



# REST API

# Implementation Guide

---

7.3.3 Release

Copyright © 2022 OneStream Software LLC. All rights reserved.

Any warranty with respect to the software or its functionality will be expressly given in the Subscription License Agreement or Software License and Services Agreement between OneStream and the warrantee. This document does not itself constitute a representation or warranty with respect to the software or any related matter.

OneStream Software, OneStream, Extensible Dimensionality and the OneStream logo are trademarks of OneStream Software LLC in the United States and other countries. Microsoft, Microsoft Azure, Microsoft Office, Windows, Windows Server, Excel, .NET Framework, Internet Information Services, Windows Communication Foundation and SQL Server are registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. DevExpress is a registered trademark of Developer Express, Inc. Cisco is a registered trademark of Cisco Systems, Inc. Intel is a trademark of Intel Corporation. AMD64 is a trademark of Advanced Micro Devices, Inc. Other names may be trademarks of their respective owners.

# Table of Contents

Introduction .....	1
REST API Overview .....	2
OneStream Web API Endpoints .....	2
Authentication .....	2
DataManagement .....	2
DataProvider .....	2
Authentication .....	3
Application API .....	5
Data Provider API v7.2.0 .....	7
OneStream REST API Implementation .....	10
OneStream WebAPI Endpoints .....	10
Configure OneStream API for External Authentication .....	27
Azure AD Configuration .....	27
Okta Configuration .....	29
Create M2M Application Registration (grant_type = client_	
credentials) .....	29
Create U2M Application Registration (grant_type = password)	
in Okta .....	30
Update the Server Configuration Utility .....	34

## Table of Contents

---

PingFederate Configuration .....	35
Update the Server Config Utility .....	37
Configure the AUD Value .....	37

# Introduction

This guide provides information about the implementation, authentication and application programming interfaces available to extend OneStream functionality.

OneStream Web API is a RESTful web service designed to expose OneStream Data Automation functions when interacting with third-party API client applications. Our Web API must be installed on a web server and be configured for external authentication providers supporting OAuth2.0/OpenID Connect authorization protocol. Identity Providers currently supported are Okta, Azure AD and PingFederate.

OneStream Web API is API client agnostic. It accepts and outputs data in JSON format making it possible for every API client application that supports this format to also interact with the service.

# REST API Overview

In this topic:

- "OneStream Web API Endpoints" below
- "OneStream REST API Implementation" on page 10
- "Configure OneStream API for External Authentication" on page 27

## OneStream Web API Endpoints

URLs are relative to query parameter api-version=5.3.0, unless otherwise noted.

### Authentication

Authentication endpoint. Represents a RESTful service for Authentication.

- POST api/Authentication/LogonAndReturnCookie  
Used primarily by the Enablement Team to verify Web API installation completed successfully. Returns a one-time cookie value that holds authentication state or a message indicating failure along with a proper HTTP code.

### DataManagement

DataManagement endpoint. Represents a RESTful service of Data Management.

- POST api/DataManagement/ExecuteSequence:  
Executes a Data Management Sequence and returns a success/failure message along with a proper HTTP code.
- POST api/DataManagement/ExecuteStep  
Executes a Data management Step and returns a success/failure message along with a proper HTTP code.

### DataProvider

DataProvider endpoint represents a RESTful service of Data Provider.

- POST `api/DataProvider/GetAdoDataSetForAdapter`:  
Executes a Data Provider HTTP Post request and returns a JSON representation of a DataSet for a given Dashboard Adapter.
- POST `api/DataProvider/GetAdoDataSetForCubeViewCommand`  
Executes a Data Provider HTTP Post request and returns a JSON representation of a DataSet for a given Cube View.
- POST `api/DataProvider/GetAdoDataSetForSqlCommand`  
Executes a Data Provider HTTP Post request and returns a JSON representation of a DataSet for a given Sql query. **Administrator role is required for this functionality.**
- POST `api/DataProvider/GetAdoDataSetForMethodCommand`  
Executes a Data Provider HTTP Post request and returns a JSON representation of a DataSet for a given pre-defined list of method commands. **Administrator role is required for this functionality.**

## Authentication

To secure REST API with OAuth 2.0, configure authentication with one of these supported external providers:

- "Azure AD Configuration" on page 27
- "Okta Configuration" on page 29
- "PingFederate Configuration" on page 35

Access tokens from the any of the above providers have short expiration times. To avoid copying the entire token value to the Authorization/Token text box, create a variable that holds the value. For every call to the external provider, the value of the access token returned will be copied to the variable.

- Create a global variable in Postman, name it appropriately, for instance `webapi_access_token`.
- In the Tests tab of the POST request to the external provider copy the script below:

```
var data = pm.response.json();  
pm.environment.set("webapi_access_token", data.access_token);
```

## Authentication API

Method	Endpoint	Description
Post	Authentication/Logon	Logs on and returns a SessionInfo (SI) object for use with other Rest API calls that accept an SI as an argument. This endpoint performs a logon only and does not open an application. This is the equivalent of entering login credentials in the Desktop App before selecting and opening an application.

### Authentication/Logon

POST <https://{BaseWebServer}/api/Authentication/Logon?api-version=7.2.0>

### Query Parameters

Key	Value	Required
api-version	7.2.0	Yes

### Authorization

Type	Value	Required
Bearer Token	(your access token)	Yes

### Headers

Key	Value	Required
Content-Type	application/json	Yes

### Request Body

Key	Type	Description	Required
BaseWebServerURL	string	Your URL for the web service	Yes

### Sample Request

```
{
  "BaseWebServerUrl": "https:// golfstream.onestreamcloud.com/OneStreamWeb"
```



## REST API Overview

---

```
}
```

### Sample Response

```
{
  "Message": "Logon succeeded.",
  "Logon SessionInfo": {
    "XfBytes": " QB8AACNodHRwOi8vbG9jYXRob3N0OjUwMDAxL09uZVN0cmVhbVdlYhQAAAB7izp1jCP3BUVr8bjD2f6KmmL5BKzh0VWUzU1MikEY0Vek0ZUIT0tUQV9NMk27tnn6+VZaR544CK1YPCFeWSBWCTmQ2ggAAAAAAAAAAAAAAAFZW4tVVMAAAAAAAAAAAAAAAAAAAAAAAAAAP//////////8P//////////AwAAABn8//8Z/P//Gfz//xn8//8Z/P//Gfz//xn8//8Z/P//Gfz//w=="
  },
  "Authorized applications": [
    "GolfStreamDemo_2022",
    "OFC_ECA_ProductMgmt",
    "OneStream_GolfStream"
  ]
}
```

## Application API

Method	Endpoint	Description
Post	Application/OpenApplication	Opens specified application. Requires a valid sessionInfo token obtained from the Authentication/Logon method.

### Application/OpenApplication

POST https://{BaseWebServer}/api/Application/OpenApplication?api-version=7.2.0

### Query Parameters

Key	Value	Required
api-version	7.2.0	Yes

### Authorization

Type	Value	Required
Bearer Token	(your access token)	Yes

### Headers

Key	Value	Required
Content-Type	application/json	Yes

### Request Body

Key	Type	Description	Required
ApplicationName	string	Name of the application to open	Yes
SI	array (bytes)	The SessionInfo (SI) object obtained from Authentication/Logon endpoint.	Yes

### Sample Request

```
{
  "ApplicationName": "GolfStreamDemo_2022",
  "SI": {
    "XfBytes": "QB8AACNodHRwOi8vbG9jYXRob3N0OjUwMDAxL09uZVN0cmVhbVdlYhQAAAB7izp1jCP3BUVr8bjD2f6KmmL5BKzhOVWUzU1MikeYOVekoZUIT0tUQV9NMk27tnn6+VZaR544CKlYPCFeWSBWCTmQ2ggAAAAAAAAAAAAAAAAAAAAAAAFZW4tVVMAAAAAAAAAAAAAAAAAAAAAAAAAAAP////////////////////////////////////////8P////////////////////////////////AwAAABn8//8Z/P//Gfz//xn8//8Z/P//Gfz//xn8//8Z/P//Gfz//xn8//8Z/P//Gfz//w=="
  }
}
```

### Sample Response

```
{
  "Message": "Open application succeeded.",
}
```

```
"Application SessionInfo": {  
  "XfBytes": "QB8AACNodHRwOi8vbG9jYWxob3N0OjUwMDAxL09uZVN0cmVhbVd1YhQAAAAep0GewgsakcN4GJDmuwyaaIMazfN/aHyhnXNLgg+hUxy6cpQIT0tUQV9NMk27tnn6+VZaR544CK1YPCFe0BusL1iM2ggUAAAArL9Q04ePExHJxVU89Y1MAeNxr8UT251U3RyZWftX0dvdGZTdHJ1YW3xShfEXWxvRb0x2hWDSCd0BwVuLVVTAAAAAAAAAAAAAAAAAAAAAAD////////wAAAAACAFABAABQAFD///8AAAAAYHddwMAAABCAfAAGfz//5z///+c////FQAQACYAIAARAGAAAwCQABn8//8Z/P//Gfz//xn8//8="
```

## Data Provider API v7.2.0

Method	Endpoint	Description
Post	DataProvider/ GetadoDataSetForAdapter	Executes a Data Provider HTTP Post request and returns a JSON representation of a DataSet for a given Dashboard Adapter. Requires a SessionInfo (SI) object obtained from Application/OpenApplication endpoint.

### DataProvider/GetAdoDataSetForAdapter

POST https://{BaseWebServer}/api/DataProvider/GetAdoDataSetForAdapter?api-version=7.2.0

### Query Parameters

Key	Value	Required
api-version	7.2.0	Yes

### Authorization

Type	Value	Required
Bearer Token	(your access token)	Yes

### Headers

Key	Value	Required
Content-Type	application/json	Yes

### Request Body

Key	Type	Description	Required
IsSystemLevel	boolean	An indication of whether the Dashboard Adapter is defined at the System Level (True) or for the specified Application (False).	Yes
AdapterName	string	The name of the Dashboard Adapter used for data retrieval.	Yes
ResultDataTableName	string	Name of the resulting table in the DataSet	Yes
CustomSubstVarsAsCommaSeparatedPairs	string	Comma separated list of Variable name/value pairs requiring a user prompt. These must be specified using the following format: "VariableName1=[VariableValue1],VariableName2=[VariableValue2],...".	No
SI	array (bytes)	The SessionInfo (SI) object obtained from Application/OpenApplication endpoint.	Yes

### Sample Request

```
{
  "IsSystemLevel": true,
  "AdapterName": "Sales Mix (WF)",
  "ResultDataTableName": "ResultsTable",
  "CustomSubstVarsAsCommaSeparatedPairs": "",
  "SI": {
    "XfBytes": " QB8AACNodHRwOi8vbG9jYWxob3N0OjUwMDAxL09uZVN0cmVhbVdlYhQAAAEp0GewgsakcN4GJdmuwyaaIMazfN/aHyhnXNLgg+hUxy6c"
```

```
pQIT0tUQV9NMk27tnn6+VZaR544CK1YPCFe0BusL1iM2ggUAAArL9Q04eP
ExHJxVU89Y1MAeNxr8UT25lU3RyZWftX0dvbGZTdHJlYw3xShfEXWxvRbO
x2hWDScd0BWVuLVVTAAAAAAAAAAAAAAAAAAAAAAD////////wAAAA
ACAFABAABQAfD///8AAAAAYHddwMAAABCAfAAGfz//5z///+c///FQAQA
CYAIAARAGAAwCQABn8//8Z/P//Gfz//xn8//8="
}
}
```

## Sample Response

```
{
  "ResultsTable": [
    {
      "RowId": 0,
      "RowName": "Row1",
      "PovCubeNameAndDesc": "GolfStream - Corporate",
      "Pov00EntityNameAndDesc": "Total GolfStream",
      "Pov02ScenarioNameAndDesc": "Actual - Actual",
      "Pov03TimeNameAndDesc": "2011M2 - Feb 2011",
      "Pov04ViewNameAndDesc": "YTD",
      "RowHdr0NameAndDesc": "Drivers",
      "RowHdr0Indent": 0,
      "Col0Hdr0NameAndDesc": "60000 - Operating Sales",
      "Col0Hdr0Indent": 0,
      "Col0Value": 25552270.482000000000000000,
      "Col0ValueAsText": "25,552,270.48"
    },
    {
      "RowId": 1,
      "RowName": "Row1",
      "PovCubeNameAndDesc": "GolfStream - Corporate",
      "Pov00EntityNameAndDesc": "Total GolfStream",
      "Pov02ScenarioNameAndDesc": "Actual - Actual",
      "Pov03TimeNameAndDesc": "2011M2 - Feb 2011",
      "Pov04ViewNameAndDesc": "YTD",
      "RowHdr0NameAndDesc": "Fairway Woods",
      "RowHdr0Indent": 0,
      "Col0Hdr0NameAndDesc": "60000 - Operating Sales",
      "Col0Hdr0Indent": 0,
      "Col0Value": 17476089.966000000000000000,
      "Col0ValueAsText": "17,476,089.97"
    }
  ]
}
```

# OneStream REST API Implementation

In this topic:

- "Authentication" on page 3
- "OneStream WebAPI Endpoints" below

## OneStream WebAPI Endpoints

This API implementation is client agnostic therefore every API test capable third-party tool can be pointed to OneStreamWeb API endpoints. This tutorial is using Postman. Note that all arguments in the body are **required** unless otherwise specified.

Versioning This implementation will start with Api-version=5.2.0

## Data Management Execute Sequence endpoint

1. Create new POST request in Postman,
2. Url= http(s)://[servername]:  
[port]/onestreamapi/api/DataManagement/ExecuteSequence?api-version=5.2.0
3. Authorization: Type=Bearer Token. Token={{webapi\_access\_token}}
4. Headers: Content-Type=application/json
5. Body (raw / JSON):

```
{
  "BaseWebServerUrl": [your web server url ],
  "ApplicationName": [your application name],
  "SequenceName": [existing sequence name],
  "CustomSubstVarsAsCommaSeparatedPairs": [comma separated list of key value
pairs as substitution variables with the following format: "VariableName1=
[VariableValue1],VariableName2=[VariableValue2],..." - Optional
}
```

6. Click Send and observe the response at the bottom pane. If successful, a message of "Data Management Sequence [sequence name] was completed" will be returned otherwise a descriptive error message will show. More details will be logged in the Error and Activity logs.

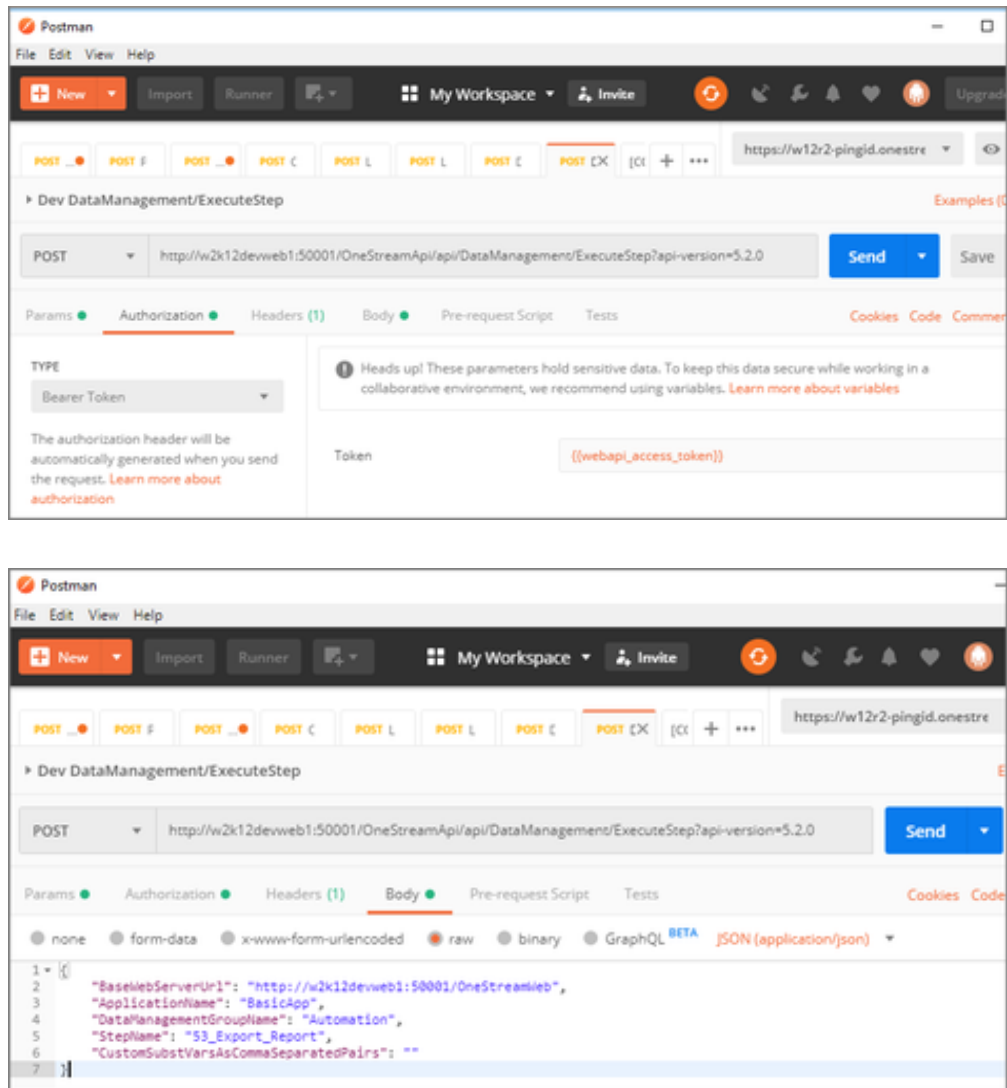
## Data Management Execute Step endpoint

1. Create new POST request in Postman,
2. Url= `http(s)://[servername]:[port]/onestreamapi/api/DataManagement/ExecuteStep?api-version=5.2.0`
3. Authorization: Type=Bearer Token. Token={{webapi\_access\_token}}
4. Headers: Content-Type=application/json
5. Body (raw / JSON):

```
{
  "BaseWebServerUrl": [your web server url ],
  "ApplicationName": [your application name],
  "DataManagementGroupName": [an existing data management group name],
  "StepName": [existing step name],
  "CustomSubstVarsAsCommaSeparatedPairs": [comma separated list of key value pairs as
substitution variables with the following format: "VariableName1=[VariableValue1],VariableName2=[VariableValue2],..." - Optional]
}
```
6. Click Send and observe the response at the bottom pane. If successful, a message of "Data Management Step [step name] was completed" will be returned otherwise a descriptive error message will show. More details will be logged in the Error and Activity logs.

## REST API Overview

---



## Data Provider GetAdoDataSetForAdapter endpoint

1. Create new POST request in Postman,
2. Url= http(s)://[servername]:  
[port]/onestreamapi/api/DataProvider/GetAdoDataSetForAdapter?api-version=5.2.0
3. Authorization: Type=Bearer Token. Token={{webapi\_access\_token}}



4. Headers: Content-Type=application/json

5. Body (raw / JSON):

```
{
  "BaseWebServerUrl": [your web server url ],
  "ApplicationName": [your application name],
  "WorkspaceName": Reserved for future use. Use an empty string. - Optional,
  "AdapterName": [existing adapter name],
  "ResultDataTableName": [name of resulting table in the DataSet],
  "CustomSubstVarsAsCommaSeparatedPairs": [comma separated list of key value pairs as
substitution variables with the following format: "VariableName1=[VariableValue1],VariableName2=
[VariableValue2],..." - Optional]
}
```

Example:

```
{
  "BaseWebServerUrl": "http://localhost:50528/OneStream",
  "ApplicationName": "GolfStream_v37",
  "IsSystemLevel": "False",
  "AdapterName": "ActivityClassListing_PLP",
  "ResultDataTableName": "ResultsTable",
  "CustomSubstVarsAsCommaSeparatedPairs": ""
}
```

6. Click Send and observe the response at the bottom pane. If successful, a JSON data table will be returned otherwise a descriptive error message will show. More details will be logged in the Error and Activity logs.

This is a returned response from the request using the above body example in Postman:

```
{
  "ResultsTable": [
    {
      "ClassID": "100_Salary",
      "Name": "100 - Salary",
      "Description": "100 - Salary",
      "ValueType": 0,
      "ValueTypeName": "Wage Percentage",
      "ClassItemID": "79b612b9-8cb4-49ca-9a0d-d13c7683a7f2",
      "Description1": "100 - Salary",
      "WeightOrValue": "1",

```

```
        "FKAccountID": "Salary_Exp",
        "Flow": "None",
        "IC": "None",
        "UD1": "None",
        "UD2": "None",
        "UD3": "None",
        "UD4": "None",
        "UD5": "None",
        "UD6": "None",
        "UD7": "None",
        "UD8": "None",
        "Sequence": 10.0,
        "FKClassID": "100_Salary"
    },
    ...
  ]}]}
```

## Data Provider GetAdoDataSetForCubeViewCommand endpoint

1. Create new POST request in Postman,
2. Url= `http(s)://[servername]:[port]/onestreamapi/api/DataProvider/GetAdoDataSetForCubeViewCommand?api-version=5.2.0`
3. Authorization: Type=Bearer Token. Token={{webapi\_access\_token}}
4. Headers: Content-Type=application/json
5. Body (raw / JSON):

```
{
  "BaseWebServerUrl": [your web server url ],
  "ApplicationName": [your application name],
  "CubeViewName": [existing Cube View name],
  "DataTablePerCubeViewRow ": [if true returns a Data Table Per Cube View row - bool],
  "ResultDataTableName": [name of resulting table in the DataSet],
  "CubeViewDataTableOptions": [set of formatting boolean options for the returned table -
Optional],
```

"CustomSubstVarsAsCommaSeparatedPairs": [comma separated list of key value pairs as substitution variables with the following format: "VariableName1=[VariableValue1],VariableName2=[VariableValue2],..." - *Optional*]

Example:

```
{
  "BaseWebServerUrl": "http://localhost:50528/OneStream",
  "ApplicationName": "GolfStream_v37",
  "CubeViewName": "Gross Margin",
  "DataTablePerCubeViewRow": false,
  "ResultDataTableName": "ResultDataTableNames",
  "CustomSubstVarsAsCommaSeparatedPairs": "",
  "CubeViewDataTableOptions": {
    "IncludeTitle": false,
    "IncludeHeaderLeftLabel1" : true,
    "IncludeHeaderLeftLabel2" : true,
    "IncludeHeaderLeftLabel3" : true,
    "IncludeHeaderLeftLabel4" : true,
    "IncludeHeaderCenterLabel1" : true,
    "IncludeHeaderCenterLabel2" : true,
    "IncludeHeaderCenterLabel3" : true,
    "IncludeHeaderCenterLabel4" : true,
    "IncludeHeaderRightLabel1" : true,
    "IncludeHeaderRightLabel2" : true,
    "IncludeHeaderRightLabel3" : true,
    "IncludeHeaderRightLabel4" : true,
    "IncludePovCube" : true,
    "IncludePovEntity" : true,
    "IncludePovParent" : true,
    "IncludePovCons" : true,
    "IncludePovScenario" : true,
    "IncludePovTime" : true,
    "IncludePovView" : true,
    "IncludePovAccount" : true,
    "IncludePovFlow" : true,
    "IncludePovOrigin" : true,
  }
}
```

```
        "IncludePovIC" : true,
        "IncludePovUD1" : true,
        "IncludePovUD2" : true,
        "IncludePovUD3" : false,
        "IncludePovUD4" : true,
        "IncludePovUD5" : false,
        "IncludePovUD6" : true,
        "IncludePovUD7" : false,
        "IncludePovUD8" : true,
        "IncludeMemberDetails": true,
        "IncludeRowNavigationLink" : true,
        "IncludeHasDataStatus" : true,
        "IncludeAnnotation" : true,
        "IncludeAssumptions" : true,
        "IncludeAuditComment" : true,
        "IncludeFootnote" : true,
        "IncludeVarianceExplanation" : true
    }
}
```

6. Click Send and observe the response at the bottom pane. If successful, a JSON data table will be returned otherwise a descriptive error message will show. More details will be logged in the Error and Activity logs.

This is a returned response from the request using the above body example in Postman:

```
{
  "ResultDataTableNames": [
    {
      "RowId": 0,
      "RowName": "Row1",
      "HeaderLeftLabel1": "",
      "HeaderLeftLabel2": "",
      "HeaderLeftLabel3": "",
      "HeaderLeftLabel4": "",
      "HeaderCenterLabel1": "",
      "HeaderCenterLabel2": "",
      "HeaderCenterLabel3": ""
    }
  ]
}
```

```
        "HeaderCenterLabel4": "",
        "HeaderRightLabel1": "",
        "HeaderRightLabel2": "",
        "HeaderRightLabel3": "",
        "HeaderRightLabel4": "",
        "PovCubeId": 5,
        ...
        "Col8VarianceExplanation": ""
    },
    ...
} }
```

## Data Provider GetAdoDataSetForSqlCommand endpoint

1. Create new POST request in Postman,
2. Url= http(s)://[servername]:[port]/onestreamapi/api/DataProvider/GetAdoDataSetForSqlCommand?api-version=5.2.0
3. Authorization: Type=Bearer Token. Token={{webapi\_access\_token}}
4. Headers: Content-Type=application/json
5. Body (raw / JSON):

```
{
  "BaseWebServerUrl": [your web server url],
  "ApplicationName": [your application name],
  "SqlQuery": [sql query statement used to return data],
  "DbLocation": [specify if data from an external database referenced in the configuration
will need to be returned - string - defaults to "Application" - Optional],
  "ResultDataTableName": [name of resulting table in the DataSet],
  "XFExternalDBConnectionNam ": [specify if DbLocation is set to "External"],
  "CustomSubstVarsAsCommaSeparatedPairs": [comma separated list of key value
pairs as substitution variables with the following format: "VariableName1=[
VariableValue1],VariableName2=[VariableValue2],..." - Optional]
}
```

Example:

```
{
  "BaseWebServerUrl": "http://localhost:50528/OneStream",
  "ApplicationName": "GolfStream_v37",
  "SQLQuery": "Select TOP 100 * from Cube",
  "ResultDataTableName": "ResultDataTableName",
  "DBLocation": "Application",
  "XFExternalConnectionName": "",
  "CustomSubstVarsAsCommaSeparatedPairs": ""
}
```

6. Click Send and observe the response at the bottom pane. If successful, a JSON data table will be returned otherwise a descriptive error message will show. More details will be logged in the Error and Activity logs.

This is a returned response from the request using the above body example in Postman:

```
{
  "ResultDataTableName": [
    {
      "CubeId": 0,
      "Name": "Houston",
      "Description": "Houston Clubs",
      "CubeType": 0,
      "IsTopLevelCube": false,
      "TimeDimProfileID": "664c9bd4-a314-4941-81be-513aeddac13a",
      "AccessGroupUniqueID": "e31054d8-83bf-4f79-b563-0e450342de9e",
      "MaintenanceGroupUniqueID": "e31054d8-83bf-4f79-b563-0e450342de9e",
      "ConsAlgorithmType": 0,
      "TransAlgorithmType": 0,
      "CalcNoneConsIfNoData": false,
      "CalcLocalCurrIfNoData": true,
      "CalcTransCurrsIfNoData": false,
      "CalcOwnerPreAdjIfNoData": false,
      "CalcShareIfNoData": false,
      "CalcElimIfNoData": false,
      "CalcOwnerPostAdjIfNoData": false,
      "BR1Name": "CorporateBusinessRules",
      "BR2Name": ""
    }
  ]
}
```

```
        "BR3Name": "",
        "BR4Name": "",
        "BR5Name": "",
        "BR6Name": "",
        "BR7Name": "",
        "BR8Name": "",
        "DefaultCurrencyId": 176,
        "FxRateTypeIdForRevExp": "89ce1f1c-c1cb-438e-9825-e00861a4fa5b",
        "FxRuleTypeIdForRevExp": 1,
        "FxRateTypeIdForAssetLiab": "89ce1f1c-c1cb-438e-9825-e00861a4fa5b",
        "FxRuleTypeIdForAssetLiab": 0,
        "XmlData": ""
    },
    ...
} } }
```

**IMPORTANT:** The Administrator role is required for this functionality.

## Data Provider GetAdoDataSetForMethodCommand endpoint

1. Create new POST request in Postman,
2. Url= http(s)://[servername]:[port]/onestreamapi/api/DataProvider/GetAdoDataSetForMethodCommand?api-version=5.2.0
3. Authorization: Type=Bearer Token. Token={{webapi\_access\_token}}
4. Headers: Content-Type=application/json
5. Body (raw / JSON):

```
{
  "BaseWebServerUrl": [your web server url ],
  "ApplicationName": [your application name],
  "MethodQuery": [method query to return data],
  "XFCommandMethodTypeId": [pre-defined list of XF method commands used by      XFDataProvider to
fill a DataSet],
  "ResultDataTableName": [name of resulting table in the DataSet],
  "CustomSubstVarsAsCommaSeparatedPairs": [comma separated list of key value pairs as
substitution variables with the following format: "VariableName1=[VariableValue1],VariableName2=
```

```
[VariableValue2],...] - Optional  
}
```

Example:

```
{  
  "BaseWebServerUrl": "http://localhost:50528/OneStream",  
  "ApplicationName": "GolfStream_v37",  
  "MethodQuery ": "{Houston}{Actual}{2018M1}{true}{}",  
  "XFCommandMethodTypeId ": "CertificationForWorkflowUnit",  
  "ResultDataTableName": "MyResultsTable",  
  "CustomSubstVarsAsCommaSeparatedPairs": ""  
}
```

**XFCommandMethodTypeId** may take any values from the list below:

```
"WorkflowCalculationEntities"  
"WorkflowConfirmationEntities"  
"WorkflowProfileAndDependentProfileEntities"  
"WorkflowProfileEntities"  
"WorkflowProfiles"  
"WorkflowProfileRelatives"  
"WorkflowStatus"  
"WorkflowStatusTwelvePeriod"  
"WorkflowAndEntityStatus"  
"JournalsForWorkflowUnit"  
"FormsStatusForWorkflowUnit"  
"ConfirmationForWorkflowUnit"  
"CertificationForWorkflowUnit"  
"ICMatchingForWorkflowUnit"  
"ICMatchingForWorkflowUnitMultiPlug"  
"ICMatchingForWorkflowUnitMultiPeriod"  
"ICMatchingPlugAccountsForWorkflowUnit"
```

6. Click Send and observe the response at the bottom pane. If successful, a JSON data table will be returned otherwise a descriptive error message will show. More details will be logged in the Error and Activity logs.

This is a returned response from the request using the above body example in Postman:

```
{  
  "MyResultsTable": [  
    {  
      "WorkflowCalculationEntities": "WorkflowCalculationEntities"  
    }  
  ]  
}
```



```
{
  "ProfileName": "Houston",
  "ProfileKey": "2f3a719e-8e26-4d8c-8cc7-4544a4812673",
  "ProfileOrder": 1,
  "ScenarioName": "Actual",
  "ScenarioKey": 0,
  "TimeKey": 2018003000,
  "TimeName": "2018M1",
  "CertProfileKey": "003e0a15-6c9a-412c-90ba-64d31040c314",
  "CertName": "Plant Certification",
  "CertDescription": "Plant Certification",
  "CertSignOffState": "Inprocess",
  "CertIsCertified": false,
  "CertCanCertify": false,
  "CertIsParentCertified": false,
  "CertAreDependantsCertified": false,
  "CertAllAnswered": false,
  "CertQuestionCount": 3,
  "CertUnansweredCount": 3,
  "CertUnansweredRate": 1.0,
  "GroupKey": "7c7fedcd-f04a-4f5b-ba13-ed1097f449a9",
  "GroupName": "SOX Plant Controller",
  "GroupDescription": "SOX Plant Controller",
  "GroupSignOffState": "Inprocess",
  "GroupAllAnswered": false,
  "GroupQuestionCount": 3,
  "GroupUnansweredCount": 3,
  "GroupUnansweredRate": 1.0,
  "QuestionUniqueID": "8a92f59c-2419-49d2-87b7-1cdfb21c7072",
  "QuestionName": "Unusual Transactions",
  "QuestionCategory": "InternalAudit",
  "QuestionRiskLevel": "High",
  "QuestionFrequency": "AllTimePeriods",
  "TimeFilterForReqFreq": "",
  "QuestionText": "Any unusual transactions booked? If so, explain. ",
}
```

```
"QuestionResponse": "-1",
"QuestionComments": "",
"QuestionResponseOptional": false,
"QuestionDeactivated": false,
"QuestionDeactivationDate": "1900-01-01T00:00:00",
"QuestionDisplayOrder": 10
},
{
  "ProfileName": "Houston",
  "ProfileKey": "2f3a719e-8e26-4d8c-8cc7-4544a4812673",
  "ProfileOrder": 1,
  "ScenarioName": "Actual",
  "ScenarioKey": 0,
  "TimeKey": 2018003000,
  "TimeName": "2018M1",
  "CertProfileKey": "003e0a15-6c9a-412c-90ba-64d31040c314",
  "CertName": "Plant Certification",
  "CertDescription": "Plant Certification",
  "CertSignOffState": "Inprocess",
  "CertIsCertified": false,
  "CertCanCertify": false,
  "CertIsParentCertified": false,
  "CertAreDependantsCertified": false,
  "CertAllAnswered": false,
  "CertQuestionCount": 3,
  "CertUnansweredCount": 3,
  "CertUnansweredRate": 1.0,
  "GroupKey": "7c7fedcd-f04a-4f5b-ba13-ed1097f449a9",
  "GroupName": "SOX Plant Controller",
  "GroupDescription": "SOX Plant Controller",
  "GroupSignOffState": "Inprocess",
  "GroupAllAnswered": false,
  "GroupQuestionCount": 3,
  "GroupUnansweredCount": 3,
  "GroupUnansweredRate": 1.0,
```

```
"QuestionUniqueID": "78e102c2-cda5-4c07-b853-416d83de5706",
"QuestionName": "Audit Transactions",
"QuestionCategory": "ExternalAudit",
"QuestionRiskLevel": "High",
"QuestionFrequency": "AllTimePeriods",
"TimeFilterForReqtFreq": "",
"QuestionText": "Any transactions to be reviewed by external audit? If so, explain. ",
"QuestionResponse": "-1",
"QuestionComments": "",
"QuestionResponseOptional": false,
"QuestionDeactivated": false,
"QuestionDeactivationDate": "1900-01-01T00:00:00",
"QuestionDisplayOrder": 20
},
{
  "ProfileName": "Houston",
  "ProfileKey": "2f3a719e-8e26-4d8c-8cc7-4544a4812673",
  "ProfileOrder": 1,
  "ScenarioName": "Actual",
  "ScenarioKey": 0,
  "TimeKey": 2018003000,
  "TimeName": "2018M1",
  "CertProfileKey": "003e0a15-6c9a-412c-90ba-64d31040c314",
  "CertName": "Plant Certification",
  "CertDescription": "Plant Certification",
  "CertSignOffState": "Inprocess",
  "CertIsCertified": false,
  "CertCanCertify": false,
  "CertIsParentCertified": false,
  "CertAreDependantsCertified": false,
  "CertAllAnswered": false,
  "CertQuestionCount": 3,
  "CertUnansweredCount": 3,
  "CertUnansweredRate": 1.0,
  "GroupKey": "7c7fedcd-f04a-4f5b-ba13-ed1097f449a9",
```

```
    "GroupName": "SOX Plant Controller",
    "GroupDescription": "SOX Plant Controller",
    "GroupSignOffState": "Inprocess",
    "GroupAllAnswered": false,
    "GroupQuestionCount": 3,
    "GroupUnansweredCount": 3,
    "GroupUnansweredRate": 1.0,
    "QuestionUniqueID": "3d9c4dcc-75fd-4568-b224-f7e428622917",
    "QuestionName": "Key Data Review",
    "QuestionCategory": "FinancialStatementReview",
    "QuestionRiskLevel": "MediumLow",
    "QuestionFrequency": "AllTimePeriods",
    "TimeFilterForReqFreq": "",
    "QuestionText": "Have all key metrics been reviewed? ",
    "QuestionResponse": "-1",
    "QuestionComments": "",
    "QuestionResponseOptional": false,
    "QuestionDeactivated": false,
    "QuestionDeactivationDate": "1900-01-01T00:00:00",
    "QuestionDisplayOrder": 30
  }
],
"MyResultsTable_SignOffCert": [
  {
    "ProfileKey": "2f3a719e-8e26-4d8c-8cc7-4544a4812673",
    "ScenarioKey": 0,
    "TimeKey": 2018003000,
    "CertProfileKey": "003e0a15-6c9a-412c-90ba-64d31040c314",
    "SignOffState": "Inprocess",
    "Comments": "Sign-Off Initialized",
    "UserKey": "2b61ed59-63ae-46f2-89aa-a8ee9f14bacd",
    "UserName": "TestUserOkta",
    "UserIPAddress": "8d3d857e-cd62-4fd9-a2ec-43b46217a036",
    "TimeStamp": "2019-11-18T14:45:00.007"
  }
]
```

```
],  
  "MyResultsTable_SignOffGroups": [  
    {  
      "ProfileKey": "2f3a719e-8e26-4d8c-8cc7-4544a4812673",  
      "ScenarioKey": 0,  
      "TimeKey": 2018003000,  
      "CertProfileKey": "003e0a15-6c9a-412c-90ba-64d31040c314",  
      "CertProfileName": "Plant Certification",  
      "GroupKey": "7c7fedcd-f04a-4f5b-ba13-ed1097f449a9",  
      "GroupName": "SOX Plant Controller",  
      "SignOffState": "Inprocess",  
      "Comments": "Sign-Off Initialized",  
      "UserKey": "2b61ed59-63ae-46f2-89aa-a8ee9f14bacd",  
      "UserName": "TestUserOkta",  
      "UserIPAddress": "8d3d857e-cd62-4fd9-a2ec-43b46217a036",  
      "TimeStamp": "2019-11-18T14:45:00.2"  
    }  
  ]  
}
```

**IMPORTANT:** The Administrator role is required for this functionality.

## Authentication Execute LogonAndReturnCookie endpoint

Returns a one-time cookie value that indicates authentication state. Used mostly by enablement team to verify the installation of web API completed successfully.

1. Create new POST request in Postman,
2. Url= http(s)://[servername]:  
[port]/OneStreamApi/api/Authentication/LogonAndReturnCookie?api-version=5.2.0
3. Authorization: Type=Bearer Token. Token={{webapi\_access\_token}}
4. Headers: Content-Type=application/json
5. Body (raw / JSON):

## REST API Overview

---

Arguments:

**"BaseWebServerUrl"**: [your web server url],

**"ApplicationName"** : [name of Application attempted to access]

<response code="200">Returns a JSON representation of the resulting DataSet.</response>

<response code="400">Bad Request. Missing Authentication arguments. </response>

<response code="500">Error Message. Authentication Failed. Please check the Error Log for more details</response>

Click Send and observe the response at the bottom pane. If successful, a one-time cookie value that indicates authentication state will be returned. Otherwise the error message will be shown. More details will be logged in the Error and Activity logs.

The screenshot shows the REST client interface with the URL `http://localhost:3403/api/Authentication/LogonAndReturnCookie?api-version=5.2.0`. The **Authorization** tab is selected. Under the **TYPE** dropdown, **Bearer Token** is chosen. A warning message states: "Heads up! These parameters hold sensitive data. To keep this data secure while working in a recommend using variables. [Learn more about variables](#)". Below this, the **Token** field contains the variable `{{webapi_access_token}}`. A **Preview Request** button is visible at the bottom left.

The screenshot shows the REST client interface with the same URL. The **Body** tab is selected. The **Body** type is set to **JSON (application/json)**. The JSON body is defined as follows:

```
1 {  
2   "BaseWebServerUrl": "http://localhost:50528/OneStream",  
3   "ApplicationName": "GolfStream_v37"  
4 }
```

# Configure OneStream API for External Authentication

We support REST API authentication with Azure AD, Okta and PingFederate. Perform the configuration for your provider:

- "PingFederate Configuration" on page 35
- "Okta Configuration" on page 29
- "Azure AD Configuration" below

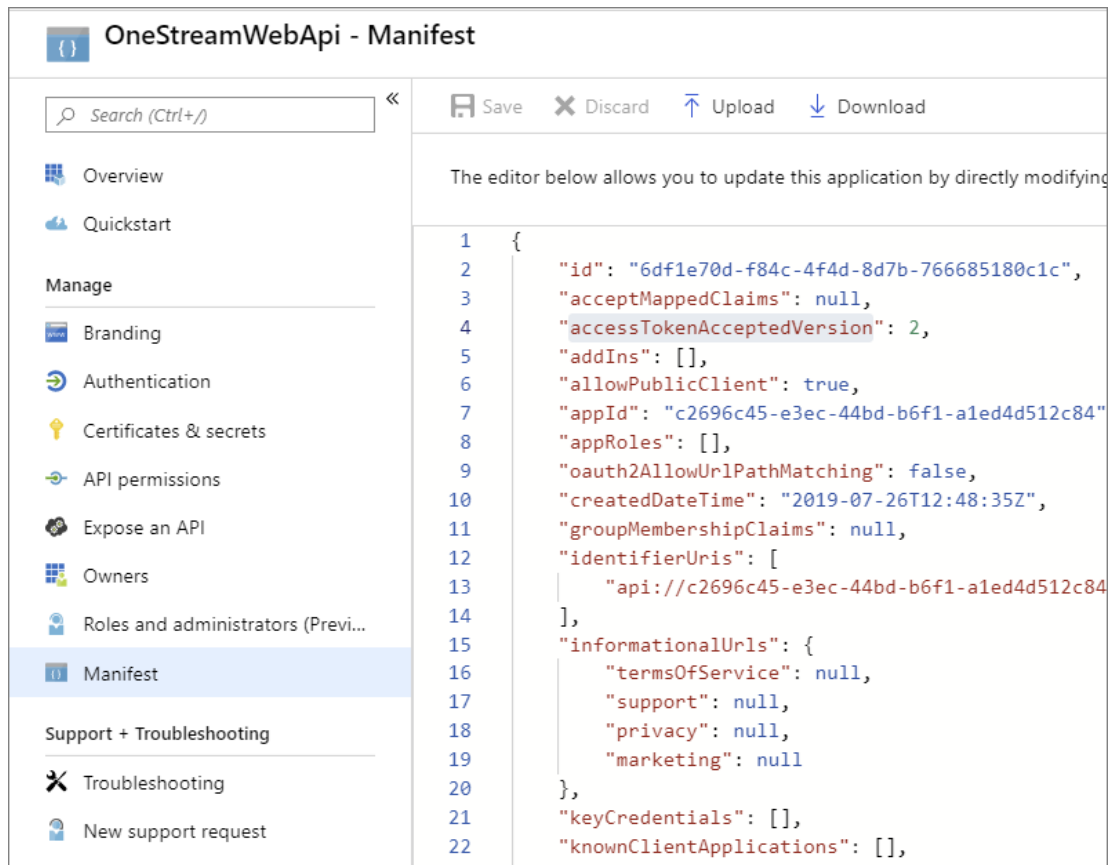
## Azure AD Configuration

Register an application:

- Sign in to the Azure portal.
- To the left, select the **Azure Active Directory** service, and then **App registrations > New registration**.
- On the **Register an application** page, enter the application's registration name and click Register.
- On the Application's **Overview** tab, note {Client Id}, {Tenant Id}
- On **Authentication > Advanced Settings** select Access and ID tokens.
- In **Default Client Type**, select **Yes** for **Treat application as a public client**.
- In Supported Account types, select **Accounts in this organizational directory only (Default Directory)**.
- Save.
- In **Certificates & secrets**, add **New client secret** and note the value. Save.
- In **Expose an API** tab, add a custom scope needed for user-machine use case. Note the scope name and the {AppId Uri} values.

## Configure OneStream API for External Authentication

- Save.
- We support v2.0 Azure endpoints, so on **Manifest**, find **accessTokenAcceptedVersion**. Set the value to **2**.



- Save.





# Okta Configuration


## Create M2M Application Registration (grant\_type = client\_credentials)

- Create a new OAuth Service App.
- Add a name, then click **OK**.
- Note the Client ID and Client Secret values. Copy these values.

[← Back to Applications](#)



### OneStreamWebApiClientCredentials

Active  [View Logs](#)

General


---

#### General Settings Edit



APPLICATION

Application label OneStreamWebApiClientCredentials

#### Client Credentials Edit

Client ID  

Public Identifier for the client that is required for all OAuth flows.

Client secret   


- Perform the tasks in the next section.

### Create U2M Application Registration (grant\_type = password) in Okta


- Create a Native App.
- Specify a Label and select the **Allowed Grant Types** shown below.
- Note the Logout Redirect URIs, Client ID, and Client Secret values. Copy these values.
- Select **Use Client Authentication**.

## Configure OneStream API for External Authentication

[← Back to Applications](#)



### OneStreamWebApiUserCreds

Active  [View Logs](#)

General Sign On Assignments

#### General Settings Edit

##### APPLICATION

Application label **OneStreamWebApiUserCreds**


Application type **Native**


Allowed grant types

Client acting on behalf of a user

- ☒ Authorization Code
- ☒ Refresh Token
- ☒ Resource Owner Password
- ☐ Implicit (Hybrid)

##### LOGIN

Login redirect URIs  **com.oktapreview.dev-992535/callback**

Logout redirect URIs 

Initiate login URI

## Configure OneStream API for External Authentication

---

Client Credentials

Edit

Client ID

00aj9gjoukO0aRHPL0h7

Public identifier for the client that is required for all OAuth flows.

Client authentication

☐ Use PKCE (for public clients)

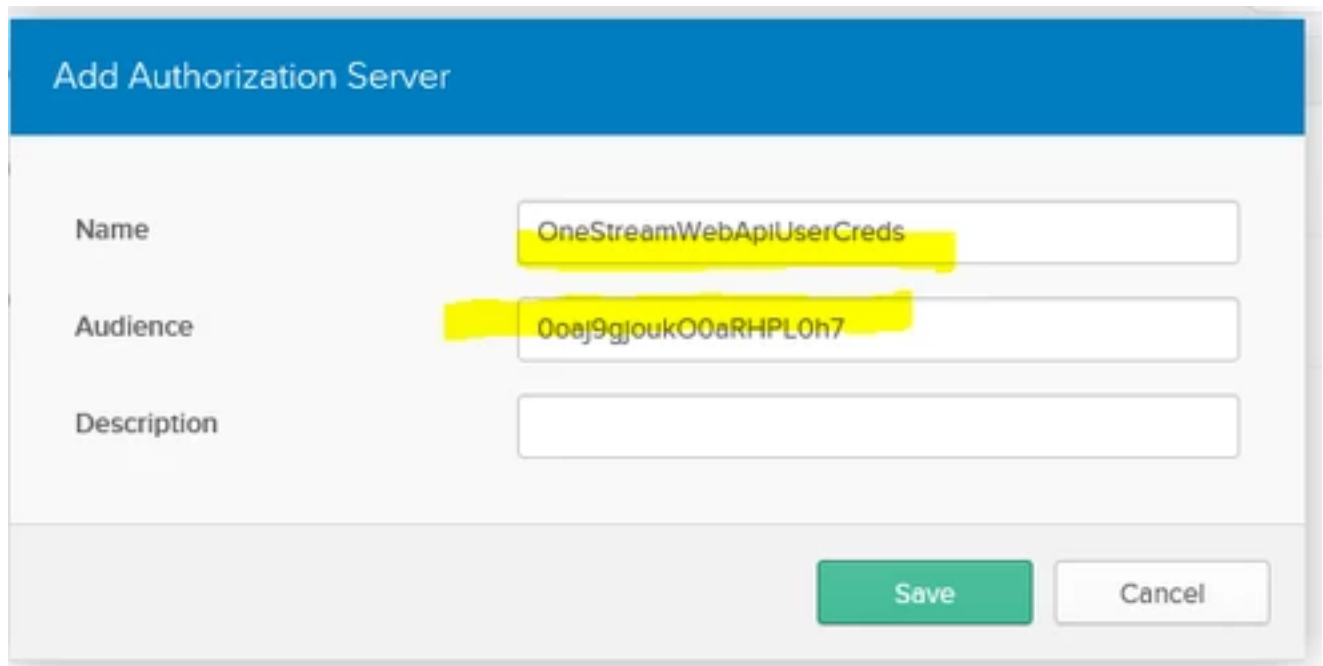
Uses Proof Key for Code Exchange (PKCE) instead of a client secret. A one-time key is generated by the client and sent with each request. Instead of proving the identity of a client, this ensures that only the client which requested the token can redeem it.

☒ Use Client Authentication

Not secure for distributed native apps. A client secret is embedded in the client and is sent with requests, proving the identity of the client.

Client secret

- Select **API > Authorization servers**.
- Click **Add Authorization Server**, provide a name, and set **Client ID** as the **Audience**.



**Add Authorization Server**

Name: OneStreamWebApiUserCreds

Audience: 00aj9gjouk00aRHPL0h7

Description:

Save Cancel

- Click **Save**.
- Add a custom scope to support the Machine-to-Machine scenario. For example:









## Configure OneStream API for External Authentication

OneStreamWebApiUserCreds Help

Active ▾

Settings **Scopes** Claims Access Policies Token Preview

+ Add Scope

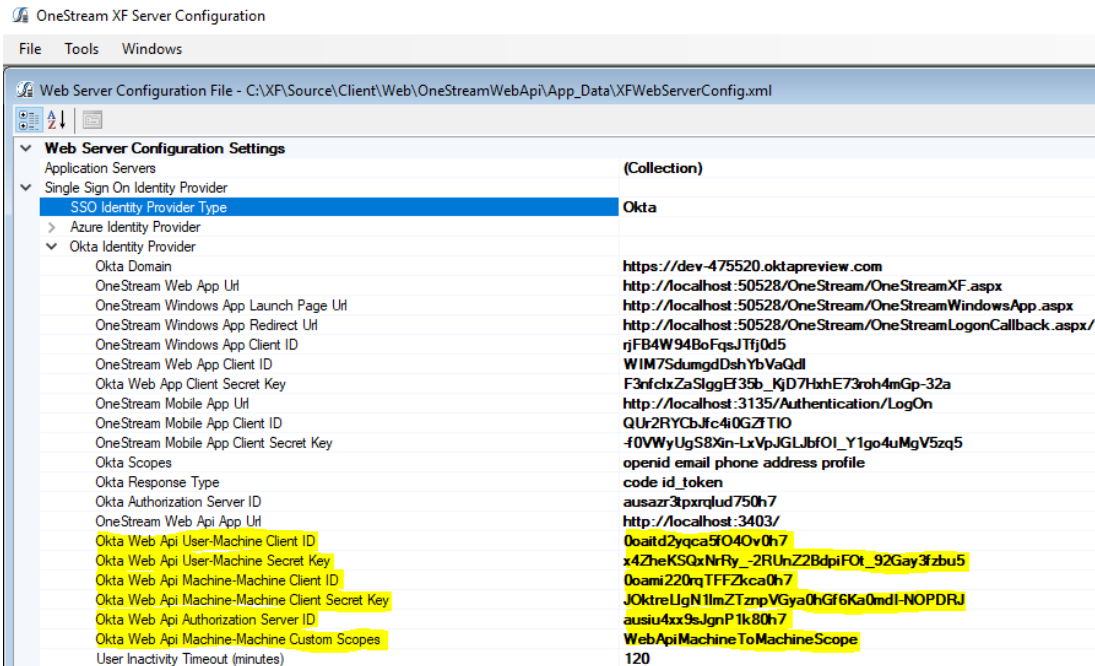
Name	Description	Default Scope	Metadata Publish	Actions
WebApiMachineToMachineScope	WebApiMachineToMachineScope	No	No	 
openid	Signals that a request is an OpenID request.	No	Yes	
profile	Allows this application to access your profile information.	No	Yes	
email	Allows this application to access your email address.	No	Yes	
address	Allows this application to access your address.	No	Yes	
phone	Allows this application to access your phone number.	No	Yes	
offline_access	Allows this application to access your data when you aren't using the application.	No	Yes	

- Perform the tasks in the next section.

## Update the Server Configuration Utility

- Launch the Server Configuration utility and enter the values for the four Web Api properties. For example:

## Configure OneStream API for External Authentication



- Save the file.
- Restart IIS.

## PingFederate Configuration

1. Add a new client in PingFederate Admin Console representing the OneStreamWeb Api application.
2. Create a U2M Application Registration (grant\_type = password).
3. Create a new Access Token Manager (ATM) and note the **Access Token Manager ID** value (Client Credentials). See Appendix 9.10 for instructions on how to setup a ATM in Ping Admin console.
4. Set **Client Authentication** to the **Client Secret** option.
5. Click **Generate Secret**, then **Update** and note the Client ID and Client Secret values.
6. In **Allowed Grant Types**, select **Authorization Code**, **Resource Owner Password Credentials**, **Client Credentials**.

## Configure OneStream API for External Authentication

7. Set **Default Access Token Manager** to the value in step 1.
8. Save.

### Client

Manage the configuration and policy information about a client.

CLIENT ID **OneStreamWebApi**

NAME **OneStreamWebApi**

DESCRIPTION **OAuth Client representing OneStreamWebApi application in PF**

CLIENT AUTHENTICATION

☐ NONE

☒ CLIENT SECRET

SECRET **XXXXXXXXXXXX** **Generate Secret**

☐ CHANGE SECRET

☐ CLIENT TLS CERTIFICATE

☐ PRIVATE KEY JWT

REQUIRE SIGNED REQUESTS ☐

REDIRECT URIS **Redirection URIs**

**https://w12r2-pingid.onestream.com:9031/as/token.oauth2/callback**

**Action**

[Edit](#) | [Delete](#)

**BYPASS AUTHORIZATION APPROVAL** ☐ Bypass

**RESTRICT COMMON SCOPES** ☐ Restrict

**EXCLUSIVE SCOPES** ☐ Allow Exclusive Scopes

**ALLOWED GRANT TYPES**

☒ Authorization Code

☒ Resource Owner Password Credentials

☒ Refresh Token

☐ Implicit

☒ Client Credentials

☐ Access Token Validation (Client is a Resource Server)

☐ Extension Grants

**RESTRICT RESPONSE TYPES** ☐ Restrict

**DEFAULT ACCESS TOKEN MANAGER** **Client Credentials** ▼

**VALIDATE AGAINST ALL ELIGIBLE ACCESS TOKEN MANAGERS** ☐

**PERSISTENT GRANTS EXPIRATION** ☐ Use Global Setting



# Update the Server Config Utility

1. Launch the Server Configuration utility and enter the values for the four Web API properties.
2. Save.
3. Restart IIS.

## Configure the AUD Value

In some installations, the Audience value is not used in the authentication process. Normal processing will cause authentication to fail if this value is not used. The **Validate Audience** option allows for disabling audience validation for these installations.

By default, this setting is **True**, which means the audience will be validated.

1. In the Web Server Configuration file, select **Single Sign On Identity Provider**.
2. In **PingFederate Identity Provider**, click the ellipsis for more details.
3. In **Validate Audience**, select **False** to disable Audience validation.

