Overpass — Web Application Penetration Test Report

Document Control

- Document title: Overpass — Web Application Penetration Test Report
- Version: v1.3.1
- Date: 2026-01-23
- Author: PREM (Portfolio)
- Classification: Public (lab environment)

Finding: Authentication Bypass via Unvalidated Session Cookie (CWE-287)\
Risk: Critical (CVSS 9.1)

Finding Metadata

- Finding ID: F01
- Title: Authentication Bypass via Unvalidated Session Cookie
- Severity: Critical
- CVSS v3.1: 9.1 (AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H)
- CWE: CWE-287 (Improper Authentication), CWE-306 (Missing Authentication for Critical Function)
- OWASP Top 10: A07:2021 — Identification & Authentication Failures
- Status: Confirmed (reproducible)
- Affected endpoints: /admin/, /secrets, /logout

> Testing was performed in a controlled lab environment.\
> No production systems or real user data were accessed.

Target Identification

- Target name: Overpass (TryHackMe lab)
- Assessment type: Web Application Penetration Test (training environment)
- Target host/IP: Lab instance (redacted)
- Environment: Isolated VM

1. Executive Summary

The application incorrectly treats any session cookie as valid authentication.\
Because tokens are not verified server■side, an attacker can access administrative pages and stored secrets without logging in.

This weakness exposes credentials and configuration data and could enable compromise of other systems that reuse those credentials. The issue is simple to exploit and must be addressed immediately.

Priority: Immediate remediation\
Business Impact: Loss of confidentiality and integrity; potential regulatory exposure\
Likelihood: High — no credentials or specialized tools required

Business Decision Summary

- Risk: Critical --- unauthorized admin access and credential exposure.\
- Cost if ignored: Possible lateral compromise, incident response costs, loss of trust, and potential compliance issues.\
- Recommended action: Replace custom auth or enforce verified server■side sessions immediately, then monitor and retest after deployment.

---

2. Scope & Methodology

Testing followed the OWASP Web Security Testing Guide (WSTG):

- Information gathering and reconnaissance\
- Authentication and session testing\
- Authorization and access■control validation\
- Input and logic testing\
- Verification and retest

Tools: Burp Suite (manual proxy), browser developer tools, raw HTTP requests.

Assumptions:

- Application is running in a lab instance\
- Admin interface is expected to be protected\
- No WAF or reverse■proxy controls were present

2.1 Environment & Testing Dates

Platform: TryHackMe (Overpass lab — simulated environment)

Instance type: Isolated training VM (no production systems)

Tester: Independent security assessment (portfolio)

Testing window: 2025-12-15 (lab executed during dedicated learning session)

Data sensitivity: Non■production, synthetic data only

> These details are included to maintain transparency and mirror real client assessment structure.

2.2 Scope Boundaries & Exclusions

In-scope

- Web application front-end

- Authentication and session behavior

- Authorization and administrative interfaces

- Read■only observation of stored secrets

Out of scope / not performed

- Denial■of■Service testing

- Brute■force or credential■stuffing attacks

- Pivoting into external infrastructure

- Social engineering or phishing simulations

Documenting these constraints prevents misinterpretation of coverage and reflects real■world engagements.

3. Application Overview

Overpass is a self-hosted password-management web application.

Key Assets

- stored secrets and credentials\
- authenticated session state\
- administrative functions\
- application configuration and keys

Authentication Behavior

- custom cookie-based authentication\
- token integrity is not verified server-side\
- authorization decisions depend on client-controlled values

The system assumes cookies are legitimate instead of validating them.

---

4. Threat Model

Authentication and Trust Boundary Flow

High-Level Trust Flow Diagram

Browser → Web Server → Auth Logic (broken) → Secrets DB

This makes it clear that authentication is trusted too early, and downstream components rely on that broken assumption.

Trust boundary: everything beyond the web server implicitly trusts the cookie value.\
Because the cookie is attacker-controlled, authentication fails application-wide.

---

5. Attack Hypotheses & Results

Hypothesis	Result	Evidence
Forged cookie may be accepted	Accepted	session=abc123 authenticates
Server validates sessions server■side	Rejected	Any cookie grants access
Access control tied to cookie presence	Accepted	/admin/ accessible with forged cookie
Logout invalidates session	Rejected	Same cookie works after logout
Sensitive data is isolated	Accepted	Access to stored SSH private key

These hypotheses guided testing rather than being written after the fact.

---

6. Vulnerability Analysis