**Implement and manage storage**

**Secure storage (**Manage storage account)

Storage accounts supports 5 different types of data objects

Blob – unstructured data, (videos, pictures) – applications needing access to data anywhere

(this is storage in Container)

File – file access to applications using SMB protocol (server message block)

Queue – asynchronous message queueing for apps deployed in azure

Table – structured noSql data, for key attribute store without schema

Disk – virtual disk used by VM’s

Storage account – entity that holds and manages the data objects

Unique namespace to storage resources

Secure and scalable

http://storageaccount001.blob.core.windows.net/containername/pic.jpg

Types of storage account:

General purpose v2 – all types of data objects, recommended, only type support data lake gen2

General purpose v1 – Legacy

BlockblobStorage – premium performance accounts for block blobs and append blobs

File storage accounts – recommended for Files

BlobStorage – legacy blob only account (use v2 instead)

Performance tiers:  
 Standard – V2, V1, Blob storage

Recommended for backup, DR and media

Premium – available only for = BlockBlob storage, FileStorage, GPv1 and GPv2 (unmanaged VHD only)

Recommended for Interactive, analytics and AI

Need to understand workload performance needs and cost on determining what tier to select

Cannot convert, need to migrate files to new storage type (no conversion after deployment)

Access Tiers:  
 Hot – Highest storage cost, lower access cost

Cold – Lower storage cost, higher access cost (30 days min)

Archive – Lowest storage cost, highest access cost (180 days min)

with early deletion cost

Replication options:  
 Local-redundant storage (LRS) – single availability zone – will not survive physical dc/zone outage

Zone-redundant storage (ZRS) – 3 sync copies on 3 zones – safe for physical dc/zone outage

Geo-Redundant storage (GRS) – LRS on primary + LRS (read only) for 2nd region

Geo-Zone-Redundant storage (GZRS) – ZRS on primary + LRS on 2nd (read only)

GRS and GZRS 2nd can only be access when failover occurs

If you want to read access even if no failover, use the below

Read-Access Geo-Redundant storage (RA-GRS)

Read-acccess Geo-Zone-Redundant storage (RA-GZRS)

Ask yourself

What is Azure DC fails?  
 What is Azure region fails?

Do you need Read access to redundant data in another azure region?

Create storage account:  
 Go to storage account > Choose performance > Choose replication > choose blob access tier > choose connectivity endpoint (public all net, public selected net, private) > choose data protection options (soft delete, etc) > choose security options (TLS, datalake storage gen2, > tags > create

Configuring storage account:

Ex. V1 to upgrade  
 Go to the storage account > Configuration > upgrade > Confirm upgrade > Change replication (other settings)

Storage account > Encryption (Microsoft manage keys or Customer manage keys) >

If customer managed > Encryption key, choose Enter KEY URI or key vault

Configure access control to storage accounts

Anonymous –

azure blobs only

Authenticated –

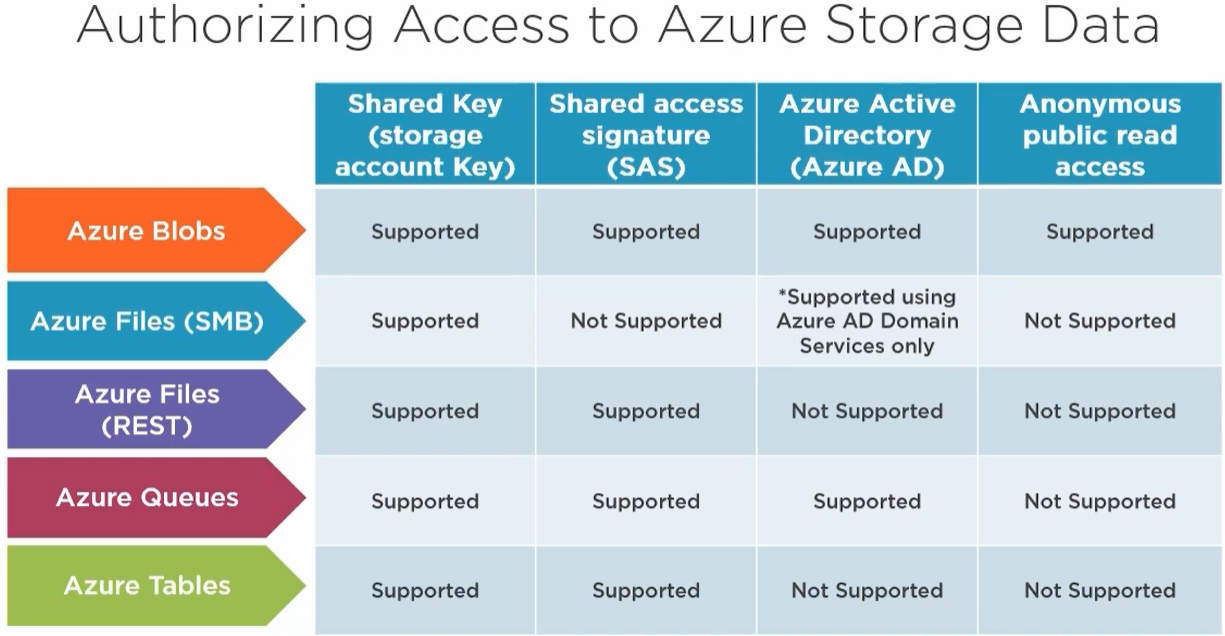
Shared access key – access to whole storage account, regenerate if compromise

-storage account > access key

Shared access signatures (SAS) – User delegation, Service, Account

-storage account > shared access signature > choose allowed services > allowed types > allowed permission > start date and end date > choose which access key > copy right away

Azure AD (RBAC) – set the data layer permission (storage account > container > blob (type) > IAM) and management resrouce permissions (storage account > IAM)



Network access control

Storage firewalls and Virtual networks

Still require prorper authorization

Layered security model

Limit access by rules – IP address, subnet/vnet

Storage account > Firewalls and virtual networks > add Vnet > IP address

**Manage storage**

Manage data in azure storage

Import/Export into/from azure job

Copy data by AZCopy

Install and use Azure storage explorer

Migrating data to azure

Via internet or Ship physical media

Azure import/Export service

Shipping disk drives

Components:  
 azure import/export service

WAImportExportTool

Drives

OVerView:

Importing files to azure - Supports blob storage and File storage

Copy files to drives and add BitLocker and NTFS

Create import job in Azure portal

Drives shipped to azure datacenter

Data is copied to storage account and customer verification

Drives are returned to the customer

Export blob from azure – supports only blobs

Create export job from azure portal

Ship drive to datacenter

Export data copied to encypted drive

Drive shipped to customer

Unlock and very drive with WAImportExport unlock command

Drive preparation with WAImportExportTool.exe – command line tool

Supports windows 64bit only

encryption and decryption

calculates how many drives needed

creation of journal files

V1 is used for Azure Blobs  
 V2 is for Azure Files

Driveset.csv - List of disks mapped to drive letters

* Can format drives with NTFS if unformatted

Dataset.csv – list of files and/or directories to be imported

* Local files mapped to azure storage endpoint

Steps (import):

Search import/export job in marketplace > Create > Type: Import> upload journal files > destination region and storage account > provide shipping address > acknowledge terms > Encryption key found on the job status

Steps (Export):  
 Search Import/export job in marketplace > Create > Type: Export > Choose blobs to export (all, selected containers, XML list) > provide shipping details > ack terms >

From Encryption blade in the export job, get the Bitlocker key > unlock using the WAImportExport

AZCopy

Command line utility

To and from Azure

AZCopy Authentication:

For BlobStorge – SAS and AzureAD

For FileStorage – SAS only

(Ensure proper roles are assigned in IAM and/or AAD)

With SAS

Azcopy <action> <source> <destination URL appended with <container>/<SAS key>

With AzureAD

Azcopy login

Azcopy <action><source><Destination>

Azure Storage Explorer

Desktop GUI to manage azure storage

Has all the functions from the Azure portal

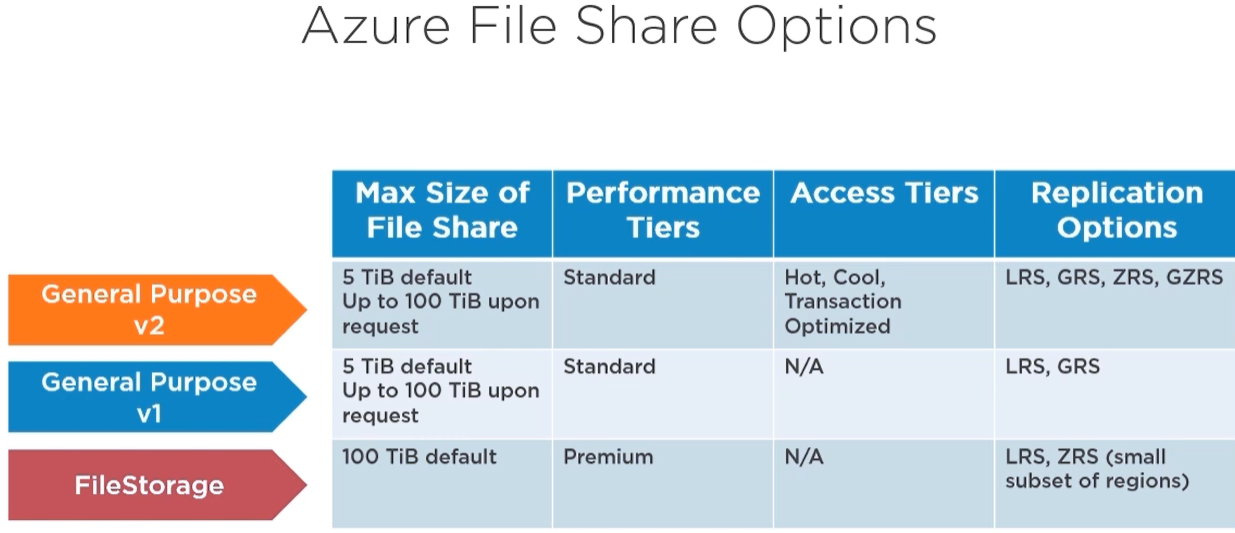
Download and install Storage Explorer > Open > add account

**Configure azure files and azure Blob Storage**

**Azure File Share** – Cloud based SMB and NFS

Port 445

100TiB out of the box



Create and configure file share

Create storage account > Add FileShare > Set capacity

Go to FileShare > Add directory (or subdirectory)

To snapshot and Backup

Go to storage account > FileShare > Directory > Snapshot/Backup

To Connect using windows client

Go to storage account > FileShare > Directory > Connect > choose OS > choose drive letter > copy the code provided

Azure Files Access Tiers

HOT – storage cost high, access low cost (general purpose)

COOL – low cost storage, high cost access (online archive)

Transaction Optimized – Highest cost of storage, lowest cost per transaction (backend store of app)

Go to storage account > FileShare > Directory > Change Tier

Azure FileShare Authentication

Azure active directory DS or ADDS

Storage account key or Share access Signature

Azure File Sync

Integrate/replicate azure fileshare to onprem fileshare

Requirements:  
 Cloud Endpoint (azure file share) -

Server Endpoint -

Sync Group -

Steps:  
 Deploy the storage sync service in azure

Create a sync group and cloud endpoint

Install azure file sync agent on windows server

Register Windows server with the storage sync service

Create a server endpoint and wait for sync

Go to Azure File Sync > Create

Go to the File sync resource > + Sync Group > choose the storage account > Choose the data file share’

Add the cloud endpoint

Download/install the azure file sync agent > Open and login > choose the storage sync service > register

Go back to sync group resource > Add server end point > choose registered server > local path on the local server

**Azure Blob Storage (Binary large objects)**

Blob type:  
 Block Blob

Append

VHD

Requirements:  
 Storage account

Container

Blob

Azure Blob Access Tiers:

HOT

COLD

ARCHIVE

Lifecycle management – Role based policy, automate the tiering

Storage account > Blob services, Lifecycle management > Add a rule > Scope (name, version, type) > If else

You can also see the Json code

Blob object replication – asynchronous copies of block blobs between storage accounts

Scenarios: min latency, increase efficiency of workloads, optimize distribution and cost

Blob replication is only for hot and cold tier, no snapshot support and no immutable blob support

Blob replication Requirements:

Source and target storage account containers already created

Versioning for blob enabled (source)

Blob storage feed enabled (source and destination)

Create blob replication steps:

Storage account > container > blob services, object replication > replication rules > container pairs (source and destination)

Exam alert: