# Docker离线安装部署文档

**环境：**

CentOS 7.2.1511 内核版本 3.10.0-327.e17.x86\_64

docker 1.12.6版本

**思路：**

1. 在可以连接外网的机器（和内网的机器系统版本一致）上通过yum下载rpm以及相关依赖和需要更新的rpm；

2. 将下载好的rpm拷贝到内网，构建yum源；

3. 使用命令 yum install docker 安装

**docker离线安装包下载：**

通过yum install命令下载相关rpm包时，因为yum命令默认keepcache=0，所以下载安装后相关的rpm包会被yum删除。但是yum提供了只下载不安装的参数，所以我们通过命令：

yum install –downloadonly –downloaddir=/home/dockerRpm docker （切勿复制粘贴）

下载相关的rpm文件到/home/dockerRpm文件夹下（包括安装包，依赖包，更新包）。至此我们已经完成相关rpm包的下载，接下来我们将拷贝到内网机器并制作yum源。

**构建本地yum源：**

1. 将dockerRpm文件夹拷贝到内网/data/下

2. 备份/etc/yum.repo.d/下的所有.repo文件：mkdir bak/ && mv \*.repo bak/

3. 创建新的repo文件，如：docker.repo，进行配置：

[docker]

name=docker

baseurl=<file:///data>/dockerRpm

gpgcheck=0

enabled=1

4. createrepo -d /data/dockerRpm 设置本地源（如果没有该命令，可通过使用ios镜像配置离线yum源后安装使用）

5. 清除缓存：yum clean all

6. 创建缓存：yum makecache

7. 查看构建的本地源：yum repolist

**安装docker：**

yum install docker

**安装过程中可能遇到的问题总结：**

1. 将命令yum install –downloadonly –downloaddir=/home/dockerRpm docker 直接复制粘贴执行，导致命令参数不被识别，直接将docker安装到了机器上，相关的rpm包被yum删除（因为默认keepcache=0。可通过 cat /etc/yum.conf | grep ‘keepcache’ 查看keepcache值，如果值为1，通过cat /etc/yum.conf | grep ‘cachedir’ 查看缓存路径，去该路径提取相关rpm包），如果使用yum卸载了docker，重新执行命令。之后又会发现相关的更新包并不会重新下载。

解决方法：

通过yum回滚命令回归到安装docker之前的版本：

(1) yum history list docker 查看安装docker之前的版本号

(2) yum history undo <版本号> 回滚到之前的版本

（如果回滚之后还是发现，没有回滚相关的更新包，可以通过查看/var/log/yum.log查看yum日志，通过命令：cat /var/log/yum.log | grep -i ‘update’ 筛选出yum更新的包）

2. 虽然筛选出来了相关的yum更新过的包，但是当你用yum重新下载该包时，又会发现提示现在系统已经安装此版本的包，而无法下载。

解决方法：

通过yum工具包里面的yumdownloader命令下载相关包。（注意：此命令只会下载你需要的包，并不会下载此包需要的依赖包。）

(1) yum install yum-util 安装yum工具包

(2) yumdownloader <包名>

3. 当你走到最后一步 yum install docker 时，或许你觉得一切都没问题了，终于搞定了。但是yum不会这么轻易的放过你，你又可能会遇到错误：“保护多版本”。T-T

解决方法：

yum install –setopt=protected\_multilib=false docker 搞定！