Essential Web Security Concepts Every Developer Should Know

Introduction

Building secure web applications is crucial to protecting user data and privacy. This article will overview core web security concepts developers should understand.

Common Web App Vulnerabilities

- SQL Injection - Malicious SQL queries inserted into entry points.
- Cross-Site Scripting (XSS) - Injecting client-side scripts into pages.
- Cross-Site Request Forgery (CSRF) - Tricking users into unwanted actions.
- Broken Authentication - Flawed auth logic enabling account takeovers.

Encryption and Hashing

- Encryption - Encoding data to protect confidentiality and privacy.
- Hashing - One-way encoding of passwords and data.
- Salting - Random data added to passwords before hashing.

Securing User Authentication

- Use bcrypt, scrypt or PBKDF2 for secure password hashing.
- Implement rate limiting on login attempts.
- Require re-authentication for sensitive actions.
- Use proven auth frameworks like Devise (Rails).

Preventing SQL Injection

- Parameterize queries to separate code and data.
- Validate and sanitize all user inputs.
- Limit account permissions and access.
- Configure firewall rules to block SQLi payloads.

Mitigating XSS Attacks

- Escape/encode untrusted outputs in HTML.
- Set HTTP headers like CSP to whitelist trusted sources.
- Validate, sanitize, and encode user inputs.

Conclusion

Learning fundamental web security concepts will enable developers to build more logic-proof and attack-resistant applications.