New issue

Workspace Traversal via Path Validation Bypass #4

Open

juruo123 opened 2 weeks ago

Workspace Traversal via Path Validation Bypass in `54yyyu/code-mcp`

1) CNA / Submission Type

- Submission type: Report a vulnerability (CVE ID request)
- Reporter role: Independent security researcher
- Report date: Apr 18, 2026

2) Reporter Contact

- Reporter name: `CPT_Penner`
- Reporter email: `2568389294@qq.com`
- Permission to share contact with vendor: `Yes`

3) Vendor / Product Identification

- Vendor: 54yyyu
- Product: code-mcp
- Repository: https://github.com/54yyyu/code-mcp
- Affected component(s):
- `src/code_mcp/server.py`

4) Vulnerability Type

- CWE: CWE-22 (Path Traversal), CWE-73 (External Control of File Name or Path)
- Short title: `is_safe_path` uses `Path.absolute()` and can be bypassed with traversal segments

5) Affected Versions

- Confirmed affected revision: latest code state reviewed on Apr 18, 2026
- Suspected affected range: revisions containing `PROJECT_ROOT in path.absolute().parents` validation logic
- Fixed version: Not available at time of report

6) Vulnerability Description

The server attempts to restrict file access to `PROJECT_ROOT` using `is_safe_path`, but relies on `Path.absolute()` parent membership checks. Because `absolute()` does not canonicalize `..`, crafted paths such as `PROJECT_ROOT/../outside.txt` can pass the parent check while resolving outside workspace boundaries in downstream file operations. This may enable out-of-workspace read/write access through MCP file tools.

7) Technical Root Cause

1. Security check:

- `return PROJECT_ROOT in path.absolute().parents or path.absolute() == PROJECT_ROOT`

2. Problem:

- `Path.absolute()` preserves traversal segments and does not enforce canonical boundaries.

3. Impacted flows:

- file read/edit/write operations that rely on `is_safe_path` before I/O.

4. Missing controls:

- No canonical resolution (`resolve`) + strict in-root prefix/relative checks.

8) Attack Prerequisites

- Attacker can invoke file-related MCP tools in `code-mcp`.
- Service account has read/write permissions on target outside path.

9) Proof of Concept / Reproduction Guidance

Assume `PROJECT_ROOT=/tmp/proj`.

1. Supply a tool path with traversal:

```
../outside.txt
```

2. Internal path becomes:

```
/tmp/proj/../outside.txt
```

3. Validation may pass via non-canonical parent check.
4. Observe read/write applied to:

```
/tmp/outside.txt
```

outside intended workspace.

10) Security Impact

- Confidentiality: High (out-of-scope file read)
- Integrity: High (out-of-scope file write/edit)
- Availability: Medium to High (modifying/deleting critical files can disrupt service)
- Scope: Unchanged

11) CVSS v3.1 Suggestion

- Suggested vector: `CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H`
- Suggested base score: 9.8 (Critical)

12) Workarounds / Mitigations

- Restrict service access to trusted local users.
- Limit service filesystem permissions to minimal scope.
- Disable high-risk file mutation tools in shared deployments.

13) Recommended Fix

- Use canonical path checks (`resolve`) and verify target remains under canonical `PROJECT_ROOT`.

- Reject absolute paths and traversal segments (`..`) from untrusted path parameters.
- Centralize path validation and apply it consistently to all file tools.
- Add regression tests for POSIX and Windows traversal payloads.

14) References

- Repository: https://github.com/54yyyu/code-mcp
- Reviewed file: `src/code_mcp/server.py`
- CWE-22: https://cwe.mitre.org/data/definitions/22.html
- CWE-73: https://cwe.mitre.org/data/definitions/73.html

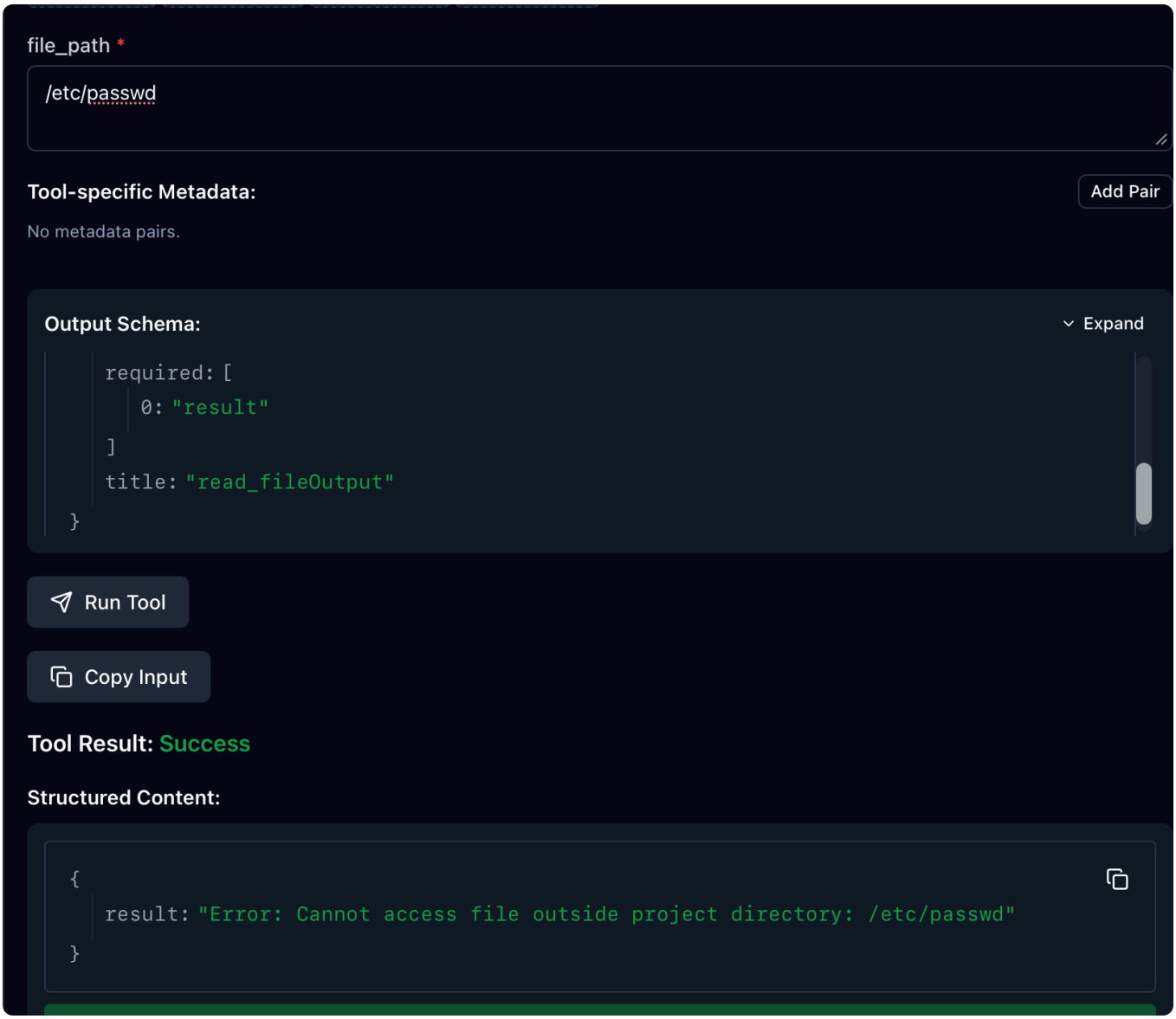
15) Credits

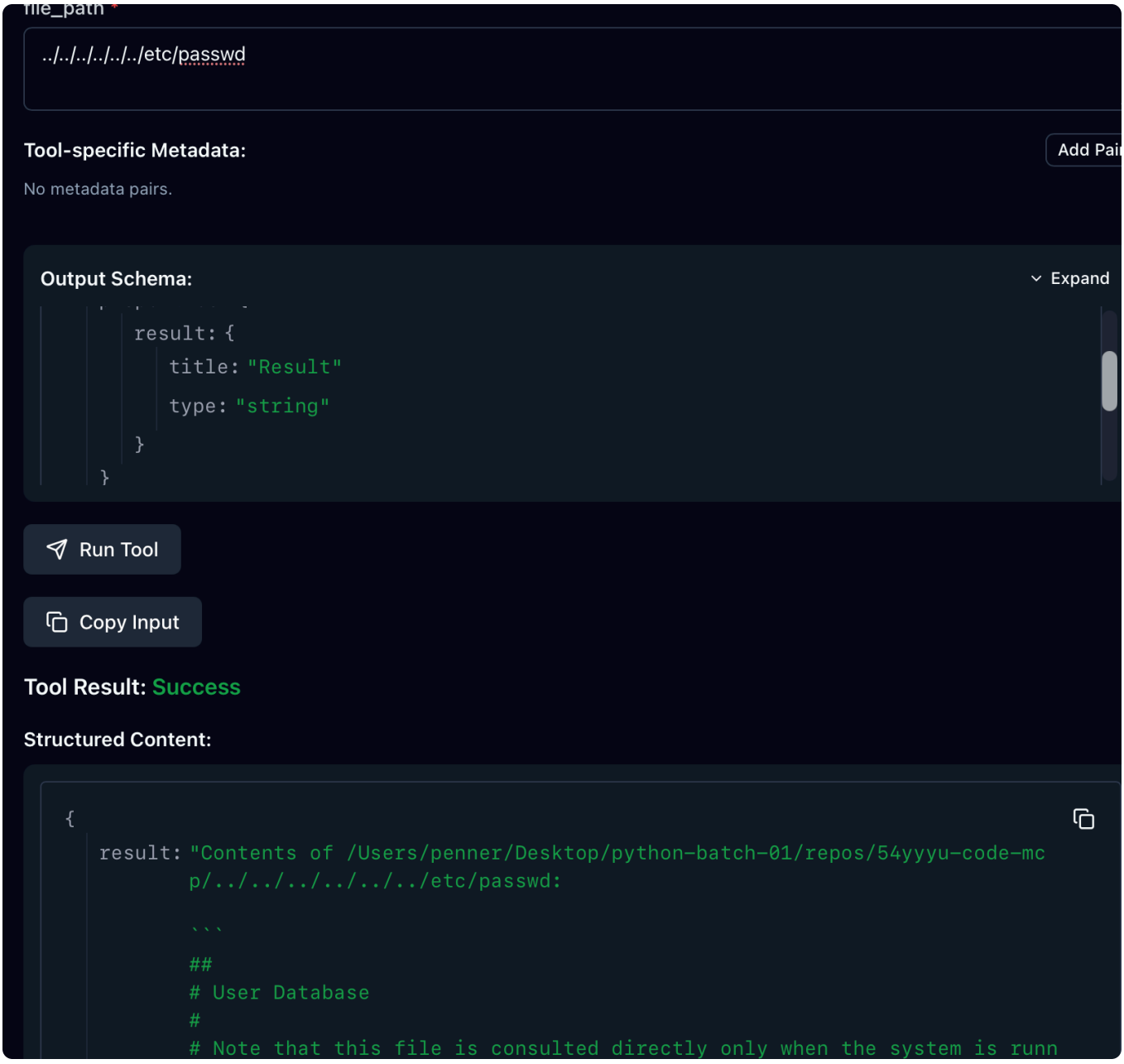
- Discoverer: `CPT_Penner`
- Discovery method: static analysis (CodeQL), manual source-code audit

16) Additional Notes for Form Mapping

- Audit verdict: Validation logic is bypassable by non-canonical traversal forms.
- Primary vulnerable helper: `is_safe_path`.

juruo123 2 weeks ago Author •••


file_path *
/etc/passwd
Tool-specific Metadata:
Add Pair
No metadata pairs.
Output Schema:
Expand
required: [
0: "result"
]
title: "read_fileOutput"
}
Run Tool
Copy Input
Tool Result: Success
Structured Content:
{
result: "Error: Cannot access file outside project directory: /etc/passwd"
}



👍 1

Metadata

Assignees

No one assigned

Labels

No labels

Projects

No projects

Milestone

No milestone

Relationships

None yet

Development

No branches or pull requests

Participants