



Send your API Information to your 3rd Party Vendor

The API URL, Key, and Secret are the 3 pieces of information needed to complete your API configuration. These pieces of information will be given to the 3rd party vendor by you so they can successfully use the API.

- 1. Determine which API the vendor uses, it will either be the **LMS API**, or the **OneRoster API**. The vendor will be able to tell you which API they use.
- 2. Once you know which API is used, determine your API URL based on the examples below. You will need to give the vendor your API URL.

LMS API URL is:

On Premises hosted customers: <https://school.k12.state.us/api>

ISCorp hosted customer: <https://skyward.iscorp.com/APIdistrictnameSTU>

Note: Be sure to change the sample URLs above to use your districts URL settings.

OneRoster API URL is:

On Premises hosted customers: <https://school.k12.state.us/api/ims/oneroster/v1p1/>

ISCorp hosted customers: <https://skyward.iscorp.com/APIdistrictnameSTU/ims/oneroster/v1p1/>

Note: Be sure to change the sample URLs above to use your districts URL settings.

- 3. Share the API URL, Key, and Secret/Key with the 3rd party vendor, you are finished!

API Configuration in SMS

- 1. The API Configuration Options and API Logging can be found by browsing to Product Setup → Contact Access → District Setup → Configuration → API Configuration.

These fields are not editable as the info is gathered when the third party connects to the API URL.

API Configuration

Appserver: :5162/asSkyMon

Authentication URL: https:// /api/token?grant_type=password&username=COMPASLEARNING&

Application URL: https:// /api/api?grant_type=password&username=COMPASLEARNING&pa

Last updated on 10/03/2016 16:27:28.997-05:00

LMS/OneRoster API Configuration

- Allow Combined Gradebook Assignments to Post Once for Each Section
- Automatically Restore Valid Deleted Assignments When Edited

Buttons: Save, Back, LMS/OneRoster API Log

Congratulations the API Server Install and Configuration is Complete!



Advanced API Information

LMS API Data Objects

The following list of Data Objects have been made available to 3rd party vendors that consume the LMS API. Documentation on the supported endpoints are found using the [Swagger Interface](#).

If the LMS/OneRoster API license is purchased the LMS API allows pass back for LMS API objects. If the license is not purchased the LMS API allows read only access to all data objects listed below.

LMS API Version 1

- AbsenceReasons
- AbsenceTypes
- Assignments
- AssignmentScores
- Courses
- Departments
- District
- EmergencyContacts
- Enrollments
- Errors
- GradingCategories
- GradingCategoryCodes
- GradingPeriods
- Guardians
- Names
- Relationships
- Schools
- Sections
- Settings
- SpecialCodes
- StaffMembers
- StandardGrades
- Standards
- Students
- Subjects



OneRoster API Data Objects

The following list of Data Objects have been made available to 3rd party vendors that consume the OneRoster API. Documentation on the supported endpoints are found on the [IMS Global](#) website. If the LMS/OneRoster API license is purchased the OneRoster API allows pass back for Categories, LineItems (Assignments), and Results (Assignments Scores). If the license is not purchased the OneRoster API allows read only access to all data objects listed below.

OneRoster API Version 1.1

- categories
- results
- lineitems
- academicssessions
- classes
- courses
- demographics
- enrollments
- gradingperiods
- orgs
- schools
- students
- teachers
- terms
- users



Advanced Configuration Options

The advanced configuration options are not necessary for most customers and are intended for special configurations where multiple environments are hosted on a shared set of web servers, or when a customer has multiple load balanced web servers.

Configuring Multiple API Instances on a Shared Web Server

The template configuration file [skyward]\ocx\skywardapi\appsettings.config.template has examples to define multiple running instances of the API on servers that are setup with multiple Skyward environments, like a Cloud Hosted server.

To run multiple API instances each instance of the API must have a set of unique keys defined for each environment. The IIS Virtual Application must match the name of the unique keys.

To configure multiple applications, copy the Multiple Applications appSettings section to the top of the appsettings.config.template file, overwriting the existing appSettings section, and then save the file as [skyward]\ocx\skywardapi\appsettings.config

```
<!--
*****
* Multiple Applications Example (Cloud Hosted) *
*****
-->
<appSettings>
  <add key="DistrictOneAPI_NSHost" value="myserver" />
  <add key="DistrictOneAPI_NSPort" value="5162" />
  <add key="DistrictOneAPI_ASName" value="asAppserverDistrictOne" />

  <add key="DistrictTwoAPI_NSHost" value="myserver" />
  <add key="DistrictTwoAPI_NSPort" value="5162" />
  <add key="DistrictTwoAPI_ASName" value="asAppserverDistrictTwo" />
</appSettings>
```

For Example , set the key values to “DistrictOneAPI” and then set the configuration values for this customer within this configuration section. Once that is done, set the key values for “DistrictTwoAPI” and set the configuration values for District Two. You can support as many instances of the API on any one IIS Web Server as you wish by adding additional keys and setting the values.

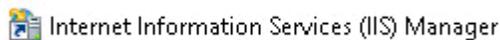
In IIS you must create an API Virtual Application for each configuration section, for the example above you would need an API Virtual Application named “DistrictOneAPI” and another named “DistrictTwoAPI”. Your API URLs would be: https://{DNSNAME}/DistrictOneAPI and https://{DNSNAME}/DistrictTwoAPI



Additional IIS Setup for Load Balanced API Servers

If your district has more than one web server that is load balanced, please follow these steps to setup the API for load balancing. The API Server must be installed on all Skyward Web Servers that participate in the load balancing of the Skyward Web Applications.

1. Select **one** of your **Skyward Web Servers** that has the **API Server** Installed. Open the **Administrative Tools** Control Panel → Open the **Internet Information Services (IIS) Manager**.



2. Expand the **IIS Server** → Expand **Sites** → Expand the **Skyward Web Site** → Select the **API Application** → Click on the **Machine Key** icon.



Machine Key

3. Under **Validation Key** → **Uncheck** the option to automatically **Generate at Run Time** → **Uncheck** the option to **Generate a Unique Key** for each application → Under **Decryption Key** → **Uncheck** the option to automatically **Generate at Run Time** → **Uncheck** the option to **Generate a Unique Key** for each application.



Use this feature to specify hashing and encryption settings for application services, such as view state, Forms authentication, membership and roles, and anonymous identification.

Validation method:

SHA1

Encryption method:

Auto

Validation key

Automatically generate at runtime

Generate a unique key for each application

E7F455A8E7460B7BFC1CC099A8338F062E0413AED401DDDC932C33414C35ED1DFD1D5C7BE

Decryption key

Automatically generate at runtime

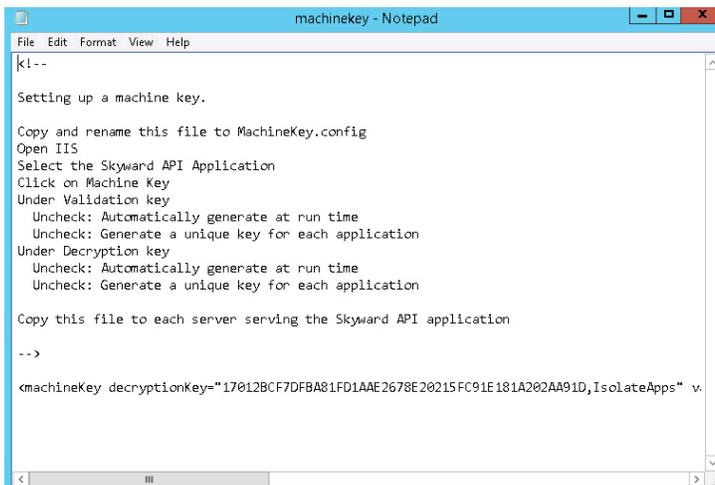
Generate a unique key for each application

17012BCF7DFBA81FD1AAE2678E20215FC91E181A202AA91D

4. On the right-hand side → Click **Generate Key** → Click **Apply**



5. Open the {**Skyward program path**}\ocx\skywardapi**MachineKey.config** file using **Notepad** and you will notice it now has a **Machine Key** populated. → No changes are necessary → Close the file after verifying the Machine Key is populated.



```
machinekey - Notepad
File Edit Format View Help
|k|--
Setting up a machine key.
Copy and rename this file to MachineKey.config
Open IIS
Select the Skyward API Application
Click on Machine Key
Under Validation key
  Uncheck: Automatically generate at run time
  Uncheck: Generate a unique key for each application
Under Decryption key
  Uncheck: Automatically generate at run time
  Uncheck: Generate a unique key for each application
Copy this file to each server serving the Skyward API application
-->
<machineKey decryptionKey="17012BCF7DFBA81FD1AAE2678E20215FC91E181A202AA91D,IsolateApps" v.
```

5. Copy the **MachineKey.config** file noted above to each **Web Server** {**Skyward program path**}\ocx\skywardapi\ folder that has the **API Server** installed.