



#### Support

RVM is maintained by community of volunteers, report issues to [RVM issues tracker](#).

If you can help or wish to become one of the maintainers - just start helping. You can find more RVM related projects at [RVM Github organization](#).

#### Donation

Love RVM? Help us maintain RVM, a donation can help with it. Want to help in other way, want to be a sponsor?

contact: [Michal Papis](#)

#### Sponsors



#### Carbon Ads



[Design and Development tips in your inbox. Every weekday. ads via Carbon](#)

## ∞Installing RVM

RVM supports most UNIX like systems and Windows (with [Cygwin](#) or [Bash on Ubuntu on Windows](#)). The basic requirements are bash, curl, gpg2 and overall GNU version of tools - but RVM tries to autodetect it and install anything that is needed.

## ∞Install GPG keys

As a first step install GPG keys used to verify installation package:

```
gpg --keyserver keyserver.ubuntu.com --recv-keys 409B6B1796C275462A1703113804BB82D39DC0E3 7D2BAF1CF37B13E2069D6956105BD0E739499BDB
```

In case you encounter an issues check [security](#).

## ∞Basic install

### ∞Ubuntu

RVM have dedicated Ubuntu package, so please follow instructions posted here: [https://github.com/rvm/ubuntu\\_rvm](https://github.com/rvm/ubuntu_rvm)

If you need a different (newer) version of RVM, after installing base version of RVM check the [Upgrading](#) section.

### ∞Any other system

Install RVM (development version):

```
\curl -sSL https://get.rvm.io | bash
```

Install RVM stable with ruby:

```
\curl -sSL https://get.rvm.io | bash -s stable --ruby
```

Additionally with rails (poor man's [railsinstaller](#)):

```
\curl -sSL https://get.rvm.io | bash -s stable --rails
```

Or with jruby, rails and puma:

```
\curl -sSL https://get.rvm.io | bash -s stable --ruby=jruby --gems=rails,puma
```

To install without rubygems-bundler and rvm gems (and also remove those gems from both global.gems and default.gems):

```
\curl -sSL https://get.rvm.io | bash -s stable --without-gems="rvm rubygems-bundler"
```

To install with hirb gem (and also add it to global.gems):

```
\curl -sSL https://get.rvm.io | bash -s stable --with-gems="hirb"
```

To install with rails and haml gems (and also add them to default.gems):

```
\curl -sSL https://get.rvm.io | bash -s stable --with-default-gems="rails haml"
```

Without [autolibs](#):

```
\curl -sSL https://get.rvm.io | bash -s -- --autolibs=read-fail
```

For a progress bar when downloading RVM / Rubies:

```
echo progress-bar >> ~/.curlrc
```

Point to be noted is, there is a backslash before **curl**. This prevents misbehaving if you have aliased it with configuration in your ~/.curlrc file.

If you're an existing RVM user and you don't want RVM to attempt to setup your shell to load RVM, you can opt out of this at install time by exporting `rvm_ignore_dotfiles=yes`, or opt out permanently by setting this in your `rvmrc`.

### ∞ You can also:

- read the [installation documentation](#) below.
- watch the most accurate (but not official) [rvm screencast](#).
- read the most accurate (but not official) [rvm cheat sheet](#).
- starting with Rails? watch the [RailsCasts.com on Getting Started with Rails](#).

## ∞ Installation explained

There are three different ways to install and configure RVM.

1. **Single-User installations** ([recommended](#)) - For an isolated install within a user's \$HOME, not for root.
2. **Multi-User installations** - For [server administrators](#) - For an installation usable by all users on the system - Please note that Single-User supersedes Multi-User. This also used to be called the System-Wide Install. **Using this type of installation without knowledge how umask works is a big security risk.**
3. **Mixed mode installations** - For an installation usable by all users on the system - with isolated rubies/gemsets within a user's \$HOME. Installation instructions are exactly the same as for Multi-User installations, the difference is in users environment.

[get.rvm.io](#) is a redirect to <https://raw.githubusercontent.com/rvm/rvm/master/binscripts/rvm-installer> You could also use full path for the installer:

```
\curl https://raw.githubusercontent.com/rvm/rvm/master/binscripts/rvm-installer | bash -s stable
```

## ∞ Installation

I recommend you [read the installation script](#) yourself. This will give you a chance to understand what it is doing *before* installing, and allow you to feel more comfortable running it if you do so.

### ∞ 1. Download and run the RVM installation script

Installing the stable release version:

```
\curl -sSL https://get.rvm.io | bash -s stable
```

To get the latest development state:

```
\curl -sSL https://get.rvm.io | bash
```

Instruct RVM to not change the shell initializations files '`rc`' / '`profile`':

```
\curl -sSL https://get.rvm.io | bash -s -- --ignore-dotfiles
```

Please note that from this point it is user responsibility to add sourcing rvm to appropriate files.

For a Multi-User install you would execute the following:

```
\curl -sSL https://get.rvm.io | sudo bash -s stable
```

**Note:** The Multi-User install instructions *must* be prefixed with the [sudo command](#). However, once the install is complete, and the instructions to add users to the rvm group is followed, the use of either sudo or rvmsudo is no longer required. The sudo command is only to temporarily elevate privileges so the installer can complete its work. If you need to use sudo or rvmsudo after the install is complete, some part of the install directions were not properly followed. This usually is because people execute the install *as root*, rather than executing the installation instructions from a non-privileged user account.

Installing a specific version:

```
\curl -sSL https://get.rvm.io | bash -s -- --version latest
\curl -sSL https://get.rvm.io | bash -s -- --branch [owner/][repo]
```

Prefix the 'bash' portion with 'sudo', of course, if you wish to apply this to a Multi-user Install. Please feel free to check out our [upgrading docs](#) for more details on branch format.

Debugging installation process:

```
\curl -sSL https://get.rvm.io | bash -s -- --trace
```

If the rvm install script complains about certificates you need to follow the displayed instructions.

**Single-User Install Location:** ~/.rvm/

If the install script is run as a standard, non-root user, RVM will install into the current users's home directory.

Modification of user configuration files (\*rc / \*profile) - RVM by default will modify user startup files, although it is not recommended you can disable automated process and do this manually:

```
\curl -sSL https://get.rvm.io | bash -s -- --ignore-dotfiles
echo "source $HOME/.rvm/scripts/rvm" >> ~/.bash_profile
```

**Multi-User Install Location:** /usr/local/rvm

If the install script is run prefixed with sudo, RVM will automatically install into /usr/local/rvm. Please see the troubleshooting page for an [important note regarding Multi-User Installs](#).

Please see the FAQ page for an [important note regarding root only installs](#).

## External tutorials

Note that that any outside tutorials are **NOT** supported whether they work or not. Tutorials are great, however we have spent massive amounts of man hours debugging the installation process. Please use the install process(es) from this site only, as this is the only supported installation types and methods.

## To update an existing RVM installation

It is safe to simply re-run the installation script again, or you can follow the [upgrading docs](#).

## ∞2. Load RVM into your shell sessions as a function

### Single-User:

The rvm function will be automatically configured for every user on the system if you install as single user. Read the output of installer to check which files were modified.

### Multi-User:

The rvm function will be automatically configured for every user on the system if you install with sudo. This is accomplished by loading /etc/profile.d/rvm.sh on login. Most Linux distributions default to parsing /etc/profile which contains the logic to load all files residing in the /etc/profile.d/ directory. Once you have added the users you want to be able to use RVM to the rvm group, those users **MUST** log out and back in to gain rvm group membership because group memberships are only evaluated by the operating system at initial login time. **Zsh not always sources /etc/profile so you might need to add this in /etc/\*\*/zprofile:**

```
source /etc/profile
```

### Mixed mode (user gemsets):

- After following above instructions for Multi-User.
- Select a user as a manager - he will be responsible for installing new rubies. This user should never run the command introduced below. If this happens, remove/rename the \${HOME}/.rvmrc, logout and then relogin. Otherwise you won't be able to install/upgrade new rubies correctly.

- For each user that want to use RVM, an additional command needs to be run (once) for each user:

```
rvm user gemsets
```

Gemsets created by these users will be hosted in their HOME directory. It's not possible to use global gemsets from system without using tricks like manually linking directories and they should not be used in mixed-mode. Please bear in mind that 'system' in this context does not refer to your distribution's ruby packages, but to the RVM Multi-User installation.

You have two possibilities to manage RVM. The first one is to add managers to the rvm group. The second one is to use separate managers with rvm sudo and privilege escalation. Note that it is not safe to use rvm sudo from mixed mode user. Both can be mixed without any side-effect. It is however very important to not enable mixed-mode gemsets or rubies for the managers. RVM is using a custom umask (umask u=rwx,g=rwx,o=rx) when installing gemsets, rubies, updating itself, etc. This should not impact your system. But if you prefer to avoid RVM messing around with your umask, you can comment the umask line in /etc/rvmrc.

This mode should also works with passenger, please follow [passenger](#) instructions. .

### 3. Reload shell configuration & test

Close out your current shell or terminal session and open a new one (preferred). You may load RVM with the following command:

```
source ~/.rvm/scripts/rvm
```

If installation and configuration were successful, RVM should now load whenever you open a new shell. This can be tested by executing the following command which should output rvm is a function as shown below.

```
type rvm | head -n 1
rvm is a function
```

**NOTE:** Before reporting problems check rvm notes as it might contain important information.

**Congratulations! You have successfully installed RVM.**

### Try out your new RVM installation

Below are some examples of how to install and use a Ruby under RVM.

Display a list of all known rubies. NOTE: RVM can install many more Rubies not listed.

```
rvm list known
# MRI Rubies
[ruby-]1.8.6[-p420]
[ruby-]1.8.7[-p374]
[ruby-]1.9.1[-p431]
[ruby-]1.9.2[-p320]
[ruby-]1.9.3[-p545]
[ruby-]2.0.0-p353
[ruby-]2.0.0[-p451]
[ruby-]2.1[.1]
[ruby-]2.1-head
ruby-head
...
```

Install a version of Ruby (eg 2.1.1):

```
rvm install 2.1
Checking requirements for openssl.
Requirements installation successful.
Installing Ruby from source to: /home/mpapis/.rvm/rubies/ruby-2.1.1, this may take a while depending on your cpu(s)...
...
Install of ruby-2.1.1 - #complete
Using /home/mpapis/.rvm/gems/ruby-2.1.1
```

Use the newly installed Ruby:

```
rvm use 2.1
Using /home/mpapis/.rvm/gems/ruby-2.1.1
```

Check this worked correctly:

```
ruby -v
ruby 2.1.1p76 (2014-02-24 revision 45161) [x86_64-linux]
```

```
which ruby
/home/mpapis/.rvm/rubies/ruby-2.1.1/bin/ruby
```

Optionally, you can set a version of Ruby to use as the default for new shells. Note that this overrides the 'system' ruby:

```
rvm use 2.1 --default
```

## ∞Enjoy using RVM!

### ∞Where to now?

If you are new to RVM I recommend that you read the [basics page](#). At the end of the basics page there are further links for getting started.

## ∞Troubleshooting Your Install

- When you run

```
curl -sSL https://get.rvm.io | bash -s stable
```

and got the notice

```
curl: (60) SSL certificate problem: unable to get local issuer certificate
More details here: http://curl.haxx.se/docs/sslcerts.html
```

ca-certificates need to be installed:

```
apt-get install ca-certificates
```

- If you open a new shell and running:

```
type rvm | head -1
```

does **not** show rvm is a function, RVM isn't being sourced correctly.

- Ensure that RVM is sourced after any path settings as RVM manipulates the path. If you don't do this, RVM may not work as expected.
- If you are using GNOME on Red Hat, CentOS or Fedora, ensure that the **Run command as login shell** option is checked under the **Title and Command** tab in **Profile Preferences**. After changing this setting, you may need to exit your console session and start a new one before the changes take affect.

[RVM Documentation Index](#) | [RVM Blog](#)

Search:

### [RVM](#)

- [About](#)
- [Autolibs](#)
- [Basics](#)
- [Best Practices](#)
- [Cli](#)
- [Configuration](#)
- [Info](#)
- [Install](#)
- [Myths](#)
- [Offline](#)
- [Prerequisites](#)
- [Security](#)
- [Upgrading](#)

### [Interpreters](#)

- [MRI/YARV \(ruby\)](#)
- [JRuby \(jruby\)](#)
- [Rubinius \(rbx\)](#)

### [Rubies](#)

- [Alias](#)
- [Default](#)

- [Docs](#)
- [Info](#)
- [Installing](#)
- [List](#)
- [Named](#)
- [Patching](#)
- [Removing](#)
- [Rubygems](#)
- [Strings](#)
- [Uninstalling](#)
- [Upgrading](#)

#### [Gemsets](#)

- [Basics](#)
- [Clear](#)
- [Copying](#)
- [Creating](#)
- [Deleting](#)
- [Emptying](#)
- [Exporting](#)
- [Global](#)
- [Ignoring](#)
- [Importing](#)
- [Initial](#)
- [Listing](#)
- [Using](#)

#### [Set Actions](#)

- [do](#)
- [tests](#)
- [benchmark](#)

#### [Workflow](#)

- [Chruby](#)
- [Completion](#)
- [Examples](#)
- [Hooks](#)
- [Irbrc](#)
- [Projects](#)
- [Prompt](#)
- [Proxy](#)
- [Rvmrc](#)
- [Screen](#)
- [Scripting](#)
- [Subshells](#)

#### [Deployment](#)

- [Best Practices](#)
- [Capistrano](#)
- [Chef](#)
- [Cron](#)
- [God](#)
- [Init D](#)
- [Passenger](#)

#### [Integration](#)

- [Bundler](#)
- [Capistrano](#)
- [Chef](#)
- [Cron](#)
- [Culerity](#)
- [Databases](#)
- [Emacs](#)
- [Engineyard](#)
- [Fish](#)

- [Gnome Terminal](#)
- [God](#)
- [Heroku](#)
- [Init D](#)
- [Jenkins](#)
- [Konsole](#)
- [Lxterminal](#)
- [Macports Before Autolibs](#)
- [Macports](#)
- [Memprof](#)
- [Passenger](#)
- [Pow](#)
- [Qtruby](#)
- [Scripting](#)
- [Simple Terminal](#)
- [Sketchup](#)
- [Sudo](#)
- [Teamecity](#)
- [Textmate](#)
- [Tk](#)
- [Ubuntu On Windows](#)
- [Vagrant](#)
- [Vim](#)
- [Xterm](#)
- [Zsh](#)

#### [Packages](#) [Presentations](#)

- [MWRC 2010](#)
- [RailsConf 2012](#)
- [RailsConf 2013](#)
- [Changelog 66 with Wayne](#)
- [RVM 2: Lessons and future](#)
- [RVM 2: Python Version Manager](#)
- [Changelog 102 with Michal](#)

#### [Help & Support](#)

- [Issues tracker](#)
- [FAQ & Notes](#)
- [Troubleshooting](#)
- [Fixing Broken SSL Certificates](#)

#### [Quality Assurance](#)

- [Software testing](#)

#### [Development](#)

- [Contributing](#)
- [Debugging](#)
- [Github](#)
- [List](#)

#### [Credits](#)

- [Contributions](#)
- [Donations](#)
- [Funding](#)

∞ Copyright © 2009-2011 [Wayne E. Seguin](#) © 2011-2017 [Michal Papis](#) © 2016-2017 [Piotr Kuczynski](#) ∞ Built with [Nanoc](#) ∞ Cached by [Fastly](#) ∞ Tested by [Travis CI](#)