




Moonshine

Tarsd kézben a teljes alkalmazást

Cél egy alkalmazás és a futtatásához szükséges teljes szoftver rendszer felügyelete és konfiguráció kezelése.

	Linode 360 
RAM	360MB
Storage	16GB
Transfer	200GB
Price	\$19.95
	109 Available
	Sign Up Now!

Miért?

Virtualizáció

1 alkalmazás - 1
kiszolgáló

Rendzergazdai
tudás követése és
verziókezelése

Telepítés
megismételhetősége
Lásd: Cloud
computing

“Egy új Rails / Merb / Sinatra
alkalmazás telepítése egyszerű –
csak használj Capistranot!”

```
ssh rails@ubuntu-my.com
```

```
sudo apt-get install mysql-server libmysql-ruby libmysqlclient15-dev
```

“Egy új Rails / Merb / Sinatra alkalmazás telepítése egyszerű – csak használj Capistranot!”

```
sudo apt-get install libmagick10 libmagick9-dev  
sudo gem install rmagick
```

Mi is egy telepítés?

Egy teljes egymásra épülő szoftver rendszer

Rengeteg egymásra épülő függőséggel

Rails v2.2.2, v1.2.3

Ruby/Ruby Enterprise

Apache/Nginx

Passenger/Thin/Mongrel

MySQL/PostgreSQL



Moonshine

Capistrano + ShadowPuppet + Puppet



Puppet

```
class ssh {
  package { ssh: ensure => installed }
  file { sshd_config:
    name => $operatingsystem ? {
      Darwin => "/etc/sshd_config",
      Solaris => "/opt/csw/etc/ssh/sshd_config",
      default => "/etc/ssh/sshd_config"
    },
    source => "puppet://server.domain.com/files/ssh/sshd_config"
  }
  service { ssh:
    name => $operatingsystem ? {
      Solaris => openssh,
      default => ssh
    },
    ensure => running,
    subscribe => [Package[ssh], File[sshd_config]]
  }
}
```

Ssh szolgáltatás
leírása

ShadowPuppet

- ▶ Ruby DSL a Puppet konfigurációs fájlaihoz

```
def ssh(options = {})
  package 'ssh', :ensure => :installed
  service 'ssh', :enable => true, :ensure => :running

  file '/etc/ssh/sshd_config',
    :mode => '644',
    :content => template(File.join(File.dirname(__FILE__), '..', 'templates', 'sshd_config')),
    :require => package('ssh'),
    :notify => service('ssh')
end
```

Moonshine

- ▶ Capistrano kiterjesztés
- ▶ Minden deploy során lefut a teljes alkalmazás stack ellenőrzése

- ▶ Mi kell a használatához?
 - Ubuntu 8.10 szerver
 - Egy felhasználó aki tud sudo-val parancsokat futtatni
 - Kód repó, amit el lehet érni erről a szerverről

Használat

1) `ruby script/plugin install`

`git://github.com/railsmachine/moonshine.git`

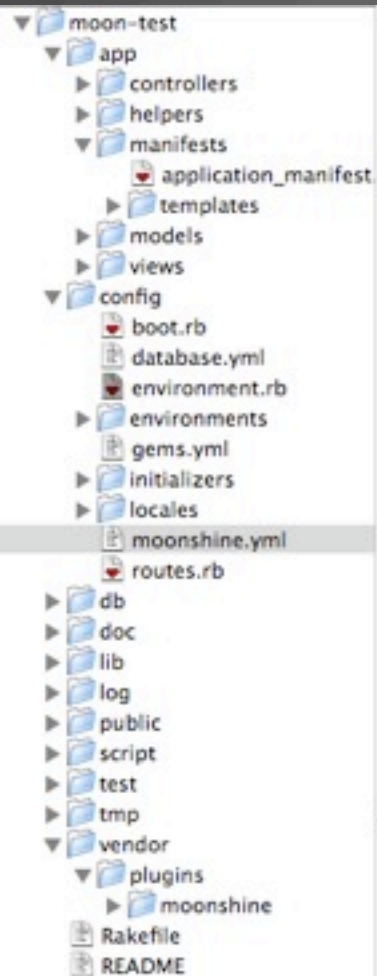
2) Ellenőrizd, hogy minden gem definiálva van-e az `environment.rb`-ben!

```
Rails::Initializer.run do |config|  
  config.gem "bj"  
  config.gem "sqlite3-ruby", :lib => "sqlite3"
```

• `ruby script/generate moonshine`

`config/moonshine.yml`

`app/manifests/application_manifest.rb`



environment.rb x moonshine.yml

```
1  :ruby: ree
2  :application: moon-test
3  :user: rails
4  #:time_zone: UTC
5  :deploy_to: /srv/moon-test
6  :domain: moon-test.com
7  #:domain_aliases:
8  # - assets1.yourapp.com
9  # - assets2.yourapp.com
10 :repository: git@github.com:username/your_app_name.git
11 # Use :app_symlinks to setup directories under public/ that you want to persist
12 # across deployments. They'll be symlinked to the shared directory.
13 #:app_symlinks:
14 # - uploads
15 #
16 # Use :local_config to copy directories straight from your local project directory
17 # to the server; these are files that you don't have/want in scm.
18 #:local_config:
19 # - config/database.yml
20 #
21 # The entries in shared_children are created in #{application}/shared
22 :shared_children:
23 - system
24 - log
25 - pids
26 - config
27 # After specifying a gem with config.gem, run 'rake moonshine:gems'
28 # to update config/gems.yml. If the gem depends on native packages,
29 # tell Moonshine here:
30 #:opt_gems:
31 # :awesomegem:
32 #   - awesome
33 #   - libawesome-dev
34 #
35 # See all the options for Passenger, Apache vhosts, and MySQL in the
36 # respective templates: moonshine/lib/moonshine/manifest/rails/templates
37 :passenger:
38   :max_pool_size: 3
39   :use_global_queue: true
40 :mysql:
41   :innodb_buffer_pool_size: 128M
42
```

Capistrano config

- ▶ capify
- ▶ config/deploy.rb lehet nagyon egyszerű, minden változó ami a moonshine.yml-ben deklarált, elérhető lesz a Capistrano számára is

```
server "moon-test.com", :app, :web, :db, :primary => true
```

Deploy

```
git add . &&
```

```
git commit -am "added moonshine" &&
```

```
git push
```

```
cap deploy:setup – szerverenként 1X elég
```

```
cap deploy – minden app deploy során
```

Hasznos tippek

Moonshine frissítése

- ▶ `rake moonshine:update`

Adatbázisban lévő adatok mentése → `db/dump`

- ▶ `rake moonshine:db:dump`

Adatbázisba visszatöltés

`rake moonshine:db:bootstrap`

Ez első telepítéskor fut le

`rake moonshine:app:bootstrap`

Ha új gem-et adunk hozzá

`rake moonshine:gem`

Linkek

- ▶ Puppet típusok leírása – ShadowPuppet ezeket valósítja meg
- ▶ <http://reductivelabs.com/trac/puppet/wiki/TypeReference>
- ▶ Moonshine
- ▶ <http://github.com/railsmachine/moonshine/>
- ▶ Moonshine wiki – rengeteg hasznos anyag
- ▶ <http://wiki.github.com/railsmachine/moonshine>
- ▶ Railsmachine blog – Moonshine, Moonshine plugin leírások