PUBLIC

DOCUMENT VERSION: 2023-11-17

Virtual Learning Service Configuration

Microsoft Teams

SAP SuccessFactors

Table of Contents

Overview

This guide provides detailed configuration information to enable SuccessFactors Learning to use Microsoft Teams as a Virtual Learning Service for instructor-led classes.

Note: It is strongly recommended that you work with a Microsoft partner to register your Microsoft Teams app and complete the required tasks within the Microsoft Azure portal.

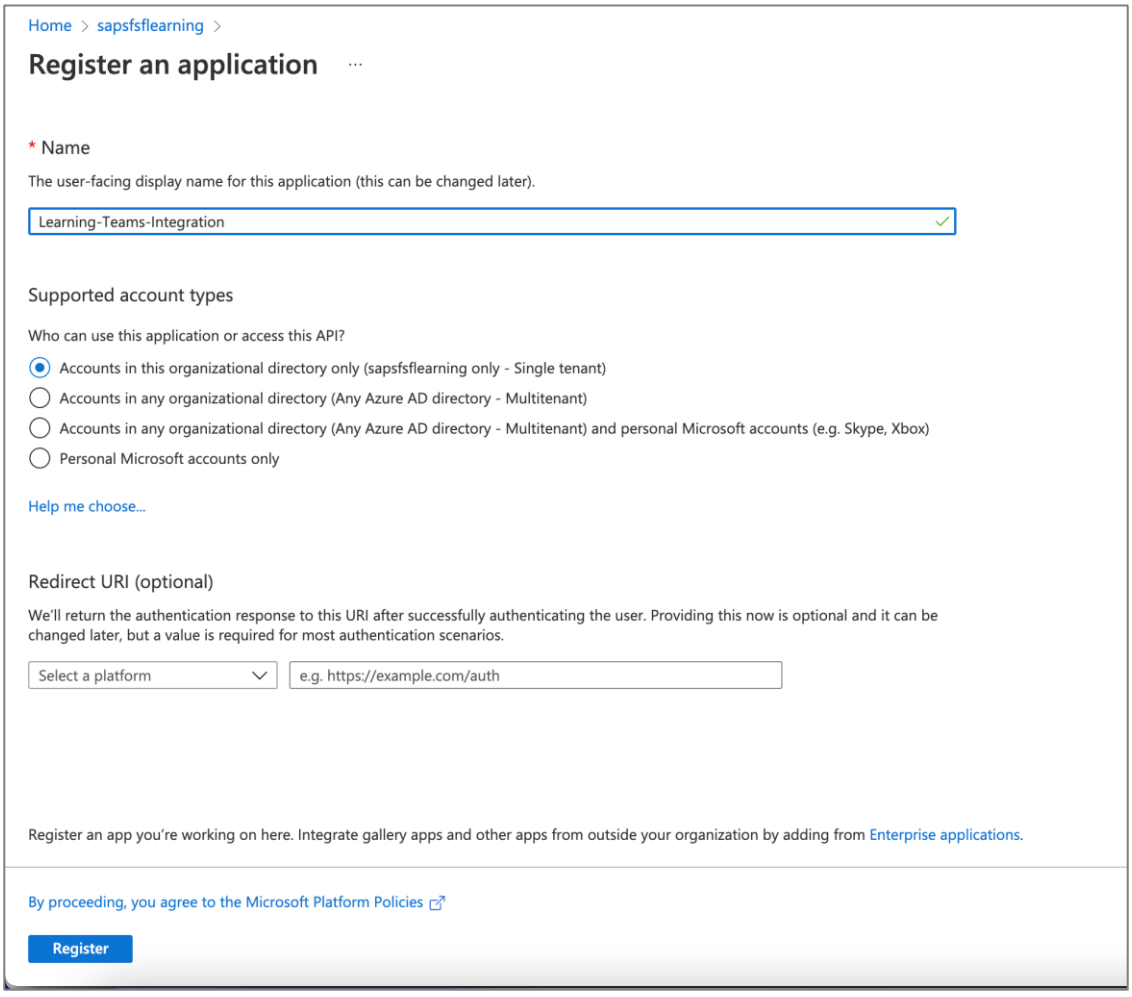
Considerations

- The instructor's ID is their Microsoft Teams email address. No password is required.
- Users can join a Microsoft Teams meeting through a URL, but only users that are registered for the class receive credit for attendance.
- If users have an alternate email address in Microsoft Azure, you must provide it in the user profile for attendance to be properly processed. You also need to specify which custom field contains the alternate email address in the VLS configuration for Microsoft Teams (Learning Administration > System Administration > Configuration > VLS Configuration).

Configuration Within Microsoft Azure

Register the Application

1. Log in to your Azure portal (https://portal.azure.com).
2. Navigate to Azure Active Directory (AD).
3. Navigate to **App registrations** and then choose **New registration**.
4. Enter the desired name.
5. For the **Supported account** types, select the first or second option. Personal Microsoft accounts aren't supported.



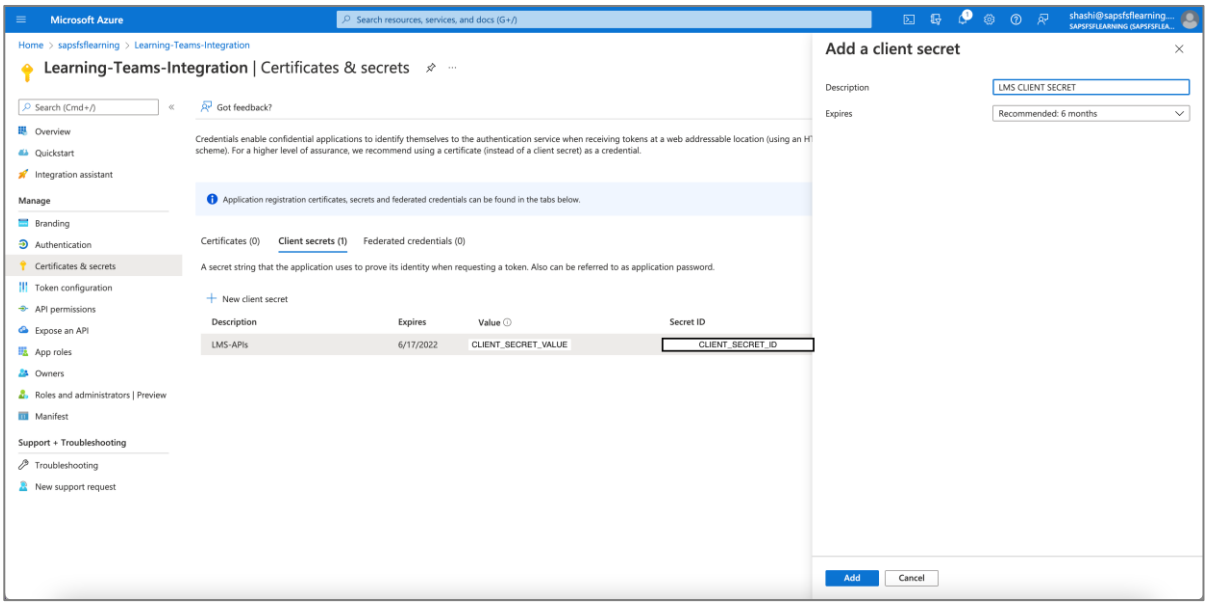
6. Choose **Register**.

7. Copy the values in the **Application (client) ID** and **Directory (tenant) ID** fields and then save in a secure, temporary location. These values are required when setting up the integration in SuccessFactors Learning Administration.

   – **Note**: A service principal is generated when an app is created in Azure AD. The credentials (application ID, directory ID, and app secret) associated with the service principal are used to authenticate the integration with your Azure AD. This allows SuccessFactors Learning to connect to your Microsoft Graph APIs.

Generate a Client Secret

1. Within your Azure portal, navigate to **Certificates & secrets** and then choose **New client secret** in the **Client secrets** section.
2. Complete the client secret properties as desired, then choose **Add**.

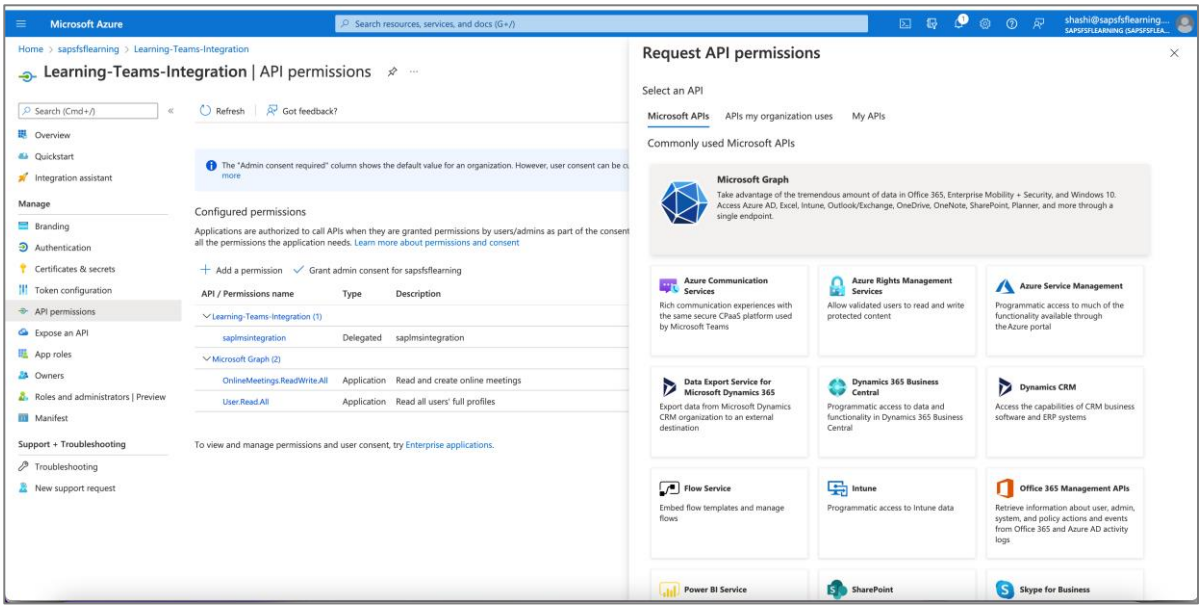


3. Copy the client secret in the **Value** column and then save in a secure, temporary location. This value is required when setting up the integration in SuccessFactors Learning Administration.

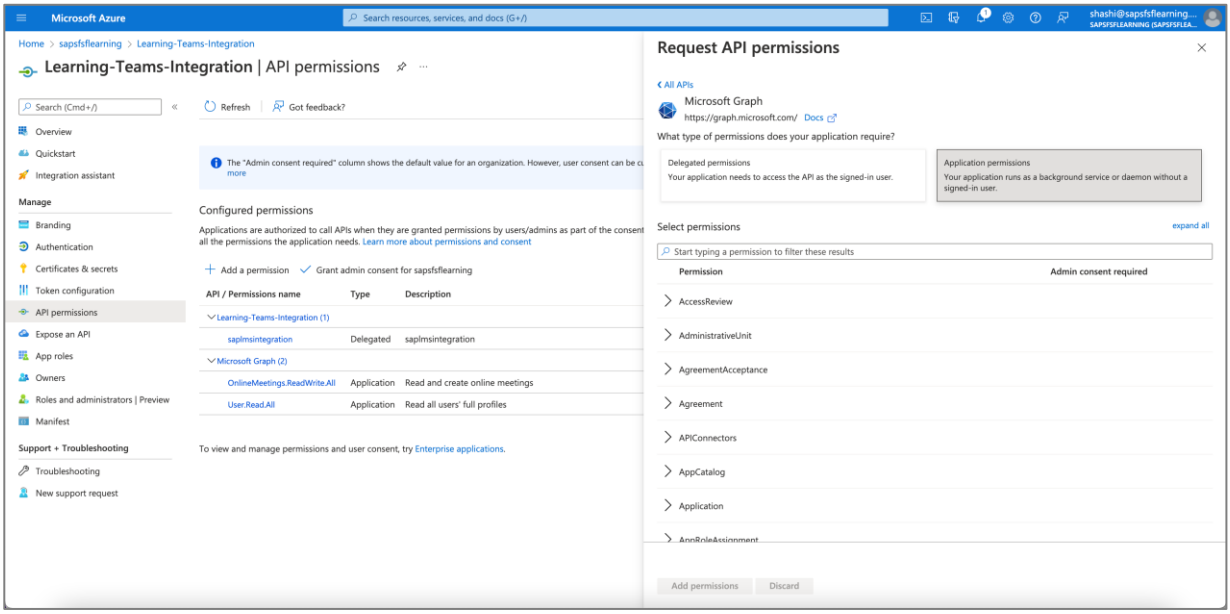
   – **Note**: When you navigate away from this page, the client secret value is masked and can't be copied. If you haven't copied the secret value, you'll need to create new client secret.

Set Microsoft Graph API Permissions

1. Within your Azure portal, navigate to **API permissions** and then choose **Add a permission.**



2. Choose **Microsoft Graph** and then choose **Application permissions**.



3. Select the following permissions:

Required Permission	Purpose
OnlineMeetings.ReadWrite.All	Used by the Learning API to schedule, update, and delete meetings.
User.Read.All	Used by the Learning API to enroll and withdraw users from a virtual time slot.
OnlineMeetingArtifact.Read.All	Used by the Learning API to process meeting attendance.

4. Choose **Add permissions**.
5. Navigate to **API permissions** and choose **Grant admin consent for [your Azure tenant name]** and then choose **Yes**. The **Status** column should display "Granted for [your Azure tenant name]" for all API permissions.

Configure the CsApplicationAccessPolicy for Application Authentication

The CsApplicationAccessPolicy policy controls which users the integration can act on behalf of. It is important that the integration can act on behalf of instructors so that it can retrieve meeting attendance information.

Prerequisites

- The steps need to be performed by an administrator with admin access to Microsoft Teams.
- You must have Powershell 5.1 or later. You can check your version by running `$Host.version` within PowerShell. Microsoft has installation instructions here: https://docs.microsoft.com/en-us/skypeforbusiness/set-up-your-computer-for-windows-powershell/download-and-install-windows-powershell-5-1
- You must have Microsoft Teams PowerShell Module installed. Microsoft has installation instructions here: https://docs.microsoft.com/en-us/microsoftteams/teams-powershell-install

Configuration Steps

1. In Powershell, connect to your Microsoft Teams using the `Connect-MicrosoftTeams` command. This will open a browser and allow you to log in with your Microsoft credentials. After successful login, the running PowerShell instance will be able to administer the Microsoft Teams instance that is associated with the account you used to log in to Microsoft.

2. Create a policy that contains the app id for the registered app you're working with.
    a. Save the Azure registered app id (GUID/UUID) into a variable: `$appId = "your-app-id"`
    b. Save the policy name into a variable: `$policyName = "Your MS Teams Policy"`
    c. Create a new policy: `new-csapplicationaccesspolicy -identity $policyName`
    d. Assign the registered application to the policy: `Set-CsApplicationAccessPolicy -identity $policyName -appids @{add=$appId }`
3. Assign that policy either globally or to individual users who will be instructors in the integration:
    - To assign the policy globally to all users use this command: `Set-CsApplicationAccessPolicy -identity "Global" -appIds @{add=$appid}`
    - To assign the policy to an individual user only use this command: `Grant-CsApplicationAccessPolicy -identity "instructorUsername@example.com" -PolicyName $policyName`

      Note: You need to run this command for each instructor separately, you cannot add multiple instructors in a single command.

Configuration Within SuccessFactors Learning

When you configure the Virtual Learning Service (VLS), you create an API connection to a virtual meeting room vendor. That connection manages the virtual training sessions: learners can launch them from their assignments and learning administrators can schedule them from the administration environment.

Create a New VLS Configuration File

1. Go to SuccessFactors Learning Administration > System Administration > Configuration > System Configuration and edit **LMS_ADMIN**.
2. In the **LMS_ADMIN** file, find `vleEnabled` and change its value to `true`. If you have created previous VLS configuration files, this will already be set to `true`.
3. Go to System Administration > Configuration > VLS Configuration and choose **Add New**.
4. Select Microsoft Teams from the **VLS Vendor** dropdown list to populate the default configuration template.
5. Edit the file and paste the values you copied from the Azure portal into the corresponding elements, as described in the following table:

Element	Value to Use in the File
api_key	The client ID, from the Application ID field on overview page in the Microsoft Azure portal.
password	The client secret, generated in the Microsoft Azure portal. **Note:** It's stored as an encrypted value so if you don't have it, you'll need to generate it.
tenant_id	The tenant ID, from the Directory ID field on the overview page in the Microsoft Azure portal.
application_id	The client ID, from the Application ID field on overview page in the Microsoft Azure portal.

6. Provide a value in the **VLS ID** field (required) and in the **Description** field (optional).
7. The save the configuration, choose **Add**.

Update the Microsoft Teams VLS Configuration File with a New Client Secret

Each time the client secret expires, you need to generate a new secret in the Azure portal and then update the VLS configuration file in SuccessFactors Learning with the new value.

1. Generate a Client Secret.
2. Go to SuccessFactors Learning Administration > System Administration > Configuration > VLS Configuration and search for and edit your Microsoft Teams configuration file.
3. In the `password` element, paste the new client secret.
4. To save, choose **Apply Changes**.