Capstone Engagement

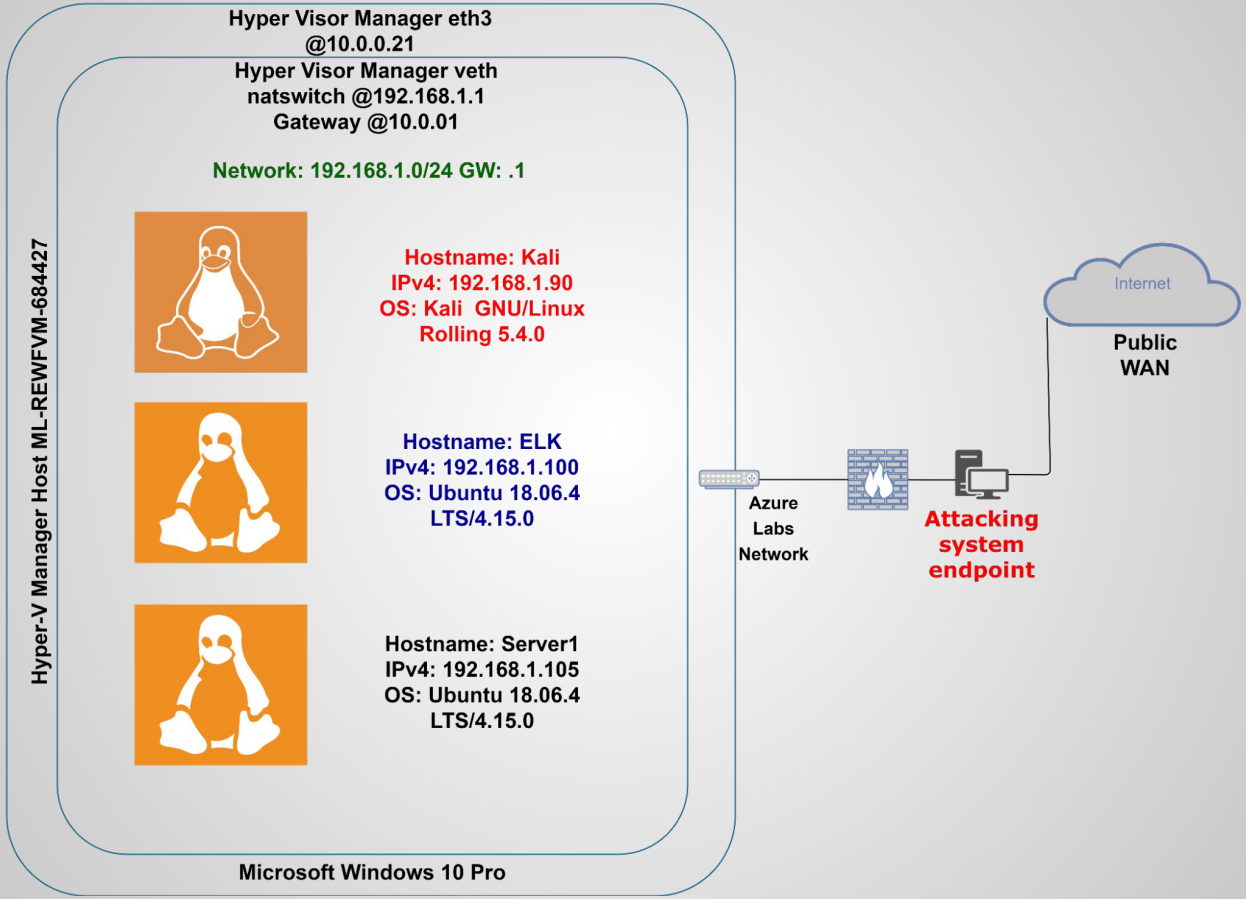
Assessment, Analysis, and Hardening of a Vulnerable System

Table of Contents

This document contains the following sections:

Network Topology

Network Topology



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	Microsoft Hyper-V Host Manager Network GW/IP Helper/NatSwitch VM storage
KALI	192.168.1.90	Pentesting VM with KALI OS attack tools, attacker
ELK	192.168.1.100	Elasticsearch, Logstash, and Kibana log server with filebeat, metricbeat and packbeat
Server1	192.168.1.105	Vulnerable server, victim

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
CWE-548: Exposure of Information Through Directory Listing	Directory information of web server content is publicly exposed.	Can provide sensitive and confidential data to attacker. Assists attacker to understand directory structure
CWE-307: Improper Restriction of Excessive Authentication Attempts	No limited number of bad password attempts and logins in place.	An attacker can leverage a brute force attack on accounts with no restrictions
LFI Vulnerability	LFI allows access into confidential files on a site.	An LFI vulnerability allows attackers to gain access to sensitive credentials. The attacker can read and/or write to these files
Users with Root Access	Access to execute all commands and access to all resources on the system.	Attacker can leverage this access to run exploits, compromise and establish persistence

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Weak Passwords	Simple passwords (no symbols, numbers, etc) can be guessed or compared to password lists for easy cracking.	The password for “ashton” which was Leopoldo was bruteforced in 40 seconds
Hashed Passwords	Non salted hash values can easily be deciphered using web tools such as www.crackstation.net	Once deciphered, the password can be used in combination of usernames to attain access to a resource
CWE-256: Unprotected Storage of Credentials	Several users stored passwords in plain text	User account “ashton” had “ryan” credentials stored in a text file which allowed the attacker access
Common and Simple Usernames	Short and common usernames can easily be guessed or found.	“ashton,” “ryan,” and “hannah”are usernames that can be guessed and used with know password lists

Exploitation: CWE-548: Exposure of Information Through Directory Listing

01

Tools & Processes

* Linux commands to get network properties of Kali attacking system.

Ifconfig: Inet 192.168.1.90 /24
Route -n: Gateway 192.168.1.1

* Nmap scan to subnet the attacking Kali system resides on.

Nmap -A 192.168.1.0/24
-Will display public files, OS fingerprinting, open ports, ssh

* Linux command to see folder directory

Wget -r -np --spider 192.168.1.105/ 2>1
| grep 192.168.1.105
| grep -v -e “Remote File”
| uniq > server1.txt

-Alternatively a web browser can be used to 192.168.1.105 to see the folder directory.

02

Achievements

* The layout of the network was found

* Hosts along with recon information such as open ports, and folder directories were exposed

*Company information such as messages and notes found in the public directories . Usernames exposed and bits of information that could assist in formulating an attack

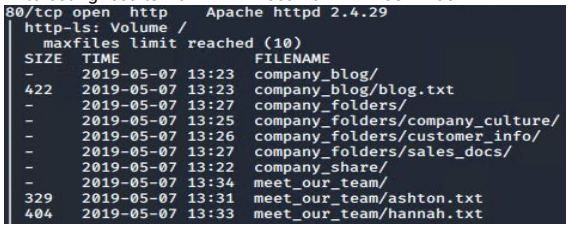
*Username “ashton” found as administrator of the “secret_folder”

03

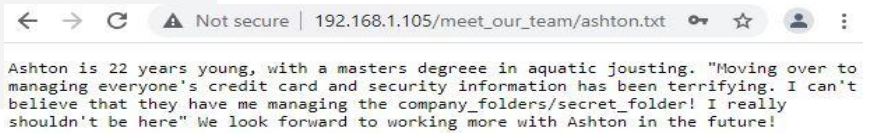
Results

Network information

Interesting results from NMAP scan on 192.168.1.105



Useful recon information



Exploitation: CWE-307: Improper Restriction of Excessive Authentication Attempts

01

Tools & Processes

*A bruteforce tool named “Hydra” from the Kali attack system was used in conjunction with the infamous rockyou.txt password list to compromise the “ashton” account as there was no lockout policies to stop the attack.

https://tools.kali.org/password-attacks/hydra

*An MD5 online decrypter was used to decipher the hashed file found in 192.168.1.105/company_fodlers/secret_folder

https://crackstation.net/

02

Achievements

*Username “ashton” was successfully bruteforce attacked within 25 seconds of launch of the tool.

*The username and password were used to gain access to: 192.168.1.105/company_folders/secret_folder

*In this folder a username “ryan” and an MD5 hash were found. The hash was put through a hash decryption tool found on https://crackstation.net/

*Instructions were found related to connecting to a server over webdav at: dav://172.16.84.205/webdav/

*Username “ryan” and unhashed password used to connect to webdav.
-This location allows for files to be added

03

Results

```
[80][http-get] host: 192.168.1.105   login: ashton   password: leopoldo
[STATUS] attack finished for 192.168.1.105 (waiting for children to complete tests)
1 of 1 target successfully completed, 1 valid password found
```

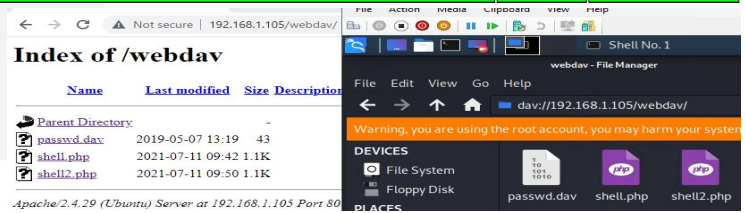
Not secure | 192.168.1.105/company_folders/secret_folde...

```
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
```

Hash	Type	Result
d7dad0a5cd7c8376eeb50d69b3ccd352	md5	linux4u



Exploitation: Reverse TCP shell - Persistent (LFI Vulnerability)

01

Tools & Processes

*Metasploit and MSFVenom were used to create a malicious payload from: php/meterpreter/reverse_tcp

*Listener port was established

*Reverse shell backdoor executed on victim “server1”

02

Achievements

*Back door shell opened back to attacker Kali system with root access to the root directory of “server1”

*flag.txt found and captured

03

Results

```
meterpreter > cd ../../../
meterpreter > ls
Listing: /
==========

Mode              Size        Type  Last modified              Name
----              ----        ----  -------------              ----
40755/rwxr-xr-x   4096        dir   2020-05-29 14:05:57 -0500  bin
40755/rwxr-xr-x   4096        dir   2020-06-28 01:13:04 -0500  boot
40755/rwxr-xr-x   3840        dir   2021-07-11 09:25:33 -0500  dev
40755/rwxr-xr-x   4096        dir   2021-07-11 07:51:06 -0500  etc
100644/rw-r--r--  16          fil   2019-05-07 14:15:12 -0500  flag.txt
40755/rwxr-xr-x   4096        dir   2020-05-19 12:04:21 -0500  home
100644/rw-r--r--  57982894    fil   2020-06-26 23:50:32 -0500  initrd.img
100644/rw-r--r--  57977666    fil   2020-06-15 14:30:25 -0500  initrd.img.old
40755/rwxr-xr-x   4096        dir   2018-07-25 18:01:38 -0500  lib
40755/rwxr-xr-x   4096        dir   2018-07-25 17:58:54 -0500  lib64
40700/rwx------   16384       dir   2019-05-07 13:10:15 -0500  lost+found
40755/rwxr-xr-x   4096        dir   2018-07-25 17:58:48 -0500  media
40755/rwxr-xr-x   4096        dir   2018-07-25 17:58:48 -0500  mnt
40755/rwxr-xr-x   4096        dir   2020-07-01 14:03:52 -0500  opt
40555/r-xr-xr-x   0           dir   2021-07-11 09:24:58 -0500  proc
40700/rwx------   4096        dir   2020-05-21 18:30:12 -0500  root
40755/rwxr-xr-x   900         dir   2021-07-11 09:26:36 -0500  run
40755/rwxr-xr-x   12288       dir   2020-05-29 14:02:57 -0500  sbin
40755/rwxr-xr-x   4096        dir   2019-05-07 13:16:00 -0500  snap
40755/rwxr-xr-x   4096        dir   2018-07-25 17:58:48 -0500  srv
100600/rw-------  2065694720  fil   2019-05-07 13:12:56 -0500  swap.img
40555/r-xr-xr-x   0           dir   2021-07-11 09:25:02 -0500  sys
41777/rwxrwxrwx   4096        dir   2021-07-11 09:25:46 -0500  tmp
40755/rwxr-xr-x   4096        dir   2018-07-25 17:58:48 -0500  usr
40755/rwxr-xr-x   4096        dir   2020-05-21 18:31:52 -0500  vagrant
40755/rwxr-xr-x   4096        dir   2019-05-07 13:16:46 -0500  var
100600/rw-------  8380064     fil   2020-06-19 06:08:40 -0500  vmlinuz
100600/rw-------  8380064     fil   2020-06-04 05:29:12 -0500  vmlinuz.old

meterpreter > cat flag.txt
b1ng0w@5h1sn@m0
meterpreter >
```

Blue Team
Log Analysis and
Attack Characterization

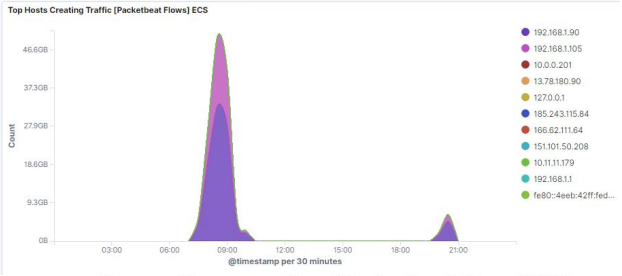
Analysis: Identifying the Port Scan

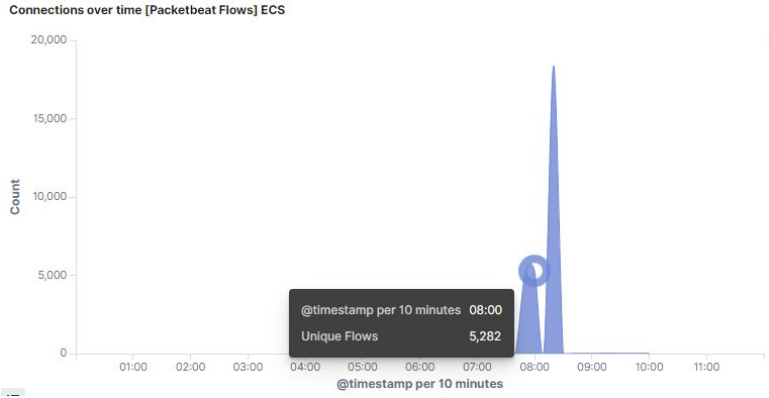
- The port scan happened around 7:40AM to 8:00AM CST on 07.11.2021
- 192.168.1.90 sent about 5000 packets indicative of a port scan.
- The port scan happened twice which I assume was done to get more information about the data with other NMAP options.

Network Traffic Between Hosts [Packetbeat Flows] ECS

Source IP	Destination IP	Source Bytes	Destination Bytes
192.168.1.90	192.168.1.100	98GB	2GB
192.168.1.90	192.168.1.105	52.6MB	93.7MB
192.168.1.90	192.168.1.1	1.1MB	108.4KB
192.168.1.90	192.168.1.90	648.2KB	615.1KB
192.168.1.90	255.255.255.255	8.5KB	0B
192.168.1.105	192.168.1.100	43.3GB	2.1GB
192.168.1.105	91.189.88.142	217KB	13.4MB
192.168.1.105	169.254.169.254	98.9KB	243KB
192.168.1.105	192.168.1.90	83.9KB	753.5KB
192.168.1.105	91.189.92.38	29.9KB	41.4KB

Export: Raw Formatted





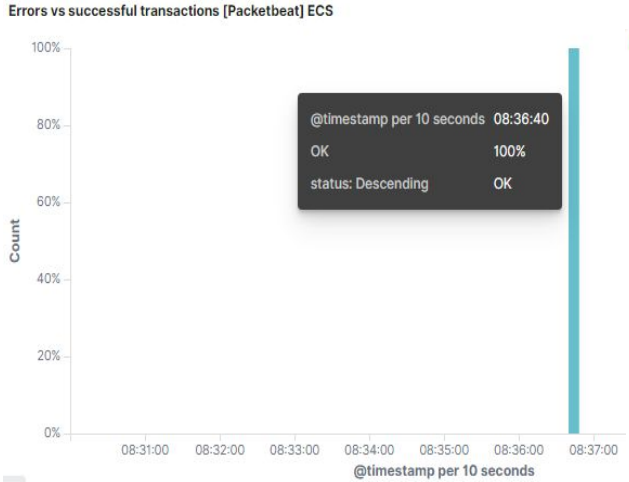
Connections over time [Packetbeat Flows] ECS

Unique Flows

@timestamp per 10 minutes	Unique Flows
07:20	6
07:30	7
07:40	1
07:50	5,047
08:00	5,282

Analysis: Finding the Request for the Hidden Directory

- A successful request for the hidden directory 192.168.1.105/company_folders/secret_folder happened at 8:36AM CST on 07.11.2021
- There were 16, 643 attempts to authenticate to this folder more than likely with the use of Hydra.
- During this same time 192.168.1.105/company_folders/secret_folder/connect_to_corporate_server was accessed



Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	Count
http://192.168.1.105/company_folders/secret_folder	16,473

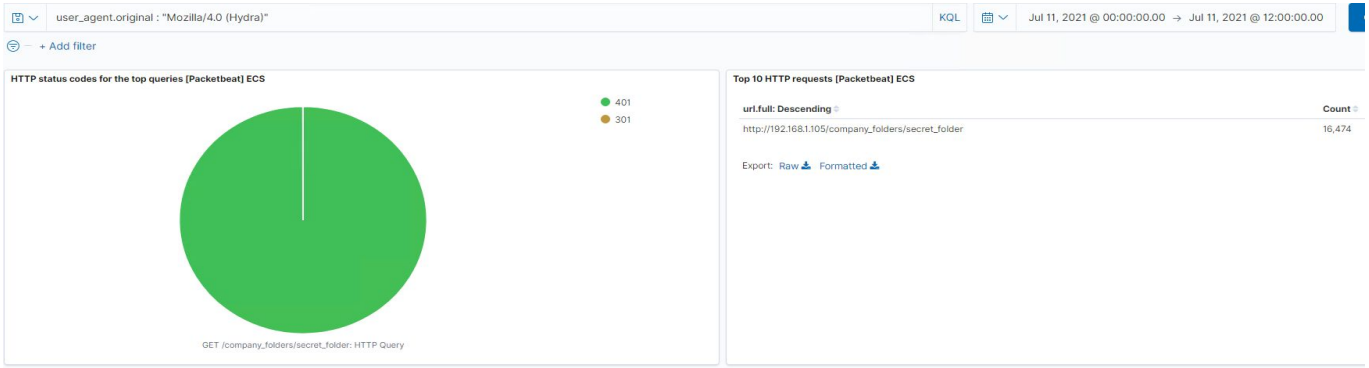
Export: Raw Formatted

Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	Count
http://192.168.1.105/company_folders/secret_folder/connect_to_corp_server	1

Analysis: Uncovering the Brute Force Attack

- It took 16,474 attempts prefer Hyrdra was able to bruteforce the “secret_folder”
- 16,643 attempts were completed in all by Hydra



	server.ip	192.168.1.105
#	server.port	80
#	source.bytes	163B
	source.ip	192.168.1.90
#	source.port	39580
t	status	OK
t	type	http
t	url.domain	192.168.1.105
t	url.full	http://192.168.1.105/company_folders/secret_folder
t	url.path	/company_folders/secret_folder
t	url.scheme	http
t	user_agent.original	Mozilla/4.0 (Hydra)

Analysis: Finding the WebDAV Connection

- There were 50 requests to 192.168.1.105/webdav during the time of the compromise
- Files requested were: 192.168.1.105/webdav/shell.php
  192.168.1.105/webdav/shell2.php
  192.168.1.105/webdav/passwd.dav

url.full: Descending	Count
http://192.168.1.105/webdav	50
http://192.168.1.105/webdav/shell.php	18
http://192.168.1.105/webdav/shell2.php	13
http://192.168.1.105/webdav/passwd.dav	5

This is when shell2.php was uploaded to 192.168.1.105/webdav

Jul 11, 2021 @ 09:50:35.165

http.request.method: put url.path: /webdav/shell2.php @timestamp: Jul 11, 2021 @ 09:50:35.165 type: http destination.ip: 192.168.1.105 destination.port: 80
destination.bytes: 535B host.name: server1 user_agent.original: gvfs/1.42.2 status: OK method: put http.request.headers.content-length: 1,113 http.request.bytes: 1.3KB
http.request.body.bytes: 1.1KB http.response.headers.content-length: 271 http.response.headers.content-type: text/html; charset=ISO-8859-1 http.response.status_phrase: created
http.response.status_code: 201 http.response.bytes: 535B http.response.body.bytes: 271B http.version: 1.1 event.kind: event event.category: network_traffic
event.dataset: http event.duration: 0.7 event.start: Jul 11, 2021 @ 09:50:35.165 event.end: Jul 11, 2021 @ 09:50:35.166 ecs.version: 1.5.0

:(

Blue Team Proposed Alarms and Mitigation Strategies

Mitigation: Blocking the Port Scan

Alarm

Below is basic logic to get a port scan scan to alert when detected.

Log and alert when 3 or more port scan requests to common or known ports come from the same IP within 1 minute

A daily report should be setup to retrieve a baseline of this incident and change the number of port scan requests if needed

Trend data should be reviewed daily

System Hardening

Security team should conduct their own port scans on their network to see the layout of the land and vulnerabilities such as exposed ports and network information

Security team should set up logging from edge devices and ingest the logs into a SIEM that can raise incidents and alerts and has reporting capabilities. Logging from critical servers should also be ingested into the SIEM to correlate data in case of compromise

IPtables on the Linux servers can be adjusted to slow down the effectiveness of a port scan. Below is a list of recommended commands to mitigate a port scan

```
iptables -A port-scan -p tcp -tcp-flags SYN,ACK,FIN,RST RST -m limit -limit 1/s -j RETURN
iptables -A port-scan -j DROP -log-level 6
iptables -A specific-rule-set -p tcp -syn -j syn-flood
iptables -A specific-rule-set -p tcp -tcp-flags SYN,ACK,FIN,RST RST -j port-scan
```

Mitigation: Finding the Request for the Hidden Directory

Alarm

Below is basic logic to receive an alert when untrusted IPs have been made to confidential data.

Log and alert when 1 or more access attempts have been made to “secret_folder” from any IP other than 192.168.1.105

A daily report should be setup to retrieve a baseline of this incident.

Trend data should be reviewed daily

System Hardening

Directory listing should be disabled by removing the “Indexes” option from the apache configuration file. The file can be found here:

Debian and Ubuntu distributions refer to Apache as “Apache2” and the configuration file of Apache2 is /etc/apache2/apache2.conf.

CentOS refer to Apache as httpd and configuration file of httpd is /etc/httpd/httpd.conf.

```
<Directory /var/www/>
        Options Indexes FollowSymLinks
        AllowOverride None
        Require all granted
</Directory>
```

In the same section allow/deny rules can be set for for allow and deny access

```
/var/www/company_folders/secret_folder/
Order allow,deny
Allow from 192.168.1.1
Allow from 192.168.1.105
Allow from 127.0.0.1
Deny from 192.168.1.90
```

Mitigation: Preventing Brute Force Attacks

Alarm

Below is basic logic to get a an alert to execute when attempted or credentialed access is allowed to protected data.

Log and alert when in network protected directories and subdirectories have more than 5 error 401 responses occurring at any time or any OK successful 200 responses from non-trusted IPs

A daily report should be setup to retrieve a baseline of this incident.

Trend data should be reviewed daily

System Hardening

A strong password policy that enforces password change, password history, multi-factor authentication, and complexity should be implemented

A lock out policy should be enforced throughout the network

A sensitive data discovery tool should be deployed by the security team and an assessment made about what other data is viewable and what data should be protected

An identity security solution should be implemented into the network as the network uses different operating systems and privileged accounts with different access

Root and administrator accounts should be disabled or at least change the names and complicate the passwords. A review of accounts that have root/admin rights should be completed

Mitigation: Detecting the WebDAV Connection

Alarm

Below is basic logic to get a an alert to be raised when connections from untrusted IPs are attempting to or get access to the WebDAV connection

Log and alert when an un/successful access attempt has been made to 192.168.1.105/webdav and 192.168.1.105/webdav/*

A daily report should be setup to retrieve a baseline of this incident.

Trend data should be reviewed daily

System Hardening

Allow and deny IPv4 addresses to directory

Locate the webdav directory in /var/www/ in either:

/etc/apache2/apache2.conf.

/etc/httpd/httpd.conf.

```
/var/www/webdav/
Order allow,deny
Allow from 192.168.1.1
Allow from 192.168.1.105
Allow from 127.0.0.1
Deny from 192.168.1.90
```

Mitigation: Identifying Reverse Shell Uploads

Alarm

Below is basic logic to get a an alert to be raised when connections from untrusted IPs are attempting to upload a file to directories

Log and alert when “put” requests are made on protected folders from non-trusted IPs

System Hardening

Security team should consider reviewing logs for WebDAV commands as below and raising alerts when they occur

-PROPFIND: Used to retrieve properties, persisted as XML, from a resource. It is also overloaded to allow one to retrieve the collection structure (a.k.a. directory hierarchy) of a remote system.

-PROPPATCH: Used to change and delete multiple properties on a resource in a single atomic act.

-MKCOL: Used to create collections (a.k.a. directory).

-COPY: Used to copy a resource from one URI to another

.

-MOVE: Used to move a resource from one URI to another.

-LOCK: Used to put a lock on a resource. WebDAV supports both shared and exclusive locks.

-UNLOCK: To remove a lock from a resource.

Furthermore, adding a line to the deny/allow list for GET POST HEAD provides another layer of defense.

```
/var/www/webdav/
Order allow,deny
Allow from 192.168.1.1
Allow from 192.168.1.105
Allow from 127.0.0.1
Deny from 192.168.1.90
Deny from all
<LimitExcept GET POST HEAD>deny from all
</LimitExcept>
```

THANK YOU FOR PLAYING!

Everything not saved will be lost.