



S2S AUTHENTICATION

Providing OAuth 2 Clients Access to the SessionM Platform

June 2019

Contents

Introduction	3
Setup.....	3
Client Credentials Request.....	3

Introduction

SessionM (SMP) has integrated OAuth 2 as an authorization framework that allows applications to acquire limited access to user accounts on an HTTP service. It works by delegating user authentication to the service that hosts the user accounts and authorizing access for third-party applications. OAuth 2 provides authorization flows for web and desktop applications as well as mobile devices.

Setup

Before you begin, be sure to:

- Create an OAuth 2 application that will make API requests of the SMP.
- Provide the API consumer with the client ID and secret of the OAuth 2 client.

Client Credentials Request

OAuth 2 clients must make a request to the SMP Identity Service with their client credentials. These include a client ID and a client secret.

Specify the request using the following endpoint and parameters:

```
POST <Identity_Service_URL>/oauth/token?grant_type=client_credentials&
client_id=my_client_id&
client_secret=my_client_secret
```

Consult the table below for parameter descriptions:

Parameter	Description
grant_type	Required. Must be <i>client_credentials</i> .
client_id	Required. Allows API to identify client application.
client_secret	Required. Obtained during setup stage.

The platform returns the following JSON response:

```
{
  "access_token": "070fd79c35ee0e51a0a1804e41d407a0824a5b6b8f3a5fa4ac792a2",
  "created_at": 1553199154,
  "expires_in": 1209599,
  "scope": "openid",
  "token_type": "bearer"
}
```

Subsequent S2S API requests to the SMP can be authenticated using the access token granted above by passing it as a bearer token in the HTTP authorization header.

For example, consider this sample endpoint, which retrieves customer profile data for a user:

```
GET /priv/v1/apps/:api_key/users/:user_id
Authorization: Bearer 070fd79c35ee0e51a0a1804e41d407a0824a5b6b8f3a5fa4ac792a2
```

Or, for incentive-related information, you could specify this endpoint to retrieve a user's progress towards the next, highest tier level in a loyalty program:

```
POST /api/1.0/user_tiers/tier_status
Authorization: Bearer 070fd79c35ee0e51a0a1804e41d407a0824a5b6b8f3a5fa4ac792a2
```