COMMUNITY CONTENT

Home › Submit › 784462

Submit #784462: premsql v0.2.1 Code Injection

Title premsql v0.2.1 Code Injection

Description A Remote Code Execution (RCE) vulnerability exists in the premsql library due to the unsafe usage of eval() on language model outputs. An attacker can use prompt injection to force the LLM to output malicious Python code, which is then executed by the server.

```python
    try:
        result = self.generator.generate(
            data_blob={"prompt": prompt},
            temperature=temperature,
            max_new_tokens=max_new_tokens,
            postprocess=False,
        )
        # VULNERABILITY HERE:
        result = eval(result.replace("null", "None"))
        error_from_model = None
        assert "alternate_decision" in result
        assert "suggestion" in result
```

The `result` variable contains the raw string output from the LLM. The application attempts to parse this as a Python dictionary using `eval()`. However, if the LLM output is manipulated to contain valid Python commands (e.g., `__import__('os').system('calc')`), `eval()` will execute them.

Source https://github.com/Ka7arotto/cve/blob/main/premsql-rce/issue.md

User Goku (UID 80486)

Submission 03/21/2026 02:50 AM (16 days ago)

Moderation 04/05/2026 07:12 AM (15 days later)

Status Accepted

VulDB entry [premAI-io premsql up to 0.2.1 followup.py eval result code injection]

Points 20

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