**Understanding Salesforce Authentication Part 1**

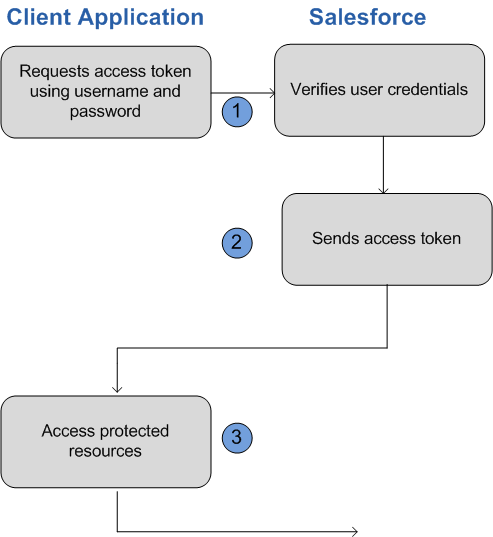
Salesforce uses the OAuth protocol to allow users of applications to securely access data using Salesforce REST APIs. Salesforce supports several different OAuth 2.0 authentication flows mentioned below:

* **Username-password flow** - Used to authenticate when the consumer already has the user’s credentials.
* **Web server flow** - Used by applications that are hosted on a secure server.
* **User-agent flow** - Used by client applications residing in the user’s device such as JavaScript, or from a mobile device or a desktop application.

In this article we will discuss how we can implement Username-password flow authentication in your .net web application.

**Understanding the Username-Password OAuth Authentication Flow**

This OAuth authentication flow involves passing the user’s credentials back and forth. Use this authentication flow only when necessary.



To Authenticate user POST request would need to be made to the appropriate Salesforce token request endpoint i.e. #2 in the list below. Primary OAuth endpoints are:

1. For authorization: <https://login.salesforce.com/services/oauth2/authorize>
2. For token requests: <https://login.salesforce.com/services/oauth2/token>
3. For revoking OAuth tokens: <https://login.salesforce.com/services/oauth2/revoke>

If you’re verifying authentication on a sandbox organization, use “test.salesforce.com” instead of “login.salesforce.com” in all the OAuth endpoints listed above.

| **Parameter** | **Description** |
| --- | --- |
| grant\_type | Must be *password* for this authentication flow. |
| client\_id | The Consumer Key from the connected app definition. |
| client\_secret | The Consumer Secret from the connected app definition. |
| username | End-user’s username. |
| password | End-user’s password. **You must append the user’s security token to their password** |

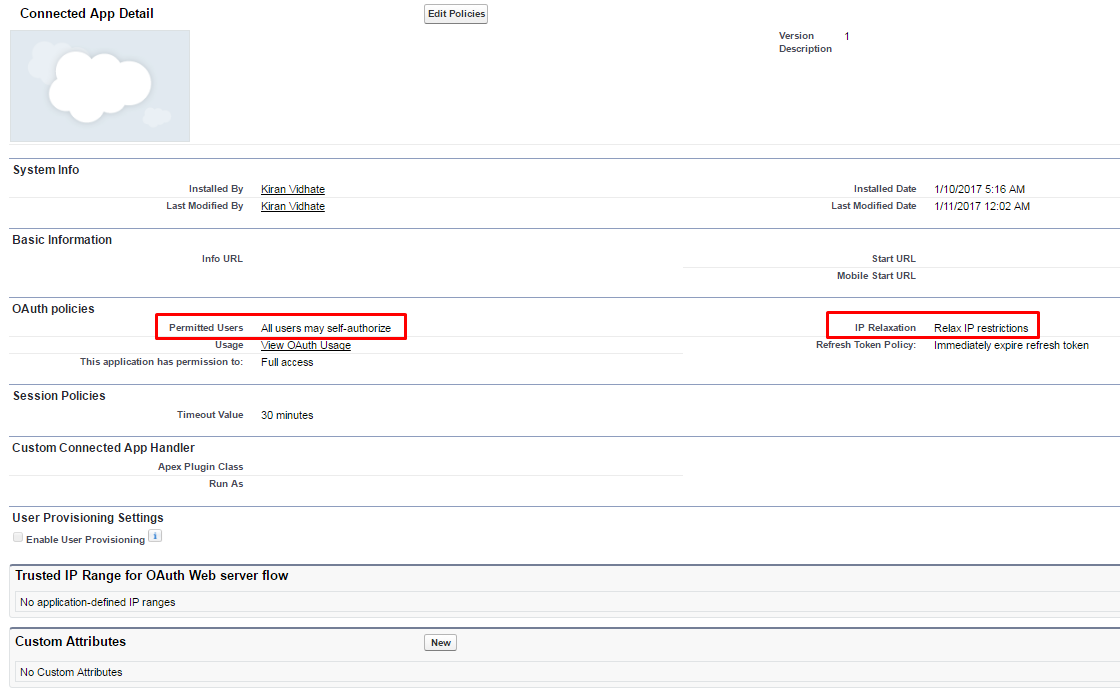
**Creating Connected App in Salesforce**

To authenticate using OAuth, you must create a connected app that defines your application’s OAuth settings for the Salesforce organization.

When you develop an external application that needs to authenticate with Salesforce, you need to define it as a new connected app within the Salesforce organization that informs Salesforce of this new authentication entry point.

Steps to create a new connected app:

* From Setup, enter Apps in the Quick Find box, then select Apps and click New to start defining a connected app.
* Enter the name of your application.
* Enter the contact email information, as well as any other information appropriate for your application.
* Select Enable OAuth Settings.
* Enter a Callback URL (Not used for for Username-Password flow). Depending on which OAuth flow you use, this is typically the URL that a user’s browser is redirected to after successful authentication.
* Add all supported OAuth scopes to Selected OAuth Scopes. These scopes refer to permissions given by the user running the connected app. Following screenshot is the connected app configuration used in this demo



* Click Save. The Consumer Key is created and displayed, and the Consumer Secret is created (click the link to reveal it). These keys should be used in the parameter list mentioned above

Attached sample application demonstrates usage of Username-Password OAuth Authentication Flow to fetch accounts list from Salesforce organization.

Salesforce,oAuth,c#