Overview

SSO (single sign on) gives clients the ability to integrate whichever user account system they have in place with Brandfolder, in order to reduce the amount of passwords and login screens users have to manage. Below are detailed descriptions of each endpoint related to Brandfolder's SSO workflow.

Sign Up

This endpoint is used for creating new users in Brandfolder's system. Clients will need to have a valid `application_id` assigned to them in order for it to work. Brandfolder will return an error if the user already exists in it's system or the token provided is invalid.

The token needs to be a valid JWT signed with the client's `application_secret` using HS256. It must contain the following payload:

```
{
  "email": "test@example.com",
  "first_name": "Test",
  "last_name": "Account"
}
```

(`first_name` and `last_name` are optional)

POST /api/v3/sso/:application_id/signup?token=JWT_TOKEN_HERE

Request Params: * application_id (Required in the URL) * token (Required: A JWT that meets the requirements above)

Example Response:

```
{
  "data": {
      "sso_token":
"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyX2tleSI6ImFiY2QtZWZnaC1pamtsIn0.JlysS
glxzfhLJgrgUrhJQbV-1rCj6-M95NjeHdv4wdc"
   }
}
```

Login

This endpoint is used to get an SSO token for Brandfolder in order to login a user at a later time. It takes a token containing the user's email. Brandfolder trusts that clients have done proper authentication themselves and returns an SSO token if their token is verified. Brandfolder will return an error if the user does not exist in it's system or the token provided is invalid.

The token needs to be a valid JWT signed with the client's `application_secret` using HS256. It must

contain the following payload:

```
{
  "email": "test@example.com"
}
```

POST /api/v3/sso/:application_id/login?token=JWT_TOKEN_HERE

Request Params: * application_id (Required in the URL) * token (Required: A JWT that meets the requirements above)

Example Response:

```
{
  "data": {
      "sso_token":
"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyX2tleSI6ImFiY2QtZWZnaC1pamtsIn0.JlysS
glxzfhLJgrgUrhJQbV-1rCj6-M95NjeHdv4wdc"
   }
}
```

Reset Password

This endpoint is used to reset a user's password when their password is updated in the client's system. It takes a token containing the user's email. Brandfolder will return an error if the user does not exist in it's system or the token provided is invalid. Brandfolder does not allow resetting passwords for users that are part of multiple organizations (for security reasons).

The token needs to be a valid JWT signed with the client's `application_secret` using HS256. It must contain the following payload:

```
{
  "email": "test@example.com",
  "password": "new_password"
}
```

POST /api/v3/sso/:application_id/reset_password?token=JWT_TOKEN_HERE

Request Params: * application_id (Required in the URL) * token (Required: A JWT that meets the requirements above)

Example Response: A 200 OK response is returned with an empty body.

Assign Permissions

This endpoint is used to assign permissions to a user for the given client's Organization,

Brandfolders, and or Collections. Brandfolder will return an error if the user does not exist in it's system or if the token provided is invalid.

The token needs to be a valid JWT signed with the client's `application_secret` using HS256. It must contain the following payload:

```
{
    "email": "test@example.com",
    "user_permissions": {
        "organizations": [
            {
                "slug": "example_org",
                "permission_level": "guest"
            }
        ],
        "brandfolders": [
            {
                "slug": "example_bf",
                "permission_level": "collaborator"
            }
        ],
        "collections": [
            {
                "slug": "example_collection",
                "permission_level": "admin"
            }
        ]
    }
}
```

POST /api/v3/sso/:application_id/assign_permissions?token=JWT_TOKEN_HERE

Request Params: * application_id (Required in the URL) * token (Required: A JWT that meets the requirements above)

Example Response: A 200 OK response is returned with an empty body.

Remove All Permissions

This endpoint is used to remove all user permissions relevant to the client's Organization, Brandfolders, and Collections within Brandfolder itself. It takes a token containing the user's email. Brandfolder will return an error if the user does not exist in it's system or the token provided is invalid.

The token needs to be a valid JWT signed with the client's `application_secret` using HS256. It must contain the following payload:

```
{
  "email": "test@example.com"
}
```

DELETE /api/v3/sso/:application_id/remove_all_permissions?token=JWT_TOKEN_HERE

Request Params: * application_id (Required in the URL) * token (Required: A JWT that meets the requirements above)

Example Response: A 200 OK response is returned with an empty body.

Get Resources

This endpoint is used to retrieve a list of Brandfolders and Collections for a given client.

The token needs to be a valid JWT signed with the client's `application_secret` using HS256. The contents of the payload can be empty:

```
{}
```

GET /api/v3/sso/:application_id/resources?token=JWT_TOKEN_HERE

Request Params: * application_id (Required in the URL) * token (Required: A JWT that meets the requirements above)

Example Response:

```
{
  "data": {
    "organization": {
      "slug": "example-organization",
      "name": "Example Organization",
      "key": "op33h3-uefow-csfq8"
    },
    "brandfolders": [
      {
        "slug": "example-brandfolder",
        "name": "Example Brandfolder",
        "key": "op33h4-5bvwew-2cgmac"
      }
    ],
    "collections": [
      {
        "slug": "example-collection",
        "name": "Example Collection",
        "key": "op33h5-uf3m0-elpfgt"
      }
    ]
  }
}
```

What to actually do with the SSO Token clients get back:

Once a client has an `sso_token` for a user, they can redirect them to `https://brandfolder.com/organizations?sso_token=SSO_TOKEN_HERE` and Brandfolder will automatically login them in.

In addition, clients can provide a `redirect` parameter in order to redirect the user to a desired Brandfolder immediately upon logging in. The redirect param should be the slug of the desired Brandfolder to redirect to. Example: `https://brandfolder.com/organizations?sso_token=TOKEN_HERE&redirect=slack` will redirect users to `https://brandfolder.com/slack` after login.

Things to Note:

- Due to security reasons, if the user in Brandfolder is part of multiple organizations, Brandfolder will not allow them to login with SSO, nor will clients be able to reset their password. However, they can still be assigned permissions via the assign permissions endpoint and all permissions relevant to a client's organization can be revoked through the remove permissions endpoint.