CANTON WEKE OTIENO

SECURITY ENGINEER TRACK

STUDENT TRACKING NUMBER; ADC-SE01-24010

CYBER SHUJAA

INSTRUCTOR: Dr. Paula

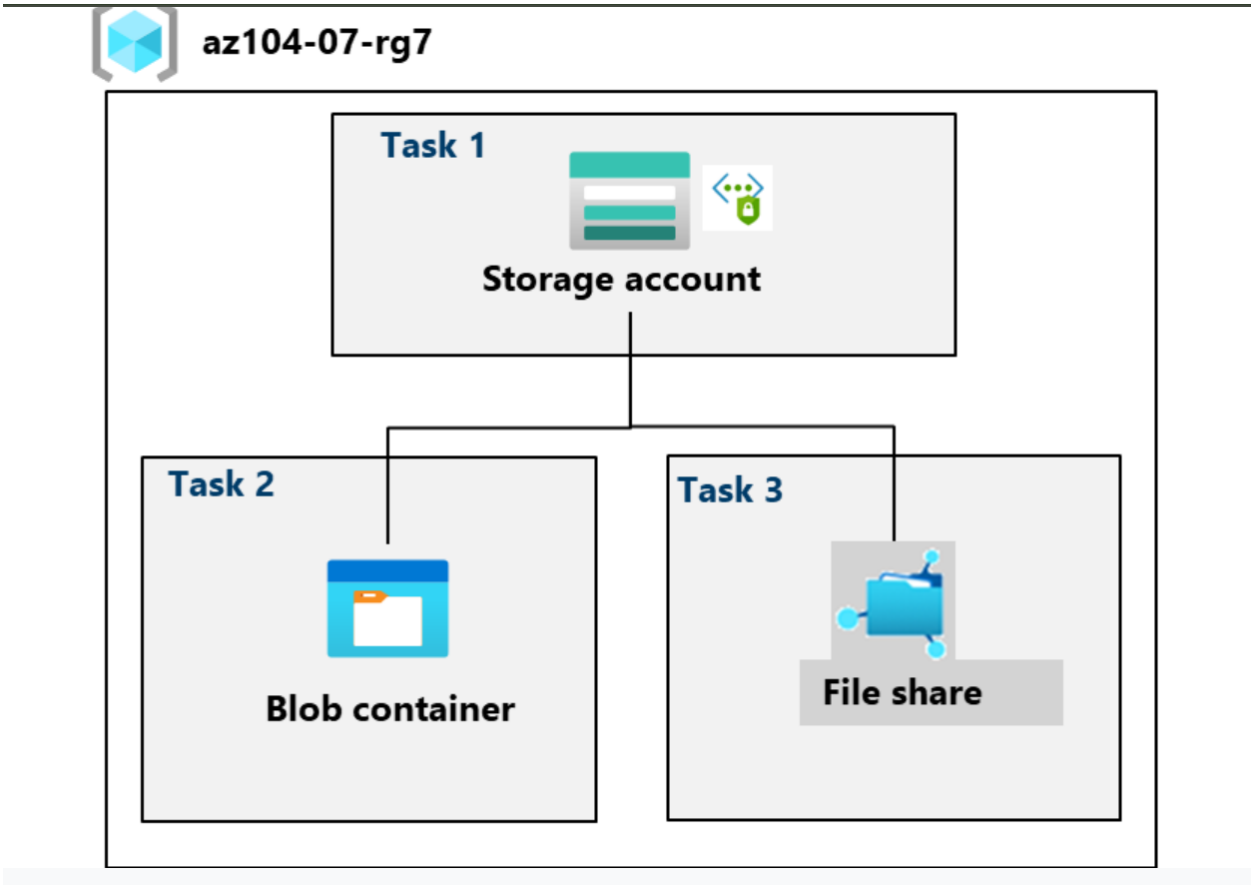
MANAGE AZURE STORAGE

Azure Storage provides a massively scalable object store. It offers a NoSQL database, a messaging store for dependable messaging, and a file system service for cloud computing. Azure Storage supports three categories of data, they include structured data, unstructured data, and finally virtual machine data. Azure Storage offers four data services that can be accessed by using an Azure storage account.

They include:

- **Azure Blob Storage (containers)**: A massively scalable object store for text and binary data.
- **Azure Files**: Managed file shares for cloud or on-premises deployments.
- **Azure Queue Storage**: A messaging store for reliable messaging between application components.
- **Azure Table Storage**: A service that stores nonrelational structured data (also known as structured NoSQL data).

Architecture diagram



Introduction

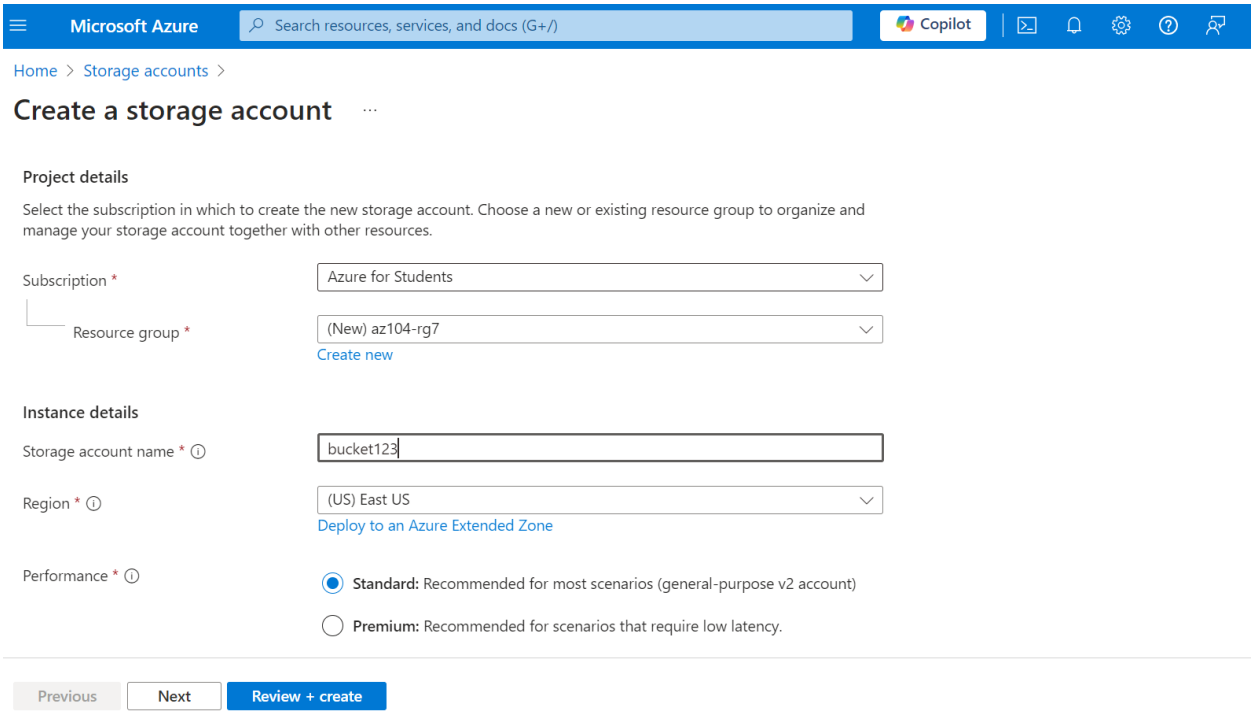
This report contains instructions on to create storage accounts for Azure blobs and azure files, how to to configure and secure blob containers. And finally, the lab illustrates how to use Storage Browser to configure and secure Azure file shares. This lab prepare individual on how to manage azure storage in the real-world scenarios. It contains step on how to fully configure and mange azure storage.

Task 1: Create and configure a storage account.

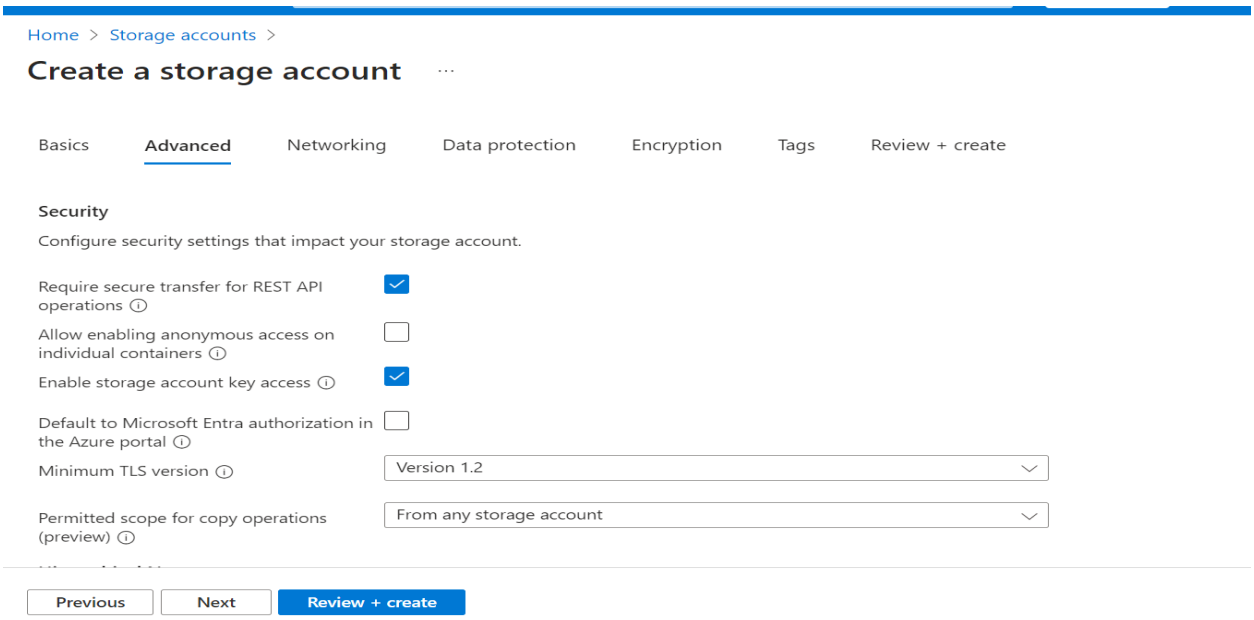
In this lab one I created and configured a storage account. The storage account used geo-redundant storage with no public access.

Instructions

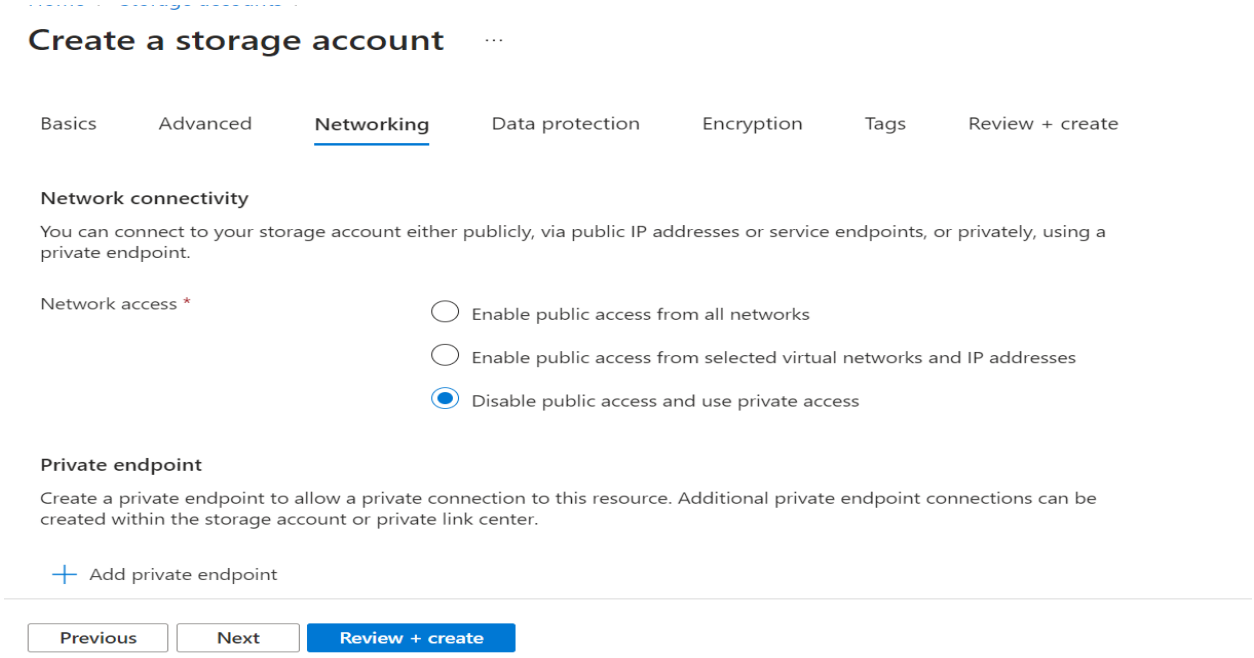
- Sign in to the Azure portal - https://portal.azure.com.
- Search for and select Storage accounts, and then click + Create.
- On the Basics tab of the Create a storage account blade, specify the following settings (leave others with their default values):



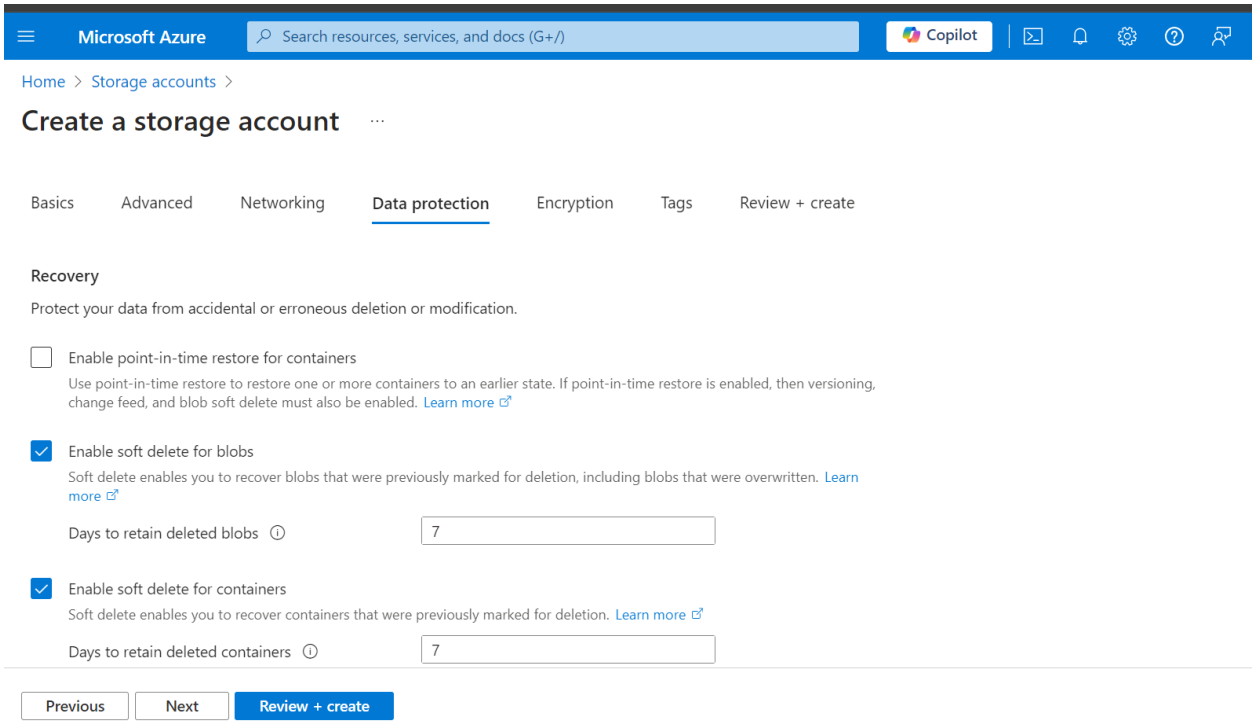
- On the Advanced tab, use the informational icons to learn more about the choices. Take the defaults.



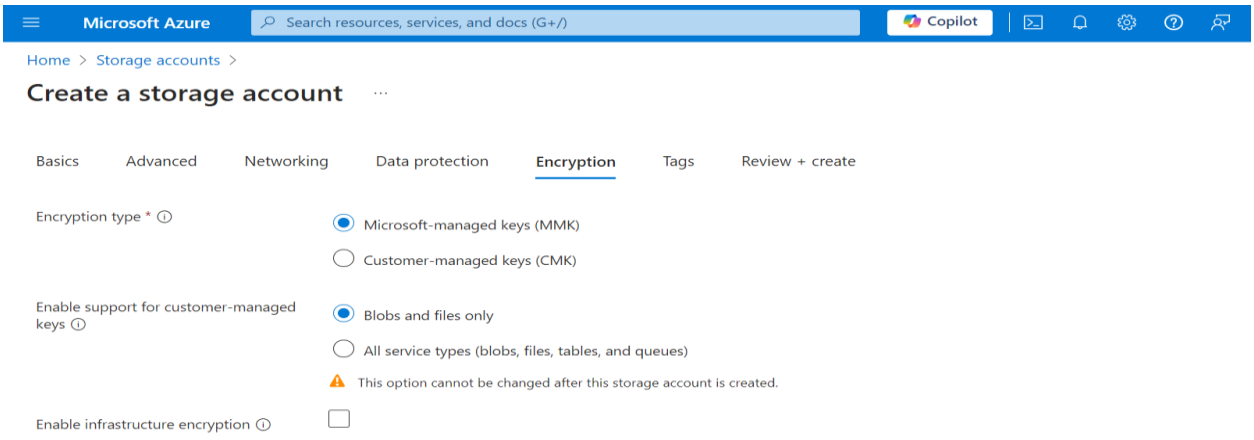
- On the Networking tab, review the available options, select Disable public access and use private access.



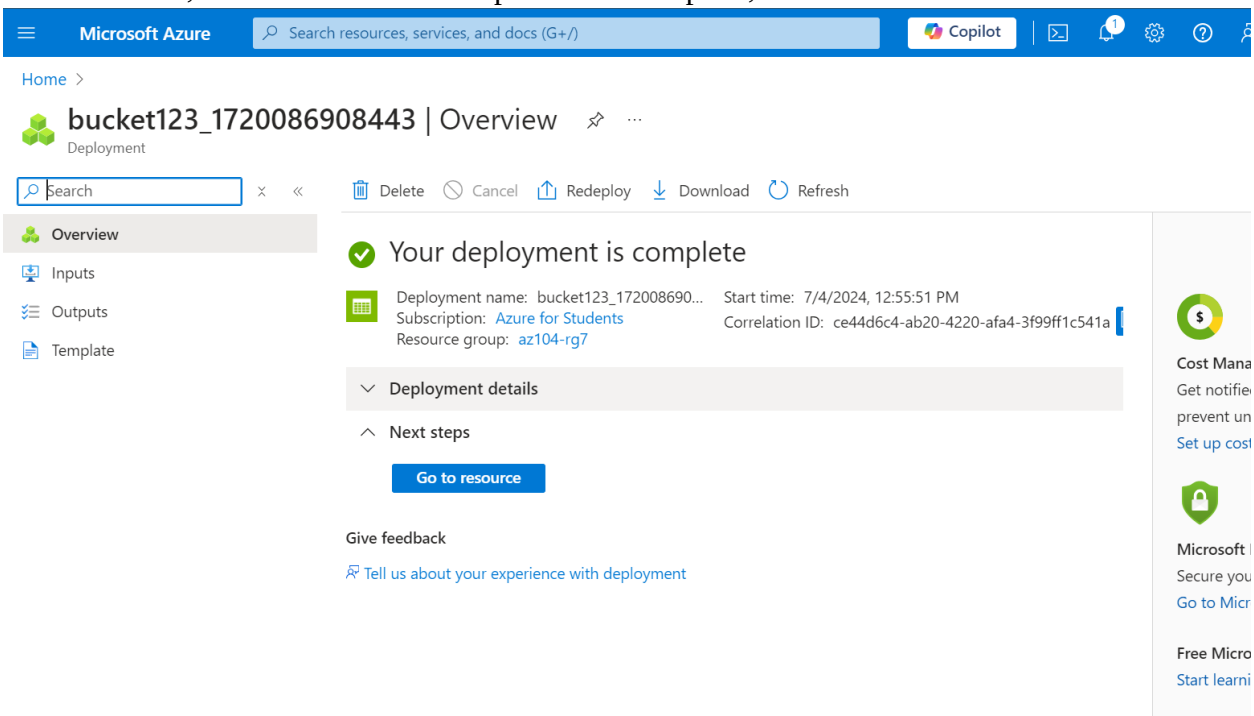
- Review the Data protection tab. Notice 7 days is the default soft delete retention policy. Note you can enable blob versioning. Accept the defaults



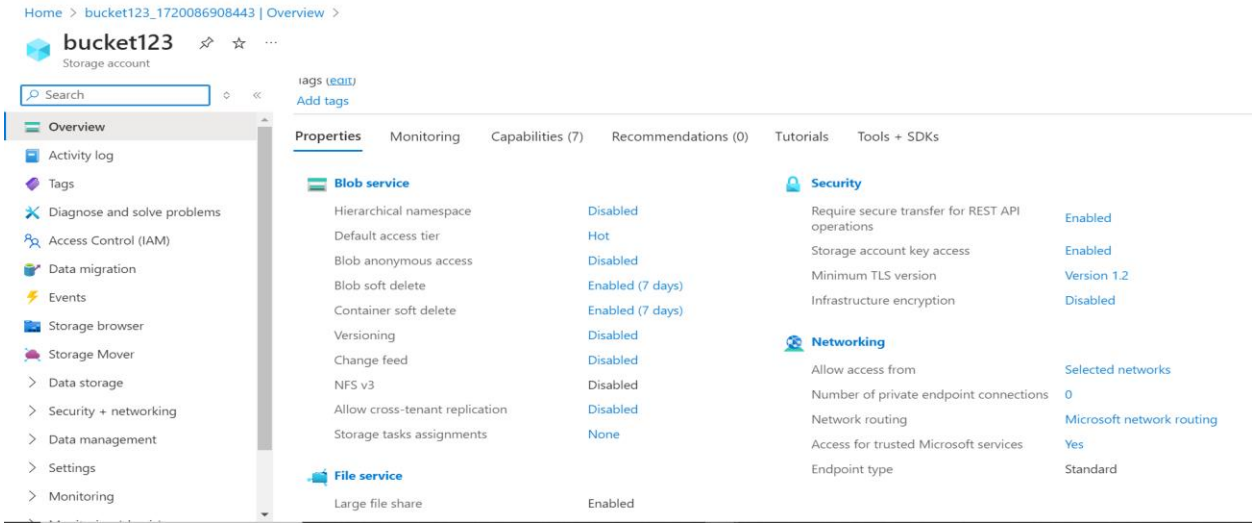
- Review the Encryption tab. Notice the additional security options. Accept the defaults.



- Select Review, wait for the validation process to complete, and then click Create.

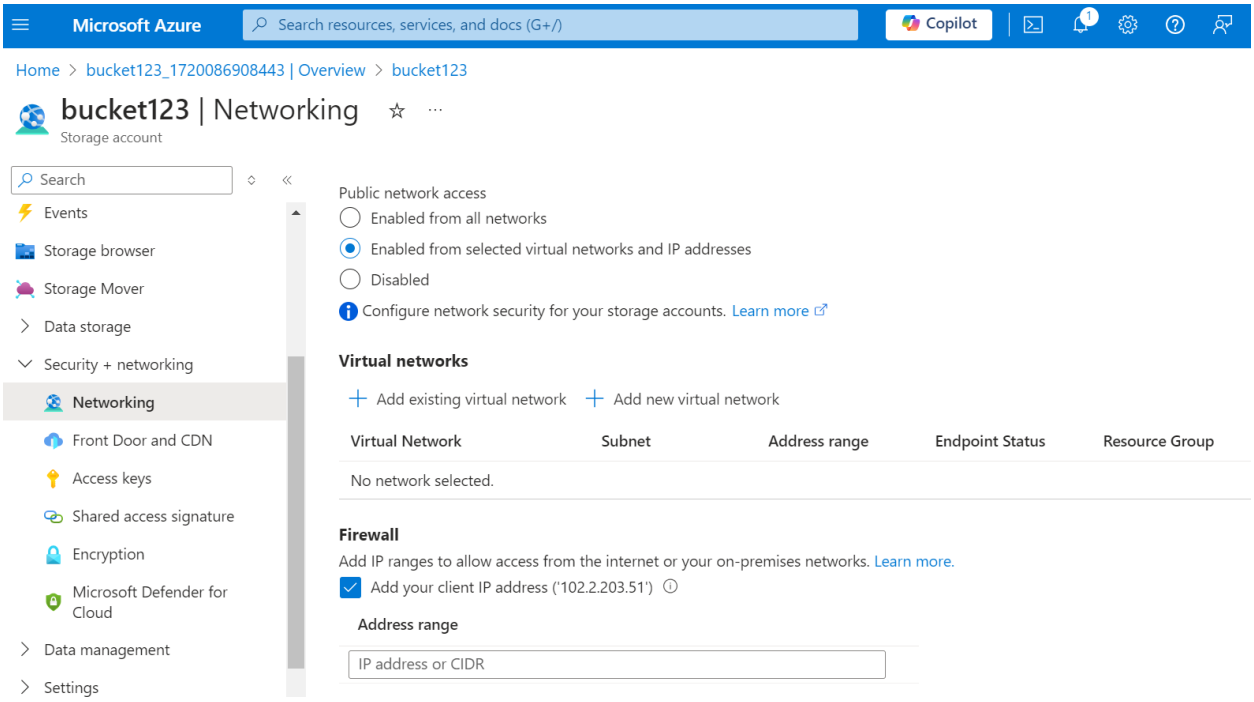


- Once the storage account is deployed, select Go to resource.
- Review the Overview blade and the additional configurations that can be changed. These are global settings for the storage account. Notice the storage account can be used for Blob containers, File shares, Queues, and Tables.

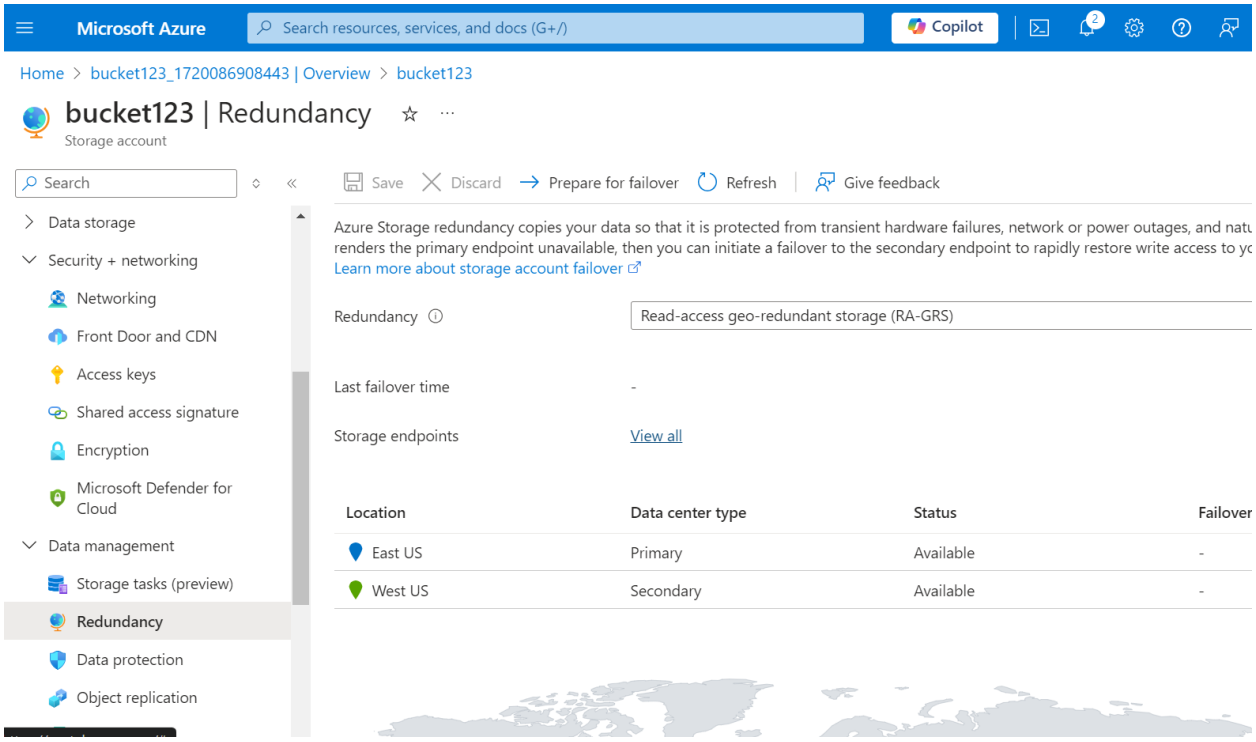


In the Security + networking section, select Networking. Notice public network access is disabled.

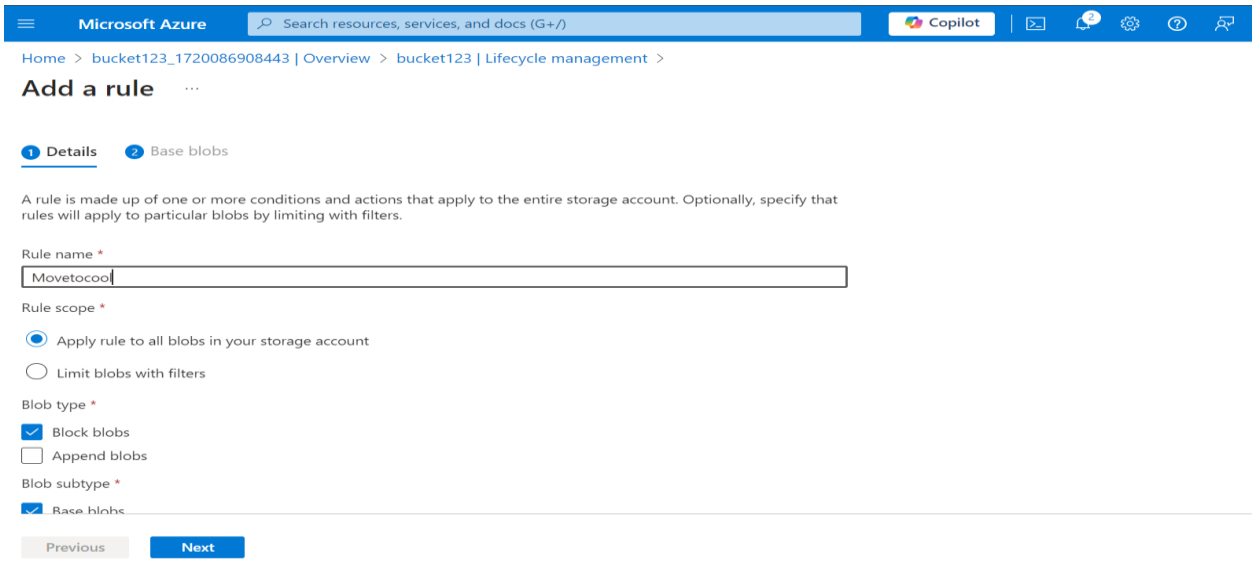
- Change the public access level to Enabled from selected virtual networks and IP addresses.
- In the Firewall section, check the box for Add your client IP address.
- Be sure to Save your changes.



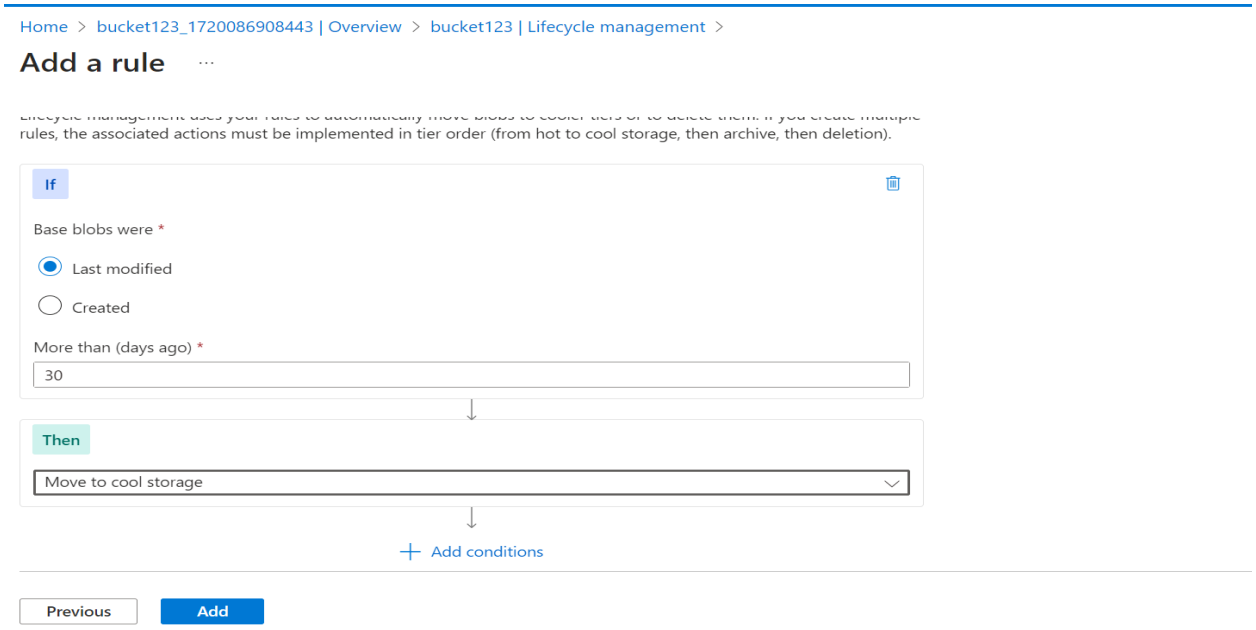
- In the Data management section, view the Redundancy blade. Notice the information about your primary and secondary data center locations.



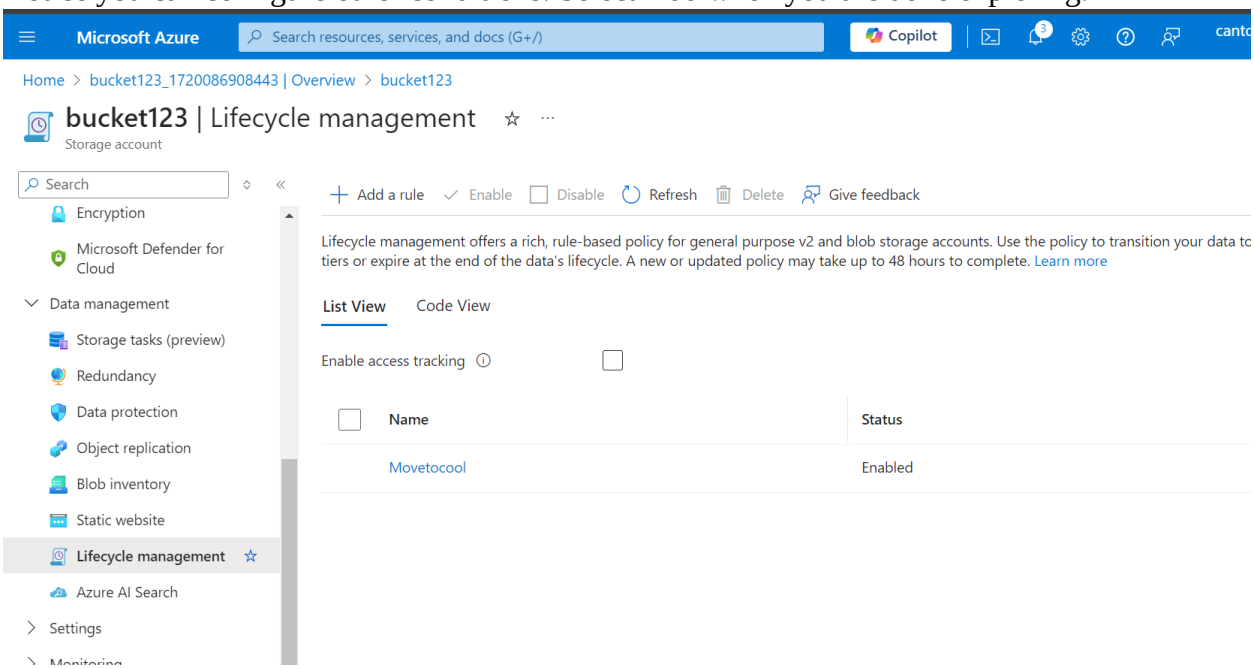
- In the Data management section, select Lifecycle management, and then select Add a rule.Name the rule Movetocool. Notice your options for limiting the scope of the rule.



- On the Base blobs tab, if based blobs were last modified more than 30 days ago then move to cool storage. Notice your other choices.



- Notice you can configure other conditions. Select Add when you are done exploring.



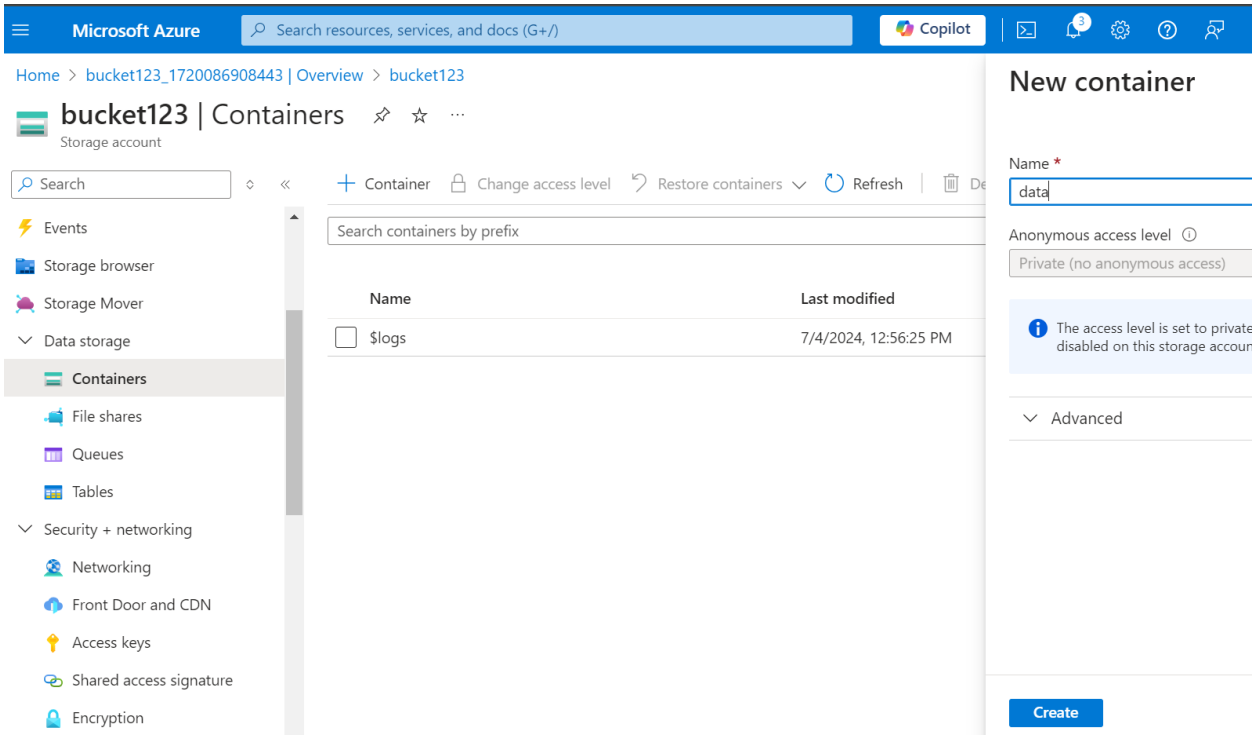
Task 2: Create and configure secure blob storage

In this lab two task, I created a blob container to help store unstructured data and uploaded an image.

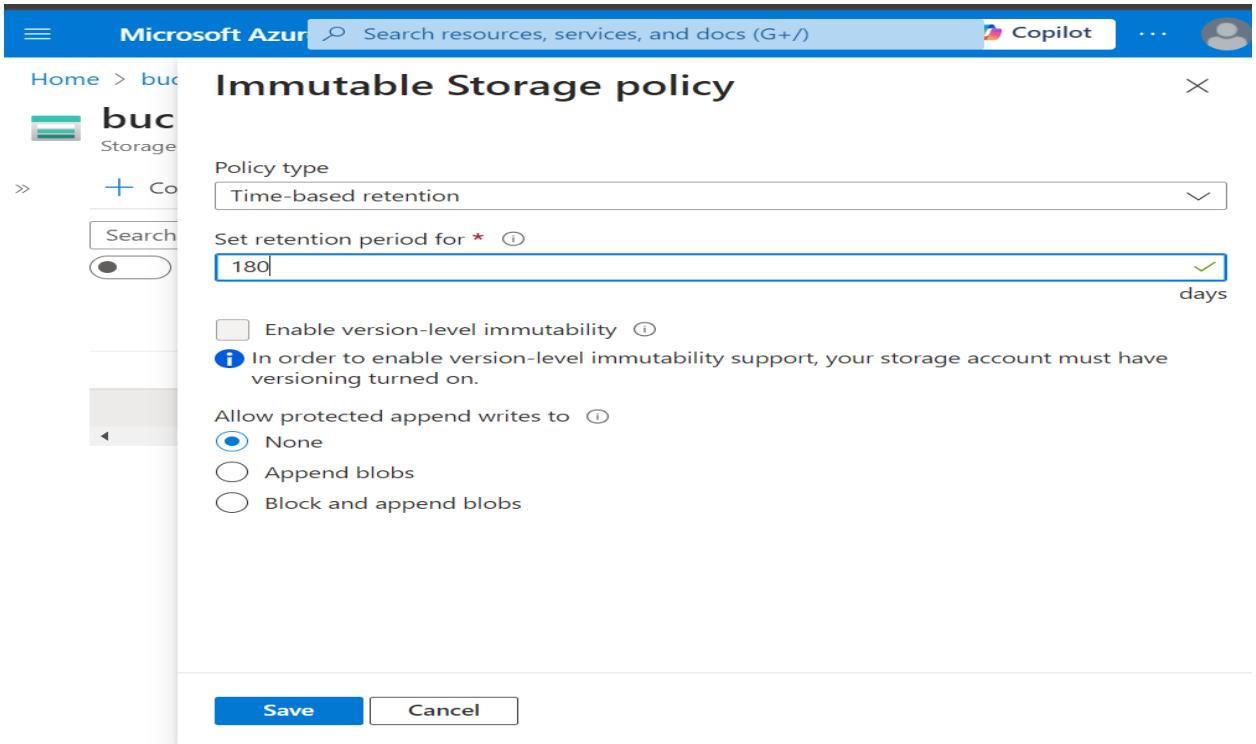
Create a blob container and a time-based retention policy

- Continue in the Azure portal, working with your storage account.

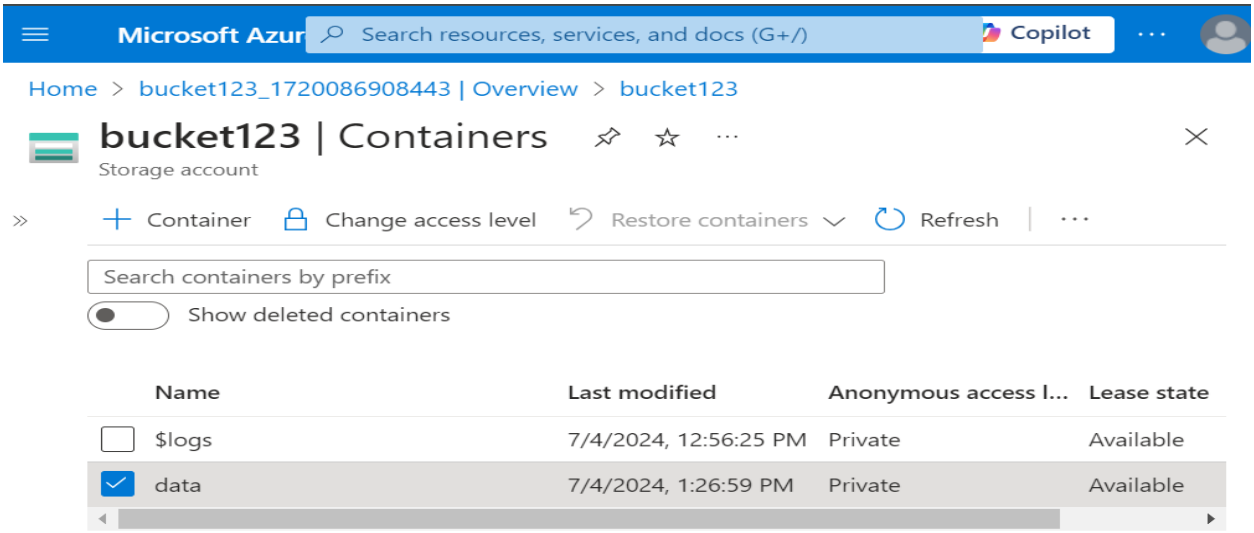
- In the Data storage section, click Containers.
- Click + Container and Create a container with the following settings**:**



- On your container, scroll to the ellipsis (…) on the far right, select Access Policy.
- In the Immutable blob storage area, select Add policy.



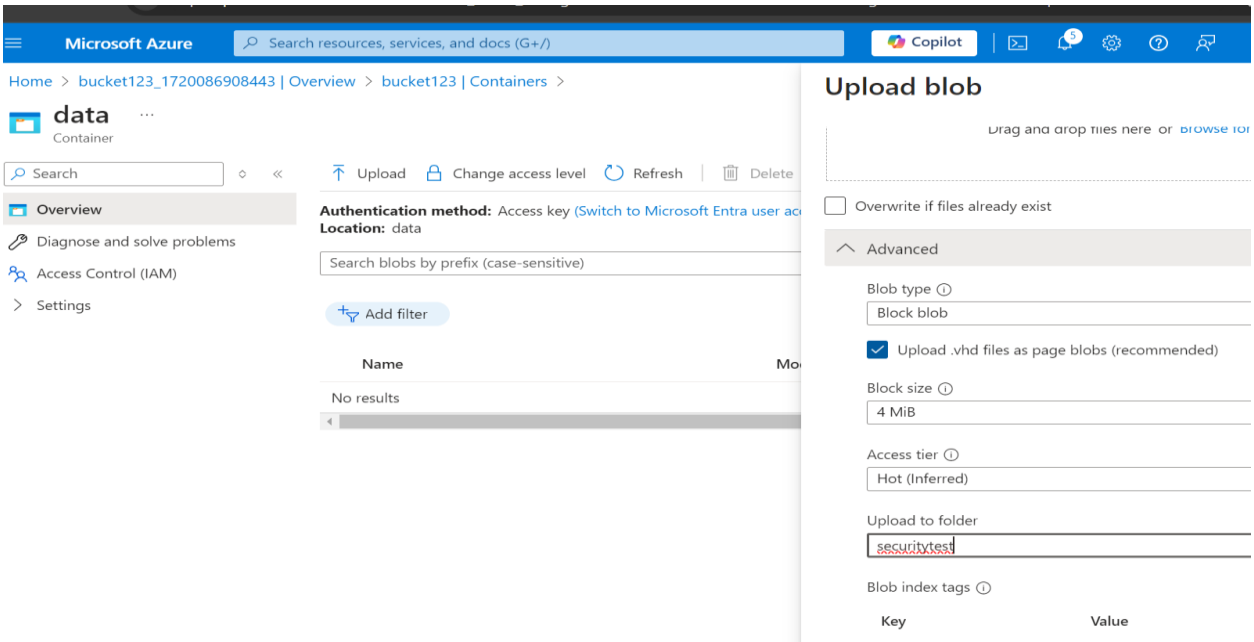
- Select Save.



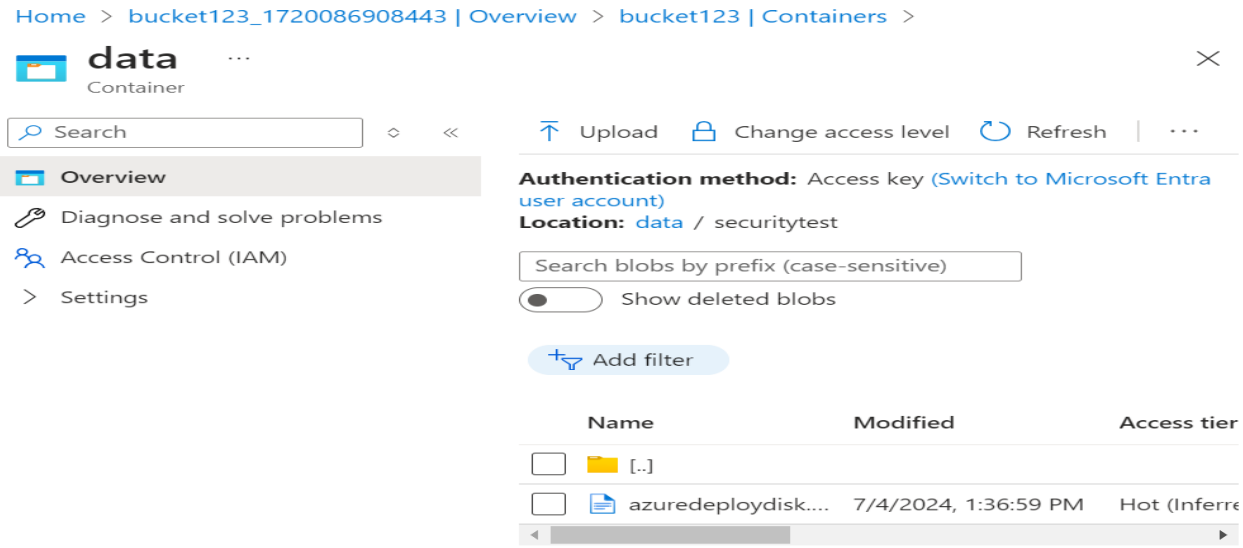
Manage blob uploads

- Return to the containers page, select your data container and then click Upload.
- On the Upload blob blade, expand the Advanced section.

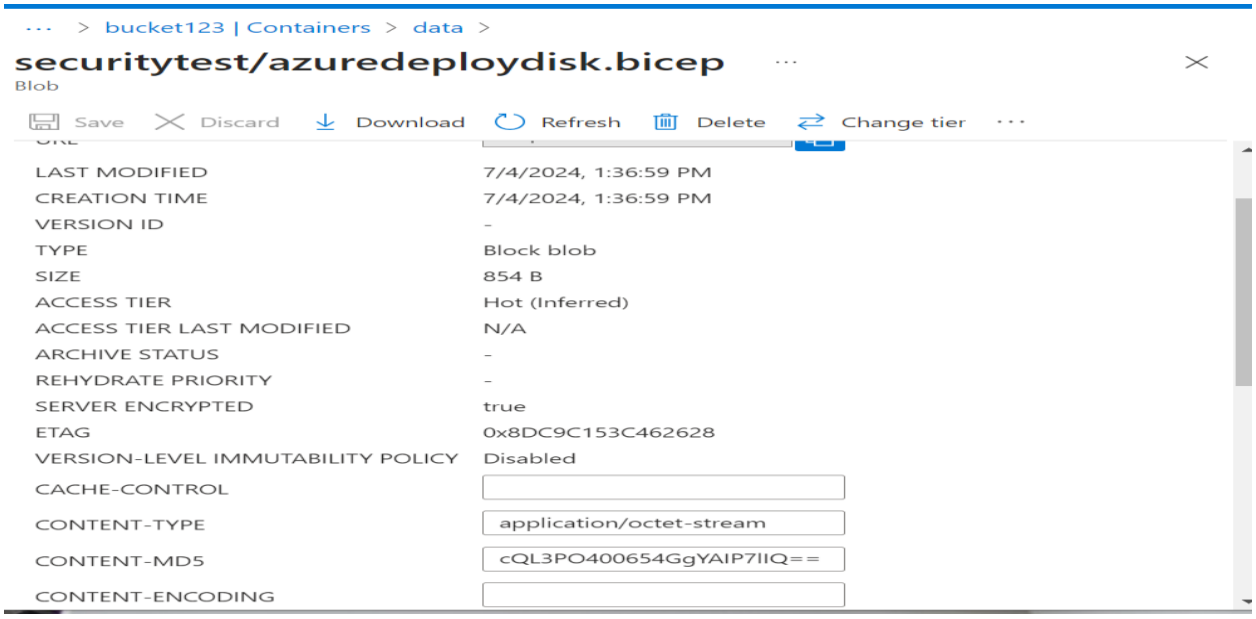
- Confirm you have a new folder, and your file was uploaded.



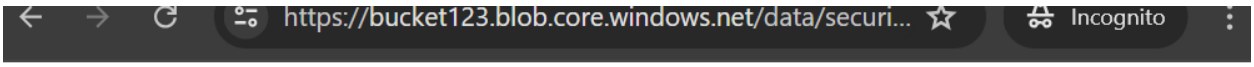
- **Confirm you have a new folder, and your file was uploaded.**



- Select your upload file and review the options including Download, Delete, Change tier, and Acquire lease.



- Copy the file URL and paste into a new Inprivate browsing window.
- You should be presented with an XML-formatted message stating ResourceNotFound or PublicAccessNotPermitted.

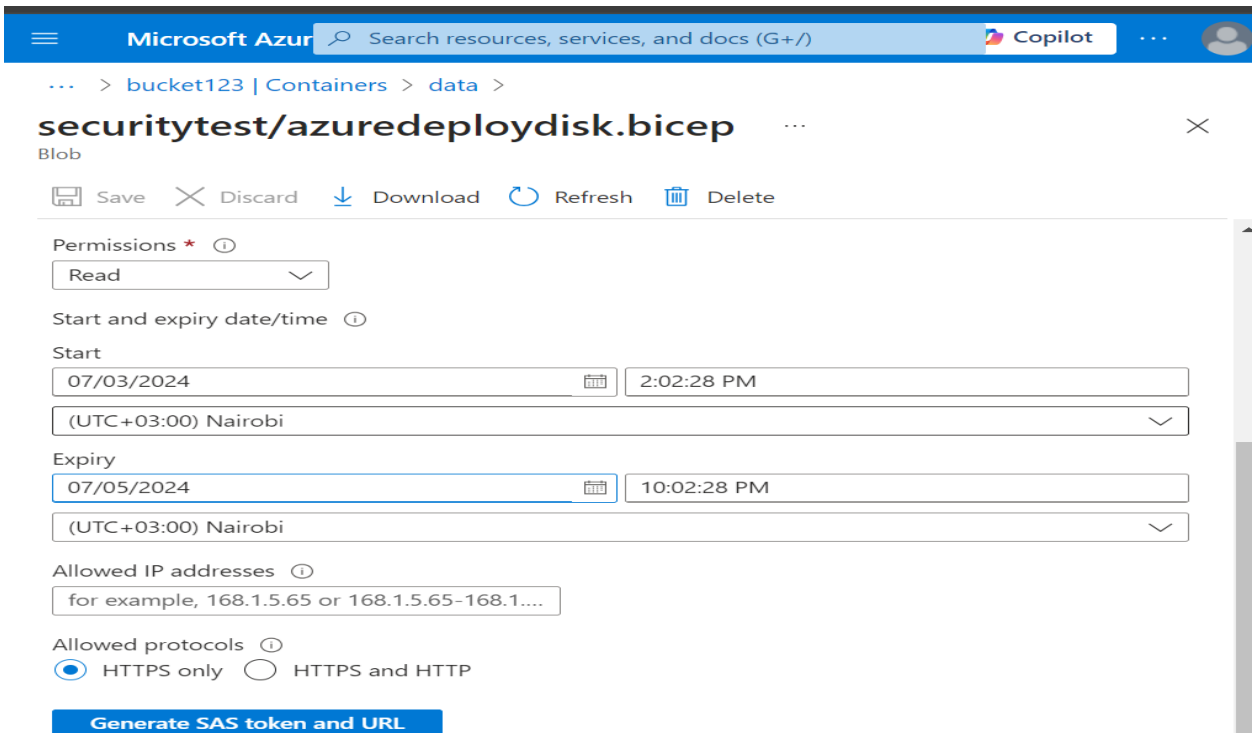


This XML file does not appear to have any style information associated with it. The document tree is shown below.

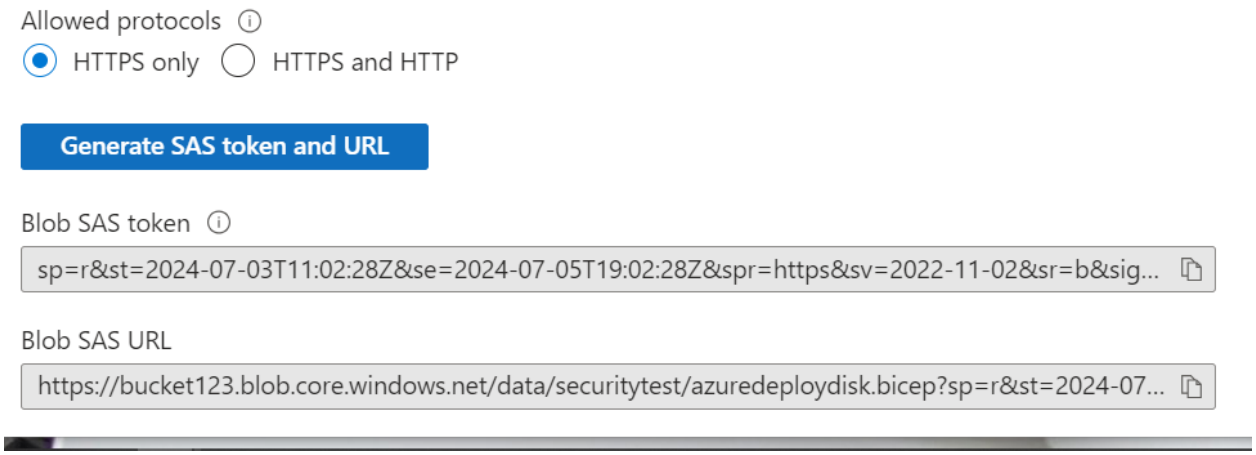

```
▼<Error>
  <Code>PublicAccessNotPermitted</Code>
  <Message>Public access is not permitted on this storage account. RequestId:bd422eab-d01e-
  0075-2100-cea3b3000000 Time:2024-07-04T10:56:08.7195342Z</Message>
 </Error>
```


Configure limited access to the blob storage

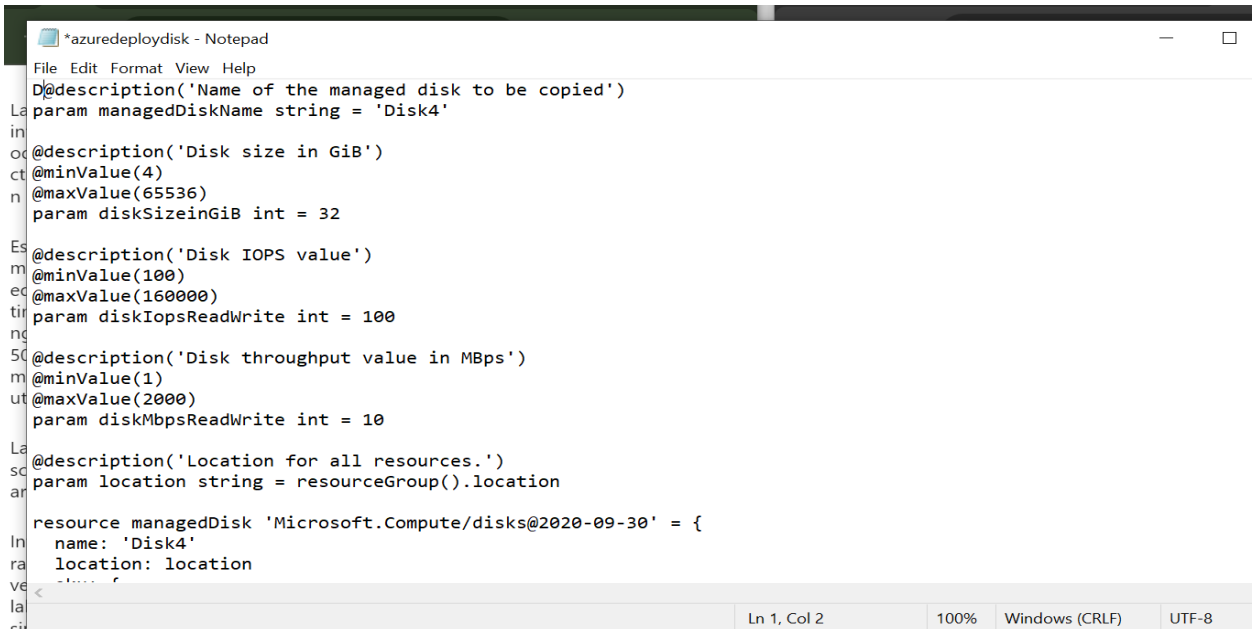
- Select your uploaded file and then on the Generate SAS tab. You can also use the ellipsis (…) to the far right. Specify the following settings (leave others with their default values):



- Click **Generate SAS token and URL**.



- Copy the **Blob SAS URL** entry to the clipboard.
- Open another InPrivate browser window and navigate to the Blob SAS URL you copied in the previous step.

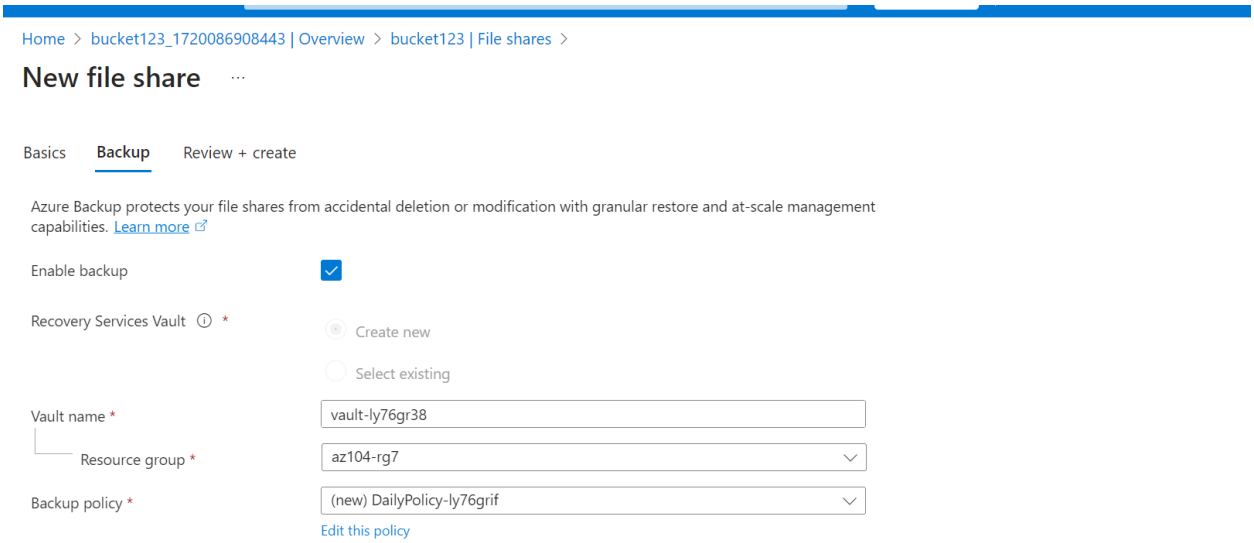


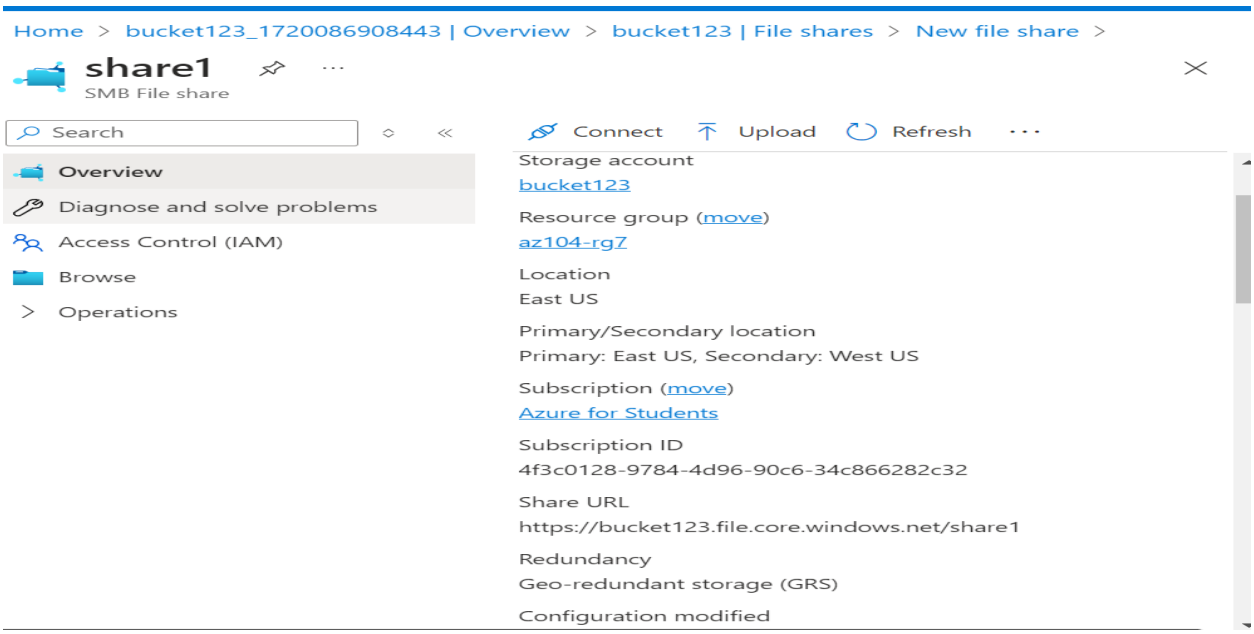
Task 3: Create and configure an Azure File storage.

In this task three I configured azure file share. Azure File Shares makes it easy for multiple users to share files and collaborate on projects

Instructions for creating File storage

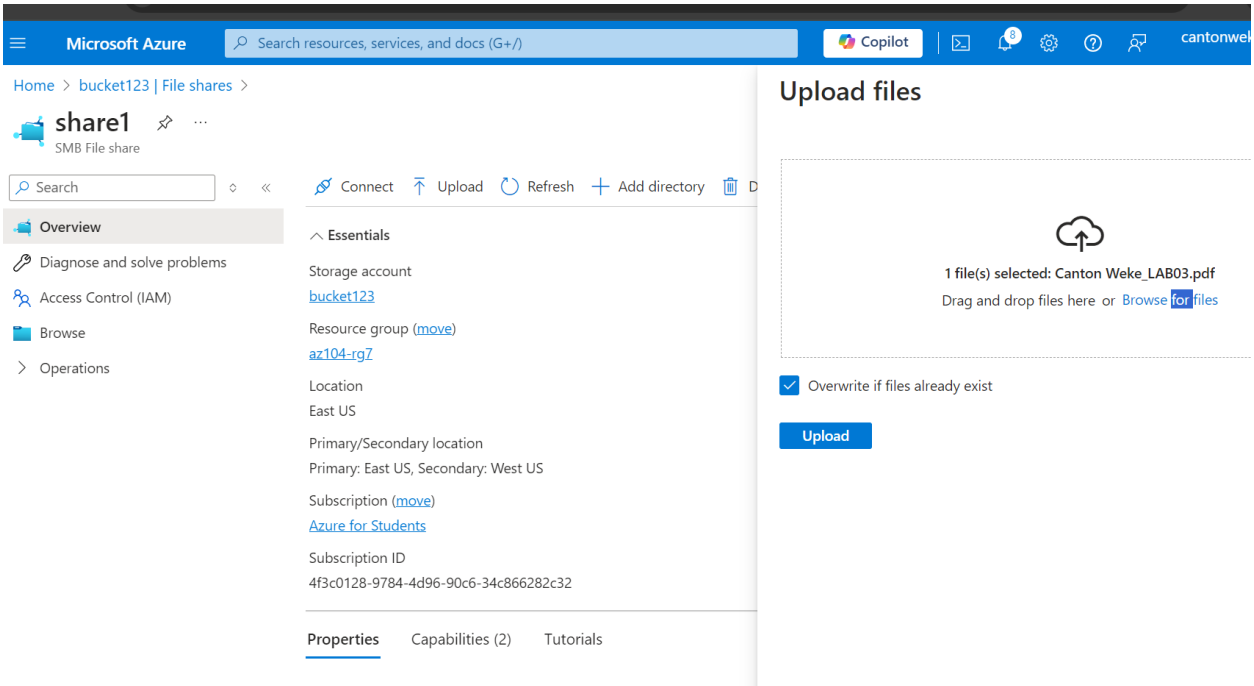
- Create the file share and upload a file
- In the Azure portal, navigate back to your storage account, in the Data storage section, click File shares.
- Click + File share and on the Basics tab give the file share a name, share1.
- Notice the Access tier options. Keep the default Transaction optimized.
- Move to the Backup tab and ensure Enable backup is not checked. We are disabling backup to simplify the lab configuration.
- Click Review + create, and then Create. Wait for the file share to deploy.





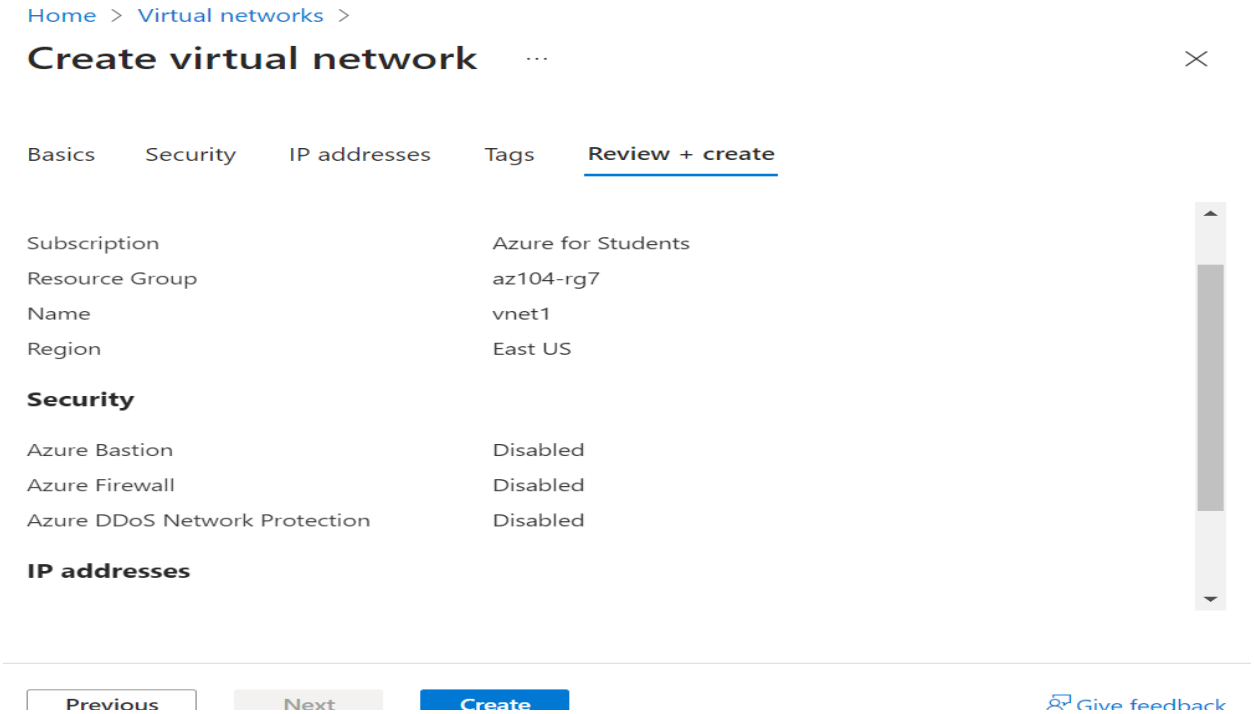
Explore Storage Browser and upload a file

- Return to your storage account and select **Storage browser**. The Azure Storage Browser is a portal tool that lets you quickly view all the storage services under your account.
- Select **File shares** and verify your **share1** directory is present.
- Select your **share1** directory and notice you can **+ Add directory**. This lets you create a folder structure.
- Select **Upload**. Browse to a file of your choice, and then click **Upload**.

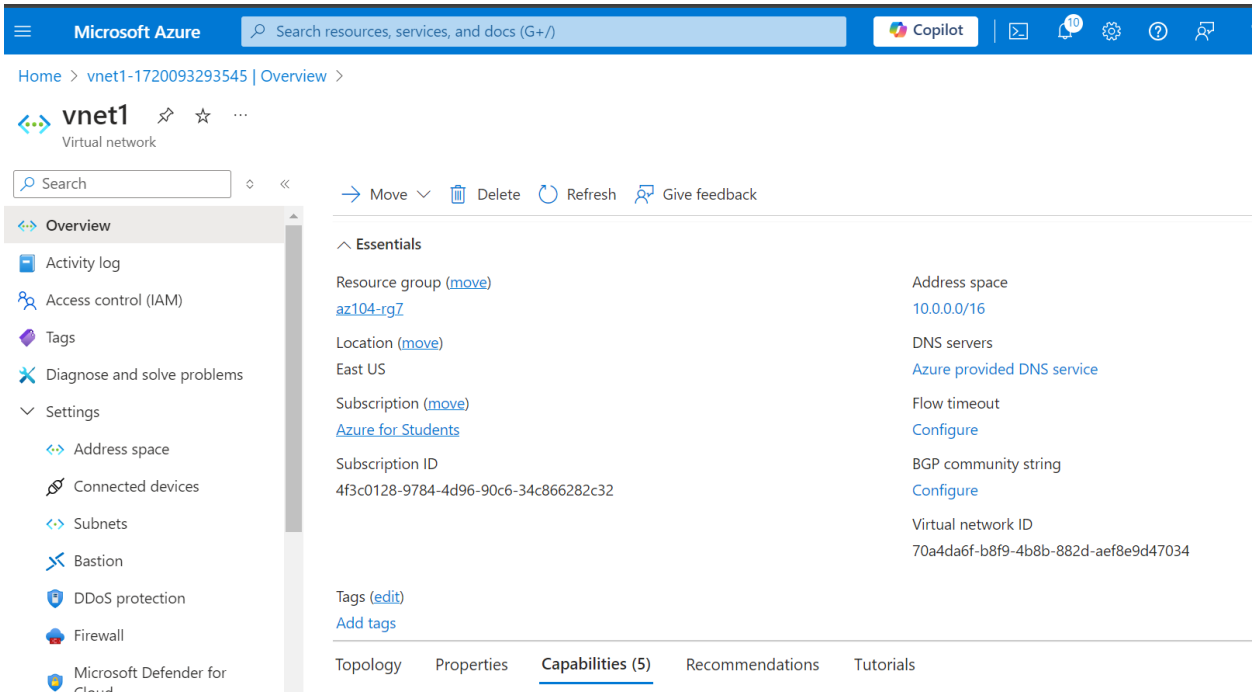


Restrict network access to the storage account

- In the portal, search for and select Virtual networks.
- Select + Create. Select your resource group. and give the virtual network a name, vnet1.
- Take the defaults for other parameters, select Review + create, and then Create.

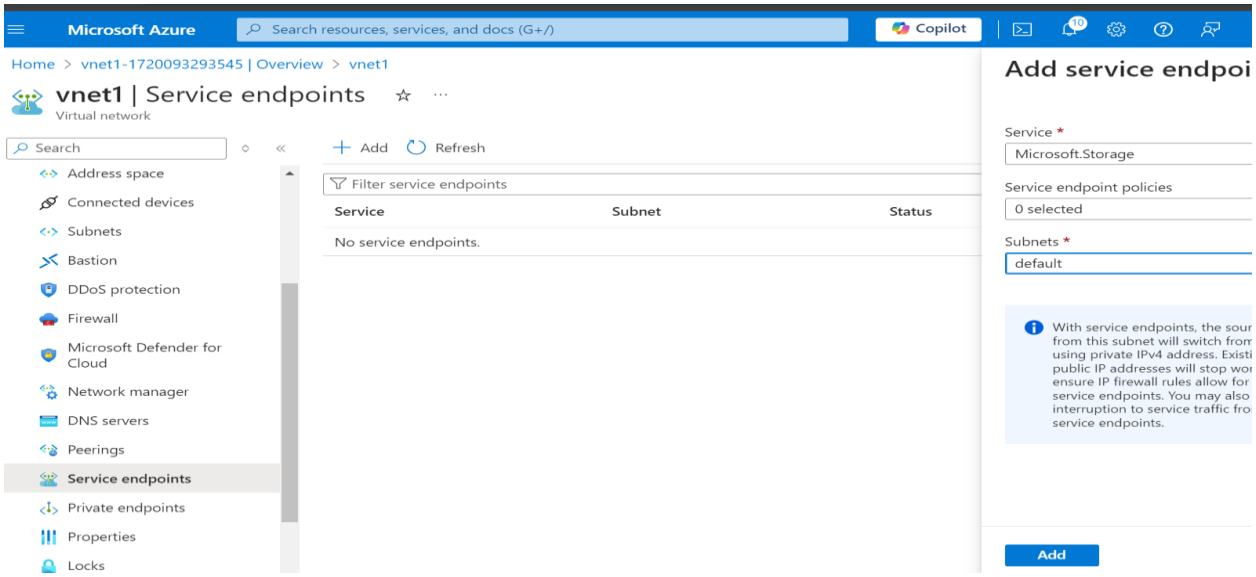


- Wait for the virtual network to deploy, and then select Go to resource.

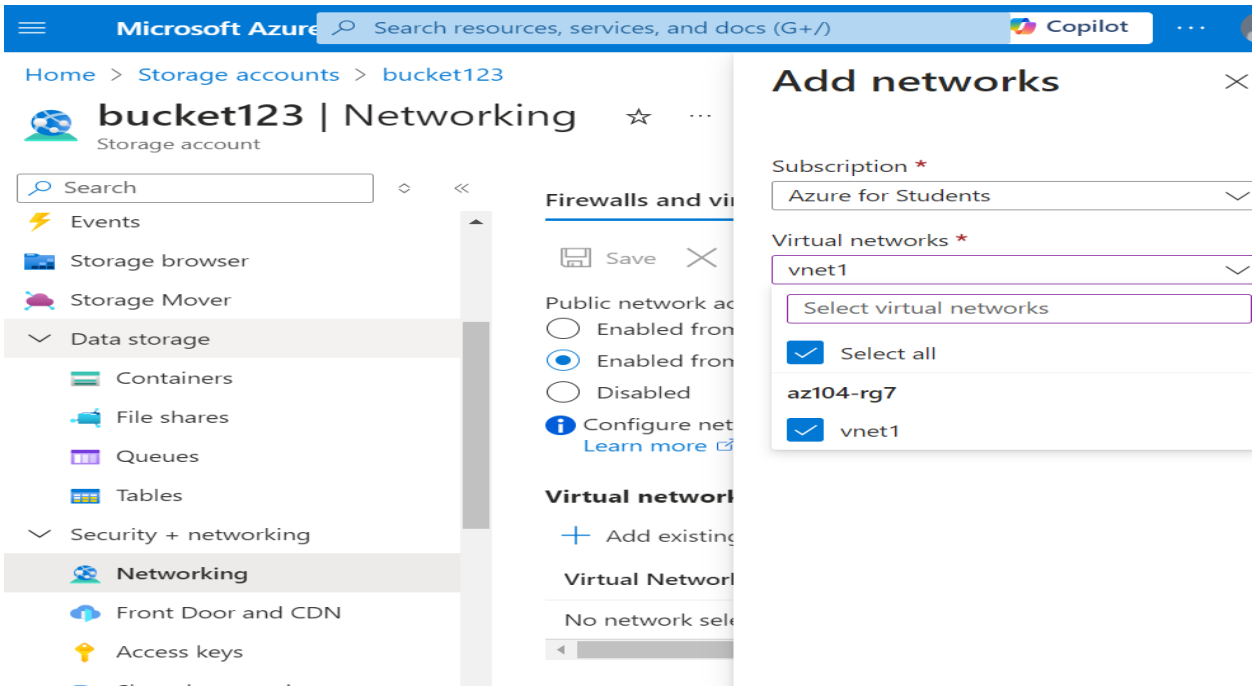


- **In the Settings section, select the Service endpoints blade.**

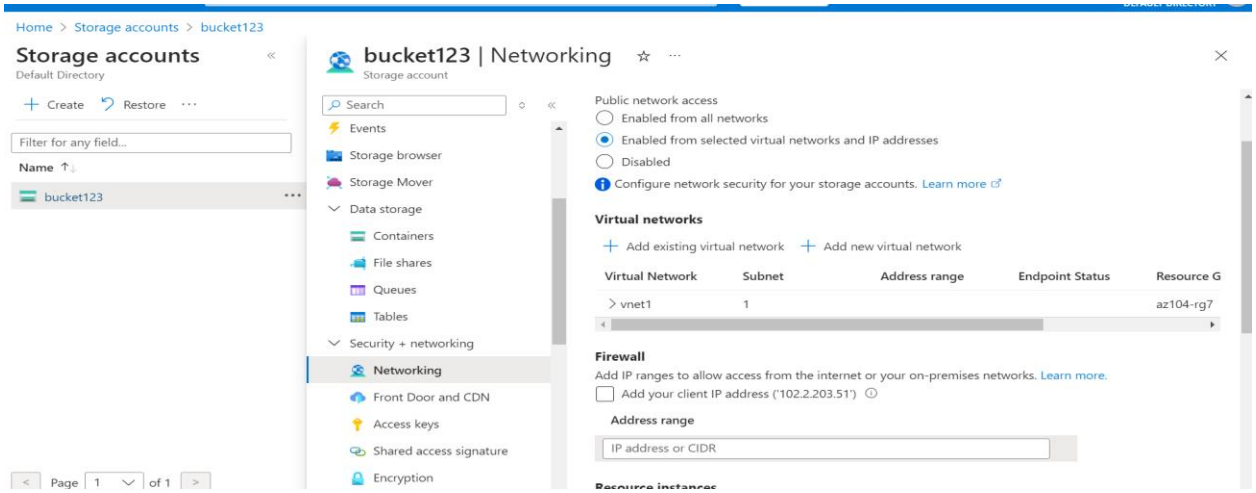
✓ Select Add.
✓ In the Services drop-down select Microsoft.Storage.
✓ In the Subnets drop-down check the Default subnet.
✓ Click Add to save your changes.
✓ Return to your storage account.

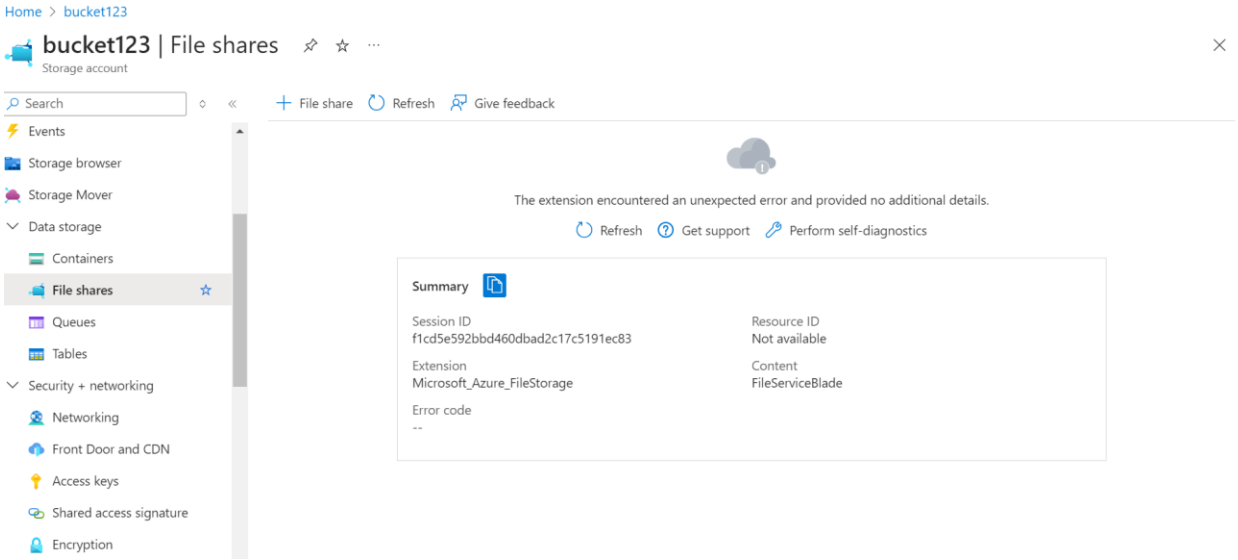


- In the Security + networking section, select the Networking blade.
- Select add existing virtual network and select vnet1 and default subnet, select Add.



- In the Firewall section, Delete your machine IP address. Allowed traffic should only come from the virtual network.
- Be sure to Save your changes.





Conclusion

This lab exercise provided a deep dive into the foundational elements of Azure storage services, highlighting the critical aspects of creating and securing various types of storage. Task one involved creating and configuring a storage account, which is the cornerstone of Azure storage services. This task reinforced the importance of proper setup and configuration to ensure scalability, cost-efficiency, and robust performance of our storage solutions. It also underscored the versatility of Azure storage accounts in supporting diverse storage needs, from general-purpose use cases to specialized scenarios requiring advanced data handling capabilities.

Task two focused on creating and securing blob storage, which is essential for storing large amounts of unstructured data like logs, media files, and backups. This task highlighted the significance of implementing security best practices, such as setting up role-based access control (RBAC) and encryption, to protect sensitive data from unauthorized access and potential breaches. We also explored how to efficiently manage and optimize blob storage to support data-intensive applications.

Finally task three addressed the creation and configuration of secure Azure file storage. This task emphasized the importance of providing secure, scalable, and accessible file shares for applications and services. It reinforced the need to implement secure access methods, such as network file sharing over SMB (Server Message Block) with encryption and integration with Azure Active Directory, to ensure data integrity and availability. Together, these tasks highlighted the critical role of secure and well-managed storage solutions in supporting the seamless operation and security of cloud-based applications in the real world scenerios.