Microsoft

AZ-800 Administer Windows Server Hybrid Core Infrastructure

Agenda AZ-800

1 Deploy and manage identity infrastructure – Windows Server
2 Deploy and manage identity infrastructure – Hybrid

3 Administering Windows Server Hybrid Core Infrastructure – Windows Server
4 Administering Windows Server Hybrid Core Infrastructure – Hybrid

5 Manage virtualization and containers – Windows Server
6 Manage virtualization and containers – Hybrid

7 Implement and manage networking infrastructure – Windows Server
8 Implement and manage networking Infrastructure – Hybrid

9 Configure storage and file services – Windows Server
10 Configure storage and file services – Hybrid ←

Configure storage and file services

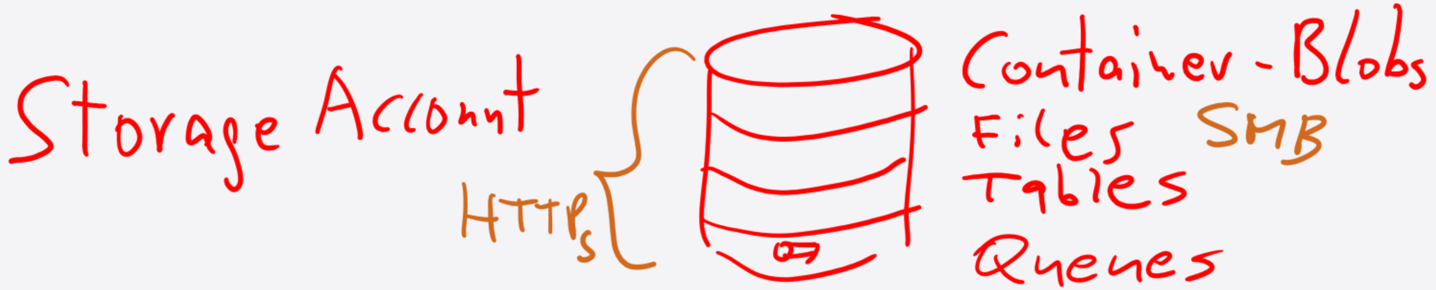
(Implementing a hybrid file server infrastructure)

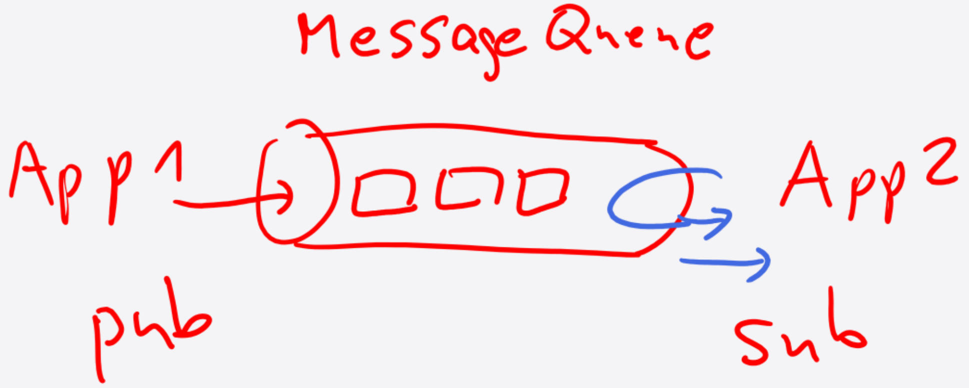
- Overview of Azure file services ✓
- Implementing Azure File Sync ✓
- Lab 10 – Implementing Azure File Sync

Overview of Azure file services

Learning Objectives – Overview of Azure file services

- Describe Azure storage services
- Configure Azure Files
- Configure connectivity to Azure Files
- Create Azure File share snapshots





Describe Azure storage services (1/2)

Four types of storage services:

- **Blobs** – Typically represent unstructured files and facilitate locking mechanisms, ensuring exclusive file access that IaaS VM disks require.
- **Tables** – Host nonrelational, semi-structured content, which consists of multiple rows of data.
- **Queues** – Offer temporary storage for messages that components of distributed applications use to asynchronously communicate with each other.
- **Files** – Provide storage for unstructured data.

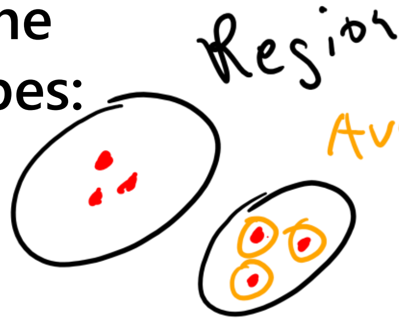
Azure Files provides the following functionality:

- Serverless deployment
- Almost unlimited storage
- Data redundancy
- Data encryption
- Access from anywhere
- Use of standard protocols
- Integration into an existing environment
- Granular file permissions
- Previous versions and backups
- Optional integration with on-premises file servers

Describe Azure storage services (2/2)

Deploy Azure Files by using the following storage account types:

- Locally redundant storage (LRS)
- Zone-redundant storage (ZRS)
- Geo-redundant storage (GRS)
- Geographically zone-redundant storage (GZRS)
- Read-access geographically-redundant storage (RA-GRS)

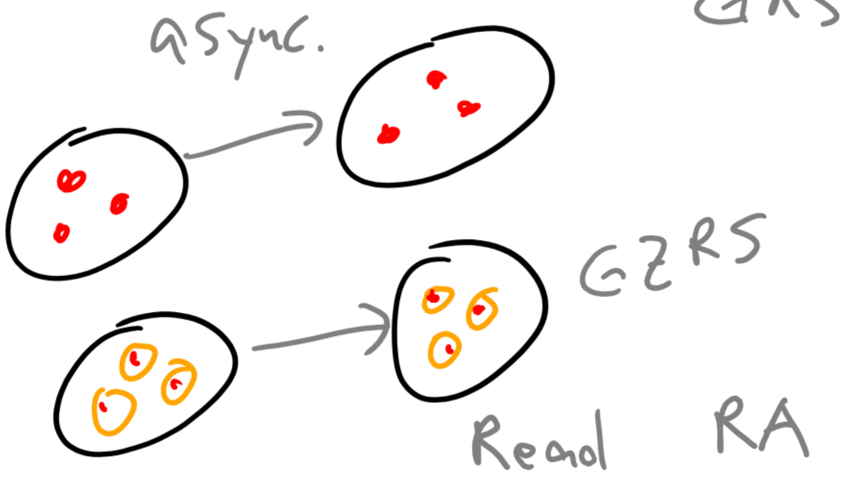


GZRS - RA

Azure Files supports two storage tiers: Premium, and standard

Common uses of Azure Files

- Replace or supplement on-premises file servers
- Lift and shift
- Backup and disaster recovery
- Azure File Sync

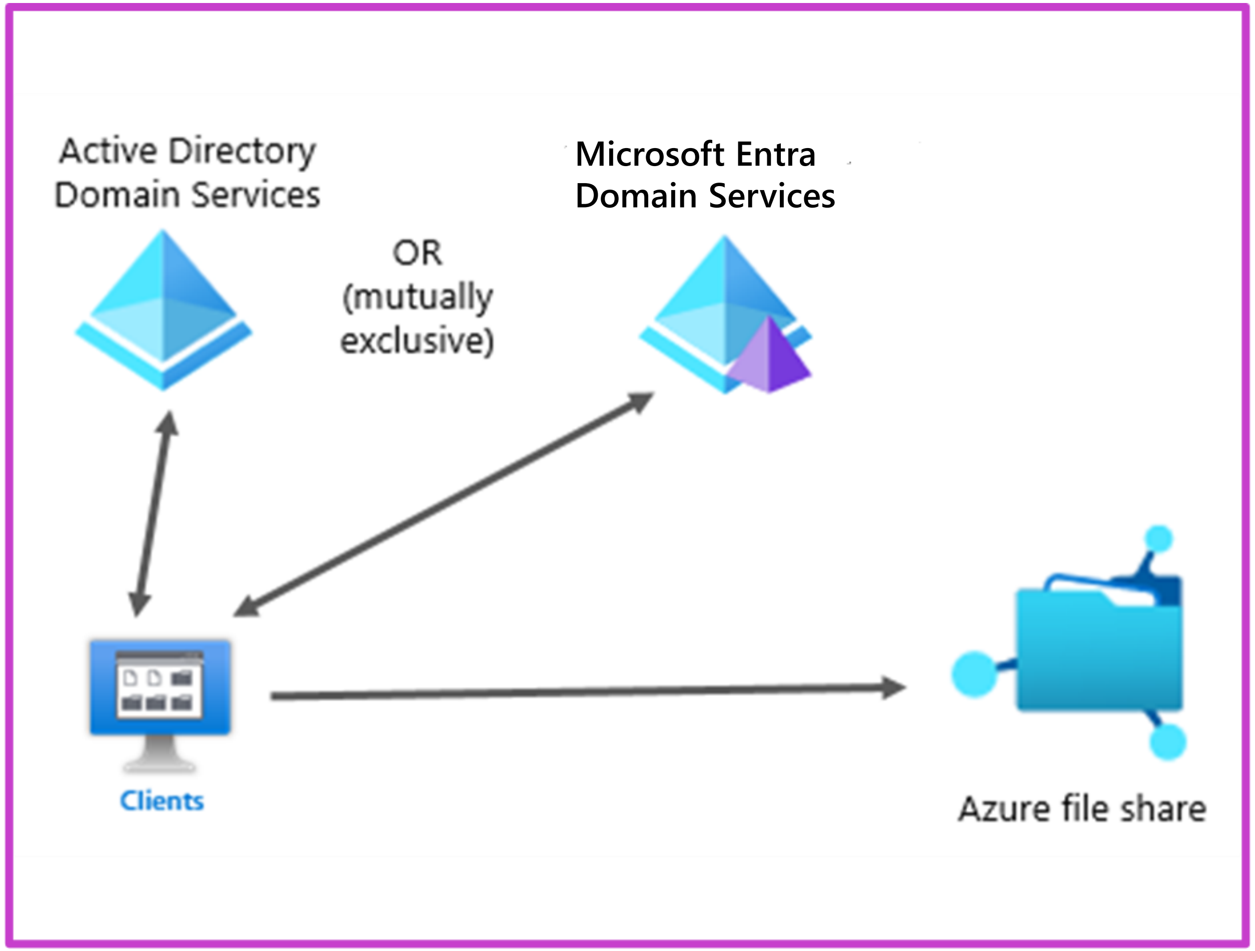


Configure Azure Files (1/3)

Access to Azure Files requires users to authenticate:

- Identity-based authentication over SMB Roles
- Access key
- A Shared Access Signature (SAS) ✓
- Microsoft Entra Kerberos for hybrid identities

The graphic on the right shows how users access files on the Azure file share with their sign-in credentials.


Active Directory Domain Services
OR (mutually exclusive)
Microsoft Entra Domain Services
Clients
Azure file share

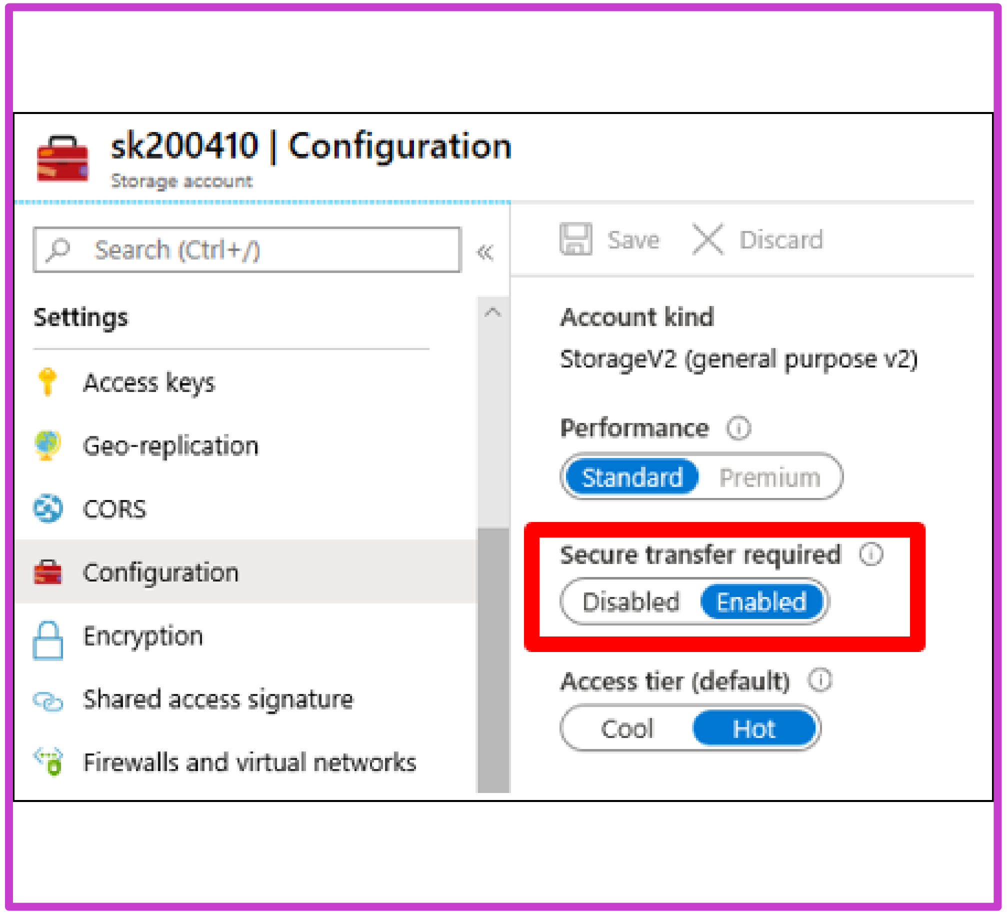
Configure Azure Files (2/3)

Azure provides the following RBAC roles for identity-based access to Azure Files:

- Storage File Data SMB Share Contributor
- Storage File Data SMB Share Elevated Contributor
- Storage File Data SMB Share Reader

Data encryption

- All data that's stored in an Azure storage account is always encrypted at rest by using Storage Service Encryption (SSE).
- By default:, data encrypts by using Microsoft managed keys, but you can choose to bring your own encryption key and all Azure storage accounts have encryption-in-transit enabled.



sk200410 | Configuration
Storage account

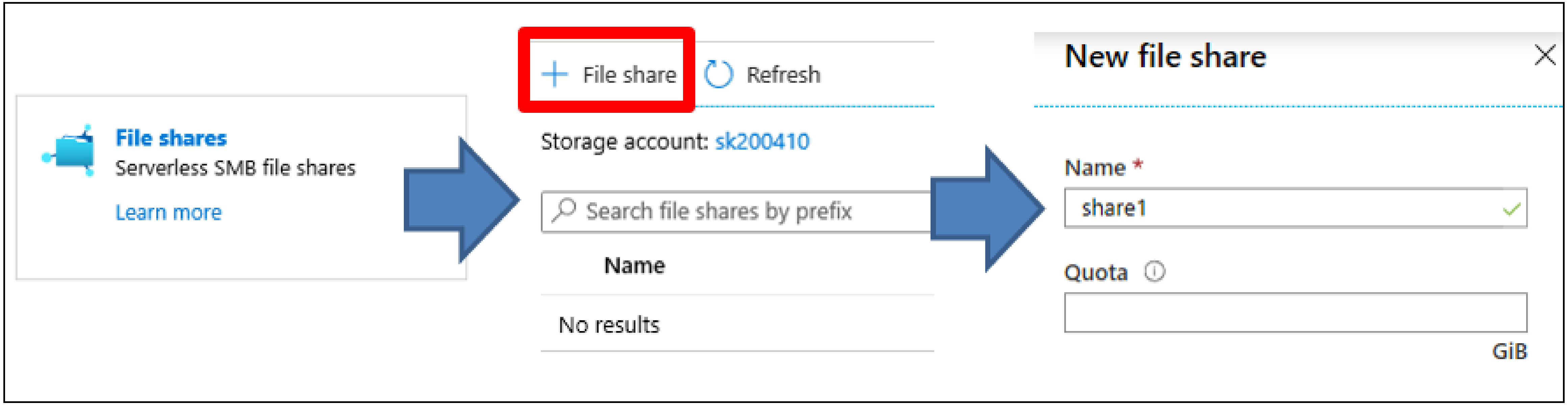
Account kind
StorageV2 (general purpose v2)

Performance: [✓ Standard] Premium

Secure transfer required: Disabled [✓ Enabled]

Access tier (default): Cool [✓ Hot]

Configure Azure Files (3/3)



To create an Azure file share, use the following procedure:

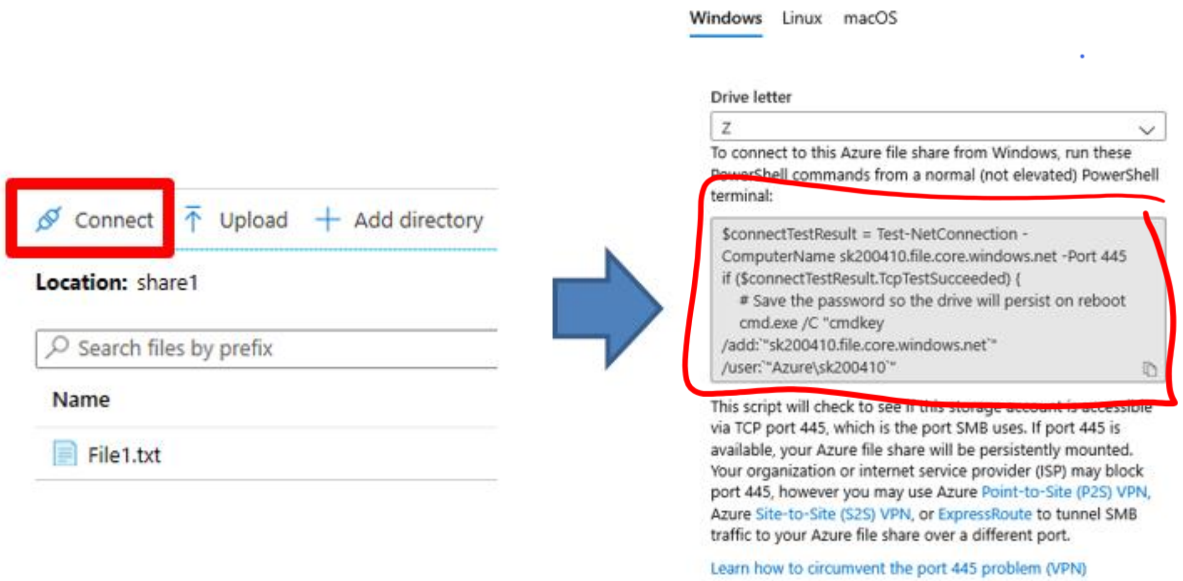
1. In the Azure portal, select the appropriate storage account.
2. In the navigation pane, under File service, select **File shares.**
3. In the details pane, on the toolbar, select + **File share.**
4. In the New file share blade, enter the desired Name and Quota values, and then select **Create.**

Configure connectivity to Azure Files

net use g
\\....\...

- By default, a storage account firewall allows access from all Azure VNets and the public internet.
- Firewall configuration also enables you to select trusted Azure platform services to access a storage account securely.
- In addition to the default public endpoint, storage accounts provide the option to have one or more private endpoints.
- To use an Azure file share, you must either mount it—which means assigning it a drive letter or mount point path—or access it through its Universal Naming Convention (UNC) path.
- The UNC path includes the Azure storage account name, the file.core.windows.net domain suffix, and the share name.

New-PSDrive



Location: share1

Drive letter: Z

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

```
$connectTestResult = Test-NetConnection -
ComputerName sk200410.file.core.windows.net -Port 445
if ($connectTestResult.TcpTestSucceeded) {
    # Save the password so the drive will persist on reboot
    cmd.exe /C "cmdkey
/add:`"sk200410.file.core.windows.net`"
/user:`"Azure\sk200410`"
```

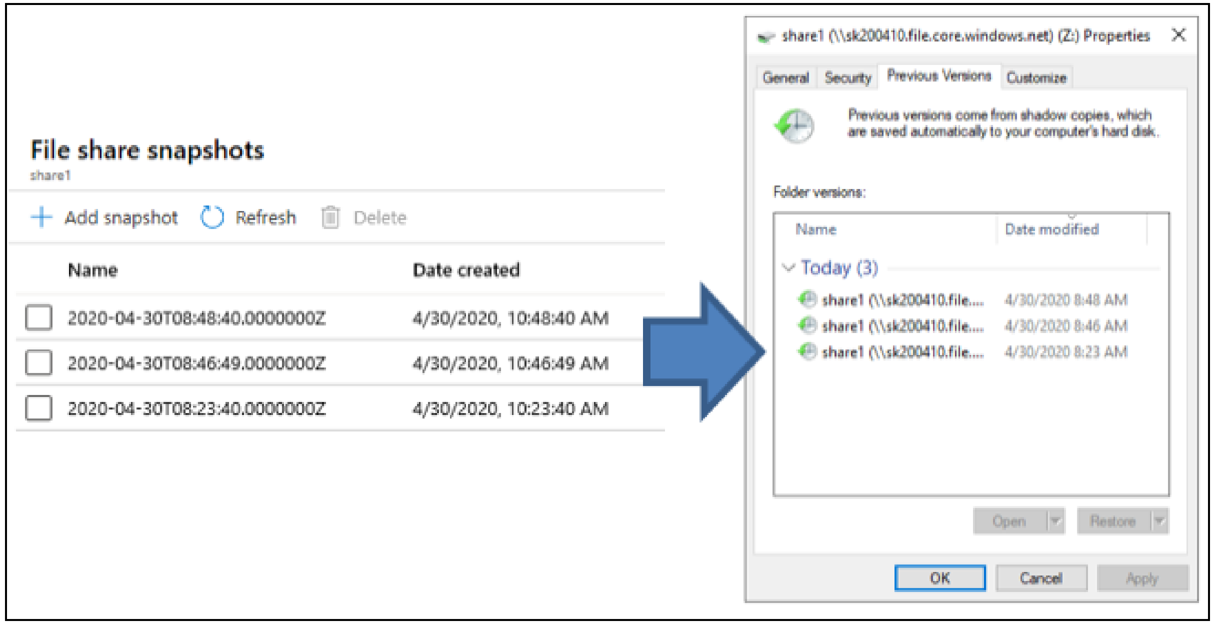
This script will check to see if this storage account is accessible via TCP port 445, which is the port SMB uses. If port 445 is available, your Azure file share will be persistently mounted. Your organization or internet service provider (ISP) may block port 445, however you may use Azure Point-to-Site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

Learn how to circumvent the port 445 problem (VPN)

Create Azure Files share snapshots

Azure Files share snapshots

- A share snapshot created at the file share level.
- You can have up to 200 snapshots per share.
- Share snapshots are incremental.
- Use snapshots in the following situations:
  - As protection against accidental deletions or unintended changes.
  - For general backup purposes.



File share snapshots

share1

	Name	Date created
□	2020-04-30T08:48:40.0000000Z	4/30/2020, 10:48:40 AM
□	2020-04-30T08:46:49.0000000Z	4/30/2020, 10:46:49 AM
□	2020-04-30T08:23:40.0000000Z	4/30/2020, 10:23:40 AM

share1 (\\sk200410.file.core.windows.net) (Z:) Properties

Previous versions come from shadow copies, which are saved automatically to your computer's hard disk.

Folder versions:

Name	Date modified
Today (3)	
share1 (\\sk200410.file....	4/30/2020 8:48 AM
share1 (\\sk200410.file....	4/30/2020 8:46 AM
share1 (\\sk200410.file....	4/30/2020 8:23 AM

Implementing Azure File Sync

Learning Objectives – Implementing Azure File Sync

- Describe Azure File Sync
- Implement Azure File Sync
- Demonstration – Deploy Azure File Sync
- Manage cloud tiering
- Migrate from DFSR to Azure File Sync
- Learning recap

Describe Azure File Sync (1/3)

Benefits of Azure File Sync

Multisite sync

File Sync can be used for multisite sync.

File Sync implements multi-master replication.

Cloud tiering

It is an optional feature that you can enable and configure for server endpoints.

With cloud tiering, you can define the percentage of free space/define whether to locally store only recently accessed files.

Can copy as many files as you want to a server endpoint, and all the files sync to the cloud endpoint.

Cloud backup

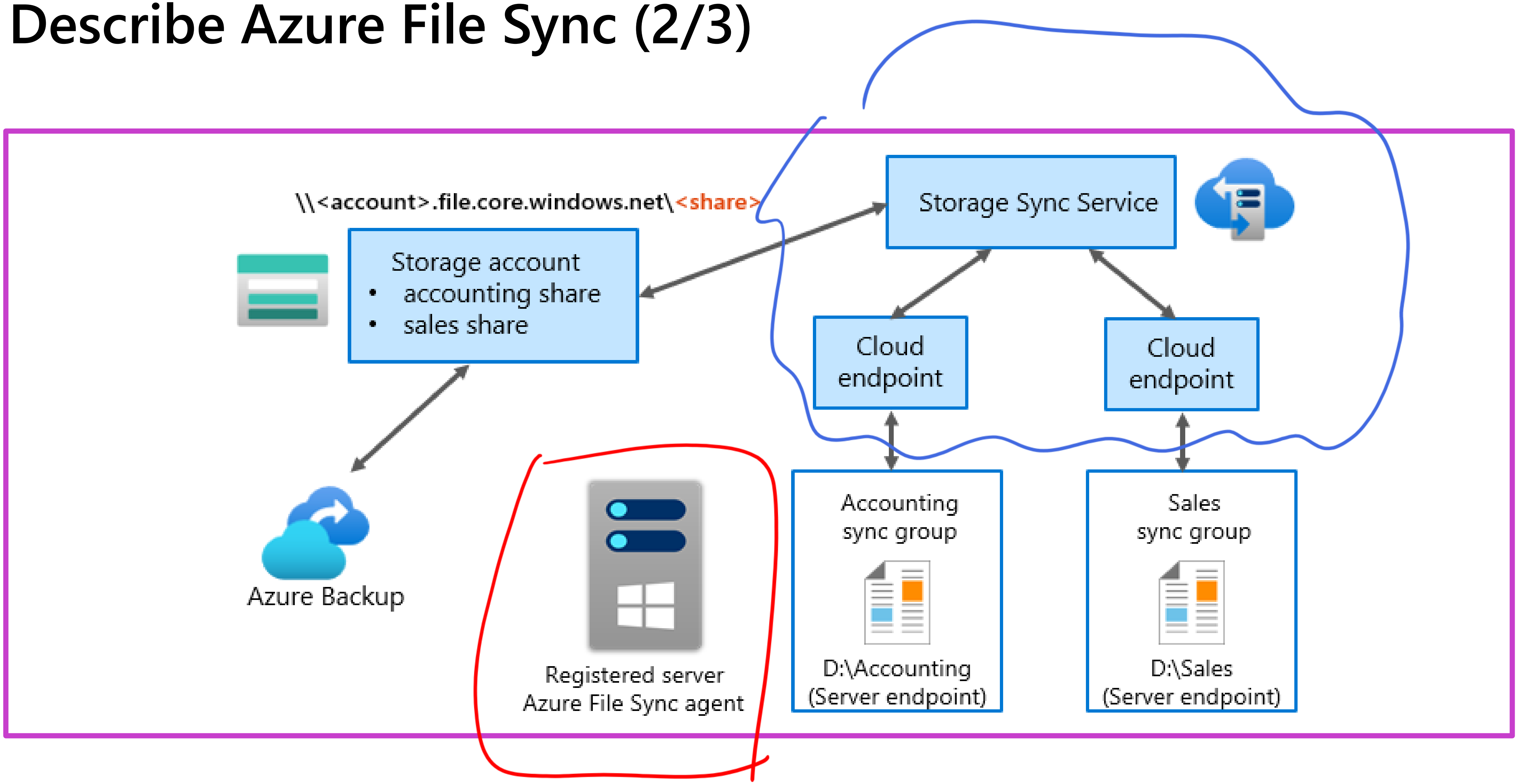
Use Azure Backup to perform one scheduled daily backup of Azure file shares, or up to four on-demand daily backups.

Azure Backup uses snapshots for creating an Azure file share backup.

Disaster recovery

Regardless of the amount of data on an Azure file share, the sync agent first pulls down the entire namespace, which makes folder structure and files almost immediately displayed and available on the new server.

Describe Azure File Sync (2/3)



Describe Azure File Sync (3/3)

Component	Description
Storage Sync Service	This is the top-level Azure resource for File Sync.
Sync group	A sync group defines the sync topology for a set of files. Endpoints within a sync group are kept in sync with each other.
Registered server	The registered server object represents a trust relationship between a server that's running Windows Server (or a cluster) and the Storage Sync Service.
Azure File Sync agent	The File Sync agent is a downloadable package that enables Windows Server to sync with an Azure file share.
Server endpoint	A server endpoint represents a specific location on a registered Windows Server computer, such as a folder or a volume.
Cloud endpoint	A cloud endpoint is an Azure file share that's part of a sync group.

Implement Azure File Sync

Implement Azure File Sync manually

To implement Azure File sync manually, you must complete the following high-level steps:

1. Deploy the Storage Sync Service
2. Install the Azure File Sync agent
3. Register Windows Server with Storage Sync Service
4. Create a sync group
5. Add server endpoints

Implement Azure File Sync by using Windows Admin Center

1. Connect Windows Server to WAC
2. Set up Azure File Sync from WAC
3. WAC deploys Azure File Sync Agent
4. WAC registers Windows Server with an FS sync group
5. Select the folders to sync

Demonstration – Deploy Azure File Sync

1. Prepare Windows Server to use with Azure File Sync
2. Deploy the Storage Sync Service
3. Install the Azure File Sync agent
4. Register Windows Server with Storage Sync Service
5. Create a sync group and a cloud endpoint
6. Create a server endpoint and enable cloud tiering

Manage cloud tiering (1/2)

Configure two policies:

Policy	Description
Volume free space policy - Always preserve the specified percentage of free space on the volume	For this policy, also called the volume free space policy, you specify the percentage of free space that must always be available on the volume on which a server endpoint is located.
Date policy - Only cache files that were accessed or modified within the specified number of days	This policy, also called the date policy, specifies that only the most recently accessed files cache locally. You define the number of days, and if a file isn’t read or written to for that many days, it’s automatically tiered.

Manage cloud tiering (2/2)

Recognize tiered files

You can recognize a tiered file in several ways, including:

- Tiered files don’t use local disk space because they’re stored on an Azure file share. Regardless of their actual size, the size on the disk is 0 bytes.
- Attributes, namely Offline, Reparse point, and Recall on data access are set on tiered files.
- Tiered files have reparse pointers set. A reparse pointer is a special pointer for the File Sync file system filter. To check whether a file has a reparse point, you can run the following command:
  `fsutil reparsepoint query <file-name>`

Tier or recall files manually

- You can manually force a file to be tiered.
- If a file is already tiered and you want to recall it, the easiest way to cache it locally is to open the file, such as by double-clicking or selecting it in File Explorer.
- You can also run the `Invoke-StorageSyncFileRecall` cmdlet, which can be especially useful if you want to recall multiple files at once.

Migrate from DFSR to Azure File Sync (1/3)

Windows Server has the following two DFS-related role services:

- DFS Namespaces
- DFS Replication

DFS and Azure File Sync

- File Sync is compatible with DFS-N and DFSR.
- You can install the File Sync agent on DFSR servers, and then sync data between those server endpoints and the cloud endpoint.
- DFSR and File Sync are both replication solutions that can work side by side.
- If you want to use File Sync and DFSR side by side, you must:
  - Disable File Sync cloud tiering on volumes with DFSR-replicated folders.
  - Not configure DFSR read-only replication folders as server endpoints.

Migrate from DFSR to Azure File Sync (2/3)

To migrate a DFSR deployment to File Sync, perform the following high-level steps:

1. Create an Azure storage account, an Azure file share, and a Storage Sync Service resource in your Azure subscription.
2. Create a Storage Sync Service sync group to represent the DFSR topology that you’re replacing. In DFSR, replication groups define the replication topology. You must define the same topology in File Sync by using sync groups.
3. Install the File Sync agent on the DFSR server that has all the data that you want to migrate.
4. Register the server in File Sync and then create the first server endpoint. Don’t enable cloud tiering for the first server endpoint.
5. Wait until all the server endpoint data syncs to the Azure file share (cloud endpoint).

Migrate from DFSR to Azure File Sync (3/3)

Migration process continued:

6. Install and register the File Sync agent on each remaining DFSR server.
7. Disable DFSR.
8. Create a server endpoint on each of the DFSR servers. Don’t enable cloud tiering.
9. Ensure that the sync completes and then test your topology as desired.
10. Retire DFSR.
11. You can now enable cloud tiering on any server endpoint you wish.

Learning recap – Implement hybrid file server infrastructure

Module assessment

Microsoft Learn Modules (docs.microsoft.com/Learn)

Implement a hybrid file server infrastructure

Lab 10 – Implementing Azure File Sync

Lab 10: Implementing Azure File Sync

Lab scenario

To address concerns regarding Distributed File System (DFS) Replication between Contoso’s London headquarters and its Seattle–based branch office, you decide to test Azure File Sync as an alternative replication mechanism between two on-premises file shares.

Objectives

- Implement DFS Replication in your on-premises environment
- Create and configure a sync group
- Replace DFS Replication with Azure File Sync-based replication
- Verify replication and enable cloud tiering
- Troubleshoot replication conflicts

Lab 7 DHCP-Server, DNS-Server

Lab 8 Azure Hub-Spoke, Network Watcher, User Defined Routes

Lab 9 iSCSI, Storage Spaces, Storage Spaces Direct

Lab 10 DFS und Azure File Sync

End of presentation