AL2024_38 MITRE Shares 2024's Top 25 Most Dangerous Software Weaknesses (22nd November 2024)

Description

MITRE has released the 2024 list of the 25 most dangerous and commonly exploited software weaknesses, based on a review of over 31,000 vulnerabilities reported between June 2023 and June 2024. Software weaknesses are flaws, bugs, vulnerabilities, and errors in the design, code, or implementation of software systems. These weaknesses, if exploited, can lead to severe consequences such as unauthorized access, data theft, denial-of-service (DoS) attacks, and even full system compromise.

The ranking of these weaknesses is based on both their severity and frequency of exploitation. MITRE analyzed vulnerability data, focusing on the flaws cataloged in Cybersecurity and Infrastructure Security Agency's (CISA's) Known Exploited Vulnerabilities (KEV) list, which contains security issues actively targeted by adversaries. This annual update highlights the critical flaws organizations must prioritize in their security strategies to avoid serious breaches.

Top Weaknesses Identified:

1. **CWE-79: Cross-site Scripting (XSS)**
   - **Score:** 56.92
   - **KEV CVEs:** 3
   - Cross-site scripting vulnerabilities allow attackers to inject malicious scripts into web pages viewed by other users, enabling them to steal sensitive information, hijack sessions, or deface websites.

2. **CWE-787: Out-of-bounds Write**
   - **Score:** 45.20
   - **KEV CVEs:** 18
   - These vulnerabilities occur when a program writes data outside the boundaries of allocated memory, potentially leading to memory corruption, system crashes, or code execution.
3. **CWE-89: SQL Injection**
   - **Score:** 35.88
   - **KEV CVEs:** 4
   - SQL injection flaws allow attackers to manipulate SQL queries to gain unauthorized access to a database, leading to data theft, data manipulation, and system compromise.
4. **CWE-352: Cross-Site Request Forgery (CSRF)**

   - **Score:** 19.57
   - **KEV CVEs:** 0
   - CSRF allows attackers to perform unauthorized actions on behalf of a logged-in user, such as making changes to account settings or initiating transactions, without the user's consent.

5. **CWE-22: Path Traversal**
   - **Score:** 12.74
   - **KEV CVEs:** 4
   - Path traversal vulnerabilities let attackers access files or directories that are outside the intended scope by manipulating file path input, leading to the exposure of sensitive data or system compromise.

Attack Details

Attackers actively exploit these vulnerabilities to achieve various malicious goals, ranging from data theft and system takeover to denial-of-service attacks. The most common tactics include:

- **Injecting malicious scripts (XSS) or commands (SQLi, Command Injection)** into web applications to hijack user sessions or gain administrative access.
- **Exploiting memory corruption vulnerabilities (Out-of-bounds Write/Read, Use After Free)** to crash applications or execute arbitrary code remotely.
- **Bypassing authentication mechanisms (Improper Authentication, Missing Authorization)** to impersonate users and gain unauthorized access to resources.
- **Exposing sensitive information (Data Exposure, Hardcoded Credentials)** by improperly managing security controls or not encrypting sensitive data adequately.
- **Manipulating file systems (Path Traversal, OS Command Injection)** to retrieve or execute unauthorized files, sometimes to deploy malware.

Remediation

To mitigate these vulnerabilities, MITRE and CISA encourage organizations to adopt proactive measures throughout the software development lifecycle, emphasizing secure coding practices, routine vulnerability assessments, and the prioritization of high-risk weaknesses:

- **Cross-Site Scripting (CWE-79):** Implement proper input sanitization and output encoding, use Content Security Policy (CSP), and ensure security mechanisms are in place for user inputs.
- **Out-of-bounds Writes (CWE-787) & Reads (CWE-125):** Apply strict bounds checking during data handling and use memory-safe programming languages when possible.
- **SQL Injection (CWE-89):** Use parameterized queries, stored procedures, and ORM frameworks to prevent direct SQL query manipulation.

- **Cross-Site Request Forgery (CWE-352):** Implement anti-CSRF tokens in web forms and verify request origins to ensure actions are made intentionally by authenticated users.
- **Path Traversal (CWE-22):** Sanitize user input paths and ensure access control restrictions prevent unauthorized access to sensitive directories.

The Guyana National CIRT recommends that users and administrators review this alert and apply it where necessary.

References

1. *CWE - 2024 CWE Top 25 most dangerous software weaknesses.* (n.d.). Retrieved from Mitre https://cwe.mitre.org/top25/archive/2024/2024_cwe_top25.html
2. Gatlan, S. (2024, November 20). *MITRE shares 2024's top 25 most dangerous software weaknesses*. Retrieved from BleepingComputer. https://www.bleepingcomputer.com/news/security/mitre-shares-2024s-top-25-most-dangerous-software-weaknesses/
3. *Known Exploited Vulnerabilities Catalog | CISA.* (n.d.). Retrieved from Cybersecurity and Infrastructure Security Agency CISA. https://www.cisa.gov/known-exploited-vulnerabilities-catalog