**OWASP Top 10 Vulnerabilities in Pet Store Application**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Risk ID** | **Risk Name** | **Description** | **Example in Pet Store Application** | **Mitigation** |
| A01:2021 | Broken Access Control | Restrictions on user actions aren't properly enforced. | A user without admin privileges can access an endpoint to modify pet information or view other users' orders. | Enforce server-side access checks for roles; deny by default; use secure coding practices |
| A02:2021 | Cryptographic Failures | Sensitive data exposure due to weak or missing encryption. | Customer passwords or payment details sent in plain text over HTTP or stored unencrypted. | Use HTTPS (TLS) for all data transmission; store passwords with proper hashing (e.g., bcrypt) |
| A03:2021 | Injection | Untrusted data sent to an interpreter as part of a command/query. | SQL injection allowing attackers to view or modify pet details by manipulating search queries. | Use parameterized queries; sanitize inputs; ORM frameworks |
| A04:2021 | Insecure Design | Design flaws that lead to security weaknesses. | Lack of input validation allows users to upload malicious files (e.g., scripts) in the pet images section. | Proper validation of file types and sizes; implement secure design principles |
| A05:2021 | Security Misconfiguration | Improperly configured security headers, error messages, cloud storage, etc. | Exposing detailed error messages like database connections or server details that can be exploited by attackers. | Configure error handling and security headers properly (e.g., HTTP security headers) |
| A06:2021 | Vulnerable and Outdated Components | Use of outdated libraries, frameworks, or software versions. | PetStore app running on outdated libraries (e.g., old version of Java or Spring Framework), which have known vulnerabilities. | Regularly update dependencies and apply security patches; use automated tools like Dependabot |
| A07:2021 | Identification and Authentication Failures | Flaws in authentication mechanisms. | The PetStore app allows weak passwords or fails to implement multi-factor authentication for critical operations (e.g., order management). | Enforce strong password policies, implement multi-factor authentication (MFA) |
| A08:2021 | Software and Data Integrity Failures | Failure to verify software or data integrity. | The PetStore app's code base could be tampered with, or unauthorized changes made to product data (e.g., pet prices or availability). | Use code signing, integrity checks, secure update mechanisms |
| A09:2021 | Security Logging and Monitoring Failures | Inadequate logging and alerting hampers attack detection and response. | PetStore app does not log failed login attempts, or does not alert administrators when suspicious activity is detected. | Implement detailed logging, centralized monitoring, and alerting for suspicious behavior |
| A10:2021 | Server-Side Request Forgery (SSRF) | The server is tricked into making unauthorized requests to internal resources. | An attacker causes the PetStore backend to make requests to internal resources (e.g., accessing metadata services of the cloud provider). | Implement server-side validation of outbound requests and disable unnecessary services |