What is Plex2Shell?

Plex2Shell is the name for a chain of four Plex Media Server vulnerabilities, CVE-2026-96652, CVE-2026-96654, CVE-2026-96655 and CVE-2026-96656, that ends in unauthenticated remote code execution. A blind SSRF in /player/timeline reaches loopback-only plugin APIs, a plugin call leaks the server owner's admin token, and that token sets a transcoder option that writes a shared library into the Codecs folder, which Plex loads on the next transcode. Builds before 1.43.3.10861 are affected. A public exploit appeared on October 16, 2026, and CISA added all four CVEs to its KEV catalog on October 20.

HOW IT WORKS

01 How the chain reaches code execution

Each step hands the attacker a capability the next one needs:

- 1. Blind SSRF from the server itself. `/player/timeline` checks only that an `X-Plex-Token` header exists, not that it is valid, so `X-Plex-Token: junk` passes. Its `protocol` parameter is spliced into a callback URL as `protocol://host:port` without validation, so a full URL makes the server send a POST from 127.0.0.1 to a destination the attacker picks, including its own loopback interface. See server-side request forgery for the general pattern.
- 2. Loopback trust. The internal `/system/agents/*` API skips token checks for loopback callers. In `searchOne`, a broken guard lets an attacker-supplied `identifier` through unencoded into an internal plugin URL, so `?` and `#` characters forge a call to any plugin function with attacker parameters.
- 3. Owner token theft. A bundled metadata agent fetches a URL on the attacker's behalf, and the plugin framework attaches the owner token to any URL whose string starts with `http://127.0.0.1`. A hostname like `127.0.0.1.<attacker-domain>` passes that string check but resolves to the attacker, who receives the server's `local-` admin token.

HOW TO DEFEND

- Update to Plex Media Server 1.43.3.10861 or newer. Check the version under Settings, General in Plex Web. Container users should pull a current image and recreate the container.
- Close the exposure. Turn off Remote Access and remove any port forward for 32400 until you are patched. If you need remote streaming, put the server behind a VPN or an authenticating reverse proxy.
- Hunt before you trust it. Check the Codecs folder, `Preferences.xml` and running processes for the indicators above. Patching does not remove a payload that is already there.
- If you find anything, rebuild. Plex recommends a clean reinstall; at minimum, delete everything in the Codecs folder, remove unexpected transcoder options, sign out all devices and change the Plex account password. Treat everything the Plex service account could read or reach as exposed, including NAS shares mounted for media.
- Watch the edge. A `POST /player/timeline` carrying a full URL in `protocol` is the first request of the chain and a good detection point on a reverse proxy.

SOURCES

- 4. File write into Codecs. With the admin token, `PUT /:/prefs` sets `TranscoderH264Options` to a debug option that dumps decoded frames to a file path inside the Codecs folder. A transcode request to `/video/:/transcode/universal` pointed at an attacker-hosted video, with an ELF hidden in the first frame, writes `x.so` there.
- 5. Library load. The bundled media library `dlopens` every `.so` in that folder with no signature or naming check. The library's constructor runs on the next transcode, as the Plex service user (on Linux, typically `plex`).

The last two steps are a file upload vulnerability in everything but name: the attacker controls where a file lands, and the server later executes it.

02 Who is exposed and what attackers are doing

Any Plex Media Server older than 1.43.3.10861 that an attacker can reach on port 32400 is in scope, whether through router port forwarding, a UPnP mapping or a cloud host. Plex's own advice is blunt: if your server is older than 1.43.3.10861 and reachable from the internet, assume it may be compromised.

Plex reported the flaws as "actively and widely exploited" on October 19, three days after the proof of concept. The payloads seen so far are cryptocurrency miners and proxyware, which rents the victim's internet connection to other people. Both are quiet and profitable, and the same access works just as well for ransomware or for pivoting into whatever else sits on the home or office network.

Indicators Plex lists:

- Processes started by Plex Media Server named `myawesomeproxy`, `mylovelyminer` or `x12345`.
- Unfamiliar `.so` or `.dll` files in the Plex Codecs folder.
- Transcoder options in `Preferences.xml` that you did not set, such as `TranscoderH264Options`.
- Outbound connections to `proxystream[.]st`.
- Sustained high CPU while nothing is playing.

[1] Plex: Important security update for Plex Media Server v1.43.2 and earlier

[2] GitHub Advisory: GHSA-5p7w-r379-3c2m (CVE-2026-96652)

[3] NVD: CVE-2026-96652

[4] NVD: CVE-2026-96656

03 Where Plex2Shell fits in the X2Shell family

Most X2Shell vulnerabilities are one bug in a ubiquitous component, like PHP2Shell in the PHP engine or Node2Shell in a bundled decoder. Plex2Shell is different: four modest bugs in one product, each trusting the step before it. Loopback is treated as an authenticated caller, a string prefix stands in for an address check, and a debug option can write anywhere. The lesson carries over to any server: a moderate SSRF is rarely moderate once you ask what the server trusts from itself.

Find the path to a shell before an attacker does.

securelayer7.net/learn/web-exploitation/what-is-plex2shell

Open online