OWASP MCP Top 10

Compliance assessment — 2025

Server: io.github.pinchwork/pinchwork

Slug: io-github-pinchwork-pinchwork
Scan id: 36df3f55-2eae-4732-89d4-1240a542bb1f

Assessed at: 2026-05-31 07:52:35 UTC
Sentinel version: 0.4.0
Rules version: 2026-04-23

' Non-compliant

DRAFT for review — not legal advice. See attestation block for verification instructions.

DRAFT

Table of contents

1. Executive summary
2. Coverage & transparency
3. Controls summary
4. Control details
5. Multi-step attack chains
6. Cryptographic attestation

1. Executive summary

Assessment of io.github.pinchwork/pinchwork against OWASP MCP Top 10: overall status non compliant. Of 10 controls, 0 met, 2 unmet, 0 partial, 8 not assessed, 0 not applicable. 2 control(s) had complete mapped-assessor execution evidence; 8 lacked sufficient execution evidence and 0 were outside the current assessor mapping. Unmet controls have findings at or above the framework's mandatory severity threshold and should be remediated before relying on this server in a regulated deployment. All claims are traceable to individual finding rows via finding_id and to the governing rule via rule_id; the enclosing signed envelope commits MCP Sentinel to the exact bytes of this report.

2. Coverage & transparency

Coverage band: minimal

Coverage ratio: 6%

Rules version: 2026-04-23

Analysis techniques applied:

- composite
- dependency-audit
- similarity
- structural

3. Controls summary

ControlName
Status
Evidence

MCP01Prompt Injection? Not assessed0

MCP02Tool Poisoning' Unmet1

MCP03 Command

Injection? Not assessed0

MCP04Data Exfiltration? Not assessed0

MCP05 Privilege Escalation? Not assessed0

MCP06 Excessive Permissions? Not assessed0

MCP07 Insecure Configuration ? Not assessed0

MCP08 Dependency Vulnerabilities ? Not assessed0

MCP09 Logging & Monitoring Failures? Not assessed0

MCP10Supply Chain Compromise' Unmet1

4. Control details

MCP01 — Prompt Injection

? Not assessed

0/24 mapped assessor rule(s) are proven to have executed; no adverse findings were observed, but the control is not assessed because execution evidence is incomplete. Unexecuted assessors: A1 (execution-ledger); A5 (execution-ledger); A7 (execution-ledger); A8 (execution-ledger); A9 (execution-ledger); B5 (execution-ledger); F1 (execution-ledger); F6 (execution-ledger); G1 (execution-ledger); G2 (execution-ledger); G3 (execution-ledger); G4 (execution-ledger); G5 (execution-ledger); H2 (execution-ledger); H3 (execution-ledger); I3 (execution-ledger); I6 (execution-ledger); I7 (execution-ledger); J3 (execution-ledger); J5 (execution-ledger); J6 (execution-ledger); N4 (execution-ledger); N9 (execution-ledger); N12 (execution-ledger).

MCP02 — Tool Poisoning

' Unmet

0/15 mapped assessor rule(s) are proven to have executed; 1 finding(s) observed (1 critical); at least one finding is at or above the high threshold (status: unmet). 15 mapped assessor(s) did not execute, so this finding-backed result does not imply complete control coverage.

Evidence:

[Critical]
Official
Namespace

Squatting
(finding ae2728ff
-7b96-4c68-bf38-
c72ea9b409e0,
confidence 90%)

SOURCE: external-content at initialize.server_name — The MCP client surfaces the server name verbatim in its approval dialog, and the LLM ingests the server name alongside the tool descriptions. A nam

Required mitigations:

- If you own the server and are NOT affiliated with the vendor whose namespace it contains, rename the server to remove the vendor token. Choose a name that makes your actual publisher identity clear. If you ARE a vendor-approved partner and intentionally use the vendor's namespace, request inclusion in the rule's OFFICIAL_NAMESPACES.verified_github_orgs list by publishing the server under a vendor-sanctioned GitHub organisation. Users deciding whether to approve the server should check the repository owner against the vendor's published list of approved partners before granting trust.

MCP03 — Command Injection

? Not assessed

0/6 mapped assessor rule(s) are proven to have executed; no adverse findings were observed, but the control is not assessed because execution evidence is incomplete. Unexecuted assessors: C1 (execution-ledger); C9 (execution-ledger); C13 (execution-ledger); C16 (execution-ledger); J2 (execution-ledger); J7 (execution-ledger).

MCP04 — Data Exfiltration

? Not assessed

0/15 mapped assessor rule(s) are proven to have executed; no adverse findings were observed, but the control is not assessed because execution evidence is incomplete. Unexecuted assessors: A3 (execution-ledger); F1 (execution-ledger); F3 (execution-ledger); F7 (execution-ledger); G7 (execution-ledger); I9 (execution-ledger); I13 (execution-ledger); K18 (execution-ledger); O4 (execution-ledger); O5 (execution-ledger); O6 (execution-ledger); O8 (execution-ledger); O9 (execution-ledger); O10 (execution-ledger); P3 (execution-ledger).

MCP05 — Privilege Escalation

? Not assessed

0/7 mapped assessor rule(s) are proven to have executed; no adverse findings were observed, but the control is not assessed because execution evidence is incomplete. Unexecuted assessors: C2 (execution-ledger); C8 (execution-ledger); C10 (execution-ledger); C12 (execution-ledger); I4 (execution-ledger); I12 (execution-ledger); J1 (execution-ledger).

MCP06 — Excessive Permissions

? Not assessed

0/7 mapped assessor rule(s) are proven to have executed; no adverse findings were observed, but the control is not assessed because execution evidence is incomplete. Unexecuted assessors: A2 (execution-ledger); B3 (execution-ledger); B7 (execution-ledger); E4 (execution-ledger); F2 (execution-ledger); I11 (execution-ledger); I16 (execution-ledger).

MCP07 — Insecure Configuration

? Not assessed

0/11 mapped assessor rule(s) are proven to have executed; no adverse findings were observed, but the control is not assessed because execution evidence is incomplete. Unexecuted assessors: B6 (execution-ledger); C7 (execution-ledger); C8 (execution-ledger); C11 (execution-ledger); C14 (execution-ledger); C15 (execution-ledger); D6 (execution-ledger); E1 (execution-ledger); E2 (execution-ledger); I15 (execution-ledger); J4 (execution-ledger).

MCP08 — Dependency Vulnerabilities

? Not assessed

0/7 mapped assessor rule(s) are proven to have executed; no adverse findings were observed, but the control is not assessed because execution evidence is incomplete. Unexecuted assessors: D1 (execution-ledger); D2 (execution-ledger); D3 (execution-ledger); D4 (execution-ledger); D5 (execution-ledger); D6 (execution-ledger); D7 (execution-ledger).

MCP09 — Logging & Monitoring Failures

? Not assessed

0/6 mapped assessor rule(s) are proven to have executed; no adverse findings were observed, but the control is not assessed because execution evidence is incomplete. Unexecuted assessors: C6 (execution-ledger); E3 (execution-ledger); K1 (execution-ledger); K2 (execution-ledger); K3 (execution-ledger); K20 (execution-ledger).

MCP10 — Supply Chain Compromise

' Unmet

0/27 mapped assessor rule(s) are proven to have executed; 1 finding(s) observed (1 critical); at least one finding is at or above the high threshold (status: unmet). 27 mapped assessor(s) did not execute, so this finding-backed result does not imply complete control coverage.

Evidence:

[Critical]
Official
Namespace
Squatting
(finding ae2728ff
-7b96-4c68-bf38-
c72ea9b409e0,
confidence 90%)

SOURCE: external-content at initialize.server_name — The MCP client surfaces the server name verbatim in its approval dialog, and the LLM ingests the server name alongside the tool descriptions. A nam

Required mitigations:

- If you own the server and are NOT affiliated with the vendor whose namespace it contains, rename the server to remove the vendor token. Choose a name that makes your actual publisher identity clear. If you ARE a vendor-approved partner and intentionally use the vendor's namespace, request inclusion in the rule's OFFICIAL_NAMESPACES.verified_github_orgs list by publishing the server under a vendor-sanctioned GitHub organisation. Users deciding whether to approve the server should check the repository owner against the vendor's published list of approved partners before granting trust.

5. Multi-step attack chains

No multi-step attack chains were synthesized for this server.

6. Cryptographic attestation

Algorithm: `HMAC-SHA256`
Key ID: `mcps-hkdf-e8990a4e1526`
Signer: `mcp-sentinel/v1`
Signed at: `2026-05-31T07:52:35.407Z`
Canonicalization: `RFC8785`

HMAC-SHA256 signature (base64, wrapped at 64 chars):

```
VoRHbw58xWj5neKFOEsK+OvFtXFaPkSZ+MKD3fEMgwc=
```

Verification instructions:

To verify this report:

1. Extract the report body (everything except the .attestation field).
2. Canonicalize the body via RFC 8785 (JCS).
3. Compute HMAC-SHA256 with the signing key for key_id "mcps-hkdf-e8990a4e1526".
4. Base64-encode the result and compare with the signature above.