

Linux????

- [1 Ubuntu 20.04 安装教程](#)
- [2 安装 sz driver](#)
- [3 21 安装 tcp 工具](#)
- [4 15 安装](#)
- [check_and_sync.sh](#)
- [ubuntu 安装 IP](#)
- [1 v10 安装](#)
- [-20250514](#)
- [Docker 安装](#)
- [安装 rulr-v4gw-adapter](#)
- [1 Ubuntu 安装](#)
- [docker-compose 安装](#)
- [Docker/DockerHub 安装 / 6 18 -](#)

? Ubuntu 20.04 ??????????

```

# Ubuntu 20.04 20TB gdisk parted fdisk
# fdisk 2TB gdisk parted
# gdisk 1 gdisk gdisk
sh
```

```
sudo apt updatesudo apt install gdisk
```

```

# 2 /dev/sdb
sh
```

```
sudo fdisk -l
```

```

# 3 gdisk gdisk gdisk
sh
```

```
sudo gdisk /dev/sdb
```

```
# gdisk
```

- n 1
- 1
-
-
- w

```

# parted 1 parted parted
sh
```

```
sudo apt updatesudo apt install parted
```

```

# 2 /dev/sdb
sh
```



```
sudo parted -l
```

```

# 3 parted parted parted
sh
```

```
sudo parted /dev/sdb
```

```
# parted
```

- 
 Ubuntu 20.04 

????sz_driver??

```
#!/bin/bash
```

```
#####  
#####  
# [] :  
# chmod +x /home/shuncom/services/scripts/sz_driver_mon.sh  
# 00 03 * * 05 /bin/bash /home/shuncom/services/scripts/sz_driver_mon.sh  
#####  
#####
```



sleep 30




```
set -i
source ~/.bashrc
```

```
# 
current_time="`date +%Y%m%d%H%M%S`"
```

```
# sz_driver pid
sz_driver_pid=`pidof sz_driver`
```

```
# [ ] [ ] [ ] [ ] [ ] [ ]  
sz_driver_mon_log="/home/shuncom/services/log/sz_server/${current_time}-sz_driver-mon.log"
```

```
#  sz_driver
if [ -z "${sz_driver_pid}" ]; then
    echo -e "${current_time} sz_driver not start" >> ${sz_driver_mon_log}
    exit 1
fi
```

```
# 
/bin/sh /home/shuncom/services/scripts/driver stop >/dev/null 2>&1
sleep 10
/bin/sh /home/shuncom/services/scripts/driver start >/dev/null 2>&1
sleep 10
/bin/sh /home/shuncom/services/scripts/driver status | grep 'is running' >/dev/null 2>&1
# 
if [ $? -ne 0 ]; then
    echo -e "${current_time} sz_driver restart failed" >> ${sz_driver_mon_log}
    exit 1
fi
# 
echo -e "${current_time} sz_driver restart success" >> ${sz_driver_mon_log}
```

exit 1

??21????tcp?????

```
#!/bin/bash
```

```
#sudo chmod +x /home/shuncom/net_conn_count.sh  
#crontab -e | 0 21 * * * /home/shuncom/net_conn_count.sh
```

```
# [ ] [ ] [ ] [ ] [ ]
```

```
timestamp=$(date +"%Y-%m-%d %H:%M:%S")
```

```
# [ ] [ ] TCP 6011 [ ] [ ] [ ] [ ] [ ]
```

```
conn_count=$(netstat -an | grep ':6011' | grep -i tcp | wc -l)
```

```
# [ ] [ ] [ ] [ ] [ ] eth0 [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ]
```

```
# [ ] [ ] 3[ ] [ ] [ ] [ ] [ ] [ ] [ ] 11[ ] [ ] [ ] [ ] [ ]
```

```
read rx_packets tx_packets <<< $(cat /proc/net/dev | awk '/eth0:/ {gsub(/:/,""); print $3, $11}')
```

```
# [ ] [ ] [ ] [ ] [ ]
```

```
echo "$timestamp TCP 6011[ ] [ ] [ ] : $conn_count, eth0[ ] [ ] [ ] : $rx_packets, [ ] [ ] [ ] : $tx_packets" >>  
/home/shuncom/net_conn_count.log
```

```
shuncom@ubuntu:~$ ./net_conn_count.sh  
shuncom@ubuntu:~$ cat net_conn_count.log  
2025-05-14 13:51:20 TCP 6011连接数: 1, eth0接收包: 975391250, 发送包: 1142964311  
shuncom@ubuntu:~$
```

????15???????

```
#!/bin/bash
```

```
#0 1 * * * /home/shuncom/auto_del_sz_log.sh
```

```
# 目录
```

```
target_dir="/home/shuncom/services/log/sz_server"
```

```
log_file="/home/shuncom/log/del_driver_log.log"
```

```
current_time="$(date '+%Y-%m-%d %H:%M:%S')"
```

```
# 删除文件
```

```
echo "[$current_time] Start deleting files older than 15 days in $target_dir" >> "$log_file"
```

```
find "$target_dir" -type f -mtime +15 -print -delete >> "$log_file" 2>&1
```

```
echo "[$current_time] Deletion complete." >> "$log_file"
```

```
echo "-----" >> "$log_file"
```

check_and_sync.sh

```
#!/bin/bash
#chmod +x /home/shuncom/bin/check_and_sync.sh
#30 1 * * * /home/shuncom/bin/check_and_sync.sh

# 检查 "shun-sync" 进程
PROCESS_NAME="shun-sync"
PROCESS_ID=$(pgrep -f $PROCESS_NAME)

# 如果进程存在
if [ ! -z "$PROCESS_ID" ]; then
    kill $PROCESS_ID
    echo "$(date): Killed process $PROCESS_NAME with PID $PROCESS_ID" >>
/home/shuncom/log/check_and_sync.log
fi

# 执行 sttsync
sh /home/shuncom/bin/sttsync
echo "$(date): Executed sttsync script" >> /home/shuncom/log/check_and_sync.log
```


ubuntu??????IP??

```
#ubuntu18.04
```

```
network:
```

```
version: 2
```

```
ethernets:
```

```
eth0:
```

```
dhcp4: no
```

```
addresses:
```

```
- 222.222.63.231/24
```

```
- 192.168.3.11/24
```

```
gateway4: 192.168.3.1
```

```
routes:
```

```
- to: 222.222.63.0/24
```

```
via: 222.222.63.1
```

```
nameservers:
```

```
addresses:
```

```
- 202.96.209.5
```

```
- 223.5.5.5
```

```
network:
```

```
ethernets:
```

```
eth0:
```

```
addresses:
```

```
- 172.17.20.17/24
```

```
- 192.168.1.17/24
```

```
gateway4: 192.168.1.1
```

```
routes:
```

```
- to: 172.17.20.0/24
```

```
via: 172.17.20.254
```

```
nameservers:
```

```
addresses:
```

```
- 114.114.114.114
```

```
version: 2
```

```
#ubuntu14.04
```

```
# IP eth0
```

```
auto eth0
```

```
iface eth0 inet static
```

```
address 192.168.1.19
```

```
netmask 255.255.255.0
gateway 192.168.1.1
dns-nameservers 192.168.0.5
```

```
# IP address of eth0:1
```

```
auto eth0:1
```

```
iface eth0:1 inet static
```

```
address 10.2.2.19
```

```
netmask 255.255.255.0
```

```
gateway 10.2.2.1
```

```
dns-nameservers 202.96.209.5
```

```
ip route add default via 192.168.1.1 dev eth0 metric 100
```

```
ip route add default via 10.2.2.1 dev eth0:1 metric 200
```



```
grep -i activation /var/log/messages
```

□□□□ V10 □□□□□□□□□□□□□□

sudo kylin-activation status

--	--	--	--	--	--	--

XXXX-XXXX-XXXX-XXXX

2024-10-01

--	--	--	--	--	--	--	--	--

--	--	--	--	--	--	--

[illegible]

GUI

xvfb[]

```
sudo yum install -y xorg-x11-server-Xvfb
```

```
xvfb-run kylin-activation status
```

??????-20250514



ubuntu18.04

```
cp /etc/apt/sources.list /etc/apt/sources.list.bak
```

```
cat > /etc/apt/sources.list << EOF
```

```
deb https://mirrors.aliyun.com/ubuntu/ bionic main restricted universe multiverse
```

```
deb-src https://mirrors.aliyun.com/ubuntu/ bionic main restricted universe multiverse
```

```
deb https://mirrors.aliyun.com/ubuntu/ bionic-security main restricted universe multiverse
```

```
deb-src https://mirrors.aliyun.com/ubuntu/ bionic-security main restricted universe multiverse
```

```
deb https://mirrors.aliyun.com/ubuntu/ bionic-updates main restricted universe multiverse
```

```
deb-src https://mirrors.aliyun.com/ubuntu/ bionic-updates main restricted universe multiverse
```

```
deb https://mirrors.aliyun.com/ubuntu/ bionic-backports main restricted universe multiverse
```

```
deb-src https://mirrors.aliyun.com/ubuntu/ bionic-backports main restricted universe multiverse
```

```
EOF
```

```
apt update && apt upgrade -y
```

```
sudo hostnamectl set-hostname rulr-1
```

```
[[[
```

```
In -sf /usr/share/zoneinfo/Asia/Shanghai /etc/localtime &&\
```

```
apt-get install ntpdate -y &&\
```

```
if ! crontab -l | grep "ntpdate" &>/dev/null; then
```

```
(echo "*/5 * * * * /usr/sbin/ntpdate ntp.aliyun.com >/dev/null 2>&1"; crontab -l) | crontab
```

```
fi
```

```
[[[
```

```
In -sf /usr/share/zoneinfo/Asia/Ho_Chi_Minh /etc/localtime &&\
```

```
apt-get install ntpdate -y &&\
```

```
if ! crontab -l | grep "ntpdate" &>/dev/null; then
```

```
(echo "*/5 * * * * /usr/sbin/ntpdate ntp.aliyun.com >/dev/null 2>&1"; crontab -l) | crontab
```

```
fi
```

```
[[[[[
```

```
In -sf /usr/share/zoneinfo/Africa/Addis_Ababa /etc/localtime &&\
```

```
apt-get install ntpdate -y &&\
```

```
if ! crontab -l | grep "ntpdate" &>/dev/null; then
```

```
(echo "*/5 * * * * /usr/sbin/ntpdate ntp.aliyun.com >/dev/null 2>&1"; crontab -l) | crontab
```

```
fi
```

```
cat >> /etc/sysctl.conf << EOF
```

```
kernel.pid_max = 4194303
```

```
#net.ipv4.tcp_tw_recycle = 0

net.ipv4.tcp_tw_reuse = 1

net.ipv4.ip_local_port_range = 1024 65000

net.ipv4.tcp_syncookies = 1

net.ipv4.tcp_max_tw_buckets = 20480

net.ipv4.tcp_max_syn_backlog = 20480

net.core.netdev_max_backlog = 262144

net.ipv4.tcp_fin_timeout = 20

vm.swappiness = 0

net.core.somaxconn = 20480

vm.overcommit_memory = 1

net.core.rmem_default = 262144

net.core.rmem_max = 16777216

net.core.wmem_default = 262144

net.core.wmem_max = 16777216

net.ipv4.tcp_wmem = 4096 131072 1048576

net.ipv4.tcp_rmem = 4096 131072 1048576

net.ipv4.tcp_mem = 786432 1048576 1572864

net.ipv4.tcp_max_orphans = 262144

net.ipv4.tcp_fastopen = 3

fs.aio-max-nr = 1048576

net.ipv4.tcp_keepalive_time = 1800

net.ipv4.tcp_keepalive_probes = 3

net.ipv4.tcp_keepalive_intvl = 15

net.ipv4.route.gc_timeout = 100
```

```
net.ipv4.tcp_syn_retries = 1
```

```
net.ipv4.tcp_synack_retries = 1
```

EOF

```
sysctl -p
```

```
cat >> /etc/security/limits.conf << EOF
```

```
* soft      nproc      65535
```

```
* hard      nproc      65535
```

```
* soft    nofile    65535
```

```
* hard    nofile    65535
```

```
root soft    nproc      65535
```

```
root hard    nproc      65535
```

```
root soft    nofile      65535
```

```
root hard  nofile  65535
```

EOF

SU -

- ulimit -n

#

```
echo $LANG
```

en_US.UTF-8

[] [] [] [] [] [] "en_US.UTF-8"[] [] [] [] [] [] [] [] [] []

en_US.UTF-8 UTF-8

```
# dpkg-reconfigure locales
```


[[[: [[[[[[[[[[

tab : [[[[[[[[[[

space : [[[[[[[[

[[docker docker-compose rulr-1 rulr-5

[[docker_debs_2021_Rulr6.zip

scp -P 53022 shuncomlog@shuncom.3322.org:/home/shuncomlog/rulr/docker_debs_2021_Rulr6.zip
./

sudo apt install -y unzip

unzip docker_debs_2021_Rulr6.zip &&\

cd docker_debs_2021_Rulr6 &&\

dpkg -i *.deb &&\

mv docker-compose-Linux-x86_64_1.29.2 /usr/bin/docker-compose &&\

chmod +x /usr/bin/docker-compose &&\

usermod -aG docker shuncom &&\

id shuncom &&\

docker -v &&\

docker-compose version

[[[[[[[[

-1

rulr-1

sudo su

```
sudo docker network create --subnet=172.18.0.0/16 rulr-network &&\
```

```
sudo docker network ls
```

```
su - shuncom
```

```
scp -P 53022 -r shuncomlog@shuncom.3322.org:/home/shuncomlog/rulr/30.16/* ./
```

```
cd /home/shuncom
```

```
tar -zxvf servers-20250512.tar.gz
```

```
tar -zxvf www-20250512.tar.gz
```

```
mkdir -p /home/shuncom/log/dubbo/registry/cache &&\
```

```
mkdir -p /home/shuncom/log/nginx &&\
```

```
mkdir -p /home/shuncom/log/shuncom-oauth2-server/tomcat &&\
```

```
mkdir -p /home/shuncom/log/web-maintain &&\
```

```
mkdir -p /home/shuncom/log/web-things &&\
```

```
mkdir -p /home/shuncom/log/web-user &&\
```

```
mkdir -p /home/shuncom/log/model-proxy-server
```

```
cat > /home/shuncom/compose/.env << EOF
```

```
RULR_MYSQL=172.17.20.17
```

```
RULR_ZOOKEEPER=172.17.20.15
```

```
RULR_REDIS=172.17.20.15
```

```
RULR_INFLUXDB=172.17.20.16
```

RULR_RABBITMQ=172.17.20.15

HOSTIP=172.17.20.11

WEB_HOST=172.17.20.11

WEB_PORT=443

RULR_1=172.17.20.11

RULR_3=172.17.20.13

RULR_4=172.17.20.14

JAVA_IMAGE=openjdk:8-alpine3.9-tini

DB_USER=shuncom

EOF

sudo chown -R root.root /home/shuncom/www &&\

sudo chown -R shuncom.shuncom /home/shuncom/www/rulr

2 cpus 1.4

vi docker-compose.yml

%s/'2.8'/'1.4'/gc

rulr-model-proxy-server.jar rulr-2

rulr-2

scp -r /home/shuncom/servers/model-proxy-server/
shuncom@172.17.20.11:/home/shuncom/servers/

docker image load -i nginx_1.20.2.tar &&\

```
docker image load -i openjdk_8-alpine3.9.tar &&\
```

```
docker image load -i openjdk_8-alpine3.9-tini.tar
```

```
docker-compose build &&\
```

```
docker-compose up -d &&\
```

```
docker-compose ps
```

```
■■■■■■■■ \RulrDeploy\ServiceComponents\ops_tools
```

```
■■ installCleanLog31.sh ■■■■
```

```
■■ rotateNginxLog31.sh ■■■■ nginx■■■
```

■■■■■■■■ **-2**

```
rulr-2
```

```
sudo su
```

```
docker network create --subnet=172.18.0.0/24 rulr-network &&\
```

```
docker network ls
```

```
su - shuncom
```

```
scp -P 53022 -r shuncomlog@shuncom.3322.org:/home/shuncomlog/rulr/30.17/* ./
```

```
cd /home/shuncom/
```

```
tar -zxvf servers-20250512.tar.gz
```

```
mkdir -p /home/shuncom/log/dtunt-connect &&\n\nmkdir -p /home/shuncom/log/engine-action-server &&\n\nmkdir -p /home/shuncom/log/engine-alarm-server &&\n\nmkdir -p /home/shuncom/log/engine-timing-server &&\n\nmkdir -p /home/shuncom/log/engine-trigger-server &&\n\nmkdir -p /home/shuncom/log/model-converter-server &&\n\nmkdir -p /home/shuncom/log/model-event-server &&\n\nmkdir -p /home/shuncom/log/model-proxy-server &&\n\nmkdir -p /home/shuncom/log/model-task-schedule
```

```
cat > /home/shuncom/compose/.env << EOF
```

```
RULR_MYSQL=172.17.20.17
```

```
RULR_ZOOKEEPER=172.17.20.15
```

```
RULR_REDIS=172.17.20.15
```

```
RULR_INFLUXDB=172.17.20.16
```

```
RULR_RABBITMQ=172.17.20.15
```

```
HOSTIP=172.17.20.12
```

```
WEB_HOST=172.17.20.11
```

```
WEB_PORT=443
```

```
RULR_1=172.17.20.11
```

```
RULR_3=172.17.20.13
```

```
RULR_4=172.17.20.14
```

```
JAVA_IMAGE=openjdk:8-alpine3.9-tini
```

```
DB_USER=shuncom
```

EOF

2 **cpus** **1.4**

%s/'2.8'/'1.4'/gc

3 **cpus** **1.5**

%s/'3'/'1.5'/gc

cd /home/shuncom/compose/ &&\

docker load -i openjdk_8-ffmpeg.tar &&\

docker load -i openjdk_8-alpine3.9.tar &&\

docker load -i openjdk_8-alpine3.9-tini.tar &&\

docker-compose build &&\

docker-compose up -d &&\

docker-compose ps

su - shuncom

chmod +x /home/shuncom/ops_tools/*

/home/shuncom/ops_tools/installCleanLog31.sh

■■■■■■■■

-3

rulr-3

sudo su

```
docker network create --subnet=172.18.0.0/16 rulr-network
```

```
docker network ls
```

```
su - shuncom
```

```
scp -P 53022 -r shuncomlog@shuncom.3322.org:/home/shuncomlog/rulr/30.18/* ./
```

```
cd /home/shuncom/
```

```
tar -zxvf servers-20250512.tar.gz
```

```
tar -zxvf www-20250512.tar.gz
```

```
su - shuncom
```

```
mkdir -p /home/shuncom/log/advice-server &&\
```

```
mkdir -p /home/shuncom/log/batch-proxy-server &&\
```

```
mkdir -p /home/shuncom/log/dtunt-connect &&\
```

```
mkdir -p /home/shuncom/log/dubbo/registry/cache &&\
```

```
mkdir -p /home/shuncom/log/model-data-resolver &&\
```

```
mkdir -p /home/shuncom/log/model-task-batch &&\
```

```
mkdir -p /home/shuncom/log/nginx &&\
```

```
mkdir -p /home/shuncom/log/pph-proxy-handler/tomcat &&\
```

```
mkdir -p /home/shuncom/log/push-tripartite &&\
```

```
mkdir -p /home/shuncom/log/shuncom-onbon-server
```

```
cat > /home/shuncom/compose/.env << EOF
```

RULR_MYSQL=172.17.20.17

RULR_ZOOKEEPER=172.17.20.15

RULR_REDIS=172.17.20.15

RULR_INFLUXDB=172.17.20.16

RULR_RABBITMQ=172.17.20.15

HOSTIP=172.17.20.13

WEB_HOST=172.17.20.11

WEB_PORT=443

RULR_1=172.17.20.11

RULR_3=172.17.20.13

RULR_4=172.17.20.14

JAVA_IMAGE=openjdk:8-alpine3.9-tini

DB_USER=shuncom

EOF

%s/'2.8'/'1.4'/gc

cd /home/shuncom/compose/ &&\

docker image load -i nginx_1.20.2.tar &&\

docker image load -i openjdk_8-alpine3.9.tar &&\

docker image load -i openjdk_8-alpine3.9-tini.tar &&\

docker-compose build &&\

docker-compose up -d &&\

docker-compose ps


```
su - shuncom
```

```
chmod +x /home/shuncom/ops_tools/*
```

```
/home/shuncom/ops_tools/installCleanLog31.sh
```

```
sudo /home/shuncom/ops_tools/rotateNginxLog31.sh
```

■■■■■■■

-4

```
rulr-4
```

```
sudo su
```

```
docker network create --subnet=172.18.0.0/16 rulr-network &&\
```

```
docker network ls
```

```
su - shuncom
```

```
■■■■■ 192.168.30.41
```

```
scp -P 53022 -r shuncomlog@shuncom.3322.org:/home/shuncomlog/rulr/30.41/* ./
```

```
cd /home/shuncom/
```

```
tar -zxvf servers-20250512.tar.gz
```

```
mkdir -p /home/shuncom/log/dtumt-connect &&\
```

```
mkdir -p /home/shuncom/log/dtumt-handler &&\
```

```
mkdir -p /home/shuncom/log/dtumt-transmission &&\
```

```
mkdir -p /home/shuncom/log/dubbo/registry/cache &&\
mkdir -p /home/shuncom/log/gateway-connect &&\
mkdir -p /home/shuncom/log/gateway-handler &&\
mkdir -p /home/shuncom/log/gateway-proxy-handler &&\
mkdir -p /home/shuncom/log/light-connect-gateway &&\
mkdir -p /home/shuncom/log/light-connect-tcp &&\
mkdir -p /home/shuncom/log/light-proxy-handler &&\
mkdir -p /home/shuncom/log/mt-event-server &&\
mkdir -p /home/shuncom/log/nginx &&\
mkdir -p /home/shuncom/log/tripartite-connect &&\
mkdir -p /home/shuncom/log/tripartite-connect-sensor
```

```
cat > /home/shuncom/compose/.env << EOF
```

```
RULR_MYSQL=172.17.20.17
```

```
RULR_ZOOKEEPER=172.17.20.15
```

```
RULR_REDIS=172.17.20.15
```

```
RULR_INFLUXDB=172.17.20.16
```

```
RULR_RABBITMQ=172.17.20.15
```

```
HOSTIP=172.17.20.14
```

```
WEB_HOST=172.17.20.11
```

```
WEB_PORT=443
```

```
RULR_1=172.17.20.11
```

```
RULR_3=172.17.20.13
```

```
RULR_4=172.17.20.14
```

```
JAVA_IMAGE=openjdk:8-alpine3.9-tini
```

```
DB_USER=shuncom
```

```
EOF
```

```
cd /home/shuncom/compose/ &&\
```

```
docker image load -i nginx-tcp-24.08.19.tar &&\
```

```
docker image load -i nginx-tcp.tar &&\
```

```
docker image load -i openjdk_8-alpine3.9.tar &&\
```

```
docker image load -i openjdk_8-alpine3.9-tini.tar
```

```
# 2
```

```
vi docker-compose.yml
```

```
%s/'2.8'/'1.4'/gc
```

```
cp docker-compose.yml docker-compose.yml.bak
```

```
vi docker-compose.yml
```

```
docker-compose.yml set paste
```

```
# nginx-tcp:
```

```
# image: nginx-tcp
```

```
# build:
```

```
# context: ./
```

```
# dockerfile: NginxTcp
```

%s/image: nginx-tcp/image: nginx-tcp:24.08.19/gc

docker-compose build &&\

docker-compose up -d &&\

docker-compose ps

■■■■■■■■ \RulrDeploy\ServiceComponents\ops_tools

■ ■ ./installCleanLog31.sh ■■■■■■

■ ■ sudo ./rotateNginxLog31.sh ■■■■ nginx■■■■

■■■■■■■■

rulr-5

sudo su

docker network create --subnet=172.18.0.0/16 rulr-network &&\

docker network ls

2 ■■■■■■

su - shuncom

scp -P 53022 -r shuncomlog@shuncom.3322.org:/home/shuncomlog/rulr/redisoorabbitmq/* ./

```
mkdir -p /home/shuncom/rulr/conf/redis &&\n\nmkdir -p /home/shuncom/logs/rabbitmq &&\n\nmkdir -p /home/shuncom/rulr/conf/rabbitmq &&\n\nchmod 757 /home/shuncom/logs/rabbitmq &&\n\nchmod 757 /home/shuncom/rulr/conf/rabbitmq &&\n\nmkdir -p /home/shuncom/rulr/conf/zookeeper &&\n\nmkdir -p /home/shuncom/compose
```

```
cat > /home/shuncom/compose/.env << EOF
```

```
RULR_MYSQL=172.17.20.17
```

```
RULR_ZOOKEEPER=172.17.20.15
```

```
RULR_REDIS=172.17.20.15
```

```
RULR_INFLUXDB=172.17.20.16
```

```
RULR_RABBITMQ=172.17.20.15
```

```
HOSTIP=172.17.20.15
```

```
WEB_HOST=172.17.20.11
```

```
WEB_PORT=443
```

```
RULR_1=172.17.20.11
```

```
RULR_3=172.17.20.13
```

```
RULR_4=172.17.20.14
```

```
JAVA_IMAGE=openjdk:8-alpine3.9-tini
```

```
DB_USER=shuncom
```

```
EOF
```

3 配置

```
cd /home/shuncom/compose &&\

cp redis-5.0.14/6379.conf ../rulr/conf/redis/ &&\

cp rabbitmq-3.7.26/rabbitmq.conf ../rulr/conf/rabbitmq/ &&\

chmod 667 /home/shuncom/rulr/conf/rabbitmq/rabbitmq.conf &&\

cp zookeeper-3.5.8/log4j.properties ../rulr/conf/zookeeper/ &&\

cp zookeeper-3.5.8/zoo.cfg ../rulr/conf/zookeeper/
```

(4) compose

```
# 配置 (docker-compose.yml)

# cd /home/shuncom/compose


docker image load -i rabbitmq_3.7.26.tar &&\

docker image load -i redis_5.0.14.tar &&\

docker image load -i zookeeper_3.5.8.tar


cp docker-compose.yml docker-compose.yml.bak

vi docker-compose.yml


mysql infuxdb


2 CPU


%s/'2.8'/'1.4'/gc


# 8G 4096 2048
```

```
vi /home/shuncom/rulr/conf/rabbitmq/rabbitmq.conf
```

```
vm_memory_high_watermark.absolute=2048MB
```

```
%s/4096/2048/gc
```

4 ☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐

```
cd /home/shuncom/compose &&\
```

```
docker-compose up -d &&\
```

```
docker-compose ps
```

5 rabbitmq ☐☐ **web** ☐☐

```
docker exec -it rulr-rabbit bash
```

```
rabbitmq-plugins enable rabbitmq_management
```

```
exit
```

6 ☐☐☐☐☐☐

(1) ☐☐ **redis**

```
docker exec -it rulr-redis bash
```

```
redis-cli -h 127.0.0.1 -a 'sz_clighting' -p 6379
```

```
127.0.0.1:6379> CONFIG get maxclients
```

```
1) "maxclients"
```

```
2) "10000"
```

```
127.0.0.1:6379> keys *
```

```
(empty list or set)
```

```
127.0.0.1:6379> exit
```

```
root@f39b51dab023:/data# redis-cli --version
```

```
redis-cli 5.0.14
```

```
root@f39b51dab023:/data# exit
```

(2) **rabbitmq**

URL: http://172.17.20.15:15672/

Username: shuncom_mq

Password: sz_clighting_mq

 WEB

```
docker exec -it rulr-rabbit rabbitmqctl status
```

(3) **zookeeper**

```
docker exec -it rulr-zookeeper bash
```

```
zkCli.sh -server 127.0.0.1:2181
```

```
[zk: 127.0.0.1:2181(CONNECTED) 0] ls /
```

```
[zookeeper]
```

```
[zk: 127.0.0.1:2181(CONNECTED) 1] quit
```

```
root@2ad68fbb302e:/apache-zookeeper-3.5.8-bin# exit
```



InfluxDB

rulr-6


```
sudo su
```

1 `influxdb`

```
wget https://dl.influxdata.com/influxdb/releases/influxdb-1.7.10_linux_amd64.tar.gz
```

```
scp -P 53022 -r shuncomlog@shuncom.3322.org:/home/shuncomlog/rulr/influxdb/influxdb-1.8.10_linux_amd64.tar.gz ./
```

2 `influxdb`

```
tar -zxvf influxdb-1.8.10_linux_amd64.tar.gz &&\
```

```
mv influxdb-1.8.10-1 influxdb &&\
```

```
useradd -M -s /usr/sbin/nologin influxdb &&\
```

```
id influxdb
```

```
uid=1601(influxdb) gid=1601(influxdb) groups=1601(influxdb)
```

3 `influxdb`

```
cat > /home/shuncom/influxdb/etc/influxdb/influxdb.conf << EOF
```

```
reporting-disabled = false
```

```
bind-address = "127.0.0.1:8088"
```

```
[meta]
```

```
dir = "/home/shuncom/influxdb/var/lib/influxdb/meta"
```

```
[data]
```

```
dir = "/home/shuncom/influxdb/var/lib/influxdb/data"
```

```
wal-dir = "/home/shuncom/influxdb/var/lib/influxdb/wal"
```

```
series-id-set-cache-size = 100
```

```
[coordinator]
```

[retention]

[shard-precreation]

[monitor]

[http]

enabled = true

bind-address = ":8086"

auth-enabled = false

[logging]

[subscriber]

[[graphite]]

[[collectd]]

[[opentsdb]]

[[udp]]

[continuous_queries]

[tls]

EOF

4 ▯ InfluxDB ▯ ▯ ▯ ▯ ▯ systemd

cat > /lib/systemd/system/influxdb.service << EOF

[Unit]

Description=InfluxDB is an open-source, distributed, time series database

Documentation=<https://docs.influxdata.com/influxdb/>

After=network-online.target

[Service]

User=influxdb

Group=influxdb

LimitNOFILE=65536

ExecStart=/home/shuncom/influxdb/usr/bin/influxd
/home/shuncom/influxdb/etc/influxdb/influxdb.conf \\${INFLUXD_OPTS}

-config

KillMode=control-group

Restart=on-failure

RestartSec=2s

[Install]

WantedBy=multi-user.target

Alias=influxdb.service

EOF

5 ❷ InfluxDB❷

```
cat > /etc/rsyslog.d/influxdb.conf << EOF
```

```
### InfluxDB Rules ###
```

```
if \${programname} == 'influxd' then {
```

```
action(type="omfile" file="/home/shuncom/influxdb/var/log/influxdb/influxd.log")
```

```
stop
```

```
}
```

```
EOF
```

```
# ❷ rsyslog❷
```

```
systemctl restart rsyslog.service
```

```
# [16 empty boxes]
```

```
cat > /home/shuncom/influxdb/etc/logrotate.d/influxdb << EOF
```

```
/home/shuncom/influxdb/var/log/influxdb/influxd.log {
```

```
daily
```

```
rotate 7
```

```
missingok
```

```
dateext
```

```
copytruncate
```

```
compress
```

```
}
```

```
EOF
```

```
chown -R influxdb.influxdb /home/shuncom/influxdb/ &&\
```

```
chmod 757 /home/shuncom/influxdb/var/log/influxdb/
```

6 [] InfluxDB

```
systemctl daemon-reload &&\
```

```
systemctl start influxdb.service &&\
```

```
systemctl enable influxdb.service &&\
```

```
systemctl status influxdb.service
```

```
cat >> ~/.bashrc << EOF
```

```
export PATH=/home/shuncom/influxdb/usr/bin:\$PATH
```

```
EOF
```

```
source ~/.bashrc
```

```
su - shuncom
```

```
cat >> ~/.bashrc << EOF
```

```
export PATH=/home/shuncom/influxdb/usr/bin:\$PATH
```

```
EOF
```

```
source ~/.bashrc
```

```
exit
```

```
influx -version
```

InfluxDB shell version: 1.8.10

7 InfluxDB

(1)

```
influx
```

```
show users
```

```
create user root with password 'root_influxdb' with all privileges
```

```
show users
```

```
exit
```

(2) influxdb

```
influxdb.conf
```

```
sudo vi /home/shuncom/influxdb/etc/influxdb/influxdb.conf
```

```
[http] "auth-enabled = false"
```

```
"auth-enabled = true"
```

(3) InfluxDB

```
systemctl restart influxdb.service
```

(4) InfluxDB

```
influx -host 127.0.0.1 -port 8086 -username root -password 'root_influxdb'
```

```
create database rulr_things;
```

2

```
use rulr_things;
```

```
CREATE RETENTION POLICY three_days ON rulr_things DURATION 3d REPLICATION 1;
```

```
CREATE RETENTION POLICY seven_days ON rulr_things DURATION 7d REPLICATION 1;
```

```
CREATE RETENTION POLICY half_year ON rulr_things DURATION 180d REPLICATION 1;
```

```
CREATE RETENTION POLICY one_year ON rulr_things DURATION 365d REPLICATION 1;
```

```
show retention policies;
```

3

```
create user shuncom with password 'shuncom_influxdb_passwd';
```

```
grant all privileges on rulr_things to shuncom;
```

```
show grants for shuncom;
```

```
exit
```

```
 Mysql
```

Mysql

rulr-7

sudo su

1 安装 mysql

```
wget https://downloads.mysql.com/archives/get/p/23/file/mysql-8.0.20-linux-glibc2.12-x86_64.tar.xz
```

```
scp -P 53022 -r shuncomlog@shuncom.3322.org:/home/shuncomlog/rulr/mysql/* ./
```

2 安装 mysql

(1) 安装 mysql

```
apt install libaio1 -y
```

(2) 安装 mysql

```
tar -xf mysql-8.0.20-linux-glibc2.12-x86_64.tar.xz &&\
```

```
mv mysql-8.0.20-linux-glibc2.12-x86_64/ /home/shuncom/mysql/
```

(3) 安装 mysql

```
groupadd -g 1600 mysql &&\
```

```
useradd -g mysql -u 1600 -M -s /sbin/nologin mysql &&\
```

```
id mysql
```

```
uid=1600(mysql) gid=1600(mysql) groups=1600(mysql)
```

(4) 配置 mysql

mysql 配置文件

mysql

```
mkdir -p /home/shuncom/mysql/data/ &&\
```

```
chown -R mysql:mysql /home/shuncom/mysql/
```

3 配置 mysql my.cnf

```
cat > /etc/my.cnf << EOF
```

```
[client]
```

```
port = 3306
```

```
default-character-set = utf8mb4
```

```
socket = /home/shuncom/mysql/data/mysql.sock
```

```
[mysql]
```

```
no-auto-rehash
```

```
[mysqld]
```

```
default-time_zone = '+00:00'
```

```
user = mysql
```

```
port = 3306
```

```
basedir = /home/shuncom/mysql
```

```
datadir = /home/shuncom/mysql/data
```

```
socket = /home/shuncom/mysql/data/mysql.sock
```

```
bind-address = 0.0.0.0
```

```
pid-file = /home/shuncom/mysql/data/mysqld.pid
```

```
character-set-server = utf8mb4
```

```
collation-server = utf8mb4_0900_ai_ci
```

```
log-error = /home/shuncom/mysql/data/mysqld.log
```

```
slow_query_log = ON
```

```
long_query_time = 2
```


slow_query_log_file = /home/shuncom/mysql/data/mysql-slow.log

max_connections = 1024

open_files_limit = 65535

innodb_buffer_pool_size = 2G

innodb_flush_log_at_trx_commit = 2

innodb_log_file_size = 256M

transaction_isolation = READ-UNCOMMITTED

transaction_isolation = READ-COMMITTE

default-storage-engine = innodb

innodb_file_per_table = on

innodb_flush_method = O_DIRECT

interactive_timeout = 1800

wait_timeout = 1800

innodb_open_files=1500

innodb_io_capacity=5000

innodb_buffer_pool_instances=4

innodb_autoextend_increment=64

innodb_purge_threads=1

innodb_purge_batch_size=150

innodb_write_io_threads=4

innodb_read_io_threads=4

innodb_concurrency_tickets=2500

explicit_defaults_for_timestamp = 1

skip-name-resolve

lower_case_table_names = 1

server-id = 1

skip-log-bin

sql_mode
STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION

[mysqldump]

quick

max_allowed_packet = 32M

EOF

4 创建 mysql 用户

```
/home/shuncom/mysql/bin/mysqld --initialize --user=mysql --basedir=/home/shuncom/mysql --datadir=/home/shuncom/mysql/data
```

5 创建 mysql 的 systemd 服务

```
cat > /lib/systemd/system/mysqld.service << EOF
```

[Unit]

Description=MySQL Server 8.0.20

Documentation=man:mysqld(8)

Documentation=<http://dev.mysql.com/doc/refman/en/using-systemd.html>

After=network.target

After=syslog.target

[Install]

WantedBy=multi-user.target

[Service]

User=mysql

Group=mysql

Type=forking

PIDFile=/home/shuncom/mysql/data/mysqld.pid

TimeoutSec=0

PermissionsStartOnly=true

ExecStartPre=/usr/local/mysql/bin/mysqld_pre_systemd

ExecStart=/home/shuncom/mysql/bin/mysqld --defaults-file=/etc/my.cnf --daemonize --pid-file=/home/shuncom/mysql/data/mysqld.pid \\${MYSQLD_OPTS}

EnvironmentFile=-/etc/sysconfig/mysql

LimitNOFILE = 65535

Restart=on-failure

RestartPreventExitStatus=1

PrivateTmp=false

EOF

6 6 mysql66

systemctl daemon-reload &&\

systemctl start mysqld.service &&\

systemctl enable mysqld.service &&\

systemctl status mysqld.service

cat >> ~/.bashrc << EOF

export PATH=/home/shuncom/mysql/bin:\\$PATH

EOF

source ~/.bashrc

```
su - shuncom
```

```
cat >> ~/.bashrc << EOF
```

```
export PATH=/home/shuncom/mysql/bin:\$PATH
```

```
EOF
```

```
source ~/.bashrc
```

```
exit
```

```
mysql --version
```

```
mysql Ver 8.0.20 for Linux on x86_64 (MySQL Community Server - GPL)
```

7 **mysql**

```
grep 'temporary password' /home/shuncom/mysql/data/mysql.log
```

```
2022-09-06T07:24:22.857237Z 6 [Note] [MY-010454] [Server] A temporary password is generated for root@localhost: x2z?tj79y,dI
```

```
mysql -uroot -p'tji?RNa1Ly9N'
```

```
ALTER USER 'root'@'localhost' IDENTIFIED BY 'Sz_clighting';
```

```
flush privileges;
```

```
exit;
```

4.1 mysql

1

```
mysql -u root -p'Sz_clighting'
```

```
CREATE DATABASE rulr_things DEFAULT CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci;
```

```
create user "shuncom"@"%" identified by "sh_clighting";
```

```
grant all on rulr_things.* to "shuncom"@"%";
```

exit

[illegible]

```
mkdir -p /home/shuncom/update_mysql/ &&\
```

```
chmod +x /home/shuncom/update_mysql/update_mysql.sh
```

2

rulr-db-migrate_vx.x.x.jar□□□□

(2) $\square\square$ jdk

```
tar -xzf jdk-8u251-linux-x64.tar.gz &&\
```

```
mv jdk1.8.0_251/ /usr/local/jdk/
```

```
cat >> ~/.bashrc << EOF
```

```
export JAVA_HOME=/usr/local/jdk
```

```
export CLASSPATH=\\$JAVA_HOME\\lib\\tools.jar;\\$JAVA_HOME\\jre\\lib\\rt.jar
```

```
export PATH=%JAVA_HOME%\bin;%PATH%
```

EOF

```
source ~/.bashrc
```

su - shuncom

```
cat >> ~/.bashrc << EOF
```

```
export JAVA_HOME=/usr/local/jdk
```

```
export CLASSPATH=\$JAVA_HOME/lib/tools.jar:\$JAVA_HOME/jre/lib/rt.jar
```

```
export PATH=%JAVA_HOME%\bin;%PATH%
```

EOF

```
source ~/.bashrc
```

```
exit
```

(4) 安装 jdk

```
java -version
```

```
java version "1.8.0_251"
```

Mysql 配置 max_binlog_cache_size

```
mysql -u root -p'Sz_clighting'
```

```
show variables like "%max_binlog%";
```

```
3 max_binlog_cache_size < 33554432
```

```
set global max_binlog_cache_size=33554432;
```

```
show variables like "%max_binlog%";
```

4 配置 max_binlog_cache_size

```
mysql -u root -p'Sz_clighting'
```

```
set global max_binlog_cache_size=8388608;
```

```
show variables like "%max_binlog%";
```

1 SQL 数据库

1 数据库 sql

mysql -u shuncom -p'sh_clighting'

use rulr_things

source /home/shuncom/v3.11.11__aliyun_init_db_dump.sql

cd /home/shuncom/update_mysql

vi update_mysql.sh

数据库 v3.11.11 数据库 v3.11.11

1 jdk /usr/local/jdk/

1

./update_mysql.sh

mkdir -p log/shuncom-db-migrate

?????Mysql 数据库

1 数据库 log_bin 数据库 skip-log-bin

2 数据库 [mysqld]

a

vi /etc/my.cnf

server-id = 1

log-bin = /home/shuncom/mysql/data/mysql-bin

binlog_cache_size = 4M

binlog_format = mixed

max_binlog_cache_size = 8M

max_binlog_size = 512M

expire_logs_days = 14

gtid_mode = ON

enforce_gtid_consistency = ON

mysql

systemctl restart mysqld.service

b

vi /etc/my.cnf

server-id = 2

slave-skip-errors=1007,1022,1050,1062,1169

relay-log=/home/shuncom/mysql/data/relay-log

max_relay_log_size=512M

relay-log-purge=ON

read-only

gtid_mode = ON

enforce_gtid_consistency = ON

❏ mysql❏

systemctl restart mysqld.service

3❏❏❏❏❏❏❏

a❏ root❏❏❏❏

mysql -u root -p'Sz_clighting'

b❏❏❏❏❏❏❏❏❏❏❏❏❏❏❏❏❏❏

CREATE USER 'replicator'@'%' IDENTIFIED BY 'Passw0rd';

GRANT REPLICATION SLAVE ON *.* TO 'replicator'@'%';

FLUSH PRIVILEGES;

SHOW MASTER STATUS\G; -- ❏❏ MASTER_LOG_FILE❏ MASTER_LOG_POS

4❏❏❏❏❏❏❏

a❏ root❏❏❏❏❏❏❏

mysql -u root -p'Sz_clighting'

b❏❏❏❏❏

CHANGE MASTER TO

MASTER_HOST='172.17.20.17',

MASTER_PORT=3306,

MASTER_USER='replicator',

MASTER_PASSWORD='Passw0rd',

```
MASTER_LOG_FILE='mysql-bin.000001',
```

```
MASTER_LOG_POS=868,
```

```
get_master_public_key=1;
```

```
CHANGE MASTER TO MASTER_DELAY = 3600;
```

```
START SLAVE;
```

```
SHOW SLAVE STATUS\G;
```

Docker????????

docker[]

docker save -o xampp_7.tar ef5954c4d0d1

#####

docker load -i xampp_7.tar

####

docker tag c1b6ba59135e amazoncorretto:8u352-alpine3.16-tini

####

docker commit <container_name_or_id> <new_image_name>:<tag>

1. []

docker-compose down

2. []

[] docker-compose []

docker-compose config | awk '/image:/{print \$2}' | xargs -I {} docker save -o {}.tar {}

[] docker-compose.yml [] .tar []

3. []

#####

#####

docker volume ls

my_volume[]

docker run --rm -v my_volume:/volume -v \$(pwd):/backup busybox tar czf /backup/my_volume.tar.gz -C /volume .

4. [] tar.gz

tar.gz []

tar czf docker_backup.tar.gz *.tar *.tar.gz

#####

[] docker_backup.tar.gz [] scp []

scp docker_backup.tar.gz user@target_server:/path/to/destination

#####

1. ❶ tar.gz ❷
❸❹❺❻❼❽❿❾ docker_backup.tar.gz ❶❷

tar xzf docker_backup.tar.gz

2. ❶❷❸❹
❶❷❸❹ Docker❶❷❸

for image in *.tar; do docker load -i \$image; done

docker load -i mongodb:24.8.9.1.tar &&\
docker load -i mysql:24.8.2.tar &&\
docker load -i rabbitmq:24.8.2.tar &&\
docker load -i redis:3.2.2.tar &&\
docker load -i szdriver:24.8.9.7.tar &&\
docker load -i javaapp:24.8.9.2.tar

3. ❶❷❸❹❺❻❼❽❿❾❶❷❸❹❺❻

❶❷❸❹❺
docker run --rm -v my_volume:/volume -v \$(pwd):/backup busybox tar xzf
/backup/my_volume.tar.gz -C /volume

4. ❶❷❸❹
❶❷ docker-compose.yml ❶❷❸❹❺❻❼❽❿❾❶❷❸❹❺❻

docker-compose up -d

????????????rulr-v4gw-adapter

```
1##### Rulr[] _New_5 (121.199.1.127) []
docker stop rulr-v4gw-adapter
docker save -o amazoncorretto.tar c1b6ba59135e
docker start rulr-v4gw-adapter
scp -P 53022 amazoncorretto.tar shuncomlog@shuncom.3322.org:/home/shuncomlog/
```

```
2#####
scp -P 53022 shuncomlog@shuncom.3322.org:/home/shuncomlog/amazoncorretto.tar ./
docker load -i amazoncorretto.tar
docker images #[] id
docker tag <[] id> amazoncorretto:8u352-alpine3.16-tini
```

```
cat >> /home/shuncom/compose/.env << EOF
JAVA_IMAGE=amazoncorretto:8u352-alpine3.16-tini
EOF
```

```
vi /home/shuncom/compose/docker-compose.yml
:set paste #[]
```

```
# []
rulr-v4gw-adapter:
  container_name: rulr-v4gw-adapter
  image: ${JAVA_IMAGE}
  volumes:
    - /home/shuncom/servers/v4gw-adapter:/rulr/app
    - /home/shuncom/config/servers:/home/shuncom/config/servers
    - /home/shuncom/log:/home/shuncom/log
  ports:
    - "9011:9011"
  working_dir: /rulr/app
  command: java -Xms256m -Xmx512m -XX:MaxDirectMemorySize=128m -jar v4gw-adapter.jar --
appMode=2
  restart: always
  networks:
    - rulr-network
  deploy:
    resources:
      limits:
        cpus: '1.4'
```

memory: 1024M

```
cd /home/shuncom/compose  
docker-compose up -d
```

??Ubuntu????????????

```
## Ubuntu
#####
##### Ubuntu#####

#####
#####
#####
#####
date
##
date#####
#####
#####
timedatectl
##
timedatectl##### UTC#####
##
Local time: Mon 2024-09-02 15:04:23 CSTUniversal time: Mon 2024-09-02 07:04:23 UTCRTC time:
Mon 2024-09-02 07:04:23Time zone: Asia/Shanghai (CST, +0800)
##
Local time#####
Universal time##### UTC##
RTC time#####
Time zone#####

#####
#####
#####
#####
timedatectl list-timezones
##
list-timezones#####
#####
##### Asia/Shanghai#####
sudo timedatectl set-timezone Asia/Shanghai
##
set-timezone Asia/Shanghai##### Asia/Shanghai
##### timedatectl#####

#####
```


[illegible]

timedatectl

☐ NTP☐☐☐☐☐☐

```
sudo systemctl status ntp
```

111

```
systemctl status ntp NTP
```

--	--	--	--	--	--	--	--

11

--	--	--	--

time_t [10] time_t [10] UTC [10]

[illegible]

NTP

NTP

11

Ubuntu

NTP

docker-compose ?????????

1. Ubuntu 23.04 安装 docker-compose
sudo apt-get install docker-compose

验证安装

1. 检查

docker-compose --version
docker-compose down

2. 创建

docker-compose.yml 文件

docker-compose config | awk '/image:/{print \$2}' | xargs -I {} docker save -o {}.tar {}

docker-compose.yml 文件 .tar 文件

3. 加载

docker load --input docker_image.tar
docker images

docker volume ls

创建 my_volume

docker run --rm -v my_volume:/volume -v \$(pwd):/backup busybox tar czf /backup/my_volume.tar.gz -C /volume .

4. 打包 tar.gz

打包 tar.gz 文件

tar czf docker_backup.tar.gz *.tar *.tar.gz

打包完成

docker_backup.tar.gz 文件 scp 到目标服务器

scp docker_backup.tar.gz user@target_server:/path/to/destination

验证

1. 检查 tar.gz 文件

docker_backup.tar.gz 文件

tar xzf docker_backup.tar.gz

2. 备份

备份 Docker 镜像

```
for image in *.tar; do docker load -i $image; done
```

3. 恢复备份

恢复

```
docker run --rm -v my_volume:/volume -v $(pwd):/backup busybox tar xzf
/backup/my_volume.tar.gz -C /volume
```

4. 部署

部署 docker-compose.yml 文件

```
docker-compose up -d
```

部署

部署

部署

```
docker-compose down
```

部署

```
docker-compose config | awk '/image:/{print $2}' | xargs -I {} docker save -o {}.tar {}
```

```
docker run --rm -v my_volume:/volume -v $(pwd):/backup busybox tar czf
/backup/my_volume.tar.gz -C /volume .
```

tar.gz

```
tar czf docker_backup.tar.gz *.tar *.tar.gz
```

部署

```
scp docker_backup.tar.gz user@target_server:/path/to/destination
```

部署

tar.gz 文件

```
tar xzf docker_backup.tar.gz
```

部署

```
for image in *.tar; do docker load -i $image; done
```

部署

```
docker run --rm -v my_volume:/volume -v $(pwd):/backup busybox tar xzf
/backup/my_volume.tar.gz -C /volume
```

||||

docker-compose up -d

|||||||||

docker-compose |||||||||||||||||||

??	???	??	??
https://mirror.ccs.tencentyun.com	???	??CDN	?????????????
https://xxx.mirror.aliyuncs.com	???	??CDN	????????????? https://help.aliyun.com/zh/acr/user-guide/accelerate-the-pulls-of-docker-official-images?spm=a2c4g.11186623.0.i7?

????

???????????????? registry mirror

Ubuntu

22.04+

Debian

10+

CentOS

8+

javascript

0

AI

```
sudo mkdir -p /etc/docker
sudo tee /etc/docker/daemon.json <<EOF
```

```
{
  "registry-mirrors": [
    "https://docker.xuanyuan.me"
  ]
}
EOF
sudo systemctl daemon-reload
sudo systemctl restart docker
```

??DockerHub ?????? docker.xuanyuan.me ???????????????

javascript 0

AI 

```
docker pull docker.xuanyuan.me/library/mysql:5.7
```

library library

library

Linux

javascript 0

11

AI

```
#!/bin/sh
cat <<-EOF > /etc/docker/daemon.json
{
  "registry-mirrors": [
    "https://docker.xuanyuan.me"
  ]
}
EOF
systemctl daemon-reload
systemctl restart docker
```

☐ Ubuntu 22.04
 ☐ Debian
 ☐ CentOS 8
 ☒ **Fedora**
☐ Arch Linux
 ☐ openSUSE
 ☐ Leap 42.1

macOS? Docker For Mac?

☐ macOS ☐ Docker For Mac ☐

1 docker Preferences

2 □ Daemon □ □ □ Registry mirrors □ □ □ □ □ □ □ □ □ □

javascript 0

--	--

AI

https://docker.xuanyuan.me

3 Apply & Restart

Windows?Docker For Windows?

Windows Docker For Windows

1. docker Docker Daemon json?

2. "registry-mirrors" https://docker.xuanyuan.me

3. Apply Docker

DockerHub	
657	https://dockerhub.icu?0913
	hub.rat.dev

DockerHub????	???????
	docker.wanpeng.top
?????? 250	https://doublezonline.cloud
?????? 76	https://docker.mrxn.net?1102 ??????
?????? 45	https://lynn520.xyz?1102 ??????
?????? 30	https://ginger20240704.asia?1102 ??????
DockerHub ?????? 102	https://docker.anyhub.us.kg?0831 ??????
Dockerhub?????? 70	https://docker.wget.at?1102 ??????
?????? 38	https://docker.aws19527.cn?0913 ??????
?????? 19	https://dislabaiot.xyz
Docker Proxy ???? 92????? 22?	https://dockerpull.com?1102 ?????? https://dockerproxy.cn?0913 ?????1102 ??????
Docker Hub Container Image Library 16	https://docker.fxxk.dedyn.io
docker-registry-mirrors 77: ?? Docker Hub, GitHub, Google, k8s, Quay, Microsoft ?????.	dhub.kubesre.xyz?0913 ??????

DockerHub????	???????
AtomHub ???????? 27??????????336??	https://atomhub.openatom.cn
DaoCloud ??? 90	https://docker.m.daocloud.io
???DockerHub????	
Docker????? 1???????????????	https://hub.uuuadc.top8?1102 ??????
Docker?????	https://docker.ckyl.me4?1102 ??????
?????? 3	https://docker.hpcloud.cloud 4?1102 ??????
	docker.1panel.live?1102 ??????
	https://dockerhub.jobcher.com 5?1102 ??????
	https://docker.chenby.cn 5?1102 ??????

████████████████ DockerHub████████████████

Docker Compose ????????????

Docker Compose ████████

Dcoker Compose ██████

██████ ████