Mindtree
A Larsen & Toubro Group Company

Basics of **web security**

Introduction

The process of securing modern web applications is a comprehensive exercise with various dimensions. Platform security is a quintessential necessity that helps organizations earn and retain customers' trust. **In this whitepaper, we discuss the key aspects of basic web security.**

Welcome to possible

A Mindtree Whitepaper

Tenets of web security

In this section, we discuss the key tenets of web security. We have shed light on the checklists, best practices, and common anti patterns for each of the tenets.

Authentication and Authorization

Authenticating the user is the start of the security journey. Based on the security needs of the web application, we can implement further security measures. Given below are the main best practices in authentication and authorization:

1. Implement stateless and token-based authentication for modern web applications. We can employ Oauth protocols for authentication.

2. Design and implement the account management best policies such as strong password policies, multi-factor authentication and time-based OTPs for sensitive functions, federated identity management, social login (login through Google, Apple, Facebook)

3. Support open standards such as SAML (Security Assertion Markup Language) and OIDC (Open ID connect) for federated authentication.

4. Employ strong cryptographic standards for sensitive data management.

5. Encryt data at rest and during transit. We can encrypt data at rest using strong cryptographic methods such as SHA-256 based encryption and we can use TLS1.2 for transport level security.

6. Audit authentication and authorization activities such as password change attempts, login/logout events and others.

7. Design and implement role-based authorization and method level security.

8. Design the single-sign-on (SSO) for various enterprise applications to provide a seamless user experience.

9. Follow the principle of least privilege and layered security at each solution layer.

Given below are some of the key validations and checks that can be employed for security testing:

1. Ensure that no PII data or sensitive data is captured in the logs

2. Test the application for horizontal escalation scenario (logged in user accessing other users' data) and vertical escalation scenario (logged in user accessing his/her manager's data)

3. Validate the scenarios using SOC2 checklist and OWASP guidelines[1,2]

Session Management

Session management involves efficiently managing the logged-in user’s scenarios. Given below are the best practices in session management:

1. Design stateless sessions to achieve scalability and use token-based authentication and avoid cookies. If cookies cannot be avoided, use HTTPOnly and Samesite flag.

2. Use only TLS v1.2 for transmitting all session data.

3. Provide features for explicit user logout and implicit session logout based on idle timeout (usually 15 mins) and enforce stricter timeout for admin sessions.

4. For sensitive applications, use CSRF tokens with each request to avoid request forgery.

5. Automate the account management process for activities such as account creation, account locking and such.

6. Don’t cache the sensitive data in browser cache and in the user's local storage.

Application security

Modern applications are built on layered architecture that mainly consists of the front end, services and database layers.

Given below are the best practices for application security:

1. Design and implement both server side and client side input data validation. Use centralized, pattern-based validation logic. Sanitize and encode the input to mitigate the injection attacks.

2. Design and implement the error handling logic that does not disclose sensitive server information to the end user.

3. Validate the application for OWASP top 10 vulnerabilities such as Injection Flaws, Broken Authentication and Session Management, Cross-Site Scripting, Insecure Direct Object References, Security Misconfiguration, Sensitive Data Exposure, Missing Function Level Access Control, Cross-Site Request Forgery (CSRF), Components with Known Vulnerabilities and Un-validated Redirects and Forwards.

4. Thoroughly validate all the third party open source libraries used in the application and check for any known vulneralibities.

5. Conduct manual and tool-based security reviews iteratively.

6. Create a threat model based on the application domain and continuously assess the vulnerabilities based on that.

7. Design and implement the security requirements for the application domain (such as HIPAA standards for health-related application, PCI-DSS standards for finance applications)

8. Implement secure headers[3]

9. Follow secure development practices during all phases of SDLC.

10. Avoid usage of local data storage for storing any session information.

11. Continuously refactor the application to remove the deprecated methods.

12. Design a blacklist of request payload values and reject all inputs matching the blacklist patterns.

13. Implement a content security policy (CSP) to mitigate injection attacks.

14. Design and use secure coding guidelines and a checklist.

15. Implement secure encoding, secure data transmission and controls using the best practices[5].

16. Implement anti-automation measures such as CAPTCHA-based validation.

17. Throughly validate the application and the used open source libraries for the presence of any back-door access using code obfuscation.

18. Enforce code and data integrity checks using signing, digests and other mechanisms.

19. Categorize the data based on the sensitive nature of the data and apply the appropriate access policies for the data.

Encryption

We need to encrypt the data at rest and during transit. Given below are the encryption-related best practices:

1. Follow the recommended cryptographic best practices[4]

2. Use SHA-256 for secure hashing of password and sensitive information[4]

Infrastructure

The servers, hardware and network components involved in running the applications fall within the scope of infrastructure. Given below are the security-related best practices for infrastructure:

1. Design and enforce mitigation measures for DDoS (distributed denial of service) using CDN (Content Delivery Network) and out-of-the-box cloud services.
2. Design the disaster recovery (DR) environment to handle security contingencies and test the DR environment thoroughly.
3. Harden the servers and block the unused ports and protocols. Reguarly install the security patches.
4. Use a web application firewall (WAF) to mitigate the common application vulneralibitiles.
5. Validate the user uploaded files and block any OS-level command execution from the application.
6. Develop a real-time security event monitoring and notification setup to respond to any security incidents in real time.
7. Disable directory browsing.
8. Take regular secure data backups in remote regions to enable business continuity.
9. Continuously perform the vulneralibiilty of the servers and monitor the vulneralibility reports
10. Continuously monitor the network using intrusion detection tools to identify any abnormal traffic patterns.

Security tools

Given below are some of the tools used for implementing security

Category	Sample tools
Security Testing	HPE WebInspect, IBM AppScan, PortSwigger Burp, Qualys, Rapid7, Veracode. Iron Wasp
Penetration testing	Burp Suite, ZAP Proxy
Code scanning	SonarQube
Vulneralibitity assessment	Blackduck
Injection Flow testing	SQLMap, Wapiti
Network Traffic Inspection	Wireshark, Google Nogotofail
Web security assessment	W3af
Web Application Firewall	AWS WAF
Intrusion detection systems	AWS GuardDuty
Vulneralibility assessment	AWS Inspector

Conclusion

We have discussed the key tenets of the web security, which include authentication and authorization, session management, application security, encryption and infrastructure security. As part of authentication and authorization, we need to create security policies and access roles, proper authentication and authorization policies, encryption standards and auditing. As part of session management, we need to design the stateless sessions and adopt secure session data management practices. Application security is mainly dependent on the functional domain of the application and the sensitive nature of the handled data. Some of the main application security measures are input validation, proper error handling, secure development practices, secure data storage and others. As part of infrastructure security, it is important to strengthen and continously assess the vulnerability of the servers.

References

1. https://buildfire.com/app-statistics/
2. https://www.fyresite.com/how-many-apps-fail/

About the author

Dr. Shailesh Kumar Shivakumar

Solution Architect

Dr. Shailesh Kumar Shivakumar has 19+ years of experience in a wide spectrum of digital technologies including, enterprise portals, content management systems, lean portals and microservices. Dr. Shailesh holds a PhD degree in computer science and has authored eight technical books published by the world's top academic publishers such as Elsevier Science, Taylor and Franscis, Wiley/IEEE Press and Apress. Dr. Shailesh has authored more than 14 technical white papers, five blogs, twelve textbook chapters for various under-graduate and post graduate programs and has contributed multiple articles. He has published 20+ research papers in reputed international journals. Dr. Shailesh holds two granted US patents, apart from ten patent applications. Dr. Shailesh has presented multiple research papers in international conferences. Dr. Shailesh's Google Knowledge Graph can be accessed at https://g.co/kgs/4YoaiN . He has successfully led several large scale digital engagements for Fortune 500 clients. Shailesh can be reached at Shaileshkumar.Shivakumarasetty@mindtree.com

About Mindtree

Mindtree [NSE: MINDTREE] is a global technology consulting and services company that enables enterprises across industries to drive superior competitive advantage, customer experiences and business outcomes by harnessing digital and cloud technologies. A digital transformation partner to more than 260 of the world's most pioneering enterprises, Mindtree brings extensive domain, technology and consulting expertise to help reimagine business models, accelerate innovation and maximize growth. As a socially and environmentally responsible business, Mindtree is focused on growth as well as sustainability in building long-term stakeholder value. Powered by more than 29,700 talented and entrepreneurial professionals across 24 countries, Mindtree — a Larsen & Toubro Group company — is consistently recognized among the best places to work. For more, please visit www.mindtree.com or @Mindtree_Ltd.