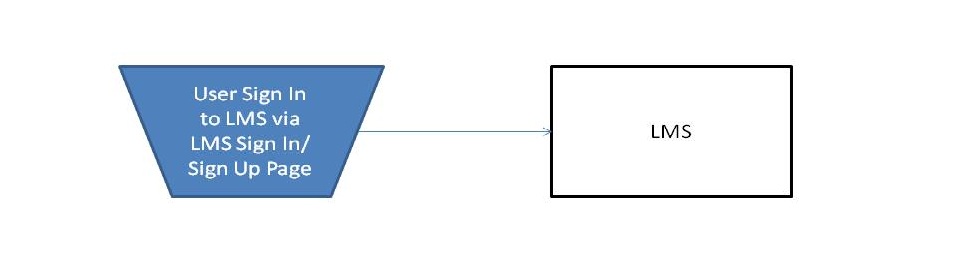
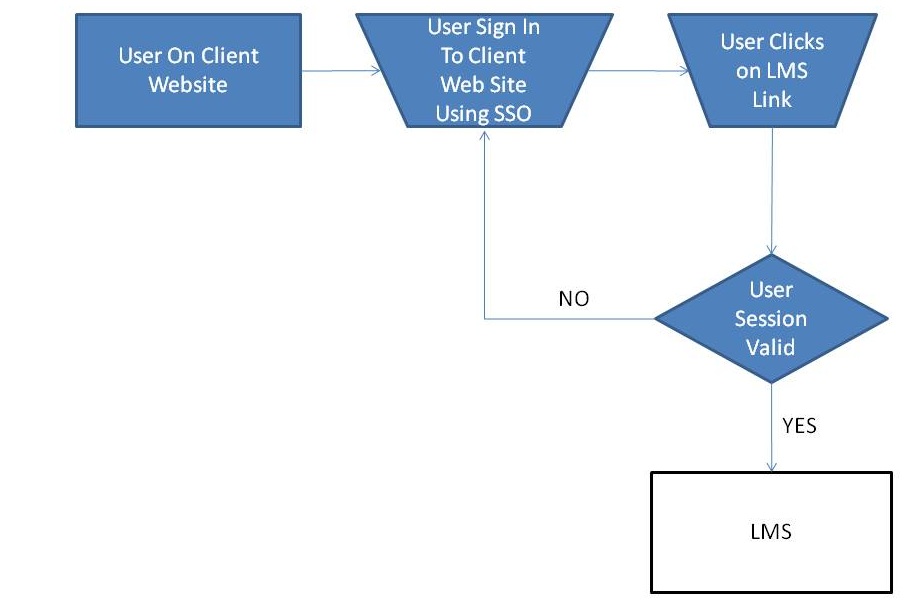
# Options for Authentication

YM requires user to be authenticated before accessing the LMS. User can be authenticated in different ways. This is very very much specific to the client environment.

There are two major ways to authenticate user into the LMS.

* User Directly Sign Up/Sign In through LMS Sign In/Sign Up Page.



* User gets redirected to LMS from Client Web Site. At this point there will be some sort of authentication transfer Client Web Site and LMS (various options to transfer control to LMS is discussed below).   
    
  

No matter what approach is followed for control transfer, we would need some minimal information to give him/her access to the LMS.

* First Name
* Last Name
* Email
* Unique User ID

Note: Some additional information is also required, along with above mandatory information. Please refer to specific integration document for details.

Below are a few approaches that are most common with our clients.

* Simple YM Gateway Passthru Approach
* Using Client SSO (Provided Client has some kind of service that we can call to authenticate)
* Sub Domain Appraoch (We can set up YM LMS to be sub domain for Client environment and read any user profile that is in their SSO).

#### Simple YM Gateway Passthru Approach

Simple YM Passthru allows learners to seamlessly log into the CrowdWisdom Learning portal from client website. Learners requiring access to the CrowdWisdom Learning Portal will be pre-authenticated by client website and then they will be redirected to the single access point URL defined below:

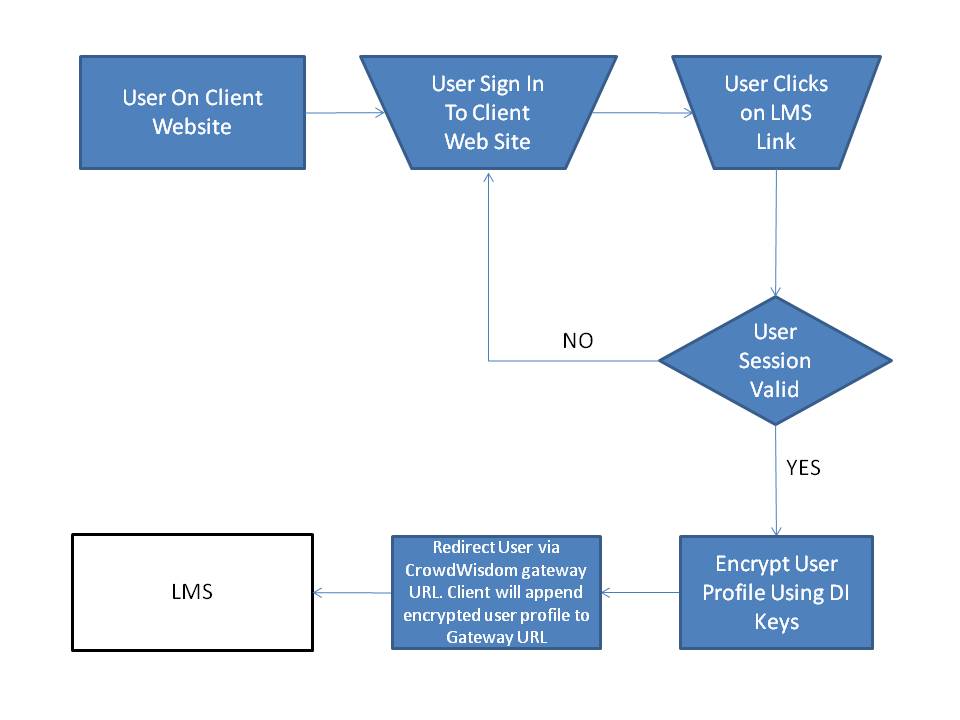
Production Base URL: http://<context>.mycrowdwisdom.com/diweb/gateway/

Staging Base URL: http://<context>.precrowdwisdom.com/diweb/gateway/

Any User profile information needed for LMS will be passed along with this URL in an encrypted format.

Resulting redirect would look something like this:  
**http://<context>.mycrowdwisdom.com/diweb/gateway/?data=dfhiusIUHREdjsdiisdhfdsdsd**

Note: The secret encryption key and initialization vector to be used in the encryption algorithm will be provided by YML in the form of base64 encoded strings.



#### *Sample Client (pseudo code)*

//prepare parameters to be passed to the gateway  
Map<String,String> params = new HashMap<String, String>();  
params.put("clientId", "123);  
...  
...  
//construct key/value pair query string  
StringBuilder sb = new StringBuilder();  
for (Map.Entry<String,String> param : params.values()) {  
if (sb.length() > 0) sb.append('&');  
sb.append(param.getKey());.append('=');

//url encode the value as this is query string  
sb.append(jURLEncoder.encode(param.getValue()));  
}

//create a cipher (TripleDES algorithm, CBC mode with PKCS5Padding)  
Cipher cipher = Cipher.getInstance("DESede/CBC/PKCS5Padding");

//decode Digital Ignite provided KEY and IV  
byte[] key = Base64.decode(KEY); //KEY is Base64 encoded  
byte[] iv = Base64.decode(IV); // IV is Base64 encoded  
SecretKey secretkey= S; // create secret key using key byte array  
IvParameterSpec ivSpec=S; //create initialization vector using the iv byte array

//initialize the cipher for encryption  
cipher.init(Cipher.ENCRYPT\_MODE, secretKey, ivSpec);

//use base64 encoding for resulting bytes  
String dataStream = Base64.encode(cipher.doFinal(sb.toString().getBytes()));

//redirect to gateway (url encode the data stream as it is being attached to the url)  
getRespone().redirect(gatewayURL + "?data=" + URLEncoder.encode(dataStream));

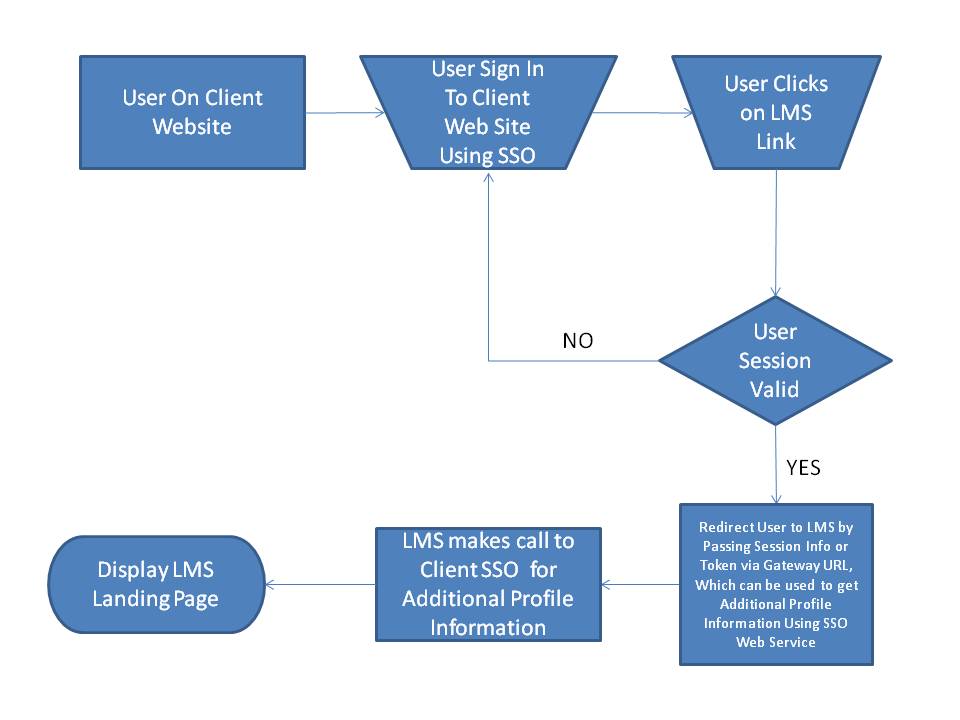
Note : Please refer to Gateway passthru document of actual implementation. <put the link to gateway doc here>

#### Client SSO

This approach is appropriate for clients who has some kind of SSO for their website.

Workflow Steps

* User Sign In/Sign Up happens on client website.
* User Clicks on a list to access LMS.
* He/She will be redirected to LMS via CrowdWiseom Gateway URL
* Along with URL Client Website will pass SSO token to LMS
* LMS makes additional call to Client SSO (provided they have some kind of web service that accepts tokens and returns Additional user profile information that is required for LMS)
* Up on successful return of profile information, he/she will be given access to LMS appropriately.



#### Configure LMS as Sub Domain for Client Environment

This approach is appropriate where LMS client would like to setup CW LMS as their sub domain.

Being a sub domain, LMS can access any session Info (client Web Site Cookies) which might hold all the profile information to give access to lms.

With Some client implantations, Cookie will hold SSO token information Which LMS can access and make an additional call to SSO web service to get additional Profile Information.

Similar to previous approach, User is redirected to CrowdWisdom LMS via Gateway URL. One difference from previous being, no information is sent to LMS via Gateway URL. Any Information needed will be available in client session cookie and LMS has access to cookies.

Assumptions

* In order to implement this approach, LMS would need an **SSL Certificate from client**.
* **YM will provide client with an IP address,** Client will have to **map Given LMS IP Address with one of their host names**

