How to Configure Customer Portal for Single Sign-On (SSO)

With Single Sign-On (SSO), a financial institution's customers use their existing log-in credentials to access Customer Portal.

Nov 21, 2018 . How to Configure

How To Configure

Welcome to Single Sign-On

Single Sign-On (SSO) with nCino's Customer Engagement Solution allows customers to access Customer Portal through their existing financial institution log-in credentials.

Financial institutions with SSO enabled can expect the following Customer Engagement Solution scenarios:

- Customer Portal users can use their already-existing online banking credentials to log on to Customer Portal. These users are navigated to the Customer Dashboard.
- A user with online banking credentials can follow a link to Customer Portal or to an online application from the financial institution's website—or an email—and access Customer Portal or a specific application using their existing credentials.
- Financial institutions can enforce duplicate detection, preventing unnecessary user records from being created in Salesforce.
- If the user logging on through SSO is an existing Portal user, the existing user (with existing license) can access Customer Portal.
- If a user is already logged in to online banking and follows a link to Customer Portal, the system does not prompt them to log in again.

Note: To ensure SSO for Customer Portal works properly, you must fully configure Customer Portal using "How to Configure Customer Portal: Community Set-Up (https://ncinocommunity.force.com/s/article/HowtoConfigureCustomerPortalCommunitySetUpPARTONE)" and the other required configuration documents linked from that article.

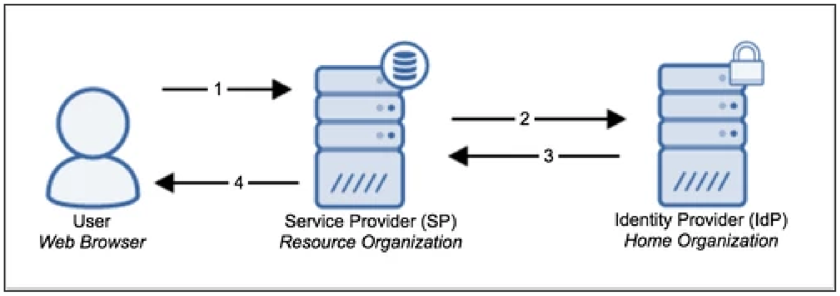
Contents

Financial Institution Requirements

To use SSO, the financial institution must provide sign-in and self-registration pages on their website. To navigate users to the Customer Portal after registering or logging in requires the custom development described in this document.

Note: When there are available nCino Platform licenses, the system automatically assigns a license to Portal users who log in through SSO. Document upload requires this license.

Glossary of Terms



The service provider is nCino's Customer Engagement Solution. The identity provider is the financial institution's federated identity system, in use for online banking.

- **Federated Identity:** A federated identity arrangement allows a user or subscriber to access multiple networks using the same identification information. The term "federated" emphasizes that there is a covenant between *independent* providers—such as between the service provider and identity provider—that ensures secure authentication if everyone follows it.
- **Service Provider (SP):** The SP is the client in the federated identity. The SP wants to provide a service but needs the users of that service to be externally authenticated.
- **Identity Provider (IdP):** The IdP is the server in the federated identity. The IdP provides authentication and ensures secure communication of that fact to an SP. (An example of an IdP is PingOne (https://www.pingidentity.com/en/products/pingone.html).)
- **Security Assertion Markup Language (SAML 2.0):** SAML is a standard (or covenant) for ensuring a user can be authenticated by an IdP for an SP.
- **Extensible Markup Language (XML):** XML sets the rules that encode documents in a human- and machine-readable format. In the case of SAML, the eXtension exists to ensure encrypted transmission of unique authentication parameters.
- **OAuth**: Another standard for ensuring user authentication, OAuth is usually used for social networks and the reverse flow of information.
- **Just In Time (JIT) Provisioning:** Through JIT Provisioning, an SP can use SSO to create user accounts "on the fly." These accounts are based on the information received from the IdP. For example, when nCino's Customer Portal installation is configured to work with a financial institution's IdP, existing users are imported at deployment time; JIT Provisioning allows for the creation or modification of new users in the IdP over time. To meet the needs of a particular implementation, the way Salesforce translates IdP information to Salesforce records (such as records on the User and Contact objects) is customizable with code.

IdP Requirements

SSO requires a financial institution to have a relationship with an Identity Provider, or IdP. This IdP, which must use SAML 2.0 authentication, is the single source of truth for customer sign-on information.

When my organization enables SSO, what happens to existing users?

Existing online banking users are added as Portal users only if the need to log in to Customer Portal arises. This prevents online banking users who do not need Portal access from consuming one of the financial institution's Portal user licenses (Customer Community Plus and Customer Community Plus Login are the accepted license types). Until a user self-registers for the Customer Engagement Solution, or follows a link to log in to an application, they will not exist in the system as a Portal user.

Can a new user register for Customer Portal through SSO?

Yes. While the configuration process may vary depending on your financial institution's IdP, the following list outlines the general user experience a new user can expect:

1. From the financial institution's website, the user navigates to the IdP's log-in page.

2. Because the user does not already have log-in credentials, they register for new credentials.
   Note: The user must provide the minimum credentials nCino requires—First Name, Last Name, Mobile Phone, and Email. During configuration, you must work with the IdP to ensure collection of these required credentials.
3. The IdP saves the user's data and registers the user. The section "Map Identifying Data from the IdP to the User Object" covers this process in more detail.
4. The system navigates the user to an nCino site, such as the customer dashboard in Customer Portal.
5. Anytime the user needs to log in to Customer Portal, they enter their SSO credentials; if IdP-provided information changes, the User fields in nCino update with the new information.

Is Brand Studio's multiple institution functionality compatible with SSO? NEW ARTICLE CONTENT

We do not currently support SSO with Brand Studio (multiple institutions). A user with an IdP account can access only one nCino community using their IdP account.

How Does the nCino System Know the User has a Password? NEW ARTICLE CONTENT

When a user logs in through SSO, the system automatically selects the User-Created Customer Engagement PW checkbox field on the user record.

Technical Requirements NEW ARTICLE CONTENT

Just In Time Handler

To enable SSO, your financial institution must use a configured Just In Time (JIT) handler. Through the SamlJitHandler interface, this handler is used to define the logic for user creation and user updates upon any log-in using SSO.

Note: The SamlJitHandler Interface is a set of function calls that determine how specific data is passed from the identity provider to the service provider. For more information about the interface and its usage and methods, refer to the Salesforce developer document "SamlJitHandler Interface (https://developer.salesforce.com/docs/atlas.en-us.apexcode.meta/apexcode/apex_interface_Auth_SamlJitHandler.htm)."

The SamlJitHandler interface executes through an executing user. As of the Fall 2018 release, nCino recommends you configure a service account (a user record not tied to a specific human user in the system) to act as the SSO executing user. This recommendation allows for record ownership continuity not tied to a particular existing administrator's user record. However, if the financial institution strongly prefers not to use a license on a service account, you can use an existing system administrator instead.

To create a service account user, complete the following:

- Create a user record named `SSO Service Account` or similar
- Set username and email to an email address or distribution list to which authorized system administrators have access
- Set profile to **System Administrator** (or an equivalent profile)
- Set license to Salesforce
- For role, you can reuse the Customer Portal Account Owner User Role (configured in "How to Configure Customer Portal: Community Set-Up (https://ncinocommunity.force.com/s/article/HowtoConfigureCustomerPortalCommunitySetUpPARTONE)") or create a new role for the purpose. If you create a new role, create a role at the top of its own branch in the existing role hierarchy. The role must not report to other roles or have other roles that report to it.

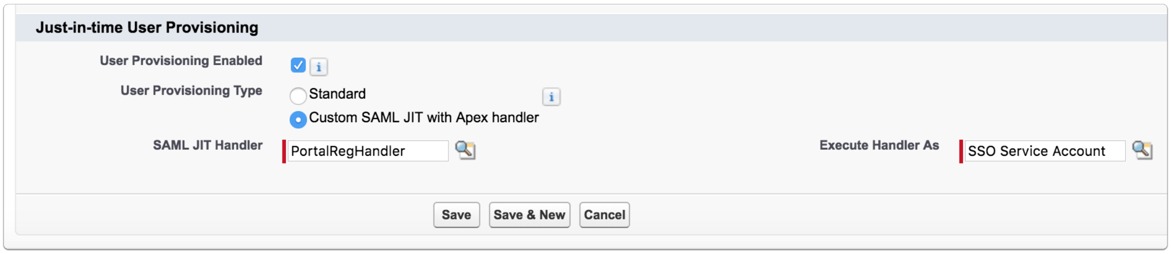
During configuration, select an Apex class to implement the SamlJitHandler interface. This class determines how inbound requests (such as a user attempting to access the Customer Portal) are handled. As a best practice, select the packaged PortalRegHandler class, which contains the custom logic to handle various registration functionality, including:

- nCino Platform license assignment
- duplicate detection
- profile assignment

- high data-volume user management
- Person Accounts support
- Financial Services Cloud (FSC) individual account model support

Follow these steps to select the class and executing user:

1. Go to **Setup > Single Sign-On Settings**.
2. Click **Edit** for the relevant setting.
3. Navigate to Just-in-time User Provisioning, and then populate the following fields:



Just-in-time User Provisioning

User Provisioning Enabled ☑

User Provisioning Type ○ Standard ● Custom SAML JIT with Apex handler

SAML JIT Handler PortalRegHandler

Execute Handler As SSO Service Account

- **User Provisioning Enabled**: Select the checkbox
- **User Provisioning Type**: Select Custom SAML JIT with Apex Handler
- **SAML JIT Handler**: Use the lookup to select PortalRegHandler (or an alternate class, per the instructions in the "Methods" section, though nCino recommends you stick with PortalRegHandler)
- **Execute Handler As**: Use the lookup to select the executing user you configured earlier in this section

4. Click **Save**.

Methods

PortalRegHandler is global and virtual. It uses two global virtual methods to handle incoming registrations from an IdP:

- **createUser**: This method handles net new IdP users without a federated identity already recognized in the nCino system. It creates a user in nCino using their specified federated identity. The system calls this method when a user logs on to the Customer Engagement Solution for the first time.
- **updateUser**: This method updates user information for users with a federated identity already stored in nCino. The system calls this method every time a user logs into the Customer Engagement Solution after their initial SSO log in.

Both methods use the following parameters:

- userId
- samlSsoProviderId
- communityId
- portalId
- federationId
- attributes
- assertion

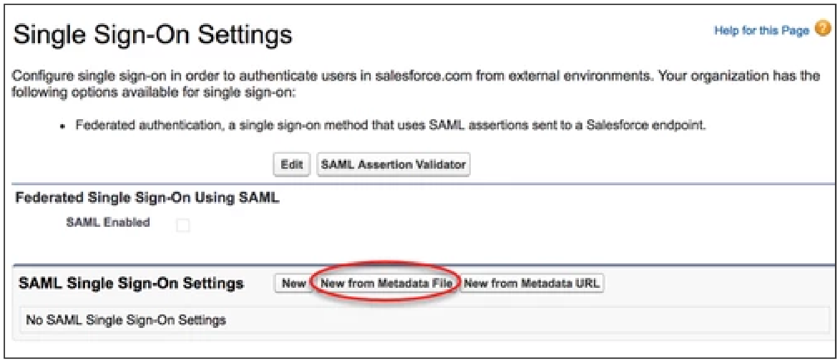
The Salesforce developer document "SamlJitHandler Interface (https://developer.salesforce.com/docs/atlas.en-us.apexcode.meta/apexcode/apex_interface_Auth_SamlJitHandler.htm)" defines the parameters. In addition to the information in that document, you must understand that attributes, also known as attestations, are custom-defined. The attributes you define aid in the duplicate detection process and in the process of verifying a potential user's identifying information. In nCino's Customer Engagement Solution, the minimum attributes are First Name, Last Name, Mobile, and Email. You can add additional attributes as needed, and attributes are case-sensitive.

If your financial institution needs additional functionality from the class used to implement the SamlJitHandler interface, contact your nCino representative for guidance. You can create a new Apex class to extend PortalRegHandler, implement Auth.SamlJitHandler, and override the createUser and updateUser methods with custom logic. Ensure the overrides perform all the productized functionality available in PortalRegHandler; to do this, invoke the parent class calls `super.createUser()` or `super.updateUser()` before or after the custom code.

Configure IdP and Customer Portal Community

The configuration steps vary based on the IdP the financial institution uses. At a high level, configuration may include:

- Creation of a new SAML application through the IdP
- Configuration of Single Sign-On Settings in the Salesforce org to:
  - create a new SAML Single Sign-On Setting record



Single Sign-On Settings

Configure single sign-on in order to authenticate users in salesforce.com from external environments. Your organization has the following options available for single sign-on:

- Federated authentication, a single sign-on method that uses SAML assertions sent to a Salesforce endpoint.

Federated Single Sign-On Using SAML

SAML Enabled □

SAML Single Sign-On Settings

No SAML Single Sign-On Settings

  - upload metadata downloaded from the SAML application
  - add a new Entity ID
  - ensure the Federation ID persists to the Federation ID field on the User record in nCino
- Generation of Login URLs that are specific to the IdP and the Salesforce Community
- Configuration of the IdP application to include the Community Login URL and Entity ID
- Mapping additional attributions as required by the financial institution (see the section "Duplicate Detection and Salesforce Record Creation" for more information)
- Configuring a registration page to allow new Portal users to register through SSO
- Configuring Community Login and Registration settings (see the section "Configure Login Settings" for more information)

Complete Salesforce Winter '18 Migration Steps

Due to changes in Salesforce Winter '18 that retire the use of the proxy.salesforce.com (http://proxy.salesforce.com) client certificate, you must switch to a self-managed client certificate. Read the Salesforce document "Retirement of Default Certificate affects SAML Single Sign On into Salesforce (https://help.salesforce.com/articleView?id=Retirement-of-Default-Certificate-affects-SAML-Single-Sign-On-into-Salesforce&language=en_US&type=1)" to determine the level of impact for your organization. Based on the applicable scenario described in the "How do I know if I am impacted?" and "What actions do I need to take?" sections of the document, follow the appropriate steps in the "Migration Steps" section.

Duplicate Detection and Salesforce Record Creation

For user record creation and duplicate detection to function properly in an org with SSO enabled, the IdP must issue claims for identifying data, which is then mapped in nCino. During implementation, the financial institution and IdP determine which identifying data serves as the user's federation identifier, or federation ID. The federation ID from the IdP is checked against the Federation ID and Username fields in nCino to determine whether a Customer Portal account already exists in nCino for the user; the system checks primarily against the federation ID, and the value in the Username field should match the user's email. When determining if duplicate records exists, the system prioritizes the federation ID but also checks against matching usernames. If either match, the system uses the existing user record in nCino rather than creating a new one.

In addition to the federation ID, the IdP must provide the following identifying data, which Salesforce requires to create a user record:

- First Name
- Last Name
- Mobile
- Email

When the user logs in to Customer Portal through SSO, the IdP passes the user's federation ID and other identifying data to nCino, and the following fields update:

Identifying Data from IdP	nCino Object	nCino Field
Federation ID	User	Federation ID
Email	User	Username, Email

Email	Account (or Relationship)	Email
Email	Contact	Email
First Name, Last Name	User	Name
First Name, Last Name	Account (or Relationship)	Name
First Name, Last Name	Contact	First Name, Last Name, Account Name

Notes:

- Since Salesforce usernames must be in the form of an email address, the user's email address is stored on both the Username and Email fields on the User object.
- If the user changes any of their identifying data with the IdP, all fields listed in the table update.
- The federation ID does not need to be in the form of an email address, though it *may* be in the form of an email address.
- As of the Fall 2018 release, if the IdP does not provide an existing user's mobile phone number but that phone number is already stored in nCino, the system no longer overwrites the stored phone number with a null value. The mobile phone number remains on the record.

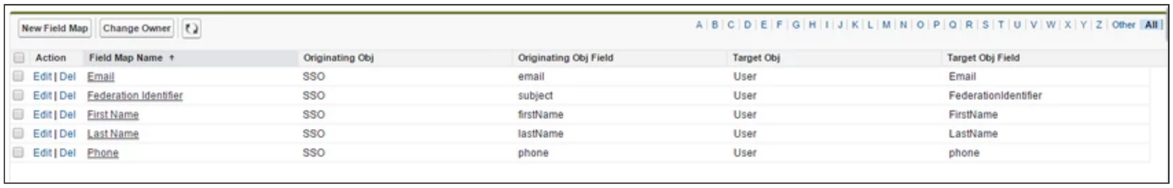
Map Identifying Data from the IdP to the User Object

nCino recommends you map the following data from the IdP to nCino:

- First Name
- Last Name
- Mobile
- Email

These four fields—and the federation ID—map to the User object by default. Additionally, you can configure field mapping to copy additional fields from the IdP to the User object, or override the out-of-the-box field mapping based on the financial institution's requirements.

For example, if your financial institution uses the IdP PingOne, you navigate to the Setup menu, then click SSO Settings, then click Attributes. You can manage the field mapping there. Next, go to the Field Maps tab in the Salesforce org and ensure that the mapping is correct. This screenshot shows an example of the minimum required field mapping.


New Field Map
Change Owner
A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | Other | All

□	Action	Field Map Name ↑	Originating Obj	Originating Obj Field	Target Obj	Target Obj Field
□	Edit | Del	Email	SSO	email	User	Email
□	Edit | Del	Federation Identifier	SSO	subject	User	FederationIdentifier
□	Edit | Del	First Name	SSO	firstName	User	FirstName
□	Edit | Del	Last Name	SSO	lastName	User	LastName
□	Edit | Del	Phone	SSO	phone	User	phone

Configure Login Settings

After you have completed SSO configuration through an IdP, you must enable Portal users to sign into the Customer Portal community using that IdP.

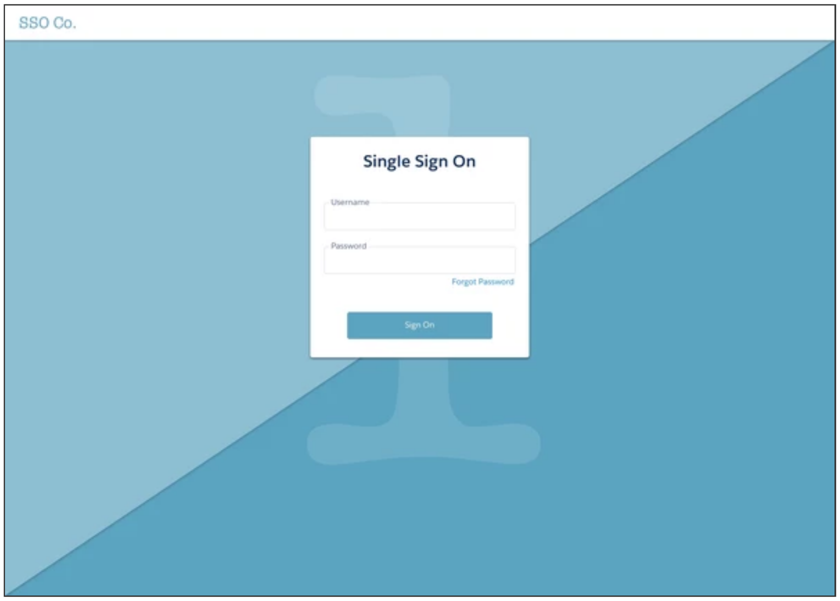
1. Go to **Setup > Feature Settings > All Communities**.
2. Click **Workspaces** next to the Customer Portal.
3. Go to **Administration > Login and Registration**.
4. Navigate to the Login section, and then to the text "Select which login options to display."
5. Select all appropriate checkboxes:
    - Select ***Financial Institution Name username and password*** to continue to allow Portal users to sign in without using SSO
    - Select the option for the configured IdP to allow users to be redirected to the IdP from the standard Customer Portal login page

Password Management

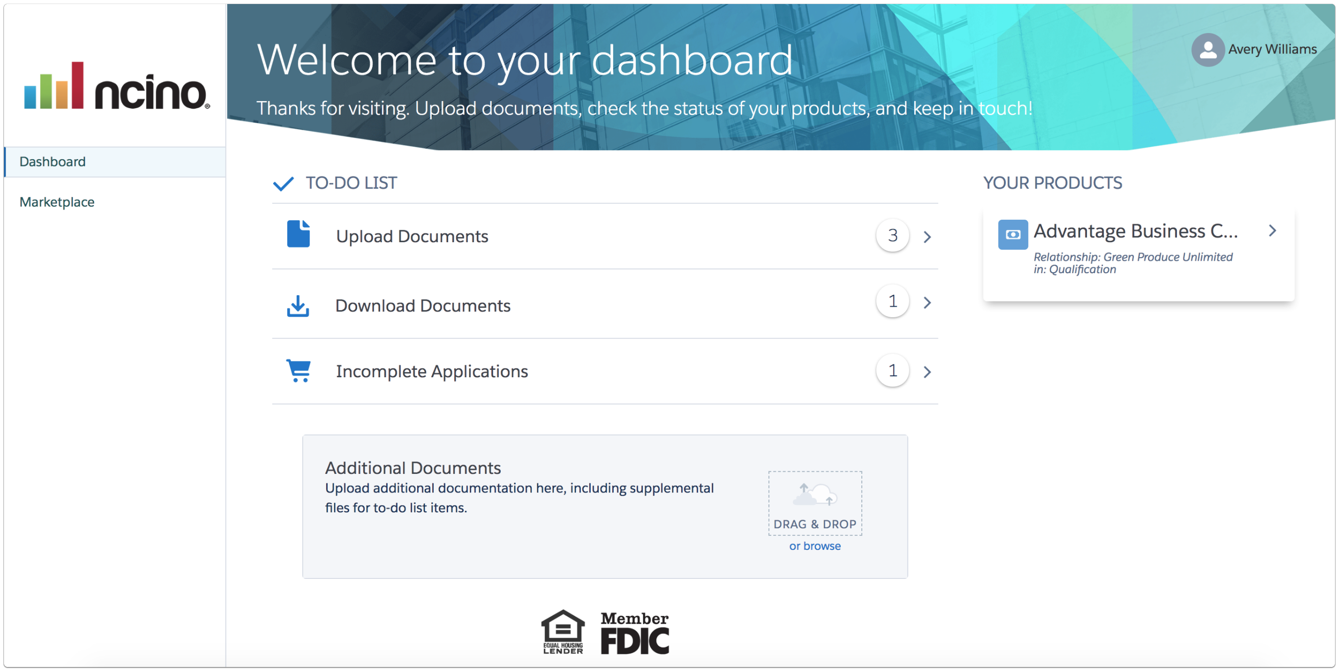
When a user logs in via SSO, the system automatically selects the checkbox field Customer Engagement Password Created on the user record. This prevents the Customer Engagement Solution from prompting the user to select or update their password, since the desired experience is for the SSO provider to handle password management.

Example User Experience

The look and feel of the registration page or SSO page configured for the financial institution varies; a user logs in via SSO through the same login page they use to access online banking or other services.



From there, the system takes the user directly to the customer dashboard:



Welcome to your dashboard

Thanks for visiting. Upload documents, check the status of your products, and keep in touch!

✓ TO-DO LIST

Upload Documents	3	›
Download Documents	1	›
Incomplete Applications	1	›

Additional Documents
Upload additional documentation here, including supplemental files for to-do list items.

DRAG & DROP
or browse

YOUR PRODUCTS

Advantage Business C... ›
Relationship: Green Produce Unlimited
in: Qualification

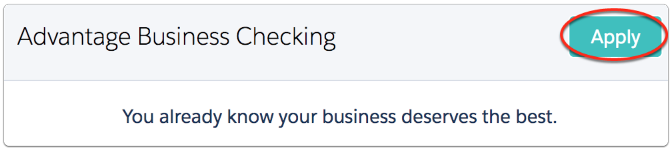
Configure SSO Log In to Specific Product Online Application

NEW ARTICLE CONTENT

Financial institutions that do not use SSO can configure self-registration URLs that include a Salesforce product ID; this creates a product-specific application experience from the moment the user registers for the first time.

To create the same experience for a financial institution that uses SSO, you must configure deep links. Deep links route a request to log in to a specific page (in this case, the online application for a certain product) through the SAML authentication process.

When you configure a deep link for an online application, the financial institution can use the deep link URL as a hyperlink or button link on their online application page. In this example, **Apply (circled in red)** directs the user to the deep link.



Advantage Business Checking

You already know your business deserves the best.

Online application deep links work for several different scenarios:

Scenario	Outcome
The user is not logged in to the IdP and clicks a deep link.	The system directs the user to the IdP log in page. Once they log in, the system redirects them to the online application referenced in the deep link.
The user is already logged in to the IdP and clicks a deep link.	The system immediately directs the user to the online application referenced in the deep link.
The user has not yet registered with the financial institution's single sign-on.	After the admin registers the user or the user self-registers with the IdP, they can log in to the Customer Engagement Solution. When they click the deep link and navigate to the Customer Engagement Solution for the first time, the system navigates them directly to the online application. Behind the scenes, the system sets up the user in the Customer Engagement Solution (including tasks such as license assignment and Federated ID generation).

Deep link configuration varies based on the IdP. This example shows the basic deep link configuration for PingOne, the IdP example used elsewhere in this documentation. Other IdPs may require similar or different URL configuration, but the basic PingOne deep link configuration combines the following:

- the log in URL for the IdP
- the Customer Portal community URL
- the product ID for the specific product online application you want to launch

This is the PingOne deep link URL pattern: `https://sso.connect.pingidentity.com/sso/sp/initsso?saasid={ (https://sso.connect.pingidentity.com/sso/sp/initsso?saasid={)`**`SaaS_GUID`**`}&`**`idpid`**`={IdP_GUID}&`**`appurl`**`={deep link}`.

SaaS_GUID refers to the ID of the SSO application configured for Customer Engagement. idpid refers to the institution's specific IdP instance. appurl refers to the URL for the requested resource (in this example, an online application page that includes a new parameter and a product ID URL parameter; these parameters indicate that the application is new and linked to a specific product).

Important: These steps show an *example* configuration. *Do not complete these steps for your financial institution*; instead, follow the IdP's exact deep link configuration policy, and use URLs and product IDs from the institution's IdP and Customer Engagement Solution.

Example deep link configuration:

1. Log in to the IdP (such as PingOne) as an administrator.
2. Copy the Initiate Single Sign-On URL and paste it into the location where you want to construct the deep link URL. For example: `https://sso.connect.pingidentity.com/sso/sp/initsso?saasid=2e098979-abcb-4785-bb3f-38496cfd0259&idpid=b307c4d9-4797-46a2-a260-594799d3d1ab`

(https://sso.connect.pingidentity.com/sso/sp/initsso?saasid=2e098979-abcb-4785-bb3f-38496cfd0259&idpid=b307c4d9-4797-46a2-a260-594799d3d1ab)

3. Add this text to the URL: `&appurl=`
4. In the Salesforce environment, go to **Setup > Feature Settings > All Communities**. Copy the URL of the Customer Portal community, and then add it to the deep link URL after &appurl=. For example: https://retail-ncinoportal-160986c2b5d-1619fc53e36.cs67.force.com (https://retail-ncinoportal-160986c2b5d-1619fc53e36.cs67.force.com)
5. Add the New and Product ID parameters to the URL: `&new=true&productID=`
6. In the Salesforce environment, navigate to the product record and copy the ID from the browser's URL area.
7. Add the Product ID after productID= in the deep link URL. For example: `a0b6A000000KmwL`
8. At this point, the link created using these examples looks like this: https://sso.connect.pingidentity.com/sso/sp/initsso?saasid=2e098979-abcb-4785-bb3f-38496cfd0259&idpid=b307c4d9-4797-46a2-a260-594799d3d1ab&appurl=https://retail-ncinoportal-160986c2b5d-1619fc53e36.cs67.force.com?new=true&productId=a0b6A000000KmwL (https://sso.connect.pingidentity.com/sso/sp/initsso?saasid=2e098979-abcb-4785-bb3f-38496cfd0259&idpid=b307c4d9-4797-46a2-a260-594799d3d1ab&appurl=https://retail-ncinoportal-160986c2b5d-1619fc53e36.cs67.force.com?new=true&productId=a0b6A000000KmwL)
9. Copy the URL after `&appurl=`. In other words, copy the part of the URL that begins with the Community URL.
10. Encode the copied portion of the URL by pasting it into a URL encoder. *This is required.* URLEncoder (https://www.urlencoder.org/) is an example encoder, but you should use the financial institution's preferred encoder.
11. When you have encoded the URL, paste the encoded URL after `&appurl=` in the deep link. At this point, the example deep link looks like this: https://sso.connect.pingidentity.com/sso/sp/initsso?saasid=2e098979-abcb-4785-bb3f-38496cfd0259&idpid=b307c4d9-4797-46a2-a260-594799d3d1ab&appurl=https%3A%2F%2Fretail-ncinoportal-160986c2b5d-1619fc53e36.cs67.force.com%3Fnew%3Dtrue%26productId%3Da0b6A000000KmwL (https://sso.connect.pingidentity.com/sso/sp/initsso?saasid=2e098979-abcb-4785-bb3f-38496cfd0259&idpid=b307c4d9-4797-46a2-a260-594799d3d1ab&appurl=https%3A%2F%2Fretail-ncinoportal-160986c2b5d-1619fc53e36.cs67.force.com%3Fnew%3Dtrue%26productId%3Da0b6A000000KmwL)
12. Use the deep link on the online banking website to direct users to the online application.

–