CSP PARTNER APPLICATION OVERVIEW

Multi-tenant application model

October 2018

Microsoft

TABLE OF CONTENTS

CSP PARNTER APPLICATION

CREATE A MICROSOFT PARTNER CENTER SERVICE PRINCIPAL

Create a Microsoft Partner Center service principal in the CSP partner's tenant, where the multitenant application is going to be created.

NOTE: For CSP partner tenants, this service principal should already exist. If not, please create using the following steps.

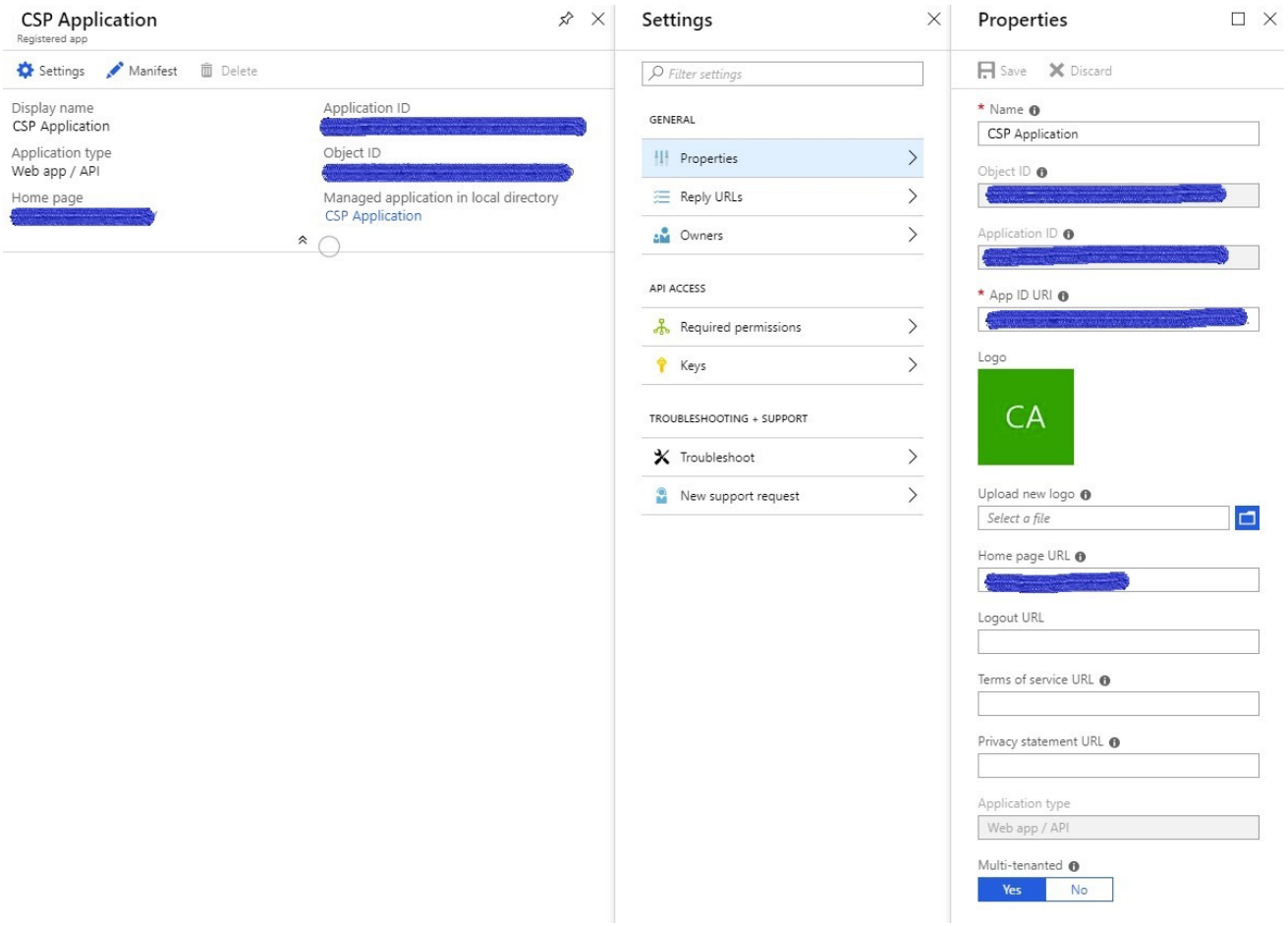
In a PowerShell window, run the following admin commands.

1. Install the AzureAD module.
    - `Install-Module "AzureAD"`
2. Run Connect-AzureAD, this will prompt for a user name and password. Please enter the tenant admin credentials.
    - `Connect-AzureAD`
3. Create a Microsoft Partner Center service principal.
    - `New-AzureADServicePrincipal -DisplayName "Microsoft Partner Center" -AppId fa3d9a0c-3fb0-42cc-9193-47c7ecd2edbd`

CREATE A MULTI-TENANT APPLICATION ON THE CSP PARNTER'S TENANT.

Please make sure that the following application properties are set for the newly created multi-tenant application.

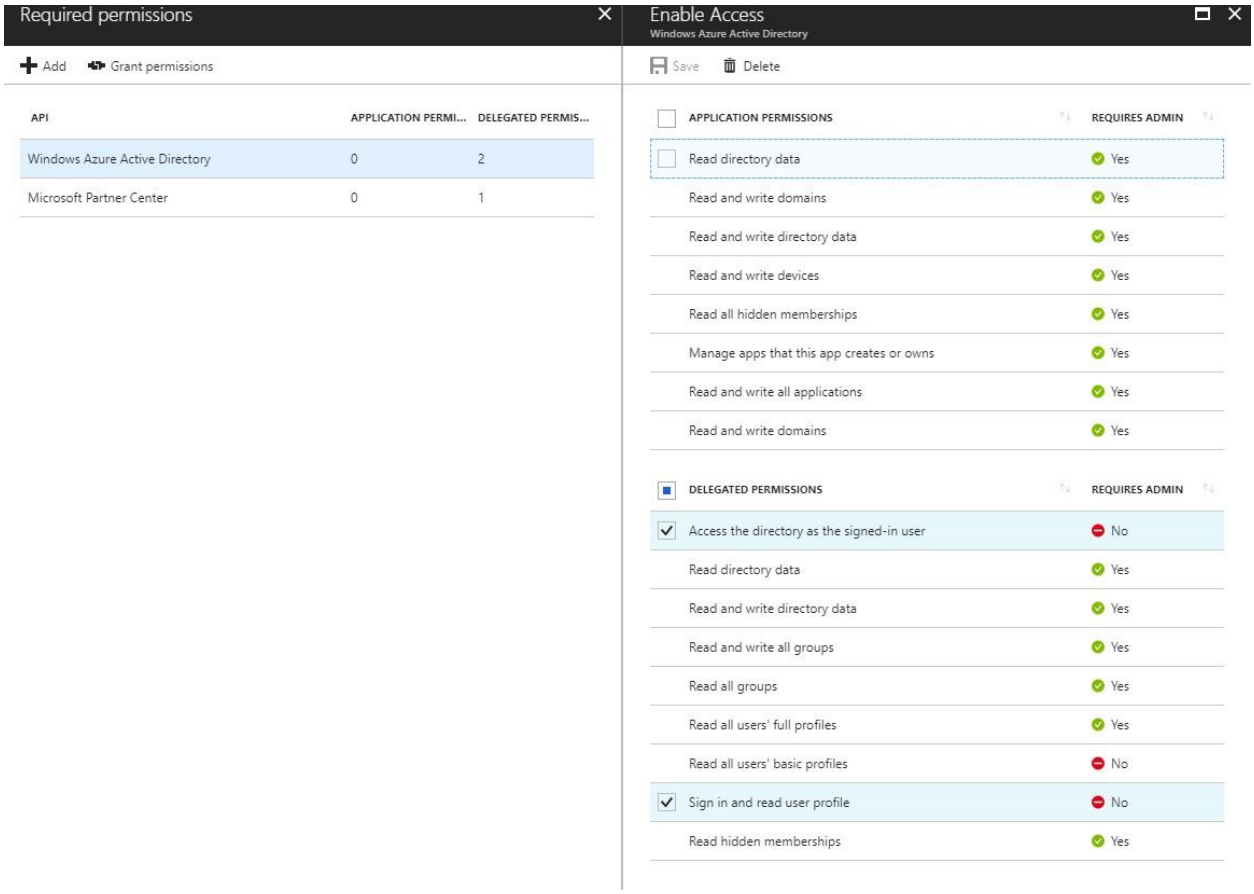
1. Have an **Application type** of "Web app / API"
2. The **Home page** URL must be your application redirect URL, which will show the consent success to the partner and collect a refresh token
3. Add a key to the web application



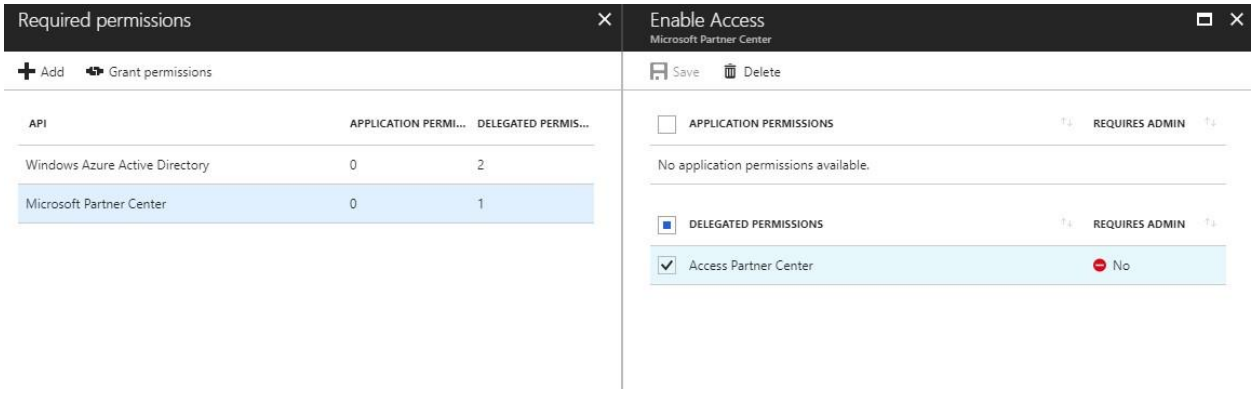
APPLICATION PERMISSIONS

Please make sure the following permissions are set for the multi-tenant application

1. Windows Azure Active Directory
    a. There should not be any direct **Application Permissions** to the multi-tenant application.
    b. **Delegated Permissions** are to be set to access Active Directory as the signed in user.



2. Microsoft Partner Center
    a. Grant **Access Partner Center** permissions under **Delegated Permissions**.



CONSENT LINK

Present the partner with the consent link and have them login with their service account to approve the CSP application to act on behalf of the service account on partner tenant.

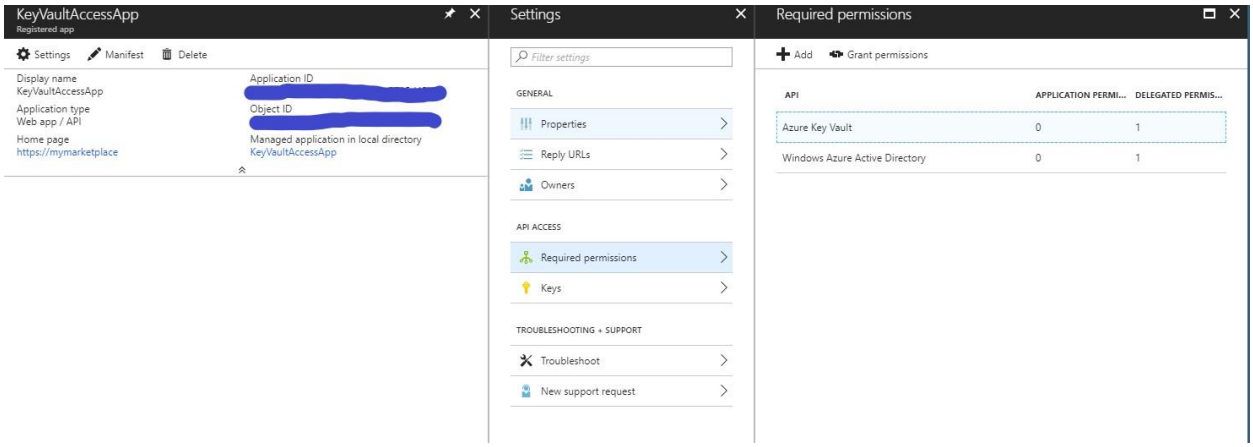
https://login.microsoftonline.com/common/oauth2/authorize?&client_id=<CSPApplicationId>&response_type=code&redirect_url=https://<CSPApplicationUrl_which_collects_refreshtoken>

KEY VAULT SETUP

CREATE A NEW WEB APPLICATION IN CSP PARNTER'S TENANT

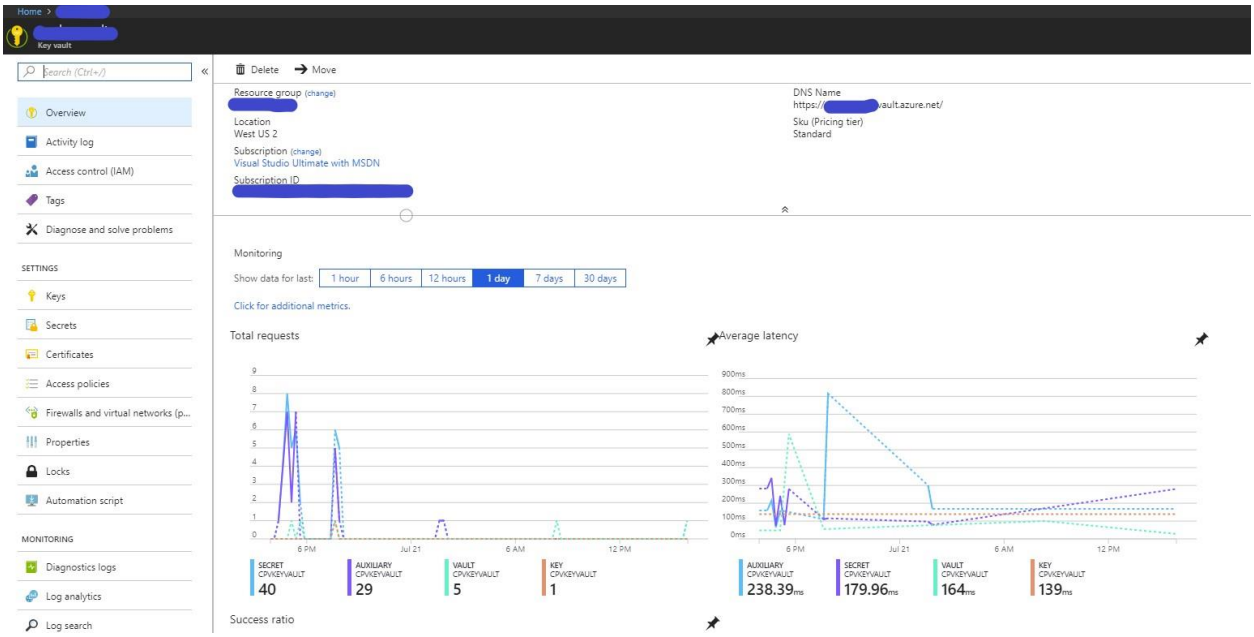
If you are using Azure Key Vault:

1. Add a key to the web application
2. Set application permissions in the **Required Permissions** tab
    a. For an **Azure Key Vault** app, under the **Delegated Permissions** section, select **Have full access to the Azure Key Vault service**
    b. For a **Windows Azure Active Directory** app, under the **Delegated Permissions** section, select **Sign in and read user profile**



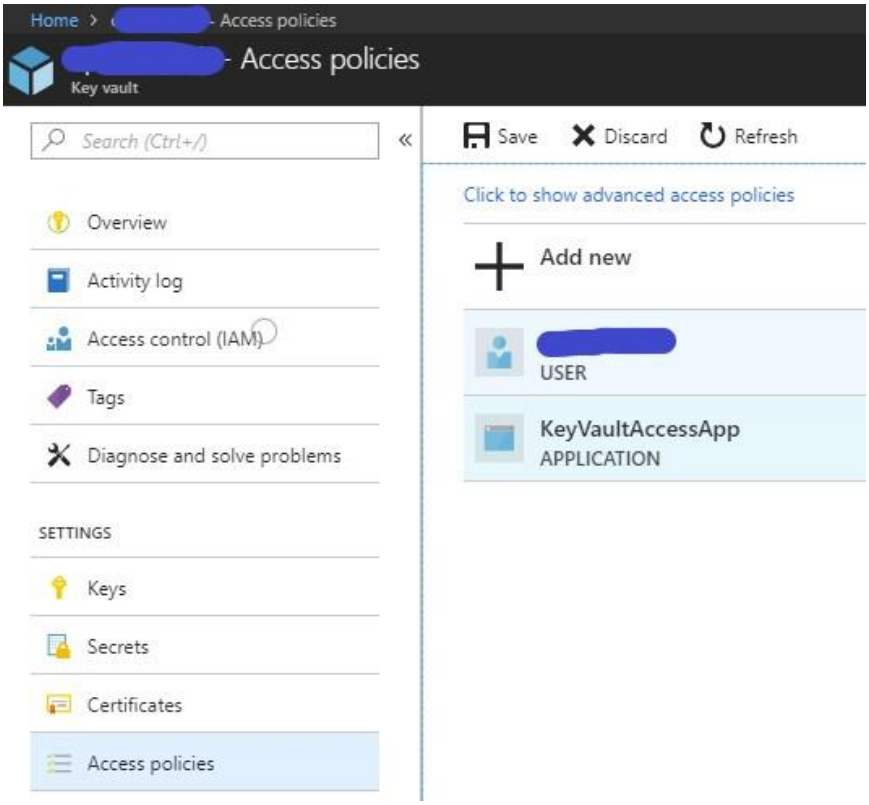
AZURE KEY VAULT SETUP

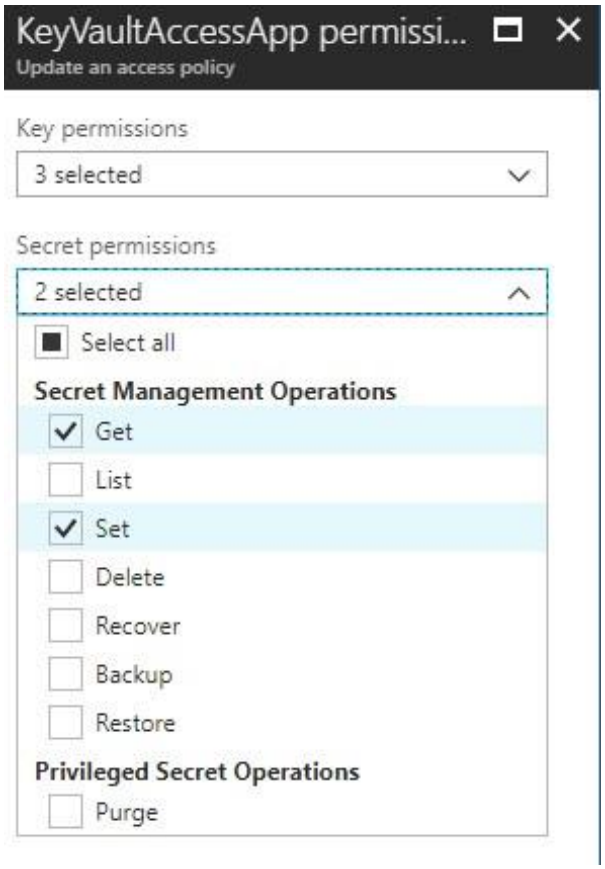
Create the Azure Key Vault with the appropriate <key-vault-name> and it will result in a DNS name like: *https://<key-vault-name>.vault.azure.net*



AZURE KEY VAULT ACCESS

In the access policies of the key vault, add the **KeyVaultAccessApp** with permissions to only manage the **Get** and **Set** aspects of a **Secret**.





PROTOTYPE CONFIGURATION

PROTOTYPE HAS TWO APPLICATIONS:

1. **Partner Consent**: Represents a web application designed to accept consent from a CSP partner and show a success message.
    a. This application will setup consent and capture the refresh token of the consented user.
    b. The consented user's refresh token is used for generating the access token for the CSP partner tenant.
2. **CSP application**: Represents a primary CSP application which calls Partner Center APIs and graph APIs to perform commerce and user actions on behalf of the partner
    a. This application retrieves the access token for a specific audience (Partner Center APIs or graph) before calling respective APIs using the refresh token that is stored securely in the key vault

CONFIGURATIONS

PARTNER CONSENT APPLICATION:

The web.config file has the following sections called out. Please update the values with corresponding application IDs and secrets. For your primary application, please use "certificate" as the web application secret instead of plain secrets because it will provide an additional layer of security.

```
<!-- AppID that represents CSP application -->
<add key="ida:CSPApplicationId" value="CSPApplicationIdValue" />
<!--
Please use certificate as your client secret and deploy the certificate to your environment.
The following application secret is for sample application only. please do not use secret directly from the
config file.
-->
<add key="ida:CSPApplicationSecret" value="CSPApplicationSecretValue" />

<!-- AppID that is given access for keyvault to store the refresh tokens -->
<add key="ida:KeyVaultClientId" value="KeyVaultClientIdValue" />

<!--
Please use certificate as your client secret and deploy the certificate to your environment.
The following application secret is for sample application only. please do not use secret directly from the
config file.
-->
<add key="ida:KeyVaultClientSecret" value="KeyVaultClientSecretValue" />

<!-- AAD instance: Global is .com, for different national clouds it changes German cloud: .de, China cloud: .cn
-->
<add key="ida:AADInstance" value="https://login.microsoftonline.com/" />
```

CSP PARNTER APPLICATION:

The app.config file has the following sections called out. Please update the values with the corresponding application IDs and secrets. For your primary application, please use “certificate” as the web application secret instead of plain secrets because it will provide and additional layer of security.

```
<!-- AppID that represents CSP application -->
<add key="ida:CSPApplicationId" value="CSPApplicationIdValue" />
<!--
Please use certificate as your client secret and deploy the certificate to your environment.
The following application secret is for sample application only. please do not use secret directly
from the config file.
-->
<add key="ida:CSPApplicationSecret" value="CSPApplicationSecretValue" />
<!-- AppID that is given access for keyvault to store the refresh tokens -->

<add key="ida:KeyVaultClientId" value="KeyVaultClientIdValue" />
<!--
Please use certificate as your client secret and deploy the certificate to your environment.
The following application secret is for sample application only. please do not use secret directly
from the config file.
-->
<add key="ida:KeyVaultClientSecret" value="KeyVaultClientSecretValue" />

<!-- AAD instance: Global is .com, for different national clouds it changes German cloud: .de,
China cloud: .cn -->
<add key="ida:AADInstance" value="https://login.microsoftonline.com/" />
```