1. 环境准备
2. 服务器环境准备

PHP环境 安装教程

https://blog.csdn.net/anzhen0429/article/details/79272893

PHP 安装需要一些扩展, 主要是redis和mysql的扩展

yum install php-mbstring php-xml php-xmlrpc php-pecl-redis php-pecl-xdebug php-xcache php-pecl-memcached php-opcache

安装 php composer 包依赖工具

<https://getcomposer.org/download/>

php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"

php -r "if (hash\_file('SHA384', 'composer-setup.php') === '544e09ee996cdf60ece3804abc52599c22b1f40f4323403c44d44fdfdd586475ca9813a858088ffbc1f233e9b180f061') { echo 'Installer verified'; } else { echo 'Installer corrupt'; unlink('composer-setup.php'); } echo PHP\_EOL;"

php composer-setup.php

php -r "unlink('composer-setup.php');"

NGINX 环境

1. 设置nginx的官方源

vim /etc/yum.repos.d/nginx.repo

[nginx]

name=nginx repo

baseurl=http://nginx.org/packages/centos/7/$basearch/

gpgcheck=0

enabled=1

手动安装 nginx 服务器

yum install nginx

配置 nginx 域名服务

Mysql 环境

安装 docker 环境 使用官方源

vim /etc/yum.repos.d/docker.repo

[dockerrepo]

name=Docker Repository

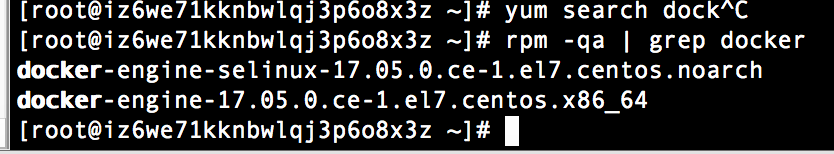
baseurl=https://yum.dockerproject.org/repo/main/centos/7/

enabled=1

gpgcheck=1

gpgkey=https://yum.dockerproject.org/gpg

完成后, yum install docker-engine, 会按照2个包表示安装完成



使用 docker 容器作为mysql 服务, 启动命令是

docker run -d -p 3306:3306 -v /opt/mysql\_data:/var/lib/mysql -e MYSQL\_ROOT\_PASSWORD=T8ceN7HxujTw%sNX --restart=always --name mysql3306 hub.c.163.com/library/mysql:5.6

mysql 服务权限

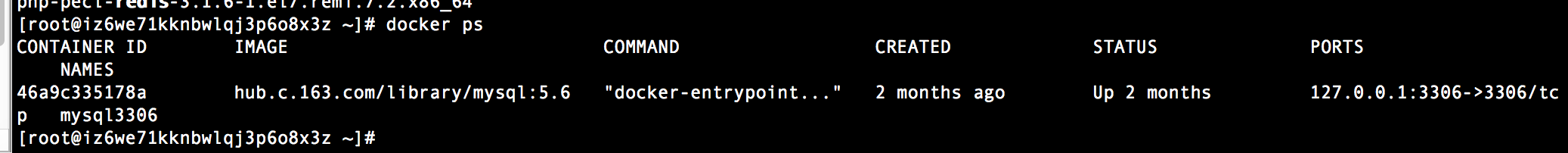
mysql5.6

root T8ceN7HxujTw%sNX

服务器启动后, 需要确认下加下应用的访问权限

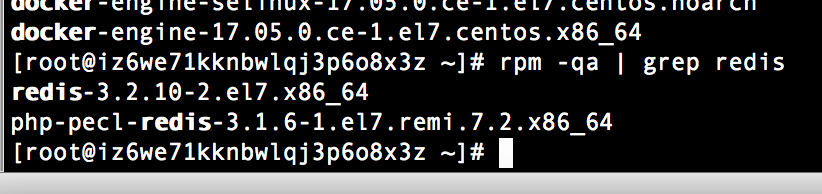
grant all privileges on db\_helpe.\* to helpe@`localhost` identified by 'Y6RZPOtMfa4!^@ww';

正常启动如下, docker ps 命令查看正在运行的 docker 容器



Redis 环境

Yum install redis



1. 线上环境配置文件:

线上配置文件放到仓库 <https://github.com/Dong-Sky/Helpe-Back/evns>

php 配置文件

默认配置没有做调整

nginx 配置文件 是线上的配置文件

nginx\_conf.d.tar.gz

nginx ssl 文件

testapi.helpe.online\_ssl.tar.gz

redis 配置文件

redis.conf 主要是加了密码访问 requirepass F2hsXBRdbqSx2DGD

mysql 配置文件可以忽略, 跟默认生成的一致

执行数据库备份

mysqldump -h 127.0.0.1 -uroot -p'T8ceN7HxujTw%sNX' db\_helpe > db\_helpe\_20180905.sql

db\_helpe\_20180905.sql.tar.gz

二. 部署项目

部署项目依赖

git clone 仓库

删除 vender 下的所有文件 和 composer.lock 文件

设置php composer 源

composer config -g repositories.packagist composer https://packagist.phpcomposer.com

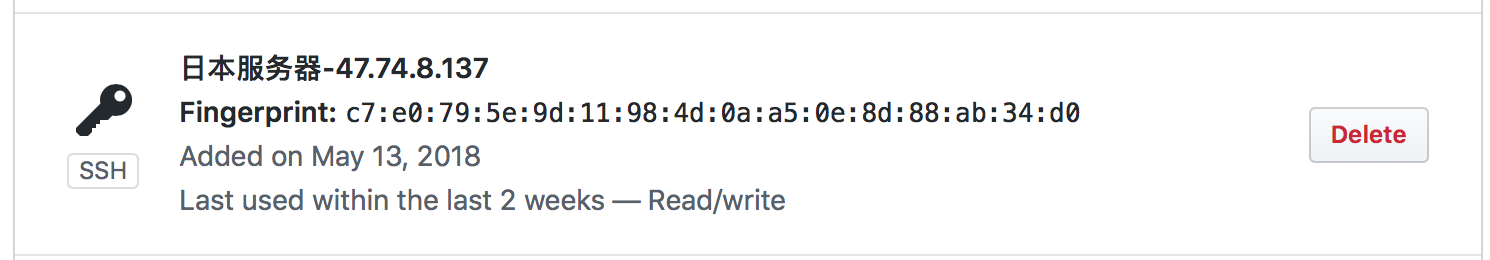
设置asset插件

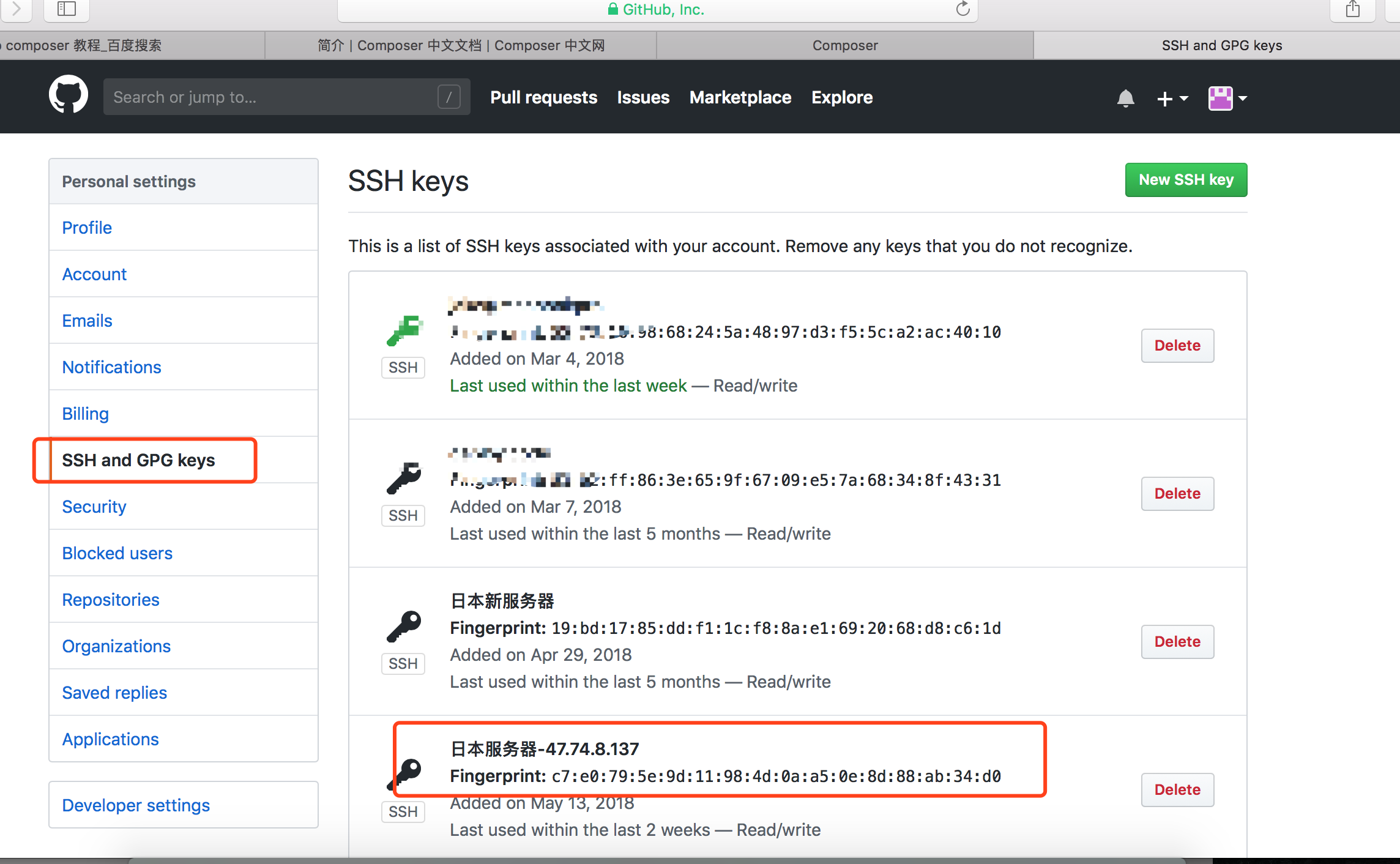
composer global require "fxp/composer-asset-plugin:^1.2.0"

克隆代码

git clone git@github.com:Dong-Sky/Helpe-Back.git

如果不需要密码访问,需要设置个人github的访问权限, 我把日志服务器的ssh key 作为部署key 设置在我个人的访问key里, 就可以通过无密码方式, 访问git 仓库





在当前项目下执行 composer install 命令, 添加依赖, 依赖添加完成,

执行当前脚本

# 上线部署到正式的目录

rsync -av --delete /opt/helpe\_bak/Helpe-Back/\* /data/code/ --exclude "images" --exclude "runtime"

# 前端部署

rsync -av --delete /opt/helpe\_bak/Helpe-HP/\* /data/wwwroot/www.helpe.online/www/ --exclude ".git"

三. 账号相关

后台管理：

helpe, rkHN7tYP32

admin, rkHN7tYP32d23s11SW

阿里云主账号登录

dmchoicer

Dong950922

阿里云管理账号密码

http://signin.aliyun.com/1382110770233645/login.htm

ft20082

q1w2e3r4

阿里云 OSS 数据访问

子账号 ft20082

AccessKeyId : LTAIMlUcOZ9ZLOXs

AccessKeySecret : yk6M46oV70Byle0bY8i0snaZC37Gqt

新服务器账号

47.74.34.112,

root, Dong950922..

root, #fa3hizow@3$1Sj#1riX^kHN77tYP!u8

47.74.8.137

root, fa3hizDw@3$1Sj1riX^kHN7tYP!u8

helpe, 1riX^kHN7tYP!u8

更改为新密码

指定修改 composer 中国源的方式

数据库权限

mysql5.6

root T8ceN7HxujTw%sNX

grant all privileges on db\_helpe.\* to helpe@`localhost` identified by 'Y6RZPOtMfa4!^@ww';

docker run -d -p 3306:3306 -v /opt/mysql\_data:/var/lib/mysql -e MYSQL\_ROOT\_PASSWORD=T8ceN7HxujTw%sNX --restart=always --name mysql3306 hub.c.163.com/library/mysql:5.6

Redis 服务器权限

redis 6379 F2hsXBRdbqSx2DGD

facebook 第三方登录

appid:212122489339371

appsecret:80f7fb624eaa4fa4142209c8c3dff9f7

mailgun 配置信息

id:dongzhutian2996@gmail.com

password:Dong950922

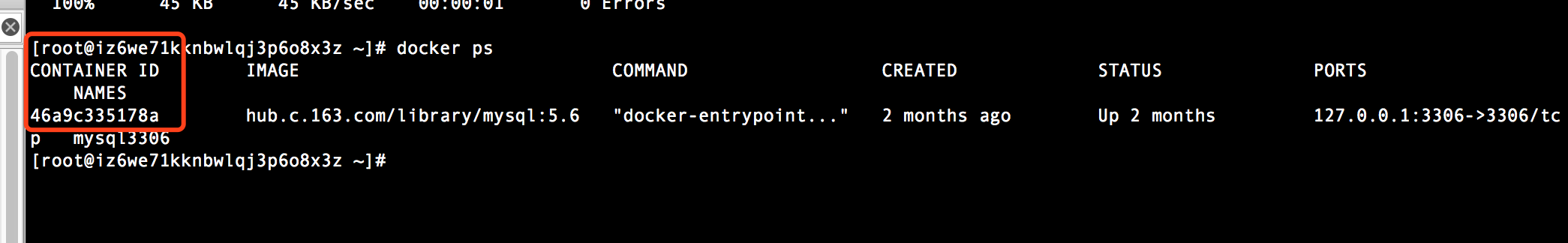
四: 日常运维

1. redis 故障

查看相关redis 问题, 进行重启, 或者调整redis内存大小

1. mysql 故障

由于是容器起的服务, 所有只需要 docker ps 下看下mysql的 contain id ,



可以通过重启命令 docker restart ID 来重启容器

如果修改配置文件, 需要修改 /opt/mysql\_data 这个目录下文件, 修改后,需要重启才能生效.

1. Php 故障

Service restart php-fpm

如果压力比较大, 可以调整PHP的参数, 已适应更大的并发需求

4. nginx 故障

最大的可能性是参数配置错误导致的.

可扩展架构图,

