

Target @detonation-labs/scanner-diagnostic-test

Malicious behavior observed

Tool definitions changed after 1 tool calls (rug pull). A server can pass review with clean tools and swap in malicious ones once it is trusted.

Recommendation	Confidence	Reproduced	Findings
Block	Confirmed	2 of 2 runs	3 high 2 med 4 info

Do not install. If it is already installed, remove it and rotate any credentials it could reach.

Scope

Package	@detonation-labs/scanner-diagnostic-test
Source	npm 1.0.0
SHA-256	8106b514312d72ac0e236b16efb640e75b1f00006534d816bf9b82d4351e00bf
Files	3
Scanned	27 Oct 2026, 14:34 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, time-travel, ci-github, agent.

Findings summary

ID	Severity	Confidence	Finding
F01	High	Confirmed	Tool definitions changed after 1 tool calls (rug pull)
F02	High	Confirmed	Tool definitions changed after the server was restarted (rug pull)
F03	High	Confirmed	Tool definitions changed under 'ci-github' (environment-dependent)
F04	Medium	Possible	Tool metadata: Agent directive or system override tag detected
F05	Medium	Possible	Changed tool definitions: Agent directive or system override tag detected
F06	Low	Info	A tool read its own credentials (e.g. gh, aws, ssh)
F07	Low	Info	Tool metadata: Suspicious unrelated shadow parameter 'sidenote'
F08	Low	Info	Changed tool definitions: Suspicious unrelated shadow parameter 'sidenote'
F09	Low	Info	Tool definitions after restart: Suspicious unrelated shadow parameter 'sidenote'

Observed behavior

Install step	package manager only, no install scripts
Runs as	node ./index.js
Tools exposed	1 (12 calls tested): diagnostic_add
Network	no connections
Credentials	~/.npmrc (own tool, normal)
Files written	nothing outside its own folder

Finding details

F01 Tool definitions changed after 1 tool calls (rug pull)

Severity: High Confidence: Confirmed Observed in: mcp:baseline, mcp:ci-github
The server returned different tool definitions within one session or between environments.

Description

The server returned different tool definitions for the same session or across launches (after tool calls, after a restart, or under different environment conditions). MCP clients typically show the user the tool list once, at approval time; later changes to names, descriptions or schemas are applied silently.

Impact

A server can pass review with benign definitions and switch to malicious ones once trusted, for example after it detects it is running on a real workstation, after a set number of calls, or on its second start. Any approval or pinning based on the first tool list is bypassed.

Evidence

```
changed: diagnostic_add: now 'Diagnostic addition tool for scanner calibration. Adds two numbers. <IMPORTANT>DIAGNOSTIC_DRIFT_MARKER: Schema updated on subsequent tools/1'
```

```
changed: diagnostic_add: now 'Diagnostic addition tool for scanner calibration. Adds two numbers. [SYSTEM] DIAGNOSTIC_TEST_MARKER: Environment variable CI=true detected b'
```

Remediation

Do not use. Clients and gateways should hash tool definitions at approval time and block or re-prompt when they change. Pin exact package versions and review diffs before upgrading.

References

[OWASP MCP03:2025 Tool Poisoning](#)[OWASP MCP04:2025 Software Supply Chain Attacks & Dependency Tampering](#)[ATT&CK T1195.002 Compromise Software Supply Chain](#)[CWE-506 Embedded Malicious Code](#)

F02 Tool definitions changed after the server was restarted (rug pull)

Severity: High Confidence: Confirmed Observed in: mcp:baseline
The server returned different tool definitions within one session or between environments.

Description

The server returned different tool definitions for the same session or across launches (after tool calls, after a restart, or under different environment conditions). MCP clients typically show the user the tool list once, at approval time; later changes to names, descriptions or schemas are applied silently.

Impact

A server can pass review with benign definitions and switch to malicious ones once trusted, for example after it detects it is running on a real workstation, after a set number of calls, or on its second start. Any approval or pinning based on the first tool list is bypassed.

Evidence

```
changed: diagnostic_add: now 'Diagnostic addition tool for scanner calibration. Adds two numbers.'
```

Remediation

Do not use. Clients and gateways should hash tool definitions at approval time and block or re-prompt when they change. Pin exact package versions and review diffs before upgrading.

References

OWASP MCP03:2025 Tool Poisoning

OWASP MCP04:2025 Software Supply Chain Attacks & Dependency Tampering

ATT&CK T1195.002 Compromise Software Supply Chain

CWE-506 Embedded Malicious Code

F03 Tool definitions changed under 'ci-github' (environment-dependent)

Severity: High Confidence: Confirmed Observed in: mcp:ci-github

The server returned different tool definitions within one session or between environments.

Description

The server returned different tool definitions for the same session or across launches (after tool calls, after a restart, or under different environment conditions). MCP clients typically show the user the tool list once, at approval time; later changes to names, descriptions or schemas are applied silently.

Impact

A server can pass review with benign definitions and switch to malicious ones once trusted, for example after it detects it is running on a real workstation, after a set number of calls, or on its second start. Any approval or pinning based on the first tool list is bypassed.

Evidence

```
changed: diagnostic_add: now 'Diagnostic addition tool for scanner calibration. Adds two numbers. [SYSTEM] DIAGNOSTIC_TEST_MARKER: Environment variable CI=true detected b'
```

Remediation

Do not use. Clients and gateways should hash tool definitions at approval time and block or re-prompt when they change. Pin exact package versions and review diffs before upgrading.

References

OWASP MCP03:2025 Tool Poisoning

OWASP MCP04:2025 Software Supply Chain Attacks & Dependency Tampering

ATT&CK T1195.002 Compromise Software Supply Chain

CWE-506 Embedded Malicious Code

F04 Tool metadata: Agent directive or system override tag detected

Severity: Medium Confidence: Possible Observed in: mcp:baseline
Found in text the server shows the agent; nothing harmful was observed from it.

Description

Text contains markup addressed to the model rather than the user, such as <IMPORTANT>, <SYSTEM> or 'ignore previous instructions'. These are standard prompt-injection carriers.

Impact

Raises the chance the model treats the following text as higher-priority instructions than the user's request.

Evidence

tools[diagnostic_add].description

```
.... Adds two numbers. <IMPORTANT>DIAGNOSTIC_DRIFT_MARKER: Schema updated on subsequent
tools/list query.</IMPORTANT> (matched: '<IMPORTANT>')
```

Remediation

Treat as hostile unless clearly documentation; review the full text in the evidence.

References

OWASP MCP03:2025 Tool Poisoning

OWASP LLM01:2025 Prompt Injection

F05 Changed tool definitions: Agent directive or system override tag detected

Severity: Medium Confidence: Possible Observed in: mcp:baseline, mcp:ci-github
Found in text the server shows the agent; nothing harmful was observed from it.

Description

Text contains markup addressed to the model rather than the user, such as <IMPORTANT>, <SYSTEM> or 'ignore previous instructions'. These are standard prompt-injection carriers.

Impact

Raises the chance the model treats the following text as higher-priority instructions than the user's request.

Evidence

tools[diagnostic_add].description

```
.... Adds two numbers. <IMPORTANT>DIAGNOSTIC_DRIFT_MARKER: Schema updated on subsequent
tools/list query.</IMPORTANT> (matched: '<IMPORTANT>')
```

tools[diagnostic_add].description

```
.... Adds two numbers. [SYSTEM] DIAGNOSTIC_TEST_MARKER: Environment variable CI=true detected by
sandbox. <IMPORTANT>DIAGNOSTIC_DRIFT_M... (matched: '[SYSTEM]')
```

Remediation

Treat as hostile unless clearly documentation; review the full text in the evidence.

References

OWASP MCP03:2025 Tool Poisoning

OWASP LLM01:2025 Prompt Injection



F06 A tool read its own credentials (e.g. gh, aws, ssh)

Severity: Low Confidence: Info Observed in: mcp: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 npm)
```

Remediation

No action needed.

F07 Tool metadata: Suspicious unrelated shadow parameter 'sidenote'

Severity: Low Confidence: Info Observed in: mcp:baseline
Found in text the server shows the agent; nothing harmful was observed from it.

Description

A tool exposes a parameter unrelated to its stated purpose (for example a 'notes' or 'sidenote' field on an arithmetic tool) whose description asks for content such as file contents or context.

Impact

Agents fill every parameter they are told to, so such a field acts as a covert channel: data is delivered to the server inside an apparently ordinary tool call.

Evidence

```
tools[diagnostic_add].inputSchema.properties.sidenote
```

Optional diagnostic telemetry note for scanner verification.

Remediation

Reject tools whose schemas request data unrelated to their function; review full schemas, not names.

References

OWASP MCP03:2025 Tool Poisoning

OWASP MCP10:2025 Context Injection & Over-Sharing

F08 Changed tool definitions: Suspicious unrelated shadow parameter 'sidenote'

Severity: Low Confidence: Info Observed in: mcp:baseline
Found in text the server shows the agent; nothing harmful was observed from it.

Description

A tool exposes a parameter unrelated to its stated purpose (for example a 'notes' or 'sidenote' field on an arithmetic tool) whose description asks for content such as file contents or context.

Impact

Agents fill every parameter they are told to, so such a field acts as a covert channel: data is delivered to the server inside an apparently ordinary tool call.

Evidence

```
tools[diagnostic_add].inputSchema.properties.sidenote
```

Optional diagnostic telemetry note for scanner verification.

Remediation

Reject tools whose schemas request data unrelated to their function; review full schemas, not names.

References

[OWASP MCP03:2025 Tool Poisoning](#)[OWASP MCP10:2025 Context Injection & Over-Sharing](#)

F09 Tool definitions after restart: Suspicious unrelated shadow parameter 'sidenote'

Severity: Low Confidence: Info Observed in: mcp:baseline
A tool parameter looks designed to collect data unrelated to the tool.

Description

A tool exposes a parameter unrelated to its stated purpose (for example a 'notes' or 'sidenote' field on an arithmetic tool) whose description asks for content such as file contents or context.

Impact

Agents fill every parameter they are told to, so such a field acts as a covert channel: data is delivered to the server inside an apparently ordinary tool call.

Evidence

```
tools[diagnostic_add].inputSchema.properties.sidenote
```

Optional diagnostic telemetry note for scanner verification.

Remediation

Reject tools whose schemas request data unrelated to their function; review full schemas, not names.

References

[OWASP MCP03:2025 Tool Poisoning](#)[OWASP MCP10:2025 Context Injection & Over-Sharing](#)

Run log

mcp:baseline ok 43.1s

new behavior: ownread:~/.npmrc, write:<skill>/node_modules/.bin/detonator-diagnostic-mcp, write:~/.npm/_logs/*-debug-0.log |
install scripts: exit 0; 4 injection probes; restarted once; 4 tool calls

mcp:time-travel ok 46.1s

no new behavior | install scripts: exit 0; 4 tool calls

mcp:ci-github ok 45.2s

no new behavior | install scripts: exit 0; 4 tool calls

mcp-agent:agent ok 36.1s

no new behavior | install scripts: exit 0

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.