

Créer un référentiel RubyGem

Serveurs Gem locaux

Utilisation du serveur par défaut de RubyGems

RubyGems est fourni avec une commande - judicieusement nommée `gem server` - utilisée pour servir les gemmes ruby installées.

```
gem server
Server started at http://0.0.0.0:8808
...
```

La commande `gem server` démarre un serveur sur un port (le port par défaut est **8808** - peut être spécifié avec **-p** ou **-port**) et servira, par défaut, tous les paquets installés sur le système local. On peut taper <http://localhost:8808> dans un navigateur pour voir un index de tous les paquets installés sur le système.

Pour utiliser le serveur gem local comme source d'installation des gemmes, il faut s'assurer que les fichiers cache des gemmes existent sur le système exécutant le serveur. On peut également spécifier un ou plusieurs répertoires pour rechercher les gemmes installées en passant les options `-d` ou `-dir` à la commande `gem server`.

```
gem server -d /path/to/your/gems
```

Pour rendre les nouveaux packages disponibles, ajouter simplement les gemmes souhaitées dans le répertoire de gemmes spécifié.

Serveur gemstash

Pour commencer, installer **gemstash**:

```
gem install gemstash
```

Une fois installé, démarrer le serveur **Gemstash** avec la commande suivante :

```
gemstash start
```

Par défaut, le serveur fonctionne sur le port **9292**.



Pour l'utiliser comme cache, on peut indiquer à **Bundler** d'utiliser **Gemstash** pour



rechercher des gemmes sur RubyGems.org :

```
bundle config mirror.https://rubygems.org http://localhost:9292
```

\\Avec cette configuration, toutes les gemmes récupérées sur RubyGems.org via bundler sont mises en cache par Gemstash.

\\On peut également envoyer ses propres gemmes et utiliser le serveur **Gemstash** comme source de gemmes privée.

Exécution de Gem in a Box

Pour un serveur avec plus de fonctionnalités, y compris la possibilité de pousser des gemmes, on peut utiliser le projet **Gem in a Box**.

Pour commencer, installer **geminabox**:

```
gem install geminabox
```

Créer un répertoire de données pour stocker les gemmes :

```
mkdir data
```

Inclure les éléments suivants dans un fichier config.ru :

```
require "rubygems"
require "geminabox"

Geminabox.data = "./data"
run Geminabox::Server
```

Et exécuter le serveur :

```
rackup
[2011-05-19 12:09:40] INFO WEBrick 1.3.1
[2011-05-19 12:09:40] INFO ruby 1.9.2 (2011-02-18) [x86_64-darwin10.5.0]
[2011-05-19 12:09:40] INFO WEBrick::HTTPServer#start: pid=60941 port=9292
```

On peut maintenant envoyer des gemmes à l'aide de la commande `gem inabox`. La première fois que l'on effectue cette opération, il faut indiquer l'emplacement du serveur gem.

```
gem build secretgem.gemspec
RubyGem a été créé avec succès
Nom : secretgem
Version : 0.0.1
Fichier : secretgem-0.0.1.gem
```

```
gem inabox ./secretgem-0.0.1.gem
Entrez l'URL racine de votre instance geminabox personnelle. (Par exemple,
http://gems/)
Hôte : http://localhost:9292
Transfert de secretgem-0.0.1.gem vers http://localhost:9292/...
```

Une interface Web est également disponible sur <http://localhost:9292>.

Utilisation des gemmes du serveur local

Que l'on utilise **Gemstash**, **Gem in a Box** ou un autre serveur de gemmes, on peut configurer RubyGems pour utiliser la source locale ou interne avec d'autres sources telles que <http://rubygems.org>.

Utiliser la commande `gem sources` pour ajouter le serveur de gemmes aux sources de gemmes à l'échelle du système. L'URL suivante est l'URL par défaut pour exécuter **Gem in a Box** via **rackup**:

```
gem sources --add http://localhost:9292
```

Installer ensuite les gemmes comme d'habitude :

```
gem install secretgem
Secretgem-0.0.1 installé avec succès
1 gem installé
```

Avec Bundler, on peut spécifier ce serveur comme source de gemmes dans le fichier Gemfile :

```
source "http://localhost:9292"
gem "secretgem"
```

```
bundle
Utilisation de secretgem (0.0.1)
Utilisation de bundler (1.0.13)
Votre bundle est terminé ! Utiliser `bundle show [gemname]` pour voir où un
gem groupé est installé.
```

From:

<http://www.ouarte.garden/> - dwndoc

Permanent link:

<http://www.ouarte.garden/doku.php?id=vagrant:ruby-gem-local>

Last update: **2025/02/19 10:59**

