

Security Advisory Report - OBSO-1905-01

Apache Tomcat for Windows CGI Servlet Command Line Argument Handling Remote Code Execution (CVE-2019-0232)

Release Date: 2019-05-07 13:07:58

Last Update: 2019-06-21 17:20:58

Summary

Update #2: no Unify product is affected.

Apache Tomcat for **Windows** contains a flaw in the CGI Servlet in `catalina/servlets/CGIServlet.java` that is triggered as improperly quoted command line arguments passed via the JRE are not properly handled. This may allow a **remote attacker to execute arbitrary code when** running on Windows in a non-default configuration in conjunction with batch files.

When running on Windows with **enableCmdLineArguments** enabled, the CGI Servlet in Apache Tomcat 9.0.0.M1 to 9.0.17, 8.5.0 to 8.5.39 and 7.0.0 to 7.0.93 is vulnerable to Remote Code Execution due to a bug in the way the JRE passes command line arguments to Windows. **The CGI Servlet is disabled by default. The CGI option enableCmdLineArguments is disable by default** in Tomcat 9.0.x (and will be disabled by default in all versions in response to this vulnerability).

The vulnerability has been patched for these versions: 7.0.94, 8.5.40, 9.0.19

Details

Common Gateway Interface (CGI) is a standard protocol to allow web servers to execute command line programs / scripts via web requests. This protocol also allows passing of command line arguments to the script or program being executed via URL parameters. The protocol itself is defined in [RFC 3875](#).

Apache Tomcat supports execution of CGI scripts / programs in a non-default configuration via [a special CGI servlet](#). This servlet also parses URL parameters and translates them into command line arguments. The actual execution of the CGI scripts happens via Java Runtime Environment (JRE)'s [java.lang.Runtime](#) class, `exec()` function.

When CGI support is enabled in Apache Tomcat in Windows, and command line argument passing is enabled, it is possible to cause command injection via parameter interpolation when calling a batch file (*.bat / *.cmd). This happens because "cmd.exe" performs interpolation on some special characters before execution which can cause other shell commands to be called. Neither Apache Tomcat or the Windows JRE perform any kind of input validation for these special characters. A partial list of these characters can be found [here](#) and [here](#). Additional information about why this issue is specific to the Windows JRE can be found in [this blog post](#) by Markus Wulftange.

<https://www.nightwatchcybersecurity.com/2019/04/30/remote-code-execution-rce-in-cgi-servlet-apache-tomcat-on-windows-cve-2019-0232/>

The detailed explanation of the jre behavior can be found in [Markus Wulfstange's blog](#) and this archived [MSDN blog](#).

Command line parsing in Windows is not consistent and therefore the implementation of proper quoting of command line argument even less.

This may allow the injection of additional arguments.

Additionally, since *CreateProcess* implicitly starts *.bat* and *.cmd* in a *cmd.exe* shell environment, even command injection may be possible.

As a sample, Java for Windows fails to properly quote command line arguments. Even with *ProcessBuilder* where arguments are passed as a list of strings:

- Argument injection is possible by providing an argument containing further quoted arguments, e.g., "arg 1" "arg 2" "arg 3".
- On *cmd.exe* process command lines, a simple '&calc&' alone suffices.

Only within the most strictly mode, the *VERIFICATION_CMD_BAT* verification type, injection is not possible:

- Legacy mode:
 - *VERIFICATION_LEGACY*: There is no *SecurityManager* present and *jdk.lang.Process.allowAmbiguousCommands* is not explicitly set to *false* (no default set)
 - allows argument injection
 - allows command injection in *cmd.exe* calls (explicit or implicit)
- Strict mode:
 - *VERIFICATION_CMD_BAT*: Most strictly mode, file ends with *.bat* or *.cmd*
 - does not allow argument injection
 - does not allow command injection in *cmd.exe* calls
 - *VERIFICATION_WIN32*: File does not end with *.bat* or *.cmd*
 - allows argument injection
 - allows command injection in *cmd.exe* calls (explicit or implicit)

However, Java's check for switching to the *VERIFICATION_CMD_BAT* mode can be circumvented by adding whitespace after the *.bat* or *.cmd*.

Note: The issue was fixed in Apache Tomcat 9.0.18 but the release vote for the 9.0.18 release candidate did not pass. Therefore, although users must download 9.0.19 to obtain a version that includes a fix for these issues, version 9.0.18 is not included in the list of affected versions.

Affected Products

Only products running on Windows in a non-default configuration in conjunction with batch files may be

affected.

Most Unify products run on Linux.

SESAP, UC Web client, Xpressions, Contact Center, License Manager, DLS and Fault Management are not affected.

All other products do not use Apache Tomcat for Windows,

Recommended Actions

Update Apache Tomcat to 7.0.94, 8.5.40, 9.0.19

References

<http://cve.mitre.org/cgi-bin/cvename.cgi?name=2019-0232>

<https://codewhitesec.blogspot.com/2016/02/java-and-command-line-injections-in-windows.html>

<https://blogs.msdn.microsoft.com/twistylittlepassagesallalike/2011/04/23/everyone-quotes-command-line-arguments-the-wrong-way/>

http://tomcat.apache.org/security-7.html#Fixed_in_Apache_Tomcat_7.0.94

http://tomcat.apache.org/security-8.html#Fixed_in_Apache_Tomcat_8.5.40

http://tomcat.apache.org/security-9.html#Fixed_in_Apache_Tomcat_9.0.18

<https://www.securityfocus.com/bid/107906> (Certificate expired)

<https://www.nightwatchcybersecurity.com/2019/04/15/upcoming-advisory-for-apache-tomcat-vulnerability-cve-2019-0232/>

<https://www.nightwatchcybersecurity.com/2019/04/30/remote-code-execution-rce-in-cgi-servlet-apache-tomcat-on-windows-cve-2019-0232/>

Advisory: OBSO-1905-01, status: general release

Security Advisories are released as part of Unify's Vulnerability Intelligence Process. For more information see <https://www.unify.com/security/advisories>.

Contact and Disclaimer

OpenScape Baseline Security Office

obso@atos.net

© Unify Software and Solutions GmbH & Co. KG 2019

Otto-Hahn-Ring 6

D-81739 München

www.unify.com

The information provided in this document contains merely general descriptions or characteristics of performance which in case of actual use do not always apply as described or which may change as a result of further development of the products. An obligation to provide the respective characteristics shall only exist if expressly agreed in the terms of contract. Availability and technical specifications are subject to change without notice.

Unify, OpenScape, OpenStage and HiPath are registered trademarks of Unify Software and Solutions GmbH & Co. KG.

All other company, brand, product and service names are trademarks or registered trademarks of their respective holders.