Vulnerability Scanning – July 14, 2025

Introduction:

To test and check for vulnerabilities with XSS attacks a manual test was conducted through all API calls in the backend and fields in the frontend. Notes, recommendations and risk levels were all noted from this manual test

Frontend Inputs Tested:

Login Page – Email, Password

- Note: Password features do not actually authenticate
- Recommendations: Secure authorization system again to ensure password protection re-established
- Risk: Very high

Register Page – Name, Email, Password

- Recommendations: N/A
- Risk: Low

Profile Page – Display Name, Email, Storage Preference, RAM Preference, Brand Preference, Budget Minimum, Budget Maximum, Rating Prefence, Country

- Note: Console logs object of profile – Need to remove this in final release
- Recommendations: Remove log statements since customers can see console logs if they open the developer tools
- Risk: High

Chat Page – User input

- Note: Chat bot recognizes and deflects code
- Recommendations: N/A
- Risk: Low

Backend Inputs Tested:

/api/auth - /register, /login

- Note: Password features do not actually authenticate for /login – Always posts success whenever valid password length string used
- Recommendation: Ensure JWT is working with this route
- Risk: Very high

/api/chats – POST /, GET /

- Note: GET / allows anyone to query database
- Recommendation: Protect route behind some auth token system
- Risk: High

/api/profiles - /verify/:token?, /resend-verification, /:userId?, /passwords/:userId?, /:userId?

- Note: PATCH /:userId? Can store data that is not selectable settings for RAM_preference
- Recommendation: Guarding to ensure only the desired inputs can be made via API calls to the backend
- Risk: Moderate

/api/products - /search

- Recommendations: N/A
- Risk: Low

/api/chats – POST, GET

- Note: POST does not sanitize data
- Recommendation: Santize data
- Risk: Moderate

/api/buywise/redirect

- Note: Sanitizes inputs for URLs
- Recommendation: N/A
- Risk: Low