OAuth 2.0 Scope Aggregation for Multi-Step AI Agent Workflows

https://datatracker.ietf.org/doc/draft-jia-oauth-scope-aggregation/

Presenter: Yukuan Jia

jiayukuan@huawei.com

On behalf of project champions: Yukuan Jia, Shuping Peng

Background

- Advances in LLM enables AI agents to plan and execute multi-step workflow to accomplish tasks

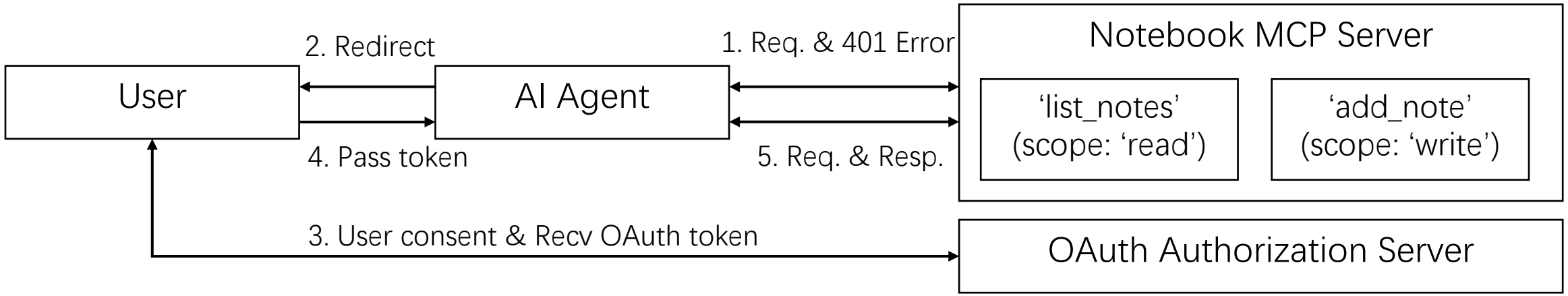
- With agent communication protocols, an AI agent can
  - call third-party tools (e.g., via MCP)
  - delegate sub-tasks to other AI agents (e.g., via A2A)

- These interoperations typically require authorization, by OAuth 2.0

Agent2Agent (A2A) Protocol

Scenario & Problem

- **Scenario**: An AI agent connected to a notebook MCP Server
  - User → AI agent: 'Summarize this week's daily reports into a weekly report note.'
  - AI agent (reasons): First call 'list_notes', then call 'add_note'.
- **Problem:** Existing agent protocols (e.g. MCP) adopt OAuth in a reactive and challenge-triggered way
  - User needs to respond to multiple consent prompts with distinct scopes, causing fatigue

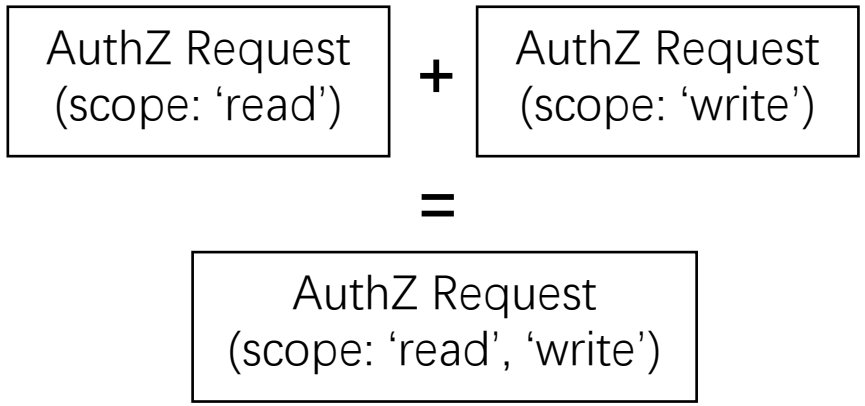


Solution

1. The server exposes the security requirement in the resource metadata

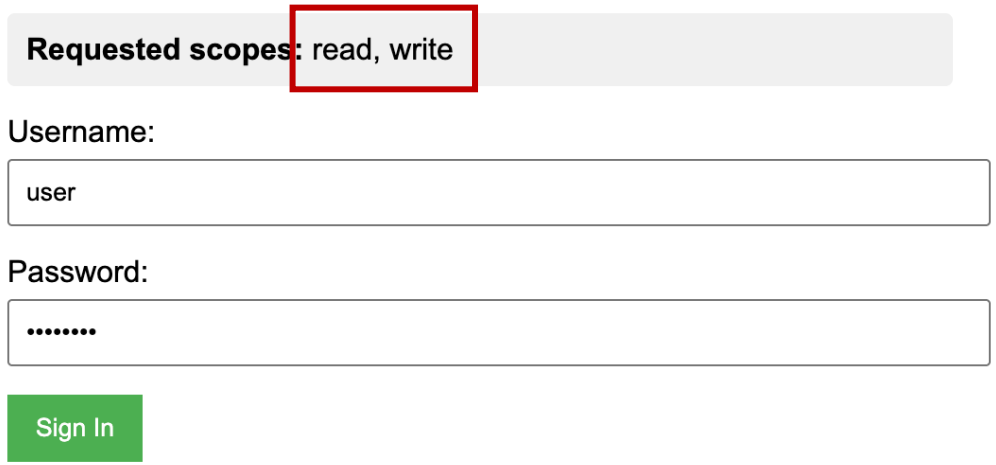
```
{
    "name": "add_note",
    "description": "...",
    "input_schema": { ... },
    "security": {
      "type": ["oauth2"],
      "scopes": ["write"],
      "as_metadata": "https://notebook.example.com/.well-known/..."
    }
}
```

2. The AI agent initiates a single OAuth flow for the aggregated scopes



Demo: Notebook MCP Server

- Authorization UI




```
============================================================
MCP Notebook Client - Aggregated Scope Authorization
============================================================
Server: http://localhost:8001/mcp
Auth Server: http://localhost:9000

Started callback server on http://localhost:3030

[Step 1] Discovering tool scope requirements...
  (Calling public tools/list to discover requirements)
  Found 4 tools:
    - list_notes: requires scopes ['read']
    - read_note: requires scopes ['read']
    - add_note: requires scopes ['write']
    - edit_note: requires scopes ['write']

  Aggregated scopes: ['read', 'write']

[Step 2] Requesting authorization with scopes: ['read', 'write']
Opening browser for authorization: http://localhost:9000/authorize?response_type=code&client_id=
800d91eb-a17a-44a7-ae0f-d683cba58a7c&redirect_uri=http%3A%2F%2Flocalhost%3A3030%2Fcallback&state
=Uu_a8QhJ25CWX2Y-a9ksoMgk3WcW1N2T1BTDMOBan_0&code_challenge=6s2_DLrD9wUp-YbEujpSVz9YL04T7pwTmphp
xV0hsh4&code_challenge_method=S256&resource=http%3A%2F%2Flocalhost%3A9000&scope=read+write

============================================================
LOGIN CREDENTIALS (demo server):
  Username: user
  Password: password
============================================================

Waiting for authorization callback...
  Authorization successful!

[Step 3] Executing operations with authorized scopes...

  Calling list_notes() - requires 'read' scope...
    SUCCESS: list_notes() completed
      {
  "id": "1",
  "title": "Test Note",
  "created_at": "%Y-%m-%d %H:%M:%S"
}

  Calling add_note() - requires 'write' scope...
    SUCCESS: add_note() completed
      {
  "id": "2",
  "title": "Aggregated Note",
  "content": "Created with aggregated scopes",
  "created_at": "%Y-%m-%d %H:%M:%S"
}
```

We will show this demo on Monday's HackDemo Happy Hour! Feedback and suggestions are welcome!