容器云基础：在 Ubuntu上安装Docker及基本用法

Docker 是一个开源项目，为开发者和系统管理员提供了一个开放的平台，在任何地方通过打包和运行应用程序作为一个轻量级的容器。Docker 在软件容器内自动部署应用程序。Docker 最开始由 Solomon Hykes 作为 dotCloud 一个内部开发项目，一个企业级的 PaaS （platform as a service 服务平台），该软件现在由 Docker 社区和 Docker 公司维护，更多 Docker 信息你可以访问：<https://docs.docker.com/>。

我们可以通过 Docker 官方提供的 KVM 与 Docker 的图片更加形象的知道什么是 Dock：

安装 Docker 所需条件：需要 64 位架构的系统和Linux 3.10 内核或更高版本。这里作者使用了[Ubuntu](http://www.linuxidc.com/topicnews.aspx?tid=2)15.04 系统的 3.19 内核版本。

**关于 Docker 再多了解一些**

在这里你可以了解到 docker 世界最基本的条件。

**Docker Images**

Docker image 是 Docker container 最基本的模板。image 通用容器使系统和应用易于安装，Docker image 是用来运行的容器，你可以找到许多 images （多种操作系统和软件已经被安装好了的 Docker）在这里 <https://hub.docker.com/>.

**Docker Container**

Docker 容器（Docker Container）是一个 Image，在运行的 Docker image 上读取和写入。Docker 是一个联合的文件系统作为容器后台，容器的任何变化，都将被保存在一个基本 image 新的层上。我们安装应用程序的层就是容器。每个在主机机上运行的容器都是独立的，因此，提供了一个安全的应用平台。

**Docker Registry**

Docker registry 是为 Docker images 提供的库。它提供了公共和私有库。公共 Docker 库被叫做 Docker Hub。这里我们能够上传 push 和 pull 我们自己的 images.

**在 Ubuntu 15.04 上安装 Docker**

以下我们将指导你如何安装 docker。在安装之前我们需要检查 kernel 版本和操作系统架构。

运行命令：

uname -a

你可以看到我们使用的是 ubuntu 15.04 64位版本和 kernel  3.19 内核。

现在运行安装 Docker 的命令：

sudo apt-get install -y docker.io

等待安装完毕，现在我们使用下面的命令启动 Docker：

systemctl start docker

运行系统引导时启用 docker，命令：

systemctl enable docker

你可能想核对一下 docker 版本：

docker version

现在，docker 已经安装在您的系统上。您可以从 Docker 库先下载 Docker Image 制作的容器。

**Docker 的基本用法**

在本节中，我将向您介绍 Docker 命令的常用选项。例如如何下载一个 docker image，打造一个容器，以及如何访问容器。

要创建一个新的容器，你应该选择一个基本 image 的操作系统，例如启动 Ubuntu 或者 [CentOS](http://www.linuxidc.com/topicnews.aspx?tid=14) 或其他系统。您可以搜索一个基本 image 使用 Docker 搜索命令：

docker search ubuntu

该命令将显示所有 ubuntu images，你可以自己尝试一下搜索 centos Images。

现在我们现在 base image到我们的服务中，使用命令：

docker pull ubuntu

现在，您可以通过使用命令来查看所有已下载的images：

docker images

Ubuntu 镜像从DockerHub/Docker Registry下载。下一步骤是创建从该镜像的容器。

要创建容器，可以使用docker create 或 docker run

docker create ubuntu:14.04

docker create 命令会创建一个新的容器，但不会启动它。所以现在你需要使用运行命令：

docker run -i -t ubuntu:14.04 /bin/bash

此命令将创建并运行一个基于 Ubuntu14.04 镜像的容器，容器内并运行一个命令/bin/bash，您将在容器内自动运行命令。

当你输入 Exit 命令退出容器时，容器也是停止运行，如果你想容器在后台运行需要在命令后面添加 -d 参数。

docker run -i -t -d ubuntu:14.04 /bin/sh -c “while true; do echo hello world; sleep 1; done”

/bin/sh -c “while true; do echo hello world; sleep 1; done” this is bash script to echo **“hello word”**forever.

现在你可以看到容器在后台运行通过命令：

docker ps

如果你想从 bash 命令看日志结果，使用命令：

docker logs NAMES/ContainerID

怎样在后台访问容器 shell？这个命令将会连接你的容器 shell：

docker exec -i -t NAMES/ContainerID

你可以看到主机名和容器ID是相等的，这意味着你在容器shell内。当你在shell 上键入’exit`，会离开的shell，但容器仍在运行。

你会经常使用的另一个命令是：

docker stop NAME/ContainerID

这将停止容器而不将其删除，这样你就可以用命令重新启动它：

docker start NAME/ContainerID

如果你想删除的容器，先停止它，然后用命令将其删除：

docker rm NAME/ContainerID