Contents

[Salesƒorce Integration with SDL WEB8 2](#_Toc468617131)

[Salesƒorce-CRM 2](#_Toc468617132)

[SDL WEB 8-CMS 2](#_Toc468617133)

[Steps to get the SF API and how to create .NET based client 2](#_Toc468617134)

[Create dummy data in SF 3](#_Toc468617135)

[Read Data using .NET Client and SF API 4](#_Toc468617136)

[Write Data in SF using .Net Client 4](#_Toc468617137)

[Let’s integrate SF with SDL WEB 8 5](#_Toc468617138)

[Summary 8](#_Toc468617139)

[Issues faced while integrating SF API with .NET client 8](#_Toc468617140)

[Error 1:- 8](#_Toc468617141)

[Solution: 8](#_Toc468617142)

[Error 2:- 9](#_Toc468617143)

[Solution 9](#_Toc468617144)

[Error 3:- 9](#_Toc468617145)

# Salesƒorce Integration with SDL WEB8

## Salesƒorce-CRM

A CRM system is a business tool that allows you to manage all your customers, partners, and prospects information all in one place. The Sales Cloud (Salesforce.com’s CRM system) is a cloud based CRM system.

For example, it helps:

1. sales teams close deals faster
2. marketing manage campaigns and track lead generation
3. service call centres reduce the time to resolve customer complaints

## SDL WEB 8-CMS

SDL Web covers four main core functional areas:

1. Web Content Management
2. Experience Optimization (including personalization)
3. Digital Media
4. Localization

## Steps to get the SF API and how to create .NET based client

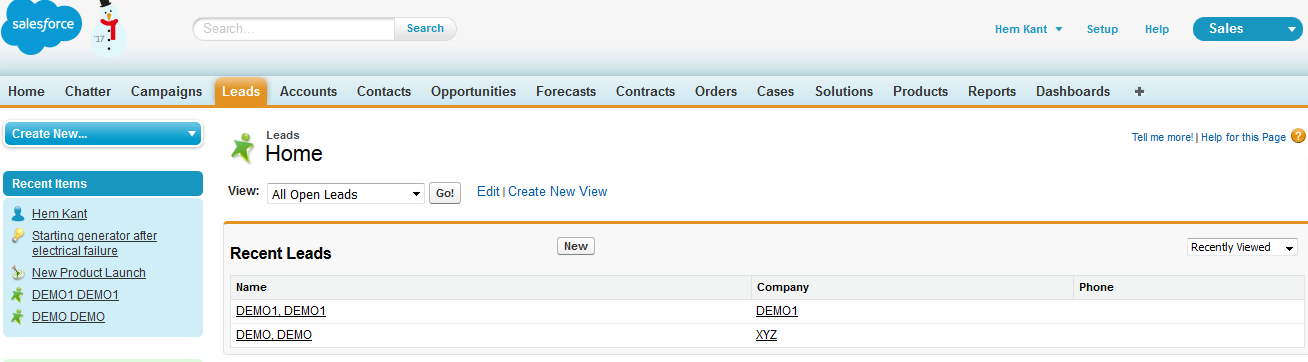
1. Go to ap2.salesforce.com and create your developer account.
2. Click on Setup and from quick find search for API.
3. Select the API, Salesforce’s WSDL allows you to easily integrate salesforce.com with your applications, and to build new applications that work with salesforce.com. To get started, download a WSDL file to a place accessible to your development environment. For complete documentation, sample code, and developer community, visit
4. <http://developer.salesforce.com/>
5. I used Enterprise WSDL
   1. A strongly typed WSDL for customers who want to build an integration with their salesforce.com organization only.



SF Enterprise WSDL API

## Create dummy data in SF

I have created some dummy data in Lead object which we will accessing using Enterprise WSDL API

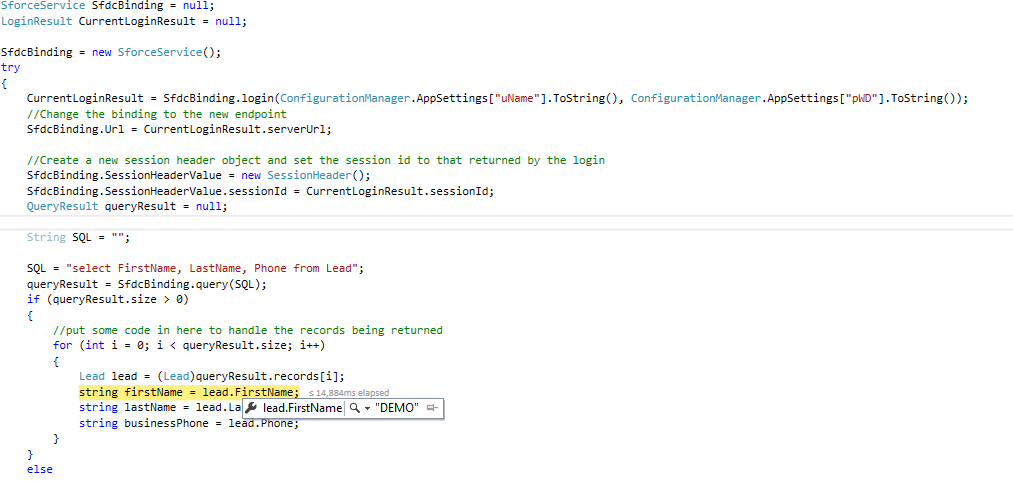


SF Interface dummy data created

## Read Data using .NET Client and SF API

Add SF Enterprise API as service reference in your project

1. Login to the Force.com API
2. Change the binding to the new endpoint
3. Create a new session header object and set the session id to that returned by the login
4. Queries are executed against the Force.com API using SOQL (Salesforce.com Object Query Language).



In the screenshot, we can see that the data created in SF is accessible using .Net client

## Write Data in SF using .Net Client

1. Instantiate the Object Being Created: Before we can execute a create() against the API, we must establish the type of object being created. In this case, we will create a new lead.



1. Establish Field Values and Object Properties



1. Execute Create Call and Capture Save Results



1. Working with the Results: The create call will obviously result in success or failure. Successful create calls will return the ID value of the resultant record created. Fortunately, the API is built to allow us to handle both scenarios within our application.
2. Data is successfully created



1. Let’s go and check in SF online. Here you can see the details which we have supplied via .NET client is saved in SF.



## Let’s integrate SF with SDL WEB 8

* 1. We already have SF API and before we start the integration of SF with SDL WEB 8 we need to first create WEB 8 Core service client to get the object.
  2. Core Service is WCF based webService.
  3. I have already created core service client using ISessionAwareCoreService.



Creating Core Service client

* 1. After this we need to read Component data using.



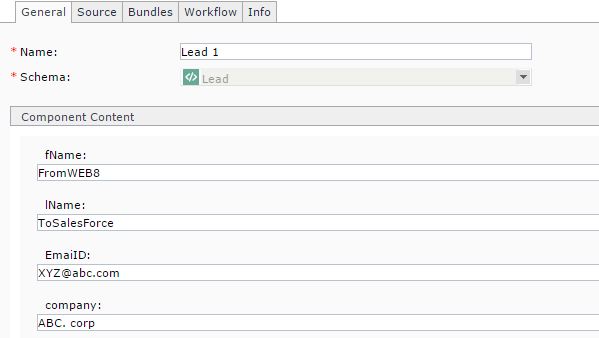
* 1. Deserialize the data in to the object, and send to SF API to create the records in SF.



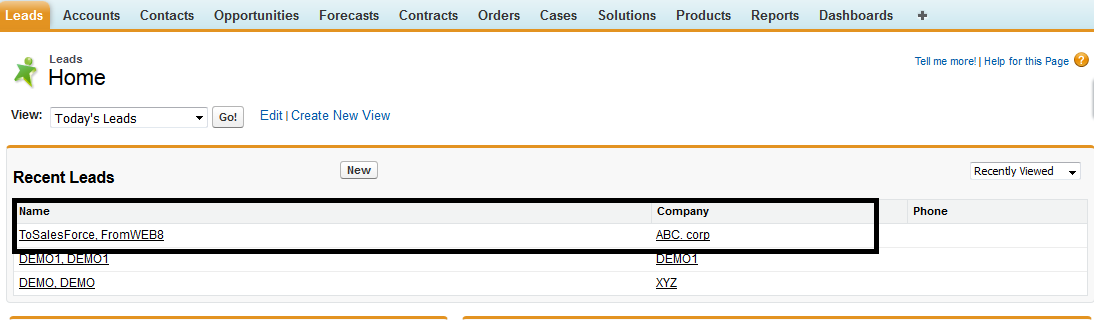
* 1. Know we have the components List from WEB 8 , run SF API create method to create new records in SF.



* 1. I have created sample component in WEB 8 which is pushed in to SF
     1. Below image is CMS component which is pushed to SF.

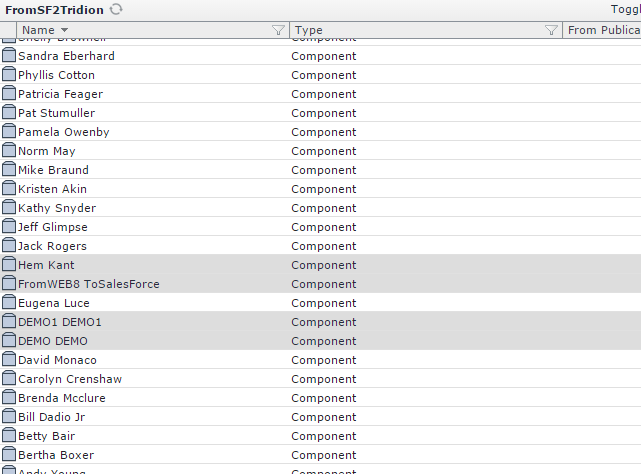


* + 1. Salesforce UI here we can see the data which was pushed from WEB 8 is successfully created in SF.



SF Interface

So, we are successfully able to push content from WEB 8 to SF lets work on the other way around so that we can say SF and WEB 8 are in sync.

1. To create content In WEB 8
   1. Need core service client I have re-used the same core service utility.
   2. Folder Location where you want to create content.
   3. Read data from SF you can check [how to read content from SF](#_Read_Data_using) .
   4. Transform the object into XML.
   5. We need to call.
2. I have imported all the SF Lead data in to Tridion. 

Tridion Interface

1. Data in SF which we imported in to Tridion. 

SF interface



## Summary

Run this service in background to keep Salesforce and Tridion in a sync also if you want to have, create custom SF objects, and add custom fields you can also do that, in this demo I have used the default object LEAD provided by SF.

Prerequisites to create custom objects in SF:

A Salesforce account in a sandbox Professional, Enterprise, Performance, or Unlimited Edition org, or an account in a Developer org.

## Issues faced while integrating SF API with .NET client

### Error 1: -

TLS 1.0 has been disabled in this organization. Please use TLS 1.1 or higher when connecting to Salesforce using https

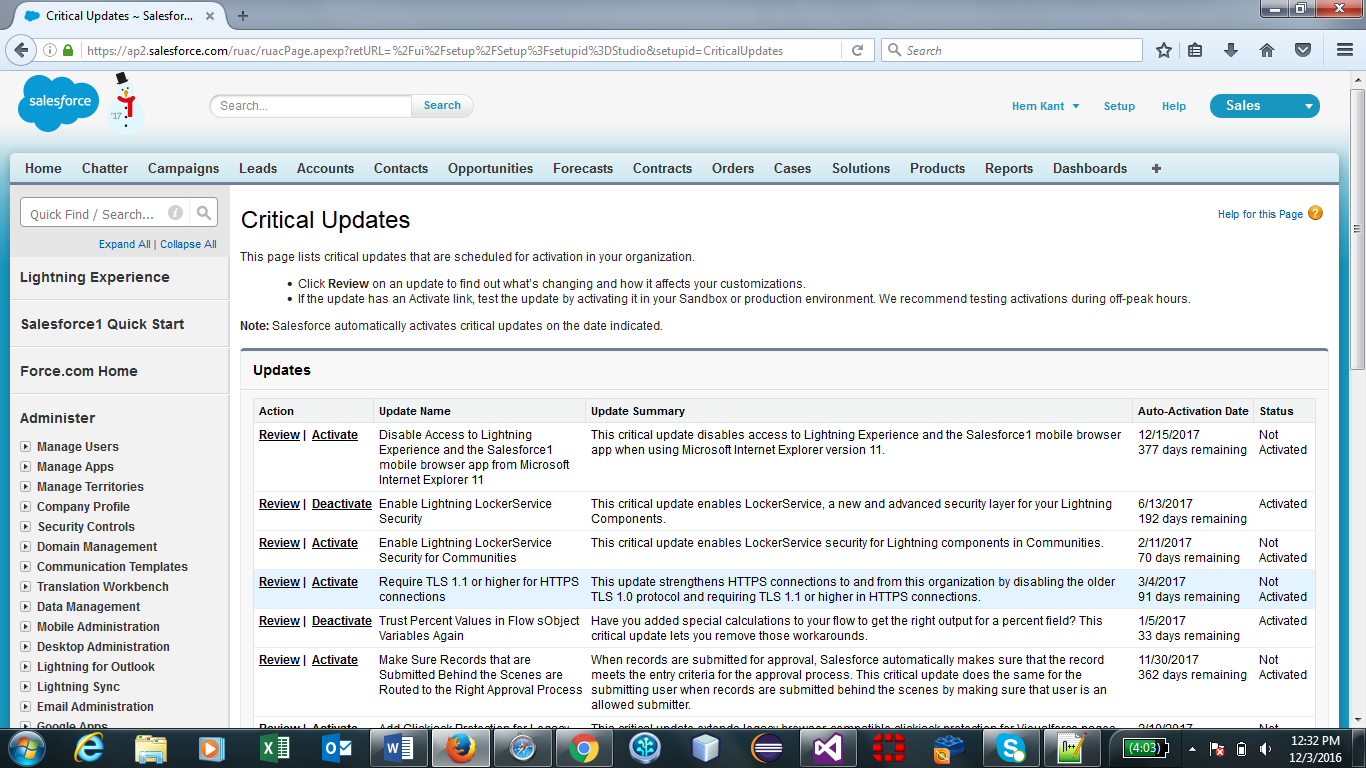
#### Solution: -

1) Go to Build > Critical Updates

2) Under the Update Name column, look for "Require TLS 1.1 or higher for HTTPS connections"

3) For the row that has "Require TLS 1.1 or higher for HTTPS connections", click the "Deactivate" link

If you want to reactivate this update in the future, just click the "Activate" link.



### Error 2: -

when I try to use this WSDL in a .NET integration I get an exception when evaluating the below line:

SforceService sfService = new SforceService ();

Unable to generate a temporary class (result=1).

error CS0030: Cannot convert type 'xxx.Salesforce.ListViewRecordColumn[]' to 'xxx.Salesforce.ListViewRecordColumn'

error CS0030: Cannot convert type 'xxx.Salesforce.ListViewRecordColumn[]' to 'xxx.Salesforce.ListViewRecordColumn'

#### Solution: -

I loaded the Reference.cs file that was created by Updating the Web Reference in Visual Studio into Notepad++ and did a search for adjacent double square brackes [][]

There were 2 lines that had the double square brackets side-by-side. I removed one of the sets, saved the .cs file and now the .net code works.

### Error 3: -

Invalid password error when log in SF from .NET client

#### Solution:-

When you access Salesforce from an IP address that’s outside your company’s trusted IP range using a desktop client or the API, you need a security token to log in. A security token is a case-sensitive alphanumeric code that you append to your password or enter in a separate field in a client application.

For example, if your password is mypassword and your security token is XXXXXXXXXX, enter mypasswordXXXXXXXXXX in the Password field

