COMMUNITY CONTENT

Submit #782107: Ollama 18.1 and previous Server-Side Request Forgery

Title	Ollama 18.1 and previous Server-Side Request Forgery
Description	Server-Side Request Forgery (SSRF) in Ollama's model pull API allows an attacker to read HTTP responses from internal services on the host running Ollama. When a user or application pulls a model from a malicious OCI registry, the registry can respond with a 307 redirect pointing blob downloads to arbitrary internal URLs such as http://127.0.0.1:6379/INFO or http://x.x.x.x/latest/meta-data/. Ollama follows this redirect, fetches the internal resource, and writes the response to a blob file on disk. The attacker can then retrieve the captured data by pushing the model back to an attacker-controlled registry. The vulnerability exists in server/download.go, which uses the Location header from an attacker-controlled redirect response as the download URL without validating the target against private or reserved IP ranges. Any Ollama deployment where the API is network-accessible is affected, including Docker containers which bind to x.x.x.x and run as root by default. An attacker with network access to the Ollama API can read responses from localhost-bound services such as databases, admin panels, key-value stores, cloud metadata endpoints, and internal APIs. Confirmed on Ollama v0.18.1 (latest stable release) and 0.13.5. The vulnerable code is unchanged on the current main branch. I have reached out to the vendor about this vulnerability and am awaiting their response.
User	davidrochester (UID 94063)
Submission	03/18/2026 05:52 AM (18 days ago)
Moderation	04/04/2026 08:29 AM (17 days later)
Status	Pending
VulDB entry	[Ollama up to 18.1 Model Pull API server/download.go server-side request forgery]
Points	17

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