# Capstone Engagement

## Assessment, Analysis, and Hardening of a Vulnerable System

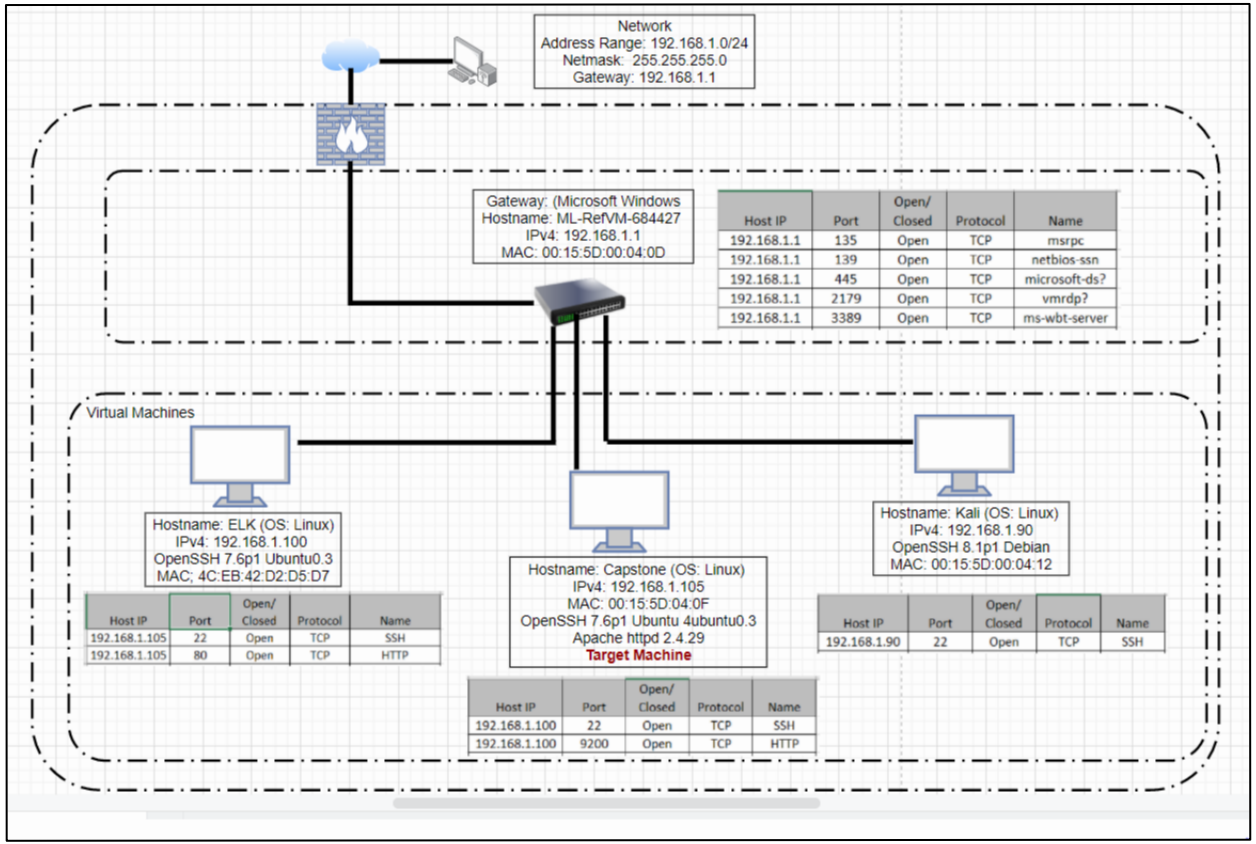
Prepared by Nicole Plybon

# Table of Contents

This document contains the following sections:

# Network Topology

# Network Topology




Network
Address Range:
192.168.1.0/24
Netmask: 255.255.255.0
Gateway: 192.168.1.1

Machines
Ipv4: 192.168.1.1
OS: Windows
Hostname: Gateway /
ML-RefVM-684427

Ipv4: 192.168.1.90
OS: Linux
Hostname: Kali

Ipv4: 192.168.1.100
OS: Linux
Hostname: ELK

Ipv4: 192.168.1.105
OS: Linux
Hostname: Capstone
Target Machine

# **Red Team**
# Security Assessment

# Recon: Describing the Target

**Nmap identified the following hosts on the network:**

| Hostname | IP Address | Role on Network |
|---|---|---|
| ML-RefVM-684427 | 192.168.1.1 | Gateway/Jump Box |
| Kali | 192.168.1.90 | Attacker's Machine |
| ELK | 192.168.1.100 | Network Security Monitor (NSM) / Log Analysis |
| Capstone | 192.168.1.105 | Webserver / Target Machine |

# Vulnerability Assessment

## The assessment uncovered the following critical vulnerabilities in the target:

| Vulnerability | Description | Impact |
|---|---|---|
| Sensitive Data Exposure | Sensitive data was accessible from the internet. Red Team was able to use the browser to read the full contents of directories on the Capstone server. | The contents of the directories revealed that Ashton is the administrator for the director: /company_folders/secret_folder |
| Security Misconfiguration / Weak Password Policy | System settings allow brute force attack of password. Passwords were easily cracked using Hydra and crackstation.net.<br><br>No lockout for failed login attempts allowing for a potential brute force attack. | Allowed the Red Team to brute force attack Ashton's password. This allowed the Red Team to gain access to the secret folder containing additional information |
| Unrestricted File Upload | Uploaded .php files represents a significant risk to applications/systems. | Allowed the Red Team to gain root access to the Capstone web server and create a backdoor to the network |

# Exploitation: Sensitive Data Exposure

01

**Tools & Processes**

Nmap was used to run a scan on IP range 192.168.1.0/24.

The scan reveled that ports 22 and 80 were open.  This allowed the Red Team to navigate to 192.168.1.105 with a web browser to gain access and naviagted to the different folders and discovered that the secret_folder existed

Nmap –sS –A 192.168.1.1/24

```
Nmap scan report for 192.168.1.105
Host is up (0.00065s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 73:42:b5:8b:1e:80:1f:15:64:b9:a2:ef:d9:22:1a:b3 (RSA)
|   256 c9:13:0c:50:f8:36:62:43:e8:44:09:9b:39:42:12:80 (ECDSA)
|_  256 b3:76:42:f5:21:42:ac:4d:16:50:e6:ac:70:e6:d2:10 (ED25519)
80/tcp open  http    Apache httpd 2.4.29
| http-ls: Volume /
|   maxfiles limit reached (10)
| SIZE  TIME              FILENAME
| -     2019-05-07 18:23  company_blog/
| 422   2019-05-07 18:23  company_blog/blog.txt
| -     2019-05-07 18:27  company_folders/
| -     2019-05-07 18:25  company_folders/company_culture/
```

# Exploitation: Sensitive Data Exposure

02

**Achievements**

The information obtained with the nmap scan and reviewing the contents of the company directories  provided further information that Ashton is the admin for the /company_folder/secret_folder/.  The seccret folder contains sensitive information which is used to execute the brute force attack and steal data.

03

**Results**

# Exploitation: Security Misconfiguration / Weak Password Policy

01

**Tools & Processes**

Executed Hydra to conduct the brute force dictionary attack using the rockyou.txt file to get Ashton's password

```
root@Kali:/usr/share/wordlists# hydra -l Ashton -P rockyou.txt -s 80 -f -V
 192.168.1.105 http-get http://192.168.1.105/company_folders/secret_folder/
```

02

**Achievements**

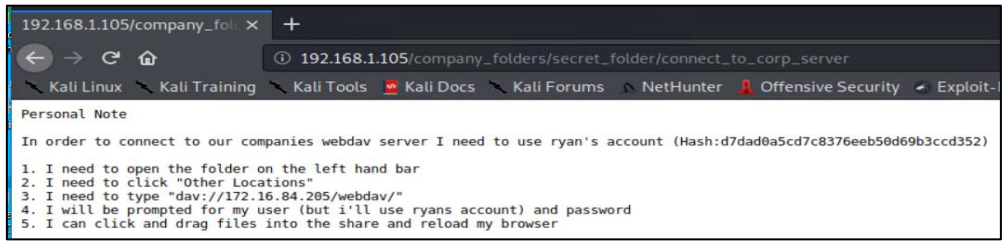
Hydra was used to brute force Ashton's password which allow us to gain access to the company's /secret_folder/.

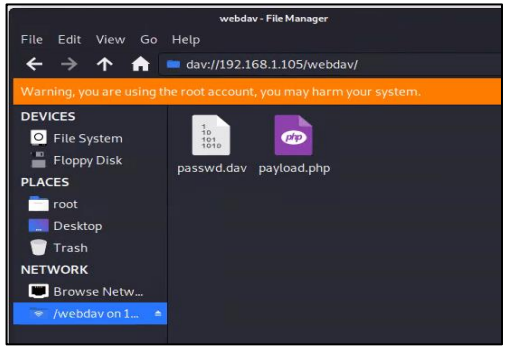
Has for Ryan's password was found and cracked allowing access to dav://192.168.1.105/webdav/

03

**Results**

```
f 14344399 [child 3] (0/0)
[80][http-get] host: 192.168.1.105   login: ashton   password: [redacted]
[STATUS] attack finished for 192.168.1.105 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2020-11-07 1
1:32:19
root@Kali:/usr/share/wordlists#
```



# Exploitation: Unrestricted File Upload

01

**Tools & Processes**

Created and uploaded msfvenom payload: php/meterpreter/reverse_tcp

Established the remote listener

Executed reverse shell backdoor on the Capstone web server

02

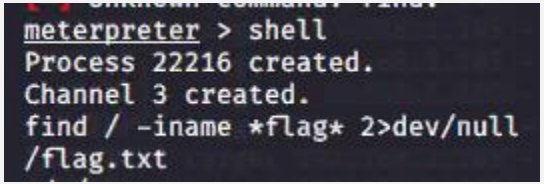
**Achievements**

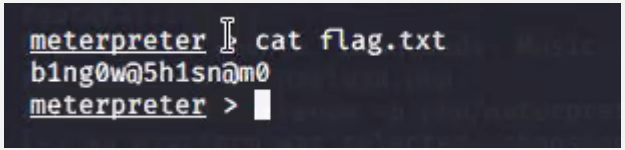
Webdav was not secure and allowed the Red Team to upload the .php file which created a reverse shell backdoor onto the Capstone 192.168.1.105 web server.

03

**Results**

Reverse shell backdoor established

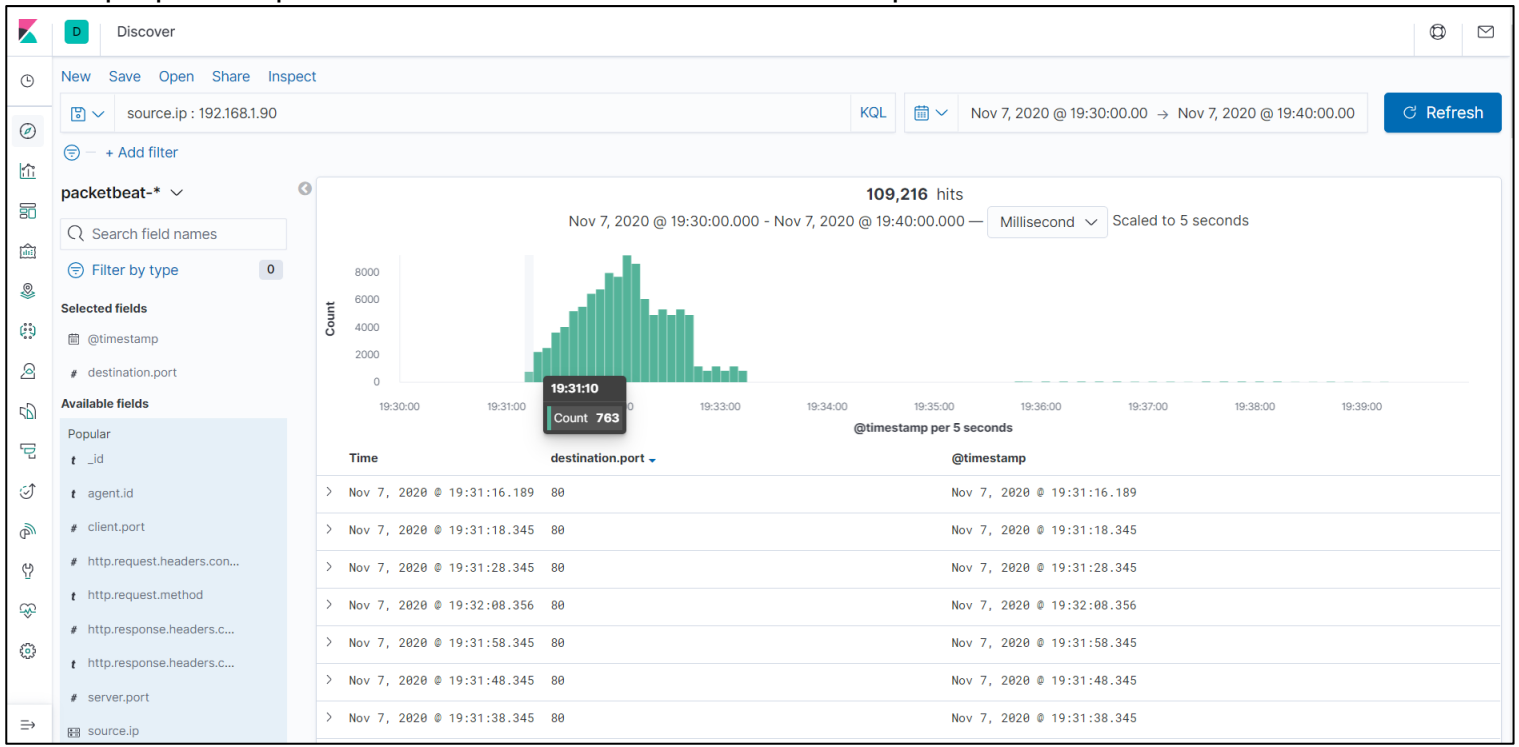


# **Blue Team**
# Log Analysis and Attack Characterization

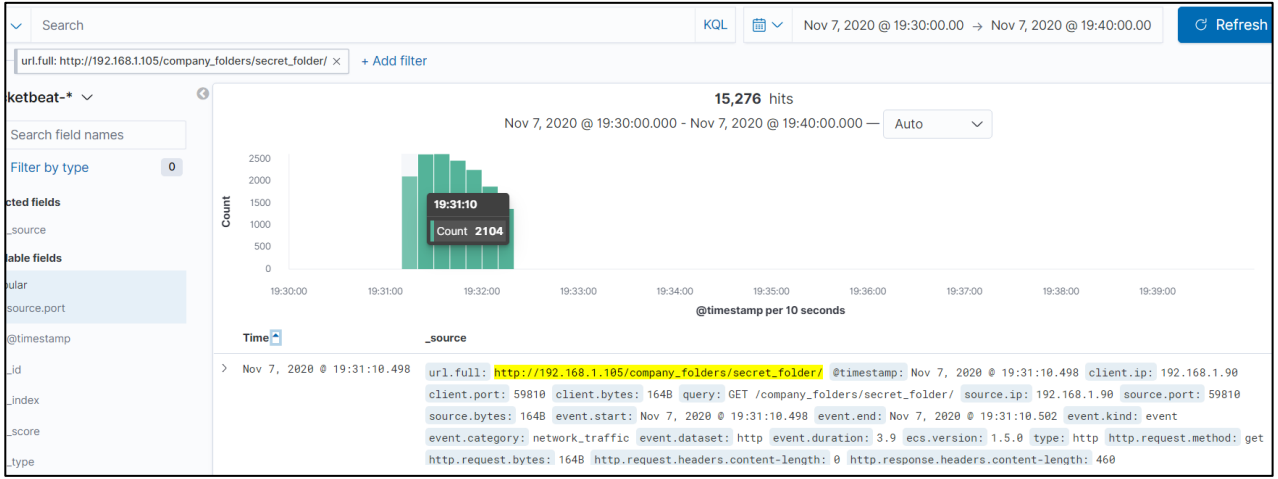
# Analysis: Identifying the Port Scan

- Port scan occurred on Nov 7th, 2020 @ 19:31:10 with 763 packet sent from 192.168.1.90
- 109, 216 packets were sent from 192.168.1.90
- Multiple ports requested at the same time are indicative of a port scan



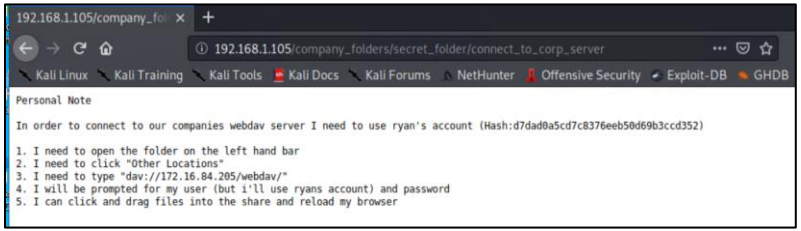
# Analysis: Finding the Request for the Hidden Directory

- The first request occurred on Nov 7th starting at 19:31:10 (7:31:10 pm)
- 15,280 requests were made to the /secret_folder, mostly durning the brute force attack



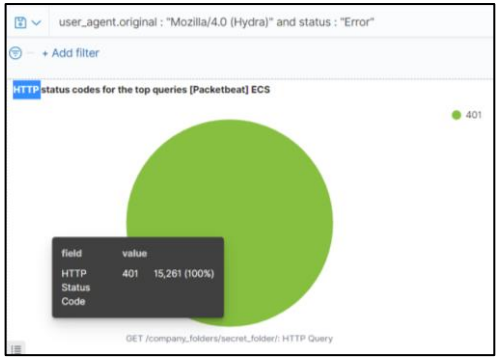

- The “http://192.168.1.105/company_folders/secret_folder/ file was attacked with 15,280 requests.  This file contains instruction for connecting to webdav and Ryan’s hashed password which was cracked using crackstation.net

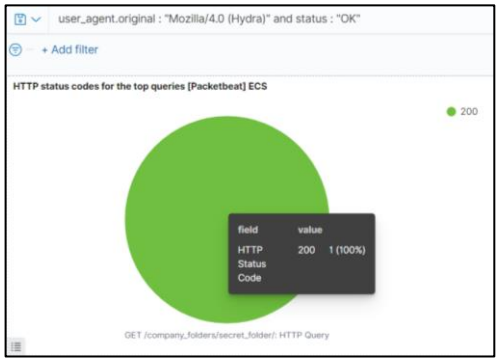
| url.full: Descending | Count |
|---|---|
| http://192.168.1.105/company_folders/secret_folder/ | 15,280 |
| http://127.0.0.1/server-status?auto= | 5,052 |

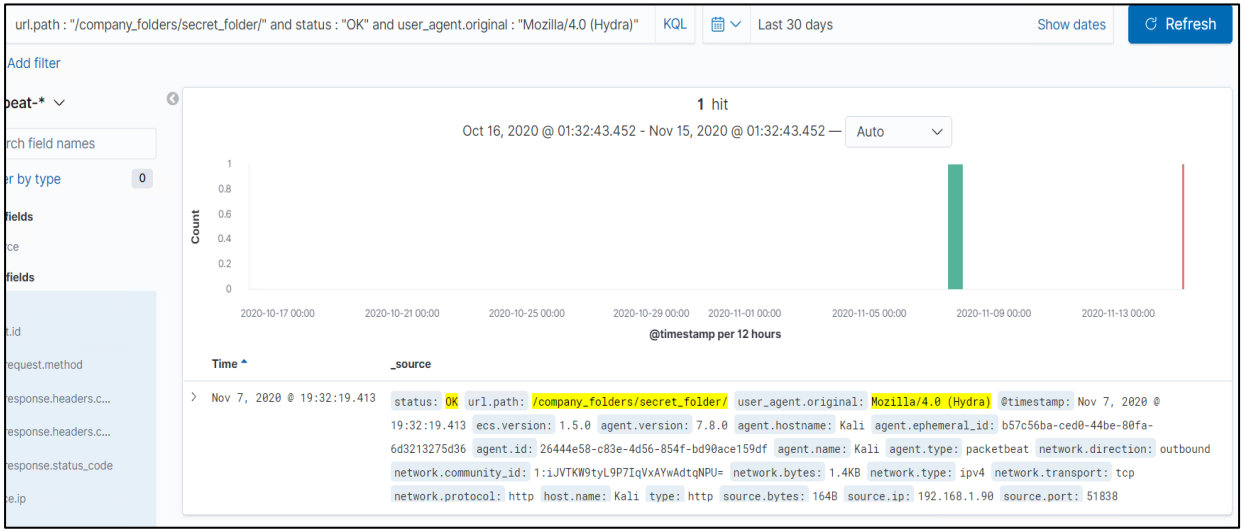


# Analysis: Uncovering the Brute Force Attack

- 15,261 requests were made in the direct brute force attack before the password was cracked
- The password was discovered on Nov 7th, 2020 at 19:32:19 (7:32:19 pm) and the attacker was able to gain access to the system and the /secret_folder/

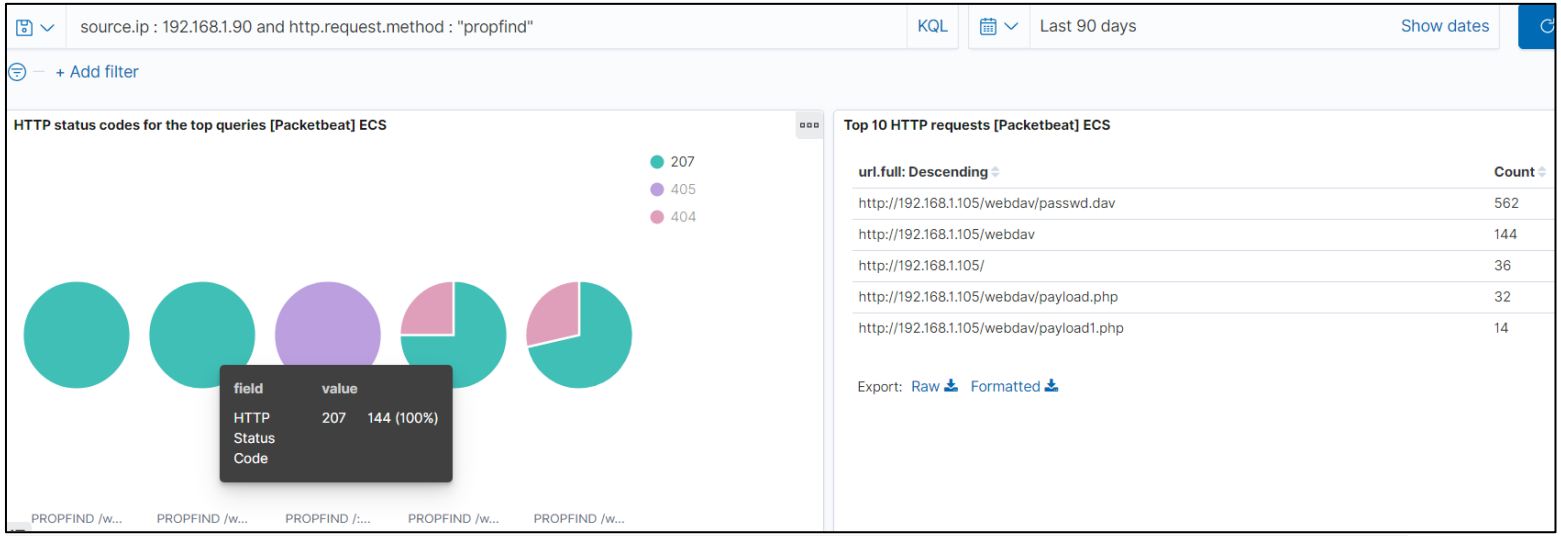


# Analysis: Finding the WebDAV Connection

- The http://192.168.1.90/webdav/ directory was requested 144 times
- The http://192.168.1.90/webdav/passwd.dav was requested 562 times
- The http://192.168.1.90/webdav/payload.php was requested 32 times
- Backdoor payload.php was uploaded on Nov 7 at 20:22:32




Nov 7, 2020 @ 20:22:32.908 url.path: /webdav/payload.php http.request.method: put @timestamp: Nov 7, 2020 @ 20:22:32.908 network.community_id: 1:pKMVEcQruM+AAZMbXtdM8MLYFS8= network.bytes: 1.9KB network.type: ipv4 network.transport: tcp network.protocol: http network.direction: outbound type: http query: PUT /webdav/payload.php server.ip: 192.168.1.105 server.port: 80 server.bytes: 537B status: OK source.bytes: 1.3KB source.ip: 192.168.1.90 source.port: 52484 method: put event.kind: event event.category: network_traffic event.dataset: http event.duration: 1.6 event.start: Nov 7, 2020 @

# **Blue Team**
# Proposed Alarms and Mitigation Strategies

# Mitigation: Blocking the Port Scan

## Alarm

**The following alarms can be set to detect future port scans.**

Alarm that detects any IP address that is scanning the network that is not a trusted IP address (Trusted IP address should be placed on the Whitelist)

Alarm that detects destination ports that are not 80 and 443 and attempted access > 100 requests per second intervals

**Alarm details/threshold:**
Alert emails and log > 100 requests per second intervals against destination ports that are not 80 and 443 and detected in the same time stamp from the same IP

## System Hardening

**What configurations can be set on the host to mitigate port scans?**
Conduct internal port scans to determine if there are more ports open than required. Check system to determine existing weak points that could be exploited.

Firewall equipped with well-audited rules, close off all unused ports (protecting ports that are exposed and their visibility), make sure that all remote users and access points are secured.

Configure a Web Application Firewall to detect and block malicious requests before they reach users applications

# Mitigation: Finding the Request for the Hidden Directory

## Alarm

**The following alarm can be set to detect future access to directories with sensitive data**

Sensitive files or directories accessed by non whitelisted IP addreses

Whitelist only the trusted IP addresses and firewall rules to deny any non-trusted IP's

**Alert details/threshold:**
Alert email and log when protected files and directories containing sensitive data are accessed by outside non-trusted IP addresses.

Alert email and log when > 1 attempt is made to access the /secret_folder/ from IP's other than on the Whitelist

## System Hardening

**What confirmation can be set on the host to block unwanted access?**

All sensitive data should be removed and never placed on public facing web servers

Managing IP's by Whitelisting and Blacklisting rules

Httpd.conf file configuration:

- Nano /etc/httpd/conf/httpd.conf (file location may vary)
- Locate directory section (/var/www/) and set as follows:

```
<Directory /var/www/>
        Order allow,deny
        Allow from 192.168.1.1
        Allow from 192.168.1.105
        Deny from all
        <LimitExcept GET POST HEAD > deny from all
        </LimitExcept>
</Directory>
```

Blacklisted IP's will not be able to access the 'secret_folder/

# Mitigation: Preventing Brute Force Attacks

## Alarm

**The following alarm can be set to detect future brute force attacks**
Any Error (401) responses detecting in > 100 requests per second intervals requests

Any OK (200) responses from a non-trusted IP address

Whitelist only the trusted IP addresses and firewall rules to deny any non-trusted IP's

**Alert details/threshold:**
Alert email and log > 5 failed login attempts within a 1 time timestamp

Alert email for any login attempts by Mozilla/4.0 (Hydra)

Alert email for any login attempts from an untrusted outside IP address attempting to log into the system

## System Hardening

Enforce a strong password policy

Limit failed login attempts

Use Captcha to ensure the user is human

Limit logins to a specified IP address or range

Two factor authentication

Unique login URLs

Monitoring logs

# Mitigation: Detecting the WebDAV Connection

## Alarm

**The following alarm can be set to detect future access to directories with sensitive data**

Monitor sensitive files or directories accessed on webdav by non whitelisted IP addreses.  Monitor with Filebeat

Whitelist only the trusted IP addresses and firewall rules to deny any non-trusted IP's

**Alert details/threshold:**
Alert email and log when protected files are accessed on the webdav directories containing sensitive data by outside non-trusted IP addresses.

Alert email and log when > 1 attempt is made to access the webdav directory from IP's other than on the Whitelist

## System Hardening

**What confirmation can be set on the host to block unwanted access?**
All sensitive data should be removed and never placed on public facing web servers

System Administors should install and configure Filebeat to monitor activity on the network

httpd.conf file configuration:

- Nano /etc/httpd/conf/httpd.conf (file location may vary)
- Locate directory section (/var/www/) and set as follows:

```
<Directory /var/www/webdav/>
        Order allow,deny
        Allow from 192.168.1.1
        Allow from 192.168.1.105
        Deny from all
        <LimitExcept GET POST HEAD > deny from all
        </LimitExcept>
</Directory>
```

# Mitigation: Identifying Reverse Shell Uploads

## Alarm

**The following alarm can be set to detect future uploads?**

Alarm when a “put” HTTP request from a non whitelisted IP indicating server files are altered

Alarm when there is a POST request that contains a file being uploaded that is not in a premitted format such as .php

**Alert details/threshold:**
Alert email and log when “put” request method is made on protected files or directories containing sensitive data are being accessed by outside non-trusted IP addresses.

Alert email and log when a forbidden file is being uploaded by a non-trusted IP address

## System Hardening

**What configuration can be set on the host to block file uploads?**
Filebeat should be enabled and configured to montior for format file types that are not permitted (i.e. .php)

Whitelisting and Blacklisting in the httpd.conf file configuration:

- Nano /etc/httpd/conf/httpd.conf (file location may vary)
- Locate directory section (/var/www/) and set as follows:

```
<Directory /var/www/>
        Order allow,deny
        Allow from 192.168.1.1
        Allow from 192.168.1.105
        Deny from all
        <LimitExcept GET POST HEAD > deny from all
        </LimitExcept>
</Directory>
```


The End