基于Docker安装部署Gitlab是基于docker镜像中心的这个镜像进行的：<https://hub.docker.com/r/sameersbn/gitlab/>

下面将几个主要步骤整理出来——

一、启动几个容器

首先启动数据库容器，下面列举了两类，分别是PostgreSQL和MySQL，可按需选择：

**Step 1. Launch a database container**

1. **PostgreSQL**

docker run --name**gitlab-postgresql** -d --restart=always \

--env 'DB\_ENCODING=utf8' \

 --env 'DB\_NAME=gitlabhq\_production' \

--env 'DB\_USER=gitlab' --env 'DB\_PASS=G!t1abPwd' \

--volume /home/docker/gitlab/postgresql:/var/lib/postgresql \

sameersbn/postgresql:9.4-14

1. **MySQL**

docker run --name **gitlab-mysql** -d --restart=always \

--env 'DB\_ENCODING=utf8' \

--env 'DB\_NAME=gitlabhq\_production' \

--env 'DB\_USER=gitlab' --env 'DB\_PASS=G!t1abPwd' \

--volume /home/docker/gitlab/mysql:/var/lib/mysql \

sameersbn/mysql:latest

启动Redis容器：

**Step 2. Launch a redis container**

docker run --name gitlab-redis -d --restart=always \

--volume /home/docker/gitlab/redis:/var/lib/redis \

sameersbn/redis:latest

启动GitLab容器，关联数据库容器和Redis容器：

**Step 3. Launch the gitlab container**

**或 --link gitlab-postgresql:postgresql \**

docker run --name gitlab -d --restart=always \

--link gitlab-redis:redisio  **--link gitlab-mysql:mysql \**

--publish 10022:22 --publish 10080:80 \

--env 'GITLAB\_HOST=175.25.23.162' \

--env 'GITLAB\_TIMEZONE=Beijing' \

--env 'GITLAB\_PORT=10080' --env 'GITLAB\_SSH\_PORT=10022' \

--env 'GITLAB\_ROOT\_PASSWORD=G!t1abPwd' \

--env 'GITLAB\_ROOT\_EMAIL=eayuncloud@eayun.cn ' \

--env 'SMTP\_USER=eayuncloud@eayun.cn' \

--env 'SMTP\_PASS=eayuncloud123' \

--env 'SMTP\_HOST=smtp.eayun.cn' \

--env 'SMTP\_PORT=25' \

--env 'SMTP\_DOMAIN=mail.eayun.cn' \

--env 'GITLAB\_SECRETS\_DB\_KEY\_BASE=long-and-random-alpha-numeric-string' \

--volume /home/docker/gitlab/gitlab:/home/git/data \

sameersbn/gitlab:8.5.5

二、尝试迁移

将原来/home/docker下的所有文件打成tar包，上传到我的阿里云主机上，修改一下路径，重新启动，查看是否数据能完整迁移，包括用户信息、项目信息。

信息完整，迁移方便。

三、Iptables

下面是docker启动后，iptables的详情，其中要关注的是关于docker的记录。具体docker网络运维，可以详细的查一下。

# Generated by iptables-save v1.4.7 on Fri Mar 18 14:52:27 2016

\*nat

:PREROUTING ACCEPT [0:0]

:INPUT ACCEPT [0:0]

:OUTPUT ACCEPT [0:0]

:POSTROUTING ACCEPT [0:0]

:DOCKER - [0:0]

-A PREROUTING -m addrtype --dst-type LOCAL -j DOCKER

-A OUTPUT ! -d 127.0.0.0/8 -m addrtype --dst-type LOCAL -j DOCKER

-A POSTROUTING -s 172.17.0.0/16 ! -o docker0 -j MASQUERADE

-A POSTROUTING -s 172.17.0.2/32 -d 172.17.0.2/32 -p tcp -m tcp --dport 80 -j MASQUERADE

-A POSTROUTING -s 172.17.0.2/32 -d 172.17.0.2/32 -p tcp -m tcp --dport 22 -j MASQUERADE

-A DOCKER ! -i docker0 -p tcp -m tcp --dport 10080 -j DNAT --to-destination 172.17.0.2:80

-A DOCKER ! -i docker0 -p tcp -m tcp --dport 10022 -j DNAT --to-destination 172.17.0.2:22

COMMIT

# Completed on Fri Mar 18 14:52:27 2016

# Generated by iptables-save v1.4.7 on Fri Mar 18 14:52:27 2016

\*filter

:INPUT ACCEPT [0:0]

:FORWARD ACCEPT [0:0]

:OUTPUT ACCEPT [0:0]

:DOCKER - [0:0]

-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT

-A INPUT -p icmp -j ACCEPT

-A INPUT -i lo -j ACCEPT

-A INPUT -p tcp -m state --state NEW -m tcp --dport 10694 -j ACCEPT

-A FORWARD -o docker0 -j DOCKER

-A FORWARD -o docker0 -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT

-A FORWARD -i docker0 ! -o docker0 -j ACCEPT

-A FORWARD -i docker0 -o docker0 -j ACCEPT

-A DOCKER -d 172.17.0.2/32 ! -i docker0 -o docker0 -p tcp -m tcp --dport 80 -j ACCEPT

-A DOCKER -d 172.17.0.2/32 ! -i docker0 -o docker0 -p tcp -m tcp --dport 22 -j ACCEPT

-A INPUT -j REJECT --reject-with icmp-host-prohibited

-A FORWARD -j REJECT --reject-with icmp-host-prohibited

COMMIT

# Completed on Fri Mar 18 14:52:27 2016