🔍 ThreatScope Security Analysis Report

Date: 5/16/2025, 8:09:39 PM

📋 SECTION 0: Executive Summary

- Sensitive data (password, email) is handled.
- CSRF protection and strong password policies need attention.

Overall Security Posture: Needs Improvement

Threat Impact: Potential for account takeover and data breaches.

Business Risk: Compromised user accounts, data leakage, reputational damage.

Suggested Next Action: Security re-architecture with focus on authentication, authorization and input validation.

📝 SECTION 1: Summarized Screen Overview

Attribute	Value
Form Action URL	/signup
HTTP Method	POST
Input Fields	email (type: email), password (type: password), confirm_password (type: password), tos_agreement (type: checkbox), signup_token (type: hidden)
Hidden Fields	signup_token

🔐 SECTION 2: Security Design Analysis

- HTTPS is assumed to be enabled, which is good.
- Proper HTTP method (POST) is used for form submission.
- Password fields should have `autocomplete="off"` or `autocomplete="new-password"` .
- Confirm password field is present; ensure client and server-side validation match.
- Signup token: Ensure it has high entropy, is unique per session, and is properly validated server-side to prevent replay attacks.

- Client-side validation: Ensure it's in place but is not the only layer of defense. Server-side validation is critical.
- Missing CSRF token protection. This is a significant vulnerability.

🧠 SECTION 3: Threat Modeling (STRIDE-based)

Spoofing

- **Threat:** Attacker spoofs a legitimate user to create an account.
- **Scenario:** Reusing email addresses or bypassing email verification.
- **Likelihood:** Medium
- **Impact:** Medium

Tampering

- **Threat:** Attacker tampers with form data during submission.
- **Scenario:** Modifying the hidden signup_token or bypassing client-side validation.
- **Likelihood:** Medium
- **Impact:** Medium

Repudiation

- **Threat:** User denies signing up.
- **Scenario:** Lack of sufficient audit logging or non-repudiation mechanisms.
- **Likelihood:** Low
- **Impact:** Low

Information Disclosure

- **Threat:** Sensitive information leakage.
- **Scenario:** signup_token is guessable or exposed. Error messages reveal sensitive info.
- **Likelihood:** Medium
- **Impact:** High

Denial of Service

- **Threat:** Attacker floods the signup endpoint.
- **Scenario:** Automated bot signup attempts.
- **Likelihood:** Low
- **Impact:** Low (rate limiting needed)

Elevation of Privilege

- **Threat:** An attacker gains unauthorized privileges.
- **Scenario:** Exploiting vulnerabilities in the signup process to gain admin access.
- **Likelihood:** Low
- **Impact:** High

🛡️ SECTION 4: Actionable Security Recommendations

- Implement CSRF protection using tokens synchronized with the server.
- Add `autocomplete="off"` or `autocomplete="new-password"` to password fields.
- Enforce a strong password policy (length, complexity).
- Implement server-side validation for all inputs, even those validated client-side.
- Use reCAPTCHA or similar mechanism to prevent bot signups.
- Ensure proper error handling to avoid information leakage. Log errors securely.
- Consider implementing an email verification process after signup.
- Implement rate limiting on the signup endpoint to prevent abuse.
- Set secure HTTP headers: `Strict-Transport-Security`, `X-Frame-Options`, `X-Content-Type-Options`, `Content-Security-Policy`.

✅ SECTION 5: Positive Security Observations

- Use of POST method for form submission.
- Presence of a confirm password field.
- Use of hidden signup_token.

📊 SECTION 6: Security Score (4/10)

4/10

Score based on the lack of CSRF protection, weak password policy, and potential information leakage. Mitigation will improve the score.

💉 SECTION 7: Manual Security Checklist for Reviewers

- ☐ Does the form use HTTPS?
- ☐ Are sensitive fields masked or obfuscated?
- ☐ Do hidden fields include secure tokens?
- ☐ Are inputs validated and sanitized client-side?
- ☐ Is CSRF protection implemented and tested?
- ☐ Are secure HTTP headers configured?
- ☐ Can input lead to XSS or data injection?
- ☐ Is data stored in cookies/localStorage secure?

Report generated by ThreatScope Security Extension