Integrate Azure with Azure DevOps for Terraform deployment.

Pre-Req Setup Guide

TABLE OF CONTENTS

Azure DevOps & Azure Setup

Overview

This document describes the requirements for the service principle that will enable a Cloud Engineer to deploy Azure resources using Terraform and Azure DevOps. The service principal requires one to create management groups at the root of the tenancy as it will need owner rights for these tasks.

These high-level permissions are obviously a concern, as such the following mitigations will be used:

- The permissions will only be allocated to the service principle during the deployment change window and will be removed from the access at the end of the change window (usually 3 hours)

- The Service Principal will be deleted at the completion of the deployment by the client or the Cloud Service Provider as part of the change control.

The full permissions are outlined below:

Requirements	Justification
Owner permissions at the tenant root group	Required to create Management groups and assign to the tenant root group. Permissions will be limited to just the change window, secret will have a 1 day time limit.
Microsoft Graph Directory.ReadWrite.All Group.ReadWrite.All User.Read User.ReadWrite.All	Required to create AAD groups for management groups Required for creating break glass accounts.

This document will guide you through the steps to setup all the pre-requirements for the Azure DevOps to Azure deployment using Terraform.

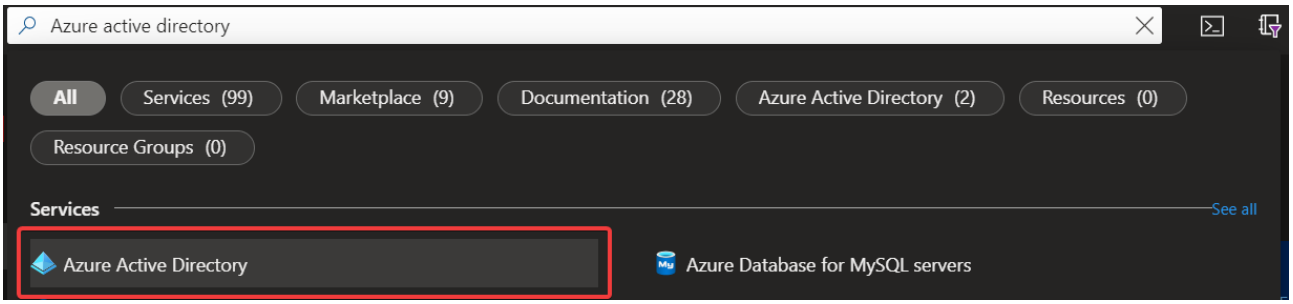
Creating the Service Principle (App Registration) Overview

As part of the Azure Terraform deployment, a Service Principle account is created so Terraform as well as Azure DevOps can authenticate and deploy within the Azure Tenant using the permissions that are assigned to the Azure Service Principle Accounts.

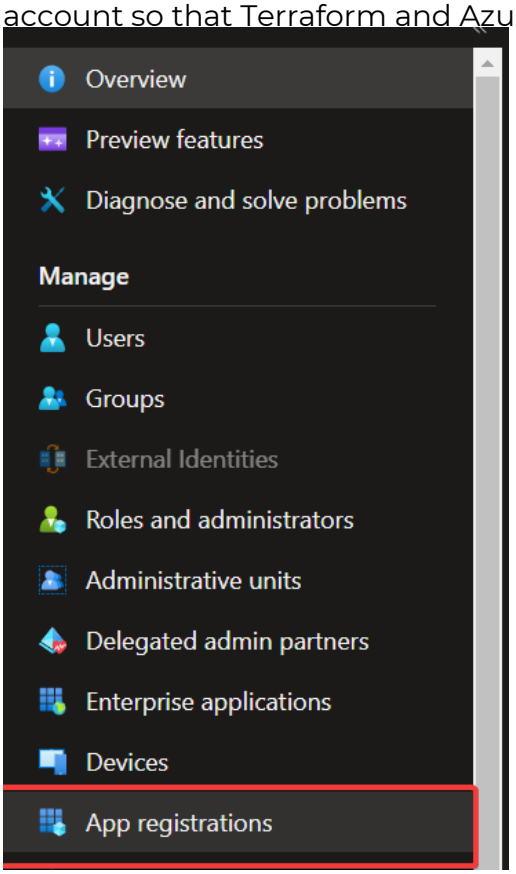
Creating the Service Principle Account.

1. Login to the Azure portal - https://portal.azure.com/ with an account that can create app registrations within Azure Active Directory, this tends to be an admin account.

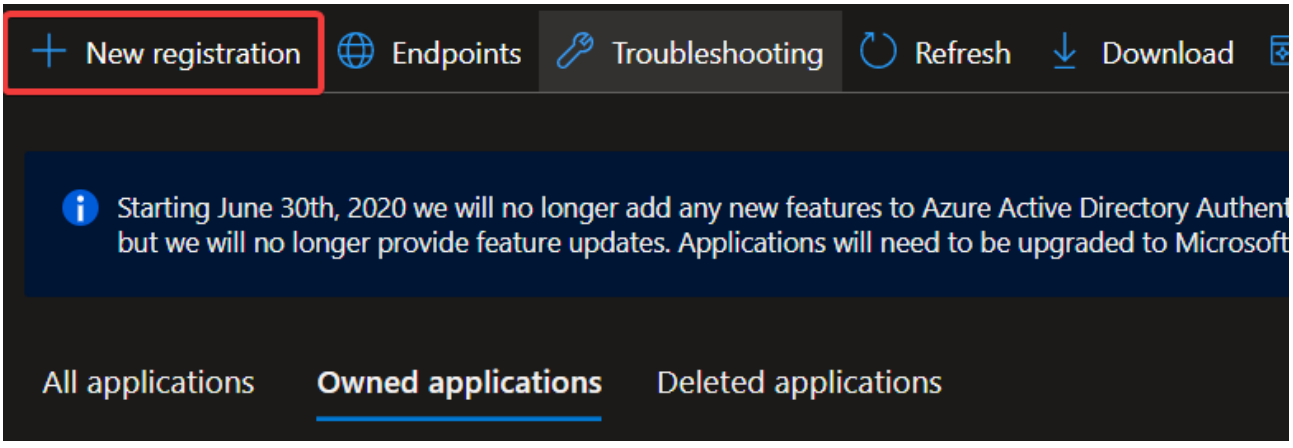
2. Search for “Azure Active Directory” within the Azure Portal



3. On the left hand menu list, click “App Registrations”, this is where we create the account so that Terraform and Azure DevOps can authenticate with Azure.



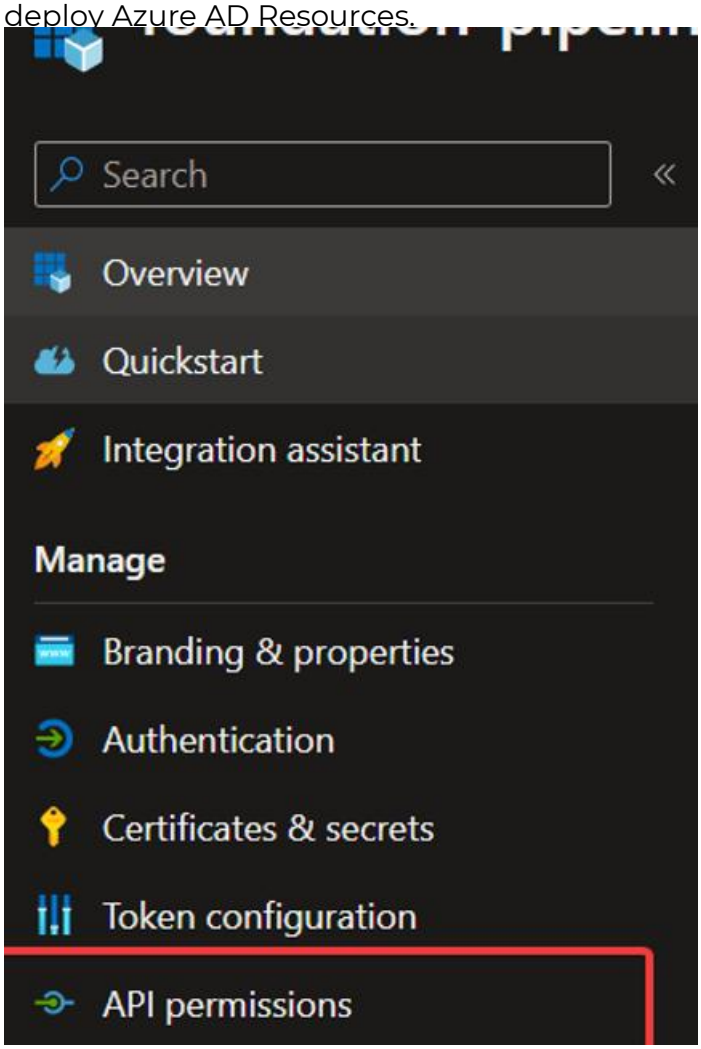
4. At the top you will see an option to create a new registration, click this and follow the below steps.



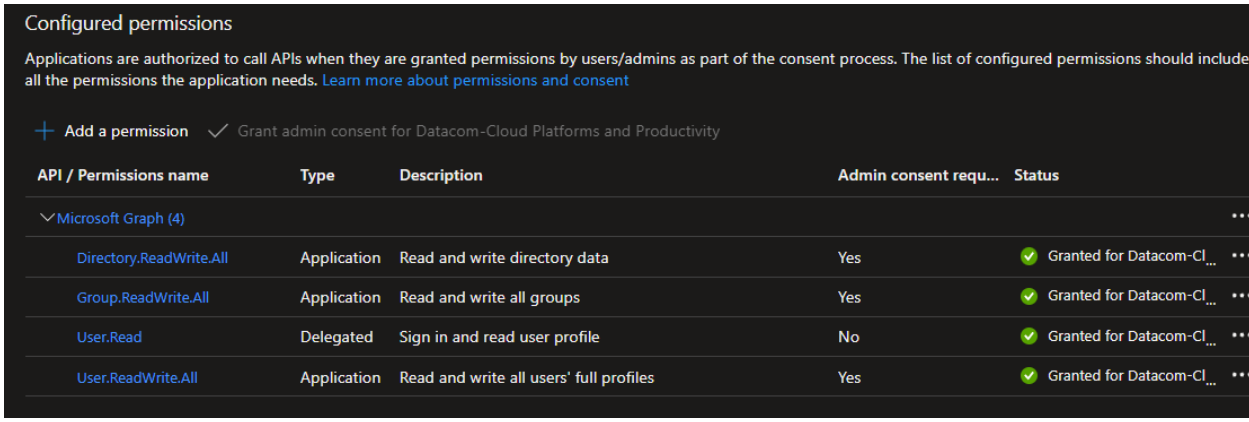
   a. Enter in a name and leave everything else as default (Including " Accounts in this organizational directory only (Default Directory only - Single tenant)" this means that it can only be used Internally), click “Register” at the bottom.

   b. Once the application has been registered it will now appear in your app-registrations under “Owned applications”

c. In the left menu pane, click “API Permissions” This is where we set the required Azure API permissions so the Service Principle account can deploy Azure AD Resources.

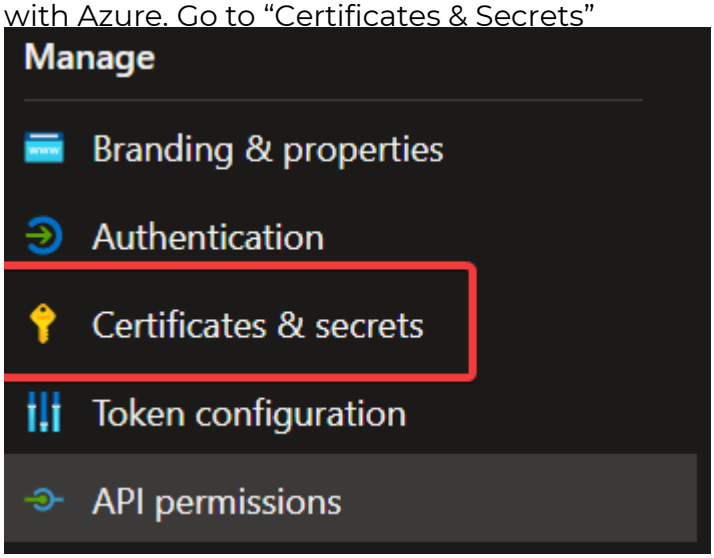


d. Click “Add a Permission” and under the Microsoft Graph select the permissions based off the image below.

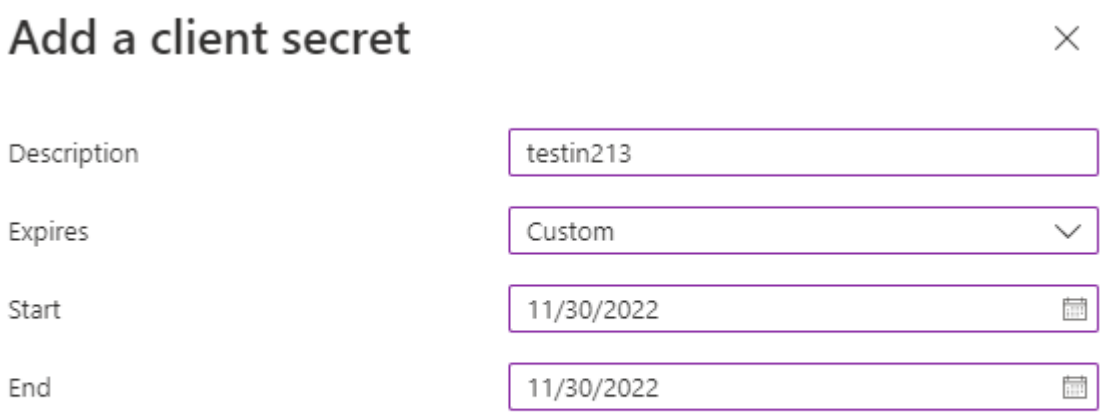


e. Once you have the permissions added, make sure you click “Grant admin consent for <Application Name>”

f. Create a Client Secret as this is needed for the pipeline to authenticate with Azure. Go to “Certificates & Secrets”



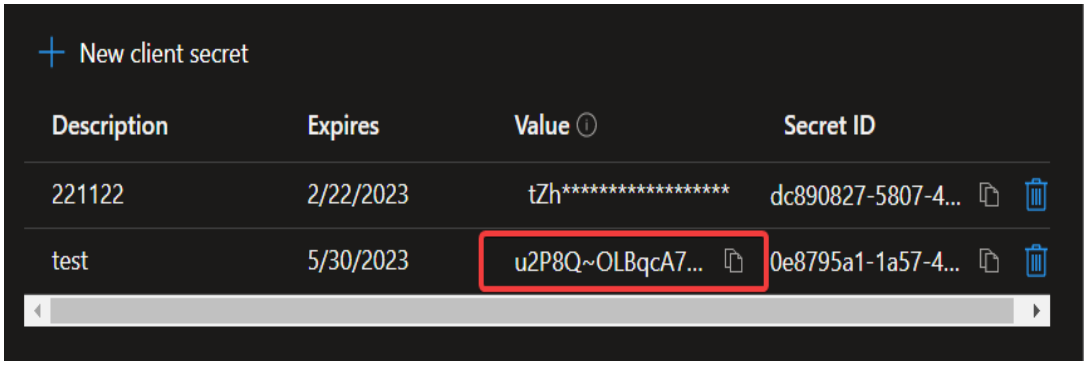
Click “New Client Secret”. enter a name, and choose a custom expiry date, and only for the day of the deployment:



This will ensure that the service principal can not be used after that day (should the service principal not be removed)

This will create a new secret that you need to keep a **note of because when you leave the page the secret will be hidden for good**. This secret needs be given to the team creating the Terraform Pipeline. Set the

secret to a short time frame



While you are in this screen, within the Overview tab within the left pane, note down the following IDs as these are also needed for the pipeline.

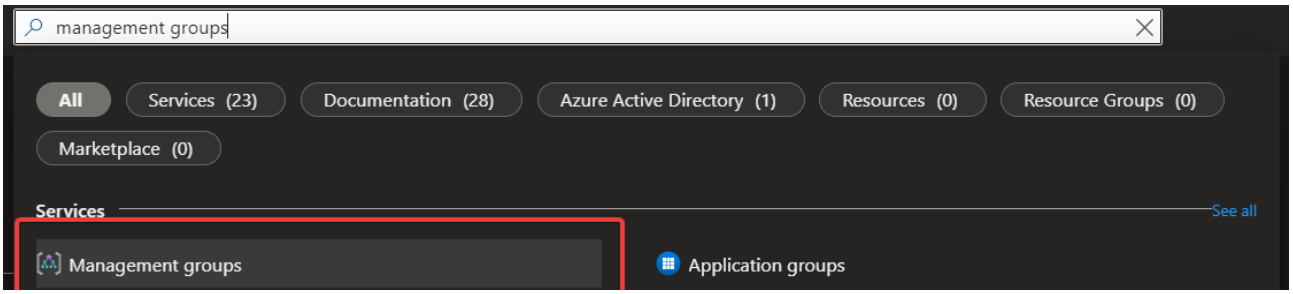
Application (client) ID and **Directory (tenant) ID**

If deploying management groups, Take Note.

Adding the Service Principle to the Root Management Group

The newly created service principle needs to be added as a Global Administrator to the Root Management group for a very limited time so the deployment can deploy management groups as well as move subscriptions and apply RBAC to them.

1. Go to “Management Groups” via the search menu.



2. Click the Root management group
3. On the left pane, click “Access Control (IAM)”
4. Click Add at the top, then click Add Role Assignment
5. Click the Owner role.
6. Click Members at the top.
7. Next to members, click Select Members
8. On the right side pane, search for the name of your service principle account.

9. Click select

10. Go to Review and assign and, click review and assign at the bottom to assign the owner role to the Service Principle.

Creating the Storage Account for Terraform State File.

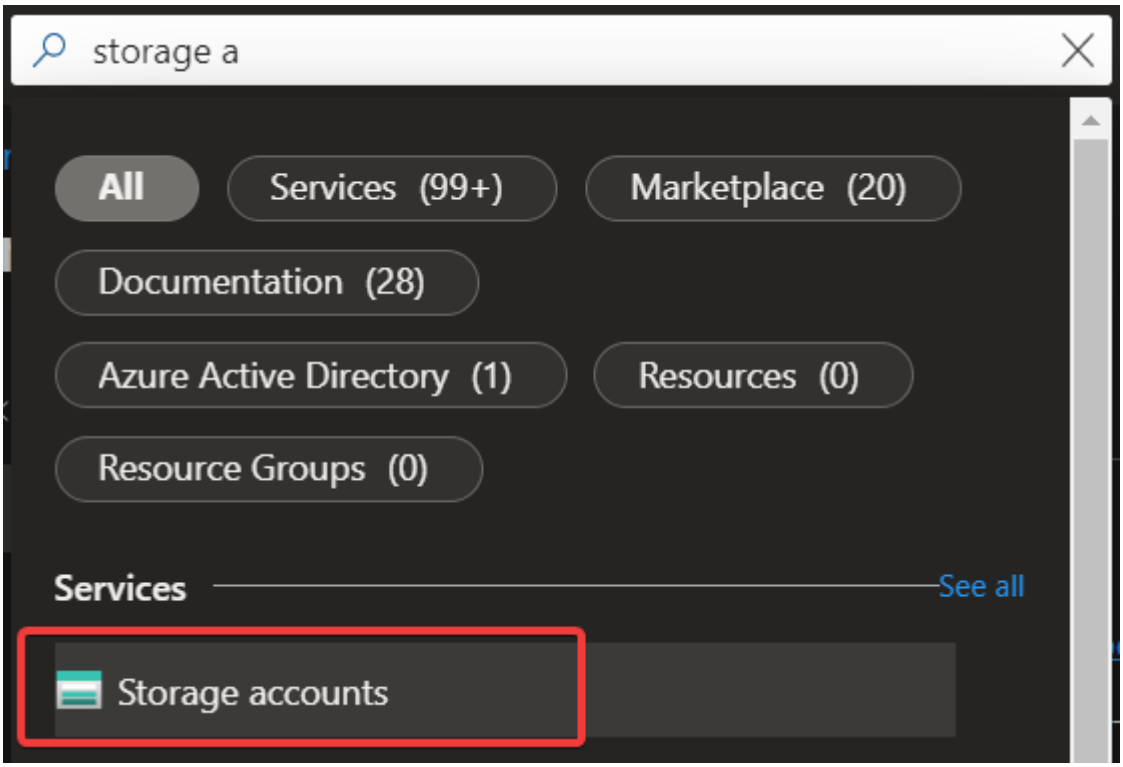
You will need to create the Storage Account manually in the Shared Services Subscription within the clients Azure tenancy. It is required to store the Terraform state file for the terraform deployment.

The following details will need to be noted as they are required for the pipeline variables.

- Storage Account Name = Has to be a unique global name conforming to customer storage account naming standard
- Storage Blob Container Name = tfstate
- Generate a SAS token at the blob level so the pipeline can authenticate with the Storage Account and add a blob file for the state.

Creating a Storage Account

1. Go to the search bar the top of the Azure portal and search for Storage Accounts, click the first option that appears.



2. Click create.
3. Make sure the subscription is the Shared Services or equivalent.
4. Create a new Resource Group and name it Terraform State while also following your naming convention
5. Create a storage account name, this needs to be globally unique.
6. Select a region and then leave all the other options as default.
7. Click review and finish the creation of the Storage Account
8. Note the storage account name down and pass this to the team deploying the pipeline.

Creating a Container

A container sits within a storage account and will contain the Terraform state file.

1. Click the new storage account and in the left pane, click Containers.

2. Click the “ + Container” at the top.
3. Name this “tfstate” and click create.

Generating the SAS Token.

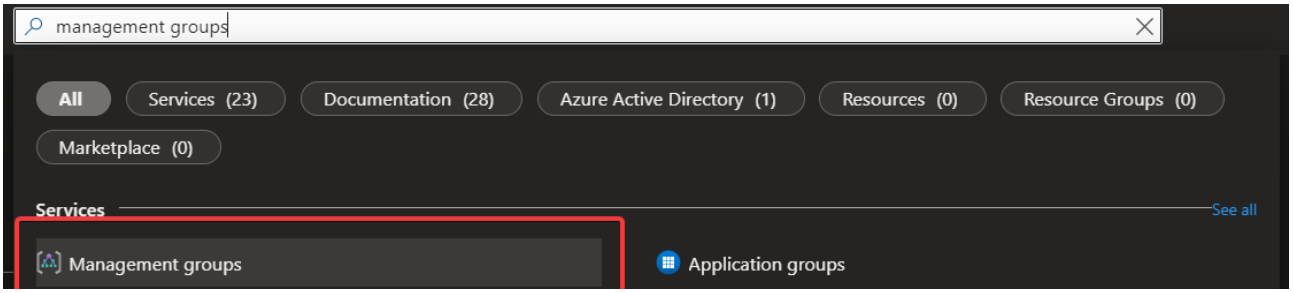
This is needed so the Terraform pipeline can authenticate with the Storage Account container.

1. Under Security + Networking, click Shared Access Signature.
2. Click Container and Object under the Allowed Resource Types heading.
3. Click Generate SAS and connection string
4. Copy the SAS Token string and pass this to the team running the Pipeline.

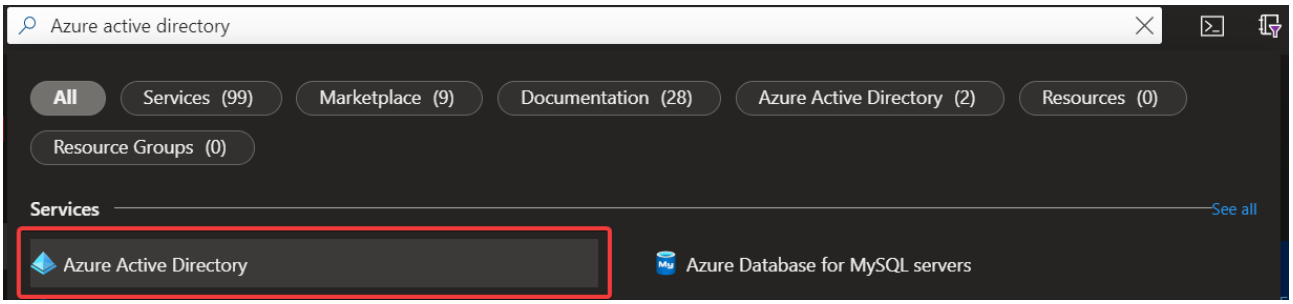
Removing the App Registration at the end of the change window

After the successful deployment, the owner permissions should be removed from the service principal, and the service principal should be deleted.

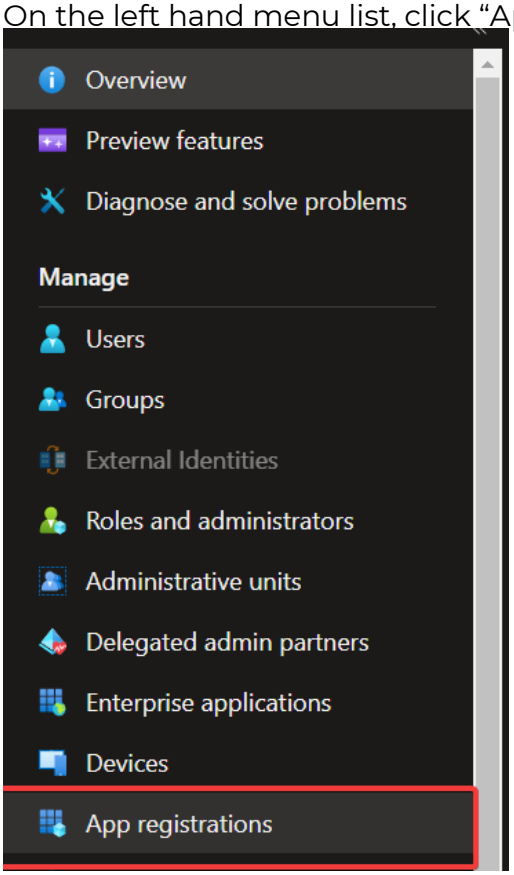
1. Go to “Management Groups” via the search menu.



2. Click the Root management group
3. Select the check mark next to the service principal, and select remove.
4. Search for “Azure Active Directory” within the Azure Portal



5. On the left hand menu list, click “App Registrations”



6. Select the previously created app registration and click delete, and then confirm that you would like to delete the service principal.