

About

With serverspec, you can write RSpec tests for checking your servers are configured correctly.

Serverspec tests your servers' **actual state** through **SSH access**, so you don't need to install any agent softwares on your servers and can use any configuration management tools, [Puppet](#), [Chef](#), [CFEngine](#) and so on.

Installation

Add this line to your application's Gemfile:

```
gem 'serverspec'
```

And then execute:

```
$ bundle
```

Or install it yourself as:

```
$ gem install serverspec
```

Getting Started

```
$ serverspec-init  
Select OS type:
```

- 1) UN*X
- 2) Windows

```
Select number: 1
```

```
Select a backend type:
```

- 1) SSH
- 2) Exec (local)

```
Select number: 1
```

```
Vagrant instance y/n: n  
Input target host name: www.example.jp  
+ spec/  
+ spec/www.example.jp/  
+ spec/www.example.jp/httpd_spec.rb  
+ spec/spec_helper.rb  
+ Rakefile
```

spec/www.example.jp/httpd_spec.rb is a sample spec file and its content is like this.

```
require 'spec_helper'  
  
describe package('httpd') do  
  it { should be_installed }  
end
```

```
describe service('httpd') do
  it { should be_enabled }
  it { should be_running }
end

describe port(80) do
  it { should be_listening }
end

describe file('/etc/httpd/conf/httpd.conf') do
  it { should be_file }
  its(:content) { should match /ServerName www.example.jp/ }
end
```

You can write spec for testing servers like this.

Serverspec with SSH backend logs in to target servers as a user configured in `~/.ssh/config` or a current user. If you'd like to change the user, please edit the below line in `spec/spec_helper.rb`.

```
user = options[:user] || Etc.getlogin
```

Run tests.

```
$ rake spec
/usr/bin/ruby -S rspec spec/www.example.jp/httpd_spec.rb
.....
```

```
Finished in 0.99715 seconds
6 examples, 0 failures
```

1