



DSP Custom Security Definitions

User Guide

Contents

| | |
|---|-----------|
| Introduction | 1 |
| Notes Regarding User Guide Example..... | 2 |
| Create Custom Security Definition | 3 |
| Step 1: Create Security Definition Data View..... | 3 |
| Step 2: Create Security Definition..... | 4 |
| Step 3: Assign Security Definition Key Columns | 4 |
| Step 4: Create Security Definition User Assignment Stored Procedure | 5 |
| Step 5: Create Security Definition User Unassignment Stored Procedure..... | 5 |
| Step 6: Create webUserDeleted_XXXXXXDel SP and Assign to Event Users - BeforeDelete | 6 |
| Step 7: Create webSecurityRoleUserAdded_XXXXXXAdd SP and Assign to Event Security Role Users - AddUsers..... | 7 |
| Step 8: Create webSecurityRoleUserDeleted_XXXXXXDel SP and Assign to Event Security Role Users - RemoveUsers . | 9 |
| Step 9: Create webSecurityRoleKeyAdded_XXXXXXAdd SP and Assign to Event Security Role Key Values - AddKeys10 | |
| Step 10: Create webSecurityRoleKeyDeleted_XXXXXXDel SP and Assign to Event Security Role Key Values - RemoveKeys..... | 12 |
| Step 11: Create webSecurityRoleDeleted_XXXXXXDel SP and Assign to Event Security Roles - BeforeDelete | 14 |
| Step 12: Create webUserSecurityKeyAdded_XXXXXXAdd SP and Assign to Event User Specific Keys - AddKeys..... | 16 |
| Step 13: Create webUserSecurityKeyDeleted_XXXXXXDel SP and Assign to Event User Specific Keys - RemoveKeys . | 17 |
| Step 14: Assign webSecurityRoleUserDeleted_XXXXXXDel SP to Event User Roles – UnassignToRole..... | 19 |
| Step 15: Assign webSecurityRoleUserAdded_XXXXXXAdd SP to Event User Roles - AssignToRole..... | 20 |
| Step 16: Assign webSecurityRoleUserDeleted_XXXXXXDel SP to Event User Roles Staging - UnassignToRole..... | 21 |
| Step 17: Assign webSecurityRoleUserAdded_XXXXXXAdd SP to Event User Roles Staging – AssignToRole..... | 22 |
| Step 18: Assign webSecurityRoleUserAdded_XXXXXXAdd SP to Event Copy User - AssignToRole | 23 |
| Step 19: Assign webUserSecurityKeyAdded_XXXXXXAdd SP to Event Copy User - AddKeys..... | 24 |
| Appendix A - Handling Security Definitions whose Data View has Multiple Key Columns..... | 25 |
| Example for Security Definition with single key column Template04_webSecurityRoleUserAdded_XXXXXXAdd | 25 |
| Example for Security Definition with 2 key column Template04_webSecurityRoleUserAdded_WebAppGroupUserAdd | 26 |
| Appendix B - Handling Security Definitions whose Event Rules Populate Multiple Tables | 27 |

Introduction

DSP version 7.1 introduced Centralized Security Management. This functionality made the day-to-day management of DSP security faster, simpler and ultimately enabled third-party tools to integrate with the DSP to control user provisioning activities.

This manual outlines the process with an example of how to create a custom security definition.

IMPORTANT! Creating custom security definitions for standard DSP delivered applications is not supported. During an upgrade, the associated views and stored procedures will be deleted. Therefore, if you have such a requirement, please contact Syniti Support at <https://support.syniti.com/>; technical experts will advise you of the best course of action.

A key enabler of this functionality was the introduction of Security Definition Events. The DSP was analyzed to understand what actions and events would warrant an associated stored procedure to be called to perform user provisioning tasks. These events and the associated DSP pages from where they are called are defined in the following table:

| Event | DSP System Administration Page |
|---------------------------------------|--------------------------------|
| Users - BeforeDelete | Users |
| Security Role Users - AddUsers | Security Role Users |
| Security Role Users - RemoveUsers | Security Role Users |
| Security Role Key Values - AddKeys | Security Role Key Values |
| Security Role Key Values - RemoveKeys | Security Role Key Values |
| Security Roles - BeforeDelete | Security Roles |
| User Specific Keys - AddKeys | User Specific Keys |
| User Specific Keys - RemoveKeys | User Specific Keys |
| User Roles - UnassignToRole | User Roles |
| User Roles - AssignToRole | User Roles |
| User Roles Staging - UnassignToRole | User Roles Staging |
| User Roles Staging - AssignToRole | User Roles Staging |
| Copy User - AssignToRole | Copy User |
| Copy User - AddKeys | Copy User |

Notes Regarding User Guide Example

The example and associated SQL templates referenced in this User Guide are based on a scenario with the following criteria:

- Security Definition’s Data View has a single key column.
- Security Definition Event Rules populate a single table.
- The user guide example requires users to download the DSP_CustomSecurityDefinition_SimpleTemplate.zip file. To download the file, contact Syniti Support at <https://support.syniti.com>. This zip file contains SQL templates that simplify the Security Definition creation process. The following table provides a matrix of which template to use for each event.

| Priority | Event | Business Rule (Template) |
|----------|---------------------------------------|---|
| 10 | Users - BeforeDelete | Template03_webUserDeleted_XXXXXXDel |
| 20 | Security Role Users - AddUsers | Template04_webSecurityRoleUserAdded_XXXXXXAdd |
| 30 | Security Role Users - RemoveUsers | Template05_webSecurityRoleUserDeleted_XXXXXXDel |
| 40 | Security Role Key Values - AddKeys | Template06_webSecurityRoleKeyAdded_XXXXXXAdd |
| 50 | Security Role Key Values - RemoveKeys | Template07_webSecurityRoleKeyDeleted_XXXXXXDel |
| 60 | Security Roles - BeforeDelete | Template08_webSecurityRoleDeleted_XXXXXXDel |
| 70 | User Specific Keys - AddKeys | Template09_webUserSecurityKeyAdded_XXXXXXAdd |
| 80 | User Specific Keys - RemoveKeys | Template10_webUserSecurityKeyDeleted_XXXXXXDel |
| 90 | User Roles - UnassignToRole | Template05_webSecurityRoleUserDeleted_XXXXXXDel |
| 100 | User Roles - AssignToRole | Template04_webSecurityRoleUserAdded_XXXXXXAdd |
| 110 | User Roles Staging - UnassignToRole | Template05_webSecurityRoleUserDeleted_XXXXXXDel |
| 120 | User Roles Staging - AssignToRole | Template04_webSecurityRoleUserAdded_XXXXXXAdd |
| 130 | Copy User - AssignToRole | Template04_webSecurityRoleUserAdded_XXXXXXAdd |
| 140 | Copy User - AddKeys | Template09_webUserSecurityKeyAdded_XXXXXXAdd |

For more advanced scenarios, refer to following appendices:

- [Appendix A - Handling Security Definitions whose Data View has Multiple Key Columns](#)
- [Appendix B - Handling Security Definitions whose Event Rules populate multiple tables](#)

Create Custom Security Definition

When creating custom security definitions, the recommendation is that they reside in non-delivered data sources or WebApps, even if the events are acting on tables that are part of delivered WebApps.

Step 1: Create Security Definition Data View

A Security Definitions Data View provides the list of allowable key values that can be assigned to a role or user.

| Security Definitions | | × FILTER APPLIED | | webap | ? | ⚙️ | |
|------------------------------------|-------------------------------------|---------------------|-------------|-----------------|---|--------|--|
| <input type="button" value="Add"/> | <input type="button" value="Edit"/> | | | | | 1 rows | |
| SECURITY DEFINITION NAME | DATA SOURCE ID | DATA VIEW | DESCRIPTION | SYSTEM PROVIDED | | | |
| CranSoft.WebAppUser | CranSoft | webWebAppSec | WebApp | | | | |
| | | | | | 1 | | |

The view must:

- Use the following naming convention: web*sec
- Include at least 1 key field associated with the items returned
- Include a descriptive name (FriendlyName) for the value.

The following is an example of a Security Definition Data View:

```

Definition webWebAppSec ?
Database CranSoft
Type View
Definition
CREATE VIEW dbo.webWebAppSec
AS
SELECT WebAppID,
       WebAppName AS FriendlyName
FROM   dbo.WebApp
WHERE  ( WebAppID = '9ac95b56-f805-4fc4-9d9b-590b05cd0e7c' )
    
```

Step 2: Create Security Definition

In this step, create a custom security definition.



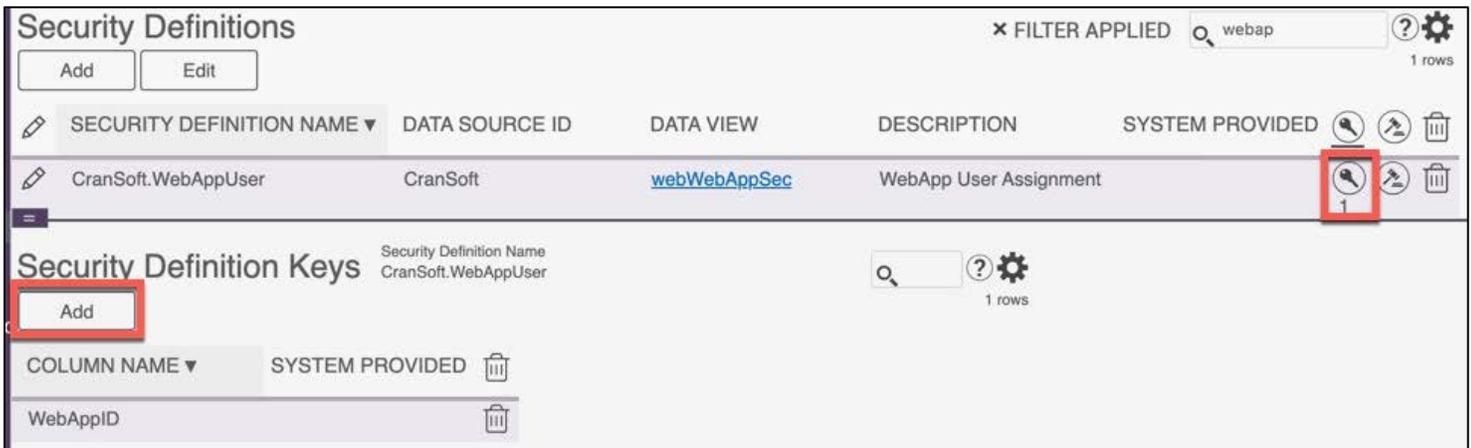
| Security Definitions | | × FILTER APPLIED | | webap | 1 rows |
|--|----------------|------------------------------|------------------------|-----------------|--------|
| <input type="button" value="Add"/> <input type="button" value="Edit"/> | | | | | |
| SECURITY DEFINITION NAME | DATA SOURCE ID | DATA VIEW | DESCRIPTION | SYSTEM PROVIDED | |
| CranSoft.WebAppUser | CranSoft | webWebAppSec | WebApp User Assignment | | 1 |

To create a custom security definition:

1. Select **Admin > Security Management > Security Definitions** in the *Navigation* pane.
2. Click **Add**.
3. Enter a name for the definition in the **Security Definition Name** field.
4. Select the data source in which the data view and event rules exist from the **Data Source ID** list box.
5. Select the data view created in [Step 1: Create Security Definition Data View](#) from the **Data View** list box.
6. Enter a description of the definition in the **Description** field.
7. Click **Save**.

Step 3: Assign Security Definition Key Columns

In this step, add the key columns associated with Security Definition Data View to the definition.



| Security Definitions | | × FILTER APPLIED | | webap | 1 rows |
|--|----------------|------------------------------|------------------------|-----------------|--------|
| <input type="button" value="Add"/> <input type="button" value="Edit"/> | | | | | |
| SECURITY DEFINITION NAME | DATA SOURCE ID | DATA VIEW | DESCRIPTION | SYSTEM PROVIDED | |
| CranSoft.WebAppUser | CranSoft | webWebAppSec | WebApp User Assignment | | 1 |

| Security Definition Keys | | Security Definition Name | 1 rows |
|------------------------------------|-----------------|--------------------------|--------|
| <input type="button" value="Add"/> | | CranSoft.WebAppUser | |
| COLUMN NAME | SYSTEM PROVIDED | | |
| WebAppID | | | |

1. Select **Admin > Security Management > Security Definitions** in the *Navigation* pane.
2. Click **Add** on the *Security Definition Keys* page.
3. Select the name of the key column from the **Column Name** list box.

4. Click **Save**.
5. Repeat if multiple key columns are used.

NOTE: Refer to [Appendix A - Handling Security Definitions whose Data View has Multiple Key Columns](#) for more information.

Step 4: Create Security Definition User Assignment Stored Procedure

In this step, create a common stored procedure that adds a user to the relevant application user table. This procedure is called from all subsequent procedures that are related to events that are intended to grant user access.

To create stored procedure webSecurity_#SEC_DEFINITION#Add using SQL template Template01_webSecurity_XXXXXXAdd.sql:

1. Open the script **template Template01_webSecurity_XXXXXXAdd.sql** in MS Server Management Studio (SSMS).
2. Replace **#DATABASE#** with the name of the database.
3. Replace **#SEC_DEFINITION#** with the short name of security definition (for consistency, use the same value on all subsequent stored procedures).
4. Replace **#TABLE#** with the name of the table that the user records needs to be inserted into.
5. Execute the modified SQL script to create the stored procedure.

Step 5: Create Security Definition User Unassignment Stored Procedure

In this step, create a common stored procedure that deletes a user from the relevant application user table. This procedure is called from all subsequent procedures (except the procedure associated with user deletion) that are related to events that are intended to remove user access.

To create the stored procedure webSecurity_#SEC_DEFINITION#Add using the SQL template Template01_webSecurity_XXXXXXAdd.sql:

1. Open the script template **Template01_webSecurity_XXXXXXAdd.sql** in SSMS.
2. Replace **#DATABASE#** with the name of the database.
3. Replace **#SEC_DEFINITION#** with the short name of security definition (for consistency, use same value on all subsequent stored procedures).
4. Replace **#TABLE#** with the name of the table that the user records needs to be inserted into.
5. Execute the modified SQL Script to create the stored procedure.

Step 6: Create webUserDeleted_XXXXXXDel SP and Assign to Event Users - BeforeDelete

In this step, create a stored procedure to delete a user from the application user access table associated with the Security Definition. This procedure simply deletes the user from the associated table(s) without any consideration for other user assignments. Then, assign this procedure to the event Users – BeforeDelete.

To create the stored procedure webUserDeleted_#SEC_DEFINITION#Del using the SQL template Template03_webUserDeleted_XXXXXXDel.sql:

1. Open the script template **Template03_webUserDeleted_XXXXXXDel.sql** in SSMS.
2. Replace the word **#DATABASE#** with the name of the database.
3. Replace the word **#SEC_DEFINITION#** with the short name of security definition (for consistency, use same value on all stored procedures).
4. Replace the word **#TABLE#** with the name of the table that the user records needs to be inserted into.
5. Execute the modified SQL script to create the stored procedure.

To assign stored procedure webUserDeleted_#SEC_DEFINITION#Del to the Users – BeforeDelete event:

1. Click the **Events** icon for the Security Definition being created.
2. Click the **Rules** icon for the **Users - BeforeDelete** event.

The screenshot displays two sections of the application interface. The top section, titled "Security Definitions", contains a table with the following data:

| SECURITY DEFINITION NAME | DATA SOURCE ID | DATA VIEW | DESCRIPTION | SYSTEM PROVIDED |
|--------------------------|----------------|--------------|------------------------|-----------------|
| CranSoft.WebAppUser | CranSoft | webWebAppSec | WebApp User Assignment | |

The bottom section, titled "Security Definition Events", shows events for the "CranSoft.WebAppUser" definition. It contains a table with the following data:

| EVENT |
|--|
| System Administration - Users - BeforeDelete |

3. Enter a value in the **Priority** field.
4. Select the data source from the **Data Source ID** list box.

NOTE: The Data Source ID is typically the same as the Data Source ID registered on the *Security Definitions* page.

5. Select the **webUserDeleted_#SEC_DEFINITION#Del** rule created previously from the **Business Rule** list box.
6. Check the **Active** check box.
7. Enter a comment that describes what the rule does in the **Comment** field.

8. Click **Save**.

The resulting record should look similar to the following:

| Security Definition Event Rules | | Security Definition Name | Event | | |
|--|----------------|------------------------------|--|--|--|
| | | CranSoft.WebAppUser | System Administration - Users - BeforeDelete | | |
| <input type="button" value="Add"/> <input type="button" value="Edit"/> | | | | | |
| PRIORITY | DATA SOURCE ID | BUSINESS RULE | ACTIVE | COMMENT | |
| 10 | CranSoft | webUserDeleted_WebAppUserDel | <input checked="" type="checkbox"/> | When a User is deleted, delete the User from WebAppUser. | |

Step 7: Create webSecurityRoleUserAdded_XXXXXXAdd SP and Assign to Event Security Role Users - AddUsers

In this step, create a stored procedure that assigns a user to all values associated with all Security Definition Key values assigned to the Role the user is being added to. Then, assign the procedure to the event Security Role Users – AddUsers.

To create the stored procedure webSecurityRoleUserAdded_#SEC_DEFINITION#Add using template Template04_webSecurityRoleUserAdded_XXXXXXAdd.sql:

1. Open the script template **Template04_webSecurityRoleUserAdded_XXXXXXAdd.sql** in SSMS.
2. Replace the word **#DATABASE#** with the name of the database.
3. Replace the word **#SEC_DEFINITION#** with the short name of security definition (for consistency, use same value on all stored procedures).
4. Replace the word **#TABLE#** with the name of the table that the user records needs to be inserted into.
5. Replace **Template01_webSecurity_#SEC_DEFINITION#** with the stored procedure created in [Step 4: Create Security Definition User Assignment Stored Procedure](#).

Refer to the following screen shots for an example.

Before:

```
EXECUTE Template01 webSecurity #SEC DEFINITION#Add
    @UserID,
    @KeyName,
    @KeyValue
    @boaUserID
```

After:

```
EXECUTE webSecurity WebAppUserAdd
    @User ID,
    @KeyName,
    @KeyValue
    @boaUser ID
```

- Execute the modified SQL script to create the stored procedure.

To assign the stored procedure `webSecurityRoleUserAdded_#SEC_DEFINITION#Add` to the event `Security Role Users - AddUsers`:

- Click the **Events** icon for the Security Definition being created.
- Click the **Rules** icon for the **Users Add Users** event.

Security Definitions × FILTER APPLIED 1 rows

| SECURITY DEFINITION NAME | DATA SOURCE ID | DATA VIEW | DESCRIPTION | SYSTEM PROVIDED |
|--------------------------|----------------|------------------------------|------------------------|-----------------|
| CranSoft.WebAppUser | CranSoft | webWebAppSec | WebApp User Assignment | |

Security Definition Events Security Definition Name: CranSoft.WebAppUser × FILTER APPLIED 1 rows

| EVENT |
|--|
| System Administration - Security Role Users - AddUsers |

- Enter a value in the **Priority** field.
- Select the data source from the **Data Source ID** list box.

NOTE: The Data Source ID is typically the same as the Data Source ID registered on the Security Definitions header.

- Select the `webSecurityRoleUserAdded_#SEC_DEFINITION` rule created previously from the **Business Rule** list box.
- Check the **Active** check box.
- Enter a comment that describes what the rule does in the **Comment** field.

The resulting record should look similar to like the following:

Security Definition Event Rules Security Definition Name: CranSoft.WebAppUser Event: System Administration - Security Role Users - AddUsers 1 rows

| PRIORITY | DATA SOURCE ID | BUSINESS RULE | ACTIVE | COMMENT |
|----------|----------------|--|-------------------------------------|---|
| 10 | CranSoft | webSecurityRoleUserAdded_WebAppUserAdd | <input checked="" type="checkbox"/> | When a User is added to a Security Role, add User to WebAppUser if the User is not already there. |

Step 8: Create webSecurityRoleUserDeleted_XXXXXXDel SP and Assign to Event Security Role Users - RemoveUsers

In this step, create a stored procedure that removes a user from all values associated with all Security Definition Key values assigned to the Role the user is being removed from. This procedure only deletes the user assignment record if the user not granted the same access from other user role assignments or if the user is not granted access by direct user security definition key value assignments. Then, assign this procedure to the event Security Role Users – RemoveUsers.

To create the stored procedure webSecurityRoleUserDeleted_#SEC_DEFINITION#Del using template Template05_webSecurityRoleUserDeleted_XXXXXXDel.sql:

1. Open the script template **Template05_webSecurityRoleUserDeleted_XXXXXXDel.sql** in SSMS.
2. Replace the word **#DATABASE#** with the name of the database.
3. Replace the word **#SEC_DEFINITION#** with the short name of security definition (for consistency, use same value on all stored proceduress).
4. Replace **Template02_webSecurity_#SEC_DEFINITION#Del** with the procedure created in [Step 5: Create Security Definition User Unassignment Stored Procedure](#).

Refer to the following screen shots for an example.

Before:

```
EXECUTE Template02 webSecurity #SEC DEFINITION#Del
@UserId,
@KeyName,
@KeyValue
```

After:

```
EXECUTE webSecurity_WebAppUserDel
@UserId,
@KeyName,
@KeyValue
```

5. Execute the modified SQL script to create the stored procedure.

To assign the stored procedure webSecurityRoleUserDeleted_#SEC_DEFINITION#Del to the event Security Role Users – RemoveUsers:

1. Click the **Events** icon for the Security Definition being created.
2. Click the **Rule** icon for the **Users - Remove Users** event.

Security Definitions × FILTER APPLIED ? ⚙️ 1 rows

| SECURITY DEFINITION NAME | DATA SOURCE ID | DATA VIEW | DESCRIPTION | SYSTEM PROVIDED |
|--------------------------|----------------|------------------------------|------------------------|-----------------|
| CranSoft.WebAppUser | CranSoft | webWebAppSec | WebApp User Assignment | |

Security Definition Events Security Definition Name: CranSoft.WebAppUser × FILTER APPLIED ? ⚙️ 1 rows

| EVENT |
|---|
| System Administration - Security Role Users - RemoveUsers |

3. Enter a value in the **Priority** field.
4. Select the data source from the **Data Source ID** list box.

NOTE: The Data Source ID is typically the same as the Data Source ID registered on the *Security Definitions* page.

5. Select the `webUserDeleted_#SEC_DEFINITION#Del` rule created previously from the **Business Rule** list box.
6. Check the **Active** check box.
7. Enter a comment that describes what the rule does in the **Comment** field.
8. Click **Save**.

The resulting record should look similar to the following:

Security Definition Event Rules Security Definition Name: CranSoft.WebAppUser Event: System Administration - Security Role Users - RemoveUsers ?

| PRIORITY | DATA SOURCE ID | BUSINESS RULE | ACTIVE | COMMENT |
|----------|----------------|--|-------------------------------------|---|
| 10 | CranSoft | webSecurityRoleUserDeleted_WebAppUserDel | <input checked="" type="checkbox"/> | When a User is removed from a Security Role, delete User from WebAppUser if the User does not have permissions through other roles or user specific keys. |

Step 9: Create webSecurityRoleKeyAdded_XXXXXXAdd SP and Assign to Event Security Role Key Values - AddKeys

In this step, create a stored procedure that assigns a user to a newly added Role Security Definition Key. Then, assign this stored procedure to the event Security Role Key Values - AddKeys.

To create the stored procedure `webSecurityRoleKeyAdded_#SEC_DEFINITION#Add` using the template `Template06_webSecurityRoleKeyAdded_XXXXXXAdd.sql`:

1. Open the script template `Template06_webSecurityRoleKeyAdded_XXXXXXAdd.sql` in SSMS.
2. Replace the word `#DATABASE#` with the name of the database.

3. Replace the word #SEC_DEFINITION# with the short name of security definition (for consistency, use same value on all stored procedures).
4. Replace **Template01_webSecurity_#SEC_DEFINITION#Add** with the procedure created in [Step 4: Create Security Definition User Assignment Stored Procedure](#).

Refer to the following screen shots for an example.

Before:

```
EXECUTE Template01 webSecurity #SEC_DEFINITION#Add
    @UserID,
    @KeyName,
    @KeyValue
    @boaUserID
```

After:

```
EXECUTE webSecurity WebAppUserAdd
    @UserID,
    @KeyName,
    @KeyValue
    @boaUserID
```

5. Execute the modified SQL script to create the stored procedure.

To assign the stored procedure webSecurityRoleKeyAdded_#SEC_DEFINITION#Add to the event Security Role Key Values – AddKeys:

1. Click the **Events** icon for the Security Definition being created.
2. Click **Rule** icon for the **Values – Add Keys** event.

The screenshot shows the 'Security Definitions' table with one row: 'CranSoft.WebAppUser' with data source 'CranSoft' and data view 'webWebAppSec'. The 'Events' icon for this definition is highlighted with a red box. Below, the 'Security Definition Events' table shows one event: 'System Administration - Security Role Key Values - AddKeys', which is also highlighted with a red box.

3. Enter a value in the **Priority** field.

4. Select the data source from the **Data Source ID** list box.

NOTE: The Data Source ID is typically the same as the Data Source ID registered on the *Security Definitions* page.

5. Select the **webSecurityRoleKeyAdded_#SEC_DEFINITION#Add** rule created previously from the **Business Rule** list box.
6. Check the **Active** check box.
7. Enter a comment that describes what the rule does in the **Comment** field.
8. Click **Save**.

The resulting record should look similar to the following:

| Security Definition Event Rules | | Security Definition Name | Event | | |
|--|----------------|---------------------------------------|--|---|--|
| | | CranSoft.WebAppUser | System Administration - Security Role Key Values - AddKeys | | |
| <input type="button" value="Add"/> <input type="button" value="Edit"/> | | | | | |
| PRIORITY | DATA SOURCE ID | BUSINESS RULE | ACTIVE | COMMENT | |
| 10 | CranSoft | webSecurityRoleKeyAdded_WebAppUserAdd | <input checked="" type="checkbox"/> | When a Key is added to a Security Role, for Users that have the Key, add User to WebAppUser if the User is not already there. | |

Step 10: Create webSecurityRoleKeyDeleted_XXXXXXDel SP and Assign to Event Security Role Key Values - RemoveKeys

In this step, create a stored procedure that removes a user from a deleted Role Security Definition Key value. This stored procedure only deletes the user assignment record if the user is not granted the same access from other user role assignments or if the user is not granted access by direct user security definition key value assignments. Then, assign this procedure to the event Security Role Key Values - RemoveKeys.

To create the stored procedure webSecurityRoleKeyDeleted_#SEC_DEFINITION#Del using template Template07_webSecurityRoleKeyDeleted_XXXXXXDel.sql:

1. Open the script template **Template07_webSecurityRoleKeyDeleted_XXXXXXDel.sql** in SSMS.
2. Replace the word **#DATABASE#** with the name of the database.
3. Replace the word **#SEC_DEFINITION#** with the short name of security definition (for consistency, use same value on all stored procedures).
4. Replace **Template02_webSecurity_#SEC_DEFINITION#Del** with the procedure created in [Step 5: Create Security Definition User Unassignment Stored Procedure](#).

Refer to the following screen shots for an example.

Before:

```
EXECUTE Template02 webSecurity #SEC_DEFINITION#Del
@UserId,
@KeyName,
@KeyValue
```

After:

```
EXECUTE webSecurity_WebAppUserDel
@User Id,
@KeyName,
@KeyValue
```

5. Execute the modified SQL script to create the stored procedure.

To assign the stored procedure webSecurityRoleKeyDeleted_#SEC_DEFINITION#Del to the event Security Role Key Values – RemoveKeys:

1. Click the **Events** icon for the Security Definition being created.
2. Click the **Rules** icon for the **Values – Remove Keys** event.

3. Enter a value in the **Priority** field.
4. Select the data source from the **Data Source ID** list box.

NOTE: The Data Source ID is typically the same as the Data Source ID registered on the *Security Definitions* page.

5. Select the **webSecurityRoleKeyDeleted_#SEC_DEFINITION#Del** rule created previously from the **Business Rule** list box.
6. Check the **Active** check box.
7. Enter a comment that describes what the rule does in the **Comment** field.

8. Click **Save**.

The resulting record should look similar to the following:

| Security Definition Event Rules | | | | |
|---------------------------------|-------------|---|--|---|
| | | Security Definition Name CranSoft.WebAppUser | Event System Administration - Security Role Key Values - RemoveKeys | |
| Add Edit | | | | |
| PRIORITY | DATA SOURCE | BUSINESS RULE | ACTIVE | COMMENT |
| ID | | | | |
| 10 | CranSoft | webSecurityRoleKeyDeleted_WebAppUserDel | <input checked="" type="checkbox"/> | When a Key is removed from a Security Role, delete User that has that Security Key from WebAppUser if the User does not have permissions through other roles or user specific keys. |

Step 11: Create webSecurityRoleDeleted_XXXXXXDel SP and Assign to Event Security Roles - BeforeDelete

In this step, create a stored procedure that removes all users from all values associated with all Security Definition Key values assigned to the deleted Role. This procedure only deletes the user assignment record if the user is not granted the same access from other user role assignments or if the user is not granted access by direct user security definition key value assignments. Then, assign this stored procedure to the event Security Roles - BeforeDelete.

To create the stored procedure webSecurityRoleDeleted_#SEC_DEFINITION#Del using the template Template08_webSecurityRoleDeleted_XXXXXXDel.sql:

1. Open the script template **Template08_webSecurityRoleDeleted_XXXXXXDel.sql** in SSMS.
2. Replace the word **#DATABASE#** with the name of the database.
3. Replace the word **#SEC_DEFINITION#** with the short name of security definition (for consistency, use same value on all stored procedures).
4. Replace **Template02_webSecurity_#SEC_DEFINITION#Del** with the procedure created in [Step 5: Create Security Definition User Unassignment Stored Procedure](#).

Refer to the following screen shots for an example.

Before:

```
EXECUTE Template02 webSecurity #SEC DEFINITION#Del
@UserId,
@KeyName,
@KeyValue
```

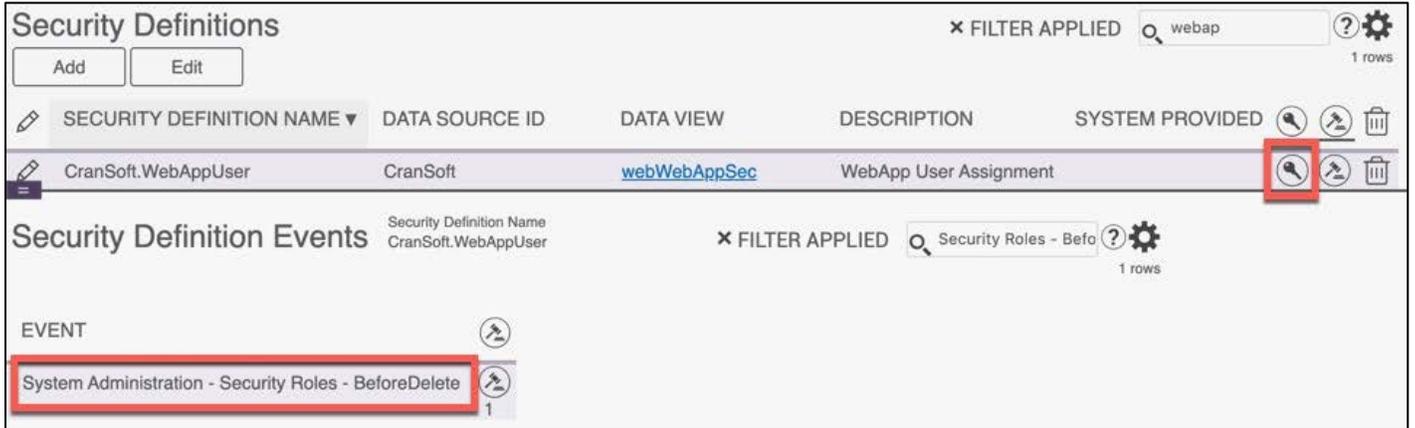
After:

```
EXECUTE webSecurity_WebAppUserDel
@User Id,
@KeyName,
@KeyValue
```

5. Execute the modified SQL script to create the stored procedure.

To assign the stored procedure `webSecurityRoleDeleted_#SEC_DEFINITION#Del` to the event Security Roles - BeforeDelete:

1. Click the **Events** icon for the Security Definition being created.
2. Click the **Rules** icon for the **Roles – Before Delete** event.



Security Definitions × FILTER APPLIED 1 rows

| SECURITY DEFINITION NAME | DATA SOURCE ID | DATA VIEW | DESCRIPTION | SYSTEM PROVIDED |
|--------------------------|----------------|------------------------------|------------------------|-----------------|
| CranSoft.WebAppUser | CranSoft | webWebAppSec | WebApp User Assignment | |

Security Definition Events Security Definition Name: CranSoft.WebAppUser × FILTER APPLIED 1 rows

| EVENT |
|---|
| System Administration - Security Roles - BeforeDelete |

3. Enter a value in the **Priority** field.
4. Select the data source from the **Data Source ID** list box.

NOTE: The Data Source ID is typically the same as the Data Source ID registered on the *Security Definitions* page.

5. Select the `webSecurityRoleKeyDeleted_#SEC_DEFINITION#Del` rule created previously from the **Business Rule** list box.
6. Check the **Active** check box.
7. Enter a comment that describes what the rule does in the **Comment** field.
8. Click **Save**.

The resulting record should look similar to the following:



Security Definition Event Rules Security Definition Name: CranSoft.WebAppUser Event: System Administration - Security Roles - BeforeDelete

| PRIORITY | DATA SOURCE ID | BUSINESS RULE | ACTIVE | COMMENT |
|----------|----------------|---|-------------------------------------|---|
| 10 | CranSoft | <code>webSecurityRoleDeleted_WebAppUserDel</code> | <input checked="" type="checkbox"/> | When a Security Role is deleted, delete User that has that Security Role from WebAppUser if the User does not have permissions through other roles or user specific keys. |

Step 12: Create webUserSecurityKeyAdded_XXXXXXAdd SP and Assign to Event User Specific Keys - AddKeys

In this step, create a procedure that assigns a user to an added User Specific Security Definition Key. Then, assign this procedure to the event User Specific Keys – AddKeys.

To create the stored procedure webUserSecurityKeyAdded_#SEC_DEFINITION#Add using the template Template09_webUserSecurityKeyAdded_XXXXXXAdd.sql:

1. Open the script template **Template09_webUserSecurityKeyAdded_XXXXXXAdd.sql** in SSMS.
2. Replace the word **#DATABASE#** with the name of the database.
3. Replace the word **#SEC_DEFINITION#** with the short name of security definition (for consistency, use same value on all stored procedures).
4. Replace **Template01_webSecurity_#SEC_DEFINITION#Add** with the procedure created in [Step 4: Create Security Definition User Assignment Stored Procedure](#).

Refer to the following screen shots for an example.

Before:

```
EXECUTE Template01 webSecurity #SEC DEFINITION#Add
    @UserID,
    @KeyName,
    @KeyValue
    @boaUserID
```

After:

```
EXECUTE webSecurity WebAppUserAdd
    @UserID,
    @KeyName,
    @KeyValue
    @boaUserID
```

5. Execute the modified SQL script to create the stored procedure.

To assign the stored procedure webUserSecurityKeyAdded_#SEC_DEFINITION#Add to the event User Specific Keys - AddKeys:

1. Click the **Events** icon for the Security Definition being created.
2. Click the **Rules** icon for the **Keys – Add Keys** event.

Security Definitions × FILTER APPLIED ? ⚙️ 1 rows

| SECURITY DEFINITION NAME ▼ | DATA SOURCE ID | DATA VIEW | DESCRIPTION | SYSTEM PROVIDED |
|----------------------------|----------------|------------------------------|------------------------|--------------------------|
| CranSoft.WebAppUser | CranSoft | webWebAppSec | WebApp User Assignment | <input type="checkbox"/> |

Security Definition Events × FILTER APPLIED ? ⚙️ 1 rows

Security Definition Name: CranSoft.WebAppUser

| EVENT |
|--|
| System Administration - User Specific Keys - AddKeys |

3. Enter a value in the **Priority** field.
4. Select the data source from the **Data Source ID** list box.

NOTE: The Data Source ID is typically the same as the Data Source ID registered on the *Security Definitions* page.

5. Select the **webSecurityRoleKeyAdded_#SEC_DEFINITION#Add** rule created previously from the **Business Rule** list box.
6. Check the **Active** check box.
7. Enter a comment that describes what the rule does in the **Comment** field.
8. Click **Save**.

The resulting record should look something like the following:

Security Definition Event Rules Security Definition Name: CranSoft.WebAppUser Event: System Administration - User Specific Keys - AddKeys

| PRIORITY ▼ | DATA SOURCE ID | BUSINESS RULE | ACTIVE | COMMENT |
|------------|----------------|---------------------------------------|-------------------------------------|---|
| 10 | CranSoft | webUserSecurityKeyAdded_WebAppUserAdd | <input checked="" type="checkbox"/> | When a User Specific Key is added, add User to WebAppUser if the User is not already there. |

Step 13: Create webUserSecurityKeyDeleted_XXXXXXDel SP and Assign to Event User Specific Keys - RemoveKeys

In this step, create a stored procedure that removes a user access from a deleted User Security Definition Key value. This stored procedure only deletes the user assignment record if the user is not granted the same access from other user role assignments or if the user is not granted access by direct user security definition key value assignments. Then, assign this procedure to the event Security Role Key Values - RemoveKeys.

To create the stored procedure webUserSecurityKeyDeleted_#SEC_DEFINITION#Del using template Template10_webUserSecurityKeyDeleted_XXXXXXDel.sql:

1. Open the script template `Template10_webUserSecurityKeyDeleted_XXXXXXDel.sql` in SSMS.
2. Replace the word `#DATABASE#` with the name of the database.
3. Replace the word `#SEC_DEFINITION#` with the short name of security definition (for consistency, use same value on all stored procedures).
4. Replace `Template01_webSecurity_#SEC_DEFINITION#Add` with the procedure created in [Step 5: Create Security Definition User Unassignment Stored Procedure](#).

Refer to the following screen shots for an example.

Before:

```
EXECUTE Template02 webSecurity #SEC_DEFINITION#Del
@UserId,
@KeyName,
@KeyValue
```

After:

```
EXECUTE webSecurity_WebAppUserDel
@UserId,
@KeyName,
@KeyValue
```

5. Execute the modified SQL script to create the stored procedure.

To assign the stored procedure `webUserSecurityKeyDeleted_#SEC_DEFINITION#Del` to event `User Specific Keys - RemoveKeys`.

1. Click the **Events** icon for the Security Definition being created.
2. Click the **Rules** icon for the **Keys - Remove Keys** event.

The screenshot displays two UI sections. The top section, titled "Security Definitions", contains a table with the following data:

| SECURITY DEFINITION NAME | DATA SOURCE ID | DATA VIEW | DESCRIPTION | SYSTEM PROVIDED |
|--------------------------|----------------|--------------|------------------------|-----------------|
| CranSoft.WebAppUser | CranSoft | webWebAppSec | WebApp User Assignment | |

The "Events" icon for the first row is highlighted with a red box. The bottom section, titled "Security Definition Events", contains a table with the following data:

| EVENT |
|---|
| System Administration - User Specific Keys - RemoveKeys |

The "Rules" icon for the first row is highlighted with a red box.

3. Enter a value in the **Priority** field.
4. Select the data source from the **Data Source ID** list box.

NOTE: The Data Source ID is typically the same as the Data Source ID registered on the *Security Definitions* page.

5. Select the **webSecurityRoleKeyDeleted_#SEC_DEFINITION#Del** rule created previously from the **Business Rule** list box.
6. Check the **Active** check box.
7. Enter a comment that describes what the rule does in the **Comment** field.
8. Click **Save**.

The resulting record should look similar to the following:

| Security Definition Event Rules | | Security Definition Name | Event |
|--|----------------|---|---|
| | | CranSoft.WebAppUser | System Administration - User Specific Keys - RemoveKeys |
| <input type="button" value="Add"/> <input type="button" value="Edit"/> | | | |
| PRIORITY | DATA SOURCE ID | BUSINESS RULE | ACTIVE COMMENT |
| 10 | CranSoft | webUserSecurityKeyDeleted_WebAppUserDel | <input checked="" type="checkbox"/> When a User Specific Key is removed, delete that user from WebAppUser unless the User has that key through other roles. |

Step 14: Assign webSecurityRoleUserDeleted_XXXXXXDel SP to Event User Roles - UnassignToRole

To assign the stored procedure webSecurityRoleUserDeleted_#SEC_DEFINITION#Del to the event User Roles - UnassignToRole:

1. Click the **Events** icon for the Security Definition being created.
2. Click the **Rules** icon for the **User Roles - Unassign to Role** event.

The screenshot shows two tables. The top table, 'Security Definitions', has columns: SECURITY DEFINITION NAME, DATA SOURCE ID, DATA VIEW, DESCRIPTION, and SYSTEM PROVIDED. A row is shown for 'CranSoft.WebAppUser' with 'CranSoft' as the data source ID and 'webWebAppSec' as the data view. A red box highlights the 'Rules' icon in the 'SYSTEM PROVIDED' column for this row. The bottom table, 'Security Definition Events', has columns: EVENT, and a 'Rules' icon. A row is shown for 'System Administration - User Roles - UnassignToRole' with a red box highlighting the 'Rules' icon.

3. Enter a value in the **Priority** field.

4. Select the data source from the **Data Source ID** list box.

NOTE: The Data Source ID is typically the same as the Data Source ID registered on the *Security Definitions* page.

5. Select the **webSecurityRoleKeyDeleted_#SEC_DEFINITION#Del** rule created previously from the **Business Rule** list box.
6. Check the **Active** check box.
7. Enter a comment that describes what the rule does in the **Comment** field.
8. Click **Save**.

The resulting record should look similar to the following:

| Security Definition Event Rules | | Security Definition Name | Event |
|--|----------------|--|---|
| | | CranSoft.WebAppUser | System Administration - Security Role Users - RemoveUsers |
| <input type="button" value="Add"/> <input type="button" value="Edit"/> | | | |
| PRIORITY | DATA SOURCE ID | BUSINESS RULE | ACTIVE COMMENT |
| 10 | CranSoft | webSecurityRoleUserDeleted_WebAppUserDel | <input checked="" type="checkbox"/> When a User is removed from a Security Role, delete User from WebAppUser if the User does not have permissions through other roles or user specific keys. |

Step 15: Assign webSecurityRoleUserAdded_XXXXXXAdd SP to Event User Roles - AssignToRole

To assign the stored procedure webSecurityRoleUserAdded_#SEC_DEFINITION#Add to the Event User Roles - AssignToRole:

1. Click the **Events** icon for the Security Definition being created.
2. Click the **Rules** icon for the **User Roles – Assign to Role** event.

The screenshot shows two parts of the application interface. The top part is the 'Security Definitions' table, which has a filter 'webap' applied. It contains one row for 'CranSoft.WebAppUser' with data source 'CranSoft' and data view 'webWebAppSec'. The 'Rules' icon for this row is highlighted with a red box. The bottom part is the 'Security Definition Events' table, which also has a filter applied. It contains one row for the event 'System Administration - User Roles - AssignToRole', which is highlighted with a red box.

3. Enter a value in the **Priority** field.
4. Select the data source from the **Data Source ID** list box.

NOTE: The Data Source ID is typically the same as the Data Source ID registered on the Security Definitions header.

5. Select the **webSecurityRoleKeyAdded_#SEC_DEFINITION#Add** rule created previously from the **Business Rule** list box.
6. Check the **Active** check box.
7. Enter a comment that describes what the rule does in the **Comment** field.
8. **Click Save.**

The resulting record should look similar to the following:

| Security Definition Event Rules | | | | |
|--|----------------|---|--|---|
| | | Security Definition Name CranSoft.WebAppUser | Event System Administration - User Roles - AssignToRole | |
| <input type="button" value="Add"/> <input type="button" value="Edit"/> | | | | |
| PRIORITY ▼ | DATA SOURCE ID | BUSINESS RULE | ACTIVE | COMMENT |
| 10 | CranSoft | webSecurityRoleUserAdded_WebAppUserAdd | <input checked="" type="checkbox"/> | When a User is added to a Security Role, add User to WebAppUser if the User is not already there. |

Step 16: Assign webSecurityRoleUserDeleted_XXXXXXDel SP to Event User Roles Staging - UnassignToRole

To assign the stored procedure webSecurityRoleUserDeleted_#SEC_DEFINITION#Del to the event User Roles Staging - UnassignToRole:

1. Click the **Events** icon for the Security Definition being created.
2. Click the **Rules** icon for the **User Roles Staging – Unassign to Role** event.

Security Definitions

× FILTER APPLIED

1 rows

| SECURITY DEFINITION NAME ▼ | DATA SOURCE ID | DATA VIEW | DESCRIPTION | SYSTEM PROVIDED | |
|----------------------------|----------------|------------------------------|------------------------|-----------------|---|
| CranSoft.WebAppUser | CranSoft | webWebAppSec | WebApp User Assignment | | <input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> |

Security Definition Events

× FILTER APPLIED

1 rows

| EVENT | |
|---|--|
| System Administration - User Roles Staging - UnassignToRole | <input checked="" type="checkbox"/> <input type="checkbox"/> |

3. Enter a value in the **Priority** field.
4. Select the data source from the **Data Source ID** list box.

NOTE: The Data Source ID is typically the same as the Data Source ID registered on the *Security Definitions* page.

5. Select the **webSecurityRoleKeyDeleted_#SEC_DEFINITION#Del** rule created previously from the **Business Rule** list box.
6. Check the **Active** check box.
7. Enter a comment that describes what the rule does in the **Comment** field.
8. Click **Save**.

The resulting record should look similar to the following:

| Security Definition Event Rules | | Security Definition Name | Event |
|--|----------------|--|---|
| | | CranSoft.WebAppUser | System Administration - User Roles Staging - UnassignToRole |
| <input type="button" value="Add"/> <input type="button" value="Edit"/> | | | |
| PRIORITY | DATA SOURCE ID | BUSINESS RULE | ACTIVE COMMENT |
| 10 | CranSoft | webSecurityRoleUserDeleted_WebAppUserDel | <input checked="" type="checkbox"/> When a User is removed from a Security Role, delete User from WebAppUser if the User does not have permissions through other roles or user specific keys. |

Step 17: Assign webSecurityRoleUserAdded_XXXXXXAdd SP to Event User Roles Staging - AssignToRole

To assign the stored procedure webSecurityRoleUserAdded_#SEC_DEFINITION#Add to the event User Roles Staging – AssignToRole:

1. Click the **Events** icon for the Security Definition being created.
2. Click the **Rules** icon for the **User Roles Staging – Assign to Role** event.
3. Enter a value in the **Priority** field.
4. Select the data source from the **Data Source ID** list box.

NOTE: The Data Source ID is typically the same as the Data Source ID registered on the *Security Definitions* page.

5. Select the **webSecurityRoleKeyAdded_#SEC_DEFINITION#Add** rule created previously from the **Business Rule** list box.
6. Check the **Active** check box.
7. Enter a comment that describes what the rule does in the **Comment** field.
8. Click **Save**.

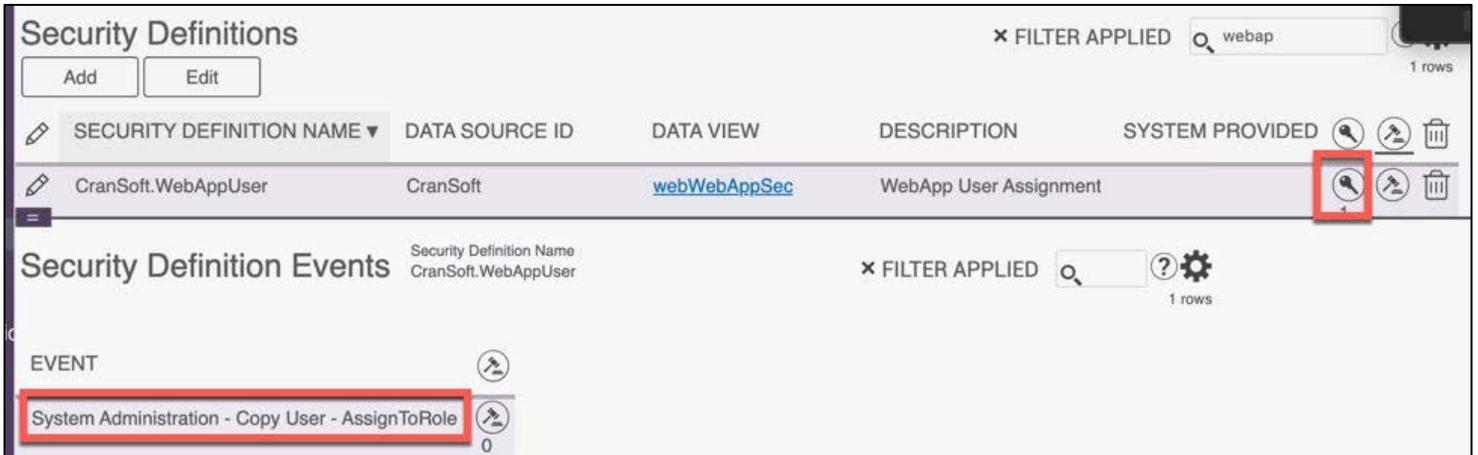
The resulting record should look similar to the following:

| Security Definition Event Rules | | Security Definition Name | Event |
|--|----------------|--|---|
| | | CranSoft.WebAppUser | System Administration - User Roles Staging - AssignToRole |
| <input type="button" value="Add"/> <input type="button" value="Edit"/> | | | |
| PRIORITY | DATA SOURCE ID | BUSINESS RULE | ACTIVE COMMENT |
| 10 | CranSoft | webSecurityRoleUserAdded_WebAppUserAdd | <input checked="" type="checkbox"/> When a User is added to a Security Role, add User to WebAppUser if the User is not already there. |

Step 18: Assign webSecurityRoleUserAdded_XXXXXXAdd SP to Event Copy User - AssignToRole

To assign the stored procedure webSecurityRoleUserAdded_#SEC_DEFINITION#Add to the event Copy User - AssignToRole:

1. Click the **Events** icon for the Security Definition being created.
2. Click the **Rules** icon for the **Copy User - Assign to Role** event.



Security Definitions × FILTER APPLIED webap 1 rows

| SECURITY DEFINITION NAME | DATA SOURCE ID | DATA VIEW | DESCRIPTION | SYSTEM PROVIDED |
|--------------------------|----------------|--------------|------------------------|-----------------|
| CranSoft.WebAppUser | CranSoft | webWebAppSec | WebApp User Assignment | |

Security Definition Events Security Definition Name: CranSoft.WebAppUser × FILTER APPLIED 1 rows

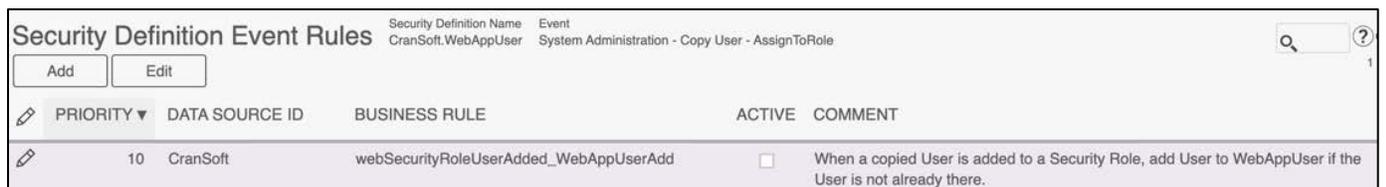
| EVENT |
|--|
| System Administration - Copy User - AssignToRole |

3. Enter a value in the **Priority** field.
4. Select the data source from the **Data Source ID** list box.

NOTE: The Data Source ID is typically the same as the Data Source ID registered on the *Security Definitions* page.

5. Select the **webSecurityRoleKeyAdded_#SEC_DEFINITION#Add** rule created previously from the **Business Rule** list box.
6. Check the **Active** check box.
7. Enter a comment that describes what the rule does in the **Comment** field.
8. Click **Save**.

The resulting record should look similar to the following:



Security Definition Event Rules Security Definition Name: CranSoft.WebAppUser Event: System Administration - Copy User - AssignToRole

| PRIORITY | DATA SOURCE ID | BUSINESS RULE | ACTIVE | COMMENT |
|----------|----------------|--|--------------------------|--|
| 10 | CranSoft | webSecurityRoleUserAdded_WebAppUserAdd | <input type="checkbox"/> | When a copied User is added to a Security Role, add User to WebAppUser if the User is not already there. |

Step 19: Assign webUserSecurityKeyAdded_XXXXXXAdd SP to Event'Copy User - AddKeys

To assign the stored procedure webUserSecurityKeyAdded_#SEC_DEFINITION#Add to the event Copy User - AddKeys:

1. Click the **Events** icon for the Security Definition being created.
2. Click the **Rules** icon for the **Copy User – Add Keys** event.

Security Definitions × FILTER APPLIED ? ⚙️ 1 rows

| SECURITY DEFINITION NAME | DATA SOURCE ID | DATA VIEW | DESCRIPTION | SYSTEM PROVIDED | |
|--------------------------|----------------|------------------------------|------------------------|-----------------|--|
| CranSoft.WebAppUser | CranSoft | webWebAppSec | WebApp User Assignment | | |

Security Definition Events × FILTER APPLIED ? ⚙️ 1 rows

Security Definition Name: CranSoft.WebAppUser

| EVENT | |
|---|---|
| System Administration - Copy User - AddKeys | 0 |

3. Enter a value in the **Priority** field.
4. Select the data source from the **Data Source ID** list box.

NOTE: The Data Source ID is typically the same as the Data Source ID registered on the *Security Definitions* page.

5. Select the **webSecurityRoleKeyAdded_#SEC_DEFINITION#Add** rule created previously from the **Business Rule** list box.
6. Check the **Active** check box.
7. Enter a comment that describes what the rule does in the **Comment** field.
8. Click **Save**.

The resulting record should look similar to the following:

Security Definition Event Rules Security Definition Name: CranSoft.WebAppUser Event: System Administration - Copy User - AddKeys ?

| PRIORITY | DATA SOURCE ID | BUSINESS RULE | ACTIVE | COMMENT |
|----------|----------------|---------------------------------------|-------------------------------------|---|
| 10 | CranSoft | webUserSecurityKeyAdded_WebAppUserAdd | <input checked="" type="checkbox"/> | When a copied User has User Specific Keys added, add User to WebAppUser if the User is not already there. |

Appendix A - Handling Security Definitions whose Data View has Multiple Key Columns

The SQL templates provided have been designed to work with Security Definitions that have a single key field. If a Security Definition Key is comprised of multiple key values, then there is some additional complexity that needs to be handled.

This additional complexity arises because DSP holds the Security Definition Key Columns in table rows whereas the usage of these columns requires that the key columns and associated values are pivoted on a single record.

An example of using a Security Definition with multi Key Columns can be found in the following ZIP file:

- DSP_CustomSecurityDefinition_ComplexExample.zip

Example for Security Definition with single key column Template04_webSecurityRoleUserAdded_XXXXXXAdd

```
INSERT INTO @RoleUserAdd
(
    RoleName,
    KeyName,
    KeyValue,
    UserID)
SELECT
    RoleName,
    KeyName,
    KeyValue,
    UserID
FROM
    DSPCommon.dbo.GetSecurityRoleUserAdded(@UserID, @RoleID, @SecurityDefinitionID)
```

Example for Security Definition with 2 key column Template04_webSecurityRoleUserAdded_WebAppGroupUserAdd

In this example, along with the addition of multiple KeyValues and KeyNames, the code pivots the data returned from the GetSecurityRuleUserAdded() table value function and moves the KeyName 'WebAppID' (Key 1) and 'GroupID' (Key 2) from records into a single row using the SQL Pivot operator. This enables simpler processing in subsequent steps.

```

INSERT INTO @RoleUserAdd
    (RoleID,
     RoleName,
     KeyName1,
     KeyValue1,
     KeyName2,
     KeyValue2,
     UserID)

SELECT
    @RoleID,
    RoleName,
    'WebAppID' as KeyName1,
    [WebAppID] as KeyValue1,
    'GroupID' as KeyName2,
    [GroupID] as KeyValue2,
    UserID
FROM
    (SELECT
        RoleName,
        SecurityDefinitionKeyID,
        UserID,
        KeyName,
        KeyValue
     FROM
        DSPCommon.dbo.GetSecurityRoleUserAdded(@UserID, @RoleID, @SecurityDefinitionID) AS src
     PIVOT ( Max(KeyValue)
            FOR KeyName IN ([WebAppID],
                           [GroupID]) ) AS Dtpivot;

```

A complete example is available within the SQL scripts in folder DSP_CustomSecurityDefinition_ComplexExample.zip.

Appendix B - Handling Security Definitions whose Event Rules Populate Multiple Tables

The recommended approach for implementing Custom Security Definitions is to have dedicated stored procedures that:

1. Handle the insertion of a user into application tables that provision user access.

NOTE: Refer to template Template01_webSecurity_XXXXXX.sql.

2. Handle the deletion of a user from application tables that provision user access.

NOTE: Refer to template Template02_webSecurity_XXXXXXDel.

The SQL templates provided assume that a single table is involved in the user provisioning. If there is a requirement to provision user access via multiple tables, then the logic for provisioning and revoking access needs to be analyzed and understood. This logic can then be embedded with the associated addition/deletion stored procedures.

An example can be found in the following ZIP file:

- DSP_CustomSecurityDefinition_ComplexExample.zip
 - Template01_webSecurity_WebAppGroupUserAdd.sql
 - Template02_webSecurity_WebAppGroupUserDel

Last Updated on 3/3/2020