

X-ruleStudio

User Guide

V6.2.17

XengineXperts, Inc.

Copyright ©2019 All rights reserved worldwide

Contents

Welcome to X-ruleStudio.....	3
The real purpose	3
Navigation.....	4
Signing In.....	4
The main screen.....	6
Managing folders	7
Managing rules	8
Overrides	12
Security	13
Roles	14
Rights	15
Users	16

Product Version: 6.2.17-build 9

Release Date: 11/10/2019

Welcome to X-ruleStudio.

X-ruleStudio is a complete IDE for designing, testing and managing business rules, Building rules in an intuitive development environment and testing them right then and there is what the developers are expecting from the modern software engineering products. In the world of business rules however, it's not just software engineers who are the rules developers. Here is our audience:

- Business analysts (BA's)
- Managers
- QA engineers and testers
- Software engineers
- Sales and marketing specialists
- Support technicians
- And the end users of products

Each one of groups is keen on keeping their objectives specific to their business operations and their routines. Understanding these routines is the key to building a versatile tool that would cater to this kind of diverse roster of users. We observed, analyzed and learned by working side-by-side with all of these groups.

Our goal is to create the best business rules management utility for them.

We hope you will enjoy using this product.

The real purpose

Why do we need rules? An excellent question. We posed this question to our prospective clients at the very beginning of this project.

The usual routine in SDLC is all about the product being built and maintained by engineering and support organizations. Everyone is ok with the building part because engineering is the most suitable organization for that task. Engineering teams possess knowledge and skills for building products which is a complicated and high pressure task.

The maintenance part is where practically all within our polling audience agreed on the fact that depending on engineering and support for every single change was a big problem. And we've seen it way too many times to disagree.

Our polling audience wanted autonomy, a way to make changes on their own and test their changes right then and there, stop shuttling back and forth to engineering and support teams and being stuck in the cycle that requires a small brigade. They wanted the software products they could call self-service.

X-ruleStudio was created with these notions in mind.

Let's dive in and explore the ways you can start using this tool.

Navigation

Below is a snapshot of the main control panel.

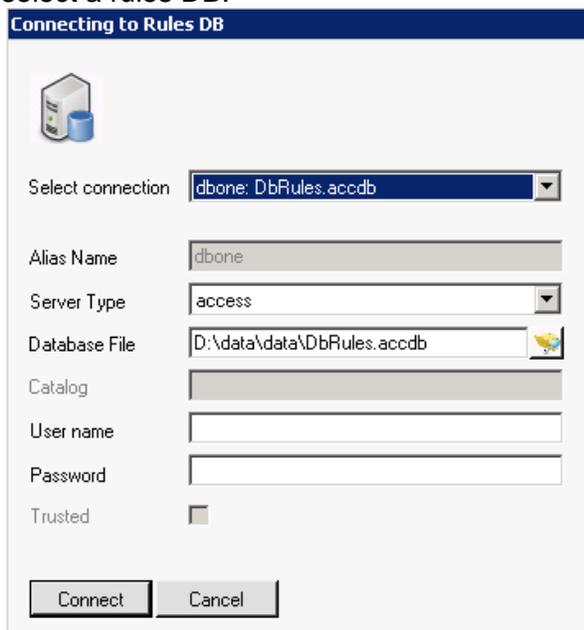


General navigation notes

X-ruleStudio implements auto-saving of the UI elements while you navigate throughout the product's screens. This means that by resizing or moving the main X-ruleStudio window or by resizing the panes you implicitly cause the dimensions and positioning of the windows saved thus making them persist between sessions. When you exit and re-enter the program the state of your session would be saved and restored automatically.

Signing In

The security of X-ruleStudio and the rules engine in general is tightly integrated with security of the hosting environment (usually domain). Moreover, roles and rights level security is in place on all levels, from rules authoring to execution including integration with external objects. This will be covered further in the section dedicated to this topic. But for now, in order to access the facility of X-ruleStudio you're required to sign in and in order to be able to sign in you need to select a rules DB.



MS Access type of database is set as default but SQL Server and Oracle are also supported. Creating an alias name for your connection is required and authentication parameters might be required as well, all depending on the security settings of your particular database.

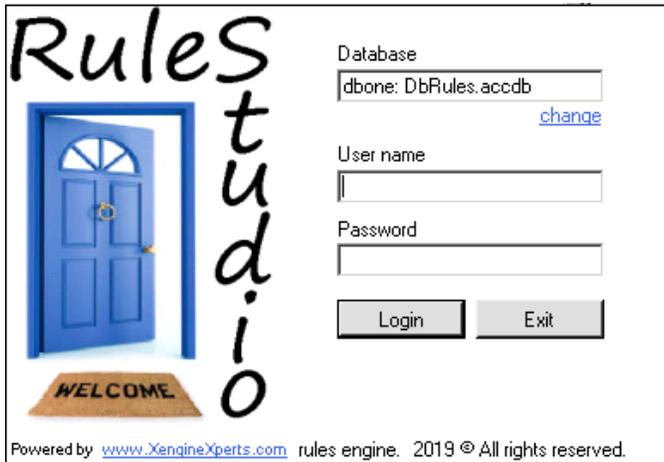
Please note that you can create any number of connections to the same or different databases, For instance you can connect to DB1 database hosted by SQL Server at "localhost" using 3 different user accounts, or a single user account.

If you succeed connecting to the database your connection parameters are stored and you bypass this step the next time you attempt to sign into X-ruleStudio.

Now you would be required to sign in with your rules DB authentication values (username/password).

There are no built-in accounts in the rules DB but you would be asked to create an administrator level username/password right at the end of installation process. Also after a successful installation, making a backup of your rules DB is a good practice. In general, performing regular backup operations is always a good practice. Contact your system administrator, DBA or our support team for more information.

And as far as your sign-in credentials – writing them down somewhere “safe” is never a good idea and neither is sharing them with other people. So choose something that you and only you would remember.



RulesStudio

Database
dbone: DbRules.accdb [change](#)

User name

Password

Login Exit

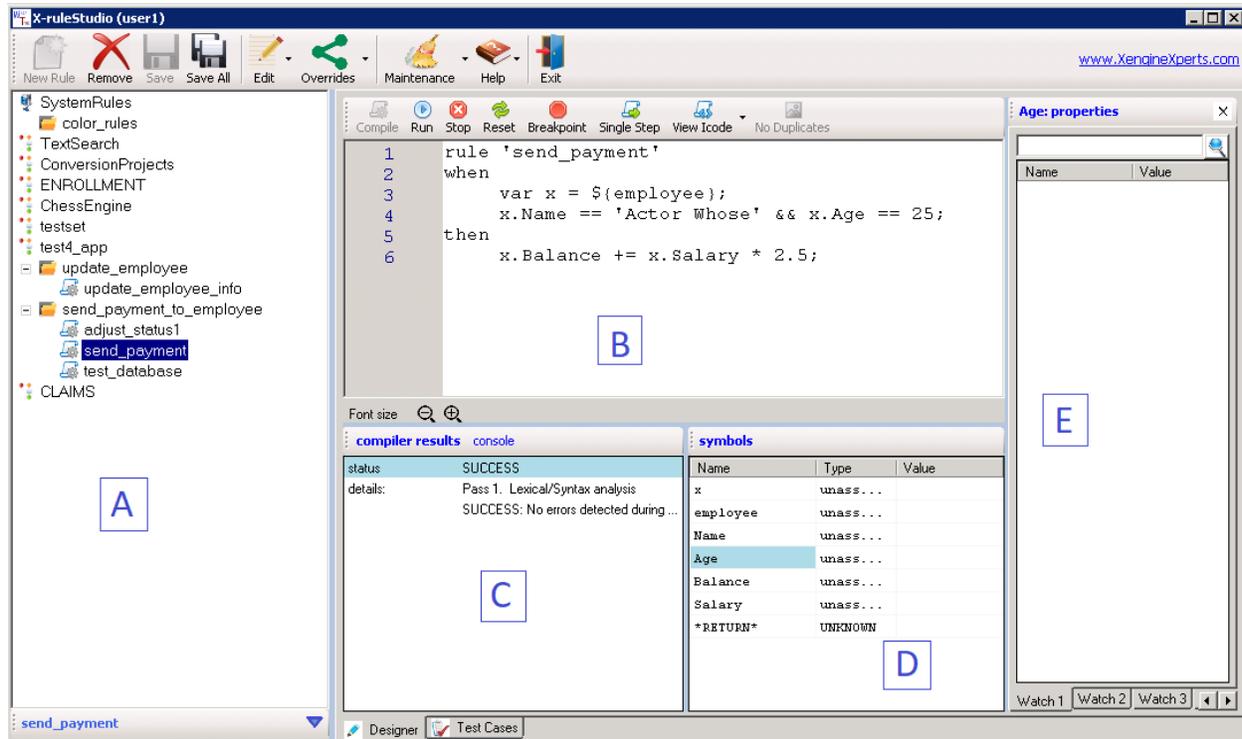
Powered by www.XengineXperts.com rules engine. 2019 © All rights reserved.

NOTE: Since authentication parameter values are not recoverable, you might need to reinstall your rules DB if you happen to forget your sign-in credentials.

By clicking [change](#) button you will be able to select another database to connect to.

The main screen

Below is view of the main screen.



Functional areas of this screen are:

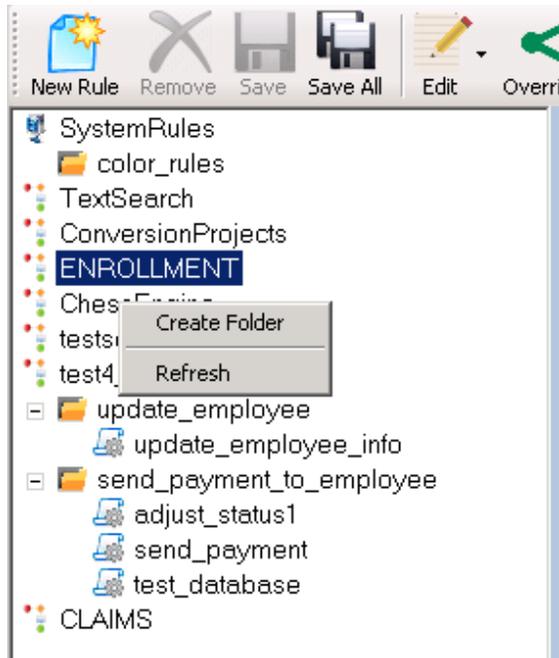
- A. View of the rule folder structure. The main idea of the folder structure resembles folder structure of the file system of any OS. Depth of subfolders is unlimited and so is the number of rules that any folder may hold, Here is a list of possible types of nodes in this structure:
 - a. Ruleset with  icon are designated for built-in (or core) rules. Contents of these folders are available for general use but are protected against any type of changes. Modifications to these folders and rules are reserved for our engineering and support teams.
 - b. Rulesets with  icon contain rules that are specific to customer built products. Contents could be other non-root folders and rules.
 - c. Folders with  icon may contain folder structures and rules.
 - d. Nodes with  icon are executable rules.
- B. Rule editor.
- C. Rules compiler results and run-time console.
- D. Rule symbols are presented here.
- E. More detailed view of the symbol values.

By clicking any folder node you would cause that folder to open and allow navigating deeper into its structure.

Managing folders

By right clicking the mouse while its pointer is hovering over the ruleset or folder nodes you gain access to the folder management functions.

Rulesets.

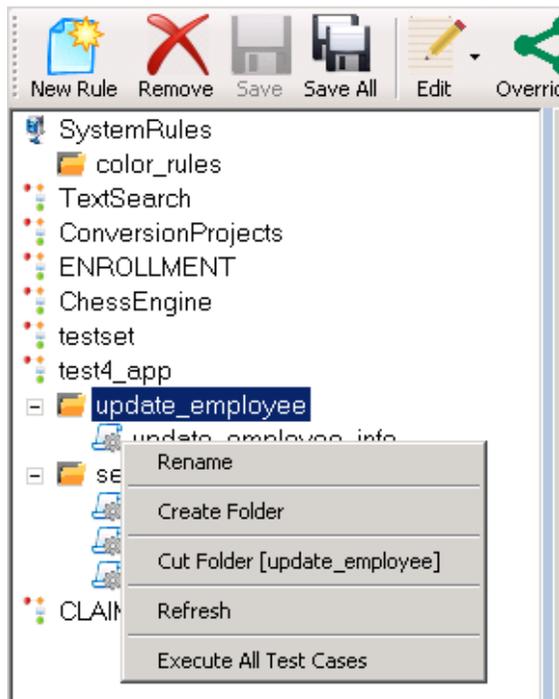


Rulesets are also root folders which can't be renamed or removed here. Special rights are required to be able to access ruleset maintenance functions. This is covered further in the document.

Creating rules is allowed at ruleset level, even without any folder structure.

 **New Rule** button is activated thus allowing you to create rules.

Regular folders



Regular folders can be renamed and removed by users who hold access rights to these functions. Other folders can be created, again a function available to those allowed.

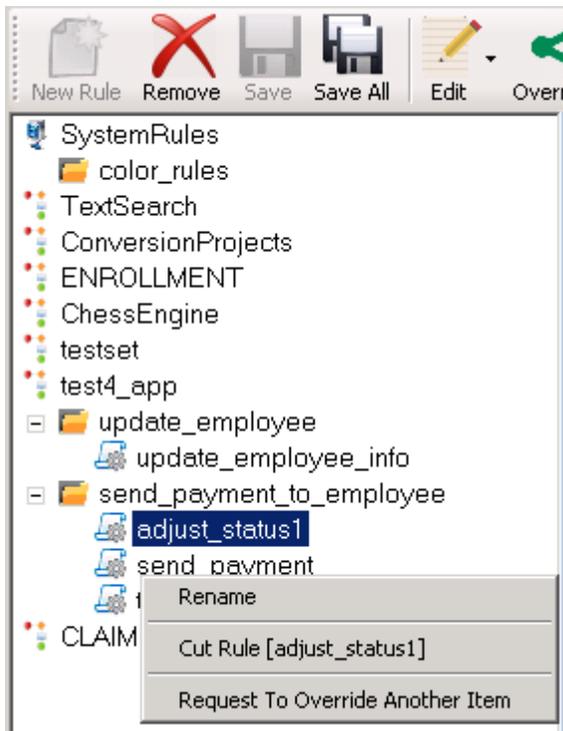
Moving folders (cutting and pasting) is also allowed.

When folders contain any rules that have test cases designed for them, the option to run all test cases becomes available.

 **New Rule** button is activated thus allowing you to create rules.

Managing rules

By right clicking the mouse while its pointer is hovering over a rule you gain access to the rule management functions.



Rules can be renamed and removed by users who hold access rights to these functions.

Moving rules (cutting and pasting) is also allowed.



The 'New Rule' button is deactivated because rules can't be containers for other rules.

You can request to have a rule of your interest override another rule or a folder or a structure of folders. This particular functionality is explained in more detail in a section dedicated to this topic, further in this document.

By left clicking a rule its contents are accessed and the script is opened in the rule editor window.

This editor is integrated directly with the rules engine which provides a script debugger. This is where you would verify that your rule's script is written correctly and its logic would operate as intended.

Text editor is equipped with all the usual navigation facilities:

- a. copy, cut and paste functions are standard and respond to all the usual keyboard shortcuts
- b. text find is also quite standard and is controlled by the following keys
 1. Ctrl-F activates "find"
 2. ESC key cancels the effects of "find" and closes the "find" window
 3. F3 key performs "find next"
 4. Shift-F3 key performs "find previous"

Let's take a closer look at the debugger.

Rules editor/debugger

The screenshot displays the Rules editor/debugger interface. The main editor shows a rule script named 'adjust_status1' with the following code:

```

1 rule 'adjust_status1'
2 when
3     var x = ${employee};
4     x.Name == 'Actor Whose' && x.Age == 25;
5 then
6     if (x.Salary < 20)
7         x.Status1 = 2;
8     else if (x.Balance > 400)
9     {
10        x.Status2 = 'B';
11        x.Status1 = 9;
12    }

```

Below the editor, there are two tabs: 'compiler results' and 'symbols'. The 'compiler results' tab shows the following output:

```

status: SUCCESS
details: Pass 1. Lexical/Syntax analysis
        SUCCESS: No errors detected during compilation

```

The 'symbols' tab shows a table of symbols:

Name	Type	Value
x	unass...	
employee	unass...	
Name	unass...	
Age	unass...	
Salary	unass...	
Status1	unass...	
Balance	unass...	
Status2	unass...	
RETURN	UNKNOWN	

At the bottom of the interface, there are buttons for 'Designer' and 'Test Cases'.

Editing is allowed when the rule is not in “running” state. The editor is switched automatically into read-only mode once the script is successfully compiled.

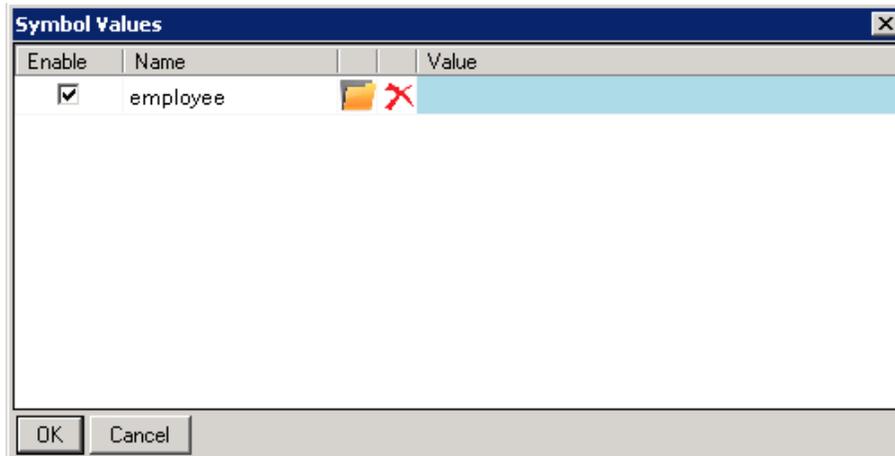


Compile button activates the compiler which parses and translates the script. Results of the compilation are produced and placed in the **compiler results** tab. By double clicking on any of the lines in this tab, more detailed view of the selected line is provided.

When no errors have been found during compilation process this button is deactivated and a set of execution type buttons to the right are activated. Also, the editor is switched to “read-only” mode.



Run button causes the script to be executed. When external variables are referenced by the script a data entry window is displayed to prompt the user to enter values for each referenced variable.



Example

- `employee` is the name of external variable that would be supplied by the external object and is expected by this rule.
- The data could be typed or fed into the rule by using an external file in which case you would click  button, browse and open the data file

NOTE: if no data is supplied on this screen the variable in question (in this case `employee`) would be “null”.



Stop button cancels the execution stage

- “Compile” button is activated again
- other buttons are de-activated, except for “No Duplicates”
- editor is switched back to allow editing and modifications of script



Reset button cancels the current script execution and resets it the state so it could be executed from the start.



Breakpoint button sets or cancels breakpoints which allow the execution to halt at a particular line. Red circles to the left of line numbers are used to identify breakpoints. When script execution reaches a line at which the breakpoint is active, the engine halts and displays values of all variables. These values would appear in the “symbols” window.



Single Step button is used to halt script execution at every line. It is a convenient, common and extremely useful feature to which the professional developers are so used. In regards to dealing with external variables the “Run” button functionality is shared, thus please refer to that section.



View Icode button is used to display the entire set of executable lines of script's code that are scheduled to be run. Using this feature requires software engineering skills and is reserved for our support team.



No Duplicates button is used to turn on and off the rule engine's feature that allows naming internal and external variables the same way. This feature is used in support of the legacy rules that have been migrated from other systems. Please contact our support team when dealing with this case.

If your script happens to contain `console.log` commands **console** window would contain output of these commands.

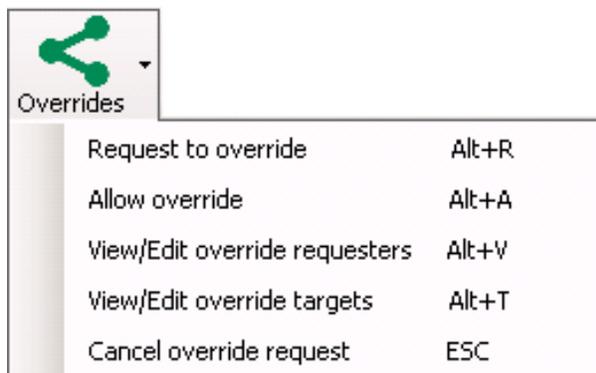
Overrides

“Overrides” is what actually makes rules a powerful concept.

Here is a common scenario.

- a. A product is built and tested by the engineers and QA testers and has gone through all sorts of processes and now it's released and deployed. Part of the product, let's say a function that calculates sales person's commission, is built as a rule.
- b. This particular rule's script is written by an engineer and the formula on which the product's business owner agreed is a simple calculation based on some percentage, let's say 10% flat rate.
- c. Several weeks after the product was installed business manager decides that this rule shouldn't use flat percentage but instead the percentage should fluctuate depending on time of the year and maybe other factors.
- d. Because the rule that came with the product is locked against any changes, which is a normal practice, business manager would need to create a new rule implementing his new algorithm. After it's tested and the manager is satisfied with his test results he/she would direct the rules engine to override the original rule with the new one.
- e. At any point the override setting can be removed and the original rule could be reverted to, thus restoring the original functionality.

“Override” is what takes rules based products to the level of “self-service”. We encourage you to analyze this scenario and come up with more of your own.



“Override” functions could be accessed at the top panel or via context sensitive menus.

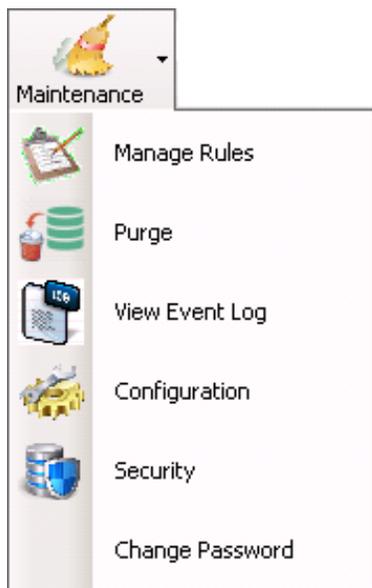
These menus become available when you set focus on folders or rules in the left pane.

NOTE: rulesets can't be part of this feature

Security

Security policies and processes of the rules engine and X-ruleStudio are based on rights. The rights that would allow or deny you access to certain features create the makeup of your security profile. That's why they are called "access rights".

Rights could be added to and removed from any user in the system. Certain combination of rights would form roles that could also be added to and removed from any user.



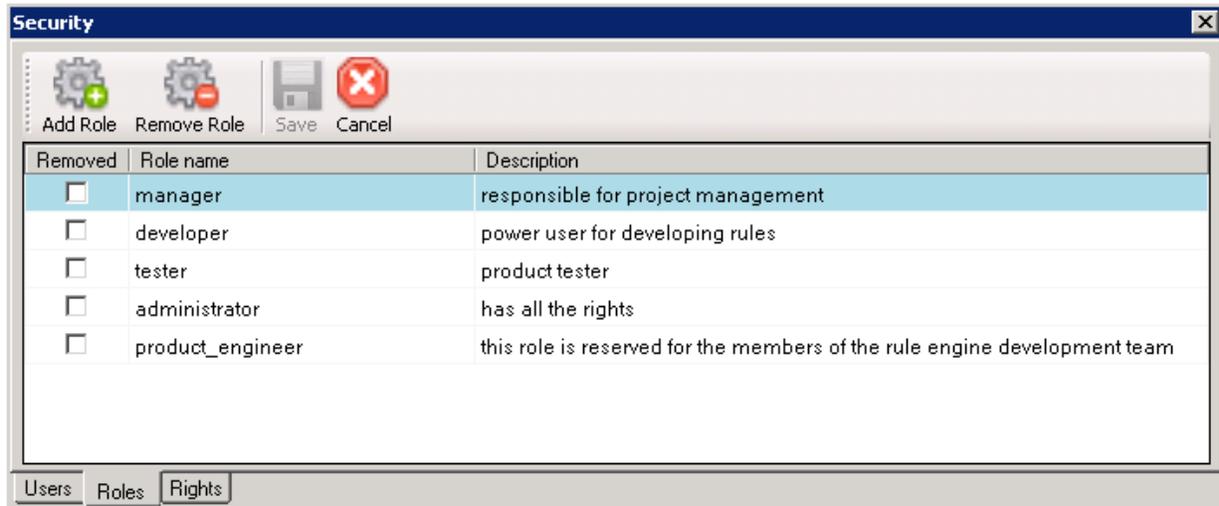
Access to the maintenance functions requires certain privileges. Only administrators and those with specific rights are allowed to use these facilities.

Here is a list of rights that control maintenance functions.

Function	Rights required
 Manage Rules	manage_rules
 Purge	purge_rules and/or purge_folders
 View Event Log	view_event_log
 Configuration	manage_configuration
 Security	Any combination of: create_user_profiles, modify_user_profiles, remove_user_profiles, restore_user_profiles, create_roles, remove_roles, create_rights, remove_rights, reset user password
Change Password	Any user who is allowed to change their own password

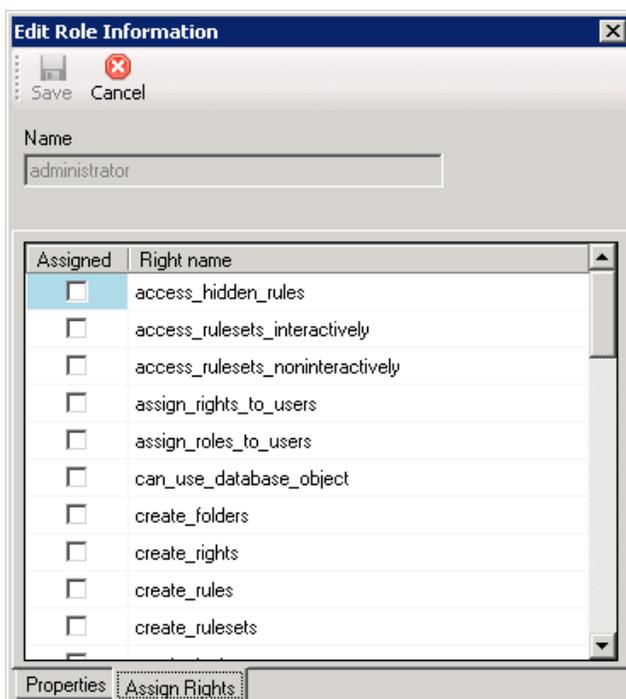
Roles

Roles management functions are available from  Security section of the maintenance button.



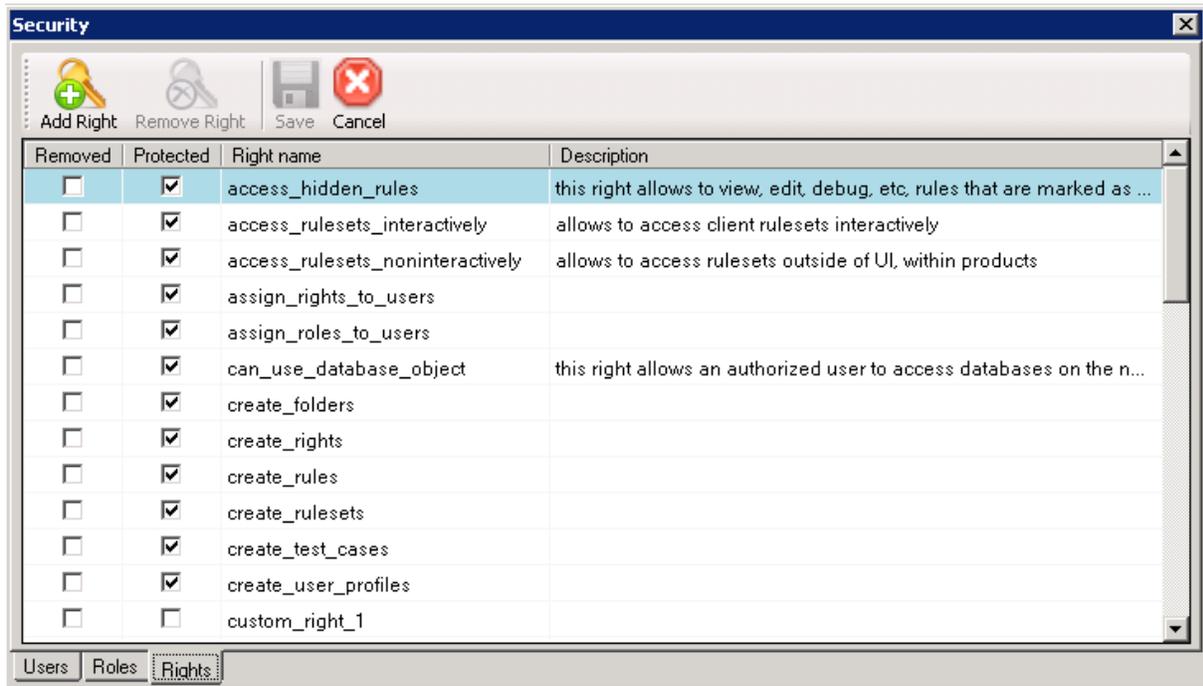
administrator and product_engineer are the only built-in roles that can't be removed, all other roles can be removed and modified and new roles can be created.

Any combination of rights can be added to or remove from any role, except the administrator.



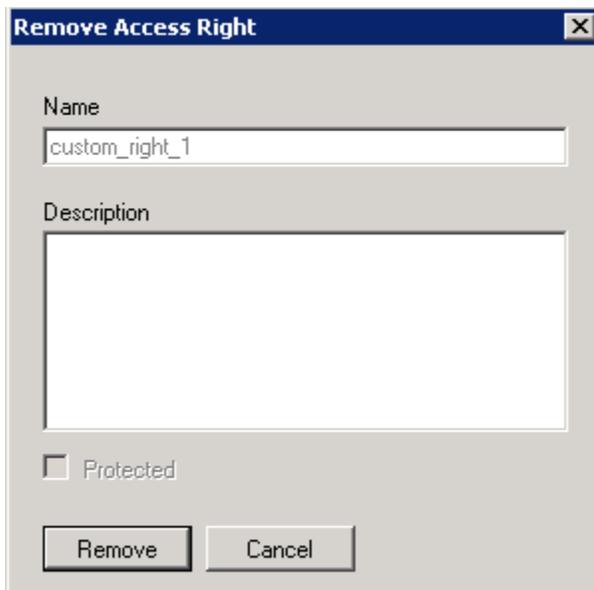
By double clicking on a role you would be able to access that role's profile and manage it. Dialog to the left is where you would be adding and removing access rights.

Rights



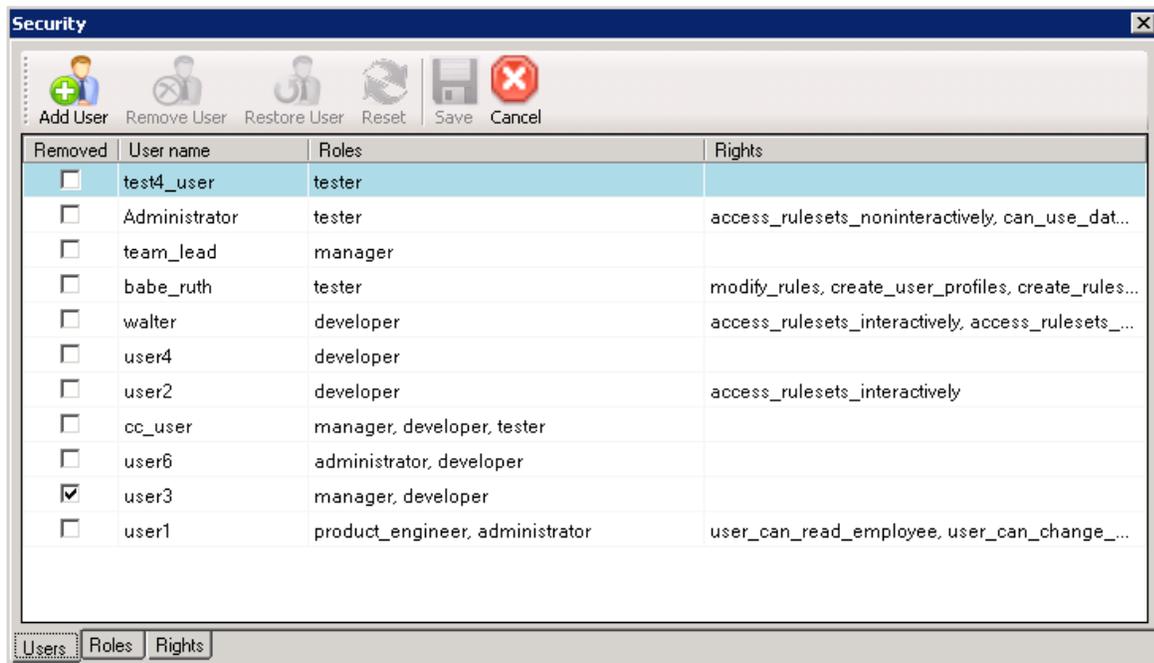
Build-in rights that are supplied with the system are considered “core” rights and cannot be removed and their protection flag cannot be turned off.

New custom rights can be created and then removed. When you attempt to remove an existing right you will be asked to verify your intention with this dialog:

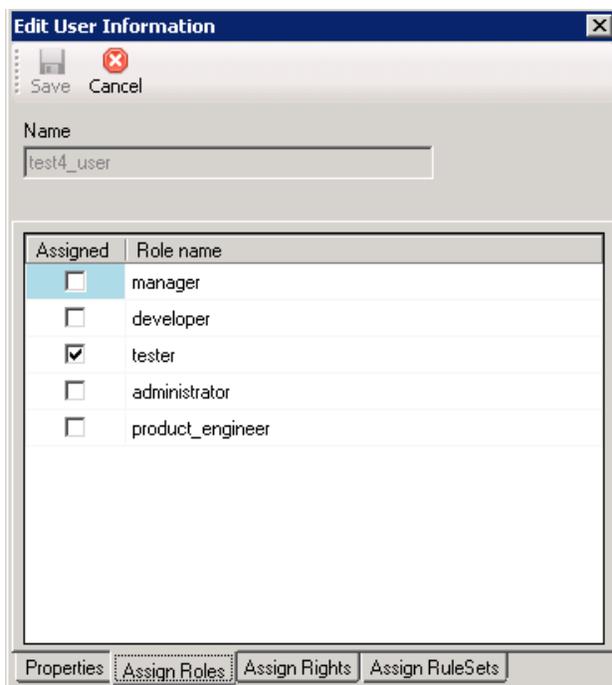


Click “Remove” button to confirm.

Users



You can access the user's profile (properties, assigned roles, rights and rulesets) by double clicking on user.



Select an appropriate tab at the button to gain access to various security settings within the selected user's profile.

