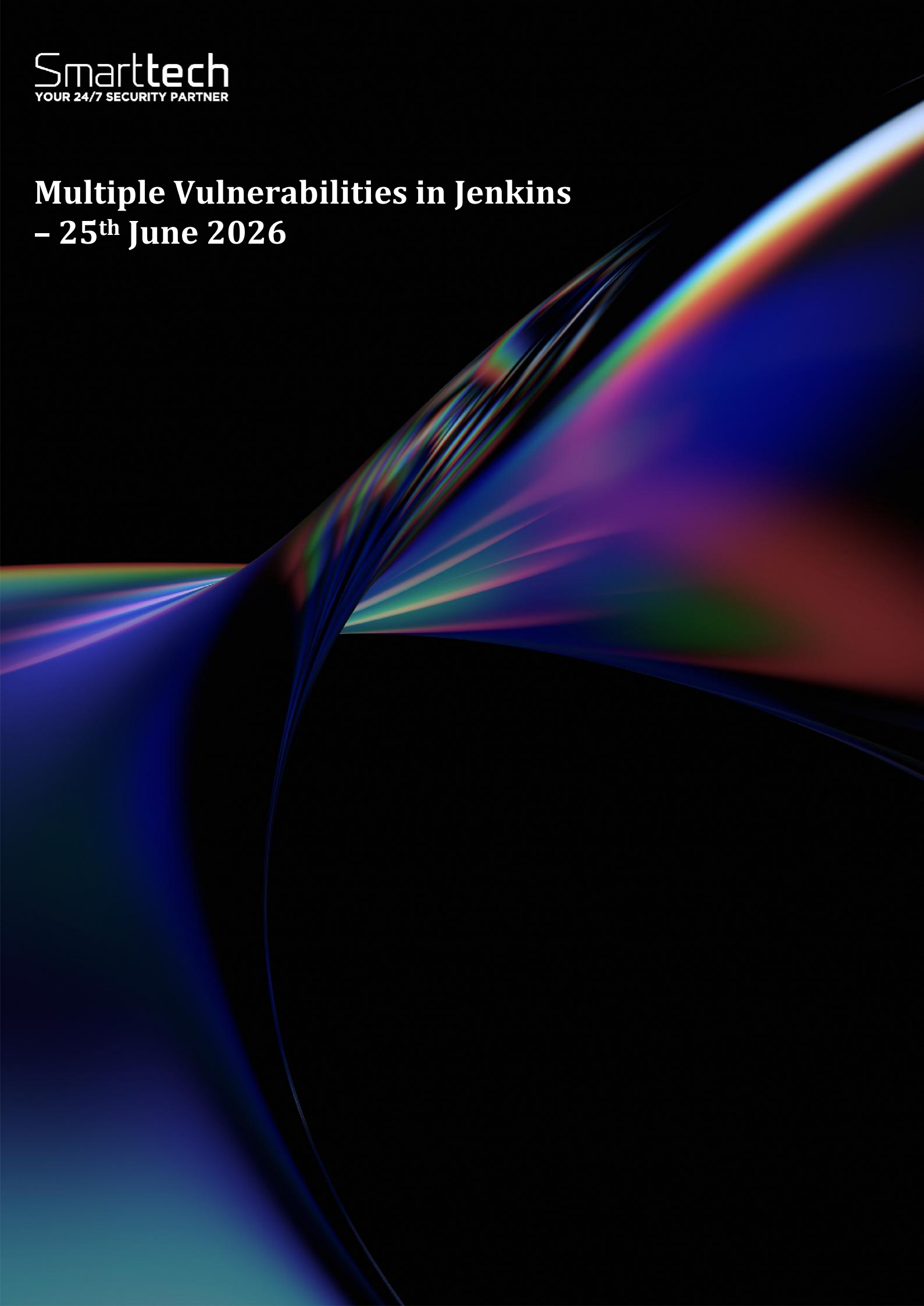
Smarttech
YOUR 24/7 SECURITY PARTNER

Multiple Vulnerabilities in Jenkins – 25th June 2026



Document ID	SMA-Threat Report
Document status	ISSUED
Issue Number	106
Authors	Dorin Constantin Banu < constantin.banu@smarttech247.com >
Verified by	Alin Curcan < alin.curcan@smarttech247.com >
Last modified	2026-06-25
Issue Date	2026-06-24

This document and any information therein are confidential property of Smarttech247 and without infringement neither the whole nor any extract may be disclosed, loaned, copied or used for manufacturing, provision of services or other purposes whatsoever without prior written consent of Smarttech247, and no liability is accepted for loss or damage from any cause whatsoever from the use of the document. Smarttech247 retain the right to alter the document at any time unless a written statement to the contrary has been appended.

Threat Reports are reports created by Smarttech247 based on high and critical severity vulnerabilities that may have a high potential to be exploited in the wild i.e. vulnerabilities that are present in most used products by companies and do not have an auto-update option or they are usually not automatically updated in case that could lead to some service disruption. This report is usually created as soon as the vulnerability is released, therefore we strongly recommend that the information is reviewed, tests are performed and patches are applied before the first proof-of-concept is released. Even though certain vulnerabilities may not have an active exploit in the wild at the time that we report on them, we take into consideration the wider risk and the impact it could have on systems, should an exploit like that be available after a while. Our duty is to report them on time and we recommend enterprises that, in order to keep critical business systems protected, they should consider, on average, ten working days to check whether or not the new vulnerability affects them, and if so, to implement actions in order to remove the risk.

Overview:

Multiple vulnerabilities have been identified in Jenkins. Successful exploitation could result in arbitrary code execution, unauthorized access, information disclosure and cross-site request forgery.

Risk

Government:

- Large and medium government entities: High
- Small government entities: High

Businesses:

- Large and medium business entities: High
- Small business entities: High

Technical summary

More details related to these vulnerabilities are as follows:

CVE ID	***Description***
Sandbox bypass vulnerability in Script Security Plugin **CVE-2026-57280** CVSS Base Score: **8.8**	Script Security Plugin provides a sandbox feature that allows running user-provided scripts safely by intercepting and checking potentially unsafe operations. Script Security Plugin 1402. v94c9ce464861 and earlier does not intercept the implicit type cast applied to each element of the iterated collection in a typed for loop (e.g. for (Type t in collection)), as this cast is performed during bytecode generation rather than in the transformed script AST. This allows attackers able to provide sandboxed scripts to invoke constructors of arbitrary types without those invocations being checked by the sandbox, bypassing the sandbox protection. This can be used to execute arbitrary code on the Jenkins controller.
Script security bypass vulnerability in Script Security Plugin **CVE-2026-57281** CVSS Base Score: **7.5**	Script Security Plugin 1402.v94c9ce464861 and earlier does not reject Groovy AST transformation annotations such as @CompileStatic and @TypeChecked that carry an extensions member, which causes Groovy to load and execute a script from the classpath at compile time, before the sandbox is applied. This may allow attackers able to define and run sandboxed scripts to execute code outside the sandbox, in the rare case that a suitable

	Groovy script is present on the classpath of the component that evaluates the script.
OS command injection vulnerability on agents in Git client Plugin **CVE-2026-57282** CVSS Base Score: **5.0**	Git client Plugin 6.6.0 and earlier does not correctly escape the workspace directory name when it is embedded into the SSH wrapper script generated by the "Manually provided keys" Git Host Key Verification strategy on Unix agents. This allows attackers able to control the name of a build's working directory (e.g. through a build parameter that determines the workspace directory) to inject shell command substitution and execute arbitrary commands on the agent. This vulnerability only has an impact when attackers can control working directories (e.g., the argument to the dir(…) Pipeline step) while not being able to control the Pipeline itself or the programs or build scripts it executes.
CSRF vulnerability and unrestricted instantiation of types in Pipeline: Groovy Plugin **CVE-2026-57283** **CVE-2026-57284** CVSS Base Score: **4.3**	Pipeline: Groovy Plugin 4331.v9d06ed4658ff and earlier does not restrict the types that can be instantiated through the Pipeline Snippet Generator, instantiating any type with a constructor annotated with @DataBoundConstructor in response to a request. This allows attackers to have Pipeline: Groovy Plugin instantiate types related to job or system configuration other than Pipeline steps. Additionally, this HTTP endpoint can be accessed using the GET method and does not require POST requests, resulting in a cross-site request forgery (CSRF) vulnerability. This allows attackers to create a script approval request attributed to another user, impersonating a trusted user when social engineering an administrator into approving a malicious script.
Missing permission check allows enumerating GitHub Enterprise server URLs in GitHub Branch Source Plugin **CVE-2026-57285** CVSS Base Score: **4.3**	GitHub Branch Source Plugin 1967.1969. v205fd594c821 and earlier does not perform a permission check in an HTTP endpoint that lists the GitHub API endpoints configured in the global plugin configuration. This allows attackers with Overall/Read permission to obtain the URLs of GitHub Enterprise servers configured by administrators.
Missing permission check in Git Parameter Plugin allows listing SCM branch and tag names **CVE-2026-57286** CVSS Base Score: **4.3**	Git Parameter Plugin 462.vdcf3df2ed2ca_ and earlier does not perform a permission check in an HTTP endpoint that populates the list of values for Git parameters by querying the SCM configured on a job, using the SCM credentials configured in Jenkins. This allows attackers with Item/Read permission to obtain information about the SCM repository used by a job they would otherwise be unable to access, such as branch names, tag names, and revision metadata.
Encrypted values of secrets in job and agent configurations not redacted by Job Configuration History Plugin **CVE-2026-57287** CVSS Base Score: **4.3**	Job Configuration History Plugin 1356. ve360da_6c523a_ and earlier does not redact the encrypted values of secrets when displaying historical job and agent configurations through its "View as XML" / "(RAW)" feature and its configuration diff views. This allows attackers with

	Item/Extended Read permission (but not Item/Configure permission) to view the encrypted values of secrets, such as build trigger tokens, that Jenkins would otherwise redact from the configuration shown to them.
LDAP injection vulnerability in Active Directory Plugin **CVE-2026-57288** CVSS Base Score: **3.7**	In Active Directory Plugin 2.41.1 and earlier, the Windows native (ADSI) authentication path does not escape the user name before building the LDAP search filter. This allows unauthenticated attackers to inject LDAP wildcard characters into the user name, enabling them to enumerate directory user and group names, and to authenticate as a matching user when they know that user's password but not their exact user name.
Missing permission check in MCP Server Plugin allows reading Pipeline replay scripts **CVE-2026-57300** CVSS Base Score: **4.3**	MCP Server Plugin 0.177.v629fdb_2557fe and earlier does not perform a permission check in the getReplayScripts MCP tool that returns the replay script of a Pipeline build. This allows attackers with Item/Read permission to obtain the Pipeline script of jobs.
SSL/TLS certificate validation unconditionally disabled by Bitbucket Push and Pull Request Plugin **CVE-2026-57289** CVSS Base Score: **4.8**	Bitbucket Push and Pull Request Plugin 3.3.8 and earlier unconditionally disables SSL/TLS certificate and hostname validation for the connections it makes to Bitbucket Server using Bearer token authentication. Because the Bearer token is transmitted in these requests, this allows attackers able to intercept network traffic to capture the token and impersonate the Jenkins controller to Bitbucket Server.
CSRF vulnerability in Priority Sorter Plugin **CVE-2026-57290** CVSS Base Score: **4.3**	Priority Sorter Plugin 936.v2c01c6b_84449 and earlier does not require POST requests in an HTTP endpoint that saves the global job priority configuration. This allows attackers to overwrite the global job priority configuration.
Missing permission checks and CSRF vulnerability in Gitee Plugin **CVE-2026-57291** **CVE-2026-57292** CVSS Base Score: **5.4**	Gitee Plugin 1288.v18b_deb_c9069b_ and earlier does not perform permission checks in several HTTP endpoints implementing form validation for its global configuration. This allows attackers with Overall/Read permission to connect to an attacker-specified URL using attacker-specified credentials IDs obtained through another method, capturing credentials stored in Jenkins. Additionally, these HTTP endpoints do not require POST requests, resulting in a cross-site request forgery (CSRF) vulnerability.
Incorrect permission check in Gitee Plugin allows enumerating credentials IDs **CVE-2026-57293** CVSS Base Score: **4.3**	Gitee Plugin 1288.v18b_deb_c9069b_ and earlier does not correctly perform a permission check in an HTTP endpoint. This allows attackers with global Item/Configure permission (while lacking Item/Configure permission on any particular job) to enumerate credentials IDs of credentials stored in Jenkins. Those can be used as part of an attack to capture the credentials using another vulnerability.
CSRF vulnerability and missing permission checks in EC2 Fleet Plugin **CVE-2026-57294** **CVE-2026-57295**	EC2 Fleet Plugin 4.2.3.539.v8fedff2a_81c3 and earlier does not perform permission checks in several HTTP endpoints used to validate cloud configurations. This allows attackers with

CVSS Base Score: **5.4**	Overall/Read permission to connect to an attacker-specified URL using attacker-specified credentials IDs obtained through another method, capturing AWS credentials stored in Jenkins. Additionally, these HTTP endpoints do not require POST requests, resulting in a cross-site request forgery (CSRF) vulnerability.
Path traversal vulnerability in External Workspace Manager Plugin **CVE-2026-57296** CVSS Base Score: **8.8**	External Workspace Manager Plugin 1.3.2 and earlier does not reject .. path segments when validating the custom workspace path provided to the exwsAllocate Pipeline step, allowing the resulting workspace path to escape the configured disk mount point. This allows attackers with Item/Configure permission to read arbitrary files on the Jenkins controller file system, which can lead to remote code execution (see Reading Files).
CSRF vulnerability and missing permission check in Contrast Continuous Application Security Plugin **CVE-2026-57297** **CVE-2026-57298** CVSS Base Score: **5.4**	Contrast Continuous Application Security Plugin 3.11 and earlier does not perform a permission check in an HTTP endpoint that tests the connection to a Contrast TeamServer. This allows attackers with Overall/Read permission to connect to an attacker-specified URL using an attacker-specified username, API key, and service key. Additionally, this HTTP endpoint does not require POST requests, resulting in a cross-site request forgery (CSRF) vulnerability.
Missing permission checks in Contrast Continuous Application Security Plugin allow enumerating Contrast metadata **CVE-2026-57299** CVSS Base Score: **4.3**	Contrast Continuous Application Security Plugin 3.11 and earlier does not perform permission checks in several HTTP endpoints that fill list box options with the names of the configured Contrast metadata. This allows attackers with Overall/Read permission to enumerate the names of configured Contrast metadata.
Builds executed on the Jenkins controller by OWASP ZAP Plugin can lead to RCE **CVE-2026-57301** CVSS Base Score: **8.8**	OWASP ZAP Plugin 1.0.7 and earlier does not support distributed builds, causing the file operations and build process of its "Automatically build ZAP" feature to be performed on the Jenkins controller rather than on the agent the build is assigned to. This allows attackers with Item/Configure permission to configure the feature to build an attacker-controlled project, executing arbitrary code on the Jenkins controller and bypassing any restriction confining the build to a specific agent. As of publication of this report, there is no fix.
Passwords stored in plain text by FitNesse Plugin **CVE-2026-57302** CVSS Base Score: **4.3**	FitNesse Plugin 1.36 and earlier stores passwords unencrypted in job config.xml files on the Jenkins controller as part of its configuration. These passwords can be viewed by users with Item/Extended Read permission or access to the Jenkins controller file system. As of publication of this report, there is no fix.
XXE vulnerability in Assembla Plugin **CVE-2026-57303** CVSS Base Score: **7.1**	Assembla Plugin 1.4 and earlier does not configure its XML parser to prevent XML external entity (XXE) attacks when parsing responses from the configured Assembla server. This allows attackers able to control the responses of the configured Assembla server to extract secrets from the Jenkins controller

	or perform server-side request forgery. As of publication of this report, there is no fix.
CSRF vulnerability and missing permission check in Assembla Plugin **CVE-2026-57304** **CVE-2026-57305** CVSS Base Score: **5.4**	Assembla Plugin 1.4 and earlier does not perform a permission check in an HTTP endpoint that tests the connection to an Assembla server. This allows attackers with Overall/Read permission to connect to an attacker-specified URL using an attacker-specified username and password. Additionally, this HTTP endpoint does not require POST requests, resulting in a cross-site request forgery (CSRF) vulnerability. This does not allow exploiting the XML external entity (XXE) vulnerability described in the previous advisory entry. As of publication of this report, there is no fix.
CSRF vulnerability and missing permission check in Zowe zDevOps Plugin **CVE-2026-57306** **CVE-2026-57307** CVSS Base Score: **4.2**	Zowe zDevOps Plugin 1.1.3.50.ve350c9b_450b_1 and earlier does not perform a permission check in an HTTP endpoint implementing a connection test. This allows attackers with Overall/Read permission to connect to an attacker-specified URL using attacker-specified credentials IDs obtained through another method, capturing credentials stored in Jenkins. Additionally, this HTTP endpoint does not require POST requests, resulting in a cross-site request forgery (CSRF) vulnerability. As of publication of this report, there is no fix.

Affected Products:

- Active Directory Plugin up to and including 2.41.1
- Assembla Plugin up to and including 1.4
- Bitbucket Push and Pull Request Plugin up to and including 3.3.8
- Contrast Continuous Application Security Plugin up to and including 3.11
- EC2 Fleet Plugin up to and including 4.2.3.539.v8fedff2a_81c3
- External Workspace Manager Plugin up to and including 1.3.2
- FitNesse Plugin up to and including 1.36
- Git client Plugin up to and including 6.6.0
- Git Parameter Plugin up to and including 462.vdcf3df2ed2ca_
- Gitee Plugin up to and including 1288.v18b_deb_c9069b_
- GitHub Branch Source Plugin up to and including 1967.1969.v205fd594c821
- Job Configuration History Plugin up to and including 1356.ve360da_6c523a_
- MCP Server Plugin up to and including 0.177.v629fdb_2557fe
- OWASP ZAP Plugin up to and including 1.0.7
- Pipeline: Groovy Plugin up to and including 4331.v9d06ed4658ff
- Priority Sorter Plugin up to and including 936.v2c01c6b_84449
- Script Security Plugin up to and including 1402.v94c9ce464861
- Zowe zDevOps Plugin up to and including 1.1.3.50.ve350c9b_450b_1

Fixed Versions:

- Active Directory Plugin should be updated to version 2.41.2
- Bitbucket Push and Pull Request Plugin should be updated to version 3.3.9
- Contrast Continuous Application Security Plugin should be updated to version 3.12
- EC2 Fleet Plugin should be updated to version 4.2.3.540.va_6eedb_7b_c112
- External Workspace Manager Plugin should be updated to version 1.4.0

- Git client Plugin should be updated to version 6.6.1
- Git Parameter Plugin should be updated to version 462.463.v496a_59f698e5
- Gitee Plugin should be updated to version 1292.v2559f2f3f2c0
- GitHub Branch Source Plugin should be updated to version 1967.1970.vd86979736546
- Job Configuration History Plugin should be updated to version 1367.vc8fa_b_15101dc
- MCP Server Plugin should be updated to version 0.178.vffe5a_e770f3b_
- Pipeline: Groovy Plugin should be updated to version 4331.4333.v50a_b_076c5199
- Priority Sorter Plugin should be updated to version 936.937.v5581d0b_2ccb_a_
- Script Security Plugin should be updated to version 1402.1405.vc96e74964250

As of publication of this report, no fixes are available for the following plugins:

- Assembla Plugin
- FitNesse Plugin
- OWASP ZAP Plugin
- Zowe zDevOps Plugin

Recommendations

Smarttech247 team recommend the following actions to be taken:

- Apply appropriate updates provided by Jenkins to vulnerable systems immediately after appropriate testing. (M1051: Update Software)
    - Safeguard 7.1 : Establish and Maintain a Vulnerability Management Process: Establish and maintain a documented vulnerability management process for enterprise assets. Review and update documentation annually, or when significant enterprise changes occur that could impact this Safeguard.
    - Safeguard 7.2: Establish and Maintain a Remediation Process: Establish and maintain a risk-based remediation strategy documented in a remediation process, with monthly, or more frequent, reviews.
    - Safeguard 7.4: Perform Automated Application Patch Management: Perform application updates on enterprise assets through automated patch management on a monthly, or more frequent, basis.
    - Safeguard 7.5 : Perform Automated Vulnerability Scans of Internal Enterprise Assets: Perform automated vulnerability scans of internal enterprise assets on a quarterly, or more frequent, basis. Conduct both authenticated and unauthenticated scans, using a SCAP-compliant vulnerability scanning tool.
    - Safeguard 7.7: Remediate Detected Vulnerabilities: Remediate detected vulnerabilities in software through processes and tooling on a monthly, or more frequent, basis, based on the remediation process.
    - Safeguard 12.1: Ensure Network Infrastructure is Up-to-Date: Ensure network infrastructure is kept up-to-date. Example implementations include running the latest stable release of software and/or using currently supported network-as-a-service (NaaS) offerings. Review software versions monthly, or more frequently, to verify software support.
    - Safeguard 18.1: Establish and Maintain a Penetration Testing Program: Establish and maintain a penetration testing program appropriate to the size, complexity, and maturity of the enterprise. Penetration testing program characteristics include scope, such as network, web application, Application Programming Interface (API), hosted services, and physical premise controls; frequency; limitations, such as acceptable

hours, and excluded attack types; point of contact information; remediation, such as how findings will be routed internally; and retrospective requirements.
    - Safeguard 18.2: Perform Periodic External Penetration Tests: Perform periodic external penetration tests based on program requirements, no less than annually. External penetration testing must include enterprise and environmental reconnaissance to detect exploitable information. Penetration testing requires specialized skills and experience and must be conducted through a qualified party. The testing may be clear box or opaque box.
    - Safeguard 18.3: Remediate Penetration Test Findings: Remediate penetration test findings based on the enterprise's policy for remediation scope and prioritization.
- Vulnerability scanning is used to find potentially exploitable software vulnerabilities to remediate them. (M1016: Vulnerability Scanning)
    - Safeguard 16.13: Conduct Application Penetration Testing: Conduct application penetration testing. For critical applications, authenticated penetration testing is better suited to finding business logic vulnerabilities than code scanning and automated security testing. Penetration testing relies on the skill of the tester to manually manipulate an application as an authenticated and unauthenticated user.
- Apply the Principle of Least Privilege to all systems and services. Run all software as a non-privileged user (one without administrative privileges) to diminish the effects of a successful attack. (M1026: Privileged Account Management)
    - Safeguard 4.7: Manage Default Accounts on Enterprise Assets and Software: Manage default accounts on enterprise assets and software, such as root, administrator, and other pre-configured vendor accounts. Example implementations can include: disabling default accounts or making them unusable.
    - Safeguard 5.4: Restrict Administrator Privileges to Dedicated Administrator Accounts: Restrict administrator privileges to dedicated administrator accounts on enterprise assets. Conduct general computing activities, such as internet browsing, email, and productivity suite use, from the user's primary, non-privileged account.
    - Safeguard 5.5: Establish and Maintain an Inventory of Service Accounts: Establish and maintain an inventory of service accounts. The inventory, at a minimum, must contain department owner, review date, and purpose. Perform service account reviews to validate that all active accounts are authorized, on a recurring schedule at a minimum quarterly, or more frequently
- Architect sections of the network to isolate critical systems, functions, or resources. Use physical and logical segmentation to prevent access to potentially sensitive systems and information. Use a DMZ to contain any internet-facing services that should not be exposed from the internal network. Configure separate virtual private cloud (VPC) instances to isolate critical cloud systems. (M1030: Network Segmentation)
    - Safeguard 12.2: Establish and Maintain a Secure Network Architecture: Establish and maintain a secure network architecture. A secure network architecture must address segmentation, least privilege, and availability, at a minimum.
- Use capabilities to detect and block conditions that may lead to or be indicative of a software exploit occurring. (M1050: Exploit Protection)
    - Safeguard 10.5: Enable Anti-Exploitation Features: Enable anti-exploitation features on enterprise assets and software, where possible, such as Microsoft® Data Execution Prevention (DEP), Windows® Defender Exploit Guard (WDEG), or Apple® System Integrity Protection (SIP) and Gatekeeper™.
- Restrict use of certain websites, block downloads/attachments, block Javascript, restrict browser extensions, etc. (M1021: Restrict Web-Based Content)

    - Safeguard 9.2: Use DNS Filtering Services: Use DNS filtering services on all enterprise assets to block access to known malicious domains.
    - Safeguard 9.3: Maintain and Enforce Network-Based URL Filters: Enforce and update network-based URL filters to limit an enterprise asset from connecting to potentially malicious or unapproved websites. Example implementations include category-based filtering, reputation-based filtering, or through the use of block lists. Enforce filters for all enterprise assets.
    - Safeguard 9.6: Block Unnecessary File Types: Block unnecessary file types attempting to enter the enterprise's email gateway.
- Remind users not to visit un-trusted websites or follow links provided by unknown or un-trusted sources. Inform and educate users regarding the threats posed by hypertext links contained in emails or attachments especially from un-trusted sources. (M1017: User Training)
    - Safeguard 14.1: Establish and Maintain a Security Awareness Program: Establish and maintain a security awareness program. The purpose of a security awareness program is to educate the enterprise's workforce on how to interact with enterprise assets and data in a secure manner. Conduct training at hire and, at a minimum, annually. Review and update content annually, or when significant enterprise changes occur that could impact this Safeguard.
    - Safeguard 14.2: Train Workforce Members to Recognize Social Engineering Attacks: Train workforce members to recognize social engineering attacks, such as phishing, pre-texting, and tailgating.

References

https://www.jenkins.io/security/advisory/2026-06-24/

CVEs

CVE-2026-57280
CVE-2026-57281
CVE-2026-57282
CVE-2026-57283
CVE-2026-57284
CVE-2026-57285
CVE-2026-57286
CVE-2026-57287
CVE-2026-57288
CVE-2026-57300
CVE-2026-57289
CVE-2026-57290
CVE-2026-57291
CVE-2026-57292
CVE-2026-57293
CVE-2026-57294
CVE-2026-57295
CVE-2026-57296
CVE-2026-57297
CVE-2026-57298
CVE-2026-57299

CVE-2026-57301
CVE-2026-57302
CVE-2026-57303
CVE-2026-57304
CVE-2026-57305
CVE-2026-57306
CVE-2026-57307

Smarttech
YOUR 24/7 SECURITY PARTNER

www.smarttech247.com