Postman AI Plugin Codex Report: What the Evidence Shows

TechRounder PDF Edition

Live article: https://www.techrounder.com/ai/postman-ai-plugin-codex-report/

By Vipin PG | Published November 4, 2026 | Updated November 4, 2026 | Format: Analysis | 7 min read

In brief

Postman says it will clarify opt-in after plugin auto-installs. Reports raise consent concerns but do not establish data theft or complete rollback.

A GitHub issue on Postman's app-support repository reports that Postman's desktop app installed Codex and Claude Code plugins during an upgrade on a Mac, without the user asking for them. A Postman log excerpt in the report records the installation. It does not show that any data was sent, and no one has reproduced the behavior in a controlled test. Postman has promised clearer opt-in, but no new consent flow or complete rollback procedure has been documented yet.

If you searched "Postman AI plugin Codex", it helps to separate what the report documents, what Postman's plugin is designed to do, and what remains uncertain. Installation, startup instructions, remote connections and data transmission are four different questions.

What the Postman AI plugin Codex report shows

The October report includes a timestamped log entry, but it covers one user's Mac. A reported installation on one machine does not prove that every Postman user or platform saw the same thing.

In GitHub issue #14370, opened October 25, 2026, the user says Postman Desktop 12.33.4 was followed by Codex and Claude Code plugin installations. A Postman log excerpt records an event at 2026-10-24T09:34:14.418Z, with the source labeled "auto-existing-upgrade" and results naming both agents as installed. This is evidence submitted in the issue, not a lab observation reproduced by the sources used for this article.

The reporter also posted excerpts from the Codex configuration showing a Postman marketplace, an enabled plugin and a trusted hook record. The report describes matching Claude Code settings and install records, plus a Postman CLI executable. These details show what the reporter found on that machine. They do not establish the behavior on clean installations or other operating systems.

The reporter said the sequence had not been reproduced on a clean Mac, and that the evidence shows an unwanted integration rather than data exfiltration. That makes the report a credible starting point for questions about consent and software behavior. It is not proof of a wider security incident.

A second user's account adds context, not a controlled test

In a November 3 comment on the same issue, another user said an existing, signed-in Postman installation updated from 12.31.3 to 12.35.1 and installed the CLI plus Codex and Claude Code plugins. Their pasted log also names "auto-existing-upgrade" and both agent installations.

That makes the report harder to dismiss as a single isolated case. It is still a user-submitted observation, and the sources document no clean-room test that reproduced it independently. The issue has also drawn write-ups from DEV Community and AI Weekly.

The plugin, desktop bundle and MCP are separate components

Postman presents its agent plugin as a way to bring API-development skills and organization context into coding tools, as its plugin documentation explains. That stated purpose does not tell us what a particular desktop update installed or what a user agreed to.

The Postman plugin repository and README describe filesystem-first API work for agents such as Codex, Claude Code and Cursor. They say the installer configures compatible agents it detects, and they document commands to view status and remove the plugin. A separate fallback uses Postman's remote Model Context Protocol (MCP) server when the CLI cannot run. MCP lets an agent call an external service, which is different from a local plugin's files and startup hooks.

Postman's download page also advertises the CLI and plugins alongside the desktop app, and its installation documentation says the CLI is installed with the desktop app. That confirms how the company currently describes its product. It does not show that every user saw a notice before an existing agent configuration changed, because a download page and an in-app update choice are different things.

The Codex startup hook changes the agent's instructions

A startup hook runs instructions when a coding agent begins a session. A hook alone does not prove the agent sent private API data to Postman.

The issue links the hook text from the plugin revision installed on the reporter's machine. It tells Codex to load Postman's API-engineering skill before broad API work. The reporter says this instruction appeared in the agent's configuration and changed how Codex approached API tasks.

Postman's README says its installer approves the Postman session-start hook during Codex installation. For a manual install, the README tells users to review the hook in Codex, and if the Codex hook API is unavailable, the installer continues and tells the user to approve the hook inside Codex. So the automated installation path includes installer-side hook approval, while the manual instructions describe review through Codex.

The reporter also found an MCP connection entry pointing to 'https://mcp.postman.com/mcp', but the cited Codex log said "Auth required." The report shows no authenticated request and does not establish that API data left the machine.

Postman has promised clearer opt-in, not a specified new flow

Postman's response acknowledges the consent issue and says it is reviewing hook behavior. The public statements reviewed do not define the revised prompt, its timing or the affected versions.

A Postman representative commenting on issue #14370 said the CLI and agent plugins are bundled with the desktop app to make setup easier, that the download page and update flow mention them, and that users can uninstall components inside the app. The same response said Postman would make opt-in clearer, especially for existing users, and would review hook approval and session-start behavior.

RuntimeWire reported on November 3 that Postman head of product Sarabmeet Kallar made the opt-in commitment through the company's public-relations agency. The report did not say whether automatic installs would stop, what the revised choice would look like, which versions or platforms were affected, or when a change would ship.

The issue, README and download page reviewed for this article on November 4 showed no testable new consent flow and no release note documenting one. That finding is limited to those pages and does not prove Postman has changed nothing elsewhere.

The evidence supports unwanted installation, not data theft

The reports document configuration changes and a remote MCP entry. They do not demonstrate that API data was transmitted, and the two claims should stay separate in security analysis and in reporting.

The first user says the evidence establishes an unwanted integration rather than exfiltration. The "Auth required" message in the MCP log excerpt is consistent with an unauthenticated or unsuccessful connection attempt, not a completed transfer. The sources include no network capture or independent forensic test.

What the reports do document is narrower, and it still matters: Postman's desktop app was reported to change plugin and instruction settings in other developer tools without the user starting those installations. That raises a consent and trust-boundary question even if no data was sent.

Until network testing or a forensic investigation turns up evidence, claims that Postman intercepted or sent API data should not be repeated as fact. An affected developer can inspect what is configured and use only documented removal instructions, keeping their limits in mind.

How to check your setup and remove the plugin

Postman documents ways to inspect and remove its plugin, and the incident report names files a user can check. Neither source shows that one uninstall command reverses every change.

The report points to Codex's '~/.codex/config.toml' for marketplace, plugin and hook-trust entries. It also describes examining Claude Code settings and installed-plugin records. These are useful places to review, but a snapshot from one Mac is not a complete inventory for every host or version.

Postman's documentation on installing the plugin into detected agents lists 'npx @postman/postman-plugin' to install, 'status' to inspect and 'remove' to uninstall. The '--agent' option limits a command to a selected agent. The README also gives host-specific removal examples, including 'agy plugin uninstall postman' for Antigravity and '/plugins remove postman' for Kimi Code.

These instructions describe what the installer says it does. They do not prove that removal clears every marketplace registration, MCP entry, trusted-hook record, cache or CLI file. Back up your configuration before changing it, check each agent's current documentation, and avoid copying untested cleanup commands from an incident thread.

What a real fix would need

The question is not whether Postman uses the word "opt-in" but whether users choose before the app changes another tool's settings.

A testable consent change should name the agents and components it will install, show an explicit choice before modifying existing users' agent configurations, and say whether the same choice applies to new installations. These criteria follow from the reported log and Postman's stated focus on existing-user opt-in. Postman has not confirmed adopting them.

Hook approval needs its own check: compare the installer’s behavior with the agent host’s trust prompt, and inspect whether the installer writes a hook-trust record. A rollback test should compare plugin enablement, marketplace registration, hook trust, MCP configuration and CLI files before and after removal on each supported host.

Until Postman publishes a specific flow and a complete removal inventory, treat its response as a commitment awaiting verification. The next useful evidence for the Postman AI plugin Codex report is a release note and a repeatable before-and-after test.

Frequently asked questions

Did Postman install Codex plugins without permission?

One user’s log shows Codex and Claude Code plugins installed during an "auto-existing-upgrade" event, and the user says they did not request it. A second user described a similar update. Neither case was independently reproduced.

Did the Postman plugin send my API data anywhere?

The reports do not show that. The MCP log excerpt says "Auth required," and the sources include no network capture or forensic analysis.

How do I remove the Postman plugin?

Postman’s documentation lists ‘npx @postman/postman-plugin remove‘, with ‘--agent‘ to target one agent. Check your configuration files afterward, because the sources do not confirm that this removes every related entry.

Has Postman fixed the issue?

Postman has said it will make opt-in clearer, especially for existing users. As of November 4, the pages reviewed showed no documented new consent flow or release note.

References

1. github.com - postmanlabs / postman-app-support - https://github.com/postmanlabs/postman-app-support/issues/14370

2. dev.to - gabrielanhaia / postman-desktop-added-itself-to-claude-code-and-codex-its-installer-trusts-its-own-hook-2oga - https://dev.to/gabrielanhaia/postman-desktop-added-itself-to-claude-code-and-codex-its-installer-trusts-its-own-hook-2oga

3. aiweekly.co - alerts / postman-desktop-silently-installs-plugins-into-codex-claude-code - https://aiweekly.co/alerts/postman-desktop-silently-installs-plugins-into-codex-claude-code

4. learning.postman.com - docs / postman-plugin - https://learning.postman.com/docs/postman-plugin/overview

5. github.com - postmanlabs / postman-plugin - https://github.com/postmanlabs/postman-plugin

6. modelcontextprotocol.io - https://modelcontextprotocol.io/

7. postman.com - downloads - https://www.postman.com/downloads/

8. learning.postman.com - docs / getting-started - https://learning.postman.com/docs/getting-started/installation/install-app

9. github.com - postmanlabs / postman-plugin - https://github.com/postmanlabs/postman-plugin/blob/6c8e452e0f4237f4bcc0901cb1ac442dec707e49/hooks/session-start-context.md

10. learning.postman.com - docs / postman-plugin - https://learning.postman.com/docs/postman-plugin/install/detected-agents