CYBER FORENSICS (CYFO) VT 24
Lab Assignment

Fernando M. Sanfeliz
Dept. of Computer and Systems Sci.
Stockholm University
Stockholm, Sweden
fesa6676@student.su.se

Gustav Mönefors
Dept. of Computer and Systems Sci.
Stockholm University
Stockholm, Sweden
gumo9296@student.su.se

Silas Pohl
Dept. of Computer and Systems Sci.
Stockholm University
Stockholm, Sweden
sipo6151@student.su.se

April 30, 2024

Assignment 1: Introduction to Network Forensics

A virtual machine running Kali 2024.1 and Wireshark (wireshark:amd64/kali-rolling 4.2.2-1) was used to analyse the .pcap files of assignment 1.

Evidence file:	`1.pcap`
MD5 Hash:	`47451679a42fc2a5a637886e97fd7283`
SHA-1 Hash:	`4623636b88b6293888a3ebcb75cffb767bd11094`

Verifying the hashes

```
$ md5sum 1.pcap
47451679a42fc2a5a637886e97fd7283  1.pcap
$ sha1sum 1.pcap
4623636b88b6293888a3ebcb75cffb767bd11094  1.pcap
```

1. What is/are the source(s) (IP address) of the suspicious traffic?

`192.0.2.245, 192.0.2.196, 192.0.2.207, 192.0.2.6, 192.0.2.25, 192.0.2.120, 192.0.2.83, 192.0.2.154, 192.0.2.253, 192.0.2.236`

2. What is the destination (IP address) of the suspicious traffic?

`192.0.2.2`

3. What is the transport layer protocol used?

Transmission Control Protocol (TCP)

4. What is/are the source port(s)?

`35356, 44463, 23784, 51136, 57003, 20920, 36927, 52048, 62151, 46528`

5. What is/are the destination port(s)?

`64354, 58034, 25895, 62694, 48897, 46680, 35104, 43120, 17166, 19043`

6. What conclusions can you draw from the type of the "attack"/activity illustrated by this pcap?

The network activity observed appears to involve attempts to establish connections with various ports on the destination address 192.0.2.2. This behavior could indicate a port scan, where the scanning party seeks to identify open ports on the target system. While not inherently malicious, port scanning can be a precursor to unauthorized access attempts or reconnaissance activities. Alternatively, the repeated connection attempts to the same destination address could suggest a potential Distributed Denial of Service (DDoS) attack. However, the number of connections initiated (10 in this case) is unlikely to overwhelm a server and cause a significant disruption in service.

Evidence file:	`2.pcap`
MD5 Hash:	`19633e3a2a3d4c315994fddc3ce7090f`
SHA-1 Hash:	`f9d5be156ca124b46450910d2b7b1e79f2f6825c`

Verifying the hashes

```
$ md5sum 2.pcap
19633e3a2a3d4c315994fddc3ce7090f  2.pcap
$ sha1sum 2.pcap
f9d5be156ca124b46450910d2b7b1e79f2f6825c  2.pcap
```

1. What is the source(s) (MAC address) of the suspicious traffic?

`CIMSYS_33:44:55 (00:11:22:33:44:55)`

2. What is/are the destination (MAC address[es]) of where the suspicious traffic is mostly directed towards?

- `Intel_83:13:e8 (00:0e:0c:83:13:e8)` received 3279 packets
- `Broadcast (ff:ff:ff:ff:ff:ff)` received 541 packets
- all other destinations received ≤ 20 packets

3. What is the link layer protocol used?

Address Resolution Protocol (ARP)

4. What is the purpose of this protocol?

The purpose of ARP is to map an IP address to a MAC address on a local network. This mapping is necessary because devices communicate using MAC addresses at the data link layer (Layer 2) of the OSI model, while IP addresses are used at the network layer (Layer 3). When a device on a LAN wants to communicate with another device, it needs to know the MAC address of the target device. ARP helps in this process by broadcasting an ARP request message to all devices on the network, asking for the MAC address corresponding to a specific IP address. The device with the matching IP address responds with its MAC address, allowing the requesting device to establish a direct communication link.[1]

5. What conclusions can you draw from the type of the attack illustrated by this pcap? How can this attack be used for launching other kinds of attacks?

The source MAC address (`CIMSYS_33:44:55`) is consistently sending the suspicious traffic. This suggests that the traffic is likely originating from a single device or attacker. The suspicious traffic is directed towards various MAC addresses, including broadcasting and `Intel_83:13:e8`, as well as other less frequent destinations, such as `ASUSTekCOMPU_1b:32:11`, `AniCommunica_91:12:a8`, etc. This pattern is indicative of ARP poisoning attacks. 516 packets are configured with "Duplicate IP addresses." ARP spoofing involves sending falsified ARP messages over a local area network. Attackers may send ARP replies with their own MAC address in response to ARP requests, causing

network devices to update their ARP caches with incorrect mappings. This allows attackers to intercept, modify, or redirect network traffic. ARP spoofing can be used as a stepping stone for launching other types of attacks, including:

- Man-in-the-Middle (MITM) Attacks: By intercepting and redirecting network traffic, attackers can eavesdrop on communications, steal sensitive information, or modify data packets.
- Denial of Service (DoS) Attacks: ARP spoofing can be used to flood a network with falsified ARP messages, leading to network congestion or disruption of communication between legitimate devices.
- Session Hijacking: By intercepting and manipulating ARP traffic, attackers can hijack established network sessions, gaining unauthorized access to sensitive systems or services.[2]

Evidence file:	`3.pcap`
MD5 Hash:	`0944977919541d4ee176450b7ce36f9d`
SHA-1 Hash:	`7349e1fea8e6ed6b4dce3f89898b1c6492f3a610`

Verifying the hashes

```
$ md5sum 3.pcap
0944977919541d4ee176450b7ce36f9d  3.pcap
$ sha1sum 3.pcap
7349e1fea8e6ed6b4dce3f89898b1c6492f3a610  3.pcap
```

1. What is the source (IP address) of the suspicious traffic?

`10.0.23.109`

2. What is the destination (IP address) of the suspicious traffic?

`80.237.98.132`

3. What is the transport layer protocol used?

Transmission Control Protocol (TCP)

4. This may be considered as not a direct attack but as a preparation step before an attack. Name the technique used and its purpose.

Mostly SYN connections (the first part of the TCP three-way handshake) are observed. TCP Retransmissions occur when the sender fails to receive an acknowledgment (ACK) from the receiver within a certain time period. This behavior indicates a SYN port scanning activity (stealth scan). The purpose of port scanning is to find out which of the ports of the target machine are open to investigate/attack them further in the next step.

Assignment 2: Suspicious Wireless Traffic

1 Introduction

In the scenario presented, we are confronted with a reported incident of unauthorized financial transactions from a user's bank account within our network. This raises concerns about the security posture of our network infrastructure and the potential presence of malicious activities.

To address these concerns, we have conducted a forensic investigation utilizing packet capture data obtained from traffic sensors strategically placed to capture network traffic traversing critical network segments. Through detailed analysis of the captured network traffic, our goal is to uncover any suspicious activities, identify potential security breaches, and gain insights into the events leading up to the reported incident.

2 Methods

Our investigation into the suspicious wireless traffic began with a thorough confirmation of the integrity of the provided evidence files: `A.pcap`, `B.pcap` and `C.pcap`. We utilized their respective SHA1 and MD5 hash values to ensure that the files had not been altered or corrupted since their acquisition.

Method and Tools for Analysing the Network Traffic

Kali Linux (VM)	`2024.1`
Wireshark	`wireshark:amd64/kali-rolling 4.2.2-1`
aircrack-ng	`aircrack-ng:amd64/kali-rolling 1:1.7-5`
WhatIsMyIPAddress	`https://whatismyipaddress.com/` (last accessed on April 28, 2024)
Wayback-Machine	`https://web.archive.org/` (last accessed on April 28, 2024)

Each evidence file (`A.pcap`, `B.pcap` and `C.pcap`) was analyzed using Wireshark 4.2.2, a powerful network protocol analyzer, running on a Kali 2024.1 virtual machine (VM) to extract relevant information regarding network traffic patterns, communication protocols, and potential anomalies.

For `A.pcap`, we focused on identifying WLAN traffic information, including the BSSID, SSID, and encryption details of the network with the most traffic. This involved examining beacon frames, probe requests, and association requests to determine network characteristics. We attempted to extract any WEP keys present in the captured WLAN traffic using the aircrack-ng tool in the Kali Linux virtual machine. We then aimed to use the extracted key to decrypt encrypted traffic and analyze its contents. We identified IP communications with the highest traffic volume within A.pcap, focusing on external IP addresses involved in the communication.

In `B.pcap`, we sought to identify the IP and MAC addresses of the firewall and the access point by analyzing network traffic. This involved examining ARP requests and responses, DHCP transactions, and other network protocols to determine the identities of these devices. We scrutinized

B.pcap for any indications of malicious activity or security breaches. This included analyzing network traffic patterns, identifying anomalous behavior, and recognizing common attack signatures associated with various cyber threats.

For C.pcap, we focused on analyzing web traffic to determine which website the client attempted to visit. Additionally, we identified the system that responded to the client's request and attempted to reconstruct the requested web page for comparison with the original. We constructed a chronological timeline of events based on the captured network traffic in C.pcap. This timeline provided a sequential overview of the interactions between network entities, aiding in understanding the sequence of events leading up to the observed network activity.

3 Results

In this analysis, suspicious wireless traffic in a small business environment was examined due to concerns raised by a user regarding unauthorized withdrawals from their bank account. The network administrator provided three files (A.pcap, B.pcap and C.pcap) for analysis. Each file was examined for relevant WLAN traffic information and potential indications of attacks.

```
$ sha1sum A.pcap
8d5fa66a0a32c3dc6769205a26a922cd5e4ef0e6  A.pcap
$ md5sum A.pcap
16dd44ba4d8842a5ffde82ba743f5c9c  A.pcap
```

The verification of the SHA1 and MD5 hash values of the evidence file A.pcap has shown that the file had not been altered or corrupted since its acquisition because the hash values match the values specified in the instructions. For File A.pcap, the BSSID and SSID of the network with the most traffic were identified as DLink_48:b0:f9 and "DSI_DSV" respectively. No specific details about encryption were found in probe packets, but WEP parameters were observed in data packets, suggesting the possibility of WEP encryption. Using the "aircrack-ng" tool in Kali VM, a WEP key "44:53:49:4C:41" was extracted, enabling decryption of all traffic for SSID "DSI_DSV". IP communications with the most traffic involved private IP addresses 192.168.1.201 and external IP addresses 173.199.116.22 (The Constant Company LLC, US) and 74.125.15.84 (Google).

private IP address	**public IP address**	**A>B (packets)**	**B>A (packets)**
192.168.1.201:27291	173.199.116.22:80	25,089 (2 MB)	52,799 (71 MB)
192.168.1.201:27271	74.125.15.84:80	6,980 (572 kB)	16,636 (22 MB)

```
$ sha1sum B.pcap
9689f1283ea8e31b6d8a99eec957d9e4fc9deb67  B.pcap
$ md5sum B.pcap
66ba2bb4b2cd73144ea1066d3bd2de8b  B.pcap
```

The verification of the SHA1 and MD5 hash values of the evidence file B.pcap has shown that the file had not been altered or corrupted since its acquisition because the hash values match the values specified in the instructions. For File B.pcap, the IP and MAC addresses of the firewall were identified as 192.168.1.201 and 00:1f:3c:6e:49:24 respectively, while the access point had the IP

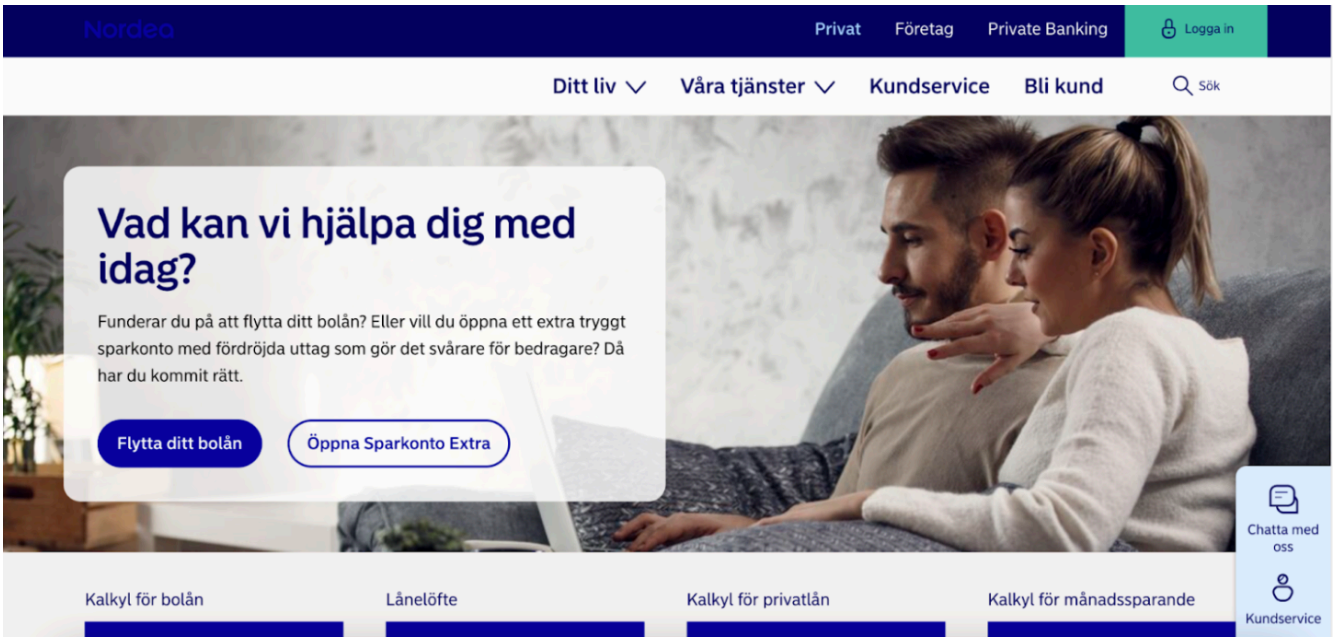
address 192.168.1.200 and MAC address 08:00:27:d2:f8:61. The investigation of the traffic indicates two types of attacks:

1. Distributed Denial of Service (DDoS) Attack - SYN Flood: The high volume anomalies identified by the continuous sending of SYN packets to specific hosts suggest SYN flood attack. It is a type of DDoS attack in which a large volume of SYN requests are sent to a target server in hopes of overwhelming it. In doing so, the attack ideally consumes enough server resources which causes the server to become unresponsive to legitimate traffic.

2. The repeated attempts to connect to various ports on the same targets using SYN packets, without further progression of TCP handshake (no ACKs following up), suggest port scanning activity.This reconnaissance tactic is commonly used by attackers to identify open ports and services available on a host which can potentially be exploited later.

```
$ sha1sum C.pcap
34a9a2422c6eca664a172c191a4c9c74af9fcef1 C.pcap
$ md5sum C.pcap
9f89be02cf88ef530df2d2930df1553d  C.pcap
```

The verification of the SHA1 and MD5 hash values of the evidence file `C.pcap` has shown that the file had not been altered or corrupted since its acquisition because the hash values match the values specified in the instructions. For File `C.pcap`, the client attempted to visit various websites (`fxfeeds.mozilla.com`, `www.stopbadware.org`, `www.nordea.se`, `statse.webtrendslive.com`, `internetbanken.privat.nordea.se`, `eplusgiro.plusgirot.se`, `internetbanken.privat.nordea.se`, `eplusgiro.plusgirot.se`, `kontoutdrag.plusgirot.se`, `solo.nordea.com`, `streaming1.nordea.com`, `gfs.nb.se`, `girovision.plusgirot.se`, `eredovisning.plusgirot.se`, `newsroom.nordea.com`, `koncernvalutakonto.plusgirot.se`, `www.checkinmusic.se`). `nordea.se` replied back.



Screenshot 1: Website nordea.se

It's worth noting that it's IP address seen is the same internal one in the server, aka 192.168.1.201, indicating interaction with a cloned version of the website. Reconstruction of the page revealed

differences from the original, indicating potential tampering. Screenshot 1 shows the original nordea webpage. On our reconstructed webpage, details wer missing, like pictures, graphics, etc. Out of the network traffic we were able to reconstruct the following timeline of events:

1. User opened internetbanken.privat.nordea.se
2. Request to get the IP Address of the DNS server was executed
3. DNS server responded with a IP address corresponding to the record of internetbanken.privat.nordea.se (probably tampered DNS record)
4. User accessed the IP address and arrived at a cloned version of the nordea website

4 Discussion

The analysis of the provided packet capture files (A.pcap, B.pcap, and C.pcap) revealed several noteworthy findings regarding suspicious wireless traffic in the small business environment, particularly concerning potential security threats and unauthorized activities.

In File `A.pcap`, the examination of WLAN traffic identified the network with the most activity. Although no specific encryption details were found in probe packets, the presence of WEP parameters in data packets suggested the possibility of WEP encryption. Through the use of the "aircrack-ng" tool, a WEP key "44:53:49:4C:41" was extracted, enabling decryption of all traffic for the SSID "DSI DSV." The analysis also highlighted IP communications involving private IP addresses 192.168.1.201 and external IP addresses 173.199.116.22 (The Constant Company LLC, US) and 74.125.15.84 (Google).

Moving on to File `B.pcap`, volumetric anomalies indicated a SYN flood attack, a form of Distributed Denial of Service (DDoS) attack. Additionally, repeated attempts to connect to various ports on the same targets using SYN packets indicated port scanning activity, a common reconnaissance tactic used by attackers.

Moving to File `C.pcap`, the client attempted to visit various websites, including "nordea.se," which responded back. Notably, the IP address seen was the same internal one on the server (192.168.1.201), indicating interaction with a cloned version of the website. Reconstruction of the webpage revealed differences from the original, indicating potential tampering. The timeline of events reconstructed from network traffic suggested DNS tampering, leading the user to access a cloned version of the Nordea website and therefore the potential leak of the banking information of the user.

In conclusion, based on the findings, it is evident that the small business network was subjected to various security threats and unauthorized activities. The presence of WEP encryption and indications of SYN flood attacks and port scanning activity suggest deliberate attempts to compromise the network's security. Furthermore, the reconstruction of the cloned Nordea website and DNS tampering point towards potential phishing or spoofing attempts aimed at obtaining sensitive information from users.

Assignment 3: Intrusion Analysis

1 Introduction

On April 28, 2023, the IT department at DSV received a concerning notification from the national computer emergency response team (CERT-SE) regarding potential unauthorized activity within the network infrastructure. The alert indicated unusual activity targeting one or more computers, with specific reference to the development test server of the Haisy student management and grade database system.

```
Date: Tues, 28 Apr 2023 11:29:03 +0100
From: CERT-SE <alert@cert.se>
To: CS2Lab <cs2lab@dsv.su.se>
Subject: Important Information from CERT-SE: Indications of Intrusion Attempt

CERT-SE has noticed unusual activity against one or more computers
in your network. See details below.

IP: 193.10.9.5
Computer name: haisy.cs2lab.dsv.su.se
Attack type: Possible intrusion attempt
Time, from circa: 2023-04-27 23:50:25 CET

Sincerely
CERT-SE
```

As part of the responsibility to maintain the security and integrity of the systems, an investigation to ascertain the nature and extent of the incident was initiated. The Haisy development test server, hosted at haisy.cs2lab.dsv.su.se, is a critical component of the infrastructure, supporting the ongoing development and testing of the Haisy system, which manages student data and academic records. Any compromise to this system could have significant ramifications for the confidentiality, integrity, and availability of sensitive information.

This report documents our investigation into the potential network intrusion, with the objective of understanding the events leading up to the alert from CERT-SE, identifying any unauthorized access or malicious activity, and assessing the impact on our systems and data. By conducting a thorough analysis of the available evidence and applying appropriate forensic techniques, we aim to provide actionable insights to mitigate risks, enhance security measures, and prevent future incidents of this nature.

2 Methods

Our investigation into the possible network intrusion targeting the Haisy development test server began with a thorough confirmation of the integrity of the provided evidence files: `haisy_4000.pcap` and `haisy.raw`. We utilized their respective SHA1 and MD5 hash values to ensure that the files had not been altered or corrupted since their acquisition.

Method and Tools for Analysing the Network Traffic

Kali Linux (VM)	`2024.1`
Wireshark	`wireshark:amd64/kali-rolling 4.2.2-1`
Geo Data Tool	`https://www.geodatatool.com/en/` (last accessed on April 28, 2024)

We proceeded to analyze the `haisy_4000.pcap` file using a Kali 2024.1 virtual machine (VM) equipped with Wireshark 4.2.2. Our primary goal was to gain insight into the network traffic associated with the suspected intrusion. We began our analysis by identifying the number of IP addresses involved in the traffic and examining the source and destination IP addresses. We also attempted to determine which ports were experiencing the highest volume of traffic, which would provide insight into the nature of the communication. We also used the geolocation tool `https://www.geodatatool.com/en/` to determine the geographic location of the identified IP addresses. This contextual information helped us understand the potential origin or source of the suspicious network activity. Throughout our analysis, we meticulously examined DNS and HTTP packets, as well as TCP streams, looking for any indicators of compromise or anomalous behavior that might indicate an attempted network intrusion.

Method and Tools for Analysing the Linux Server Image

Windows 10 (VM)	`10.0.19045 (Build 19045)`
Autopsy	`4.21.0`

We then turned our attention to examining the `haisy.raw` file in a Windows 10 environment running Autopsy 4.21.0. The goal of this phase of the investigation was to delve deeper into the system-level activity during the incident. Our analysis covered several key areas, including user account management, command execution, and software usage. We began by identifying existing users on the system and examining their account activity for signs of unauthorized access or suspicious behavior. At the same time, we analyzed the commands executed during the relevant time period to identify any anomalies or indicators of malicious activity. In addition, we evaluated the services running on the system at the time of the incident to assess their relevance to the investigation and potential impact on system security. To supplement our analysis, we examined specific log files associated with the running services, focusing on events and activities that could provide evidence of a potential attack or compromise. By meticulously reviewing the system logs, we aimed to uncover any traces of unauthorized access, privilege escalation, or suspicious activity indicative of a network intrusion. Finally, we synthesized the results to construct a comprehensive timeline of events related to the suspected intrusion. This timeline provided a chronological sequence of activities leading up to and during the incident, allowing us to gain a clear understanding of the nature and scope of the attempted intrusion.

3 Results

3.1 Investigating the Network Traffic

```
$ sha1sum haisy_4000.pcap
5d50246cd8ed94b9d39d60b4008a2ead1e3cba50  haisy_4000.pcap

$ md5sum haisy_4000.pcap
8f7f17adf4de26e88dd2841dca174b02  haisy_4000.pcap

$ ls -l haisy_4000.pcap
-rw-r--r-- 1 kali kali 35116793 Apr 23 08:20 haisy_4000.pcap
```

The verification of the SHA1 and MD5 hash values of the evidence file `haisy_4000.pcap` has shown that the file had not been altered or corrupted since its acquisition because the hash values match the values specified in the instructions. The file has a size of 35.1168MB.

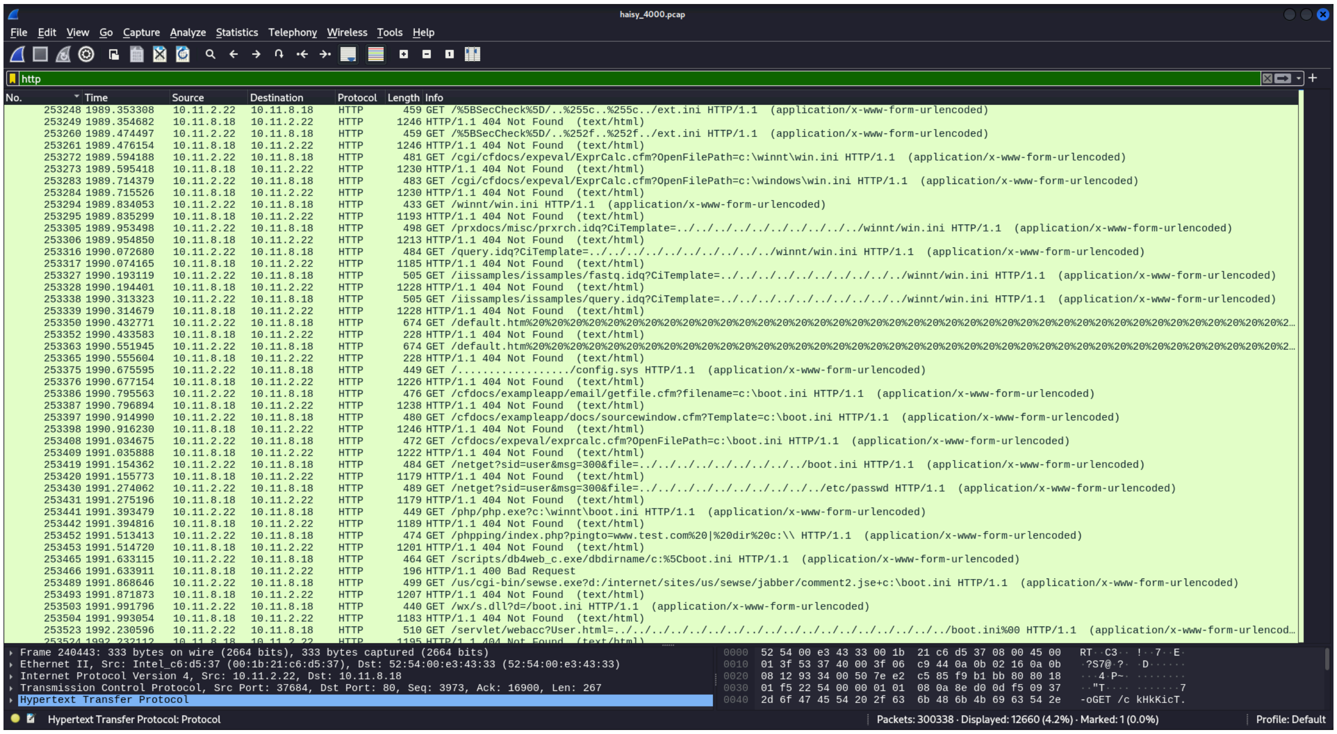
Address	Packets	Bytes	Tx Packets	Tx Bytes	Rx Packets	Rx Bytes
10.11.2.22	22,056	12 MB	13,936	3 MB	8,120	9 MB
10.11.8.17	258	27 kB	129	15 kB	129	12 kB
10.11.8.18	298,170	30 MB	146,257	19 MB	151,913	11 MB
37.120.246.146	274,967	18 MB	137,455	7 MB	137,512	10 MB
37.120.246.151	148	153 kB	8	5 kB	140	149 kB
91.198.174.192	19	2 kB	8	1 kB	11	1 kB
130.237.157.47	20	2 kB	9	1 kB	11	1 kB
130.237.157.97	176	19 kB	81	9 kB	95	10 kB
130.237.161.25	60	7 kB	27	4 kB	33	3 kB
172.17.3.3	109	53 kB	53	6 kB	56	47 kB
193.11.30.171	92	12 kB	42	7 kB	50	5 kB
209.51.188.174	177	132 kB	113	125 kB	64	7 kB
209.51.188.233	88	55 kB	52	52 kB	36	4 kB

Screenshot 2: Statistics > Endpoints > IPv4

The file `haisy_4000.pcap` contains 300338 captured network packets in total and 13 unique IP addresses occur. Of these IP addresses, 10.11.8.18 (146k transmitted packets, 151k received packets), 37.120.246.146 (137k transmitted packets, 137k received packets), 10.11.2.22 (14k transmitted packets, 8k received packets), 10.11.8.17 (129 transmitted packets, 129 received packets), 209.51.188.174 (113 transmitted packets, 64 received packets) and 130.237.157.97 (8 transmitted packets, 140 received packets) generated the most traffic. The two internal IP addresses that occur most frequently (10.11.8.18 and 10.11.2.22) are the Linux server hosting `haisy.cs2lab.dsv.su.se` and the reverse proxy. The top five ports for Linux server (10.11.8.18) are 80, 59808, 42250, 44779, 44216 and for the reverse proxy (10.11.2.22) are 33688, 42598, 49402, 39140, 48330. The only port used for 37.120.246.146 is 60836.

The three most common external IPs can be traced back geographically to Romania, Bucharest (37.120.246.146), United States, Boston (209.51.188.174) and Sweden, Stockholm (130.237.157.97).

When looking at the DNS related packets, it is noticeable that attempts were made to call domains that do not exist. For example, attempts were made to call up www.daisy.dsv.su.se or www.daisy.dsv.su.se.cs2lab.dsv.su.se. Domains such as fr.wikipedia.org or fsf.org, which have nothing directly to do with the daisy service, were called. In addition, there are DNS queries resolving to CNAME records such as mimas.dsv.su.se. CNAME records are used to alias one name to another. The presence of CNAME records isn't unusual, but it's important to verify that these canonical names are legitimate and expected within the network environment.



Screenshot 3: Filter "http"

When looking at the HTTP traffic, it is noticeable that many GET requests were made to url paths that do not exist. For example, attempts were made to call `/iissamples/issamples/fastq.idq?CiTemplate=../../../../../../../../../../../../winnt/win.ini`, `/readme.html` or `/admin.html` on the Linux server (10.11.8.18). This could be a sign that an attacker tried a directory/path traversal attack.

3.2 Investigating the Linux Server Image

```
$ sha1sum haisy.raw
6d08e3ec0c3caac2979070913010c1753c48f66f  haisy.raw

$ md5sum haisy.raw
89fd1b9b40f2b7793440a4a13d045837  haisy.raw

$ ls -l haisy.raw
-rw-r--r-- 1 kali kali 16106127360 Apr 23 09:21 haisy.raw
```

The verification of the SHA1 and MD5 hash values of `haisy.raw` has shown that the file had not

been altered or corrupted since its acquisition because the hash values match the values specified in the instructions. The file has a size of 16.1061GB. The image contains 5 partitions and the operating system according to the lsb-release file is Ubuntu 13.10 codename saucy.

```
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/bin/sh
(...)
postgres:x:107:114:PostgreSQL administrator,,,:/var/lib/postgresql:/bin/bash
whoopsie:x:108:115::/nonexistent:/bin/false
tomcat7:x:109:117::/usr/share/tomcat7:/bin/false
erika:x:1000:1000:Erika Thuning,,,:/home/erika:/bin/bash
```

The file `etc/passwd` contains entries for 29 user accounts whereby a home directory exists only for the user erika. The `.bash_history` of the user erika reveals several activities that could be considered suspicious:

- Repeated SSH Service Commands (`sudo /etc/init.d/ssh status, sudo status /etc/init.d/ssh, sudo start /etc/init.d/ssh, sudo /etc/init.d/ssh`): The frequent stopping, starting, and status checking of the SSH service could indicate trouble with the SSH service, or it might suggest that someone was trying to gain persistent access.
- Network Interface Cycling (`sudo ifconfig eth0 down, sudo ifconfig eth0 up, ...`): Regularly bringing the network interface eth0 down and back up is unusual and could be an attempt to evade network monitoring or reset network connections after unauthorized activities.
- Bypassing SSL Certificate Verification (`wget --no-check-certificate http://daisy.dsv.su.se, wget --no-check-certificate dsv.su.se, ...`): The use of –no-check-certificate with wget could suggest And HTTP (no encryption) is used.
- Editing Apache Configuration Files (`nano /etc/apache2/apache2.conf, cd /etc/apache2/, nano apache2.conf, ...`): Changes to the Apache configuration could be legitimate, but they can also indicate an attempt to alter the web server's behavior for malicious purposes, such as setting up a reverse proxy for traffic redirection.
- Direct Interaction with MySQL Database (`mysql -u erika -p < haisy_students_2023.sql, mysql -u root -p haisy < haisy_students_2023.sql, ...`): Importing/exporting data using the MySQL command line with root access can be a standard administrative task, but it can also be a way to inject malicious data or exfiltrate information.

Examination of `/var/log/syslog` shows that routine tasks such as DHCP renewals and periodic cron jobs were executed. Nevertheless, several segfaults occurred at different times. Segfaults indicate that a program has attempted to access memory for which it did not have permission, or that it has attempted to execute a process that was not allowed. The affected processes include QIRCKvwYm, oymZYryK, wbMEAsyiwVoHrV, GmBxfGdRbZmmDc, rvqSmSIKwxGhuXI, LkAjRzqB, gBWWFnVPKxb, zrrleSXSDFxI and LqqCzyBp. These processes appear to have been deleted,

which could indicate that they are temporary files or malware.

`/var/log/auth.log` also contains suspicious entries. The entry at Apr 24 00:03:46 indicates that the user "erika" executed the nano command with root privileges. The user "erika" seems to have edited a file (index.php) using nano. Another log entry at Apr 24 00:04:07 shows the user "erika" using sudo to run nano on the file index.php, again with root privileges. The entry at indicates that a cron session was opened for the root user. However, there is no corresponding "session closed" entry. It's unusual for a session to remain open without being closed properly. This might indicate a potential issue or anomaly.

According to the IT department, the Linux server was only running Tomcat and Apache services. In the `/etc/init.d` directory we found indication that also mysql, postgresql and samba were running. Especially the last one could be of interest, because samba is a file sharing service, that could potentially be used to exfiltrate information from the system.

Investigating the Tomcat log files showed that numerous HTTP requests attempting to include or execute a remote file (cirt.net/rfiinc.txt) on the server were executed. These requests are generating 404 errors, indicating that the requested files were not found on the server. This pattern of requests is indicative of an attempted Remote File Inclusion (RFI) attack. In an RFI attack, the attacker tries to exploit vulnerabilities in web applications by including and executing remote files hosted on external servers. Given the frequency and consistency of these requests, it's likely that there's an automated script or bot attempting to exploit potential vulnerabilities in the server or web application.

The log entries of the Apache server indicate several HTTP requests made by the IP address 10.11.2.22 to various endpoints on a server. The requests include attempts to access the root directory ("/"), "admin.html," "index.php," "php," "info.php," and many other endpoints that do not exist on the server, resulting in 404 Not Found errors. Additionally, there are requests made by a tool called Nikto/2.1.6, which appears to be a web vulnerability scanner. It tries various URLs with different extensions and parameters, likely attempting to find vulnerabilities or discover sensitive files on the server. This is consistent with the findings in network traffic with HTTP filter.

3.3 Timeline of Interesting Events

- **23/Apr/2023 22:17:13 (Apache Log File)** "GET /admin.html HTTP/1.1" 404 499 "-" "Mozilla/5.0 (Android 13; Mobile; rv:109.0) Gecko/112.0 Firefox/112.0" start of attack against Apache Server (potential path traversal attack to find vulnerabilities or public files)
- **24/Apr/2023 00:03:46 (System Log Files)**: The user "erika" executed the nano command with root privileges. The user "erika" seems to have edited a file (index.php) using nano.
- **24/Apr/2023 00:04:07 (System Log Files)**: The user "erika" used sudo to run nano on the file index.php, again with root privileges.
- **24/Apr/2023 06:25:01 (System Log Files)**: A cron session was opened for the root user. However, there is no corresponding "session closed" entry. It's unusual for a session to

remain open without being closed properly. This might indicate a potential issue or anomaly.

- **27/Apr/2023 23:49:17 (Tomcat Log Files)**: "GET /ckHkKicT.db HTTP/1.1" 404 973 (start of potential remote file inclusion attack). Matches with packet 239104 in `haisy_4000.pcap` (1832.858680 10.11.2.22 10.11.8.18 HTTP 329 GET /ckHkKicT.db HTTP/1.1). Therefore timestamp 15/Nov/2011 07:55:12 could correlate to 27/Apr/2023 23.49.17

4 Discussion

The comprehensive analysis of the server image and network traffic data reveals a series of suspicious activities and potential security threats targeting the Linux server. These activities encompass a wide range of techniques commonly associated with malicious intent, including repeated SSH service commands, network interface cycling, bypassing SSL certificate verification, editing Apache configuration files, direct interaction with the MySQL database, and attempted remote file inclusion attacks. Additionally, the presence of segfaults in the syslog further underscores the severity of the situation, suggesting possible exploitation attempts or system instability.

The involvement of multiple external IPs, traced back to Romania, the United States, and Sweden, adds another layer of complexity to the investigation. These IPs are associated with various suspicious activities, including attempted remote file inclusion attacks and probing of the server for vulnerabilities using tools like Nikto/2.1.6. Such coordinated and persistent probing from geographically diverse locations is highly indicative of a targeted attack aimed at compromising the server's security.

Moreover, the activities attributed to the user "erika" on the Linux server raise significant red flags. The repeated execution of sensitive commands with root privileges, such as SSH service manipulation, network interface cycling, SSL certificate bypassing, Apache configuration file editing, and direct interaction with the MySQL database, strongly suggest an insider threat or unauthorized access by an individual with malicious intent.

The culmination of these findings leads to the conclusion that the system has likely been subjected to a sophisticated and coordinated attack. The combination of external probing, suspicious user behavior, and anomalous system activities indicates a concerted effort to compromise the server's integrity and potentially exfiltrate sensitive data or establish persistent access for future exploitation. Immediate action is imperative to mitigate the damage and prevent further compromise. This includes, patching known vulnerabilities, resetting compromised credentials, enhancing access controls, and implementing robust monitoring mechanisms to detect and respond to future threats proactively. Furthermore, it's essential to review and reinforce security protocols, conduct employee training on cybersecurity best practices, and remain vigilant against evolving attack vectors. Collaborating with law enforcement and cybersecurity experts may also be necessary to investigate the incident thoroughly and hold accountable those responsible for the attack

References

[1] "An Ethernet Address Resolution Protocol: Or Converting Network Protocol Addresses to 48.bit Ethernet Address for Transmission on Ethernet Hardware." RFC 826, 1982.

[2] V. Ramachandran and S. Nandi, "Detecting ARP Spoofing: An Active Technique," in *Information Systems Security* (S. Jajodia and C. Mazumdar, eds.), (Berlin, Heidelberg), pp. 239–250, Springer Berlin Heidelberg, 2005.