

Ruby on Rails

For cPanel & WHM version 68

(cPanel >> Home >> Software >> Ruby on Rails)

Overview

Warning:

We **deprecated** this interface in cPanel & WHM version 66. To create a Ruby application in cPanel & WHM version 66 and later, perform the following steps:

1. Ask your hosting provider to install Ruby on your system via the following [EasyApache 4 RPMs](#):
 - The `ea-apache24-mod_env` module.
 - The `ea-ruby24-ruby-devel` module.
2. Create a Ruby application on your server. For an example of how to do this, read our [How to Create Ruby Web Applications](#) documentation.
3. Use cPanel's [Application Manager](#) interface (cPanel >> Home >> Software >> Application Manager) to register the application.

This interface allows you to create and deploy a Ruby on Rails® application.

Notes:

- To use this feature, your system administrators **must** enable it for you.
- Before you deploy a Ruby on Rails application, read the [Ruby on Rails](#) documentation.

Create a Ruby on Rails application

After you confirm that your server uses a functional and accessible version of Ruby, you can create and deploy a Ruby on Rails environment.

Because Ruby on Rails runs on its own server, it requires the following steps to set up the environment:

1. Enter your application's name in the *App Name* text box.
2. To load when the server boots, select the *Load on Boot?* checkbox.
3. Enter the application path in the *Application Path* text box.

Note:

If you set an application path in your `public_html/` directory, anyone on the web can view your source code and configuration files.

4. In the *Environment* menu, select the type of Ruby on Rails environment that you prefer.
5. Click *Create*.

You can perform the following actions on applications in the *Ruby on Rails Applications* table:

- To redirect to an application's Rails server, click *URL*.
- To run an application, click *Run*.
- To stop an application, click *Stop*.
- To load an application when the server boots, select the *Load on Boot?* checkbox.
- To toggle an application between a production or development environment, select or deselect the *Production* checkbox.
- To delete an application, click *Delete*.

Create a Rewrite

Because Ruby on Rails uses its own server and does not respond on the standard HTTP port number 80, you **must** specify the port where site visitors can access your Ruby on Rails application (for example, `example.com:12001`).

In This Document

Related Documentation

- [Ruby on Rails](#)
- [RubyGems](#)
- [Application Manager](#)

For Hosting Providers

- [Module Installers](#)
- [Guide to Ruby Installations - Uninstall Ruby - rpm](#)
- [Guide to Ruby Installations - Uninstall Ruby - Manual Compilation](#)
- [Guide to Ruby Installations - Requirements](#)
- [Guide to Ruby Installations - Install Ruby](#)

Warnings:

- Ruby application port numbers increment automatically with every new application. If you are on a shared server, the system assigns the next available port number to your application.
- Your first application will not necessarily run on port 12001.

To configure Ruby to respond without the need for visitors to specify a port number, create a rewrite.

To configure a rewrite, perform the following steps:

1. In the *Create A Rewrite* table, click *Create Rewrite* for the appropriate application
2. From the menu, select the domain, subdomain, or addon domain on which you wish to run your Ruby application.
 - To run the application from a new page within your domain, enter the page name that you wish to use in the available text box.
3. Click *Save*.

You can perform the following actions in the *Current Rewrites* table:

- To change an application path name, enter the path name in the text box.
- To save an application rewrite, click *Save*.
- To delete an application rewrite, click *Delete*.

Additional applications and rewrites

To run multiple Ruby applications, you **must** configure additional ports to respond with Ruby on Rails and, if necessary, additional rewrites to disguise the port change.

- To add additional Rails applications, follow the steps in the [Create a Ruby on Rails application](#) section.
- The new application will respond on the next available port number (for example, if your first application runs on port 12001, the second may run on port 12002 or higher).

Warning:

Ruby application port numbers increment automatically with every new application. If you are on a shared server, the system assigns the next available port number to your application. Also, your applications' port numbers may **not** be consecutive.

- To configure an additional rewrite, follow the steps in the [Create a Rewrite](#) section.
- You will need to use a new domain, addon domain, subdomain, or page (for example, `example.com/RubyApplication`) for the rewrite.

Note:

You **cannot** configure two Ruby applications to respond on the same domain, subdomain, addon domain, or page.

Note for resellers and web hosts

You can limit the number of Rails applications that your customers can install in WHM's [Modify an Account](#) interface (*WHM >> Home >> Account Functions >> Modify An Account*). Perform the following steps to set the maximum number of Rails applications:

1. Click the domain name that you wish to limit.
2. In the *Max Mongrel instances (Ruby on Rails)* text box, enter the maximum number of Rails applications that you wish to allow your customers to install.
3. Click *Save*.