Microsoft EntraID Authentication

Copyright © 2025 by

Robert M. Dondero, Ph.D.

Princeton University

EntraID Authentication

- Microsoft EntraID authentication

    - **Part 1**: User logs into EntraID server
        - Unnecessary if user is already logged into EntraID server
        - User must provide credentials
    - **Part 2**: User logs into PennyAdmin
        - User need not provide credentials

EntraID Authentication

- See **<u>PennyAdminEntraID</u>** app

Agenda

- Microsoft EntraID Authentication
  - **The code**
  - Running locally
  - Running on Render
  - Assessment

The Code

- See **<u>PennyAdminEntraID</u>** app (cont.)
    - runserver.py
    - penny.sql, penny.sqlite
    - database.py
    - header.html, footer.html
    - index.html, show.html,
    - add.html, delete.html, reportresults.html
    - penny.py

Agenda

- Microsoft EntraID Authentication
  - The code
  - **Running locally**
  - Running on Render
  - Assessment

Running Locally

- See **<u>PennyAdminEntraID</u>** app (cont.)

  - How to run it on your local computer…

Running Locally

1. Create and activate an appropriate Python virtual environment. To do that, issue these commands:

```
mkdir ~/.virtualenvs
python -m venv ~/.virtualenvs/somedir
source ~/.virtualenvs/somedir/bin/activate
```

2. Install modules into your Python virtual environment. To do that, in your app directory issue this command:

```
python -m pip install -r requirements.txt
```

Running Locally

3. Register your app (**http://localhost:5000**) with Microsoft Entra ID.

    3.1. Browse to **https://entra.microsoft.com**, and log in using your Princeton email address and password.
    3.2. In the left panel, click on "Entra ID" to expand that item.
    3.3. In the left panel under "Entra ID", click on "App registrations".
    3.4. Click on "+ New Registration".
    3.5. For "Name" enter `PennyEntraId`.
    3.6. For Redirect URI choose `Web` and enter the URL **http://localhost/getAToken**.
    3.7. Click "Register".
    3.8. Note the Application (client) ID; you'll need it in the next step.
    3.9. Click on the link immediately under "Client credentials".
    3.10. Click on "+ New client secret"
    3.11. For "Description" enter **PennyEntraId Secret**.
    3.12. For "Expires" choose something reasonable.
    3.13. Click "Add".
    3.14. Note the client secret; you'll need it in the next step.

Running Locally

4. Define these environment variables. It would be common to define
them in a **.env** file in your app directory.

```
AUTHORITY="https://login.microsoftonline.com/
    2ff60116-7431-425d-b5af-077d7791bda4"
REDIRECT_URI="http://localhost:5000/getAToken"
CLIENT_ID="yourEntraIdClientId"
CLIENT_SECRET="yourEntraIdClientSecret"
SCOPE="User.Read"
ENDPOINT="https://graph.microsoft.com/v1.0/me"
```

5. Run the server. To do that, in your app directory issue this
command:

```
python runserver.py
```

Running Locally

```
6. Browse to http://localhost:5000.

   6.1. You must use localhost as the host. Using the real IP address
        of your computer won't work properly. Using 127.0.0.1 won't
        work properly.

6.2. Authenticate using Microsoft Entra ID.

6.3. When prompted for an author name, enter Ker.
```

Running Locally

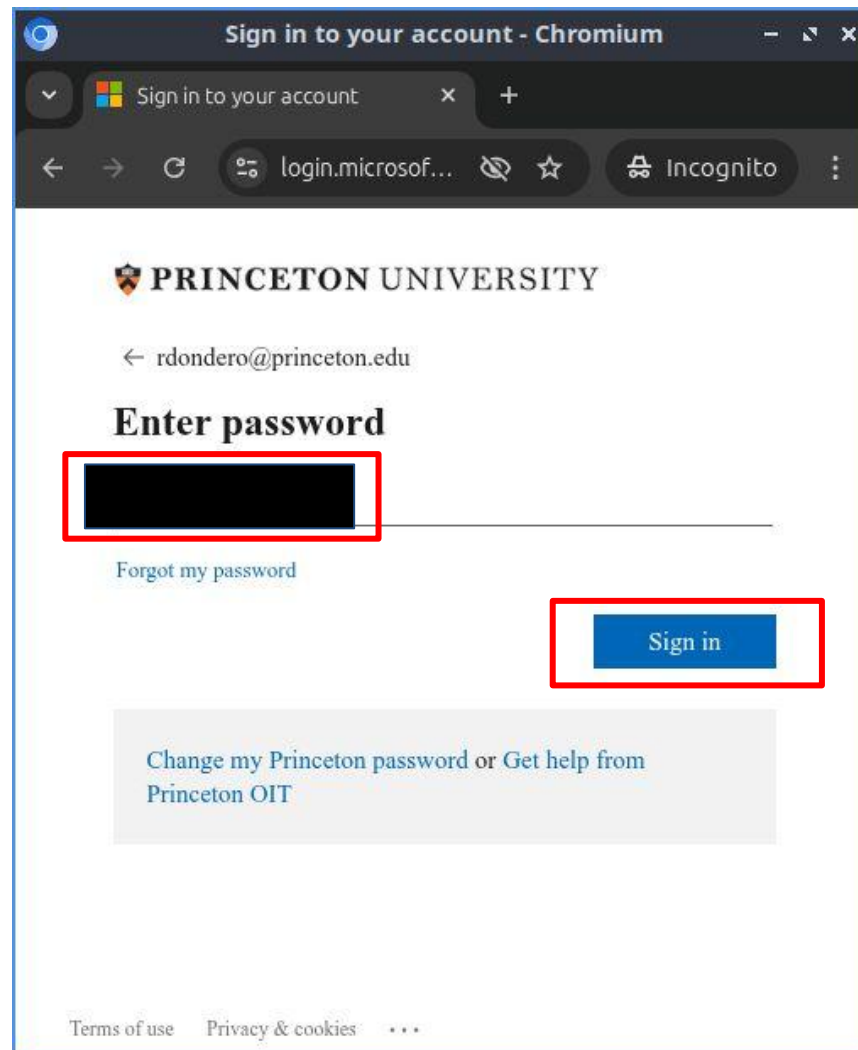- See **<u>PennyAdminEntraID</u>** app (cont.)

Running Locally
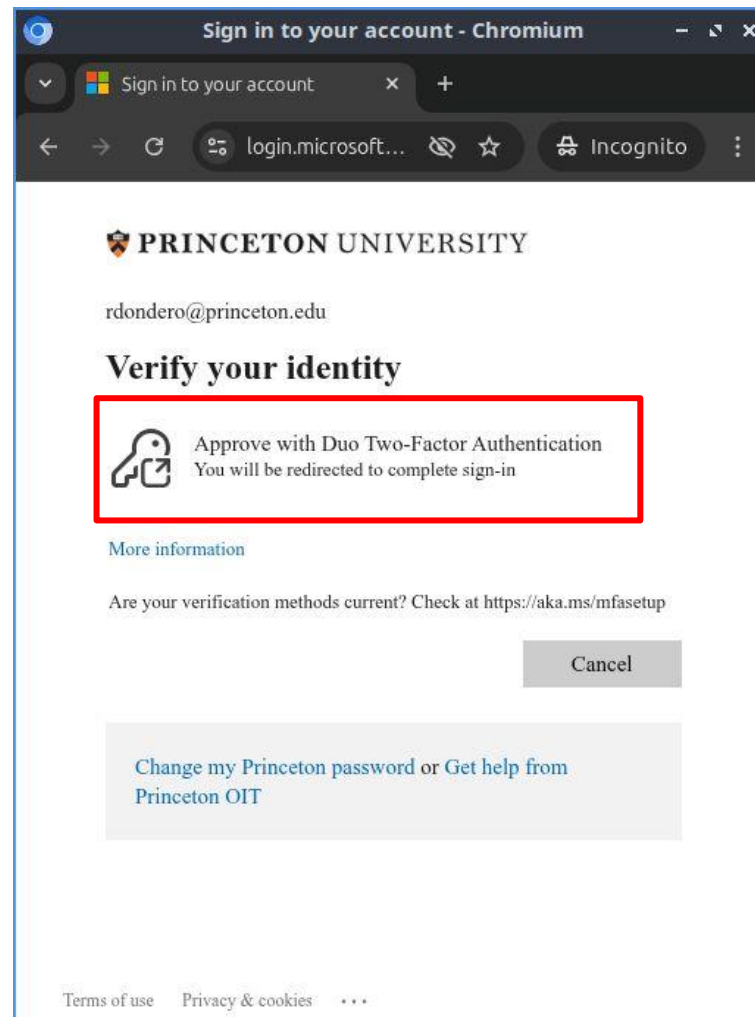
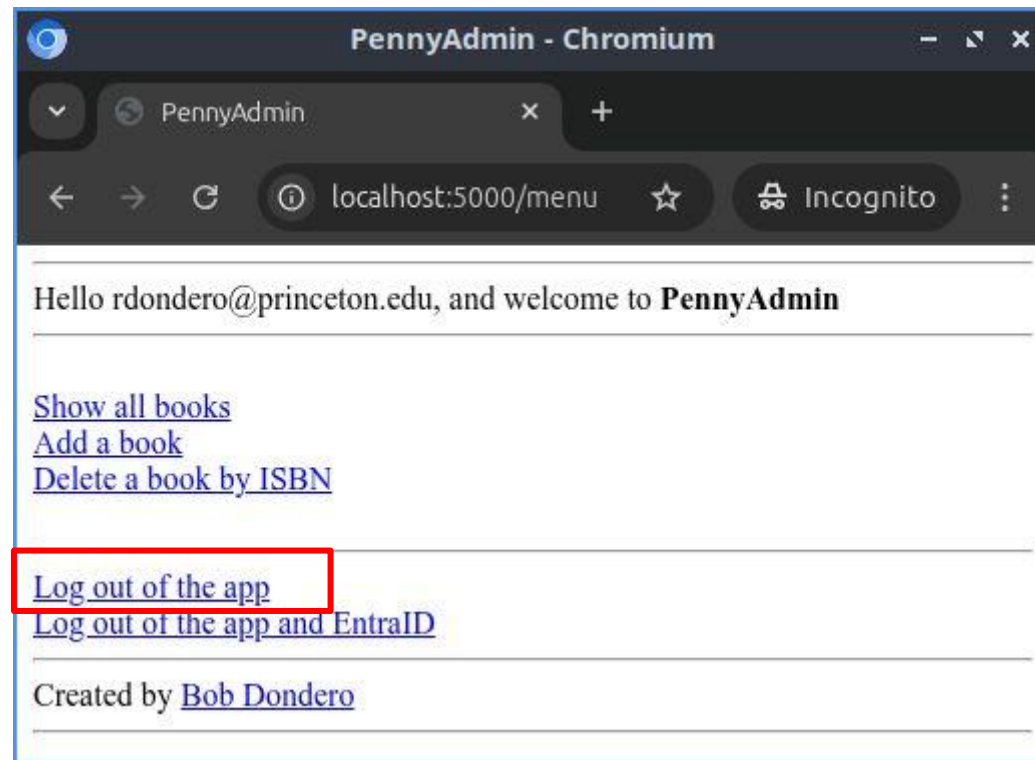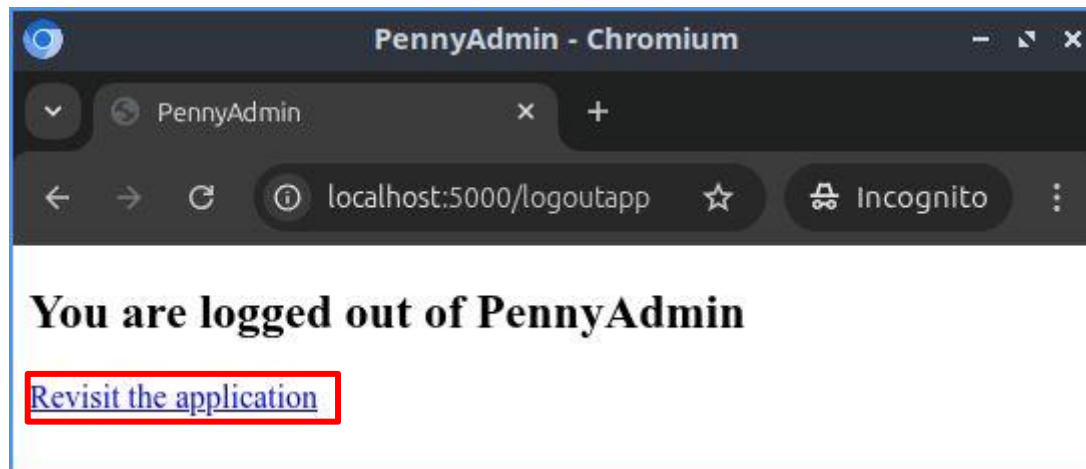- See **<u>PennyAdminEntraID</u>** app (cont.)

Running Locally

- See **PennyAdminEntraID** app (cont.)

Running Locally

- See **PennyAdminEntraID** app (cont.)

Running Locally

- See **<u>PennyAdminEntraID</u>** app (cont.)

Running Locally

- See **<u>PennyAdminEntraID</u>** app (cont.)
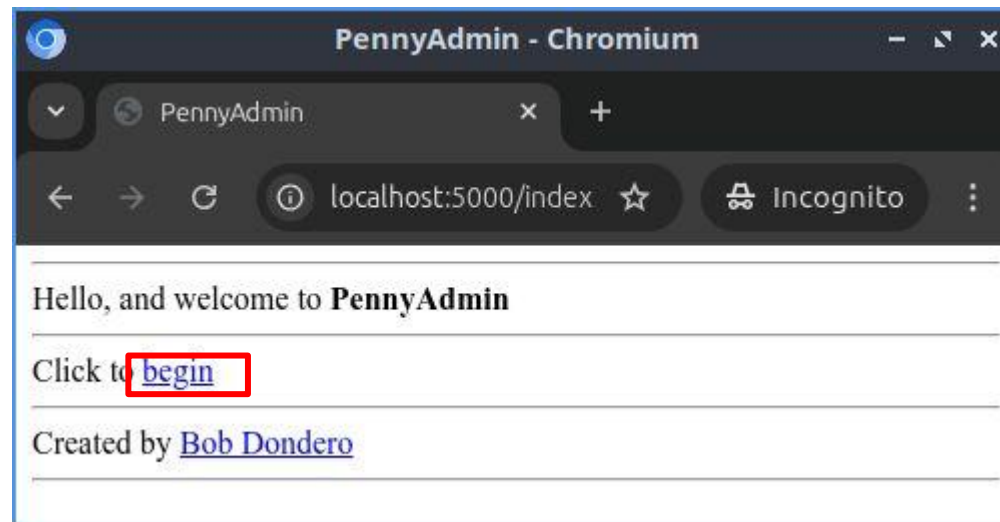
Running Locally

- See **PennyAdminEntraID** app (cont.)

Running Locally

- See **PennyAdminEntraID** app (cont.)
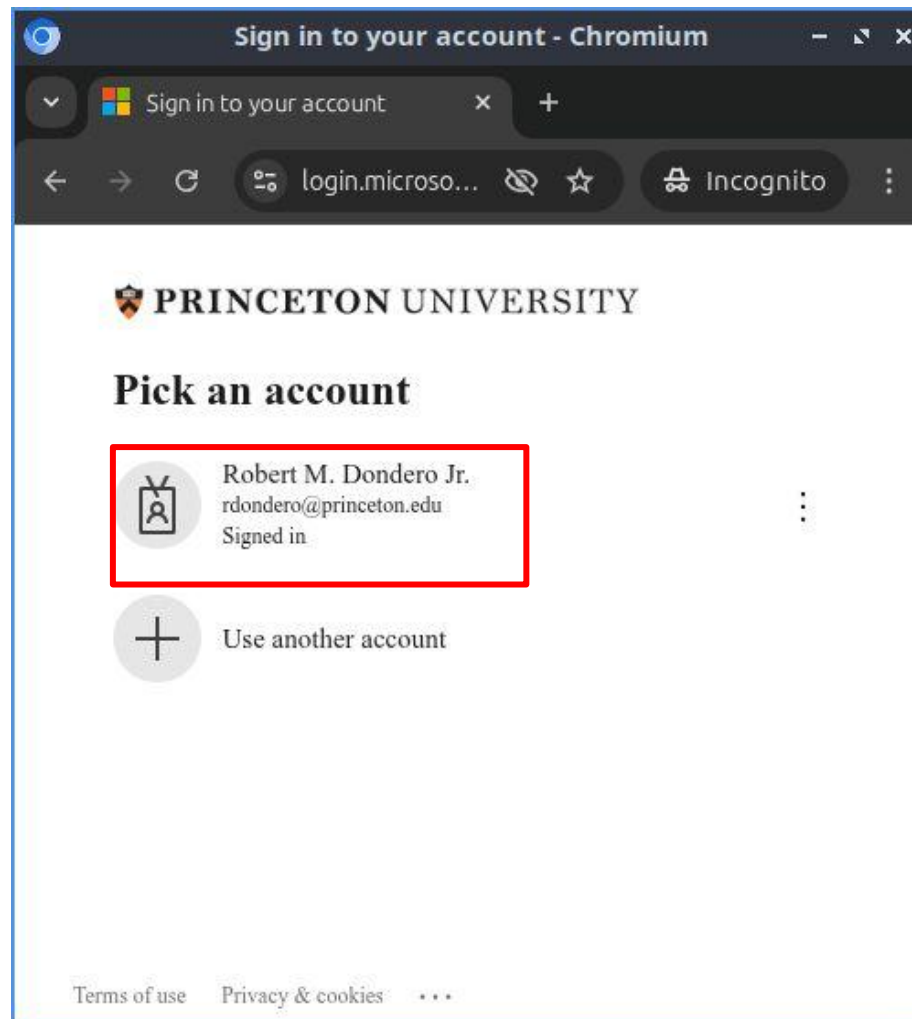
Running Locally

- See **<u>PennyAdminEntraID</u>** app (cont.)

Running Locally

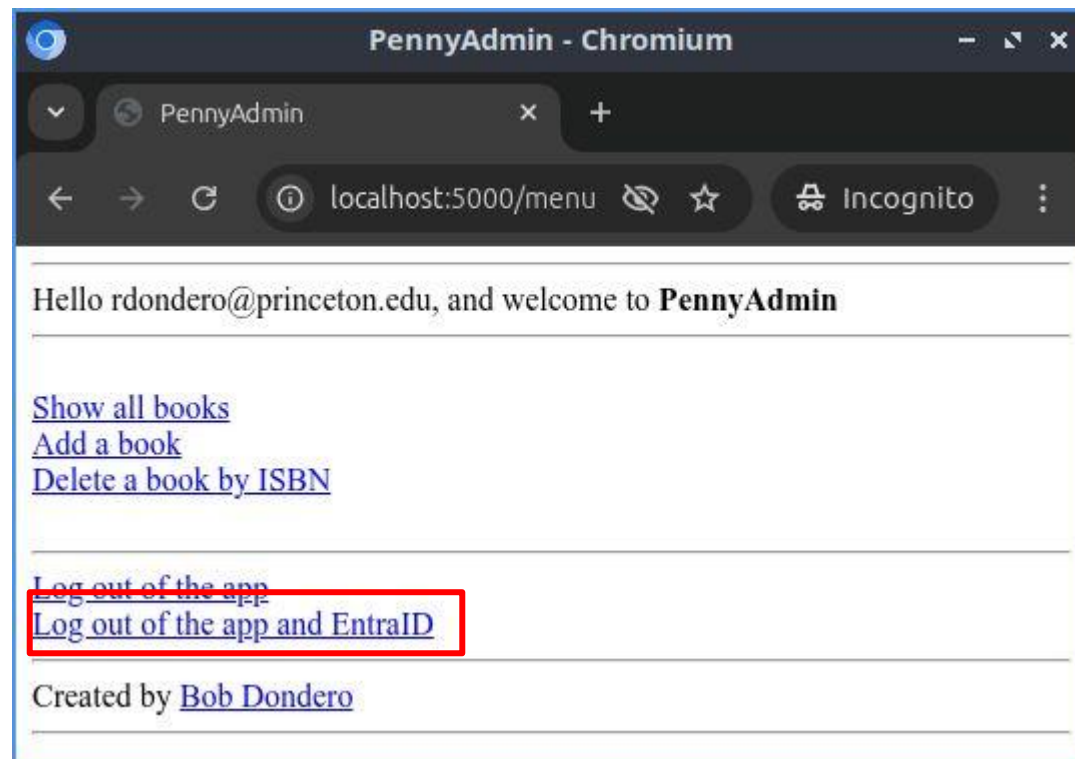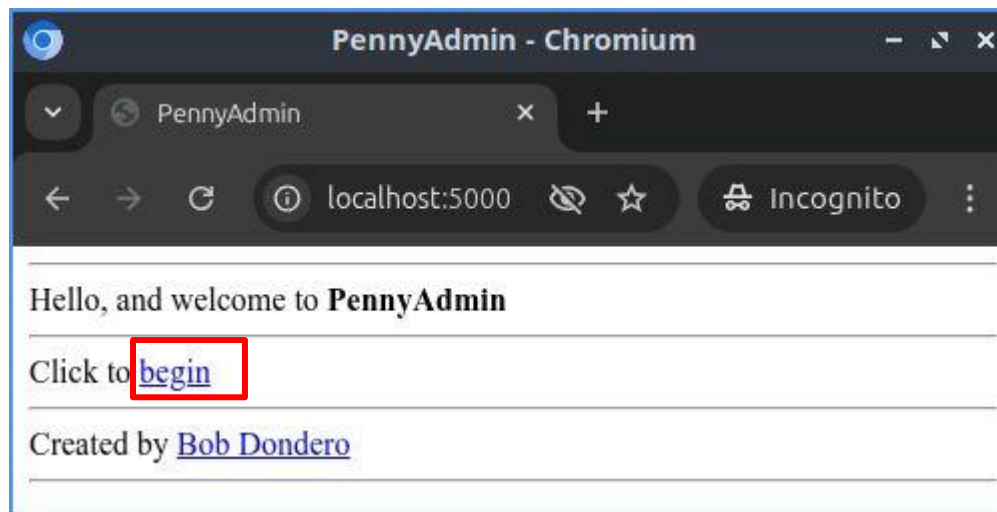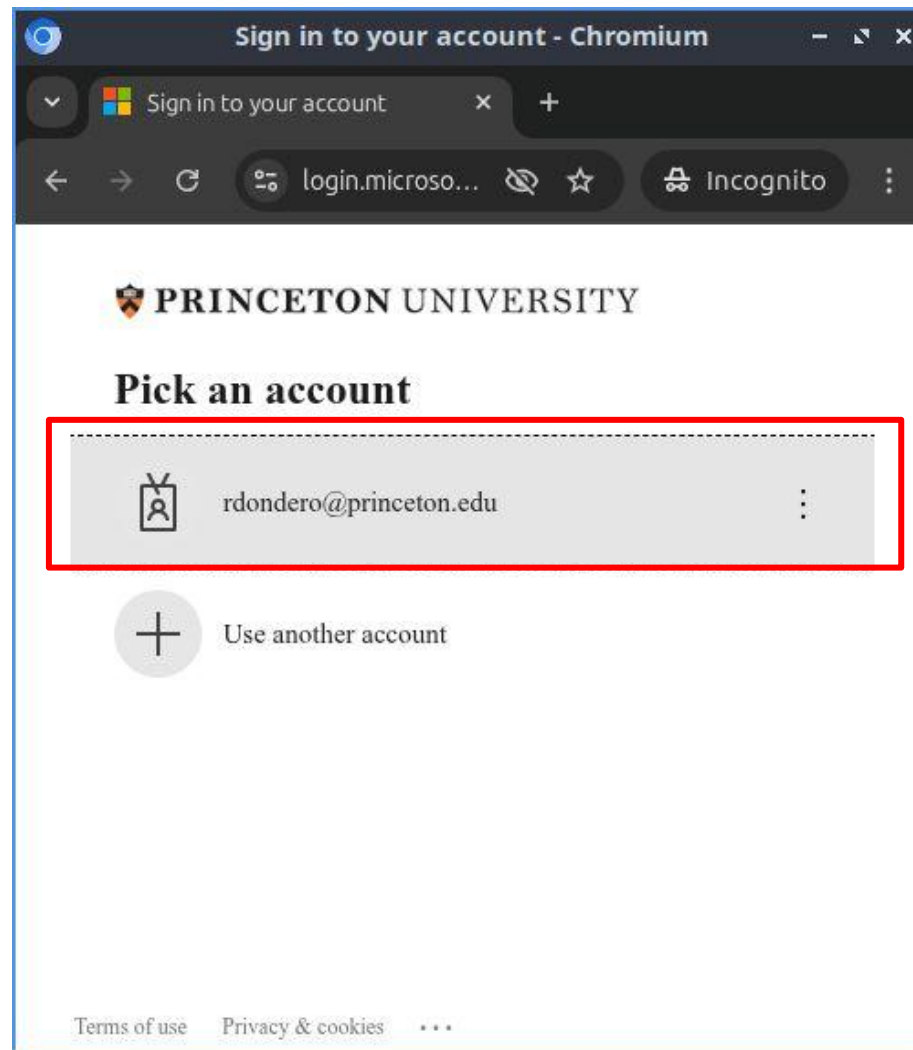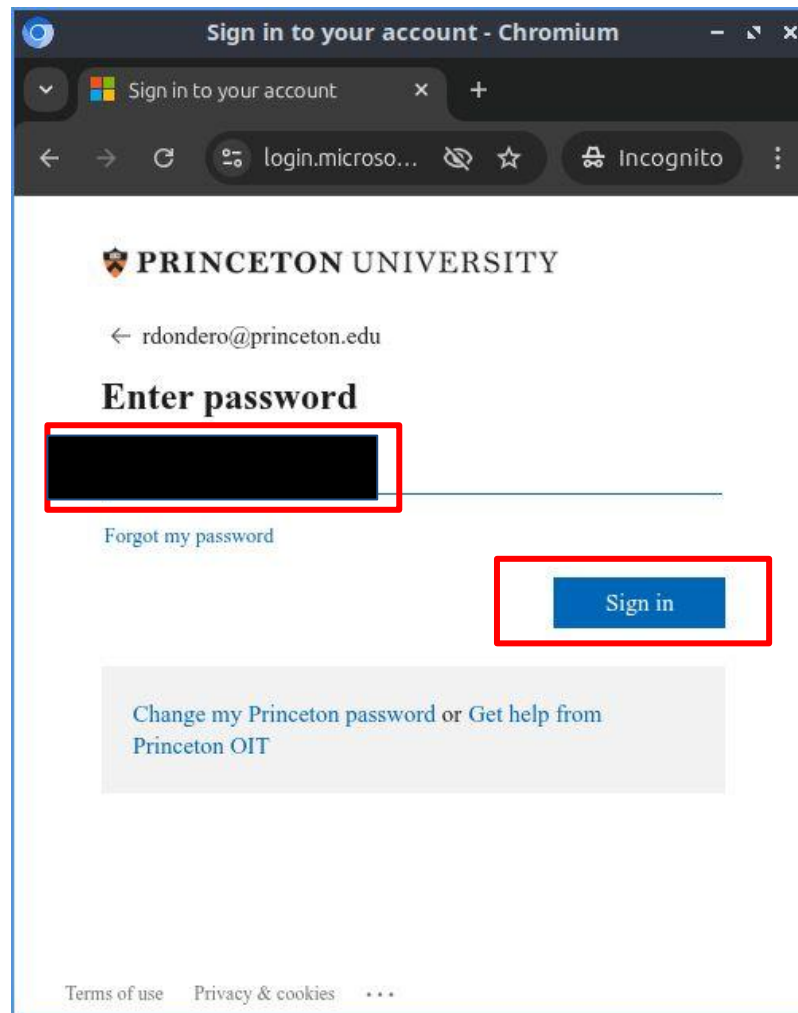- See **<u>PennyAdminEntraID</u>** app (cont.)

Running Locally

- See **PennyAdminEntraID** app (cont.)

Running Locally

- See **PennyAdminEntraID** app (cont.)

Running Locally

- See **PennyAdminEntraID** app (cont.)

Running Locally

- See **PennyAdminEntraID** app (cont.)
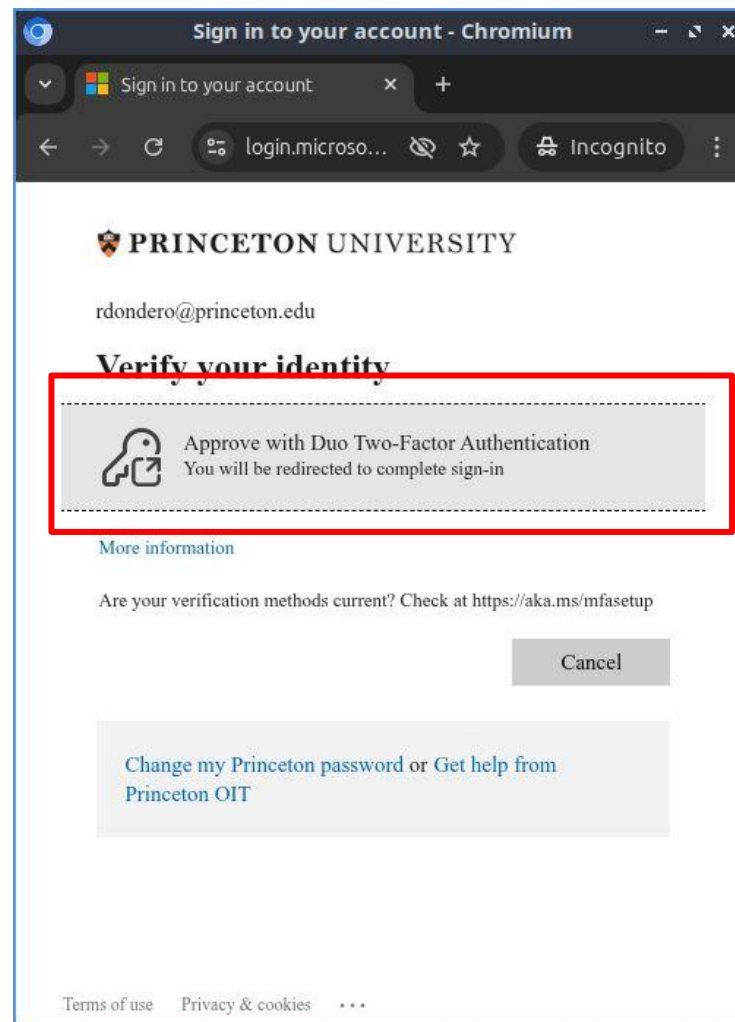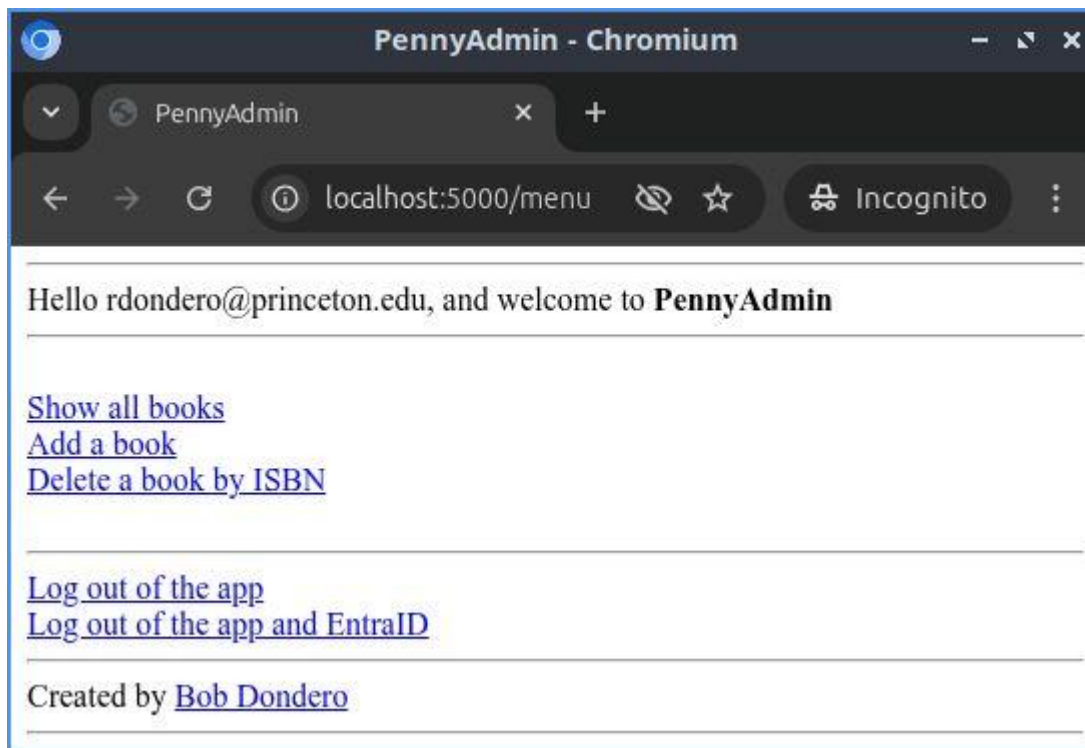
Agenda

- Microsoft EntraID Authentication
  - The code
  - Running locally
  - **Running on Render**
  - Assessment

Running on Render

- See **<u>PennyAdminEntraID</u>** app (cont.)

  - How to deploy it to Render.com and run it there…

Running on Render

```
1. Deploy the app to Render as a **Web Service**.

   1.1. Provide this as the build command:

       pip install -r requirements.txt

   1.2. Provide this as the start command:

       gunicorn penny:app

   1.3. Define these environment variables:

       AUTHORITY="https://login.microsoftonline.com/
           2ff60116-7431-425d-b5af-077d7791bda4"
       REDIRECT_URI="http://localhost:5000/getAToken"
       CLIENT_ID="yourEntraIdClientId"
       CLIENT_SECRET="yourEntraIdClientSecret"
       SCOPE="User.Read"
       ENDPOINT="https://graph.microsoft.com/v1.0/me"
```

Running on Render

```
2. Register your Render app with Microsoft Entra ID.

   2.1. Again browse to `https://entra.microsoft.com`.

   2.2. Follow the instructions given above, except...

      2.2.1. For "Redirect URIs" add
             https://yourappname.onrender.com/getAToken.

3. Browse to `https://yourappname.onrender.com`.

4. Authenticate using Microsoft Entra ID.

5. When prompted for an author name, enter  Ker.
```

Agenda

- Microsoft EntraID Authentication
  - The code
  - Running locally
  - Running on Render
  - **Assessment**

Assessment

- **Pros**
  - Application need not manage usernames or passwords
  - Application *cannot* access passwords!
  - Application is constrained to one user community
  - More standards-based than CAS
  - Princeton OIT is phasing in
  - No need for Princeton OIT intervention

Assessment

- **Cons**
  - Complex
  - Adds overhead, but only during user's first visit to the app per browser session
  - Application is constrained to one user community!