

Configure storage accounts

1. Introduction

<https://learn.microsoft.com/en-us/training/modules/configure-storage-accounts/1-introduction>

Introduction

Completed

Azure Storage is Microsoft's cloud storage solution for modern data storage scenarios.

Suppose you work for a large e-commerce company that needs to store and serve a vast number of product images to its customers. The company wants a scalable and reliable solution that can handle high traffic and ensure data durability. They want to quickly restore data if there's an outage.

In this module, you learn how to configure storage accounts and select appropriate storage types in Azure. The module covers topics such as implementing replication strategies, and configuring secure access to storage.

The goal of this module is to provide Azure Administrators with the knowledge and skills to effectively configure and manage Azure storage accounts.

Learning objectives

In this module, you learn how to:

- Identify features and usage cases for Azure storage accounts.
- Select between different types of Azure Storage and create storage accounts.
- Select a storage replication strategy.
- Configure secure network access to storage endpoints.

Skills measured

The content in the module helps you prepare for [Exam AZ-104: Microsoft Azure Administrator](#).

Prerequisites

- Experience with the Azure portal.
- Familiarity with managing different types of data storage.

2. Implement Azure Storage

<https://learn.microsoft.com/en-us/training/modules/configure-storage-accounts/2-implement-azure-storage>

Implement Azure Storage

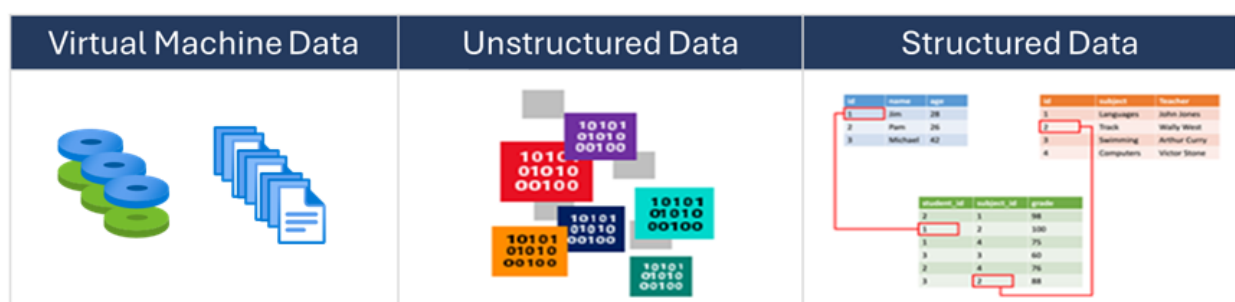
Completed

Azure Storage is Microsoft's cloud storage solution for modern data storage scenarios. Azure Storage offers a massively scalable object store for data objects. It provides a file system service for the cloud, a messaging store for reliable messaging, and a NoSQL store.

Azure Storage is an AI-ready service that you can use to store files, messages, tables, and other types of information. You use Azure Storage for applications like file shares. Developers use Azure Storage for working data. Working data includes websites, mobile apps, and desktop applications. Azure Storage is also used by IaaS virtual machines, and PaaS cloud services.

Things to know about Azure Storage

You can think of Azure Storage as supporting three categories of data: structured data, unstructured data, and virtual machine data. Review the following categories and think about which types of storage are used in your organization.



Category	Description	Storage examples
Virtual machine data	Virtual machine data storage includes disks and files. Disks are persistent block storage for Azure IaaS virtual machines. Files are fully managed file shares in the cloud.	Storage for virtual machine data is provided through Azure managed disks. Data disks are used by virtual machines to store data like database files, website static content, or custom application code. The number of data disks you can add depends on the virtual machine size.

Category	Description	Storage examples
Unstructured data	Unstructured data is the least organized. The format of unstructured data is referred to as <i>nonrelational</i> .	Unstructured data can be stored by using Azure Blob Storage and Azure Data Lake Storage. Blob Storage is a highly scalable, REST-based cloud object store. Azure Data Lake Storage is the Hadoop Distributed File System (HDFS) as a service.
Structured data	Structured data is stored in a relational format that has a shared schema. Structured data is often contained in a database table with rows, columns, and keys. Tables are an autoscaling NoSQL store.	Structured data can be stored by using Azure Table Storage, Azure Cosmos DB, and Azure SQL Database. Azure Cosmos DB is a globally distributed database service. Azure SQL Database is a fully managed database-as-a-service built on SQL.

Things to consider when using Azure Storage

As you think about your configuration plan for Azure Storage, consider these prominent features.

- **Consider durability and availability.** Azure Storage is durable and highly available. Redundancy ensures your data is safe during transient hardware failures. You replicate data across datacenters or geographical regions for protection from local catastrophe or natural disaster. Replicated data remains highly available during an unexpected outage.
- **Consider secure access.** Azure Storage encrypts all data. Azure Storage provides you with fine-grained control over who has access to your data.
- **Consider scalability.** Azure Storage is designed to be massively scalable to meet the data storage and performance needs of modern applications.
- **Consider manageability.** Microsoft Azure handles hardware maintenance, updates, and critical issues for you.
- **Consider data accessibility.** Data in Azure Storage is accessible from anywhere in the world over HTTP or HTTPS. Microsoft provides SDKs for Azure Storage in various languages. You can use .NET, Java, Node.js, Python, PHP, Ruby, Go, and the REST API. Azure Storage supports scripting in Azure PowerShell or the Azure CLI. The Azure portal and Azure Storage Explorer offer easy visual solutions for working with your data.
- **Consider SFTP support.** Blob Storage can use SFTP (SSH File Transfer Protocol), so you can keep using existing SFTP tools to move files directly to and from blobs. To use SFTP, enable hierarchical namespace (HNS). You can turn it on when you create the storage account (Advanced tab) or later under Settings → Configuration.
- **Consider NFSv3 protocol support.** Blob Storage can also be accessed using NFSv3, which lets Linux clients mount a container like an NFS share. NFSv3 can simplify migrations from Linux file workloads to Azure.

- **Consider default authorization preferences.** In the Azure portal, you can enable **Default to Microsoft Entra authorization**. This authentication makes role-based access control (RBAC) the default instead of shared access keys, which can improve security.

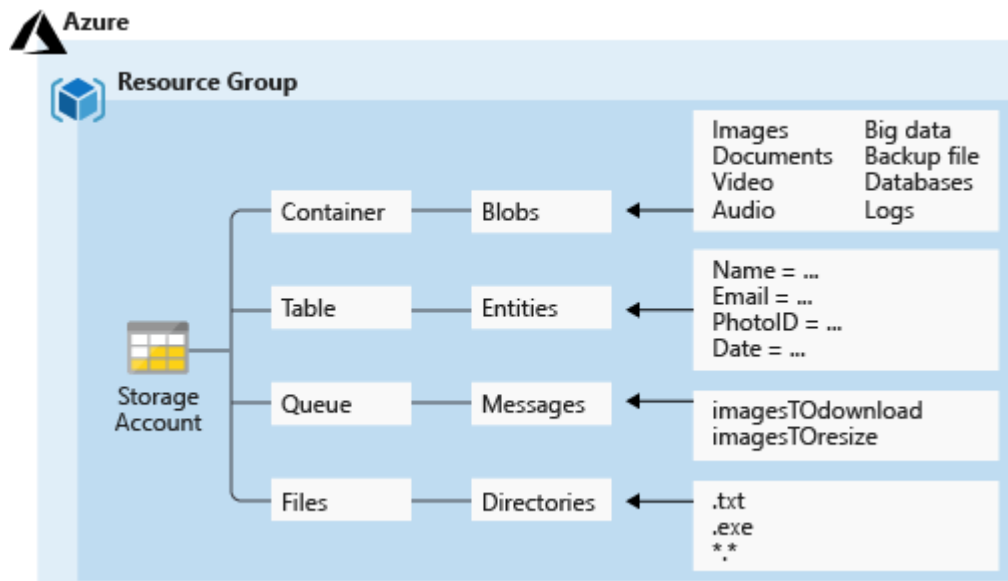
3. Explore Azure Storage services

<https://learn.microsoft.com/en-us/training/modules/configure-storage-accounts/3-explore-azure-storage-services>

Explore Azure Storage services

Completed

Let's examine the details of these services.



Azure Blob Storage

[Azure Blob Storage](#) is Microsoft's object storage solution for the cloud. Blob Storage is optimized for storing massive amounts of unstructured or *nonrelational* data, such as text or binary data. Blob

Storage is ideal for:

- 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, and archiving.
- Storing data for analysis by an on-premises or Azure-hosted service.

Objects in Blob Storage can be accessed from anywhere in the world via HTTP or HTTPS. Users or client applications can access blobs via URLs, the Azure Storage REST API, Azure PowerShell, the Azure CLI, or an Azure Storage client library. The storage client libraries are available for multiple languages, including .NET, Java, Node.js, Python, PHP, and Ruby.

Azure Files

[Azure Files](#) enables you to set up highly available network file shares. Shares can be accessed by using the Server Message Block (SMB) protocol and the Network File System (NFS) protocol. Multiple virtual machines can share the same files with both read and write access. You can also read the files by using the REST interface or the storage client libraries.

File shares can be used for many common scenarios:

- Many on-premises applications use file shares. This feature makes it easier to migrate those applications that share data to Azure. If you mount the file share to the same drive letter that the on-premises application uses, the part of your application that accesses the file share should work with minimal, if any, changes.
- Configuration files can be stored on a file share and accessed from multiple virtual machines. Tools and utilities used by multiple developers in a group can be stored on a file share, ensuring that everybody can find them, and that they use the same version.
- Diagnostic logs, metrics, and crash dumps are just three examples of data that can be written to a file share and processed or analyzed later.

The storage account credentials are used to provide authentication for access to the file share. All users who have the share mounted should have full read/write access to the share.

Azure Queue Storage

[Azure Queue Storage](#) is used to store and retrieve messages. Queue messages can be up to 64 KB in size, and a queue can contain millions of messages. Queues are used to store lists of messages to be processed asynchronously.

Consider a scenario where you want your customers to be able to upload pictures, and you want to create thumbnails for each picture. You could have your customer wait for you to create the thumbnails while uploading the pictures. An alternative is to use a queue. When the customer finishes the upload, you can write a message to the queue. Then you can use an Azure Function to retrieve the message from the queue and create the thumbnails. Each of the processing parts can be scaled separately, which gives you more control when tuning the configuration.

Azure Table Storage

[Azure Table storage](#) is a service that stores nonrelational structured data (also known as structured NoSQL data) in the cloud, providing a key/attribute store with a schemaless design. Because Table storage is schemaless, it's easy to adapt your data as the needs of your application evolve. Access to Table storage data is fast and cost-effective for many types of applications, and is typically lower in cost than traditional SQL for similar volumes of data. In addition to the existing Azure Table Storage service, there's a new Azure Cosmos DB Table API offering that provides throughput-optimized tables, global distribution, and automatic secondary indexes.

Things to consider when choosing Azure Storage services

As you think about your configuration plan for Azure Storage, consider the prominent features of the types of Azure Storage and which options support your application needs.

- **Consider storage optimization for massive data.** Azure Blob Storage is optimized for storing massive amounts of unstructured data. Objects in Blob Storage can be accessed from anywhere in the world via HTTP or HTTPS. Blob Storage is ideal for serving data directly to a browser, streaming data, and storing data for backup and restore.
- **Consider storage with high availability.** Azure Files supports highly available network file shares. On-premises apps use file shares for easy migration. By using Azure Files, all users can access shared data and tools. Storage account credentials provide file share authentication to ensure all users who have the file share mounted have the correct read/write access.
- **Consider storage for messages.** Use Azure Queue Storage to store large numbers of messages. Queue Storage is commonly used to create a backlog of work to process asynchronously.
- **Consider storage for structured data.** Azure Table Storage is ideal for storing structured, nonrelational data. It provides throughput-optimized tables, global distribution, and automatic secondary indexes. Because Azure Table Storage is part of Azure Cosmos DB, you have access to a fully managed NoSQL database service for modern app development.

4. Determine storage account types

<https://learn.microsoft.com/en-us/training/modules/configure-storage-accounts/4-determine-storage-account-kinds>

Determine storage account types

Completed

General purpose Azure storage accounts have two basic [types](#): Standard and Premium.

Things to know about storage account types

Standard storage accounts are backed by magnetic hard disk drives (HDD). A standard storage account provides the lowest cost per GB. You can use Standard storage for applications that require bulk storage or where data is infrequently accessed.

Premium storage accounts are backed by solid-state drives (SSD) and offer consistent low-latency performance. You can use Premium storage for Azure virtual machine disks with I/O-intensive applications like databases.

Note

You can't convert a Standard storage account to a Premium storage account or vice versa. You must create a new storage account with the desired type and copy data, if applicable, to a new storage account. All storage account types are encrypted by using Storage Service Encryption (SSE) for data at rest.

Storage account	Supported services	Redundancy options	Recommended usage
Standard general-purpose v2	Blob Storage (including Data Lake Storage), Queue Storage, Table Storage, and Azure Files	LRS, GRS, RA-GRS, ZRS, GZRS, RA-GZRS	Standard storage account for most scenarios, including blobs, file shares, queues, tables, and disks (page blobs).
Premium block blobs	Blob Storage (including Data Lake Storage)	LRS, ZRS	Premium storage account for block blobs and append blobs. Recommended for applications with high transaction rates. Use Premium block blobs if you work with smaller objects or require consistently low storage latency. This storage is designed to scale with your applications.
Premium file shares	Azure Files	LRS, ZRS	Premium storage account for file shares only. Recommended for enterprise or high-performance scale

Storage account	Supported services	Redundancy options	Recommended usage
			applications. Use Premium file shares if you require support for both Server Message Block (SMB) and NFS file shares.
Premium page blobs	Page blobs only	LRS only	Premium high-performance storage account for page blobs only. Page blobs are ideal for storing index-based and sparse data structures, such as operating systems, data disks for virtual machines, and databases.

Note

Administrators managing existing Azure subscriptions may encounter legacy storage account types such as General-purpose v1 (GPv1) and legacy BlobStorage accounts. Microsoft recommends upgrading legacy accounts to General-purpose v2 for access to all current capabilities. Upgrades are supported in-place via the Azure portal, Azure CLI, or PowerShell.

Tip

Before continuing, consider working through the [Create a storage account](#) training module.

5. Determine replication strategies

<https://learn.microsoft.com/en-us/training/modules/configure-storage-accounts/5-determine-replication-strategies>

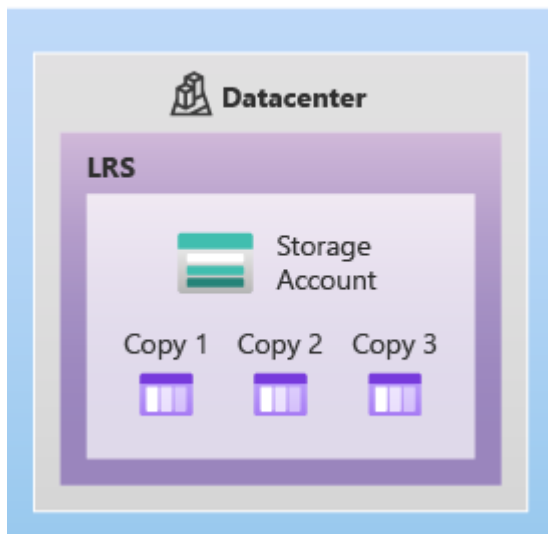
Determine replication strategies

Completed

The data in your Azure storage account is always replicated to ensure durability and high availability. [Azure Storage replication](#) copies your data to protect from planned and unplanned events. These events range from transient hardware failures, network or power outages, massive natural disasters, and so on. You can choose to replicate your data within the same data center, across zonal data centers within the same region, and even across regions. Replication ensures your storage account meets the Service-Level Agreement (SLA) for Azure Storage even if there are failures.

Locally redundant storage

Primary region

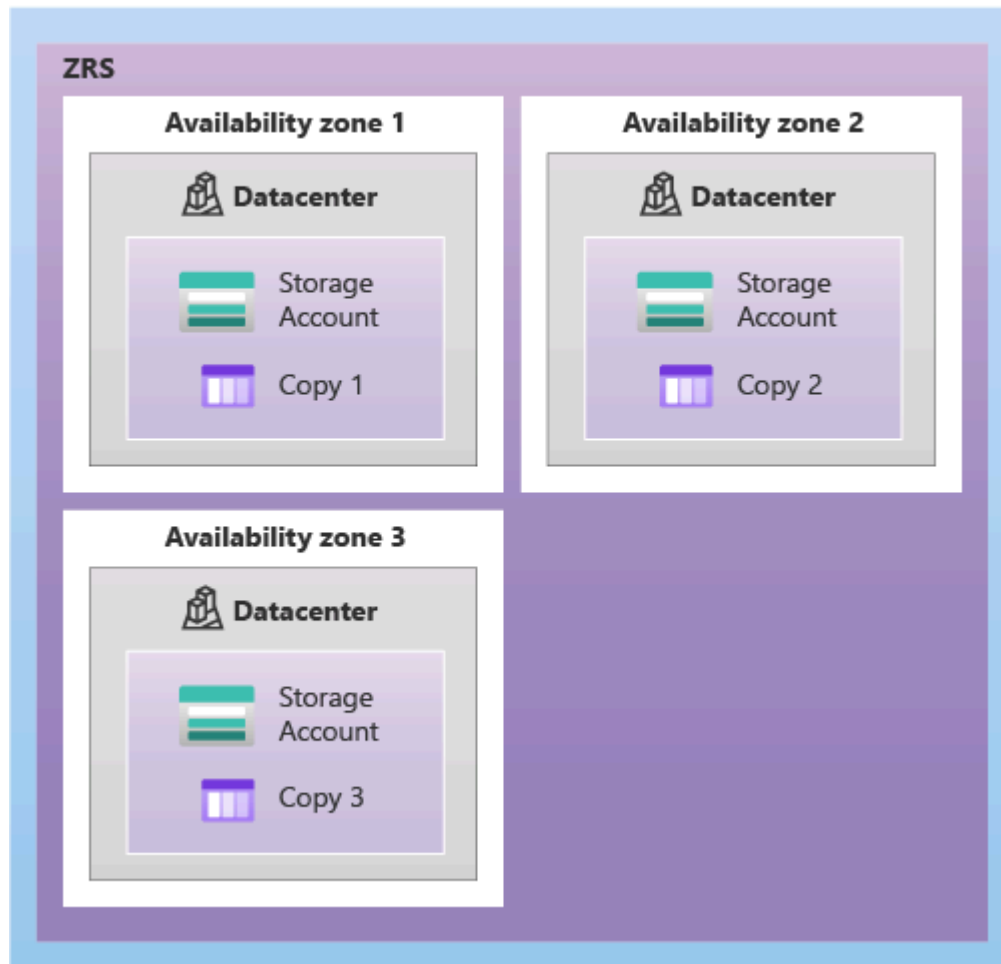


Locally redundant storage is the lowest-cost replication option and offers the least durability compared to other strategies. If a data center-level disaster occurs, such as fire or flooding, all replicas might be lost or unrecoverable. Despite its limitations, LRS can be appropriate in several scenarios:

- Your application stores data that can be easily reconstructed if data loss occurs.
- Your data is constantly changing like in a live feed, and storing the data isn't essential.
- Your application is restricted to replicating data only within a location due to data governance requirements.

Zone redundant storage

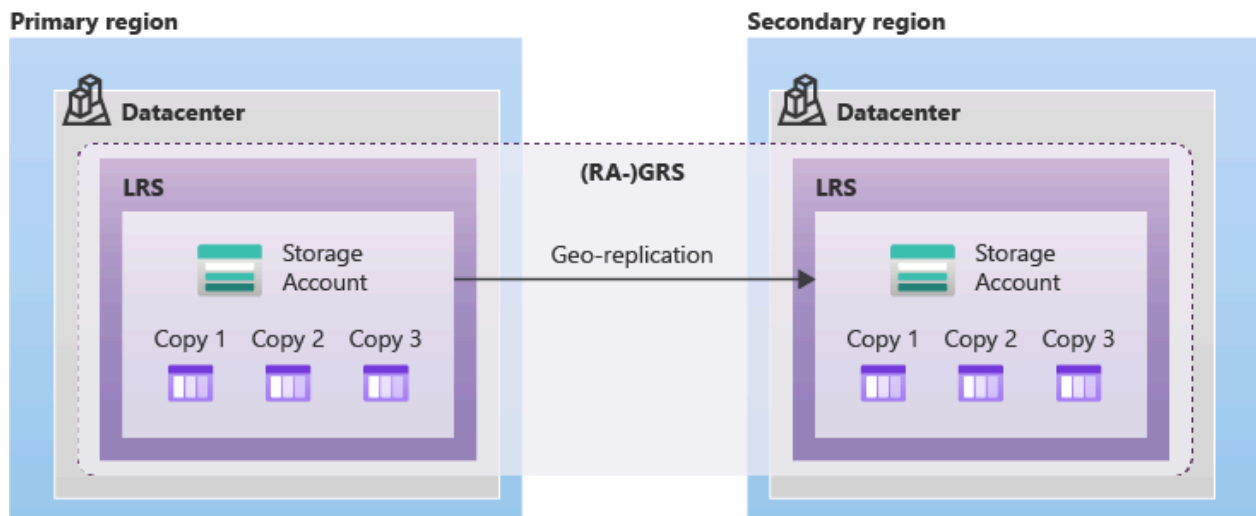
Primary region



Zone redundant storage synchronously replicates your data across three storage clusters in a single region. Each storage cluster is physically separated from the others and resides in its own availability zone. Each availability zone, and the ZRS cluster within it, is autonomous, and has separate utilities and networking capabilities. Storing your data in a ZRS account ensures you can access and manage your data if a zone becomes unavailable. ZRS provides excellent performance and low latency.

- ZRS isn't currently available in all regions.
- Changing to ZRS from another data replication option requires the physical data movement from a single storage stamp to multiple stamps within a region.

Geo-redundant storage



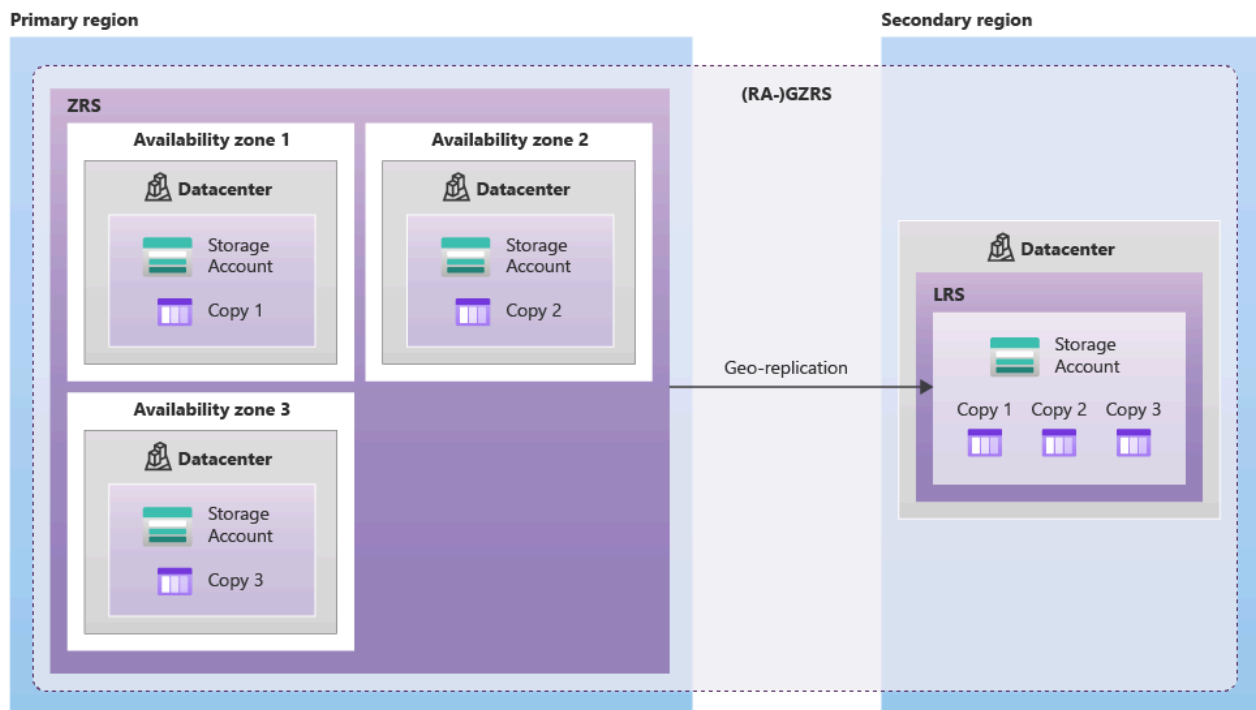
Geo-redundant storage replicates your data to a secondary region (hundreds of miles away from the primary location of the source data). GRS provides a higher level of durability even during a regional outage. GRS is designed to provide at least 99.99999999999999% **(16 9's) durability**. When your storage account has GRS enabled, your data is durable even when there's a complete regional outage or a disaster where the primary region isn't recoverable.

If you implement GRS, you have two related options to choose from:

- **GRS** replicates your data to another data center in a secondary region. The data is available to be read only if Microsoft initiates a failover from the primary to secondary region.
- **Read-access geo-redundant storage (RA-GRS)** is based on GRS. RA-GRS replicates your data to another data center in a secondary region, and also provides you with the option to read from the secondary region. With RA-GRS, you can read from the secondary region regardless of whether Microsoft initiates a failover from the primary to the secondary.

For a storage account with GRS or RA-GRS enabled, all data is first replicated with locally redundant storage. An update is first committed to the primary location and replicated by using LRS. The update is then replicated asynchronously to the secondary region by using GRS. Data in the secondary region uses LRS. Both the primary and secondary regions manage replicas across separate fault domains and upgrade domains within a storage scale unit. The storage scale unit is the basic replication unit within the datacenter. Replication at this level is provided by LRS.

Geo-zone redundant storage



Geo-zone-redundant storage combines the high availability of zone-redundant storage with protection from regional outages as provided by geo-redundant storage. Data in a GZRS storage account is replicated across three Azure availability zones in the primary region, and also replicated to a secondary geographic region for protection from regional disasters. Each Azure region is paired with another region within the same geography, together making a regional pair.

With a GZRS storage account, you can continue to read and write data if an availability zone becomes unavailable or is unrecoverable. Additionally, your data is also durable during a complete regional outage or during a disaster in which the primary region isn't recoverable. GZRS is designed to provide at least 99.99999999999999% (16 9's) durability of objects over a given year. GZRS also offers the same scalability targets as LRS, ZRS, GRS, or RA-GRS. You can optionally enable read access to data in the secondary region with read-access geo-zone-redundant storage (RA-GZRS).

Tip

Microsoft recommends using GZRS for applications that require consistency, durability, high availability, excellent performance, and resilience for disaster recovery. Enable RA-GZRS for read access to a secondary region when there's a regional disaster.

Things to consider when choosing replication strategies

Let's examine the scope of durability and availability for the different replication strategies. The following table describes several key factors during the replication process, including node unavailability within a data center, and whether the entire data center (zonal or nonzonal) becomes unavailable. The table identifies read access to data in a remote, geo-replicated region during region-wide unavailability, and the supported Azure storage account types.

Node in data center unavailable	Entire data center unavailable	Region-wide outage	Read access during region-wide outage
- LRS - ZRS - GRS - RA-GRS - GZRS - RA-GZRS	- ZRS - GRS - RA-GRS - GZRS - RA-GZRS	- GRS - RA-GRS - GZRS - RA-GZRS	- RA-GRS - RA-GZRS

6. Access storage

<https://learn.microsoft.com/en-us/training/modules/configure-storage-accounts/6-access-storage>

Access storage

Completed

Every object you store in Azure Storage has a unique URL address. Your storage account name forms the *subdomain* portion of the URL address. The combination of the subdomain and the domain name, which is specific to each service, forms an endpoint for your storage account.

Let's look at an example. If your storage account name is *mystorageaccount*, default endpoints for your storage account are formed for the Azure services as shown in the following table:

Service	Default endpoint
Container service	<code>// mystorageaccount .blob.core.windows.net</code>
Table service	<code>// mystorageaccount .table.core.windows.net</code>
Queue service	<code>// mystorageaccount .queue.core.windows.net</code>
File service	<code>// mystorageaccount .file.core.windows.net</code>

We create the URL to access an object in your storage account by appending the object's location in the storage account to the endpoint.

For example, to access the *myblob* data in the *mycontainer* location in your storage account, we use the following URL address:

`// mystorageaccount .blob.core.windows.net/ mycontainer / myblob .`

Configure custom domains

You can configure a [custom domain](#) to access blob data in your Azure storage account. As we reviewed, the default endpoint for Azure Blob Storage is `\<storage-account-name>.blob.core.windows.net`. If you map a custom domain and subdomain, such as `www.contoso.com`, to the blob or web endpoint for your storage account, your users can use that domain to access blob data in your storage account.

Direct mapping lets you enable a custom domain for a subdomain to an Azure storage account. For this approach, you create a `CNAME` record that points from the subdomain to the Azure storage account.

The following example shows how a subdomain is mapped to an Azure storage account to create a `CNAME` record in the domain name system (DNS):

- Subdomain: `blobs.contoso.com`
- Azure storage account: `\<storage account>.blob.core.windows.net`
- Direct `CNAME` record: `contosoblobs.blob.core.windows.net`

7. Secure storage endpoints

<https://learn.microsoft.com/en-us/training/modules/configure-storage-accounts/7-secure-storage-endpoints>

Secure storage endpoints

Completed

In the Azure portal, each Azure service requires certain steps to configure the service endpoints and restrict network access.

To access these settings for your storage account, you use the **Firewalls and virtual networks** settings. You add the virtual networks that should have access to the service for the account. - This setting restricts access to your storage account from specific subnets on virtual networks or public IPs.

Security + networking

Networking

Azure CDN

Access keys

Firewalls and virtual networks

Custom domain

Save Discard Refresh

Public network access

☒ Enabled from all networks

☐ Enabled from selected virtual networks and IP addresses

☐ Disabled

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

The service endpoints for a storage account provide the base URL for any blob, queue, table, or file object in Azure Storage. Use this base URL to construct the address for any given resource.

Search (Ctrl+ /)

Refresh

Settings

Configuration

Data Lake Gen2 upgrade

Resource sharing (CORS)

Advisor recommendations

Endpoints

Locks

Monitoring

Insights

Alerts

Metrics

Workbooks

Diagnostic settings

Logs

Provisioning state Succeeded

Created 5/5/2022, 8:27:25 PM

Storage account resource ID /subscriptions/ <subscription-id> ..

Blob service

Resource ID /subscriptions/ <subscription-id> ..

Blob service https://storagesamplesdnszone4.z10.blob.storage.a: ..

File service

Resource ID /subscriptions/ <subscription-id> ..

File service https://storagesamplesdnszone4.z10.file.storage.azl ..

Queue service

Resource ID /subscriptions/ <subscription-id> ..

Queue service https://storagesamplesdnszone4.z10.queue.storage. ..

Things to know about configuring service endpoints

Here are some points to consider about configuring service access settings:

- You can configure the service to allow access to one or more public IP ranges.
- Subnets and virtual networks must exist in the same Azure region or region pair as your storage account.

Important

Be sure to test the service endpoint and verify the endpoint limits access as expected.

Things to know about configuring private endpoints

In addition to service endpoints, Azure Storage supports private endpoints for enhanced security and network isolation. Private endpoints are the recommended approach for production workloads requiring secure access.

A private endpoint uses a private IP address from your virtual network to bring the Azure Storage service into your VNet. All traffic between your VNet and the storage service goes over the Microsoft backbone network, eliminating exposure to the public internet.

Key differences from service endpoints

- Private endpoints assign a private IP from your VNet to the storage account, keeping all traffic within the Microsoft backbone. Use private endpoints for production workloads requiring complete network isolation and compliance requirements
- Service endpoints keep the storage account on its public endpoint but restrict access to specific VNets and subnets. Use service endpoints for development scenarios or when you need simpler configuration with some public internet access

Tip

Learn more with the [Secure and isolate access to Azure resources by using network security groups and service endpoints](#) training module. This module has a sandbox where you can restrict access to Azure Storage by using service endpoints.

8. Module assessment

<https://learn.microsoft.com/en-us/training/modules/configure-storage-accounts/8-knowledge-check>

Module assessment

Completed

9. Summary and resources

<https://learn.microsoft.com/en-us/training/modules/configure-storage-accounts/9-summary-resources>

Summary and resources

Completed

In this module, you learned about Azure Storage and how to create a storage account.

The main takeaways from this module are:

- Azure Storage provides a range of storage options for different types of data, including virtual machine data, unstructured data, and structured data.
- There are different types of storage accounts available, each with its own features and pricing model. It's important to consider the specific requirements of your application when choosing the right storage account type.
- Azure Storage offers four data services: Azure Blob Storage, Azure Files, Azure Queue Storage, and Azure Table Storage. Each service is optimized for different types of data and has its own use cases and benefits.
- Replication is an important consideration for ensuring data durability and high availability. Azure Storage offers different replication strategies to choose from based on your requirements.
- Configuring custom domains and secure endpoints allow you to access and secure your storage account in Azure.

Learn more with Copilot

Copilot can assist you in configuring Azure infrastructure solutions. Copilot can compare, recommend, explain, and research products and services where you need more information. Open a Microsoft Edge browser and choose Copilot (top right) or navigate to copilot.microsoft.com. Take a few minutes to try these prompts and extend your learning with Copilot.

- What is an Azure storage account? What type of storage accounts are available?
- Explain for a nontechnical person Azure data redundancy for storage accounts.

Learn more with Azure documentation

- [Storage account overview](#). This article is your starting point for learning about Azure storage accounts.
- [Azure storage redundancy](#). This article reviews how to tradeoff cost and availability when selecting a redundancy option.

Learn more with self-paced training

- [Create an Azure storage account](#). Learn how to create an Azure Storage account with the correct options for your business needs.
- [Design and implement private access to Azure Services](#). Learn how to implement private access to Azure Services with Azure Private Link, and virtual network service endpoints.