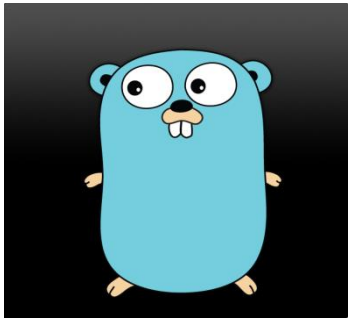
基于Harbor的高可用企业级私有容器镜像仓库部署实践

Tony Bai
@Neusoft Cloud Technology

About Me

- 白明 (Tony Bai)
- @Neusoft Cloud Technology
- Gopher
- Translator & Author
- GopherChina lecturer
- Blogger
- mainly focus on docker & kubernetes recently



OSC 源创会 Opensource Innovation Meetup | IT大咖说 知识分享平台

Tony Bai

一个程序员的心路历程

- 关于我
- 文章列表

docker

kubernetes by Google

五年前

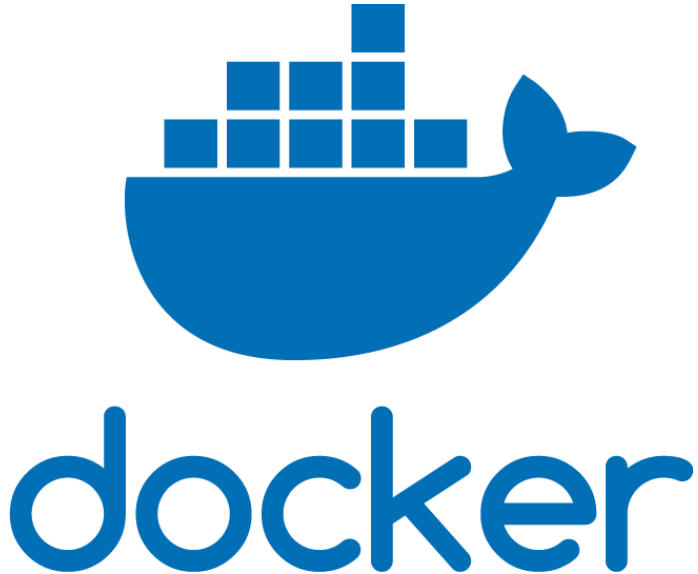

vmware™


VM


Xen™


KVM

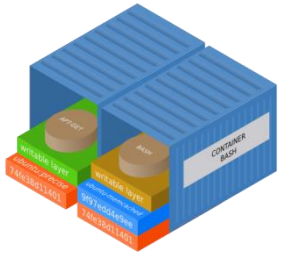
Now


docker

What is Docker

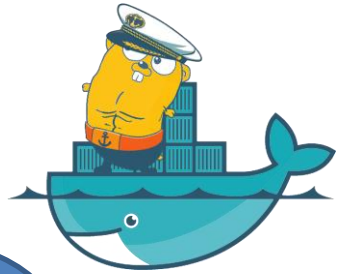
Solaris container
by Sun at 2005

Linux container(LXC)
by Google at 2008



+ Union File System

Docker
by dotCloud at 2013



+ Developer eXperience(DX)

namespaces Cgroups

build, ship and run any app and anywhere

docker run ubuntu “echo hello”

After 4 years

Watch 3,293 Unstar 45,770 Fork 13,627

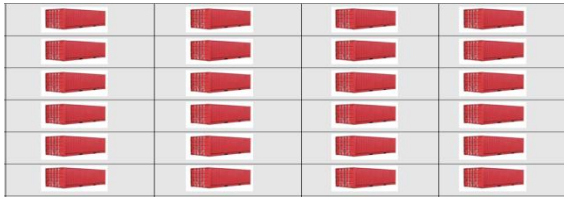
Docker bring us

OSC 源创会 Opensource Innovation Meetup | IT大咖说 知识分享平台

1. 交付标准化

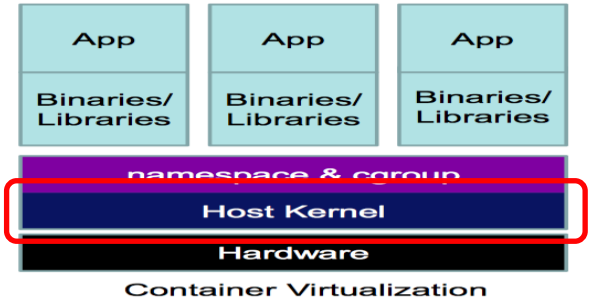
VS.

2. 执行高效化

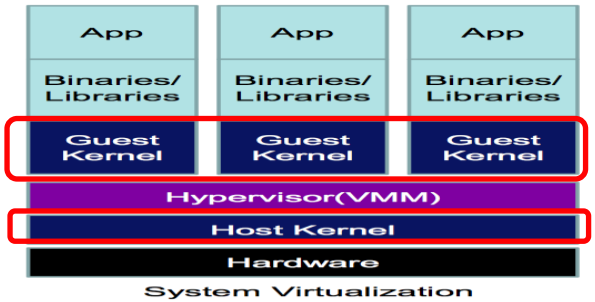


VS.

3. 资源集约化



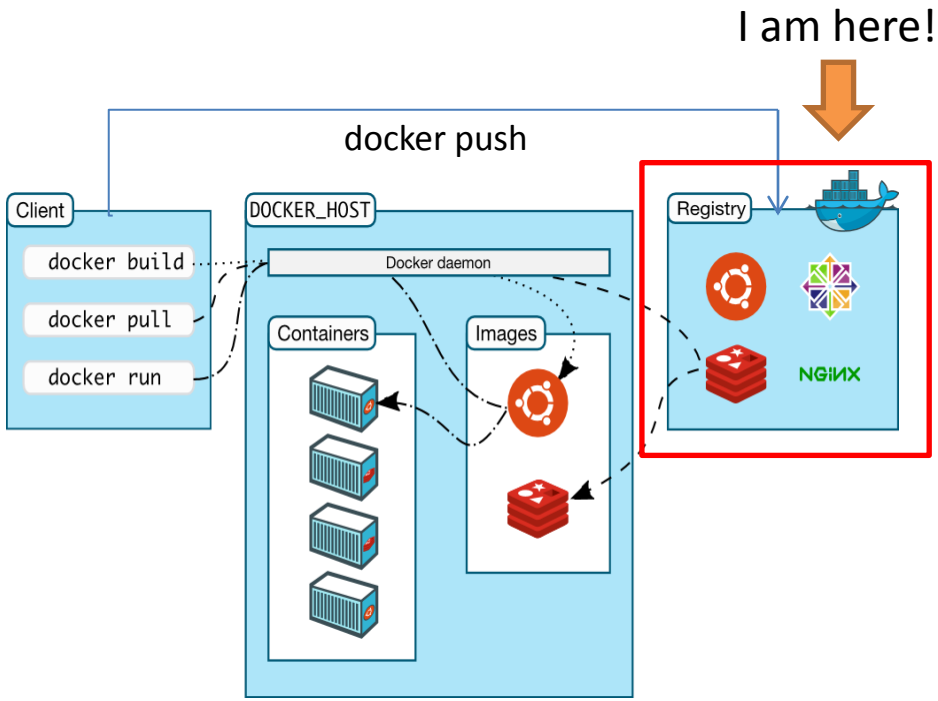
VS.




docker
REGISTRY
TM & © 2015 Docker, Inc.

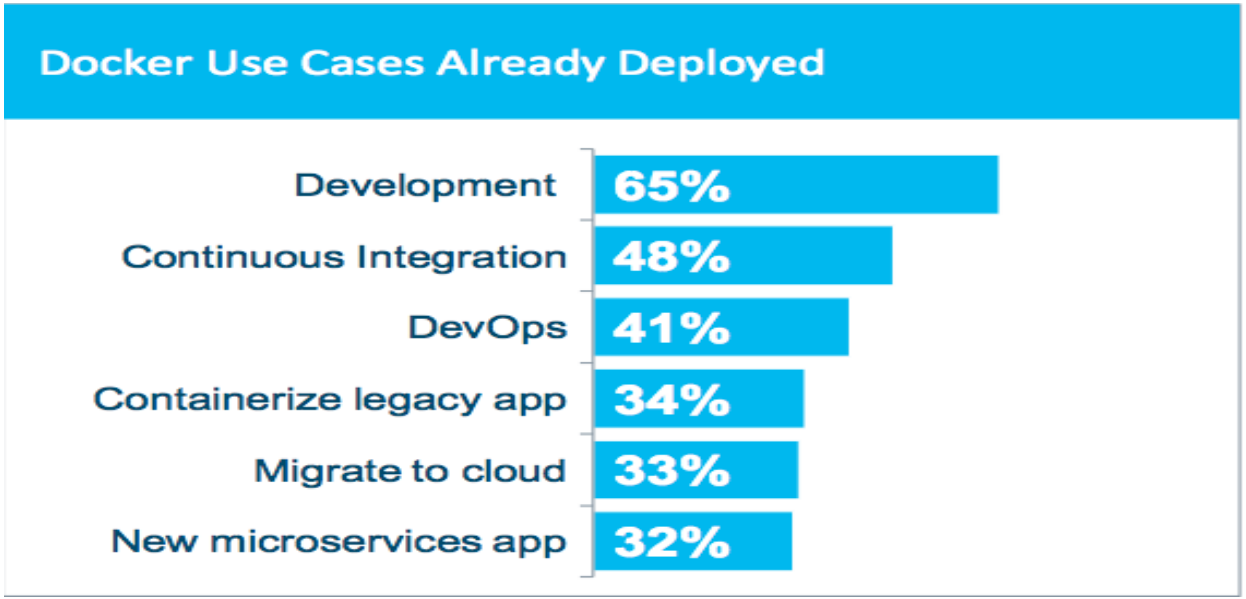
What is Docker Registry

- Docker Registry : 官方镜像存储、管理和分发工具
- 最新实现是distribution，实现了registry2.0协议
- 官方仓库: hub.docker.com
- 国内一般采用加速器



启动一个registry

```
docker run -d -p 5000:5000 --restart=always --name registry registry:2
```



From Docker 2016 survey

Private Registry

- 便于集成到内部CI/CD系统中；
- 对镜像更灵活全面地掌控；
- 数据传输性能更好；
- 出于安全考虑。

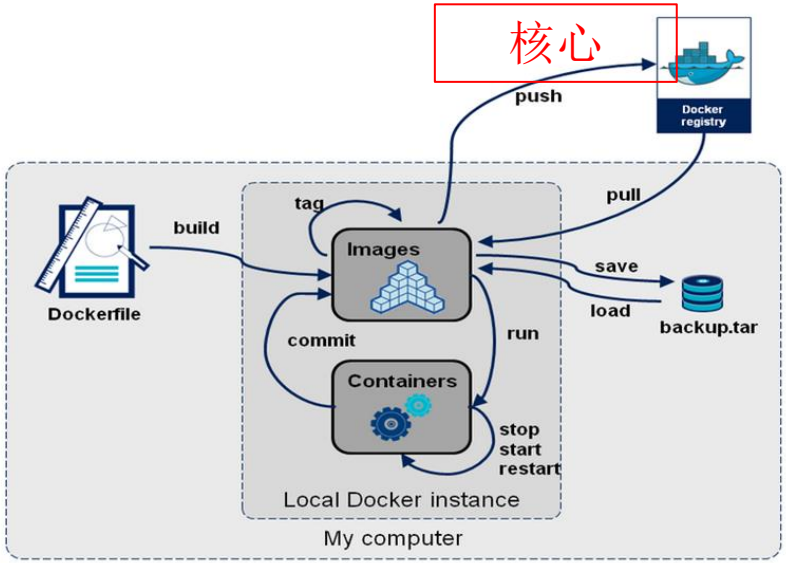
VS.

Features of HARBOR

- VMware中国团队开源的企业级镜像仓库项目，聚焦镜像仓库的企业级需求：
  - 支持基于角色的访问控制RBAC；
  - 支持镜像复制策略（PUSH）；
  - 支持无用镜像数据的自动回收和删除；
  - 支持LDAP/AD认证；
  - Web UI；
  - 提供审计日志功能；
  - 提供RESTful API，便于扩展；
  - 支持中文&部署Easy。

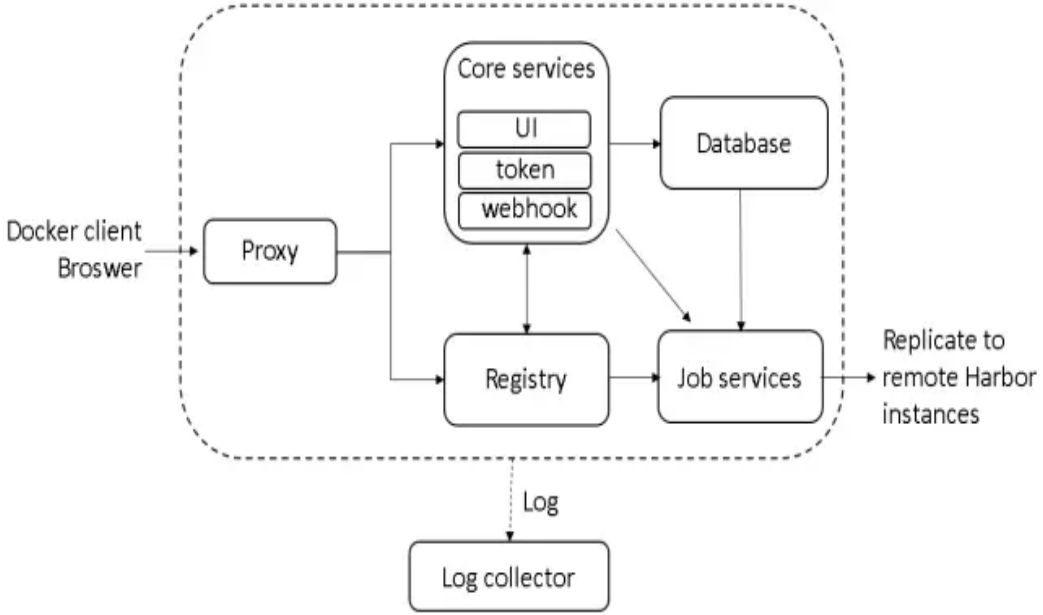
Why High Availability Harbor

- 开发运维一体化流水线的核心组件；
- 单一Registry实例无法满足企业内部大量节点上传和下载的性能需求。



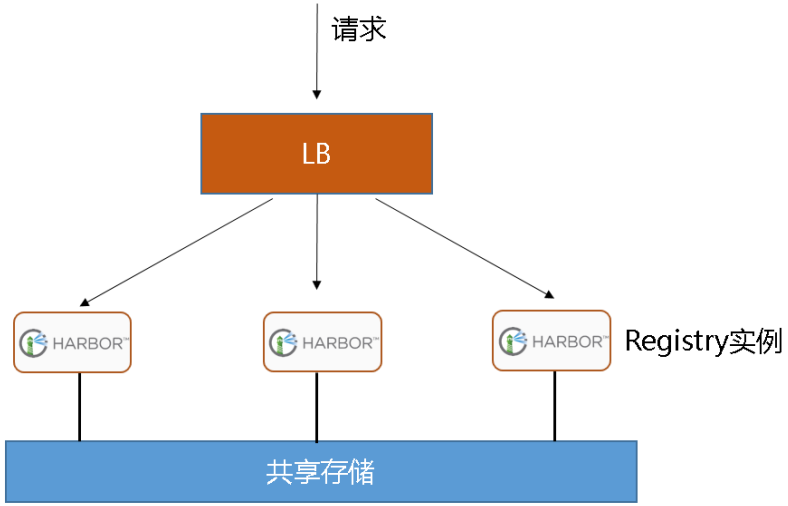
Harbor Architecture

- Proxy(Nginx)
- Registry
- Core Services
  - UI
  - Token Service
  - Webhook
- Job Services

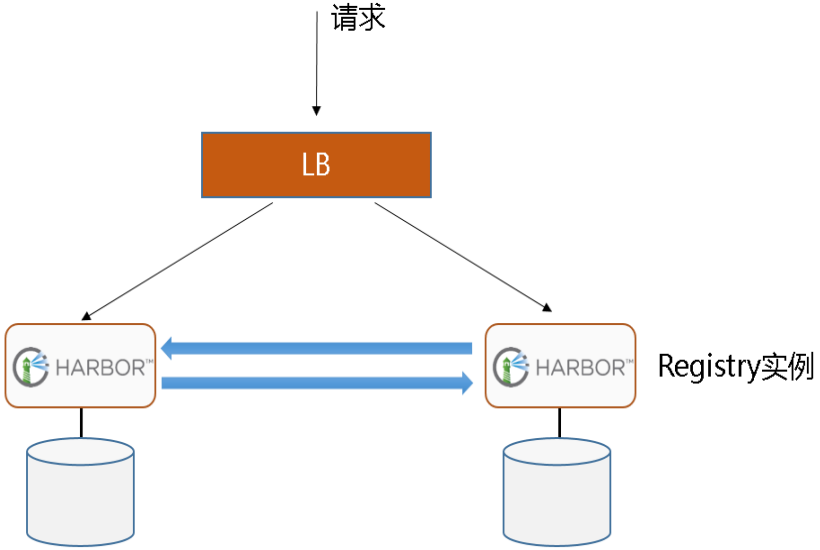


HA Solutions

Solution1: 基于共享存储



Solution2: 基于镜像复制



Cons and Pros

Solution1: 基于共享存储

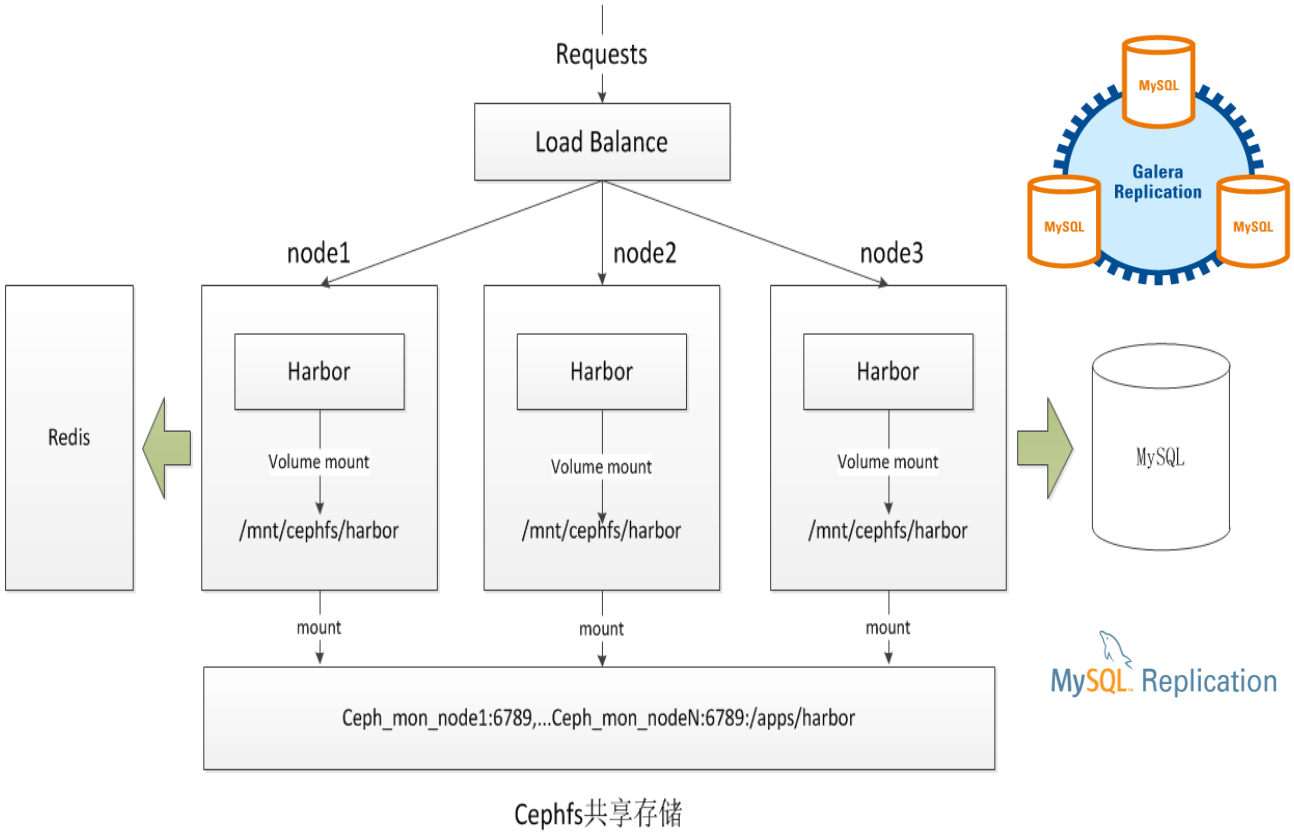
- 优点
  - 标准方案，Scaling好
  - 数据实时一致
- 不足
  - 门槛高，需要具备现成的共享存储
  - 搭建难度略高

Solution2: 基于镜像复制

- 优点
  - 门槛低，搭建简便
- 不足
  - Scaling差，甚至是不能
  - 镜像复制延迟，导致数据阶段性不一致
  - 添加Project时，需手工维护复制规则

Solution details

- Based on CephFS
- External MySQL cluster
- Share sessions in Redis




OSC 源创会
Opensource Innovation Meetup
IT大咖说
知识分享平台
Do It Now!

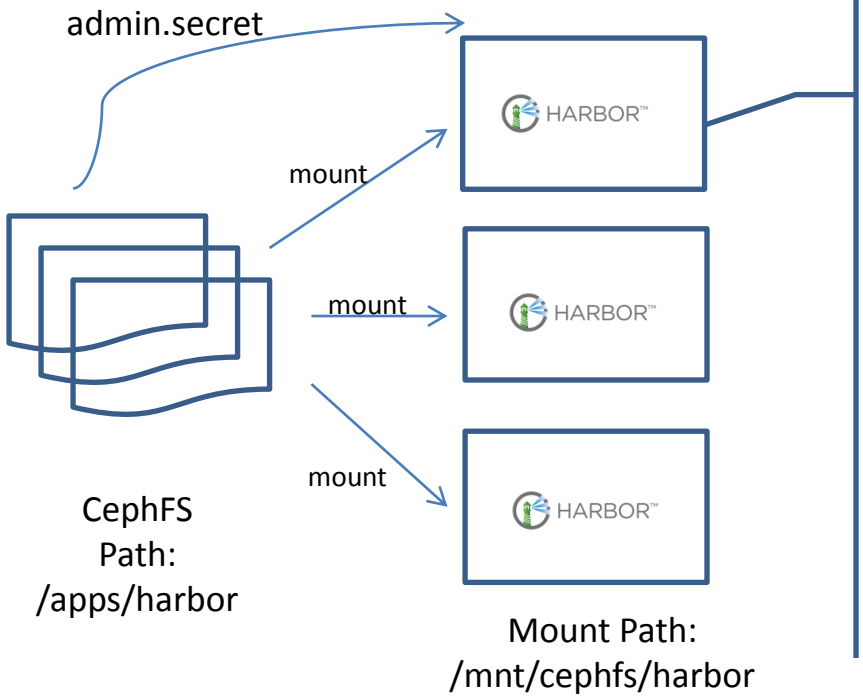
Environment & Prerequisites

- 三台VM(Ubuntu 16.04及以上版本)；
- CephFS、MySQL、Redis已就绪；
- Harbor v1.1.0及以上版本；

```
root@ubuntu-db-1:~/harbor-install/harbor# ls -F
common/  docker-compose.notary.yml  docker-compose.yml  harbor.cfg  install.sh*  LICENSE  NOTICE  prepare*
```

- 一个域名： hub.tonybai.com:8070

Step1: mount CephFS

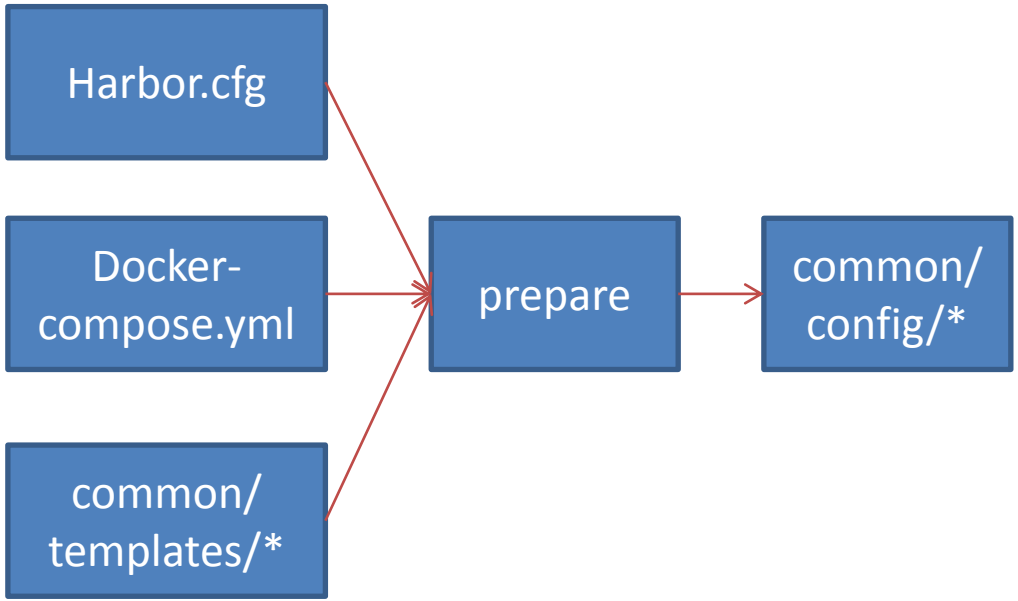


针对每个harbor node:

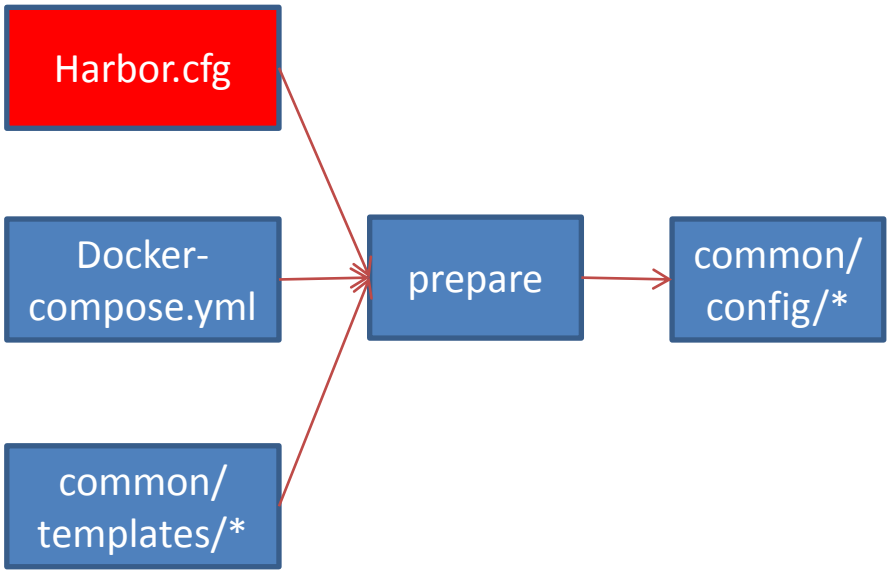
1. apt install ceph-fs-common
2. /etc/fstab添加一行：

```
xx.xx.xx.xx:6789:/apps/harbor
/mnt/cephfs/harbor ceph
name=harbor,secretfile=/etc/ceph/a
dmin.secret,noatime,_netdev 0 2
```

3. mount –a生效


Harbor.cfg
Docker-compose.yml
common/templates/*
prepare
common/config/*

Step2: modify harbor.cfg

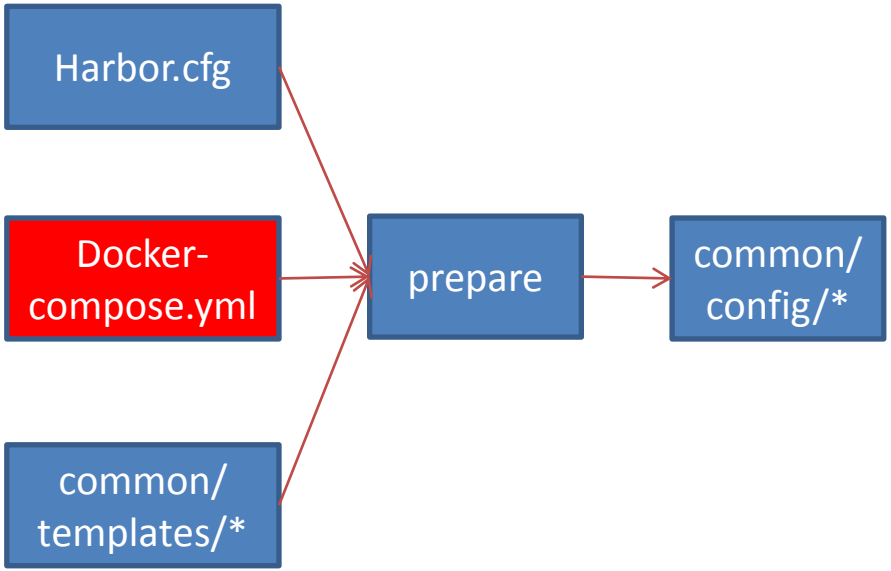


修改如下几项配置：

hostname = hub.tonybai.com:8070

customize_crt = off（保留一个node上的为on）

Step3: modify docker-compose.yml



1. 修改volumes路径

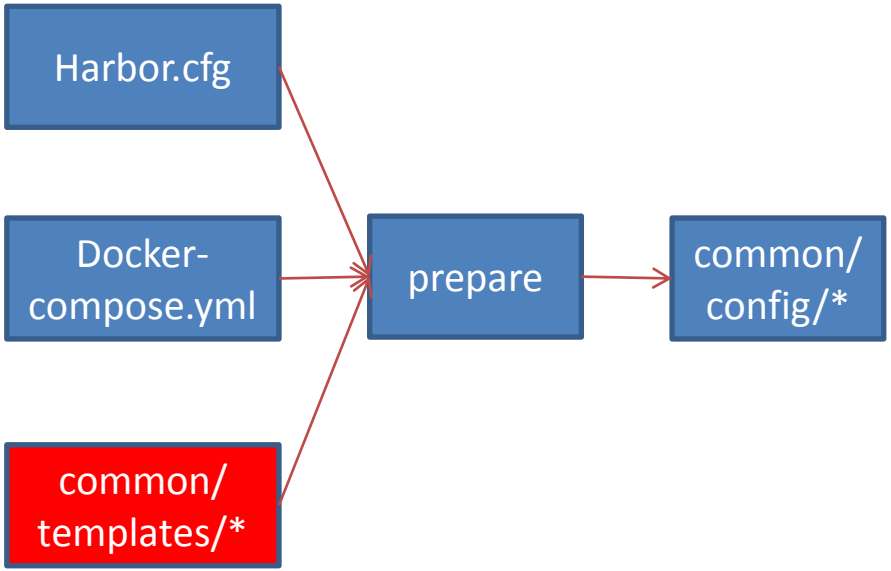
/data/xxx ->
/mnt/cephfs/harbor/data/xxx

2. 删除mysql service以及其他service对mysql service的依赖(depends_on)

3. 修改对proxy外服务端口

```
ports:
    - 8070:80
```

Step4: external db & redis



- common/templates/adminserver/env

```
MYSQL_HOST=harbor_host
MYSQL_PORT=3306
MYSQL_USR=harbor
MYSQL_PWD=harbor_password
RESET=true
```

- common/templates/ui/env

```
_REDIS_URL=redis_ip:6379,100,password,0
```

Step5: prepare&launch harbor

- ./prepare
- docker-compose up –d
- docker-compose ps

```
root@yypdcom1:~/harbor-install/harbor# docker-compose ps
      Name                     Command               State
------------------------------------------------------------
harbor-adminserver   /harbor/harbor_adminserver       Up
harbor-jobservice    /harbor/harbor_jobservice        Up
harbor-log           /bin/sh -c crond && rm -f  ...   Up
harbor-ui            /harbor/harbor_ui                Up
nginx                nginx -g daemon off;             Up
registry             /entrypoint.sh serve /etc/ ...   Up
```

- http://hub.tonybai.com:8070

Troubleshooting

- docker login hub.tonybai.com:8070 failed

现象：

Error response from daemon: Get https://hub.tonybai.com:8070/v1/users/: http: server gave HTTP response to HTTPS client

解决方法：

/etc/docker/daemon.json中添加insecure-registry

```
{
    "insecure-registries": ["hub.tonybai.com:8070"]
}
```

Troubleshooting

- docker login hub.tonybai.com:8070 有时成功，有时failed

现象：

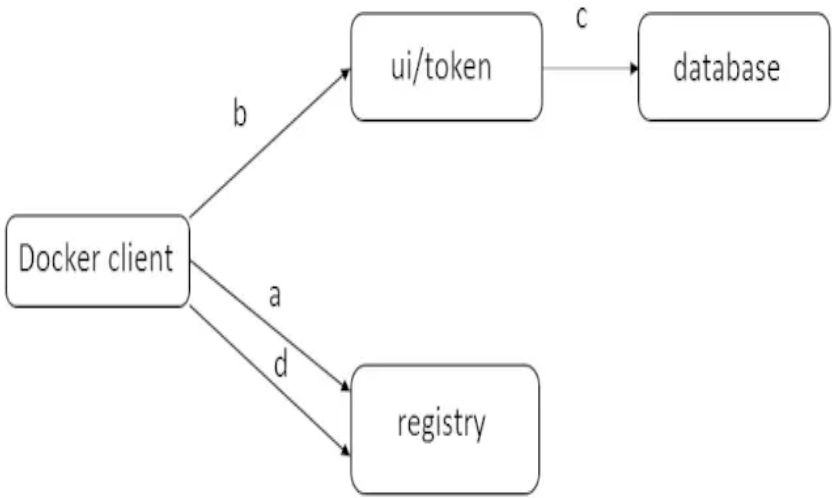
```
# docker login -u user -p passwd http://hub.tonybai.com:8070
Login Succeeded

# docker login -u user -p passwd http://hub.tonybai.com:8070
Error response from daemon: login attempt to
http://hub.tonybai.com:8070/v2/ failed with status: 401
Unauthorized
```

Troubleshooting

解决方法：

- *将一个harbor node上的common/config/ui/private_key.pem和common/config/registry/root.crt复制到其他harbor node;*
- *重新创建harbor各组件。*



资料

- Harbor install 录屏
  https://pan.baidu.com/s/1o8JYKEe

Thank You