**Deploy your SharePoint client-side web part to Azure CDN**

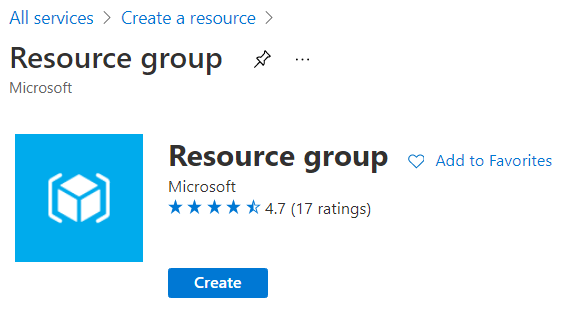
Steps

1. Create a Resource Group
2. Create Storage account
3. Enable Azure CDN for the storage account
4. Create Container
5. Storage account access key
6. Create the Webpart
7. Update Azure CDN reference **./config**
8. Deploy the web part code in Azure CDN
9. Deploy the webpart app in appcatalog
10. Add the webpart and test it in SharePoint Online

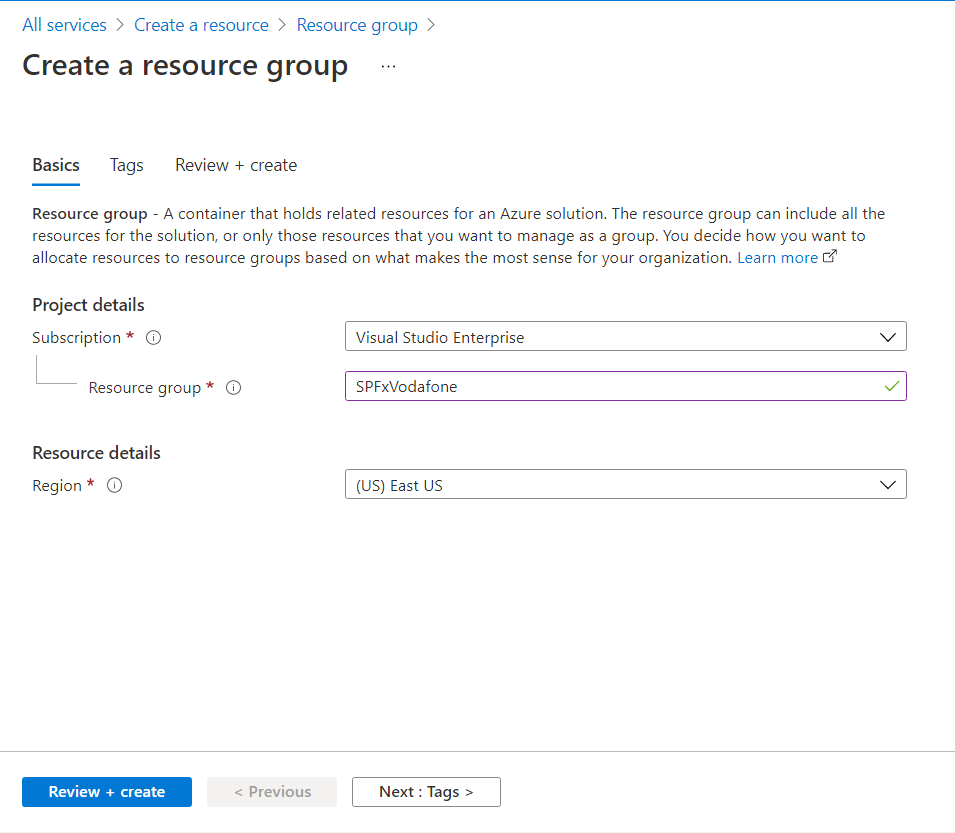
Create a Resource Group

Go to <https://portal.azure.com/#allservices> 🡪 login

* Create a resource
* Search Resource group



Click Create



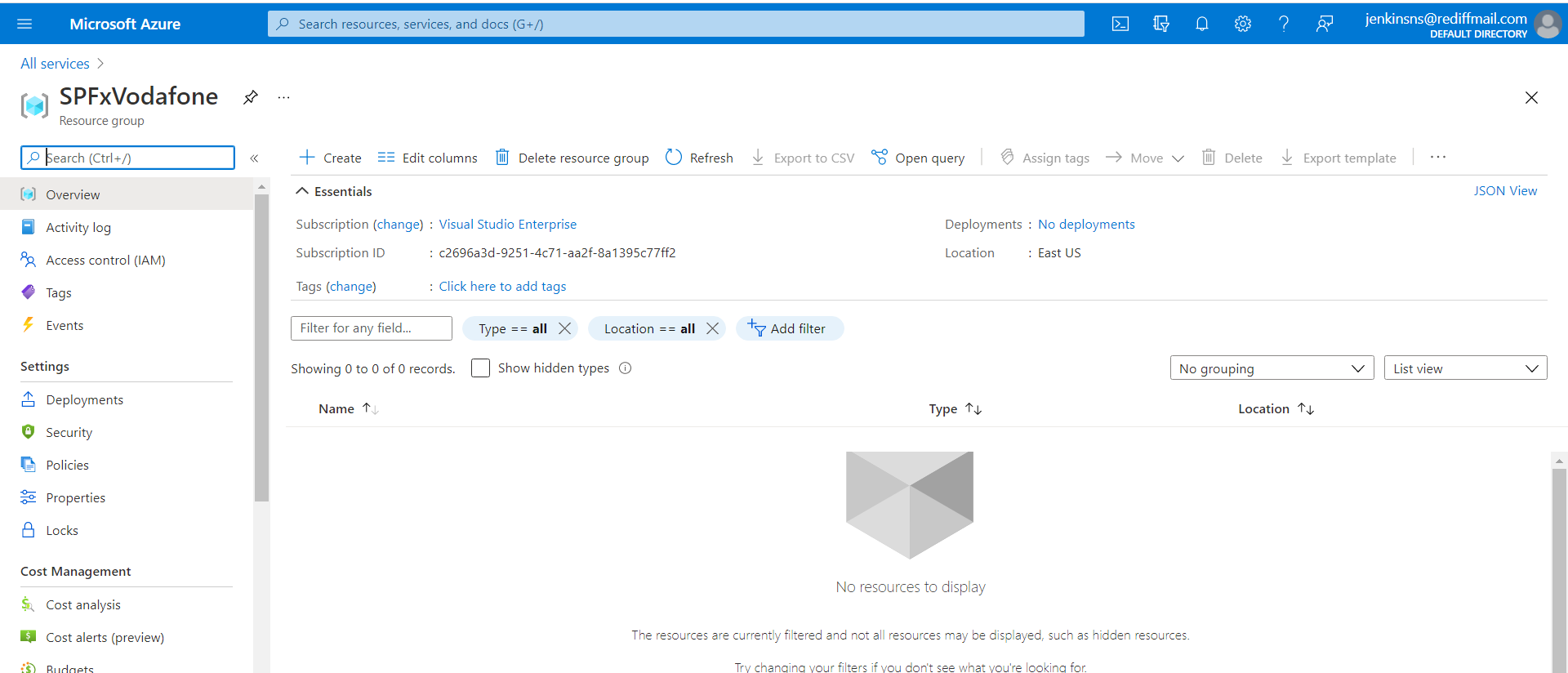
Give Resource group name

Click Review + create

Click create

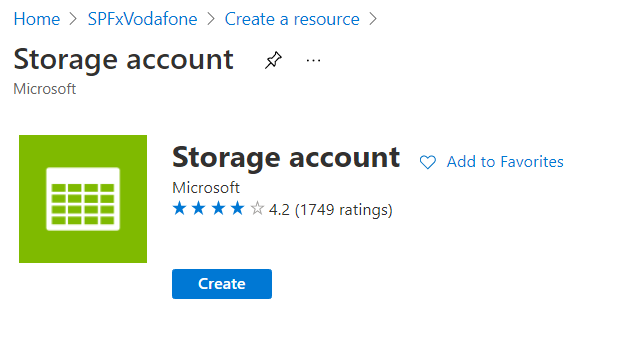
After created pin it

Then go to the Resource Group

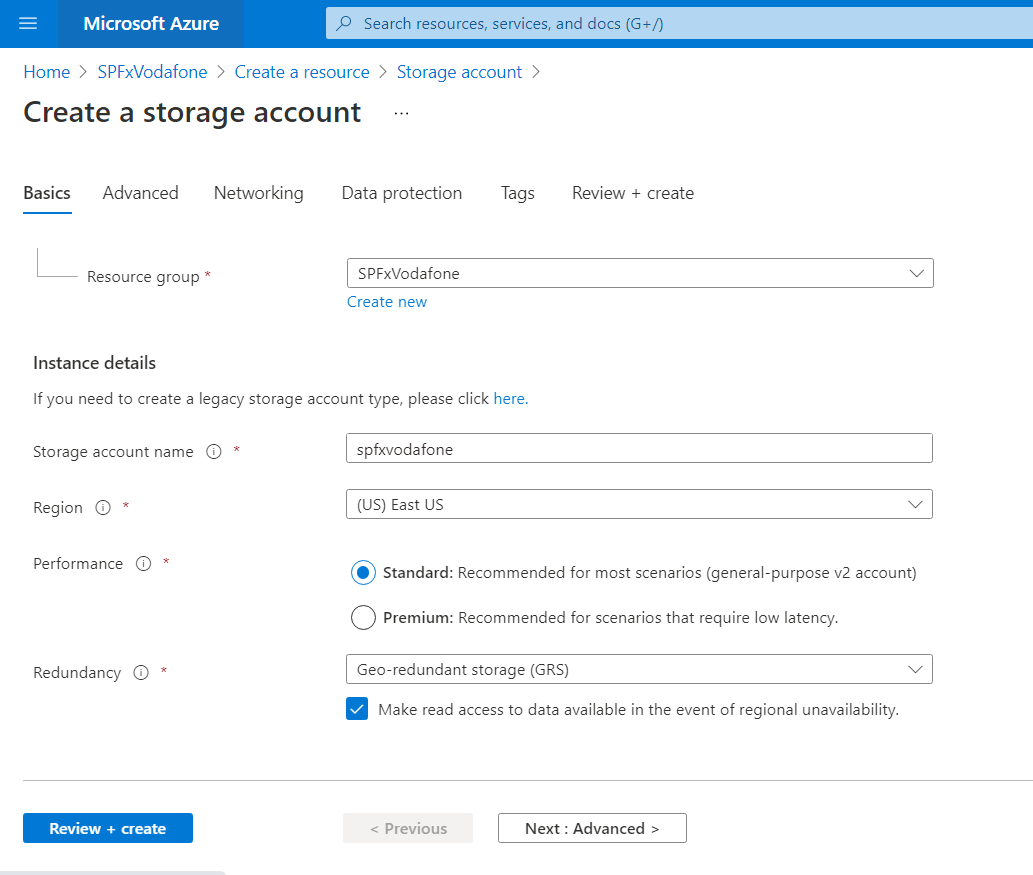


Create Storage account

1. Click in Resource Group and search Storage account



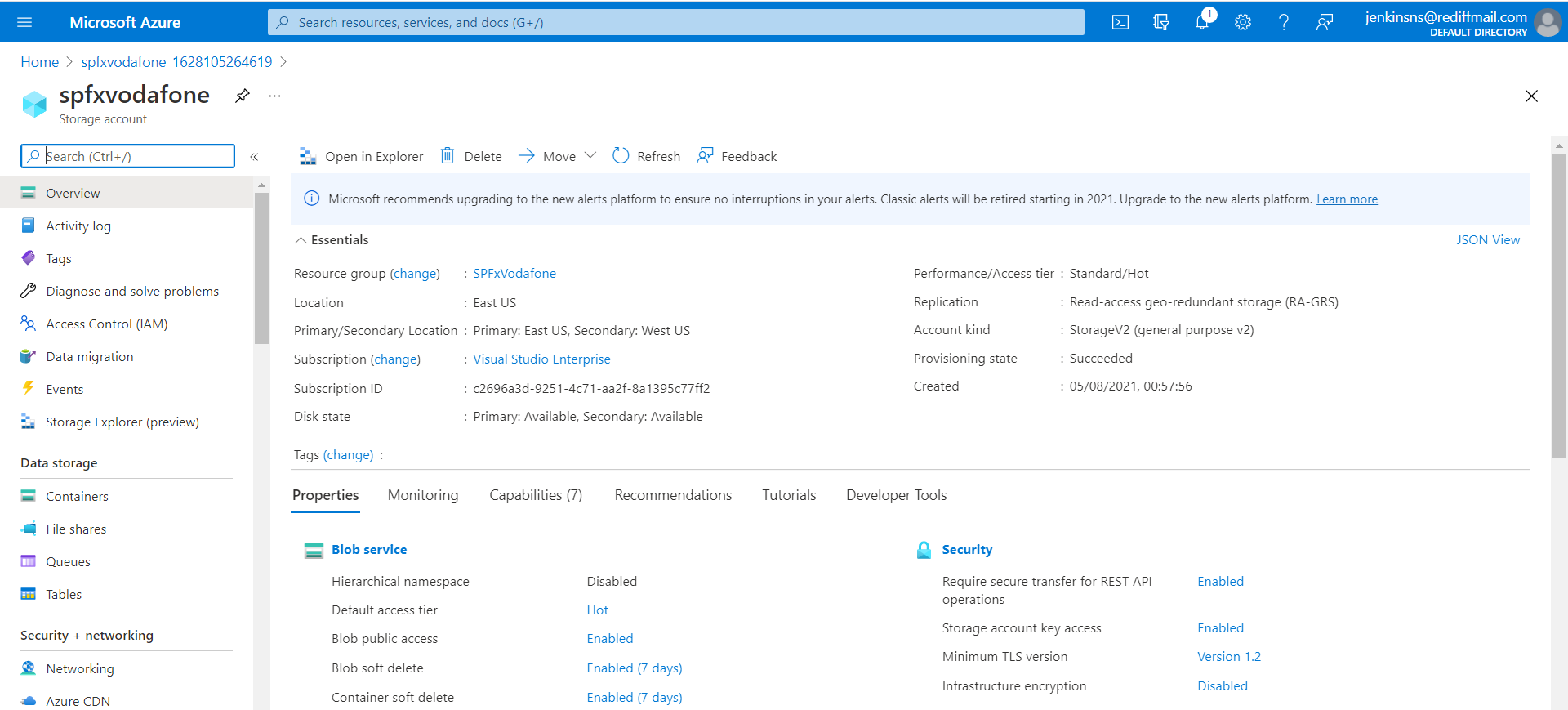
1. Click create



Click Review + create

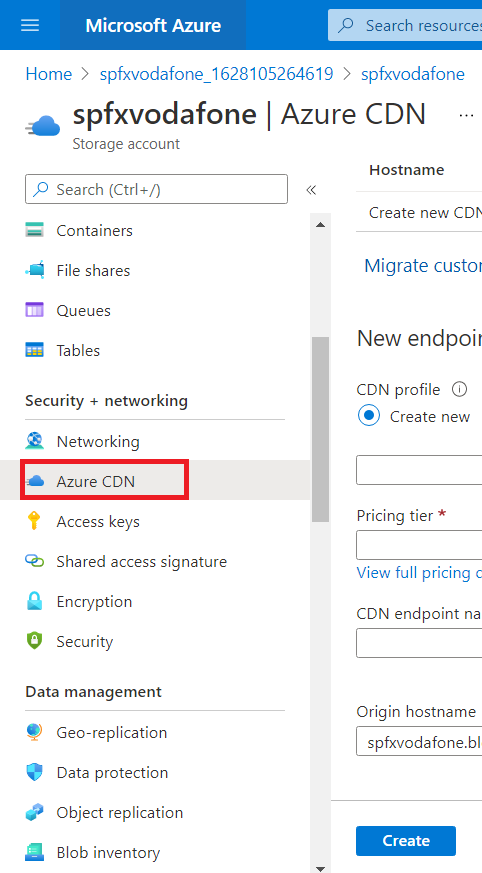
Click create

Creating the storage account might take several minutes to complete. Once creation is complete, select Go to resource to open the storage account's page for the next step.

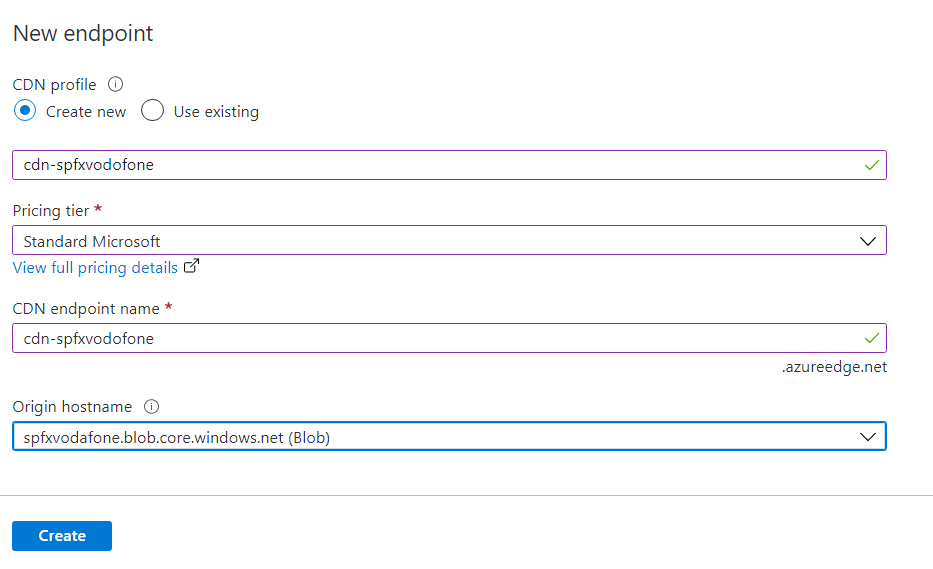


**Enable Azure CDN for the storage account**

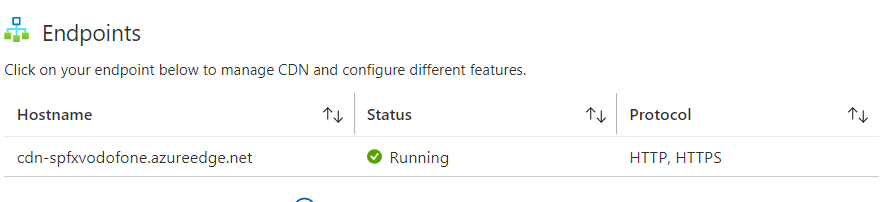
On the page for your storage account, select Blob service > Azure CDN from the left menu. The Azure CDN page appears.



1. **CDN profile** - Select Create new and enter your profile name, for example, **cdn-spfxvodofone**. A profile is a collection of endpoints.
2. **Pricing tier -** Select one of the Standard options, such as Standard Microsoft.
3. **CDN endpoint name** - Enter your endpoint hostname, such as cdn-spfxvodofone. This name must be globally unique across Azure because it's to access your cached resources at the URL <endpoint-name>.azureedge.net.
4. **Origin hostname** - By default, a new CDN endpoint uses the hostname of your storage account as the origin server.



Click Create

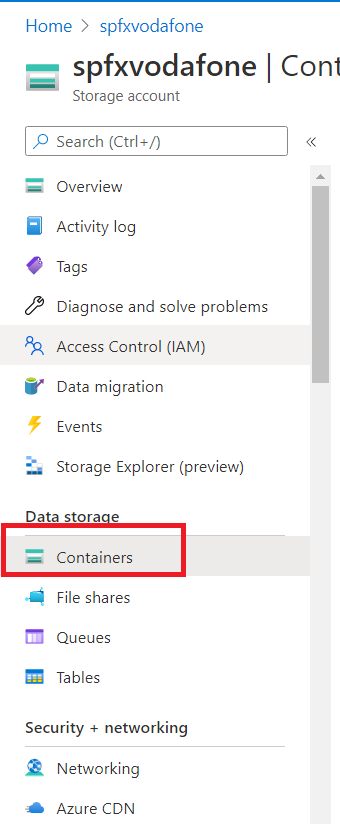


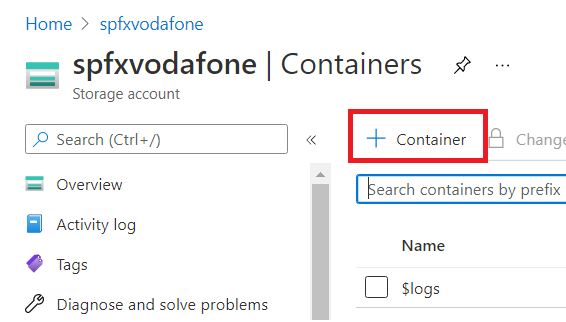
Now we have the new storage account endpoint **spfxvodofone.blob.core.windows.net.**

And host name - **cdn-spfxvodofone.azureedge.net**

Create Container

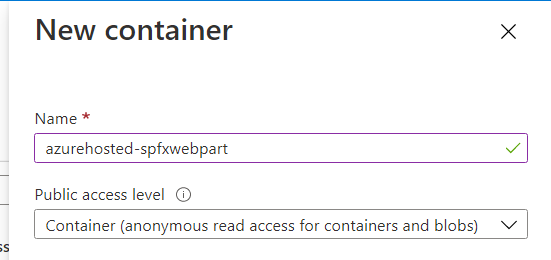
Click Containers



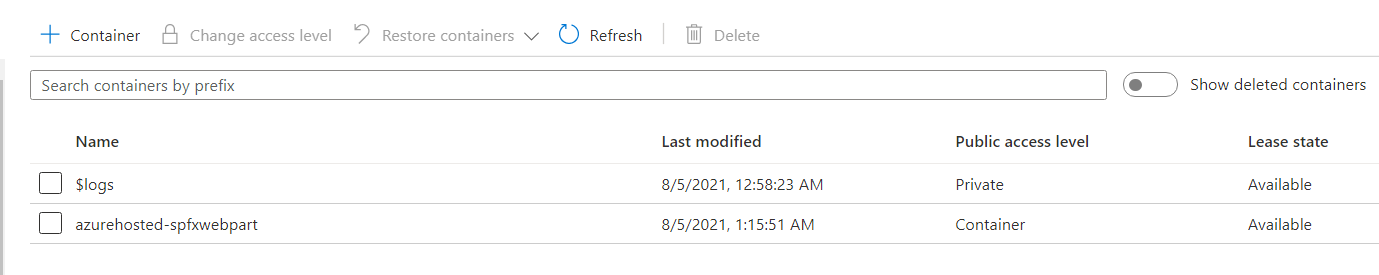
  
Select the + Container and create a new container with the following:

Name: azurehosted-spfxwebpart

Access type: Container



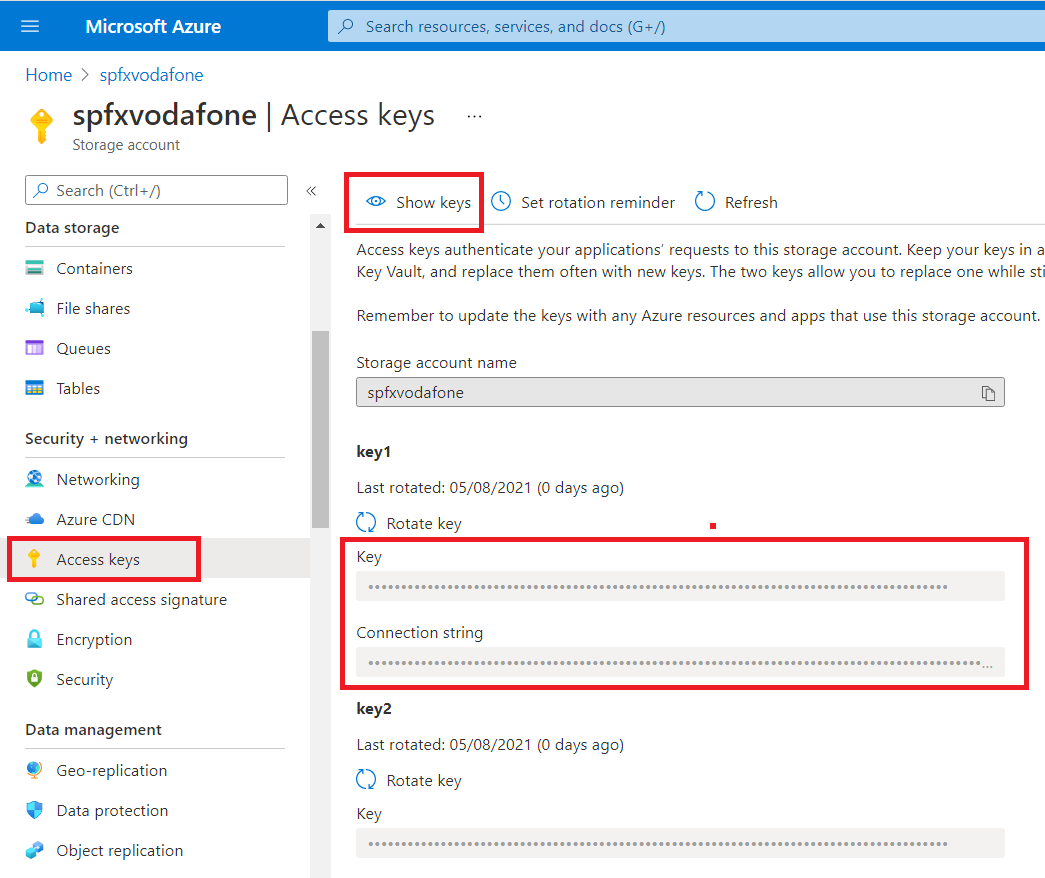
Click Create



Now azurehosted-spfxwebpart container ready

**Storage account access key**

In the storage account dashboard, select Access Key in the dashboard, and copy one of the access keys.



Click Show keys link above and copy the first key and connection string

**Get below Details**

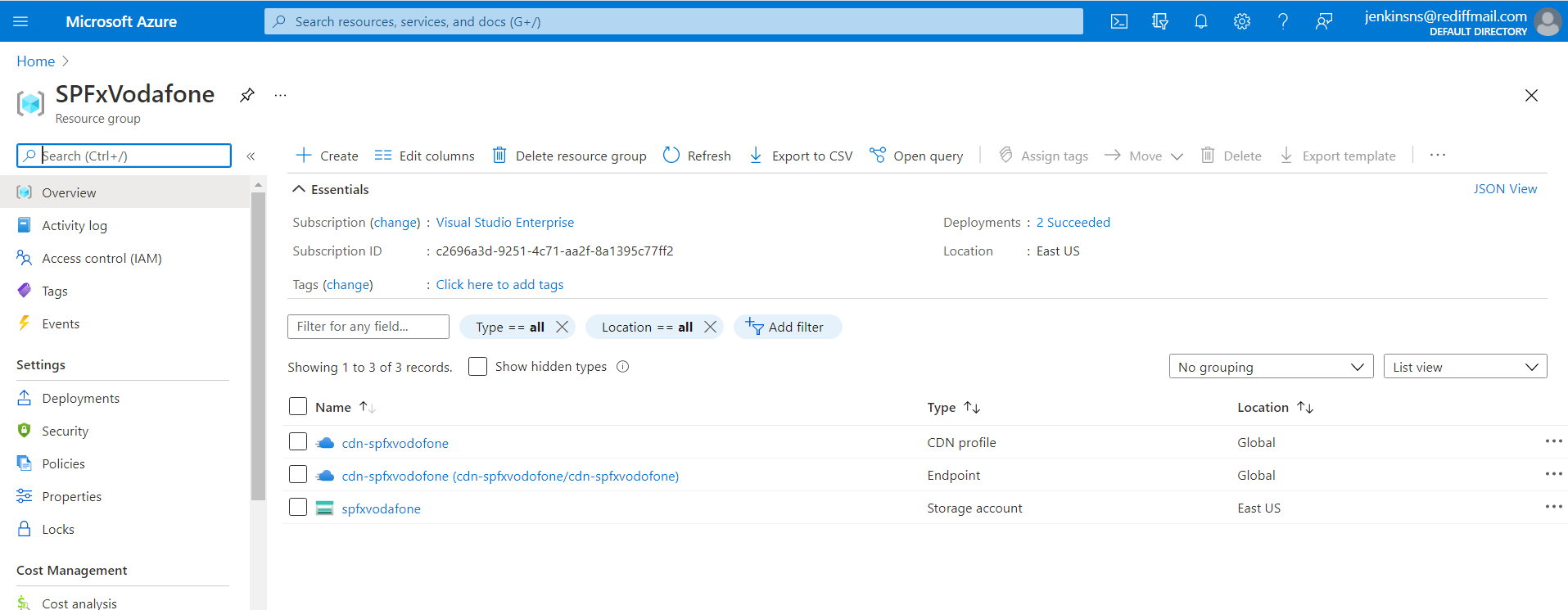
**Storage Account** : spfxvodafone

**Container** : azurehosted-spfxwebpart

**EndPoint** : cdn-spfxvodofone.azureedge.net

<https://cdn-spfxvodofone.azureedge.net/azurehosted-spfxwebpart/>

**Access key** : \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*



**Create New WebPart**

Create a new project directory in your preferred location:

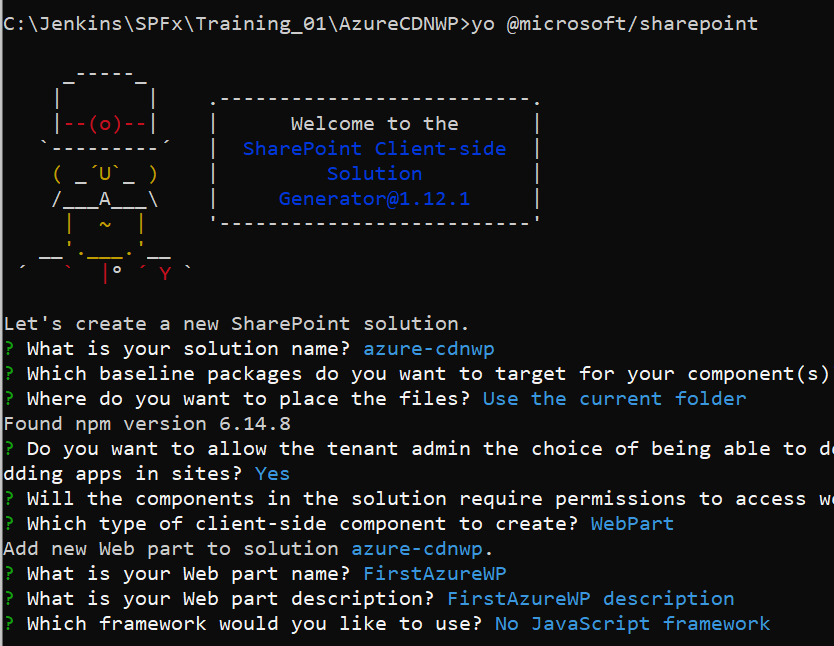
**md AzureCDWP**

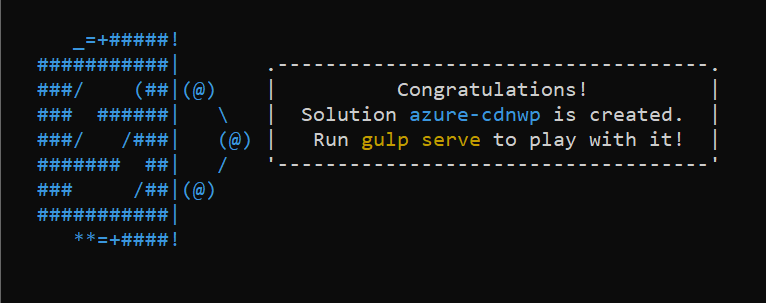
Go to the project directory:

**cd AzureCDWP**

Create a new SharePoint Framework solution by running Yeoman SharePoint Generator:

yo @microsoft/sharepoint





WebPart Created

Open in VS code

Code .

**Configure Azure Storage account details**

Open ./config/deploy-azure-storage.json file. This is the file that contains your Azure Storage account details.

{

  "$schema": "https://developer.microsoft.com/json-schemas/spfx-build/deploy-azure-storage.schema.json",

  "workingDir": "./release/assets/",

  "account": "<!-- STORAGE ACCOUNT NAME -->",

  "container": "azure-cdnwp",

  "accessKey": "<!-- ACCESS KEY -->"

}

Replace the account, container, accessKey with your storage account name, BLOB container, and storage account access key respectively and workingDir is the directory where the web part assets are located.

{

    "$schema": "https://developer.microsoft.com/json-schemas/spfx-build/deploy-azure-storage.schema.json",

    "workingDir": "./release/assets/",

    "account": "spfxvodafone",

    "container": "azurehosted-spfxwebpart",

    "accessKey": "ISGgeTrefzrrIoSMxPHYyvtARKzpLiiYoCISFhfHnSvA8DcommSPvbF+ml77hn7Sy+dsDW6K6KwIh3gPtM4rHQ=="

}

Save the file.

**Configure the web part to load from CDN**

For the web part to load from your CDN, you need to tell it your CDN path.

Switch to Visual Studio Code and open the **./config/write-manifests.json** file.

Enter your CDN base path for the **cdnBasePath** property.

{

  "$schema": "https://developer.microsoft.com/json-schemas/spfx-build/write-manifests.schema.json",

  "cdnBasePath": "<!-- PATH TO CDN -->"

}

To

{

  "$schema": "https://developer.microsoft.com/json-schemas/spfx-build/write-manifests.schema.json",

  "cdnBasePath": "https://cdn-spfxvodofone.azureedge.net/azurehosted-spfxwebpart/"

}

Save the file.

Go to **./config/package-solution.json and** update **includeClientSideAssets** true to false

"includeClientSideAssets": **false**,

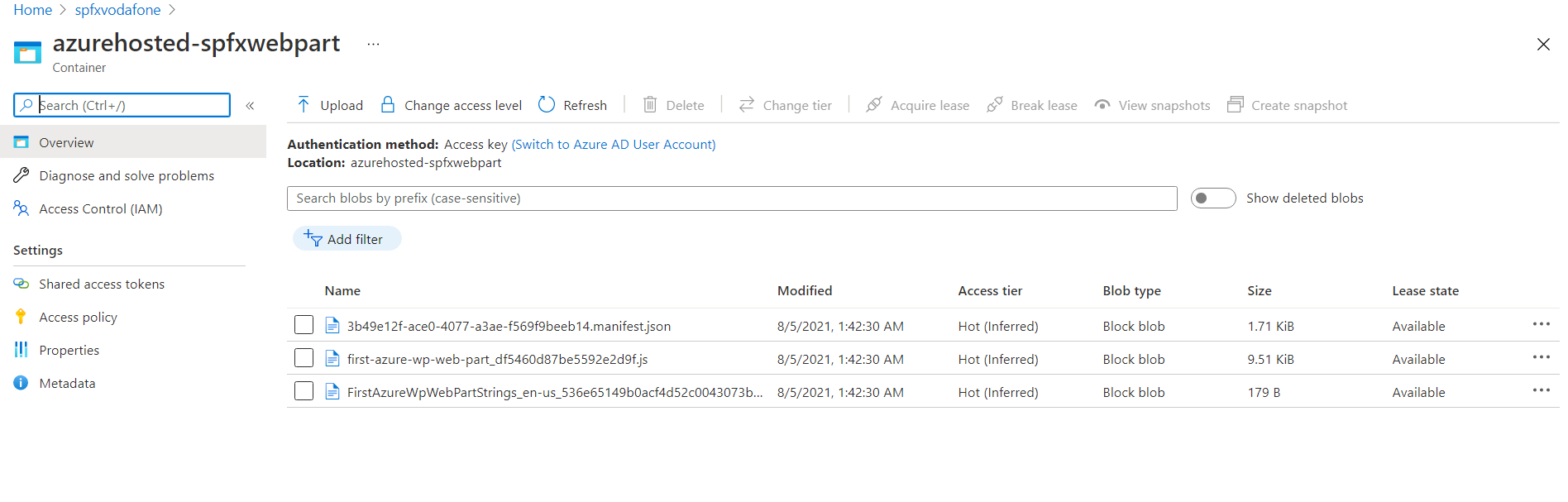
**Build and deploy**

gulp build

gulp bundle --ship

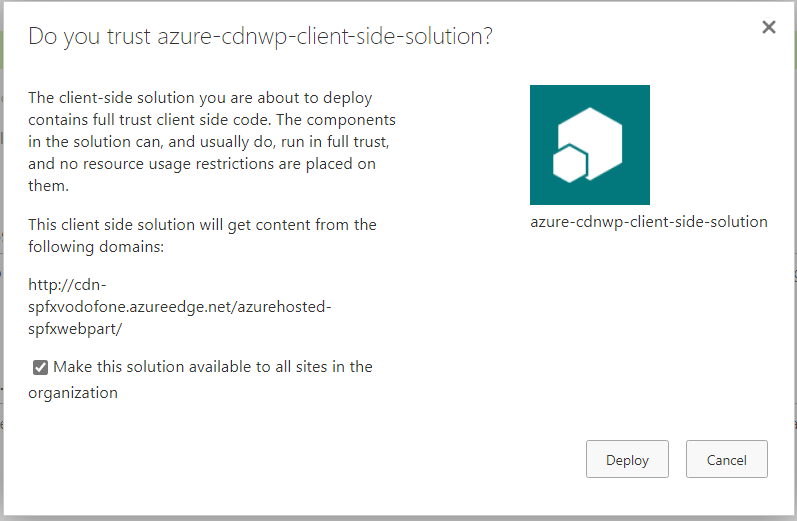
gulp deploy-azure-storage

verify it deployed in Azure CDN container



gulp package-solution --ship

Drag and drop in appcatalog



Check **checkbox** – Make this solution available to all sites in the organization

**Test the Deployed in Azure CDN Web Part**

Add the webpart in your Site collection

