



## Modul 4: Storage Management Fundamentals

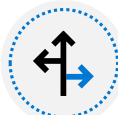
# Lesson 1: Volumes and file systems in Windows Server



# Volumes and file systems in Windows Server



Overview of file systems in Windows Server



Why use ReFS in Windows Server?



Overview of disk volumes



Overview of File Server Resource Manager



Manage permissions on volumes



# Overview of file systems in Windows Server

When selecting a file system, consider the differences between FAT, NTFS file system, and ReFS:

- FAT provides:
  - Basic file system
  - Partition size limitations
  - FAT32 to enable larger disks
  - exFAT developed for flash drives
- NTFS provides:
  - Metadata
  - Auditing and journaling
  - Security (ACLs and encryption)



# Overview of file systems in Windows Server

- ReFS provides:
  - Backward compatibility support for NTFS
  - Enhanced data verification and error correction
  - Support for larger files, directories, and volumes)



# Why use ReFS in Windows Server?

ReFS has many advantages over NTFS:

- Metadata integrity with checksums
- Integrity streams with user data integrity
- Allocation on write transactional model
- Large volume, file, and directory sizes ( $2^{78}$  bytes with 16 KB cluster size)
- Storage pooling and virtualization
- Data striping for performance and redundancy
- Disk scrubbing for protection against latent disk errors
- Resiliency to corruptions with recovery
- Shared storage pools across machines



# Overview of disk volumes

When selecting a type of disk for use in Windows Server, you can choose between:

- Basic disk
- Dynamic disk

Regardless of which type of disk you use, you must configure the following volumes on one of the server's hard disks:

- System volumes
- Boot volumes



# Overview of disk volumes

In Windows Server, if you are using dynamic disks, you can create a number of different types of disk volumes:

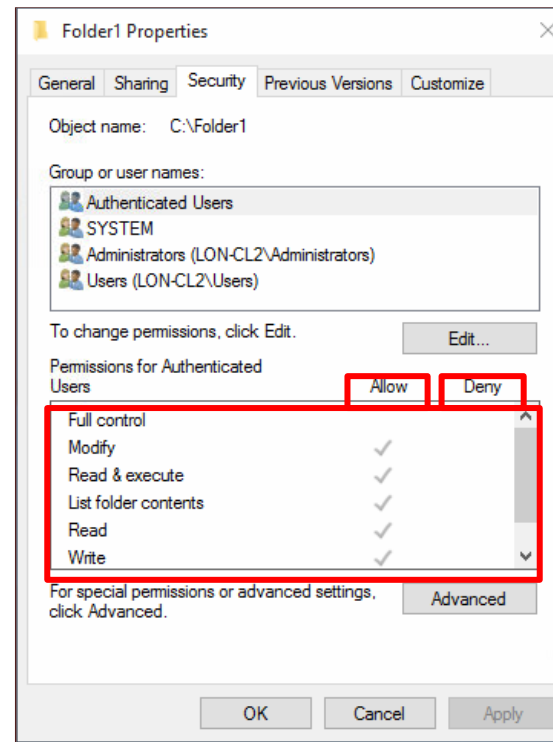
- Simple volumes
- Spanned volumes
- Striped volumes
- Mirrored volumes
- RAID-5 volumes



# Manage permissions on volumes

Control access on NTFS and ReFS file system:

- Can be added for groups, users, and computers:
  - Cumulative for group members
- Can be assigned to:
  - Files
  - Folders
  - Volumes (root folder)
- Permissions:
  - Allow
  - Deny (takes precedence)
- Basic and advanced



# Lesson 2: Implementing sharing in Windows Server



# Implementing sharing in Windows Server



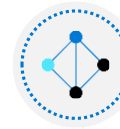
What is SMB?



Configure SMB shares



Best practices for sharing resources



Overview of NFS

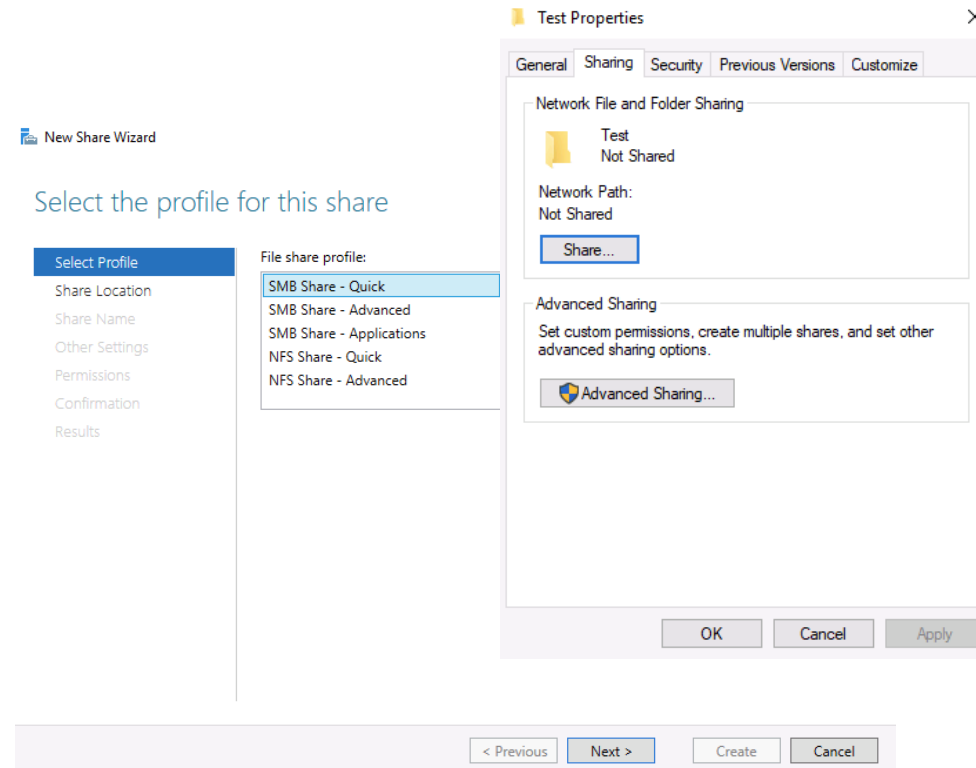


# What is SMB?

- SMB is the file-sharing protocol that Windows client and server operating systems use
- Each new version has additional features
- SMB 3.0 introduced large performance benefits
- SMB 3.0.2 added:
  - Scale-Out File Server
  - Removable SMB 1.x
- SMB 3.1.1 added:
  - Pre-authentication integrity
  - SMB encryption improvements
  - Cluster dialect fencing

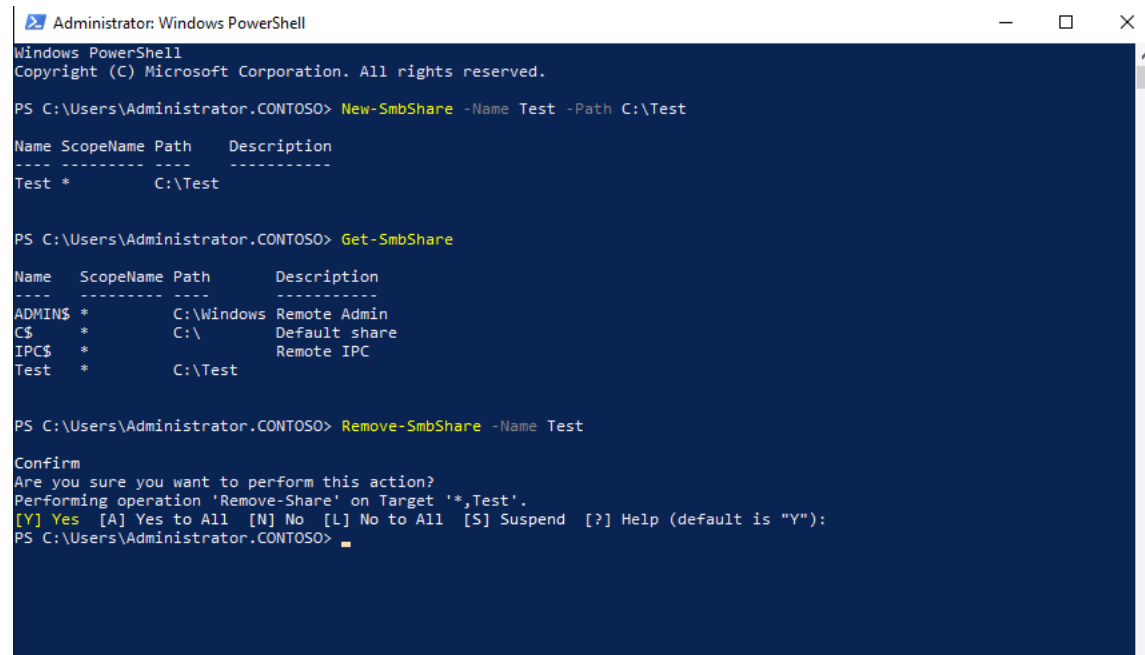
# Configure SMB shares

- File Manager
  - Network File and Folder Sharing
  - Advanced Sharing
- Server Manager
  - Quick
  - Advanced
  - Applications



# Configure SMB shares

- Windows PowerShell cmdlets for SMB share management:
  - New-SmbShare
  - Set-SmbShare
  - Remove-SmbShare
  - Get-SmbShare
  - Get-SmbSession
  - Get-SmbOpenFile
  - Set-SmbBandwidthLimit



```
Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Users\Administrator.CONTOSO> New-SmbShare -Name Test -Path C:\Test

Name ScopeName Path Description
----
Test * C:\Test

PS C:\Users\Administrator.CONTOSO> Get-SmbShare

Name ScopeName Path Description
----
ADMIN$ * C:\Windows Remote Admin
C$ * C:\ Default share
IPC$ * Remote IPC
Test * C:\Test

PS C:\Users\Administrator.CONTOSO> Remove-SmbShare -Name Test

Confirm
Are you sure you want to perform this action?
Performing operation 'Remove-Share' on Target '*,Test'.
[Y] Yes [A] Yes to All [N] No [L] No to All [S] Suspend [?] Help (default is "Y"):
PS C:\Users\Administrator.CONTOSO>
```



# Overview of NFS

- NFS is a file system based on open standards
- Current version is 4.1
- Windows NFS components include:
  - Client for NFS
  - Server for NFS
- Support for Kerberos v5 authentication
- Primary uses for NFS:
  - Storage for VMware virtual machines
  - Sharing data across multiple operating systems
  - Sharing data across different IT infrastructures after a company merger



# Understanding SMB 3.1.1 protocol security

- Understanding SMB 3.1.1 protocol security
- SMB 3.1.1 encryption requirements
- Configuring SMB encryption on SMB shares
- Disabling SMB 1.0
- Walkthrough: Disabling SMB 1.0, and configuring SMB encryption on shares





# SMB 3.1.1 encryption requirements

| Operating system                      | Windows 10<br>Windows Server 2016/19 | Windows 8.1<br>Windows Server 2012 R2 | Windows 8<br>Windows Server 2012 | Windows 7<br>Windows Server 2008 R2 | Windows Vista<br>Windows Server 2008 | Previous versions |
|---------------------------------------|--------------------------------------|---------------------------------------|----------------------------------|-------------------------------------|--------------------------------------|-------------------|
| Windows 10<br>Windows Server 2016/19  | SMB 3.1.1                            | SMB 3.02                              | SMB 3.0                          | SMB 2.1                             | SMB 2.0.2                            | SMB 1.x           |
| Windows 8.1<br>Windows Server 2012 R2 | SMB 3.02                             | SMB 3.02                              | SMB 3.0                          | SMB 2.1                             | SMB 2.0.2                            | SMB 1.x           |
| Windows 8<br>Windows Server 2012      | SMB 3.0                              | SMB 3.0                               | SMB 3.0                          | SMB 2.1                             | SMB 2.0.2                            | SMB 1.x           |
| Windows 7<br>Windows Server 2008 R2   | SMB 2.1                              | SMB 2.1                               | SMB 2.1                          | SMB 2.1                             | SMB 2.0.2                            | SMB 1.x           |



# Configuring SMB encryption on SMB shares

Use Windows PowerShell to enable encrypted SMB:

- For an existing file share:
  - **Set-SmbShare –Name <sharename> -EncryptData \$true**
- To encrypt all sharing on a file server:
  - **Set-SmbServerConfiguration –EncryptData \$true**
- To create a new SMB file share and enable SMB encryption simultaneously:
  - **New-SmbShare –Name <sharename> -Path <pathname> –EncryptData \$true**

# Disabling SMB 1.0

You can disable SMB 1.x support by using Windows PowerShell:

- For Windows 7, Windows Server 2008 R2, Windows Vista, and Windows Server 2008:
  - **Set-ItemProperty -Path "HKLM:\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters" SMB1 -Type DWORD -Value 0 –Force**
- For Windows 8/Windows Server 2012 or newer systems:
  - **Set-SmbServerConfiguration –EnableSMB1Protocol \$false**
- To uninstall SMB 1.x from Windows 8.1 and newer:
  - **Remove-WindowsFeature FS-SMB1**

# Lesson 3: Configure Storage Accounts



# Configure Storage Accounts Introduction



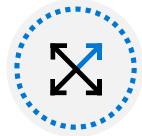
Implement Azure Storage



Explore Azure Storage Services



Determine Storage Account Kinds



Determine Replication Strategies



Access Storage



Secure Storage Endpoints

# Implement Azure Storage



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

Durable, secure, scalable,  
managed, accessible

Storage for virtual machines,  
unstructured data and  
structured data

Two tiers: Premium and  
Standard

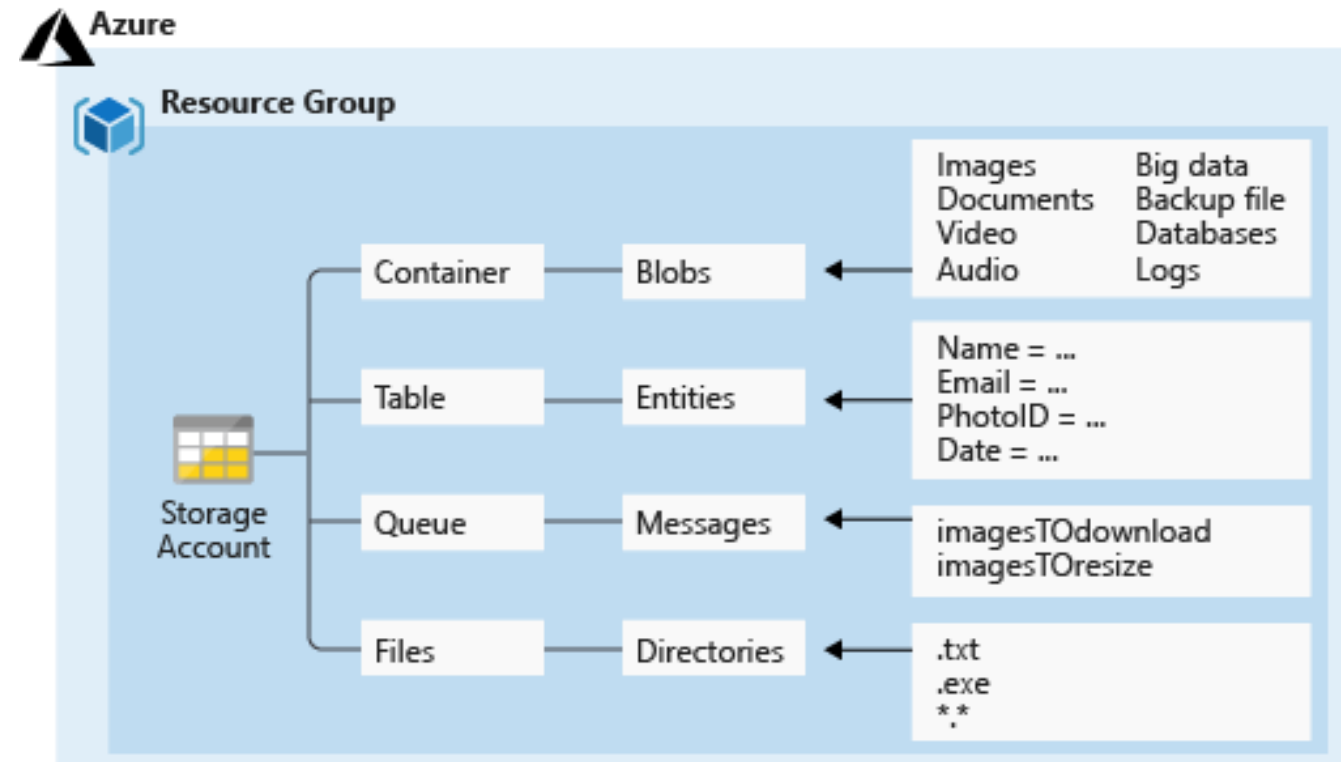
# Explore Azure Storage Services

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





# Determine Storage Account Kinds

| Storage Account             | Recommended usage   |
|-----------------------------|---|
| Standard general-purpose v2 | Most scenarios including Blob, File, Queue, Table, and Data Lake Storage.   |
| Premium block blobs         | 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.   |

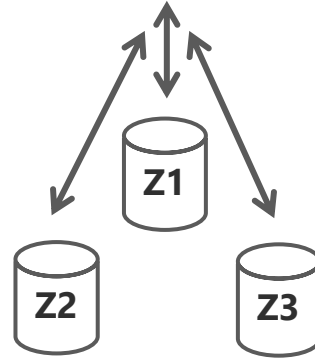
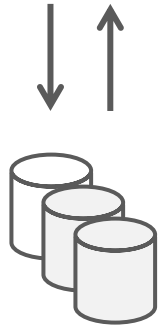


All storage accounts are encrypted using Storage Service Encryption (SSE) for data at rest

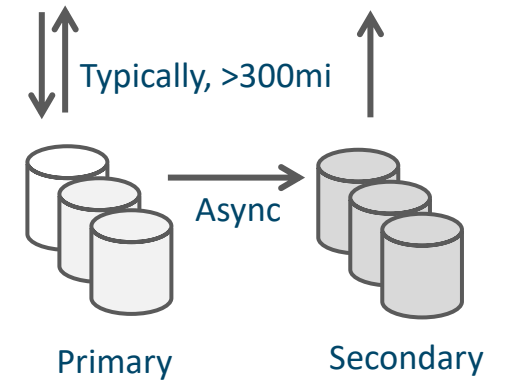
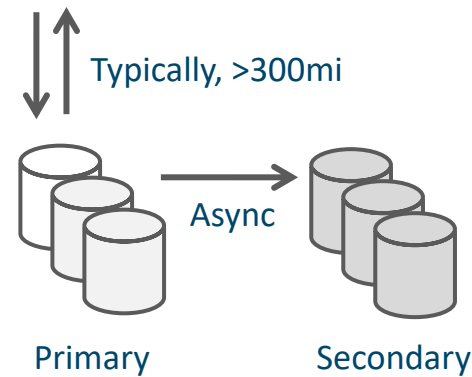


# Determine Replication Strategies (1 of 2)

## Single region



## Multiple regions



### 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

### GRS

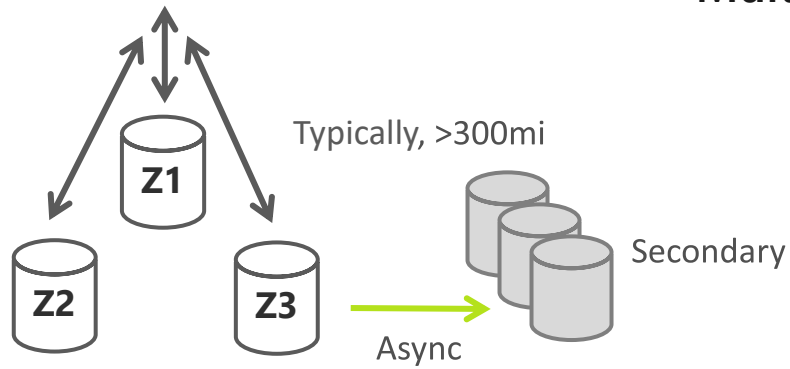
- Six replicas, two regions (three per region)
- Protects against major regional disasters
- Asynchronous copy to secondary

### RA-GRS

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

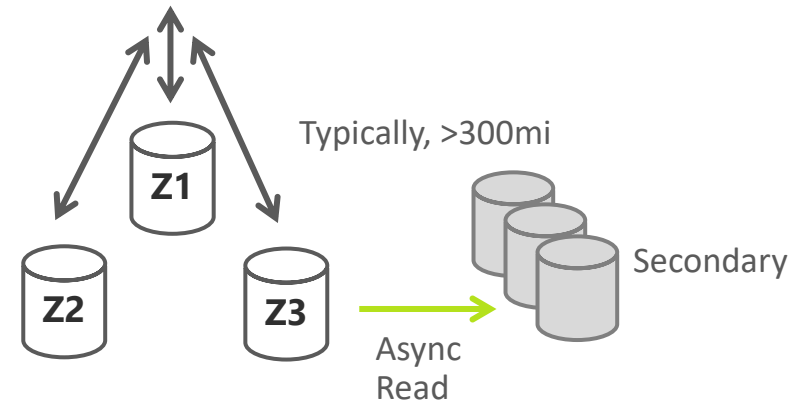
# Determine Replication Strategies (2 of 2)

## Multiple regions



### 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`

If you prefer you can configure a custom domain name

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

# Secure Storage Endpoints

storage987123 | Firewalls and virtual networks

Storage account

Search (Ctrl+ /)

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Data transfer

Events

Storage Explorer (preview)

Save Discard Refresh

Allow access from

All networks Selected networks

Configure network security for your storage accounts. [Learn more.](#)

Virtual networks

Secure your storage account with virtual networks. [+ Add existing virtual network](#) [+ Add new virtual network](#)

| Virtual Network | Subnet   | Address range | Endpoint Status | Resource Group |
|-----------------|----------|---------------|-----------------|----------------|
| ▼ vnet01        | 1        |               |                 | Demo           |
|                 | subnet01 | 10.1.0.0/24   | ✓ Enabled       | Demo           |

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

# Lesson 3: Configure Blob Storage



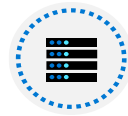
# Configure Blob Storage Introduction



Implement Blob Storage



Create Blob Containers



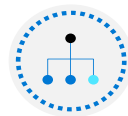
Create Blob Access Tiers



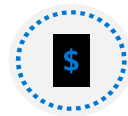
Add Blob Lifecycle Management Rules



Determine Blob Object Replication



Upload Blobs



Understand Storage Pricing



# Implement Blob Storage

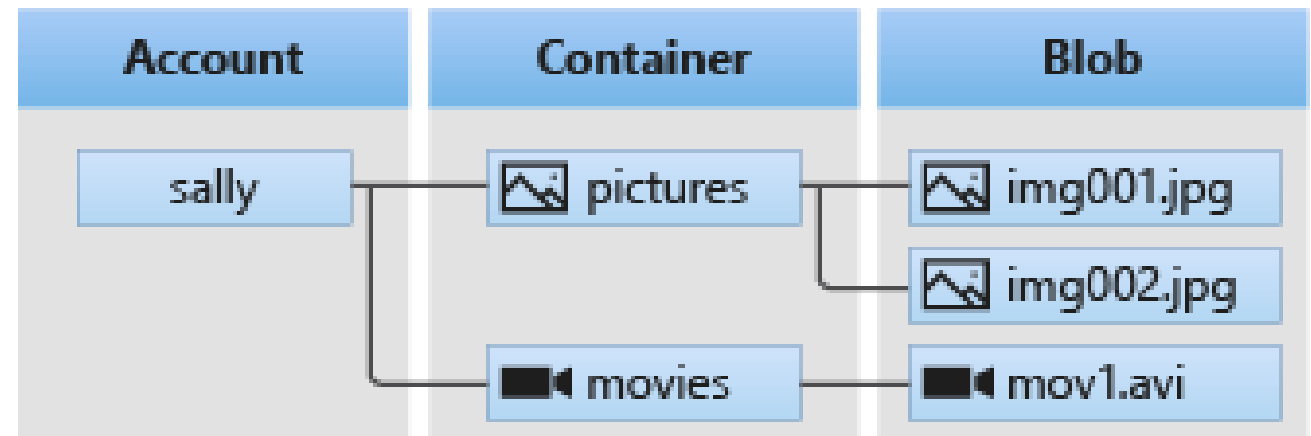
Stores unstructured data in the cloud

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



# Create Blob Containers

All blobs must be in a container

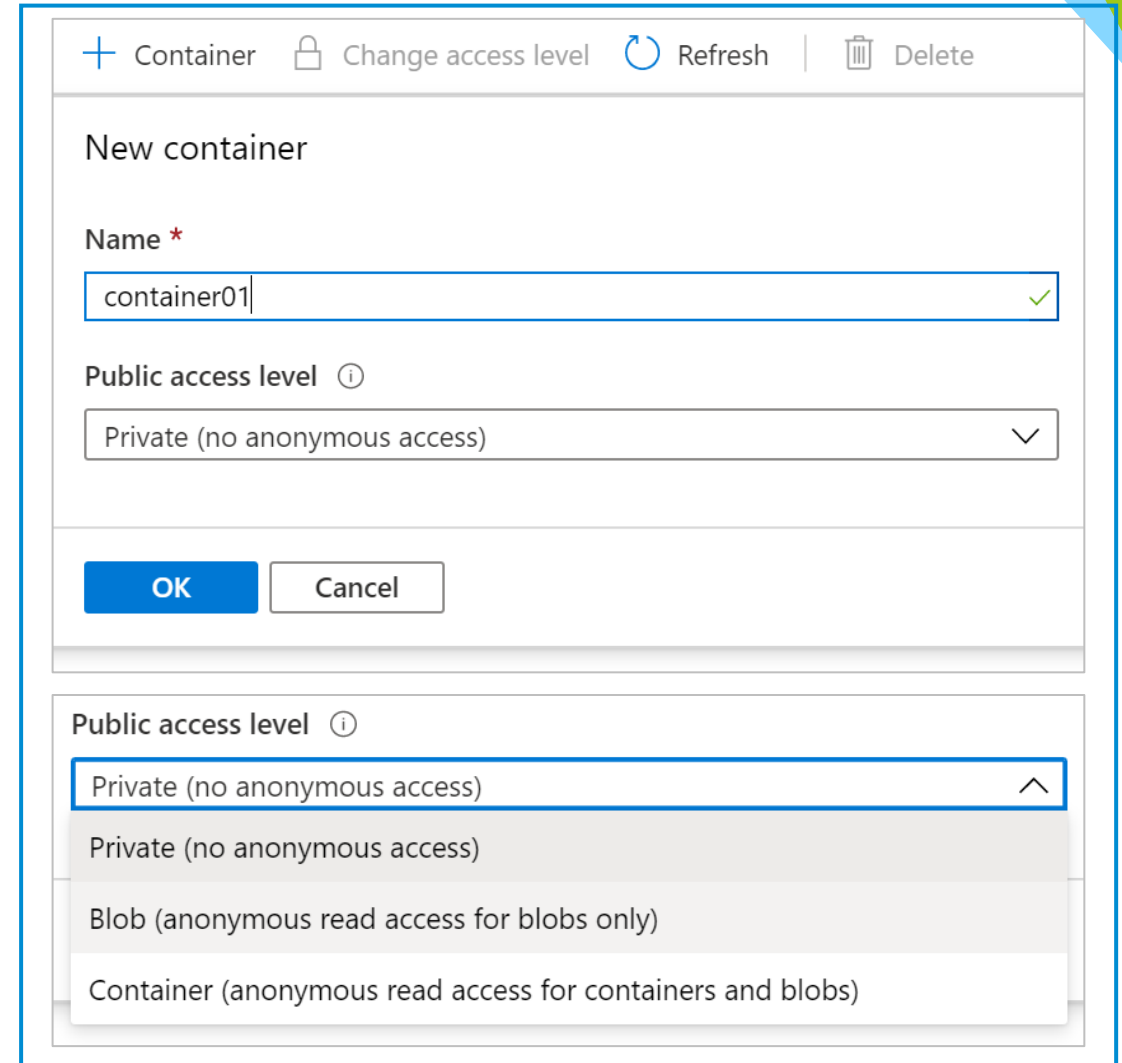
Accounts have unlimited containers

Containers can have unlimited blobs

**Private blobs** – no anonymous access

**Blob access** – anonymous public read access for blobs only

**Container access** – anonymous public read and list access to the entire container, including the blobs



Container Change access level Refresh Delete

New container

Name \*  
container01 ✓

Public access level ⓘ  
Private (no anonymous access) ▼

OK Cancel

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)



# Create Blob Access Tiers

**Hot tier** – Optimized for frequent access of objects in the storage account

**Cool tier** – Optimized for storing large amounts of data that is infrequently accessed and stored for at least 30 days

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

## Access Tier

Optimize storage costs by placing your data in the appropriate access tier.

Hot (Inferred)

Hot (Inferred)

Cool

Archive



You can switch between these access tiers at any time

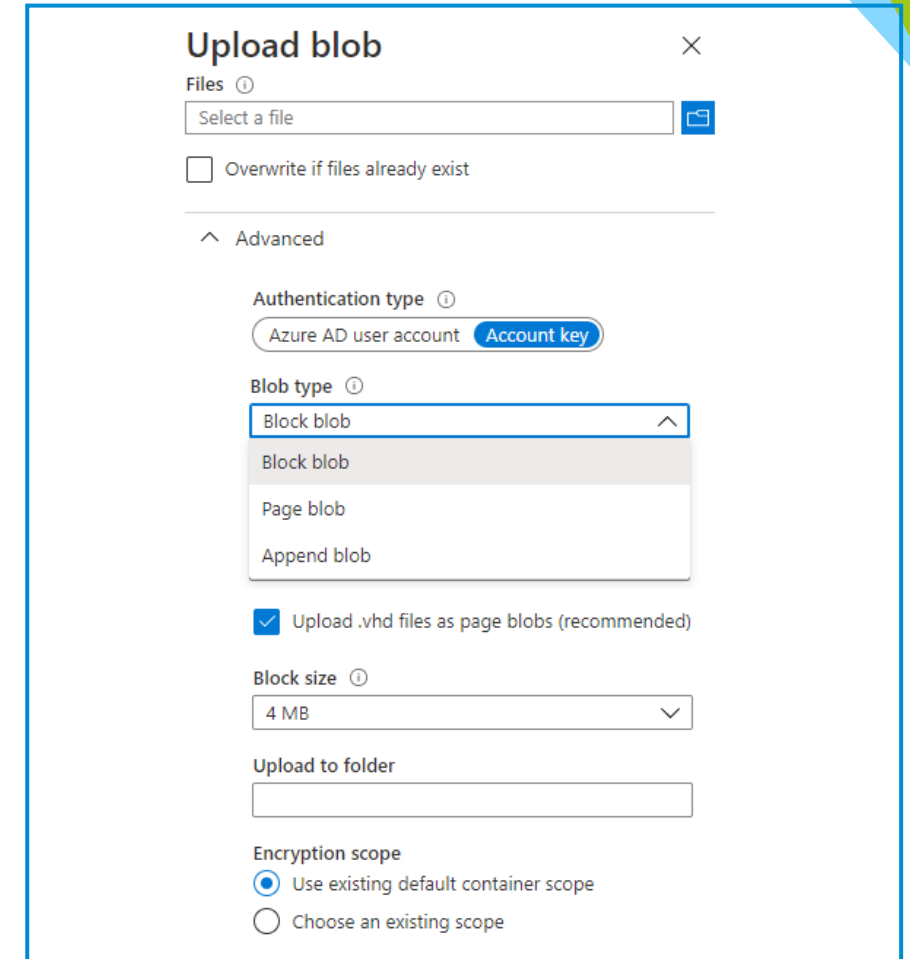
# Upload Blobs

Authentication type – Azure AD user account or Account key

Block blobs (default) – useful for storing text or binary files

Page blobs – more efficient for frequent read/write operations

Append blobs – useful for logging scenarios



The screenshot shows the 'Upload blob' dialog box with the following settings:

- Files:** A text input field with a file icon button.
- ☐ Overwrite if files already exist
- Advanced:**
  - Authentication type:** Two buttons: 'Azure AD user account' and 'Account key' (selected).
  - Blob type:** A dropdown menu showing 'Block blob' (selected), 'Block blob', 'Page blob', and 'Append blob'.
  - ☒ Upload .vhd files as page blobs (recommended)
  - Block size:** A dropdown menu showing '4 MB'.
  - Upload to folder:** An empty text input field.
  - Encryption scope:** Two radio buttons: 'Use existing default container scope' (selected) and 'Choose an existing scope'.

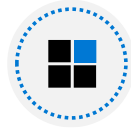


You cannot change a blob type once it has been created

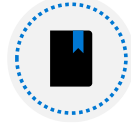
# Lesson 4: Configure Azure Files



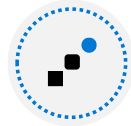
# Configure Azure Files



Compare Files to Blobs



Manage File Shares



Create File Share Snapshots

# Compare Files to Blobs

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

# Manage File Shares

File share quotas

Windows – ensure port 445 is open

Linux – mount the drive

MacOS – mount the drive

Secure transfer required – SMB 3.0 encryption

Windows Linux macOS

Drive letter

Z

To connect to this Azure file share from Windows, run these PowerShell commands from a normal (not elevated) PowerShell terminal:

```
$connectTestResult = Test-NetConnection -  
ComputerName storage987123.file.core.windows.net -  
Port 445  
if ($connectTestResult.TcpTestSucceeded) {  
    # Save the password so the drive will persist on reboot  
    cmd.exe /C "cmdkey  
/add:`"storage987123.file.core.windows.net`"
```

This script will check to see if this storage account is accessible via TCP port 445, which is the port SMB uses. If port 445 is available, your Azure file share will be persistently mounted. Your organization or internet service provider (ISP) may block port 445, however you may use Azure [Point-to-Site \(P2S\) VPN](#), Azure [Site-to-Site \(S2S\) VPN](#), or [ExpressRoute](#) to tunnel SMB traffic to your Azure file share over a different port.

# Create File Share Snapshots



 Add snapshot  Refresh  Delete

Name

Date created

Initiator



2020-03-12T00:58:38.0000000Z

3/11/2020, 8:58:38 PM -

Incremental snapshot that captures the share state at a point in time

Is read-only copy of your data

Snapshot at the file share level, and restore at the file level

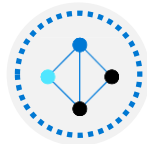
- Protection against application error and data corruption
- Protection against accidental deletions or unintended changes
- General backup purposes

# Lesson 4: Configure Azure Storage with Tools

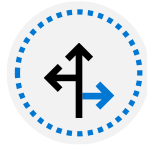




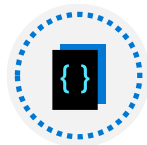
# Configure Azure Storage with Tools



Use Azure Storage Explorer



Use the Import and Export Service



Use AzCopy



# Use Storage Explorer

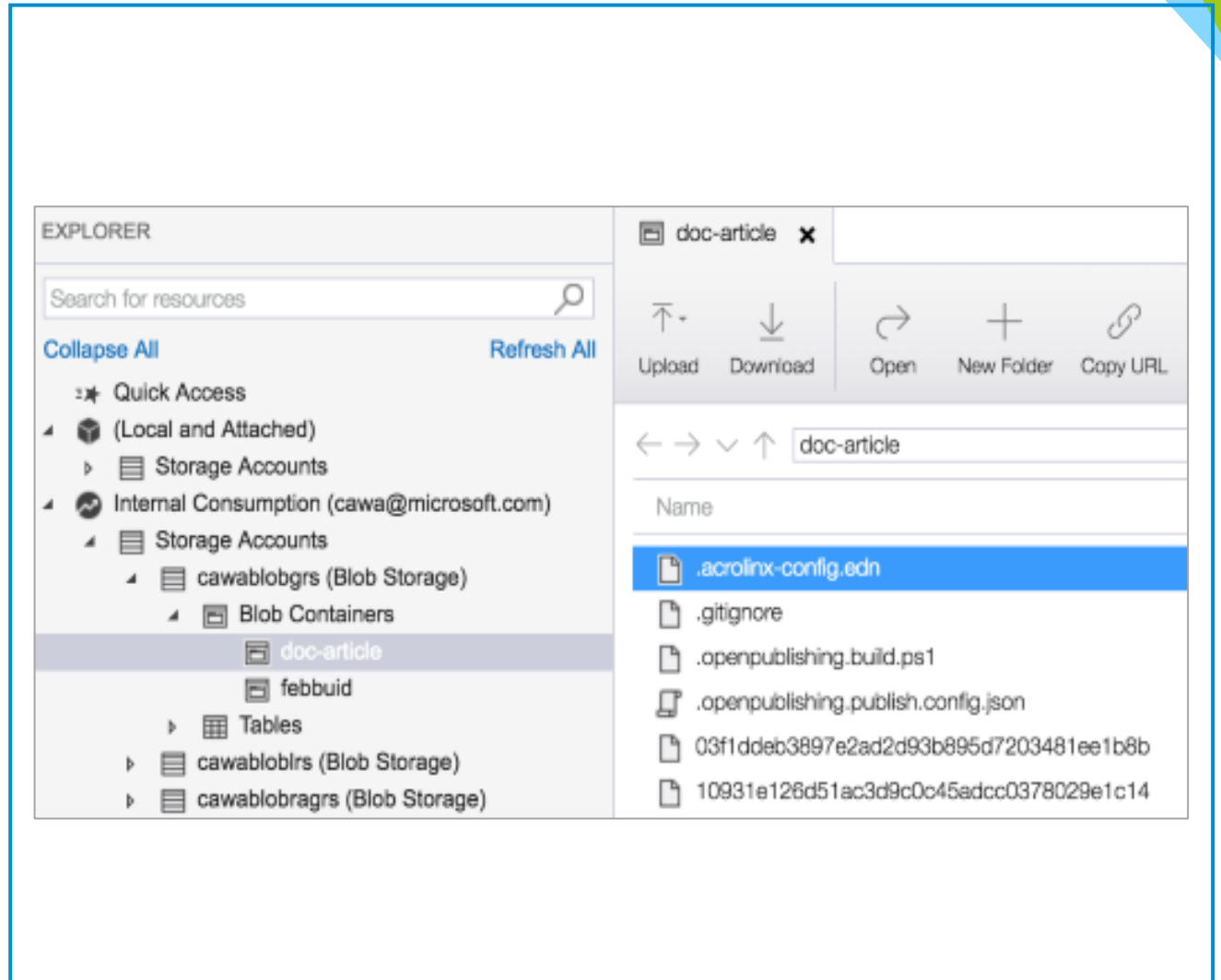
Access multiple accounts and subscriptions

Create, delete, view, edit storage resources

View and edit Blob, Queue, Table, File, Cosmos DB storage and Data Lake Storage

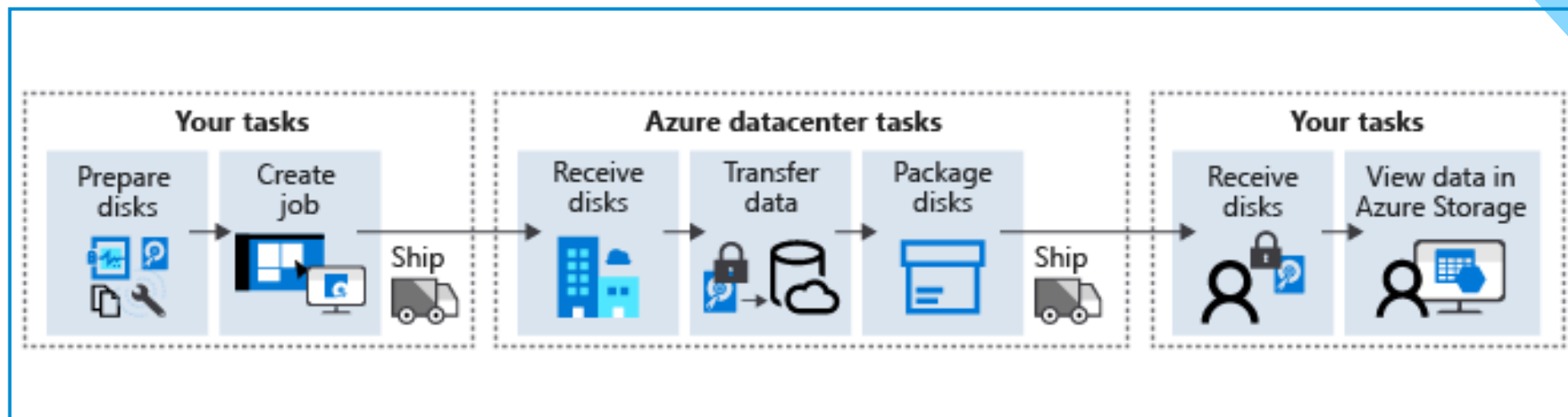
Obtain shared access signature (SAS) keys

Available for Windows, Mac, and Linux

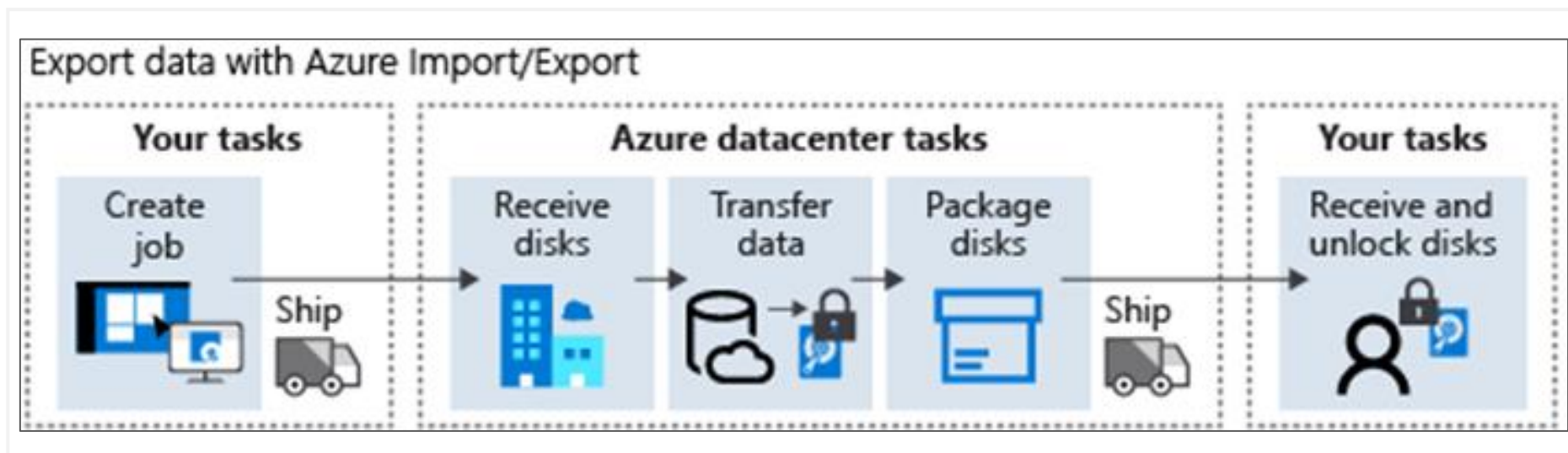


# Use the Import and Export Service

Import jobs move large amounts of data to Azure blob storage or files



Export jobs move large amounts of data from Azure blob storage (not files)



# Use AzCopy

```
azcopy copy [source] [destination] [flags]
```

Command line utility

Designed for copying data to and from Azure Blob, File, and Table storage

Available on Windows, Linux, and MacOS

Authentication options include Active Directory or SAS token

The End