Incident Report - Premium House Lights

Course 12: Capstone Project Incident Report

By Evelyn Wolfe | September 23 2024 Cyber Security Flex Cohort

Executive Summary

On Tuesday, February 19, 2022, Premium House Lights was targeted by a fast-moving cyberattack that resulted in the theft of sensitive customer information. In under five minutes, the attacker exploited a vulnerable upload feature on the company's public website, allowing them to deploy a malicious script to the WebServer. Once inside the internal environment, the attacker scanned the network, located the customer database, and copied its entire contents. Shortly after the intrusion, a ransom email was received demanding cryptocurrency in exchange for not leaking the stolen data. The message included real customer names and phone numbers, confirming the authenticity and severity of the breach.

The attack exposed several critical weaknesses in the company's digital environment. These included the absence of controls to prevent unsafe file uploads, minimal internal segmentation to prevent lateral movement between systems, and a lack of monitoring that might have detected the attack sooner. In immediate response, the System Administration Team took the affected servers offline, restored them from clean backups, and blocked the attacker's known IP addresses. Internal teams also began a detailed review of logs and systems to assess the full extent of the compromise and prepare a broader incident response.

This report presents a detailed timeline of the attack, supported by log evidence, captured network traffic, and attacker behavior patterns. Based on this analysis, Premium House Lights is working to implement a range of cybersecurity improvements aligned with recognized standards, including NIST SP 800-53, ISO/IEC 27001, and PIPEDA. Planned actions include securing file upload paths, deploying a web application firewall (WAF), enforcing least privilege access controls, and establishing a formal ransomware response plan. These steps are intended to address current gaps while strengthening the organization's long-term cybersecurity posture.

Timeline of the Incident

The following timeline provides the speed in which the attack happened, and a breakdown of events that took place during the unauthorized access and data exfiltration affecting the PHL WebServer and database server. Timestamps are shown in both Eastern Standard Time (EST) and Coordinated Universal Time (UTC) for clarity. This timeline is based on correlated evidence pulled from web access logs, shell command logs, MySQL database activity, and packet captures from Wireshark. The activity of the attack took less than five minutes to exfiltrate the data of our clients, and move it off company owned servers. *All times referenced in the report are in Eastern Standard Time (EST), unless otherwise noted.*

Time (EST) Feb 19, 2022	**Time (UTC) Feb 20, 2022**	**Event Description**
21:58:40	02:58:40	Attacker received a 200 OK response while accessing the /uploads/ directory, confirming it was publicly accessible and directory listing was enabled.
21:59:04	02:59:04	The attacker uploaded a web shell (shell.php) to the /uploads/ directory. A second 200 OK response confirmed the file upload was successful. This marked the point of initial compromise.
21:59:45	02:59:45	The attacker began an ARP requests to discover devices on the 10.10.1.x subnet, the internal WebServer IP Address. Allowing the attacker to identify the Database Server as 10.10.1.3
21:59:47	02:59:47	The attacker probed 10.10.1.3 for an open port to laterally moving, finding port 3306 open, and moving from WebServer to Database Server
22:00:55	03:00:55	The attacker used the uploaded shell to gain elevated access and executed commands with root privileges. They initiated a MySQL session using sudo mysql -u root -p.
22:01:45	03:01:45	The attacker ran mysqldump to export the phl database contents into a file called phl.db, containing sensitive customer information.
22:02:36	03:02:36	The phl.db file was deleted from the local system, likely as an attempt to cover their tracks.
22:02:26 to 22.02:30	03:02:26 to 03:02:30	Wireshark packet capture shows encrypted SSH traffic from internal IP 147.182.157.9 to 178.62.228.28 over TCP port 22, confirming the moment of data exfiltration.
22:00:48 to 22:02:56	03:00:48 to 03:02:56	Post-Exfiltration Persistence Attempt: Following the data exfiltration, Wireshark data indicates the attacker attempted to establish persistence and fallback access channels. On the database server, a Telnet connection was initiated, an insecure protocol rarely used in modern environments, suggesting an effort to maintain internal access. Meanwhile, the WebServer continued to show activity on port 4444, commonly associated with Metasploit's Meterpreter, which is often used to leave a backdoor for remote control or future exploitation.

Technician Analysis

Attack Origin, Review, and Impact

The attack on Premium House Lights (PHL) began on February 19, 2022, at 21:58 when the IP address 138.68.92.163 and 138.122.33.221 first appeared in the web access logs for www.premiumhouselights.com. This was followed closely by probing activity captured in Wireshark, originating from **134.122.33.221**, as shown in *Image 2*. The attacker was actively scanning for vulnerabilities, sending repeated requests that initially resulted in HTTP 404 errors — indicating failed attempts to locate accessible pages or directories (*see Image 1.*)

```
138.68.92.163 - - [19/Feb/2022:21:58:22 -0500] "GET /randomfile1 HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:22 -0500] "GET /frand2 HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:22 -0500] "GET /index HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:22 -0500] "GET /archive HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:22 -0500] "GET /02 HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:22 -0500] "GET /register HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:22 -0500] "GET /en HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:22 -0500] "GET /forum HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:23 -0500] "GET /software HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
138.68.92.163 - - [19/Feb/2022:21:58:23 -0500] "GET /downloads HTTP/1.1" 404 437 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
```

Image 1: WebServer Probing Attempts
Example of the attempts at probing by the IP 138.67.92.163.

344	2022-02-19 21:58:22.250140	134.122.33.221	138.68.92.163	HTTP	505 HTTP/1.1 404 Not Found (text/html)
345	2022-02-19 21:58:22.347570	138.68.92.163	134.122.33.221	TCP	68 54944 → 80 [ACK] Seq=129 Ack=438 Win=63872 Len=0 TSval=1054345922 TSecr=4059173918
346	2022-02-19 21:58:22.347570	138.68.92.163	134.122.33.221	HTTP	191 GET /frand2 HTTP/1.1
347	2022-02-19 21:58:22.347632	134.122.33.221	138.68.92.163	TCP	68 80 → 54944 [ACK] Seq=438 Ack=252 Win=65152 Len=0 TSval=4059174016 TSecr=1054345922
348	2022-02-19 21:58:22.347889	134.122.33.221	138.68.92.163	HTTP	505 HTTP/1.1 404 Not Found (text/html)
349	2022-02-19 21:58:22.445211	138.68.92.163	134.122.33.221	TCP	68 54944 → 80 [ACK] Seq=252 Ack=875 Win=63744 Len=0 TSval=1054346019 TSecr=4059174016
350	2022-02-19 21:58:22.445551	138.68.92.163	134.122.33.221	HTTP	190 GET /index HTTP/1.1
351	2022-02-19 21:58:22.445581	134.122.33.221	138.68.92.163	TCP	68 80 → 54944 [ACK] Seq=875 Ack=374 Win=65152 Len=0 TSval=4059174114 TSecr=1054346020
352	2022-02-19 21:58:22.445831	134.122.33.221	138.68.92.163	HTTP	505 HTTP/1.1 404 Not Found (text/html)
353	2022-02-19 21:58:22.543216	138.68.92.163	134.122.33.221	TCP	68 54944 → 80 [ACK] Seq=374 Ack=1312 Win=63744 Len=0 TSval=1054346117 TSecr=4059174114
354	2022-02-19 21:58:22.543329	138.68.92.163	134.122.33.221	HTTP	192 GET /archive HTTP/1.1
355	2022-02-19 21:58:22.543358	134.122.33.221	138.68.92.163	TCP	68 80 → 54944 [ACK] Seq=1312 Ack=498 Win=65152 Len=0 TSval=4059174211 TSecr=1054346118
356	2022-02-19 21:58:22.543629	134.122.33.221	138.68.92.163	HTTP	505 HTTP/1.1 404 Not Found (text/html)
357	2022-02-19 21:58:22.641382	138.68.92.163	134.122.33.221	HTTP	187 GET /02 HTTP/1.1
358	2022-02-19 21:58:22.641679	134.122.33.221	138.68.92.163	HTTP	505 HTTP/1.1 404 Not Found (text/html)
359	2022-02-19 21:58:22.739437	138.68.92.163	134.122.33.221	HTTP	193 GET /register HTTP/1.1
360	2022-02-19 21:58:22.739747	134.122.33.221	138.68.92.163	HTTP	505 HTTP/1.1 404 Not Found (text/html)
361	2022-02-19 21:58:22.837568	138.68.92.163	134.122.33.221	HTTP	187 GET /en HTTP/1.1
362	2022-02-19 21:58:22.837966	134.122.33.221	138.68.92.163	HTTP	505 HTTP/1.1 404 Not Found (text/html)

Image 2: Wireshark Probing Attempts
Wireshark cross reference to the logs from WebServer

At 21:58, the attacker received an HTTP 200 OK response when attempting to access the */uploads/* directory, essentially a green checkmark from the server confirming that the directory existed and could be reached from the internet. This response told the attacker that the path was valid and likely unprotected, signaling a potential vulnerability. Armed with this confirmation, the attacker proceeded just a moment later, at 21:59, to upload a malicious file by targeting */uploads/shell.php*. Once again, the server responded with a 200 OK, indicating that the upload had not only succeeded but that the file was now stored and accessible on the server. This marked the moment the attacker gained a foothold, opening the door to remote command-line access into the underlying WebServer (*see Image 3 and Image 4*).

```
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"
```

Image 3: WebServer Data Logs
Timestamps of the attacker receiving the “200 Ok Response”

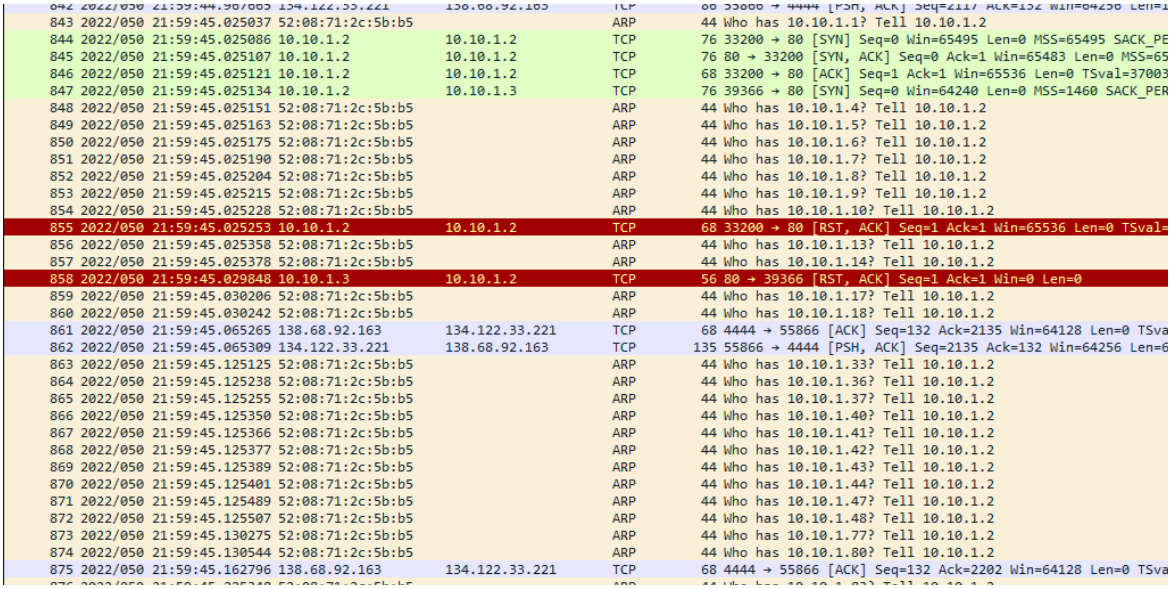
739	2022-02-19 21:58:40.813039	134.122.33.221	138.68.92.163	HTTP	1183 HTTP/1.1 200 OK (text/html)
740	2022-02-19 21:58:40.912018	138.68.92.163	134.122.33.221	TCP	68 54946 → 80 [FIN, ACK] Seq=10879 Ack=39277 Win=64128 Len=0 TSval=1054364486 TSecr=4059192481
741	2022-02-19 21:58:40.912143	134.122.33.221	138.68.92.163	TCP	68 80 → 54946 [FIN, ACK] Seq=39277 Ack=10880 Win=64256 Len=0 TSval=4059192580 TSecr=1054364486
742	2022-02-19 21:58:41.009744	138.68.92.163	134.122.33.221	TCP	68 54946 → 80 [ACK] Seq=10880 Ack=39278 Win=64128 Len=0 TSval=1054364584 TSecr=4059192580
743	2022-02-19 21:58:53.218396	20.119.213.210	134.122.33.221	TCP	76 33860 → 63643 [SYN] Seq=0 Win=64240 Len=0 MSS=1440 SACK_PERM TSval=3508770235 TSecr=0 WS=128
744	2022-02-19 21:58:53.218441	134.122.33.221	20.119.213.210	TCP	56 63643 → 33860 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
745	2022-02-19 21:58:55.612427	138.68.92.163	134.122.33.221	TCP	76 54948 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=1054379185 TSecr=0 WS=128
746	2022-02-19 21:58:55.612478	134.122.33.221	138.68.92.163	TCP	76 80 → 54948 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PERM TSval=4059207281 TSecr=1054379185 WS=128
747	2022-02-19 21:58:55.711642	138.68.92.163	134.122.33.221	TCP	68 54948 → 80 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=1054379285 TSecr=4059207281
748	2022-02-19 21:58:55.711642	138.68.92.163	134.122.33.221	HTTP	154 GET /uploads/ HTTP/1.1
749	2022-02-19 21:58:55.711715	134.122.33.221	138.68.92.163	TCP	68 80 → 54948 [ACK] Seq=1 Ack=87 Win=65152 Len=0 TSval=4059207380 TSecr=1054379286
750	2022-02-19 21:58:55.712217	134.122.33.221	138.68.92.163	HTTP	1183 HTTP/1.1 200 OK (text/html)
751	2022-02-19 21:58:55.809683	138.68.92.163	134.122.33.221	TCP	68 54948 → 80 [ACK] Seq=87 Ack=1116 Win=64128 Len=0 TSval=1054379384 TSecr=4059207380
752	2022-02-19 21:58:55.810125	138.68.92.163	134.122.33.221	TCP	68 54948 → 80 [FIN, ACK] Seq=87 Ack=1116 Win=64128 Len=0 TSval=1054379385 TSecr=4059207380
753	2022-02-19 21:58:55.810225	134.122.33.221	138.68.92.163	TCP	68 80 → 54948 [FIN, ACK] Seq=1116 Ack=88 Win=65152 Len=0 TSval=4059207478 TSecr=1054379385
754	2022-02-19 21:58:55.907775	138.68.92.163	134.122.33.221	TCP	68 54948 → 80 [ACK] Seq=88 Ack=1117 Win=64128 Len=0 TSval=1054379482 TSecr=4059207478
755	2022-02-19 21:58:59.806434	20.104.250.91	134.122.33.221	TCP	76 33954 → 63643 [SYN] Seq=0 Win=64240 Len=0 MSS=1440 SACK_PERM TSval=121354095 TSecr=0 WS=128

Image 4: Wireshark - Directory Probe - 200 Code Ok
Cross reference to the timeline when the attacker was able to transfer the file, line 739 is the point they received OK for “Get /upload/ HTTP/1.1”

786	2022/050 21:59:04.073598	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
787	2022/050 21:59:04.073651	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 WS=128
788	2022/050 21:59:04.171702	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
789	2022/050 21:59:04.171795	138.68.92.163	134.122.33.221	HTTP	589 POST /uploads/shell.php HTTP/1.1 (application/x-www-form-urlencoded)
790	2022/050 21:59:04.171843	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
791	2022/050 21:59:04.191040	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	2022/050 21:59:04.289759	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	2022/050 21:59:04.289822	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	2022/050 21:59:04.291723	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	2022/050 21:59:04.389586	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	2022/050 21:59:04.389627	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	2022/050 21:59:04.487209	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

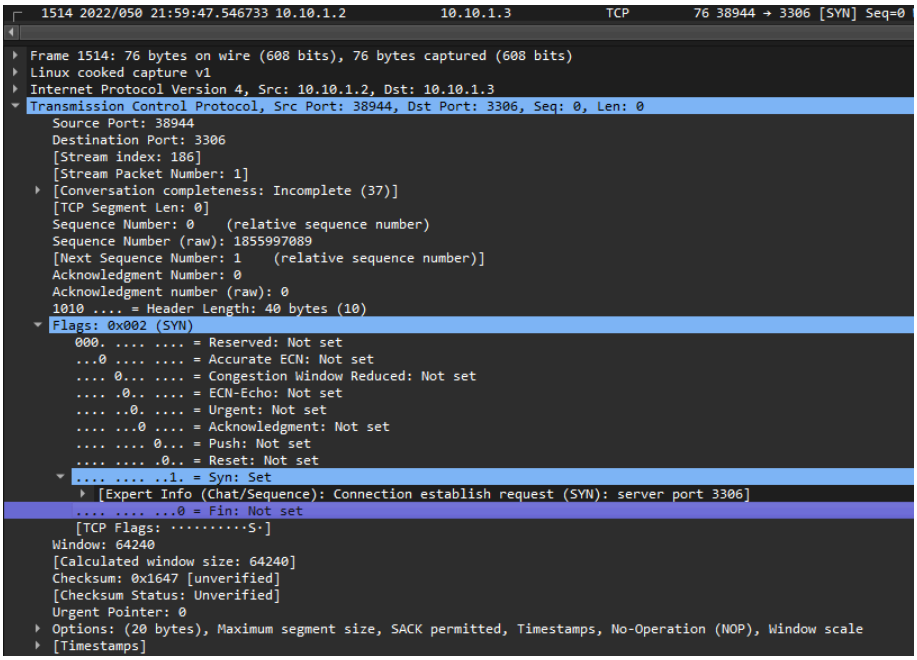
Image 5: Wireshark - Web Shell Script Execution
Wireshark Packet #789 confirms the exact moment the web shell (/upload/shell.php) is activated, with a POST request issued from the attacker’s IP.

After gaining access to the WebServer (internal IP: 10.10.1.2), the attacker began searching the internal network for other systems. At 21:59:45 (Packet #843, Image 6), they sent out requests to identify nearby devices, which led them to 10.10.1.3, the internal Database Server. Although the scan only took a few seconds, it generated thousands of entries in the network logs. Packet #1514 (Image 7) and Packet #49 (Image 8) show that the attacker successfully connected to the database through its open MySQL port. This connection confirmed that the database was accessible, and marked the moment the attacker moved deeper into the network. While their original public IP addresses (138.68.92.163 and 134.122.33.221) are not visible during this phase, the network activity clearly shows how they moved from one system to another.



843	2022/050 21:59:45.025037	52:08:71:2c:5b:b5		ARP	44 Who has 10.10.1.1? Tell 10.10.1.2
844	2022/050 21:59:45.025086	10.10.1.2	10.10.1.2	TCP	76 33200 → 80 [SYN] Seq=0 Win=65495 Len=0 MSS=65495 SACK_PE
845	2022/050 21:59:45.025107	10.10.1.2	10.10.1.2	TCP	76 80 → 33200 [SYN, ACK] Seq=0 Ack=1 Win=65483 Len=0 MSS=65
846	2022/050 21:59:45.025121	10.10.1.2	10.10.1.2	TCP	68 33200 → 80 [ACK] Seq=1 Ack=1 Win=65536 Len=0 TSval=37003
847	2022/050 21:59:45.025134	10.10.1.2	10.10.1.3	TCP	76 39366 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PER
848	2022/050 21:59:45.025151	52:08:71:2c:5b:b5		ARP	44 Who has 10.10.1.4? Tell 10.10.1.2
849	2022/050 21:59:45.025163	52:08:71:2c:5b:b5		ARP	44 Who has 10.10.1.5? Tell 10.10.1.2
850	2022/050 21:59:45.025175	52:08:71:2c:5b:b5		ARP	44 Who has 10.10.1.6? Tell 10.10.1.2
851	2022/050 21:59:45.025190	52:08:71:2c:5b:b5		ARP	44 Who has 10.10.1.7? Tell 10.10.1.2
852	2022/050 21:59:45.025204	52:08:71:2c:5b:b5		ARP	44 Who has 10.10.1.8? Tell 10.10.1.2
853	2022/050 21:59:45.025215	52:08:71:2c:5b:b5		ARP	44 Who has 10.10.1.9? Tell 10.10.1.2
854	2022/050 21:59:45.025228	52:08:71:2c:5b:b5		ARP	44 Who has 10.10.1.10? Tell 10.10.1.2
855	2022/050 21:59:45.025253	10.10.1.2	10.10.1.2	TCP	68 33200 → 80 [RST, ACK] Seq=1 Ack=1 Win=65536 Len=0 TSval=
856	2022/050 21:59:45.025358	52:08:71:2c:5b:b5		ARP	44 Who has 10.10.1.13? Tell 10.10.1.2
857	2022/050 21:59:45.025378	52:08:71:2c:5b:b5		ARP	44 Who has 10.10.1.14? Tell 10.10.1.2
858	2022/050 21:59:45.029848	10.10.1.3	10.10.1.2	TCP	56 80 → 39366 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
859	2022/050 21:59:45.030206	52:08:71:2c:5b:b5		ARP	44 Who has 10.10.1.17? Tell 10.10.1.2
860	2022/050 21:59:45.030242	52:08:71:2c:5b:b5		ARP	44 Who has 10.10.1.18? Tell 10.10.1.2
861	2022/050 21:59:45.065265	138.68.92.163	134.122.33.221	TCP	68 4444 → 55866 [ACK] Seq=132 Ack=2135 Win=64128 Len=0 TSva
862	2022/050 21:59:45.065309	134.122.33.221	138.68.92.163	TCP	135 55866 → 4444 [PSH, ACK] Seq=2135 Ack=132 Win=64256 Len=6
863	2022/050 21:59:45.125125	52:08:71:2c:5b:b5		ARP	44 Who has 10.10.1.33? Tell 10.10.1.2
864	2022/050 21:59:45.125238	52:08:71:2c:5b:b5		ARP	44 Who has 10.10.1.36? Tell 10.10.1.2
865	2022/050 21:59:45.125255	52:08:71:2c:5b:b5		ARP	44 Who has 10.10.1.37? Tell 10.10.1.2
866	2022/050 21:59:45.125350	52:08:71:2c:5b:b5		ARP	44 Who has 10.10.1.40? Tell 10.10.1.2
867	2022/050 21:59:45.125366	52:08:71:2c:5b:b5		ARP	44 Who has 10.10.1.41? Tell 10.10.1.2
868	2022/050 21:59:45.125377	52:08:71:2c:5b:b5		ARP	44 Who has 10.10.1.42? Tell 10.10.1.2
869	2022/050 21:59:45.125389	52:08:71:2c:5b:b5		ARP	44 Who has 10.10.1.43? Tell 10.10.1.2
870	2022/050 21:59:45.125401	52:08:71:2c:5b:b5		ARP	44 Who has 10.10.1.44? Tell 10.10.1.2
871	2022/050 21:59:45.125489	52:08:71:2c:5b:b5		ARP	44 Who has 10.10.1.47? Tell 10.10.1.2
872	2022/050 21:59:45.125507	52:08:71:2c:5b:b5		ARP	44 Who has 10.10.1.48? Tell 10.10.1.2
873	2022/050 21:59:45.130275	52:08:71:2c:5b:b5		ARP	44 Who has 10.10.1.77? Tell 10.10.1.2
874	2022/050 21:59:45.130544	52:08:71:2c:5b:b5		ARP	44 Who has 10.10.1.80? Tell 10.10.1.2
875	2022/050 21:59:45.162796	138.68.92.163	134.122.33.221	TCP	68 4444 → 55866 [ACK] Seq=132 Ack=2202 Win=64128 Len=0 TSva

Image 6: Wireshark Packet #843: ARP Scan from WebServer
ARP probe observed in Packet #843, showing the attacker querying 10.10.1.1 from 10.10.1.2.



```
1514 2022/050 21:59:47.546733 10.10.1.2          10.10.1.3          TCP       76 38944 → 3306 [SYN] Seq=0
Frame 1514: 76 bytes on wire (608 bits), 76 bytes captured (608 bits)
Linux cooked capture v1
Internet Protocol Version 4, Src: 10.10.1.2, Dst: 10.10.1.3
Transmission Control Protocol, Src Port: 38944, Dst Port: 3306, Seq: 0, Len: 0
    Source Port: 38944
    Destination Port: 3306
    [Stream index: 186]
    [Stream Packet Number: 1]
  ▸ [Conversation completeness: Incomplete (37)]
    [TCP Segment Len: 0]
    Sequence Number: 0    (relative sequence number)
    Sequence Number (raw): 1855997089
    [Next Sequence Number: 1    (relative sequence number)]
    Acknowledgment Number: 0
    Acknowledgment number (raw): 0
    1010 .... = Header Length: 40 bytes (10)
  ▾ Flags: 0x002 (SYN)
      000. .... .... = Reserved: Not set
      ...0 .... .... = Accurate ECN: Not set
      .... 0... .... = Congestion Window Reduced: Not set
      .... .0.. .... = ECN-Echo: Not set
      .... ..0. .... = Urgent: Not set
      .... ...0 .... = Acknowledgment: Not set
      .... .... 0... = Push: Not set
      .... .... .0.. = Reset: Not set
    ▾ .... .... ..1. = Syn: Set
      ▸ [Expert Info (Chat/Sequence): Connection establish request (SYN): server port 3306]
        .... .... ...0 = Fin: Not set
        [TCP Flags: ··········S·]
    Window: 64240
    [Calculated window size: 64240]
    Checksum: 0x1647 [unverified]
    [Checksum Status: Unverified]
    Urgent Pointer: 0
  ▸ Options: (20 bytes), Maximum segment size, SACK permitted, Timestamps, No-Operation (NOP), Window scale
  ▸ [Timestamps]
```

Image 7: Wireshark #1514 - SYN from WebServer to MySQL(3306) Connection
This shows that IP address 10.10.1.3 responded to a connection request on port 3306, confirming it was open, allowing the attacker to move from 10.10.1.2 to the DB Server.

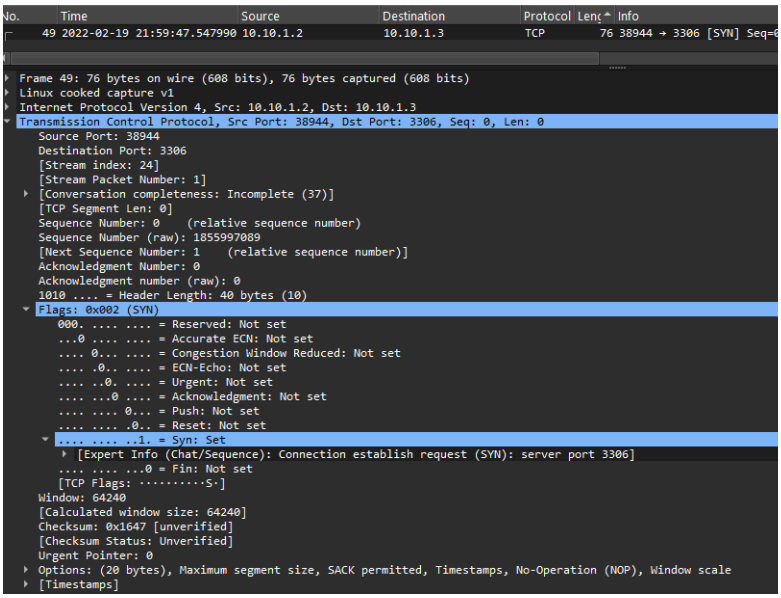


Image 8: Wireshark Packet #49 - SYN-ACK from DB Server Accepting the Connection
This reflects the same time and data that the WebServer requested from the Database Server, allowing connection coming from 10.10.1.2 to 10.10.1.3

Both systems resided on the same VLAN, with no network segmentation in place, a significant oversight that allowed seamless movement between systems. On the database server, the attacker carried out two primary actions. First, shell activity recorded in *phl_database_shell.txt* confirms that they obtained root-level access and used *mysqldump* to export the entire customers table, effectively creating a full copy of all stored customer records. Second, they attempted to establish a backdoor connection via the TelNet protocol, which is outdated and not used in modern production environments (*see Image 9*). After they ran the script to remove the client information from the database, the attacker attempted to delete the *phl.db* file with command rm phl.db. *This activity is captured and timestamped in Image 10, showing direct evidence of the export operation taking place.*

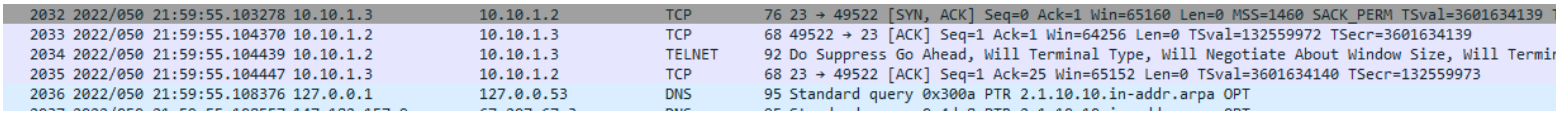


Image9: Wireshark Capture of TelNet Protocol Opening

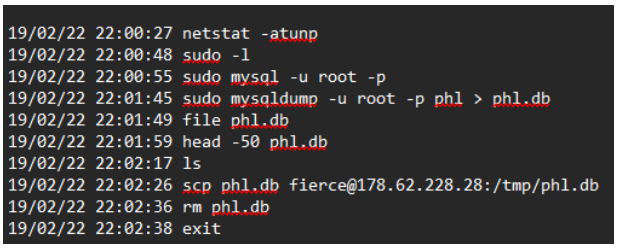
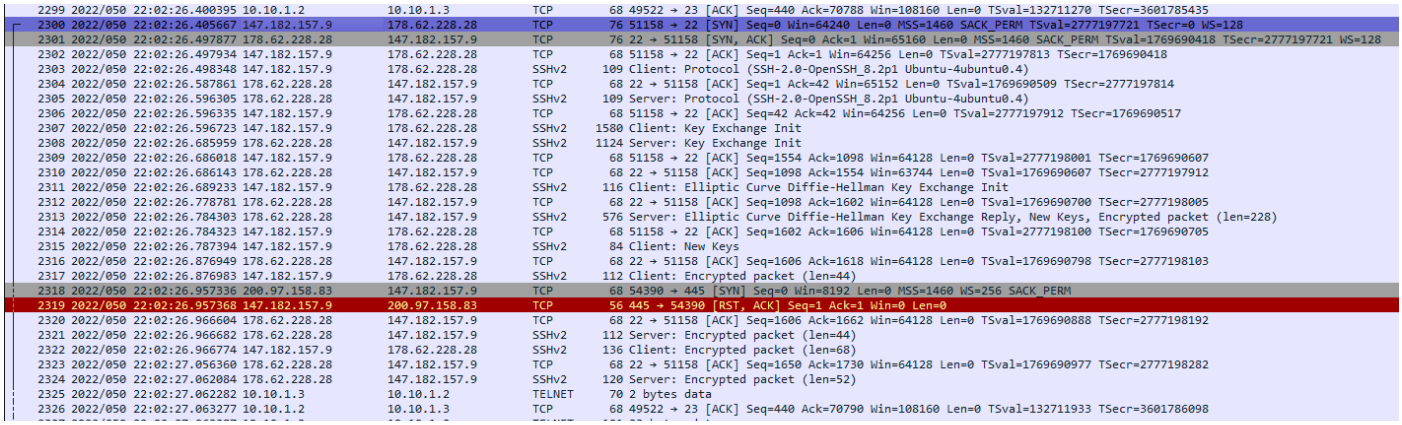


Image 10: PHL Database Logs
Showing the attacker transferring the client data to remote IP 178.62.228.28

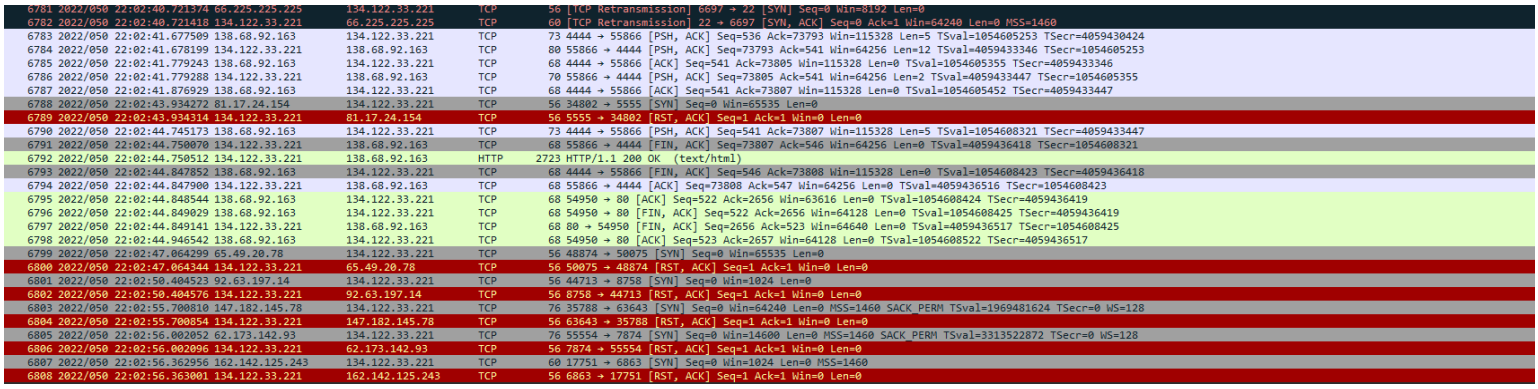
The database dump was then exfiltrated to an external server at IP 178.62.228.28, confirming that sensitive client data had been removed from the environment. At approximately 22:02:26, Wireshark captured a secure connection from the internal database server (our public IP 147.182.157.9 from our internet service provider) to this external IP. The timing of this connection aligns precisely with the execution of the *mysqldump* command, strongly indicating that this was the moment the stolen database was transferred offsite (*see Image 11).* While the contents of the transmission were encrypted and unreadable, the size and timing of the data flow provide strong evidence confirming the breach and exfiltration event.



```
2299 2022/050 22:02:26.400395 10.10.1.2          10.10.1.3          TCP       68 49522 → 23 [ACK] Seq=440 Ack=70788 Win=108160 Len=0 TSval=132711270 TSecr=3601785435
2300 2022/050 22:02:26.405667 147.182.157.9      178.62.228.28      TCP       76 51158 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=2777197721 TSecr=0 WS=128
2301 2022/050 22:02:26.497877 178.62.228.28      147.182.157.9      TCP       76 22 → 51158 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK_PERM TSval=1769690418 TSecr=2777197721 WS=128
2302 2022/050 22:02:26.497934 147.182.157.9      178.62.228.28      TCP       68 51158 → 22 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=2777197813 TSecr=1769690418
2303 2022/050 22:02:26.498348 147.182.157.9      178.62.228.28      SSHv2    109 Client: Protocol (SSH-2.0-OpenSSH_8.2p1 Ubuntu-4ubuntu0.4)
2304 2022/050 22:02:26.587861 178.62.228.28      147.182.157.9      TCP       68 22 → 51158 [ACK] Seq=1 Ack=42 Win=65152 Len=0 TSval=1769690509 TSecr=2777197814
2305 2022/050 22:02:26.596305 178.62.228.28      147.182.157.9      SSHv2    109 Server: Protocol (SSH-2.0-OpenSSH_8.2p1 Ubuntu-4ubuntu0.4)
2306 2022/050 22:02:26.596335 147.182.157.9      178.62.228.28      TCP       68 51158 → 22 [ACK] Seq=42 Ack=42 Win=64256 Len=0 TSval=2777197912 TSecr=1769690517
2307 2022/050 22:02:26.596723 147.182.157.9      178.62.228.28      SSHv2   1580 Client: Key Exchange Init
2308 2022/050 22:02:26.685959 178.62.228.28      147.182.157.9      SSHv2   1124 Server: Key Exchange Init
2309 2022/050 22:02:26.686018 147.182.157.9      178.62.228.28      TCP       68 51158 → 22 [ACK] Seq=1554 Ack=1098 Win=64128 Len=0 TSval=2777198001 TSecr=1769690607
2310 2022/050 22:02:26.686143 178.62.228.28      147.182.157.9      TCP       68 22 → 51158 [ACK] Seq=1098 Ack=1554 Win=63744 Len=0 TSval=1769690607 TSecr=2777197912
2311 2022/050 22:02:26.689233 147.182.157.9      178.62.228.28      SSHv2    116 Client: Elliptic Curve Diffie-Hellman Key Exchange Init
2312 2022/050 22:02:26.778781 178.62.228.28      147.182.157.9      TCP       68 22 → 51158 [ACK] Seq=1098 Ack=1602 Win=64128 Len=0 TSval=1769690700 TSecr=2777198005
2313 2022/050 22:02:26.784303 178.62.228.28      147.182.157.9      SSHv2    576 Server: Elliptic Curve Diffie-Hellman Key Exchange Reply, New Keys, Encrypted packet (len=228)
2314 2022/050 22:02:26.784323 147.182.157.9      178.62.228.28      TCP       68 51158 → 22 [ACK] Seq=1602 Ack=1606 Win=64128 Len=0 TSval=2777198100 TSecr=1769690705
2315 2022/050 22:02:26.787394 147.182.157.9      178.62.228.28      SSHv2     84 Client: New Keys
2316 2022/050 22:02:26.876949 178.62.228.28      147.182.157.9      TCP       68 22 → 51158 [ACK] Seq=1606 Ack=1618 Win=64128 Len=0 TSval=1769690798 TSecr=2777198103
2317 2022/050 22:02:26.876983 147.182.157.9      178.62.228.28      SSHv2    112 Client: Encrypted packet (len=44)
2318 2022/050 22:02:26.957336 200.97.158.83      147.182.157.9      TCP       68 54390 → 445 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM
2319 2022/050 22:02:26.957368 147.182.157.9      200.97.158.83      TCP       56 445 → 54390 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2320 2022/050 22:02:26.966604 178.62.228.28      147.182.157.9      TCP       68 22 → 51158 [ACK] Seq=1606 Ack=1662 Win=64128 Len=0 TSval=1769690888 TSecr=2777198192
2321 2022/050 22:02:26.966682 178.62.228.28      147.182.157.9      SSHv2    112 Server: Encrypted packet (len=44)
2322 2022/050 22:02:26.966774 147.182.157.9      178.62.228.28      SSHv2    136 Client: Encrypted packet (len=68)
2323 2022/050 22:02:27.056360 178.62.228.28      147.182.157.9      TCP       68 22 → 51158 [ACK] Seq=1650 Ack=1730 Win=64128 Len=0 TSval=1769690977 TSecr=2777198282
2324 2022/050 22:02:27.062084 178.62.228.28      147.182.157.9      SSHv2    120 Server: Encrypted packet (len=52)
2325 2022/050 22:02:27.062282 10.10.1.3          10.10.1.2          TELNET    70 2 bytes data
2326 2022/050 22:02:27.063277 10.10.1.2          10.10.1.3          TCP       68 49522 → 23 [ACK] Seq=440 Ack=70790 Win=108160 Len=0 TSval=132711933 TSecr=3601786098
```

Image 11: Wireshark Client Data Record
Wireshark capture shows the data transfer between PHL and the attacker.

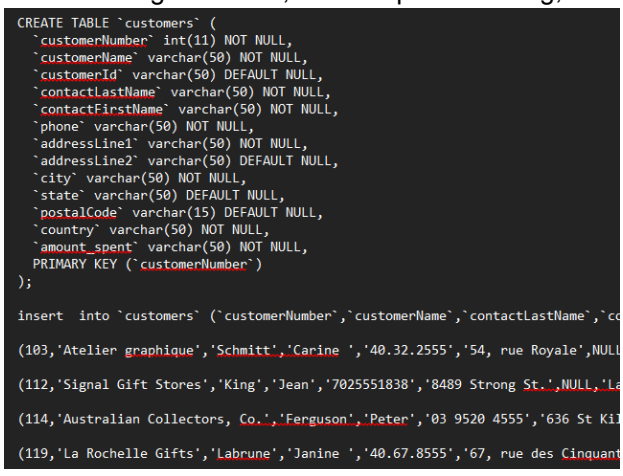
Following the completion of the data exfiltration, the attacker appeared to initiate a final method of maintaining access to the environment. Activity was detected on TCP port 4444 from the WebServer, a port commonly associated with Metasploit's Meterpreter — a known tool for establishing backdoors that allow remote control or delayed exploitation. Wireshark packet analysis (*Image 12*) confirms sustained traffic over this port, beginning shortly after the exfiltration event and continuing until the connection was closed at 22:02:56. This suggests the attacker was either staging for future reentry or verifying the availability of a persistent access channel before disengaging.



```
6781 2022/050 22:02:40.721374 66.225.225.225     134.122.33.221     TCP       56 [TCP Retransmission] 6897 → 22 [SYN] Seq=0 Win=8192 Len=0
6782 2022/050 22:02:40.721418 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
6783 2022/050 22:02:41.677509 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
6784 2022/050 22:02:41.678199 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
6785 2022/050 22:02:41.779243 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
6786 2022/050 22:02:41.779288 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
6787 2022/050 22:02:41.876929 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
6788 2022/050 22:02:43.934272 81.17.24.154       134.122.33.221     TCP       56 34802 → 5555 [SYN] Seq=0 Win=65535 Len=0
6789 2022/050 22:02:43.934314 134.122.33.221     81.17.24.154       TCP       56 5555 → 34802 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
6790 2022/050 22:02:44.745173 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
6791 2022/050 22:02:44.750070 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
6792 2022/050 22:02:44.750512 134.122.33.221     138.68.92.163      HTTP    2723 HTTP/1.1 200 OK  (text/html)
6793 2022/050 22:02:44.847852 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
6794 2022/050 22:02:44.847900 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
6795 2022/050 22:02:44.848544 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
6796 2022/050 22:02:44.849029 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
6797 2022/050 22:02:44.849141 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
6798 2022/050 22:02:44.946542 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
6799 2022/050 22:02:47.064299 65.49.20.78        134.122.33.221     TCP       56 48874 → 50075 [SYN] Seq=0 Win=65535 Len=0
6800 2022/050 22:02:47.064344 134.122.33.221     65.49.20.78        TCP       56 50075 → 48874 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
6801 2022/050 22:02:50.404523 92.63.197.14       134.122.33.221     TCP       56 44713 → 8758 [SYN] Seq=0 Win=1024 Len=0
6802 2022/050 22:02:50.404576 134.122.33.221     92.63.197.14       TCP       56 8758 → 44713 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
6803 2022/050 22:02:55.700810 147.182.145.78     134.122.33.221     TCP       76 35788 → 63643 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=1969481624 TSecr=0 WS=128
6804 2022/050 22:02:55.700854 134.122.33.221     147.182.145.78     TCP       56 63643 → 35788 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
6805 2022/050 22:02:56.002052 62.173.142.93      134.122.33.221     TCP       76 55554 → 7874 [SYN] Seq=0 Win=14600 Len=0 MSS=1460 SACK_PERM TSval=3313522872 TSecr=0 WS=128
6806 2022/050 22:02:56.002096 134.122.33.221     62.173.142.93      TCP       56 7874 → 55554 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
6807 2022/050 22:02:56.362956 162.142.125.243    134.122.33.221     TCP       60 17751 → 6863 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
6808 2022/050 22:02:56.363001 134.122.33.221     162.142.125.243    TCP       56 6863 → 17751 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
```

Image 12 - Wireshark Packet Capture of Port 4444

Further analysis confirmed the attacker accessed and extracted the entire contents of the customers table, which included: ***full names, phone numbers, email addresses, physical mailing addresses, unique customer ID numbers, and amount spent.*** This personally identifiable information (PII) was successfully removed from PHL's environment during the exfiltration phase. The incident represents a critical breach of sensitive client data and highlights several systemic weaknesses, including insufficient segmentation, lack of upload filtering, and missing detection controls.

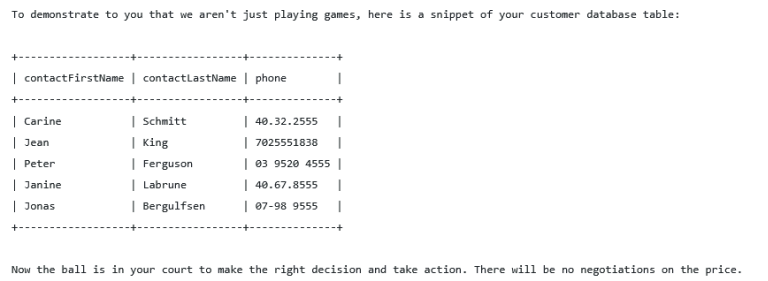


```
CREATE TABLE `customers` (
  `customerNumber` int(11) NOT NULL,
  `customerName` varchar(50) NOT NULL,
  `customerId` varchar(50) DEFAULT NULL,
  `contactLastName` varchar(50) NOT NULL,
  `contactFirstName` varchar(50) NOT NULL,
  `phone` varchar(50) NOT NULL,
  `addressLine1` varchar(50) NOT NULL,
  `addressLine2` varchar(50) DEFAULT NULL,
  `city` varchar(50) NOT NULL,
  `state` varchar(50) DEFAULT NULL,
  `postalCode` varchar(15) DEFAULT NULL,
  `country` varchar(50) NOT NULL,
  `amount_spent` varchar(50) NOT NULL,
  PRIMARY KEY (`customerNumber`)
);

insert  into `customers` (`customerNumber`,`customerName`,`contactLastName`,`co

(103,'Atelier graphique','Schmitt','Carine ','40.32.2555','54, rue Royale',NULL

(112,'Signal Gift Stores','King','Jean','7025551838','8489 Strong St.',NULL,'La

(114,'Australian Collectors, Co.','Ferguson','Peter','03 9520 4555','636 St Kil

(119,'La Rochelle Gifts','Labrune','Janine ','40.67.8555','67, rue des Cinquant
```

Image 13 - Example of the contents that was stolen

Following the exfiltration of sensitive client data, Premium House Lights received a direct extortion email from the threat actor, identifying themselves as "The 4C484C Group." The message was sent from the email address 4C484C@qq.com and addressed to support@premiumhouselights.com. The attacker claimed to be in possession of the company's customer database and demanded a ransom of 10 Bitcoin (BTC) to prevent the public release of the data. The ransom was to be sent to the following wallet address: 1JQqFLmAp5DQJbdD3ThgEiJGSmX8eaaBid.

The email specified a payment deadline of Monday at 10:00 AM UTC, threatening to publish the data on https://pastebin.com if the demand was not met. To validate their claim, the attacker included a snippet of the stolen customer database, displaying actual customer names and phone numbers — matching the records found in the compromised customers table as well as excerpt from the email (*see Image 7*). This communication not only confirms the authenticity of the breach, but also elevates the incident to a ransom-driven data extortion case. The attacker demonstrated access to the exact dataset verified in internal logs, and set a clear financial threat timeline, introducing urgent risk and reputational exposure to the organization.



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 price.

Image 14: Excerpt from Email
Excerpt from email with the demands and customer info.

Insight into How Systems Were Accessed

The attacker exploited a publicly accessible upload folder on the company's website — a misconfiguration that allowed unrestricted file uploads. By placing a web shell in this directory, the attacker gained remote access to the server, confirmed by an HTTP 200 OK response.

From there, they moved laterally to the internal database server, which was reachable due to the lack of network segmentation. No firewall or access controls were in place to isolate the WebServer from more sensitive systems. Once on the database server, the attacker used root-level credentials to export sensitive customer data via mysqldump, then exfiltrated the data to an external server.

The absence of real-time monitoring, privilege restrictions, and segmentation allowed this attack to proceed undetected — from initial access to data theft — in less than ten minutes.

Weaknesses That Allowed for This Incident to Occur

Several gaps in the system's configuration and security controls made it easy for the attacker to gain access, escalate privileges, and move laterally without detection:

- **Unprotected Upload Folder:**
  - The website had a publicly accessible upload area with no file type or size restrictions
- **Directory Browsing Was Enabled:**
  - The attacker could see a full list of files already present in the folder
- **No Block on Running Uploaded Files:**
  - Uploaded scripts were not sandboxed or blocked from execution
- **No Web Application Firewall (WAF):**
  - There were no security tools in place to detect, flag, or block remote shell activity over HTTP
- **Flat Network Architecture:**
  - Both the web and database servers were located on the same VLAN without segmentation.
  - This made lateral movement trivial and bypassed any need for additional authentication or access layers.
- **Excessive Privileges Granted to Services:**
  - The attacker was able to gain root-level access on both servers,
    - Either from misconfigured permissions or default credentials
- **No Logging or Real-Time Monitoring:**
  - There was no SIEM, intrusion detection system (IDS), or behavior-based monitoring tool in place.
- **No Ransomware or Extortion Response Plan:**
  - The organization was unprepared for the extortion attempt and had no formal policy in place for handling threats involving stolen data and cryptocurrency demands.

Incident Response

Recommended Steps to Contain and Remediate the Incident Appropriately

Following the identification of the intrusion, several immediate and long-term actions were recommended to contain the threat and minimize further exposure:

- Immediate Isolation of the WebServer - **Done**
  - The compromised WebServer needed to be removed immediately from the network
    - *This was done as soon as PHL was made aware of the breach*
    - This helped cut off the attacker's access and stopped any further commands from being run through the uploaded shell file
  - ***Source: NIST SP 800-53*** *- IR-4 (Incident Handling),* ***ISO/IEC 27001*** *- 5.29 (Information Security During Disruption)*
- Resetting Compromised Login Credentials
  - Any login credentials that may have been exposed, especially ones tied to the database or admin accounts need to be reset.
    - This included the root-level database access that the attacker used.
  - ***Source: NIST SP 800-53*** *- IA-5 (Authenticator Management),* ***ISO/IEC 27001*** *- 5.15 (Access Control Policy), 5.18 (Access Rights)*
- Blocking the Attacker's IP Addresses - **Done**
  - The attacker's original IP addresses were flagged: 138.68.92.163, 134.122.33.221, and 178.62.228.28
    - These were blocked by the firewall configuration to prevent any future attempts to reconnect.
  - ***Source: NIST SP 800-53*** *- SC-7* (Boundary Protection), ***ISO/IEC 27001*** *- 8.16 (Monitoring Activities)*
- Scanning for Persistence and Other Threats
  - The affected systems need to be reviewed and scanned for malware, backdoors, or unauthorized scripts left behind
  - Any suspicious files or indicators of persistence will be removed to prevent reinfection
  - ***Source: NIST SP 800-53*** *- SI-3 (Malicious Code Protection), SI-4 (System Monitoring),* ***ISO/IEC 27001*** *- 8.25 (Secure Development Lifecycle)*
- Preserving Digital Evidence
  - Key forensic artifacts collection, including:
    - server logs, network captures, shell command histories, and database query logs
  - Keeping these artifacts secured to support internal investigations and potential legal action
  - ***Source: NIST SP 800-53*** *- IR-5 (Incident Monitoring), AU-12 (Audit Record Generation),* ***ISO/IEC 27001*** *- 5.27(Learning from Information Security Incidents)*
- Escalating the Ransomware Threat
  - The extortion email received from "The 4C484C Group" needs to be documented and escalated to both internal leadership and external partners
  - This will include alerting legal counsel and preparing for coordination with law enforcement and cybercrime authorities
  - ***Source: NIST SP 800-53*** *-IR-6 (Incident Reporting), IR-8 (Incident Response Plan),* ***ISO/IEC 27001*** *- 5.29 (Information Security During Disruption)*
- Notifying the Right People: Legal and Regulatory
  - Based on the nature of the breach and confirmed exposure of customer PII, internal privacy officers and legal teams will be engaged to assess potential reporting obligations under data privacy laws
  - Notifications to regulators and affected individuals will initiated where required
  - ***Source: NIST SP 800-53*** *- PL-2 (System Security and Privacy Policy), IR-4 (Incident Handling),* ***ISO/IEC 27001*** *- 5.30(ICT Readiness for Business Continuity)*

Steps to Contain and Remediate the Incident

After confirming the attack, the following actions are to be taken to eliminate the threat, restore secure operations, and prevent similar breaches in the future:

- Rebuilding the Compromised System
  - Affected systems are rebuilt from clean backups
  - No files from the compromised environment are be reused unless they were verified to be safe
  - All software will be updated to current versions with hardened configurations
  - ***Source: NIST SP 800-53*** *- SI-2 (Flaw Remediation),* ***ISO/IEC 27001*** *- 8.27(Secure System Architecture and Engineering Principles)*
- Securing the Upload Folder
  - Restrict access The /uploads/ directory
  - Directory listing need to be disabled, executable files blocked, and only safe file types (e.g., images, PDFs) will be allowed
  - All uploads will need to be automatically scanned for threats.
  - ***Source: NIST SP 800-53*** *- SI-10(Information Input Validation),* ***ISO/IEC 27001*** *- 8.26(Application Security Requirements), 8.32(Change Management)*
- Tightening Account and Access Settings
  - Privileges across the environment needs to be reviewed, and permissions reduced
  - The database server only accepts local connections, and administrative access is limited and monitored.
  - Default or unnecessary root-level access will be eliminated.
  - ***Source: NIST SP 800-53*** *- AC-2(Account Management), AC-6 (Least Privilege),* ***ISO/IEC 27001*** *- 5.15 (Access Control Policy), 5.18 (Access Rights), 8.3(Information Access Restriction)*
- Installing a Web Security Filter / Web Application Firewall (WAF)
  - A security filter needs to be installed to inspect web traffic in real time
  - Detection of anomalies like shell uploads or SQL injection attempts will be blocked before reaching the server
  - ***NIST SP 800-53*** SC-7 (Boundary Protection), *SI-4 (System Monitoring),* ***ISO/IEC 27001*** *- 8.16 (Monitoring Activities)*
- Improving Logging and Monitoring
  - Centralized logging and alerting need to be implemented to detect unusual activity
    - including: unexpected uploads, large data transfers, or login attempts
  - ***Source: NIST SP 800-53*** *- SI-4(System Monitoring), AU-6(Audit Record Review, Analysis, and Reporting),* ***ISO/IEC 27001*** *- 8.15 (Logging), 8.16 (Monitoring Activities)*
- Preparing for Extortion and Ransomware Threats
  - A formal Ransomware Response Plan needs to be introduced
    - includes procedures for handling digital extortion emails, cryptocurrency wallet tracing, law enforcement contact, and secure communication protocols during a live threat scenario
  - ***Source: NIST SP 800-53*** *- IR-4 (Incident Handling), IR-8 (Incident Response Plan),* ***ISO/IEC 27001*** *- 5.29 (Information Security During Disruption), 5.30(ICT Readiness for Business Continuity)*

Post-Incident Recommendations

After reviewing how this incident occurred, several key areas were identified where the company can strengthen its systems and reduce the risk of future attacks. These recommendations are focused on improving security, limiting unnecessary access, and making it easier to detect suspicious activity before it becomes a serious issue.

Protecting Against Similar Attacks in the Future

- **Secure File Uploads on the Website**
  - Make sure the website only allows safe file types (like images or PDFs) to be uploaded
    - Dangerous files — such as scripts or programs — should be blocked automatically
  - Uploaded files should be scanned for viruses and stored in a safer location that prevents them from being run or accessed directly
  - ***Source: NIST SP 800-53*** *- SI-10(Information Input Validation),* ***ISO/IEC 27001*** *- 8.32(Change Management), 8.27(Secure System Architecture and Engineering Principles), PIPEDA - Safeguard*
- **Hide File Folders from Public View**
  - Disable the ability for outsiders to view folders like the one used for uploads
  - If no main page is present, the system should show a "restricted access" message instead of listing the folder's contents
  - ***Source: NIST SP 800-53*** *- AC-3(Account Enforcement), AC-4(Information Flow Enforcement),* ***ISO/IEC 27001*** *- 8.26(Application Security Requirements), 8.27(Secure System Architecture and Engineering Principles), PIPEDA - Safeguard*
- **Apply the Principle of Least Privilege**
  - Everyone — whether it's a person or a system process — should only have access to the files and tools they absolutely need to do their job
    - This reduces the chances of someone misusing or accidentally exposing sensitive information.
    - For example, the WebServer should not have full control over the database unless it is absolutely necessary.
  - ***Source: NIST SP 800-53*** *- AC-2(Account Management), AC-6 (Least Privilege),* ***ISO/IEC 27001*** *- 5.18 (Access Rights), 8.3(Information Access Restriction), PIPEDA - Safeguard*
- **Limit Access to What's Needed**
  - Review all staff and system accounts to make sure they only have access to what they need, nothing more
  - Remove administrative access (like full system control) from anyone who doesn't absolutely require it to do their job
  - ***Source: NIST SP 800-53*** *- AC-2(Account Management), AC-5(Separation of Duties), AC-6 (Least Privilege),* ***ISO/IEC 27001*** *- 5.15 (Access Control Policy) 5.18 (Access Rights), 8.3(Information Access Restriction), PIPEDA - Safeguard*
- **Separate Critical Systems**
  - Keep different parts of the system (such as the website and database) isolated from one another
  - If one part is compromised, it doesn't automatically give access to everything else.
  - ***Source: NIST SP 800-53*** *- SC-7(21)(Boundary Protection:Isolation of System Components), SC-32(System Partitioning),* ***ISO/IEC 27001*** *- 8.22(Segregation of Networks),8.27(Secure System Architecture and Engineering Principles), PIPEDA - Safeguard*
- **Install Security Monitoring Tools**
  - Set up security tools that can detect unusual activity
    - such as someone trying to upload a harmful file or move data out of the system — and send alerts when these things happen.
  - ***Source: NIST SP 800-53*** *- SI-4(System Monitoring), AU-6(Audit Record Review, Analysis, and Reporting), IR-5(Incident Monitoring),* ***ISO/IEC 27001*** *- 5.29 (Information Security During Disruption), 8.16 (Monitoring Activities), PIPEDA - Safeguard*
- **Keep an Eye on Outbound Data**
  - Monitor when large amounts of data are being sent out of the network or when unknown servers are being contacted.
    - These signs can indicate when sensitive data may be leaving the system without permission.
  - ***Source: NIST SP 800-53*** *- SC-7(11)(Boundary Protection: Restrict Incoming Communication Traffic), SI-5(1)(Security Alerts, Advisories, and Directive: Automated Alerts and Advisories), AC-4(Information Flow Enforcement),* ***ISO/IEC 27001*** *- 8.16 (Monitoring Activities), 5.29 (Information Security During Disruption), PIPEDA - Safeguard*

- **Establish a Ransomware and Extortion Response Plan**
  - Develop and document a formal response process for handling digital extortion attempts and ransomware threats
  - Include decision tree for ransom response, law enforcement involvement, internal escalation paths, and public communications
  - Designate responsible personnel and legal points of contact ahead of time
  - ***Source: NIST SP 800-53*** *- IR-4 (Incident Handling), IR-8 (Incident Response Plan)* ***ISO/IEC 27001*** *- 5.29 (Information Security During Disruption), 6.1.3(Actions to Address Risks & Opportunities), PIPEDA - Safeguard*

- **Strengthen Email Threat Detection and Filtering**
  - Enhance email security system to detect and quarantine extortion threats, phishing attempts, and email from suspicious domains
  - Use threat intelligence feeds to block known attacker email addresses and domains. For example: gg.com in this case
  - ***Source: NIST SP 800-53*** *- SI-8(Spam Protection), SC-7* (Boundary Protection)*, IR-5 (Incident Monitoring),* ***ISO/IEC 27001*** *- 8.16 (Monitoring Activities), PIPEDA - Safeguard*
- **Review Data Retention and Encryption Practices**
  - Evaluate how customer data is stored, including whether it is encrypted at rest and in transit
  - Ensure that any database backup, exports, or log containing PII are also encrypted and access is restricted
  - Regularly purge unnecessary customer data to limit what's exposed in the event of a breach
  - ***Source: NIST SP 800-53*** *- SC-12 (Cryptographic Key Establishment and Management), SC-28 (Protection of Information at Rest), MP-6 (Media Sanitization) ,* ***ISO/IEC 27001*** *- 8.10(Information Deletion), 8.25(Secure Development Lifecycle), PIPEDA - Limiting Use, Disclosure, and Retention*

Adjustments to Security Policy

- **Update the Rules for File Handling on the Website**
  - Create clear rules for what kinds of files are allowed to be uploaded, and make sure any custom scripts that handle uploads are reviewed for security. Include web security filters as part of any future website updates.
  - ***NIST SP 800-53*** *- SI-10(Information Input Validation), SA-11(3)(Developer Testing and Evaluation: Independent Verification of Assessment Plans and Evidence),* ***ISO/IEC 27001*** *- 8.32(Change Management), 8.27(Secure System Architecture and Engineering Principles)*
- **Strengthen How Access is Managed**
  - Replace traditional passwords with stronger authentication methods where possible. Require periodic password changes and limit how administrator-level accounts are used.
  - ***NIST SP 800-53*** *- IA-2(Identification and Authentication(Organizational Users)), IA-5(18)(Authenticator Management: Password Managers), AC-2(Account Management),* ***ISO/IEC 27001*** *- 5.15 (Access Control Policy) 5.18 (Access Rights), 8.3(Information Access Restriction), 8.4(Access to Source Code)*
- **Improve How Logs Are Collected and Reviewed**
  - Keep system and access logs for at least 90 days, and make sure they are automatically collected and reviewed regularly. Set up alerts for specific warning signs — like uploads to suspicious paths or attempts to copy sensitive data
  - ***NIST SP 800-53*** *- AU-6(Audit Record Review, Analysis, and Reporting), AU-12(Audit Record Generation), SI-4(System Monitoring),* ***ISO/IEC 27001*** *- 8.15(Logging), 8.16 (Monitoring Activities), 5.29 (Information Security During Disruption)*
- **Invest in Staff Security Awareness**
  - Train staff and technical teams regularly on safe practices — including how to recognize suspicious behavior, avoid misconfigurations, and follow secure procedures when working with sensitive systems.
  - ***NIST SP 800-53*** *- AT-2(Literacy Training and Awareness), AT-3(Role-Based Training)* ***ISO/IEC 27001*** *- 6.3(Security Awareness, Education, and Training)*
- **Have a Formal Response Plan in Place**
  - Maintain a clear, step-by-step incident response plan that outlines who does what during a security breach. This plan should cover investigation, containment, recovery, and any communication that needs to go out to stakeholders or regulators. It should also be tested regularly so the team is ready if something happens again.
  - ***NIST SP 800-53*** *- IR-1 to IR-8(Coverage of IR-1: Policy and Procedures, IR-2: Incident Response Training, IR-3: Incident Response Testing, IR-4: Incident Handling, IR-5: Incident Monitoring, IR-6: Incident Reporting, IR-7: Incident Response Assistance, IR-8: Incident Response Plan), CP-2(Contingency Plan),* ***ISO/IEC 27001*** *- 5.27(Learning from Information Security Incidents), 5.29 (Information Security During Disruption), 5.30(ICT Readiness for Business Continuity)*

Appendix - Supporting Material and Evidence

The following appendix includes technical evidence, artifacts, and reference materials that support the findings and timeline outlined in this incident report. These items provide context for the attacker's activities, data exfiltration, and the recommended response actions.

- **Web Access Logs & Directory Probing**
  - Image1 - 404 Error (Directory Not Found)
    - Timestamp: 21:58:22 - WebServer Data Logs
  - Image2 - 404 Packet View in Wireshark
    - Timestamp: 21:58:22
  - Image3 - 200 OK Confirmation from Upload Directory
    - Timestamp: 21:58:40
  - Image4 - Wireshark Confirmation of Directory Access (Line 739)
    - Timestamp: 21:58:40
- **Shell Upload & Execution**
  - Image3 - WebServer Logs: Shell Upload to /upload/shell.php
    - Timestamp: 21:59:04
  - Image5 - Wireshark View: Shell Upload Triggered
    - Timestamp: 21:59:04
- **Internal Reconnaissance & Lateral Movement**
  - Image 6 - Wireshark Capture: ARP Probing for Internal Hosts
    - Timestamp: 21:59:45
  - Image7 - Wireshark Capture: Port 3306 Connection Initiated (WebServer to DB)
    - Timestamp: 21:59:47
  - Image8 - Wireshark Capture: Port 3306 Connection Accepted (Confirmed from DB)
    - Timestamp: 21:59:47
- **Database Exfiltration**
  - Image10 - mysqldump Execution Log & Data Exfil Command
    - Timestamp: 22:00:27 to 22:02:38
  - Image11 - Wireshark: Client Disconnect / Final Exfil Data
    - Timestamp: 22:02:26
  - Image13 - Partial Screenshot of Exported Customer Table (PII-Adjusted)
- **Persistence & Backdoor Attempts**
  - Image9 - Telnet Session from DB to WebServer
    - Timestamp: 21:59:55
  - Image12 - Ongoing WebServer Traffic on Port 4444
    - Timestamp: 22:02:00 – 22:03:00
- Threat Email Evidence
  -
- **Database Activity**:
  - Image14
    - Sender: 4C484C@qq.com
    - Pastebin Threat + BTC Wallet: *1JqqFL...*

Citations:

ISO/IEC -27001:2022 References:

Admin. (2025, February 3). ISO 27001:2022 Annex A 5.15 - Access Control. ISMS.online. https://www.isms.online/iso-27001/annex-a/5-15-access-control-2022/

Admin. (2025b, February 10). ISO 27002:2022 - Control 5.18 - Access Rights. ISMS.online. https://www.isms.online/iso-27002/control-5-18-access-rights/

Admin. (2025c, February 11). ISO 27002:2022 - Control 5.27 - Learning from Information Security Incidents. ISMS.online. https://www.isms.online/iso-27002/control-5-27-learning-from-information-security-incidents/

Admin. (2025d, February 11). ISO 27002:2022 - Control 5.29 - Information Security During Disruption. ISMS.online. https://www.isms.online/iso-27002/control-5-29-information-security-during-disruption/

Admin. (2025b, February 3). ISO 27001:2022 Annex A 5.30 - ICT Readiness for Business Continuity. ISMS.online. https://www.isms.online/iso-27001/annex-a/5-30-readiness-for-business-continuity-2022/

Admin. (2023, December 14). ISO 27001 Requirement 6.1 – Actions to address risks & opportunities. ISMS.online. https://www.isms.online/iso-27001/actions-to-address-risks-opportunities/

Admin. (2025a, January 29). ISO 27001:2022 Annex A 6.3 – Information Security Awareness, Education, and Training. ISMS.online. https://www.isms.online/iso-27001/annex-a/6-3-information-security-awareness-education-training-2022/

Admin. (2025g, February 17). ISO 27002:2022 - Control 8.10 - Information Deletion. ISMS.online. https://www.isms.online/iso-27002/control-8-10-information-deletion/

Admin. (2025m, February 17). ISO 27002:2022 - Control 8.15 - Logging. ISMS.online. https://www.isms.online/iso-27002/control-8-15-logging/

Admin. (2025b, January 30). ISO 27001:2022 Annex A 8.16 – Monitoring Activities. ISMS.online. https://www.isms.online/iso-27001/annex-a/8-16-monitoring-activities-2022/

Admin. (2025b, January 31). ISO 27001:2022 Annex A 8.22 – Segregation of Networks. ISMS.online. https://www.isms.online/iso-27001/annex-a/8-22-segregation-of-networks-2022/

Admin. (2025i, February 17). ISO 27002:2022 - Control 8.25 - Secure Development Life Cycle. ISMS.online. https://www.isms.online/iso-27002/control-8-25-secure-development-life-cycle/

Admin. (2025c, January 31). ISO 27001:2022 Annex A 8.26 – Application Security Requirements. ISMS.online. https://www.isms.online/iso-27001/annex-a/8-26-application-security-requirements-2022/

Admin. (2025d, January 31). ISO 27001:2022 Annex A 8.27 – Secure System Architecture and Engineering Principles. ISMS.online. https://www.isms.online/iso-27001/annex-a/8-27-secure-system-architecture-engineering-principles-2022/

Admin. (2025b, January 31). ISO 27001:2022 Annex A 8.3 – Information Access Restriction. ISMS.online. https://www.isms.online/iso-27001/annex-a/8-3-information-access-restriction-2022/\

Admin. (2025h, February 3). ISO 27001:2022 Annex A 8.32 – Change Management. ISMS.online. https://www.isms.online/iso-27001/annex-a/8-32-change-management-2022/

NIST SP 800-53:

JOINT TASK FORCE, Ross, W. L., Jr., & Copan, W. (2020). NIST Special Publication 800-53 Revision 5 Security and

Privacy Controls for information systems and organizations. U.S. Department of Commerce. https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-53r5.pdf

PIPEDA

Office of the Privacy Commissioner of Canada. (2020, August 13). PIPEDA Fair Information Principle 5 – Limiting use, disclosure, and Retention. https://www.priv.gc.ca/en/privacy-topics/privacy-laws-in-canada/the-personal-information-protection-and-electronic-documents-act-pipeda/p_principle/principles/p_use/

Office of the Privacy Commissioner of Canada. (2021, August 13). PIPEDA Fair Information Principle 7 – Safeguards. https://www.priv.gc.ca/en/privacy-topics/privacy-laws-in-canada/the-personal-information-protection-and-electronic-documents-act-pipeda/p_principle/principles/p_safeguards/