**ASIM SHIKHAR 21BCE2595**

**ASSIGNMENT = 1**

***What is OWASP ?***

OWASP, or the Open Web Application Security Project, is a non-profit organization focused on improving the security of software. It is a global community of security experts, developers, and organizations that work together to create resources, tools, and best practices to help organizations design, develop, and maintain secure web applications and APIs.

**ITS PRIMARY OBJECTIVES ?**

**Education**: OWASP provides a wealth of educational materials, including guides, documentation, videos, and training courses, to help developers and security professionals understand and address web application security issues.

**Tools and Resources**: OWASP produces a wide range of open-source tools and resources that assist in identifying and mitigating web application vulnerabilities. Some of the most well-known tools include the OWASP ZAP (Zed Attack Proxy) for penetration testing and the OWASP Top Ten, a list of the most critical web application security risks.

**Best Practices**: OWASP promotes best practices and methodologies for secure application development through documents like the OWASP Development Guide and Cheat Sheets.

**Community and Conferences**: OWASP facilitates collaboration and knowledge sharing through local chapters, conferences, and community-driven projects.

**Vulnerability Awareness**: OWASP raises awareness about web application security vulnerabilities and threats, helping organizations and individuals stay informed about emerging risks and attack techniques.

***OWASP TOP 10 REPORT”S***

|  |  |
| --- | --- |
| **S.No** | **Name of the Vulnerability** |
| 1 | Broken Access Control |
| 2 | Cryptographic Failures |
| 3 | Injection |
| 4 | Insecure Design |
| 5 | Security Misconfiguration |
| 6 | Vulnerable and Outdated Components |
| 7 | Identification and  Authentication Failures |
| 8 | Software and Data Integrity Failures |
| 9 | Security Logging and  Monitoring Failures |
| 10 | Server Side Request Forgery |

***Vulnerabilities ?***

1. ***Broken Access Control***

Broken Access Control occurs when a system fails to properly restrict or manage user access

to resources. This can result in unauthorized users gaining access to sensitive data, functionality, or resources. It's a critical security issue because it can lead to data breaches, unauthorized actions, and compromise of system integrity. This weakness often arises due to improper authentication and authorization mechanisms.

**USAGE OF CWE 201:**

CWE-201: Information Exposure Through Sent Data is a Common Weakness Enumeration (CWE) category that describes a type of software vulnerability where sensitive information is inadvertently exposed or leaked through data sent to a user or system, often as part of error messages or responses. This exposure can occur due to poor design, configuration, or error-handling practices in software applications. CWE-201 is also sometimes referred to as "Information Exposure Through Response Headers" because it frequently involves HTTP response headers**.**

**BUSINESS IMPACT:**

CWE-201 relates to the inadvertent exposure of sensitive information through data transmission. When combined with Broken Access Control, it can amplify the risk. For example, if a user with insufficient access privileges attempts to access a restricted resource, the system may respond with an error message that includes sensitive information. This exposed data could be used maliciously, potentially leading to identity theft, social engineering attacks, or legal issues. The business consequences can include reputational damage, regulatory fines, customer trust erosion, and legal actions.

1. ***Cryptographic Failures (CWE-310):***

Cryptographic Failures encompass various weaknesses related to the improper implementation or use of cryptographic algorithms and mechanisms. These issues can lead to vulnerabilities such as weak encryption, poor key management, or insecure random number generation. When cryptographic systems are compromised, attackers can decrypt sensitive data or impersonate legitimate entities, compromising the confidentiality and integrity of information.

**USAGE OF CWE 310:**

CWE-310 is used to identify weaknesses in cryptographic implementations. Developers and security experts use this CWE to ensure that encryption and cryptographic processes are correctly implemented and configured. This involves selecting strong encryption algorithms, securely managing encryption keys, and protecting against common cryptographic vulnerabilities, such as weak ciphers or poor key management.

**BUSINESS IMPACT:**

Cryptographic failures can lead to the compromise of sensitive data, including customer information, financial records, and trade secrets. This can result in regulatory penalties, loss of customer trust, legal actions, and damage to the company's reputation. It may also impact compliance with data protection laws.

1. ***Injection(CWE 77)***

Injection attacks occur when untrusted data is embedded within a command or query in an insecure manner. Common types include SQL Injection, Cross-Site Scripting (XSS), and Command Injection. These vulnerabilities can allow attackers to manipulate the behavior of a system, execute arbitrary code, or steal data. Injection flaws are often found in web applications, where user input isn't properly sanitized or validated before being processed.

**USAGE OF CWE 77:**

CWE-77 covers vulnerabilities related to injection attacks, including SQL Injection, Cross-Site Scripting (XSS), and Command Injection. Developers use this CWE to identify and prevent security flaws that allow attackers to inject malicious code or data into an application. This includes input validation, parameterized queries, and output encoding to prevent injection attacks

**BUSINESS IMPACT:**

Injection attacks can allow attackers to manipulate systems and steal or corrupt data. This can lead to service downtime, data breaches, financial losses, and legal repercussions. Businesses can suffer from reputational damage, loss of customer trust, and regulatory fines due to the exposure of sensitive information.

1. ***Insecure Design (CWE-749):***

Insecure Design issues stem from poor architectural choices or fundamental design flaws in a system or application. These vulnerabilities are challenging to mitigate since they're rooted in the system's structure. Insecure design can lead to a wide range of security problems, from inadequate input validation to weak access controls. It's crucial to identify and address these issues during the design phase to create a secure system.

**USAGE OF CWE 749:**

CWE-749 addresses security weaknesses that originate from flawed software design choices. It is used during the design and architecture phase of software development. Security architects and designers use this CWE to identify and rectify design flaws that could lead to security vulnerabilities. This involves making informed design decisions, threat modeling, and adhering to secure design principles

**BUSINESS IMPACT:**

Insecure design can lead to systemic vulnerabilities that are costly to remediate after implementation. It can result in significant development and maintenance expenses, delays in product delivery, and increased exposure to security risks. Additionally, it may damage the company's reputation if security flaws become public.

1. ***Security Misconfiguration (CWE-732):***

Security Misconfiguration refers to vulnerabilities that arise from improperly configured systems or software components. This could include leaving default settings enabled, exposing unnecessary services, or failing to apply security patches. Attackers often target misconfigured systems because they provide a low-hanging fruit for exploitation.

**USAGE OF CWE 732:**

CWE-732 focuses on security issues caused by misconfigurations in software, servers, and systems. Security administrators and developers use this CWE to identify and rectify misconfigurations that may expose vulnerabilities. It involves regular system reviews, adherence to security best practices, and proper configuration management to prevent security misconfigurations**.**

**BUSINESS IMPACT:**

Security misconfigurations can expose systems to a wide range of threats. This may result in unauthorized access, data breaches, service disruptions, and potential financial losses. The organization may also face legal and regulatory consequences for failing to protect sensitive data adequately.