Web Design & Programming

Web Security

Xavier Belanger

Security is a process,
not a product.

Bruce Schneier

Security at the System Level

- Apply security updates in a timely manner; check in a test environment before deploying to production systems.
- Limit accesses to the minimum, use strong credentials.
- Compartmentalize: each component (web server, database, application, …) should run on different systems.
- Backup all data, check and validate the restore process.

Security at the Network Level

- Use encryption (TLS), validate the security level on a regular basis.
- Plan for network access redundancy (multiple hosting locations, multiple Internet accesses); use load balancers.
- Plan for protection against malicious traffic (DoS/DDoS).
- Use firewall rules to restrict accesses, use a Web Application Firewall (WAF) for the web traffic.

Security at the Service Level

- Your web server must be properly configured, default configuration is usually not secure.
- Test and check for proper error management, logging, access limits.
- Apply updates for the web server, language and libraries in a timely manner.

Security at the Database Level

- Be in compliance with the laws and reglementations, including specific ones (by industry sector, geography, …) regarding data collection, storage and stewardship.
- Check and sanitize database queries.
- Limit access to only what is needed (use views).
- Validate the data added to the database.
- Backup all data, check and validate the restore process.

Security at the Website Level

- Validate all the code that you are using for compliance (HTML, CSS, PHP, …).
- Use application web scanners to check your entire website:
  - Nikto: https://cirt.net/nikto2
  - OWASP ZAP: https://www.zaproxy.org/
- Maintain your code up-to-date with changes and new versions.

Dependencies Security

- Validate all third party components used in your code.
- Ensure that suppliers, contractors and hosting providers are reliable and meet the requirements that you have set.
- Perform periodical reviews.

OWASP

- **Open Web Application Security Project**
  - https://owasp.org/
- **Top 10 Web Application Security Risks**
  - https://owasp.org/www-project-top-ten/
- **OWASP Cheat Sheet Series**
  - https://cheatsheetseries.owasp.org/index.html

Most Critical Vulnerabilities

- **Injection**: weak or nonexistent input validation, allowing an attacker to execute arbitrary code.
- **Broken Authentication**: user authentication mechanisms allowing weak passwords, brute-force attempts, simple password resets, ...
- **Data Exposure**: files or database not being properly protected

security.txt

- The security.txt file is a standard based text file where information on how to report security issues.
- It contains a way to reach out to a contact person, and additional details such as the preferred languages or an encryption key.
- Not limited to web security issues, this is an entry point for any vulnerability in a product or service for an organization.
- See RFC 9116 for a full description, or https://securitytxt.org/

If you think technology
can solve your security problems,
then you don’t understand the problems
and you don’t understand the technology.

Bruce Schneier

Training and Awareness

- Stay informed about security issues, new offensive and defensive techniques.
- Read books, blogs, articles, …
- Get involved with vendors, open-source community and peers.
- Attend to a training session/workshop/conference on a regular basis (once a year minimum).

- OWASP Logo: OWASP Foundation, Inc.
  https://owasp.org/www-policy/operational/branding