Grouper Entra ID Provisioner (Current) Azure O365

The info on this page applies to Grouper v4 and above.
For Azure provisioning in Grouper versions before Grouper 2.6, see this page.
Grouper versioning information is here.

Links

- Azure portal
- User documentation

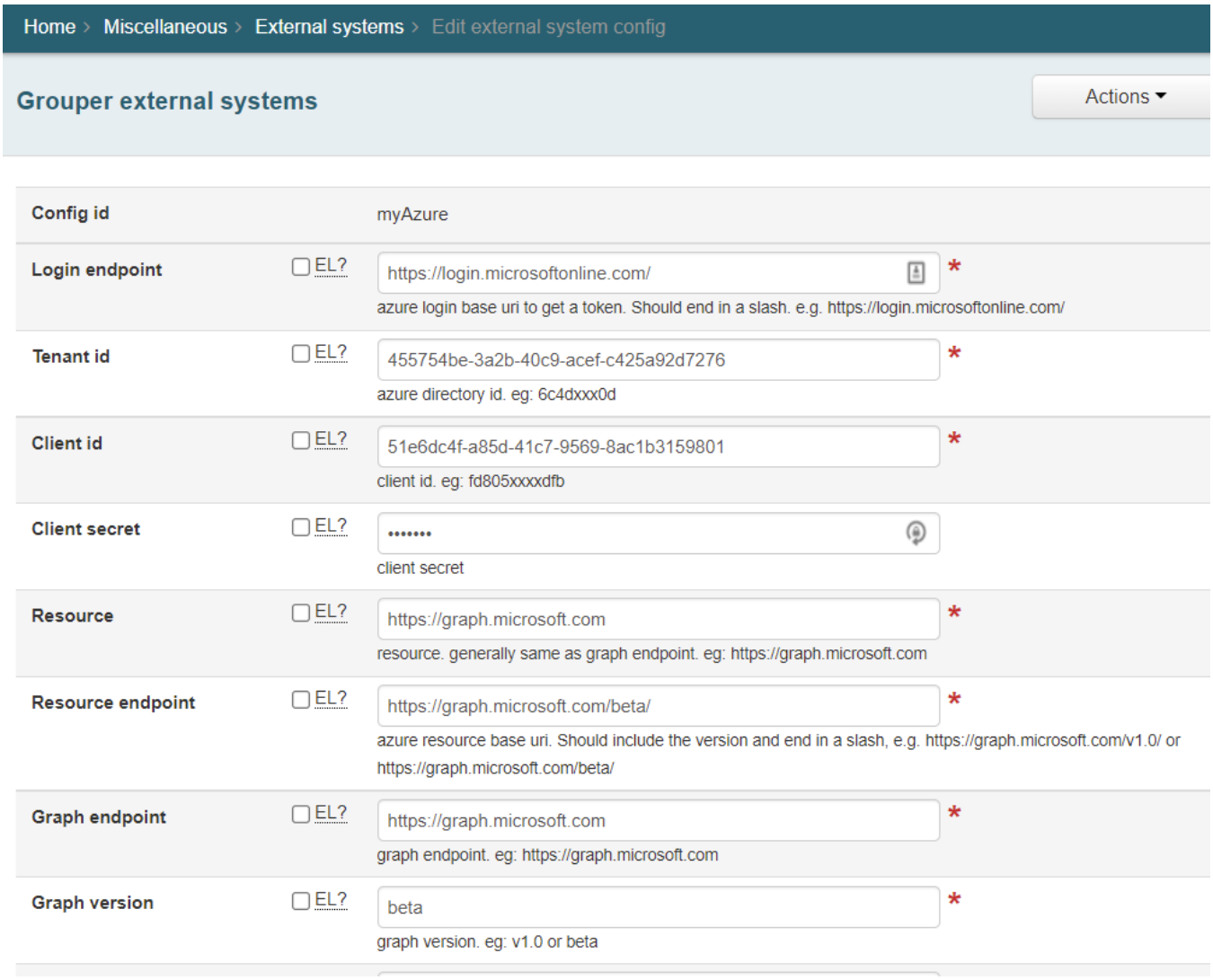
Demo

- Movie: Grouper provisioning framework Azure demo (v2.6.15): https://youtu.be/abTkJVBMr1M
- Config (grouper-loader.properties from version 2.6.15). Note, you should configure this in the provisioning configuration wizard.

```
grouper.azureConnector.myAzure.clientId = 7e590d54-d6af-4b07-b2aXXXXXXX
grouper.azureConnector.myAzure.clientSecret = *******
grouper.azureConnector.myAzure.graphEndpoint = https://graph.microsoft.com
grouper.azureConnector.myAzure.graphVersion = beta
grouper.azureConnector.myAzure.loginEndpoint = https://login.microsoftonline.com/
grouper.azureConnector.myAzure.resource = https://graph.microsoft.com
grouper.azureConnector.myAzure.resourceEndpoint = https://graph.microsoft.com/beta/
grouper.azureConnector.myAzure.tenantId = 5e7fa4df-8d24-4c33-bdaXXXXXXXXXXX

changeLog.consumer.provisioner_incremental_azureProvisioner.class = edu.internet2.middleware.grouper.
changeLog.esb.consumer.EsbConsumer
changeLog.consumer.provisioner_incremental_azureProvisioner.provisionerConfigId = azureProvisioner
changeLog.consumer.provisioner_incremental_azureProvisioner.publisher.class = edu.internet2.middleware.
grouper.app.provisioning.ProvisioningConsumer
changeLog.consumer.provisioner_incremental_azureProvisioner.publisher.debug = false
changeLog.consumer.provisioner_incremental_azureProvisioner.quartzCron = 0 * * * * ?

otherJob.provisioner_full_azureProvisioner.class = edu.internet2.middleware.grouper.app.provisioning.
GrouperProvisioningFullSyncJob
otherJob.provisioner_full_azureProvisioner.provisionerConfigId = azureProvisioner
otherJob.provisioner_full_azureProvisioner.quartzCron = 0 36 6 * * ?

provisioner.azureProvisioner.addDisabledFullSyncDaemon = true
provisioner.azureProvisioner.addDisabledIncrementalSyncDaemon = true
provisioner.azureProvisioner.azureExternalSystemConfigId = myAzure
provisioner.azureProvisioner.azureGroupType = true
provisioner.azureProvisioner.class = edu.internet2.middleware.grouper.app.azure.GrouperAzureProvisioner
provisioner.azureProvisioner.entityAttributeValueCache0entityAttribute = id
provisioner.azureProvisioner.entityAttributeValueCache0has = true
provisioner.azureProvisioner.entityAttributeValueCache0source = target
provisioner.azureProvisioner.entityAttributeValueCache0type = entityAttribute
provisioner.azureProvisioner.entityAttributeValueCacheHas = true
provisioner.azureProvisioner.entityMatchingAttribute0name = userPrincipalName
provisioner.azureProvisioner.entityMatchingAttributeCount = 1
provisioner.azureProvisioner.groupAttributeValueCache0groupAttribute = id
provisioner.azureProvisioner.groupAttributeValueCache0has = true
provisioner.azureProvisioner.groupAttributeValueCache0source = target
provisioner.azureProvisioner.groupAttributeValueCache0type = groupAttribute
provisioner.azureProvisioner.groupAttributeValueCacheHas = true
provisioner.azureProvisioner.groupMatchingAttribute0name = displayName
provisioner.azureProvisioner.groupMatchingAttributeCount = 1
provisioner.azureProvisioner.hasTargetEntityLink = true
provisioner.azureProvisioner.hasTargetGroupLink = true
provisioner.azureProvisioner.logAllObjectsVerbose = true
provisioner.azureProvisioner.logCommandsAlways = true
```

```
provisioner.azureProvisioner.numberOfEntityAttributes = 2
provisioner.azureProvisioner.numberOfGroupAttributes = 3
provisioner.azureProvisioner.operateOnGrouperEntities = true
provisioner.azureProvisioner.operateOnGrouperGroups = true
provisioner.azureProvisioner.operateOnGrouperMemberships = true
provisioner.azureProvisioner.provisioningType = membershipObjects
provisioner.azureProvisioner.selectAllEntities = true
provisioner.azureProvisioner.showAdvanced = true
provisioner.azureProvisioner.startWith = this is start with read only
provisioner.azureProvisioner.subjectSourcesToProvision = jdbc
provisioner.azureProvisioner.targetEntityAttribute.0.name = id
provisioner.azureProvisioner.targetEntityAttribute.1.name = userPrincipalName
provisioner.azureProvisioner.targetEntityAttribute.1.translateExpression = ${ grouperProvisioningEntity.
subjectId + '@mchyzergmail.onmicrosoft.com' }
provisioner.azureProvisioner.targetEntityAttribute.1.translateExpressionType = translationScript
provisioner.azureProvisioner.targetGroupAttribute.0.insert = false
provisioner.azureProvisioner.targetGroupAttribute.0.name = id
provisioner.azureProvisioner.targetGroupAttribute.0.showAdvancedAttribute = true
provisioner.azureProvisioner.targetGroupAttribute.0.showAttributeCrud = true
provisioner.azureProvisioner.targetGroupAttribute.0.update = false
provisioner.azureProvisioner.targetGroupAttribute.1.name = displayName
provisioner.azureProvisioner.targetGroupAttribute.1.translateExpressionType =
grouperProvisioningGroupField
provisioner.azureProvisioner.targetGroupAttribute.1.translateFromGrouperProvisioningGroupField =
extension
provisioner.azureProvisioner.targetGroupAttribute.2.name = mailNickname
provisioner.azureProvisioner.targetGroupAttribute.2.translateExpressionType =
grouperProvisioningGroupField
provisioner.azureProvisioner.targetGroupAttribute.2.translateFromGrouperProvisioningGroupField =
extension
```

External system



grouper-loader.properties for local testing

```
grouper.azureConnector.myAzure.clientId = 51e6dc4f-a85d-41c7-9569-8ac1b3159801
grouper.azureConnector.myAzure.clientSecret = *******
grouper.azureConnector.myAzure.graphEndpoint = https://graph.microsoft.com
grouper.azureConnector.myAzure.graphVersion = beta
grouper.azureConnector.myAzure.groupLookupAttribute = displayName
grouper.azureConnector.myAzure.groupLookupValueFormat = ${group.getName()}
grouper.azureConnector.myAzure.loginEndpoint = https://login.microsoftonline.com/
grouper.azureConnector.myAzure.resource = https://graph.microsoft.com
grouper.azureConnector.myAzure.resourceEndpoint = https://graph.microsoft.com/beta/
grouper.azureConnector.myAzure.tenantId = 455754be-3a2b-40c9-acef-c425a92d7276
```

Provisioning fields and attributes

Item	Type	Description
id	field	uuid from Azure
displayName	field	group name in Azure

Azure group types

Documentation

Grouper development team testing

Set this in grouper.hibernate.properties (or set env var: GROUPER_MOCK_SERVICES=true)

```
grouper.is.mockServices = true
```

test config

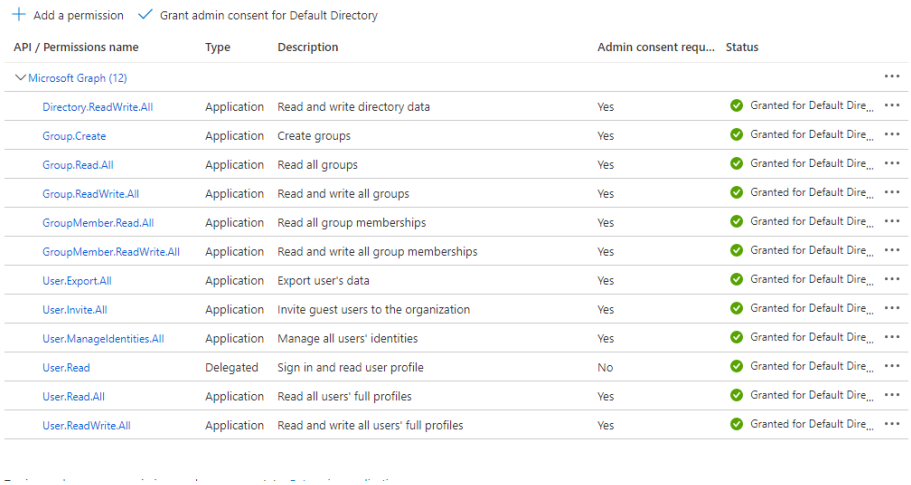
```
grouper.azureConnector.azureTest.clientId = fd805xxxxdfb
grouper.azureConnector.azureTest.clientSecret = *******
grouper.azureConnector.azureTest.graphEndpoint = https://graph.microsoft.com
grouper.azureConnector.azureTest.graphVersion = v1.0
grouper.azureConnector.azureTest.loginEndpoint = http://localhost:8400/grouper/mockServices/azure/auth/
grouper.azureConnector.azureTest.resource = https://graph.microsoft.com
grouper.azureConnector.azureTest.resourceEndpoint = http://localhost:8400/grouper/mockServices/azure/
grouper.azureConnector.azureTest.tenantId = 6c4dxxx0d
```

Set up Azure

1. Sign up with Azure
2. On the left menu, go to Azure Active Directory
3. Create a new app registration
    a. Select: Who can use this app: Accounts in any organizational directory (Any Azure AD directory - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)
4. After the app is registered, click on API Permissions and give Microsoft graph access
    a. Give full permissions for Directory, Group, User, and GroupMember
    b. Grant admin consent for default directory
    c. Check https://jwt.ms with the token, should see


```
"aio": "E2ZgYMg3P9LM3fjtorfnJ0amv+keAA==",
"app_displayname": "vivek-chris-test-july-2022",
"appid": "1ed9f56e-96ef-441d-bf1a-8391747e8c7f",
"appidacr": "1",
"idp": "https://sts.windows.net/a443dbb3-818b-432e-95ec-79acff176ff1/",
"idtyp": "app",
"oid": "fc9f70fa-4518-4c5b-ad13-7a0bfd6c7df1",
"rh": "0.AVkAs9tDpIuBLkOV7Hms_xdv8QMAAAAAAAAAwAAAAAAAAACdAAA.",
"roles": [
  "User.ReadWrite.All",
  "Group.Read.All",
  "Group.Create",
  "Group.ReadWrite.All",
  "User.Invite.All",
  "User.Read.All",
  "GroupMember.Read.All",
  "User.Export.All",
  "User.ManageIdentities.All",
  "GroupMember.ReadWrite.All"
],
"sub": "fc9f70fa-4518-4c5b-ad13-7a0bfd6c7df1",
"tenant_region_scope": "NA",
"tid": "a443dbb3-818b-432e-95ec-79acff176ff1",
"uti": "9sqiXw7uVUSbagwOcWAQAA",
"ver": "1.0",
"wids": [
  "0997a1d0-0d1d-4acb-b408-d5ca73121e90"
],
```

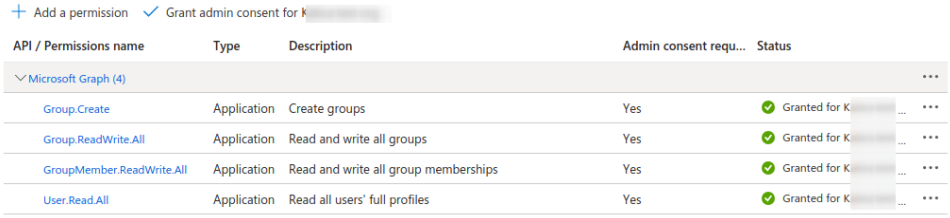

    d. Permissions look like this. Note you can clamp down these permissions as needed



+ Add a permission ✓ Grant admin consent for Default Directory

API / Permissions name	Type	Description	Admin consent requ...	Status
Microsoft Graph (12)				
Directory.ReadWrite.All	Application	Read and write directory data	Yes	Granted for Default Dire...
Group.Create	Application	Create groups	Yes	Granted for Default Dire...
Group.Read.All	Application	Read all groups	Yes	Granted for Default Dire...
Group.ReadWrite.All	Application	Read and write all groups	Yes	Granted for Default Dire...
GroupMember.Read.All	Application	Read all group memberships	Yes	Granted for Default Dire...
GroupMember.ReadWrite.All	Application	Read and write all group memberships	Yes	Granted for Default Dire...
User.Export.All	Application	Export user's data	Yes	Granted for Default Dire...
User.Invite.All	Application	Invite guest users to the organization	Yes	Granted for Default Dire...
User.ManageIdentities.All	Application	Manage all users' identities	Yes	Granted for Default Dire...
User.Read	Delegated	Sign in and read user profile	No	Granted for Default Dire...
User.Read.All	Application	Read all users' full profiles	Yes	Granted for Default Dire...
User.ReadWrite.All	Application	Read and write all users' full profiles	Yes	Granted for Default Dire...

   e. From basic testing, if using read-only entities, the following Admin consent permissions seems to work (should not need any User consent grants):



+ Add a permission ✓ Grant admin consent for K

API / Permissions name	Type	Description	Admin consent requ...	Status
Microsoft Graph (4)				
Group.Create	Application	Create groups	Yes	Granted for K
Group.ReadWrite.All	Application	Read and write all groups	Yes	Granted for K
GroupMember.ReadWrite.All	Application	Read and write all group memberships	Yes	Granted for K
User.Read.All	Application	Read all users' full profiles	Yes	Granted for K

   f. For even tight permissions, see below for setting the Grouper service account as the owner for new groups

5. On the left, under Certificates and Secrets, create a new secret
6. When testing using Postman, you will only need the secret value to get access token which will be used to call the graph API
7. To get an access token, make a POST call to https://login.microsoftonline.com/a98c57b9-a771-4c01-b69b-83cceb36c834/oauth2/v2.0/token (id is the directory tenant id)
8. Under form data send these four key values. client_id = clientId, scope = https://graph.microsoft.com/.default, client_secret = clientSecret, grant_type=client_credentials
   a. Content-type: application/x-www-form-urlencoded
   b. Post body looks like this:

```
client_id=aea2eb2a-bc4f-4ae5-a315-3XXXXX&scope=https%3A%2F%2Fgraph.microsoft.com%2F.
default&grant_type=client_credentials&client_secret=ewC8Q~yGN4dyBaSYBrOXXXXXXXXXX
```

   c. Configure external system in grouper-loader.properties

```
grouper.azureConnector.azure.clientId = aea2eb2a-bc4f-4ae5-a315-38XXXXX
grouper.azureConnector.azure.clientSecret = ewC8Q~yXXXXX
grouper.azureConnector.azure.graphEndpoint = https://graph.microsoft.com
grouper.azureConnector.azure.graphVersion = beta
grouper.azureConnector.azure.loginEndpoint = https://login.microsoftonline.com/
grouper.azureConnector.azure.resource = https://graph.microsoft.com
grouper.azureConnector.azure.resourceEndpoint = https://graph.microsoft.com/beta/
grouper.azureConnector.azure.tenantId = 5e7fa4df-8d24XXXXXXX
```

9. The client id is the Application (client) ID next to Directory tenant id on the Overview page of the app.
10. The response from the above POST call will give you an access token in the body which we will use to access graph APIs like https://graph.microsoft.com/v1.0/groups
11. For the above request, send Authorization header with value Bearer <access token>

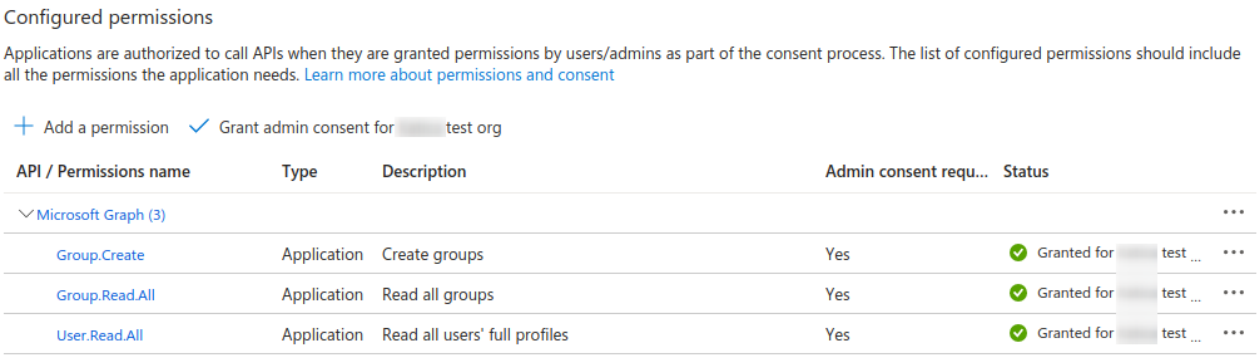
Add Microsoft certificate for graph apis

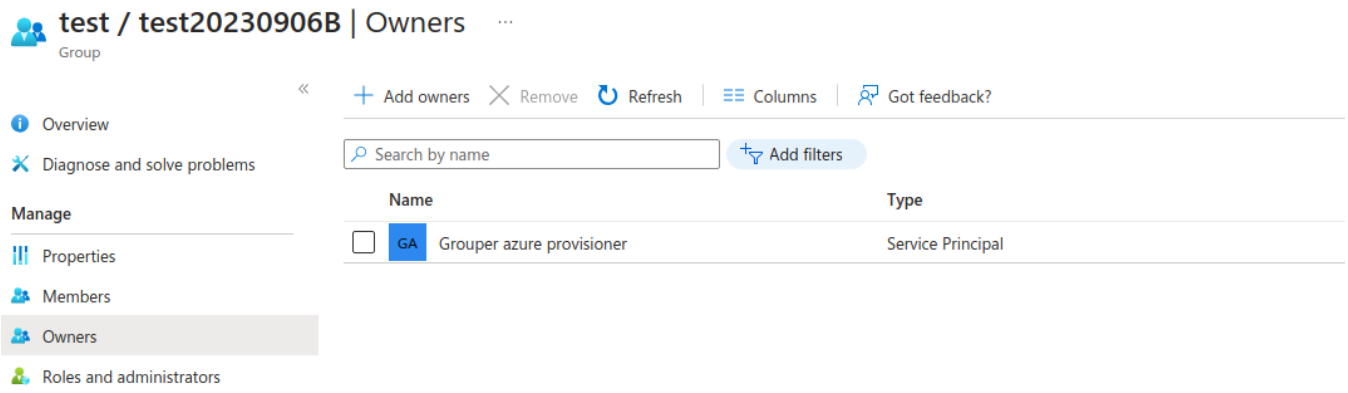
1. Go to https://graph.microsoft.com/applications in your browser and download the certificate by clicking on the padlock sign in the address bar.

2. Find out the path to the security directory inside the jre. e.g. /Library/Java/JavaVirtualMachines/jdk1.8.0_65.jdk/Contents/Home/jre/lib/security
3. From the terminal run "sudo keytool -import -alias microsoft.graph -keystore cacerts -file ~/graph.microsoft.com.cer
4. For the password enter: changeit

Setting the Grouper service account as the owner of new groups (to reduce Azure privileges) v4.2.0+

If the service account Grouper uses for Graph API calls can be set as the owner of a managed group, the Azure application no longer needs the privileges to update all groups and memberships. It only needs the Group.Create privilege to create a new group, Group.Read.All to find groups to match with Grouper, and User.Read.All to resolve target entities. While there is an option in the Azure provisioner to set the groupOwner attribute with one or more entities, the ability to add non-users (i.e. service accounts) is only available v4.2.0.





The "groupOwners" attribute is normally expected to be a Grouper subject ID or identifier used for search/match to resolve to an Azure object URL. But this lookup always uses the /users api endpoint, which means the service account can't be added this way. Since v4.2.0, an already-resolved URL can be entered in this field, and this URL can be either for a user or a service account. There are two ways to specify the account.

1) https://graph.microsoft.com/v1.0/servicePrincipals/{id}

2) https://graph.microsoft.com/v1.0/servicePrincipals(appId='{appId}')

where {id} is the "Object ID" for the object in the Enterprise Applications page (it's not the App registrations page, but you can get to it from there by clicking "Managed application in local directory"). The {appId} is the application ID, also called the client ID, and is easier to find, being on the Enterprise applications page, the App registrations page, and various other pages when you look at the service account details. The appId is also the "Client ID" value in the external system configuration for Azure.

Properties

GA

Name

Grouper azure provisioner

Application ID

db5ee87b-a33b-45d9-b023-...

Object ID

0106c3fd-ae2e-41f2-9e69-68...

Target Group attr groupOwners

Configuration for the group attribute

Group attr groupOwners - name	☐ EL?	groupOwners * Attribute name is the key in the key/value pairs for this group. Generally you will have metadata for group type so you should not have attributes for: groupTypeMailEnabled, groupTypeMailEnabledSecurity, groupTypeSecurity, groupTypeUnified, isAssignableToRole.
Group attr groupOwners - translation type	☐ EL?	staticValues If this is a simple translation straight from field, then select GrouperProvisioningGroupField since you do not have to worry about a script typo and the performance is better. Or if the attribute should have static values, then select staticValues. Otherwise configure a translation script.
Group attr groupOwners - translate from static values	☐ EL?	https://graph.microsoft.com/v1.0/servicePrincipals(appId='db5ee87b * Specify static value. Value should be comma-separated if this is a multi-valued attribute.