Task 2:Report on Tomcat Vulnerability

Name:Aakanksha Kalyani

Title : Out-of-date Version (Tomcat)

Domain : http://zero.webappsecurity.com/

Subdomain : http://zero.webappsecurity.com/resources/js/

IDENTIFIED VERSION : 7.0.70 (contains 4 critical and 44 other vulnerabilities)

LATEST VERSION : 10.0.5

Steps to reproduce :

Step 1: Add Website you want to s to the dialogue box.

Step 2: Define the customization option to scan as per your need

Step 3: It will start scanning it automatically

Step 4: Choose any one of the Critical Vulnerabilities.

Step 5: Write Reports for the vulnerability you wanted to write. Make sure the report should not be the same as in Netsparker.

Impact : Apache Tomcat uses a package renamed copy of Apache Commons FileUpload to implement the file upload requirements of the Servlet specification. A denial of service vulnerability was identified in Commons FileUpload that occurred when the length of the multipart boundary was just below the size of the buffer (4096 bytes) used to read the uploaded file. This caused the file upload process to take several orders of magnitude longer than if the boundary was the typical tens of bytes long.

Mitigation : Upgrade your installation of Tomcat to the latest stable version.

References : KNOWN VULNERABILITIES IN THIS VERSION

Apache Tomcat Deserialization of Untrusted Data Vulnerability

The fix for CVE-2020-9484 was incomplete. When using Apache Tomcat 10.0.0-M1 to 10.0.0, 9.0.0.M1 to 9.0.41, 8.5.0 to 8.5.61 or 7.0.0. to 7.0.107 with a configuration edge case that was highly unlikely to be used, the Tomcat instance was still vulnerable to CVE-2020-9494. Note that both the previously published prerequisites for CVE-2020-9484 and the previously published mitigations for CVE-2020-9484 also apply to this issue.

External References :

CVE-2021-25329

Apache Tomcat Exposure of Sensitive Information to an Unauthorized Actor

When serving resources from a network location using the NTFS file system, Apache Tomcat versions 10.0.0-M1 to 10.0.0-M9, 9.0.0.M1 to 9.0.39, 8.5.0 to 8.5.59 and 7.0.0 to 7.0.106 were susceptible to JSP source code disclosure in some configurations. The root cause was the unexpected behavior of the JRE API File.getCanonicalPath() which in turn was caused by the inconsistent behavior of the Windows API (FindFirstFileW) in some circumstances.