

Replied Mail View

View Other Replies asivakumar ▾

Rate This Reply

Invalid ▾

saveeditforwardshow mailadd to featuresadd to KBadd to patchadd to promisesclose

From AdventNet security-support<security-support@adventnet.com>
Replied By **asivakumar@adventnet.com**
To Cpecina@acmepacket.com
rbodathula@velankani.com
jjerry@velankani.com
CC ssahu@velankani.com
rkarthik@velankani.com
jscali@acmepacket.com
security-support@adventnet.com
Date 2005-09-01 00:06:23 Asia/Calcutta
Subject [Ticket ID : 11717367] Re:ACMEPACKET: Confusing select options for operations in the Operation tree

Priority: 3 (Normal)

NormalView

Ticket ID : 11717367

Ticket Title : Re:ACMEPACKET: Confusing select options for operations in the Operation tree

Dear Chris & team,

This is an interesting discussion and let me try my best to answer your question. Before that i would like to provide you an short update.

For this servicepack release, in webclient we planned to provide complete security administration support. This includes User, Group and Operation configuration support in webclient itself. At the same time, velankani has asked us to provide only two STATE for operation type(for java client security administration). Therefore we planned to have only TWO STATE for operation type: TICK and CHECKED boxes and will eliminate the BLANK state. In the webclient security administration(new) , we started implementation with only two state and the same will be maintained for the java client too. So we started analysing the feasibility for the same and started exploring the functionality.

Here comes the answer for your queries. Lets start with definitions:

Operations can be configured at TWO level:

1. Group Specific level. Operation will be assigned by selecting the group.
2. User Specific level. Operation will be assigned by selecting the User.

1. Definition for the Check box states:

CHECKED - Operation not permitted for the Group/User. FIXED.

TICKED - Operation permitted for the Group/User. FIXED

BLANK/UNCHECKED - Onceagain operation is not permitted for the Group. But **MUTABLE** for Users through USER SPECIFIC permission setting. Not clear!! Let us see this with examples:

Consider Group **Group_1** has the following permission setting:

1. Security Administration - CHECKED - (i.e) Not authorized to perform security administration work.
2. System Administration - CHECKED - Not authorized
3. Alert User Operation - TICKED - Authorized to perform the alert related operation.
4. Event User Operation - TICKED - Authorized
5. Map Editing Operation - TICKED - Authorized

Also there are bunch of Operations like

6. PollingObjects
7. Polling Units
8. Provisioning
9. Threshold Object, etc. All these operations will fall under **UNCHECKED** state.

According to Authorization engine, for Group **Group_1** only Operation (3), (4) and (5) are authorized and all other permissions are considered **not authorized**.

Now User **guest** is assigned to (Group) **Group_1**. Therefore the group permission settings of **Group_1** will be applied to **guest**. Now **guest** badly need a permission to configure the PollingObjects but the Authorization engine will not permit the **guest** to configure because UNCHECKED state for **Group_1** is considered as **not authorized**.

What is User Specific Permission:

For a given User, It is possible to assign additional (User specific) permission apart from the group inherited permissions with the following rule applied:

RULE 1: If any user is assigned to group, it is not possible for user to modify the Group inherited permission. (ie) In the security admin UI, TICK & CHECKED will be disabled for the User operation settings in the operation tree.

RULE 2: Only the operation that are in BLANK/UNCHECKED (undefined) state is allowed for modification for assigning the user specific permission. Note: Permission can be Include or Exclude.

Great!. Now i can assign the **PollingObjects** operation to user **guest** through **User specific permission** setting as the operation is in UNCHECKED state for group **Group_1**.

Advantages of this model:

1. For a User/Group, only operations that are Included and Excluded (TICK and CHECKED respectively) will be maintained in the SecuritydbData.xml and in the Database. UNCHECKED state will not be maintained. **This saves us the space and increases the processing speed. We will not maintain all the operations status for all the users/group.**

2. Has two level of operation permission settings 1. Group Specific Level and 2. User Specific Level

If we maintain only TWO STATE:

<!--StartFragment -->Consider for a given group, all the available operations(Consider if we have hundreds of operations) will fall in any one of the two state. (TICK and CHECKED/UNCHECKED).

1. Currently for the Group/User, only the Included and Excluded state will be maintained in the Database and in the SecurityDBdata.xml file. The undefined state will be not be stored in the database/xml file. If we removed the undefined(UNCHECKED) state, all the available permission has to be stored in the database/xml file for every user & group. This will increase the storage space and the processing of the XML, which will be a performance hit if we have more number of users/groups and permissions.

2. The User based operations setting functionality will be lost completely as we will disable the operation corresponds to the group. If we attempt to have two state, this will be a functionality breakage where customer will never accept. I believe you too.

Finally, we believe the existing functionality should be preserved with the following changes:

<!--StartFragment -->1. F1/Help link explaining the usage and the feature of Operation tree and the reason for maintaining the three states.

2. Having Tooltip & legend/Description in the Operations tree panel

3. Currently for the *USER *permission setting panel, there is no demarcation between the disabled(Inherited from the assigned groups) and the editable fields. The disabled fields should be shaded with different color/image.

4. All the above will be backed by help documentation in our developer guide.

We believe the above stand is acceptable.

Still you would like to have only two state, kindly write to us. Since security administration is an open source project, customer can modify on thier own. We will guide you for that. Kindly note that you cannot runaway from the demerits we listed above.

Even we can have a conference call to resolve this issue.

Kindly get back to us for further clarification in this regard.

Thanks and regards,

A Sivakumar

Oops!!. In webclient, (html form)checkbox will maintain only two state. I have to use other methodology to

achieve the feature of having three states.

>----- <Cpecina@acmepacket.com> wrote: -----

Jerald,

I do not understand your explanation.

Here is my definition of checkbox states:

Checked: operation is allowed

Not checked: operation is not allowed

What is your definition?

You comments on whether or not the data is saved to the database have no bearing on the user.

What does the user care about our internal storage mechanism choices? What meaning is an empty checkbox supposed to convey?

Also, I have no idea what you mean by the following statement: users will not be able to differentiate between the operations that are excluded at grouplevel and the operations that are not defined at all. This will add more confusions to the end user.

Operations that are not defined should not appear in the list at all. What are you talking about?

Please clarify the AdventNet position on checking, x'ing out, or leaving empty the boxes in the security administration operations screen.

Chris

Ticket ID : 11717367

Ticket Title : Re:ACMEPACKET: Confusing select options for operations in the Operation tree

Hi Ramu,

We analysed the requirement. We find that the three states present now (Empty, Check & UnCheck), are essensial to capture the inclusion / exclusion of the operations at database level.

As you might be aware, only the Check and UnCheck operations will be stored in the database. Operations denoted by empty icons will not be stored in the database. Not having the Empty state, all the operations should be stored in the database and it will lead to heavy performance loads that might degrade the responsiveness of the client. Hence it is very necessary to maintain three states.

We understand that this requirement raises, because having three states, end users may not be clear of what each state means. This might be due to a less clear documentation. Hence we can address this requirement by making the description clear that it does not confuse the end user and delivers the purpose of each state.

Apart from the above reasons, there is one more disadvantage of having only Empty and Check states. ie: users will not be able to differentiate between the operations that are excluded at group level and the operations that are not defined at all. This will add more confusions to the end user..

If you still believe, that you would like to have only Empty and Check states, it can not be done at the state level, since it might affect the performance directly. But that can be done at the icon level, by displaying the same icon (empty icon) for both the states (Empty and UnCheck). This will need few source changes in the Security Administration project. But this will result in the previously mentioned disadvantage. Please let us know if you still would like to go for this option.

If you need more clarification, please let us know the same. We can even schedule a conf call, if necessary.

Thanks & Regards,
Jerald.
Dear Support,

I haven't got any response for this issue too.

Please let me know the status.

Regards,

Ramu

Dear Support,

We are using Acme packet License.

There are 3 possible selections on an operation

in a Operation tree

1. Empty ("")
2. Cross (X)
3. Tick (V)

These are basically confusing for an end user.
Can we just have the
first (Empty) and the last one (Tick) alone?

Since we are customizing the security
administration project in
AdventNet, Please let us know what to modify to
achieve this.

Regards,

Ramu

=====

Ramu Bodathula

Velankani Software Private Ltd.,

43 Electronics City, Phase -2

Hosur Road, Bangalore 561 100

Phone: 91.80.28522507/509 ext 4062

Direct : 91.80.25014062

Mobile: 91.9845594663

Fax : 91.80.25014444

Home : 91.80.26556888

email : rbodathula@velankani.com

URL : www.velankani.com

"Every Customer is a Reference Customer"

=====