

Plugin installation

Redmine installation procedure is explained at <http://www.redmine.org/projects/redmine/wiki/redmineinstall>.
To install one or more AlphaNodes plugins, run **all** 7 steps.

1. Backup

Backup of your **database** and all your **Redmine directories**, first !

2. Requirement and system preparation

2.1 Requirements

Make sure that the following requirements are fulfilled:

2.1.1 Minimum requirements

- Redmine version = 4.1.x
- Ruby version >= 2.5.x
- Database
 - MySQL >= 5.7 *make sure all database tables using the same storage engine (InnoDB is recommended) and character set (utf8mb4 is recommended). You can use our article [Redmine_with_MySQL](#) to fix it.*
 - or PostgreSQL >= 9.6

2.1.2 System recommendation

- Redmine version: 4.1.3
- Ruby version: 2.6.7
- Database: PostgreSQL 13
- Operating system: Linux 64bit (e.g. Debian)

2.2 System preparation

Delete previous installed AlphaNodes plugin:

```
cd $REDMINE_ROOT/plugins
rm -rf
additional additional_tags redmine_reporting redmine_db redmine_hrm redmine_passwords redmine_wiki_guide
```

Remove all plugins, which are not compatible with the new plugins.
Learn more about [Plugin compatibility](#) with other plugins in our FAQ.

2.3 Check

- Do all requirements fit and are all system preparations done?
- Start with 3.

3. Plugin/Package installation

Alert

Make sure, that (if you are using more commercial AlphaNodes plugins) they are all available in the **same version**.

3.1 Package installation

All *AlphaNodes Redmine plugins* require to install [additional](#) plugin first. It is available for free at GitHub.

```
cd $REDMINE_ROOT
git clone -b stable https://github.com/alphanodes/additional.git plugins/additional
```

After this, put all your AlphaNodes plugins, you want to install to directory \$REDMINE_ROOT/plugins and unzip the plugin there.

3.2 Check

- Do the plugins exist in the correct directory? \$REDMINE_ROOT/plugins/PLUGIN_NAME/init.rb
- Are all plugins installed to: \$REDMINE_ROOT/plugins
- Proceed with 4

4. Run Bundle

4.1 Bundle

Install / update the necessary Ruby (Gem) packages:

```
cd $REDMINE_ROOT
bundle config set --local without 'development'
bundle update
```

If you installed Redmine with the official documentation [RedmineInstall#Step-4-Dependencies-installation](#), existing [bundler](#) configuration will be used.

4.2 Check

- If no error occurred go to 5.
- Did an error occur? Fix the error, first. Only run next step, if you solved the error.

5. Database changes

5.1 Run plugin migration

Enter the following code in order to make necessary database changes:

```
cd $REDMINE_ROOT
bundle exec rake redmine:plugins:migrate RAILS_ENV=production
```

5.2 Check

- If no error occurred go to 6.
- Did an error occur? Fix the error, first. Only run next step, if you solved the error.

6. Plugin specific tasks

Some plugins need special tasks for installation / update.
Go through this section one by one.

- If there is a plugin that you have, do the following activities
- If there is a plugin that you do not have, skip the task and proceed with the next

6.1 Plugin: redmine_automation

6.1.1 Create default data (time periods)

If you had already installed a previous version, skip this task.

```
cd $REDMINE_ROOT
bundle exec rake redmine:automation:defaults RAILS_ENV=production
```

6.1.2 Setup cronjob

It is necessary to set up a cronjob on the server for the automatic execution of the tasks.

For this purpose the following *rake* task must be executed:

```
* /5 * * * * cd $REDMINE_ROOT && RAILS_ENV=production bundle exec rake redmine:automation:jobs # run all async jobs
```

This checks every 5 minutes whether automated tasks are available for execution or not.

So the smallest unit is 5 minutes - if a smaller unit is used in the rules, they are still only executed every 5 minutes.

If you want to use different time queries for the rules, repeating issues and repeating workload bookings, you can use the following instead.

This is useful, for example, if you want to reduce server load.

```
0 */2 * * * * cd $REDMINE_ROOT && RAILS_ENV=production bundle exec rake redmine:automation:rules # run rules
*/5 * * * * cd $REDMINE_ROOT && RAILS_ENV=production bundle exec rake redmine:automation:repeat_issues # run issue repeats
*/30 * * * * cd $REDMINE_ROOT && RAILS_ENV=production bundle exec rake redmine:automation:repeat_time_entries # run time entry repeats
```

The areas are executed individually, the task *redmine:automation:jobs* must not be executed again!

Again for clarification: either *redmine:automation:jobs* OR *redmine:automation:rules*, *redmine:automation:repeat_issues* and *redmine:automation:repeat_time_entries*.

6.2 Plugin: redmine_db

Nothing to do for this plugin, go to next step.

6.3 Plugin: redmine_passwords

If you had already installed a previous version, skip these tasks.

6.3.1 Initial settings

Start the initial settings by using the following code:

```
cd $REDMINE_ROOT
bundle exec rake redmine:passwords:defaults RAILS_ENV=production
```

6.3.2 Encrypt sensitive data (optional)

redmine_passwords version 3.0.3 or newer uses Redmine default functionality to encrypt sensitive data. We strongly recommend to run our script [Encrypting-password-fields](#) to encrypt your passwords.

6.4 Plugin: redmine_wiki_guide

To parse existing wiki pages for link creation, run:

```
cd $REDMINE_ROOT
bundle exec rake redmine:wiki_guide:parse_links RAILS_ENV=production
```

6.5 Plugin: redmine_devops

Nothing to do for this plugin, go to next step.

6.6 Plugin: redmine_reporting

If you had already installed a previous version, skip this task.

In order to make *Redmine Reporting* work with your previous data it is necessary to migrate them.

The migration preprocesses existing data information and incorporates them into the current analysis.

```
cd $REDMINE_ROOT
bundle exec rake redmine:reporting:migrate RAILS_ENV=production
```

6.7 Plugin: redmine_hrm

Nothing to do for this plugin, go to next step.

7. Restart application server

Finally you need to restart the application server (e.g. [Passenger](#), [unicorn](#) or [puma](#)). For Apache with Passenger under Debian or Ubuntu use the following code:

```
systemctl restart apache2
```

Done! The latest plugin has been successfully installed and is listed under the following URL [/admin/info](#) .

- Go to the Redmine administration area for further plugin configuration in the plugin settings. Check the user permissions (section "Roles & Permissions").

Back to your [Projects](#)

TAGs

Documentation