Stage – 1:

List of Vulnerability Table —

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

REPORT:-

1) Vulnerability Name :- SQL Injection (SQLi)

CWE :- CWE-89 (Improper Neutralization of Special Elements in SQL Commands)

OWASP/SANS Category :- OWASP Top 10 (A03:2021 - Injection) / SANS 25 (#1 - SQL Injection)

Description :- SQL Injection (SQLi) is one of the most common and dangerous vulnerabilities in web applications. It occurs when an attacker is able to inject malicious SQL queries or commands into the input fields of a web application (such as login forms, search boxes, or URL parameters) that are not properly sanitized or validated by the application.

Business Impact :-

Direct Financial Loss: If an attacker uses SQLi to steal sensitive financial data (such as credit card numbers, bank account information, or transaction histories), it can lead to fraud, unauthorized transactions, or identity theft.

Ransomware: An attacker might inject SQLi to deliver ransomware payloads that encrypt sensitive business data, demanding a ransom payment in exchange for restoring access to the data.

Cost of Remediation: Identifying, mitigating, and fixing a SQLi vulnerability requires significant resources, including the cost of cybersecurity tools, security audits, hiring external security consultants, and applying patches.

2) Vulnerability Name :- Cross-Site Scripting (XSS)

CWE :- CWE-79 (Improper Neutralization of Input During Web Page Generation)

OWASP/SANS Category :- OWASP Top 10 (A07:2021 - Identification and Authentication Failures) / SANS 25 (#2 - XSS)

Description :- Cross-Site Scripting (XSS) is a common vulnerability that allows attackers to inject malicious scripts into web pages viewed by other users. These scripts are typically written in JavaScript and are executed by the victim's browser. XSS vulnerabilities occur when a web application does not properly validate or sanitize user input, allowing attackers to inject executable code into the content that is delivered to users' browsers**.**

Business Impact :-

- Session hijacking leading to account takeovers.
- Data theft (stealing cookies, personal information).
- Reputation damage if malicious scripts alter website content.

3) Vulnerability Name :- Broken Authentication

CWE :- CWE-287 (Improper Authentication)

OWASP/SANS Category :- OWASP Top 10 (A07:2021 - Identification and Authentication Failures) / SANS 25 (#3 - Broken Authentication)

Description :- Broken Authentication refers to vulnerabilities in the authentication mechanism of an application or system that allows attackers to bypass or compromise the authentication process. This can occur due to weak, improper, or misconfigured authentication methods, allowing unauthorized users to access sensitive data or perform unauthorized actions. Broken authentication can arise from a wide range of flaws, including the mishandling of credentials, session management issues, or a failure to implement proper security measures.

Business Impact :-

- User account compromise, leading to identity theft.
- Privilege escalation, allowing attackers to gain admin access.
- Financial and reputational loss due to unauthorized transactions.

4) Vulnerability Name :- Insecure Direct Object References (IDOR)

CWE :- CWE-639 (Authorization Bypass Through User-Controlled Key)

OWASP/SANS Category :- OWASP A06:2021 (Vulnerable and Outdated Components)

Description :- Insecure Direct Object References (IDOR) is a type of security vulnerability that occurs when an application provides direct access to objects (such as files, database records, or URLs) based on user-controlled input, but fails to properly validate the user's authorization to access those objects. In an IDOR attack, an attacker can manipulate the input (such as a URL parameter or a form field) to gain unauthorized access to objects or resources that they should not be able to access**.**