

---

# Installationsanleitung für Redmine

Stand: 01.11.2016

Jan Niggemann <jan.niggemann@hz6.de>

Copyright © 2013-2016 Jan Niggemann

## Lizenz



Diese Anleitung ist unter CC BY-NC-SA 3.0 DE<sup>1</sup> lizenziert.

## Zusammenfassung

Anleitung um Redmine (inkl. Apache, MySQL, Passenger, Ruby) unter Debian Jessie zu installieren

## Inhaltsverzeichnis

1. Ruby, Rails, Gems und Passenger .....	2
1.1. Vorbereiten des Systems .....	2
1.2. Ruby herunterladen, bauen und installieren .....	2
1.3. Prüfen ob Ruby funktioniert .....	3
1.4. OpenSSL für Ruby bauen .....	3
1.5. Ist der Befehl Gem jetzt installiert? .....	3
1.6. Rails installieren .....	3
1.7. Passenger installieren (Applicationserver) .....	3
1.8. Apache konfigurieren .....	4
1.9. Das Modul Passenger aktivieren .....	4
2. Redmine installieren .....	4
2.1. Redmine herunterladen .....	4
2.2. Das System weiter vorbereiten .....	4
2.3. Datenbank anlegen .....	4
2.4. Datenbankverbindung einrichten .....	4
2.5. Secret Token erstellen .....	5
2.6. Datenbank füllen (Tabellen anlegen) .....	5
2.7. Zugriffsrechte auf File-Ebene setzen .....	5
2.8. Redmine Installation prüfen .....	5
3. Apache und Passenger konfigurieren .....	5
3.1. Apache konfigurieren (subURI deployment) .....	5

---

<sup>1</sup> <http://creativecommons.org/licenses/by-nc-sa/3.0/de/deed.de>

# 1. Vorwort

Nach dem Lesen dieser Anleitung und dem Ausführen aller Schritte sollten Sie eine funktionierende Redmine-Instanz haben. Wenn etwas nicht funktioniert oder ein Schritt fehlschlägt, erstellen Sie bitte einen Post mit Details im Forum (Englisch)<sup>2</sup> und beziehen Sie sich auf diese Anleitung und die Schritt-Nummer. Ihr Post sollte auch die Ausgabe von `gem env` und/oder `RAILS_ENV=production / opt/redmine/script/about` enthalten. Wenn Sie einen Fehler entdecken, informieren Sie mich bitte per email.

Ich warte dieses Handbuch in meiner Freizeit und freue mich über eine kleine Aufmerksamkeit von

meinem  <sup>3</sup> oder Spenden via  <sup>4</sup>.

# 2. Über diese Anleitung

Dies ist eine Schritt-für-Schritt Anleitung um Redmine unter "Debian stable" (im Moment "Jessie" genannt) zu installieren. Sie ist für Menschen geschrieben die sich mit Debian, der Shell, MySQL, Apache und Linux generell schon etwas auskennen. Um diese Anleitung kurz zu halten wird nicht jeder einzelne Schritt im Detail erklärt. Filenamen und Pfade in dieser Anleitung können sich ändern. Ich tue mein Bestes um die Daten aktuell zu halten, aber bitte prüfen Sie was Sie tun. Bitte beachten Sie: Ich gehe davon aus das Sie wissen wie Sie Ihr Werkzeug benutzen müssen. Sollte das nicht der Fall sein (wenn Sie z.B. nicht wissen wie man den MySQL Client benutzt oder Apache neu startet) benutzen Sie bitte die Suchmaschine Ihrer Wahl, informieren Sie sich, und lesen Sie erst dann weiter.

# 3. Textauszeichnungen in diesem Dokument

Text type	Bedeutung
monotype font	Benutzerein- bzw. -ausgaben und Konfigurationsdateien
↵	Zeile geht unten noch weiter

# 1. Ruby, Rails, Gems und Passenger

## 1.1. Vorbereiten des Systems

```
apt-get install gcc build-essential zlib1g zlib1g-dev zlibc ↵
libssl-dev libyaml-dev libcurl4-openssl-dev ↵
apache2-mpm-prefork apache2-dev libapr1-dev ↵
libxslt-dev checkinstall 
apt-get build-dep ruby 
```

## 1.2. Ruby herunterladen, bauen und installieren

Bitte nach jeder Zeile  drücken

```
cd ~
wget -c ftp://ftp.ruby-lang.org/pub/ruby/2.3/ruby-2.3.1.tar.gz
tar xvfz ruby-2.3.1.tar.gz
```

<sup>2</sup> <http://www.redmine.org/projects/redmine/boards/2>

<sup>3</sup> <http://www.amazon.de/registry/wishlist/GAOFFPW3WLH2>

<sup>4</sup> [https://www.paypal.com/cgi-bin/webscr?cmd=\\_s-xclick&hosted\\_button\\_id=GLW5CMTZFBP8A](https://www.paypal.com/cgi-bin/webscr?cmd=_s-xclick&hosted_button_id=GLW5CMTZFBP8A)

```
cd ruby-2.3.1
./configure --enable-pthread --prefix=/usr/local
make
su
checkinstall --type=debian --install=yes --fstrans=no --pakdir='~'
```

### 1.3. Prüfen ob Ruby funktioniert

```
ruby -v 
```

Die Ausgabe sollte ungefähr so aussehen:

```
ruby 2.3.1pXXX (2016-04-26 revision XXXXX) [x86_64-linux]
```

### 1.4. OpenSSL für Ruby bauen

Bitte nach jeder Zeile  drücken

```
cd ext/openssl/
ruby extconf.rb
make
su
checkinstall --type=debian --install=yes --fstrans=no --pakdir='~'
```

### 1.5. Ist der Befehl Gem jetzt installiert?

Seit Ruby 1.9 ist RubyGems per default mit dabei, daher sollte jetzt der Befehl "gem" verfügbar sein. Wenn die Installation fehlerfrei war wird der folgende Befehl eine Versionsnummer ausgeben (z.B. 1.8.2x):

```
gem -v 
```

Jetzt können wir rdoc installieren:

```
gem install rdoc 
```

### 1.6. Rails installieren

Anforderungen hier prüfen: RedmineInstall<sup>5</sup> und die benötigte Version beim installieren angeben (andernfalls wird die neueste Version installiert)

```
gem install rails -v=4.2.7.1 --no-ri --no-rdoc 
```

#### **Anmerkung**

Eventuell erhalten Sie die Fehlermeldung "*no such file to load --zlib (LoadError)*". In diesem Fall müssen Sie zuerst zlib installieren:

```
cd ruby-2.3.1/ext/zlib/
ruby extconf.rb
make
make install
```

### 1.7. Passenger installieren (Applicationserver)

```
gem install passenger 
passenger-install-apache2-module 
```

---

<sup>5</sup> <http://www.redmine.org/projects/redmine/wiki/RedmineInstall>

## 1.8. Apache konfigurieren

Folgendes in `/etc/apache/mods-available/passenger.load` einfügen (wenn nötig Pfade anpassen).

**Das Folgende ist eine einzelne Zeile!**

```
LoadModule passenger_module ↵  
/usr/local/lib/ruby/gems/1.9.1/gems/passenger-4.0.46/ext/ ↵  
apache2/mod_passenger.so
```

Und das hier kommt nach `/etc/apache/mods-available/passenger.conf` (wenn nötig Pfade anpassen).

```
PassengerRoot /usr/local/lib/ruby/gems/1.9.1/gems/passenger-4.0.46  
PassengerRuby /usr/local/bin/ruby  
PassengerDefaultUser www-data
```

## 1.9. Das Modul Passenger aktivieren

```
a2enmod passenger 
```

# 2. Redmine installieren

## 2.1. Redmine herunterladen

Holen Sie das neueste zip von [redmine.org](http://redmine.org)<sup>6</sup> und extrahieren sie es nach `/opt/redmine`

## 2.2. Das System weiter vorbereiten

Hinweis: Die Installation von `libmagickwand-dev` installiert eine Menge Pakete (`depends / recommends`). Falls ein Fehler bei der Installation des bundles auftritt, können Sie versuchen, die `-dev` Pakete zu bereinigen (`purge`).

Bitte nach jeder Zeile  drücken

```
gem install bundler mysql2  
apt-get install libmagickwand-dev libmysqlclient-dev  
cd /opt/redmine  
bundle install --without postgresql
```

## 2.3. Datenbank anlegen

Vorraussetzung für diesen Schritt ist, dass Sie MySQL bereits installiert haben. Sollte das nicht der Fall sein, installieren Sie bitte MySQL mit dem Paketmanager Ihrer Distribution. Die folgenden Befehle müssen Sie in Ihrem MySQL client ausführen, nicht auf der Shell! Wenn Sie nicht wissen wie Sie Ihren MySQL Client starten, durchsuchen Sie bitte das Internet und machen Sie sich vertraut mit Basisaufgaben (Anmeldung, Userverwaltung, Datenbankverwaltung,...)

Bitte nach jeder Zeile  drücken

```
create database redmine character set utf8;  
create user 'redmine'@'localhost' identified by 'XXX';  
grant all privileges on redmine.* to 'redmine'@'localhost';
```

## 2.4. Datenbankverbindung einrichten

Folgendes in `/opt/redmine/config/database.yml` schreiben

---

<sup>6</sup> <http://www.redmine.org/projects/redmine/wiki/Download>

```
production:
  adapter: mysql2
  database: redmine
  host: localhost
  username: redmine
  password: XXX
```

## 2.5. Secret Token erstellen

```
rake generate_secret_token 
```

## 2.6. Datenbank füllen (Tabellen anlegen)

```
RAILS_ENV=production rake db:migrate 
```

## 2.7. Zugriffsrechte auf File-Ebene setzen

Bitte nach jeder Zeile  drücken

```
cd /opt/redmine
mkdir tmp tmp/pdf public/plugin_assets
chown -R www-data:www-data files log tmp public/plugin_assets
chmod -R 755 files log tmp public/plugin_assets
```

## 2.8. Redmine Installation prüfen

```
bundle exec rails server webrick -e production 
```

Öffnen Sie jetzt <http://localhost:3000> und sehen Sie Redmine "in Aktion".

# 3. Apache und Passenger konfigurieren

In dieser Anleitung konfigurieren wir den Einsatz in einer sub-URI. Lesen Sie andere Anleitungen auf [redmine.org](http://redmine.org) wenn Sie lieber einen name-based virtual host konfigurieren möchten.

## 3.1. Apache konfigurieren (subURI deployment)

```
ln -s /opt/redmine/public /var/www/redmine 
```

Fügen Sie dies in `/etc/apache2/sites-available/redmine` ein: Hinweis: Für Rails 3 nennt sich die Option *RackBaseURI*, in Rails 2 wurde sie *RailsBaseURI* genannt. Auf keinen Fall heißt sie *RakeBaseURI*.

```
<Location /redmine>
  RailsEnv production
  RackBaseURI /redmine
  Options -MultiViews
</Location>
```

Die neue Konfiguration wird wie folgt aktiviert:

```
a2ensite redmine 
```

Starten Sie Apache neu, prüfen Sie ob <http://yourhost.com/redmine> funktioniert und freuen Sie sich wenn ja :-). Wenn Sie etwas unerwartetes sehen, posten Sie bitte Details im Forum<sup>7</sup> (Englisch) und

---

<sup>7</sup> <http://www.redmine.org/projects/redmine/boards/2>

nehmen Sie Bezug auf diese Anleitung und die Kapitel-Nummer. Bitte fügen Sie die Ausgabe von `gem env` **und/oder** `RAILS_ENV=production /opt/redmine/script/about` **bei**.