You use the security model in Microsoft Dynamics CRM to protect the data integrity and privacy in a Microsoft Dynamics CRM organization. The security model also promotes efficient data access and collaboration. The goals of the model are as follows:

* Provide a multi-tiered licensing model for users.
* Grant users access that allows only the levels of information required to do their jobs.
* Categorize users and teams by security role and restrict access based on those roles.
* Support data sharing so that users can be granted access to objects they do not own for a one-time collaborative effort.
* Prevent access to objects a user does not own or share.

You combine business units, role-based security, record-based security, and field-based security to define the overall access to information that users have in your Microsoft Dynamics CRM organization.

Team:

User and team management is the area of Microsoft Dynamics CRM where you can create and maintain user accounts and profiles.

A *user* is any person who works for a business unit who uses Microsoft Dynamics CRM. Each user has a user account. All users must be associated with only one business unit. This association controls which customer data the user will have access to. Included in the user's account is information such as the user's telephone numbers, email address, and a link to the user's manager. Each user has privileges and rights to manage their own personal settings. Each user corresponds to a user in the Active Directory for that organization. When you create a user, you must assign the user to at least one security role. Even if the user is part of a team that has assigned roles, the user should be assigned to a role.

A *team* is a group of users. Teams let users across an organization collaborate and share information.

A *team* is a group of users. Individual users are called team members. A team reports to a *business unit* and team members may be from different business units. Creating a team gives greater flexibility for sharing business objects and collaboration. A team can own records or the records can be shared with the team.

With *owner* teams or *access* teams, you can easily share business objects and collaborate with the users across business units in Microsoft Dynamics CRM. A team belongs to one business unit, but it can include users from other business units. A user can be associated with more than one team.

An owner team owns records and has security roles assigned to the team. The team’s privileges are defined by these security roles. In addition to privileges provided by the team, team members have the privileges defined by their individual security roles and by the roles from other teams in which they are members. A team has full access rights on the records that the team owns.

* Team members require different access rights on records  
  you can share a record with several Access teams, with each team providing different access rights on the record. For example, one team is granted only the Read access right on the account, while another team is granted the Read, Write and Share access rights on the same account.
* A unique set of users requires access to a single record without having ownership of the record

User can be assigned any security role which exists in their business unit.

Security roles are linked with the user business unit to calculate what records the user can access.

Their primary function was to allow the sharing of records and/or views with multiple people in a bulk fashion rather than having to share records one by one.

The security model in CRM makes it very easy to create permissions for users in the same Business Unit (BU) to see and collaborate on each other’s records. You simply give users BU scope to the entity and privilege(s) in question, such as a Contact’s *read* and *update* privileges.

For example, a security role with BU scope on Contacts for *read* and *update* will let anyone with this role update all of the contacts in one’s own BU (meaning the Contact Owner belongs to the same BU as the user).

You can create Teams to allow for collaboration on records “among users of different BUs” that couldn’t be solved with the permissions available in the previous security model.

Without Teams, the only way to let people in sibling BUs share/collaborate on records is to give the users “Organization” level permissions, which means access to all records in all BUs which is not usually desirable.

**Record Access Implications**

An individual CRM user has permissions that are accumulated across all security roles to which they are assigned; and accumulated across all inherited security roles based on the Team(s) on which they are Members. So if a User has a personal security role that doesn’t provide the permission to *read* accounts, but this same User belongs to a Team with a security role that DOES have the Account *read* permission, then the User will be allowed to *read* accounts. In other words, the privileges are *additive* across user security roles and inherited team-based security roles.

**Business Unit Scope on Team Security Role**

When you assign a team permission at the BU scope level, this means the user’s privileges will include all of the records of that type that are owned by users in the BU to which the team belongs. For example, assume the Advisors team is part of the Consultants Unlimited (aka CU) Business Unit and the Advisors team has a security role with contact update permissions of BU scope. And assume Earl and Jamie are in the Associate Advisors (aka AA) Business Unit, but they are still members of the Advisors team. Both Earl and Jamie will inherit the permissions of the Advisors team, therefore they will be able to update contacts owned by any user in the CU Business Unit, along with any contact owned by the Advisors team.

This brings up a point in which the team architecture still leaves something to be desired. Suppose you wanted to give Earl and Jamie privileges to update contacts owned by either Advisors or Associate Advisors. You would need to create 2 teams, one that resides in each of those Business Units, and make Earl and Jamie members of both teams.

*Insight: it would have been great to have the ability to assign security roles from different BUs to the one team, so that one team could have a security role from Advisors with BU privileges, and also a security role from Associate Advisors with BU privileges, which would give the team members “all” privileges from the security roles of both BUs, but this isn’t possible in CRM 2011. A team can only have security roles from the BU in which it resides. Starting in CRM 2013 there is a new concept of “Access Teams” which may solve this dilemma.*

**Parent: Child Business Unit Scope on Team Security Role**

This one requires some explaining up front for those who aren’t familiar with what this privilege means, since its name is anything but intuitive. This privilege level grants a user access to any records in the BU in which he resides, and those in any BU below it. For example, if a user in the Consultants Unlimited BU had Parent: Child BU privileges to read contacts, he could see the contacts owned by:

· Any user of the Consultants Unlimited BU

· Any user of the Advisors BU

· Any user of the Associate Advisors BU

· Any user of Adv Farm 1 or Adv Farm 2 BUs

If a CRM Team has this privilege level, it grants the specified privileges from the security role to records owned by any user of the BU that owns the CRM Team, and to any records owned by users of child BUs. For example, if the Associate Advisors team were to be assigned to the Advisors BU, and if the team’s security role had Parent: Child BU privileges to read accounts, then Earl and Jamie could read accounts owned by:

· Any user of the Advisors BU

· Any user of the Adv Farm 1 or Adv Farm 2 BUs

Which brings us to the final points about data access: Using Views.

**Access level** controls what the user can see (read)/edit

**Privilege** controls what you can do to the entity

You cannot delete or disable the root business unit

You can rename

**Disabling Business Units**

You need to know what happens when you disable a business unit, what happens to all the users who have that business unit as their default business unit?

When you disable a business unit, it also disables all the child business units below it (and thus all the users who those business units as their default business unit).

none of the data is affected by disabling business units, its only the users who cannot then log in but it is important to take into account all the child business units will also be disabled.

The users are not disabled but cannot login into CRM whilst the business unit is disabled.  As soon as the business unit is enabled they will be able to log into CRM again.

If you want to delete the business unit then you will need to change all the users/teams that are assigned to that business unit.  You also need to disable the business unit before you delete it.

We have people work in more than one business unit and wear different 'hats' depending on what business unit they are representing.

For example a Senior Manager in marketingbusiness unit may resign, and in the interim a Finance Director from Finance Business unit may take up his job until a new person can be found.

The FD is assigned the business unit of Finance but he now also works in marketing business unit.

How can this be accommodated in Dynamics CRM?

If a user who was working in Finance needed to work in Marketing, the classical answer would be to move up them up the hierarchy into the root where they have access to all the children (assuming their security role gives them access to child BUs). However in this case that also gives them access to Sales & Service which maybe undesirable.

Teams are a newer feature which allow you work in multiple business units without having to exist in the root business unit (or have organization wide permissions). By adding the user from Finance to the Marketing Team they get access to Marketing and Finance, but not Sales & Service.

[When to use access teams](javascript:void(0)):

A unique set of users requires access to a single record without having an ownership of the record.

Access teams are a quick way to share records on a record basis. People get added to the team. CRM manages the team and the user manages the templates to control access and the team members.

So, in CRM 2013 we have the introduction of **Access Teams** where privileges can get set for an individual record level, and since Access Teams don’t have security roles associated with them, the performance implications involved with multiple security roles and changing of the security roles have been eliminated.