

DEVOPS with MULTI-CLOUD

Practice Tasks

Institute Name : V Cube software solutions
Course : DevOps with Multi-Cloud
Batch : 30
Trainer : Krishna reddy sir

Prepared by : G.Bhavish
(MCD-AZ30-024)

TASK-16 :- Service & Private Endpoints. (Storage Account).

Date : 09/02/26

Objective :-

To securely connect Azure services to a Virtual Network by restricting access over the Azure backbone network using Service Endpoints and enabling private access through a private IP address using Private Endpoints, thereby preventing exposure to the public internet.

Service Endpoint :-

A Service Endpoint extends a Virtual Network to Azure services, allowing secure communication over the Azure backbone network without using the public internet.

Private Endpoint :-

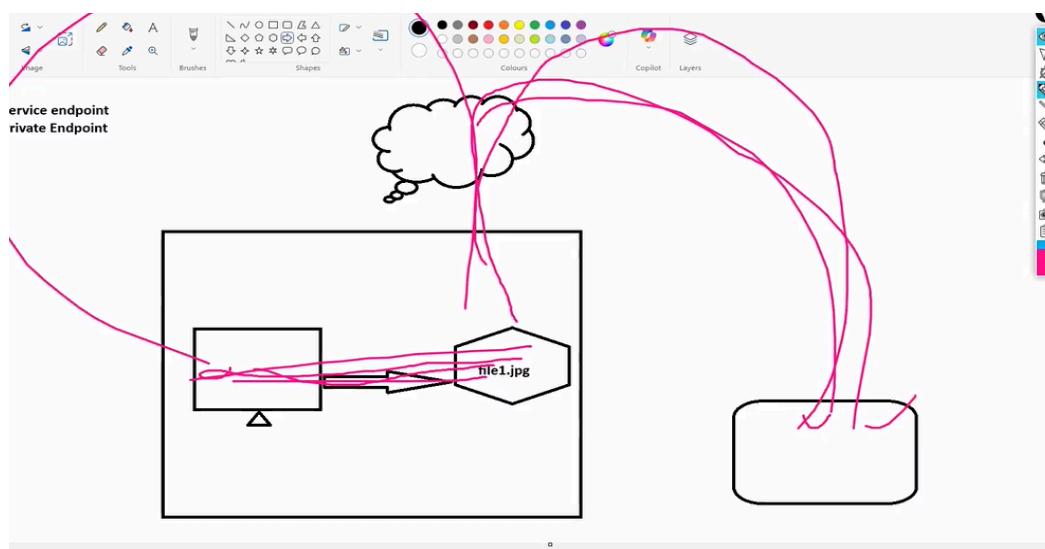
A Private Endpoint provides a private IP address from your Virtual Network to an Azure service, enabling secure access without exposing the service publicly.

→ When we create a Storage Account, it is created by a default public endpoint. If we access storage account with a public endpoint, we are connected over the internet.

→ if we connect over the internet, we will be exposed and our data will not be safe.

→ so for the internal connection we need to enable **“Service end point”**.

→ whenever we enable the service end point, it uses the backbone network i.e internal network for connecting to the storage account.



→ connecting to a storage account by a

Internal vm - Intranet.

Outsiders - Internet.

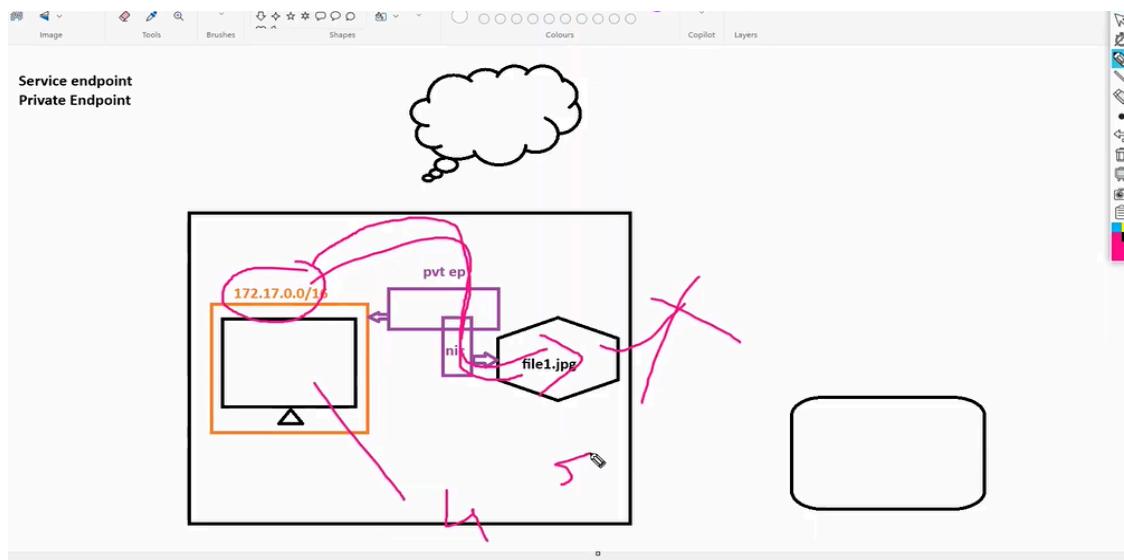
→ we can internally transfer the data to a storage account by using service endpoint. And outsiders can also access the data over the internet.

● **Note** :-

- We can configure service & private endpoint to any PaaS model
- We can find our ipv4&v6 in whatismyipaddress.com .

→ Unlike the service endpoint, for completely blocking the outside internet access to storage account we use the **“Private Endpoint”**.

→ when we configure a private end point ,one private ip is attached to our storage account so that we can access it internally.



→ Once we create a private endpoint

- Get one nic card which is attached to storage account
- The pub ip will get deleted.
- Picks one ip address from subnet range.
- In the backend one private dns is created.

→ And the dns maps the name with the pvt-ip.

→ with this private ip we can connect to a storage account internally & also blocks other connections.

“Service endpoint - for internal & outsider connection”.

“Private endpoint - for internal & more secure connection”.

→ create a vm and storage account, also create a container and add an image.

The screenshot shows the Microsoft Azure portal interface. The main title bar says "Microsoft Azure" and "Search resources, services, and docs (G+)". The user's email "gilakarabablu@gmail.com" is at the top right. The page title is "Home > CreateVm-MicrosoftWindowsServer.WindowsServer-202-20260210134901 | Overview". On the left, there's a sidebar with "Overview" selected, along with other options like Activity log, Access control (IAM), Tags, Diagnose and solve problems, Monitor, Resource visualizer, Favorites, Connect, Networking, and Network settings. The main content area is titled "Essentials" and displays the following details:

Resource group	Operating system
STORAGE-RG	Windows (Windows Server 2025 Datacenter Azure Edition)
Status	Size
Running	Standard E2s v3 (2 vcpus, 16 GiB memory)
Location	Primary NIC public IP
Canada Central	52.138.17.179
Subscription	1 associated public IPs
Azure subscription 1	VN01/SN01
Subscription ID	DNS name
24c251a0-1cbb-4a97-8576-0fd76172d25c	Not configured

Fig (1) created a windows machine.

The screenshot shows the Microsoft Azure portal interface. The main title bar says "Microsoft Azure" and "Search resources, services, and docs (G+)". The user's email "gilakarabablu@gmail.com" is at the top right. The page title is "Home > bhavish629 - Microsoft Azure | Overview". On the left, there's a sidebar with "Overview" selected, along with other options like Activity log, Tags, Diagnose and solve problems, Access Control (IAM), Data migration, Events, Storage browser, Storage Mover, Partner solutions, Resource visualizer, and Data storage. The main content area is titled "Essentials" and displays the following details:

Resource group	Performance
STORAGE-RG	Standard
Location	Replication
canadacentral	Locally-redundant storage (LRS)
Subscription	Account kind
Azure subscription 1	StorageV2 (general purpose v2)
Subscription ID	Provisioning state
24c251a0-1cbb-4a97-8576-0fd76172d25c	Succeeded
Disk state	Created
Available	10/02/2026, 13:51:26

Fig (2) created a storage account.

→ create a container and add an image, try to access the image by browsing.

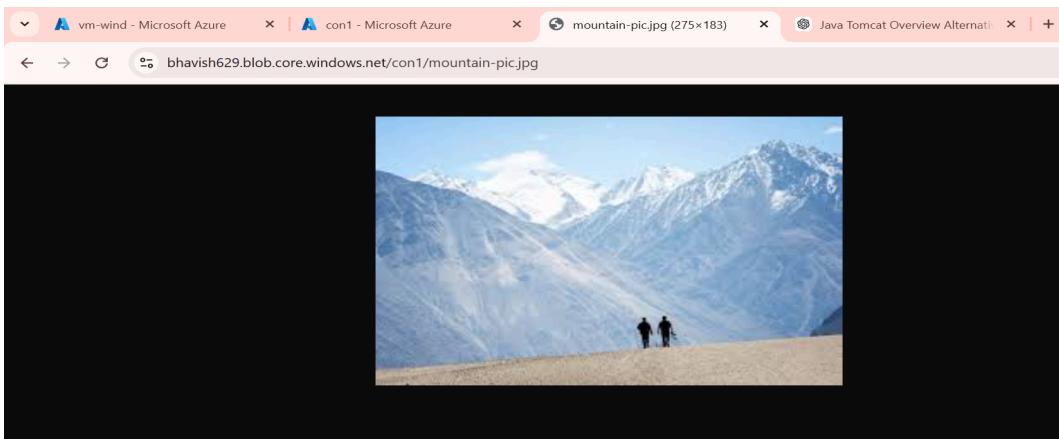
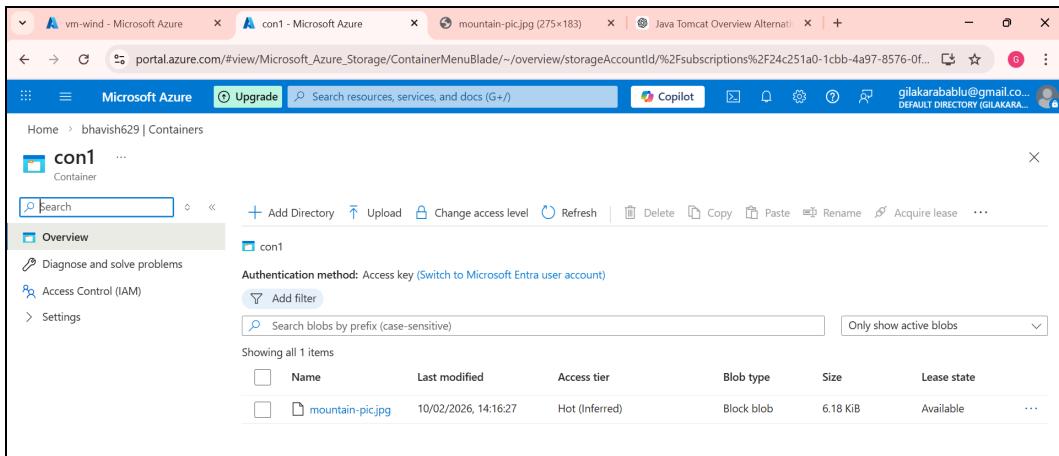


Fig (3&4) created a container and added an image.
→ Now let's configure the service endpoint.

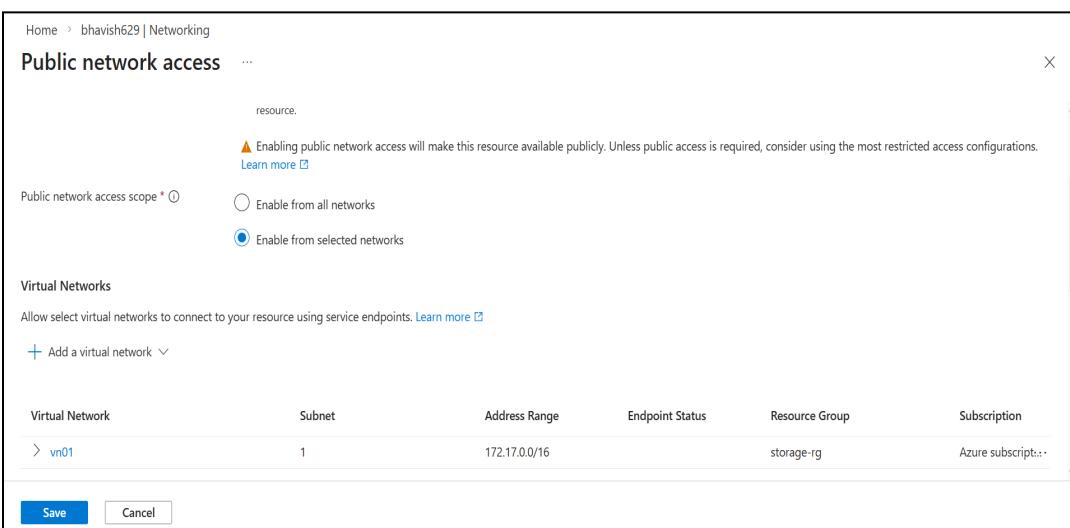


Fig (5) enabling service endpoint by selected networks.

→ the service endpoint is enabled at subnet level. Here we enabled selected networks and added machines vnet. Now only the machine can access the file.

→ we can also add our laptop ip address , so that we can also access the file.(add laptop ip in client ip.)

→ Now let's create a Private Endpoint.

- Search private endpoint and create,we can give the details in the basics, resources and vn.

The screenshot shows the Microsoft Azure portal's Overview page for a Private Endpoint named "storage-nic". The page has a left sidebar with links for Activity log, Access control (IAM), Tags, Diagnose and solve problems, Resource visualizer, Settings, Monitoring, Automation, and Help. The main content area is titled "Overview" and contains the following details:

Essentials	Value
Resource group	(move) STORAGE-RG
Location	Canada Central
Subscription	(move) Azure subscription 1
Subscription ID	24c251a0-1ccb-4a97-8576-0fd76172d25c
Provisioning state	Succeeded
Tags	(edit) Add tags

On the right side, there are additional details:

- Virtual network/subnet: VN01/SN01
- Network interface: storage-nic-nic
- Private link resource: bhavish629
- Target sub-resource: blob
- Connection status: Approved
- Request/Response: Auto-Approved

fig(6) created a private endpoint.

→ Now only the vm can access the file in the storage account.

→ There are four ways to connect to the storage account.

- Portal Access - connecting by azure portal.
- Access keys

- SAS Keys
- RBAC Rules.

→ Access Keys :-

- Using access keys we can connect to the storage account, we can add and download data.
- The issue is it gives all the admin rights on the storage account.

→ SAS Keys :-

- Shared Access Signature, used to restrict the access like which data storage can be accessed.
- Eg:- only blob, only file and permissions like read, write, which ip address and how much time also.

→ RBAC Rules :-

- Roll Based Access Control, used to restrict even more than sas keys.
- Can give access to a particular user to a particular role with permissions & restrictions.
- Eg:- Sr.engg : Admin/Owner
Jr.engg : Can be accessed only till his work or job role.

NOTE :-

1. SAS Keys can restrict and give permissions to all the users.
2. RBAC Rules can restrict & give permission to a particular person and to only his work.

→ the above three ways(access keys,sas keys and rbac rules) used to connect to the storage account via the “Storage Explorer”.

Note :- before using these keys and rules enable to all networks from selected networks.

→ Now download the Storage Explorer in the vm, and connect to storage account via keys and rules.

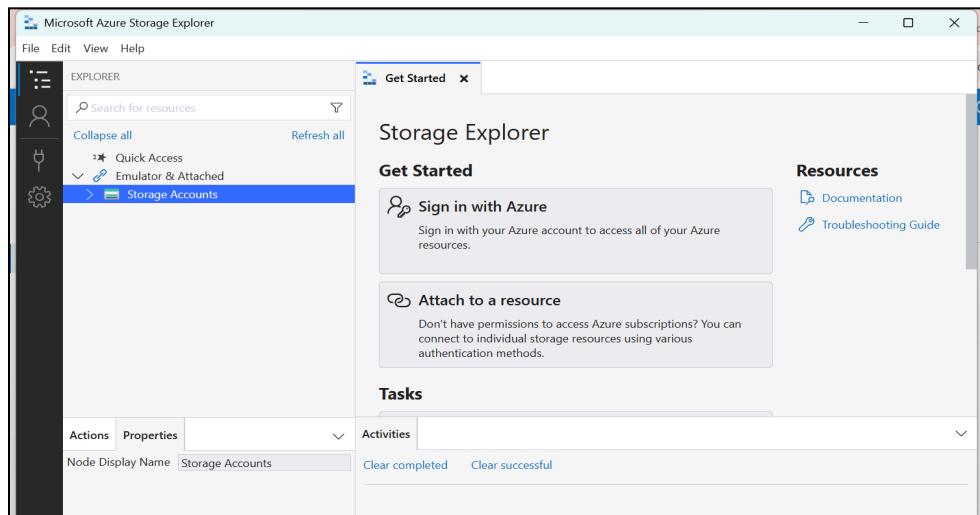


Fig (7) downloaded storage explorer.

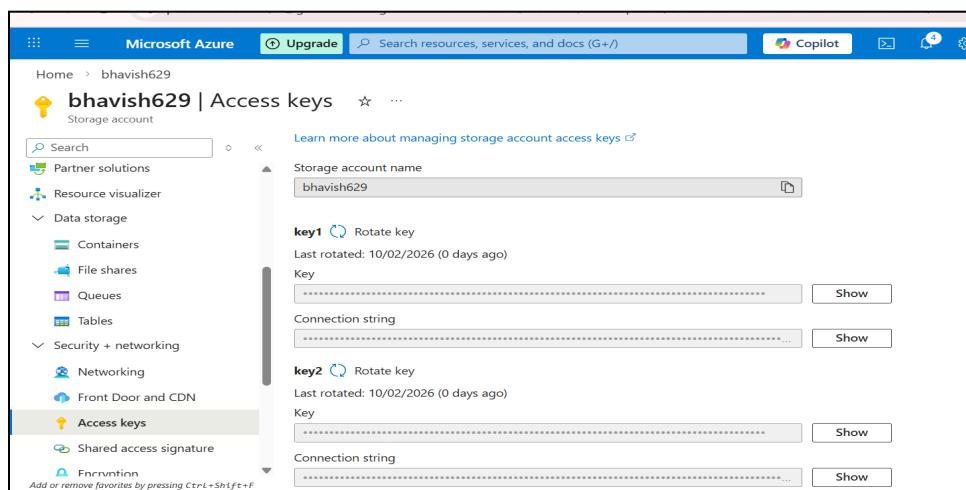


Fig (8) Access keys and connecting string.

→ use the access keys and connecting string and account name to connect the storage account.

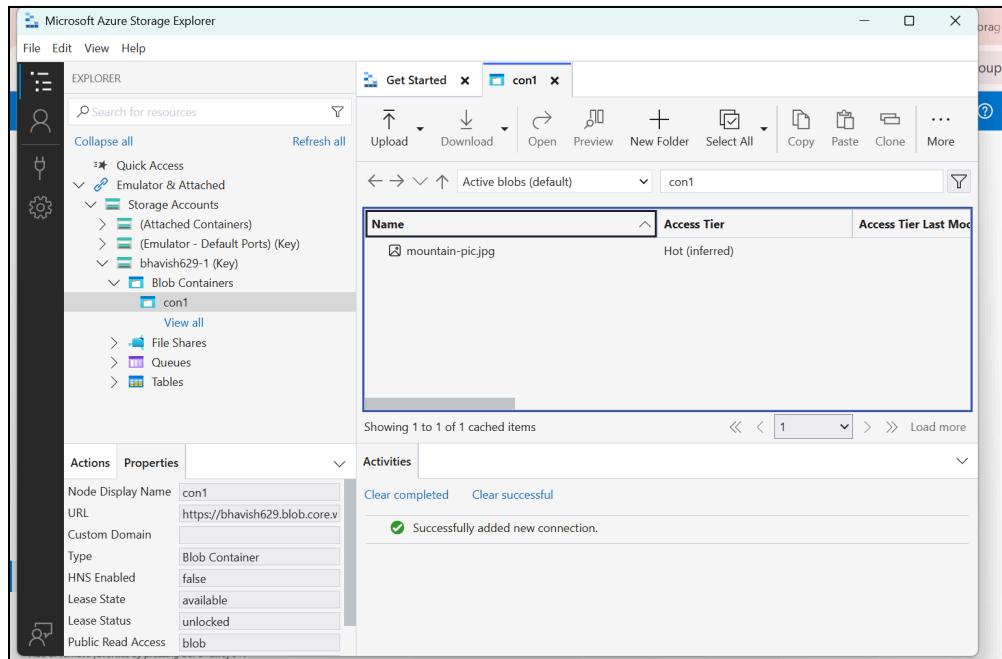


Fig (9) connected to the storage account using access keys.

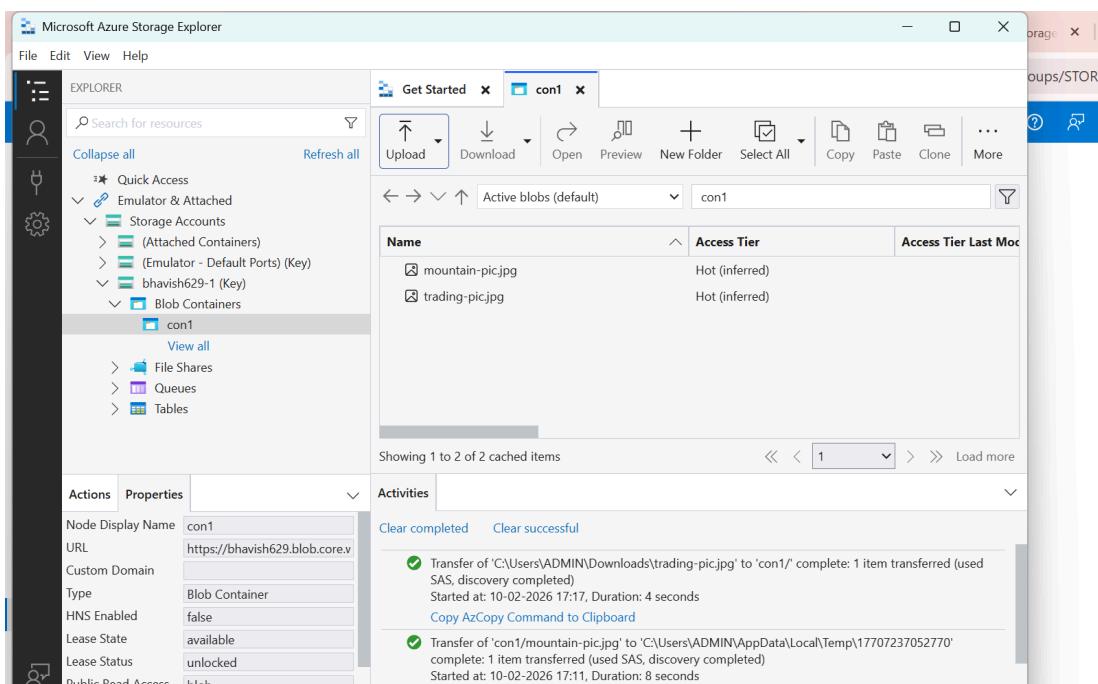


Fig (10) uploaded an image in the storage account using storage explorer.

→ now let's access with sas keys

A shared access signature (SAS) is a URI that grants restricted access rights to Azure Storage resources. You can provide a shared access signature to clients who should not be trusted with your storage account key but whom you wish to delegate access to certain storage account resources. By distributing a shared access signature URI to these clients, you grant them access to a resource for a specified period of time.

An account-level SAS can delegate access to multiple storage services (i.e. blob, file, queue, table). Note that stored access policies are currently not supported for an account-level SAS.

Learn more about creating an account SAS

Allowed services: Blob (checked), File, Queue, Table

Allowed resource types: Service (checked), Container (checked), Object (checked)

Allowed permissions: Read (checked), Write (checked), Delete (checked), List (checked), Add (checked), Create (checked), Update (unchecked), Process (checked), Immutable storage (checked), Permanent delete (checked)

Blob versioning permissions: Enables deletion of versions (checked)

Allowed blob index permissions: Read/Write (checked), Filter (checked)

fig(11) permissions in shared access signature.

Preferred routing tier: Basic (default) (radio button selected), Microsoft network routing, Internet routing

Some routing options are disabled because the endpoints are not published.

Signing key: key1

Generate SAS and connection string

Connection string: BlobEndpoint=https://bhavish629.blob.core.windows.net/QueueEndpoint=https://bhavish629.queue.core.windows.net/FileEndpoint=https://bhavish629.file.cor...

SAS token: sv=2024-11-04&ss=b&srt=sco&sp=rw&clciytxf&se=2026-02-10T19:50:56Z&st=2026-02-10T11:35:56Z&spr=https&sig=G3MDoqSapdZXOvPKi7Z8bmCb5pzXH...

Blob service SAS URL: https://bhavish629.blob.core.windows.net/?sv=2024-11-04&ss=b&srt=sco&sp=rw&clciytxf&se=2026-02-10T19:50:56Z&st=2026-02-10T11:35:56Z&spr=https&...

Fig (12) sas keys and connecting string.

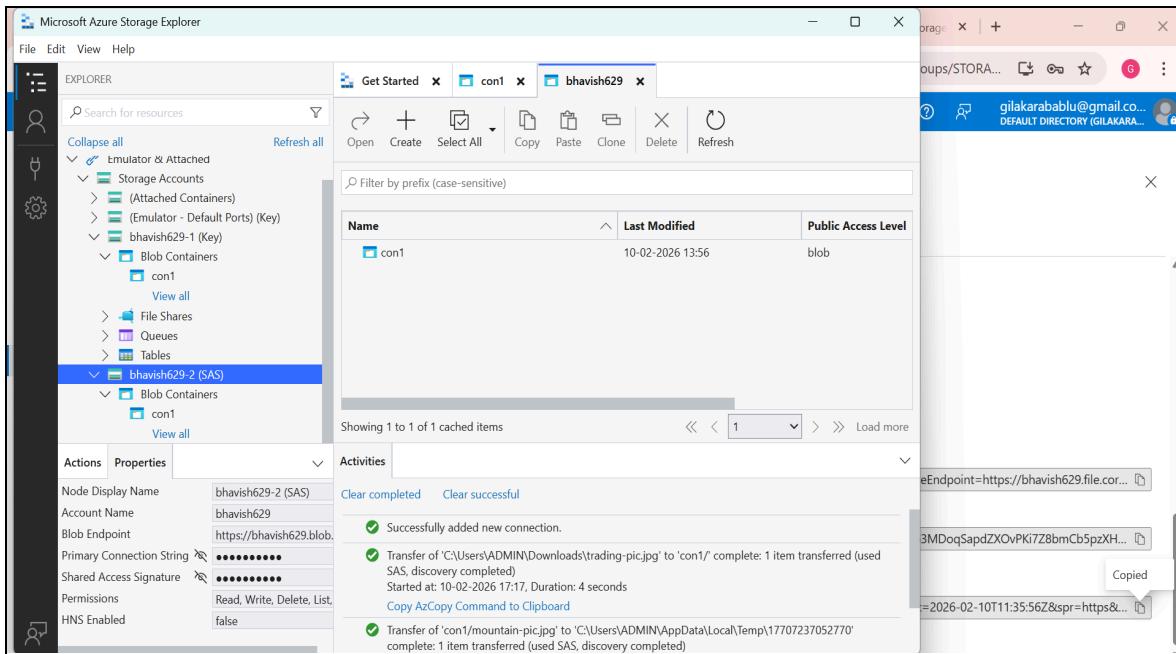


Fig (13) connected to the storage account using sas keys.

→ Roll Based Access Control.

Name	Description	Type	Category	Details
Reader	View all resources, but does not allow you to make any changes.	BuiltInRole	General	View
Advisor Reviews Contributor	View reviews for a workload and triage recommendations linked to them.	BuiltInRole	None	View
API Management Service Contributor	Can manage service and the APIs	BuiltInRole	Integration	View
API Management Service Operator Role	Can manage service but not the APIs	BuiltInRole	Integration	View
API Management Service Reader Role	Read-only access to service and APIs	BuiltInRole	Integration	View
API Management Service Workspace API ...	Has read access to tags and products and write access to allow: assigning APIs to products, assigning tag...	BuiltInRole	None	View

fig(14) adding a role assignment.

→ to add a role goto IAM access control.

The screenshot shows the 'Add role assignment' blade in the Microsoft Azure portal. The 'Conditions' tab is selected. Under 'Selected role', 'Storage Blob Data Reader' is listed. Below it, there's a 'Condition' section with a '+ Add condition' link and the word 'None'.

fig(15) assigning a storage blob data reader.

→ create a file share.

The screenshot shows the 'File shares' blade for a storage account named 'devops'. It displays the following details:

Setting	Value
Storage account	krisstore298734
Resource group (..)	Storage-RG
Location	East US
Subscription (move)	KrishnaReddy-DEV-ENV
Subscription ID	adfed678-4682-4bb0-a62f-2ebd77f373fd

Properties:

Category	Value
Size	Maximum storage (GiB): 102400 Used storage capacity (GiB): 0 Access tier: Transaction optimized
Feature status	Soft delete: Disabled Large file shares: Enabled
Identity-based access	Directory service: Not configured

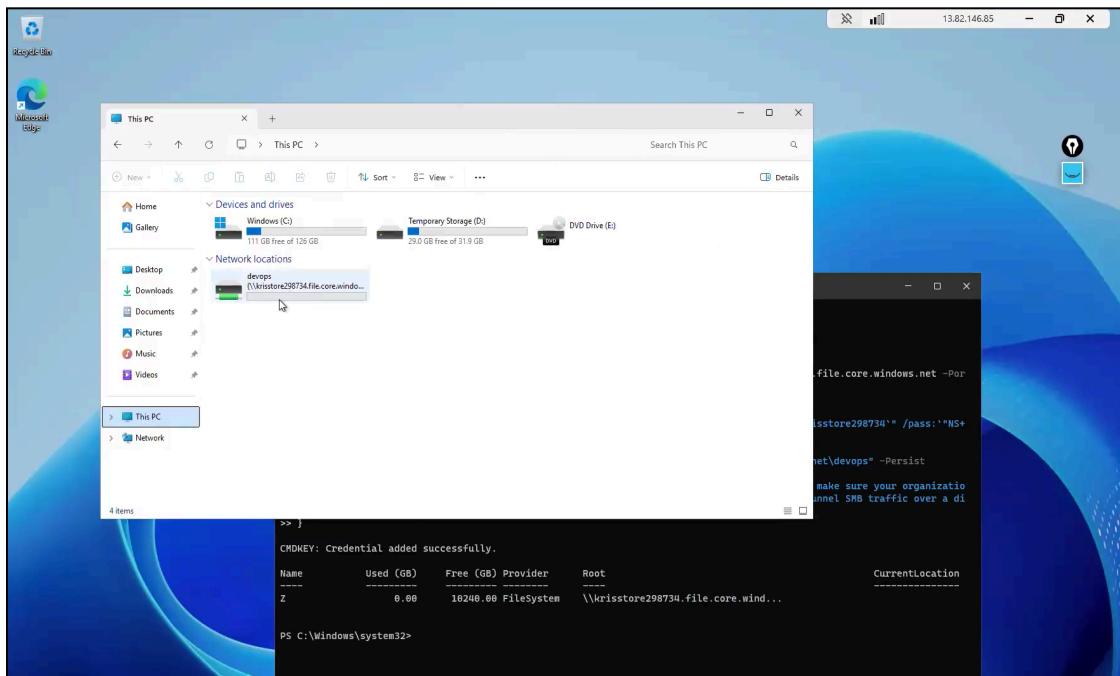
fig(16) file share is created.

The screenshot shows the Microsoft Azure Storage center interface. A new file share named 'devops' has been created under the 'krisstore298734' storage account. The 'Properties' tab is selected, displaying details like Maximum storage (GiB) at 102400 and Access tier set to 'Transaction optimized'. A tooltip provides a PowerShell script for connecting to the share via SMB.

```
$connectTestResult = Test-NetConnection -ComputerName krisstore298734.file.core.windows.net -Port 445
if ($connectTestResult.TcpTestSucceeded) {
    # Save the password so the drive will persist on reboot
    cmd.exe /C "cmdkey /add:"krisstore298734.file.core.windows.net" /user:"localhost\krisstore298734" /pass:"NS+Q6cD1R9t8k5dnqM7n2di88aY/3LweACc8aN5ImQ43dMGd2rX05jgO5Jbr/ERbGQkTXYK+AStAu3pQ=="
    # Mount the drive
    New-PSDrive -Name Z -PSProvider FileSystem -Root "\\\krisstore298734.file.core.windows.net\devops" -Persist
} else {
    Write-Error -Message "Unable to reach the Azure storage account via port 445. Check to make sure your organization or ISP is not blocking port 445, or use Azure P2S VPN, Azure S2S VPN, or Express Route to tunnel SMB traffic over a different port."
}
```

Fig (17) creating a z drive in the windows machine.

→ copy & paste the above script [fig(17)] in the windows machine powershell to create a Zdrive.



fig(18) created a z drive in the windows machine.

→ QUEUES :-

- When we create a queue we get an endpoint, this endpoint is given to the sql developer, he will add in the database, so that all the data will be stored in the Queues.

The screenshot shows the Microsoft Azure portal interface for a storage account named 'krisstore298734'. The left sidebar navigation menu is visible, with 'Queues' selected under the 'Data storage' category. The main content area displays a table titled 'Search queues by prefix' with one entry: 'Queue' (asdfs) and 'Url' (https://krisstore298734.queue.core.windows.net/asdfs). The URL is highlighted with a blue selection bar.

fig(19) created a queue.

→ TABLES :-

- When we create a Table we get an endpoint, this endpoint is given to the sql developer, he will add in the database, so that all the data will be stored in the Tables.

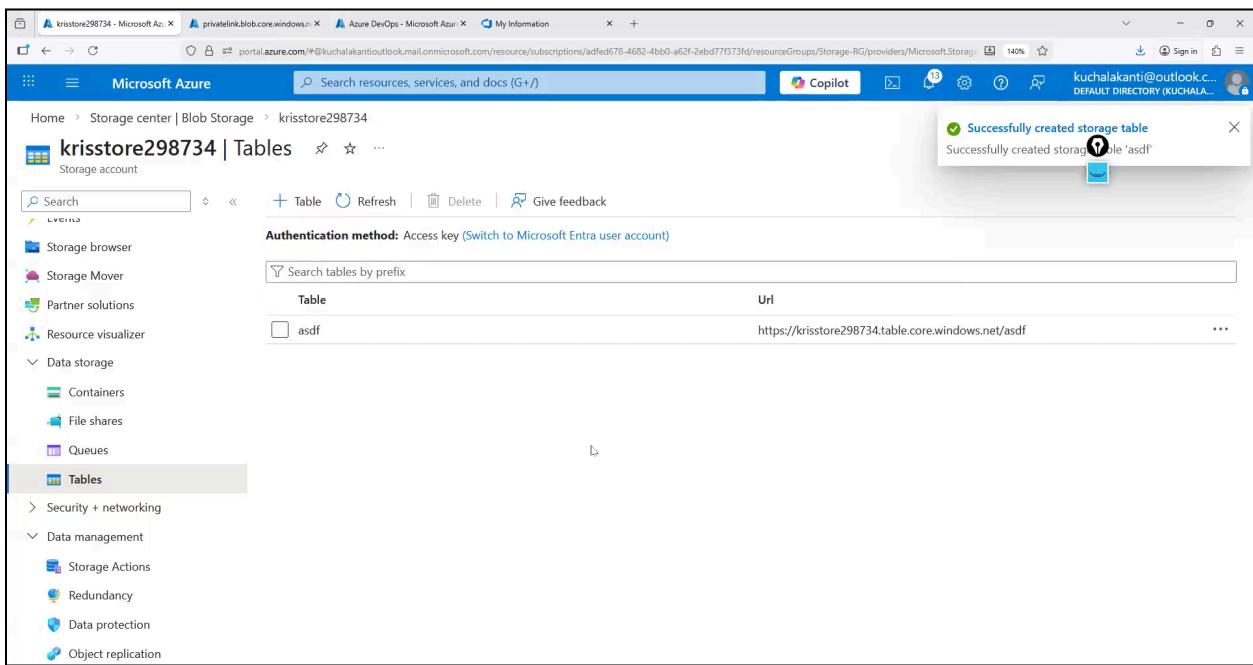


Fig (20) a table is created.