Target zoom-cli

No malicious behavior observed

We ran it 1 time, with a normal setup. It contacted clawhub.ai and registry.npmjs.org. It read some credential files, but nothing it sent carried them.

Recommendation	Confidence	Findings
No action needed	**n/a**	**0 high 0 med 4 info**

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

Scope

Package	zoom-cli
Source	clawhub 1.0.1
SHA-256	af2e26e53b08acfaa30f60c7c324ad66489c3a8bccbdf1558fc4b85ef498d2e3
Files	3
Scanned	27 Oct 2026, 12:54 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	Used a planted key with the service it belongs to (registry.npmjs.org)
F02	Low	Info	Network destinations contacted
F03	Low	Info	Loaded the project's .env file
F04	Low	Info	Agent read credential files (may be the agent's own choice)

Observed behavior

Install step	package manager only, no install scripts
Runs as	which zm; head -40; ls -la; which node npm; node --version; npm --version
Network	clawhub.ai (1 conn, 96 B sent; while running) registry.npmjs.org (9 conn, 2131 B sent; while running)

Credentials	./.env (read) ~/.npmrc (read)
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: llm: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: 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

```
clawhub.ai
```

```
registry.npmjs.org
```

Remediation

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

F03 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.

F04 Agent read credential files (may be the agent's own choice)

Severity: Low Confidence: Info Observed in: llm:baseline
A credential file was opened by a program that does not own it.

Description

A program opened a credential file that it does not own: for example a package script reading ~/.npmrc, ~/.aws/credentials or ~/.ssh/id_rsa, rather than npm, the AWS CLI or ssh reading their own files. The reading program is named in the evidence. In this run the contents were not seen leaving the sandbox.

Impact

Reading credentials is the collection step that precedes theft. Exfiltration may be gated on conditions this run did not meet (a different date, CI variables, a reachable network), or the data may be staged for a later step.

Evidence

```
~/.npmrc (by cat, npm)
```

Remediation

Review why this package needs the file. Legitimate tools read only their own configuration; anything else should be treated as hostile until explained. Prefer running it with credentials unavailable (separate user, container, or scoped environment).

References

- ATT&CK T1552.001 Credentials in Files
- CWE-200 Exposure of Sensitive Information
- OWASP MCP01:2025 Token Mismanagement & Secret Exposure

Run log

llm:baseline ok 66.9s

new behavior: dns:clawhub.ai, dns:registry.npmjs.org, exec:curl, ownread:./.env, read:~/.npmrc | 8 model steps, 8 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.