CST 8912 lab 3

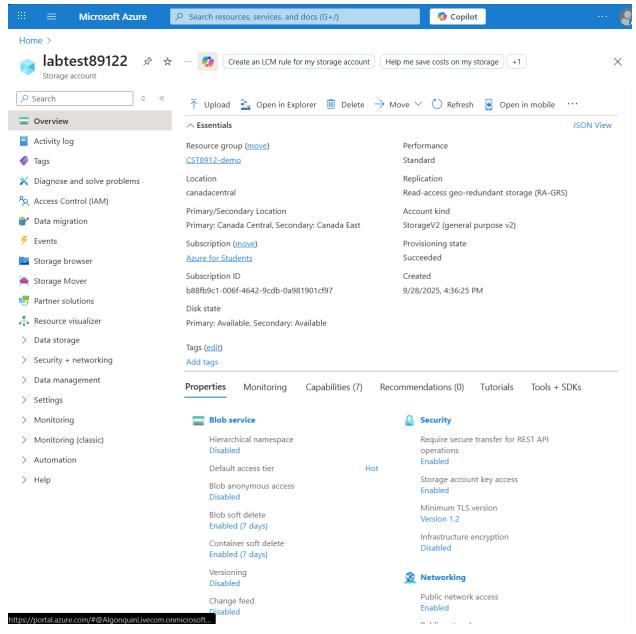
By: Obaida Kandakji ST#: 041272028

Introduction

The purpose of this lab was to gain hands-on experience with Microsoft Azure Storage Accounts, blob containers, and Shared Access Signatures (SAS). By completing the lab, I learned how redundancy options affect reliability, how access tiers help manage costs, how to securely delegate access using SAS, and how to automate data management with lifecycle rules.

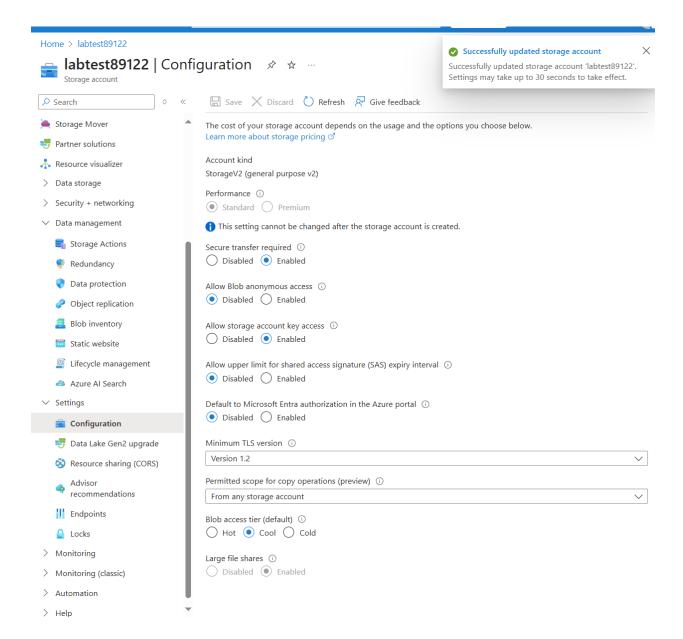
Step 1 - Create a Storage Account

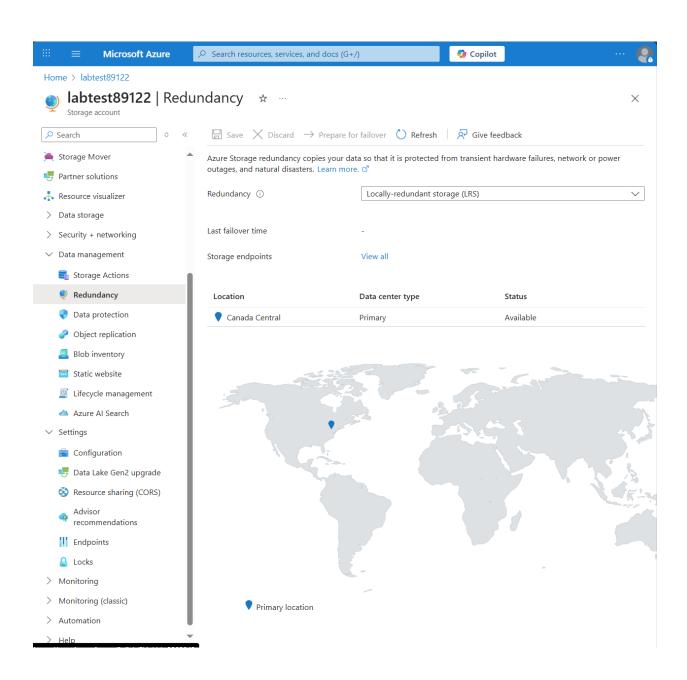
I created a storage account named **labtest8912** in the resource group **CST8912-demo**, in the **Canada Central** region. For redundancy, I selected **Geo-redundant storage (GRS)**.



Step 2 - Configure Redundancy and Access Tier

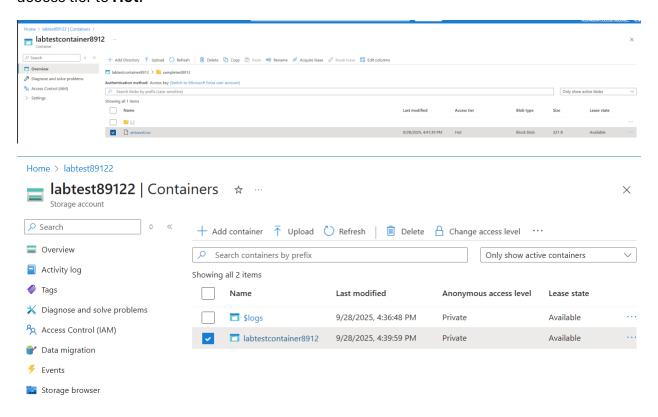
Next, I explored the redundancy settings and changed them from **GRS** to **LRS** (**Local-redundant storage**). I also set the default blob access tier for the account to **Cool**.





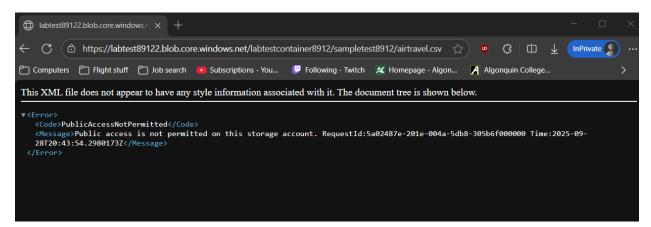
Step 3 - Create a Container and Upload a Blob

I created a container named **labtestcontainer8912** with the public access level set to **Private**. Inside the container, I created a virtual folder named **sampletest8912** and uploaded a CSV file from the provided sample dataset. I then changed the uploaded blob's access tier to **Hot**.



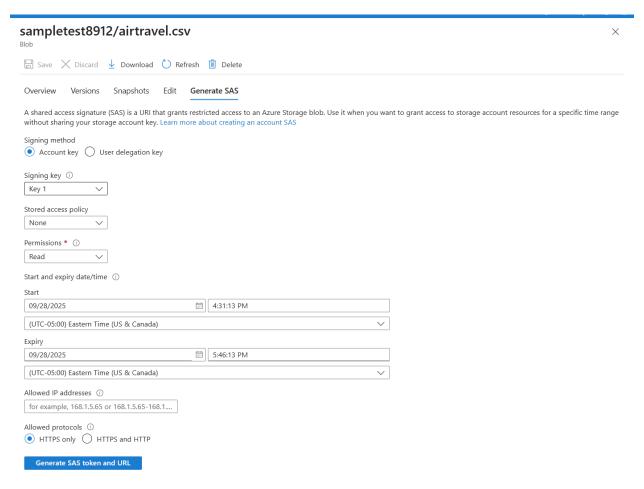
Step 4 - Test Private Access

To confirm that containers are private by default, I copied the blob's URL and tested it in a private/incognito browser window. As expected, I was denied access because no SAS token was provided.



Step 5 - Generate and Test SAS

I generated a **Shared Access Signature (SAS)** for the uploaded blob, granting **read-only** permissions with an expiry time. Using the SAS URL in an incognito browser window, I was able to successfully access and download the blob.





Step 6 - Create Lifecycle Management Rule

I created a lifecycle management rule named **myrule8912**. The rule was scoped to **base block blobs** and set to move any blob last modified more than 15 days ago to **Cool storage**.

Add a rule ...

Details Base blobs Filter set

Lifecycle management uses your rules to automatically move blobs to cooler tiers or to delete them. If you create multiple rules, the associated actions must be implemented in tier order (from hot to cool storage, then archive, then deletion).

If

Base blobs were *

Last modified

Created

More than (days ago) *

15

Then

Move to cold storage

Home > CST8912-demo > labtest89122 | Lifecycle management >

Update a rule

Details Base blobs

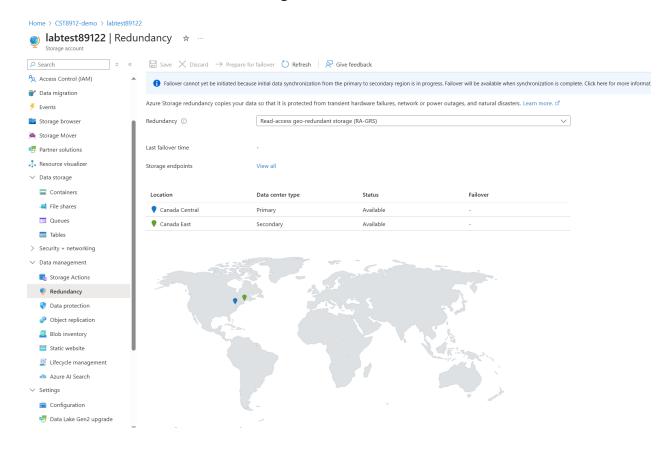
Lifecycle management uses your rules to automatically move blobs to cooler tiers or to delete them. If you create multiple rules, the associated actions must be implemented in tier order (from hot to cool storage, then archive, then deletion).



Added stretch goal of autodeletion after 30days

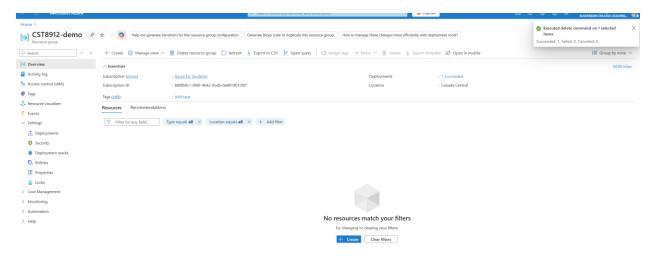


Switched back to RA-GRS for stretch goal



Step 7 - Cleanup

Finally, I deleted the resource group **CST8912-demo** to clean up all resources and avoid unnecessary charges.



Reflection

This lab provided valuable hands-on practice with Azure Storage services. I gained experience creating and configuring a storage account, exploring redundancy options, and understanding how blob access tiers balance cost with performance. I also learned the importance of security in cloud storage: while containers are private by default, SAS tokens provide controlled, time-limited access. Finally, I saw how lifecycle rules can help automate data management and cost optimization. Overall, this lab reinforced key cloud computing principles of resiliency, scalability, and secure access management.