iRODS HTTP API v0.3.0
with OpenID Connect
Updates since UGM 2023

v0.1.0
- 88 issues closed - 10 bugs, 57 enhancements

v0.2.0
- 57 issues closed - 11 bugs, 25 enhancements
- Simplified OIDC configuration
- Improved separation between HTTP status codes and iRODS status codes
- Improved API documentation
- Improved API usage by constraining input requirements
- Improved stability
- Configuration validation on server startup

v0.3.0
- 6 issues closed - 1 bug, 4 enhancements
- Improved support for OIDC - Protected Resource mode
- Improved support for TLS between HTTP API and iRODS server
OAuth & OIDC in v0.3.0

- Three Major Features
  - OAuth 2.0 Confidential Client
  - Alternate User Mapping
  - HTTP API as an OAuth 2.0 Protected Resource
- Link to PR
  - https://github.com/irods/irods_client_http_api/pull/252
• OAuth 2.0 Client Authentication
  ■ Currently Support Password Based Authentication
  ■ Both Client and Protected Resource modes supported
Alternate User Mapping

- Previously required mapping in OpenID Provider
- Provide mapping in configuration file
Alternate User Mapping

```json
...,
"openid_connect": {
  ...
  "user_attribute_mapping": {
    "rodsBob": {
      "email": "bob@bobtopia.example",
      "sub": "a.very.real.sub",
      "phone_number": "56709"
    },
    "rodsAlice": {
      "email": "al-1s@wonderland.example",
      "sub": "a.different.sub"
    }
  }
...}
```

User Mapping Example
Alternate User Mapping

- Protected Resource Mode
  - Map via Introspection Endpoint

- Client Mode
  - Map via OpenID Connect ID Token

- Information received dependent on configuration
Alternate User Mapping

```json
{
    "active": true,
    "client_id": "l238j323ds-23ij4",
    "username": "jdoe",
    "scope": "read write dolphin",
    "sub": "Z5O3upPC88QrAjx00dis",
    "aud": "https://protected.example.net/resource",
    "iss": "https://server.example.com/",
    "exp": 1419356238,
    "iat": 1419350238,
    "extension_field": "twenty-seven"
}
```

Token Introspection Example
HTTP API as an OAuth 2.0 Protected Resource

- Removes HTTP API from OAuth authentication flows
  - Simplifies Code Executed
  - Streamlines Integration with OpenID Provider
- Only handle Access Token
- Currently Supports OAuth 2.0 Introspection Endpoint
HTTP API as an OAuth 2.0 Protected Resource

Example of Protected Resource Communications
• OAuth 2.0 Security Best Practices Draft (Work in Progress)
  - Resource Owner Password Credentials MUST NOT be used

• OAuth 2.1 Draft (Work in Progress)
  - Resource Owner Password Credentials Omitted
  - Removal of Implicit Grant
References

- OAuth 2.0
  - https://www.rfc-editor.org/rfc/rfc6749
- OpenID Connect Core
  - https://openid.net/specs/openid-connect-core-1_0.html
- OpenID Connect Client Discovery
  - http://openid.net/specs/openid-connect-discovery-1_0.html
- OAuth 2.1 Draft
- OAuth 2.0 Security Best Current Practice Draft
- OAuth 2.0 Token Introspection
Future Work

High Priority
- Make write operation web-friendly
- Log client IP or other identifier(s) to distinguish users in log output

Medium Priority
- Externalize OIDC user mapping
- Update to use 4.3.2 GenQuery2 API
- Implement missing iRODS API operations

Considering
- Status / Cancellation operations for active transfers
- Extending the lifetime of Basic Authentication tokens on use
- Using API documentation generation tool
Thank you!

Questions?