

# RKT: Présentation et construction

## Table of Contents

- RKT: Présentation et construction
- Présentation
  - Description du stage1
- Paramètres de construction
  - Paramètres de construction des flavors de stage1
  - Paramètres de configuration de l'image Stage1 par défaut
    - Configuration des flavors
    - Configuration détaillée
  - Emplacement de l'image stage1 par défaut
  - Paramètres spécifiques à la flavor
    - Flavor src
    - Flavors coreos et kvm
  - Essai
  - Sécurité
  - Développement
- Dépendances d'exécution
- Exemple de configuration et de construction de rkt
  - Récupération du dépôt rkt
  - Installation de Go Programming Language
  - Construction de rkt
- Quelques commandes utiles
  - Construction et exécution des tests:
  - Exécution des tests fonctionnels après la génération:
  - Exécution d'un test spécifique:
  - Utilisation simple du conteneur rkt (exécuter, quitter, supprimer):
  - Suppression de toutes les interfaces réseau créées par rkt:

RKT, à prononcer “rock-it”, est un container runtime créé par les équipes de CoreOS, fonctionnant en couches avec le support de plusieurs moteurs d'exécution pour les containers.

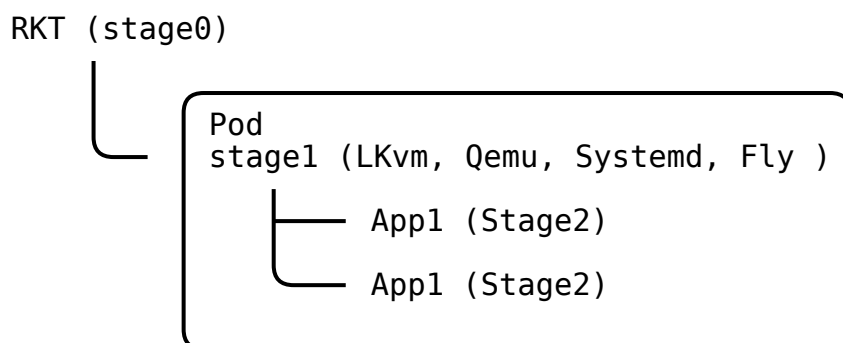
## Présentation

Les différentes couches sont les suivantes:

- **en stage0**, RKT lui même, le binaire n'étant qu'une UX/API pour le moteur d'exécution du container
- **en stage1** se trouve le moteur d'exécution du container. Comme évoqué précédemment, plusieurs sont disponibles. Par défaut, un container est lancé via systemd-nspawn. Le container

peut aussi démarrer via un simple chroot, nommé fly par les développeurs de RKT. Enfin, l'isolation matérielle peut aussi être réalisée sur cette couche par l'intermédiaire de LKVM (une version allégée de KVM) ou en core Qemu

- **le stage2** comprend la ou les applications qui doivent fonctionner dans le container lancé. La notion de Pod par RKT regroupe à la fois le moteur d'exécution et les applications containerisées, ce qui est au final démarré par le binaire rkt. Illustrée, cette composabilité interne peut être vue de la façon suivante:



## Description du stage1

Le fly stage1 est un stage1 alternatif qui exécute un ACI à application unique avec seulement une isolation par chroot.

La motivation de la fonctionnalité Fly est d'ajouter la possibilité d'exécuter des applications avec des privilèges complets sur l'hôte tout en bénéficiant de la gestion des images et de la découverte de rkt.

Par rapport au stage1 par défaut, aucun gestionnaire de processus n'est impliqué dans ce stage1. Il s'agit d'une illustration visuelle des différences dans l'arborescence de processus entre le stade par défaut et le stade fly1:

stage1-coreos.aci:

```

OS hôte
├─ rkt
│   └─ systemd-nspawn
│       └─ systemd
│           └─ chroot
│               └─ user-app1
  
```

stage1-fly.aci:

```

OS hôte
├─ rkt
│   └─ chroot
│       └─ user-app1
  
```

L'application rkt configure des montages de liaison pour /dev, /proc, /sys et les volumes fournis par l'utilisateur. En plus des montages de liaison, un montage tmpfs supplémentaire est effectué dans

/tmp. Une fois les montages configurés, rkt chroote vers RootFS de l'application et exécute enfin l'application.

## Paramètres de construction

### Paramètres de construction des flavors de stage1

|                                    |                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>--with-stage1-flavors</code> | Ce paramètre prend une liste séparée par des virgules de toutes les versions que le système de génération doit assembler. En fonction d'une configuration d'image stage1 par défaut, cette liste est par défaut vide ou définie sur <b>coreos</b> , <b>kvm</b> , <b>fly</b> pour, respectivement, une configuration détaillée et une configuration de saveur. Notez que la spécification de ce paramètre ne signifie pas nécessairement que rkt les utilisera à la fin. |
|------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

Les saveurs disponibles sont:

- **coreos** - il prend **systemd** et **bash** à partir d'une image CoreOS Container Linux PXE; utilise **systemd-nspawn**
- **kvm** - il prend **systemd**, **bash** et autres binaires d'une image Container Linux PXE; utilise **lkvm** ou **qemu**
- **src** - il construit **systemd**, prend **bash** de l'hôte au moment de la construction; utilise le **systemd-nspawn** intégré
- **host** - il prend **systemd** et **bash** de l'hôte au moment de l'exécution; utilise **systemd-nspawn** de l'hôte
- **fly** - approche chroot uniquement pour les conteneurs à isolation minimale à application unique; implémentation native de Go

La flavor host est probablement la saveur la mieux adaptée aux distributions qui ont des règles strictes sur les sources logicielles.

|                                                     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|-----------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>--with-stage1-flavors-version-override</code> | Ce paramètre prend un numéro de version pour devenir la version de toutes les versions intégrées de stage1. Normalement, sans ce paramètre, les images ont la même version que rkt lui-même. Ce paramètre peut être utile pour les distributions qui fournissent souvent des versions corrigées de logiciels en amont sans changer le numéro de version majeur / mineur / correctif, mais plutôt ajouter un suffixe numérique. Un exemple d'utilisation pourrait passer <code>--with-stage1-flavors-version-override = 0.12.0-2</code> , donc la nouvelle image aura une version 0.12.0-2 au lieu de 0.12.0. Ce paramètre affecte également la version par défaut de l'image stage1 dans la configuration de saveur. |
|-----------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

### Paramètres de configuration de l'image Stage1 par défaut

Les paramètres décrits ci-dessous affectent le traitement de l'image stage1 par défaut de rkt. rkt

essaie d'abord de trouver l'image stage1 dans le magasin en utilisant le nom et la version par défaut de l'image stage1. Si cela échoue, rkt essaiera de récupérer l'image dans le magasin à partir de l'emplacement par défaut de l'image stage1.

Il existe deux façons mutuellement exclusives de spécifier un nom et une version d'image par défaut de stage1:

- configuration de flavor
- configuration détaillée

## Configuration des flavors

La configuration des flavors n'a qu'un seul paramètre. Ce type de configuration est plutôt un outil pratique pour la configuration détaillée.

|                                            |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|--------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>--with-stage1-default-flavour</code> | Il prend un nom de la flavor de l'image stage1 construite et, sur cette base, il configure le nom et la version par défaut de l'image stage1. Dans ce cas, l'image stage1 par défaut est souvent quelque chose comme <code>coreos.com/rkt/stage1-</code> . La version par défaut de stage1 est généralement juste la version rkt, sauf si elle est remplacée par <code>--with-stage1-flavors-version-override</code> . Il s'agit de la configuration par défaut si aucune saveur ni configuration détaillée n'est utilisée. La saveur d'image stage1 par défaut est la première saveur de la liste dans <code>--with-stage1-flavors</code> . |
|--------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

## Configuration détaillée

La configuration détaillée a deux paramètres, les deux doivent être fournis. Ce type de configuration peut être utilisé pour faire de l'implémentation stage1 tierce l'image stage1 par défaut utilisée par rkt.

|                                            |                                                                          |
|--------------------------------------------|--------------------------------------------------------------------------|
| <code>--with-stage1-default-name</code>    | Ce paramètre indique quel est le nom de l'image stage1 par défaut.       |
| <code>--with-stage1-default-version</code> | Ce paramètre indique quelle est la version de l'image stage1 par défaut. |

## Emplacement de l'image stage1 par défaut

|                                                     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|-----------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>--with-stage1-default-location</code>         | Ce paramètre indique à rkt où trouver l'image stage1 par défaut si elle n'est pas trouvée dans le magasin. Pour la configuration détaillée, la valeur par défaut de ce paramètre est vide, donc s'il n'est pas fourni, vous pouvez être obligé d'informer rkt de l'emplacement de l'image stage1 lors de l'exécution. Pour la configuration de saveur, la valeur par défaut est également vide, ce qui indique à rkt de rechercher l'image dans le répertoire où se trouve le binaire rkt, à moins qu'il ne soit remplacé au moment de l'exécution. Normalement, ce paramètre doit être une URL, avec un schéma ou un chemin absolu. |
| <code>--with-stage1-default-images-directory</code> | Ce paramètre indique à rkt où se trouve le répertoire contenant toutes les images stage1. La valeur doit être un chemin absolu. Dans ce répertoire, toutes les versions construites des images stage1 doivent être installées. Le drapeau <code>--stage1-from-dir</code> rkt recherchera des images dans ce répertoire. La valeur par défaut de ce paramètre est <code>&lt;libdir&gt; / rkt / stage1-images</code> , où <code>&lt;libdir&gt;</code> est un emplacement spécifique à la distribution pour le stockage des fichiers dépendants de l'archive.                                                                           |

## Paramètres spécifiques à la flavor

### Flavor src

La saveur src fournit des paramètres pour spécifier certains détails spécifiques à git du référentiel systemd.

|                                             |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|---------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>--with-stage1-systemd-src</code>      | Ce paramètre prend une URL vers un référentiel git systemd. La valeur par défaut est <a href="https://github.com/systemd/systemd.git">https://github.com/systemd/systemd.git</a> . Vous voudrez peut-être le modifier pour pointer le système de génération pour utiliser un référentiel local.                                                                                                                                                                                 |
| <code>--with-stage1-systemd-version</code>  | Ce paramètre spécifie le système version d à construire. Les noms de version sont généralement sous la forme de <code>v&lt;numéro&gt;</code> , où <code>numéro</code> est une version systemd. La valeur par défaut est <code>v999</code> .                                                                                                                                                                                                                                     |
| <code>--with-stage1-systemd-revision</code> | Ce paramètre prend soit un nom de balise soit un nom de branche. Vous pouvez utiliser le maître de nom de branche pour tester la version de pointe de systemd ou toute branche de travail ou nom de balise. Étant donné que les noms de branche arbitraires n'impliquent pas la version de systemd en cours de construction, la version réelle de systemd est spécifiée à l'aide de <code>--with-stage1-systemd-version</code> . La valeur par défaut est <code>master</code> . |

## Flavors coreos et kvm

Les flavors coreos et kvm fournissent des paramètres liés à l'image CoreX Container Linux PXE.

|                                                            |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>--with-coreos-local-pxe-image-path</code>            | Ce paramètre est utilisé pour pointer le système de génération vers une image PXE Container Linux locale. Cela peut être utile pour certains emballeurs, où tout téléchargement sur le réseau est un non-non. Le paramètre prend des chemins relatifs ou absolus. La valeur par défaut est vide, donc l'image sera téléchargée sur le réseau. Si ce paramètre est spécifié, alors <code>--with-coreos-local-pxe-image-systemd-version</code> doit également être spécifié.                                                                             |
| <code>--with-coreos-local-pxe-image-systemd-version</code> | Le système de génération n'a aucun moyen fiable de déduire automatiquement la version de systemd que contient l'image Container Linux PXE, il a donc besoin d'aide. Ces paramètres indiquent au build systemd quelle est la version de systemd dans l'image PXE locale. La valeur doit être comme le nom de la balise dans le référentiel git systemd, c'est-à-dire <code>&lt;v &lt;numéro&gt;</code> , comme la <code>v229</code> . Si ce paramètre est spécifié, alors <code>--with-coreos-local-pxe-image-path</code> doit également être spécifié. |

## Essai

Il n'y a qu'un seul indicateur pour les tests - pour activer les tests fonctionnels.

|                                        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>--enable-functional-tests</code> | Les tests fonctionnels sont désactivés par défaut. Il y a certaines conditions à remplir pour pouvoir les exécuter. Les tests sont exécutables uniquement sous Linux. Les tests doivent être exécutés en tant que root, donc le système de build utilise sudo pour y parvenir. Notez que l'utilisation de sudo peut tuer la non-interactivité du système de génération, alors assurez-vous que si vous l'utilisez dans certains CI, l'utilisateur CI est un sudoer et n'a pas besoin de mot de passe. De plus, lorsque vous essayez d'exécuter des tests fonctionnels avec la saveur d'hôte de l'image stage1, l'hôte doit être géré par systemd d'au moins la version v220. Si l'une des exigences ci-dessus n'est pas remplie et que la valeur du paramètre est oui, configure va se renflouer. Cela peut ne pas être idéal dans un environnement CI, il existe donc une troisième valeur possible pour ce paramètre - "auto". "auto" permettra des tests fonctionnels si toutes les conditions sont remplies. Sinon, il les désactivera sans aucune erreur. |
|----------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

## Sécurité

Ces indicateurs sont liés à la sécurité.

|                           |                                                                                                                                                                                                                                                                                           |
|---------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>--enable-tpm</code> | Cette option pour activer la journalisation sur le TPM est définie par défaut. Pour que la journalisation fonctionne, TrouSerS est requis. Définissez cette option sur auto pour activer conditionnellement les fonctionnalités TPM en fonction de la prise en charge de la construction. |
|---------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

|                                   |                                                                                                                                                            |
|-----------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>--enable-insecure-go</code> | Cette option pour autoriser la construction de rkt avec go ayant des problèmes de sécurité connus n'est pas définie par défaut. Utilisez-le avec prudence. |
|-----------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------|

## Développement

|                                         |                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|-----------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>--enable-incremental-build</code> | Cette option active la compilation incrémentielle. Ceci est utile pour le développement local. Contrairement à une version build, cette option permet l'installation go vs vs build qui diminue le temps de compilation incrémentiel. Notez que cette option n'est pas prise en charge dans les versions de compilation croisée. Pour cette raison, l'option de génération incrémentielle ne doit pas être utilisée pour les versions de version. |
|-----------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

## Dépendances d'exécution

Linux 3.18+ (idéalement 4.3+ pour que la superposition sur superposition fonctionne), avec les options suivantes configurées:

```
CONFIG_CGROUPS
CONFIG_NAMESPACES
CONFIG_UTS_NS
CONFIG_IPC_NS
CONFIG_PID_NS
CONFIG_NET_NS
CONFIG_OVERLAY_FS (nice to have)
```

## Exemple de configuration et de construction de rkt

installation des dépendances

### Récupération du dépôt rkt

Dans cet exemple, `~/Repos` est un espace de travail personnel où tous les dépôts sont stockés

```
mkdir ~/Repos && cd ~/Repos
mkdir -p ~/.local/gopath/src/github.com/rkt
sudo apt-get install git
git -C ~/.local/gopath/src/github.com/rkt clone
https://github.com/rkt/rkt.git
ln -s ~/.local/gopath/src/github.com/rkt/rkt rkt
```

# Installation de Go Programming Language

```
cd ~/Downloads
wget https://storage.googleapis.com/golang/go1.6.1.linux-amd64.tar.gz
tar -xvf go1.6.1.linux-amd64.tar.gz
mv go ~/.local
```

Ajouter les variables GO au fichier .bashrc:

```
export PATH=~/.local/bin:~/.local/go/bin:$PATH
export GOPATH=~/.local/gopath
export GOROOT=~/.local/go
```

## Construction de rkt

Exécuter l'autogen et configurer les commandes avec les arguments appropriés, par exemple (kvm comme flavor):

```
./configure --enable-tpm=no --with-stage1-flavors=kvm,fly --with-coreos-  
local-pxe-image-path=/src/rkt/common/pxe.img --with-coreos-local-pxe-image-  
systemd-version=v233
```

Maintenant, construire rkt avec:

```
make V=2 -j
```

## Quelques commandes utiles

### Construction et exécution des tests:

```
./tests/build-and-run-tests.sh -f kvm
```

### Exécution des tests fonctionnels après la génération:

```
make functional-check
```

## Execution d'un test spécifique:

```
make functional-check GO_TEST_FUNC_ARGS='-run TEST_NAME_HERE'
```

## Utilisation simple du conteneur rkt (exécuter, quitter, supprimer):

```
sudo ./build-rkt-*/bin/rkt run --insecure-options=image --interactive  
docker://busybox  
exit  
sudo ./build-rkt-*/bin/rkt gc --grace-period=0
```

## Suppression de toutes les interfaces réseau créées par rkt:

```
for link in $(ip link | grep rkt | cut -d':' -f2 | cut -d'@' -f1);  
do  
    sudo ip link del "${link}"  
done
```

From:

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

Permanent link:

<http://www.ouarte.garden/doku.php?id=virtualisation:rkt-overview>

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

