



北京 上海 珠海 苏州 杭州

主办: PyChina.org, GDG

2014 年 11 月 10 日

leo@docker.cn

Docker & registry

孟繁亮

Docker.cn

agenda

- ☞ **Docker local**
- ☞ **Docke client®istry&index**
- ☞ **Docker Hub & Registry API**
- ☞ **扩展 Docker API**
- ☞ **Python docker-registry**

2014 年 11 月 10 日

leo@docker.cn

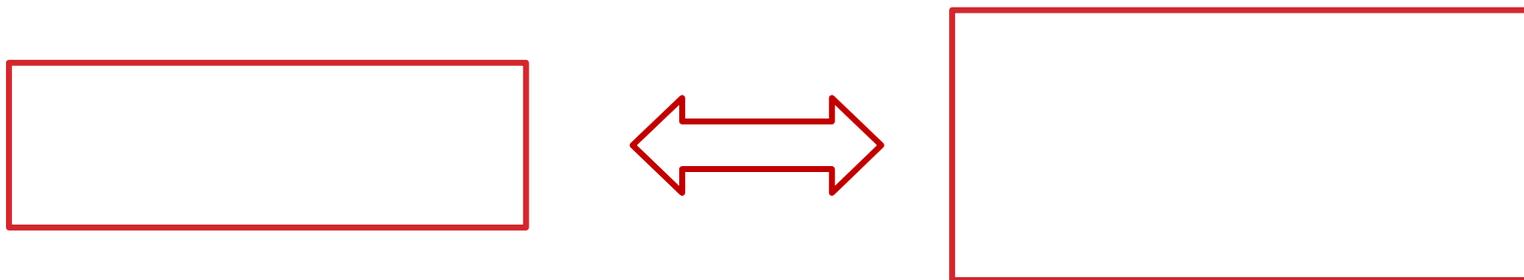
Docker Python project

- 🔗 **docker-registry**
- 🔗 **fig**
- 🔗 **docker-py**
- 🔗 **bender**
- 🔗 **openstack-docker**
- 🔗 **openstack-heat-docke**
- 🔗 **docker-status**

2014 年 11 月 10 日

leo@docker.cn

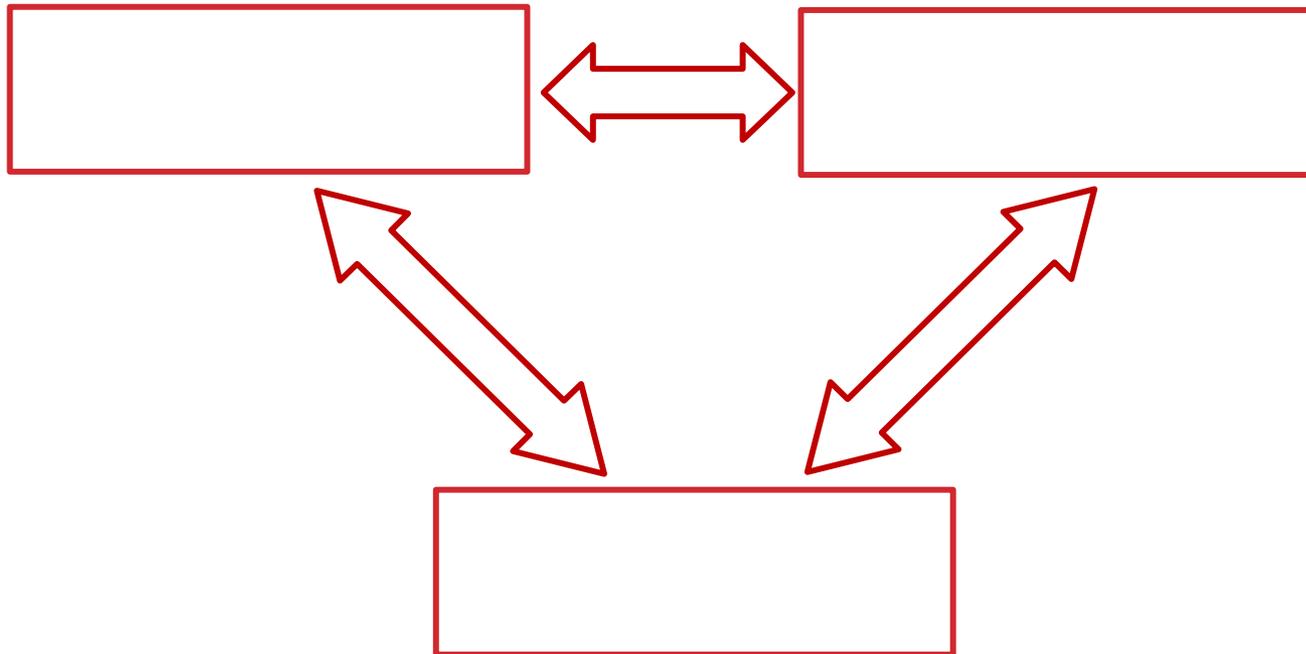
Docker - local



2014 年 11 月 10 日

leo@docker.cn

docker-registry



2014 年 11 月 10 日

leo@docker.cn

Docker local storage

- 存储路径 `/var/lib/docker` 或者 `docker -d -g /xxx/somewhere`
- 镜像仓库信息存储
 - `repositories-aufs`
 - `repositories-devicemapper ...`
- AUFS** : `diff/layer/mnt`
- Devicemapper** : `devicemapper/metadata/mnt`
- gz**、**bzip2**、**gzip**、**xz**

2014 年 11 月 10 日

leo@docker.cn

Docker hub & Registry 概念

- ❧ **[standalone]** Hub 和 Registry 是否同一个程序提供服务
- ❧ **[standalone=true]** 在 HTTP HEADER 的 Authorization 中使用 Basic Authorization 验证
- ❧ **[standalone=false]** 在 HTTP HEADER 的 Authorization 字段使用 Token 验证
- ❧ **[X-Docker-Token=true]** 生成一个 Token 返回, 否则 docker client 会报 Index response didn't contain an access token 的错误
- ❧ **[Cookie]** docker client 支持 Cookie, 所以在 PUSH 的第一次调用的时候就处理

2014 年 11 月 10 日

leo@docker.cn

Docker login

- ⌘ **[PUT /v1/users]** 创建用户；如果 **HTTP STATUE CODE** 返回 **401** ， **docker client** 会进行登录。
- ⌘ **[GET /v1/users]** 用户登录：使用 **BASIC AUTHORIZATION** 验证。
 - ⌘ **Authorization: Basic ZnNrOmZsdfdf==**

2014 年 11 月 10 日

leo@docker.cn

Docker PUSH

- ❧ **[PUT /v1/repositories/<namespace>/<repository>]** 把镜像仓库所有的镜像和标签信息组织为 **JSON** 格式 **PUT** 到服务器。
- ❧ **[GET /v1/images/<image>/json]** 获取镜像的 **JSON** 信息，如果 **HTTP STATUS CODE** 是 **200**，判定服务器已经有此镜像，不需要再上传；如果是 **404**，判断服务器不存在此镜像，进行上传。
- ❧ **[PUT /v1/images/<image>/json]** 把镜像的 **JSON** 信息发送到服务器。
- ❧ **[PUT /v1/images/<image>/layer]** 把镜像目录压缩为 **tar** 流，放在 **HTTP BODY** 里面发送到服务器
- ❧ **[PUT /v1/images/<image>/checksum]** 在 **HTTP HEADER** 中发送镜像的 **checksum** 值，**X-Docker-Checksum** 和 **X-Docker-Checksum-Payload**
- ❧ **[PUT /v1/repositories/<namespace>/<repository>/tags/<tag>]** 向服务器写入镜像和标签的对应信息
- ❧ **[PUT /v1/repositories/<namespace>/<repository>/images]**

2014 年 11 月 10 日

leo@docker.cn

Docker PULL

- ⌘ **[GET /v1/repositories/<namespace>/<repository>/images]** 从服务器读取镜像仓库的所有镜像信息
- ⌘ **[GET /v1/repositories/<namespace>/<repository>/tags]** 从服务器读取镜像仓库的标签 信息
- ⌘ **[GET /v1/images/<image>/ancestry]** 获取一个镜像的祖先信息。
- ⌘ **[GET /v1/images/<image>/json]** 获取镜像的 **JSON** 信息。
- ⌘ **[GET /v1/images/<image>/layer]** 获取一个镜像的 **tar** 文件。

2014 年 11 月 10 日

leo@docker.cn

Docker ENcrypt

- 通过增加 **HTTP HEADER** 信息扩展 **Registry API** 。
- 在服务器和 **docker client** 的 **HTTP HEADER** 中增加 **[X-Docker-Encrypt]** ，相互通知是否支持镜像加密。
- 在服务器和 **docker client** 的 **HTTP HEADER** 中增加 **[X-Docker-Sign]** ，通知加密和解密的密码签名以验证。

2014 年 11 月 10 日

leo@docker.cn

Python docker-registry src

核心 py

- **app.py**
- **index.py**
- **search.py**
- **tags.py**

关键路由

- **/v1/_ping**
- **/v1/search**
- **/v1/users**
- **/v1/repositories**
- **/v1/images**

2014 年 11 月 10 日

leo@docker.cn

End & Thanks

2014 年 11 月 10 日

leo@docker.cn