

# LOCKDOWN



version 0.9.5 and beyond...

# Options

**who\_did\_it**

:current\_user\_id

**session\_timeout**

1 hour

**session\_timeout\_method**

:clear\_session\_values

**logout\_on\_access\_violation**

false

**default\_who\_did\_it**

usually a constant like SYSTEM

**access\_denied\_path**

/

**successful\_login\_path**

/

**links\_separator**

'|'

# Helper Methods

## **add\_lockdown\_session\_values(user)**

sets current\_user\_id and access\_rights in session

## **current\_user\_id**

session[:current\_user\_id]

## **logged\_in?**

return true if current\_user\_id.to\_i > 0

## **reset\_lockdown\_session**

[:expiry\_time, :current\_user\_id, :access\_rights]

## **session\_access\_rights\_include?(controller\_action)**

returns true if access\_rights array include? “#{controller}/#{action}”

# More helper methods

**links( link\_to('A',some\_a\_url), link\_to('B',some\_b\_url))**

returns only the links that the user has access to

**current\_user\_access\_in\_group?(:group\_symbol)**

Returns true or false if user has access to ANY permission in the group. It does NOT require that the user has access to ALL permissions associated to the group.

**current\_user\_access\_in\_perm?(:permission\_symbol)**

Returns true or false if user has access to ANY controller/action in the permission. It does NOT require that the user has access to ALL controller/actions associated to the permission.

**current\_user\_is\_admin?**

Simple enough.

# init.rb ... RULES!

# Simple

```
set_permission(:manage_products).with_controller(:products)
```

# Only Methods

```
set_permission(:my_account).  
  with_controller(:users).  
    only_methods(:edit, :update, :show)
```

# Except

```
set_permission(:manager).  
  with_controller(:users).  
    except_methods(:destroy)
```

# init.rb ... RULES EVEN MORE!

```
# Chain 'em for a ridiculously long permission.  
# Note: Context is kept, so it will throw an exception if you call a method out of  
sequence.
```

```
set_permission(:ridiculous).  
  with_controller(:a_controller).  
  and_controller(:b_controller).  
    only_methods(:method_a, :method_b)  
  and_controller(:c_controller).  
    except_methods(:method_x).  
  and_controller(:d_controller).  
  and_controller(:e_controller)
```

```
# User Groups are easy  
set_user_group(:user_group_name, :perm_1, :perm_2, :perm_3)
```

**\*\*Permissions and User Groups are synchronized with database.**

# init.rb ... FUTURE!

```
set_permission(:my_account).  
  with_controller(:users).  
    only_methods(:show, :update).  
  to_model(:user).  
    where(:current_user_id). # method available in controller scope  
    equals(:id) # method on model
```

```
set_permission(:manage_timesheet).  
  with_controller(:timesheets).  
    only_methods(:show, :update).  
  to_model(:timesheet).  
    where(:current_user_id).  
    is_in(:editor_ids)
```

## More Info

<http://stonean.com/wiki/lockdown>

<http://rubytrends.com>