CFEngine

< The CFEngine Blog

# CVE-2026-24710, CVE-2026-24711 & CVE-2026-24712 - Injection & broken access control

cve policy-language mission-portal security

Posted by Ole Herman Elgesem

February 9, 2026

Share this

We have recently discovered and fixed multiple security issues in CFEngine. These discoveries were made by ethical hackers as a part of our HackerOne bug bounty program. All of these issues can be remediated by upgrading to CFEngine 3.27.0, 3.24.3, 3.21.8, or later versions. We have no indications of these issues being exploited or known outside of the company and the security researchers that reported them. All of these issues require authentication / some initial level of access - they cannot be exploited on their own, without first compromising an account, or similar.

## CVE-2026-24710 - Missing input sanitization / injection in Mission Portal



was susceptible to injection (JavaScript XSS,

ectly sanitizing user input and/or not

Tahsin Akbar Ohi and i-forgot-it for

### Impact

These issues could enable an attacker to inject malicious code into Mission Portal and:

- Run JavaSript when another user, for example admin, visits a page (XSS).
- Run shell commands on the hub (Command injection).
- Extract information through timing based methods (blind SQL injection).

### Remediation

CVE-2026-24710 affects CFEngine Enterprise 3.26.0, 3.24.2, 3.21.7, and earlier versions. The issues have been fixed in 3.27.0, 3.24.3, 3.21.8, and later versions. If you are using one of the affected versions, we recommend upgrading.

Our documentation has details on how to upgrade CFEngine:

https://docs.cfengine.com/docs/lts/getting-started/01-installation/upgrading

## CVE-2026-24711 - Broken access control in Mission Portal

Issues around access control could enable an attacker with access to a low privilege user account to get access to more information than they should have, and in the most severe case enable them to escalate their privileges to become an admin. Thank you to Tahmid Akbar Omim for discovering and responsibly disclosing these issues.



ploited without first compromising a user account could use this to bypass access nub and ultimately the whole infrastructure

### Remediation

CVE-2026-24711 affects CFEngine Enterprise 3.26.0, 3.24.2, 3.21.7, and earlier versions. The issues have been fixed in 3.27.0, 3.24.3, 3.21.8, and later versions. If you are using one of the affected versions, we recommend upgrading.

Our documentation has details on how to upgrade CFEngine:

https://docs.cfengine.com/docs/lts/getting-started/01-installation/upgrading

## CVE-2026-24712 - Missing input sanitization / command injection in CFEngine policy language

In CFEngine policy language, you can write policy to manage files, users, packages, groups, and so on. Each of these resources are managed by their respective promise types. You'd expect that `files` promises can only be used to manage files, `users` only to manage users and so on. Normally, the risk of someone "injecting" commands into policy is not a concern, it's a programming language and there is a `commands` promise type for running arbitrary commands.

However, it can matter in some situations - if the policy uses data from an external source, and you expect it to only do one thing. The external source might have some access control mechanisms and assumptions, for example giving somone access to *only* manage users.

Thus, to account for the possibility of such cases (depending on what is in your custom policy) we have identified and fixed cases where it was possible to put (inject) shell commands into places where it should not be possible. Thank you to Dipesh Thakur for discovering and responsibly



the main concern comes with custom policy

manage users, packages, and similar. In

such cases, you might have a system where you expect a user to only be able to delete users (for example), but through external data, an attacker could also run (inject) arbitrary shell commands.

### Remediation

CVE-2026-24712 affects CFEngine 3.26.0, 3.24.2, 3.21.7, and earlier versions. Both CFEngine Enterprise and CFEngine Community are affected. It has been fixed in 3.27.0, 3.24.3, 3.21.8, and later versions. If you are using one of the affected versions, we recommend upgrading.

Our documentation has details on how to upgrade CFEngine:

https://docs.cfengine.com/docs/lts/getting-started/01-installation/upgrading

## Contact

For help with upgrading or additional questions, please contact support at:

https://support.northern.tech

Explore posts by tags

announcements ansible api bash c case-studies
cf-remote cf-runagent cf-secret change-in-behavior cmdb
community compliance containers custom-promise-types cve
design developers docker feature-friday git
guest-blog-posts holiday inventory mission-portal modules
performance policy-analyzer
reporting scalability security
webinars white-papers

Recent posts

CFEngine 3.24.4 and 3.27.1 released

May 8, 2026

May the 4th be with you and your data

May 4, 2026

Show notes: The agent is in - Episode 60 - Improved package management on Enterprise Linux

April 30, 2026