

[Cosmos DB] CosmosDB linked service with Service Principal, MI, UAMI authentication test connection failed due to RBAC

Last updated by | Jackie Huang | Jan 4, 2022 at 12:24 AM PST

Contents

- [Issue](#)
- [Solution](#)
 - [Option 1: Azure CLI](#)
 - [Option 2: PowerShell](#)
 - [Option 3: ARM Template](#)
- [Tips](#)
 - [What are the IDs when using service principal authenticati...](#)
 - [What is the managed identity object ID?](#)
- [Enjoy](#)

Issue

When customer use Cosmos linked server with managed identity authentication, found the following error

```
CosmosDbSqlApi operation Failed.  
ErrorMessage:  
Request blocked by Auth cosmos-supplychainhub-dev-aue:  
Request is blocked because principal [b89axxxx-xxxx-xxxx-xxxx-xxxxxxxxxxx] does not have required RBAC permissi  
Learn more: https://aka.ms/cosmos-native-rbac.  
ActivityId: 7e13xxxx-xxxx-xxxx-xxxx-xxxxxxxxxxx, Microsoft.Azure.Documents.Common/2.14.0, Windows/10.0.14393 co
```

Solution

This is a known issue, for now, if you add the custom role from the portal, you will realize Microsoft.DocumentDB/databaseAccounts/readMetadata doesn't exist. The workaround is we need to create the custom role by Azure CLI, PowerShell or ARM template.

Make sure to properly install AZ in local PC or start a PS window from Azure portal.

Option 1: Azure CLI

1. Create a custom role definition

```
// Save the following content to role-definition.json file
{
  "RoleName": "AdfReadWriteRole",
  "Type": "CustomRole",
  "AssignableScopes": ["/"],
  "Permissions": [{
    "DataActions": [
      "Microsoft.DocumentDB/databaseAccounts/readMetadata",
      "Microsoft.DocumentDB/databaseAccounts/sqlDatabases/containers/*",
      "Microsoft.DocumentDB/databaseAccounts/sqlDatabases/containers/items/*"
    ]
  }]
}
```

```
resourceGroupName=<myResourceGroup>
accountName=<myCosmosAccount>
```

```
az cosmosdb sql role definition create --account-name $accountName --resource-group $resourceGroupName --body
```

2. Assign the role to the service principal or managed identity

```
roleDefinitionId='<roleDefinitionId>' # as fetched above
principalId='<service principal or managed identity object ID>'
```

```
az cosmosdb sql role assignment create --account-name $accountName --resource-group $resourceGroupName --scope
```

For more details, please refer to [Configure role-based access control for your Azure Cosmos DB account with Azure AD](#).

Option 2: PowerShell

1. Create a custom role definition


```
$resourceGroupName=<myResourceGroup>
$accountName=<myCosmosAccount>
$roleName=<AdfReadWriteRole>

New-AzCosmosDBSqlRoleDefinition -AccountName $accountName `
  -ResourceGroupName $resourceGroupName `
  -Type CustomRole -RoleName $roleName `
  -DataAction @( `
    'Microsoft.DocumentDB/databaseAccounts/readMetadata',
    'Microsoft.DocumentDB/databaseAccounts/sqlDatabases/containers/*', `
    'Microsoft.DocumentDB/databaseAccounts/sqlDatabases/containers/items/*') `
  -AssignableScope "/"
```

2. Assign the role to the service principal or managed identity

```
$roleDefinitionId="<roleDefinitionId>" # as fetched above
$principalId="<service principal or managed identity object ID>"
```

```
New-AzCosmosDBSqlRoleAssignment -AccountName $accountName `
  -ResourceGroupName $resourceGroupName `
  -RoleDefinitionId $roleDefinitionId `
  -Scope "/" `
  -PrincipalId $principalId
```

For more details, please refer to [Configure role-based access control for your Azure Cosmos DB account with Azure AD](#) .

Option 3: ARM Template

1. Create a custom role definition

PUT <https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.DocumentDB/databaseAccounts/{accountName}/roleDefinitions/{roleDefinitionId}>

```
// Request body
{
  "properties": {
    "roleName": "AdfReadWriteRole",
    "type": "CustomRole",
    "assignableScopes": [
      "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.DocumentDB/databaseAccounts/{accountName}"
    ],
    "permissions": [{
      "dataActions": [
        "Microsoft.DocumentDB/databaseAccounts/readMetadata",
        "Microsoft.DocumentDB/databaseAccounts/sqlDatabases/containers/*",
        "Microsoft.DocumentDB/databaseAccounts/sqlDatabases/containers/items/*"
      ]
    }]
  }
}
```

2. Assign the role to the service principal or managed identity

PUT <https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.DocumentDB/databaseAccounts/{accountName}/roleDefinitions/{roleDefinitionId}>

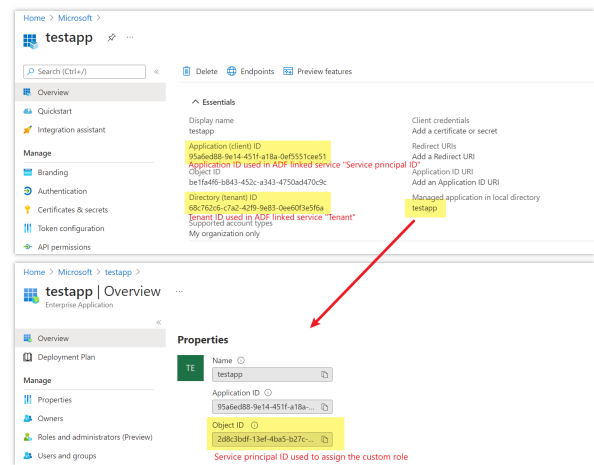
```
// Request body
{
  "properties": {
    "roleDefinitionId": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.DocumentDB/databaseAccounts/{accountName}/roleDefinitions/{roleDefinitionId}",
    "scope": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.DocumentDB/databaseAccounts/{accountName}",
    "principalId": "{service principal or managed identity object ID}"
  }
}
```

The roleDefinitionId and roleAssignmentId are customized random GUIDs.

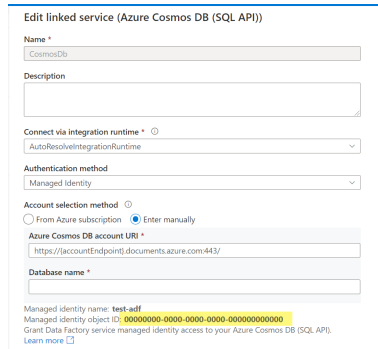
For more details, please refer to [Create Update Sql Role Definition](#) , [Create Update Sql Role Assignment](#) .

Tips

What are the IDs when using service principal authentication?

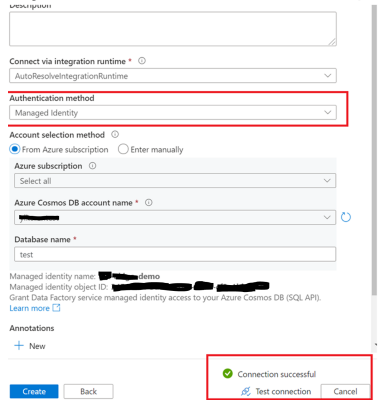


What is the managed identity object ID?



Enjoy

Try to test connect in ADF portal.



How good have you found this content?

4/6/23, 8:54 AM

[Cosmos DB] CosmosDB linked service with Service Principal, MI, UAMI authentication test connection failed due to RBAC - Overv...

