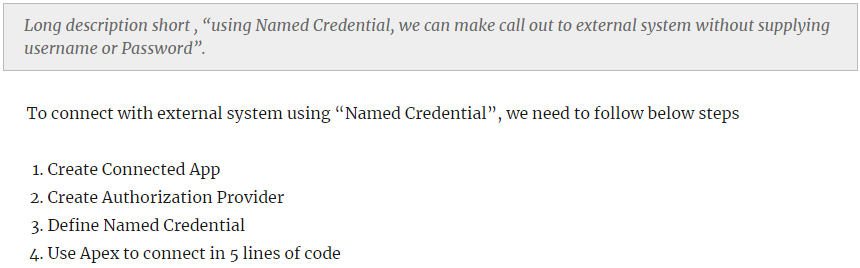
**Salesforce to Salesforce integration using Named Credentials in 5 lines**

**What is Named Credential ?**

A named credential specifies the URL of a callout endpoint and its required authentication parameters in one definition. You can simplify the setup of authenticated Apex callouts by specifying a named credential as the callout endpoint. You can instead specify a URL as the callout endpoint and register that URL in your organization’s remote site settings. In that case, however, you handle the authentication in your code. Doing so can be less secure and especially complicated for OAuth authentication.



**Login to Salesforce from Salesforce using Authentication Provider**

There are many ways to login to your Salesforce instance, using Google, Facebook, Linked, Twitter and even from other Salesforce Organization. I am sure many of readers has multiple Salesforce instances and its hard to remember password of each. We can connect every Salesforce instances and login using only one. In this post we will see, how we can login to one Salesforce from other using built in [Authentication provider](https://help.salesforce.com/apex/HTViewHelpDoc?id=sso_authentication_providers.htm) from Salesforce.

Throughout this article I will use term “service provider Salesforce instance” for Organization where I need to go after login and “Authentication Provider instance” which will authenticate user and will act as source organization for login.

[on Login to Salesforce from Salesforce using Authentication Provider](https://www.jitendrazaa.com/blog/salesforce/login-to-salesforce-from-salesforce-using-authentication-provider/#comments)

There are many ways to login to your Salesforce instance, using Google, Facebook, Linked, Twitter and even from other Salesforce Organization. I am sure many of readers has multiple Salesforce instances and its hard to remember password of each. We can connect every Salesforce instances and login using only one. In this post we will see, how we can login to one Salesforce from other using built in [Authentication provider](https://help.salesforce.com/apex/HTViewHelpDoc?id=sso_authentication_providers.htm) from Salesforce.

Throughout this article I will use term “service provider *Salesforce instance*” for Organization where I need to go after login and “*Authentication Provider instance*” which will authenticate user and will act as source organization for login.

**Enable MyDomain**

First step to start with Authentication Provider is to setup [my domain](https://help.salesforce.com/apex/HTViewHelpDoc?id=domain_name_overview.htm) in your “*service provider Salesforce instance*“. This step is important so that it will display all available Authentication provider for that Salesforce instance.

**Create Connected app**

If you want to login from Facebook, LinkedIn or any other web application, you need to inform Salesforce that those applications are legitimate and this is very important piece of OAuth2. One of major difference between OAuth1 and OAuth2 is that OAuth2 provides scope where you can set what specific permission this Connected App will need.

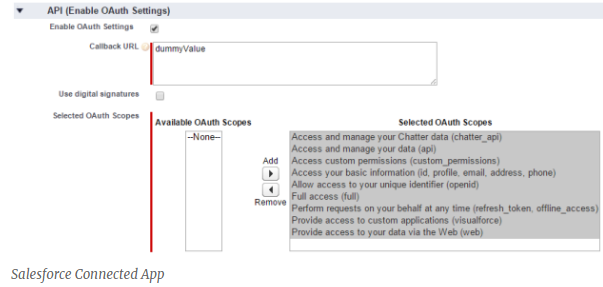
Connected App also has “Consumer Key” and “Consumer Secret” which is equivalent to “username” and “password” for that App.

Other important setting, connected app has “Callback URL“. This is the URL where “Authentication Provider instance” should return after providing access. Even if somehow “Consumer Key” and “Consumer Secret” is compromised, it will return to Callback URL which is your application.

In this post we need “service provider Salesforce instance” to be logged in from “Authentication Provider instance“. So “Authentication Provider instance” should be able to identify that request is coming from “service provider Salesforce instance“. **Therefore Connected App needs to be created in “service provider Salesforce instance” (SP).**

To create Connected App in “service provider Salesforce instance“, Navigate to “Setup | Build | Create | Apps | Connected Apps” and click on New. Provide All information except “Callback URL”. We will comeback again on this step later to provide Callback URL.

****

****

Once you save this Connected app, it will provide “Consumer Key” and “Consumer Secret”, we need this information in next step.

**Create Authorization Provider**

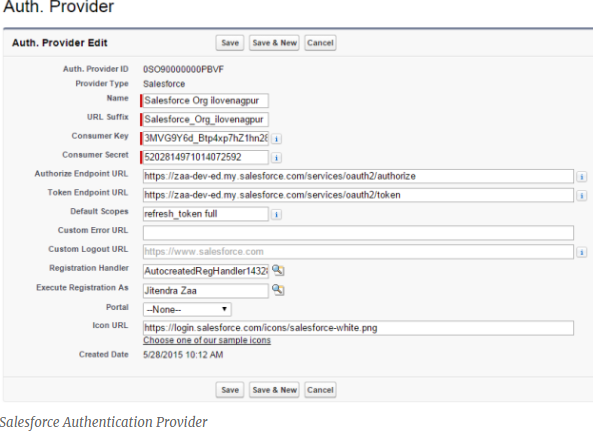
**Its time to create Authorization Provider in “Authentication provider Salesforce instance (IDP)”.**

Navigate to “Setup | Administer | Security Controls | Auth. Providers | Create New”.

Select “Salesforce” as provider Type.

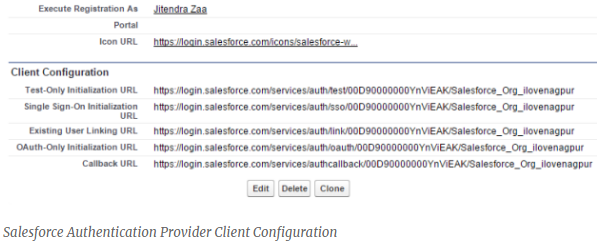
We need to provide “Consumer Key” and “Consumer Secret” created in previous step. Also one important setting is “[Default Scope](https://help.salesforce.com/HTViewHelpDoc?id=remoteaccess_oauth_scopes.htm&language=en_US)“, it should have value as “refresh\_token full”. “refresh\_token” and “full” should be separated by space. Authorize Endpoint URL should be something like “https://AuthenticationProviderinstance/services/oauth2/authorize” and Token Endpoint URL “https://AuthenticationProviderinstance/services/oauth2/token“.

Click on “Automatically create a registration handler template”, it will generate one apex class, in my case auto generated apex class name is “AutocreatedRegHandler1432826053915”. Then Select User who should be used to execute this Apex class when user tries to login.

****

**Set Callback URL in Connected App**

Once you save “Auth. Provider” in previous step, it will provide you list of URL as shown in below image. Copy Callback URL and edit Connected App we created in **service provider Salesforce instance** and set this URL.

****

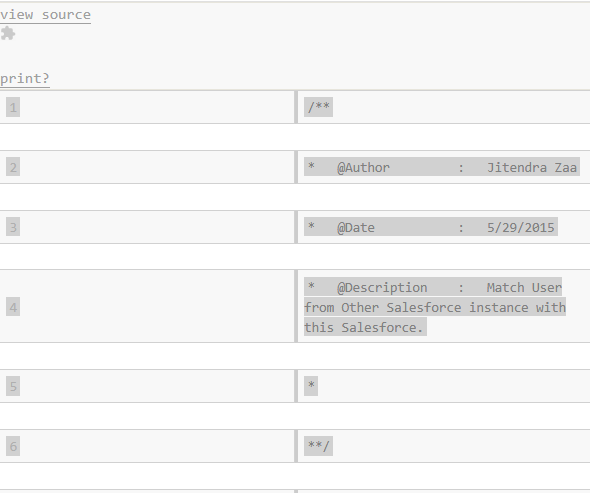
you can even test your application using “Test-Only initialization URL”, however in our case we need to modify our Apex class, so need to wait.

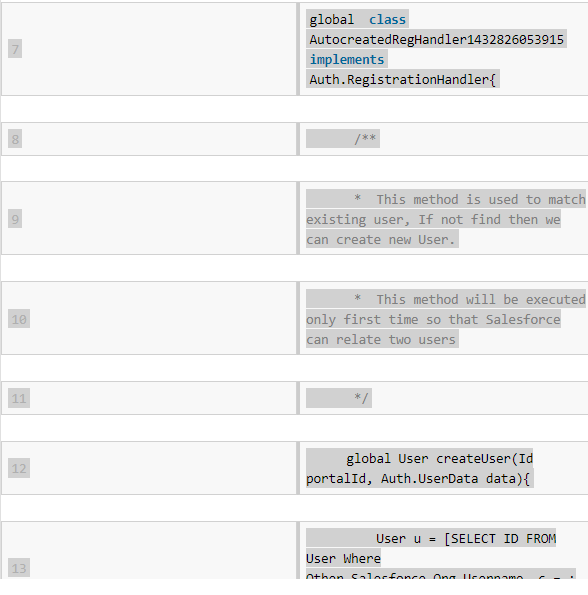
**Create field in User Object**

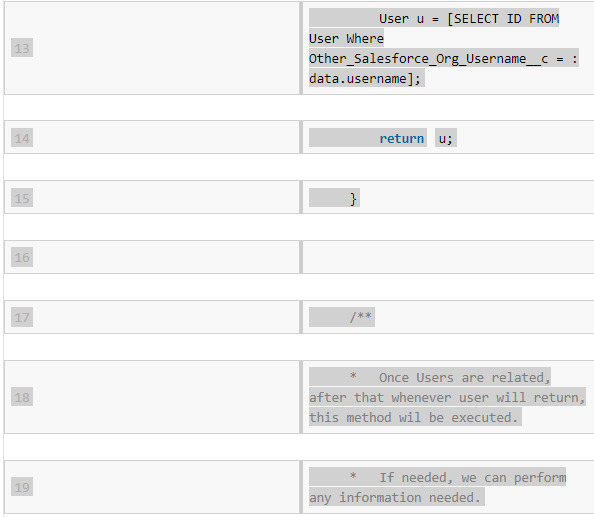
We are almost there, but we have one problem to address. When someone tries to login from “Authentication Provider instance”, how we will match which user from “Authentication Provider instance” matched with user in “service provider Salesforce instance” ? We will need to create one field on User object which will save username of other Salesforce instance. In this case, I have created field “Other Salesforce Org Username”. Once field is created, populate every user record so that we should be able to match them.

**Update Auto generated Registration Handler Apex class**

Replace content of Apex class by below code, in our case class name is “AutocreatedRegHandler1432826053915”.

****

****

****

****

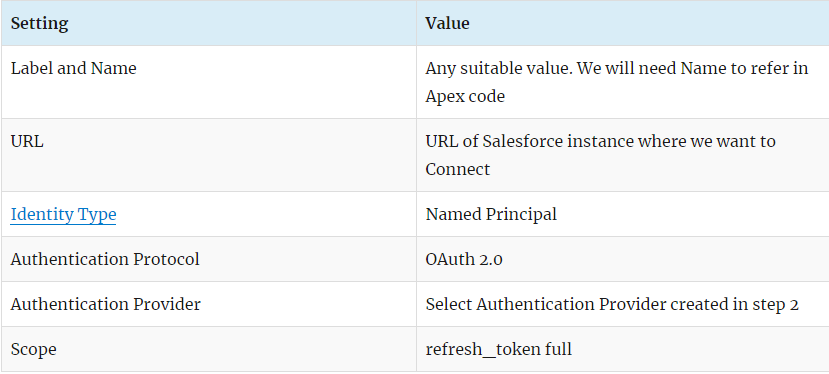
**Add Salesforce button on Login Page**

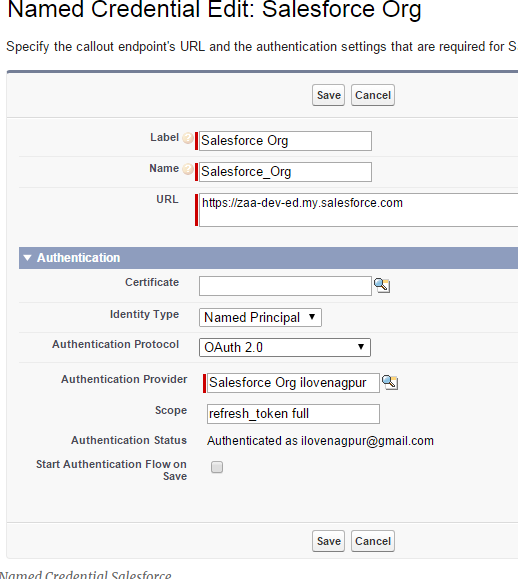
Navigate to “Setup | Administer | Domain Management | My Domain | Authentication Configuration | Edit”.

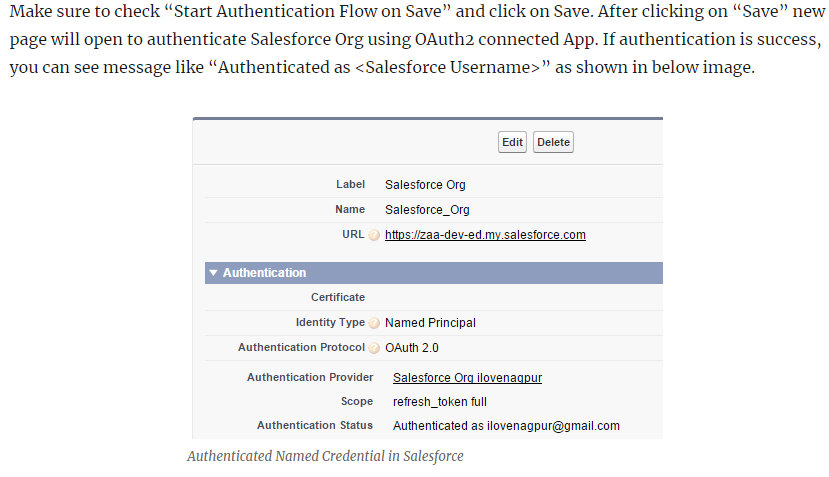
**Creating Named Credential**

Now, directly start with step 3 to create Named credential by navigating to “Setup | Administer | Security Controls | Named Credentials | New Named Credential “.

Provide below information in settings :







Now execute below code in anonymous Apex and you should be able to see list of all REST API available in Salesforce instance.

