



docker

Docker Tutorials

Communicating Between Containers

Introduction

- This scenario explores how to allow multiple containers to communicate with each other. The steps will explain how to connect a data-store, in this case, **Redis**, to an application running in a separate container.



Step 1 - Start Redis

- 连接到容器的最常见的情况是连接数据存储的应用程序(eg. redis,mysql,postgresql)。创建链接的关键方面是容器的名称。所有容器都有名字，但为了使链接更容易，重要的是定义正在连接的容器的名字。
 - *docker run -d --name <friendly-name> <image-name>*
- Run a redis server with a friendly name of *redis-server* which we'll connect to in the next step. This will be our source container.
 - *docker run -d --name redis-server redis*
 - ```
$ docker run -d --name redis-server redis
12d490011847b3f40bd5fd8ae5602e8bdc5182769f6b9dba523e20174449ee14
```

## Step 2 - Create Link ❶

- To connect to a source container you use the *--link* *<container-name | id>:<alias>* option when launching a new container. The container name refers to the source container we defined in the previous step while the alias defines the friendly name of the host.

## Step 2 - Create Link②

- In this example, we bring up a **Alpine** container which is linked to our redis-server. We've defined the alias as redis. When a link is created, Docker will do two things
  - Docker will *set some environment variables based on the linked to the container.*
  - *docker run --link redis-server:redis alpine env*
  - Docker will update the **HOSTS** file of the container with an entry for our source container with three names, the original, the alias and the hash-id.
  - *cat /etc/hosts*
  - *docker run --link redis-server:redis alpine cat /etc/hosts*
- Test Connection
  - *docker run --link redis-server:redis alpine ping -c 1 redis*

# Step 3 - Connect To App

- *docker run -d -p 3000:3000 --link redis-server:redis katacoda/redis-node-docker-example*
- Test Connection
  - *curl docker:3000*

# Step 4 - Connect to Redis CLI

In the same way, you can connect source containers to applications, you can also connect them to their own CLI tools.

- Launching Redis CLI
  - *docker run -it --link redis-server:redis redis redis-cli -h redis*

# Reference

- <https://www.katacoda.com/courses/docker/5>