

AZ-104

Administer Azure Storage





Extra ID

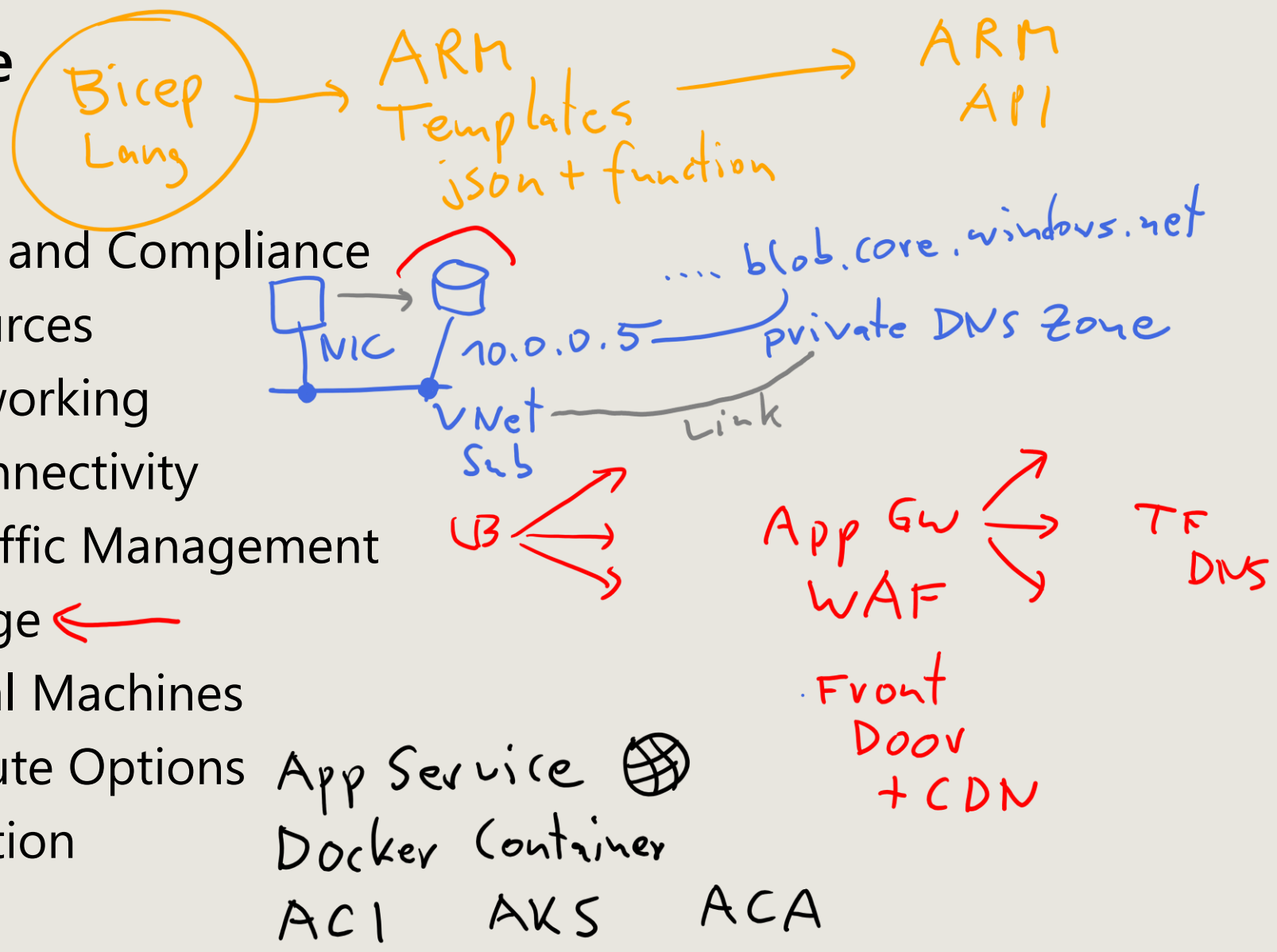
AZ-104 Course Outline

- 01: Administer Identity
- 02: Administer Governance and Compliance
- 03: Administer Azure Resources
- 04: Administer Virtual Networking
- 05: Administer Intersite Connectivity
- 06: Administer Network Traffic Management
- 07: Administer Azure Storage ←
- 08: Administer Azure Virtual Machines
- 09: Administer PaaS Compute Options
- 10: Administer Data Protection
- 11: Administer Monitoring

Lab

Lab

Lab Lab Lab



Learning Objectives - Administer Azure Storage

- Configure Storage Accounts ✓
- Configure Blob Storage
- Configure Storage Security
- Configure Azure Files and File Sync
- Lab 07 – Manage Azure Storage

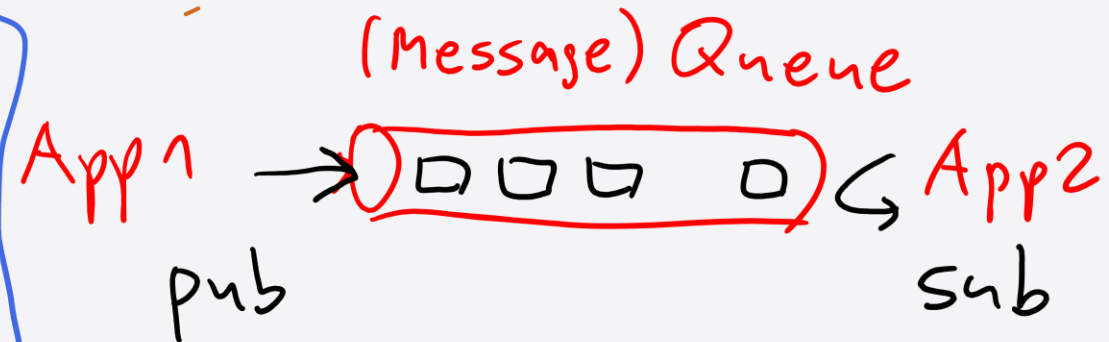
Storage API	Portal
azcopy	PS CLI



Container	Blobs	HTTPS
Files	SNB	HTTPS
Tables		HTTPS
Queue		HTTPS

Identity-based access

→ SA = File Server
"NTFS" Permissions
Share Permission RBAC



Configure Storage Accounts



Explore Azure Storage Services

General Purpose v2

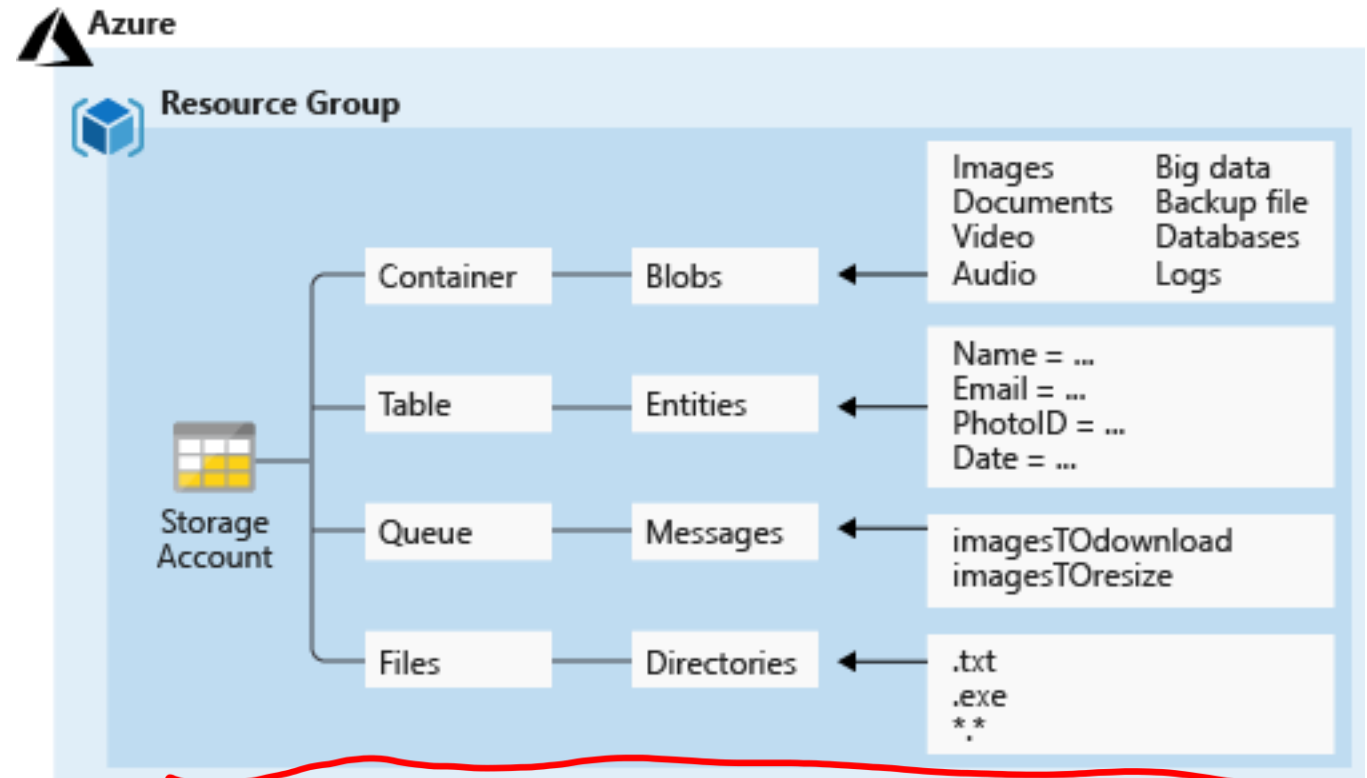
A service that you can use to store files, messages, tables, and other types of information

Azure Containers: A massively scalable object store for text and binary data

Azure Tables: Ideal for storing structured, non-relational data

Azure Queues: A messaging store for reliable messaging between application components

Azure Files: Managed file shares for cloud or on-premises deployments



Key Vault — RSA Keys
Secrets
Cert

Determine Storage Account Kinds

All storage accounts are encrypted using Storage Service Encryption (SSE) for data at rest

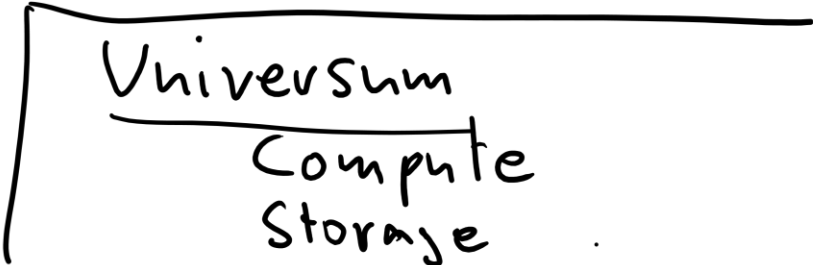
Storage Account	Recommended usage
Standard general-purpose v2	Most scenarios including Blob, File, Queue, Table, and Data Lake Storage.
Premium block blobs Disk	Block blob scenarios with high transactions rates, or scenarios that use smaller objects or require consistently low storage latency.
Premium file shares	Enterprise or high-performance file share applications.
Premium page blobs	Premium high-performance page blob scenarios.

HDD

SSD

Backup append blob

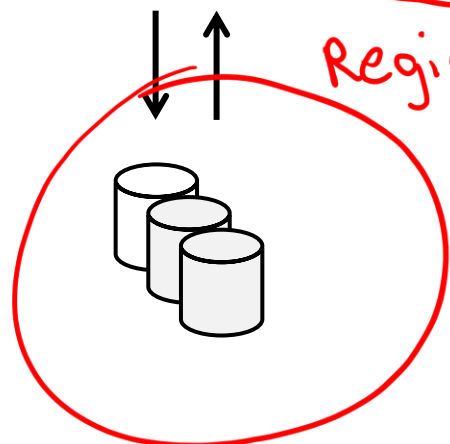
Recovery Service Vault



Determine Replication Strategies (1 of 2)

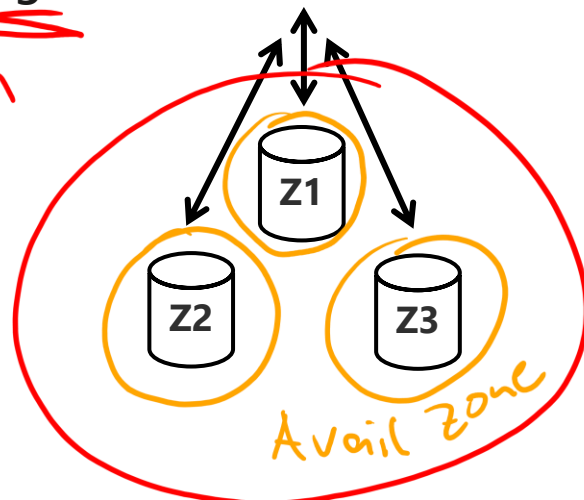
~~Single region~~

Region



LRS

- Three replicas, one region
- Protects against disk, node, rack failures
- Write is acknowledged when all replicas are committed
- Superior to dual-parity RAID

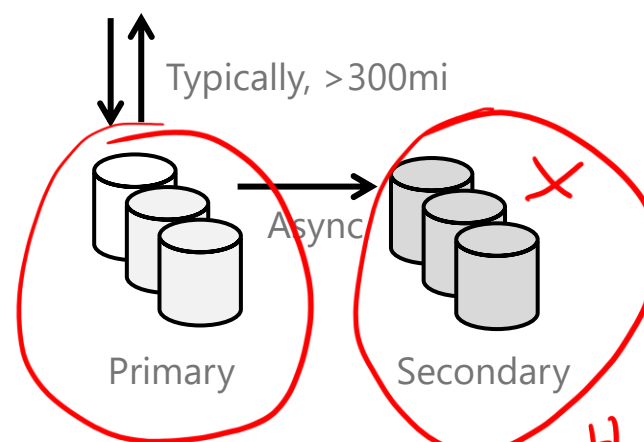


ZRS

- Three replicas, three zones, one region
- Protects against disk, node, rack, and zone failures
- Synchronous writes to all three zones

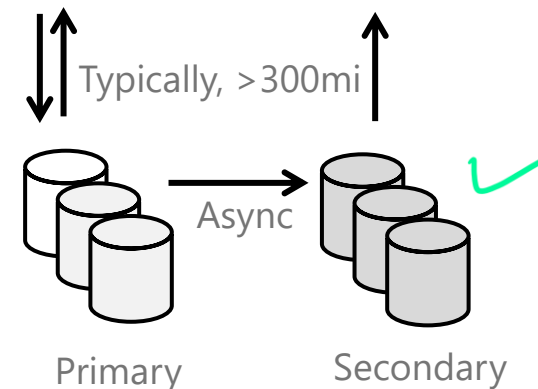
~~Multiple regions~~

2



GRS

- Six replicas, two regions (three per region)
- Protects against major regional disasters
- Asynchronous copy to secondary

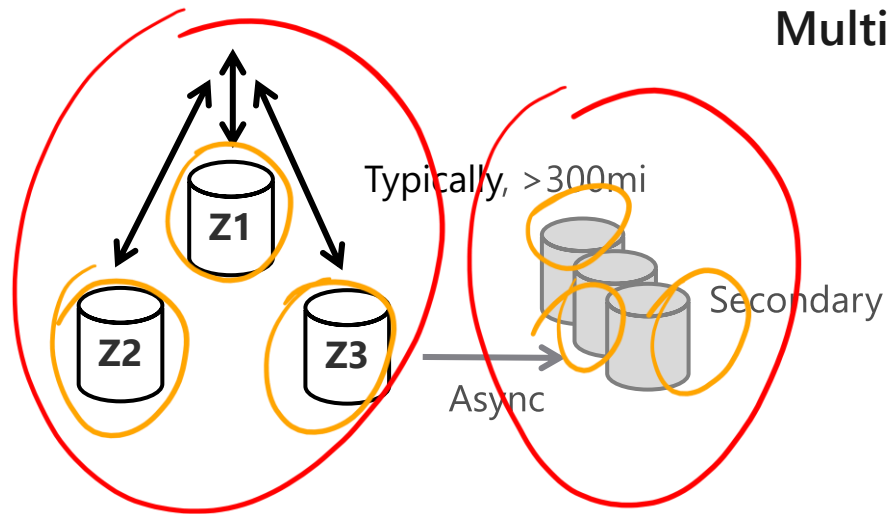


RA-GRS

- GRS + read access to secondary
- Separate secondary endpoint
- Recovery point objective (RPO) delay to secondary can be queried

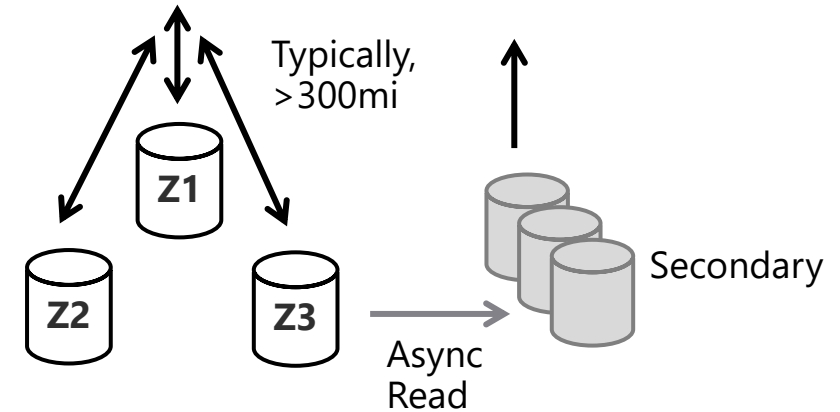


Determine Replication Strategies (2 of 2)



GZRS

- Six replicas, 3+1 zones, two regions
- Protects against disk, node, rack, zone, and region failures
- Synchronous writes to all three zones and asynchronous copy to secondary



RA-GZRS

- GZRS + read access to secondary
- Separate secondary endpoint
- RPO delay to secondary can be queried

Access Storage

Every object has a unique URL address – based on account name and storage type

Container service: https://mystorageaccount.blob.core.windows.net

Table service: https://mystorageaccount.table.core.windows.net

Queue service: https://mystorageaccount.queue.core.windows.net

File service: https://mystorageaccount.file.core.windows.net

lower Letters + numbers

Custom name

- If you prefer you can configure a custom domain name

CNAME record	Target
blobs.contoso.com	contosoblobs.blob.core.windows.net




Tenant
... onmicrosoft.com
az.training
Custom name

Secure Storage Endpoints

Firewalls and Virtual Networks restrict access to the Storage Account from specific Subnets on Virtual Networks or public IP's

Subnets and Virtual Networks must exist in the same Azure Region or Region Pair as the Storage Account


Firewalls and virtual networks Custom domain *private EP*


 Save  Discard  Refresh

Public network access

☒ Enabled from all networks

☐ Enabled from selected virtual networks and IP addresses

 ☐ Disabled


 All networks, including the internet, can access this storage account. [Learn more](#)

Network Routing

Determine how you would like to route your traffic as it travels from its source to an Azure endpoint. Microsoft routing is recommended for most customers.

Routing preference ⓘ

☒ Microsoft network routing ☐ Internet routing

 The current combination of storage account kind, performance, replication, and location does not support network routing.

Learning Recap – Configure Storage Accounts



**Check your
knowledge
questions and
additional
study**

- Create an Azure Storage account
- Provide disaster recovery by replicating storage data across regions and failing over to a secondary location

Configure Blob Storage



AWS: S3 Bucket

Implement Blob Storage

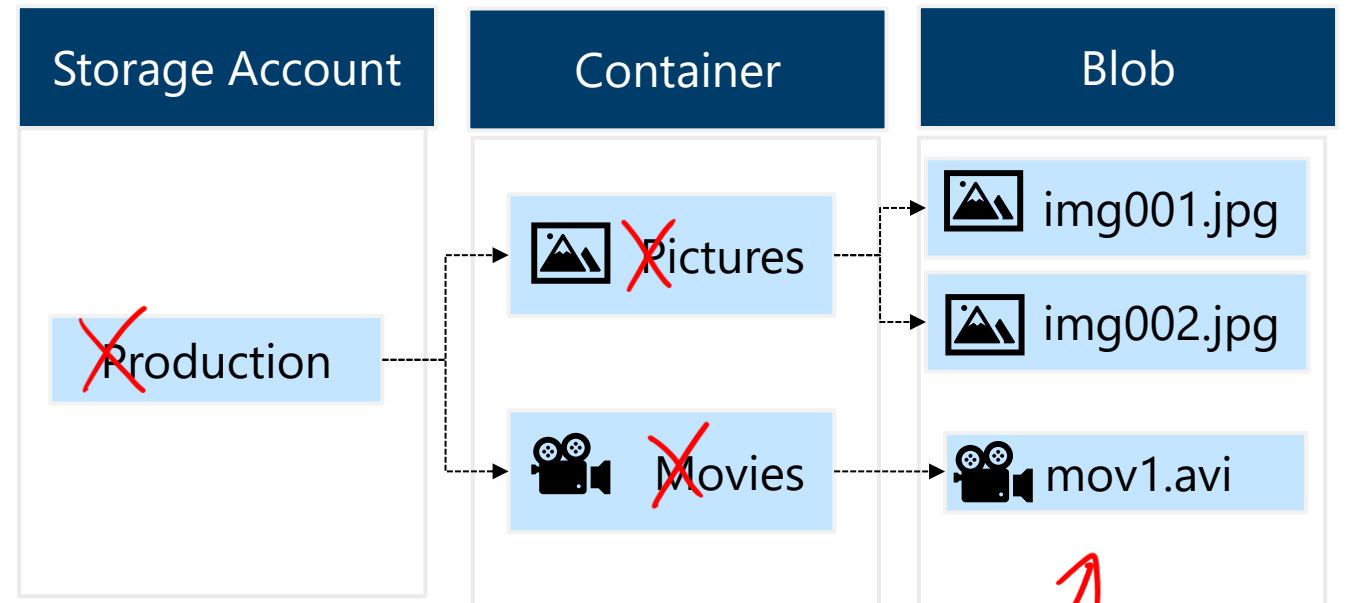
Stores unstructured data in the cloud

Can store any type of text or binary data

Also referred to as *object storage*

Common uses:

- Serving images or documents directly to a browser
- Storing files for distributed access
- Streaming video and audio
- Storing data for backup and restore, disaster recovery, archiving
- Storing data for analysis by an on-premises or Azure-hosted service



upload

Create Blob Containers

All blobs must be in a container

Accounts have unlimited containers

Containers can have unlimited blobs

Restrict access using the public access level

[+ Container](#) [🔒 Change access level](#) [🔄 Refresh](#) | [🗑️ Delete](#)

New container

Name ^{*}

container01 ✓

Public access level ⓘ

Private (no anonymous access) ^

Private (no anonymous access)

Blob (anonymous read access for blobs only)

Container (anonymous read access for containers and blobs)

OK

Cancel

Create Blob Access Tiers

HDD

* Default

Hot tier – Data that is accessed or modified frequently

30 Tage

Cool tier – Data that is infrequently accessed or modified and stored for at least 30 days

Cold tier – Data that is infrequently accessed or modified and stored for at least 90 days

Archive – Data that can tolerate several hours of retrieval latency and will remain in the Archive tier for at least 180 days

Tapes?

12h

Change tier

infoicon.jpg

Optimize storage costs by placing your data in the appropriate access tier. [Learn more](#)

Access tier

Hot (Inferred)

Hot (Inferred)

Cool

Cold

Archive

Add Blob Lifecycle Management Rules

Transitioning of blobs to a cooler storage tier to optimize for performance and cost

Delete blobs at the end of their lifecycle

Apply rules to filtered paths in the Storage Account

Add a rule ...

✓ Details 2 Base blobs 3 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) *

Enter a value

Then

Delete the blob

Move to cool storage

For infrequently accessed data that you want to keep on cool storage for at least 30 days.

Move to cold storage

For rarely accessed data that you want to keep for at least 90 days.

Move to archive storage

Use if you don't need online access and want to keep the object for 180 days or longer.

Delete the blob

Deletes the object per the specified conditions.

Determine Blob Object Replication

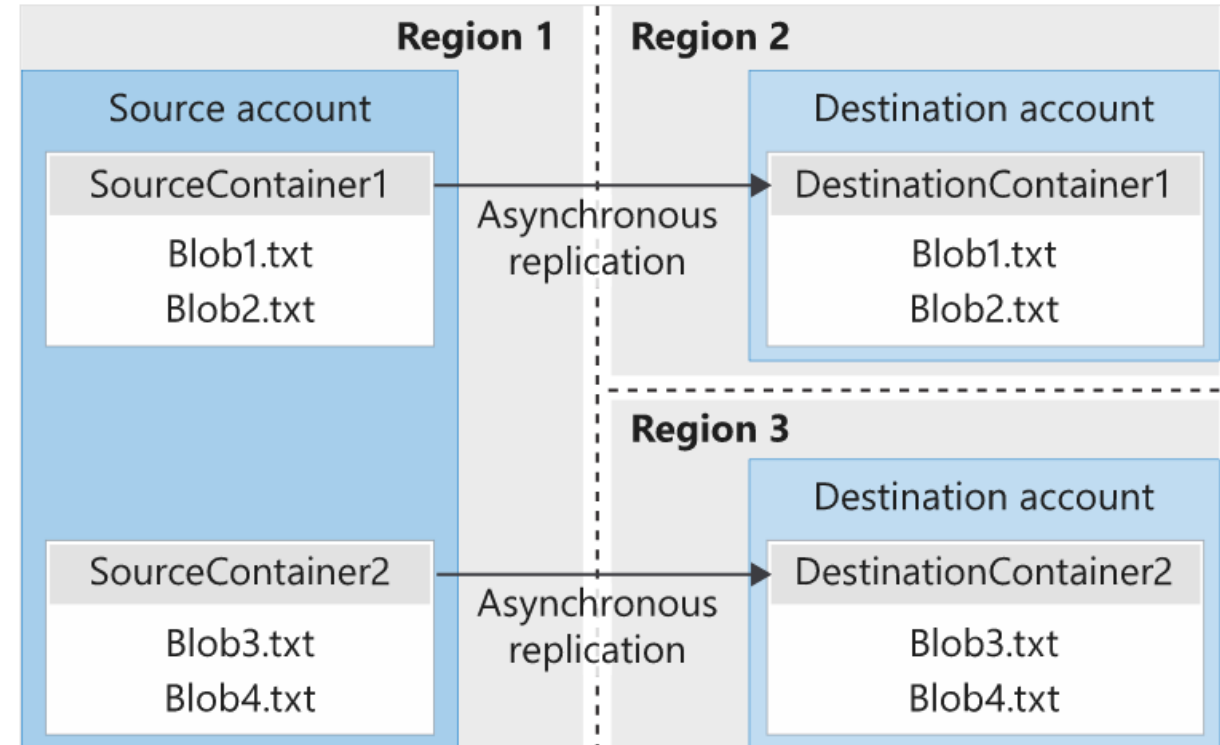
Asynchronous to any other Region

Minimizes latency for read requests

Increases efficiency for compute workloads

Optimizes data distribution

Optimizes costs

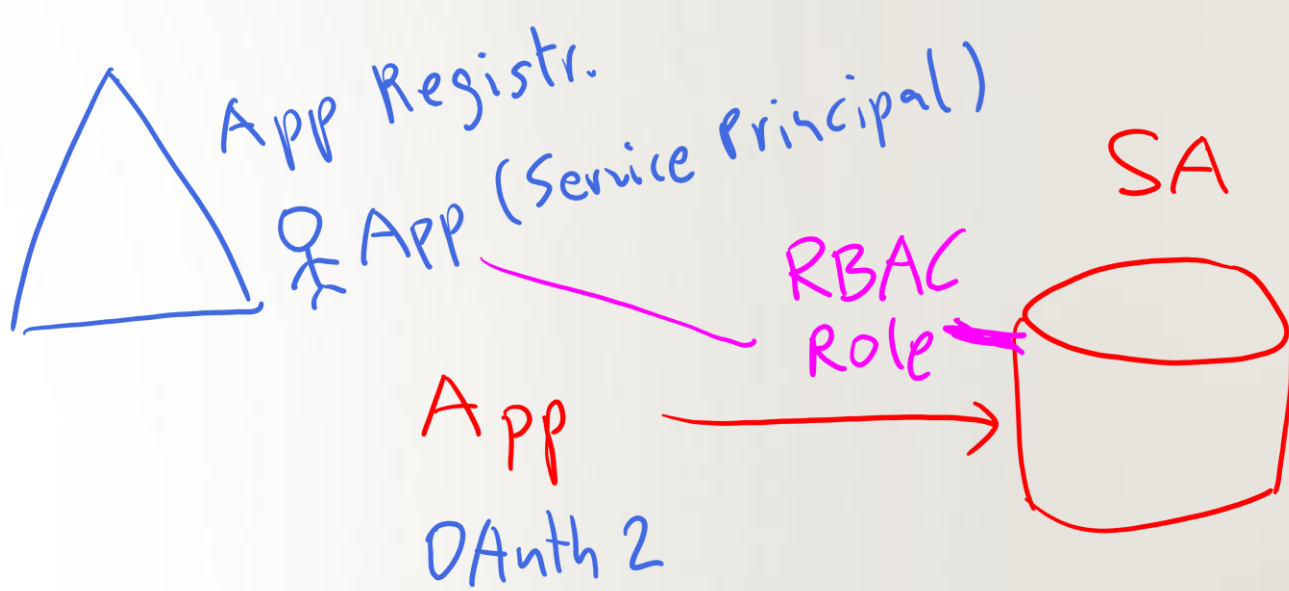


Learning Recap - Configure Blob Storage



**Check your
knowledge
questions and
additional
study**

- Optimize storage performance and costs using Azure Blob storage tiers
- Gather metrics from your Azure Blob Storage containers



RBAC
Role
Actions[]
Data Actions[*]

Configure Storage Security

Review Storage Security Strategies



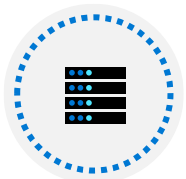
Storage Service Encryption



Authentication with Entra ID and RBAC ✓



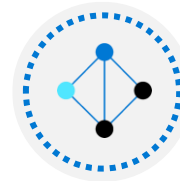
Client-side encryption, HTTPS, and SMB 3.0 for data in transit



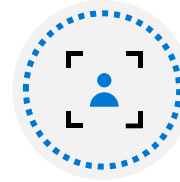
Azure disk encryption



Shared Access Signatures – delegated access



Shared Key – encrypted signature string



Anonymous access to containers and blobs

SA key 1 key 2
(connection string)

SAS

Read von bis

Sig

verschl. key 2

Create Shared Access Signatures

Provides delegated access to resources

Grants access to clients without sharing your storage account keys


The account SAS delegates access to resources in one or more of the storage services


The service SAS delegates access to a resource in just one of the storage services

Signing method ⓘ
☒ Account key ☐ User delegation key

Signing key ⓘ
Key 1 ▼

Permissions * ⓘ
Read ▼

Start and expiry date/time ⓘ
Start
02/01/2021 
(UTC-08:00) Coordinated Universal Time-08 ▼

Expiry
02/02/2021 
(UTC-08:00) Coordinated Universal Time-08 ▼

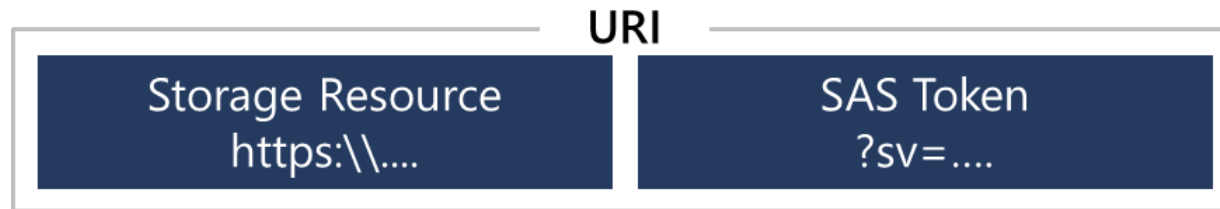
Allowed IP addresses ⓘ
for example, 168.1.5.65 or 168.1.5.65-168.1....

Allowed protocols ⓘ
☒ HTTPS ☐ HTTP

Generate SAS token and URL

Identify URI and SAS Parameters

- A SAS is a signed URI that points to one or more storage resources
- Consists of a storage resource URI and the SAS token



<https://myaccount.blob.core.windows.net/?sp=r&st=2020-05-11T18:31:43Z&se=2020-05-12T02:31:43Z&spr=https&sv=2019-10-10&sr=b&sig=j0qABJZHfUVEBQ3yVn7kWiCKl00sxCiK1rzEchfAz8U%3D>

Includes parameters for resource URI, storage services version, services, resource types, start time, expiry time, resource, permissions, IP range, protocol, signature

Determine Storage Service Encryption

You can use your own key (next topic)

Protects your data for security and compliance

Automatically encrypts and decrypts your data

Encrypted through 256-bit AES encryption

Is enabled for all new and existing storage accounts and cannot be disabled

Is transparent to users

Encryption

 Save  Discard

Storage service encryption protects your data at rest. Azure Storage encrypts your data as it's written in our datacenters, and automatically decrypts it for you as you access it.

By default, data in the storage account is encrypted using Microsoft Managed Keys. You may choose to bring your own key.

Please note that after enabling Storage Service Encryption, only new data will be encrypted, and any existing files in this storage account will retroactively get encrypted by a background encryption process.

[Learn More about Azure Storage Encryption](#) 

Encryption type

- ☒ Microsoft Managed Keys
- ☐ Customer Managed Keys

Create Customer Managed Keys

Use the Azure Key Vault to manage your encryption keys


Create your own encryption keys and store them in a key vault

Use Azure Key Vault's APIs to generate encryption keys

Custom keys give you more flexibility and control

Encryption type

- ☐ Microsoft Managed Keys
- ☒ Customer Managed Keys

i The storage account named 'storage987123' will be granted access to the selected key vault. Both soft delete and purge protection will be enabled on the key vault and cannot be disabled. [Learn more about customer managed keys](#) 

Encryption key

- ☐ Enter key URI
- ☒ Select from Key vault

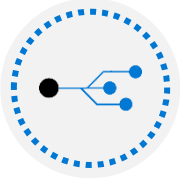
Key vault and key *

Key vault: keyvault987123

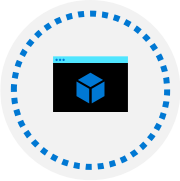
Key: storagekey

[Select a key vault and key](#)

Apply Storage Security Best Practices



Always use HTTPS to create or distribute a SAS



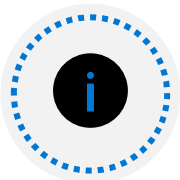
Reference stored access policies where possible



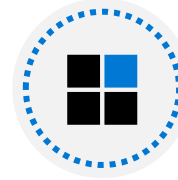
Use near-term expiration times on an ad hoc SAS



Use Storage Analytics to monitor your application



Be careful with SAS start time



Be specific with the resource to be accessed



Understand that your account will be billed for any usage



Validate data written using SAS



Don't assume SAS is always the correct choice

Learning Recap - Configure Storage Security



Check your
knowledge
questions and
additional
study

- Secure your Azure Storage account
- Control access to Azure Storage with shared access signatures

Configure Azure Files and File Sync



net use * ||

New-PS Drive

Compare storage for file shares and blob data

Feature	Description	When to use
Azure Files	<p>SMB interface, client libraries, and a REST interface that allows access from anywhere to stored files</p> <p>\\download69118\videos\ UNC</p>	<ul style="list-style-type: none">• <u>Lift and shift</u> an application to the cloud• Store shared data across multiple virtual machines• Store development and debugging tools that need to be accessed from many virtual machines
Azure Blobs	<p>Client libraries and a REST interface that allows unstructured data (flat namespace) to be stored and accessed at a massive scale in block blobs</p>	<ul style="list-style-type: none">• Support streaming and random-access scenarios• Access application data from anywhere

Manage File Shares

File share quotas

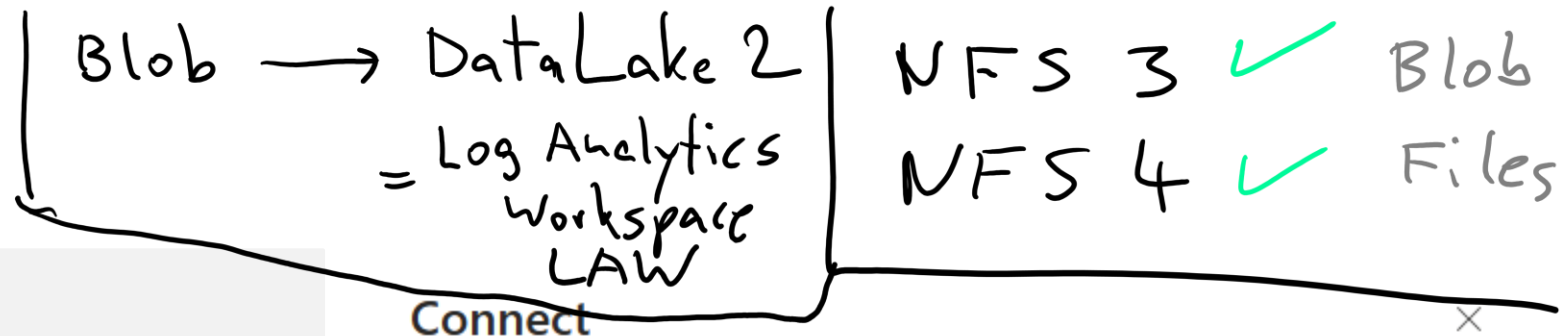
Windows – ensure port 445 is open

New-PS Drive

Linux – mount the drive

MacOS – mount the drive

Secure transfer required – SMB 3.0 encryption



Connect

publicwebsite

Windows

Linux

macOS

To connect to this Azure file share from Windows, choose from the following authentication methods and run the PowerShell commands from a normal (not elevated) PowerShell terminal:

Drive letter

Z

Authentication method



Active Directory or Microsoft Entra



Storage account key






Connecting to a share using the storage account key is only appropriate for admin access. Mounting the Azure file share with the Active Directory or Microsoft Entra identity of the user is preferred. [Learn more](#)

Show Script

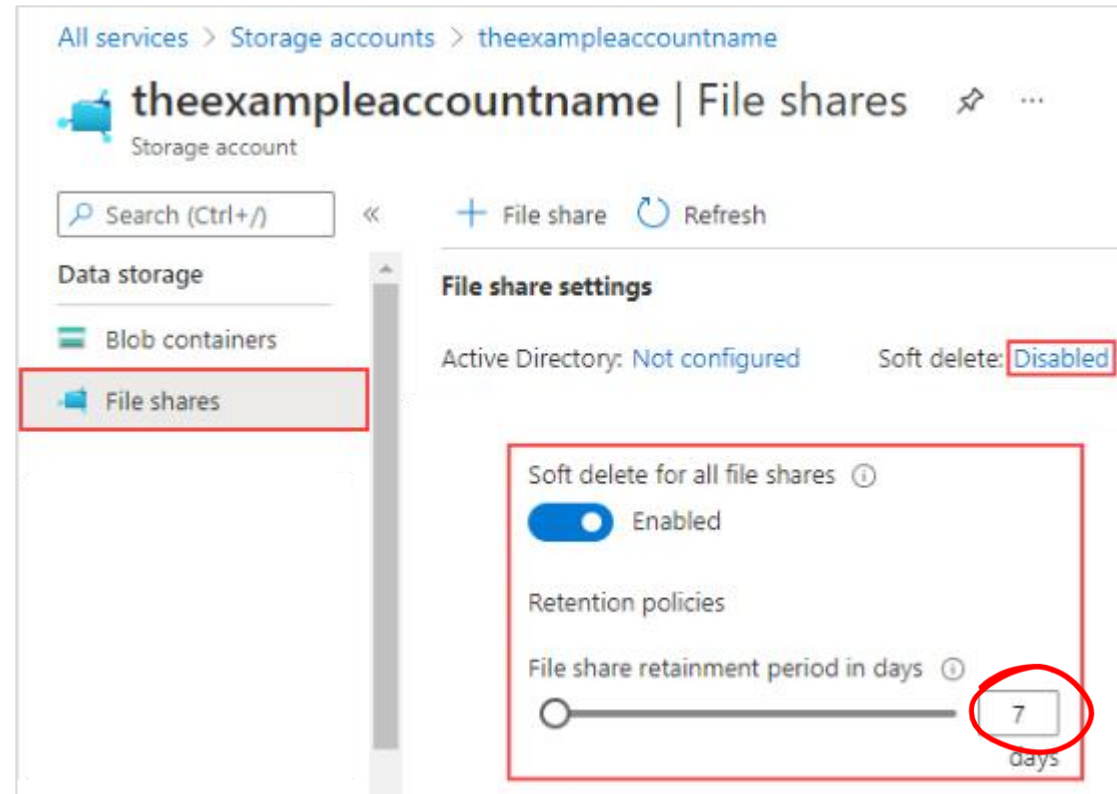
Create File Share Snapshots

- Protection against application error and data corruption
- Protection against accidental deletions or unintended changes
- Support backup and recovery
- Incremental snapshot that captures the share state at a point in time
- Snapshot at the *file share level*, and restore at the *file level*
- Is read-only copy of your data

 Add snapshot  Refresh  Delete		
Name	Date created	Initiator
<input type="checkbox"/> 2020-03-12T00:58:38.0000000Z	3/11/2020, 8:58:38 PM	-

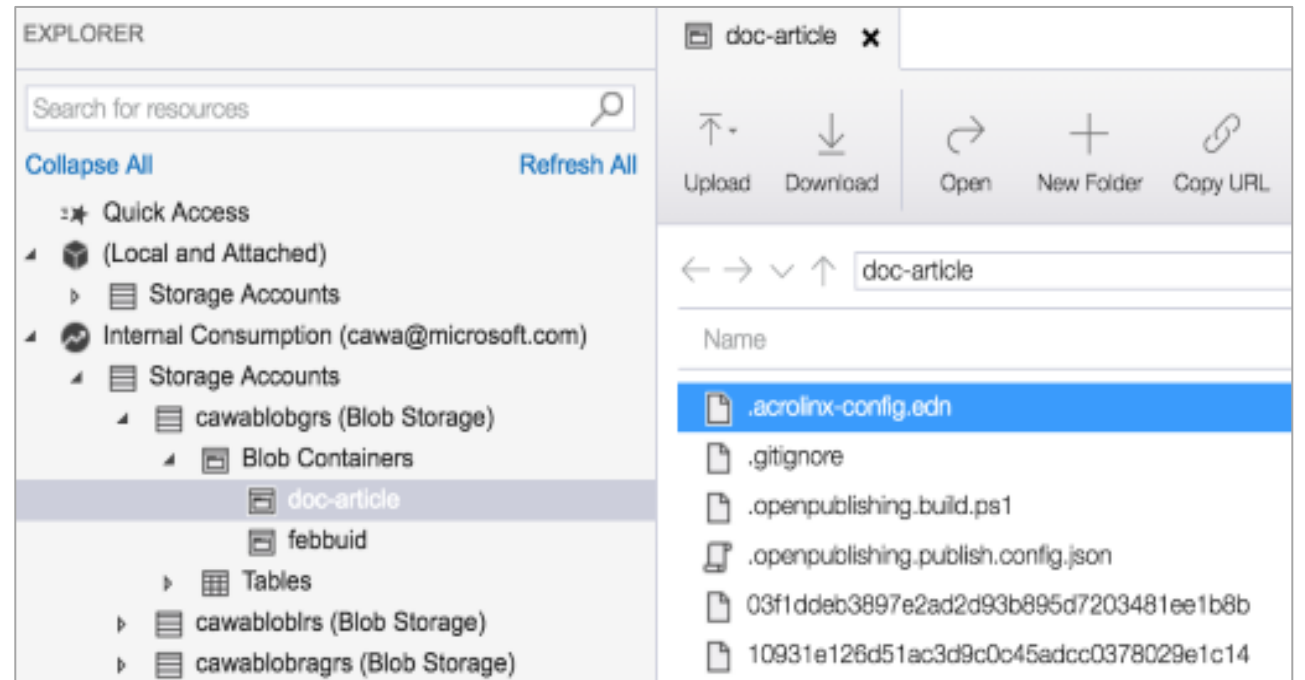
Implement soft delete for Azure Files

- Recovery from accidental data loss
 - Major change or upgrade scenarios
 - Business continuity – ransomware situations
 - Data compliance retention
-
- Enabled at the storage account level
 - Transitions content to a soft deleted state
 - Provides a retention period of 1 and 365 days
 - Works on new or existing file shares
 - Doesn't work for ~~NFS~~ shares



Use Azure Storage Explorer

- Download and install
- Access multiple accounts and subscriptions
- Create, delete, view, edit storage resources
- View and edit Blob, Queue, Table, File, Cosmos DB storage and Data Lake Storage
- Obtain shared access signature (SAS) keys
- Available for Windows, Mac, and Linux



Also consider portal-based Azure Storage Browser and Azure Storage Mover

Learning Recap - Configure Azure Files and File Sync



**Check your
knowledge
questions and
additional
study**

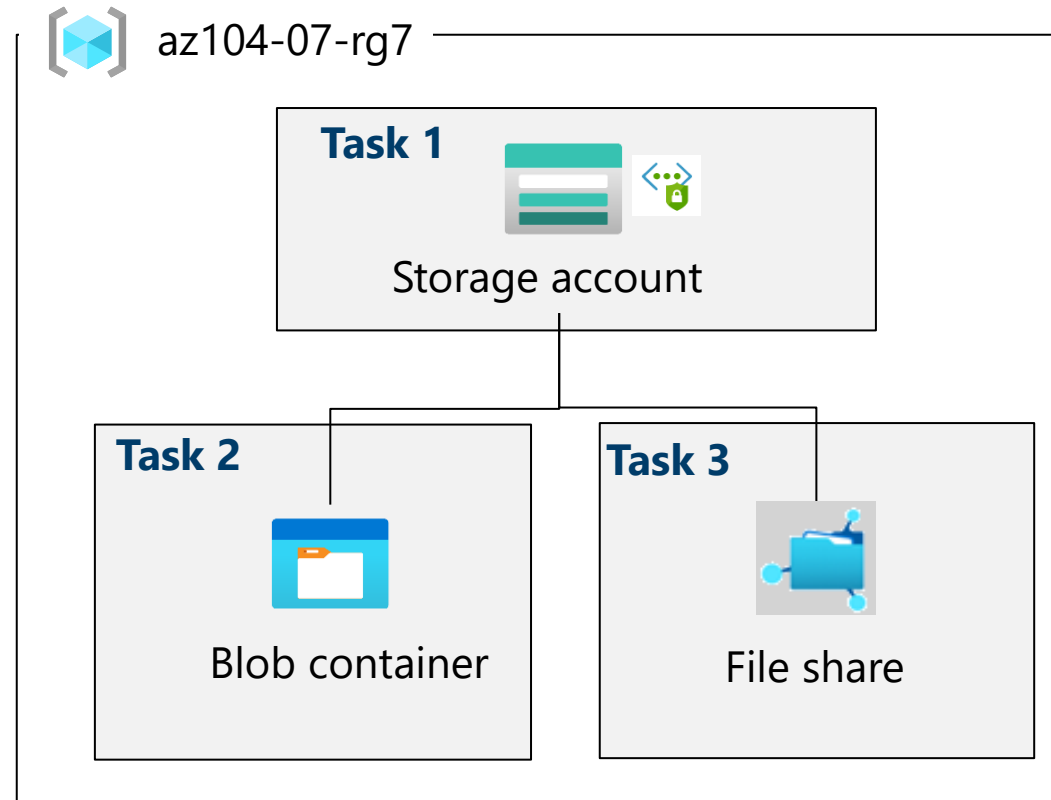
Reference Learn modules

- Configure Azure Files and Azure File Sync
- Implement a hybrid file server infrastructure
- Upload, download, and manage data with Azure Storage Explorer
- Copy and move blobs from one container or storage account to another using the AzCopy command

Lab – Manage Azure Storage



Lab 07 – Architecture diagram



End of presentation

