Microsoft Dynamics CRM Security Provider Module for Sitecore CMS 6.0-6.5

Developer's Guide

A developer's guide for the CRM security provider

Table of Contents

Chapter 1

Introduction

Sitecore provides a flexible security system based on the ASP.NET security providers model. It allows Sitecore users to use different data storages as a source for security objects in Sitecore.

The Microsoft Dynamics CRM Security Provider allows you to manage the vital customer data stored in your CRM system directly in Sitecore. It allows seamless integration and access between your CRM system and your Sitecore website.

The module uses CRM web services to represent data from the Microsoft CRM System as security objects in Sitecore. In Sitecore, CRM contacts are rendered as users and CRM marketing lists as roles.

- **Chapter 1. Introduction**
  The first chapter is a brief introduction to this guide.
- **Chapter 2. Installation and Configuration**
  The second chapter covers useful installation and configuration aspects.
- **Chapter 3. User Guide**
  The third chapter gives practical tips to module usage.
- **Chapter 4. FAQ**

Chapter 2

Installation and Configuration

This chapter describes installation and configuration instructions for this module.

Installation of the module requires previous installation of a Sitecore package and modifying Sitecore configuration files.

This chapter contains the following sections:

- Installing a Sitecore Package
- Modifying the Configuration Files

2.1 Installing a Sitecore Package

The Microsoft Dynamics CRM Security Provider module is distributed as a Sitecore package. You can use the Installation Wizard to install it. To open the wizard, in the Sitecore desktop, click **Sitecore**, **Development Tools**, **Installation Wizard**.

2.1.1 Package Content

The package contains the following elements:

- CRM Security Provider assembly files.
- The CRM Security Provider configuration file.
- The CRM Contact profile item and its template.

2.1.2 Installation Prerequisites

To install the CRM Provider module, you need:

- Sitecore CMS 6.0.1 rev. 090317 (Sitecore CMS 6.0.1 rev. 090212 + Sitecore CMS 6.0.1 Update rev. 090317 package) or later.
- Microsoft CRM system, version 3 or later.

2.2 Modifying the Configuration Files

To configure the module to connect with CRM and to represent CRM information in Sitecore in the way that your organization requires, you must make some manual changes to the `web.config` file. You must edit the `/App_Config/ConnectionStrings.config` and the `/App_Config/Security/Domains.config.xml` files. You also need to check a setting in the `/App_Config/Include/crm.config` file.

For more information, see the *Configuring Unique Key Property* section.

2.2.1 Connecting to the CRM system

Note
Configuring the Microsoft Dynamics CRM Security Provider module is identical both for Microsoft CRM On-Premise v4 and Microsoft CRM On-Premise v5.

Configuring the Microsoft Dynamics CRM Security Provider module is identical both for Microsoft CRM Online v4 and Microsoft CRM Online v5.

To use the Microsoft Dynamics CRM Security Provider module, you must have a properly configured Microsoft CRM System to connect with.

Add a connection string for the CRM system Web service to the `<connectionString>` element of `/App_Config/ConnectionStrings.config` file. If you're using a SQLite installation, you must make similar changes in the `/App_Config/ConnectionStringsSQLite.config` file.

The configuration file should look like this:

```
        <connectionStrings>
          <add name="CRMConnString" connectionString="CRM:url=crm_url;user id=user;password=password;organization=crm_organization;authentication type=authentication_type;partner=liveID_Partner;environment=liveID_environment" />
        </connectionStrings>
```

The `<connectionStrings>` element can have a number of connection strings defined. Each one is defined by the `<add>` tag. This element has a number of attributes, but here we use only two of them:

Attribute	Description
Name	The name of the connection string. Any entity using this connection string will address it by this name.
connectionString	The connection string itself.

The Microsoft Dynamics CRM Security Provider module supports the following connection string format:

- Each string starts with the `CRM` prefix.
- A URL attribute that contains a URL to the CRM Web service.
- The last part is the login credentials that are used to enter the CRM system.

The connection string should look like this:

```
        CRM:url=<url_to_crm_webservice>;user id=<user_name>;password=<user_password>
organization=crm_organization;authentication type=authentication_type
```

The connection string parts:

- For Microsoft CRM System v3:

`url` — the URL of the CRM web service; it should contain `http://<crm_host>[:<crm_host_port>]/mscrmservices/<2006>/crmservice.asmx`.

- For Microsoft CRM System v4 and v5:

  `url` — the URL of the CRM web service; it should contain `http://<crm_host>[:<crm_host_port>]/mscrmservices/<2007>/crmservice.asmx.`

URL for CRM Online Connection String

To access the CRM Online web services the URL should start with https and `<crm_host>` should include api information for example `https://<organization_name>.api.crm.dynamics.com/...`

- `user id` — the name of the user in CRM or in Windows Live service (if Passport authentication type is used) which will be used to connect the CRM and retrieve/update data in it.
- `password` — the user's password for the CRM or Windows Live service.
- `organization` — the name of the organization unit in CRM which will be used for the connection; this is required for the connection to CRM v4 and v5.
- `authentication type` — the type of the authentication; valid values are "0", "1" and "2" which corresponds to AD (on-premise deployment), Passport (CRM Online) and SPLA (Internet-facing deployment) authentication type in CRM; this is required for the connection to CRM v4 and v5.
- `partner` — Live ID partner property, used with the Passport authentication type; this is required for the connection to CRM Online instance for example *crm.dynamics.com*.
- `environment` — Live ID environment property, used with the Passport authentication type; this is required for the connection to CRM Online instance, for example, *Production*.
- `unsafeAuthenticatedConnectionSharing` — Defines the authentication behavior of the CRM services. For more information, see http://msdn.microsoft.com/en-us/library/system.net.httpwebrequest.unsafeauthenticatedconnectionsharing.aspx. The default value is *true* for performance purposes.
- `preAuthenticate` — Defines the authentication behavior of the CRM services. For more information, see http://msdn.microsoft.com/en-us/library/system.net.webrequest.preauthenticate.aspx. The default value is *true* for performance purposes.

The CRM Web Service v3 in CRM 4

If you use Dynamics CRM v4 you can use the same connection string used for Dynamics CRM v3. In this case providers will have all the limitations of Dynamics CRM v3.

Incorrect Connection String

If you can't establish a connection with the connection string you provide, Sitecore will still work. However, the CRM data won't be retrieved.

2.2.2 Adding a New Domain

Open the `/App_Config/Security/Domains.config.xml` file and add the following line to the domains element:

```
        <domain name="crm" ensureAnonymousUser="false" defaultProfileItemID="{DDEDA46F-169B-4A70-8732-DBD3F407AF2E}"/>
```

The `defaultProfileItemID` attribute defines the profile item that is used for users from the domain if the profile isn't set for the user explicitly.

To specify an email as a user name, add the `AccountNameValidation=".*"` attribute to the domain element.

2.2.3 Configuring ASP.NET Security Providers

The Dynamics CRM Security Provider module uses the ASP.NET security model architecture.

The module uses a set of basic providers to manage users, roles, and profile properties.

Service	Description
Membership provider	Provides a set of operations to get users, create, update, delete them, and also to perform some other operations, such as validating a user (by username and password) and changing the user password. For more information about the membership service, visit the MSDN library.
Role provider	Provides a set of operations to get roles, create, delete them, add users to and remove users from roles. For more information about the roles service, visit the MSDN library.
Profile provider	Provides a set of operations to get/set the properties for a user profile, as well as various actions for a profile objects (delete/find profiles, etc.) For more information about the profile service, visit the MSDN library.

Configuring Switching Providers

For providers to work together with the standard Sitecore security providers (and other custom providers), you must make the following changes:

- Change the `realProviderName` attribute value of the sitecore provider element under system.web>membership>providers to "switcher". It should look like this:

```
<add name="sitecore"
type="Sitecore.Security.SitecoreMembershipProvider, Sitecore.Kernel"
realProviderName="switcher" providerWildcard="%" raiseEvents="true" />
```

- Change the `realProviderName` attribute value of the sitecore provider element under system.web>membership>roleManager to "switcher". It should look like this:

```
<add name="sitecore" type="Sitecore.Security.SitecoreRoleProvider,
Sitecore.Kernel" realProviderName="switcher" raiseEvents="true" />
```

- Change the `defaultProvider` attribute value of the system.web>profile element to "switcher". It should look like this:

```
<profile defaultProvider="switcher" enabled="true"
inherits="Sitecore.Security.UserProfile, Sitecore.Kernel">
```

After this you need to configure the CRM security providers (see the next chapters of this guide) and add the following elements:

- to the sitecore>switchingProviders>membership element (the order is not important)

```
<provider providerName="crm" storeFullNames="false" wildcard="%"
domains="crm" />
```

- to the sitecore>switchingProviders>roleManager element (the order is not important)

```
    <provider providerName="crm" storeFullNames="false" wildcard="%"
domains="crm" />
```

- to the sitecore>switchingProviders>profile element (it must come before the sql definition in this group)

```
    <provider providerName="crm" storeFullNames="false" wildcard="%"
domains="crm" />
```

For more information, see the Low Level Sitecore Security and Custom Providers article http://sdn5.sitecore.net/SDN5/Articles/Security/Low_level_Sitecore_Security_and_Custom_Providers.aspx

Configuring the Membership Provider

Add the following element to the system.web>membership>providers section of the `web.config` file:

```
      <add name="crm" type="CRMSecurityProvider.CRMMembershipProvider, CRMSecurityProvider"
readOnly="false" connectionStringName="CRMConnString" applicationName="sitecore"
minRequiredPasswordLength="1" minRequiredNonalphanumericCharacters="0"
passwordStrengthRegularExpression=".*" requiresUniqueEmail="false"
passwordFieldName="new_sitecorepassword" autoCreatePasswordField="false" />
```

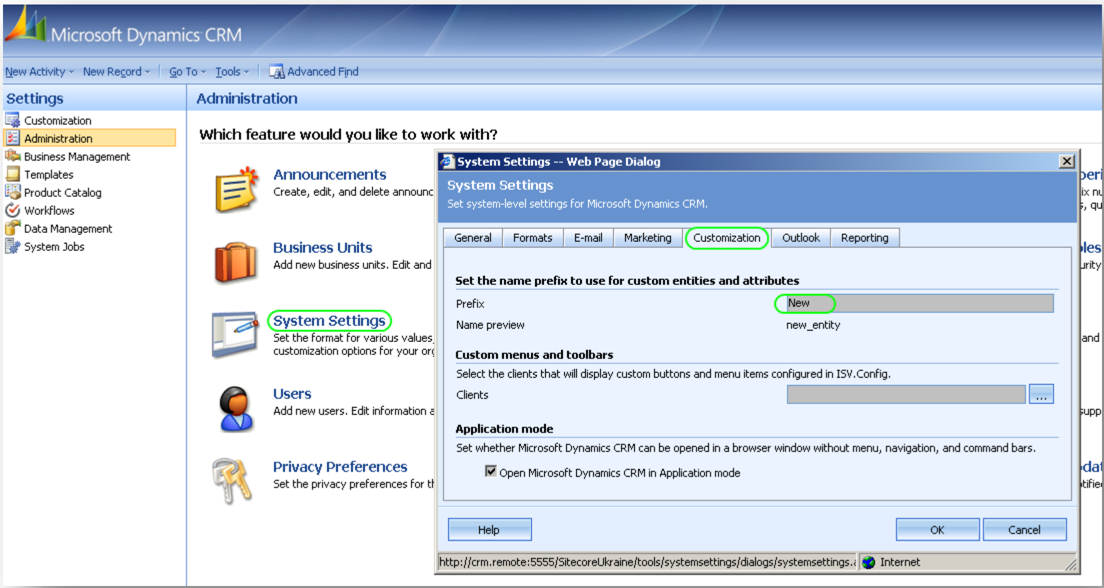
You can change the attribute's values or use the default values (the attributes are not specified).

Attribute	Description
name	The provider name. In general, this can be any string value, unique within a set of membership providers. It is crm in our example.
type	The full name of the provider class.
readOnly	Defines whether the provider works in read-only or read-write mode. The default value is true (read-only mode).
connectionStringName	The name of the connection string. It is `CRMConnString` in our example.
applicationName	A standard attribute of any provider. It defines the area of visibility of the provider data. It should be `sitecore` in our example. See the MSDN documentation for details
minRequiredPasswordLength	A minimum number of characters required for the user password. The default value is 7.
minRequiredNonalphanumericCharacters	A minimum number of non-alphanumeric characters required for the user password. The default value is 1.
requiresUniqueEmail	Defines whether a provider requires unique emails to be set for each user. By default it is set to `false`.
passwordFieldName	Defines the field in a CRM contact that will be used for storing the password.
autoCreatePasswordField="false"	Defines whether the field for storing the password should be created automatically, if it doesn't exist. Supported only by CRM v4 and v5.
passwordStrengthRegularExpression	Regular expression used to evaluate the password. The default value is no regular expression.

To disable the reset password option, you must specify the `enablePasswordReset` attribute and set its value to `false` in the previous configuration.

Leave `passwordFieldName` empty if the password isn't needed (validating/logging in will not be used for represented users).

If the password field is configured to be created automatically (`autoCreatePasswordField="true"`), the `passwordFieldName` value should start with the prefix configured to be used for the custom entities in CRM for example, "**new**_sitecorepassword". You can find the valid value of the prefix in the CRM Settings, Administration section:



Configuring the Role Provider

Add the following element to the `system.web<roleManager>` providers section of the `web.config` file:

```
        <add name="crm" type="CRMSecurityProvider.CRMRoleProvider, CRMSecurityProvider"
connectionStringName="CRMConnString" readOnly="false"/>
```

The attributes of this element are described in the previous table and have the same meaning for the role provider.

Configuring the Profile Provider

Add the following element to the system.web>profile>providers section of the `web.config` file:

```
        <add name="crm" type="CRMSecurityProvider.CRMProfileProvider, CRMSecurityProvider"
connectionStringName="CRMConnString" readOnly="false"/>
```

Add the following profile properties to the system.web>profile>properties section of the `web.config` file:

```
        <add type="System.String" name="Country" customProviderData="crm|address1_country"/>
        <add type="System.String" name="City" customProviderData="crm|address1_city"/>
        <add type="System.String" name="BirthDate" customProviderData="crm|birthdate"/>
        <add type="System.String" name="DoNotEmail" customProviderData="crm|donotemail"/>
```

This list contains the most common properties. The list can be expanded with other properties.

Adding a Profile Property

Define the properties by adding the following element to the properties section:

```
        <add type="System.String" name="<property_name>"
customProviderData="crm|[contact_attribute_schema_name]"/>
```

Attribute	Description
Name	The unique name of the property.
Type	The .NET type of the property. This is the type that the property will have in the ASP.NET environment.
customProviderData	Any data required for the provider serving this property. In this case, crm means that this property is to be handled by the crm provider. The contact_attribute_schema_name specifies the name of the corresponding CRM contact entity attribute. The crm provider expects the pipe separator (|) between the attribute parts. crm_type element The earlier module versions require the type of the attribute to be specified. Now it is defined automatically by the module. If the third element is defined, it will be ignored.

The CRM profile provider supports the following types of the CRM attributes:

- the ‘nvarchar’ and ntext CRM types
- the ‘bit’ CRM type
- the ‘datetime’ CRM type
- the ‘float’ CRM type
- the ‘int’ CRM type
- the ‘money’ CRM type
- the ‘picklist’ CRM type

Note
This is a full list of the supported CRM types no more types are supported in the current module.

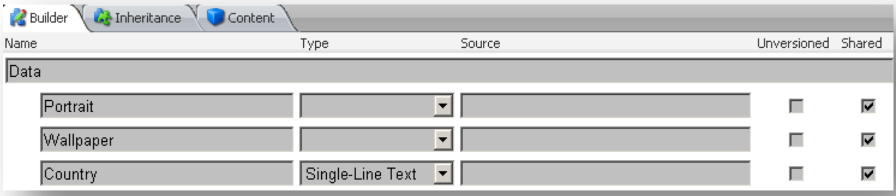
Extending the Sitecore Profile Item Template

After you have defined the custom properties in the `web.config` file, you should extend the Sitecore template to make the properties accessible from the Sitecore CMS security applications.

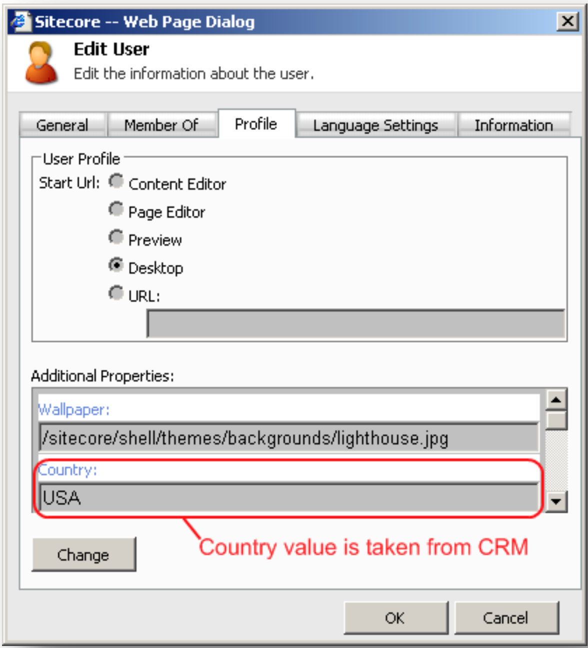
To make the properties accessible:

1. Start Sitecore CMS and login as an administrator and switch to the core database.
2. Open the **Content Editor**, and in the content tree, browse to the `/sitecore/templates/System/Security/CrmContact` template.
3. Add a new field to the template and set its name to exactly the same value as the appropriate property name (for example, Country) in the `web.config` file. Set the type to **Single-Line**

Text and select the **Shared** check box.



From now on when you edit any user from the CRM system in the Sitecore **User Manager**, you will see the added profile property in the **Edit User** dialog box in the **Profile** tab with the value taken from the corresponding contact in the CRM system.



Note:
These instructions assume that the Sitecore user uses the default profile item — `sitecore/system/Settings/Security/Profiles/CrmContact` item — which is used when no profile item is defined for the user.

2.2.4 Adding Domain-Provider Mappings

Open the `web.config` file, and browse to the `<switchingProviders>` element in the `<sitecore>` section. This section contains three groups: `<membership>`, `<roleManager>` and `<profile>`.

1. Add the following line to the `<membership>` group (the order is not important):

```
<provider providerName="crm" storeFullNames="false" wildcard="%"
domains="crm" />
```

2. Add the following line to the `<roleManager>` group (the order is not important):

```
<provider providerName="crm" storeFullNames="false" wildcard="%"
domains="crm" />
```

3. Add the following line to the `<profile>` group (it must come before the sql definition in this group):

```
<provider providerName="crm" storeFullNames="false" wildcard="%"
domains="crm" />
```

The only difference the order makes is to the order that the users/roles are displayed in the Sitecore CMS 6 security tools. For instance, if you put the *crm* membership mapping before the *sql* one, you'll see the represented CRM users in **User Manager** before the default Sitecore CMS ones.

2.2.5 Configuring Unique Key Property

The `/App_Config/Include/crm.config` file contains the `Crm.UniqueKeyProperty` setting. The setting defines the CRM contact field which will be used as the unique key for the represented Sitecore users. Any string field can be used as the unique key.

This means that:

- The value of the field is used as a represented user name.
- If there are a few contacts with the same value in the field, only the first one is represented.
- The value of the field is used as the login name if validating/logging in is used.

Note
If you leave the setting empty, the `emailaddress1` is used as the unique key.

2.2.6 Other Settings

The provider uses caching of the users by default to improve performance and reduce the number of requests to the CRM system. By default, the size of the cache is 2MB.

If you want to change the size or disable the CRM users' cache, add the following setting to the configuration file and set the required value ("0" will disable the cache):

```
<setting name="Crm.Caching.UserCacheSize" value="2MB" />
```

You can also configure the lifetime of the user entries in the cache using the following setting:

```
<setting name="Crm.Caching.UserCacheLifetime" value="00:02:00" />
```

There are also caches for members of a role and for roles a user is member of:

```
<setting name="Crm.Caching.MemberOfCacheSize" value="2MB" />
<setting name="Crm.Caching.MembersCacheze" value="2MB" />
```

One more cache is for the CRM metadata. This is used by the profile provider to store the data about the property types:

```
<setting name="Crm.Caching.MetadataCacheSize" value="2MB" />
```

The Profile Prefetch setting determines the list of properties that are loaded together with a CRM contact. After the properties have been loaded, they are cached. Sitecore doesn't send any requests to the CRM server if everything that Sitecore needs is loaded.

```
<setting name="Crm.Profile.Prefetch" value="" />
```

There is a setting that determines how much information will be sent to the log file in the course of a CRM request. Valid values are None, Error, Warning, Info, Details. The default value is Warning.

```
<setting name="Crm.LoggingLevel" value="Warning" />
```

Detailed logging level adds timing messages to the log. You can see how much time each provider action and its parts take.

Chapter 3

User Guide

The module supports read-only and read-write modes, which is regulated by the *readOnly* attribute in the provider definition.

This chapter describes basic scenarios for the use of security objects by the Microsoft Dynamics CRM Security Provider module for both modes.

This chapter contains the following sections:

- Read-Only Mode
- Read-Write Mode

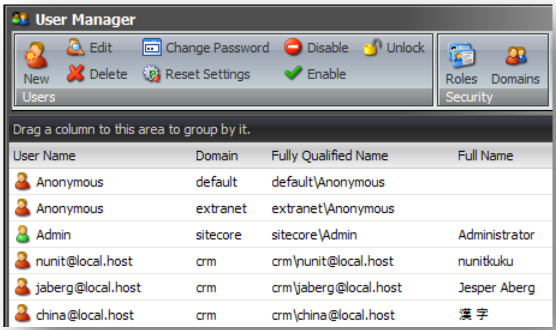
3.1 Read-Only Mode

Read-only mode is used to connect to the Microsoft CRM system to only read the data (except for some special cases which will be described later in this chapter). This means that the module does not allow you to update the data in the CRM system from Sitecore.

For more information about the *readOnly* attribute, see the section *Configuring ASP.NET Security Providers*.

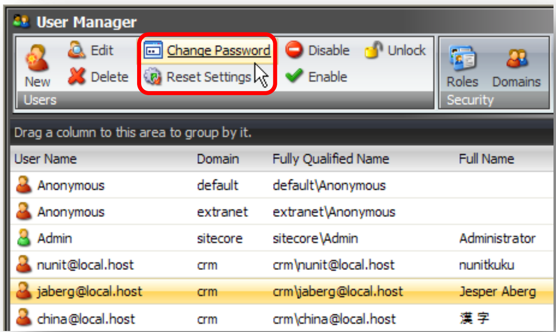
3.1.1 Managing Users

Once you have configured the membership provider, you can start the **User Manager** and see the list of users taken from the CRM contacts.



The module only transfers active contacts from the CRM system. The name of a represented user is the value of the contact field which is configured as a ‘unique key’. The Fullname and Email core properties are published with the values of “fullname” (firstname + lastname) and “emailaddress1” contact fields respectively. The Comment core property is published with the value of the “description” field of the contact.

If the provider is configured to store the password for represented users, you can reset/change the user password.

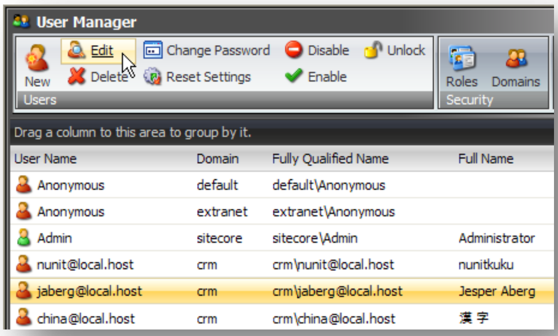


You cannot create, edit or delete users in read-only mode. The provider also does not support disabling/enabling, locking/unlocking users.

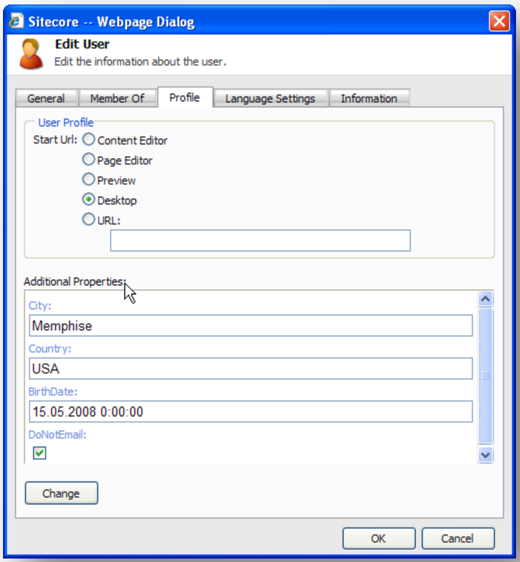
3.1.2 Viewing Profile Properties

After you configure the profile provider, you can represent the contact entity fields from the CRM system to Sitecore user profile object.

To see the represented properties, in the **User Manager**, select a user and then click **Edit**.



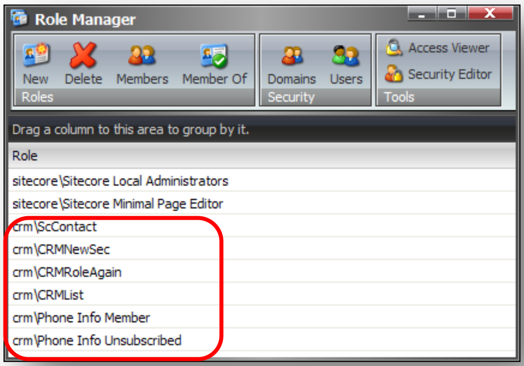
In the **Edit User** dialog box, in the **Profile** tab, you can see a list of properties including those with values from the CRM contact entity fields.



Note
In read-only mode, you cannot update the values of the properties.

3.1.3 User Membership

If the role provider is configured, the marketing lists with the contact member type from the CRM system are represented in Sitecore as Sitecore roles.



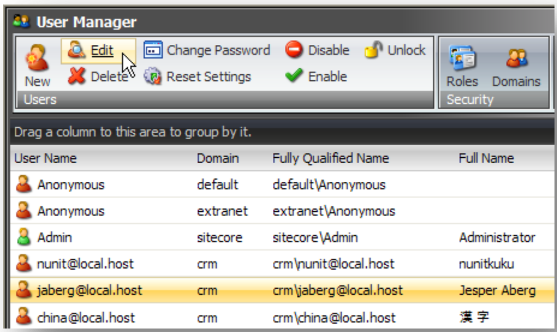
Members of the represented roles correspond to the marketing list members that are represented as Sitecore users.

Notice that in read-only mode you can change membership (add or remove contacts from the marketing list) of the represented users/roles. Therefore the CRM user that will be used to contact the CRM system must have appropriate permissions.

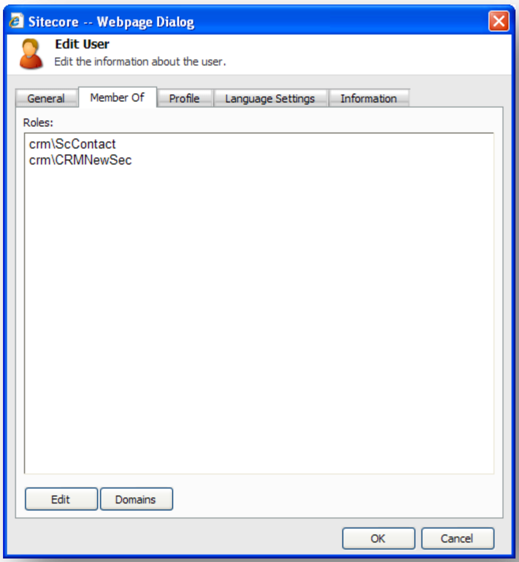
You can change the user membership, in the **User Manager** and in the **Role Manager**.

To change the user membership in **User Manager**:

1. Select the user whose membership you want to change.
2. In the **Users** group, click the **Edit**.

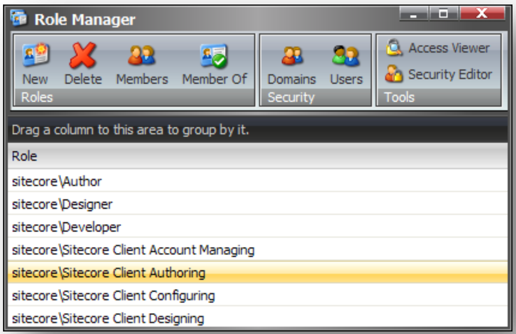


3. Click the **Member** tab and manage the user's roles.

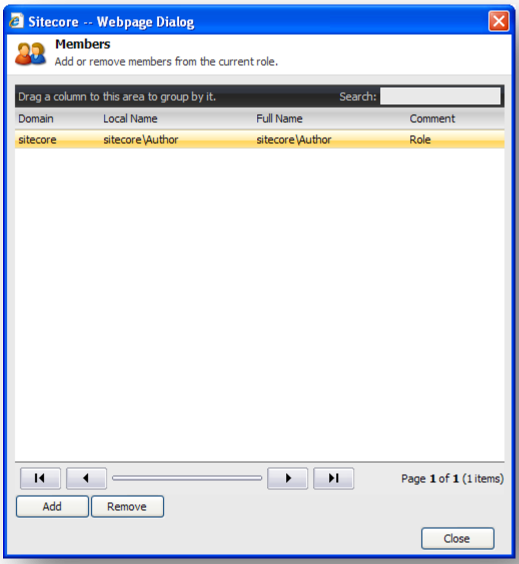


To change user membership in the Role Manager:

1. Open **Role Manager**.



2. Select a role and click **Members**.



Here you can manage members of the role.

Here are some important notes:

- You CAN change membership in read-only mode.
- You can add any CRM user/role to the roles from other domains.
- The CRM role can contain any role from other domain.
- The CRM role can't contain users from other domain; in this case you need to use Roles-In-Roles mechanism: create role in other domain, add there users you want to be members of the CRM role, add created role to the CRM role.

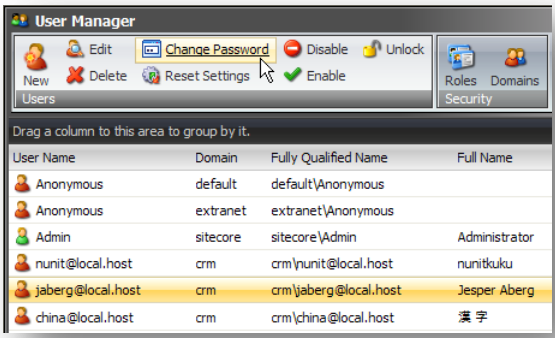
3.1.4 Validating/Logging in Represented Users

If you have configured the module to store the password — the field that stores the password is defined and exists — you can set the password for a represented user to access Sitecore. The field stores the hash value of the password. You cannot retrieve the password. This means that you can only reset and change the password.

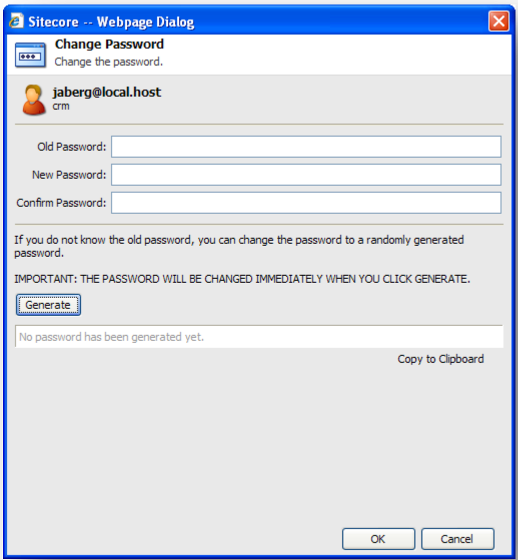
Newly represented users do not have any password; and you therefore need to reset their password.

To reset a user password:

1. Open the **User Manager**.



2. Select the user that you want to edit and click **Change Password.**



3. In the **Change Password** dialog box you can see that all the fields are empty. You must generate a password.
4. Click **Generate**.
5. After you generate a random password, you must reset it in the **Change Password** dialog box.

Note
You can reset/change password in read-only mode. The CRM user that connects to the CRM system must have the appropriate permissions.

Note
If the user who doesn't have a password tries to login the warning message is added to the log files.

3.1.5 Read-only Mode Summary

Here is a list of the actions that you can perform in read-only mode:

- You can view contacts from the CRM system as Sitecore users in the **User Manager**.

- You can view the contact field values in a represented user profile object.
- You cannot add, edit or remove CRM contacts from Sitecore.
- You can view active marketing lists with the contact member type from the CRM system as Sitecore roles in Role Manager.
- You can change the members of the marketing lists from Sitecore in the read-only mode.
- The users can have passwords and these passwords can be reset or changed in read-only mode.

3.2 Read-Write Mode

When the module is configured to use read-write mode, you can update data in the Microsoft CRM system directly from Sitecore.

To configure the module to read-write mode, add the `readOnly=false` attribute to membership, roles and profile providers. For more information about adding the `readOnly=false` attribute to membership, roles and profile providers, see the section *Configuring ASP.NET Security Providers*.

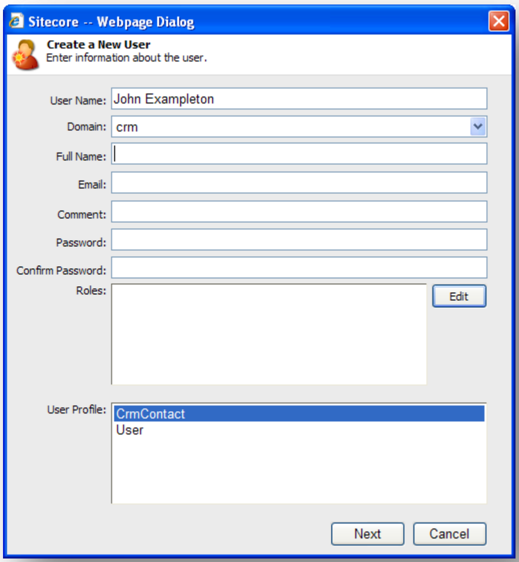
3.2.1 Managing Users

Read-write mode allows you to create, edit, and delete represented users from Sitecore. You can perform all the standard actions to a represented user and the changes are immediately applied to the contact entity in the CRM system.

Creating a CRM Contact from Sitecore

To create CRM contacts from within Sitecore:

1. Open the **User Manager** and in the **Users** group, click **New** to open the **Create a New User** dialog box.
2. In the **User Name** field, enter a user name.



   The **User Name** field is used as the contact property configured to be the unique key. For more information, see section *Configuring Unique Key Property*.

   If you type an email as the user name, you must configure the regular expression for the account name. For more information, see the section *Adding a New* Domain.

3. Select the domain which is served by the module.
4. In the **Full Name** field, type the full name of the contact.

   It is parsed in the following way: the first word before the space character is the firstname of the contact, the other part is the lastname of the contact.

   If the fullname contact property is configured to be used as unique key, the Full Name should have the same value as User Name. If they are different, there might be some unpredictable issues. The new contact will be created with the fullname from Full Name field (not User Name).

5. Type the contact email address in the **E-mail** field. This is used as emailaddress1 field value. This field is mandatory due to the Create User form validation.

   If the emailaddress1 contact property is configured to be used as unique key, the User Name and Email field must have the same value, otherwise the user cannot be created.

6. Type the contact description in the **Comment** field. This field is optional.

7. Type the password in the **Password** and **Confirm Password** fields. You need to type something because of validator. The value is ignored if the module isn't configured to store the password.

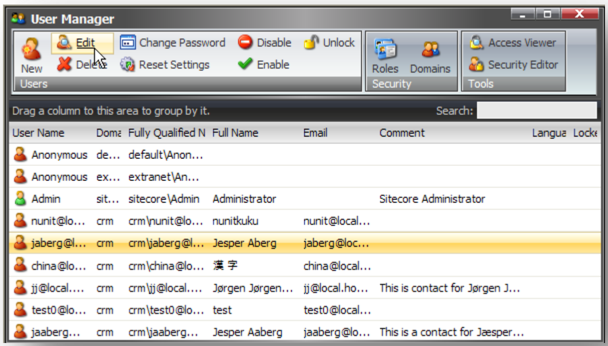
8. Select the *CrmContact* profile.

After you click **Next**, the contact will be created in the CRM system. The corresponding user will be shown in **User Manager** or the error message will be shown in the dialog.

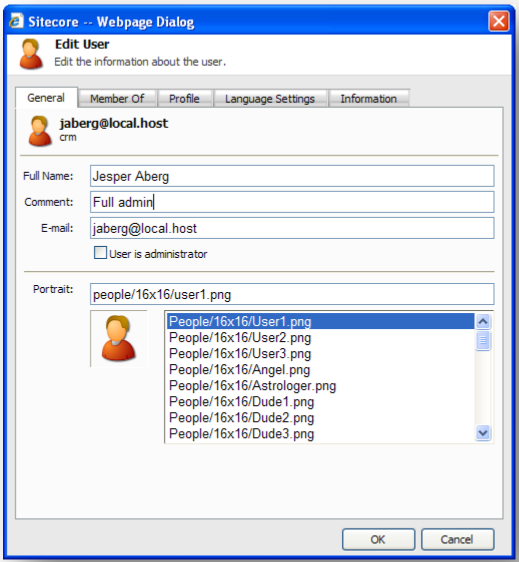
Editing a CRM Contact from Within Sitecore

To edit a CRM contact:

1. Open the **User Manager**.



2. Select a user from the domain which is served by the module and click **Edit**.



3. You can modify the **Full Name**, **E-mail** and **Comment** fields. They will update the following fields of the contact in the CRM system:

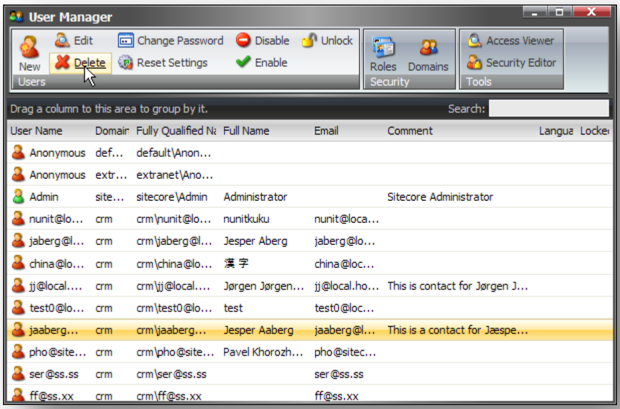
- fullname (firstname, lastname)
- emailaddress1
- description

Note
If either the emailaddress1 or the fullname contact property is configured to be used as a unique key and you change either the **Email** or **Full Name** field within Sitecore, you also rename the CRM user. If you do this, all the Sitecore data associated with the user (security permissions, role membership from other domains) will be missing. If you change the user information back again, the data is restored.

Removing CRM Contacts from Sitecore

To delete a CRM contact:

1. Open the **User Manager**.
4. Select the user that you want to delete from the domain that is served by the module.



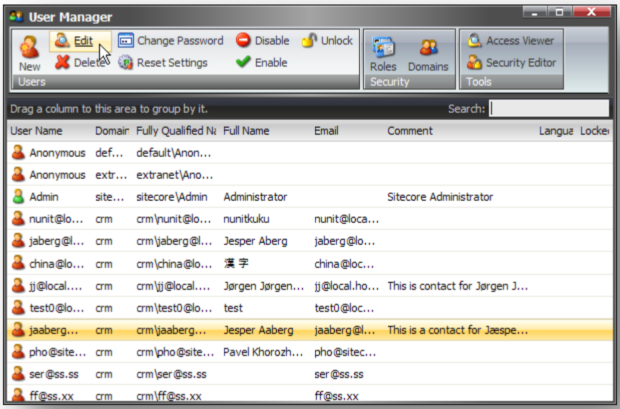
2. Click **Delete**.

Note
When you use Sitecore to remove a user that has a corresponding contact in the CRM system, the contact isn’t actually removed from CRM. The state of the corresponding contact in CRM system changes to “Inactive” instead. The contact won’t be visible in Sitecore after this.

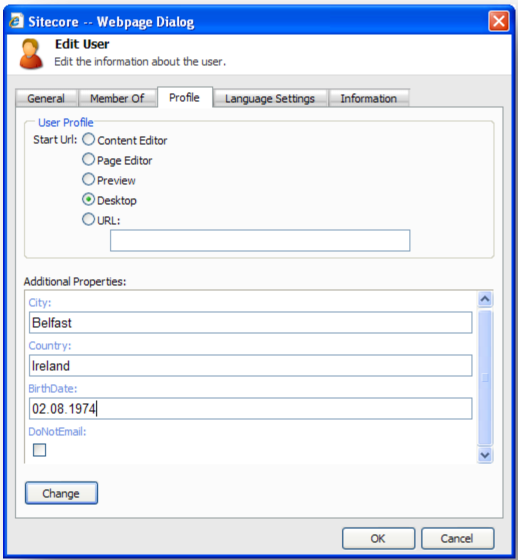
3.2.2 Editing Profile Properties

To edit the profile property:

1. Open the **User Manager**.



2. Select the user whose profile property you want to edit and click **Edit.**



3. Click the **Profile** tab and update the profile property that you want to edit.
4. Click **OK**. The changes are saved to the contact on the CRM server.

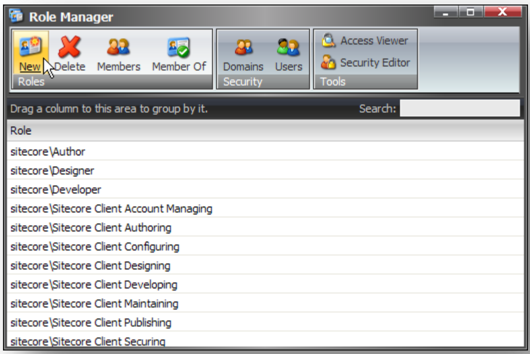
3.2.3 Managing Roles

You can add and remove CRM marketing lists from Sitecore.

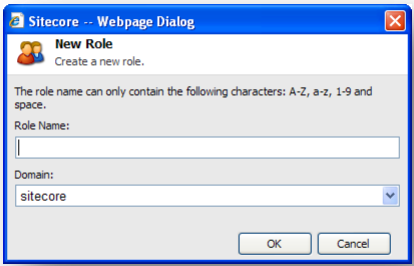
Adding a New Marketing List

To add a new CRM marketing list from Sitecore:

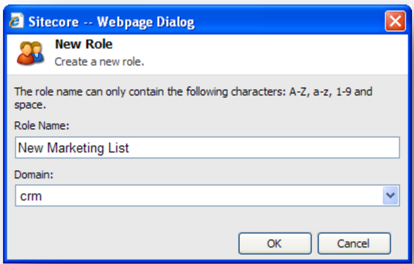
1. Open the **Role Manager**.



2. Click **New** and the **New Role** dialog box is displayed.



3. In the **Role name** field, enter a name for the marketing list and in the **Domain** dropdown list, select the domain which is served by the module form.



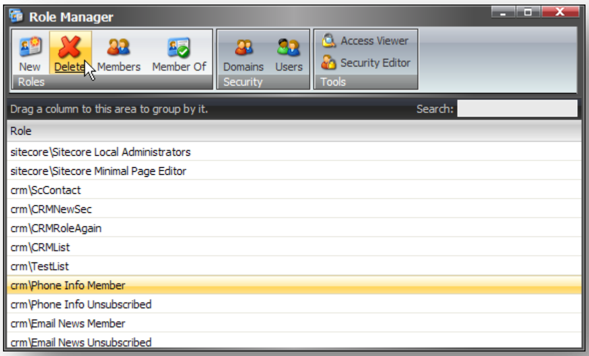
4. Click **OK**.

Note
In the CRM system a new marketing list is created with *a contact member type*.

Removing a Marketing List

To remove a marketing list:

1. Open the **Role Manager**.



2. Select a role from the domain which is served by the module.
3. Click **Delete**.

Note
When you use Sitecore to remove a role that has a corresponding marketing list in the CRM system, the marketing list isn't actually removed from the CRM. The state of the corresponding marketing list in CRM system changes to "Inactive" instead. The marketing list won't be visible in Sitecore after this.

3.2.4 Read-Write Mode Summary

Read-write mode includes all read-only mode features.

In read-write mode you can:

- You can create, edit, and remove CRM contacts from Sitecore.
- You can update profile properties which will update the field values of the CRM contact.
- The other features work the same as in read-only mode.

Chapter 4

FAQ

This chapter contains some of frequently asked questions and the answers to them.

4.1 FAQ

1. Q: I can't see entire entities from the CRM system in Sitecore.

   A: The provider:

   - Only shows the active entities from the CRM system;
   - Only shows selected entities (if you have a few contacts with the same unique key, only the first one will be represented).

2. Q: I can't see all the marketing lists from the CRM system in Sitecore.

   A: The provider:

   - Only shows the active marketing lists from CRM;
   - Only shows the marketing lists with the Contact member type. The provider only shows Contacts entities; it shows neither accounts nor leads from CRM.

3. Q: I can't make any changes to the represented entities.

   A: By default, the provider works in read-only mode. It doesn't allow changes to the CRM database.

4. Q: How can I add a user from another domain to a role from the CRM domain?

   A: you cannot do this because there is no appropriate user in the CRM source. You should use the Roles-In-Roles mechanism to make this work.

5. Q: I can't see any changes when I try to disable/enable/unlock a user from the CRM domain.

   A: The Microsoft Dynamics CRM Security Provider module doesn't support these actions.

6. Q: CRM users don't have access to items in the Content Editor although read permission is granted.

   A: You should grant the permission directly to the role the user is a member of (and not to the superimposed role). You can also use the Everyone role to grant this to all users.

7. Q: When I start the User Manager (or open the Select Account dialog box), it takes a lot of time to load. How can I make it load faster?

   A: You may experience some performance issues if your CRM system contains a lot of contacts that need to be represented in Sitecore. Here are a couple of ways to improve the performance of the security applications in Sitecore:

   - Make sure user caching is enabled for the module. For more information, see the section *Other Settings* of this guide.
   - Make sure that manual paging is used by the application grid. To check this, go to the application form file. It is the `/UserManager/UserManager.aspx` file for User Manager and the `/SelectAccount/SelectAccount.xaml.xml` file for the Select Account dialog box.In the `/sitecore/shell/Applications/Security` folder find the Grid element with "Users" ID, check the element contains ManualPaging attribute and its value is "true". If the attribute isn't present or its value is "false", you should update the application (correcly) to use manual paging.