AAD on Azure Database for PostgreSQL Flexible Server

Prerequisites:

The below steps 1 and 2 need to be performed by AAD tenant administrator and this is **one time activity per tenant**.

Install AzureAD PowerShell: AzureAD Module
Step 1: Connect-AzureAD -TenantId <customer tenant id>
Step 2: New-AzureADServicePrincipal -AppId 5657e26c-cc92-45d9-bc47-9da6cfdb4ed9

The below step can be performed by anyone who wants to create a AAD enabled flexible server
Step 3: Create a new PostgreSQL Flexible server instance. Enable AAD authentication during server provisioning or you can enable it after creating the server using the steps below.

Limitations:

1. Private preview is only available for the customers who are under NDA.

2. AAD feature can only be enabled for newly created servers during private preview. AAD can be enabled for existing servers during AAD public preview.

3. PGbouncer is currently unsupported in private preview.

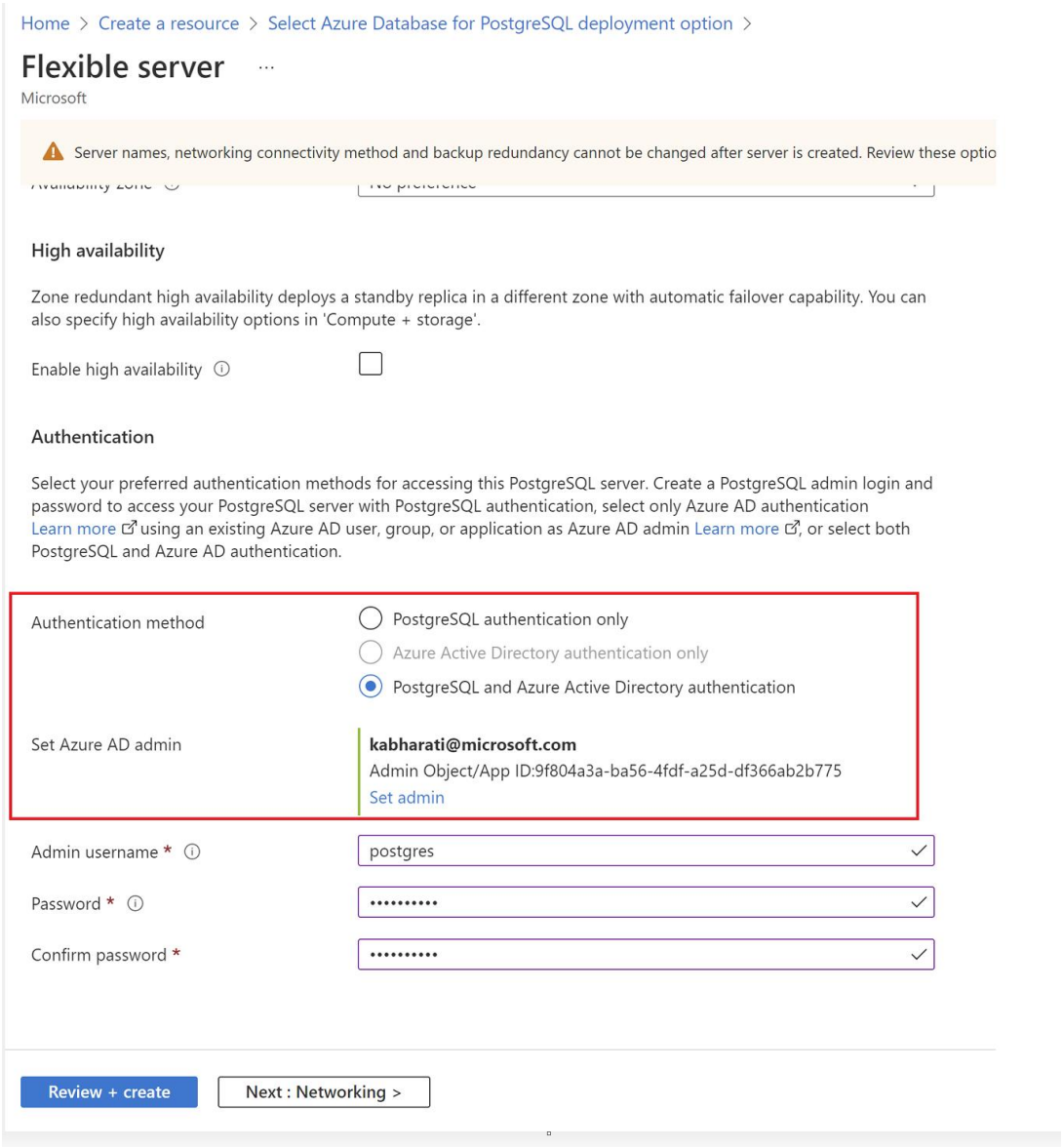
4. Disabling PostgreSQL authentication (support for Azure AD authentication only) is unavailable in private preview and will be available during public preview.

5. Private preview supports PG version 11, 12,13 and we will support PG 14 starting from next month.

Enable AAD on Azure Database for PostgreSQL Flexible Server during Server Provisioning

Please check **PostgreSQL and Azure Active Directory authentication** during Flexible server creation.

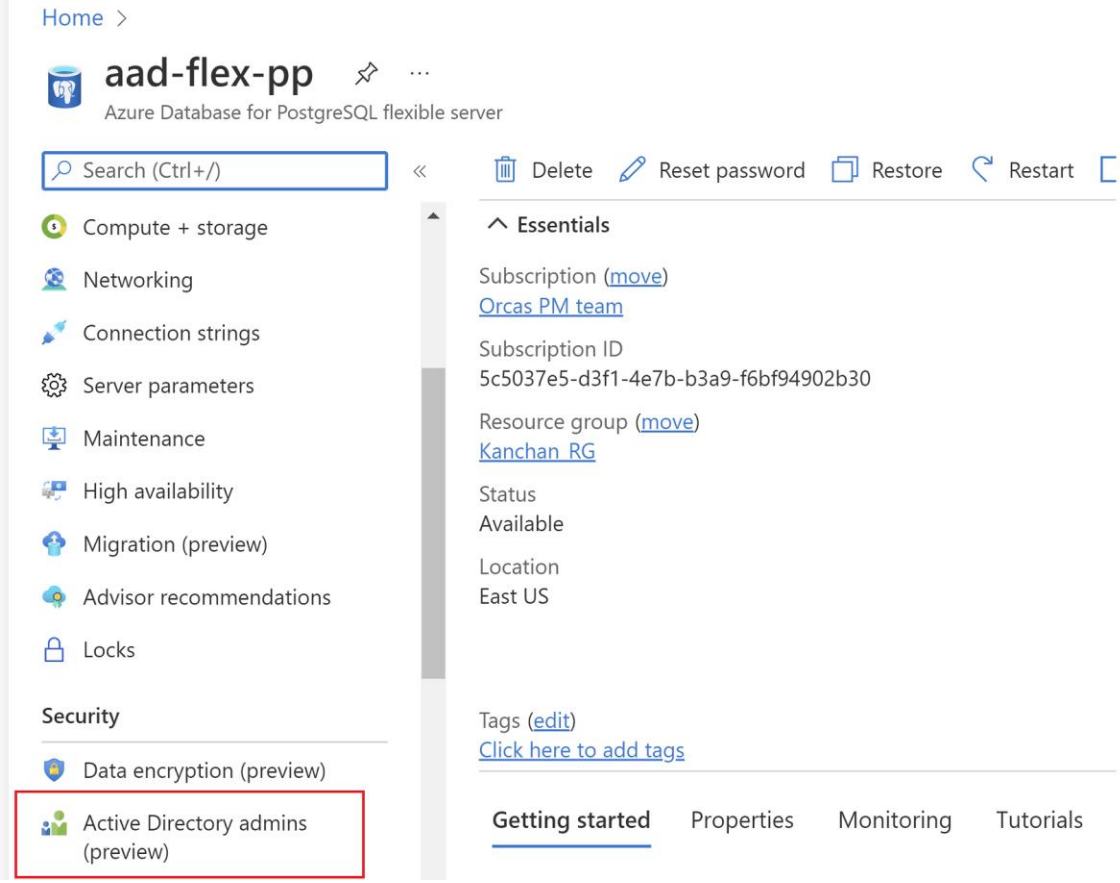
Click **Set admin** and add your Azure AD admin or group.



Enable AAD on Azure Database for PostgreSQL Flexible Server Post Server Creation

a. In the Azure Portal click on your Azure Database for PostgreSQL Flexible Server and under security you should see **Active Directory admins (Preview)** tab in the left-hand pane if your subscription has private preview access.

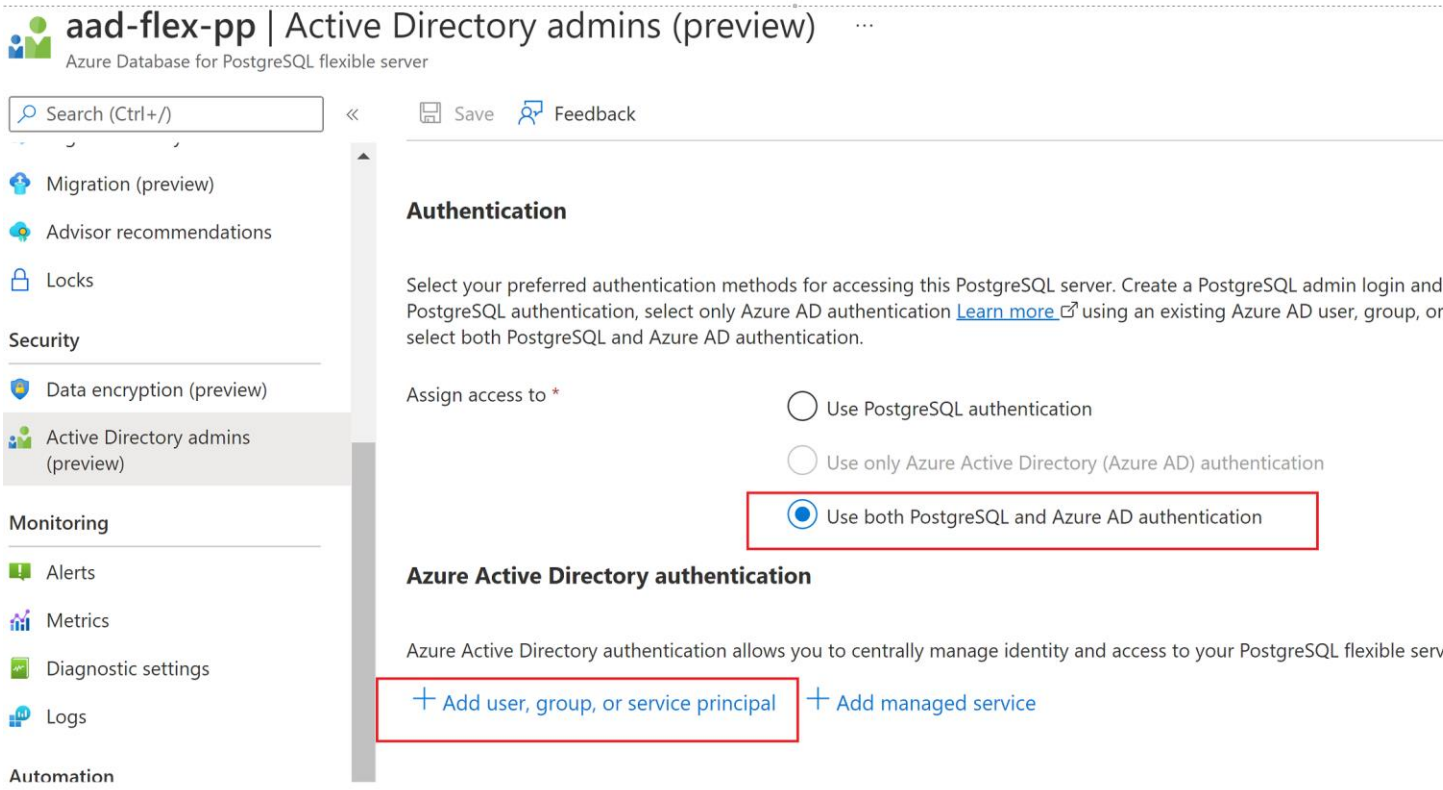
If you don't see this, then please reach out to AskAzureDBforAADPGFlex@microsoft.com and request AAD private preview access.

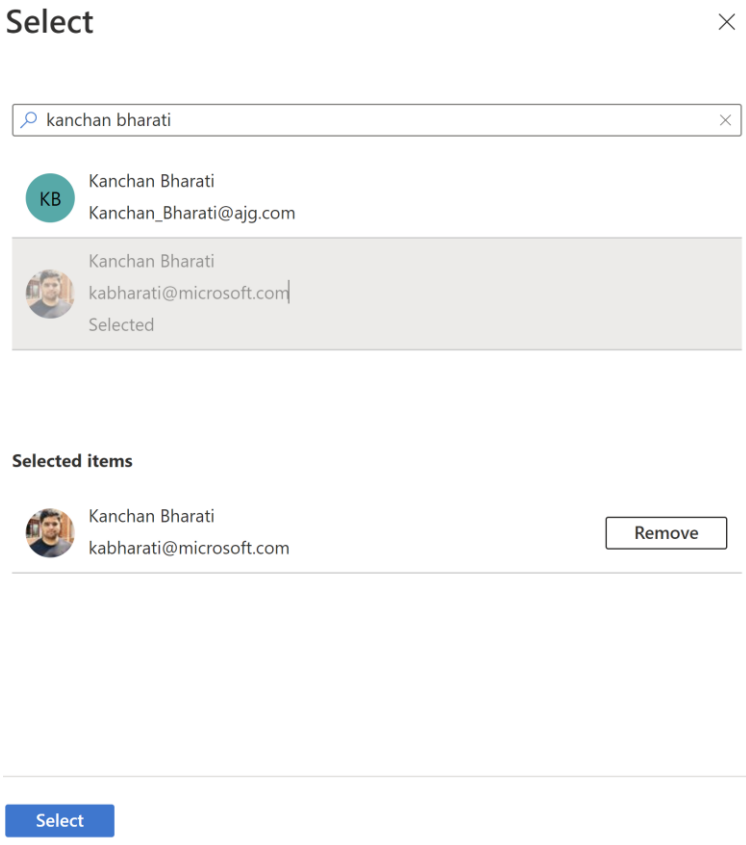


b. Select **Use both PostgreSQL and Azure AD authentication** button to enable Azure AD authentication for your flexible server.

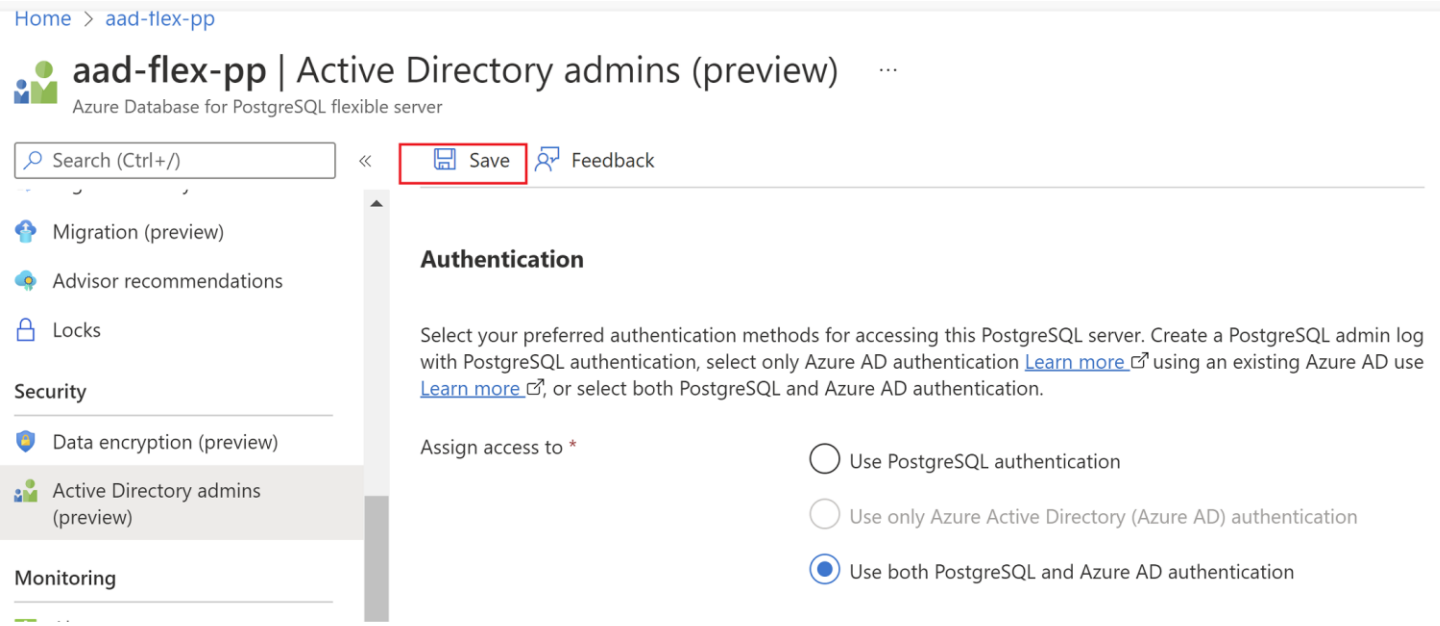
c. You can also add AAD admin by clicking on **Add user, group or service principal** selecting the appropriate **AAD admin or group.**

Note: Use **only Azure Active Directory (Azure AD) authentication** is currently greyed out and this feature will be supported during public preview (July).


aad-flex-pp | Active Directory admins (preview)
Azure Database for PostgreSQL flexible server
Search (Ctrl+/)
Save
Feedback
Migration (preview)
Advisor recommendations
Locks
Security
Data encryption (preview)
Active Directory admins (preview)
Monitoring
Alerts
Metrics
Diagnostic settings
Logs
Automation
Authentication
Select your preferred authentication methods for accessing this PostgreSQL server. Create a PostgreSQL admin login and
PostgreSQL authentication, select only Azure AD authentication Learn more using an existing Azure AD user, group, or
select both PostgreSQL and Azure AD authentication.
Assign access to *
Use PostgreSQL authentication
Use only Azure Active Directory (Azure AD) authentication
Use both PostgreSQL and Azure AD authentication
Azure Active Directory authentication
Azure Active Directory authentication allows you to centrally manage identity and access to your PostgreSQL flexible serv
Add user, group, or service principal
Add managed service

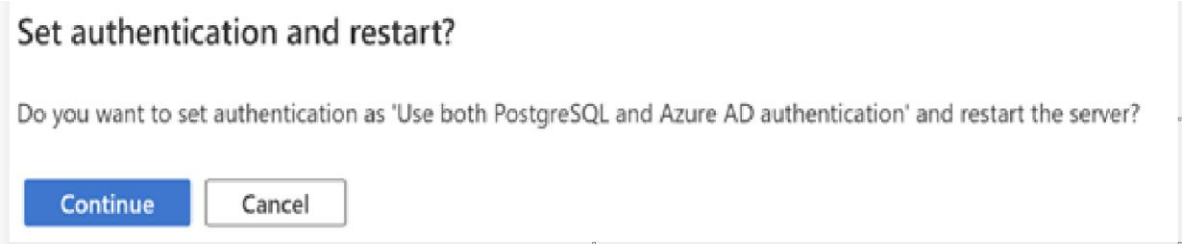

Select
kanchan bharati
KB
Kanchan Bharati
Kanchan_Bharati@ajg.com
Kanchan Bharati
kabharati@microsoft.com
Selected
Selected items
Kanchan Bharati
kabharati@microsoft.com
Remove
Select

d. Click on **the Save** button

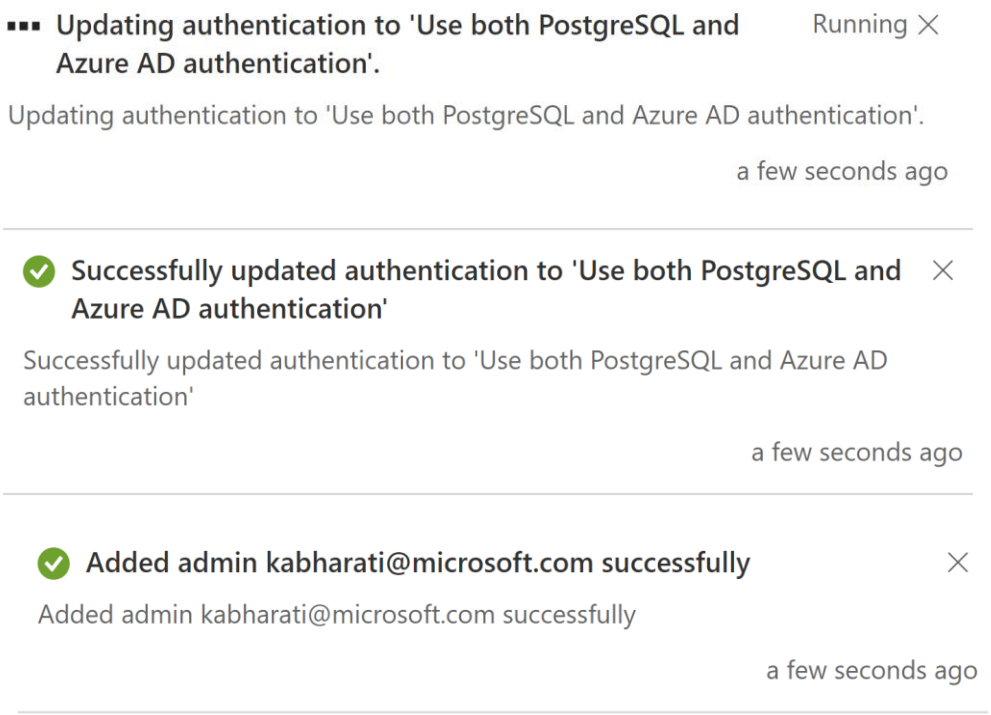


e. Click on **continue** to enable AAD for your flexible server

NOTE: Enabling/ Disabling AAD will result in a service restart so please plan accordingly.

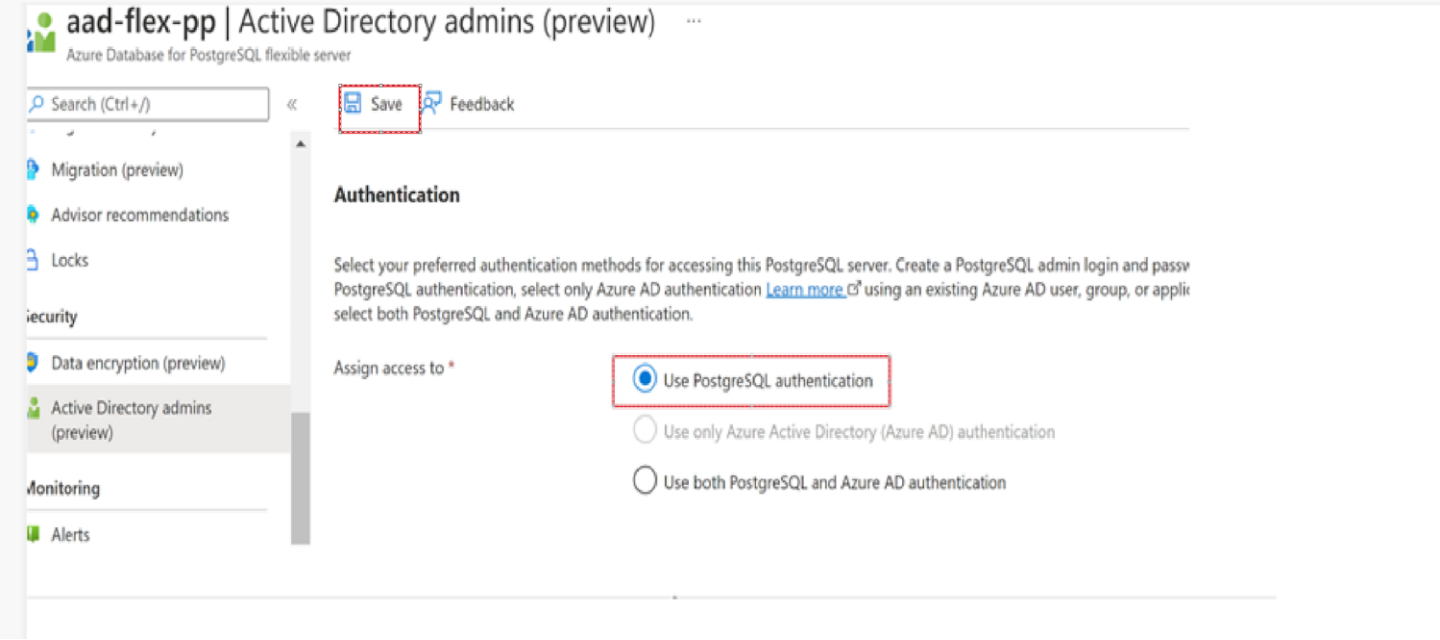


f. You should see below notifications once AAD is enabled and AAD admin is successfully configured. This operation should not take more than 2-3 mins approximately.

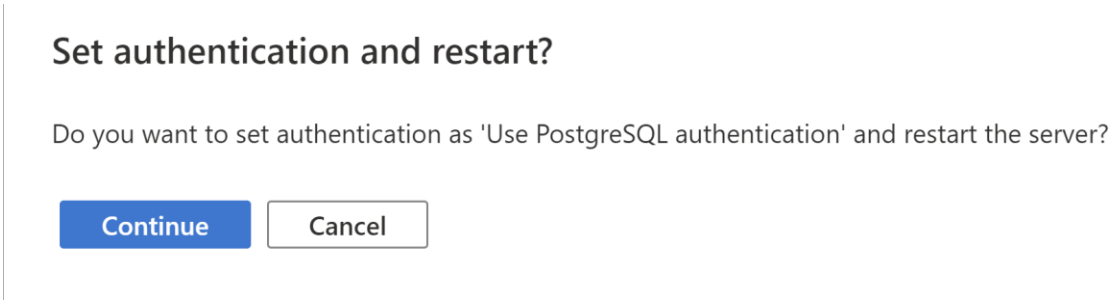


Disable AAD on Azure Database for PostgreSQL Flexible Server

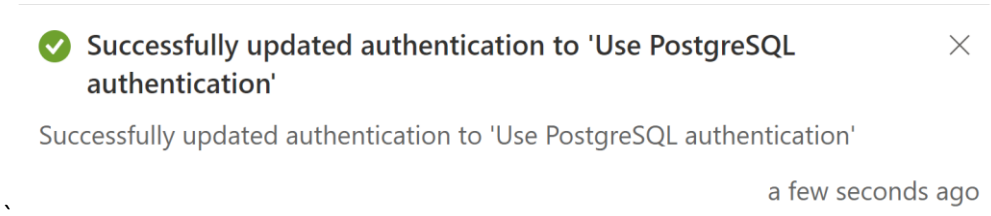
a. To disable AAD change Authentication mode from **Use both PostgreSQL and Azure AD authentication** → **Use PostgreSQL authentication** and click **Save**



b. Click **continue**



c. Your PostgreSQL flexible server is now configured to use PostgreSQL authentication.



`

Adding Multiple AAD Admins / Groups to your Flexible Server.

Flexible Server supports adding multiple add admins which was missing in our Single Server offering and you can use the steps below to configure this.

1. Go to **Active Directory admins (Preview)** tab in the left-hand pane.
2. Click on **Add user, group or service principal** and select a single AAD admin or multiple AAD admins / groups as per your requirement and click on **select**
3. Once added you should see multiple AAD admins as shown in below screenshot.

Select

PGFlex

pgflex
8d015e40-9ba6-4b69-92ad-c5e8a8710b99

P
PGFlexAADadmins
PGFlexAADadmins@service.microsoft.com
Selected

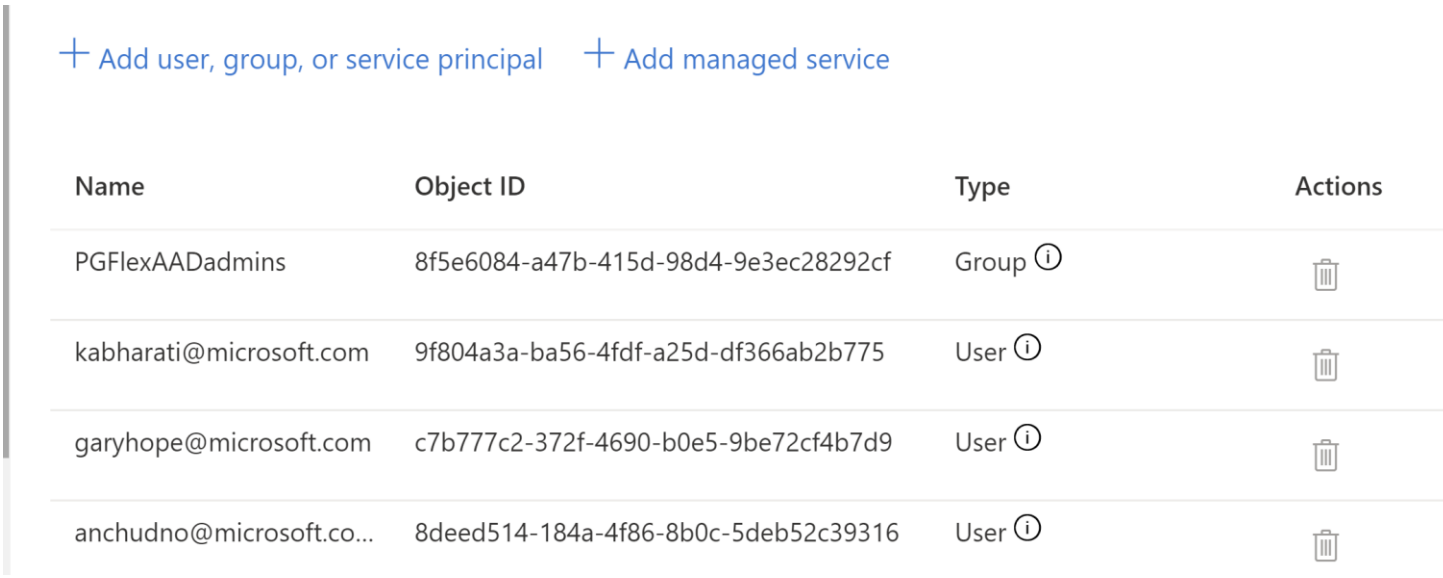
Selected items


Andrey Chudnovskiy
anchudno@microsoft.com
Remove


Gary Hope
garyhope@microsoft.com
Remove

P
PGFlexAADadmins
PGFlexAADadmins@service.microsoft.com
Remove

Select



NOTE: Deleting AAD admin from the portal is currently not supported (delete buttons are greyed out) in private preview and but you can delete the AAD admin directly by logging into your Azure PostgreSQL Flexible Server and deleting the corresponding role.

Connect to Azure Database for PostgreSQL Flexible Server using AAD authentication

Authenticate with Azure AD as a single user

Step 1: Login to the user's Azure subscription

Start by authenticating with Azure AD using the Azure CLI tool. This step is not required in Azure Cloud Shell.

```
az login
```

The command will launch a browser window to the Azure AD authentication page. It requires you to give your Azure AD user ID and the password.

Step 2: Retrieve Azure AD access token

Invoke the Azure CLI tool to acquire an access token for the Azure AD authenticated user from step 1 to access Azure Database for PostgreSQL.

Example (for Public Cloud):

```
az account get-access-token --resource https://ossrdbms-aad.database.windows.net
```

The above resource value must be specified exactly as shown. For other clouds, the resource value can be looked up using:

```
az cloud show
```

For Azure CLI version 2.0.71 and later, the command can be specified in the following more convenient version for all clouds:

```
az account get-access-token --resource-type oss-rdbms
```

After authentication is successful, Azure AD will return an access token:

JSONCopy

```
{
  "accessToken": "TOKEN",
  "expiresOn": "...",
  "subscription": "...",
  "tenant": "...",
  "tokenType": "Bearer"
}
```

The token is a Base 64 string that encodes all the information about the authenticated user, and which is targeted to the Azure Database for PostgreSQL service.

Step 3: Use token as password for logging in with client psql

When connecting you need to use the access token as the PostgreSQL user password.

When using the psql command line client, the access token needs to be passed through the PGPASSWORD environment variable, since the access token exceeds the password length that psql can accept directly:

Windows Example:

CMD

set PGPASSWORD=<copy/pasted TOKEN value from step 2>

PowerShell

$env:PGPASSWORD='<copy/pasted TOKEN value from step 2>'

Linux/macOS Example:

shell

export PGPASSWORD=<copy/pasted TOKEN value from step 2>

Now you can initiate a connection with Azure Database for PostgreSQL like you normally would:

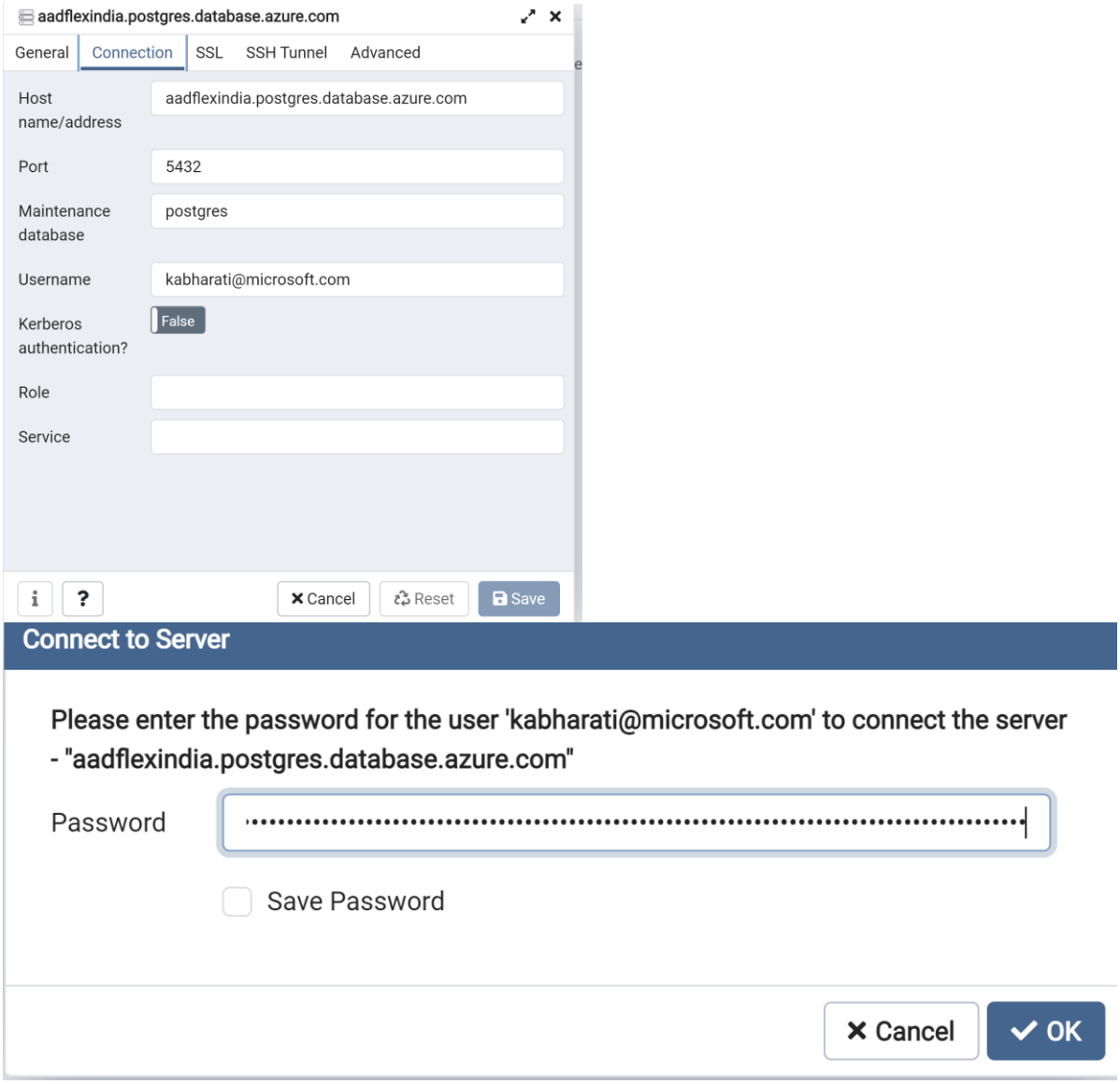
shell

psql "host=mydb.postgres... user=user@tenant.onmicrosoft.com@mydb dbname=postgres sslmode=require"

Step 4: Use token as a password for logging in with PgAdmin

To connect using Azure AD token with **pgAdmin** you need to follow the next steps:

1. Uncheck the connect now option at server creation.
2. Enter your server details and username in the connection tab and save. **Unlike Azure PostgreSQL Single Server you no longer need to append @servername to your username.**
3. From the browser menu, click connect to the Azure Database for PostgreSQL server
4. Enter the AD token password when prompted.



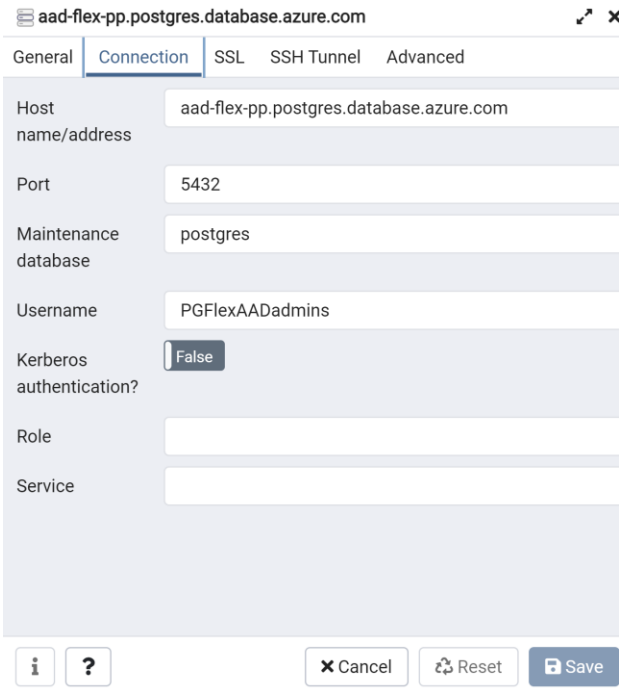
Important considerations when connecting:

- user@tenant.onmicrosoft.com is the name of the Azure AD user
- Make sure to use the exact way the Azure user is spelled - as the Azure AD user and group names are case sensitive.
- If the name contains spaces, use \ before each space to escape it.
- The access token validity is anywhere between 5 minutes to 60 minutes. We recommend you get the access token just before initiating the login to Azure Database for PostgreSQL.

You are now authenticated to your Azure Database for PostgreSQL server using Azure AD authentication.

Authenticate with Azure AD as a group member

1. To login as AD group please follow the same steps 1-4 mentioned above to retrieve the token.
2. Uncheck the connect now option at server creation.
3. Enter your server details and AD group name in the connection tab and save. **Unlike single server you no longer need to append @servername to your groupname.**
4. From the browser menu, click connect to the Azure Database for PostgreSQL server
5. Enter the AD token password when prompted.



Important considerations when connecting as a group member:

- groupname is the name of the Azure AD group you are trying to connect
- Make sure to use the exact way the Azure AD group name is spelled.
- Azure AD user and group names are case sensitive
- When connecting as a group, use only the group name not the alias of a group member.

- If the name contains spaces, use \ before each space to escape it.
- The access token validity is anywhere between 5 minutes to 60 minutes. We recommend you get the access token just before initiating the login to Azure Database for PostgreSQL.

You are now authenticated to your PostgreSQL server using Azure AD authentication.

Vnet enabled Servers :

AAD is a multi-tenant application and required outbound connectivity to perform certain operations like adding AAD admins/ groups etc. If your flexible server is Vnet enabled please plan on adding one or both rules below to ensure AAD connections are not failing depending on your network topology.

* An outbound NSG rule to allow virtual network traffic to reach AzureActiveDirectory service tag only.

*If you're using a proxy then please add a new firewall rule to allow http/s traffic to reach AzureActiveDirectory service tag only.