DATA PRIVACY & GOVERNANCE HUB

OpenAI Codex Authentication Tokens Stolen in Supply Chain Attack

JULY 1, 2026

Key Takeaways

- OpenAI Codex authentication tokens were stolen through a supply chain attack.
- The tool used in the attack was a malicious npm package known as codexui-android.
- Developers were targeted in a malicious supply chain campaign via a seemingly legitimate remote web user interface (UI) version of OpenAI Codex.
- Package supply chain attacks remain a successful way to compromise organizations by targeting trusted tools and dependencies instead of organizations themselves.

What Happened

A malicious Node.js package manager (npm) package, codexui-android, was designed to target developers using OpenAI Codex. npm is a platform for developing JavaScript applications and allows individuals to use prewritten JavaScript modules in their projects. The package was featured on GitHub and npm as a remote web interface for Codex and gained more than 29,000 weekly downloads. Researchers discovered that malicious code was added a month after the release, indicating trust was being built before the attack occurred.

Although the package functioned normally, hidden code secretly scanned and stole users' Codex authentication tokens, transferring them to an attacker-controlled server. The GitHub source code appeared clean, making the malicious activity undetectable. The npm account associated with the package is "fruins" (aka Igor Levochkin).

Additionally, researchers found that an Android application, OpenClaw Codex Claude AI Agent, had the same npm package and forwarded stolen credentials to the same server. The app had

more than 50,000 downloads, while another related Android app, linked to BrutalStrike, had over 10,000 downloads.

Privacy and Governance Concerns

Researchers explained that the main concern is the attackers' ability to maintain access to stolen Codex credentials. Since version 0.1.82, the package covertly sent users' Codex authentication tokens to a remote server controlled by the attacker, masquerading as Sentry. Due to 29,000 weekly npm downloads and over 60,000 mobile app installations, many developers may have been affected. This raises concerns that attackers could gain continuous access to developers' information, such as their accounts, activities, and code.

This incident also raises major concerns and questions regarding transparency. When the issue was reported on GitHub, the npm account owner initially stated that account access had been lost but later modified that statement, claiming the issue was under investigation.

While the account owner stated there was no distribution of credential data to third parties, no clear explanation was given for why the token-collecting code was included and why user credentials were required.

Why It Matters / Policy Considerations

The stolen authentication tokens in the codexui-android npm supply chain attack suggest that current safeguards may not be sufficient. Supply chain attacks often exploit trusted applications, making them difficult to identify and increasing the likelihood that users will install them. Concerns about stolen authentication tokens, which contain sensitive information, have been raised due to the increased use of AI-driven tools.

Organizations can use specific verification techniques, such as multi-factor authentication and shorter token lifetimes, to strengthen safeguards and lower risks. Monitoring tools such as Seceon can help detect unusual access behavior and suspicious token activity.

The incident also highlights the need and significance of transparency and accountability when addressing security breaches. In this case, researchers discovered contradictory explanations from those involved; organizations should provide prompt and unambiguous information

during investigations. Implementing third-party security assessments can also help improve accountability and identify issues that may be overlooked.