😊 Azure File Share

Azure File Share is a cloud-based file storage service provided by Microsoft Azure. It allows users to create and manage file shares in the cloud, which can be accessed and managed like traditional file shares in an on-premises network. Here are some key features and benefits of Azure File Share:

Key Features

1. **SMB Protocol Support**:
    - Supports the Server Message Block (SMB) protocol, making it compatible with Windows, macOS, and Linux operating systems.
2. **Azure Storage Integration**:
    - Integrated with Azure Storage, providing scalable and secure file storage options.
3. **Access Control**:
    - Uses Azure Active Directory (AAD) for identity-based authentication and provides role-based access control (RBAC).
4. **File Sync**:
    - Azure File Sync can be used to synchronize files between on-premises Windows Servers and Azure File Share, enabling hybrid cloud scenarios.
5. **Backup and Restore**:
    - Supports Azure Backup to create backup copies of your file shares and restore them when needed.
6. **High Availability and Durability**:
    - Provides redundancy and high availability options, ensuring that your data is safe and accessible.

Benefits

1. **Scalability**:
    - Easily scales up or down based on your storage needs, without the need to manage physical hardware.
2. **Cost-Effectiveness**:
    - Pay only for what you use, with options to optimize costs through tiered storage.
3. **Accessibility**:
    - Access your file shares from anywhere with an internet connection, enabling remote work and collaboration.

4. **Security**:
    - Offers advanced security features, including encryption at rest and in transit, and integration with Azure security services.
5. **Simplified Management**:
    - Centralized management through the Azure portal, PowerShell, or Azure CLI.

Common Use Cases

1. **File Storage**:
    - Store and share files, including documents, media files, and application data.
2. **Lift and Shift**:
    - Migrate existing applications that use file shares to Azure without changing the application code.
3. **Dev/Test Environments**:
    - Quickly set up and tear down environments with the necessary file shares.
4. **Backup and Disaster Recovery**:
    - Use as part of a backup and disaster recovery strategy to ensure data availability.

Getting Started

1. **Create a Storage Account**:
    - First, create an Azure Storage account in the Azure portal.
2. **Create a File Share**:
    - Within the storage account, create a new file share and specify the quota and other settings.
3. **Connect to the File Share**:
    - Connect to the file share from your local machine or server using the SMB protocol.
4. **Manage and Monitor**:
    - Use the Azure portal, PowerShell, or CLI to manage and monitor your file shares.

Azure File Share is a versatile and powerful solution for cloud-based file storage, providing the benefits of scalability, security, and ease of use.

🧐 What are we doing in this lab?

In this process, you are setting up an Azure File Share, creating directories, and uploading files to it. The end goal is to connect this file share to a local machine or a virtual machine (VM) to access and manage the files stored in the Azure cloud.

Summary of Steps:

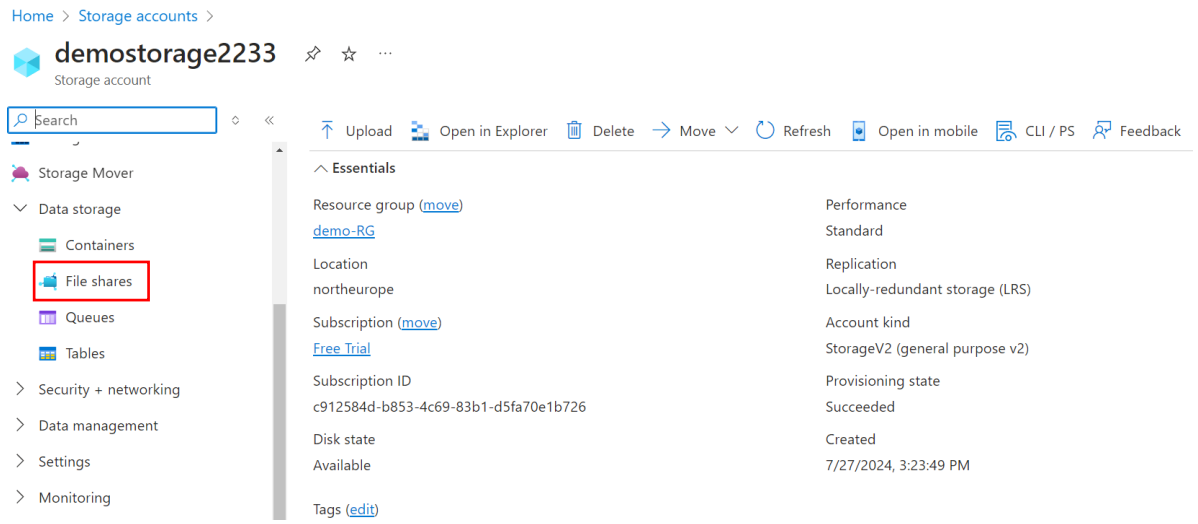
1. **Set Up Azure File Share**:
    - Log in to Azure Portal, navigate to Storage Account, and create a new file share.
    - Name the file share and configure its settings.
2. **Create Directory and Upload Files**:
    - Create a directory within the file share and upload some files to it.
3. **Connect to File Share**:
    - Generate a connection script in the Azure Portal.
    - Use the script in PowerShell on your local machine to map the file share as a network drive (Z drive).
4. **Map Network Drive in Windows Explorer**:
    - Map the file share as a network drive in Windows Explorer using the connection details.

End Goal:

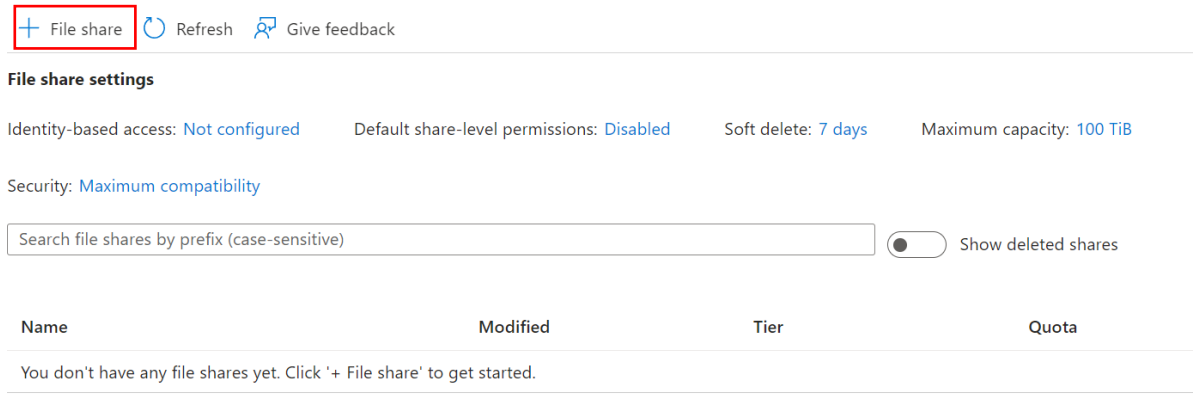
The end goal is to have a network drive on your local machine that connects to the Azure File Share, allowing you to access and manage the files stored in Azure as if they were on a local drive. This enables seamless integration and access to cloud storage from your local environment.

😄 To begin with the Lab

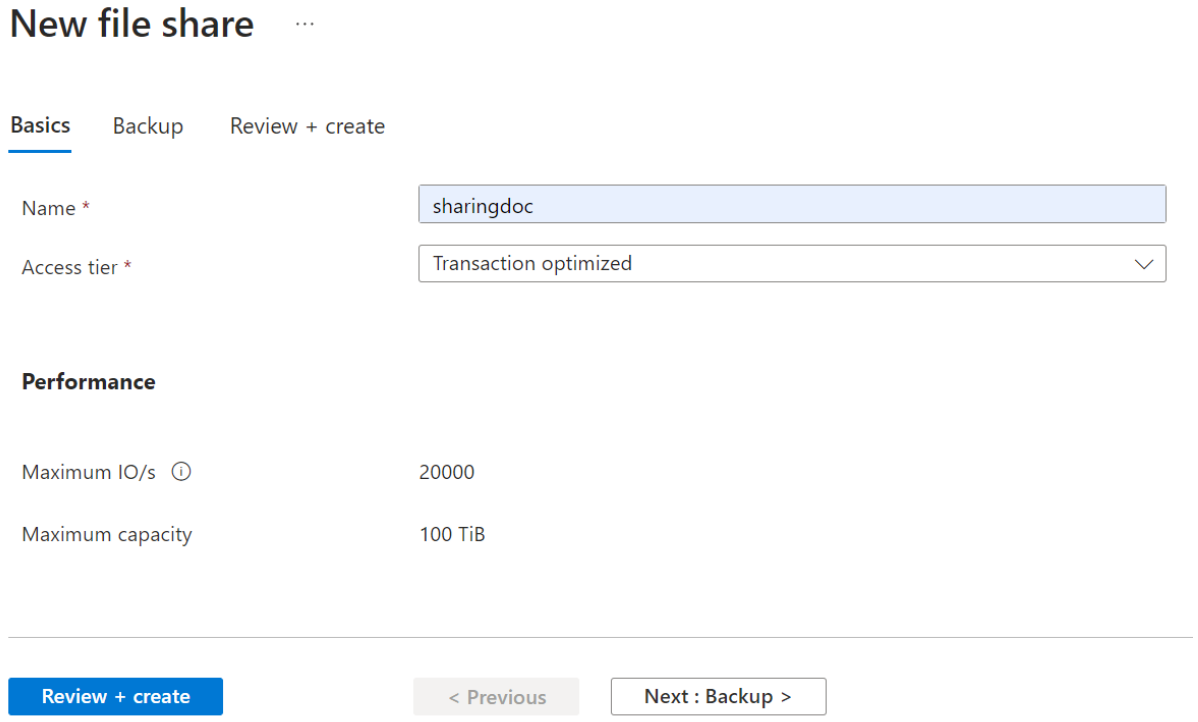
1. Log in to Azure Portal navigate to Storage Account then open your storage account.
2. Then from the left pane you will option for File Share under Data storage section. Click on file share as highlighted below.



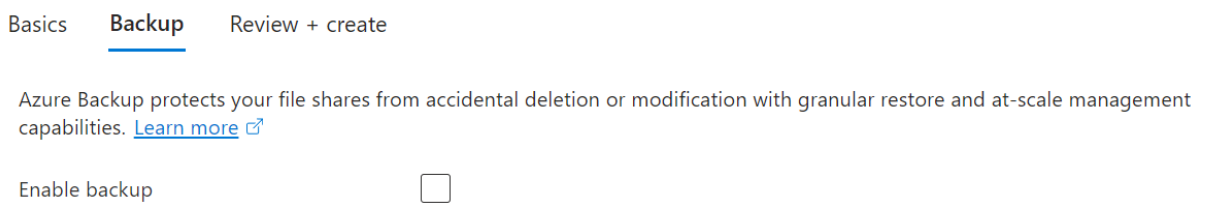
3. Now to create your file share you need to click on add file share.



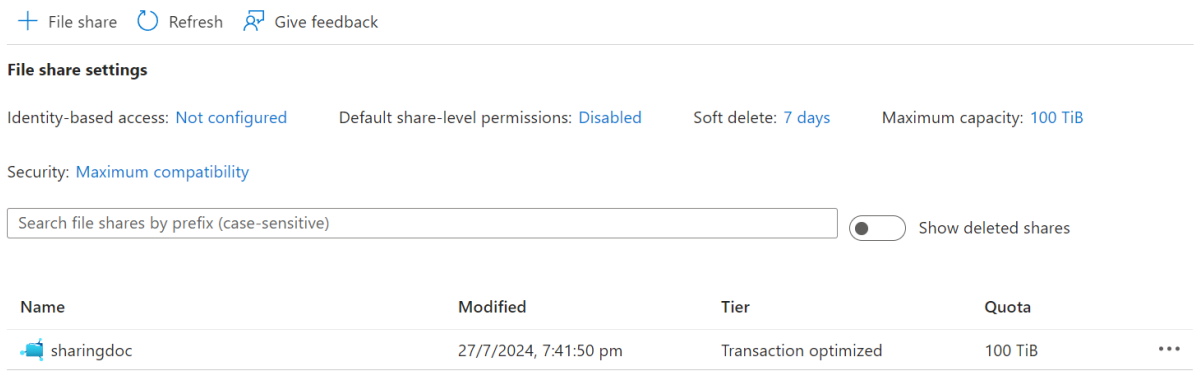
4. Now you need to give it a name to your file share and then keep the access tier to default then go to backup.



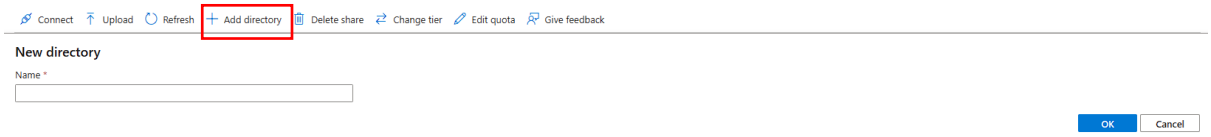
5. In the backup you need to turn off it. After that move to the review page and create your file share.



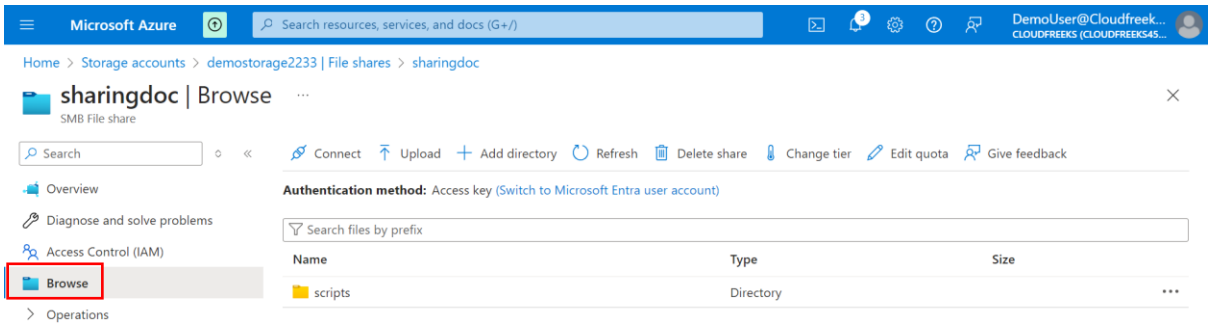
6. Below you can see your file share. Now you need to go inside of it.



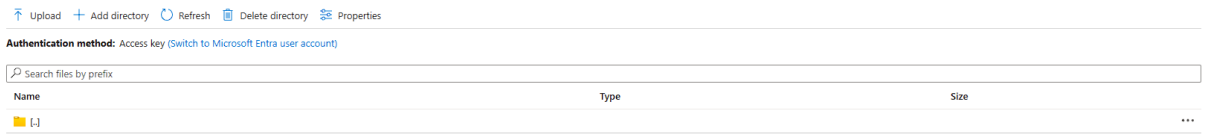
7. Now you will see the option to add directory click on it.

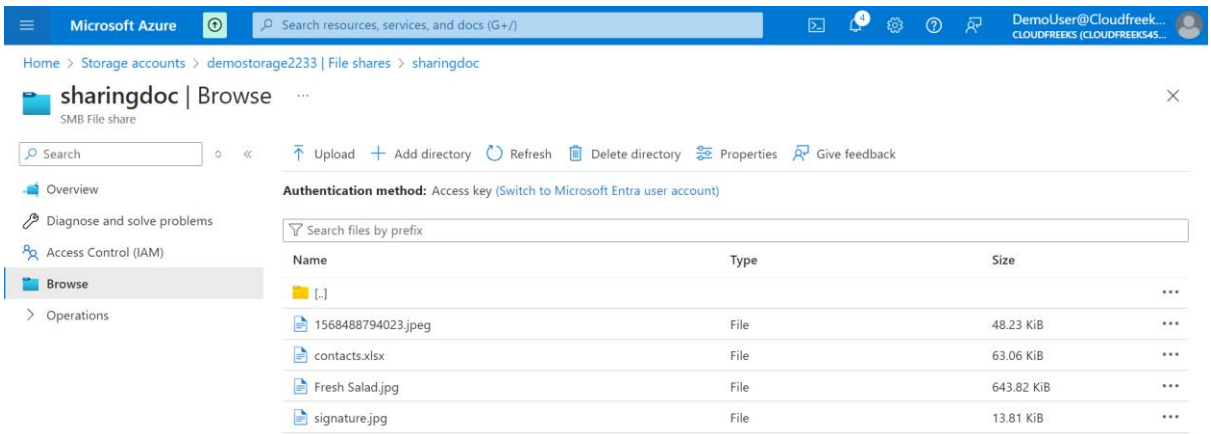


8. After creating the directory, navigate to browse. You will find your new directory there.
9. And if you click on it a single time you will move inside it.



10. In here you need to upload some random files.
11. There you can see all your random files.

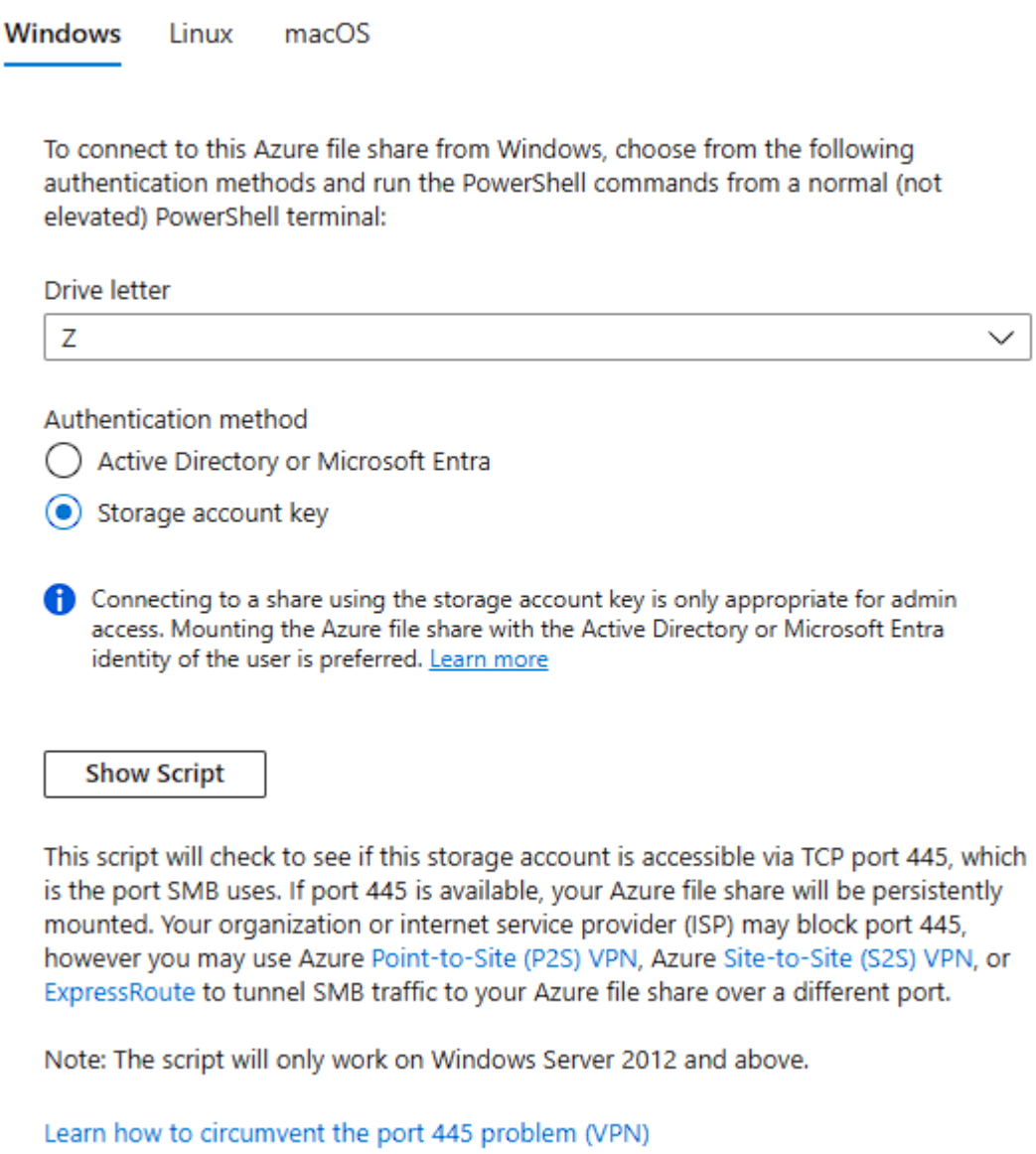




12. Now if you click on any of the files you will see its URL.
13. But if you take the URL and go on to a new tab, it gives you some errors. And that's because this service is different from the blob service. With the blob service, every file is uploaded as an object. And each object gets its unique URL.
14. But this is a file share. You have to connect to the file share.
15. So, a user can now connect to the file share from their local machine, or even a VM can connect to the file share.

16. So, now if you go to the overview of the file share.
17. You will see this connect option, click on it.

18. So, you can connect it with either Windows, Linux, or even macOS.
19. Now if you will just click on the Show script option.



20. You can see your script here in place.

Hide Script

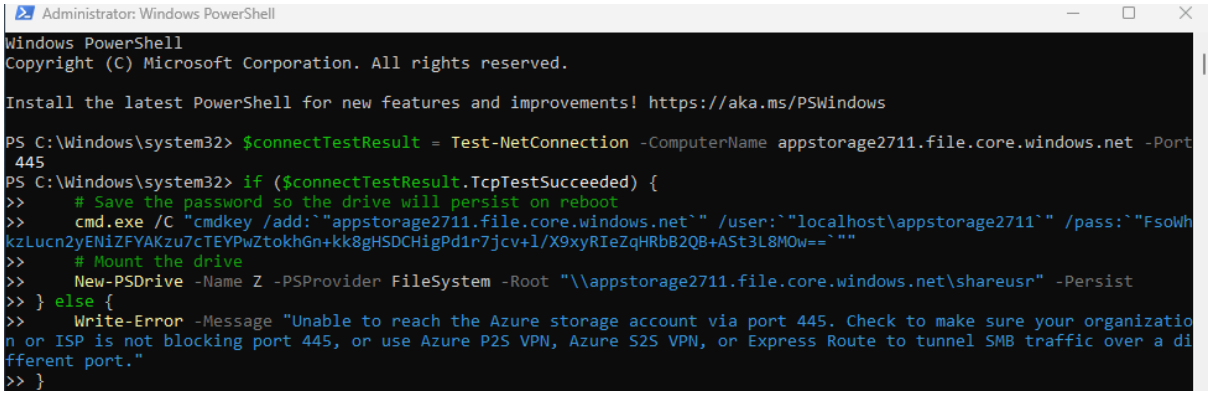
```
$connectTestResult = Test-NetConnection -ComputerName
appstorage2711.file.core.windows.net -Port 445
if ($connectTestResult.TcpTestSucceeded) {
    # Save the password so the drive will persist on reboot
    cmd.exe /C "cmdkey /add:`"appstorage2711.file.core.windows.net`"
/user:`"localhost\appstorage2711`"
/pass:`"FsoWhkzLucn2yENiZFYAKzu7cTEYPwZtokhGn+kk8gHSDCHigPd1r7jcv
+l/X9xyRIeZqHRbB2QB+ASt3L8MOw==`""
    # Mount the drive
    New-PSDrive -Name Z -PSProvider FileSystem -Root
"\\appstorage2711.file.core.windows.net\shareusr" -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."
}
```

This script will check to see if this storage account is accessible via TCP port 445, which is the port SMB uses. If port 445 is available, your Azure file share will be persistently mounted. Your organization or internet service provider (ISP) may block port 445, however you may use Azure Point-to-Site (P2S) VPN, Azure Site-to-Site (S2S) VPN, or ExpressRoute to tunnel SMB traffic to your Azure file share over a different port.

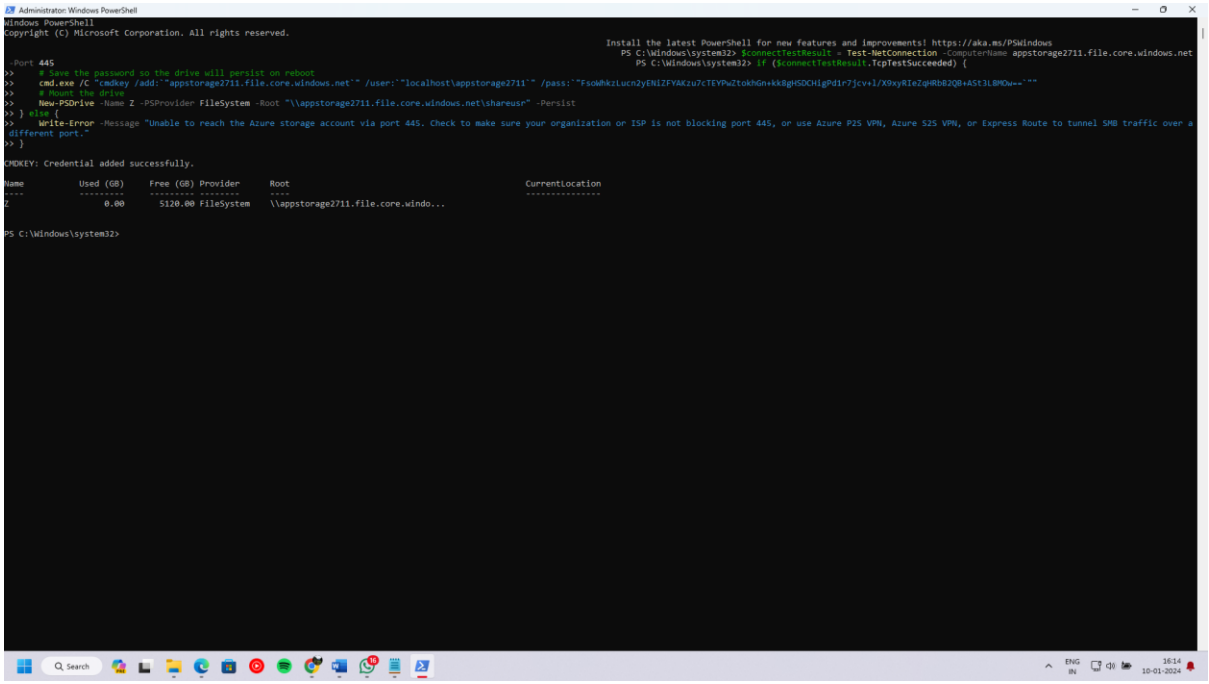
Note: The script will only work on Windows Server 2012 and above.

Learn how to circumvent the port 445 problem (VPN)

21. Now if you copy this script and **open PowerShell in your local machine as an administrator paste this script onto the PowerShell.**
22. The first thing is going to do is it's going to check whether you can connect. To the storage account.
23. Because you could be having a firewall, especially if you're connecting from a company laptop that might not allow connections on Port 445 onto the Azure storage account, onto the file share service.



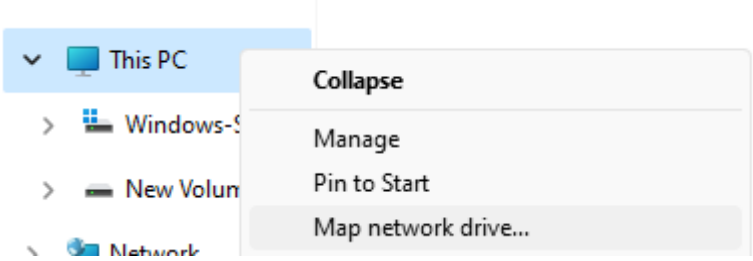
24. You can see that it has been connected.
25. Now just hit enter on your keyboard. Now it is going map the Z drive onto the file share.



26. Now go into the Z drive, then do a directory mapping on it and you can see your scripts folder right here.
27. Now you will go into your scripts folder and do a directory mapping on it.
28. You will see all the files that you uploaded onto the file share.

```
PS C:\Windows\system32> z:
PS Z:\> dir


    Directory: Z:\


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        10-01-2024     15:32                scripts


PS Z:\> cd scripts
PS Z:\scripts> dir


    Directory: Z:\scripts


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----        10-01-2024     15:53            113 Dockerfile
-a----        10-01-2024     15:53            186 IIS.ps1
-a----        10-01-2024     15:53             65 install_web.sh
-a----        10-01-2024     15:53           1650 main.py
-a----        10-01-2024     15:53             58 script.yml


PS Z:\scripts> _
```

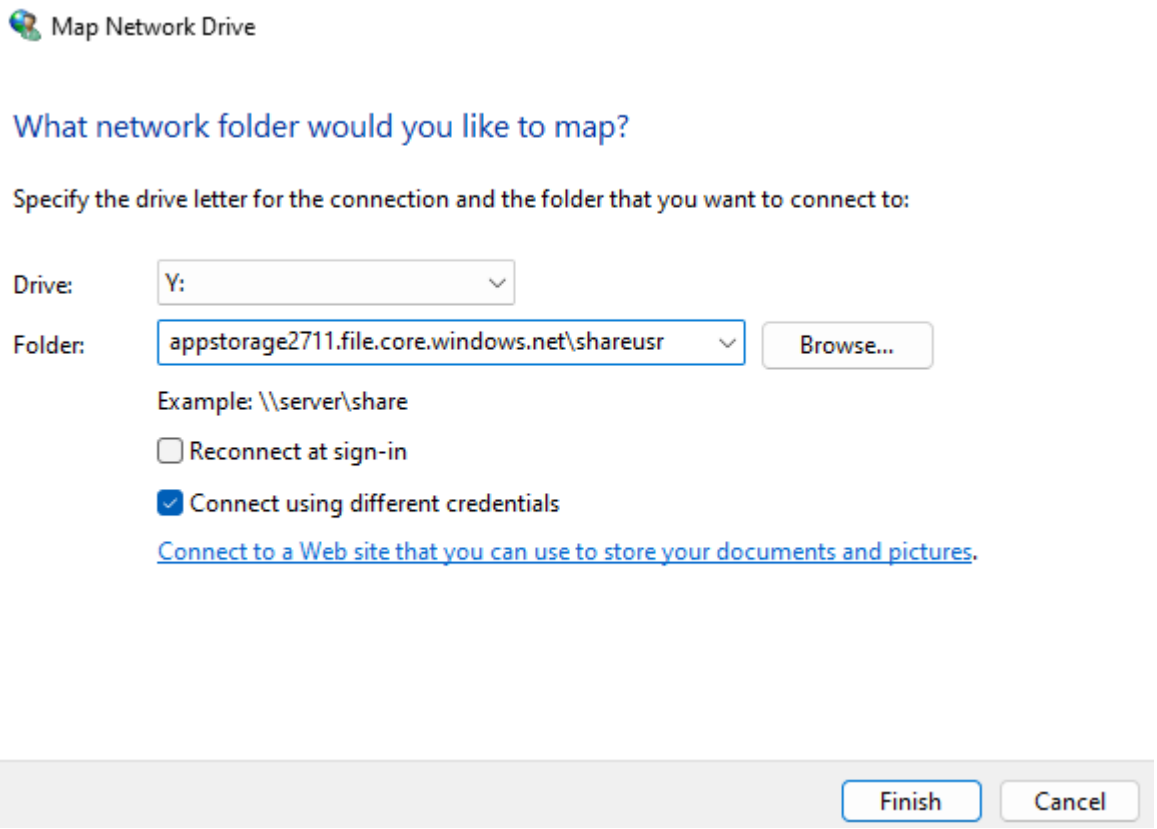
29. Now, even from your Windows Explorer, you should be able to map this network drive or the file share.
30. Go to your Windows Explorer and then right-click on This PC, then select Map network drive.



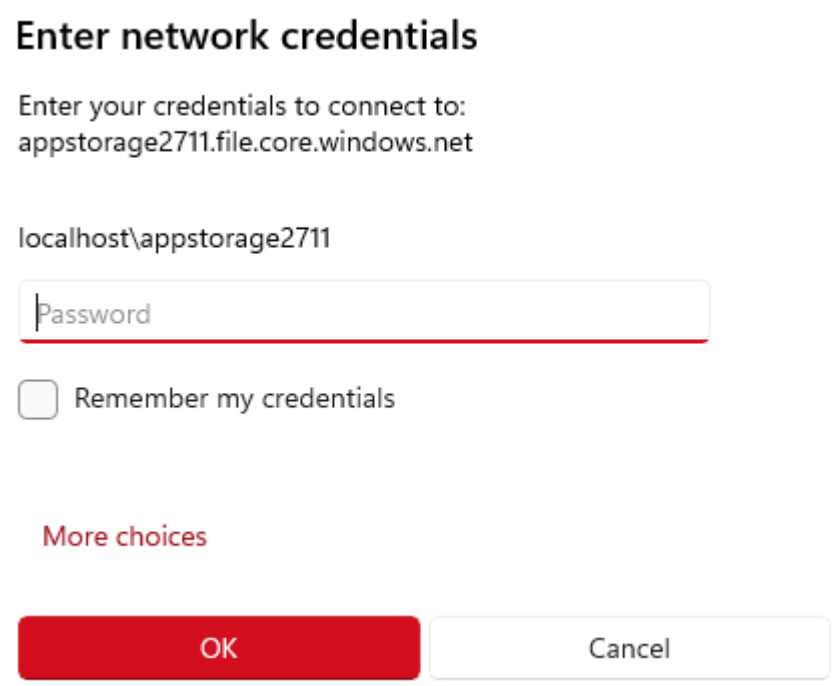
31. Now you have to this link from your script on the Portal.

```
$connectTestResult = Test-NetConnection -ComputerName
appstorage2711.file.core.windows.net -Port 445
if ($connectTestResult.TcpTestSucceeded) {
    # Save the password so the drive will persist on reboot
    cmd.exe /C "cmdkey /add:`"appstorage2711.file.core.windows.net`"
/user:`"localhost\appstorage2711`"
/pass:`"FsoWhkzLucn2yENiZFYAKzu7cTEYPwZtokhGn+kk8gHSDCHigPd1r7jcv
+l/X9xyRIeZqHRbB2QB+ASt3L8MOw==`""
    # Mount the drive
    New-PSDrive -Name Z -PSProvider FileSystem -Root
"\\appstorage2711.file.core.windows.net\shareusr" -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."
}
```

32. Now paste it here in the mapping drive.
33. Select connection using different credentials. Then click on finish.



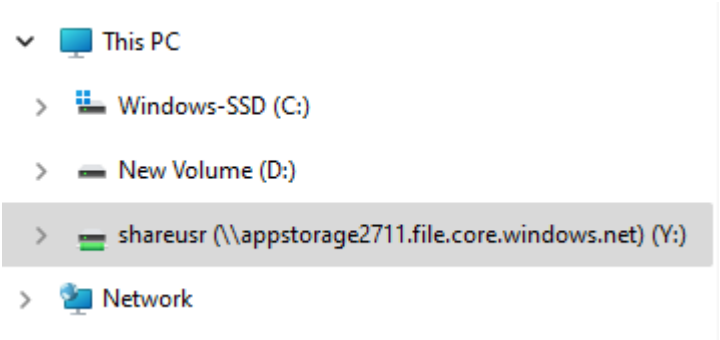
34. You will see that is asking for a password to connect.



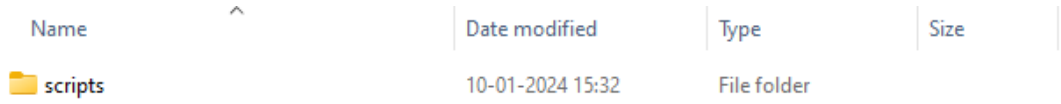
35. Go back to the portal and copy the password from there.
36. Then paste it into the network credentials.

```
$connectTestResult = Test-NetConnection -ComputerName
appstorage2711.file.core.windows.net -Port 445
if ($connectTestResult.TcpTestSucceeded) {
    # Save the password so the drive will persist on reboot
    cmd.exe /C "cmdkey /add:`"appstorage2711.file.core.windows.net`"
/user:`"localhost\appstorage2711`"
/pass:`"FsoWhkzLucn2yENiZFYAKzu7cTEYPwZtokhGn+kk8gHSDCHigPd1r7jcv
+l/X9xyRIeZqHRbB2QB+ASt3L8MOw==`""
```

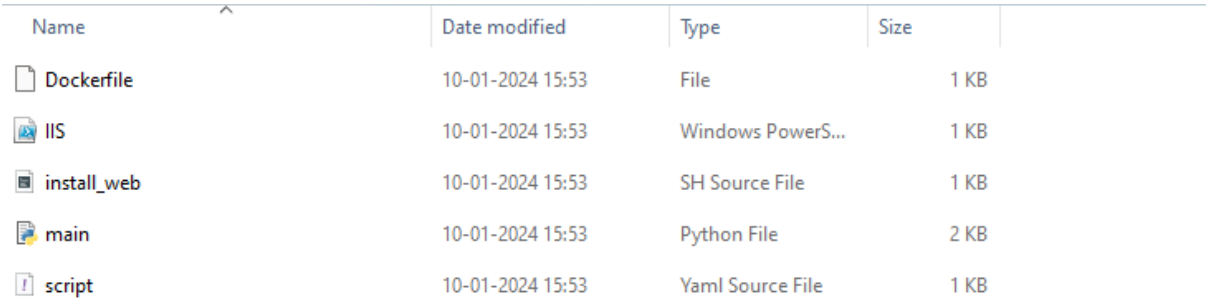
37. Now here you can see your shareusr present in your local machine Windows Explorer.



38. There is the script folder.



39. And here are the files that you have uploaded onto your file share.



Name	Date modified	Type	Size
Dockerfile	10-01-2024 15:53	File	1 KB
IIS	10-01-2024 15:53	Windows PowerS...	1 KB
install_web	10-01-2024 15:53	SH Source File	1 KB
main	10-01-2024 15:53	Python File	2 KB
script	10-01-2024 15:53	Yaml Source File	1 KB

To disconnect it from your local machine, right-click on the shareusr and you will see a disconnect option.