CentOS 7 Install Harbor

CentOS Docker 安装

使用官方安装脚本自动安装

$ curl -fsSL https://get.docker.com | bash -s docker --mirror Aliyun

$ sudo usermod -aG docker bigred

卸载旧版本

$ sudo yum remove docker docker-client docker-client-latest docker-common docker-latest docker-latest-logrotate docker-logrotate docker-engine

设置仓库

$ sudo yum install -y yum-utils device-mapper-persistent-data lvm2

使用以下命令来设置稳定的仓库。

$ sudo yum-config-manager --add-repo https://download.docker.com/linux/centos/docker-ce.repo

安装 Docker Engine-Community

$ sudo yum -y install docker-ce docker-ce-cli containerd.io --skip-broken

启动 Docker

$ sudo systemctl start docker

通过运行 hello-world 映像来验证是否正确安装了 Docker Engine-Community 。

$ sudo docker run hello-world

安裝 docker-compose

$ sudo curl -L "https://github.com/docker/compose/releases/download/1.28.6/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose

$ sudo chmod +x /usr/local/bin/docker-compose

$ sudo ln -s /usr/local/bin/docker-compose /usr/bin/docker-compose

查看版本

$ sudo docker version

$ docker-compose version

安裝 Harbor

下載 Harbor 離線版，並解壓縮。

$ wget https://github.com/vmware/harbor/releases/download/v1.2.0/harbor-offline-installer-v1.2.0.tgz

$ tar -xvf harbor-offline-installer-v1.2.0.tgz

建立目錄，並產生 SSL 憑證

$ mkdir cert && cd cert

$ openssl req -sha256 -x509 -days 365 -nodes -newkey rsa:4096 -keyout harbor.imarslan.com.key -out harbor.imarslan.com.crt

查看目錄，記住產生的 SSL 憑證路徑及名稱

（/home/$USERNAME/cert/harbor.imarslan.com.crt）

$ ll

total 8

-rw-rw-r-- 1 bigred bigred 1911 3月 30 17:00 harbor.imarslan.com.crt

-rw-rw-r-- 1 bigred bigred 3268 3月 30 17:00 harbor.imarslan.com.key

編輯 harbor 設定檔

$ cd ;cd harbor ;sudo nano -c harbor.cfg

[line 5/107]hostname = harbor.imarslan.com

[line 9/107]ui\_url\_protocol = https

[line 24/107]ssl\_cert = /home/bigred/cert/harbor.imarslan.com.crt

[line 25/107]ssl\_cert\_key = /home/bigred/cert/harbor.imarslan.com.key

開始安裝 Harbor

$ cd ;cd harbor ;sudo ./install.sh

[sudo] password for bigred:

[Step 0]: checking installation environment ...

Note: docker version: 20.10.5

Note: docker-compose version: 1.28.6

[Step 1]: loading Harbor images ...

Loaded image: vmware/registry:2.6.2-photon

Loaded image: photon:1.0

Loaded image: vmware/notary-photon:signer-0.5.0

Loaded image: vmware/clair:v2.0.1-photon

Loaded image: vmware/harbor-ui:v1.2.0

Loaded image: vmware/harbor-log:v1.2.0

Loaded image: vmware/harbor-db:v1.2.0

Loaded image: vmware/nginx-photon:1.11.13

Loaded image: vmware/postgresql:9.6.4-photon

Loaded image: vmware/harbor-adminserver:v1.2.0

Loaded image: vmware/harbor-jobservice:v1.2.0

Loaded image: vmware/notary-photon:server-0.5.0

Loaded image: vmware/harbor-notary-db:mariadb-10.1.10

[Step 2]: preparing environment ...

Clearing the configuration file: ./common/config/adminserver/env

Clearing the configuration file: ./common/config/ui/env

Clearing the configuration file: ./common/config/ui/app.conf

Clearing the configuration file: ./common/config/ui/private\_key.pem

Clearing the configuration file: ./common/config/db/env

Clearing the configuration file: ./common/config/jobservice/env

Clearing the configuration file: ./common/config/jobservice/app.conf

Clearing the configuration file: ./common/config/registry/config.yml

Clearing the configuration file: ./common/config/registry/root.crt

Clearing the configuration file: ./common/config/nginx/cert/harbor.imarslan.com.crt

Clearing the configuration file: ./common/config/nginx/cert/harbor.imarslan.com.key

Clearing the configuration file: ./common/config/nginx/nginx.conf

loaded secret from file: /data/secretkey

Generated configuration file: ./common/config/nginx/nginx.conf

Generated configuration file: ./common/config/adminserver/env

Generated configuration file: ./common/config/ui/env

Generated configuration file: ./common/config/registry/config.yml

Generated configuration file: ./common/config/db/env

Generated configuration file: ./common/config/jobservice/env

Generated configuration file: ./common/config/jobservice/app.conf

Generated configuration file: ./common/config/ui/app.conf

Generated certificate, key file: ./common/config/ui/private\_key.pem, cert file: ./common/config/registry/root.crt

The configuration files are ready, please use docker-compose to start the service.

[Step 3]: checking existing instance of Harbor ...

Note: stopping existing Harbor instance ...

Removing harbor-log ... done

Removing network harbor\_harbor

[Step 4]: starting Harbor ...

Creating network "harbor\_harbor" with the default driver

Creating harbor-log ... done

Creating harbor-db ... done

Creating registry ... done

Creating harbor-adminserver ... done

Creating harbor-ui ... done

Creating nginx ... done

Creating harbor-jobservice ... done

✔ ----Harbor has been installed and started successfully.----

Now you should be able to visit the admin portal at [https://192.168.105.230](https://192.168.105.230/).

For more details, please visit <https://github.com/vmware/harbor> .

> 以上表示安裝成功。

若出現以下訊息，再次執行即可。

ERROR: for harbor-ui UnixHTTPConnectionPool(host='localhost', port=None): Read timed out. (read timeout=60)

ERROR: for ui UnixHTTPConnectionPool(host='localhost', port=None): Read timed out. (read timeout=60)

ERROR: An HTTP request took too long to complete. Retry with --verbose to obtain debug information.

If you encounter this issue regularly because of slow network conditions, consider setting COMPOSE\_HTTP\_TIMEOUT to a higher value (current value: 60).

访问页面

開啟瀏覽器，輸入：http://192.168.105.230

使用默认的账号:admin

密码:Harbor12345

Harbor 
A Not secure 192.168.105.230/harbor/sign-in 
Q Search Harbor 
VMware Harbor 
admin 
Cl Remember me 
LOG IN 
Sign up for an account 
Forgot password 

登录验证

登录成功之后，效果如下：

Harbor 
Projects 
Logs 
Administration 
Users 
Replication 
Configuration 
Q Search Harbor. 
Projects 
PROJECT 
Project Name 
library 
PROJECTS 
REPOSITORIES 
O PRIVATE 
O PRIVATE 
O puauc 
@ English v 
I TOTAL 
O TOTAL 
All Projects 
admin v 
5 
STORAGE 
Access Level 
Public 
Role 
Project Admin 
Repositories Count 
Creation Time 
3/30/2021, 6:26 PM 
I-loflt 

启动和重启

Harbor 的日常运维管理是通过 docker-compose 来完成的，

Harbor 本身有多个服务进程，都放在docker容器之中运行，可以通过docker ps命令查看。

测试推送镜像

登录到另外一台服务器，确保docker已经安装好了。

本地配置仓库地址

由于默认的harbor，已经创建了一个公开项目 libary。

$ sudo nano /etc/docker/daemon.json

{

　　"registry-mirrors": ["https://njrds9qc.mirror.aliyuncs.com"],

　　"insecure-registries":["192.168.105.230"]

}

重启 Harbor

$ sudo docker-compose up -d

harbor-log is up-to-date

harbor-db is up-to-date

registry is up-to-date

harbor-adminserver is up-to-date

harbor-ui is up-to-date

harbor-jobservice is up-to-date

nginx is up-to-date