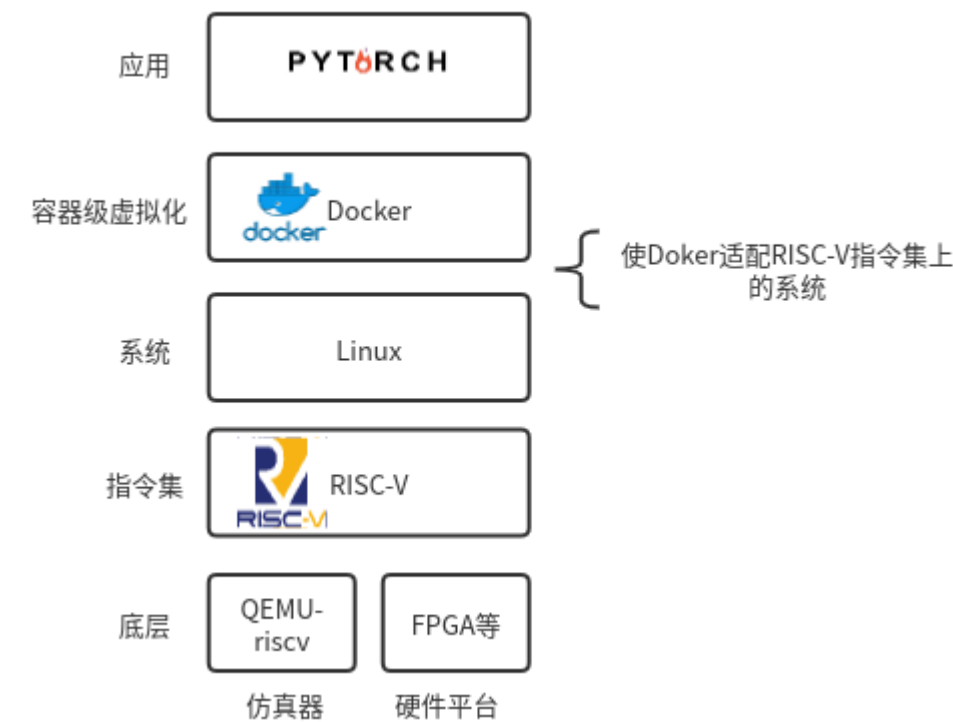


Report 0319

工作进度

为了在riscv架构下的linux中实现虚拟化，我们之前的系统架构如下：



- 上周之前我遇到的问题主要有：
- 使用的Linux镜像功能不完整，缺少安装Docker的相关依赖，无法联网，无法与宿主机通信，不方便调试等；导致工作前进。

这两周我编译了新的linux系统镜像，解决了大部分问题，现在已经有有了一个较好的调试环境：

- 可以访问网络，支持git等，可以直接下载软件包进行安装；
- 可以通过ssh和宿主机通信；

但是我在安装Docker的过程中，由于系统的底层架构问题，出现如下情形：

安装过程参考[官方文档](#)

- 2、直接从网络下载源码安装

```
[root@stage4 docker]# sudo dnf install docker-ce docker-ce-cli containerd.io
Last metadata expiration check: 15:02:22 ago on Mon Mar 16 14:30:42 2020.
Error:
Problem 1: conflicting requests
- package docker-ce-cli-1:18.09.0-3.fc28.x86_64 does not have a compatible architecture
- package docker-ce-cli-1:18.09.1-3.fc28.x86_64 does not have a compatible architecture
- package docker-ce-cli-1:18.09.2-3.fc28.x86_64 does not have a compatible architecture
- package docker-ce-cli-1:18.09.3-3.fc28.x86_64 does not have a compatible architecture
- package docker-ce-cli-1:18.09.4-3.fc28.x86_64 does not have a compatible architecture
- package docker-ce-cli-1:18.09.5-3.fc28.x86_64 does not have a compatible architecture
- package docker-ce-cli-1:18.09.6-3.fc28.x86_64 does not have a compatible architecture
- package docker-ce-cli-1:18.09.7-3.fc28.x86_64 does not have a compatible architecture
- package docker-ce-cli-1:18.09.8-3.fc28.x86_64 does not have a compatible architecture
- package docker-ce-cli-1:18.09.9-3.fc28.x86_64 does not have a compatible architecture
Problem 2: conflicting requests
- package docker-ce-18.03.1.ce-3.fc28.x86_64 does not have a compatible architecture
- package docker-ce-18.06.0.ce-3.fc28.x86_64 does not have a compatible architecture
- package docker-ce-18.06.1.ce-3.fc28.x86_64 does not have a compatible architecture
- package docker-ce-18.06.2.ce-3.fc28.x86_64 does not have a compatible architecture
- package docker-ce-18.06.3.ce-3.fc28.x86_64 does not have a compatible architecture
- package docker-ce-3:18.09.0-3.fc28.x86_64 does not have a compatible architecture
- package docker-ce-3:18.09.1-3.fc28.x86_64 does not have a compatible architecture
- package docker-ce-3:18.09.2-3.fc28.x86_64 does not have a compatible architecture
- package docker-ce-3:18.09.3-3.fc28.x86_64 does not have a compatible architecture
- package docker-ce-3:18.09.4-3.fc28.x86_64 does not have a compatible architecture
- package docker-ce-3:18.09.5-3.fc28.x86_64 does not have a compatible architecture
- package docker-ce-3:18.09.6-3.fc28.x86_64 does not have a compatible architecture
- package docker-ce-3:18.09.7-3.fc28.x86_64 does not have a compatible architecture
- package docker-ce-3:18.09.8-3.fc28.x86_64 does not have a compatible architecture
- package docker-ce-3:18.09.9-3.fc28.x86_64 does not have a compatible architecture
Problem 3: conflicting requests
- package containerd.io-1.2.0-1.2.beta.2.fc28.x86_64 does not have a compatible architecture
- package containerd.io-1.2.0-2.0.rc.0.1.fc28.x86_64 does not have a compatible architecture
- package containerd.io-1.2.0-2.2.rc.2.1.fc28.x86_64 does not have a compatible architecture
- package containerd.io-1.2.0-3.fc28.x86_64 does not have a compatible architecture
- package containerd.io-1.2.2-3.3.fc28.x86_64 does not have a compatible architecture
- package containerd.io-1.2.2-3.fc28.x86_64 does not have a compatible architecture
- package containerd.io-1.2.4-3.1.fc28.x86_64 does not have a compatible architecture
```

报错信息说明**does not have a compatible architecture**

- 2、直接安装静态二进制文件

```
[root@stage4 ~]# tar xzvf docker-19.03.8.tgz
docker/
docker/ctr
docker/containerd-shim
docker/docker
docker/docker-proxy
docker/dockerd
docker/containerd
docker/runc
docker/docker-init
[root@stage4 ~]# ls
docker  docker-19.03.8.tgz
[root@stage4 ~]# cd docker
[root@stage4 docker]# ls
containerd  ctr  docker-init  dockerd
containerd-shim  docker  docker-proxy  runc
[root@stage4 docker]# sudo ./docker run hello-world
./docker: ./docker: cannot execute binary file
[root@stage4 docker]# ./docker
-bash: ./docker: cannot execute binary file: Exec format error
[root@stage4 docker]#
```

执行依然出错，因为官方给出的二进制文件中，并没有适配riscv架构：



所以目前来说，直接在riscv架构上安装Docker是不可行的，所以我想尝试了另外两种方法：

- 1、在riscv架构的linux系统中模拟目标硬件 通过运行一个全功能模拟器，我们可以启动一个可以运行Linux 操作系统的通用 arm/x86_64等架构的虚拟机，然后在虚拟机中设置开发环境，编译应用程序。整个模拟系统同样可以使用QEMU实现。
- 2、通过QEMU模拟目标架构的用户空间 在Linux 系统上，QEMU 有另外一种操作模式，可以通过用户模式模拟器来运行非本地架构的二进制程序。该模式下，QEMU 会跳过方法 1 中描述的对整个目标系统硬件的模拟，取而代之的是通过 binfmt_misc 在Linux 内核注册一个二进制格式处理程序，将陌生二进制代码拦截并转换后再执行，同时将系统调用按需从目标系统转换成当前系统。最终对于用户来说，实现在本机运行这些异构二进制程序。其他架构的二进制程序可以在Docker提供的页面中找到。

目前的进展

方法 1 的思路更加简单直接，我先尝试了按照方法 1 来做，在虚拟机中安装QEMU，build一个x86_64架构：

```
configure linux-headers qemu-storage-daemon.c
[root@stage4 qemu]# mkdir build
[root@stage4 qemu]# cd build/
[root@stage4 build]# ls
[root@stage4 build]# ../configure --target-list=x86_64-linux-user
ERROR: glib-2.48 gthread-2.0 is required to compile QEMU
[root@stage4 build]#
```

出现的问题是，缺少两个依赖文件：glib-2.48, gthread-2.0

下周会努力尝试安装好这两个依赖文件，如果因为架构问题安装失败的话，会继续尝试方法 2。