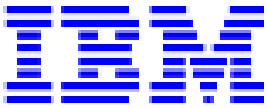
IBM FileNet Content Manager

FileNet Salesforce Connector Configuration Guide

Version 5.9.0



Copyright

Before you use this information and the product it supports, read the information in "Notices" on page 45.

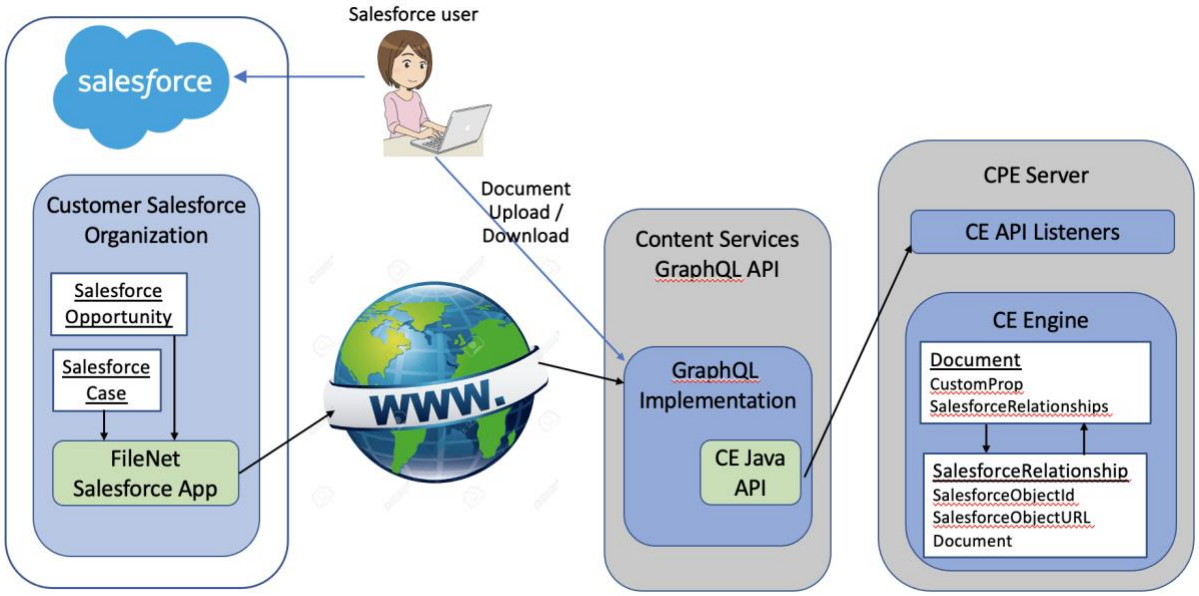
Contents

Overview

The IBM FileNet Salesforce Connector App enables Salesforce users to store their attachments as documents in their organization's FileNet Content Manager object store instead of storing the documents in Salesforce.

This document outlines the steps needed to prepare the environment on the FileNet side, as well as the steps to configure the app on the Salesforce side.

The IBM FileNet Content Services GraphQL API provides the connection between the IBM FileNet Connector for Salesforce app and the IBM FileNet Content Engine server. The following graphic describes this connection:



The FileNet Salesforce Connector App connects to the IBM FileNet Content Services GraphQL API through a secure HTTPS connection. The GraphQL API then makes calls to the Content Platform Engine server through the Content Engine Java API to interact with Documents and with SalesforceRelationship objects that associate Documents with Salesforce records.

Note that while all query and data retrieval operations are made from the Connector App running within a Salesforce.com server, there are two operations where the user's browser will connect directly to the Content Services GraphQL API server. This direct communication occurs only for document upload and download operations.

In addition to the GraphQL API configuration, you also prepare the Content Platform Engine object store and configure permissions for the content.

On the Salesforce side, you install and configure the FileNet Salesforce Connector App, then make updates to your Salesforce views to make the object store content visible and available to Salesforce users. You also configure authentication to control access to your content.

Supported Content Services GraphQL Versions

FileNet Salesforce Connector version 5.9.0 requires Content Services GraphQL version 5.5.8 or higher. However for configurations where support of multiple Salesforce organizations is needed, then version 5.5.8 IF001 or above of Content Services GraphQL API Is required.

Preparing the Content Platform Engine environment

To prepare the Content Platform Engine for integration with the FileNet Salesforce Connector App, you must deploy and configure the Content Services GraphQL API, prepare the object store, and assign appropriate permissions to the document classes.

Deploying the Content Services GraphQL API

The Salesforce app uses the Content Services GraphQL API to connect to the Content Platform Engine. The connection requires the GraphQL API to be accessible from the public internet. As a result, the GraphQL API cannot be behind a firewall or VPN.

To deploy and configure the IBM FileNet Content Services GraphQL API, follow the instructions in the following Knowledge Center topic:

https://www.ibm.com/docs/en/filenet-p8-platform/5.5.x?topic=cpdstnd-v554-later-optional-configuring-content-services-graphql-api

Note the following configuration requirements that are specific for use of the GraphQL API with the FileNet Salesforce Connector App:

CORS.xml

By default, the IBM Content Services GraphQL service will not trust calls coming from Salesforce. To allow the GraphQL service to trust incoming calls from Salesforce.com, and to allow Salesforce web pages to trust data coming from the Content Services GraphQL service, a Cross Origin Resource Sharing (CORS) configuration must be established between the two services. Configuring Salesforce to allow Resource Sharing (CORS) with the Content Platform Engine server, in this document, describes CORS settings that are required on the Salesforce side. This section describes the CORS settings required on the IBM Content Services GraphQL Service side.

The IBM Content Services GraphQL service allows the following CORS options to be set to control access by external web sites:

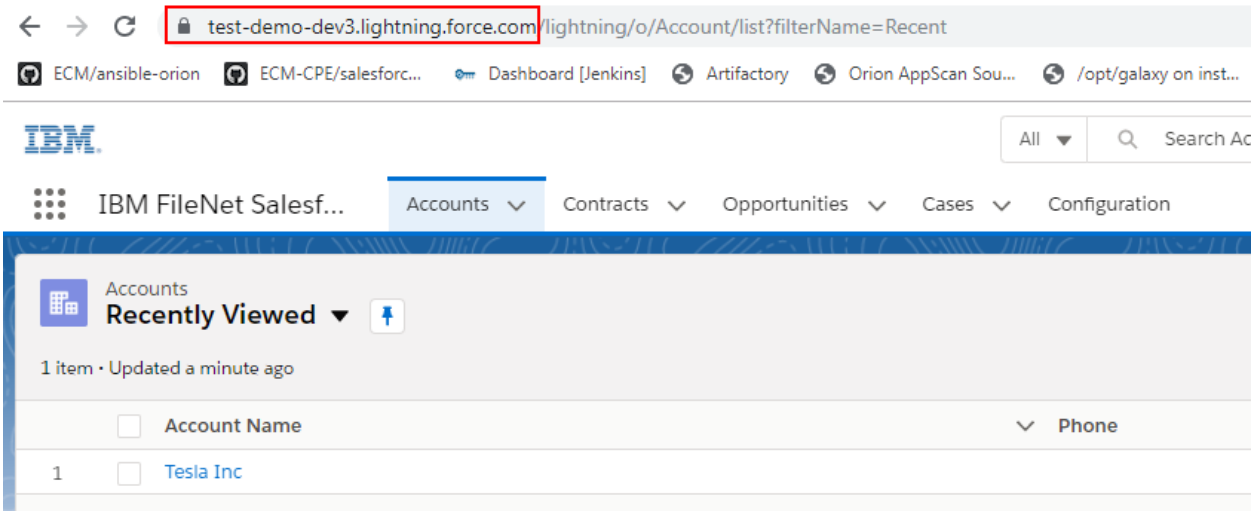
- Allowed HTTP methods
- Allowed HTTP Request Headers
- Allowed HTTP Response Headers
- Allow Credential
- Max Age

To allow calls from Salesforce.com, you must create a cors.xml file, and place it in the `${server-config-dir}/configDropins/overrides directory` of your WebSphere Liberty server, to allow calls from your Salesforce Organization's domain URL. The file should contain the following:

```
<?xml version='1.0' encoding='UTF-8'?>

<server>

<!--
https://www.ibm.com/support/knowledgecenter/en/SSEQTP_liberty/com.ibm.websphere.liberty.autogen.nd.doc/ae/rwlp_config_cors.html -->

  <cors domain="/content-services-graphql"

        allowedOrigins="https://<Salesforce_Org_URL>"

        allowedMethods="GET, POST, OPTIONS"

        allowedHeaders="Connection,Pragma,Cache-Control, ECM-CS-XSRF-Token,

XSRFtoken,Origin,User-Agent,Content-Type,Content-Length,Accept-Control-Request-Method,Accept-Control-Request-Headers,Accept,Referer,Accept-Encoding,Accept-Language,DNT,Host,Content-Length,Cache-control,Cookie,Authorization,X-ECM-SF-ORG-ID"

        exposeHeaders="Content-Disposition,Content-Length,Content_Type, ECM-CS-XSRF-Token,Content-Language,X-Powered-By,Date,Allow,Transfer-Encoding,$WSEP,DNT,Access-Control-Allow-Credentials,Access-Control-Allow-Headers,Access-Control-Allow-Max-Age,Access-Control-Allow-Methods,Access-Control-Allow-Origin,Access-Control-Expose-Headers,Connection,Cache-control,Cookie,x-content-download,X-ECM-SF-ORG-ID"

        allowCredentials="true"

        maxAge="3600" />

</server>
```

You must replace the **<Salesforce_Org_URL>** placeholder text above with your Salesforce Organization's instance domain URL (without any context root). In the following example, the domain URL is:

test-demo-dev3.lightning.force.com



If you need to configure multiple Salesforce organizations to use your object stores, then you will need to add multiple URL's in the allowedOrigins attribute above.

Authentication

When you deploy the Content Services GraphQL API, it is recommended that you initially configure Basic Authorization to get the connector app working as part of a test or development environment.

However, for production environments it is recommended to use OAuth for authentication.

There are separate sections later in this document on Configuring OAuth authentication, and Configuring Basic authentication. Please refer to those sections for details. See the instructions

in the Advanced Authentication section of the Content Services GraphQL API documentation for details of the Content Platform Engine side of this configuration.

Important:

Configuring the FileNet Connector for Salesforce app to use OAuth requires that your IBM FileNet Content Manager users use the same name to authenticate as the corresponding Salesforce user (for example, username@organization.com). If the same names are used in both places, then the Content Services GraphQL server can be configured to trust the OAuth/OIDC tokens that are generated by Salesforce.

Installing the Salesforce Integration Extensions Add-On

The Salesforce integration requires that you install a custom feature AddOn in each Object Store that is exposed to Salesforce. This AddOn defines Content Engine classes and properties that are used by the Salesforce app to associate Content Engine documents to Salesforce objects.

You download the add-on extension files, register the add-on with the server, then install the add-on. Perform these configuration steps as a P8admin in the Administration Console for Content Platform Engine.

To enable the add-on:

1. Download the following IBM Salesforce Integration Extensions AddOn files from this GitHub location:

   ```
   FNCE_SalesforceIntegrationAddOn.desc
   SalesforceIntegrationExtensions.xml
   ```

```
SalesforceIntegrationPostImportScript.js
```

2. Copy the files into a single directory on the server where you can access the IBM Administration Console for Content Platform Engine.
3. Start the New Add-On wizard in the administration console:
    a. In the domain navigation pane, navigate to **Global Configuration > Data Design > Add-ons**.
    b. Right-click the **Add-ons** node and click **New Add-On**.
4. Use the following values in the wizard:
    Select **Use the descriptor method**.
    Add-on descriptor file: Browse to FNCE_SalesforceIntegrationAddOn.desc.
    In the **Create the Add-on** dialog box: select **Optional**.
    Import data set: SalesforceIntegrationExtensions.xml
    Post-import script: SalesforceIntegrationPostImportScript.js
    For the remaining fields, accept default settings.
5. Click **Finish** to complete the wizard.
6. In the domain navigation pane of the Administration Console for Content Platform Engine, click the object store that you want to use with the app.
7. In the object store navigation pane, click the name of the object store (the top-level item).
8. From the **Actions** menu in the object store tab, click **Install Add-On Features**.
9. Select the SaleforceIntegrationAddOn, and click **OK**.

Setting permissions on document classes for use in Salesforce

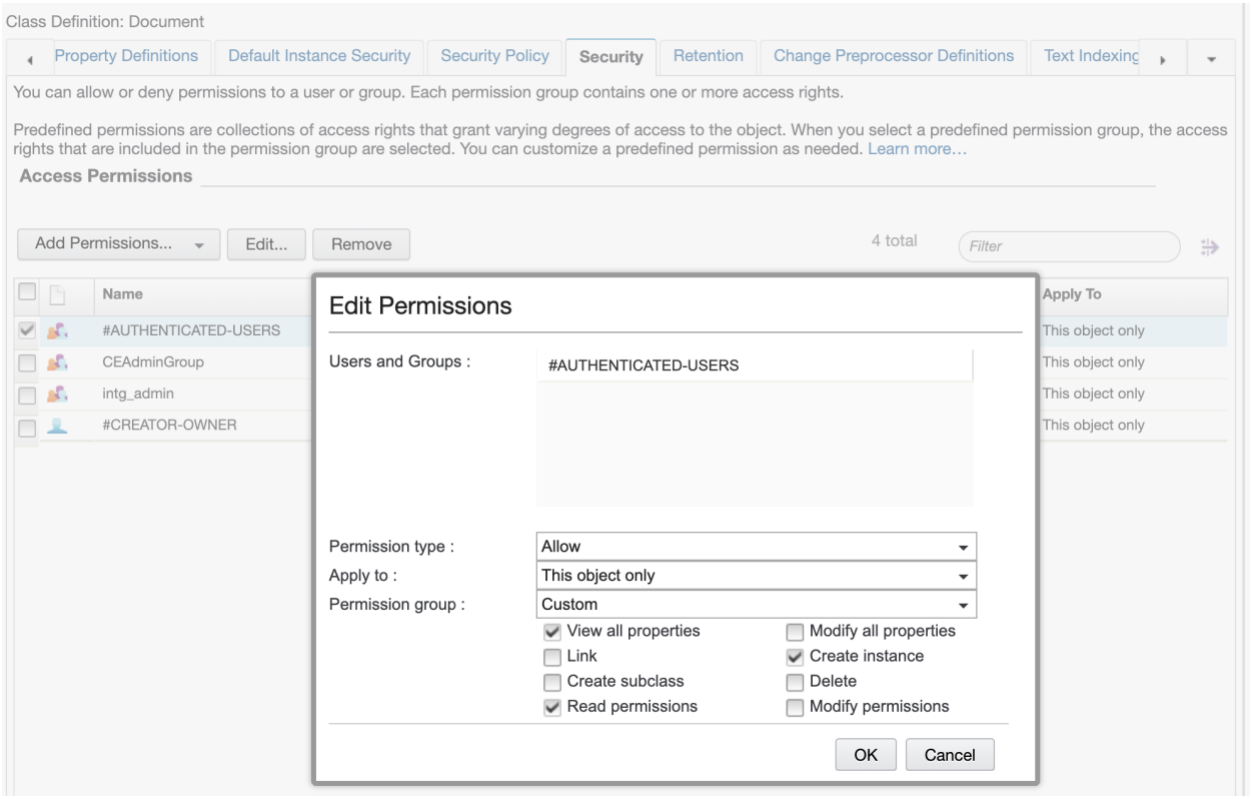
When you configure a Salesforce organization to access an IBM FileNet object store through the app, you select a set of document classes to be available to your Salesforce users for creating new documents. The settings in these document classes influence the permissions that Salesforce users have on documents that are exposed through the app. You determine what access different users should have to documents through the Salesforce app by configuring permissions on your doc classes via the Administration Console for Content Platform Engine.

Although not required, it is recommended to create a new set of classes in your object store for your Salesforce documents. You can then set permission on these classes as described in the

next section without any impact to other document classes that you might already have in your object store.

Editing document class security

There is one Document Class security permission that will impact Salesforce users: Create Instance permission.



The Document Class Security tab in the Administration Console for Content Platform Engine shows the default permission settings for a Content Engine document class. To be usable by end users, the document class should grant View All Properties, Read Permissions, and Create Instance permission to all users. Only users with Create Instance permission on a Document class are able to create a document of that class.

If your document class uses this default setting to grant these three permissions to #AUTHENTICATED_USERS, then nothing further needs to be done in the Document Class permission settings. However, if you are not granting these permissions to

#AUTHENTICATED_USERS, then create a group containing all of your Salesforce users, and use the Administration Console for Content Platform Engine to add a permission to each document class that will be exposed in Salesforce, granting these permissions to the Salesforce user's group.

Editing the document class Default Instance Owner

The owner of a document gets implicit permissions on the document, including the ability to read, view, edit, and change permissions. Therefore, the owner of a document can always view and edit the document through the Salesforce application.

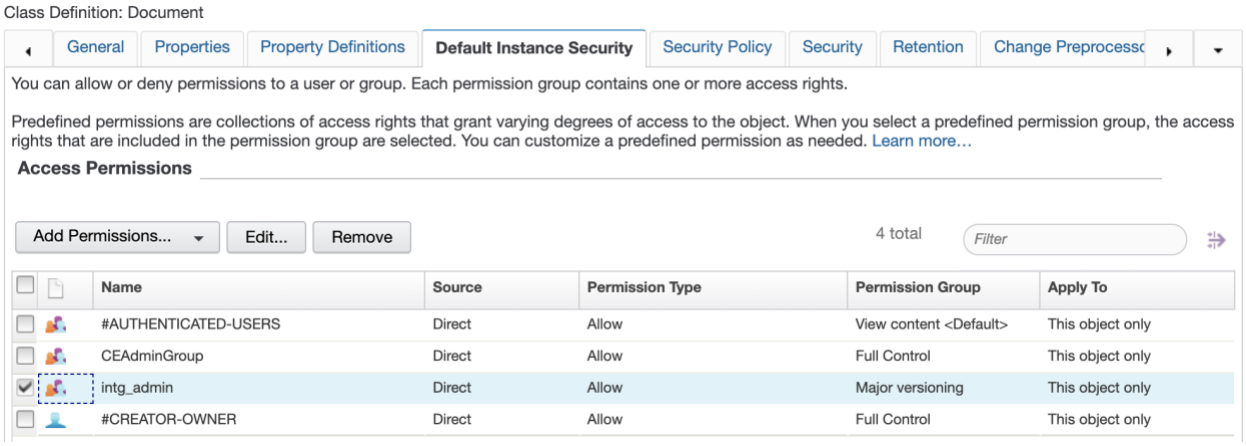
The Default Instance Owner property of a document class determines who the default owner will be for a new document of the document class. FileNet applications (including the Salesforce app) typically rely on this document class setting to control who the owner of new documents will be.

The Default Instance Owner of a document class can be viewed and edited in the Administration Console for Content Platform Engine by scrolling to the bottom of the Default Instance Security tab:

For document classes that are exposed in Salesforce, the default setting of #CREATOR_OWNER should be maintained for the Default Instance Owner property. This setting means that the person who creates a document is always it's owner.

Editing the document class default instance security

The permissions on documents created by the IBM FileNet Salesforce Connector app are determined by the Default Instance Security settings on the exposed document classes. The following screenshot shows an example of the Default Instance Security for a document class:



Configuring users who can view existing documents and their properties

In the example above, the #AUTHENTICATED_USERS pseudo-group (which represents all users who are able to login to the server) is granted the View Content security group. This security group includes the following permissions: View all properties, View content, Read permissions. These permissions allow users to view document content and properties from the Salesforce app, but not the ability to edit document properties or create new versions of a document.

When an object store is created, one or more user groups can be specified as object store users. If such a group is specified at object store creation time, then they will be given the View content access group for all document classes by default. If no such group is specified at object store creation time, then #AUTHENTICATED_USERS is granted the View Content access group for all document classes by default.

Customers may edit the Default Instance Security settings to limit access to smaller groups, or to remove all access, and instead use security mechanisms other than Default Instance Security to control document permissions.

For the Salesforce environment, we recommend that Default Instance Security should grant View Content either to #AUTHENTICATED_USERS, or to a group that will contain users who should have read-only access to all Salesforce documents.

Note that if a user has only View Content permission, then they will be able to edit properties and content for documents that they create, but not for documents that were created by any other user.

Configuring users who can edit properties or create new versions of existing documents

In the example above, the intg_admin group is granted the Major Versioning permission group. This security group includes permissions to modify properties or create new versions, in addition to the permissions that the View Content group has. Users who should be allowed edit documents that were created by other Salesforce users should be given this permission.

If a customer wanted to grant permission to edit document properties, but not permission to create new document versions, then the group should be granted the Modify Properties permission group, rather than the Major Versioning permission group.

For the Salesforce environment, we recommend that Default Instance Security should grant Major Versioning permission to either some or all Salesforce users, by assigning this permission group to a group that contains these users (or to #AUTHENTICATED_USERS).

In the example above, the #CREATOR_OWNER pseudo-user, and as the CEAdminGroup, have both been granted the Full Control permission group. These permissions allow the grantees to view and edit the documents, as well as to remove them from the Salesforce record.

For the Salesforce environment, we recommend that Default Instance Security should grant Full Control permission to a group who are admins within Salesforce, as well as to the #CREATOR_OWNER pseudo-user.

Configuring properties to be synchronized with fields on Salesforce records

Whether you create new document classes for use by the Salesforce Connector, or choose to use existing document classes, you have some options for which properties on your classes are visible and settable in Salesforce. You also have the option to create some new properties to hold copies of Salesforce fields from the Salesforce records where you have added FileNet document attachments.

You can create special properties on your document class that are automatically populated with values from the Salesforce record that a document is added from. When creating a new document, the IBM FileNet Salesforce Connector automatically looks for any properties on the target document class whose symbolic name begins with the prefix “Fnsf”. If any properties with this prefix are found, then the Connector attempts to find a Salesforce field with a matching name, and then populates the FileNet property with the value from that field during document creation.

For example, if the Salesforce record where a document is being added has a field whose name is “AccountName”, and a property exists on the target document class with the symbolic name

“FnsfAccountName”, then that property is automatically populated with the value of the “AccountName” field.

When you configure an object store in Salesforce, you can select the document classes that are exposed to Salesforce users, and select which document properties of those classes Salesforce users can set the values for when they create a new FileNet document through Salesforce. If a property with an “Fnsf” prefix in its symbolic name is selected, then it appears on the New Document dialog of the Document List View widget as read-only. The user cannot override the value from the Salesforce record.

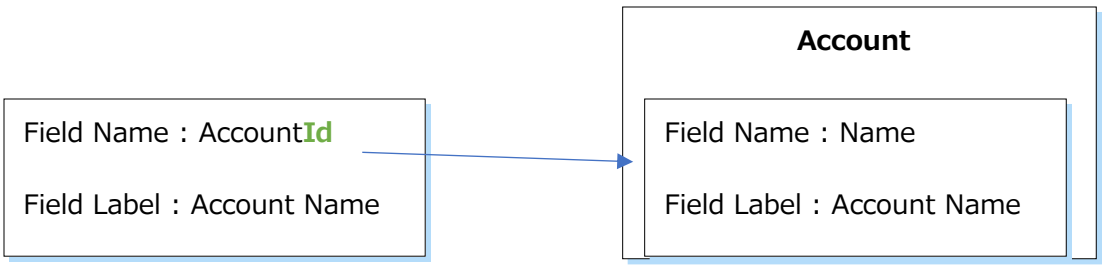
If you want to store a copy of the entire Salesforce record in your FileNet object store, you can do that by creating a property with the symbolic name “FnsfJSONRecord” in your target document class. If this property exists, then the JSON for the Salesforce record is copied into it when a document is added to the record through the IBM FileNet Salesforce Connector. Note that this property must be a string property, with the UsesLongColumn flag set to true, because this data can be moderately large.

Synchronizing Salesforce Lookup fields

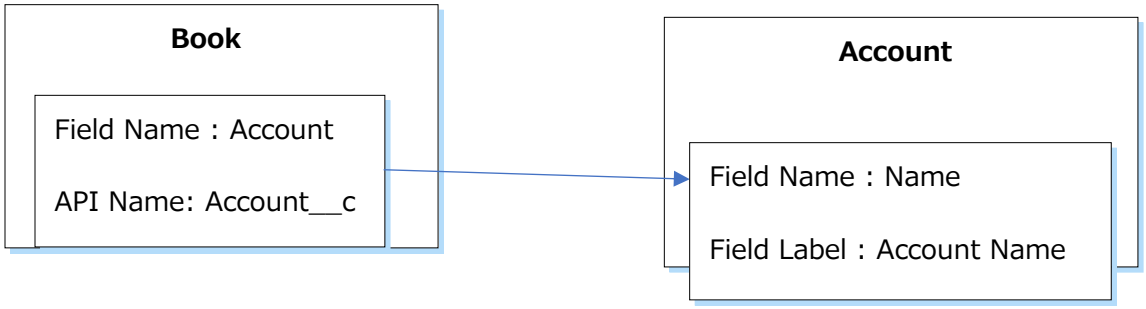
A Salesforce record can have a field which is a reference to the value of a field on a different Salesforce record. This type of field is called a Lookup field. When you are using a Lookup field between two standard Salesforce records, there is no need for any special handling to synchronize it. However, if you have a Lookup field in a custom Salesforce record, then the name of the Lookup field is different, and the name of the FileNet property needed to synchronize it is different.

In a standard Salesforce record, the name of a Lookup field must always end with “Id”. For example, the Salesforce Opportunity record has a field called “AccountName” (Field Label = “Account Name”) who’s Field Name is AccountId. This field is a reference to the “Name” field of an Account record. In this case, you would simply create a FileNet property called “FnsfAccountName” to synchronize the property.

Opportunity



For a custom Salesforce record, these lookup fields work differently. it is not required that the Field Name end with “Id”, but the field being referred to is designated by the API Name attribute, and “__c” is appended to the field name of the lookup field



In this case, the FileNet property must include the name of the referenced object, followed by “__r” and then the name of the referenced field. For example, if a custom Salesforce object references the same Account Name field as a lookup field, as in the example shown above, then the FileNet property needed to synchronize that field is named “FnsfAccount__rName”.

Limitations

Note that there are some limitations of this field synchronization feature:

- The property on the FileNet document has the value of the Salesforce field at the point in time when the document is first added as an attachment on the Salesforce record. If the value of the Salesforce field is subsequently updated, the value of the property on the FileNet document is not updated.
- If the document is later added to additional Salesforce records through the **Add To Record** feature, it is not updated with field values from the new record.
- Only fields that are of string, date, integer, or picklist value can be synchronized through this feature. Salesforce fields of type Lookup which reference fields on other Salesforce records can be synchronized, as long as the field that they reference is of a supported type. Salesforce fields of type Formula can also be synchronized, if the formula yields a value that is a supported type. The data type of the Salesforce field must match the data type of the FileNet property, or the synchronization does not work
- If the length of the Salesforce string exceeds the length of the corresponding FileNet property, then the string is truncated to fit in the FileNet property

Preparing the Salesforce Organization

A Salesforce administrator user must install and configure the IBM FileNet Salesforce Connector app with a Salesforce Organization.

Note that you must accept the license agreement for the Salesforce Connector application before the app can be installed. The specific terms and conditions for this application are available in the installation wizard, and also on the AppExchange listing page. License terms for the prerequisite products are accepted during the purchasing of those products.

Installing the FileNet Salesforce Connector app in a Salesforce Organization

To install the FileNet Salesforce Connector app:

1. From the Salesforce Organization, click the AppExchange link.
2. Browse for and select the FileNet Salesforce Connector app.
3. Supply values for the AppExchange installation wizard. When asked, choose **Install for Admin Users only.**
4. Click **Install now** to complete the wizard.

It can take a few minutes for the installation to complete.

Configuring Salesforce after installation

After you install the app in the Salesforce organization, you must perform additional configuration steps in Salesforce to make the app features available for users.

Creating a CSP Trusted URLs for the Salesforce Organization

You must create a CSP Trusted Site in the Salesforce Organization so that the IBM FileNet Salesforce Connector can work securely with the Content Services GraphQL server.

To create the CSP Trusted Site:

1. From Setup, use the **Quick Find** field to find CSP Trusted URLs.
2. From **CSP Trusted URLs**, click **New Trusted URL**
3. For **API Name**, enter `IBM_Content_Services_Endpoint`.
4. For **URL**, enter the URL for the IBM Content Services GraphQL API.
5. Select the **Active** checkbox.
6. For **Context**, select the **All** option (default).
7. Click **Save**.

See the following Salesforce documentation for more details:
https://help.salesforce.com/articleView?id=csp_trusted_sites.htm&type=5

SSL Certificate requirement on the Content Services GraphQL server

When IBM FileNet Salesforce Connector App sends an outbound message to the Content Services GraphQL server, using a CSP Trusted Site, the Salesforce.com organization acts a client that will only trust the target host (that is the Content Service GraphQL server) if it presents an SSL Certificate signed by a root Certification Authority (CA). Self-signed certificates cannot be used by the target host. Also the target host URL must be specified using a registered DNS domain name, matching the domain name in the SSL certificate; the target host URL cannot be specified using its public IP address. If a reverse proxy like IBM HTTP Server or Nginx is used in front of Content Services GraphQL server then the root CA SSL certificate must be installed on the reverse proxy. Refer the reverse proxy documentation on SSL configuration.

Configuring Salesforce to allow Resource Sharing (CORS) with the Content Platform Engine server

By default, browsers do not allow a request to one web site to retrieve many types or resources from a different web site. Cross Origin Resource Sharing (CORS) is a standard that allows a primary web site to request access to resources from a secondary web site, and for the secondary web site to be configured to allow this access. The IBM FileNet Connector for Salesforce requires CORS access to be configured on both the Salesforce Organization, and on the IBM Content Services GraphQL API service.

To configure CORS within Salesforce, the Salesforce admin must add a trusted domain to the Salesforce organization's CORS whitelist.

To configure resource sharing:

1. From Setup, use the Quick Find field to find **CORS**.
2. Click **New**.
3. For **Origin URL Pattern**, enter the domain of the IBM Content Services GraphQL API.
4. Click **Save**.

Configuring Authentication to the IBM Content Services GraphQL API Service

Salesforce supports following authentication protocols:

- User Password
- OAuth 2.0
- OAuth 2.0 JSON Web Token (JWT)
- JWT Token Exchange
- AWS Signature Version 4

IBM FileNet Salesforce Connector App supports User Password (aka BasicAuth) and OAuth 2.0 JSON Web Token (JWT). Both protocols require creating a NamedCredential and completing a set of additional configuration steps.

To configure the app to use OAuth authentication, see the instructions in section 5 of this document. To configure the app for BasicAuth authentication, see the instructions in section 6 of this document.

Configuring Salesforce users who have administrator access for the Connector

When the IBM FileNet Connector for Salesforce is installed, it creates two Permission Sets, which control which users have which access levels within the connector. The first of these is the **IBM FileNet Admin** Permission Set

Only administrative users should have access to sync an IBM FileNet Content Manager object store with a Salesforce organization, or to configure which properties are editable in the document properties dialog. To grant Salesforce users this access, they must be added to the **IBM FileNet Admin** Permission Set.

To configure admin users:

1. In the **Setup** menu, go to **Users in the Administration**, and select **Permission Sets**.
2. Select the **IBM FileNet Admin** permission set.
3. Click **Manage Assignments** on the permission set screen.
4. Select the user or users that you want to add to the permission set, or unselect users that you would like to remove from the permission set, and click **Assign**.
5. Click **Save** to save your changes.

Configuring Salesforce users who have non-admin access to IBM FileNet documents

The second Permission Set that is created during the installation is the **IBM FileNet User** Permission Set. All users who need runtime access to retrieve, create, or update documents through the IBM FileNet Salesforce Connector must be added to the **IBM FileNet User** Permission Set.

To add non-administrator users:

1. In the **Setup** menu, go to **Users in the Administration**, and select **Permission Sets**.
2. Select the **IBM FileNet User** permission set.
3. Click **Manage Assignments** on the permission set screen.
4. Select the user or users that you want to add to the permission set, or unselect users that you would like to remove from the permission set, and click **Assign**.
5. Click **Save** to save your changes.

Configuring custom HTTP headers for calls to GraphQL

The IBM FileNet Connector for Salesforce makes calls to the IBM Content Services GraphQL API in order to create, search, retrieve, or update documents. In some customer configurations, this GraphQL API traffic must flow through an intermediate router or gateway device, and it is necessary to have custom HTTP headers set in order to flow correctly through these network devices. To support this type of requirement, a Salesforce admin may set custom headers through the following process:

1. In the App Launcher, select the IBM FileNet Salesforce app
2. On the More menu, select “Custom headers”
3. Use the “New” button to bring up the “New Custom Header” dialog
4. Enter the Name of the header, and it’s Value
5. Use the Save button to save the new header

Existing custom headers can be edited or deleted from the “Custom headers” page. No more than 100 custom headers can be created.

Note that when a header is added, you must also edit the “allowedHeaders” field of the CORS.xml file on your Content Services GraphQL server, so that requests that contain the new header will be allowed.

Configuring Salesforce Digital Experience Community Site

The IBM Document List widget can be added to the Salesforce Digital Experience Community record pages.

To add the IBM Document List Widget on the Salesforce Community pages:

1. Create a new Permission set with the following details:
   Label: IBM FileNet Customer Community Plus Login
   API Name: IBM_FileNet_Customer_Community_Plus_Login
   Description: Grant access to Customer Community Plus Login user for IBM Salesforce Connector Application
   License: Customer Community Plus Login

2. Update IBM FileNet Customer Community Plus Login Permission set with the following:
   1. Add "IBM FileNet Salesforce Connected App" app to Assigned Connected App.
   2. Object Settings:
      a. Account: Read access on Object Permissions
      b. Custom Headers – Read access on Object Permissions and All Read access on Field Permissions.
      c. Ecm Metadata Choice List Items: Read access on Object Permissions and All Read access on Field Permissions.
      d. Ecm Metadata Class Descriptors: Read access on Object Permissions and All Read access on Field Permissions.
      e. Ecm Metadata Property Descriptors: Read access on Object Permissions and All Read access on Field Permissions.
      f. Ecm Object Stores: Read access on Object Permissions and All Read access on Field Permissions.

   3. Apex Class Access: Add all Apex classes from IBM_FN_CM namespace.
   4. Visualforce Page Access: Add the IBM_FN_CM.viewerPage
   5. External Credential Principal Access: For Basic Authentication add the IBMFileNetExternalCredential_BasicAuth. For OAuth/JWT add the IBMFileNetExternalCredential_JWT.
   6. Customer Permissions: Add IBM_FN_CM.IBM FileNet User Document List
   7. Assign all Community Users to this permission set.

3. Sharing Access
   a. From Setup, find Sharing Setting, and update the Default External Access on Ecm Object Store to **Public Read Only**.
4. In the Site Administration, under Members section, select this Permission set.

Configuring the Salesforce Organization to use object stores

Once the IBM FileNet Salesforce Connector app has been installed and all of the post-install configuration steps are complete, you are ready to configure an object store for use in your Salesforce organization. Multiple object stores can be configured with a single organization if you choose.

For each object store that is configured, the admin chooses the document classes to be available for Salesforce users to store documents. Additionally, for each document class that is selected, the admin can select the subset of properties that a user can set when they create a new document or update existing documents.

You must be a member of the **IBM FileNet Admin** Permission Set to perform any of the actions in this section.

Configuring an object store

You use the Connector app to choose an object store and select the Document classes and associated properties that you want to use for your application.

To configure an object store:

1. Use the App Launcher to navigate to the **IBM FileNet Salesforce Connector** tab.
2. Select the **Configuration** menu option.
3. From the drop-down list of object stores, select the object store that you want to configure.
4. From the list of available document classes for the object store you that selected, use the checkboxes to select or unselect classes.
5. For each document class that you select, use the **Select Properties** link next to the class to configure the properties to make editable for that class. In the **Properties to Display** dialog, you can select the properties to display from the list of available properties, and you can also use the up and down arrows to adjust the order in which the properties are displayed.

An asterisk (*) before the property name indicates that the property is automatically populated from the corresponding Salesforce field. This auto-population occurs whether or not you select it as a property to display. If you do select it as a property to display, then it appears as a read-only property in the document properties dialog.

Note: You can select up to fifty Document classes. The limit on the total number of selected Document classes depends on the number of Properties and the Choice List of the Document class. You may experience the Salesforce rate limit for API callout requests when selecting more than seventy-five Document classes. You can purchase additional API calls from Salesforce to increase the number of calls per day to allow selecting additional Document Classes.

6. When you have completed setting up your document classes and properties, click **Save** to save your choices.

If you get an error when you attempt to configure an object store that says that the Salesforce Integration Extensions AddOn has not been installed, refer to the instructions in Installing the Salesforce Integration Extensions Add-On.

Note that the Configure object store action can be run multiple times for an object store, to change the selected document classes, or to change the list of fields that are selected for a given class. After an object store has been configured once, the **Reconfigure** button can be used to configure it again.

After you configure an object store, you add a Document List widget to the relevant Salesforce pages to make the object store visible to end users.

Adding the Documents List widget to your Salesforce Organization screens

Note: Only a Salesforce system administrator can modify a page layout, as described in this section. The edit page link is not visible to a standard user.

For each configured object store, you must add the IBM Documents List Widget to the pages where you want IBM FileNet documents to be accessible. The widget can be added to any Salesforce page, including pages for custom Salesforce record types. The widget can also be

added to Record details pages within Salesforce Digital Experience Communities. The following steps use the Account page as an example.

To add the Documents List widget to a page:

1. Go to the App Launcher, and start the IBM FileNet Salesforce Connector app.
2. Click the **Accounts** tab.
3. Select an individual account, and go to the **Related** tab. The IBM Documents widget is not visible.
4. From the **Setup** menu, select **Edit page**.
5. From the Lightning Components bar, under **Custom > Managed** menu, select the **DocumentsList** component and drag and drop it to your desired location on the **Related Items** tab.
6. Make this component visible to users that have permission to access the component:
    a. Click **Add Filter**.
    b. Click the **Advanced** tab, click **Select**, and from the **Type of filter** list, click Permission.
    c. From the **Permissions** drop down menu, select **Custom permissions**.
    d. From **Custom permissions**, select IBM_FileNet_User_Document_List.
    e. Click **Done**.
    f. From Filter Type dialog, make sure that the operator is Equal, and Value is True, then click **Done**.
7. Optionally adjust the attributes of the DocumentList component. When the component is selected within Edit page, it's attributes can be adjusted in the dialog on the right side of the screen:
    a. Title: controls the title of the DocumentList component
    b. Object Store: controls the object store that the DocumentList component will issue its queries against
    c. VisibleRows: controls the maximum number of rows that will be shown in the dialog
    d. Hide Document Actions: To hide the action from the Document Action Context Menu, enter the list of actions (separated by a comma. e.g., UploadNewVersion,AddToRecord,Delete).
    The valid action names are as follows:
        1. Properties
        2. View
        3. Download
        4. UploadNewVersion

5. RemoveFromRecord
6. AddToRecord

e. Select Document Classes: This option allows you to only display documents of specific Document Class. Enter the list of Document Class Symbolic Name (separated by a comma).
f. Allowed File Extensions: This option enables the uploading of files with only specified file extensions or MIME types. The following file extensions and MIME types are supported.

MIME Type	**Extension**	**Kind of document**
application/json	.json	JSON format
application/pdf	.pdf	Adobe Portable Document Format (PDF)
application/octet-stream	.bin	Any kind of binary data
application/epub+zip	.epub	Electronic publication (EPUB)
application/vnd.oasis.opendocument.text	.odt	OpenDocument text document
application/vnd.oasis.opendocument.spreadsheet	.ods	OpenDocument spreadsheet document
application/x-msdownload	.exe	Executable file
application/vnd.openxmlformats-officedocument.presentationml.template	.potx	Microsoft PowerPoint Template
application/x-7z-compressed	.7z	7-zip archive
application/zip	.zip	ZIP archive
application/gzip	.gz	GZip Compressed Archive
application/x-gzip	.gz	GZip Compressed Archive
application/vnd.ms-excel	.xls	Microsoft Excel
application/vnd.openxmlformats-officedocument.spreadsheetml.sheet	.xlsx	Microsoft Excel (OpenXML)
application/vnd.ms-powerpoint	.ppt	Microsoft PowerPoint
application/vnd.openxmlformats-officedocument.presentationml.presentation	.pptx	Microsoft PowerPoint (OpenXML)
application/msword	.doc	Microsoft Word
application/vnd.openxmlformats-officedocument.wordprocessingml.document	.docx	Microsoft Word (OpenXML)
audio/mpeg	.mp2	MP2 audio
audio/mpeg	.mp3	MP3 audio
audio/ogg	.ogg	Ogg audio
audio/wav	.wav	Waveform Audio Format
audio/flac	.flac	Free Lossless Audio Codec
font/otf	.otf	OpenType font
font/ttf	.ttf	TrueType Font
image/jpeg	.jpeg, .jpg	JPEG images

image/png	.png	Portable Network Graphics
image/gif	.gif	Graphics Interchange Format (GIF)
image/webp	.webp	WEBP image
image/bmp	.bmp	Windows OS/2 Bitmap Graphics
image/tiff	.tiff, .tif	Tagged Image File Format (TIFF)
image/svg+xml	.svg	Scalable Vector Graphics (SVG)
text/plain	.txt	Text, (generally ASCII or ISO 8859-n)
text/csv	.csv	Comma-separated values (CSV)
text/css	.css	Cascading Style Sheets (CSS)
text/html	.html, .htm	HyperText Markup Language (HTML)
text/javascript	.js	JavaScript
text/xml	.xml	XML
video/mpeg	.mpeg, .mpg	MPEG Video
video/ogg	.ogv	Ogg video
video/webm	.webm	WEBM video
video/quicktime	.mov	QuickTime Movie
video/x-matroska	.mkv	Matroska Video
video/3gpp	.3gp	3GPP video container

8. Click **Save** to save the edited layout
9. To activate these changes, click **Activation**. Click **Assign as Org Default**, then click **Save**.
10. Click Back to return to the main screen.

Repeat the steps for other Salesforce pages, as needed.

Customizing the User Experience Programmatically

In addition to deploying the Document List component, you can also add FileNet Salesforce Connector capabilities to Salesforce pages using programmatic control. You can change the behavior of the Document List component, and add a different component which allows documents to be uploaded without showing the list of existing documents. To do this, you must wrap the IBM Document list view component, and/or the IBM Add Document component, with a custom Salesforce Lightning Component that controls them programmatically. Some of the customizations that are possible programmatically are:

- Setting the default document class when a new document is created
- Setting defaults for properties when a new document is created

- Hiding some or all of the properties that are shown when a new document is created
- Hiding options on the context menu of the Document list view control
- Exposing an Add document control which allows a new document to be uploaded and attached to a Salesforce record, without showing the list of existing documents for the record

After you have created a custom Lightning component that wraps either the IBM Document List Widget, or the IBM Add Document component, then you can add your custom component to the target page, to get the customized behavior. Note that when adding a component that wraps the IBM Add Document component, you must set the Object Store that attribute for the wrapping component, when the component is selected.

For more details on programmatic control of the user experience, see the Salesforce Connector for FileNet Programming Guide.

Assigning Licenses for the IBM FileNet Salesforce Connector Package

To assign licenses to Salesforce AppExchange users:

1. From Setup, enter **Installed Packages** in the Quick Find box, then select **Installed Packages** to find the IBM FileNet Salesforce Connector package.
2. Click the **Manage Licenses** link before the package name.
3. Click **Add Users**.
4. Click the checkbox in the Available users section to select users
5. Click **Add** in Selected Users.

Removing an object store

If you want to disassociate an object store from a Salesforce organization, you can do so by selecting the object store in the **Configure Object Store** tab, and then using the **Remove** button.

This action prevents any new documents from being added to this object store, as well as the listing or viewing of any existing documents. It does not, however, remove the Document List widget for the Object Store from the Salesforce pages where it is in use. The Document List widgets will show an error when pages that contain the widget are viewed.

If you use the **Remove** button to remove an object store association, and you then re-configure the object store, then all of the documents will come back. If you do not intend to immediately re-configure the object store, then you should manually remove the Document List widget from all Salesforce pages prior to performing the Remove operation.

Configuring OAuth authentication

You can configure the IBM FileNet Salesforce Connector app to authenticate to the IBM Content Services GraphQL server by using the OAuth 2.0 protocol.

Configuring OAuth authentication in Salesforce

The IBM FileNet Connector for Salesforce app references a Salesforce Named Credential when it accesses the IBM Content Services GraphQL server. The Named Credential is configured to perform all of the necessary authentication steps when a connection from Salesforce to the IBM Content Services GraphQL server is made.

Named Credential

A Named Credential is a configuration that declaratively manages a Salesforce Organization's authentication to an external service. The credential specifies the URL of the external service and its authentication parameters. The connection between Salesforce and the external service is established using the Authentication Protocol parameter in a Named Credential.

Prior to the 5.9.0 release of the IBM Salesforce Connector for FileNet, customers were instructed to manually create a Named Credential for use by the connector app. However, In the Winter '23 release of Salesforce, a new Named Credential was introduced that is more extensible, customizable, and secure. The previous Named Credentials are now treated as Legacy Named Credentials. Legacy Named Credentials are deprecated and unsupported in the current release of Salesforce.

It's important to note that the Apex code from the managed package can no longer access Named Credentials created manually by Salesforce Admins in the subscriber org. For this reason, the prior approach of creating new Named Credentials will no longer work. For more information, please refer to the note section in the relevant Salesforce article.

https://developer.salesforce.com/docs/atlas.en-us.apexref.meta/apexref/apex_ConnectAPI_NamedCredentials_static_methods.htm

As Legacy Named Credentials will be phased out in the future, the IBM Salesforce Connector has been updated to support the new Named Credentials. With the release of IBM Salesforce Connector v5.9.0, support for Legacy Named Credentials has been removed. Upon installation

or upgrade to v5.9.0, a new set of Named Credentials will be added to the subscriber org, which Salesforce Admins can configure. Two placeholder Named Credentials are created during the IBM Salesforce Connector installation:**IBMFileNetCredential_JWT** and **IBMFileNetCredential_BasicAuth.** The following sections will detail how these placeholder named credentials are modified to enable either OAuth or BasicAuth for the subscriber org.

OAuth 2.0 JSON Web Token (JWT)

OAuth 2.0 is an open standard which allows simple and powerful server-to-server API integration. The main advantage of using the OAuth 2.0 protocol over the User Password protocol is that Salesforce users do not need to manage their own credentials for the external system.

The OAuth 2.0 JWT is an OAuth flow, similar to Web Server flow within OAuth 2.0, which uses a JWT format of the OAuth access token. With this flow, Salesforce creates a token using the logged on user's identity, and digitally signs the token using a private key. Whenever the IBM FileNet Salesforce Connector App makes a callout to IBM Content Services server, Salesforce sets this token in the Authorization header of the HTTP Request object.

The value of the Authorization header looks like the following:

```
Authorization: Bearer eyJraWQiOiJTY...eyJpc3MiO...FEjHNzZ...
```

IBM FileNet Content Services server parses the token and verifies the signature using the SSL Certificate stored in server's KeyStore. The server also checks the expiration time in the token. An error response is returned if signature verification fails or token is already expired.

To configure OAuth 2.0 JWT:

1. Generate a new Self-Signed or CA Signed certificate.
2. Create a Connected App.
3. Create a Named Credential.

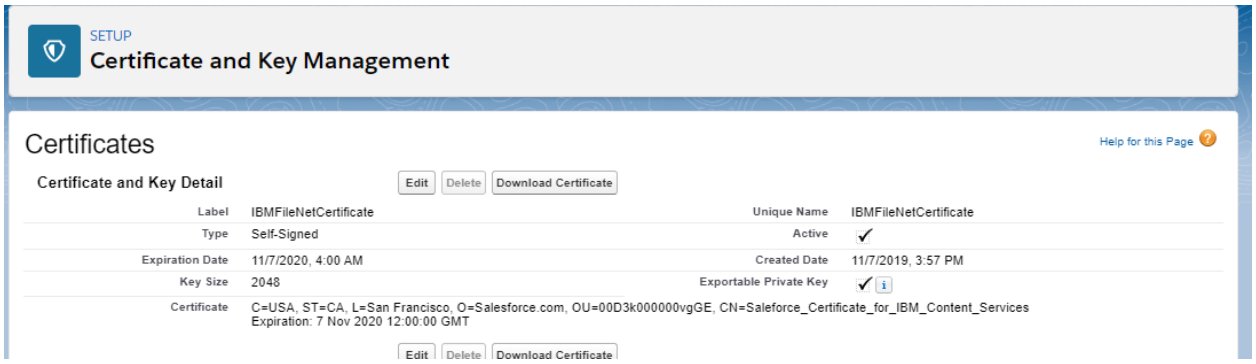
Generating a new self-signed certificate

You use Salesforce to generate the new certificate.

To generate a new self-signed certificate:

1. From Setup, enter Certificate in the Quick Find Box, then select **Certificate and Key Management.**
2. Click **Create Self-Signed Certificate**.
3. Enter the following fields:
   - Label: IBMFileNetCertificate
   - Unique Name: IBMFileNetCertificate
4. Click **Exportable Private Key**.
5. Click **Save**.
6. Click **Download Certificate**.

The certificate is downloaded to your system. Make a note of the location so you can use this certificate in a later configuration procedure.



Creating a connected app

You use Salesforce to create a connected app.

To create a connected app:

1. From **Setup**, enter App Manager in the Quick Find Box, then select **App Manager**.
2. Click **New Connected App**.
3. Enter the following fields:
   - **Connected App Name**: IBM FileNet Salesforce Connected App
   - **App Name**: IBM_FileNet_Salesforce_Connected_App
   - **Contact Email**: Enter your e-mail address
4. In the API (Enable OAuth Settings) section, click **Enable OAuth Settings**.
5. In **Callback URL**, enter the IBM Content Services redirect URL, for example,

   Content Services deployed on WebSphere Liberty:

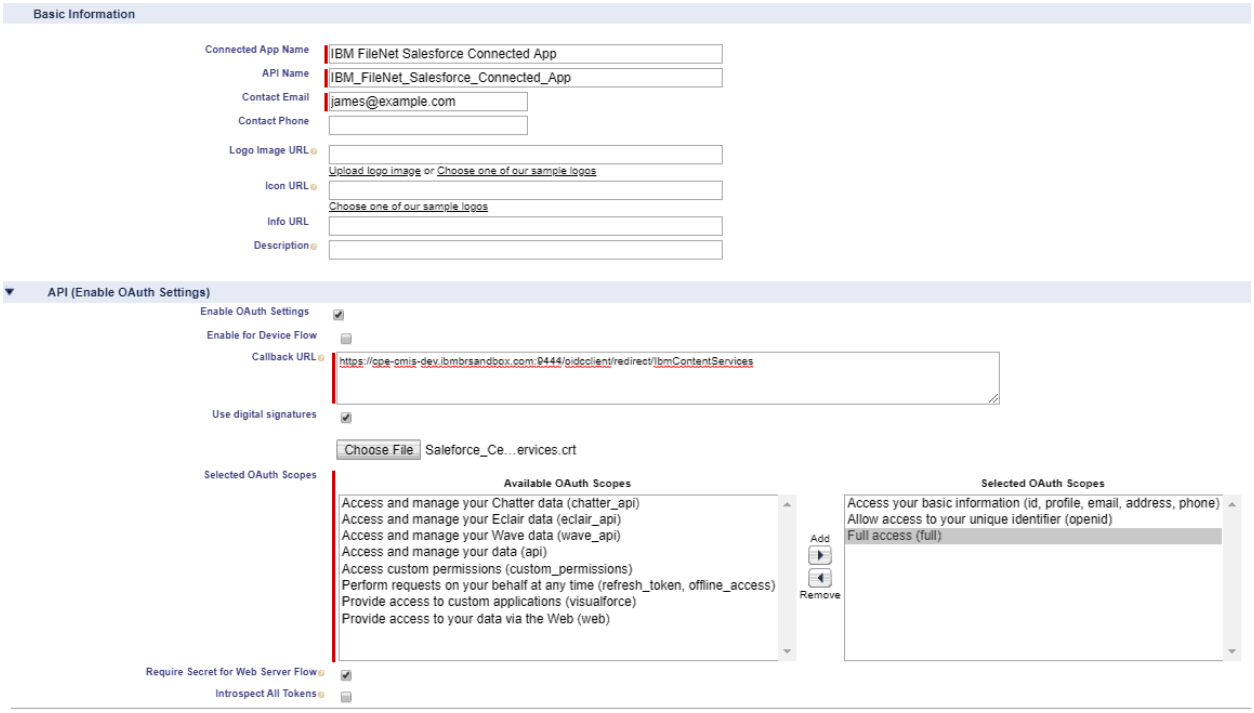
   https://cpe-dev.ibmbrsandbox.com:9444/oidcclient/redirect/IbmContentServices

   Content Services deployed on traditional WebSphere Application Server (tWAS)

   https://cpe-dev.ibmbrsandbox.com:9444/oidcclient/IbmContentServices

   Note: The tWAS path does not include /redirect.

   You can change the callback URL at later time. If you don't know the URL when creating the Connected App, you can enter a placeholder URL, and change it at a later date.
6. Select **Use digital signature**, click **Choose File**, and select the certificate that you downloaded to your system in Generating a new self-signed certificate.
7. From **Available OAuth Scopes**, add the following to **Selected OAuth Scopes**:
   - Access your basic information (id, profile, email, address phone)
   - Allow access to your unique identifier (openid)
   - Perform requests on your behalf at any time (refresh_token, offline_access)
8. Check **Require Secret for Web Server Flow**.



9. Confirm all fields and click **Save**.
10. On the Salesforce status message, click Continue.
11. From the IBM FileNet Salesforce Connected App page, in the **API (Enable OAuth Settings)** section, copy the value for the **Consumer Key** in the text editor. You will use this value in Creating a Named Credential.

The screenshot below shows an example of the Connected App details:

Configuring Salesforce users who can use the Connected App to authenticate

When using OAuth, users must reference the **IBM FileNet Salesforce Connected App** Connected App directly from their profiles. This approach requires that they have access to the Connected App. There are two options to complete this step, depending on whether you want all users to be able to use the connector, or only certain users.

Option 1: Allow access for all users in the Salesforce Organization (Recommended)

If you want all Salesforce users within an organization to have non-administrative access to the IBM FileNet Salesforce connector, add the Connected App that was created in a previous step to the Standard Profile for the organization (or the user profile of your choice).

1. In **Setup**, go to **Profiles**.
2. Select the **Standard User** profile (or the user profile of your choice).
3. Click **Assign Connected Apps**.
4. Click **Edit** for **Assign Connected Apps**.
5. Add the **IBM FileNet Salesforce Connected** App to the list of Enabled Connected Apps.
6. Click **Save**.

Option 2: Allow access for a subset of Salesforce Organization users

If you only want a subset of your users to have access, then you must create a third Permission Set in addition to the two that were created automatically when the app was installed. Use the following steps to configure this access.

1. In the **Setup** menu, go to **Administration > Users**, and select **Permission Sets**.
2. Click **New** to create a new Permission Set
3. For **Label**, enter IBM FileNet Connected App
4. For **API Name**, enter IBM_FileNet_Connected_App
5. Leave the **Session Activation Required** checkbox unchecked.
6. Leave the **License** dropdown set to Salesforce
7. Click **Save** to save the Permission Set.

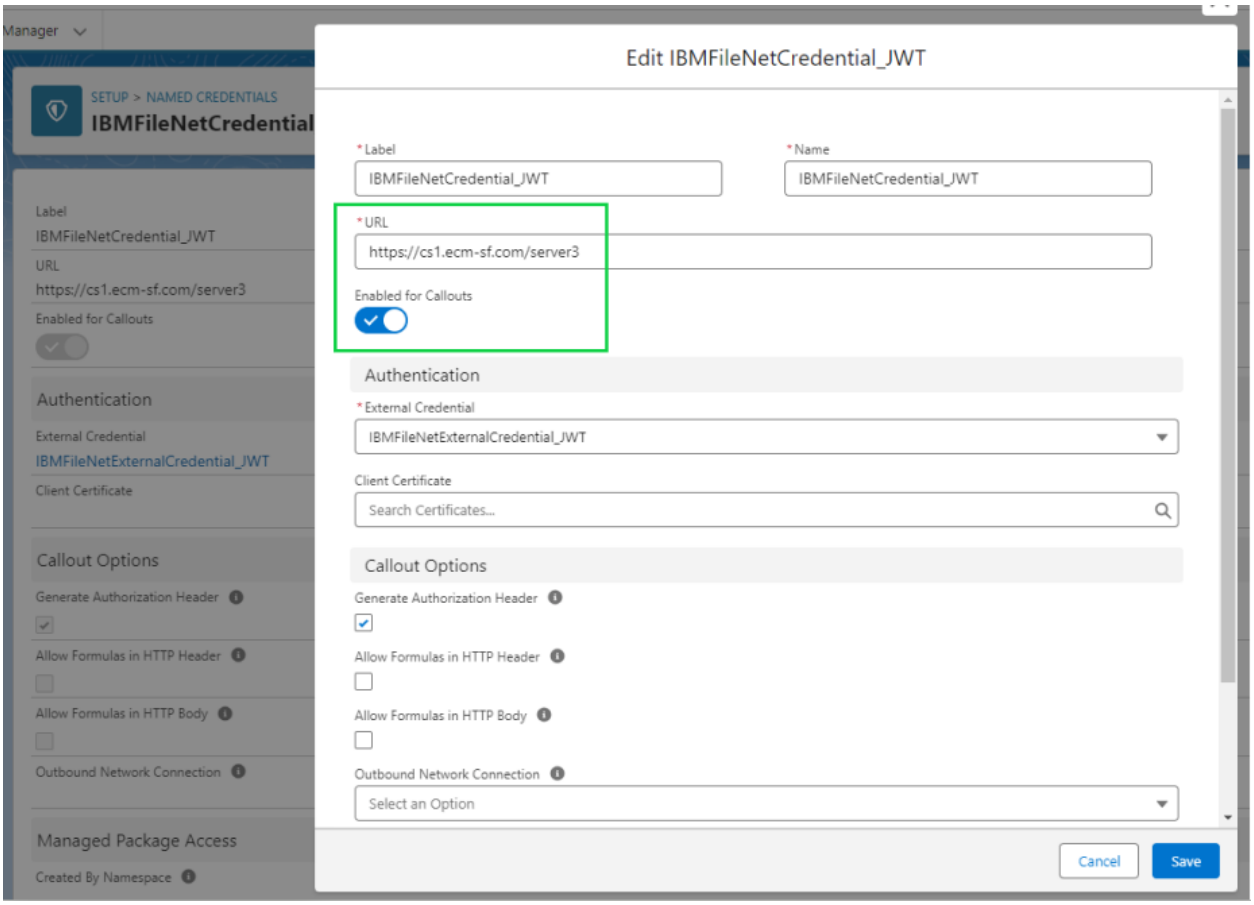
8. In the Apps section for the Permission Set that you just created, click on **Assigned Connected Apps**.
9. Click **Edit**.
10. In the **Installed Connected Apps** list, select the **IBM FileNet Salesforce Connected App** Named Credential and move it to the **Enabled Connected Apps** list.
11. Click **Save** to save your changes.
12. On the Permission set screen, click **Manage Assignments**.
13. Select the user or users that you want to add to the permission set (or unselect users who you would like to remove from the permission set), and click **Assign**.
14. Click **Save** to save your changes.

Updating a Named Credential

As discussed in an earlier section, the Salesforce Connector creates two placeholder Named Credentials during installation. In this section, we will update one of these, the OAuth Named Credential:

1. From **Setup**, enter Named in the Quick Find Box, then select **Named Credentials**.
2. In the Named Credential tab, Click **IBMFileNetCredential_JWT**, click Edit action on top right.
3. In the URL field, enter the URL of your IBM FileNet Content Services API server.
4. Enable the toggle checkbox "Enabled for Callouts".
   Note: Only one Named Credential can be enabled at a time, either **IBMFileNetCredential_JWT** or **IBMFileNetCredential_BasicAuth.**
5. Click Save.

The screenshot below shows an example of the Named Credential details:



Updating an External Credential

The external credential associated with a Named Credential needs to be configured before you can use the Named Credential. This external credential contains the configuration details needed by Salesforce to connect and authenticate to IBM FileNet Content Services API.

1. Click the **Named Credential** Link from the left side navigation menu or Click the **Named Credential** breadcrumb link.
2. Click **IBMFileNetExternalCredential_JWT** from Named Credential tab.
3. In the Issuer field, Paste the Consumer key from Creating a connected app.
4. In the Subject field, enter {!$User.Username}.
   The Subject refers to a Salesforce user. When Salesforce creates a JSON Web Token (JWT), the subject property in the JWT payload is set with the value from the “Subject” field. This field is mapped to UserNameAttribute of the Directory Configuration on Content Platform Engine.

If your Salesforce organization is configured to use SAML for federated authentication, then set the “Subject” field to {!$User.FederationIdentifier}.

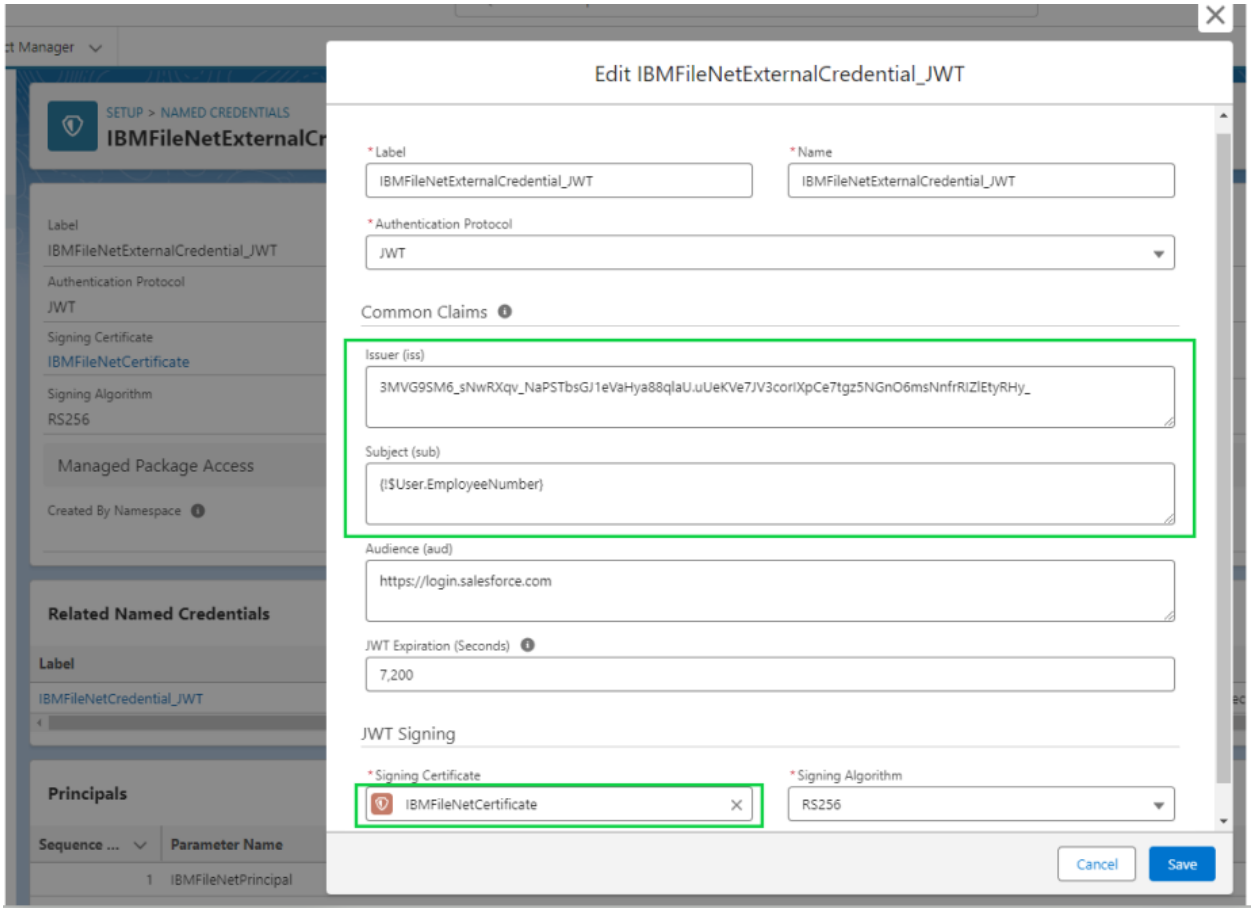
If the $User.Username does not match with the value of UserNameAttribute in the Directory Configuration on Content Platform Engine and if the $User.Email does match with the value of UserNameAttribute, then set the “Subject” field to {!$User.Email}.

If the $User.Username does not match with the value of UserNameAttribute in the Directory Configuration on Content Platform Engine and if the $User.EmployeeNumber does match with the value of UserNameAttribute, then set the “Subject” field to {!$User.EmployeeNumber}.

When neither $User.Username, $User.EmployeeNumber and $User.Email match with the value of UserNameAttribute in the Directory Configuration on Content Platform Engine, and if the $User.Alias does match with the value of UserNameAttribute, then set the “Subject” field to {!$User.Alias}.

5. The signing certificate should already be set to **IBMFileNetCertificate**. If it is not, please go ahead and select it.
6. Click Save.

The screenshot below shows an example External Credential configuration:



Enable User External Credentials

A Salesforce user performing a callout to IBM Content Services needs a profile or permission set-based access to external credentials. In this section we will create a new Permission set and enable External credentials access.

1. In the **Setup** menu, go to **Administration > Users**, and select **Permission Sets**.
2. Click **New** to create a new Permission Set
3. For **Label**, enter IBM FileNet Named Credential
4. For **API Name**, enter IBM_FileNet_Named_Credential
5. Leave the **Session Activation Required** checkbox unchecked.
6. Leave the **License** dropdown set to Salesforce
7. Click **Save** to save the Permission Set.
8. In the Apps section for the Permission Set that you just created, click on **External Credential Principal Access**.
9. Click **Edit**.

10. In the **Available External Credential Principals** list, select the **IBMFileNetCredential_JWT** and move it to the **Enabled External Credential Principals List**.
11. Click **Save** to save your changes.
12. On the Permission set screen, click **Manage Assignments**.
13. Select the user or users that you want to add to the permission set (or unselect users who you would like to remove from the permission set), and click **Assign**.
14. Click **Save** to save your changes.

If you are upgrading from an earlier release, then, after verifying that the new Named Credentials are working as expected, you can delete any legacy **IBMFilenetCredential**.

See the following Salesforce documentation topic for more details on OAuth Authorization Flows:

https://help.salesforce.com/articleView?id=remoteaccess_authenticate_overview.htm&type=5

Support for multiple Salesforce Organizations

Customers who have multiple Salesforce organizations can store documents from all of these organizations into object stores in one FNCM domain. In this configuration, each Salesforce organization appears as a different OAuth IdP. The Content Services GraphQL API must be configured to support each of these IdP's, and to distinguish which IdP an incoming request is authenticated against. The instructions for setting up this multiple IdP configuration differ depending on whether Content Services GraphQL API is hosted in a container, or in a traditional WebSphere environment.

The Salesforce Connector for FileNet sends an HTTP header which can be used to distinguish which Salesforce organization a request came from. This header is named "X-ECM-SF-ORG-ID", and it's value will contain the Salesforce organization Id.

Configuring OAuth authentication on the Content Services server deployed on WebSphere Liberty

The email address of the logged-on Salesforce user is used as part of the identity propagated to Content Services and the Content Platform Engine. To be able to access Content Services, the user's email address must exist as the shortName attribute in the Content Platform Engine LDAP server for the user.

Installing the Salesforce SSL certificate on Content Services server

You install the SSL certificate that you generated on the SalesForce server on the Content Services server to establish a secure connection. You downloaded this certificate as part of Generating a new self-signed certificate.

To install the Salesforce SSL certificate:

1. Copy the SSL certificate that you downloaded from Salesforce to the overrides folder on your Content Services server that contains your trust store. For example:
   `/opt/ibm/wlp/output/defaultServer/configDropins/overrides`.
   Create a new trust store and import the certificate to the Content Services server by using a command similar to the following example:

   ```
   keytool -import -file Saleforce_Certificate_for_IBM_Content_Services.crt -
   alias Saleforce_Certificate_for_IBM_Content_Services -keystore
   graphqlTrustStore.p12 -storetype pkcs12
   ```

2. Because keytool lowercases certificate alias names, verify what your certificate is called after importing by using a command similar to the following:
   `keytool -list -keystore graphqlTrustStore.p12 -storetype pkcs12`

   Typical output for this command might be like the following example:

   ```
   Keystore type: PKCS12

   Keystore provider: SunJSSE


   Your keystore contains 2 entries


   default, Oct 14, 2019, PrivateKeyEntry,
   ```

```
Certificate fingerprint (SHA1):
BB:BB:4E:2F:62:4D:93:EC:29:3B:D3:3A:D2:84:53:40:FE:AE:E8:BC

saleforce_certificate_for_ibm_content_services, Nov 18,
2019, trustedCertEntry,

Certificate fingerprint (SHA1):
3D:42:08:D3:51:1B:86:05:65:C2:C6:A3:C3:E9:73:C3:8D:A0:14:5
```

Creating the Open ID Connect client configuration file

You configure your Content Services server as an Open ID Connect client by adding an oidc.xml configuration file to the overrides directory for your Liberty server.

To create the OIDC client configuration file:

1. Create a file called oidc.xml, using the following example contents, and save it to the ${server-config-dir}/configDropins/overrides folder for your Liberty server:

```
<?xml version='1.0' encoding='UTF-8'?>

<server>

    <featureManager>

        <feature>openidConnectClient-1.0</feature>

        <feature>transportSecurity-1.0</feature>

    </featureManager>


    <openidConnectClient

       id="IbmContentServices"

issuerIdentifier="3MVG9_XwsqeYoueKsnUYSJ4OcVmjZvYlP6nOCuXA4ZfmRuD2toaZ121BKU.tqddt0V4l
7j.1YFkf9WC7i7BY8"

        trustStoreRef="graphqlTrustStore"

        trustAliasName="saleforce_certificate_for_ibm_content_services"

        realmName="localRealm"

        audiences="https://login.salesforce.com"

        inboundPropagation="required"

        httpsRequired="true"
```

```
        mapIdentityToRegistryUser="true"
        tokenReuse="true"
        isClientSideRedirectSupported="false"
        signatureAlgorithm="RS256"
        userIdentifier="sub"
        uniqueUserIdentifier="sub"
        userIdentityToCreateSubject="sub">
    </openidConnectClient>
    <keyStore id="graphqlTrustStore"
location="/opt/ibm/wlp/usr/servers/defaultServer/configDropins/overrides/graphqlTrustS
tore.p12" type="PKCS12" password="changeit" />


</server>
```

2. Update the following properties in the openidConnectClient stanza:

    - **Id**: Must match the last part of the Callback URL defined on Salesforce in Creating a connected app, for example, https://cpe-cmis-dev.ibmbrsandbox.com:9444/oidcclient/redirect/IbmContentServices
    - **issuerIdentifier**: The value for the Consumer Key that is defined on Salesforce in Creating a connected app.
    - **trustStoreRef**: Trust store in which you imported the Salesforce SSL certificate of your Connected App in Installing the Salesforce SSL certificate on Content Services server.
    - **trustAliasName**: SSL certificate alias of the Salesforce certificate you imported in Installing the Salesforce SSL certificate on Content Services server.
    - **realmName**: Name of the Content Platform Engine LDAP realm that contains Salesforce user email addresses.

3. Add the keyStore stanza for the graphqlTrustStore into which you imported the Salesforce certificate.

Supporting Multiple Salesforce Organizations

If you need to connect multiple Salesforce organizations to the same FileNet domain, then multiple providers must be specified in the oidc.xml file. You will need to create multiple openidConnectClient elements, and specify a unique authFilter for each, to specify an HTML header which WebSphere Liberty can use to distinguish which IdP a request came from. The authFilter will use the X-ECM-SF-ORG-ID header mentioned above. Here is an example:

```
<authFilter id="authFilter1">
    <requestHeader
        id="sfOrg1"
        name="x-ecm-sf-org-id"
        value="00D5Y000001NHaw"
        matchType="contains" />
</authFilter>
```

For more details on this, see the “Support Multiple OpenId Connect Providers” section in the Liberty OpenId Connect documentation here:

https://openliberty.io/docs/latest/reference/feature/openidConnectClient-1.0.html#filter

To find the Salesforce Organization ID:

1. From Setup, enter Information in quick Find box, then select Company Information

2. On the right page, lookup "Salesforce.com Organization ID" in the Organization Detail section.

Configuring OAuth authentication on the Content Services server deployed on traditional WebSphere Application Server (tWAS)

The email address of the logged-on Salesforce user is used as part of the identity propagated to Content Services and the Content Platform Engine. To be able to access Content Services, the user’s email address must exist as the shortName attribute in the Content Platform Engine LDAP server for the user.

To install and configure the Content Services GraphQL API on a traditional WebSphere Application Server (tWAS) use the following documentation.

https://www.ibm.com/support/pages/node/6459811

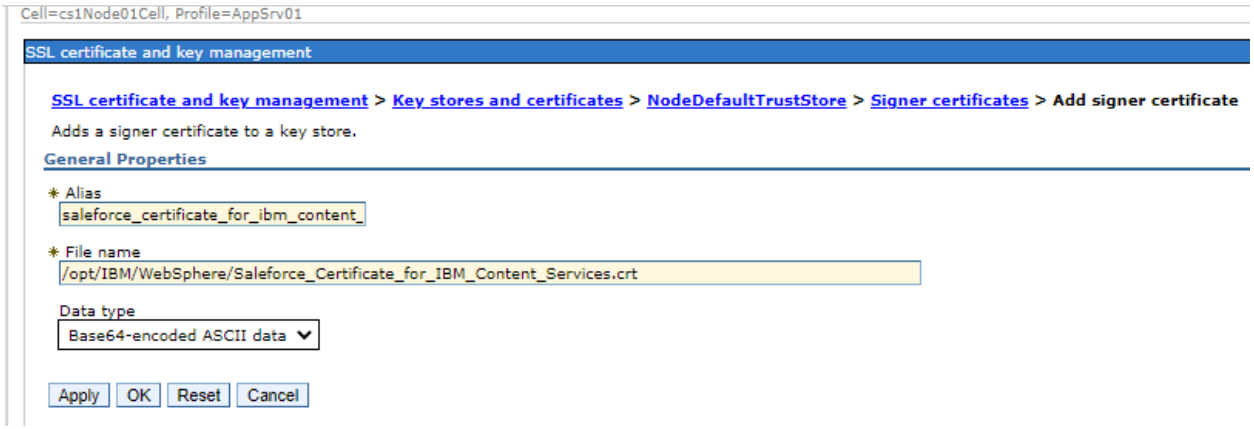
Important Note: Do not follow the Configure OAuth/OIDC section in above documentation.

Installing the Salesforce SSL certificate on Content Services server

You install the SSL certificate that you generated on the Salesforce server on the Content Services server to establish a secure connection. You downloaded this certificate as part of Generating a new self-signed certificate.

To install the Salesforce SSL certificate:

1. Copy the SSL certificate that you downloaded from Salesforce to the /opt/IBM/WebSphere folder.
2. Login to tWAS administration console, navigate to Security > SSL certificate and key management. In the Related items section, click Key stores and certificates.
3. Click NodeDefaultTrustStore, in Additional Properties section, click Signer Certificates and click Add.
4. Enter Alias as saleforce_certificate_for_ibm_content_services and File name as shown below:



5. Click OK button and click save change to the master configuration.

Install the WebSphere OpenID Connect Application

Open the command or shell prompt on the WebSphere Application Server and install the WebSphereOIDCRP.ear.

```
cd <app_server_root>/bin (e.g. /opt/IBM/WebSphere/AppServer/bin)
```

To install on a single server

```
./wsadmin.sh -profileName AppSrv01 -f installOIDCRP.py install no
deName serverName
```

To install on a cluster environment

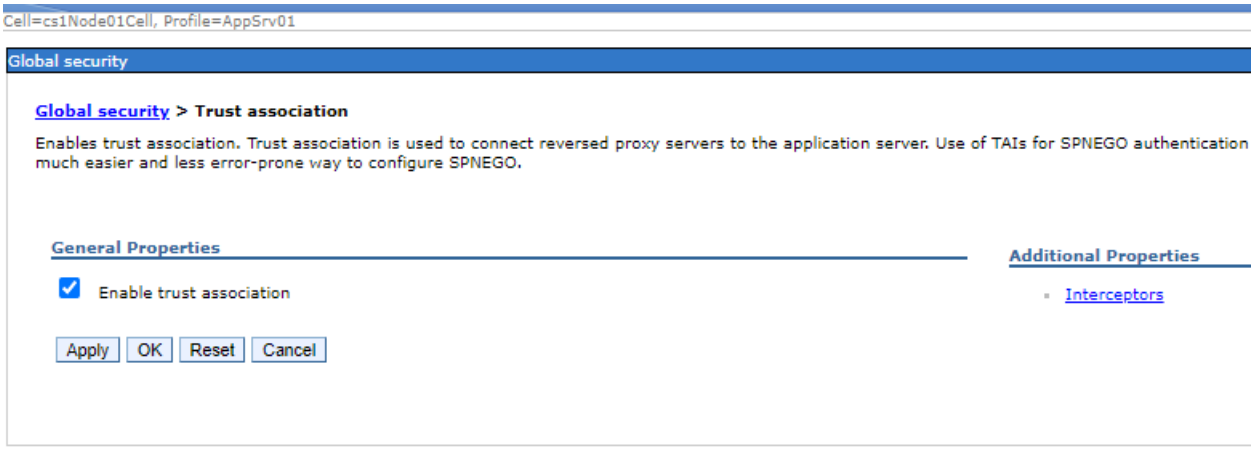
```
./wsadmin.sh -f installOIDCRP.py install clusterName
```

where:

- **AppSrv01** is the profile name of the target application server
- **nodeName** is the node name of the target application server
- **serverName** is the server name of the target application server
- **clusterName** is the name of the cluster on which OpenID Connect RP is to be installed

Configure OIDC Association Interceptor on Content Services

1. Navigate to Security > Global security > Authentication > Web and SIP security > Trust association
2. Select Enable trust association, click apply and save changes to master configuration.



3. Navigate to Security > Global security > Authentication > Web and SIP security > Trust association, click on Interceptor link, then click New and enter Class Name: **com.ibm.ws.security.oidc.client.RelyingParty**

In the custom properties, enter all the properties shown in following table:

Name	Value
provider_1.identifier	IbmContentServices
provider_1.signVerifyAlias	Saleforce_Certificate_for_IBM_Content_Services
provider_1.audiences	https://login.salesforce.com
provider_1.useRealm	<realm name from Global Security > Realm Name>
provider_1.interceptedPathFilter	/content-services-graphql.*
provider_1.userIdentifier	sub
provider_1.useJwtFromRequest	required
provider_1.uniqueUserIdentifier	sub
provider_1.httpsRequired	true
provider_1.mapIdentityToRegistryUser	true
provider_1.tokenReuse	true
provider_1.verifyIssuerInIat	true
provider_1.issuerIdentifier	<Consumer key from IBM FileNet Salesforce Connected App>
provider_1.headerName	Authorization

Refer to following link on details of these properties.
https://www.ibm.com/support/knowledgecenter/en/SSEQTP_9.0.5/com.ibm.websphere.base.doc/ae/csec_oidprop.html

Supporting Multiple Salesforce Organizations

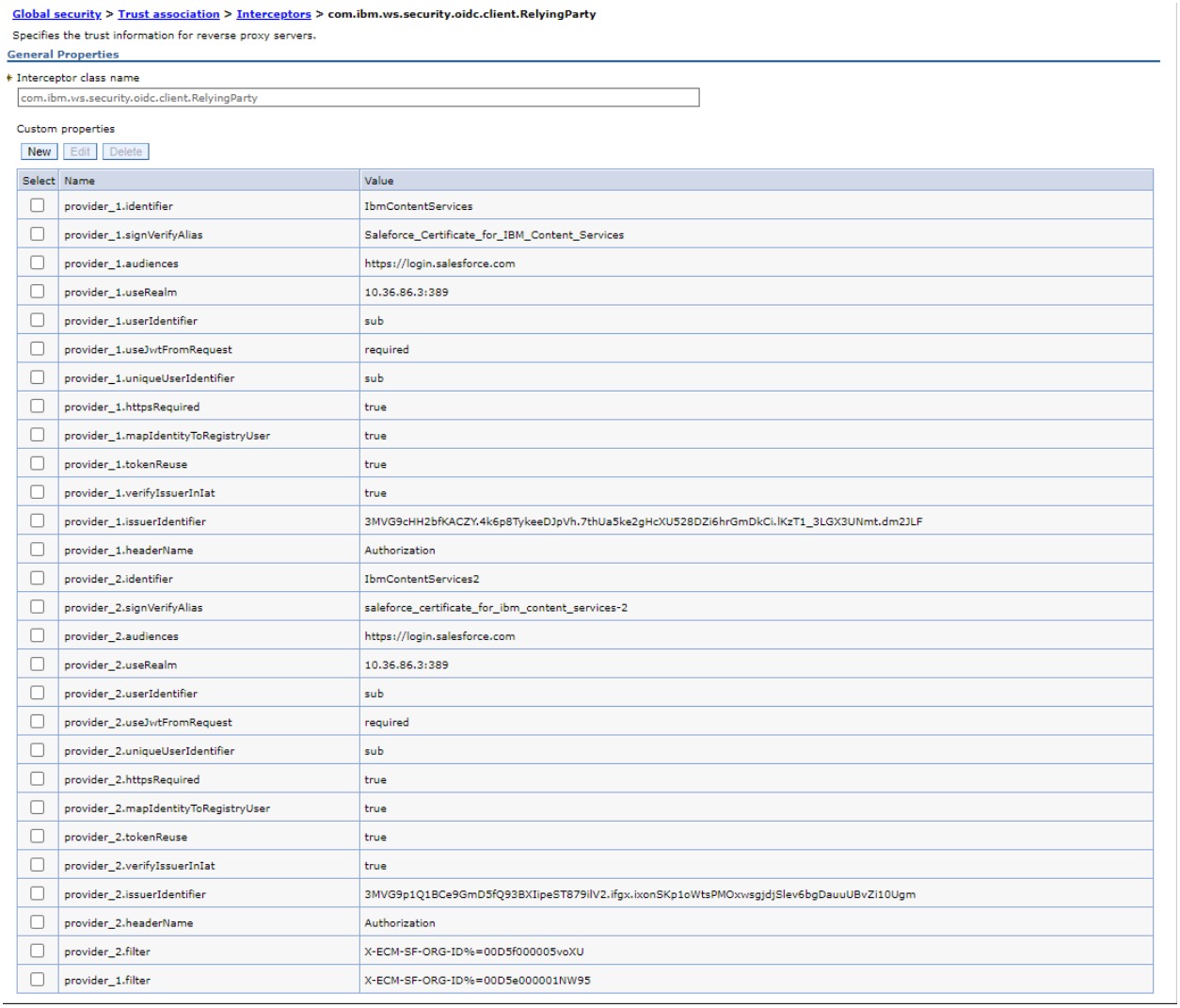
If support for multiple Salesforce organizations is needed, then each Salesforce organization will appear as a separate OAuth IdP, and therefore multiple OAuth providers must be configured. A provider_2 section would be created to support a 2nd OAuth provide, specifying the properties needed to connect to the 2nd IdP. In this case, WebSphere needs to use a filter condition to determine which provider to use for an incoming request. The Salesforce connector will send an HTTP header named X-ECM-SF-ORG-ID for this purpose, where the value of this header is the name of the Salesforce organization. Each provider must set the value of this filter. For example: provider_1.X-ECM-SF-ORG-ID has value <Salesforce org Id>

To find the Salesforce Organization ID:

1. From Setup, enter Information in quick Find box, then select Company Information

2. On the right page, lookup "Salesforce.com Organization ID" in the Organization Detail section.

Refer to following link on the details of TAI filter:
https://www.ibm.com/docs/en/was/9.0.5?topic=swss-saml-web-single-sign-sso-trust-association-interceptor-tai-custom-properties#rwbs_samltaiproperties__samltaifilterprop



Global security > Trust association > Interceptors > com.ibm.ws.security.oidc.client.RelyingParty

Specifies the trust information for reverse proxy servers.

General Properties

* Interceptor class name

com.ibm.ws.security.oidc.client.RelyingParty

Custom properties

New | Edit | Delete

Select	Name	Value
	provider_1.identifier	IbmContentServices
	provider_1.signVerifyAlias	Saleforce_Certificate_for_IBM_Content_Services
	provider_1.audiences	https://login.salesforce.com
	provider_1.useRealm	10.36.86.3:389
	provider_1.userIdentifier	sub
	provider_1.useJwtFromRequest	required
	provider_1.uniqueUserIdentifier	sub
	provider_1.httpsRequired	true
	provider_1.mapIdentityToRegistryUser	true
	provider_1.tokenReuse	true
	provider_1.verifyIssuerInIat	true
	provider_1.issuerIdentifier	3MVG9cHH2bfKACZY.4k6p8TykeeDJpVh.7thUa5ke2gHcXU528DZi6hrGmDkCi.lKzT1_3LGX3UNmt.dm2JLF
	provider_1.headerName	Authorization
	provider_2.identifier	IbmContentServices2
	provider_2.signVerifyAlias	saleforce_certificate_for_ibm_content_services-2
	provider_2.audiences	https://login.salesforce.com
	provider_2.useRealm	10.36.86.3:389
	provider_2.userIdentifier	sub
	provider_2.useJwtFromRequest	required
	provider_2.uniqueUserIdentifier	sub
	provider_2.httpsRequired	true
	provider_2.mapIdentityToRegistryUser	true
	provider_2.tokenReuse	true
	provider_2.verifyIssuerInIat	true
	provider_2.issuerIdentifier	3MVG9p1Q1BCe9GmD5fQ93BXIipeST879ilV2.ifgx.ixonSKp1oWtsPMOxwsgjdjSlev6bgDauuUBvZi10Ugm
	provider_2.headerName	Authorization
	provider_2.filter	X-ECM-SF-ORG-ID%=00D5f000005voXU
	provider_1.filter	X-ECM-SF-ORG-ID%=00D5e000001NW95

Click Apply and save changes to master configuration.

4. Navigate to Security > Global security > Custom Properties

Click the New... button and define the following custom properties.

- Name: **com.ibm.websphere.security.InvokeTAIbeforeSSO**
- Value: **com.ibm.ws.security.oidc.client.RelyingParty**

Note: if property exists, add this to existing value, separated by a comma to create a list

- Name: **com.ibm.websphere.security.performTAIForUnprotectedURI**
- Value: **true**

Add support for the SAMESITE Cookie attribute

1. Navigate to Servers > Server Types > WebSphere application servers > <server>
2. In Container settings, open Web Container setting > Web container transport chains
3. Click on HttpQueueInboundDefaultSecure, click HTTP inbound channel (HTTP 4), click custom properties in Additional Properties section.
4. Click New… and enter Name = sameSiteNone and Value = ECM-CS-XSRF-Token, click OK and save changes to master configuration as shown below:

JVM Arguments

Verify the values of the following JVM arguments on the WebSphere Application server:

-Decm.content.remote.cpeuri=<CPE_SERVER_WEB_SERVICE_ENDPOINT_URL>

-Dcom.filenet.authentication.wsi.AuthTokenOrder=ltpa,oauth

-Dcom.filenet.authentication.wsi.AutoDetectAuthToken=true

-Decm.content.graphql.xsrf.validate.disable=FALSE

-Decm.content.graphql.disable.xsrf.validation.for.ping=true

-Dcom.ibm.ecm.content.graphql.enable.graphiql=false

-Decm.content.graphql.cors.enable=true

-Decm.content.graphql.cors.origin.url= https://<Salesforce_Org_URL>

-Decm.content.graphql.cors.allow.methods=GET,POST,OPTIONS,PUT,DELETE,HEAD

-Decm.content.graphql.cors.allow.credentials.boolean=true

-Decm.content.graphql.cors.allow.headers=Connection,Authorization,Pragma,Cache-Control,Navigator-Client-Build,ECM-CS-XSRF-Token,XSRFtoken,Origin,User-Agent,Content-Type,Content-Length,Navigator-Client-

Identity,Accept-Control-Request-Method,Accept-Control-Request-Headers,Accept,Referer,Accept-Encoding,Accept-Language,DNT,Host,Content-Length,Cache-control,Cookie,Access-Control-Allow-Origin, X-ECM-SF-ORG-ID

-Decm.content.graphql.cors.expose.headers=Content-Disposition,Content-Length,ECM-CS-XSRF-Token,Content_Type,Content-Language,X-Powered-By,Date,Allow,Transfer-Encoding,$WSEP,DNT,Access-Control-Allow-Credentials,Access-Control-Allow-Headers,Access-Control-Allow-Max-Age,Access-Control-Allow-Methods,Access-Control-Allow-Origin,Access-Control-Expose-Headers,Connection,Cache-control,Cookie,x-content-download,X-ECM-SF-ORG-ID

-Decm.content.graphql.cors.max.age.seconds=86400

Configuring Basic Authentication

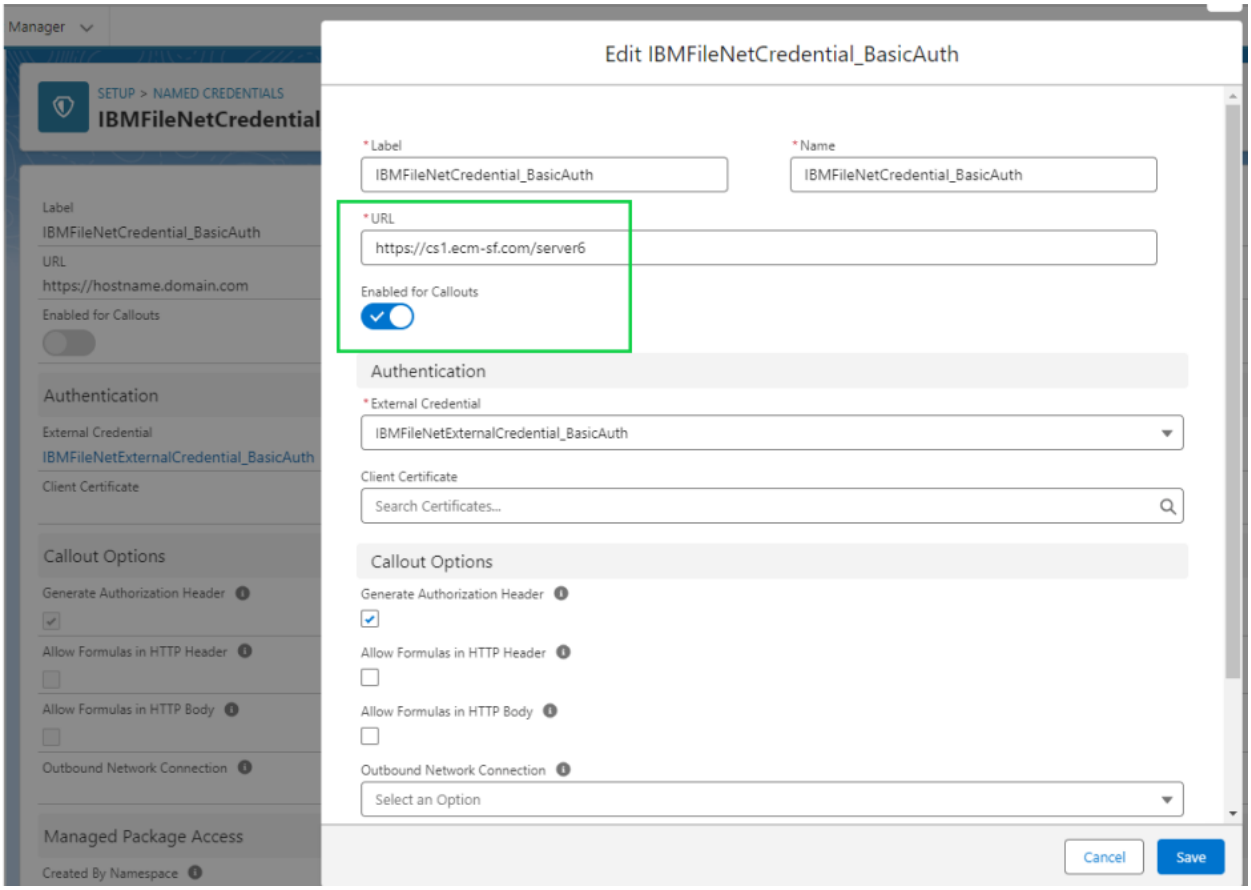
You can configure the IBM FileNet Connector for Salesforce to connect to the IBM Content Services GraphQL API using the BasicAuth protocol. This method is recommended only for development and test environments. For production environments, it is recommended to use the OAuth mechanism as described in the previous section.

Configuring a Salesforce Named Credential for the FileNet server

The IBM FileNet Salesforce Connector requires a Salesforce Named Credential to control authentication options for calls to the remote IBM Content Services GraphQL API. In this section, we will update the BasicAuth Named Credential which was created during installation of the Salesforce Connector app:

1. From **Setup**, enter Named in the Quick Find Box, then select **Named Credentials**.
2. In the Named Credential tab, Click **IBMFileNetCredential_BasicAuth**, click Edit action on top right.
3. In the URL field, enter the URL of your IBM FileNet Content Services API server.
4. Enable the toggle checkbox "Enabled for Callouts".
5. **Note:** Only one Named Credential can be enabled at a time, either **IBMFileNetCredential_JWT** or **IBMFileNetCredential_BasicAuth.**
6. Click Save.

The screenshot below shows an example of the Named Credential details:



Updating Named Principal

A **Named Principal** identity type applies the same credential for authenticating all users in the Salesforce org.

1. From **Setup**, enter Named in the Quick Find Box, then select **Named Credentials**.
2. In the Named Credential tab, Click **IBMFileNetExternalCredential_BasicAuth.**
3. In the Principals section, for **IBMFileNetPrincipal**, click action menu and select Edit.
4. Enter the username for the IBM FileNet domain in the Username field.
5. Enter the password for the IBM FileNet domain in the password field.
6. Click Save

A **Per User** identity type provides access control at the individual user level. Each user must configure their username and password credentials for the FileNet environment in their user profile, as described in a **Configuring IBM FileNet Salesforce Connector authentication – per user** section.

Configuring a Named Principal Identity is simpler and straightforward compared to a Per User Identity. You need to configure either the Named Principal or Per User Identity type for authenticating to IBM Content Services.

For further details on the Named Credentials, see the following Salesforce documentation topic:

https://help.salesforce.com/articleView?id=named_credentials_about.htm&type=5

Configuring Salesforce users who can use the External Credential to authenticate

When using BasicAuth, users must reference the **IBMFileNetCredential_BasicAuth** Named Credential directly from their profiles. This approach requires that they have access to the Named Credential. There are two options to complete this step, depending on whether you want all users to be able to use the connector, or only certain users.

Option 1: Allow access for all users in the Salesforce Organization (Recommended)

If you want all Salesforce users within an organization to have non-administrative access to the IBM FileNet Salesforce connector, add the Named Credential that was created in a previous step to the Standard Profile for the organization (or the user profile of your choice).

Update Standard Profile

1. In **Setup**, go to **Profiles**.
2. Select the **Standard User** profile (or the user profile of your choice).
3. Click **Enabled External Credential Principal Access URL**.
4. Click **Edit** for **Enable External Credential Principal Access**.
5. Add the **IBMFileNetCredential_BasicAuth** from available External Credential Principals to the list of Enabled External Credential Principals.
6. Click **Save**.

Update System Administrator Profile

1. In **Setup**, go to **Profiles**.
2. Select the **System Administrator** profile (or the user profile of your choice).
3. Click **Enabled External Credential Principal Access URL**.
4. Click **Edit** for **Enable External Credential Principal Access**.

5. Add the **IBMFileNetCredential_BasicAuth** from available External Credential Principals to the list of Enabled External Credential Principals.
6. Click **Save**.

Option 2: Allow access for a subset of Salesforce Organization users

If you only want a subset of your users to have access, then you must create a third Permission Set in addition to the two that were created automatically when the app was installed. Use the following steps to configure this access.

1. In the **Setup** menu, go to **Administration > Users**, and select **Permission Sets**.
2. Click **New** to create a new Permission Set
3. For **Label**, enter IBM FileNet Named Credential
4. For **API Name**, enter IBM_FileNet_Named_Credential
5. Leave the **Session Activation Required** checkbox unchecked.
6. Leave the **License** dropdown set to Salesforce
7. Click **Save** to save the Permission Set.
8. In the Apps section for the Permission Set that you just created, click on **External Credential Principal Access**.
9. Click **Edit**.
10. In the **Available External Credential Principals** list, select the **IBMFileNetCredential_BasicAuth** and move it to the **Enabled External Credential Principals List**.
11. Click **Save** to save your changes.
12. On the Permission set screen, click **Manage Assignments**.
13. Select the user or users that you want to add to the permission set (or unselect users who you would like to remove from the permission set), and click **Assign**.
14. Click **Save** to save your changes.

Configuring IBM FileNet Salesforce Connector authentication – per user

If the IBM Salesforce Connector app is configured to use Basic Auth, each user must **Allow Access** to authenticate to an external organization from to their individual profile and specify

their credentials for the FileNet environment in External Credentials. Any user who needs access to FileNet documents must perform the following steps.

Creating Per User Principal

A Per user authentication enables access control at the individual user level, which allows each user's access rights managed separately.

1. From **Setup**, enter Named in the Quick Find Box, then select **Named Credentials**.
2. In the Named Credential tab, Click **IBMFileNetExternalCredential_BasicAuth.**
3. In the Principals section, if there is an existing **IBMFileNetPrincipal entry**, click the action menu and select and click Delete.
4. Click the New button in the Principals section. In the Parameter Name field, enter **IBMFileNetPrincipal**.
5. Select the Per User Principal in the Identity Type field.
6. Click Save.

To configure authentication as an end user:

1. Navigate to your user profile in Salesforce.
2. Click Setting and click on **External Credentials**.
3. Click **Allow Access**.
4. Enter the username for the IBM FileNet domain in the Username field.
5. Enter the password for the IBM FileNet domain in the password field.
6. Click **Allow Access**.

Uninstalling the IBM FileNet Salesforce Connector app

You can remove the installed package. When a package is removed, all the components within that package are also removed.

Note: When a Salesforce standard object has a reference to the package component, for example, assigned users to a Permission Set, Salesforce prevents you from uninstalling the package. You must delete those references first before you uninstall the package.

To remove the app from your organization:

1. From the **Setup** menu, go to **Permission Sets**, and remove all users from the **IBM FileNet Users** permission set. Do the same for the **IBM FileNet Admins** permission set. These permission set users must be removed before the app can be uninstalled.
2. Go to the page layouts where the IBM Document List widget was previously added, and remove the IBM Document List lightning component from each of these pages. These document list components must also be removed before the app can be uninstalled.
3. From the **Setup** menu, go to **Installed Packages**. Select the **Uninstall** action for the IBM FileNet Salesforce Connector package.
4. Scroll to the bottom and check the checkbox to confirm that you want to uninstall the package and permanently delete its components, and click **Uninstall** to complete the action.
5. From the **Setup** menu, go to the **CSP Trusted Sites** page, and delete the site that was created for the IBM Content Services GraphQL endpoint.
6. From the **Setup** menu, go to the **CORS** page, and delete the whitelisted origin for the IBM Content Service GraphQL endpoint.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing IBM Corporation J74/G4 555 Bailey Avenue

San Jose, CA 95141 U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing Legal and Intellectual Property Law IBM Japan, Ltd. 19-21, Nihonbashi-Hakozakicho, Chuo-ku Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of

those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

```
IBM Corporation J46A/G4 555 Bailey Avenue San Jose, CA 95141-1003
U.S.A.
```

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Trademarks

IBM, the IBM logo and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries. Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates. Other product and service names might be trademarks of IBM or other companies.