

Target breathing-brand

No malicious behavior observed

We ran it 1 time, with a normal setup. It made no network connections. 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	breathing-brand
Source	clawhub 1.0.0
SHA-256	d6b897461f36a05cb294a18517383bc187e2bed69a99f8b82956e1011ca61b31
Files	10
Scanned	26 Oct 2026, 18:25 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. 1 runs: baseline.
Note	batch:clawhub:2026-10-26

Findings summary

ID	Severity	Confidence	Finding
F01	Low	Info	Loaded the project's .env file

Observed behavior

Install step	none
Runs as	ls -la; ls -R skill; head -50; head -100; cat README.md; grep -v node_modules
Network	no connections
Credentials	./.env (read)
Files written	./brand/_t.md (written)



Finding details

F01 Loaded the project's .env file

Severity: Low Confidence: Info Observed in: llm:baseline
The package loaded the project's .env file.

Description

The process loaded the project's .env file. Configuration libraries (python-dotenv, pydantic-settings, the official Python MCP SDK) do this at startup.

Impact

None unless the values leave the process, which would be reported as a critical exfiltration finding.

Evidence

```
./.env (by sed)
```

Remediation

No action needed. Keep long-lived secrets out of project .env files where possible.

Run log

llm:baseline ok 30.2s

new behavior: ownread:./.env, write:./brand/_t.md | token budget reached

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.