**SECURITY TESTING**

|  |  |  |
| --- | --- | --- |
| **ID** | **Vulnerability Name** | **Description** |
| A01:2021 | **Broken Access Control** | Failures that allow users to act outside their intended permissions (e.g., accessing other users' accounts, modifying data). |
| A02:2021 | **Cryptographic Failures** | Inadequate encryption or use of outdated algorithms leading to data exposure or compromise. Previously called Sensitive Data Exposure. |
| A03:2021 | **Injection** | User input not properly sanitized, allowing attackers to inject code (e.g., SQL, XSS). |
| A04:2021 | **Insecure Design** | Flaws due to insecure architecture/design rather than implementation bugs. Encourages threat modeling and secure design practices. |
| A05:2021 | **Security Misconfiguration** | Default settings, incomplete setups, or open cloud storage that expose systems to attacks. |
| A06:2021 | **Vulnerable and Outdated Components** | Use of software libraries or frameworks with known vulnerabilities that haven’t been updated or patched. |
| A07:2021 | **Identification and Authentication Failures** | Failures in identity verification or session management (e.g., broken authentication mechanisms). |
| A08:2021 | **Software and Data Integrity Failures** | Lack of integrity checks for software updates, critical data, or CI/CD pipelines. Insecure Deserialization now included here. |
| A09:2021 | **Security Logging and Monitoring Failures** | Lack of proper logging and alerting mechanisms makes detection and incident response difficult. |
| A10:2021 | **Server-Side Request Forgery (SSRF)** | Web server is tricked into making requests to internal or unintended resources, often leading to sensitive data leaks or server-side abuse. |