Installation Guide

Salesforce Connect 1.4.1

A guide to installing Salesforce Connect for Sitecore Experience Platform

sitecore®
Own the experience™

Table of Contents

Chapter 1

Introduction

This chapter provides background information on Salesforce Connect.

1.1 Overview

Salesforce Connect provides the ability to make data from Salesforce available in Sitecore and to make data from Sitecore available in Salesforce. It offers valuable functionality that can be implemented without the involvement of developers, while also offering a complete set of extensibility and customization features in order to ensure that the product fits into your business, rather than having to make your business work around the product.

Salesforce Connect was designed with several personas in mind.

1.1.1 Marketers

A CRM stores information that is an essential part of an organization's sales and marketing activities. Salesforce Connect enables marketers to leverage information from Salesforce in order to build personalized, contextualized digital experiences.

Salesforce Connect allows marketers to:

- Personalize digital experiences using contacts and campaigns from Salesforce.
- Create email segments using information from Salesforce.
- Track visitor activity on Sitecore-delivered digital experiences using Salesforce campaigns.

1.1.2 Sitecore administrators

Sitecore administrators are responsible for configuring Sitecore so that the relevant CRM data is available to Sitecore, and that the relevant Sitecore data is available to the CRM. Salesforce Connect is designed to make this as easy and adaptable as possible.

Salesforce Connect allows Sitecore administrators to:

- Configure the data synchronization process using the Sitecore Content Editor.
- Extend the data synchronization process without having to involve developers.

1.1.3 Developers

One of the goals of Salesforce Connect is to minimize the need to involve developers when integrating CRM and Sitecore. Traditionally, CRM integration requires a large amount of involvement from developers.

CRM systems are designed to handle the unique way each business operates. As a result, no two CRM implementations are the same. While this is great for businesses (because it allows the software to accurately reflect the business), it complicates integration (because the more customized a system is, the more custom logic must be implemented in order to handle the customizations). Developers are responsible for implementing this custom logic.

Salesforce Connect provides a framework that enables Sitecore administrators to handle much of this custom logic. But there are always going to be cases where developers are needed. For these cases Salesforce Connect reduces the amount of work that is needed by focusing the developer's attention to specific components that can be incorporated into the framework.

Salesforce Connect allows developers to:

- Build custom components that can fit into existing synchronization processes.
- Change existing components in order to accommodate specific business requirements.

1.2 Architecture

Salesforce Connect provides a fairly wide variety of functionality. This functionality is implemented in separate products. As a result, Salesforce Connect is a combination of components from multiple products.

You can use Salesforce Connect without having a strong understanding of the different components it is made up of. However, if you understand the different components it will be easier for you to use and manage Salesforce Connect.

Salesforce Connect is made up of, primarily, two components:

- Sitecore CRM Connect
- Sitecore Data Exchange Framework

1.2.1 Sitecore CRM Connect

Sitecore CRM Connect is an integration framework that provides compatibility between Sitecore and CRM systems. It offers a simplified API for interacting with CRMs in a consistent way. This makes it possible for Sitecore to develop CRM-aware features that are compatible with any CRM.

Salesforce Connect provides an implementation of Sitecore CRM Connect that is able to interact with Salesforce. This implementation is called the "Salesforce Provider for Sitecore CRM Connect".

1.2.2 Sitecore Data Exchange Framework

Sitecore Data Exchange Framework is an integration framework is an ETL (extract, transform, load) framework for Sitecore. It allows you to read data from a source system, transform it into a format that is compatible with a target system, and then load the data into the target system.

Salesforce Connect includes extensions to Sitecore Data Exchange Framework that allow Salesforce Connect to map and exchange data between Sitecore and Salesforce. These extensions are called the "Salesforce Provider for Sitecore Data Exchange Framework".

1.3 Terminology

In order to effectively use Salesforce Connect, you must understand some fundamental terminology. This section provides a glossary of terms.

1.3.1 Synchronization process modeling

"Synchronization process" refers to the process used to read data from one system (called the "source" system) and write that data to another system (called the "target" system). One of the features of Salesforce Connect is that you can synchronize data between Salesforce and Sitecore.

You will not see this term in the product. This is a description of something that the software enables. It is used to describe the software in action. More precise terminology is used to describe how, exactly, the software provides the ability to create, manage and run synchronization processes.

Term	Description	Product that provides the component
Pipeline Batch	The configuration of a specific synchronization process. Running a synchronization process involves running a Pipeline Batch. A Pipeline Batch has one or more Pipelines assigned. These Pipelines are handled in the order in which they are assigned to the Pipeline Batch.	Data Exchange Framework
Pipeline Batch Processor	Implements the logic that runs a Pipeline Batch. This component implements the logic that determines the order in which the Pipeline assigned to the Pipeline Batch are run, and what happens if an error occurs while a Pipeline is run. A Pipeline Batch Processor is assigned to a Pipeline Batch.	Data Exchange Framework
Pipeline	A collection of instructions. A Pipeline does not do anything itself but have a name and a set of instructions in a specific order. These instructions are called Pipeline Steps. A Pipeline has one or more Pipeline Steps assigned. The order that the Pipeline Steps are assigned is important because this is the order in which they are handled when a Pipeline Batch is run. A Pipeline may be used by one or more Pipeline Batches.	Data Exchange Framework
Pipeline Processor	Implements the logic that runs a Pipeline. This component implements the logic that determines the order in which the Pipeline Steps assigned to the Pipeline are run, and what happens if an error occurs while a Pipeline Step is run. A Pipeline Processor is assigned to a Pipeline.	Data Exchange Framework

Term	Description	Product that provides the component
Pipeline Step	A specific instruction that is used when a Pipeline Batch is run. A Pipeline Step may be used by one Pipeline. A Pipeline Step performs a very specific instruction. Examples include reading entities from Salesforce, looping through those entities, and updating a contact in Sitecore using data from a CRM contact. Each of these instructions is represented by a separate step. A synchronization process may consist of a dozen or more different types of steps.	Salesforce Provider Sitecore Provider
Pipeline Step Processor	Implements the logic that runs a Pipeline Step. This component implements the logic that determines how the settings on the Pipeline Step are used to run an instruction. A Pipeline Processor is assigned to a Pipeline.	Salesforce Provider Sitecore Provider

1.3.2 Data access

"Data access" refers to how data is read from and written to various data sources. A "source system" is a system from which data can be read. A "target system" is a system to which data can be written.

Term	Description	Product that provides the component
Endpoint	A representation of a system that may be a source system, a target system, or both a source and a target system.	Salesforce Provider Sitecore Provider
Filter Expression	An object that defines a condition (or set of conditions) that must be met in order for an entity to be read from Salesforce. For example, a Filter Expression may specify that only contacts from a certain account should be read.	Salesforce Provider
Value Reader	Provides the ability to read a value from a source object. If you are a .NET developer, a Value Reader is like a property-getter on a class.	Data Exchange Framework Salesforce Provider Sitecore Provider
Value Writer	Provides the ability to write a value to a target object. If you are a .NET developer, a Value Writer is like a property-setter on a class.	Data Exchange Framework Salesforce Provider Sitecore Provider
Value Accessor	Associates a Value Reader with a Value Writer. If are a.NET developer, a Value Accessor is like a property on an class.	Data Exchange Framework

Term	Description	Product that provides the component
Value Accessor Set	The collection of Value Accessors that define the ability to read and write values on a specific type of source and target objects. If you are a .NET developer, a Value Accessor Set is like a class.	Data Exchange Framework

1.3.3 Data mapping

"Data mapping" refers to the ability to associate a value from a source object with a value on a target object, for the purpose of reading a value from the source object and writing that value to the target object.

Term	Description	Product that provides the component
Value Mapping	The object on which 2 Value Accessors are specified: one represents the component that can read from the source object, and the other represents the component that can write to the target object.	Data Exchange Framework
Value Mapping Set	The collection of Value Mappings that defines the ability to map multiple values from a source object to a target object. When the mappings specified in the Value Mapping Set are applied, it is said to "run". In other words, running a Value Mapping Set will result in the values being read from the source object and written to the target object.	Data Exchange Framework

1.3.4 Converter

A converter, given an object, can create a new object of a different type.

The main use of converters in Data Exchange Framework is to convert a Sitecore item into the appropriate framework object. For example, a Sitecore item represents a specific Value Reader. A converter is used to transform the Sitecore item into a Value Reader.

Note
This type is needed because it is possible to use the framework outside of Sitecore. When the framework runs outside of Sitecore, the configuration is still stored in Sitecore. An external application will read the configuration from Sitecore (meaning it will read Sitecore items) and then use converters to transform the Sitecore items into the appropriate framework object.

1.3.5 Tenant

A Tenant represents a single brand, business unit, or organization. It allows you to isolate the configuration for one brand from another brand.

Chapter 2

Installation

This chapter guides you through the process of installing Salesforce Connect.

2.1 Prerequisites

This section explains the requirements for running Salesforce Connect.

2.1.1 Salesforce

You need a Salesforce user that Salesforce Connect will use to interact with Salesforce. The privileges assigned to this user will determine how much data can be shared between Salesforce and Sitecore. It is recommended you give the user the following privileges, at a minimum:

- Create, Read, Write on the object Contact
- Create, Read, Write on the object Campaign

Note
If you do not have access to a Salesforce account, you can create a free developer account. More information is available at https://developer.salesforce.com.

2.1.2 Sitecore

This product requires Sitecore 8.1 update 1 (or greater).

If you are using Solr for the analytics index, you must use Sitecore 8.1 update 2 (or greater).

If you are deploying Sitecore to Microsoft Cloud using Web Apps, you must use Sitecore 8.2 update 1 (or greater).

Note
Salesforce Connect and Dynamics CRM Connect cannot run concurrently on the same Sitecore server. Future versions of these products will support this.

2.1.3 Sitecore xDB Cloud

At the time this document was written, xDB Cloud does not support custom contact facets or processing/aggregation logic. As a result, the functionality that depends on these features are unavailable to customers using xDB Cloud:

- Personalization based on Salesforce campaign membership
- Segmentation based on Salesforce campaign membership
- Updating Salesforce contacts using data from Sitecore

Note
If you attempt to use the functionality described above with xDB Cloud, you will not see any errors in your local Sitecore log. You will need to look at the log for the aggregation server, which is available from Sitecore Product Support.

Note
A future version of Sitecore Experience Platform is expected to provide full support for xDB Cloud.

2.2 Installation

This section describes how to install Salesforce Connect on your Sitecore servers.

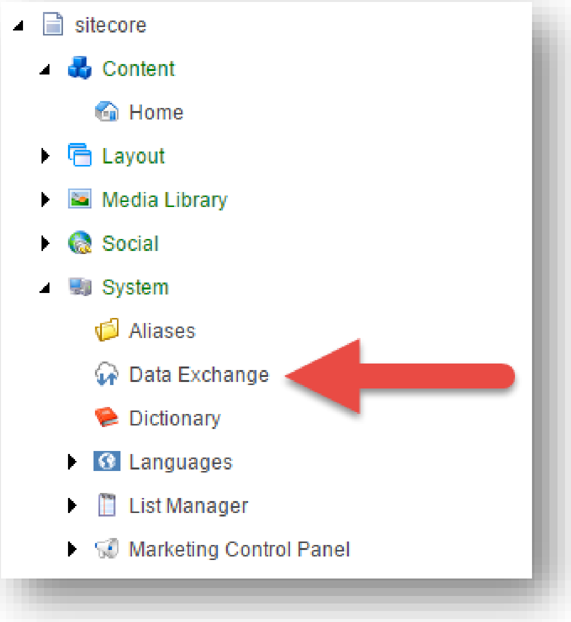
2.2.1 Content Management Role

Typically, data synchronization runs on a content management server. This requires the entire Salesforce Connect product to be installed.

Note
You must install the Sitecore packages in the order listed below. If you install the packages in a different order, your tenants will not be created properly and you will be unable to synchronize data between Salesforce and Sitecore.

1. In Sitecore, open Installation Wizard.
2. Install the "Sitecore Data Exchange Framework" package.
3. Install the "Sitecore Provider for Data Exchange Framework" package.
4. Install the "Salesforce Provider for Data Exchange Framework" package.

You can confirm the module was installed by opening Content Editor and locating the item `/sitecore/system/Data Exchange`.



2.2.2 Content Delivery Role

In most cases, data synchronization will not run on Sitecore delivery servers. The data that is synchronized, however, is used on those servers. As a result, all parts of Salesforce Connect must be installed on Sitecore delivery servers.

2.2.3 Aggregation Server Installation

If you have a dedicated aggregation server, all parts of Salesforce Connect must be installed on the aggregation server.

2.2.4 Reporting Server Installation

If you have a dedicated reporting server, all parts of Salesforce Connect must be installed on the reporting server.

2.2.5 Azure Installation

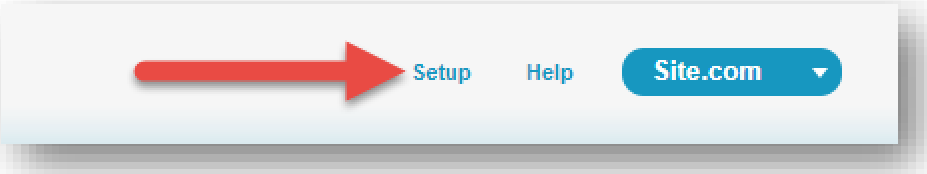
If you are deploying Salesforce Connect to Azure, you must enable the Azure Search.

1. Rename the file `Sitecore.Connect.Crm.Local.Lucene.config` to `Sitecore.Connect.Crm.Local.Lucene.config.example`.
2. Rename the file `Sitecore.Connect.Crm.Local.Azure.config.example` to `Sitecore.Connect.Crm.Local.Azure.config`.

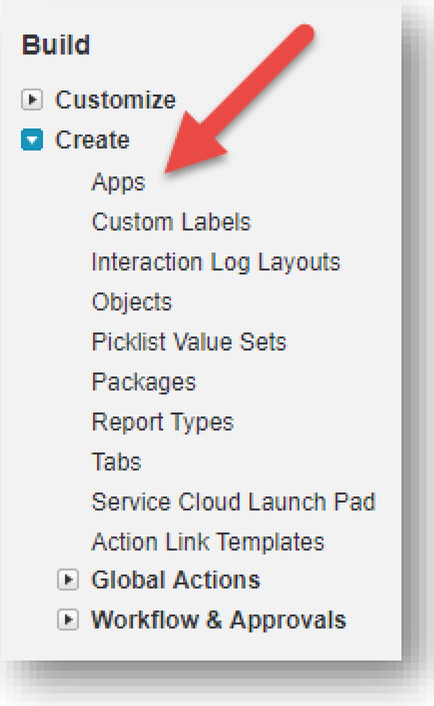
2.2.6 Salesforce Configuration

In order for Salesforce Connect to communicate with Salesforce, the following configuration is needed:

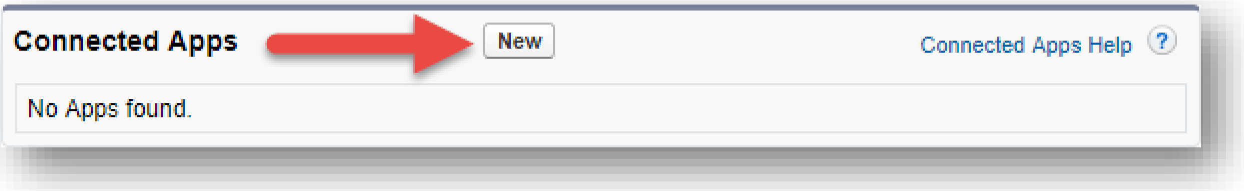
1. Log into your Salesforce account.
2. In the top menu, click **Setup**.



3. In the left menu, click **Create > Apps**.



4. In the Connected Apps section, click **New**.

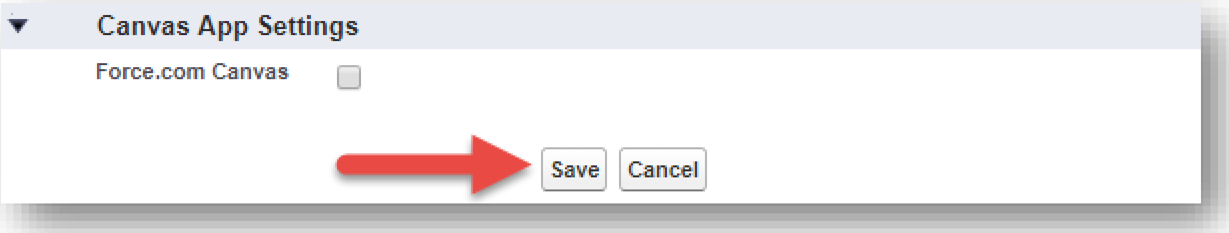


No Apps found.

5. Enter the following values:

Field name	Value	Notes
Connected App Name	Sitecore Data Exchange App	The specific value does not matter.
API Name	Sitecore_Data_Exchange_App	Salesforce generates this value based on the app name. It is recommended that this value not be changed.
Contact Email	[your email address]	The specific value does not matter.
Enable OAuth Settings	Checked	
Callback URL	http://localhost	This value is not used, so the value you enter does not matter.
Selected OAuth Scopes	Access and manage your data (api)	
Require Secret for Web Server Flow	Checked	

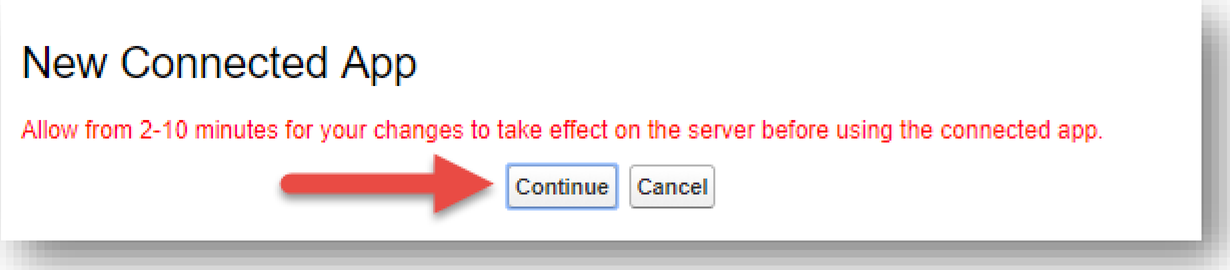
6. At the bottom of the screen, click **Save**.



▼ Canvas App Settings

Force.com Canvas □

7. Salesforce will inform you a new connected app was created. Click **Continue**.



New Connected App

Allow from 2-10 minutes for your changes to take effect on the server before using the connected app.

2.3 Salesforce Connection String

In order for Sitecore to connect to Salesforce, a connection string must be added to the Sitecore server.

2.3.1 Get Connection String Values from Salesforce

Before you can configure the connection string, there are several values you must collect from Salesforce:

User ID

This is the user ID that Sitecore will use to call the Salesforce API.

This user ID does not have to have administrator rights, but it must have sufficient rights to perform the activities you expect to be able to from Sitecore. For example:

- In order to have Salesforce contacts created in Sitecore, the user must have read-access on contacts.
- In order to have Salesforce campaigns created in Sitecore, the user must have read-access on campaigns.
- In order to push contact data from Sitecore into Salesforce, the user must have write-access on contacts.

Password

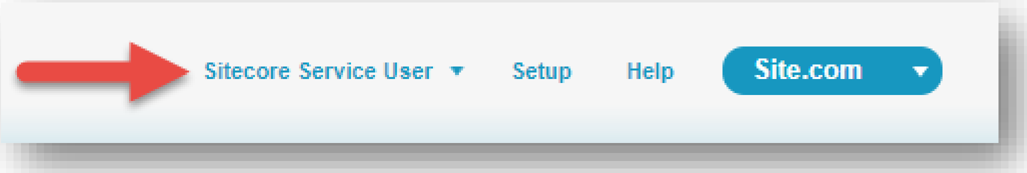
This is the password for the Salesforce user that Sitecore will use to call the Salesforce API.

Note
Your password cannot contain the ; character (semicolon).

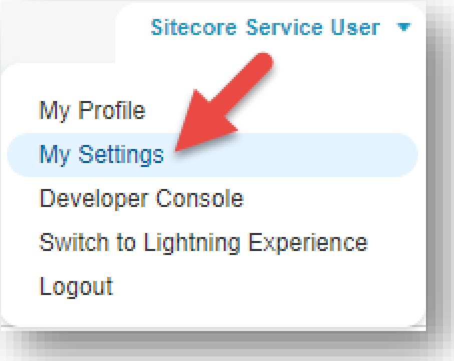
Security token

This value come from the Salesforce user whose connected app. To find this value, do the following:

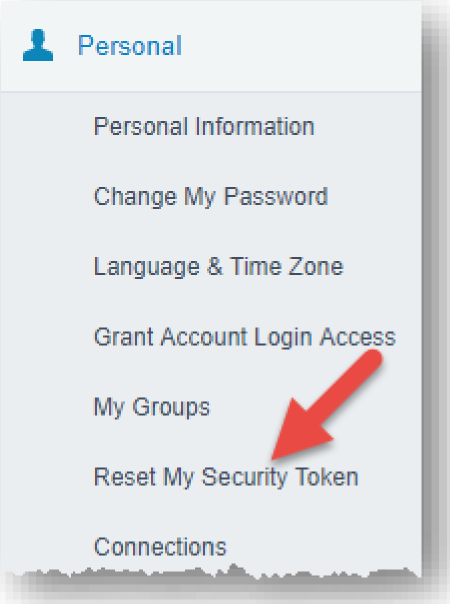
1. In Salesforce, log in with the user ID you plan to use in the connection string.
2. In the top menu, click the user name.



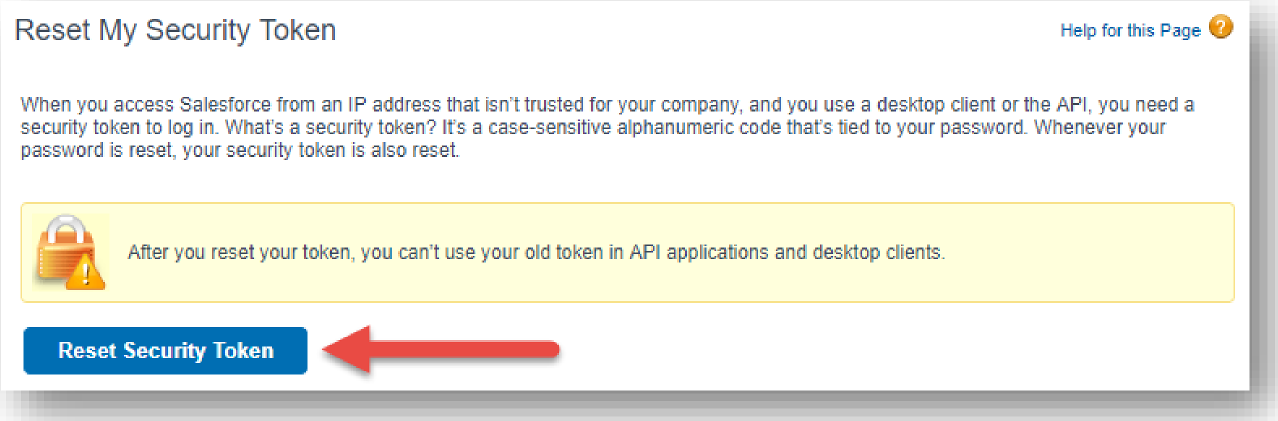
3. Click **My Settings**.



4. In the left menu, click **Personal > Reset My Security Token**.



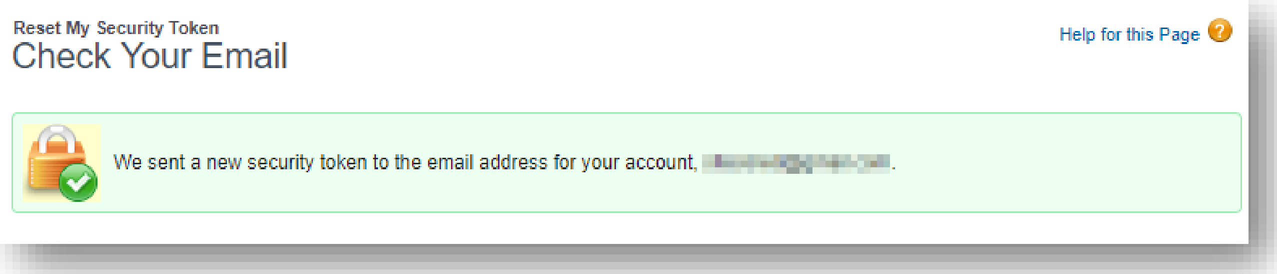
5. Read the warning on the screen carefully. If you are OK with the change that will be made, click **Reset Security Token**.


Reset My Security Token

When you access Salesforce from an IP address that isn't trusted for your company, and you use a desktop client or the API, you need a security token to log in. What's a security token? It's a case-sensitive alphanumeric code that's tied to your password. Whenever your password is reset, your security token is also reset.

After you reset your token, you can't use your old token in API applications and desktop clients.

6. Salesforce will inform you that the new security token will be emailed to you.

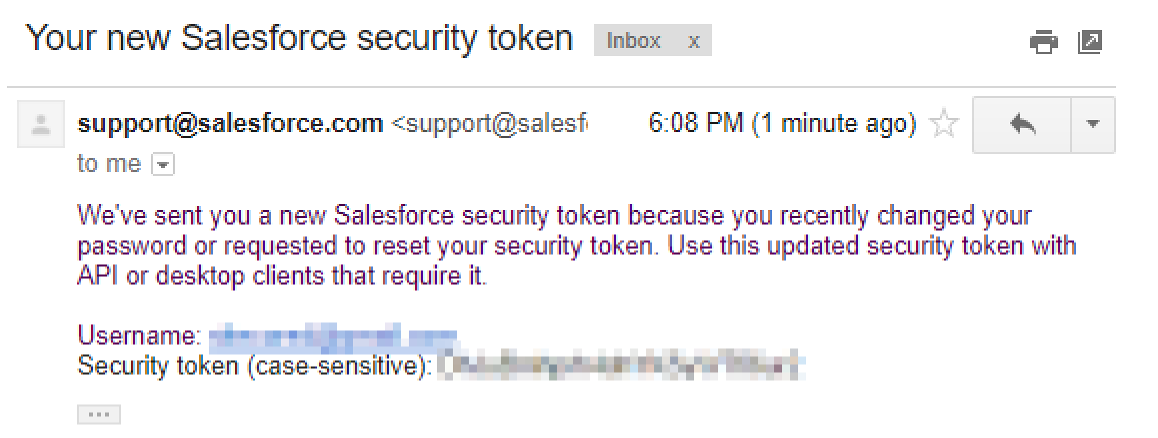


Reset My Security Token

Check Your Email

We sent a new security token to the email address for your account, [redacted].

7. In your email you will find the new security token value.



Your new Salesforce security token

support@salesforce.com <support@salesf

6:08 PM (1 minute ago)

to me

We've sent you a new Salesforce security token because you recently changed your password or requested to reset your security token. Use this updated security token with API or desktop clients that require it.

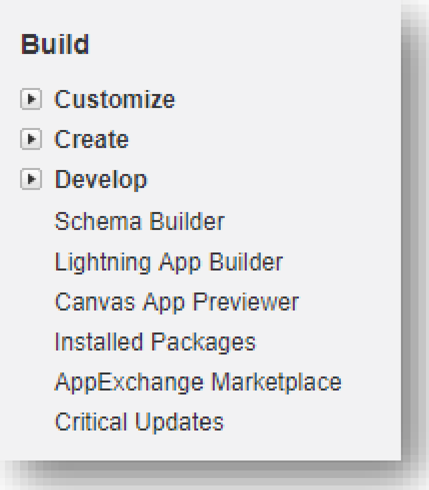
Username: [redacted]

Security token (case-sensitive): [redacted]

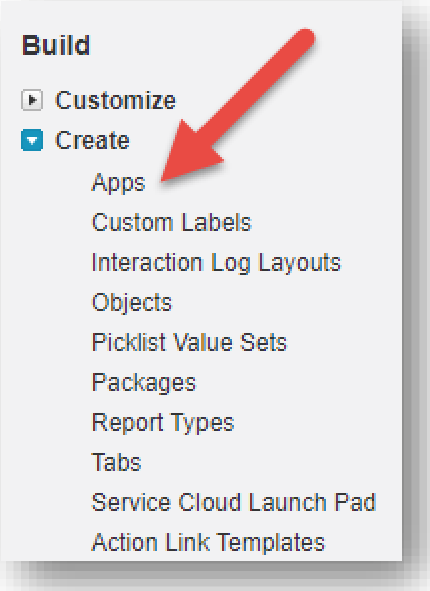
Client ID and Secret Key

These values come from the Salesforce connected app. To find these values, do the following:

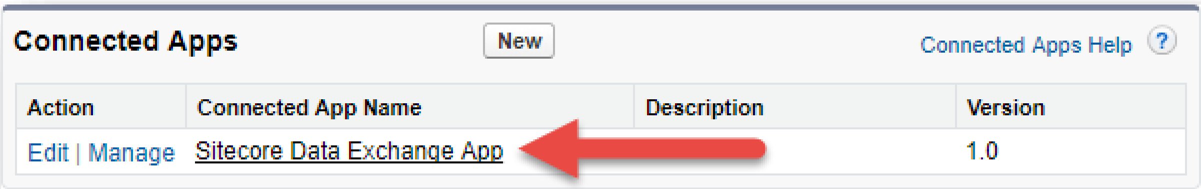
1. In Salesforce, on the left menu, scroll down to the section **Build**.



2. Navigate to **Create > Apps**.



3. In the section **Connected Apps**, click the name of the connected app you created.



Action	Connected App Name	Description	Version
Edit | Manage	Sitecore Data Exchange App		1.0

4. In the section API (Enable OAuth Settings), you will see the values for **Consumer Key** and **Consumer Secret**. Copy these values.

Note
In the connection string, the **consumer key** value is called the **client id**, and the **consumer secret** value is called the **secret key**.

2.3.2 Add the Connection String to Sitecore

Add the following connection string to the ConnectionStrings.config file on your Sitecore server. Be sure to replace the values in brackets with the values you collected from Salesforce:

```
        <add name="mysf" connectionString="user id=[user id];password=[password];client
id=[client id];secret key=[secret key];security token=[security token]" />
```

The following is an example of what the connection string will look like after you have replaced the values in brackets:

```
        <add name="mysf" connectionString="user id=someone@email.com;password=b;client
id=GEH9zlTNB8o8BA45pAeDtC8W.DIqrAzuky2uffEEOwtHxIEhLzkmPwjz8KF_vzWY8dNIfurWHpfbZPGdtc3b;secret
key=546856899979835412З;security token=g3ygFuNzGgm33YTfsM3WKG3AA" />
```