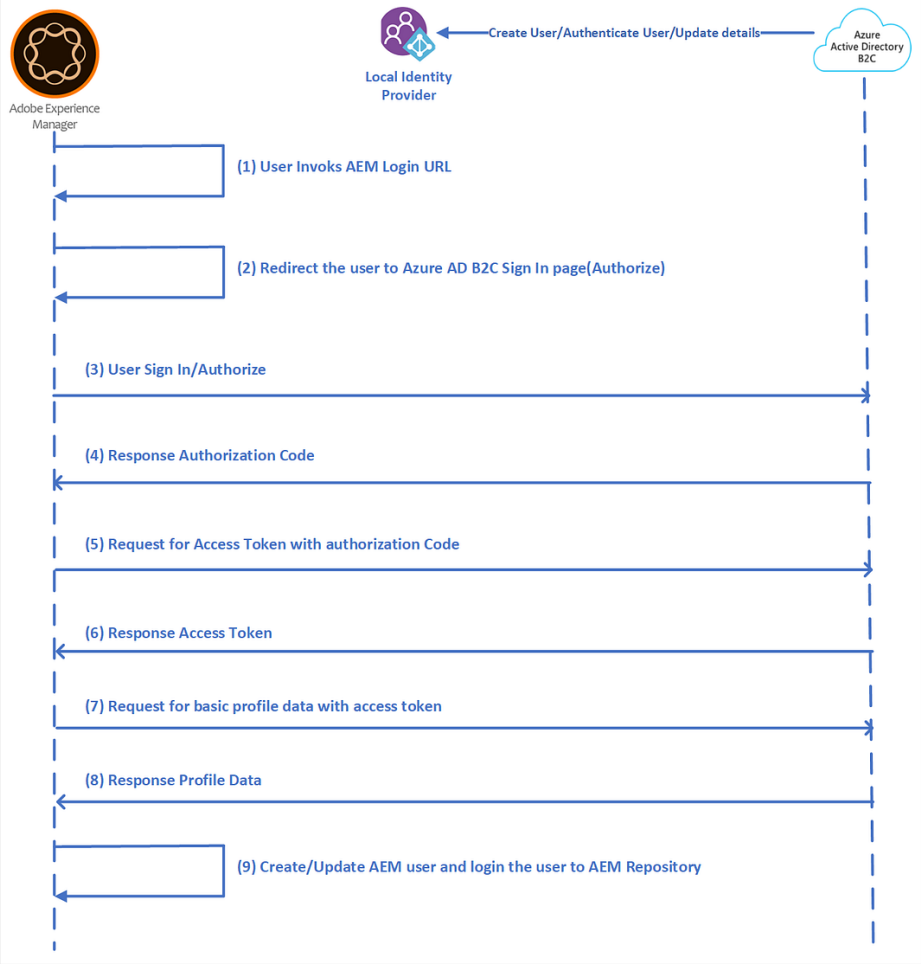
FRIDAY, DECEMBER 10, 2021

Enable User Authentication for AEM Websites — Azure AD B2C OAuth 2.0

Please note, sharing this post based on my learning and understanding may not be the right solution for your use cases; considering security and other factors, consult AEM/Azure AD B2C experts before enabling any user authentications solutions.

In my earlier post, we have seen how to use the Azure AD B2C SAML standard to enable authenticated websites in AEM. In this post, let us now see how to use the OAuth 2.0 standard to configure Authenticated websites in AEM using Azure AD B2C. The recommendation from Microsoft is to use OpenID Connect to enable the authentication for websites, but AEM currently won't support OpenID connect OOTB — may need to build a custom authentication handler to support OpenID Connect with AEM.



Azure AD B2C Configurations:

AZURE ACTIVE DIRECTORY B2C TENANT:

Before your applications can interact with Azure Active Directory B2C (Azure AD B2C), they must be registered in a tenant that you manage.

Refer to https://docs.microsoft.com/en-us/azure/active-directory-b2c/tutorial-create-tenant for more details on configuring B2C tenants.

Signing and encryption keys for Identity Experience Framework:

As a next step, Create the signing and encryption keys

The Identity Experience Framework should be defined to support the user authentication through AD B2C Local accounts.

Refer to https://docs.microsoft.com/en-us/azure/active-directory-b2c/tutorial-create-user-flows?pivots=b2c-custom-policy#add-signing-and-encryption-keys-for-identity-experience-framework-applications for more details on enabling Signing and encryption keys for Identity Experience Framework.

Register Identity Experience Framework applications:

- 2015 (83)
- 2014 (90)
- 2013 (38)
- 2012 (123)
- 2011 (60)

Azure AD B2C requires you to register two applications that it uses to sign up and sign in users with local accounts: *IdentityExperienceFramework*, a web API, and *ProxyIdentityExperienceFramework*, a native app with delegated permission to the IdentityExperienceFramework app. Your users can sign up with an email address or username and a password to access your tenant-registered applications, which creates a "local account." Local accounts exist only in your Azure AD B2C tenant.

Refer to https://docs.microsoft.com/en-us/azure/active-directory-b2c/tutorial-create-user-flows?pivots=b2c-custom-policy#register-identity-experience-framework-applications for more details on registering identity experience framework applications.

Custom Policies:

As a next step to enable the required custom policies, the starter pack can be downloaded from https://github.com/Azure-Samples/active-directory-b2c-custom-policy-starterpack, I am going to enable OAuth 2.0 with a Local account.

Customized LocalAccount policies to enable SignIn/SignUp/Profile Edit can be downloaded from https://github.com/techforum-repo/youttubedata/tree/master/active-directory-b2c/oauth/b2c-policies

Update the tenant name references in TrustFrameworkBase.xml, TrustFrameworkExtensions.xml, TrustFrameworkLocalization.xml, SignUpOrSigninSAML.xml, and ProfileEdit.xml

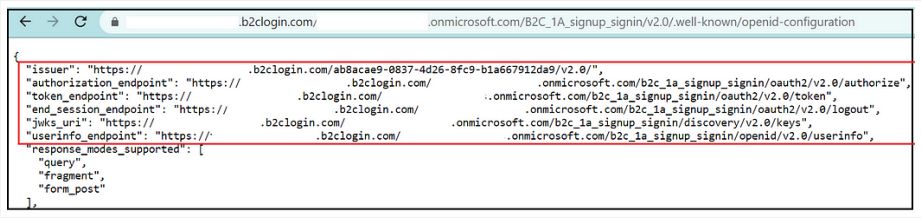
Replace ProxyIdentityExperienceFrameworkAppId and IdentityExperienceFrameworkAppId values in TrustFrameworkExtensions.xml.

Currently the Graph API is not supported to fetch the user profile details from Azure AD B2C Local Accounts, the user info endpoints can be enabled through custom policy to fetch the required profile details of the logged in user

Refer to https://docs.microsoft.com/en-us/azure/active-directory-b2c/userinfo-endpoint?pivots=b2c-custom-policy for more details on enabling user info endpoint.

The issuer id that needs to be configured in TrustFrameworkExtensions.xml along with other end point details can be copied from openid-configuration URL — https://tenantn.b2clogin.com/tenantname.onmicrosoft.com/<policy-name>/v2.0/.well-known/openid-configuration

e.g https://albinsblog.b2clogin.com/albinsblog.onmicrosoft.com/B2C_1A_signup_signin/v2.0/.well-known/openid-configuration



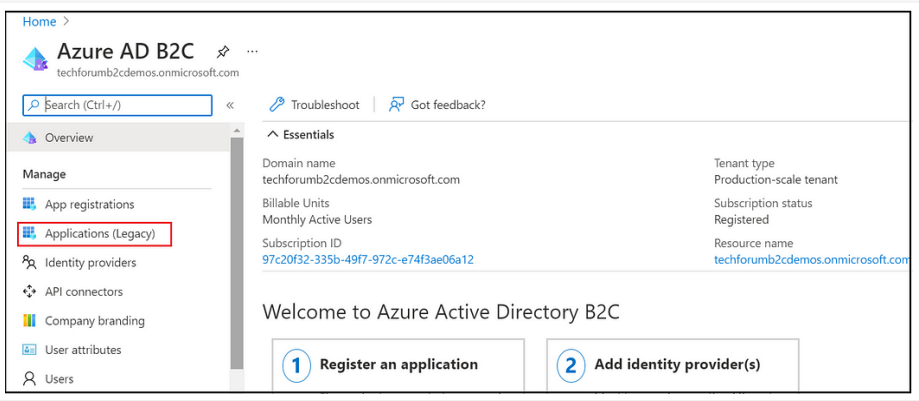
```
<Item Key="issuer">https://yourtenant.b2clogin.com/11111111-1111-1111-1111-
111111111111/v2.0/</Item>
```

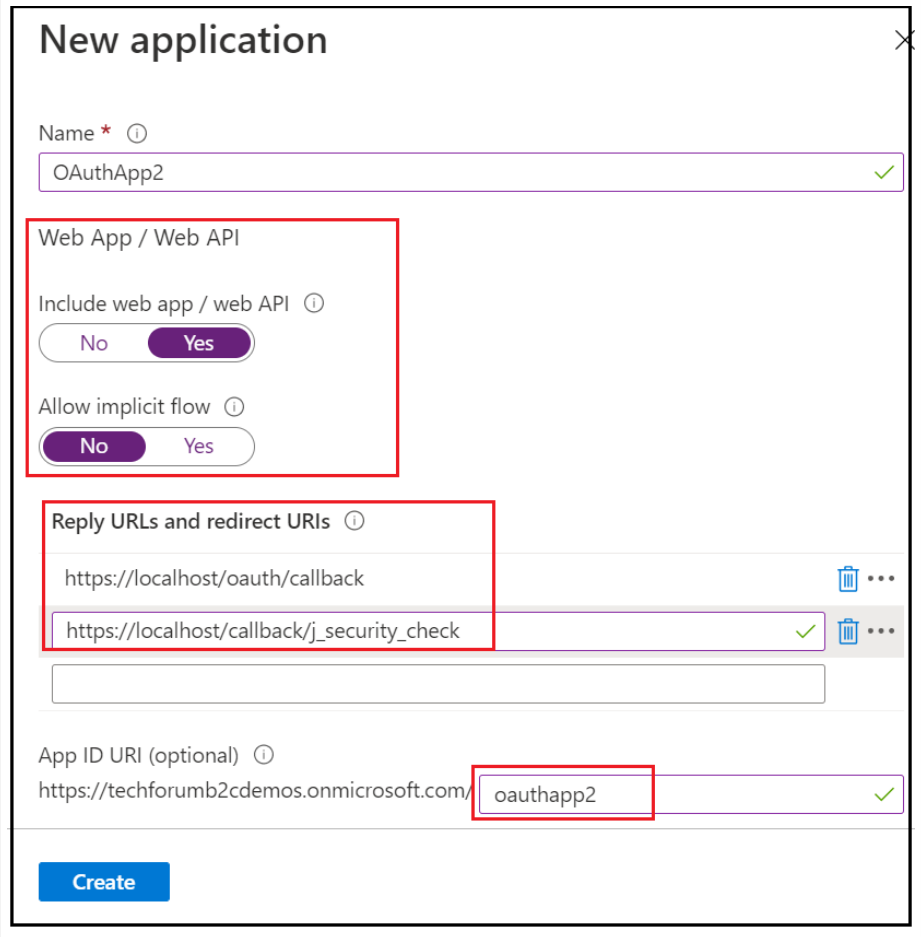
Now upload the policies to Azure AD B2C “Identity Experience Framework” in the below order

```
TrustFrameworkBase.xml
TrustFrameworkLocalization.xml
TrustFrameworkExtensions.xml
SignUpOrSigninSAML.xml
ProfileEdit.xml
```

AZURE AD B2C APP SETUP:

Not setup Azure AD B2C app through Azure portal, I was getting version mismatch issue while enabling the APP through latest App registration link, registered the app through legacy app registration link to overcome the issue.



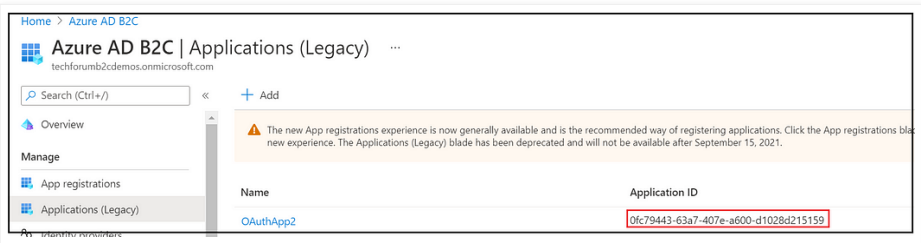


Ensure the below reply URL's are enabled — I am using localhost for demo but use the actual website domain

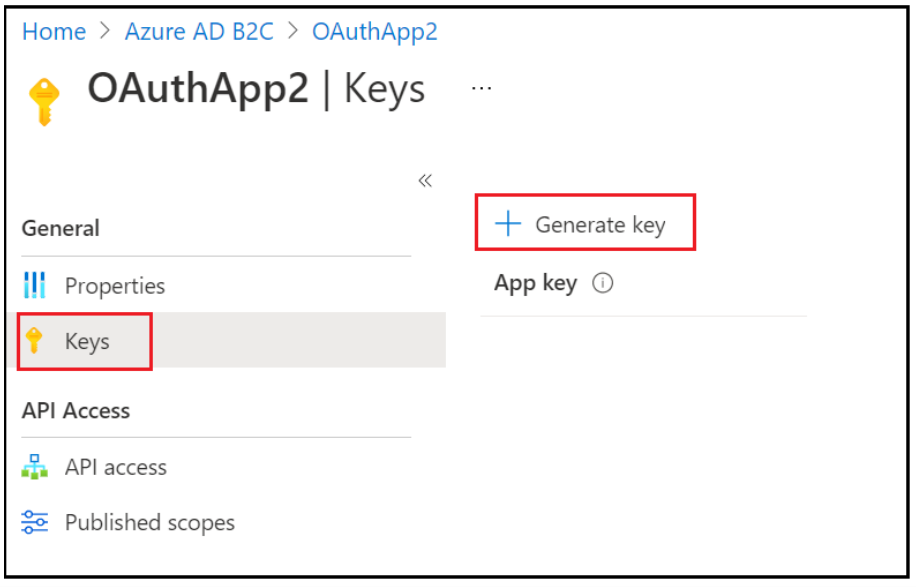
https://localhost/**oauth/callback**

https://localhost/**callback/j_security_check**

Copy the application id, this value is required while configuring the OAuth handler in AEM



Now configure the secret key to the app



Copy the secret , this value is required while configuring the OAuth handler in AEM

AEM Configurations:

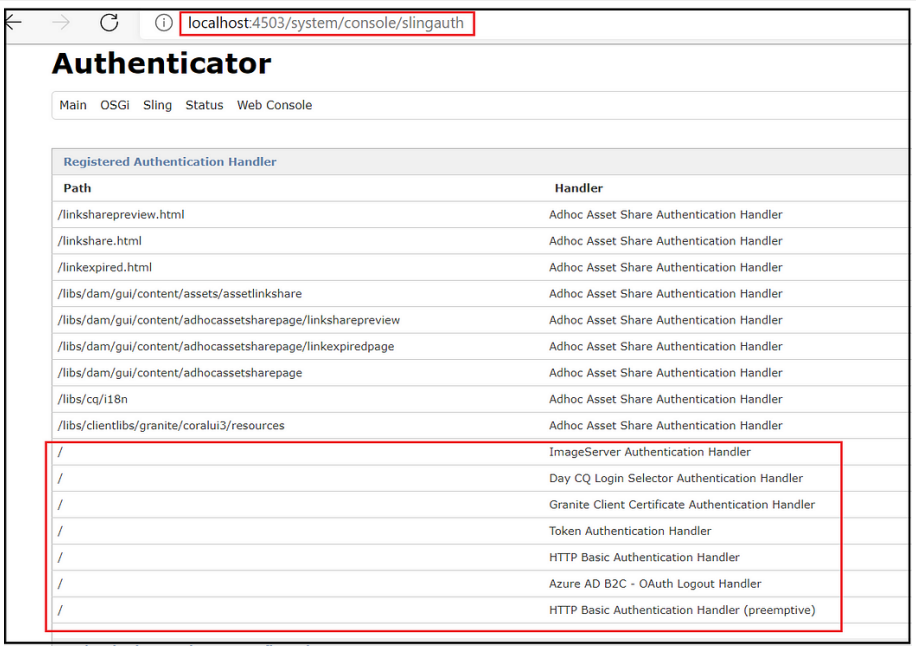
ENABLE SSL FOR AEM SERVERS:

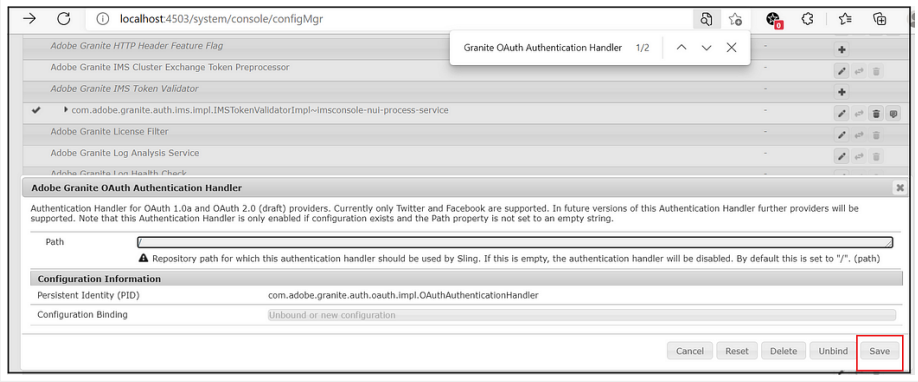
The return URL's should be enabled through HTTPS, even for local testing the HTTPS is mandatory.

Refer to https://experienceleague.adobe.com/docs/experience-manager-learn/foundation/security/use-the-ssl-wizard.html?lang=en for enabling the SSL for AEM server(the same self-signed certificate can be downloaded for testing, I have modified the default AEM SSL port to 443 ”

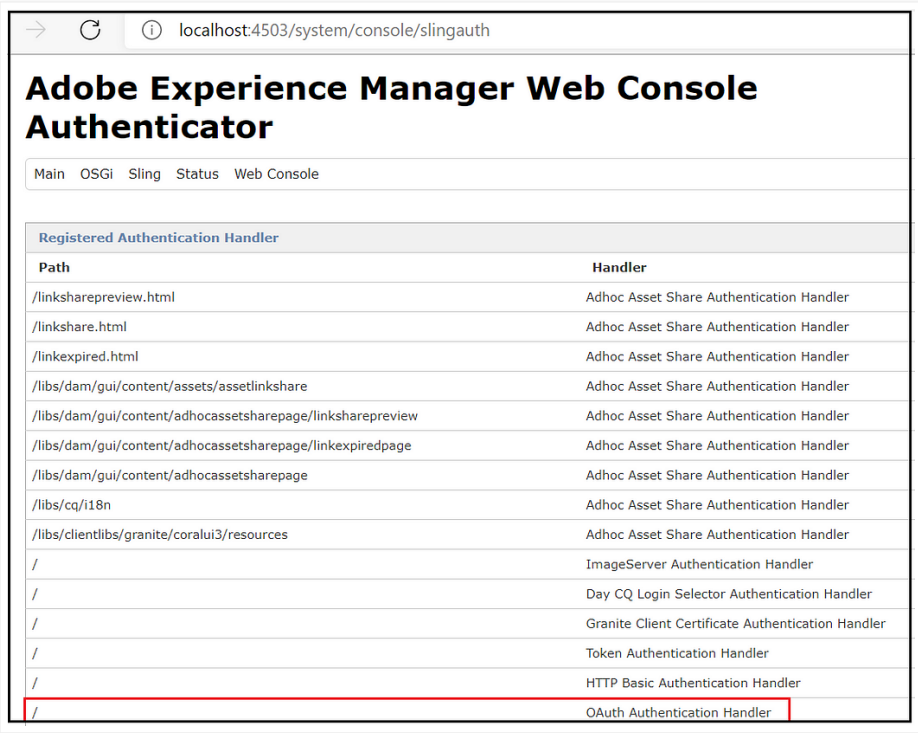
ENABLE OAUTH AUTHENTICATION HANDLER:

By default, “Adobe Granite OAuth Authentication Handler” is not enabled by default, the handler can be enabled by opening and saving without doing any changes.





Now the OAuth Authentication handler is enabled



CUSTOM AZURE AD B2C OAUTH PROVIDER:

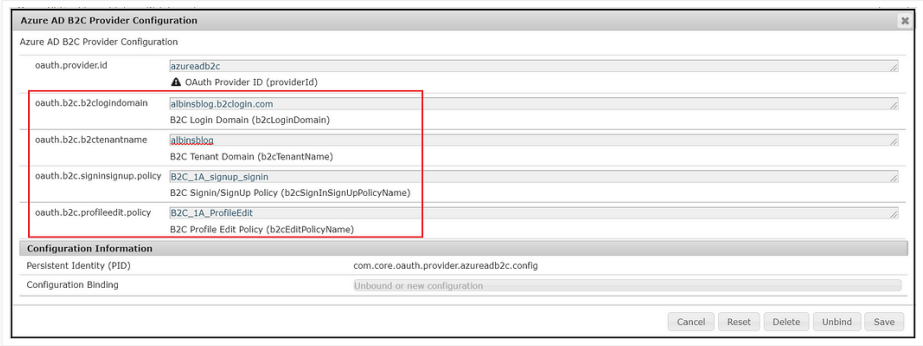
AEM won't support Azure AD B2C OAuth authentication OOTB, define a new provider to support the authentication with Azure AD B2C. Refer to https://github.com/techforum-repo/youttubedata/tree/master/active-directory-b2c/oauth/azureadb2c-oauth-provider for custom provider to support Azure AD B2C Oauth authentication. The bundle enabled for AEM as a Cloud Service SDK, may require some changes to deploy the provider bundle to other AEM versions.

The latest SDK versions supports com.adobe.granite.auth.oauth.Provider2 to enable logout functionality(refer to https://github.com/techforum-repo/youttubedata/blob/master/active-directory-b2c/oauth/azureadb2c-oauth-provider/src/main/java/com/core/oauth/provider/azureadb2c/AzureADB2COAuth2ProviderImpl.java) but unfortunately the logout functionality wont provide any option to enable site specific post logout redirects on multisite scenarios, to handle this I have enabled a Custom Authentication Handler(refer to https://github.com/techforum-repo/youttubedata/blob/master/active-directory-b2c/oauth/azureadb2c-oauth-provider/src/main/java/com/core/oauth/provider/azureadb2c/AzureADB2COAuthLogoutHandler.java) that will redirect the user to root page of the website(the behavior can be

customized as required) — the logout can be initiated by adding “operation=b2clogout” on current URL.

CONFIGURE OAUTH APPLICATION AND PROVIDER:

Now configure the “Azure AD B2C Provider Configuration” with Azure AD B2C specific configurations




```
// Configuration created by Apache Sling JCR Installer
{
  "b2cTenantName":"albinsblog",
  "b2cSignInSignUpPolicyName":"B2C_1A_signup_signin",
  "b2cLoginDomain":"albinsblog.b2clogin.com",
  "b2cEditPolicyName":"B2C_1A_ProfileEdit"
}
```

Configure “Adobe Experience Manager Web Console — Configuration”

Enable the Azure AD B2C application id copied in the earlier step into **oauth.client.id** and **oauth.scope**.

Enable Azure AD B2C application secret copied in the earlier step into **auth.client.secret**

Enable a unique site specific config id(**oauth.config.id**) — this value will be used while invoking the authentication for a specific URL.

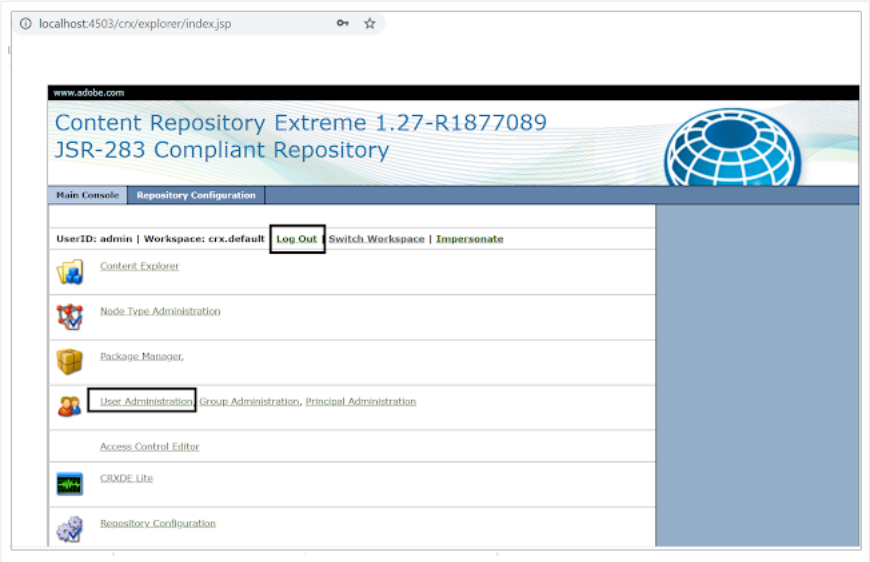
Ensure the oauth.config.provider.id is configured as **azureadb2c**

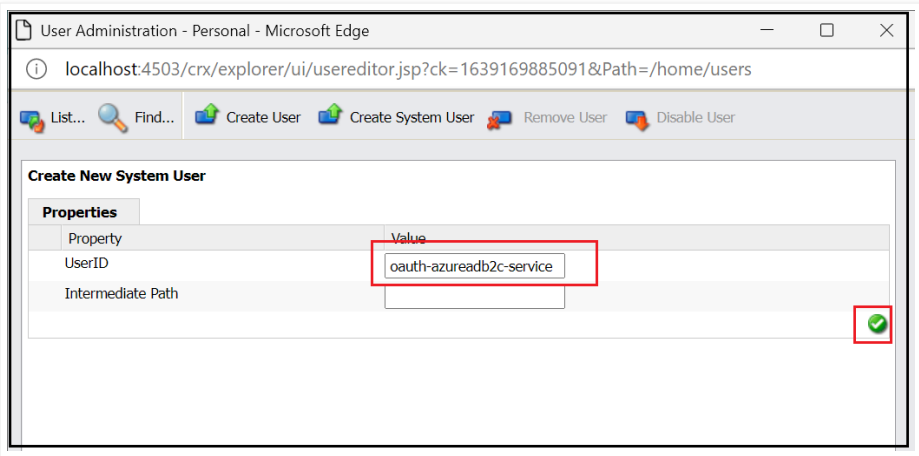
```
{
  "oauth.client.id":"cc3b55ac-3904-4140-xxxx",
  "oauth.config.id":"azureadb2csite1",
  "oauth.config.provider.id":"azureadb2c",
  "oauth.create.users":true,
  "oauth.csrf.state.protection":false,
  "oauth.scope":[
    "cc3b55ac-3904-4140-xxxx"
  ],
  "oauth.client.secret":"j36E.l)xxxx"
}
```

CONFIGURE SERVICE USER:

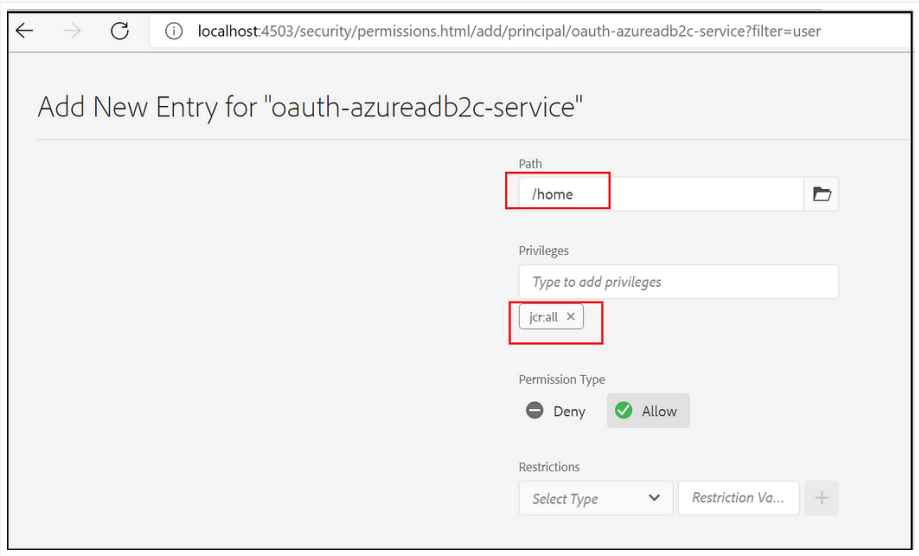
Enable the service user with required permissions to manage the users in the system, Create a system user with name “oauth-azureadb2c-service”, navigate

to http://localhost:4503/crx/explorer/index.jsp and login as an admin user and click on user administration



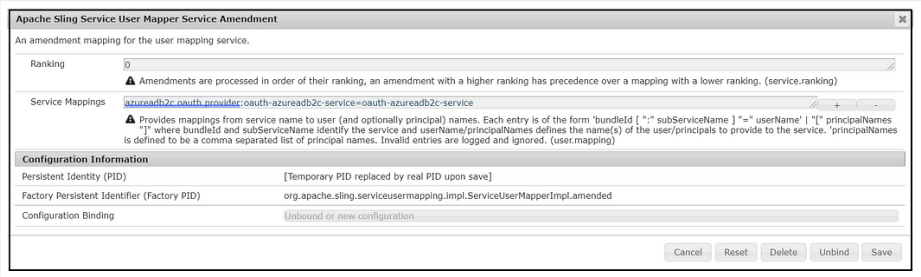


Now enable the required permissions for the user — AEM Security | Permissions

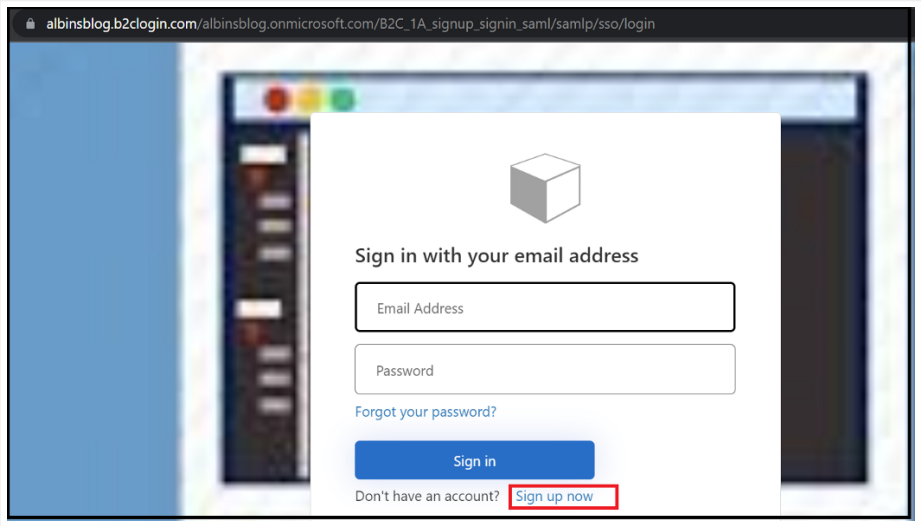


Now enable the service user mapping for provider bundle — add an entry into Apache Sling Service User Mapper Service Amendment

```
{
  "user.mapping":[
    "azureadb2c.oauth.provider:oauth-azureadb2c-service=oauth-azureadb2c-service"
  ]
}
```



Now the users will be directed to the Azure AD B2C login page while accessing https://localhost/j_security_check?configid=azureadb2csite1 (configid specified while enabling the OAuth Authentication Handler)



The user can signup for a new account or login through existing account. The user will be redirect to the root page on successful authentication — https://localhost/, to redirect the user to the current page or to a specific page on successful authentication use the following URL pattern to access the authentication — https://localhost/**content/wknd/us/en.html**/j_security_check?configid=azureadb2csite1, now the user will be redirected to **/content/wknd/us/en.html** on successful authentication.

Now by accessing the following URL — https://localhost/system/sling/logout.html?operation=b2clogout the user will be logged out from the website.

USER SYNC:

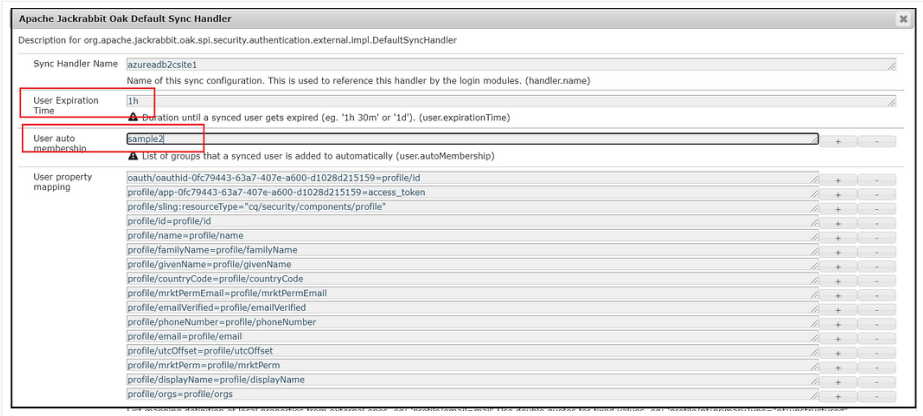
Whenever the profile data is changed (e.g familyName and givenName) in Azure AD B2C through the profile edit functionality the same will be reflected in AEM on subsequent logins based on the “Apache Jackrabbit Oak Default Sync Handler” configuration.

AEM OOTB creates an “Apache Jackrabbit Oak Default Sync Handler” configuration specific to each OAuth provider implementation.

The sync handler syncs the user profile data between the external authentication system and the AEM repository.

Configure “User auto membership” property with required AEM groups, the users should be added into while creating the users in AEM — ensure the group is created with required permissions before configuring the sync handler.

The user profile data is synced based on the User Expiration Time setting, the user data will get synced(including the group changes) on the subsequent login after the synced user data expired(default is 1 hr.). Modify the configurations based on the requirement.




```
// Configuration created by Apache Sling JCR Installer
{
  "user.expirationTime":"30m",
  "group.expirationTime":"1h",
  "user.membershipNestingDepth:Integer":1,
  "user.propertyMapping":[
    "oauth/oauthid-cc3b55ac-3904-4140-a204-fd0c72ee5cb6=profile/id",
    "profile/app-cc3b55ac-3904-4140-a204-fd0c72ee5cb6=access_token",
    "profile/sling:resourceType=\"cq/security/components/profile\"",
    "profile/id=profile/id",
    "profile/name=profile/name",
    "profile/familyName=profile/familyName",
    "profile/givenName=profile/givenName",
    "profile/countryCode=profile/countryCode",
    "profile/mrktPermEmail=profile/mrktPermEmail",
    "profile/emailVerified=profile/emailVerified",
    "profile/phoneNumber=profile/phoneNumber",
    "profile/email=profile/email",
    "profile/utcOffset=profile/utcOffset",
    "profile/mrktPerm=profile/mrktPerm",
    "profile/displayName=profile/displayName",
    "profile/orgs=profile/orgs"
  ],
  "user.autoMembership":[
    "sample2"
  ],
  "handler.name":"azureadb2csite1",
  "user.pathPrefix":"azureadb2c",
  "user.disableMissing":true
}
```

Profile Edit:

To edit the profile for a logged-in user, the user should be redirected to authorize end point with Profile Edit Policy
e.g https://albinsblog.b2clogin.com/albinsblog.onmicrosoft.com/B2C_1A_ProfileEdit/oauth2/v2.0/authorize

Created a Custom Java Filter to handle the profile edit scenario, the filter sends the profile edit request to Azure AD B2C if the URL contains the "operation=profileedit" parameter.

Refer to the custom java filter here — https://github.com/techforum-repo/youttubedata/tree/master/active-directory-b2c/saml/aem-filter

Now the logged in users(if not users will be directed to the login page first) will be sent to the Profile edit page while accessing the URL with "operation=profileedit" and "configId=azureadb2csite1"(configid value will change based on your configuration)
e.g. https://localhost**/content/wknd/us/en.html**?operation=profileedit&configId=azureadb2csite1(use the current page path so the user will be redirected to the same page on successful profile edit)

The user will be redirected to **/content/wknd/us/en.html** on successful profile edit.

You can now create a small component to display the login status in the web page, the logged in user details can be fetched through a AJAX service call
to https://localhost/libs/granite/security/currentuser.json

Login — https://localhost/content/wknd/us/en.html/j_security_check?configid=azureadb2csite1

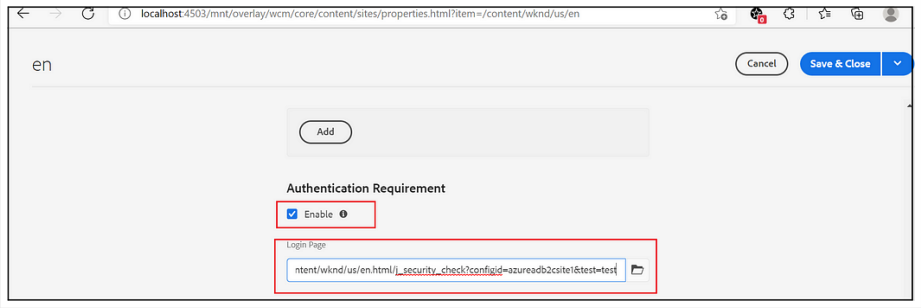
Log out URL —https://localhost/system/sling/logout.html?operation=b2clogout

Profile Edit URL — https://localhost/content/wknd/us/en.html?operation=profileedit&configId=azureadb2csite1

The "Sign In/Sign Up/Profile Edit" UI branding and the view can be customized based on the project need, also custom DNS can be enabled to support the company-specific login URL — e.g login.mysite.com

CUG(Closed User Group):

Another option is using CUG to enable the authentication for specific pages, the authentication will be requested while the user accessing that specific page and child pages. The CUG can be enabled through Page Properties — Advanced tab.

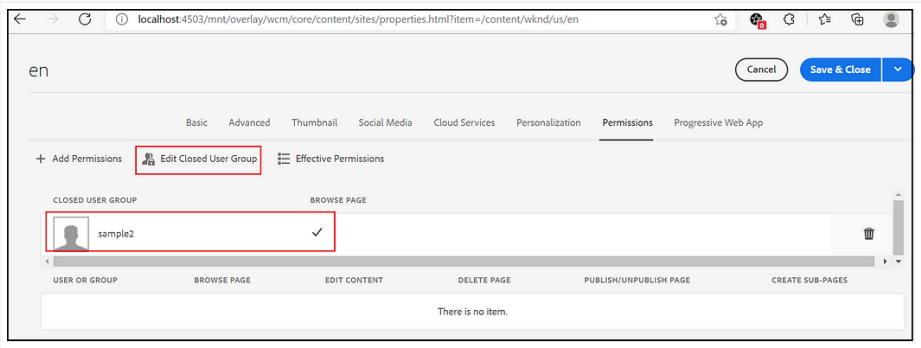


The Login page should enabled in the following patter <<current Page URL>>/j_security_check?configid=azureadb2csite1&test=test(the config id value change based on your OAuth handler configuration)

e.g /content/wknd/us/en/test.html/j_security_check?configid=azureadb2csite1&test=test

Unfortunately only enabling the url /content/wknd/us/en/test.html/j_security_check?configid=azureadb2csite1 won't work as AEM internally append .html towards end of the login URL, to avoid this append a additional random query parameter(test=test) as a workaround(not deep dived may be another better solution)

Configure the user's groups that will have the access under permissions tab — enabling "sample2" user group configured as part of the OAuth User Sync Handler.



Multi-Site Support:

The basic configuration discussed here will helps us to support the authentication for multiple sites using site specific(Domain) login/logout/profile edit URLS but if the website required different permissions enable site specific OAuth handler also better to enable site specific Application in Azure AD B2C with redirect URL's specific to a website.

The site specific authentication can be invoked through corresponding URL — **Site1:**

Login — https://mysite1.com/content/test/us/en.html/j_security_check?configid=azureadb2csite1

Log out URL — https://mysite1.com/system/sling/logout.html?operation=b2clogout

Profile Edit URL — https://mysite1.com/content/test/us/en.html?operation=profileedit&configId=azureadb2csite1

Site2:

Login — https://mysite2.com/content/test2/us/en.html/j_security_check?configid=azureadb2csite2

Log out URL — https://mysite2.com/system/sling/logout.html?operation=b2clogout

Profile Edit URL — https://mysite2.com/content/test2/us/en.html?operation=profileedit&configId=azureadb2csite2

Encapsulated Token/HMAC key synchronization:

To make the authentication stateless across multiple publisher instances, enable Encapsulated Token also sync the HMAC key across all the publishers. Refer to https://medium.com/r/?url=https%3A%2F%2Fhelpx.adobe.com%2Fexperience-manager%2F6-5%2Fsites%2Fadministering%2Fusing%2Fencapsulated-token.html%23Introduction for more details on Encapsulated Token and the steps to enable Encapsulated Token and to sync the HMAC keys.

Users synchronization

When multiple publishers are used to serving the content in a load-balanced setup, the user-created in one of the publishers should be synced to the rest of the publisher for seamless authentication.

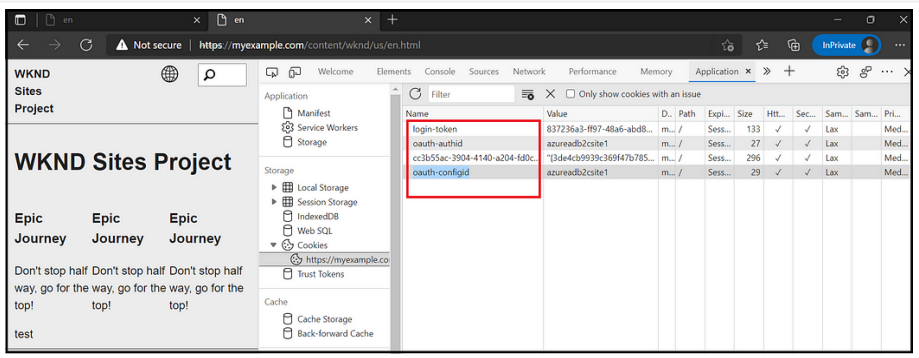
Refer to https://experienceleague.adobe.com/docs/experience-manager-65/administering/security/sync.html for more details on enabling user sync through Sling Distribution.

Dispatcher/CDN Configurations:

Add the following rule to the dispatcher website farm configuration /filter section

```
/0101 { /type "allow" /method "GET" /url "*/oauth/callback" }
/0102 { /type "allow" /method "GET" /url "*/callback/j_security_check" }
```

Also, ensure the following cookies “login-token”, “oauth-authid”, “oauth-configid”, “Cookie with the Azure AD B2C client ID” are whitelisted from CDN.



Logger Configuration:

If required you can set up a custom Logger in order to debug any issues

Logger:

```
com.core.oauth.provider.azureadb2c

com.adobe.granite.auth.oauth
```


Posted by Tech Forum at 12/10/2021 05:33:00 PM

Labels: Adobe Experience Manager

1 comment:

Magic KM May 17, 2022 at 4:46 AM

Hi the post is excellent and very detailed. One question: If the IDP does not expose a URL for fetching user properties then the integration fails because it expects IDP to expose URL for profile. Do you know how to work with that?

Reply

To leave a comment, click the button below to sign in with Google.

SIGN IN WITH GOOGLE

Newer Post Home Older Post

Subscribe to: Post Comments (Atom)