(http://toolkt.com/site)

FEB
12

♥
2

f
(http://www.facebook.com/sharer/sharer.php?u=http://toolkt.com/site/#portfolio-page)
(http://twitter.com/share?url=http://toolkt.com/site/#portfolio-page&lang=en&text=Check%20out%20this%20awesome%20project:&)
g+
(https://plus.google.com/share?url=http://toolkt.com/site)

Security in OpenERP

Security is a major concern when any application is considered. As of OpenERP, security is an important factor and it is easy to provide security in modular basis.

Group-based Access Control

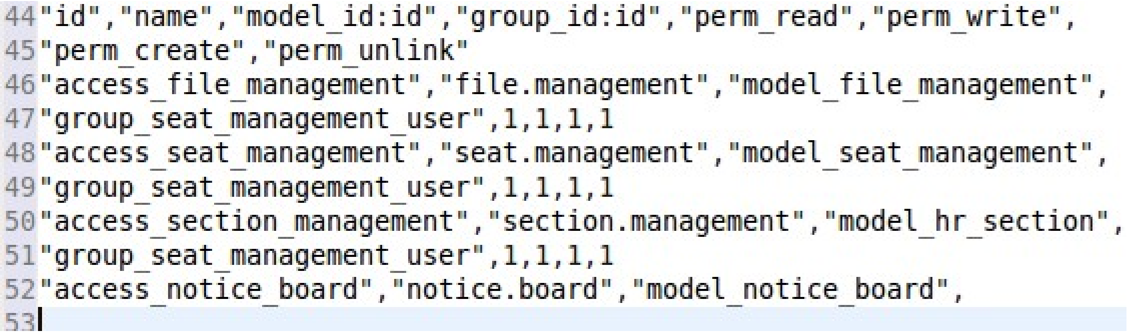
Groups can be created as normal records and menu access may be granted via menu definitions. However, even without a menu, objects may be accessible indirectly. In order to prevent this, object level permissions must be defined for groups. These permissions are usually inserted via CSV files in the security folder inside modules. The example below provide a basic view of providing group based access rights.

Defining Groups

```
<record id="group_seat_management_user" model="res.groups">
    <field name="name">Seat Administration / Manager </field>
</record>

<record model='ir.ui.menu' id='menu_seat_management'>
    <field name="groups_id" eval="[(6,0,[ref('group_seat_management_user')])]" />
</record>
```

Defining Access Rights (ir.model.access.csv)



```
"id","name","model_id:id","group_id:id","perm_read","perm_write",
"perm_create","perm_unlink"
"access_file_management","file.management","model_file_management",
"group_seat_management_user",1,1,1,1
"access_seat_management","seat.management","model_seat_management",
"group_seat_management_user",1,1,1,1
"access_section_management","section.management","model_hr_section",
"group_seat_management_user",1,1,1,1
"access_notice_board","notice.board","model_notice_board",

```

- **id** = unique identity for the permission (Example: MY_MODULE_res_partner_manager)
- **name** = unique name for the permission (Example: res_partner manager)
- **model_id/id** = the model unique name of the class you want apply permission on (Example model_res_partner)
- **group_id/id** = group permission apply on (yopu can define it in xml group file or call an existing group with syntax module.group_id)
- **perm_read,perm_write,perm_create,perm_unlink** = the 4 values for the relative permission to read, write,create,unlink record on defined class. 1 is True (you can do this action) and 0 is Faslse (you can't)

A simple way to understand this file is to read an existing csv in some OpenERP base module like sale, account, product, etc…

Menu Access definition

```
<menuitem "submenu_seat_management" name="Seat Administration"
action="action_seat_management" parent="file_menu_config"
groups="group_seat_management_user" />
```

Also, don't forget to include the xml files in the OpenERP Descriptor File.

Access Rules

Access Right (http://bazaar.launchpad.net/~openerp/openobject-addons/7.0/view/head:/project/security/ir.model.access.csv#L2) This is how you can give read/write/create/delete rights in group on particular object by creating `ir.model.access.csv` file. See line number 1 & 2.

Access Rules (http://bazaar.launchpad.net/~openerp/openobject-addons/7.0/view/head:/project/security/project_security.xml#L50) This is how you can create access rules for particular object and groups by creating xml file.

For example:

```
    <record model="ir.rule" id="ir_values_my_costume_rule">
        <field name="name">My Rule Name</field>
        <field name="model_id" ref="model_your_model_name"/>
        <field name="domain_force">[('field','operator','value'),('user_id','=',user.i
d)]</field>
        <field name="perm_read" eval="True"/>
        <field name="perm_write" eval="True"/>
        <field name="perm_unlink" eval="True"/>
        <field name="perm_create" eval="True"/>
    </record>
```

You must pass `model_` before model name in `<field name="model_id" ref="model_your_model_name"/>` like this: `model_sale_order` or `model_project_task`.

Here in `eval` you can either pass `True` or `False` as per you need.

Record Rules can be defined from the menu also without creating any file : **Settings->Technical->Security->Record Rules.**

There are three main fields that you need to configure carefully in order to define "Record Rule"

1. **Object:** On which you want to apply record rule. (in this example it is "Task" object).
2. **Domain:** Setup domain for filtering the data.
3. **Groups:** Add group for which you want to apply this record rule. If nothing to add then this rule is apply globally which is usually used to configuring multi-company record rule.

I am going to explain such access rule by taking the example of "Task" object of OpenERP.

In my example, suppose my requirement is like this:

The user 'rch' can access only list of tasks of following kind…

1. list of all tasks which is not assigned to any user. i.e.('user_id','=',False)
2. list of all tasks Which is assigned to user 'rch'. i.e.('user_id','=',user.id)

3. list of all tasks of all the project’s for which he is a member of. i.e.(‘project_id.members’,’in’, [user.id])
4. list of all tasks of the project for which he is a project manager. i.e.(‘project_id.user_id’,’=’,user.id)

Configure your record rule as follow:

1. Name: Tasks According to User and Project.
2. Object: Task.
3. Domain: [‘|’,’|’,’|’,(‘user_id’,’=’,False),(‘user_id’,’=’,user.id),(‘project_id.members’,’in’, [user.id]),(‘project_id.user_id’,’=’,user.id)].
4. Groups: project/User.

Now add this group (project/User) to user ‘rch’.

Be the first to post a comment.

How to Increase Disk Space in VMware

(http://toolkt.com/site/how-to-increase-disk-space-in-vmware/)