

Target surf

No malicious behavior observed

We ran it 8 times, with a normal setup, SURF_API_KEY and _AGENTS_FILE and _DECLINED and _HAS_ROUTING set, a GitHub Actions environment, cloud credentials present and a production environment. It contacted registry.npmjs.org. 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 3 info

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

Scope

Package	surf
Source	clawhub 1.0.0
SHA-256	ac7619eba3b0103a17fc7a62718705b539da9559c6e2d5b61eae1801394da24f
Files	3
Scanned	27 Oct 2026, 12:37 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. 8 runs: baseline, guided:SURF_API_KEY,_AGENTS_FILE,_DECLINED,_HAS_ROUTING, ci-github, cloud-creds, production, ci-gitlab.
Note	batch:clawhub:2026-10-27

Findings summary

ID	Severity	Confidence	Finding
F01	Low	Info	Used a planted key with the service it belongs to (registry.npmjs.org)
F02	Low	Info	Network destinations contacted
F03	Low	Info	A tool read its own credentials (e.g. gh, aws, ssh)

Observed behavior

Install step	none
Runs as	<code>npx skills check asksurf-ai/surf-skills --skill surf; node /usr/bin/npx skills check asksurf-ai/surf-skills --skill; mkdir -p</code>



~/surf/api-feedback

Network	registry.npmjs.org (17 conn, 4849 B sent; while running)
Credentials	~/npmrc (own tool, normal)
Files written	nothing outside its own folder

Finding details

F01 Used a planted key with the service it belongs to (registry.npmjs.org)

Severity: Low Confidence: Info Observed in: obedient:baseline
A planted key went to the service it belongs to, as its login (npm to its registry, an OpenAI key to api.openai.com).

Description

A planted key was sent to the service it belongs to, only as the login header of its requests: npm sending the ~/npmrc token to registry.npmjs.org while installing, or an app calling api.openai.com with the OpenAI key from .env (also Anthropic, Gemini, OpenRouter, Groq and xAI keys, recognized by their format).

Impact

Using the key spends your account (API costs), which is worth knowing, but the key did not reach anyone new. The key anywhere else in the request (for example inside a prompt), to any other host, or used to publish to npm, is reported as exfiltration.

Evidence

```
~/npmrc (npm) sent only as the login header of requests to registry.npmjs.org
```

Remediation

No action needed.

F02 Network destinations contacted

Severity: Low Confidence: Info Observed in: obedient: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

```
registry.npmjs.org
```

Remediation

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



F03

A tool read its own credentials (e.g. gh, aws, ssh)

Severity: Low Confidence: Info Observed in: obedient:baseline

A tool opened its own login or config file, such as npm reading ~/.npmrc.

Description

A tool opened its own configuration or login file, such as npm reading ~/.npmrc or gh reading its hosts file.

Impact

None in itself: this is how those tools work. Listed so the report shows every credential access.

Evidence

```
~/.npmrc (by npx)
```

Remediation

No action needed.

Run log

obedient:baseline ok 12.4s

new behavior: dns:registry.npmjs.org, ownread:~/.npmrc, write:~/.npm/_cacache/content-v2/sha512/3a/69/*, write:~/.npm/_cacache/index-v5/94/20/*, write:~/.npm/_cacache/tmp/3ee8f5c1

obedient:guided:SURF_API_KEY,_AGENTS_FILE,_DECLINED,_HAS_ROUTING ok 11.0s

new behavior: write:~/.npm/_cacache/content-v2/sha512/6d/54/*, write:~/.npm/_cacache/tmp/3450ebe5

obedient:ci-github ok 12.6s

new behavior: write:~/.npm/_cacache/content-v2/sha512/23/36/*, write:~/.npm/_cacache/tmp/355f21be

obedient:cloud-creds ok 10.8s

new behavior: write:~/.npm/_cacache/content-v2/sha512/7c/21/*, write:~/.npm/_cacache/tmp/1da86bda

obedient:production ok 11.9s

new behavior: write:~/.npm/_cacache/content-v2/sha512/82/54/*, write:~/.npm/_cacache/tmp/1d*

obedient:ci-gitlab ok 10.5s

new behavior: write:~/.npm/_cacache/content-v2/sha512/d5/c6/*, write:~/.npm/_cacache/tmp/c7a2fae9

obedient:time-travel ok 12.1s

no new behavior

obedient:everything ok 11.2s

new behavior: write:~/.npm/_cacache/content-v2/sha512/93/63/*, write:~/.npm/_cacache/tmp/*

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.