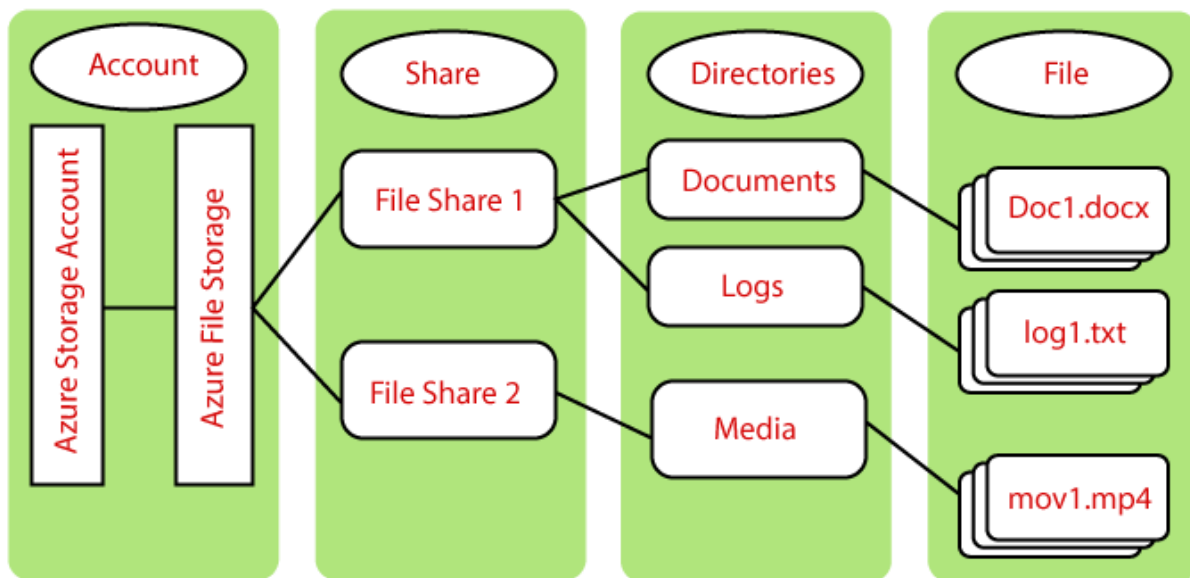


Azure File Storage Service

Azure file storage mainly can be used if we want to have a shared drive between two servers or across users. In that case, we will go for Azure file storage. In the Azure file storage structure, the first thing we need to have is an Azure storage account. Azure file storage is offered under the umbrella of the Azure storage account. And once we have created an Azure storage account, we'll create a file share.

We can create an unlimited number of file shares within a storage account. Once we create a file share, then we can create directories, just like folders, and then we can upload files into it. Once we create a file share, we can mount that on any virtual machine, whether it is in Azure or outside.

Azure file storage



Some of the concepts related to Azure file storage:

- **Storage Account:** All access to Azure Storage service is done through a storage account. We need to keep in mind scalability and performance targets when we might have Azure file storage, blob storage, tables, and queues. All of them might be sharing the same performance targets under the storage account, so the same limitations of storage account will be shared-by across all services.

- **Share:** A file storage Share is an SMB file share in Azure. The directories and files must be created in a parent's Share, so we can't create a directory directly in a storage account. We need file storage share, first created, and then we can create directories to upload. An account may contain an unlimited number of shares, and a Share can store an unlimited number of files, up to 5 TB total capacity of the file share. But, in case we need more than the full capacity, then we can create another file share.
- **Directory:** It is an optional hierarchy of directories.
- **File:** A file in the share. A file may be up to 1 TB in size.
- **URL format:** For a request to an Azure file share made with the file REST protocol

File Storage Data Access methods

Azure files offer two, built-in, convenient data access methods that you can use separately, or in combination with each other, to access your data:

- **Direct Cloud Access:** Windows, MacOS, or Linux can mount any Azure file share with the industry-standard Server Message Block (SMB) protocol or via the File REST API. But if we use the SMB protocol, then we need to take care of two things. First, if we are mounting the file share on a VM in Azure, then the SMB client in the OS must support at least SMB 2.1. Secondly, if we want to mount file share on an on-premises system such as a user's workstation, then the SMB client supported by workstation must be at least SMB 3.0 with encryption enabled. And if we are mounting Azure file share within our on-premises data center, then it should be 3.0 minimum version of SMB client, but if we are installing VM in Azure, it can be 2.1.
- **Azure File Sync:** With Azure File Sync, *Shares* can be replicated to on-premises or Azure Windows Servers. Our users can access the file share through the Windows Server, such as SMB or NFS share. Synchronization of any frequently accessed files will be kept in the server endpoint, and any infrequently accessed data will be moved to Azure file share. In that way, we can get the speed of the delivery of the data to your users and, at the same time, save the storage.

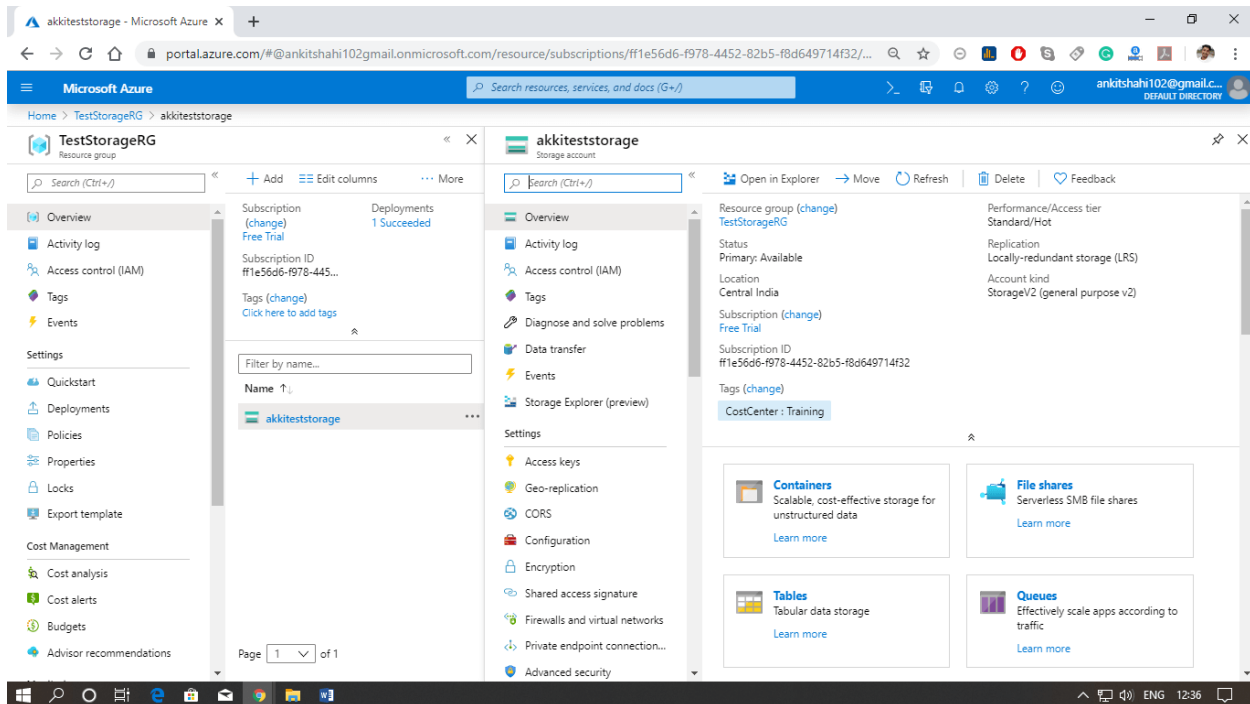
Data transfer method

When we create an Azure file share, and we have a large file share already inside our on-premises data center. Then with the help of the option below, we can transfer those files.

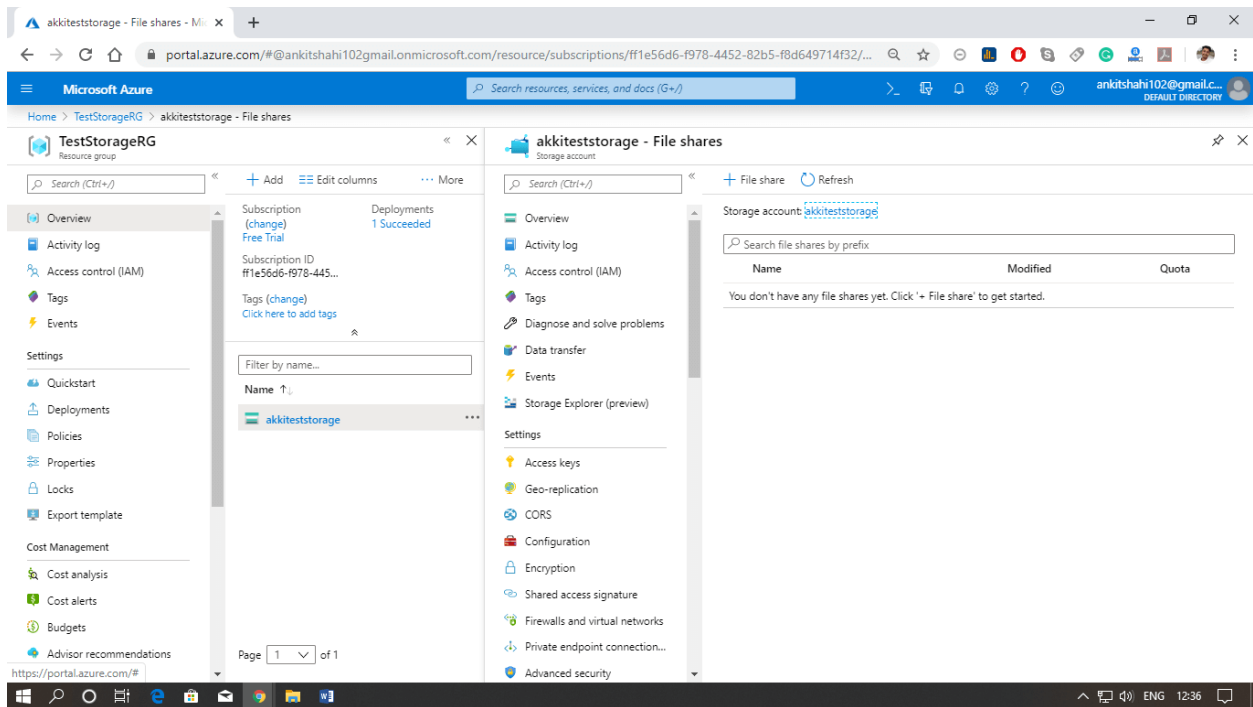
- **Azure file sync:** As a part of the first sync between an Azure file share (a "cloud Endpoint") and a windows directory namespace (a "Server Endpoint"), Azure File Sync will replicate all data from the existing file share to Azure Files.
- **Azure Import/export:** If we have terabytes of data, which we need to transfer into Azure files. And if we start moving data using Azure file sync, it might consume all the bandwidth, or it might be a slow process that may take months. In that case, you can use Azure import/export. Microsoft will provide a hard disk to you to move all the data in the hard drive and ship that hard drive back to Microsoft, and Microsoft will load the data from the hard-disk and into an Azure data center.
- **Robocopy:** Robocopy is a well-known copy tool that ships with windows and windows Server. We can use it to transfer the data into Azure files by mounting the locally shared files and then using the mounting location as a destination in the robocopy command.
- **AzCopy:** It is a command-line utility tool, which we can use for copying the data to and from Azure files. It can be used for blob storage also, and you can use Azcopy with simple commands. It provides excellent performance and is available for Windows and Linux.

Creating a File Storage in Azure

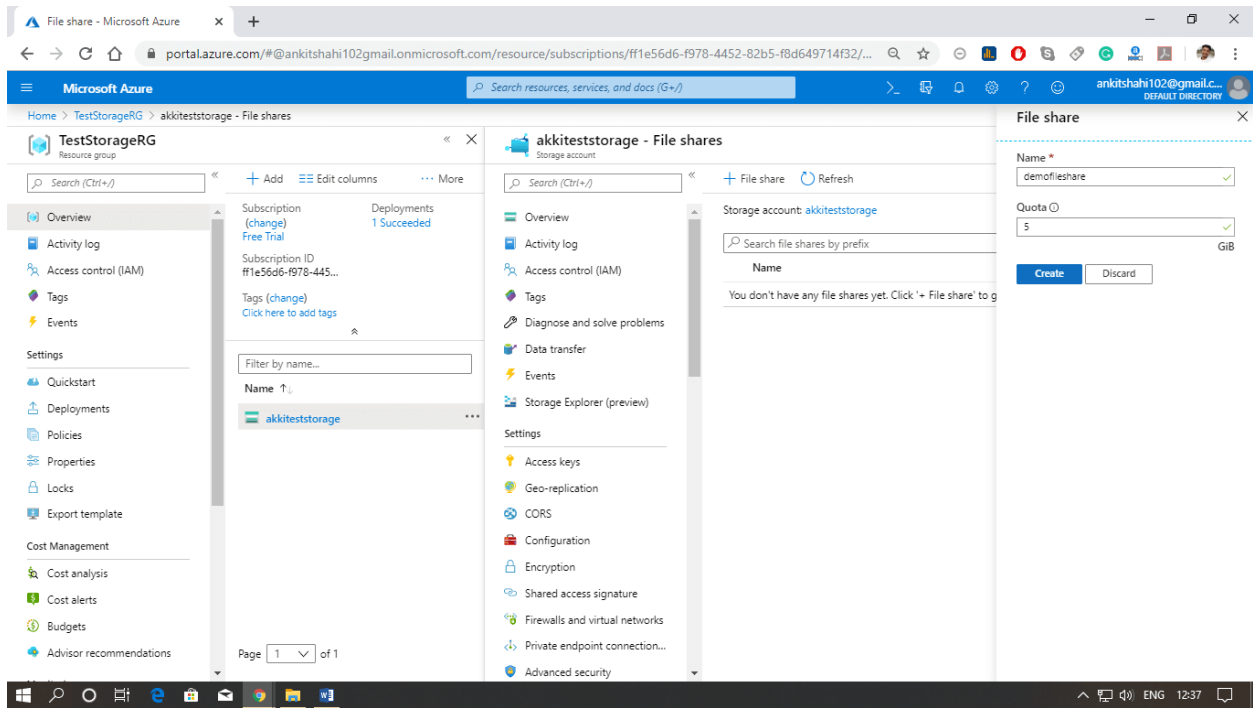
Step 1: Let's go into Azure storage Account, then scroll down and click on Files.



Step 2: Click on +File Share.



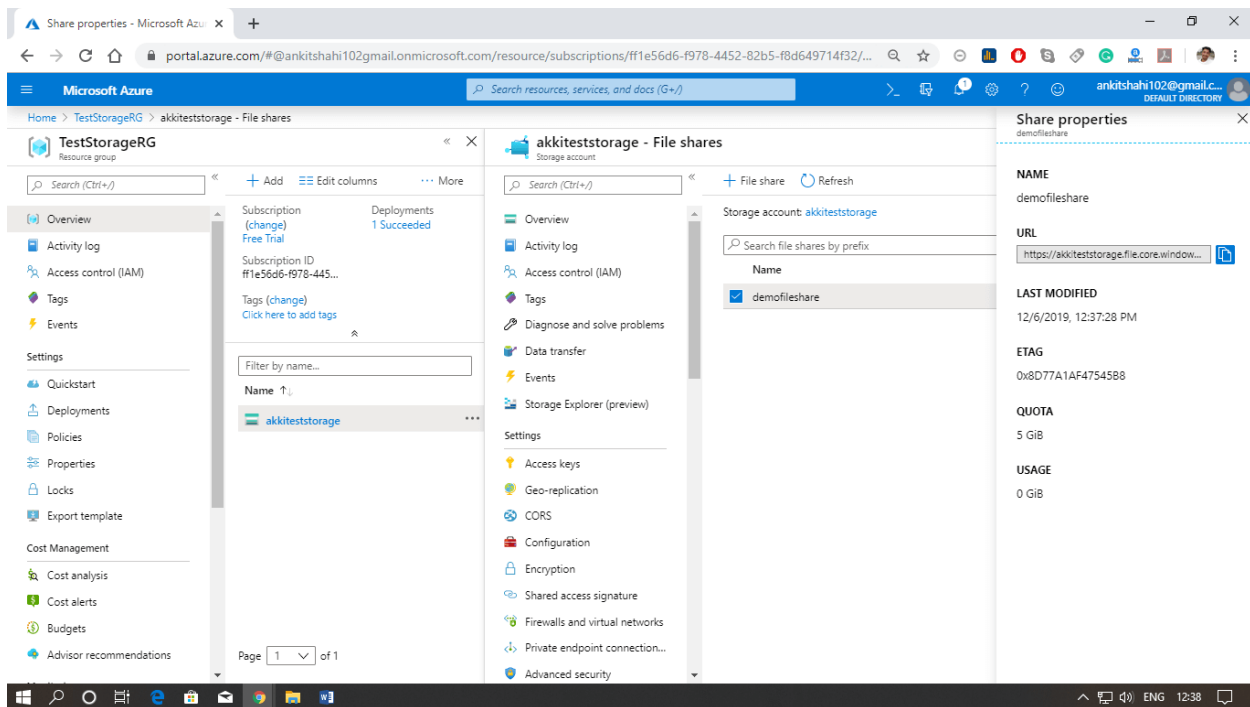
Step 3: Provide the name and Quota of the file share, then you will get a notification of the successful creation of File share.



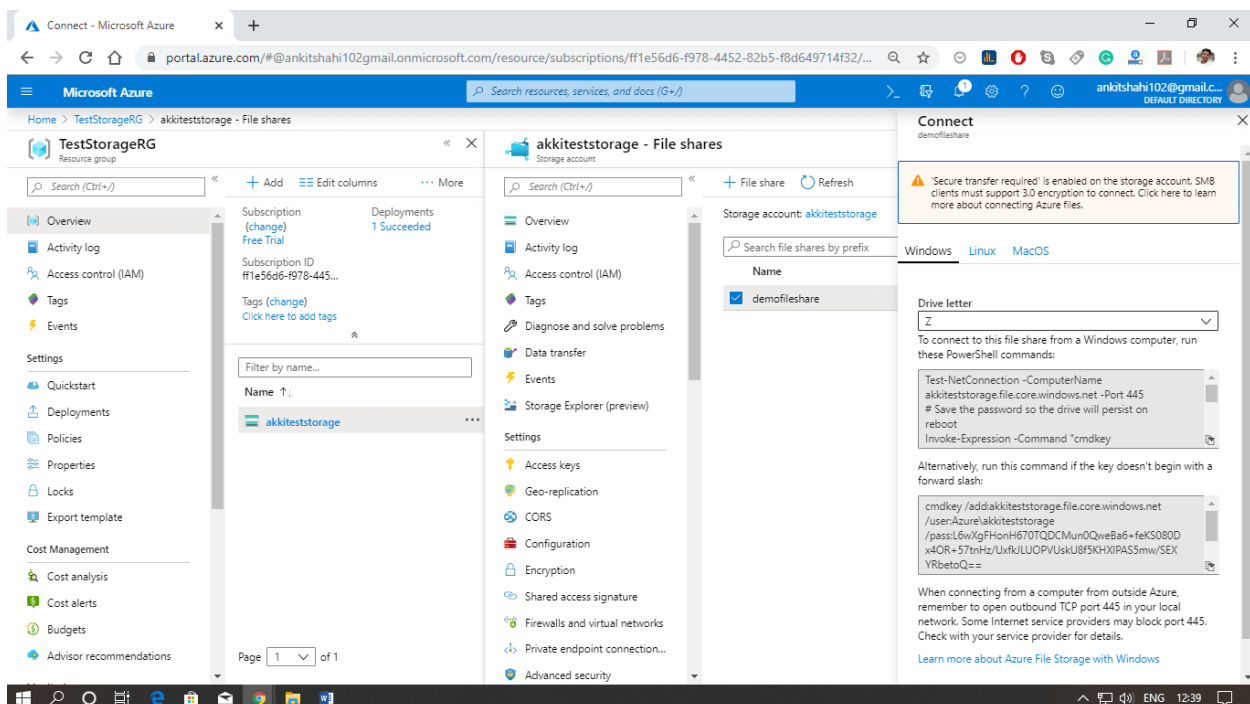
The screenshot shows the Microsoft Azure portal interface. The left sidebar contains navigation options like Overview, Activity log, Access control (IAM), Tags, Events, Settings, Quickstart, Deployments, Policies, Properties, Locks, Export template, Cost Management, Cost analysis, Cost alerts, Budgets, and Advisor recommendations. The main content area is titled 'akkiteststorage - File shares' and shows a table with one file share named 'demofiles' with a quota of 5 GiB. A notification banner at the top right states 'Successfully created storage file share'.

Step 4: Click on the File share property where you can see the URL and also the quota you assigned and how much it is used.

The screenshot shows the Microsoft Azure portal interface. The left sidebar contains navigation options like Overview, Activity log, Access control (IAM), Tags, Events, Settings, Quickstart, Deployments, Policies, Properties, Locks, Export template, Cost Management, Cost analysis, Cost alerts, Budgets, and Advisor recommendations. The main content area is titled 'akkiteststorage - File shares' and shows a table with one file share named 'demofiles' with a quota of 5 GiB. A context menu is open over the file share, showing options like Properties, Connect, Quota, Edit metadata, Access policy, View snapshots, and Delete share.



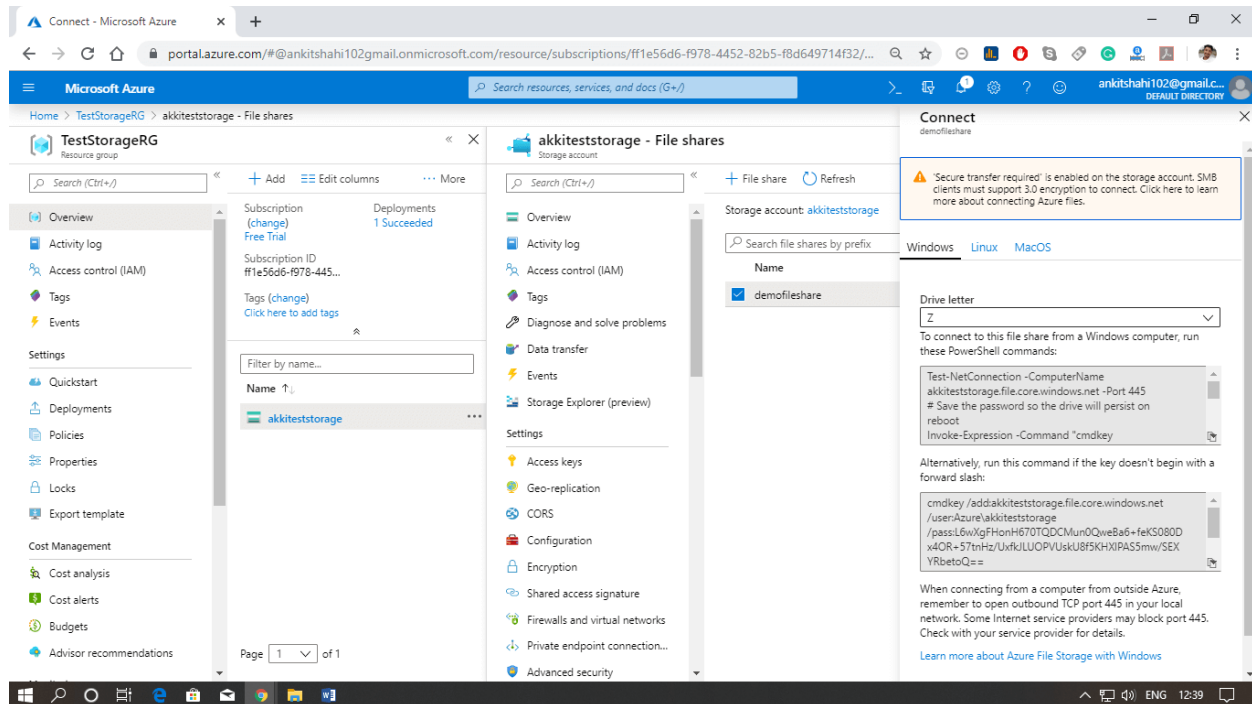
Step 5: And the second option is to connect, here you can see the PowerShell command and normal command to connect this File Share on a Windows computer.



here are other options also given in the following figures.

Step 6: Click on File share and open it, where you can see the Access Control tab. You can use the Active Directory to control access to Azure file share. It is currently in preview.

Step 7: To mount this file share with windows virtual machine Click on Connect.



Step 8: Open the command line and copy the command given in the Connect window. If the command is executed successfully, your file share would be mounted with a virtual machine.

Connect - Microsoft Azure

portal.azure.com/#blade/Microsoft_Azure_FileStorage/FileShareMenuBlade/overview/storageAccountId/%2Fsubscriptions%...

Microsoft Azure

Search resources, services, and docs (G+)

Home > TestStorageRG > akkiteststorage - File shares > demofilesare

demofilesare

File share

Search (Ctrl+F)

Overview

Access Control (IAM)

Settings

Access policy

Properties

Administrator: Command Prompt

```
Microsoft Windows [Version 10.0.18362.476]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Windows\system32>cmdkey /add:akkiteststorage.file.core.windows.net /user:Azure\akkiteststorage /pass:L6wXgFHonH670TQD
CMun0QweBa6+feKS080Dx40R+57tnHz/Uxfk3LU0PVUskU8f5KHx1PAS5mw/SEXYRbetoQ==

CMDKEY: Credential added successfully.

C:\Windows\system32>net use Z: \\akkiteststorage.file.core.windows.net\demofilesare /persistent:Yes
The command completed successfully.

C:\Windows\system32>
```

Connect

demofilesare

Learn more about Azure File Storage with Windows

12:46