Get an Azure Active Directory token using a service principal

07/15/2020 • 4 minutes to read •

In this article

This article describes how a service principal defined in Azure Active Directory (Azure AD) can also act as a principal on which authentication and authorization policies can be enforced in Azure Databricks. Service principals in an Azure Databricks workspace can have different fine-grained access control than regular users (user principals).

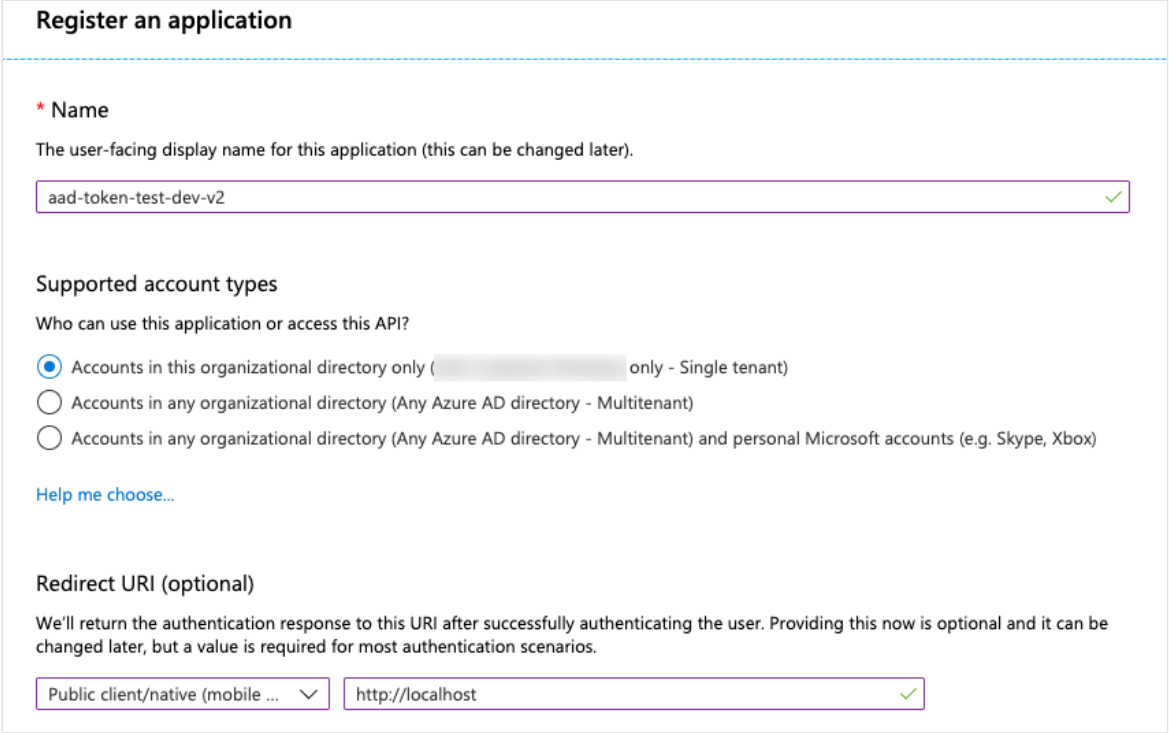
A service principal acts as a client role and uses the OAuth 2.0 code grant flow to authorize to Azure Databricks resources.

You can manage service principals using the Databricks SCIM API (ServicePrincipals) API or use the following procedure in Azure portal.

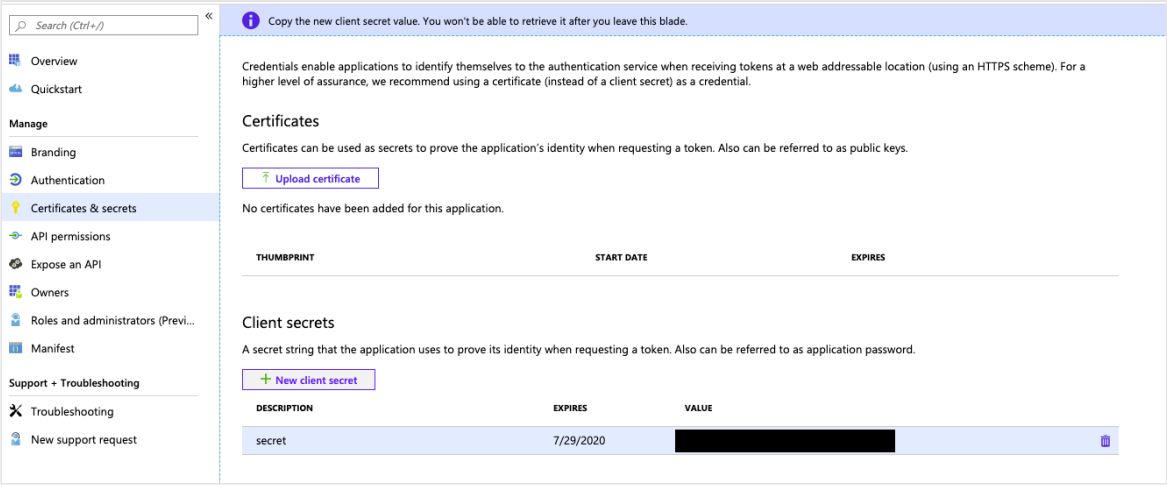
You can also use the Azure Active Directory Authentication Library (ADAL) to programmatically get an Azure AD access token for a user. See Get an Azure Active Directory token using Azure Active Directory Authentication Library.

Provision a service principal in Azure portal

1. Log in to Azure portal.
2. Navigate to **Azure Active Directory** > **App Registrations** > **New Registrations**. You should see a screen similar to this:



3. Click **Certificates & secrets** and generate a new client secret.



4. Copy and store that secret in a secure place as this secret is the password for your application.

5. Click **Overview** to look at details like Application (client) ID and Directory (tenant) ID.

Use an app identity to access resources covers how you can provision an application (service principal) in Azure AD.

Get an Azure Active Directory access token

To access the Databricks REST API with the service principal, you get an Azure AD access token for the service principal. You can use the client credentials flow to get an access token (with the AzureDatabricks login application as the resource).

Replace the following parameters in the `curl` request:

Parameter	Description
Tenant ID	Tenant ID in Azure AD. Go to **Azure Active Directory > Properties > Directory ID**.
Client ID	The application (service principal) ID of the application you registered in Provision a service principal in Azure portal.
Azure Databricks resource ID	`2ff814a6-3304-4ab8-85cb-cd0e6f879c1d`.
Application secret	The secret generated for the application.

Bash Copy

```bash
curl -X GET -H 'Content-Type: application/x-www-form-urlencoded' \
-d 'grant_type=client_credentials&client_id=<client-id>&resource=
<azure_databricks_resource_id>&client_secret=<application-secret>' \
https://login.microsoftonline.com/<tenant-id>/oauth2/token
```

The response should look like:

JSON Copy

```json
{
  "token_type": "Bearer",
  "expires_in": "599",
  "ext_expires_in": "599",
  "expires_on": "1575500666",
  "not_before": "1575499766",
  "resource": "2ff8...f879c1d",
  "access_token": "ABC0eXAiOiJKV1Q......un_f1mSgCHlA"
}
```

The `access_token` in the response is the Azure AD *access token*.

Use an Azure AD access token to access the Databricks REST API

Admin user login

If any of the following are true, you must be in a Contributor or Owner role on the workspace resource in Azure to log in using the service principal access token:

- The service principal does not belong to the workspace.
- The service principal belongs to the workspace, but you want to add it automatically as an admin user.
- You do not know the org ID of your workspace but you know the workspace resource ID in Azure.

You must provide:

- The `X-Databricks-Azure-Workspace-Resource-Id` header, which contains the ID of the workspace resource in Azure. You construct the ID using the Azure subscription ID, resource group name, and workspace resource name.
- A management access token for the Azure Resource Management endpoint.

Get the Azure Management Resource endpoint token

Replace the following parameters in the `curl` request:

Parameter	Description
Tenant ID	Tenant ID in Azure AD. Go to **Azure Active Directory** > **Properties** > **Directory ID**.
Client ID	The application (service principal) ID of the application you registered in Provision a service principal in Azure portal.
Management Resource endpoint	`https://management.core.windows.net/`.
Application secret	The secret generated for the application.

Bash Copy

```bash
curl -X GET -H 'Content-Type: application/x-www-form-urlencoded' \
-d 'grant_type=client_credentials&client_id=<client-id>&resource=
<management-resource-endpoint>&client_secret=<application-secret>' \
https://login.microsoftonline.com/<tenantid>/oauth2/token
```

The response should look like:

JSON Copy

```
{
  "token_type": "Bearer",
  "expires_in": "599",
  "ext_expires_in": "599",
  "expires_on": "1575500666",
  "not_before": "1575499766",
  "resource": "https://management.core.windows.net/",
  "access_token": "LMN0eXAiOiJKV1Q......un_f1mSgCHlA"
}
```

The `access_token` in the response is the *management endpoint access token*.

Use the management endpoint access token to access the Databricks REST API

Parameter	Description
Databricks instance	URL of your Databricks instance.
Access token	Access token obtained in Get an Azure Active Directory access token.
Management access token	Management endpoint access token obtained in Get the Azure Management Resource endpoint token.
Subscription ID	Subscription ID of the Azure Databricks resource.
Resource group name	Name of the Azure Databricks resource group.
Workspace name	Name of the Azure Databricks workspace.

Bash Copy

```bash
curl -X GET \
-H 'Authorization: Bearer <access-token>' \
-H 'X-Databricks-Azure-SP-Management-Token: <management-access-token>' \
-H 'X-Databricks-Azure-Workspace-Resource-Id:
/subscriptions/<subscription-id>/resourceGroups/<resource-group-
name>/providers/Microsoft.Databricks/workspaces/<workspace-name>' \
https://<databricks-instance>/api/2.0/clusters/list
```

A sample request will look like:

Bash Copy

```
curl -X GET \
-H 'Authorization:Bearer ABC0eXAiOiJKV1Q......un_f1mSgCHlA' \
-H 'X-Databricks-Azure-SP-Management-Token:
LMN0eXAiOiJKV1Q......un_f1mSgCHlA' \
-H 'X-Databricks-Azure-Workspace-Resource-Id:
/subscriptions/3f2e4d...2328b/resourceGroups/Ene...RG/providers/Microsoft
.Databricks/workspaces/demo-databricks' \
https://<xxxx>.azuredatabricks.net/api/2.0/clusters/list
```

Non-admin user login

> **Note**
>
> Prior to this login, the service principal must be added to the workspace either as part of the **admin user login** or using the **Add service principal** endpoint.

Use the access token as the `Bearer` token and provide the org ID of the workspace in the `X-Databricks-Org-Id` header.

Parameter	Description
Databricks instance	URL of your Databricks instance. See Get workspace, cluster, notebook, model, and job identifiers.
Access token	Token returned from the request in Get an Azure Active Directory access token.
Databricks workspace org ID	The org ID of the workspace. See Get workspace, cluster, notebook, model, and job identifiers.

Use an access token to access the Databricks REST API

Bash Copy

```
curl -X GET \
-H 'Authorization: Bearer <access-token>' \
-H 'X-Databricks-Org-Id: <workspace-org-id>' \
https://<databricks-instance>/api/2.0/clusters/list
```