Enable AD Authentication For Azure File Share - Draft Version

Steps to enable AD authentication for AFS

1. Raise a request for **IAM Team** to create **computer object** (similar to **storage account name** which must be **15 char or less**) in ENT domain (ent.bhpbilliton.net). Below is the request content (Technical Work Request for IAM Team):

Here is the catalog link Service Catalogue - Support Central Portal (service-now.com), Select **Team** as **Identity & Access management (IAM) Support** and provide the below content as Description**.**

> Hi IAM Team,
>
> Please create a computer object in AD with name `<STORAGE ACCOUNT NAME>` and run below commands to update SPN and encryption.
>
> 1. Set SPN - `Set-ADcomputer -Identity <account-name> -ServicePrincipalNames @{Add="cifs/it1sccm011gaa.file.core.windows.net"}`
> 2. Set AES256 encryption - `Set-ADComputer -Identity <account-name>$ -Server ent -KerberosEncryptionType "AES256"`
> 3. Initially set a temp password for the computer account and we will share the actual password after setting the feature on storage account. (AD have one time password reset policy enabled)
>
> ```
> Set-ADAccountPassword -Identity <account-name>$ -Reset -NewPassword (ConvertTo-SecureString -
> AsPlainText "<ENTER-temp-pass-here>" -Force)
> ```
>
> 1. Share SID for computer account. Run below command
>
> For Multiple -> `Get-ADComputer -Filter "name -like 'it1afs*'" | Select Name,SID`
>
> For Single -> `Get-ADComputer -Filter "name -eq '<account-name>'" -Properties sid | select name, sid`

2. Set the feature flag on the target storage account with AD domain information and storage account SID which we got from IAM Team.

```
Set-AzStorageAccount `
        -ResourceGroupName "<Storage-account-RG>" `
        -Name "<STORAGE-ACCOUNT-NAME>" `
        -EnableActiveDirectoryDomainServicesForFile $true `
        -ActiveDirectoryDomainName "ent.bhpbilliton.net" `
        -ActiveDirectoryNetBiosDomainName "ent.bhpbilliton.net" `
        -ActiveDirectoryForestName "ent.bhpbilliton.net" `
        -ActiveDirectoryDomainGuid "7d851265-2d75-4385-9609-
df9157b3e7f6" `
        -ActiveDirectoryDomainsid "S-1-5-21-1427962766-63821886-
607533713" `
        -ActiveDirectoryAzureStorageSid "<UPDATE-SID-HERE>"
```

2.1 Create the Kerb key for the storage account using below commands.

```
New-AzStorageAccountKey -ResourceGroupName "<RG-Name>" -Name "<account-name>" -KeyName kerb1
```

```
Get-AzStorageAccountKey -ResourceGroupName "<RG-Name>" -Name "<account-name>" -ListKerbKey | where-object
{$_.Keyname -contains "kerb1"}
```

> 3. Provide below command to IAM Team to update Kerb key as password for computer account.
>
> ```
> Set-ADAccountPassword -Identity <account-name>$ -Reset -NewPassword (ConvertTo-SecureString -
> AsPlainText "<ENTER-KERB-KEY-HERE>" -Force)
> ```

3. Now assign an Azure role to allow access to your Pa5 account/ad group on file share, using the Azure Portal.

#Below command Assign the built-in role to the target identity(ad group/pa5account): **Storage File Data SMB Share Reader**, **Storage File Data SMB Share Contributor**, **Storage File Data SMB Share Elevated Contributor**

```
az role assignment create --role "<role-name>" --assignee <user-principal-name> --scope "/subscriptions
/<subscription-id>/resourceGroups/<resource-group>/providers/Microsoft.Storage/storageAccounts/<storage-
account>/fileServices/default/fileshares/<share-name>"
```

Refer Control access to Azure file shares - on-premises AD DS authentication | Microsoft Docs to understand more around Azure File Share access control.

4. Now ask Application Team to try accessing the file share using their AD accounts (PA5 ID).

5. In case of any authentication/network error, run below debug command. This debug cmdlet conduct a set of basic checks on your AD configuration with the logged on AD user.

```
Debug-AzStorageAccountAuth -StorageAccountName $StorageAccountName -ResourceGroupName $ResourceGroupName -
Verbose
```

You can also refer Enable AD DS authentication to Azure file shares | Microsoft Docs if you require any detailed information.

- Just in case you see below error, consider to update computer account password (generate new Kerb key)

> The Kerberos client received a KRB_AP_ERR_MODIFIED error from the server it1sccm021gjg. The target name used was cifs /it1sccm021gjg.file.core.windows.net. This indicates that the target server failed to decrypt the ticket provided by the client. This can occur when the target server principal name (SPN) is registered on an account other than the account the target service is using. Ensure that the target SPN is only registered on the account used by the server. This error can also happen if the target service account password is different than what is configured on the Kerberos Key Distribution Center for that target service. Ensure that the service on the server and the KDC are both configured to use the same password. If the server name is not fully qualified, and the target domain (file.core.windows.net) is different from the client domain (ENT.BHPBILLITON.NET), check if there are identically named server accounts in these two domains, or use the fully-qualified name to identify the server.

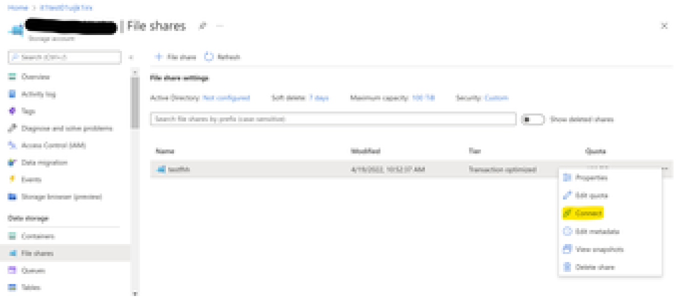
- To test File share mount on windows, try using below command and set NTFS permissions (if required)

```
net use : <storage-account-name>.file.core.windows.net
```

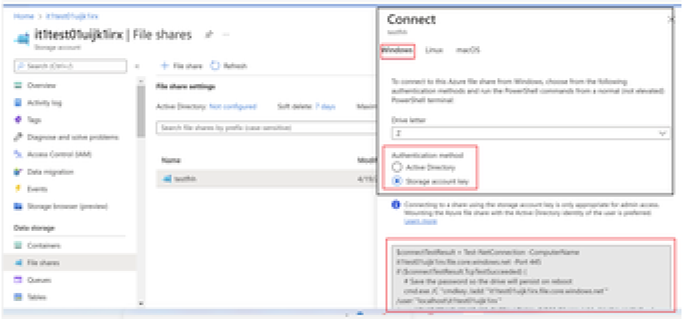
Connect File share from Windows and Linux

To connect to this Azure file share from **Windows** follow the below steps -

- Go to file share, click to see more (3 dots) and select connect.



- On the connect screen, select Windows or Linux depending your requirement.
- For Windows you can select authentication method as storage account key or active directory. Select the desired authentication type and copy the Powershell script.



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: Z

Authentication method
○ Active Directory
● Storage account key

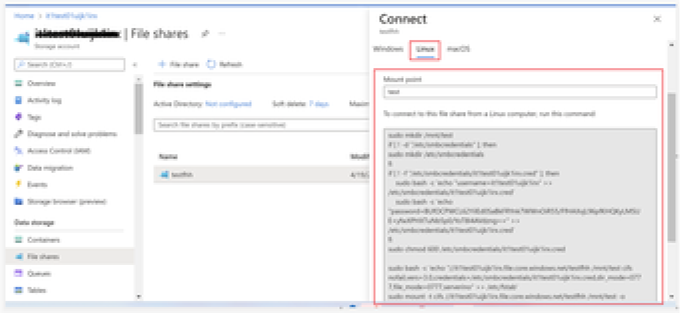
Connecting to a share using the storage account key is only appropriate for admin access. Mounting the Azure file share with the Active Directory identity of the user is preferred. Learn more

```
$connectTestResult = Test-NetConnection -ComputerName
it1test01uijk1lrx.file.core.windows.net -Port 445
if ($connectTestResult.TcpTestSucceeded) {
    # Save the password so the drive will persist on reboot
    cmd.exe /C "cmdkey /add:`"it1test01uijk1lrx.file.core.windows.net`"
/user:`"localhost\it1test01uijk1lrx`"
```

- Now you can directly run this Powershell script on Windows (VM) to mount the file share.

To connect to this Azure file share from **Linux** follow the below steps

- Select Linux tab on the connect page, provide mount point name and now copy the shell script and run it on Linux VM where you want to mount the file share.



Mount point

test

To connect to this file share from a Linux computer, run this command:

```
sudo mkdir /mnt/test
if [ ! -d "/etc/smbcredentials" ]; then
sudo mkdir /etc/smbcredentials
fi
if [ ! -f "/etc/smbcredentials/[illegible].cred" ]; then
    sudo bash -c 'echo "username=[illegible]" >>
/etc/smbcredentials/[illegible].cred'
    sudo bash -c 'echo
"password=[illegible]" >>
/etc/smbcredentials/[illegible].cred'
fi
sudo chmod 600 /etc/smbcredentials/[illegible].cred

sudo bash -c 'echo "//[illegible].file.core.windows.net/[illegible] /mnt/test cifs
nofail,vers=3.0,credentials=/etc/smbcredentials/[illegible].cred,dir_mode=077
7,file_mode=0777,serverino" >> /etc/fstab'
sudo mount -t cifs //[illegible].file.core.windows.net/[illegible] /mnt/test -o
```