App.conf settings Login Control.

In addition to the normal appSettings that are available with your application, an additional set of settings is added to your app config by this control, they are outlined below.

<add key="enableJava" value="false" />   
<add key="CrmDeploymentType" value="True" />  
<add key="CrmOrg" value=" " />  
<add key="CrmPort" value="" />  
<add key="CrmServerName" value=" " />  
<add key="UseDefaultCreds" value="False" />  
<add key="CrmUserName" value=" " />  
<add key="CrmDomain" value=" " />  
<add key="CrmPassword" value=" " />  
<add key="CacheCredentials" value=" True" />  
<add key="CrmUseOnPrem" value=" True" />  
<add key="CrmUseSSL" value=" False" />  
<add key="CrmOnlineRegion" value=" " />  
<add key="AuthHomeRealm" value=" " />

EnableJava – Not required… Defaults to True.  
This setting is used to enable or disable loading of the Java Support Libraries.

CrmDeploymentType– Not required, Valid values are Prem , Online, O365. If not present Prem is assumed

AuthHomeRealm – Not required, Utilized when Claims based authentication is used for CRM and is populated by the user login process. Can be pre-populated

CrmOnlineRegion – Not required, only present if CRM online is used. Defaults to the CRM region that a given online org is provisioned from, as determined by the Live Login process.

CrmOrg – Not required – Defaults to determined value based on available orgs on the target CRM server.   
If present on load, this value is used by the auto login process, and prepopulated into the CRM Org during the login process, note that this is the Unique Name of the CRM org, not the friendly name. The Unique name can be found in the developer tools area of the CRM 2011 settings \ customizations.

CrmPort – Not required – Defaults to user entered value.   
If present on load, this value is used by the auto login process, and prepopulated into the CRM Port Field on the login form.

CrmServerName - Not required – Defaults to user entered value.   
If present on load, this value is used by the auto login process, and prepopulated into the CRM Server Name Field on the login form. Not that this is just the server name or IP, no other qualifiers or ports are allowed

CrmUseSSL – Not required – Defaults to false.  
This determines if Http:// or Https:// is used to connect to the server.

UserDefaultCreds – Not required – Defaults to true.   
If present on load, this value is used by the auto login process, and prepopulated into the Use Default Creds Field on the login form.

CrmUserName - Not required – Defaults to user entered value.   
If present on load, this value is used by the auto login process, and prepopulated into the CRM Server Login Field on the login form.

CrmDomain - Not required – Defaults to user entered value.   
If present on load, this value is used by the auto login process, and prepopulated into the CRM Server Domain Field on the login form.

CrmPassword - Not required – Defaults to user entered value.   
If present on load, this value is used by the auto login process, and prepopulated into the CRM Server Password Field on the login form.

CacheCredentials – Not required – Defaults to True,  
if false on load, automatically clears the password field and presented the login field.

Using the HomeRealmStore.xml file for Claims Authentication.

When using Claims authentication, a Home Realm is required. The home realm information includes the url for the claims management server.

To simplify this process, the HomeRealmStore.xml file contains the required Home Realm information and a friendly name that will allow the user to select what realm they would like to auth from.

You can find an example of this file in the Model folder of the Template project.

\*added in Update 3,   
Depending on the configuration of the Claims process with CRM, there appears to be the need to allow for a claims authentication without the Home Realm being specified.   
To support this, when creating the home realm entry in the HomeRealmsStore.xml, use the value <http://blank> for the URL of the home realm, this will cause the system to pass a NULL for the home realm URI.

A Post build event can be added to automatically copy the home realms xml file to the output directory, an example if that event is here:

REM copy the HomeRealms.xml file to the output directory  
copy "$(ProjectDir)\Model\HomeRealmsStore.xml" "$(TargetDir)" /y

Example of a HomeRealmStoreFile:

<ClaimsHomeRealmOptions>

<HomeRealm DisplayName="fabrikam.com" Uri="https://federoad-1.federoad.com/adfs/services/trust/mex/" />

<HomeRealm DisplayName="contosoTel.com" Uri="https://blank" />

</ClaimsHomeRealmOptions>

Trace support for Login control.

The DynamicsCrm.CrmConnectControl and supporting DynamicsCrm.Connector controls support tracing via the .net tracing system.

You can enable tracing for the DynamicsCrm.CrmConnectControl using the source :

<source name="Microsoft.Uii.AddOn.DynamicsCrm.CrmConnectControl"

switchName="Microsoft.Uii.AddOn.DynamicsCrm.CrmConnectControl"

switchType="System.Diagnostics.SourceSwitch">

<listeners>

<add name="console" type="System.Diagnostics.DefaultTraceListener" />

<remove name="Default"/>

</listeners>

</source>

And the switch   
<add name="Microsoft.Uii.AddOn.DynamicsCrm.CrmConnectControl" value="Verbose"/>

To enable Tracing for the DynamicsCrm.Connector, use the source:

<source name="Microsoft.Uii.AddOn.DynamicsCrm.Connector.CrmWebSvcActions"

switchName="Microsoft.Uii.AddOn.DynamicsCrm.Connector.CrmWebSvcActions"

switchType="System.Diagnostics.SourceSwitch">

<listeners>

<add name="console" type="System.Diagnostics.DefaultTraceListener" />

<remove name="Default"/>

<add name ="fileListener"/>

</listeners>

</source>  
And Switch:   
<add name="Microsoft.Uii.AddOn.DynamicsCrm.Connector.CrmWebSvcActions" value="Verbose" />