SecretCouncil - HTB Writeup

Category: Web
Difficulty: Medium
Flag: HTB{l33t_5p34k_m4573r_1337}

Deployment Note (Important for HTB Review Team)
The files included in this submission are not intended to be downloaded or inspected directly by players.
The challenge is designed to be solved exclusively through the deployed instance, as the full source code would allow bypassing the intended exploit chain and identifying alternate solutions that are not part of the challenge experience.
The ZIP package is provided solely for:
- Challenge deployment
- HTB infrastructure testing
- Internal review of logic and security
- Confirmation that the intended chain functions as designed
Players should only interact with the running hosted instance, not the underlying application files.

Description
The Secret Council has convened to discuss the rise of AI manipulation.
Only the worthy may enter their inner circle.

Challenge Overview
SecretCouncil is a chained web exploitation challenge involving role manipulation, client-side trust exploitation, leetspeak decoding, and a string-escape-based XSS. The challenge incorporates intentional misdirection to confuse automated reasoning models while remaining intuitive for human solvers. A fake admin panel and misleading UI elements give AI tools false paths to pursue, matching the challenge theme of AI manipulation.

Author Notes for HTB Review Team
- Several components — the admin panel, leetspeak PIN, and karma logic — are intentionally crafted to mislead AI tools while being solvable by humans.
- The admin panel serves as a honeypot-style dead end: attractive but nonfunctional.
- AI tends to hallucinate logic inside it; human testers instantly recognize the dead end.
- Alternate solves exist but still demonstrate proper security reasoning.

Vulnerability Summary

1. Role Escalation via Registration Manipulation
A flaw allows:
role = form_role or 'elite'
Submitting role= with no value silently grants elite privileges.

2. Leetspeak PIN Challenge
Elite users face a 12-digit code such as 133704137701 mapped via:
1=l, 3=e, 7=t, 0=o, 4=a

3. Client-Side Karma Trust Issue
After solving the PIN: karma = 50
But the site uses localStorage for final privilege checks.
Players can simply do:

localStorage.setItem('karma', '100')

4. Markdown + JS String Escape XSS
User Markdown is embedded inside a JS string:
var userContent = "{{ rendered|safe }}";
Payload:
"; get('flag.txt').then(alert); //

5. Flag Endpoint Protected by Custom Header
Requires:
X-Karma: 100
Valid session
Karma >= 50
Calibration complete

Full Exploitation Walkthrough

1. Register as Elite via Abuse
Send role= with no value to become elite without PIN verification.

2. Solve Leetspeak PIN
Example: 133704137701 → leetolaetool

3. Overwrite Karma
localStorage.setItem('karma', '100');

4. Inject Payload
"; get('flag.txt').then(alert); //

5. Retrieve Flag
Server returns:
HTB{l33t_5p34k_m4573r_1337}

Unintended but Valid Alternate Solve
Using cookies + manual header:
curl -s -b cookies.txt -H "X-Karma: 100" http://localhost:1337/flag.txt

Admin Panel Dead-End Note
The admin panel is intentionally useless. It serves as an AI confusion point and provides no real advantage.

Conclusion
SecretCouncil demonstrates the dangers of trusting client-side values, mismanaging role assignment, and rendering unsafe Markdown. Combined with AI-targeted misdirection, it delivers a unique and thematic challenge experience.

FLAG:
HTB{l33t_5p34k_m4573r_1337}