

Target architecture-principles

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	architecture-principles
Source	clawhub 1.0.0
SHA-256	5ec76a5e5c547163cb40fdb3f16d60e7cb8704beb4dfa4bc35441f8269c4200e
Files	3
Scanned	27 Oct 2026, 12:04 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:2026-10-27

Findings summary

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

Observed behavior

Install step	none
Runs as	ls -la; head -50; find ./skill -type f; head -20; cat skill/SKILL.md; ls -R .
Network	no connections
Credentials	./.env (read)
Files written	nothing outside its own folder



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 10.2s

new behavior: ownread:./ .env | 3 model steps, 4 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.