Penetration Test Report

Client: VulnLawyers

Test conducted by: Baby Goat Pentesters
Test Period: July 15 - July 27, 2025

Report version: 1.0
Classification: Confidential

Confidentiality Statement

This document contains confidential and security-sensitive information resulting from a penetration test conducted for VulnLawyers. It is intended solely for the use of authorized personnel within the organization.

No part of this document may be disclosed, reproduced, or distributed in any form to third parties without the prior written consent of VulnLawyers.

Disclaimer

This report reflects the results of a penetration test performed within a specific scope and time period. It does not guarantee full security, and any changes made after the test are not covered.

Baby Goat Pentesters are not responsible for any issues, damages, or losses resulting from the use of this report.

The client is solely responsible for implementing the recommendations and maintaining the ongoing security of their systems.

Table of contents

Contact Information

Name	Title	Email Address
Baby Goat Pentesters		
Lone	Lead Penetration Tester	l@fake-email.com
VulnLawyers		

Version History

Date	Version	Notes	Author
July 15, 2025	0.1	Draft report	Lone
July 25, 2025	0.2	Import pentest findings	Lone
July 27, 2025	1.0	Delivered/Final Report	

Executive Summary

Between July 15 and July 28, 2025, Baby Goat Pentesters conducted an external penetration test targeting VulnLawyers public-facing infrastructure. The goal was to assess the exposure to real-world attacks and provide actionable insights to strengthen the overall security posture.

The test revealed several high-risk vulnerabilities. These findings demonstrate potential for full system compromise if exploited by a threat actor.

The report outlines each issue, its risk, and specific remediation steps to address them.

Risk summary

Severity	# of Findings
🔴 Critical	0
🟠 High	2
🟡 Medium	1
🟢 Low	1
⚪ Informational	0

Testing summary

Methodology

The test followed a structured four-phase approach:

1. **Planning** – Scope and rules of engagement were defined by VulnLawyer.
2. **Discovery** – Enumeration, scanning, and vulnerability analysis.
3. **Attack** – Exploitation of discovered flaws and privilege escalation.
4. **Reporting** – Documentation of all findings and recommendations.

Tools used

- Testing was performed using tools such as **Dirsearch**, **Vhosts** and **Caido**.

Key observations

Strengths

- **Use of HTTPS:** All tested web applications enforced HTTPS, reducing exposure to man-in-the-middle attacks.

- **No Unauthenticated Administrative Interfaces:** No critical admin panels were accessible without authentication.

- **No SQL/NoSQL Injection Vulnerabilities:**The application demonstrated robust input validation and proper sanitization, indicating effective protection against these common injection attacks.

Weaknesses

- **Insecure Direct Object Reference (IDOR):** Authenticated users were able to access data belonging to other users by modifying object identifiers in the request (e.g., user IDs). This could lead to unauthorized data access.

- **Information Disclosure:** Multiple endpoints exposed sensitive user data such as names and email addresses without adequate access controls. Although some endpoints required authentication, access to other users' data was not restricted.

- **Server-Side Request Forgery (SSRF):** A critical endpoint was found vulnerable to SSRF, allowing internal network access via manipulated request parameters.

- **Brute-force Weaknesses:** User accounts were susceptible to password brute-forcing due to predictable usernames discovered via exposed data and a lack of rate limiting.

- **Application Mapping via Verbose Error Messages:** Certain endpoints revealed details about 'hidden' pages, which could aid in future exploitation.

Recommendations

Based on the vulnerabilities discovered during testing, the following recommendations are provided to improve the overall security posture of the application and its supporting services:

1. Implement Proper Access Control

- Enforce strict access control on all user-related endpoints. Ensure that authenticated users can only access their own data.
- Apply object-level authorization checks in all API endpoints (e.g., user ID verification against session or token).

2. Remove Sensitive Information from Public Endpoints

- Review and restrict access to endpoints such as `/data/users` that expose user details like names and email addresses.
- Avoid including sensitive fields such as passwords or flags in API responses, even in internal environments.

3. Prevent Insecure Direct Object References (IDOR)

- Replace predictable identifiers (e.g., sequential user IDs) with non-guessable alternatives like UUIDs.
- Use authorization logic to ensure only the owner of a resource can access or modify it.

4. Harden SSRF-Prone Endpoints

- Validate and sanitize all user-supplied URLs or parameters that could be used for outbound requests.
- Consider implementing a network-level firewall or metadata protection (e.g., blocking internal IP ranges).

5. Enforce Rate Limiting and Brute-Force Protection

- Introduce rate limiting and account lockout mechanisms to mitigate brute-force attempts.
- Monitor login attempts and alert on suspicious patterns.

6. Disable Verbose Error Messages

- Remove or sanitize any error messages or debug information that may reveal server structure, endpoint existence, or other internal implementation details.

Prioritization

- **High priority:** Fix IDOR, SSRF, and information disclosure affecting user data.
- **Medium priority:** Address brute-force weaknesses and verbose error messages.
- **Ongoing:** Improve access control architecture and apply secure development best practices.

Scope

- betelgeuse.ctfio.com
- Any discovered subdomains

Out-of-scope

- No 'Denial of Service' attacks were performed.

Client Allowances

- VulnLawyers did not provide any credentials or other testing assistance.

Findings

This section provides a detailed overview of all vulnerabilities and security weaknesses identified during the penetration test of VulnLawyers' infrastructure. Each finding includes a severity rating based on its potential impact and exploitability, a Common Vulnerability Scoring System (CVSS) score, the affected endpoint(s), the type of vulnerability, and a Proof of Concept (POC) demonstrating the vulnerability.

The findings are prioritized to assist VulnLawyers in efficiently addressing the most critical risks first, ensuring a structured approach to remediation efforts.

Finding Severity Ratings

The following table defines levels of severity and corresponding CVSS score range that are used throughout the document to assess vulnerability and risk impact.

Severity	CVSS	Definition
Critical	9.0-10.0	Exploitation is straightforward and usually results in system-level compromise. It is advised to form a plan of action and patch immediately.
High	7.0-8.9	Exploitation is more difficult but could cause elevated privileges and potentially a loss of data or downtime. It is advised to form a plan of action and patch as soon as possible.
Medium	4.0-6.9	Vulnerabilities exist but are not exploitable or require extra steps such as social engineering. It is advised to form a plan of action and patch after high-priority issues have been resolved.
Low	0.1-3.9	Vulnerabilities are non-exploitable but would reduce an organization's attack surface. It is advised to form a plan of action and patch during the next maintenance window.
Informational	N/A	No vulnerability exists. Additional information is provided regarding items noticed during testing, strong controls, and additional documentation.

Finding 1- Information Disclosure via 'Denied' to Internal Endpoint

Severity: Low
CVSS Score: 5.3 (Medium)
Affected Endpoint: https://betelgeuse.ctfio.com/login
Vulnerability Type: Information Disclosure / Application Mapping
Note: Although the technical CVSS score is 5.3, this finding is considered 'Low severity' in context, as no sensitive data or functionality was exposed.
Proof of Concept (POC):

Finding 2 - Server-Side Request Forgery (SSRF) on /lawyers-only
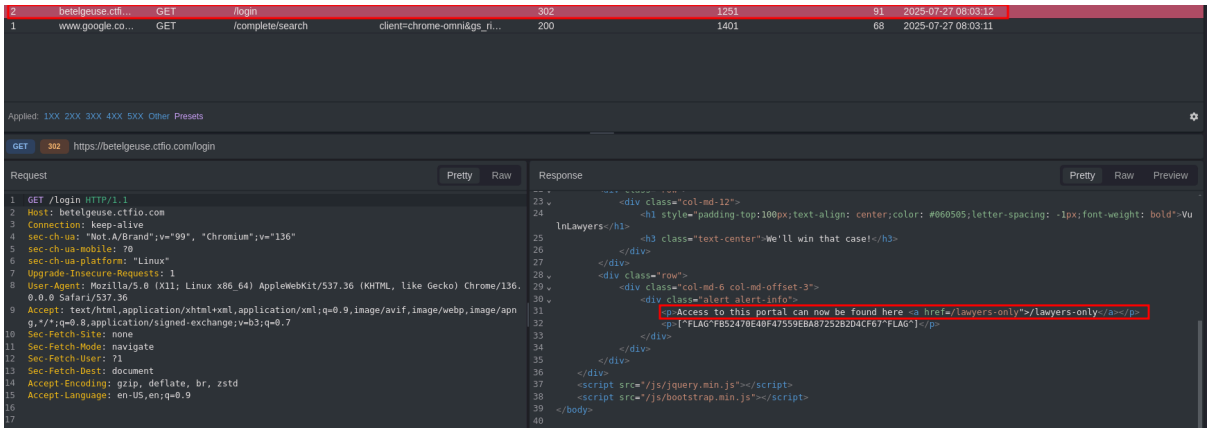
Severity: High
CVSS Score: 7.1 (High)
Affected Endpoint: https://betelgeuse.ctfio.com/lawyers-only
Vulnerability Type: Server-Side Request Forgery (SSRF)
Proof of Concept (POC):

Additional information on SSRF:
https://owasp.org/www-community/attacks/Server_Side_Request_Forgery
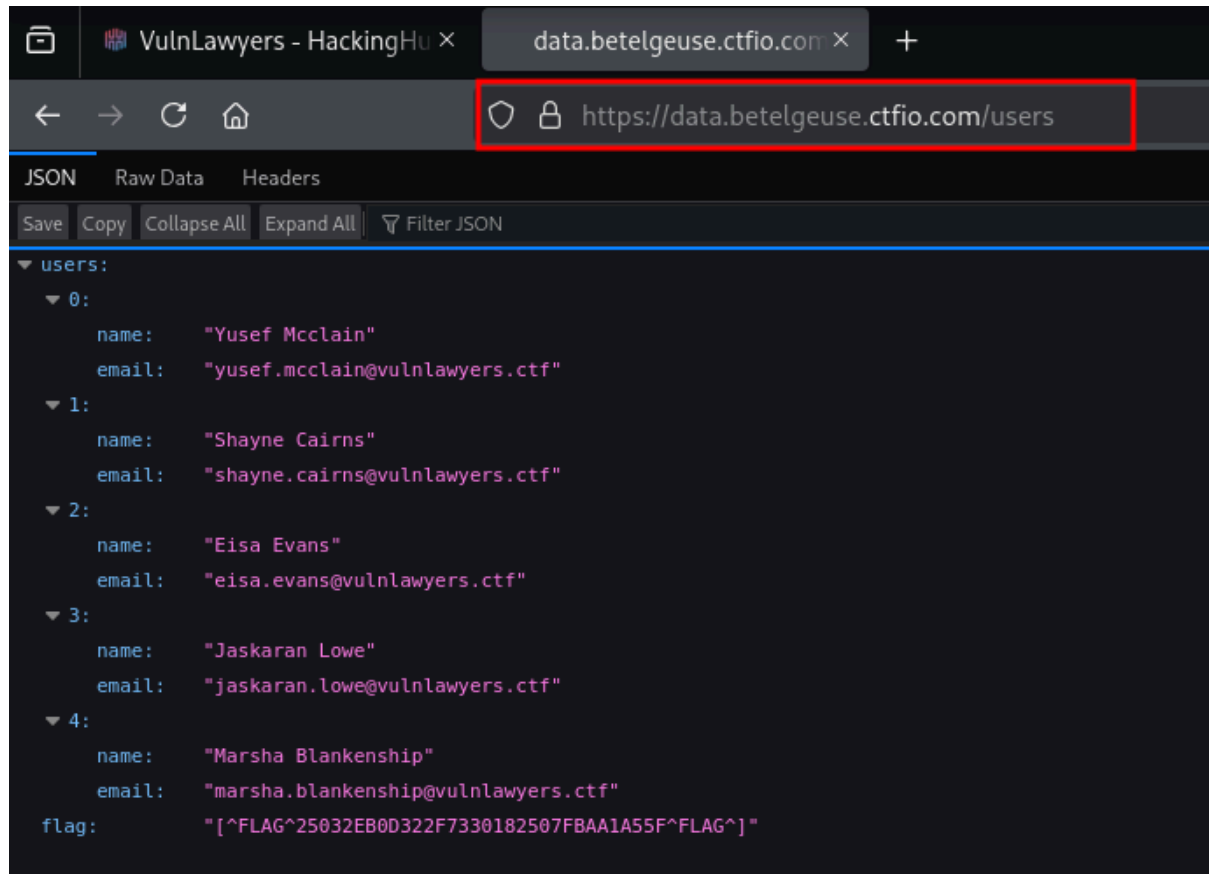
Finding 3 - Information Disclosure on /data/users

Severity: Medium
CVSS Score: 6.5 (Medium)
Affected Endpoint: http://data.betelgeuse.ctfio.com/users
Vulnerability Type: Information Disclosure
Proof of Concept (POC):

Finding 4 - Insecure Direct Object Reference (IDOR)

Severity: High
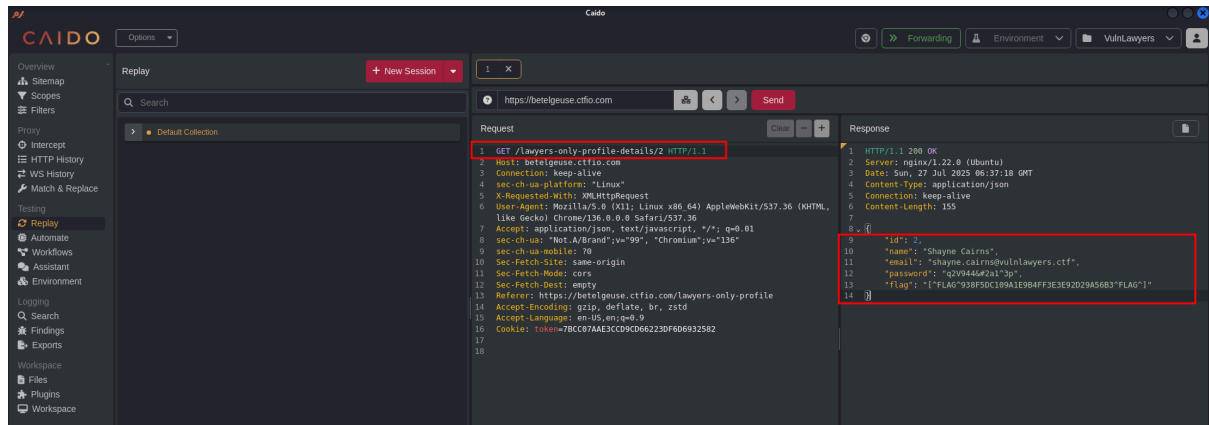**CVSS Score:** 8.8 (High)
Affected Endpoint: http://data.betelgeuse.ctfio.com/users
Vulnerability Type: Insecure Direct Object Reference (IDOR) / Broken Access Control
Proof of Concept (POC):

Additional information on IDOR:

https://owasp.org/www-project-web-security-testing-guide/latest/4-Web_Application_Security
_Testing/05-Authorization_Testing/04-Testing_for_Insecure_Direct_Object_References