**Stage – 1:**

**List of Vulnerability Table ➖**

|  |  |  |
| --- | --- | --- |
| **S.no** | **Vulnerability Name** | **CWE - No** |
| **1**  **2**  **3**  **4**  **5** | SQL Injection  Cross-Site Scripting (XSS)  Broken Authentication  Sensitive Data Exposure  Security Misconfiguration | CWE-89  CWE-79  CWE-287  CWE-200  CWE-16 |

**REPORT:-**

1. **Vulnerability Name:-** SQL Injection

**CWE : -** CWE-89

**OWASP/SANS Category:-** Injection Flaws

**Description:-** SQL Injection occurs when an attacker inserts malicious SQL queries into input fields to manipulate the database. This can lead to unauthorized access, data leakage, and even deletion of critical information.

**Business Impact**::-  Unauthorized access to sensitive data

 Data corruption or deletion

 Loss of database integrity

1. **Vulnerability Name:-** Cross-Site Scripting (XSS)

**CWE : -** CWE-79

**OWASP/SANS Category:-** Cross-Site Scripting

**Description:-** XSS attacks occur when an attacker injects malicious scripts into a trusted website. This allows them to steal user data, manipulate website content, or redirect users to harmful sites.**.**

**Business Impact**::-

* User session hijacking
* Theft of credentials or personal data
* Spread of malware

1. **Vulnerability Name:-** Broken Authentication

**CWE : -** CWE-287

**OWASP/SANS Category:-** Identification and Authentication Failures

**Description:-** Broken authentication occurs when an application fails to properly enforce authentication mechanisms, allowing attackers to bypass login credentials, session management, or exploit weak passwords. This can lead to unauthorized access to user accounts or administrative controls.

**Business Impact**::-

* Unauthorized access to accounts
* Identity theft and fraud
* Privilege escalation, leading to full system compromise

1. **Vulnerability Name:-** Sensitive Data Exposure

**CWE : -** CWE-200

**OWASP/SANS Category:-** Sensitive Data Exposure

**Description:-** Sensitive data exposure occurs when applications do not properly protect confidential information, such as passwords, credit card details, or personal data. This can happen due to weak encryption, improper data storage, or insecure transmission.

**Business Impact**::-

• Leakage of personally identifiable information (PII)  
• Financial fraud and identity theft  
• Non-compliance with data protection regulations (e.g., GDPR, HIPAA)

1. **Vulnerability Name:-** Security Misconfiguration

**CWE : -** CWE-16

**OWASP/SANS Category:-** Security Misconfiguration

**Description:-** Security misconfiguration happens when an application, database, or server is not properly secured, leaving it vulnerable to attacks. This includes using default credentials, unnecessary services, and misconfigured security headers.

**Business Impact**::-

• Exposure of sensitive information  
• Increased attack surface for hackers  
• Exploitation of weak security controls

**Stage – 2** : **Selecting Some Features and Grouping Them**

In this stage, we carefully selected key cybersecurity vulnerabilities and categorized them based on their impact, exploitation method, and security domain. The primary features were grouped as follows:

1. **Injection Attacks:**
   * SQL Injection (CWE-89)
   * Cross-Site Scripting (XSS) (CWE-79)
2. **Authentication and Access Control Issues:**
   * Broken Authentication (CWE-287)
3. **Data Protection and Privacy Threats:**
   * Sensitive Data Exposure (CWE-200)
4. **Configuration and Implementation Flaws:**
   * Security Misconfiguration (CWE-16)

Each category was analyzed based on real-world case studies and security frameworks. The vulnerabilities were mapped to security solutions such as firewalls, encryption, zero-trust models, and AI-driven threat detection.

**Overview:**

This stage focuses on utilizing Nessus, a widely used vulnerability assessment tool, to identify security weaknesses in a given network or web application. Nessus provides a detailed analysis of vulnerabilities by scanning systems for outdated software, misconfigurations, and other potential security threats. The key aspects of this stage include:

* Setting up Nessus for scanning target systems.
* Identifying vulnerabilities and classifying them based on severity levels (Critical, High, Medium, Low, and Informational).
* Analyzing vulnerability reports to understand potential attack vectors.
* Providing recommendations for remediation and strengthening security posture.

By leveraging Nessus, we aimed to gain hands-on experience with vulnerability assessment and understand its significance in proactive cybersecurity measures.

**What I Understood About Nessus**

Nessus is a powerful tool for automated security scanning, widely used by cybersecurity professionals for vulnerability management. From this stage, I learned the following key points:

* **Ease of Use and Configuration:** Nessus provides a user-friendly interface with pre-configured policies that simplify vulnerability scanning.
* **Comprehensive Vulnerability Detection:** It detects a wide range of vulnerabilities, including outdated software, missing patches, misconfigurations, and weak credentials.
* **Severity Classification:** The tool categorizes vulnerabilities based on severity, allowing security teams to prioritize remediation efforts effectively.
* **Automated and Scheduled Scanning:** Nessus enables continuous monitoring of security posture by scheduling automated scans.
* **Integration with Security Operations:** The generated reports can be integrated with SIEM solutions to enhance threat detection and response capabilities.

Through this analysis, I understood the critical role Nessus plays in vulnerability management and how organizations can leverage it to mitigate cybersecurity risks effectively.

**Target website ➖** Example vulnerable website for penetration testing (e.g., testsite.com):

**Target ip address:-** 192.168.1.10

**List of vulnerability ➖**

|  |  |  |  |
| --- | --- | --- | --- |
| **s.no** | **Vulnerability name** | **Severity** | **plugins** |
| 1 | SQL Injection | High | 10076 |
| |  | | --- | | 2 | | Cross-Site Scripting (XSS) | Medium | 20007 |
| 3 | Broken Authentication | High | 30012 |
| 4 | Sensitive data Exposure | Critical | 40123 |
| 5 | |  | | --- | |  |   Security Misconfiguration | Medium | 50214 |

**REPORT:-**

**Vulnerability Name:** SQL Injection  
**Severity:** High  
**Plugin:** 10076  
**Port:** 80 (HTTP)

**Description:**  
SQL Injection vulnerability exists in the web application, allowing attackers to manipulate database queries by injecting malicious SQL statements. This can lead to unauthorized data access, information leakage, or complete database compromise.

**Solution:**

* Implement prepared statements and parameterized queries.
* Use web application firewalls (WAF) to filter malicious queries.
* Conduct regular security audits and penetration testing.

**Business Impact:**

* Unauthorized access to sensitive user information.
* Financial losses due to data breaches.

Legal and compliance penalties (e.g., GDPR, HIPAA violations).