SOHAIB KHAN
2022551
CYBER SECURITY
SSD WEEK : 2

Week 2: Current Risks - Secure Notes Application

Component: Authentication

Use Case: Elevation of Privilege

- CRT1. Threat: Attackers access the system taking advantage of broken authentication
- Risk Levels: Inherent: Critical | Current: Critical | Projected: Critical
- State: Exposed
- CR1. Countermeasure: Implement server-side access control checks (Status: RECOMMENDED)
- CRT2. Threat: Attackers exploit flaws in access control systems
- Risk Levels: Inherent: Critical | Current: Critical | Projected: Critical
- State: Exposed
- CR2. Countermeasure: Implement secure session management (Status: RECOMMENDED)
- CR3. Countermeasure: Implement Multi-Factor Authentication (MFA) (Status: RECOMMENDED)

Use Case: Tampering

- CRT3. Threat: Attackers capitalize on security misconfigurations
- Risk Levels: Inherent: High | Current: High | Projected: High
- State: Exposed
- CR4. Countermeasure: Conduct regular security audits and reviews (Status: RECOMMENDED)

Use Case: Information Disclosure

- CRT4. Threat: Attackers get access to sensitive data
- Risk Levels: Inherent: Critical | Current: Critical | Projected: Critical
- State: Exposed
- CR5. Countermeasure: Use encryption to protect sensitive data (Status: RECOMMENDED)

Component: Backend

Use Case: Tampering

- CRT5. Threat: Attackers exploit misconfiguration and vulnerable third-party dependencies
- Risk Levels: Inherent: High | Current: High | Projected: High
- State: Exposed
- CR6. Countermeasure: Comprehensive configuration hardening and dependency auditing (Status: RECOMMENDED)
- CRT6. Threat: Attackers execute Injection Attacks
- Risk Levels: Inherent: High | Current: High | Projected: High
- State: Exposed
- CR7. Countermeasure: Use parameterized queries and input validation (Status: RECOMMENDED)

Use Case: Spoofing

- CRT7. Threat: Attackers gain elevated privileges or unauthorized access
- Risk Levels: Inherent: High | Current: High | Projected: High
- State: Exposed
- CR8. Countermeasure: Implement multi-layered security for authentication and access control (Status: RECOMMENDED)

Use Case: Information Disclosure

- CRT8. Threat: Attackers gather useful information from inadequate error handling
- Risk Levels: Inherent: High | Current: High | Projected: High
- State: Exposed
- CR9. Countermeasure: Implement generic error messages and proper exception handling (Status: RECOMMENDED)
- CRT9. Threat: Attackers take advantage of insecure communication channels and unprotected sessions
- Risk Levels: Inherent: High | Current: High | Projected: High
- State: Exposed
- CR10. Countermeasure: Implement comprehensive secure communication and session management protocols (Status: RECOMMENDED)
- CRT10. Threat: Attackers take advantage of weak encryption of sensitive data
- Risk Levels: Inherent: High | Current: High | Projected: High
- State: Exposed
- CR11. Countermeasure: Implement and maintain Advanced Encryption Standards (AES) with effective key management (Status: RECOMMENDED)

Component: Database

Use Case: Denial of Service (DoS)

- CRT11. Threat: Attackers perform a DoS attack on the database
- Risk Levels: Inherent: High | Current: High | Projected: High
- State: Exposed

- CR12. Countermeasure: Implement rate limiting and resource allocation (Status: RECOMMENDED)

Use Case: Information Disclosure

- CRT14. Threat: Attackers exfiltrate data due to insecure backup procedures
- Risk Levels: Inherent: Critical | Current: Critical | Projected: Critical
- State: Exposed
- CR15. Countermeasure: Implement secure backup procedures with encryption and access controls (Status: RECOMMENDED)
- CRT15. Threat: Attackers exploit misconfigurations in PostgreSQL settings
- Risk Levels: Inherent: Critical | Current: Critical | Projected: Critical
- State: Exposed
- CR16. Countermeasure: Harden PostgreSQL configuration and restrict network access (Status: RECOMMENDED)
- CRT16. Threat: Attackers exploit SQL Injection vulnerabilities
- Risk Levels: Inherent: High | Current: High | Projected: High
- State: Exposed
- CR17. Countermeasure: Use parameterized queries and validate inputs (Status: RECOMMENDED)

Component: Encryption

Use Case: Tampering

- CRT21. Threat: Attackers exploit misconfigurations and vulnerable third-party libraries
- Risk Levels: Inherent: High | Current: High | Projected: High
- State: Exposed
- CR22. Countermeasure: Perform comprehensive configuration hardening and dependency auditing (Status: RECOMMENDED)

Use Case: Information Disclosure

- CRT25. Threat: Attackers take advantage of insecure communication channels
- Risk Levels: Inherent: High | Current: High | Projected: High
- State: Exposed
- CR26. Countermeasure: Implement TLS encryption for all connections (Status: RECOMMENDED)

Component: Frontend & Web Security

Use Case: Spoofing

- CRT29. Threat: Attackers can deceive users into clicking on hidden elements
- Risk Levels: Inherent: High | Current: High | Projected: High

- State: Exposed
- CR30. Countermeasure: Implement frame-busting scripts, X-Frame-Options, and Content Security Policy (CSP) (Status: RECOMMENDED)

Use Case: Tampering

- CRT35. Threat: Attackers can inject malicious scripts into web pages
- Risk Levels: Inherent: High | Current: High | Projected: High
- State: Exposed
- CR36. Countermeasure: Implement input validation, output encoding, and enforce CSP (Status: RECOMMENDED)

Conclusion

Overall Risk Assessment:

- High-Risk Components: Authentication, Backend, Database, Encryption, Web Security
- Critical Threats Identified: Broken authentication, SQL injection, weak encryption, security misconfigurations
- Key Recommendations:
- Implement strong authentication and authorization controls
- Use encryption for all sensitive data and communication
- Perform regular security audits and penetration testing
- Harden server configurations and restrict unnecessary access

Threat	**Attack Vector**	**Risk Level**	**Mitigation Strategy**
Unauthorized Access	Brute-force login attack	High	Strong passwords & rate limiting
Data Breach	Stolen database or logs	High	Encryption of stored data
CSRF Attacks	Malicious requests via web forms	Medium	Use CSRF tokens
SQL Injection	Injecting malicious SQL in login forms	High	Input validation & prepared statements
Session Hijacking	Stealing user session cookies	High	Secure cookies & HTTPS enforcement