THE LOGISTICS FLOW.

Partner Authentication RIOIAM

Scope of this document

This document only describes how the partner gets access to the data of RIO customers and how the permissions are managed. It does not describe billing, business APIs or how the partner application is integrated into the marketplace.

Partner responsibilities

- Provide the following information
    - Name, short description (1-2 sentences) and a more detailed description of the application
    - Contact name and email address. This should be a mailing list of people responsible for application, not an individual person which quickly gets outdated. It will be used for important announcements, updates and possibly warnings in case of abuse.
    - Required OAuth scopes (the list is currently not available yet) and permissions on the RIO platform
    - Required grant types
    - Callback URL for subscription confirmation callbacks (if applicable)
- Rotate client secret regularly, if you have a non-public client. The client secret will expire if it is not rotated. (not ready yet, but available soon)
- React to emails send to the contact email address within a few days. Keep it up to date.

Multi-tenancy

There will be a way to have development sandboxes for partners. The concept is not ready yet. But the API endpoints will be the same for production and development and the tenant can be switched by an X-TenantID header. There will be only one client_id and client_secret that can be used for all tenants. We might consider providing additional test clients bound to a specific tenant for testing purposes. The development sandbox is a long-term goal and will not be available soon.

THE LOGISTICS FLOW.

Marketplace integration for partners with backend integration

Partner application registration

Whenever a partner application is registered, the following happens:

- A new client for the partner application is registered at the authorization server.
- The partner gets a client id and secret to initialize his application
- The partner application is added to the marketplace

Secret rotation

All shared secrets have to be rotated regularly. There are 2 secrets that we share with a partner application:

- The client secret. It's used to authenticate the partner application. The authorization server exposes an API endpoint that allows updating the client secret by the client itself. If a client does not update the secret withing 14 days, the secret expires.
- The partner callback secret. This secret authenticates the RIO partner integration service to the callback URL of the partner application. It not clear yet, if this secret is rotated by us or the partner. But it will be passed in each callback the partner receives from RIO. An alternative would be to get rid of the partner callback secret and use signed tokens instead.

Note: The API for that is not ready yet, but will be available at some point in 2019. Partners will get a notification and will be required to start rotating their secret within 3 months after that notification.

Product subscription

There will be a callback that notifies partner automatically whenever a customer books an application in the marketplace. This callback will also provide the partner with an integration id for each customer that books the application. Currently this is still a manual process. The integration id is required to get access tokens which allow access to the RIO APIs. The partner application is using the custom partner_integration grant type describes below.

Partner integration grant type

The partner integration grant types provide a technical user for each account that has a subscription to the partner product using only the partner applications client_id and client_secret. This grant must only be used by confidential clients, that can safely keep a client_secret.

Parameters:

- Authorization (header, required): HTTP Basic authentication with client_id and client_secret
- Grant type (body, required): partner_integration
- Scope (body, optional): The requested scope of the access token. Usually unset. This gives the token the maximal scope allowed for the client.
- Integration id (body, required): The integration id given to the partner during product activation

Example request:

```
POST /oauth/token HTTP/1.1
Host: auth.iam.rio.cloud
Authorization: Basic czZCaGRSa3F0MzpnWDFmQmF0M2JW
Content-Type: application/x-www-form-urlencoded
Accept: application/json
grant_type=partner_integration&integration_id=58cfbc07-4424-45b5-8638-
f24f9f734fcb
```

The response looks like specified by RFC 6749:

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
Cache-Control: no-store
Pragma: no-cache

{
  "access_token":"2YotnFZFEjr1zCsicMWpAA",
  "token_type":"bearer",
  "expires_in":3600,
  "scope":"scope1 scope2"
}
```

The partner application can use the access token in the response to access the APIs in the scope directly. A refresh token is not issued, since the partner application can get a new access token whenever it needs to.

Example with curl:

```
curl --user ${client_id}:${client_secret} -k -d
"grant_type=partner_integration&integration_id=${integration_id}"
https://auth.iam.rio.cloud/oauth/token
```

We reserve the right to modify any of the above mentioned.