Target frame-security-scan

No malicious behavior observed

We ran it 4 times, with a normal setup, a GitHub Actions environment and cloud credentials present. It contacted github.com. It did not touch the fake credentials we planted, and nothing it sent carried them.

Recommendation	Confidence	Findings
No action needed	**n/a**	**0 high 0 med 1 info**

Nothing observed calls for action. This is evidence, not a guarantee.

Scope

Package	frame-security-scan
Source	clawhub 1.0.0
SHA-256	5353595d4fda7fc8a687d215a2310796112d60cb3de2a91cd7abc429b665c329
Files	4
Scanned	26 Oct 2026, 19:35 UTC
Engine	detonator 0.5.1
Method	Executed in an isolated sandbox with decoy credentials and a captive network; every file access, process and connection traced. 4 runs: baseline, ci-github, cloud-creds.
Note	batch:clawhub:2026-10-26

Findings summary

ID	Severity	Confidence	Finding
F01	Low	Info	Network destinations contacted

Observed behavior

Install step	none
Runs as	ls -la; which -a frame; grep -i frame; ls /usr/local/bin /usr/bin; pip show frame; head -2
Network	github.com (2 conn, 418 B sent; while running)
Credentials	no planted credential touched
Files written	nothing outside its own folder

Finding details

F01 Network destinations contacted

Severity: Low Confidence: Info Observed in: llm:baseline
The package connected to these hosts.

Description

Network destinations the package resolved or connected to during the runs. All traffic was captured by the sandbox; nothing reached the internet.

Impact

Context only. Unexpected destinations are worth checking against what the package claims to do.

Evidence

```
github.com
```

Remediation

Confirm each host is expected (the package's API, its registry). Allowlist expected egress in production.

Run log

obedient:baseline ok 1.0s
no new behavior

obedient:ci-github ok 0.9s
no new behavior

obedient:cloud-creds ok 0.8s
no new behavior

llm:baseline ok 16.8s
new behavior: dns:github.com | 4 model steps, 3 commands

Observed behavior only. A clean result does not prove a package is safe. Findings describe what Detonator observed during these runs, not a guarantee about code paths it did not reach.