Using the MobileFirst Server to authenticate external resources

Overview

While MobileFirst Platform provides the ability to deploy **Adapters** to the MobileFirst server to serve resources, you can also serve resources using your own server, using your preferred technology.

You can protect and authenticate those **external resources** using the MobileFirst security framework.

This tutorial covers how a resource server can validate, and introspect (get token data) about a MobileFirst Platform OAuth token.
The validation/introspection of the token is done using a RESTful request to the MobileFirst Server, specifically the introspection endpoint.

This can either be done entirely with custom code, or using one of MobileFirst's helper libraries that encapsulate part of the flow.

Flow

(Insert diagram here)

1. Client with the MobileFirst Platform SDK makes a resource request (or any HTTP request) to a protected resource with or without the `Authorization` header (token).
2. The resource server obtains a token for the scope `authorization.introspect` and then uses this token as authorization to the introspection endpoint.
3. The resource server takes the authorization token, and validates/introspects the client token.
4. If the the MobileFirst Authorization Server determined that the token is invalid (or doesn't exist), the resource server should redirect the client to obtain a new token for the required scope. This happens internally when using the MobileFirst Client SDK.

Confidential Client

For the external resource server to be able to use the scope `authorization.introspect`, your server needs to be registered as a **confidential client** in MobileFirst.

In the MobileFirst Operations Console, under **Settings → Confidential Clients**, add a new entry. Choose a client ID and API secret. Make sure to set `authorization.introspect` as the **Allowed Scope**.

Since the introspection endpoint is an internal protected resource, the resource server will need to obtain a token in order to send any data to it. If you attempt to make a request to the introspection endpoint without an authorization header, you will receive 401 response.

Implementations