Configuring OpenID Connect

You can configure DTP to accept authentication from an OpenID Connect identity provider. This enables you to manage user authentication outside of Parasoft. Support for OpenID Connect is disabled by default. In this section:

Basic Configuration

Configuration is required in DTP, as well as in the OIDC server.

OIDC Server Configuration

1. If you have not already done so, register DTP with your OpenID Connect identity provider. You can get the values for the attributes used in the oidc.json file from the authorization server (Keycloak, connect2id, etc.).
2. Register the necessary redirect URIs so that the OIDC server knows where to send the user after authentication. The following redirect URIs that should be registered:

   Required:

   - <dtp_server>/pst/login/oauth2/code/dtp
   - <dtp_server>/pstsec/login/oauth2/code/dtp (User Administration)
   - <dtp_server>/grs/login/oauth2/code/dtp (Required for Report Center/Enterprise Pack)

   Optional. The following redirect URIs are optional and only need to be registered to enable log in directly through individual applications:

   - <dtp_server>/tcm/login/oauth2/code/dtp (Team Server)
   - <dtp_server>/licenseserver/login/oauth2/code/dtp (License Server)
   - <data_collector>/login/oauth2/code/dtp (Data Collector)

DTP Configuration

1. Open the oidc.json file located in the <DTP_DATA_DIR>/conf directory.
2. Specify the following values (all attributes are required):

   ```
   "enabled": true,
   "issuerUri": "<path to the authorization server>",
   "clientId": "<ID provided by the OpenID Connect server>",
   "clientSecret": "<password provided by OpenID Connect server>",
   "usernameAttribute": "preferred_username"
   "adminUsers": ["<known user designated as the Parasoft admin>"]
   ```

   The value of the `issureUri` parameter is the URI of the Authorization Server. The URI will be appended with `/.well-known/openid-configuration` to build the complete discovery endpoint when file is processed.

   The `usernameAttribute` is set to `preferred_username` by default. You should only change this value if the OpenID Connect server returns a different value. See Example Configurations for additional information.

   The value of the `adminUsers` parameter is an array of existing users in your organization that should be granted administrator privileges upon logging in. Also see User Administration Overview for information about permissions.

   > ⚠ The oidc.json file should be configured prior to the admin users logging in for the first time, otherwise the users will be added to the database without the permissions necessary for performing administrative functions.

3. Save your changes and restart DTP services.

When you go to the DTPlogin page, you will be redirected to the OpenID Connect authentication interface. After specifying your credentials, you will be logged in and redirected back to DTP.

Example Configurations

The following examples are intended to help you understand how to connect DTP to your identity access management system. Refer to the documentation for your software for implementation details.

Keycloak

The following configurations are prerequisites for configuring OpenID Connect for Keycloak as described in this example:

- Keycloak should be using RS256 as the default signature algorithm.
- The access token from Keycloak should include user information available that can also be retrieved from the Keycloak `userinfo API` endpoint.
- The following redirect URIs should be registered:
  - host:port/* (default windows port is 80, linux port is 8080)
  - host:8314/*
  - host:8082/* (for Data Collector upload form)

  If wildcards are not used, then individual URIs for Report Center, User Administration, Team Server, License Administration need to be added. See OIDC Server Configuration for additional information about registering redirect URIs.

In this example, `demo` is the name of the realm and two administrator users (`admin1` and `admin2`) will be created.

```
"enabled": true,
"issuerUri": "https://host:8095/auth/realms/demo/",
"clientId": "pstsec",
"clientSecret": "4d35ef23-aec5-44d7-9c59-18092bd619e8",
"usernameAttribute": "preferred_username"
"adminUsers": ["admin1","admin2"]
```

Refer to the Keycloak documentation for additional information.

Google

The following example demonstrates how to configure OpenID Connect for Google. In this example, two administrator users (`admin1` and `admin2`) will be created.

```
"enabled": true,
"issuerUri": "https://accounts.google.com",
"clientId": "<clientId-from-google>",
"clientSecret": "<clientSecret-from-google>",
"usernameAttribute": "given_name",
"adminUsers": ["admin1","admin2"]
```

Refer to Google's documentation for additional information.

Connect2id

The following example demonstrates how to configure OpenID Connect for connect2id.

Users must access DTP over HTTPS when using connect2id as the OpenID Connect provider.

In this example, `c2id` is the name of the realm. Two administrator users (`admin1` and `admin2`) will be created.

```
"enabled": true,
"issuerUri": "https://host:port/c2id",
"clientId": "<clientId-from-c2id>",
"clientSecret": "<clientSecret-from-c2id>",
"usernameAttribute": "sub",
"adminUsers": ["admin1","admin2"]
```

Refer to the connect2id documentation for additional information.

Known Limitations

The first time you log into DTP through connect2id, you may receive an "Invalid Request" error. To resolve the error, use a different browser or clear the cache of your current browser.

Using APIs in OpenID Connect Mode

When DTP is in OpenID Connect mode, you cannot access either DTP or Enterprise Pack APIs using basic authentication. Instead, you must pass an access token to the API endpoint in the request header using the `Authorization` property. The token is passed using the following format:

`Authorization: Bearer <access token>`

Refer to your identity access management software for information on how to obtain an access token.

You must pass the token for every API call, regardless of the method. In the following example, a token with the value "`1234567890`" is passed to the DTP build API endpoint.

```
curl -X GET -H "Authorization: Bearer 1234567890" http://dtp.host.com/grs/api/v1.7/builds?limit=1000&offset=0
```