COMMUNITY CONTENT

Home Submit 793438

Submit #793438: Daniel Miessler Personal AI Infra (PAI) 2.3.0 Command Injection

Title Daniel Miessler Personal AI Infra (PAI) 2.3.0 Command Injection

Description

Security Advisory: Command Injection in Daniel Miessler's Personal AI Infrastructure (PAI)

Date: March 12, 2026
Researcher: David Gilmore
Project: danielmiessler/Personal_AI_Infrastructure (PAI)
Affected Version: v2.3.0 and prior
Vulnerability Type: Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection') (CWE-78)
Severity: Critical (CVSS v3.1: 9.8)

1. EXECUTIVE SUMMARY

A critical security vulnerability was identified in the Parser skill of the Personal AI Infrastructure (PAI). The vulnerability allows for remote command injection when the system processes a maliciously crafted URL. Because PAI is designed to scrape and analyze web content as a core function, an attacker can trigger this vulnerability by simply providing a URL containing shell metacharacters, leading to full compromise of the host system.

2.1 Technical Description

The vulnerability is located in the tool-calling logic of the Parser skill. The application uses a shell-based approach to execute curl for web scraping. Specifically, the user-provided URL is directly interpolated into a template string that is then executed via a shell command without sufficient sanitization.

Affected Component: Skills/Parser/Tools/parse_url.ts (or equivalent shell script in earlier versions)

Root Cause: Use of shell-interpolated strings for executing external CLI tools like curl or yt-dlp.

2.2 Vulnerable Pattern

```
// Vulnerable Logic
const { stdout } = await exec(`curl -L "${userProvidedUrl}"`);
```

Community Content

Submissions are made by VulDB community users. VulDB is *not responsible* for their content nor the links to external sources.

Please use the raw information shown and the links listed *with caution*. They might contain malicious and harmful actions, code or data.

The corresponding VulDB entries contain the moderated, verified, and normalized information provided within the raw submission.

Documentation

- Submission Policy
- Data Processing
- CVE Handling

An attacker can provide a URL such as https://example.com"; rm -rf / # to terminate the curl command and execute arbitrary subsequent commands.

3. PROOF OF CONCEPT (PoC)

3.1 Reproduction Steps

Start the PAI environment.

Provide the following command to the assistant:
Analyze the content of this URL: https://google.com"; touch /tmp/pai_vulnerable #

The AI agent invokes the Parser skill.

The underlying shell executes: curl -L "https://google.com"; touch /tmp/pai_vulnerable #"

Result: The file /tmp/pai_vulnerable is created on the host system, confirming arbitrary code execution.

4. IMPACT ASSESSMENT

Successful exploitation allows a remote attacker (or a malicious document containing a URL) to:

Host Compromise: Gain full shell access to the machine running PAI.

Data Exfiltration: Access the PAI_DIRECTORY, which contains personal context, life goals (Telos), and sensitive history.

Lateral Movement: Use the PAI host as a jumping-off point to attack other devices on the local network.

5. REMEDIATION RECOMMENDATIONS

5.1 Immediate Fix

Replace shell execution with structured execution that does not involve a shell interpreter. Use execFile or spawn where arguments are passed as an array.

Secure Implementation:

```
import { execFile } from 'child_process';
// The URL is passed as a separate argument, preventing shell interpretation
const { stdout } = await execFile('curl', ['-L', validatedUrl]);
```

5.2 Strategic Mitigations

URL Validation: Implement a strict regex-based allowlist for URLs and block internal IP ranges (SSRF protection).

Native APIs: Use native HTTP libraries (like fetch or axios) instead of calling out to system binaries like curl.

Sandbox Environment: Run PAI skills within a containerized or sandboxed environment with restricted filesystem access.

This report was generated following successful disclosure to Daniel Miessler.

https://github.com/danielmiessler/Personal_AI_Infrastructure/pull/659

Source	https://github.com/danielmiessler/Personal_AI_Infrastructure/pull/659
User	davidgilmore (UID 96940)
Submission	03/31/2026 07:51 AM (13 days ago)
Moderation	04/12/2026 09:43 AM (12 days later)
Status	Accepted
VulDB entry	[danielmiessler Personal_AI_Infrastructure up to 2.3.0 parse_url.ts os command injection]
Points	20