Smarttech
YOUR 24/7 SECURITY PARTNER

Multiple Vulnerabilities in Jenkins – 30th April 2026

Document ID	SMA-Threat Report
Document status	ISSUED
Issue Number	59
Authors	Dorin Constantin Banu < constantin.banu@smarttech247.com >
Verified by	Alin Curcan < alin.curcan@smarttech247.com >
Last modified	2026-04-30
Issue Date	2026-04-29

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, affecting Credentials Binding Plugin, GitHub Plugin, GitHub Branch Source Plugin, HTML Publisher Plugin, Matrix Authorization Strategy Plugin, Microsoft Entra ID (previously Azure AD) Plugin and Script Security Plugin. Successful exploitation could allow for remote code execution, information disclosure, unauthorized access, cross-site scripting and phishing attacks.

Risk

Government:

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

Businesses:

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

Technical summary

More details related to these vulnerabilities are as follows:

CVE ID	***Description***
Missing permission check in Script Security Plugin allows enumerating pending and approved classpaths **CVE-2026-42519** CVSS Base Score: **4.3**	Script Security Plugin 1399.ve6a_66547f6e1 and earlier does not perform a permission check in an HTTP endpoint. This allows attackers with Overall/Read permission to enumerate pending and approved Script Security classpaths. Script Security Plugin 1402.v94c9ce464861 requires Overall/Administer permission to enumerate pending and approved Script Security classpaths.
Path traversal vulnerability in Credentials Binding Plugin **CVE-2026-42520** CVSS Base Score: **7.5**	Credentials Binding Plugin 719.v80e905ef14eb_ and earlier does not sanitize file names for file and zip file credentials. This allows attackers able to provide credentials to a job to write files to arbitrary locations on the node filesystem. If Jenkins is configured to allow a low-privileged user to configure file or zip file credentials used for a job running on the built-in node, this can lead to remote code execution. Credentials Binding Plugin 720.v3f6decef43ea_ sanitizes the

	file name provided for file and zip file credentials, preventing path traversal.
Unsafe deserialization allows invoking parameterless constructors in Matrix Authorization Strategy Plugin **CVE-2026-42521** CVSS Base Score: **6.5**	Matrix Authorization Strategy Plugin 2.0-beta-1 through 3.2.9 (both inclusive) invokes parameterless constructors of classes specified in configuration when deserializing inheritance strategies, without restricting the classes that can be instantiated. This can be abused by attackers with Item/Configure permission to instantiate arbitrary types, which may lead to information disclosure or other impacts depending on the classes available on the classpath. Matrix Authorization Strategy Plugin 3.2.10 verifies that the class being instantiated is an inheritance strategy implementation, preventing instantiation of arbitrary types.
Missing permission check in GitHub Branch Source Plugin allows performing a connection test **CVE-2026-42522** **CVSS Base Score: 4.3**	GitHub Branch Source Plugin 1967.vdea_d580c1a_b_a_ and earlier does not perform a permission check in a method implementing form validation. This allows attackers with Overall/Read permission to connect to an attacker-specified URL with attacker-specified GitHub App credentials. GitHub Branch Source Plugin 1967.1969.v205fd594c821 requires Overall/Manage permission to perform the connection test.
XSS vulnerability in GitHub Plugin **CVE-2026-42523** **CVSS Base Score: 9.0**	GitHub Plugin 1.46.0 and earlier improperly processes the current job URL as part of JavaScript implementing validation of the feature "GitHub hook trigger for GITScm polling". This results in a stored cross-site scripting (XSS) vulnerability exploitable by non-anonymous attackers with Overall/Read permission. GitHub Plugin 1.46.0.1 no longer processes the current job URL as part of JavaScript implementing validation of the feature "GitHub hook trigger for GITScm polling".
XSS vulnerability in legacy wrapper file in HTML Publisher Plugin **CVE-2026-42524** **CVSS Base Score: 8.0**	HTML Publisher Plugin 427 and earlier does not escape job name and URL in the legacy wrapper file. This results in a stored cross-site scripting (XSS) vulnerability exploitable by attackers with Item/Configure permission. HTML Publisher Plugin 427.1 escapes job name and URL when generating the legacy wrapper file.
Open redirect vulnerability in Microsoft Entra ID (previously Azure AD) Plugin **CVE-2026-42525** **CVSS Base Score: 4.3**	Microsoft Entra ID (previously Azure AD) Plugin 666.v6060de32f87d and earlier does not restrict the redirect URL after login. This allows attackers to perform phishing attacks by having users go to a Jenkins URL that will forward them to a different site after successful authentication. Microsoft Entra ID (previously Azure AD) Plugin 667.v4c5827a_e74a_0 only redirects to relative (Jenkins) URLs.

Affected Products:

- Credentials Binding Plugin up to and including 719.v80e905ef14eb_
- GitHub Plugin up to and including 1.46.0
- GitHub Branch Source Plugin up to and including 1967.vdea_d580c1a_b_a_
- HTML Publisher Plugin up to and including 427
- Matrix Authorization Strategy Plugin up to and including 3.2.9
- Microsoft Entra ID (previously Azure AD) Plugin up to and including 666.v6060de32f87d
- Script Security Plugin up to and including 1399.ve6a_66547f6e1

Fixed Versions:

- Credentials Binding Plugin should be updated to version 720.v3f6decef43ea_
- GitHub Plugin should be updated to version 1.46.0.1
- GitHub Branch Source Plugin should be updated to version 1967.1969.v205fd594c821
- HTML Publisher Plugin should be updated to version 427.1
- Matrix Authorization Strategy Plugin should be updated to version 3.2.10
- Microsoft Entra ID (previously Azure AD) Plugin should be updated to version 667.v4c5827a_e74a_0
- Script Security Plugin should be updated to version 1402.v94c9ce464861

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/2026-04-29/

CVEs

CVE-2026-42519
CVE-2026-42520
CVE-2026-42521
CVE-2026-42522
CVE-2026-42523
CVE-2026-42524
CVE-2026-42525

Smarttech
YOUR 24/7 SECURITY PARTNER

www.smarttech247.com