Malware Detection and Classification

Team Members:

Naladala Navya

Sanisetty Hema Sagar

Kurra Naveen Abhiram

Vishnubhatla V L Sruta Keerthi

Vulnerabilities Report for Main Website

1. **Vulnerability Name:**
   Protection Mechanism Failure (Web Application Potentially Vulnerable to Clickjacking)

   CWE: CWE-693
   OWASP Category: A04:2021 – Broken Access Control

   Description:
   This weakness covers three distinct situations. A "missing" protection mechanism occurs when the application does not define any mechanism against a certain class of attack. An "insufficient" protection mechanism might provide some defences - for example, against the most common attacks - but it does not protect against everything that is intended. Finally, an "ignored" mechanism occurs when a mechanism is available and in active use within the product, but the developer has not applied it in some code path.

   Business Impact:
   Protection Mechanism Failure in web applications potentially vulnerable to clickjacking can result in severe business impacts. Clickjacking is a malicious technique where attackers deceive users into performing unintended actions. When protection mechanisms fail, it can lead to unauthorized data exposure, financial losses, and reputational damage. This, in turn, can erode trust, harm customer relationships, and lead to legal consequences, making robust security measures crucial for preserving a business's integrity and success.

Vulnerability Path: https://vtop2.vitap.ac.in/vtop/initialProcess

Steps to Reproduce:

Identify a security feature or protection mechanism in the application (e.g., authentication, access control).
Attempt to bypass or defeat the security mechanism.
Observe whether the application allows unauthorized access or circumvention.
If you successfully bypass the protection mechanism, you have reproduced a protection mechanism failure issue.

Recommendation:

Return the X-Frame-Options or Content-Security-Policy (with the 'frame-ancestors' directive) HTTP header with the page's response.
This prevents the page's content from being rendered by another site when using the frame or iframe HTML tags.

2. Vulnerability Name:

Improper Input Validation (CGI Generic XSS)

CWE: CWE-20
OWASP Category: A04:2021 – Injection

Description:

The product receives input or data, but it does not validate or incorrectly validates that the input has the properties that are required to process the data safely and correctly.

Business Impact:

Improper input validation, specifically in the context of CGI scripts, can lead to Cross-Site Scripting (XSS) vulnerabilities. The business impact of this vulnerability includes risks of data theft, unauthorized access, and damage to a company's reputation, potentially resulting in financial losses, legal liabilities, and loss of customer trust.

Vulnerability Path: https://vtop2.vitap.ac.in/vtop/initialProcess

Steps to Reproduce:

Identify a user input field in the target application, such as a search box or login form.
Attempt to enter malicious input, like SQL code, HTML tags, or special characters, in the input field.
Submit the input and observe if the application processes it without validation or sanitization.

If the application accepts and processes the input without any validation or sanitization, you have successfully reproduced the issue.

Recommendation:

The recommended solution for Improper Input Validation (CGI Generic XSS) is to implement robust input validation and sanitization mechanisms to prevent malicious code injection in web applications. This involves validating and encoding user inputs, utilizing security libraries, and employing web application firewalls to mitigate Cross-Site Scripting (XSS) vulnerabilities.

3. Vulnerability Name:

Deserialization of Untrusted Data ('Cross-site Scripting') (Apache Tomcat 9.0.0.M1)

CWE: CWE-502

OWASP Category: A08:2021 – Software & Data Integrity Failure

Description:

The product deserializes untrusted data without sufficiently verifying that the resulting data will be valid.

Business Impact:

The Deserialization vulnerabilities can lead to remote code execution, potentially allowing an attacker to execute arbitrary code on the server. This can result in data breaches, unauthorized access, and data manipulation, leading to reputational damage and legal consequences.

Vulnerability Path: https://vtop2.vitap.ac.in/vtop/initialProcess

Steps to Reproduce:

I. Identify the Target Application: Find a web application that uses deserialization to process incoming data, e.g., session data stored in cookies.

II. Create a Malicious Serialized Object: Craft a malicious serialized object that contains code you want to execute on the server.

III. Inject the Malicious Object: Inject the malicious serialized object into the application by sending it in a request. This might involve setting a cookie with the malicious data.

IV. Trigger Deserialization: Send a request to the application to trigger the deserialization process.

V. Exploit the Vulnerability: If the application does not properly validate the deserialized data, it may execute the malicious code contained within the serialized object.

Recommendation:

The recommendation for deserialization of untrusted data (Cross-site Scripting or XSS) is to implement strong input validation, sanitize user inputs, and use secure serialization libraries to prevent malicious code execution when processing data from untrusted sources.

4. Vulnerability Name:

Unrestricted Upload of File with Dangerous Type (Apache Tomcat 9.0.0.M1)

CWE: CWE-434

OWASP Category: A05:2021 – Security Misconfiguration

Description:

The product allows the attacker to upload or transfer files of dangerous types that can be automatically processed within the product's environment.

Business Impact:

Allowing users to upload and execute dangerous file types can lead to server compromise, data loss, and unauthorized access to sensitive information. This may result in legal and regulatory consequences.

Vulnerability Path: https://vtop2.vitap.ac.in/vtop/initialProcess

Steps to Reproduce:

I. Identify the File Upload Functionality: Find a web application that allows users to upload files, such as images or documents.

II. Create a Malicious File: Create a malicious file, such as a PHP script, and give it an innocuous name and extension that the application allows (e.g., "innocent.jpg").

III. Upload the Malicious File: Use the application's file upload functionality to upload the malicious file. The application may not properly validate the file type.

IV. Access the Uploaded File: Access the uploaded file through the application. Depending on the application's security controls, you may be able to execute the malicious code contained in the uploaded file, e.g., by accessing it through a URL.

Recommendation:

The recommendation or solution for unrestricted file uploads of dangerous types is to implement strict file type validation and content scanning to prevent malicious files from being uploaded.

5. Vulnerability Name:

Improper Certificate Validation (Apache Tomcat 9.0.0.M1)

CWE: CWE-295

OWASP Category: A02:2021 – Cryptographic failures

Description:

The product does not validate, or incorrectly validates, a certificate.

Business Impact:

The Improper certificate validation can expose users to man-in-the-middle attacks and unauthorized data access. This can harm trust in the application, leading to loss of customers and legal consequences.

Vulnerability Path: https://vtop2.vitap.ac.in/vtop/initialProcess

Steps to Reproduce:

I. Set Up a Man-in-the-Middle Attack: Intercept network traffic between a client and the server using tools like Wireshark to act as a man-in-the-middle.

II. Present a Forged Certificate: Generate a self-signed or forged SSL/TLS certificate to impersonate the server. Present this certificate during the SSL/TLS handshake.

III. Intercept and Modify Traffic: Intercept the SSL/TLS handshake between the client and server, presenting the forged certificate during the handshake.

IV. Exploit the Vulnerability: If the application does not properly validate the server's certificate, the client may accept the forged certificate, allowing you to intercept and modify traffic between the client and server.

Recommendation:

The recommendation for improper certificate validation is to ensure that digital certificates are thoroughly validated during authentication processes, using up-to-date and secure methods, to prevent unauthorized access and protect against security vulnerabilities.

6. Vulnerability Name:

Uncontrolled Resource Consumption (Apache Tomcat 9.0.0.M1)

CWE: CWE-400

OWASP Category: A08:2021 – Software & Integrity Failure

Description:

The product does not properly control the allocation and maintenance of a limited resource, thereby enabling an actor to influence the number of resources consumed, eventually leading to the exhaustion of available resources.

Business Impact:

Uncontrolled resource consumption can lead to denial of service (DoS) attacks, causing system downtime, loss of revenue, and reputational damage.

Vulnerability Path: https://vtop2.vitap.ac.in/vtop/initialProcess

Steps to Reproduce:

I. Identify the Target Application: Find an application that handles resource-intensive operations.

II. Generate Resource-Intensive Requests: Create and send a large number of resource-intensive requests to the application. For example, sending multiple requests that require significant CPU, memory, or network bandwidth.

III. Observe Resource Exhaustion: Monitor the application to observe how it handles the resource-intensive requests. If the application doesn't have proper resource controls, it may become slow or unresponsive, leading to a potential DoS condition.

Recommendation:

The recommendation for uncontrolled resource consumption is to implement strict resource monitoring and management protocols to ensure that resources such as CPU, memory, and bandwidth are used efficiently and within predefined limits. This may involve using resource management tools, setting usage quotas, and optimizing code or configurations to prevent excessive resource consumption.

7. Vulnerability Name:

Improper Locking (Apache Tomcat 9.0.0.M1)

CWE: CWE-667

OWASP Category: A07:2021 – Identification &Authentication Failure

Description:

The product does not properly acquire or release a lock on a resource, leading to unexpected resource state changes and behaviours.

Business Impact:

Using weak or broken cryptographic algorithms can lead to data breaches, compromised confidentiality, and loss of trust in the security of the application.

Vulnerability Path: https://vtop2.vitap.ac.in/vtop/initialProcess

Steps to Reproduce:

Reproducing this vulnerability typically involves analysing the application's use of cryptography and identifying instances where weak or broken algorithms are used. Exploiting it would depend on the specific cryptographic context in the application.

Recommendation:

The Improper locking is a software issue were multiple threads or processes access shared resources without proper synchronization. The recommended solution is to implement thread-safe mechanisms, such as mutexes or semaphores, to ensure exclusive access to shared data, preventing data corruption or race conditions.

8. Vulnerability Name:

Session Fixation (Apache Tomcat 9.0.0.M1)

CWE: CWE-384

OWASP Category: A07:2021 – Identification &Authentication Failure

Description:

Authenticating a user, or otherwise establishing a new user session, without invalidating any existing session identifier gives an attacker the opportunity to steal authenticated sessions.

Business Impact:

Session fixation can lead to unauthorized access, data exposure, and identity theft, resulting in reputational damage and legal consequences.

Vulnerability Path: https://vtop2.vitap.ac.in/vtop/initialProcess

Steps to Reproduce:

I. Identify the Target Application: Find a web application that uses session management.

II. Obtain a Session ID: Obtain a valid session ID from the application, either by registering an account or using an existing session.

III. Share the Session ID: Share the valid session ID with the victim, for example, by sending a link with the session ID as a URL parameter.

IV. Victim Uses the Session: Encourage the victim to use the session by clicking on the link or accessing the application with the provided session ID.

V. Exploit the Vulnerability: If the application does not properly handle session management and associates the session ID provided by the attacker with the victim's account, the attacker can impersonate the victim.

Recommendation:

The Session fixation is a security vulnerability where an attacker can fix or set a user's session ID, potentially gaining unauthorized access to the user's account. To mitigate session fixation, implement robust session management practices, such as generating a new session ID after login, using secure session cookies, and regularly rotating session IDs during user interactions to make it harder for attackers to predict or fix a session ID.

9. Vulnerability Name:

Information Disclosure

CWE: CWE-200

OWASP Category: A03:2021 - Cryptographic Failures

Description:

The product exposes sensitive information to an actor that is not explicitly authorized to have access to that information.

Business Impact:

Information disclosure jeopardizes privacy, competitive standing, and trust, potentially resulting in legal actions, financial losses, and reputational harm, undermining an organization's security, prosperity, and image.

Vulnerability Path: https://vtop2.vitap.ac.in/vtop/initialProcess

Steps to Reproduce:

I. Identify the vulnerable system or code.
II. Reproduce the issue by capturing relevant data or using known attack vectors.
III. Analyse the results to confirm the information disclosure.
IV. Fix the vulnerability by patching, modifying code, or adjusting configurations.
V. Test the fix thoroughly, monitor for any recurrence, and document the process.

Recommendation:

Information disclosure refers to the unintentional or unauthorized release of sensitive data. To prevent it, organizations should implement robust cybersecurity measures, including encryption, access controls, regular security audits, and employee training, to safeguard sensitive information and ensure compliance with data protection regulations.