



# KodeKloud

© Copyright KodeKloud

Follow us on <https://kodekloud.com/> to learn more about us.

# Administer Azure Storage

© Copyright KodeKloud

These Learn modules are part of the AZ-104: Implement and manage storage in Azure (<https://docs.microsoft.com/learn/patterns/az-104-manage-storage/>) learning path.

# Learning Objectives

- 01 Configure Storage Accounts
- 02 Configure Blob Storage
- 03 Configure Azure Files
- 04 Configure Storage Security

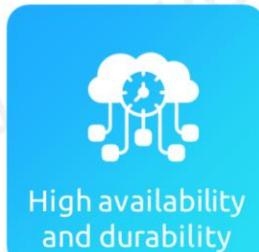


# Storage Accounts

© Copyright KodeKloud

- A repeatable way to deliver software and infrastructure code to its destination
- Build and test code
- Deploy to on-prem or cloud resources
- CICD isn't just for software developers. It's also for infrastructure pros

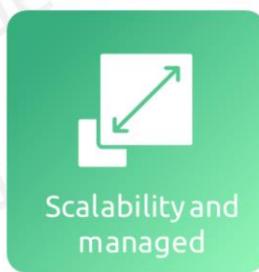
# Storage Accounts



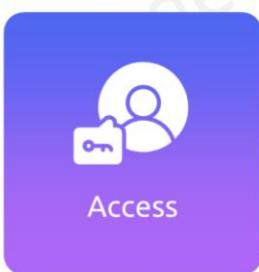
High availability and durability



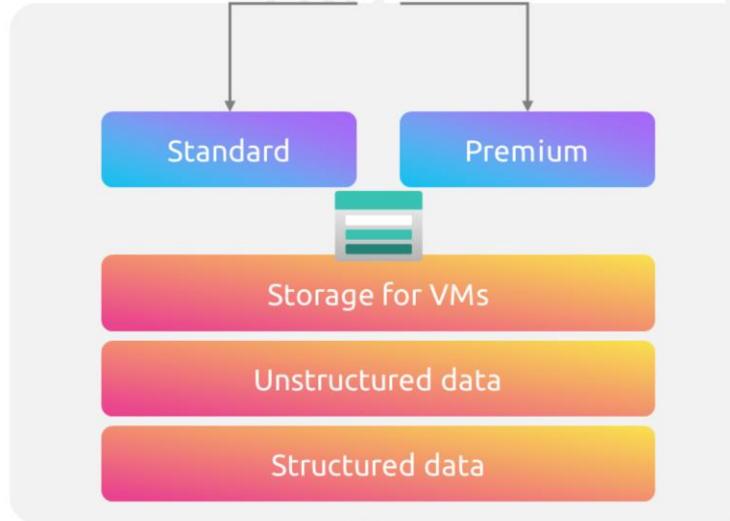
Security



Scalability and managed



Access



© Copyright KodeKloud

## High availability and durability

Storage account comes with different redundancies to fulfill your durability requirements. Data stored in the storage account can be replicated to different datacenters and even across regions ensuring high availability for the data.

## Security

By default, all data written to the storage account is encrypted by Storage Encryption Service. To access the data storage

accounts, provide different authorization methods such as storage keys, shared access signature, and Azure AD.

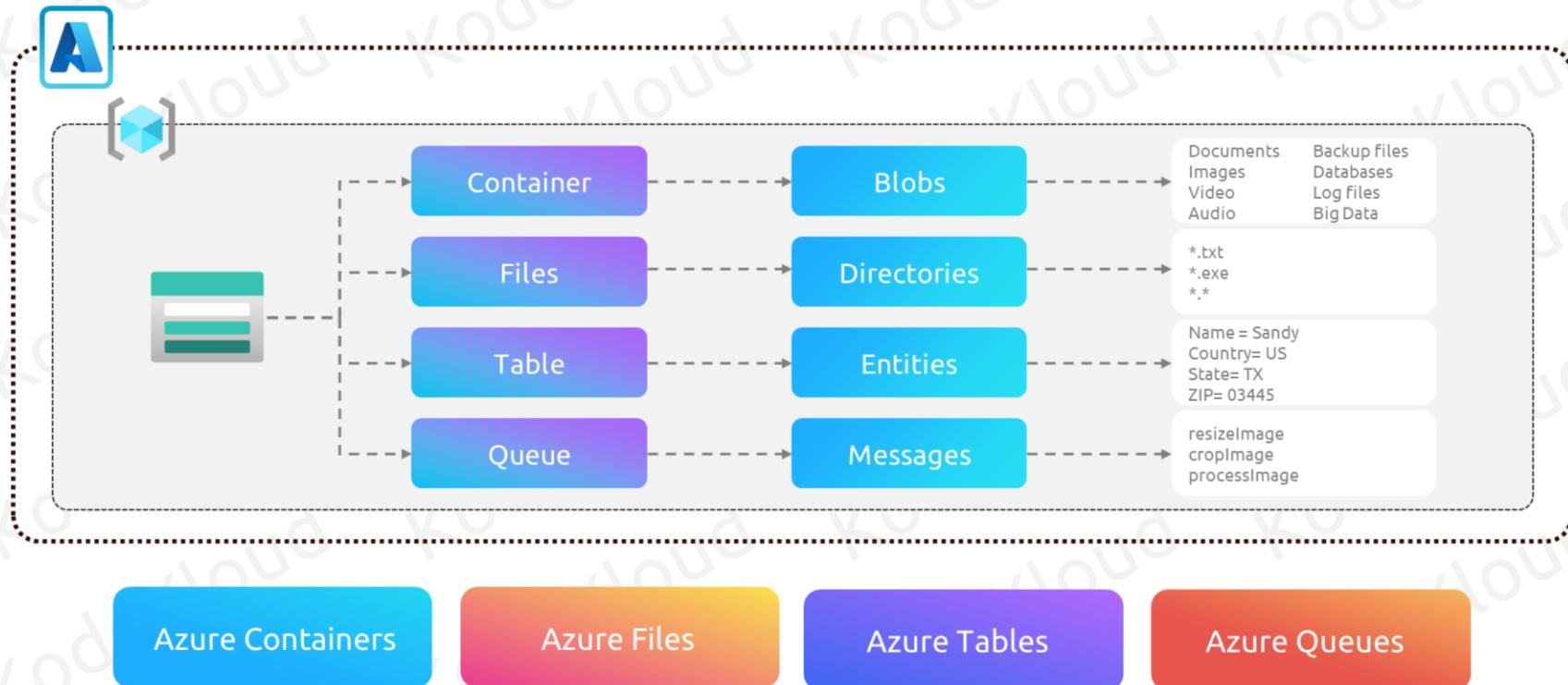
### **Scalability and Managed**

Azure Storage is a platform managed service, depending upon the requirement it will automatically scale the storage and performance.

### **Access**

HTTP or HTTPS can be used to access the data that is stored in Azure Storage. With the help SDKs provided by Microsoft, developers can easily integrate Azure Storage with their code. Azure Storage also supports Azure PowerShell, Azure CLI and REST API.

# Storage Services



© Copyright KodeKloud

## Azure Containers

An object store with immense scaling capability.

Ideal for storing unstructured data such as text or binary data.

## Azure Files

### Managed file share

Used to provision highly available file shares in cloud that can be mounted to cloud and on-premises machines.

## Azure Tables

### NoSQL datastore

Ideal for storing structured non-relational data

## Azure Queues

### Messaging store

Used to store messages and retrieve messages between application components that needs to be processed asynchronously.

# Storage Account Types

Type	Services	Performance tiers	Replication options
Blob storage	Blob	Standard	LRS, GRS, RA-GRS
General Purpose V1	Blob, File, Queue, Table, and Disk	Standard, Premium	LRS, GRS, RA-GRS
General Purpose V2	Blob, File, Queue, Table, and Disk	Standard, Premium	LRS, ZRS, GRS, RA-GRS, GZRS, RA-GZRS
Block blob storage	Blob	Premium	LRS, ZRS
File storage	Files	Premium	LRS, ZRS



# Storage Redundancy

© Copyright KodeKloud

- A repeatable way to deliver software and infrastructure code to its destination
- Build and test code
- Deploy to on-prem or cloud resources
- CICD isn't just for software developers. It's also for infrastructure pros

# Storage Replication – Locally Redundant Storage



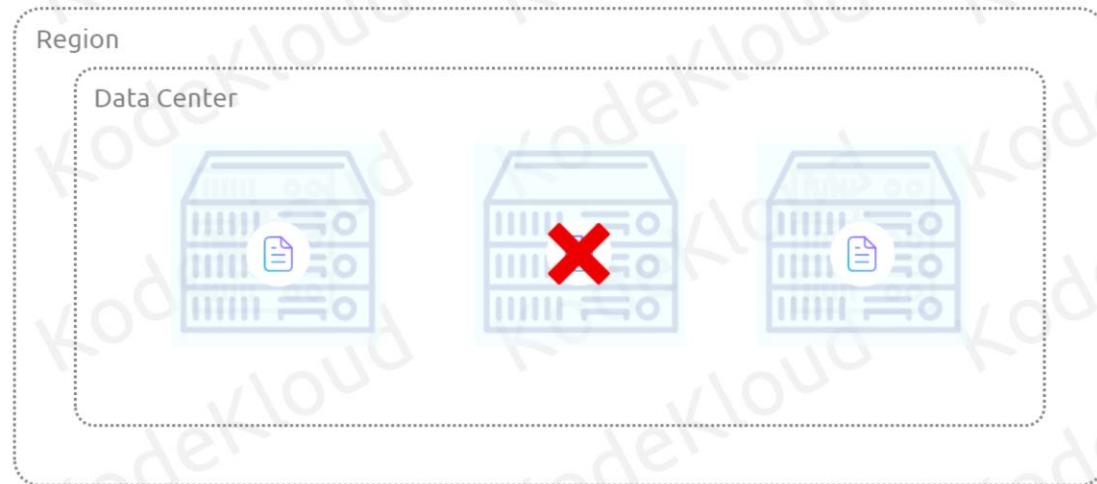
Replication



Durability



Chances of failure



© Copyright KodeKloud

## Replication

Data is replicated and will retain three copies of data across fault domain within a single datacenter. Since the data is replicated only within a single data center, LRS is the cheapest option.

## Durability

LRS offers 99.999999999 (11 9's) of durability. Data stored in LRS is protected from hardware failures as the data is

stored in different fault domains.

### **Chances of failure**

As the replicated copies are stored within a single datacenter, if the entire datacenter is down, then the data will not be available

# Storage Replication – Zone-Redundant Storage



Replication



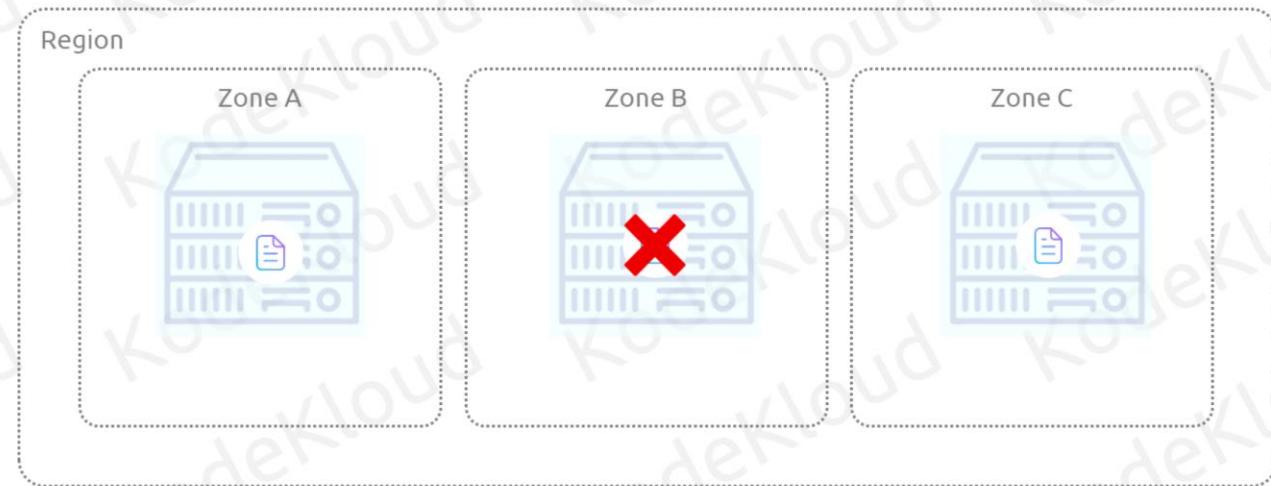
Durability



Chances of failure



Storage Account - ZRS



© Copyright KodeKloud

## Replication

Data is replicated and will retain three copies of data across availability zones within a single region.

## Durability

ZRS offers 99.9999999999 (12 9's) of durability. Data stored in ZRS is protected from datacenter failures as each zone where the datacenter resides is physically separated from each other.

### **Chances of failure**

As the replicated copies are stored within a single region, if the entire region goes down, then the data will not be available

# Storage Replication – Geo-Redundant Storage



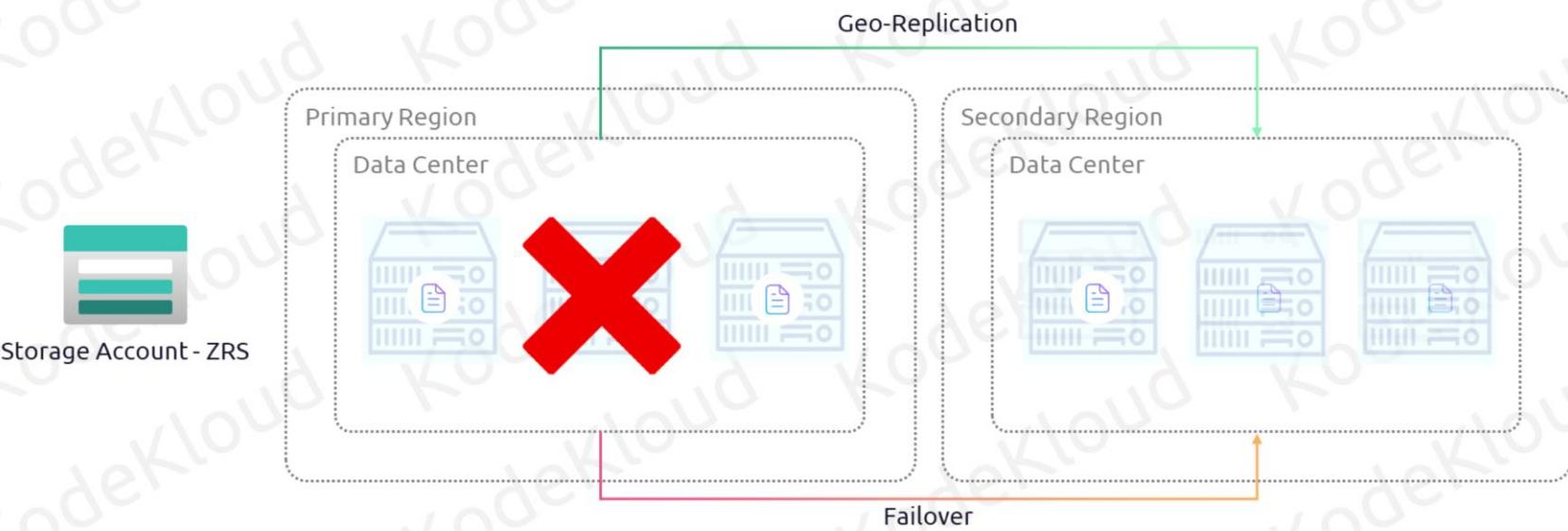
Replication



Durability



Considerations



© Copyright KodeKloud

## Replication

Data is replicated across three fault domains in a datacenter which is part of the primary region and is asynchronously replicated to secondary region where we will have three copies across fault domains.

## Durability

GRS offers 99.99999999999999 (16 9's) of durability. If the primary region goes down, a failover will happen, and

secondary region will become available for read requests.

### **Considerations**

The primary region will be available for all operations and secondary will be only available after failover. The failover can be Microsoft initiated or customer initiated.

# Storage Replication – Read Access Geo-Redundant Storage



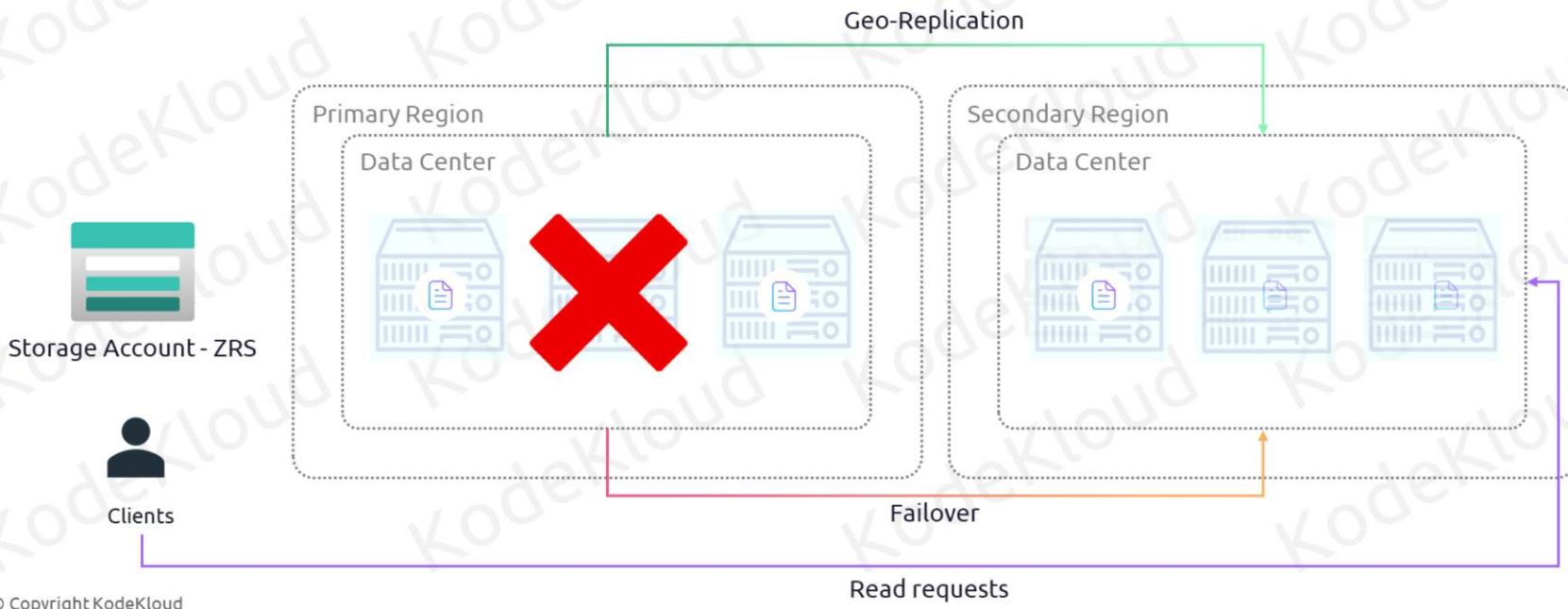
Replication



Durability



Considerations



## Replication

Data is replicated across three fault domains in a datacenter which is part of the primary region and is asynchronously replicated to secondary region where we will have three copies across fault domains.

## Durability

GRS offers 99.99999999999999 (16 9's) of durability. If the primary region goes down, a failover will happen, and

secondary region will become available for read requests.

### **Considerations**

The primary region will be available for all operations and secondary will be only available after failover. The failover can be Microsoft initiated or customer initiated.

# Storage Replication – Geo-Zone-Redundant Storage



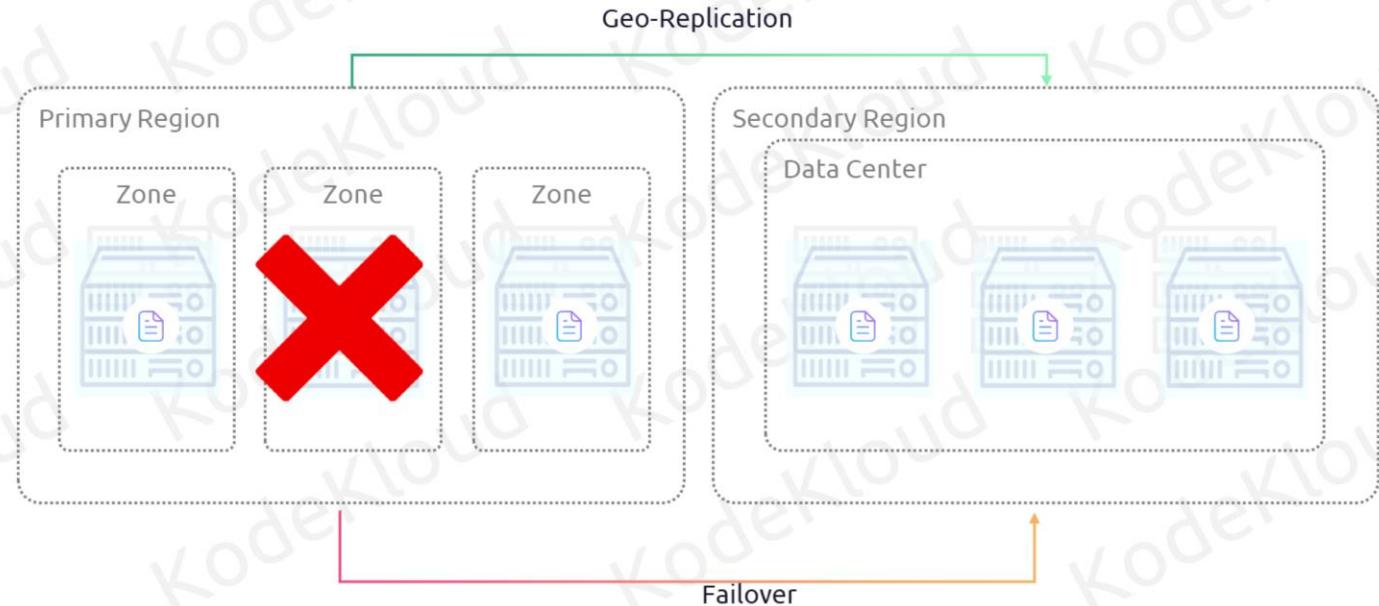
Replication



Durability



Considerations



© Copyright KodeKloud

## Replication

Three copies will be spread across availability zones within the primary region and is asynchronously replicated to secondary region where we will have three copies across fault domains.

## Durability

GZRS offers 99.99999999999999 (16 9's) of durability

## **Considerations**

Here secondary region will be always available regardless of whether there is a failover or not.

# Storage Replication – Read Access Geo-Zone-Redundant Storage



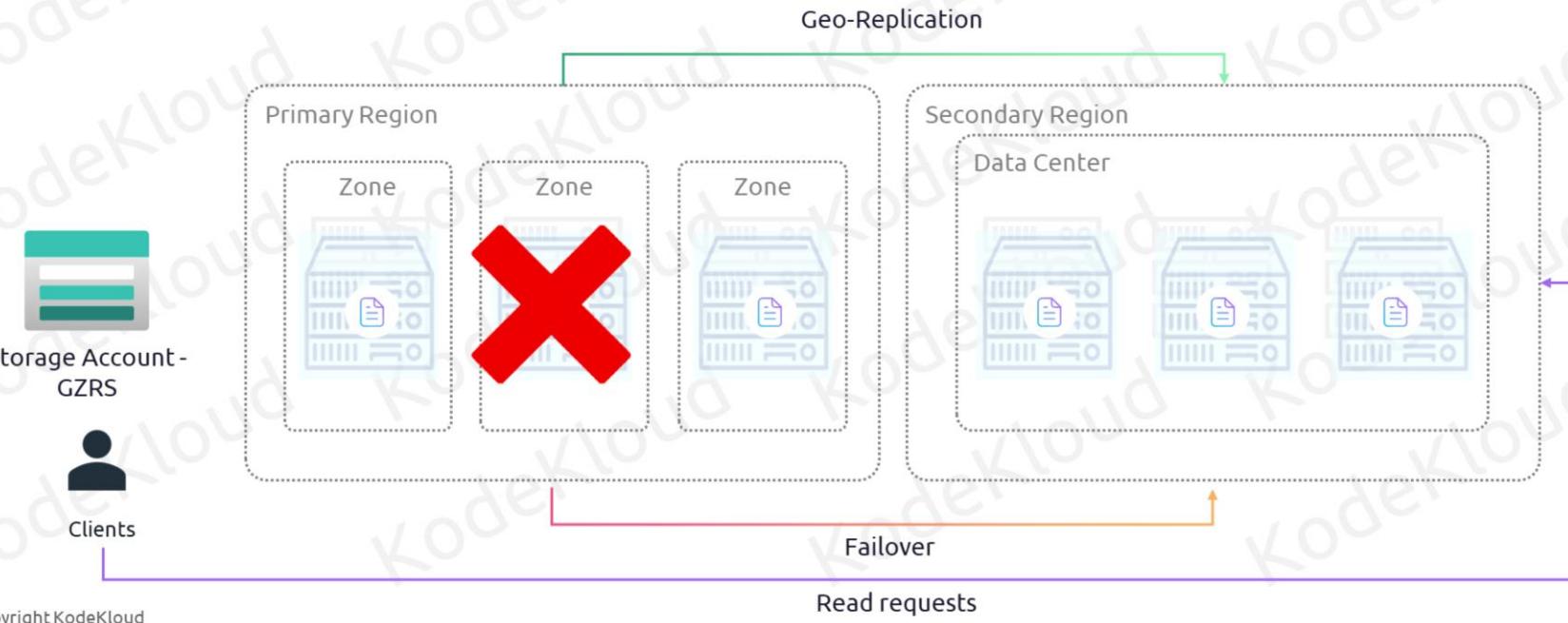
Replication



Durability



Considerations



## Replication

Three copies will be spread across availability zones within the primary region and is asynchronously replicated to secondary region where we will have three copies across fault domains.

## Durability

GZRS offers 99.99999999999999 (16 9's) of durability

### **Considerations**

Here secondary region will be always available regardless of whether there is a failover or not.



# Accessing storage endpoints

© Copyright KodeKloud

- A repeatable way to deliver software and infrastructure code to its destination
- Build and test code
- Deploy to on-prem or cloud resources
- CICD isn't just for software developers. It's also for infrastructure pros

# Accessing Storage Endpoints

Based on the storage account name and the service, every service has its own unique endpoint.

<protocol>://<storage account name>.<service>.core.windows.net  
http, https      Your storage account name   blob, queue, file, table

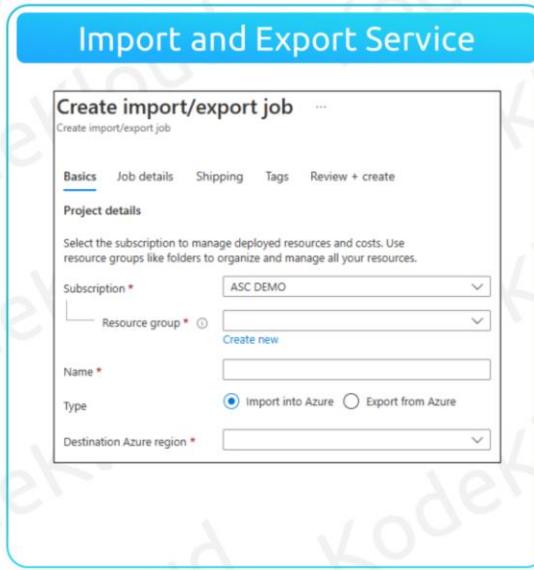
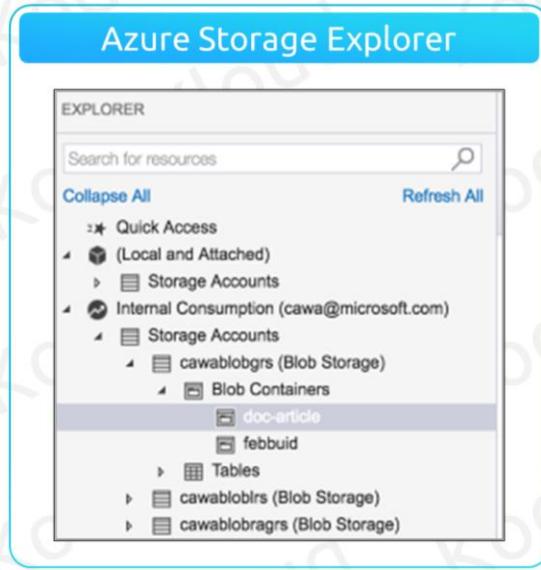
For a storage account named “kodekloud”, the endpoints will be:

Service	Endpoint
Container service	https://kodekloud.blob.core.windows.net
Queue service	https://kodekloud.queue.core.windows.net
File service	https://kodekloud.file.core.windows.net
Table service	https://kodekloud.table.core.windows.net

If needed, we can use our own custom domain with CNAME mapping.

DNS CNAME entry	Alias
blobs.kodekloud.com	kodekloud.blob.core.windows.net

# Configure Storage With Tools (optional)



© Copyright KodeKloud

This used to be an entire Learn module. Now it is a summary slide. The removed slides are at the end of the presentation.

Get started with Storage Explorer - <https://docs.microsoft.com/azure/vs-azure-tools-storage-manage-with-storage-explorer>

Upload, download, and manage data with Azure Storage Explorer - <https://docs.microsoft.com/learn/modules/upload-data-to-azure-storage>

[download-and-manage-data-with-azure-storage-explorer/](#)

Use the Azure Import/Export service to export data from Azure Blob storage -  
<https://docs.microsoft.com/azure/storage/common/storage-import-export-data-from-blobs>

Export large amounts of data from Azure by using Azure Import/Export -  
<https://docs.microsoft.com/learn/modules/export-data-with-azure-import-export/>

Use the Azure Import/Export service to import data to Azure Blob Storage -  
<https://docs.microsoft.com/azure/storage/common/storage-import-export-data-to-blobs>

Get started with AzCopy - <https://docs.microsoft.com/azure/storage/common/storage-use-azcopy-v10?toc=/azure/storage/files/toc.json>

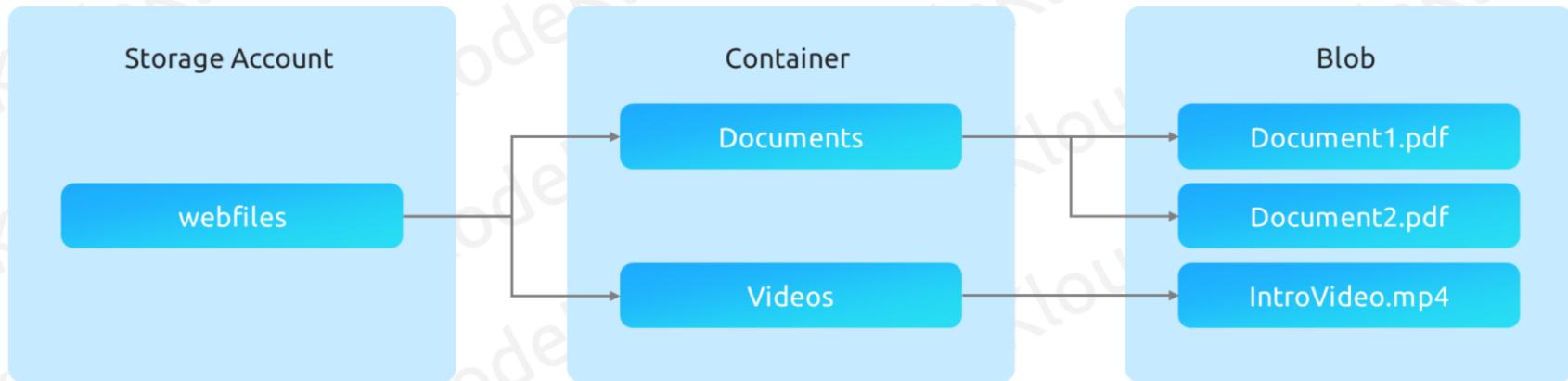


# Configuring Azure Blob Storage

© Copyright KodeKloud

- A repeatable way to deliver software and infrastructure code to its destination
- Build and test code
- Deploy to on-prem or cloud resources
- CICD isn't just for software developers. It's also for infrastructure pros

# Azure Containers (Blob Storage)



© Copyright KodeKloud

Embed images or documents in webpages

Stream video and audio directly to browser

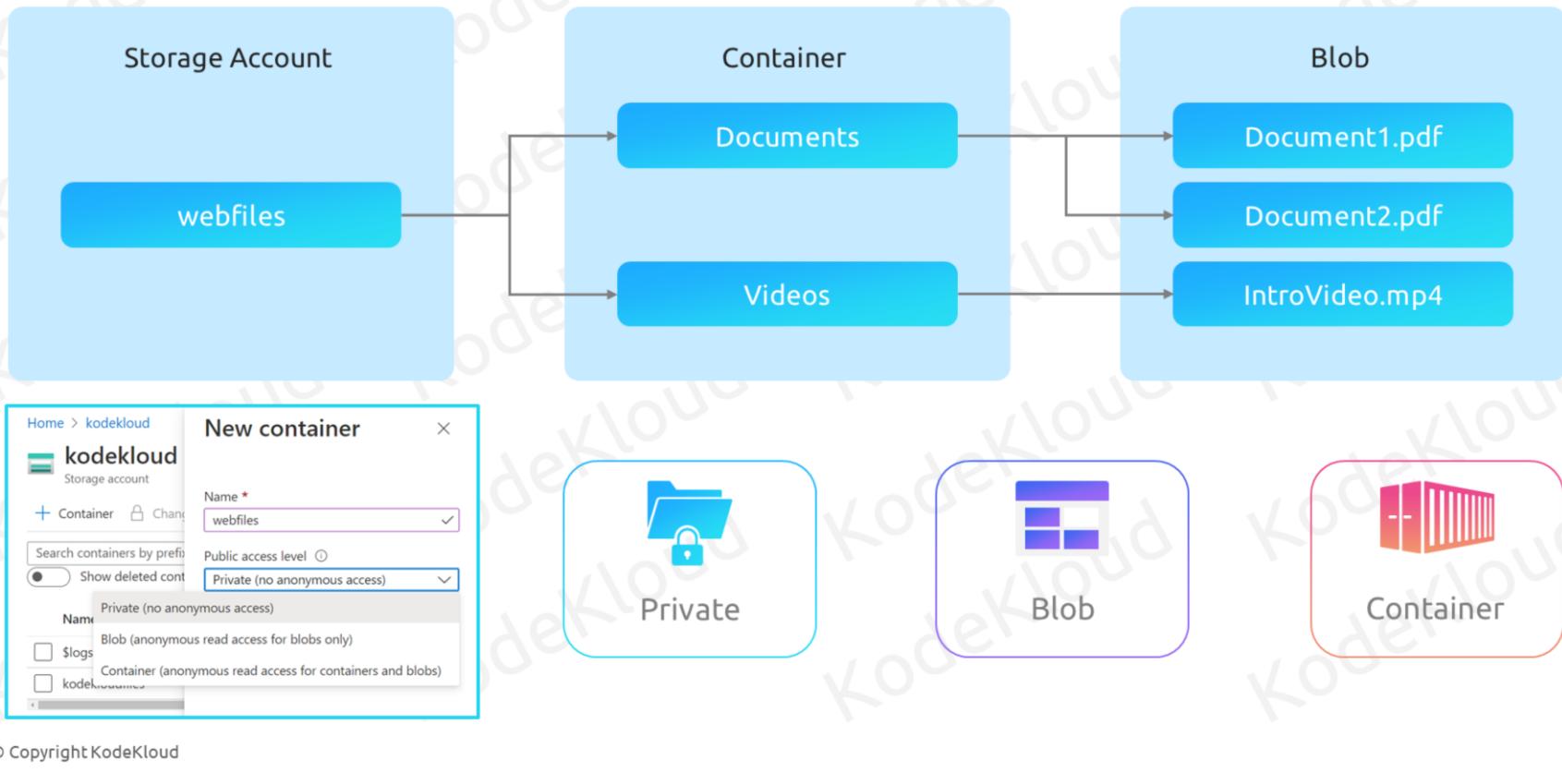
Strong files for distribution for example installation packages on websites

Act as a disaster recovery site for your on-premises site

Backup, recovery and archiving

Store data for analysis which can be accessed by tools like Power BI

# Creating Containers



## Private

No anonymous access to data stored in the container

## Blob

Anonymous read access to blobs only

**Container**



# Storage Tiers

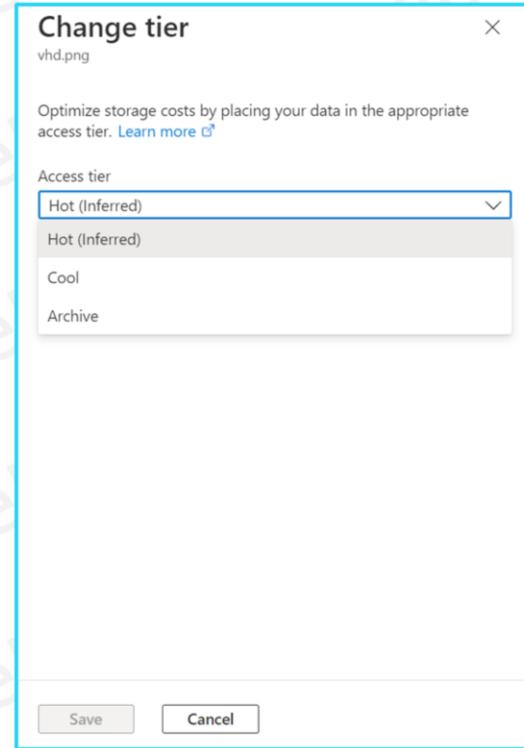
© Copyright KodeKloud

- A repeatable way to deliver software and infrastructure code to its destination
- Build and test code
- Deploy to on-prem or cloud resources
- CICD isn't just for software developers. It's also for infrastructure pros

# Storage Access Tiers



Tier	Storage Cost	Access Cost
Hot	\$\$\$	\$
Cool	\$\$	\$\$
Archive	\$	\$\$\$



© Copyright KodeKloud

## Hot

Ideal for storing data that is frequently accessed.

## Cool

Ideal for storing large amounts data that is not accessed frequently and is stored for at least 30 days.

## **Archive**

Ideal for data that can tolerate several hours of retrieval latency and will remain the archive tier for at least 180 days.

*Access tiers can be switched any time as required*



# Lifecycle Management

© Copyright KodeKloud

- A repeatable way to deliver software and infrastructure code to its destination
- Build and test code
- Deploy to on-prem or cloud resources
- CICD isn't just for software developers. It's also for infrastructure pros

# Blob Lifecycle Management

-  Policy-based transition
-  Delete blobs and snapshots
-  Filtering option
-  Target different types

© Copyright KodeKloud

## Policy based transition

We can transition blobs to cooler tiers automatically based on the last modified date.

## Delete blobs and snapshots

Besides transitioning to cooler tiers, LCM can be used to delete blobs and blob snapshots after X number of days if they are not modified.

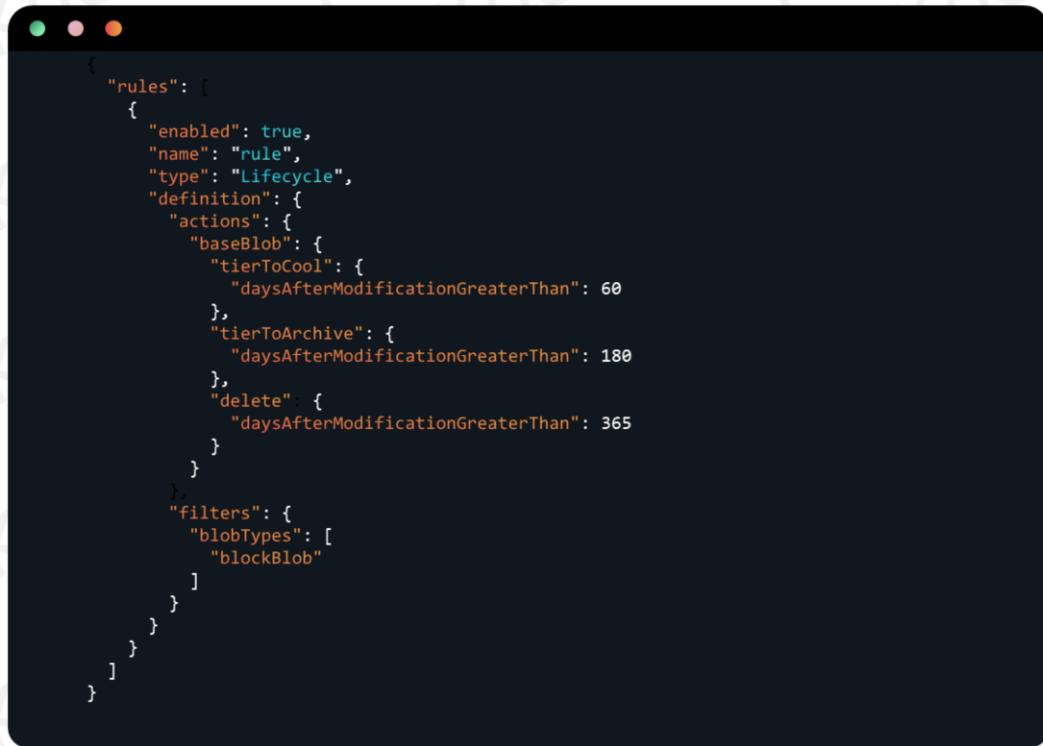
### **Filtering option**

We can apply the policy to all the blobs in the storage or limit blobs with filters

### **Target different types**

LCM can target block blobs and append blobs and further apply to sub types such as base blobs, versions and snapshots.

# Blob Lifecycle Management



```
{  
  "rules": [  
    {  
      "enabled": true,  
      "name": "rule",  
      "type": "Lifecycle",  
      "definition": {  
        "actions": {  
          "baseBlob": {  
            "tierToCool": {  
              "daysAfterModificationGreaterThan": 60  
            },  
            "tierToArchive": {  
              "daysAfterModificationGreaterThan": 180  
            },  
            "delete": {  
              "daysAfterModificationGreaterThan": 365  
            }  
          }  
        },  
        "filters": {  
          "blobTypes": [  
            "blockBlob"  
          ]  
        }  
      }  
    }  
  ]  
}
```



Policy-based transition



Delete blobs and snapshots



Filtering option



Target different types

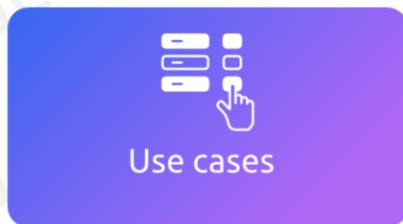
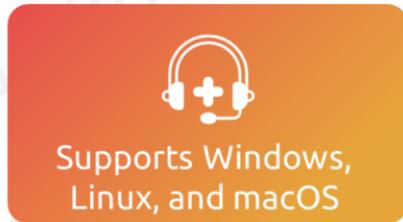


# Creating Azure File Share

© Copyright KodeKloud

- A repeatable way to deliver software and infrastructure code to its destination
- Build and test code
- Deploy to on-prem or cloud resources
- CICD isn't just for software developers. It's also for infrastructure pros

# Creating Azure File Share



**Connect**

cloudshell

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:

Authentication method:

Active Directory  
 Storage account key

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

© Copyright KodeKloud

## Enterprise grade file share

With file shares, we can share files across virtual machines and non-Azure workloads. Any number of Azure or non-Azure virtual machines can mount and work on the file share simultaneously. Also supports backup and snapshot for data recovery.

## Supports Windows, Linux and macOS

Azure provides easy to use scripts to mount the file share to Windows, Linux and macOS computers. Computers can interact with Azure File share as they work with on-premises file shares. Port 445 needs to be open for SMB traffic.

### **Use cases**

Firstly, we can decommission on-premises file share and migrate to Azure Files. It can be used for storing diagnostic data, tool and utilities which needs to be shared with teams.



# Securing storage endpoints

© Copyright KodeKloud

- A repeatable way to deliver software and infrastructure code to its destination
- Build and test code
- Deploy to on-prem or cloud resources
- CICD isn't just for software developers. It's also for infrastructure pros

# Securing Storage Endpoints

The screenshot shows the Azure Storage account settings page for 'alsoficshell' under the 'Networking' tab. The 'Private endpoint connections' tab is selected. A blue callout box at the top right says 'Set up Private Endpoint'. A purple callout box points to the 'Public network access' section, which includes options for 'Enabled from all networks' (radio button), 'Enabled from selected virtual networks and IP addresses' (radio button, selected), and 'Disabled'. A note below says 'Configure network security for your storage accounts.' A yellow callout box points to the 'Virtual networks' section, which lists 'Add existing virtual network' and 'Add new virtual network' buttons, and a table with columns for Virtual Network, Subnet, Address range, Endpoint Status, Resource Group, and Subscription. A pink callout box points to the 'Firewall' section, which allows adding IP ranges or client IP addresses, and an 'Address range' input field.

Set up Private Endpoint

Control public access to storage account

Restrict access to specific VNets using service endpoints

Allow IP ranges from internet or on-premises

Public network access

- Enabled from all networks
- Enabled from selected virtual networks and IP addresses
- Disabled

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

Virtual networks

Add existing virtual network Add new virtual network

Virtual Network	Subnet	Address range	Endpoint Status	Resource Group	Subscription
No network selected.					

Firewall

Add IP ranges to allow access from the internet or your on-premises networks. [Learn more](#).

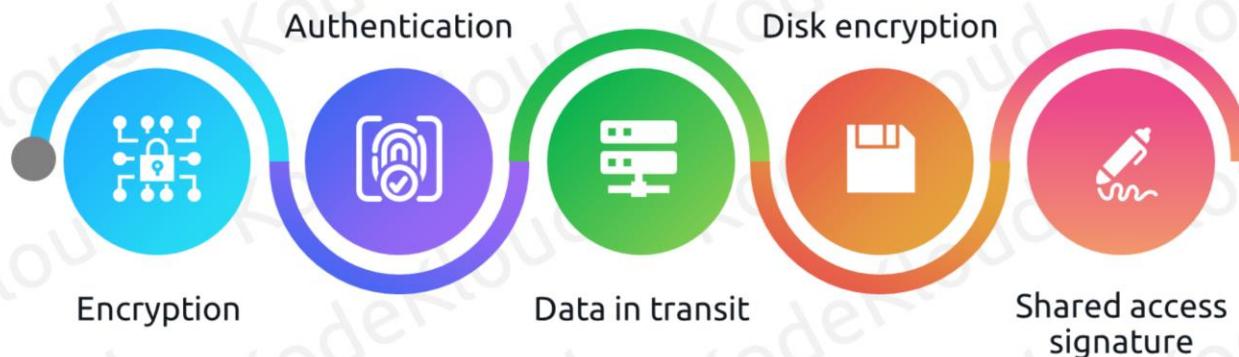
Add your client IP address ('117.216.20.36')

Address range

IP address or CIDR

© Copyright KodeKloud

# Storage Security Capabilities



© Copyright KodeKloud

## Encryption

By default, without any additional configuration, all data written to the storage account is encrypted by Storage Service Encryption (SSE)

## Authentication

With help of Azure AD and RBAC, we can authenticate and requests and provide authorization to storage services.

### **Data in transit**

Client-side encryption, HTTPS, and SMB 3.0 is used to secure data in transit.

### **Disk encryption**

OS and Data disks of Linux and Windows VMs can be encrypted using Azure Disk Encryption (ADE).

### **Shared access signature**

Fine tuned granular access can be given to storage services with the help of SAS.

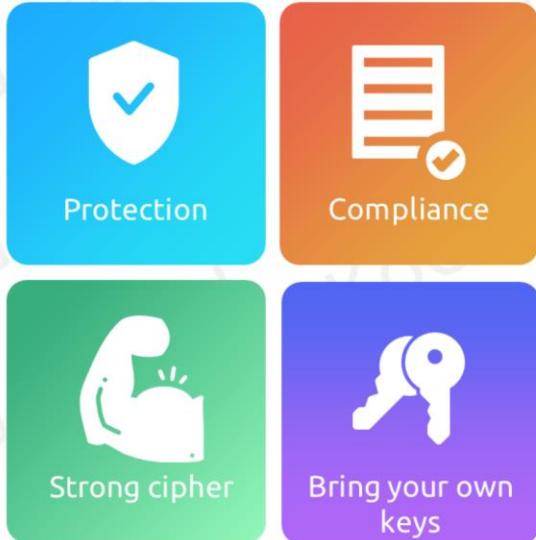


# Storage Service Encryption (SSE) and Azure Disk Encryption (ADE)

© Copyright KodeKloud

- A repeatable way to deliver software and infrastructure code to its destination
- Build and test code
- Deploy to on-prem or cloud resources
- CICD isn't just for software developers. It's also for infrastructure pros

# Storage Service Encryption (SSE)



**Encryption**   [Encryption scopes](#)

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.

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 selection**

Enable support for customer-managed keys [\(i\)](#)  
Blobs and files only

Infrastructure encryption [\(i\)](#)  
Disabled

Encryption type

Microsoft-managed keys  
 Customer-managed keys

© Copyright KodeKloud

## Protection

Data at rest is protected using SSE. All data written to Azure Disks, Blob, File, Queue, and Table is encrypted using SSE and is decrypted when the data is retrieved.

## Compliance

Organizations doesn't need develop in-house encryption methods to encrypt data stored in Azure storage. Using SSE

organizations can meet their compliance and security requirements.

### **Strong cipher**

SSE uses 256-bit AES encryption to encrypt the data. The encryption, decryption, data management and key management is done by storage service. SSE cannot be disabled.

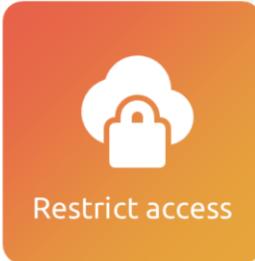
### **Bring your own keys**

If you would like to control the encryption keys and their rotation, you replace Microsoft managed keys with Customer Managed Keys. You need to create an Azure Key Vault to store the key and the storage service will retrieve the key from Key Vault for encryption and decryption.

# Azure Disk Encryption (ADE)



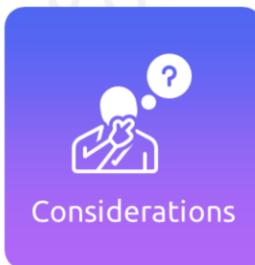
Encrypt disks



Restrict access



Encrypted  
backup



Considerations

The screenshot shows the 'Disk settings' dialog box for a virtual machine named 'dc-server'. It has sections for 'Ultra disk' (with 'Enable Ultra disk compatibility' set to 'No'), 'Encryption at host' (with 'Encryption at host' set to 'No'), and 'Encryption settings' (which includes a note about ADE providing volume encryption for OS and data disks). A dropdown menu under 'Disks to encrypt' shows options: 'None', 'None', 'OS disk', and 'OS and data disks'. At the bottom are 'Save' and 'Cancel' buttons.

© Copyright KodeKloud

## Encrypt disks

Using ADE, we can encrypt OS and Data Disks of Windows and Linux virtual machines. ADE uses BitLocker for Windows and DM-Crypt for Linux to encrypting the disks. Encryption keys are stored in Azure Key Vault.

## Restrict access

Since the disk is encrypted, only the VM owner will be able to retrieve the data stored in the VM. If anyone downloads

the VHD and attaches to another VM, without the keys, they will not be able to read the data.

### **Encrypted backup**

When you are using Azure Backup, the encryption keys are backed up to the recovery service vault. Also, the backups are encrypted. ASE uses AES 256-bit encryption.

### **Considerations**

If you are encrypting both OS and Data disk, there will be a small performance impact due to the encryption and decryption activity. The impact is very minimal, however, if your application is CPU intensive then you can skip the OS disk and encrypt data disk only to enhance performance.



# Configuring storage access

© Copyright KodeKloud

- A repeatable way to deliver software and infrastructure code to its destination
- Build and test code
- Deploy to on-prem or cloud resources
- CICD isn't just for software developers. It's also for infrastructure pros

# Configuring storage access



© Copyright KodeKloud

## Storage Account Keys

Two 512-bit keys will be generated for every storage account, and this can be rotated. Account keys are like root passwords, and we need to secure them to avoid unauthorized access.

## Shared access signature

Delegate access to storage at a very granular level. SAS are generated using account keys but with fine tuned access.

## **Azure AD**

Using Azure AD and RBAC we can authenticate and authorize requests from users. Currently Azure AD authentication is supported by Blobs, Queues, and Tables only. For Files, SMB access can be given with the help of AAD Domain Services.

## **Anonymous**

We can enable anonymous access to our blobs and containers. As the request is anonymous, we don't need pass any authorization header.

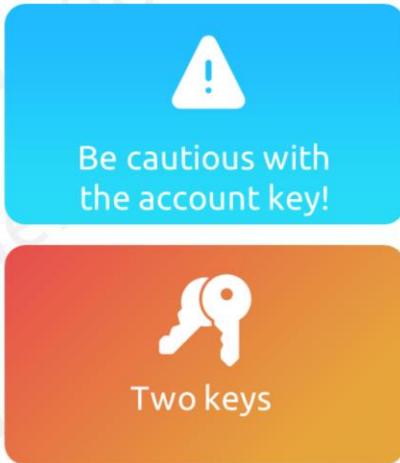


# Storage Account Keys

© Copyright KodeKloud

- A repeatable way to deliver software and infrastructure code to its destination
- Build and test code
- Deploy to on-prem or cloud resources
- CICD isn't just for software developers. It's also for infrastructure pros

# Storage Account Keys



The screenshot shows the Azure Storage Account Keys page. It displays two keys: key1 and key2. Key1 was last rotated 2 days ago and its key value is LQVU0JF+e0Bjg84X3Fw+pyl9I1J7ZbJsAvJZeD2YJiKTx9sdu90nWen6HFKWOqdpK4... . Its connection string is DefaultEndpointsProtocol=https;AccountName=alsoficssell;AccountKey=LQVU0... . Key2 was also last rotated 2 days ago and its key value is qRFvqNojdj3ISTWmLnZjjjXKXFczS7bFV5FtuDW+4Ircqy6iR3XuAXamd26cIK1BRG... . Its connection string is DefaultEndpointsProtocol=https;AccountName=alsoficssell;AccountKey=qRFvq... . Both keys have a "Rotate key" button next to them.

© Copyright KodeKloud

S

## Be cautious with the account key!

Account key is like the root password, the user possessing the account keys can perform any action against the storage account. Microsoft recommends to save the key to Azure Key Vault and regularly rotate them.

## Two keys

Azure provides two 512-bit keys for every storage account. You can either one of these in your API calls in your

authorization header. Users with permission to **Microsoft.Storage/storageAccounts/listkeys/action** can view, read or copy the key via Azure Portal, Azure CLI, and Azure PowerShell.

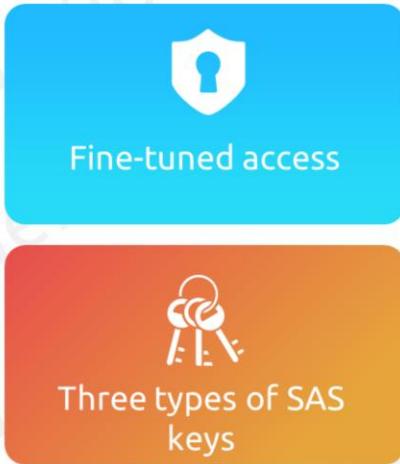


# Shared Access Signature

© Copyright KodeKloud

- A repeatable way to deliver software and infrastructure code to its destination
- Build and test code
- Deploy to on-prem or cloud resources
- CICD isn't just for software developers. It's also for infrastructure pros

# Shared Access Signature



Allowed services ⓘ  
 Blob  File  Queue  Table

Allowed resource types ⓘ  
 Service  Container  Object

Allowed permissions ⓘ  
 Read  Write  Delete  List  Add  Create  Update  Process  Immutable storage

Blob versioning permissions ⓘ  
 Enables deletion of versions

Allowed blob index permissions ⓘ  
 Read/Write  Filter

Start and expiry date/time ⓘ  
Start   12:01:40 PM  
End   8:01:40 PM

(UTC+05:30) Chennai, Kolkata, Mumbai, New Delhi

Allowed IP addresses ⓘ  
For example, 168.1.5.65 or 168.1.5.65-168.1.5.70

Allowed protocols ⓘ  
 HTTPS only  HTTPS and HTTP

© Copyright KodeKloud

S

## Fine tuned access

Instead of giving full access via account keys we can fine tune the access via SAS. We can control the allowed services, allowed resource types, permissions, start time, end time, IP address and protocol using SAS

## Three types of SAS keys

- ✓ User delegation SAS

- ✓ Service SAS
- ✓ Account SAS

## Shared Access Signature

URL

**Resource Endpoint**

<https://kodekloud.blob.core.windows.net>

**SAS token**

?sv=2020-08-04&ss=bfqt&srt=sc&sp=rwdlacup.....

# Shared Access Signature

Name	Excerpt	Explanation
Resource URI	<code>https://kodekloud.blob.core.windows.net</code>	Blob endpoint
Storage service version	<code>sv=2020-08-04</code>	Version of the storage service
Services	<code>ss=bftq</code>	SAS applies to blob, file, table and queue
Resource type	<code>srt=sc</code>	SAS applies to service and container level operations
Permissions	<code>sp=rwdlacup</code>	Supports read, write, delete, list, add, create, and update
Start time	<code>st=2022-05-19T06:31:40Z</code>	Start date and time in UTC
End time	<code>se=2022-05-19T14:31:40Z</code>	End date and time in UTC
IP address range	<code>sip=168.11.12.13-168.11.12.19</code>	Allowed IP range
Protocol	<code>spr=https</code>	Only HTTPS requests are allowed
Signature	<code>sig=66iXqzZSakarJO5J210%2ByoPRVXTeT%2FTJcHHSEkUjHr0%3D</code>	Unique signature which is HMAC computed over a string to sign and key using SHA256, then Base64 encoding on top of that.

© Copyright KodeKloud



## Fine tuned access

Instead of giving full access via account keys we can fine tune the access via SAS. We can control the allowed services, allowed resource types, permissions, start time, end time, IP address and protocol using SAS

## Three types of SAS keys

- ✓ User delegation SAS

- ✓ Service SAS
- ✓ Account SAS



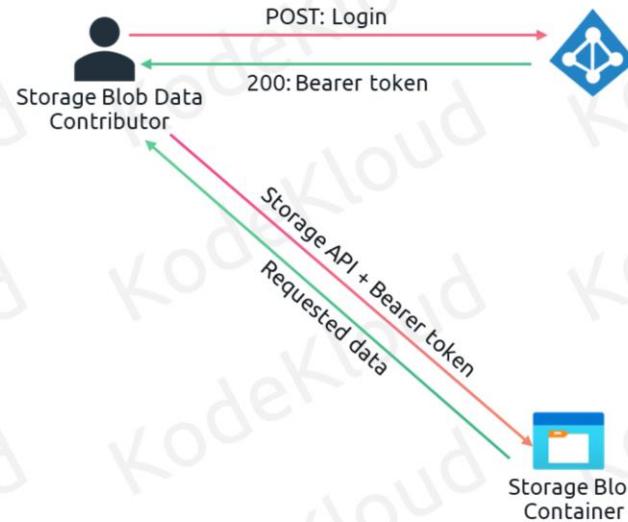
# Microsoft Entra ID Authentication

© Copyright KodeKloud

- A repeatable way to deliver software and infrastructure code to its destination
- Build and test code
- Deploy to on-prem or cloud resources
- CICD isn't just for software developers. It's also for infrastructure pros

# Microsoft Entra ID Authentication

-  Secure way of authenticating
-  Requires dedicated RBAC roles



© Copyright KodeKloud

## Secure way of authenticating

Microsoft recommends using Azure AD authentication for accessing Blobs, Queues and Tables. Azure AD integrates features such as MFA, Conditional Access to enhance the request to access storage.

## Requires dedicated RBAC roles

Even if you are the Owner or Contributor of the subscription, you would still require storage specific RBAC to authorize

storage access requests. These RBAC can be assigned to any scope and the access will be inherited. Example: Storage Blob Data Owner, Storage Queue Data Contributor.



# KodeKloud

© Copyright KodeKloud

Follow us on <https://kodekloud.com/> to learn more about us.