AAD Auth User Guide

Create Resource

1. Make sure you have Az module installed: https://www.powershellgallery.com/packages/Az/ (>version 1.4.0)

2. Login

   `Connect-AzAccount`

3. Make sure you are in the right subscriptio.

   `Set-AzContext -Subscription "3a96ef56-xxxx-xxxx-xxxx-fb125c2b8798"`

4. Create a resource with a custom subdomain

   `$resource = (New-AzCognitiveServicesAccount -ResourceGroupName meng-test -Name test-cm-aad -Type "ContentModerator" -SkuName "S0" -Location "Central US EUAP" -CustomSubdomainName "cmaad")`

5. Record your Endpoint

   `$resource`

```
ResourceGroupName               : meng-test
AccountName                     : test-cm-aad
Id                              : /subscriptions/c830bb7a-83f5-45e3-81fc-3c2053e7d16f/resourceGroups/meng-test/providers/Microsoft.CognitiveServices/accounts/test-cm-aad
Endpoint                        : https://cmaad.cognitiveservices.azure.com/
Location                        : Central US EUAP
Sku                             : Microsoft.Azure.Management.CognitiveServices.Models.Sku
AccountType                     : ContentModerator
ResourceType                    : Microsoft.CognitiveServices/accounts
Etag                            : "03001b95-0000-3300-0000-63636f830000"
ProvisioningState               : Succeeded
CustomSubDomainName             : cmaad
PublicNetworkAccess             : Enabled
Identity                        :
Encryption                      :
UserOwnedStorage                :
PrivateEndpointConnections      : {}
ApiProperties                   :
Properties                      : Microsoft.Azure.Management.CognitiveServices.Models.AccountProperties
RestrictOutboundNetworkAccess   :
AllowedFqdnList                 :
DisableLocalAuth                :
NetworkRuleSet                  :
Capabilities                    : {DynamicThrottling, VirtualNetworks, Container}
Tags                            :
```

Create AAD app

0. Create AAD Application

   `New-AzADApplication -DisplayName "cmaadtestapp"`

```
PS C:\WINDOWS\system32> New-AzADApplication -DisplayName "cmaadtestapp"

DisplayName    Id                                    AppId
-----------    --                                    -----
cmaadtestapp   6a41b30d-812b-4d42-8429-389dde217cb2  7335ecd3-a1a2-4fb9-8137-ef26fe560c74
```

1. Create Service Principal

   `New-AzADServicePrincipal -ApplicationId "<AAD AppId got above>"`

```
PS C:\WINDOWS\system32> New-AzADServicePrincipal -ApplicationId 7335ecd3-a1a2-4fb9-8137-ef26fe560c74

DisplayName    Id                                    AppId
-----------    --                                    -----
cmaadtestapp   e0ca9098-d9f7-4ed4-aa37-bae8c9013239  7335ecd3-a1a2-4fb9-8137-ef26fe560c74
```

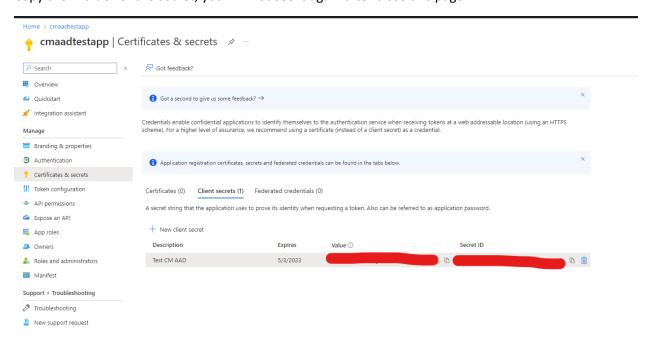0. Get the object Id and assign Cognitive Services User role on the resource created above

$objectId=(Get-AzADServicePrincipal -DisplayName cmaadtestapp).Id
New-AzRoleAssignment -ObjectId $objectId -scope $resource.Id -
RoleDefinitionName "Cognitive Services User"

```
PS C:\WINDOWS\system32> $objectId=(Get-AzADServicePrincipal -DisplayName cmaadtestapp).Id
PS C:\WINDOWS\system32> New-AzRoleAssignment -ObjectId $objectId -scope $resource.Id -RoleDefinitionName "Cognitive Services User"
WARNING: We have migrated the API calls for this cmdlet from Azure Active Directory Graph to Microsoft Graph.
Visit https://go.microsoft.com/fwlink/?linkid=2181475 for any permission issues.

RoleAssignmentName : 02f892cf-be43-49e7-80e2-a85b4aad4df4
RoleAssignmentId   : /subscriptions/c830bb7a-83f5-45e3-81fc-3c2053e7d16f/resourceGroups/meng-test/providers/Microsoft.CognitiveServices/accounts/test-cm-aad/providers/Microsoft.Authorization/roleAssignments/02f892cf-be43-49e7-80e2-a85b4
                     aad4df4
Scope              : /subscriptions/c830bb7a-83f5-45e3-81fc-3c2053e7d16f/resourceGroups/meng-test/providers/Microsoft.CognitiveServices/accounts/test-cm-aad
DisplayName        : cmaadtestapp
SignInName         :
RoleDefinitionName : Cognitive Services User
RoleDefinitionId   : a97b65f3-24c7-4388-baec-2e87135dc908
ObjectId           : e0ca909B-d9f7-4ed4-aa37-bae8c9013239
ObjectType         : ServicePrincipal
CanDelegate        : False
Description        :
ConditionVersion   :
Condition          :
```

Validate AAD Access

1. Go to your AAD app in Azure Portal and create a secret
   Copy the "Value" of the secret, you will not see it again after close this page

Home > cmaadtestapp

🔑 **cmaadtestapp | Certificates & secrets** 📌 ⋯

🔍 Search «	🗨 Got feedback?
⊞ Overview	
🏁 Quickstart	ℹ️ Got a second to give us some feedback? → ✕
🚀 Integration assistant	
Manage	Credentials enable confidential applications to identify themselves to the authentication service when receiving tokens at a web addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.
▤ Branding & properties	
⊙ Authentication	ℹ️ Application registration certificates, secrets and federated credentials can be found in the tabs below. ✕
🔑 Certificates & secrets	
⫼ Token configuration	Certificates (0) **Client secrets (1)** Federated credentials (0)
→ API permissions	A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.
☁ Expose an API	+ New client secret
🏢 App roles	
👥 Owners	Description Expires Value ⓘ Secret ID
👤 Roles and administrators	Test CM AAD 5/3/2023
▦ Manifest	
Support + Troubleshooting	
🔧 Troubleshooting	
👤 New support request	

2. Get Access Token

$SecuredPassword = ConvertTo-SecureString '<Secret Value copied above>' -
AsPlainText -Force
$ApplicationId = "<AAD App Id got above>"
$TenantId = "<Your Tenant Id>"
$Credential = New-Object -TypeName
System.Management.Automation.PSCredential ($ApplicationId,$SecuredPassword)
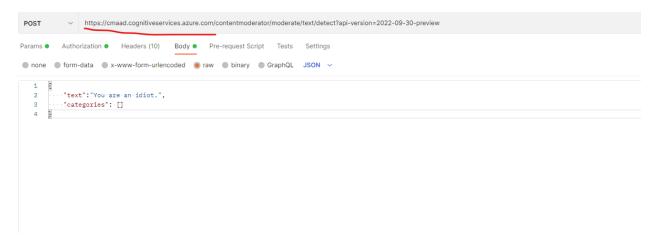Connect-AzAccount -ServicePrincipal -TenantId $TenantId -Credential $Credential
$token= Get-AzAccessToken -ResourceUrl https://cognitiveservices.azure.com/
$token.Token

PS C:\WINDOWS\system32> $token= Get-AzAccessToken -ResourceUrl "https://cognitiveservices.azure.com/"
PS C:\WINDOWS\system32> $token.Token
eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsIng1dCI6IjlaUXBKM1VvVmpBWVhZR2FYRUpsOGxWMFRPSSIsImtpZCI6IjlaUXBKM1VvVmpBWVhZR2FYRUpsOGxWMFRPSSJ9.eyJhdWQiOiJodHRwczovL2NvZ25pdGl2ZXNlcnZpY2VzLmF6dXJlLmNvbSBiLCJpc3MiOiJodHRwczovL3N0cy53aW5kb3dzLm5ldC83...
[token data]

3. Make HTTP Request by Postman

Use your endpoint:

> POST ⌄ | https://cmaad.cognitiveservices.azure.com/contentmoderator/moderate/text/detect?api-version=2022-09-30-preview

Params ● | Authorization ● | Headers (10) | Body ● | Pre-request Script | Tests | Settings

⚪ none ⚪ form-data ⚪ x-www-form-urlencoded 🔴 raw ⚪ binary ⚪ GraphQL JSON ⌄

```
1  {
2      "text":"You are an idiot.",
3      "categories": []
4  }
```

Choose "Bearer Token" as authorization method and fill in the token you got from last step:

> POST ⌄ | https://cmaad.cognitiveservices.azure.com/contentmoderator/moderate/text/detect?api-version=2022-09-30-preview | **Send** ⌄

Params ● | Authorization ● | Headers (10) | Body ● | Pre-request Script | Tests | Settings | Cookies

Type | Bearer Token ⌄

The authorization header will be automatically generated when you send the request. Learn more about authorization ↗

⚠ Heads up! These parameters hold sensitive data. To keep this data secure while working in a collaborative environment, we recommend using variables.
Learn more about variables ↗ | ✕

Token | eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsIng...