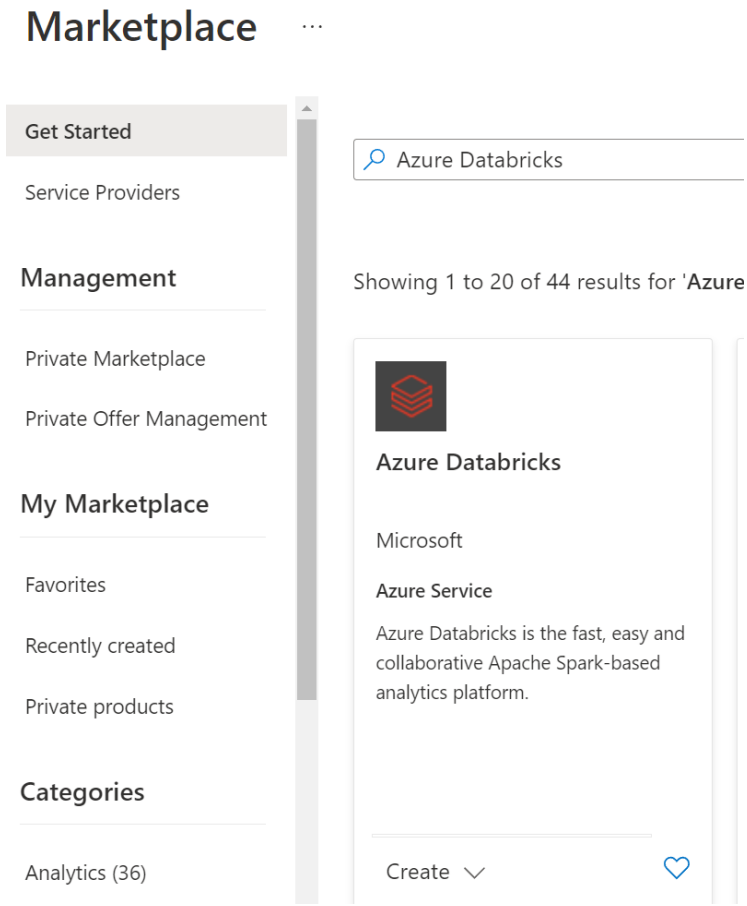
Azure Databricks Setup

Contents

This walkthrough provides the steps required to set up a Databricks workspace in Azure, create a compute capability in that workspace, and then connect the workspace to the storage account created in assignment 1.

Create Azure Databricks Resource



Home > MarchMadness > Marketplace > Azure Databricks >

Create an Azure Databricks workspace ...

Basics | Networking | Advanced | Tags | Review + create

Project Details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * : david.froslie Student Subscription

Resource group * : MarchMadness

Create new

Instance Details

Workspace name * : MarchMadnessADB

Region * : East US

Pricing Tier * : Standard (Apache Spark, Secure with Azure AD)

Create an Azure Databricks workspace ...

Basics | Networking | Advanced | Tags | Review + create

Deploy Azure Databricks workspace with Secure Cluster Connectivity (No Public IP) : ○ Yes ◉ No

Deploy Azure Databricks workspace in your own Virtual Network (VNet) : ○ Yes ◉ No

Create an Azure Databricks workspace ...

Validation Succeeded

Basics Networking Advanced Tags **Review + create**

Summary

Basics

Workspace name	MarchMadnessADB
Subscription	david.froslie Student Subscription
Resource group	MarchMadness
Region	East US
Pricing Tier	standard

Networking

Deploy Azure Databricks workspace with Secure Cluster Connectivity (No Public IP)	No
Deploy Azure Databricks workspace in your own Virtual Network (VNet)	No

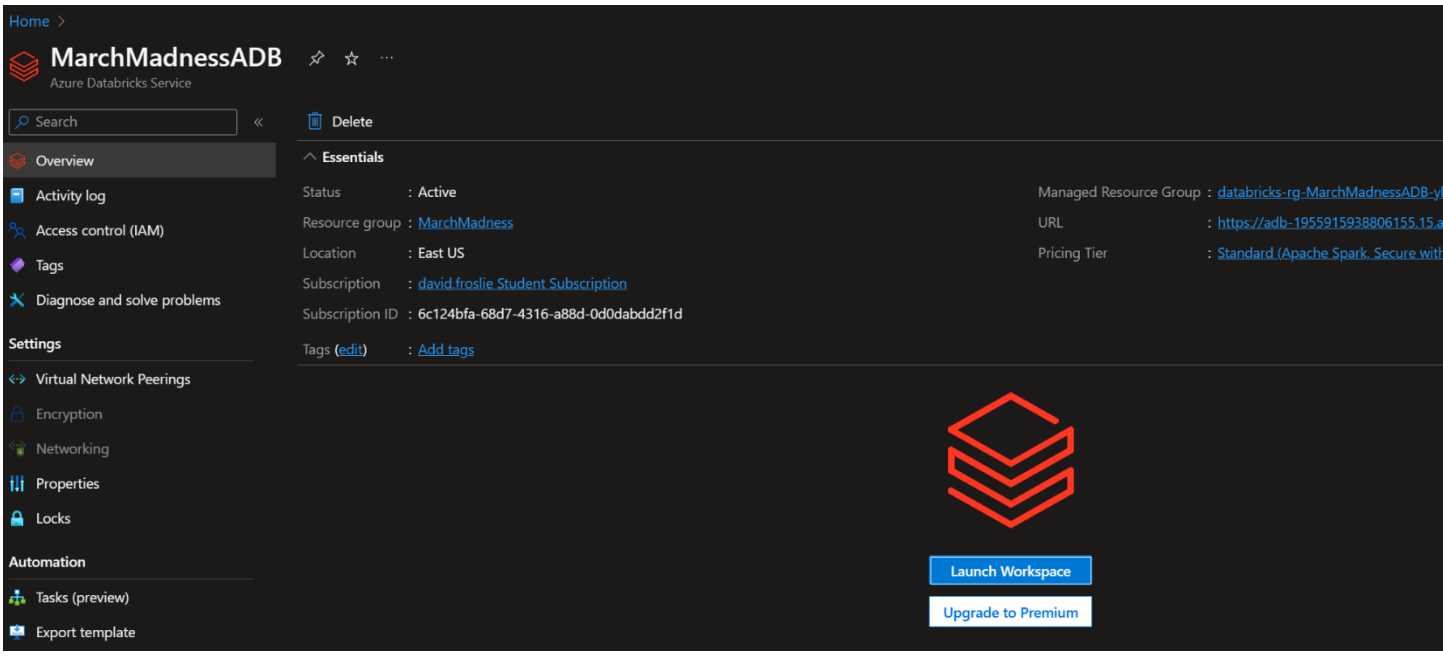
Advanced

Enable Infrastructure Encryption	No

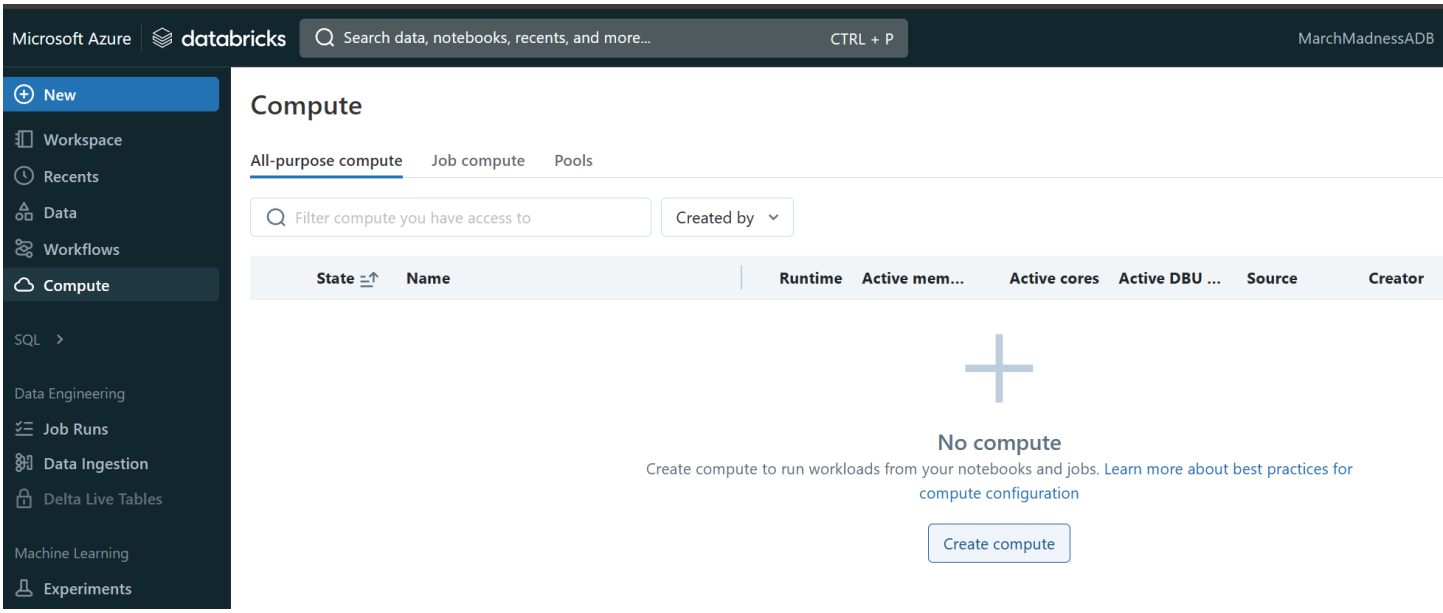
Create < Previous Download a template for automation

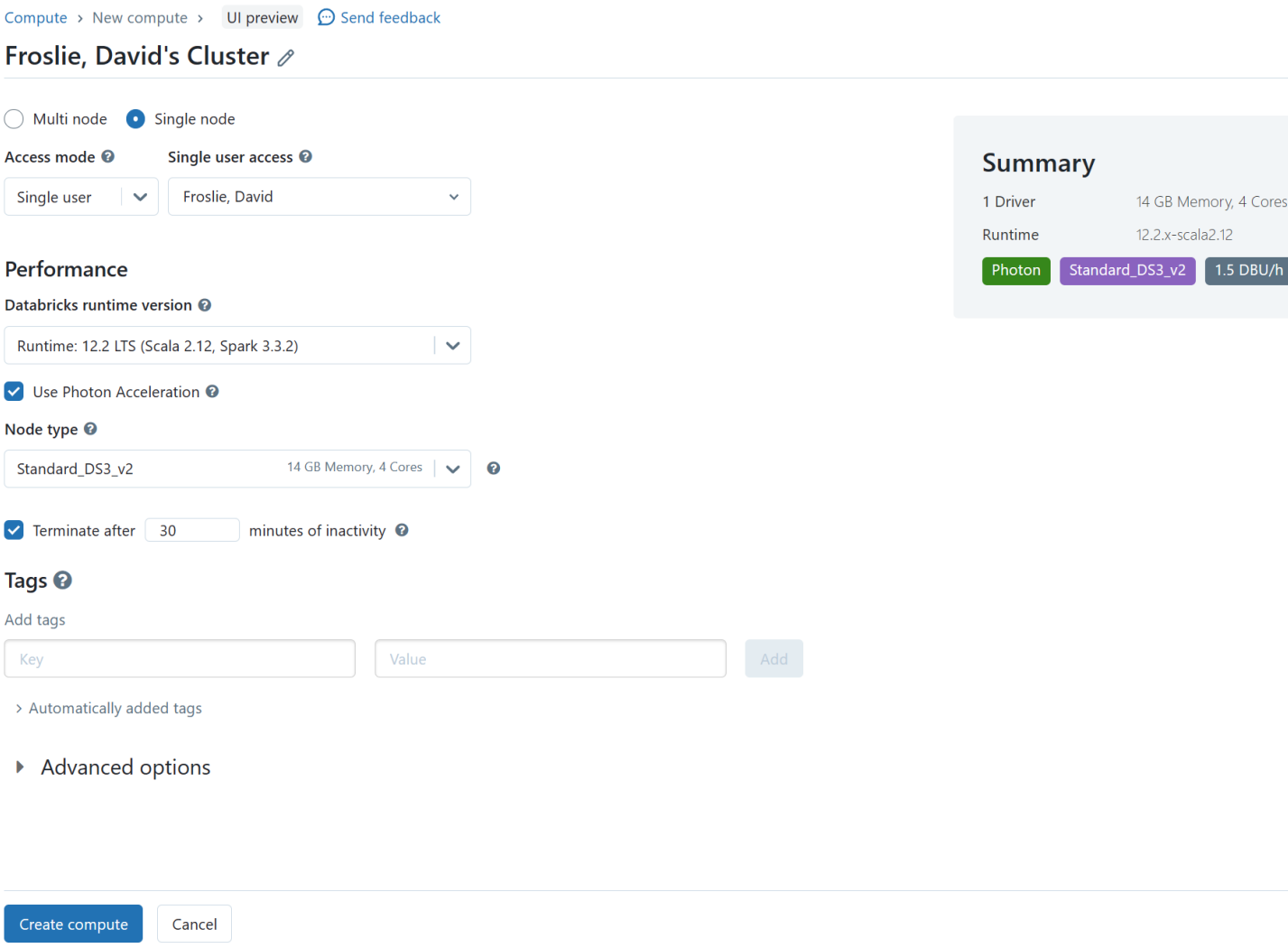
Set up minimalistic compute (0.75 dbu / hour, or $0.30 / hour)

Navigate to the Azure resource that you just created. Databricks operates in a workspace set that you go to be clicking on Launch Workspace.



After logging in, navigate to the Compute page and click Create Compute. Create a Single node "cluster".





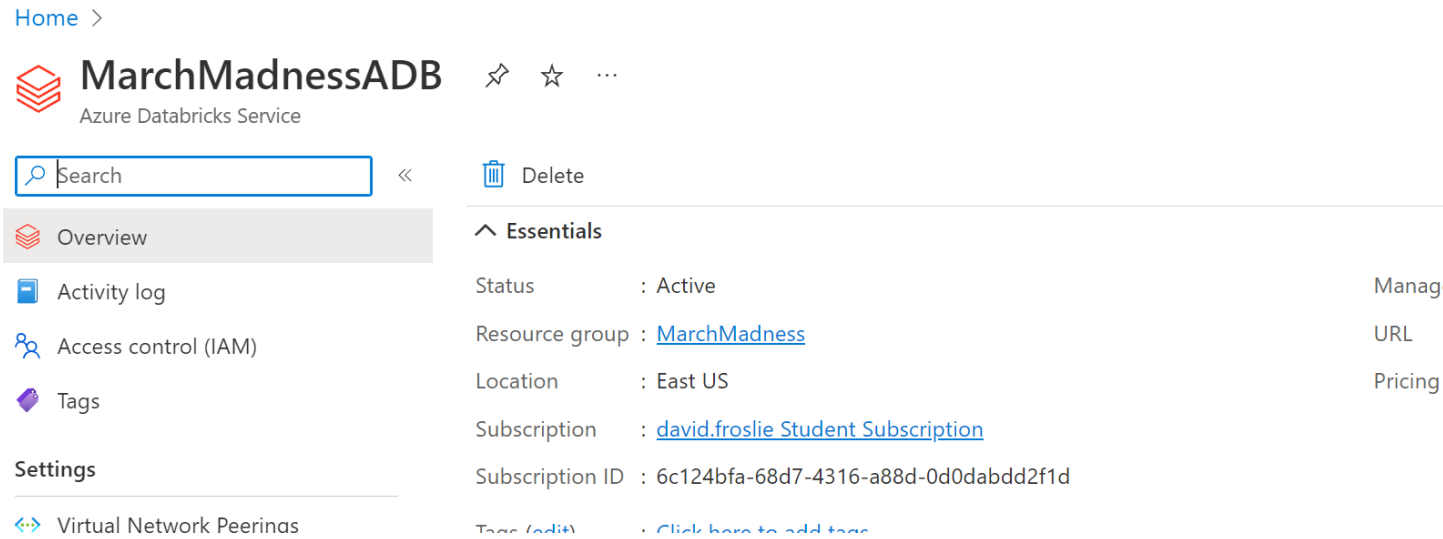
Set up access to storage account

Use the key vault and storage account created in assignment 1 to connect the Databricks instance to the storage account. This is done in multiple steps:

1. Create a "secret scope" in Databricks that points to the key vault.
2. Create a secret in the key vault that has the access key to the storage account.
3. At run-time, specify the secret scope and the secret in the Spark configuration. This will enable access to the storage account.

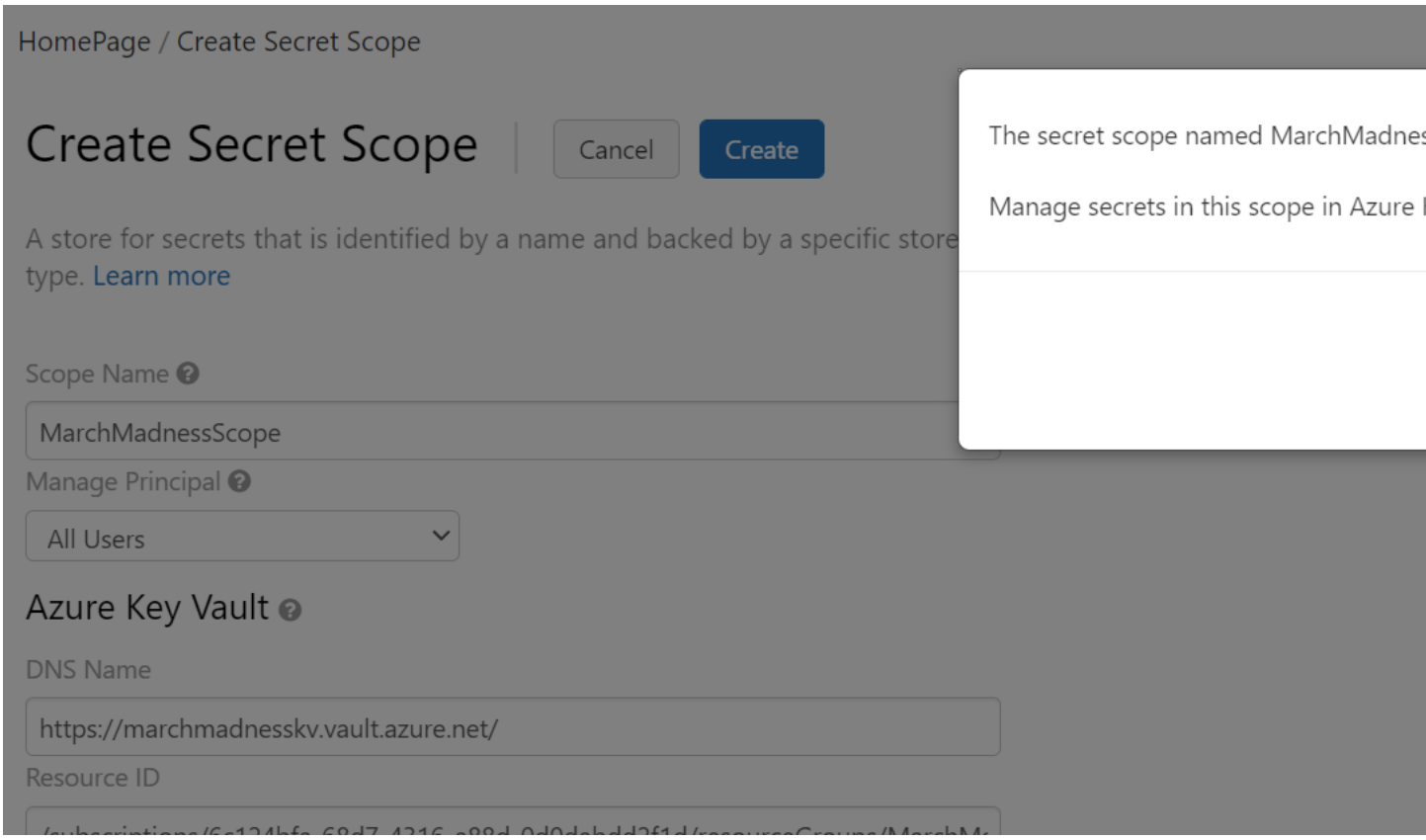
Create secret scope in ADB

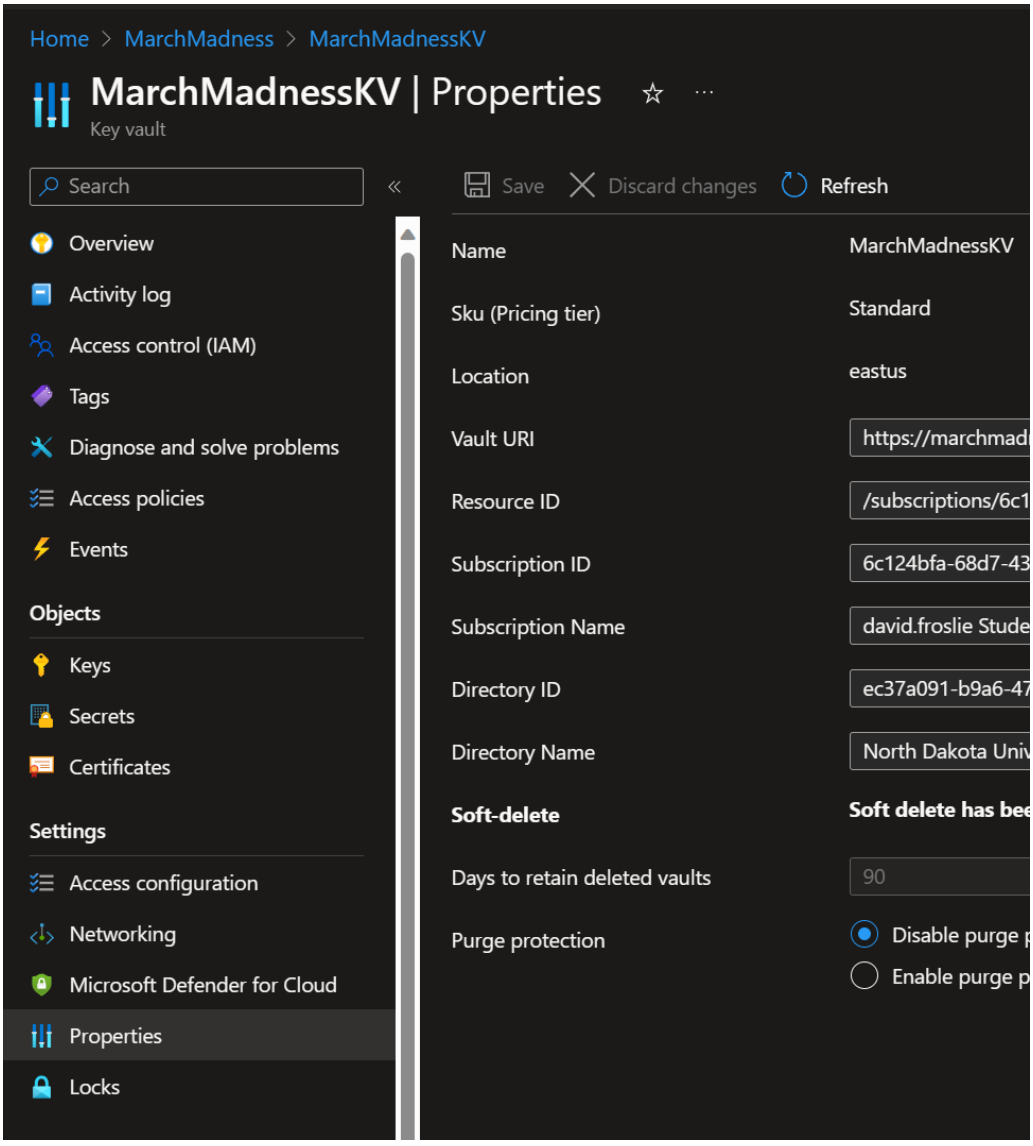
Follow the steps in to create the secret scope in Databricks: https://learn.microsoft.com/en-us/azure/databricks/security/secrets/secret-scopes#--create-an-azure-key-vault-backed-secret-scope



Substitute the URL shown in previous screenshot for <databricks-instance> in the following to navigate to the Create Secret Scope page: https://<databricks-instance>#secrets/createScope

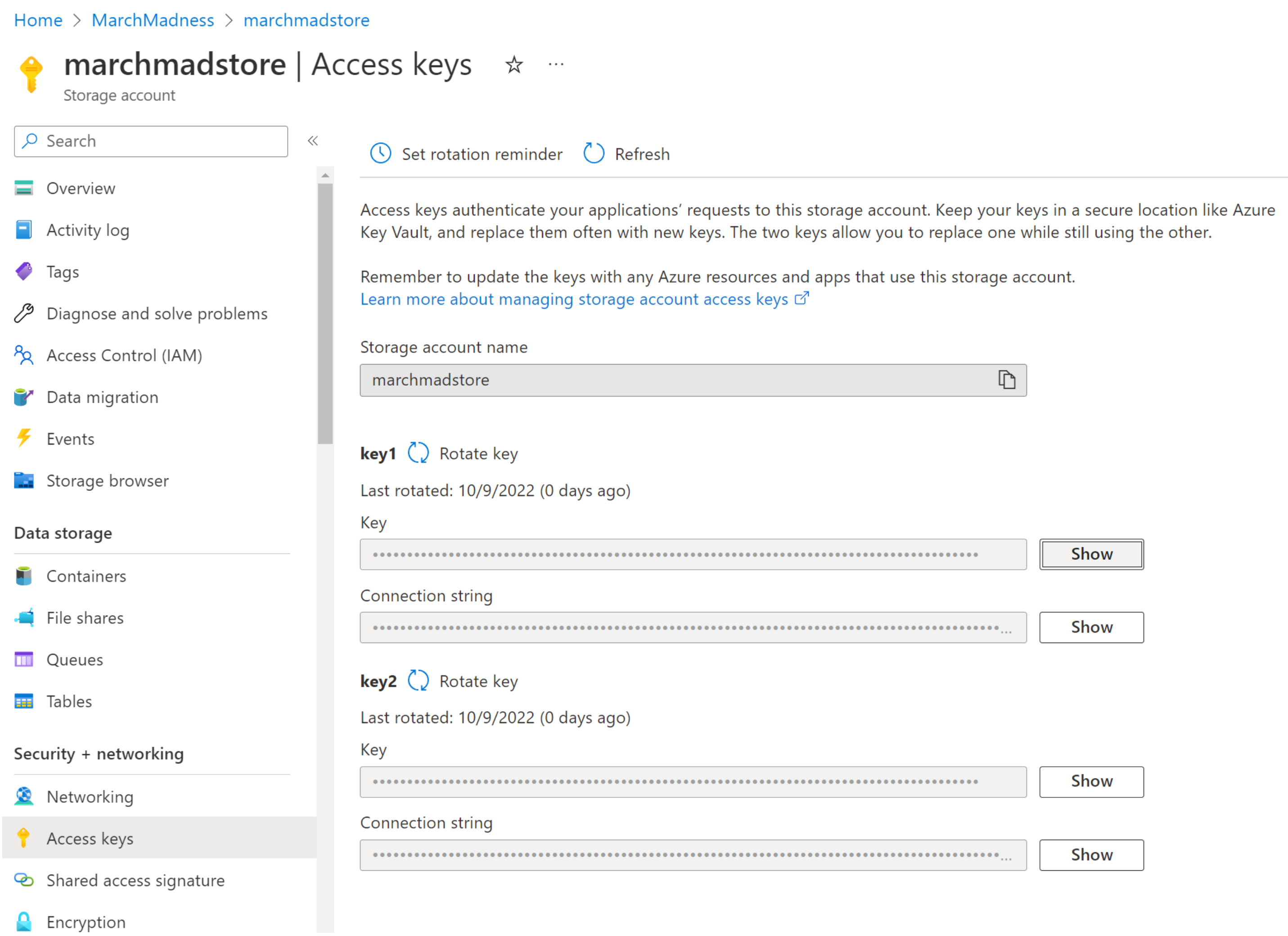
Fill in the values similar to what is shown below. The Azure Key Vault DNS and Resource ID can be found on the Key Vault Properties page as shown in the next screen shot (note - use Vaul URI for the DNS Name).


HomePage / Create Secret Scope
Create Secret Scope
Cancel
Create
A store for secrets that is identified by a name and backed by a specific store
type. Learn more
Scope Name
MarchMadnessScope
Manage Principal
All Users
Azure Key Vault
DNS Name
https://marchmadnesskv.vault.azure.net/
Resource ID
The secret scope named MarchMadnes
Manage secrets in this scope in Azure

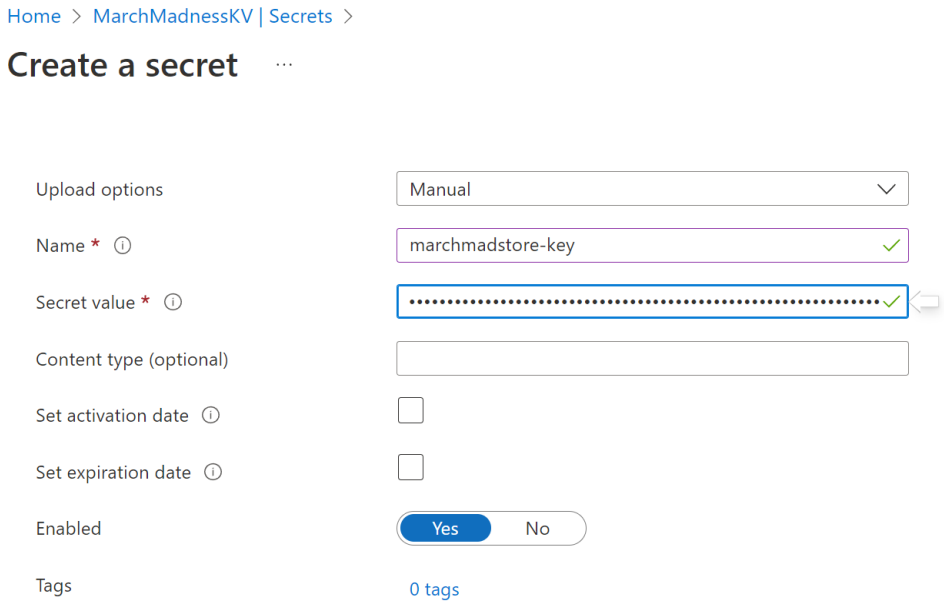


Create a key with the storage account access key

Retrieve the Key from key 1 under Access keys for the storage account or create a Shared access signature (SAS) key and copy the resulting key.

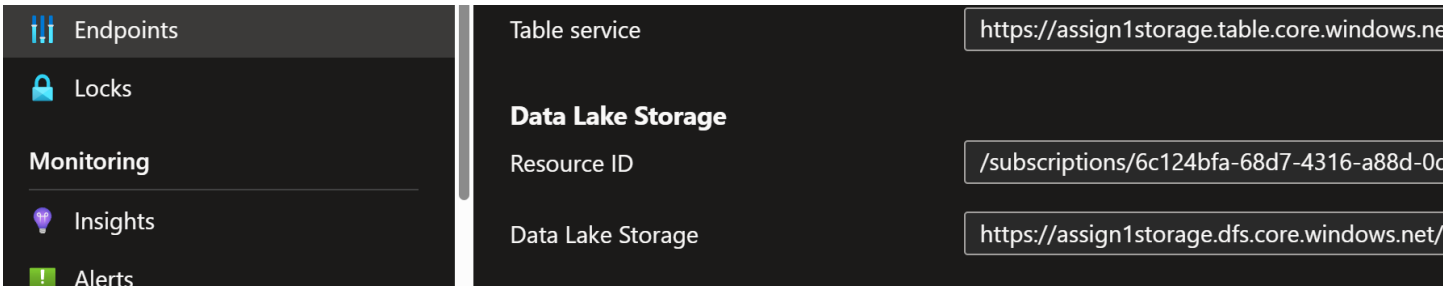

Home > MarchMadness > marchmadstore
marchmadstore | Access keys
Storage account
Search
Overview
Activity log
Tags
Diagnose and solve problems
Access Control (IAM)
Data migration
Events
Storage browser
Data storage
Containers
File shares
Queues
Tables
Security + networking
Networking
Access keys
Shared access signature
Encryption
Set rotation reminder
Refresh
Access keys authenticate your applications' requests to this storage account. Keep your keys in a secure location like Azure Key Vault, and replace them often with new keys. The two keys allow you to replace one while still using the other.
Remember to update the keys with any Azure resources and apps that use this storage account.
Learn more about managing storage account access keys
Storage account name
marchmadstore
key1
Rotate key
Last rotated: 10/9/2022 (0 days ago)
Key
Show
Connection string
Show
key2
Rotate key
Last rotated: 10/9/2022 (0 days ago)
Key
Show
Connection string
Show

Create a secret in your key vault to store the storage account key.



Runtime

Use the secret in the API call to connect the storage for ADB.



Extract the storage end point from the storage account in the Azure portal as shown above. Use the scope and key that you set up earlier in this process.

```
storage_end_point = "assign1storage.dfs.core.windows.net"

my_scope = "MarchMadnessScope"

my_key = "assign1-key"

spark.conf.set(

  "fs.azure.account.key." + storage_end_point,
```

```
    dbutils.secrets.get(scope=my_scope, key=my_key))
# Replace the container name (assign-1-blob) and storage account name (assign1storage) in the uri.
uri = "abfss://assign-1-blob@assign1storage.dfs.core.windows.net/"
```