

Nextcloud servers backdoored via flaw in Collabora's built-in CODE app



Attackers exploited a critical flaw in the optional Collabora Online - Built-in CODE Server app to plant web shells on Nextcloud servers. Three administrators reported it on GitHub; the first logged attack was on September 10.

- The affected component is the optional app Collabora Online - Built-in CODE Server, not Nextcloud core.
- Attackers used the flaw to place web shells on Nextcloud servers.
- Three administrators reported the issue on GitHub.
- The first logged attack dates to September 10.
- A web shell runs any command sent in a web request, with the rights of the web server account.

Attackers have exploited a critical flaw in an optional Nextcloud app, Collabora Online - Built-in CODE Server, to install web shells on Nextcloud servers. According to The Hacker News, three administrators reported the problem on GitHub, and the first logged attack occurred on September 10.

Why it matters

A web shell is a small file that executes any command an attacker sends through a web request, using the rights of the web server account. Once one is in place, the attacker has a persistent backdoor on the server, which can remain even if the original flaw is later fixed.

The issue sits in an optional app rather than in Nextcloud itself, so only organisations that installed and enabled the built-in CODE server are likely to be exposed. The source excerpt does not provide further technical detail, a CVE identifier or patch information.

What to do next

Institutions running Nextcloud should first confirm whether this app is installed. If it is, treat the server as potentially compromised until checked, review the vendor and GitHub advisories for the fixed version, and look for unexpected files in web-accessible directories. Because the source does not describe a fix, verify remediation status directly with the vendor before relying on any update alone.

What to do now

1. Check whether Nextcloud servers have the Collabora Online - Built-in CODE Server app installed and enabled.
2. Where it is present, disable or restrict the app until the vendor confirms a fix.
3. Scan web directories and recent file changes on those servers for unknown files that could be web shells.
4. Review web server logs for unusual requests since September 10 and preserve them for investigation.
5. If compromise is suspected, isolate the server, rotate credentials it can access and follow your incident response plan, including notifying the relevant regulator where required.