# 基础环境准备工作：

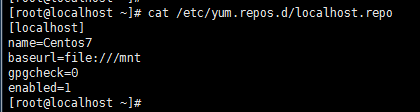
1、配置IP地址



root/root

2、mount /dev/sr0 /mnt #挂载光盘并安装本地源

连接光盘



4安装工具与阿里源

yum -y install vim wget lrzsz unzip zip tcl telnet zlib zlib-devel pcre pcre-devel gcc gcc-c++ openssl openssl-devel libevent libevent-devel perl net-tools

[root@235test nginx]# vim /etc/selinux/config #disabled(关闭) permissive(宽松模式) enforcing(强制模式)

[root@235test nginx]#setenforce 0 #0宽松模式 1强制模式

[root@235test nginx]# getenforce

[root@235test vhost]# systemctl stop firewalld

[root@235test vhost]# systemctl disable firewalld

下载阿里镜像

wget -O /etc/yum.repos.d/CentOS-Base.repo <http://mirrors.aliyun.com/repo/Centos-6.repo> #6专用

wget -O /etc/yum.repos.d/CentOS-Base.repo <http://mirrors.aliyun.com/repo/Centos-7.repo> #7专用

yum makecache

[root@localhost ~]# hostnamectl set-hostname 235test #永久修改主机名

# nginx 安装：

<http://nginx.org/download/> nginx下载地址,下载了ngixn-1.15.12.tar.gz

ngxin -V #先查看原nginx安装了哪些模块

安装依赖包：

[root@235test ~]# yum -y install gcc zlib zlib-devel pcre pcre-devel openssl openssl-devel

[root@235test ~]# pcre-config --version

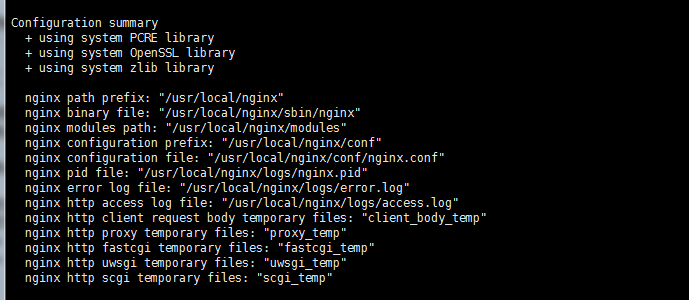
将安装包放入linux 系统进行编译安装

[root@235test ~]# tar -xf nginx-1.15.12.tar.gz

[root@235test ~]# cd nginx-1.15.12

[root@235test nginx-1.15.12]# ./configure --prefix=/usr/local/nginx --with-http\_stub\_status\_module --with-http\_flv\_module --with-http\_ssl\_module --with-http\_gzip\_static\_module --with-pcre --with-http\_realip\_module --with-http\_sub\_module --add-module=/opt/fastdfs-nginx-module-master/src

注意这里的坑，要想用fastdfs注意，先安装好



将原服务器的nginx配置文件拷贝到新服务器

[root@server1 conf]# scp nginx.conf root@192.168.10.235:/usr/local/nginx/conf/

[root@dev vhost]# scp test-yqtadmin.siyuanv.cn.conf root@192.168.10.235:/usr/local/nginx/conf/vhost/

创建原服务器目录结构：

[root@235test conf]# mkdir sslkey

[root@235test conf]# mkdir vhost

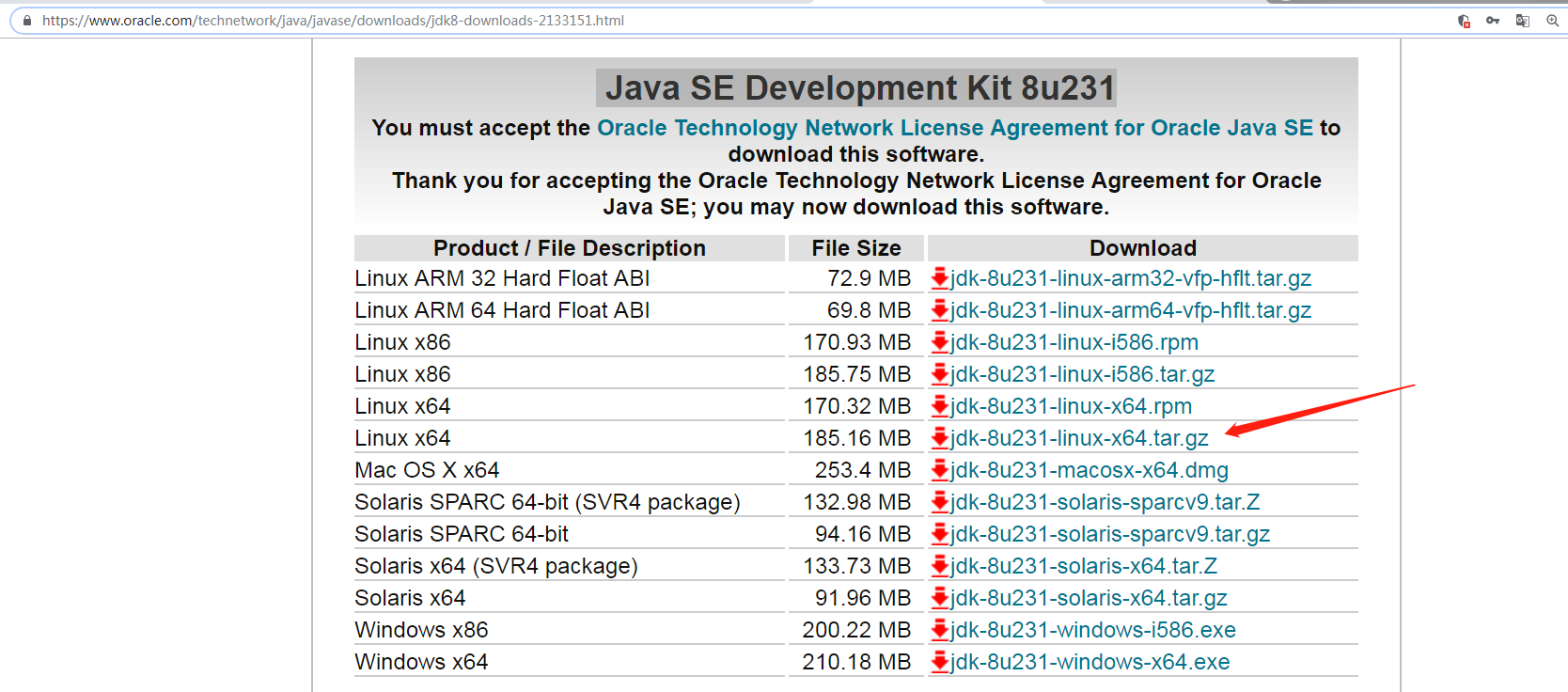
1放入元气推的前端目录：dist至/usr/local/nginx/yqt-admin目录中

2 C:\Windows\System32\drivers\etc windows系统的hosts文件目录。

# JAVA环境安装

下载地址：<https://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

需要登录oracle账户



下载后将包放入linux

[root@235test ~]# tar -xf jdk-8u231-linux-x64.tar.gz

[root@235test ~]# mv jdk1.8.0\_231 /usr/local/jdk1.8

[root@235test jdk1.8]# vim /etc/profile

JAVA\_HOME=/usr/local/jdk1.8

export PATH=$JAVA\_HOME/bin:$PATH

export CLASSPATH=$JAVA\_HOME/lib/tools.jar:$JAVA\_HOME/lib/dt.jar

[root@235test jdk1.8]# java -vsersion

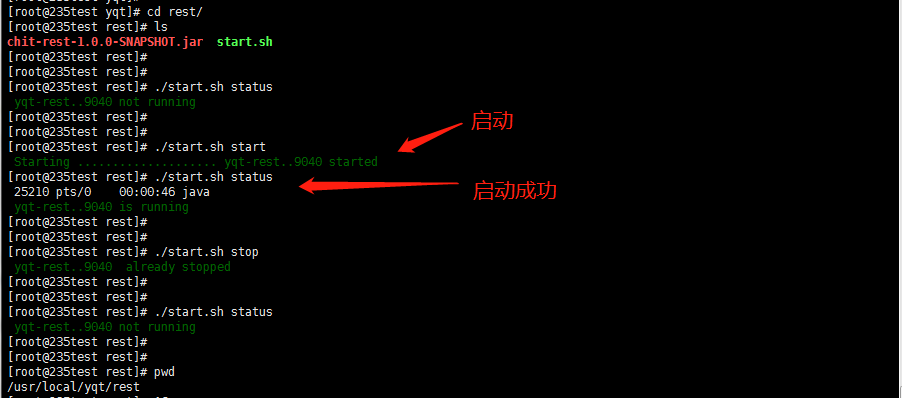
测试启动jar包是否成功

[root@235test nginx]# cd /usr/local/

[root@235test local]# mkdir -p yqt/rest

[root@localhost rest]# scp start.sh root@192.168.10.235:/usr/local/yqt/rest

[root@localhost rest]# scp chit-rest-1.0.0-SNAPSHOT.jar root@192.168.10.235:/usr/local/yqt/rest



# fastdfs安装

[root@db1 opt]# scp fastdfs-5.11.tar.gz root@192.168.10.235:/opt

[root@db1 opt]# scp fastdfs-nginx-module-master.zip root@192.168.10.235:/opt

[root@db1 opt]# scp libfastcommon-1.0.36.tar.gz root@192.168.10.235:/opt

## libfastcommon安装：

[root@235test opt]# tar -xf libfastcommon-1.0.36.tar.gz

[root@235test opt]# cd libfastcommon-1.0.36

[root@235test libfastcommon-1.0.36]# ./make.sh && ./make.sh install



## fastdfs-5.11安装

[root@235test opt]# tar -xf fastdfs-5.11.tar.gz

[root@235test opt]# cd fastdfs-5.11

[root@235test fastdfs-5.11]# ls

client conf fastdfs.spec init.d make.sh README.md stop.sh test

common COPYING-3\_0.txt HISTORY INSTALL php\_client restart.sh storage tracker

[root@235test fastdfs-5.11]#

目录说明：tracker（监控服务） test（测试服务） storage存储服务

[root@235test fastdfs-5.11]# mkdir /etc/fdfs

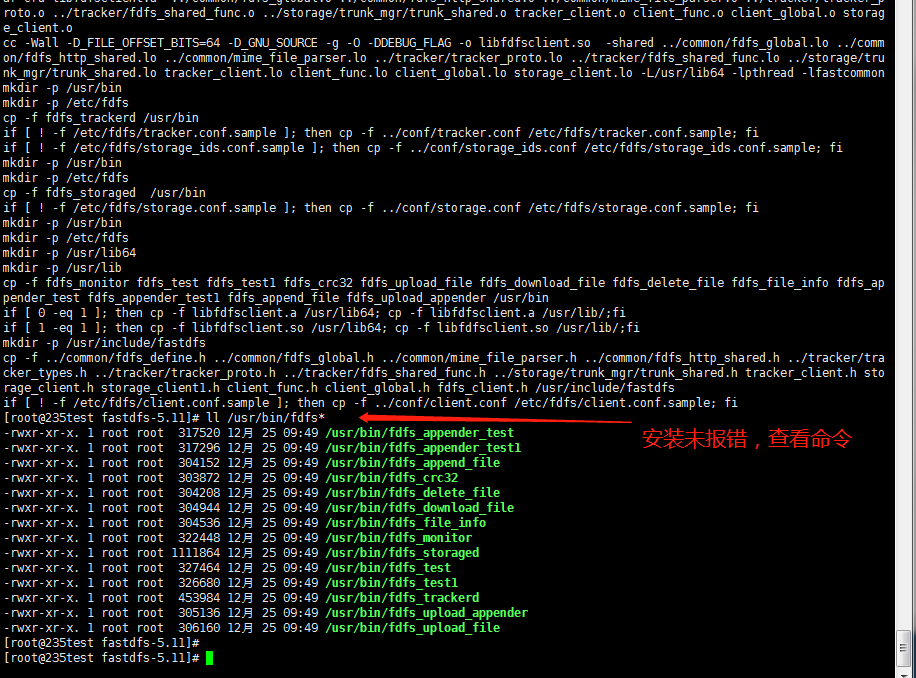
[root@235test fastdfs-5.11]# cp conf/mime.types /etc/fdfs

[root@235test fastdfs-5.11]# cp conf/http.conf /etc/fdfs

[root@235test fastdfs-5.11]# ls /etc/fdfs

http.conf mime.types

[root@235test fastdfs-5.11]# ./make.sh && ./make.sh install

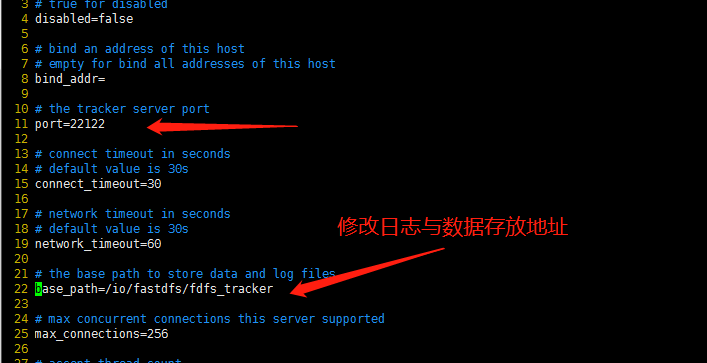


### 安装fstdfs的tracker监听服务：

[root@235test /]# cd /etc/fdfs/

[root@235test fdfs]# cp tracker.conf.sample tracker.conf

[root@235test fdfs]# vim tracker.conf



base\_path=/io/fastdfs/fdfs\_tracker

[root@235test fdfs]# mkdir -p /io/fastdfs/fdfs\_tracker

[root@235test fdfs]# fdfs\_trackerd /etc/fdfs/tracker.conf #启动fdfs\_trackerd监控服务

[root@235test fdfs]# pwd

/etc/fdfs

[root@235test fdfs]# ls /io/fastdfs/fdfs\_tracker/ #

data logs

[root@235test fdfs]# netstat -unltp |grep 22122

tcp 0 0 0.0.0.0:22122 0.0.0.0:\* LISTEN 31928/fdfs\_trackerd

[root@235test fdfs]#

[root@235test fdfs]# fdfs\_trackerd /etc/fdfs/tracker.conf stop #停止trackerd监听

waiting for pid [31928] exit ...

pid [31928] exit.

[root@235test fdfs]#

[root@235test fdfs]# netstat -unltp |grep 22122

[root@235test fdfs]#

[root@235test fdfs]# fdfs\_trackerd /etc/fdfs/tracker.conf start #启动trackerd监听

[root@235test fdfs]# netstat -unltp |grep 22122

tcp 0 0 0.0.0.0:22122 0.0.0.0:\* LISTEN 31953/fdfs\_trackerd

[root@235test fdfs]#

### 安装fstdfs的storage监听服务：

[root@235test fdfs]# pwd

/etc/fdfs

[root@235test fdfs]#

[root@235test fdfs]# cp storage

storage.conf.sample storage\_ids.conf.sample

[root@235test fdfs]#

[root@235test fdfs]# cp storage.conf.sample storage.conf

[root@235test fdfs]#

[root@235test fdfs]# vim storage.conf

41 base\_path=/io/fastdfs/fdfs\_storage/base #storage运行数据与日志

109 store\_path0=/io/fastdfs/fdfs\_storage/storage0 #资源存储目录

118 tracker\_server=192.168.10.235:22122

[root@235test fdfs]# mkdir -p /io/fastdfs/fdfs\_storage/base

[root@235test base]# mkdir /io/fastdfs/fdfs\_storage/storage0

[root@235test base]# fdfs\_storaged /etc/fdfs/storage.conf #初次启动storage服务

[root@235test base]#

[root@235test base]# fdfs\_storaged /etc/fdfs/storage.conf stop #停止storage服务

waiting for pid [32011] exit ...

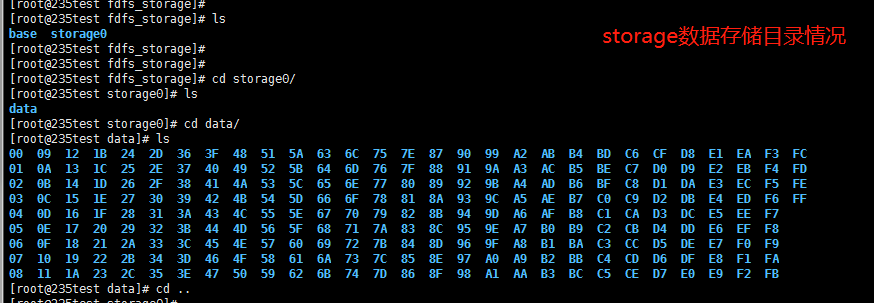
pid [32011] exit.

[root@235test base]# fdfs\_storaged /etc/fdfs/storage.conf start #启动storage服务

[root@235test base]#

[root@235test base]# netstat -unltp |grep 23000

tcp 0 0 0.0.0.0:23000 0.0.0.0:\* LISTEN 32027/fdfs\_storaged



### clinet应用

上面没问题之后，可以在本机上传图片进行测试图片服务器是否正常

[root@235test fdfs\_storage]# cd /etc/fdfs/

[root@235test fdfs]# vim client.conf

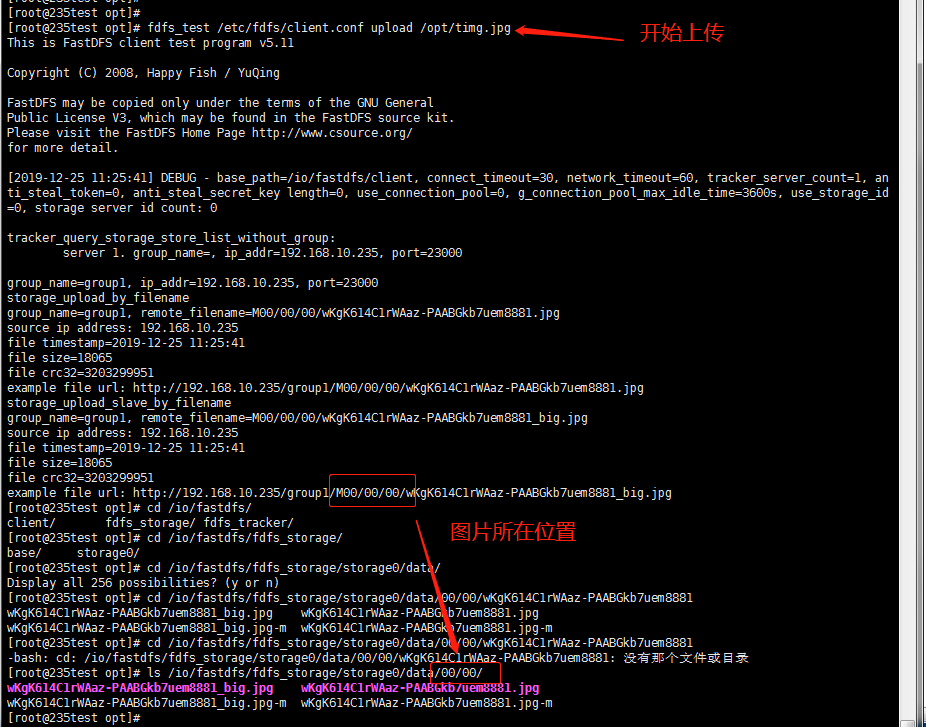
10 base\_path=/io/fastdfs/client

14 tracker\_server=192.168.10.235:22122

[root@235test fdfs]# mkdir -p /io/fastdfs/client

[root@235test opt]# rz -E #上传timg.jpg图片至opt目录

[root@235test opt]# fdfs\_test /etc/fdfs/client.conf upload /opt/timg.jpg #上传至服务器



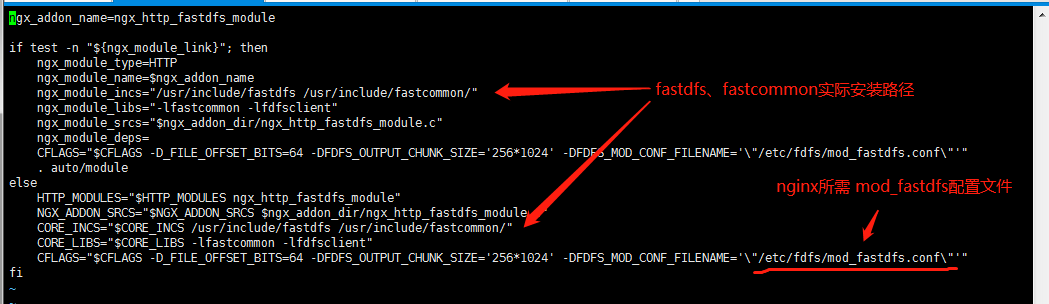
http://192.168.10.235/group1/M00/00/00/wKgK614C1rWAaz-PAABGkb7uem8881.jpg #图片路径，下面将用来测试

## fastdfs-nginx-module

[root@235test src]# pwd

/opt/fastdfs-nginx-module-master/src

[root@235test src]# vim config



[root@235test fdfs]# vim /etc/fdfs/mod\_fastdfs.conf

10 base\_path=/tmp #运行目录

40 tracker\_server=192.168.10.235:22122

53 url\_have\_group\_name = true

62 store\_path0=/io/fastdfs/fdfs\_storage/storage0

配置nginx

[root@235test vhost]# vim fastdfs.conf

server {

listen 80;

server\_name localhost;

location ~ /group1/M00 {

root /io/fastdfs/fastdfs\_storage/data;

ngx\_fastdfs\_module;

}

}

[root@235test vhost]# ../../sbin/nginx -t

[root@235test vhost]# ../../sbin/nginx -s reload

访问：http://192.168.10.235/group1/M00/00/00/wKgK614C1rWAaz-PAABGkb7uem8881.jpg 验证

总结：

1. nginx安装时定义了--add-module=/opt/fastdfs-nginx-module-master/src模块，
2. /opt/fastdfs-nginx-module-master/src/config 文件定义了fastcommon与fastdfs安装路径，并读取/etc/fdfs/mod\_fastdfs.conf文件
3. /etc/fdfs/mod\_fastdfs.conf文件配置了tracker\_server地址端口和存储地址

# mysql

下载地址：<https://dev.mysql.com/downloads/mysql/>

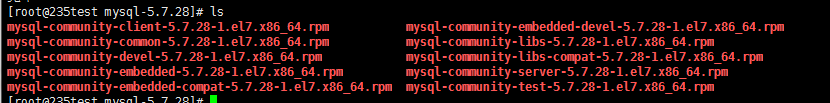
！！下载可直接使用wget命令更快，windows版也可

安装相关依赖：

[root@235test mysql-5.7.28]# yum -y install perl-Data-Dumper perl-JSON perl-Time-HiRes

[root@235test ~]# mkdir mysql-5.7.28

[root@235test ~]# tar -xf mysql-5.7.28-1.el7.x86\_64.rpm-bundle.tar -C mysql-5.7.28



主要安装如下四个包：

mysql-community-common-5.7.28-1.el7.x86\_64.rpm

mysql-community-libs-5.7.28-1.el7.x86\_64.rpm

mysql-community-client-5.7.28-1.el7.x86\_64.rpm

mysql-community-server-5.7.28-1.el7.x86\_64.rpm

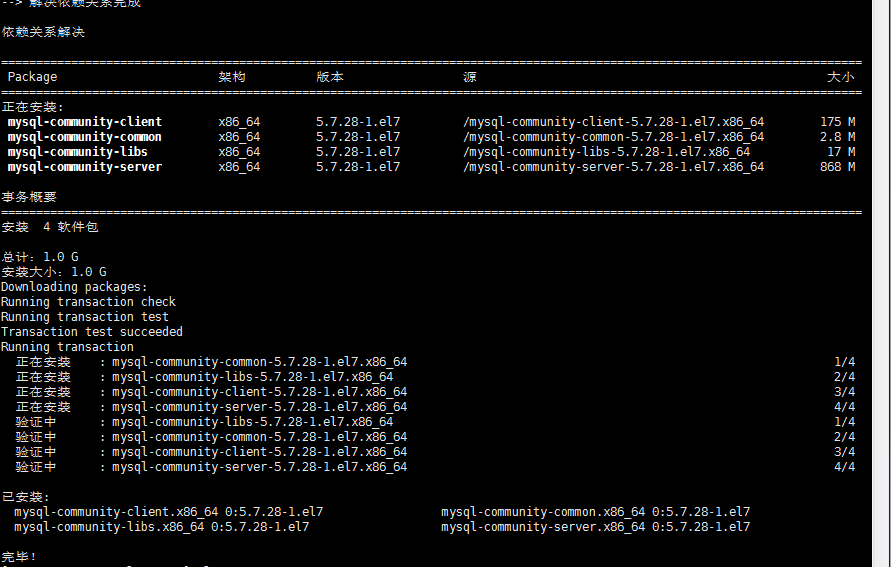
[root@235test mysql-5.7.28]# rpm -qa |grep -i mariadb

mariadb-libs-5.5.56-2.el7.x86\_64

[root@235test mysql-5.7.28]# yum remove -y mariadb-libs #卸载自带的mariadb

**安装mysql四个：**

[root@235test mysql-5.7.28]# yum -y install mysql-community-common-5.7.28-1.el7.x86\_64.rpm mysql-community-libs-5.7.28-1.el7.x86\_64.rpm mysql-community-client-5.7.28-1.el7.x86\_64.rpm mysql-community-server-5.7.28-1.el7.x86\_64.rpm



注：此时是用rpm -qa |greo mysql 无法查询到安装，用rpm -ivh 4个包名 安装一次即可

调试：

[root@235test mysql]# systemctl start mysql #启动正常后做如下操作

[root@235test mysql]# systemctl stop mysql

[root@235test mysql-5.7.28]# vim /etc/my.cnf #将db1的数据库配置文件拷贝到此修改

[root@235test mysql-5.7.28]# mkdir -p /io/mysql\_data/mysql

[root@235test mysql-5.7.28]# cd /io/mysql\_data/

[root@235test mysql\_data]# chown -R mysql.mysql /io/mysql\_data/mysql

[root@235test mysql\_data]# cd ..

[root@235test mysql]# cp -pr /var/lib/mysql /io/mysql\_data/ #拷贝默认数据库所有文件到指定数据库目录

[root@235test mysql-5.7.28]#mkdir -p /io/mysql\_data/binlog #创建binlog日志目录

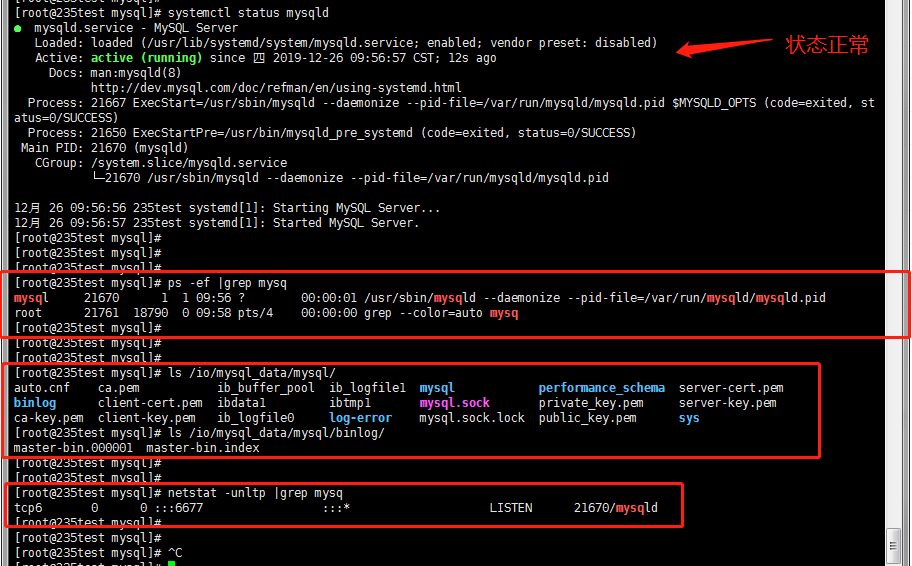
[root@235test mysql\_data]# chown mysql:mysql /io/mysql\_data/binlog

[root@235test mysql-5.7.28]#mkdir -p /io/mysql\_data/log-error #创建数据库错误日志目录

[root@235test mysql\_data]# chown mysql:mysql /io/mysql\_data/log-error

[root@235test mysql]# systemctl start mysql #再次启动

[root@235test mysql]# systemctl status mysqld



[root@235test ~]# vim /etc/my.cnf

[mysqld]

skip-grant-tables #在mysqld的配置下添加，跳过密码验证登录，仅限于首次登录，重启生效

[root@235test mysql]# mysql #登录

mysql>

mysql> show databases;

mysql>

[root@235test ~]# mysql

mysql> show grants for root@"%"; #配置文件中启用了-skip-grant-tables，所以无法使用本语句查询

ERROR 1290 (HY000): The MySQL server is running with the --skip-grant-tables option so it cannot execute this statement

重置root密码：

mysql> use mysql

mysql> select host,user,authentication\_string from user;

mysql> update mysql.user set authentication\_string=password("123456") where host="localhost" and user="root";

mysql> flush privileges;

mysql> show grants; #查看本用户的权限，不修改密码无法查询

ERROR 1820 (HY000): You must reset your password using ALTER USER statement before executing this statement.

mysql> show variables like "%password%" ; #查看密码要求，不修改密码无法查询

ERROR 1820 (HY000): You must reset your password using ALTER USER statement before executing this statement.

mysql> alter user root@"localhost" identified by "123456"; #密码不符合要求

ERROR 1819 (HY000): Your password does not satisfy the current policy requirements

mysql> alter user root@"localhost" identified by "Yuanqi@1688" ;

Query OK, 0 rows affected (0.02 sec)

mysql> show grants;

+---------------------------------------------------------------------+

| Grants for root@localhost |

+---------------------------------------------------------------------+

| GRANT ALL PRIVILEGES ON \*.\* TO 'root'@'localhost' WITH GRANT OPTION |

| GRANT PROXY ON ''@'' TO 'root'@'localhost' WITH GRANT OPTION |

+---------------------------------------------------------------------+

2 rows in set (0.00 sec)

# redis

redis 较为精简,源码安装，只要make 和make install就OK，不需要./config

[root@db1 opt]# scp redis-3.0.7.tar.gz [root@192.168.10.235:/opt/](mailto:root@192.168.10.235:/opt/)

[root@235test opt]# cd redis-3.0.7

[root@235test opt]#tar -xf redis-3.0.7.tar.gz

[root@235test redis-3.0.7]# ls

[root@235test redis-3.0.7]# make

[root@235test redis-3.0.7]# make install



[root@235test redis-3.0.7]# mkdir /etc/redis

[root@db1 redis]# scp redis.conf root@192.168.10.235:/etc/redis/

配置：

42 daemonize yes

107 logfile "/var/log/redis/redis\_6379.log"

191 dir "/io/redis"

[root@235test redis-3.0.7]# mkdir /io/redis

[root@235test redis-3.0.7]# mkdir /var/log/redis

[root@235test redis-3.0.7]# redis-server /etc/redis/redis.conf #指定配置文件启动

[root@235test redis-3.0.7]# ps -ef |grep redis

root 48746 1 0 18:07 ? 00:00:00 redis-server 0.0.0.0:6379

[root@235test redis-3.0.7]# redis-cli #登录

127.0.0.1:6379> ping #测试返回PONG则成功

PONG

127.0.0.1:6379>

127.0.0.1:6379> shutdown #关闭redis

not connected>

not connected> exit

[root@235test redis-3.0.7]# ps -ef |grep redis

root 49108 21171 0 18:15 pts/1 00:00:00 grep --color=auto redis

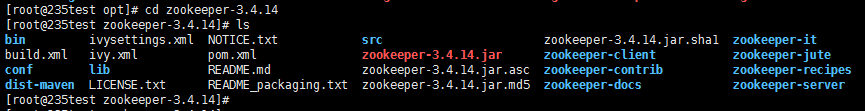
.

# zookeeper

<https://mirrors.tuna.tsinghua.edu.cn/apache/zookeeper/> 3.4.14

[root@235test opt]# tar -xf zookeeper-3.4.14.tar.gz

[root@235test opt]# cd zookeeper-3.4.14



[root@235test opt]# mv zookeeper-3.4.14 /usr/local/

[root@235test opt]# cd /usr/local/zookeeper-3.4.14/

[root@235test zookeeper-3.4.14]# mkdir data

[root@235test zookeeper-3.4.14]# mkdir logs

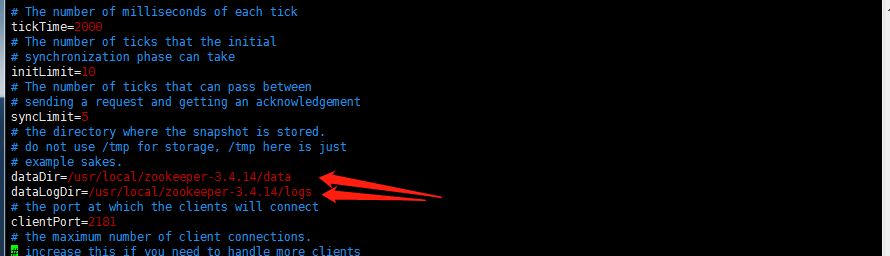
[root@235test zookeeper-3.4.14]# cd conf/

[root@235test conf]# ls

configuration.xsl log4j.properties zoo\_sample.cfg

[root@235test conf]# cp zoo\_sample.cfg zoo.cfg

[root@235test conf]# vim zoo.cfg



[root@235test zookeeper-3.4.14]# ./bin/zkServer.sh start conf/zoo.cfg

ZooKeeper JMX enabled by default

Using config: conf/zoo.cfg

Starting zookeeper ... STARTED

[root@235test zookeeper-3.4.14]#

[root@235test zookeeper-3.4.14]# ./bin/zkServer.sh status

ZooKeeper JMX enabled by default

Using config: /usr/local/zookeeper-3.4.14/bin/../conf/zoo.cfg

Mode: standalone #模式为独立模式

[root@235test data]# ps -ef |grep zookeeper

[root@235test data]# netstat -untlp |grep 2181

tcp6 0 0 :::2181 :::\* LISTEN 100924/java

[root@235test data]#

# Mongodb

<https://www.mongodb.com/download-center/community>

windows下载太慢，直接wget 下载

版本号为偶数的是稳定版，奇数是开发版

[root@db1 ~]# mongo --version #db1在用的事3.4.30版

MongoDB shell version v3.4.10

[root@235test opt]# wget https://fastdl.mongodb.org/linux/mongodb-linux-x86\_64-rhel70-3.6.16.tgz

[root@235test opt]# tar -xf mongodb-linux-x86\_64-rhel70-3.6.16.tgz

[root@235test opt]#mv mongodb-linux-x86\_64-rhel70-3.6.16 /usr/local/mongodb-3.6.16

[root@235test opt]# cd /io

[root@235test io]# mkdir mongodb

[root@235test io]# cd mongodb/

[root@235test mongodb]# mkdir db

[root@235test mongodb]# mkdir logs

[root@235test mongodb]# ls

[root@235test mongodb]# vim mongodb.conf

dbpath = /io/mongodb/db

logpath = /io/mongodb/logs/mongodb.log

logappend = true #启动时追加日志，默认是覆盖

pidfilepath = /var/run/mongo.pid

port = 27017

fork = true

[root@235test mongodb-3.6.16]# ./bin/mongod -f /io/mongodb/mongodb.conf #启动

about to fork child process, waiting until server is ready for connections.

forked process: 117522

child process started successfully, parent exiting

[root@235test mongodb-3.6.16]#

[root@235test mongodb-3.6.16]# netstat -unltp |grep 27017

tcp 0 0 127.0.0.1:27017 0.0.0.0:\* LISTEN 117522/./bin/mongod

[root@235test mongodb-3.6.16]#

[root@235test mongodb-3.6.16]# ps -ef |grep mongod

root 117522 1 10 17:41 ? 00:00:03 ./bin/mongod -f /io/mongodb/mongodb.conf

root 117574 18790 0 17:41 pts/4 00:00:00 grep --color=auto mongod

[root@235test mongodb-3.6.16]# ./bin/mongod -f /io/mongodb/mongodb.conf --shutdown #停止

killing process with pid: 117522

[root@235test mongodb-3.6.16]# ps -ef |grep mongod

[root@235test mongodb-3.6.16]# netstat -unltp |grep 27017

[root@235test opt]# /usr/local/mongodb-3.6.16/bin/mongo

> show dbs

admin 0.000GB

config 0.000GB

local 0.000GB

>

> exit

bye

# 新服务器磁盘处理：

## db1详情

[root@db1 ~]# df -h #挂载详情

文件系统 容量 已用 可用 已用% 挂载点

/dev/vda1 40G 6.8G 31G 19% /

devtmpfs 16G 0 16G 0% /dev

tmpfs 16G 0 16G 0% /dev/shm

tmpfs 16G 364K 16G 1% /run

tmpfs 16G 0 16G 0% /sys/fs/cgroup

/dev/vdb 880G 444G 393G 54% /io

tmpfs 3.2G 0 3.2G 0% /run/user/0

[root@db1 ~]# fdisk -l #磁盘详细情况

磁盘 /dev/vda：42.9 GB, 42949672960 字节，83886080 个扇区

Units = 扇区 of 1 \* 512 = 512 bytes

扇区大小(逻辑/物理)：512 字节 / 512 字节

I/O 大小(最小/最佳)：512 字节 / 512 字节

磁盘标签类型：dos

磁盘标识符：0x635e6c7d

设备 Boot Start End Blocks Id System

/dev/vda1 2048 83886079 41942016 83 Linux

磁盘 /dev/vdb：959.9 GB, 959925190656 字节，1874853888 个扇区

Units = 扇区 of 1 \* 512 = 512 bytes

扇区大小(逻辑/物理)：512 字节 / 512 字节

I/O 大小(最小/最佳)：512 字节 / 512 字节

[root@db1 ~]# file -s /dev/vdb #磁盘文件格式

/dev/vdb: Linux rev 1.0 ext4 filesystem data, UUID=c55a3e01-6aac-48f5-8f21-13999f861e1d (needs journal recovery) (extents) (64bit) (large files) (huge files)

[root@db1 ~]# lsblk #查看磁盘情况

NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT

sr0 11:0 1 1024M 0 rom

vda 253:0 0 40G 0 disk

└─vda1 253:1 0 40G 0 part /

vdb 253:16 0 894G 0 disk /io

[root@db1 ~]# parted -l

Model: Virtio Block Device (virtblk)

Disk /dev/vda: 42.9GB

Sector size (logical/physical): 512B/512B

Partition Table: msdos

Disk Flags:

Number Start End Size Type File system 标志

1 1049kB 42.9GB 42.9GB primary ext4

Model: Virtio Block Device (virtblk)

Disk /dev/vdb: 960GB

Sector size (logical/physical): 512B/512B

Partition Table: loop

Disk Flags:

Number Start End Size File system 标志

1 0.00B 960GB 960GB ext4

[root@db1 ~]# lsblk -f

NAME FSTYPE LABEL UUID MOUNTPOINT

sr0

vda

└─vda1 ext4 80b9b662-0a1d-4e84-b07b-c1bf19e72d97 /

vdb ext4 c55a3e01-6aac-48f5-8f21-13999f861e1d /io

[root@db1 ~]# vi /etc/fstab #开机自挂载vdb磁盘

# /etc/fstab

# Created by anaconda on Tue May 3 13:48:10 2016

# Accessible filesystems, by reference, are maintained under '/dev/disk'

# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info

#

UUID=80b9b662-0a1d-4e84-b07b-c1bf19e72d97 / ext4 defaults 1 1

/dev/vdb /io ext4 defaults 1 1

~

## yqt详情：

[root@yqt ~]# df -h

Filesystem Size Used Avail Use% Mounted on

/dev/vda1 50G 1.7G 45G 4% /

devtmpfs 7.8G 0 7.8G 0% /dev

tmpfs 7.8G 0 7.8G 0% /dev/shm

tmpfs 7.8G 492K 7.8G 1% /run

tmpfs 7.8G 0 7.8G 0% /sys/fs/cgroup

tmpfs 1.6G 0 1.6G 0% /run/user/0

[root@yqt ~]# fdisk -l

Disk /dev/vda: 53.7 GB, 53687091200 bytes, 104857600 sectors

Units = sectors of 1 \* 512 = 512 bytes

Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk label type: dos

Disk identifier: 0x000a57df

Device Boot Start End Blocks Id System

/dev/vda1 \* 2048 104855551 52426752 83 Linux

Disk /dev/vdb: 107.4 GB, 107374182400 bytes, 209715200 sectors

Units = sectors of 1 \* 512 = 512 bytes

Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes

[root@yqt ~]# file -s /dev/vdb #查看文件系统，未格式化之前

/dev/vdb: data

[root@yqt ~]# lsblk

NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT

vda 253:0 0 50G 0 disk

└─vda1 253:1 0 50G 0 part /

vdb 253:16 0 100G 0 disk

[root@yqt ~]# parted -l #查看磁盘，未格式化之前

Model: Virtio Block Device (virtblk)

Disk /dev/vda: 53.7GB

Sector size (logical/physical): 512B/512B

Partition Table: msdos

Disk Flags:

Number Start End Size Type File system Flags

1 1049kB 53.7GB 53.7GB primary ext4 boot

Error: /dev/vdb: unrecognised disk label

Model: Virtio Block Device (virtblk)

Disk /dev/vdb: 107GB

Sector size (logical/physical): 512B/512B

Partition Table: unknown

Disk Flags:

[root@yqt ~]# lsblk -f

NAME FSTYPE LABEL UUID MOUNTPOINT

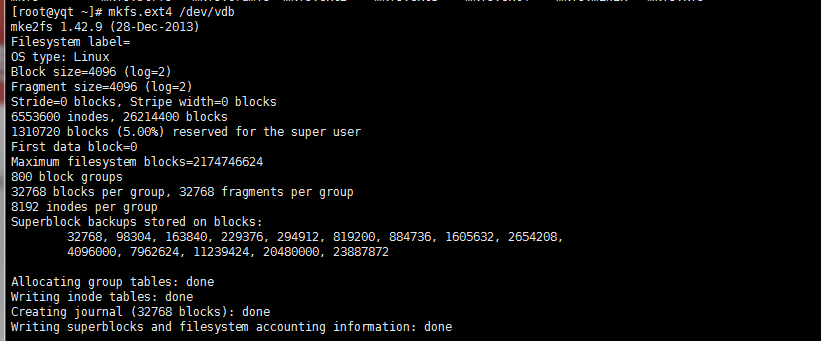
vda

└─vda1 ext4 b98386f1-e6a8-44e3-9ce1-a50e59d9a170 /

vdb

**新磁盘所需操作：**

[root@yqt ~]# mkfs.ext4 /dev/vdb



[root@yqt ~]# lsblk -f

NAME FSTYPE LABEL UUID MOUNTPOINT

vda

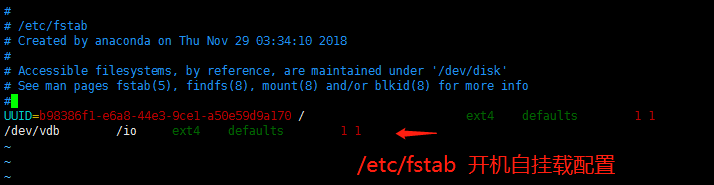
└─vda1 ext4 b98386f1-e6a8-44e3-9ce1-a50e59d9a170 /

vdb ext4 b8d72a47-b709-42c3-86c7-a5621166b8c8

[root@yqt ~]#

[root@yqt ~]# mount /dev/vdb /io

[root@yqt ~]# vim /etc/fstab



[root@yqt ~]# mount -a #检测fstab文件是否配置正确，无报错即为正确

[root@yqt ~]# df -h

Filesystem Size Used Avail Use% Mounted on

/dev/vda1 50G 1.7G 45G 4% /

devtmpfs 7.8G 0 7.8G 0% /dev

tmpfs 7.8G 0 7.8G 0% /dev/shm

tmpfs 7.8G 492K 7.8G 1% /run

tmpfs 7.8G 0 7.8G 0% /sys/fs/cgroup

tmpfs 1.6G 0 1.6G 0% /run/user/0

/dev/vdb 99G 61M 94G 1% /io

[root@yqt ~]# parted -l

Model: Virtio Block Device (virtblk)

Disk /dev/vda: 53.7GB

Sector size (logical/physical): 512B/512B

Partition Table: msdos

Disk Flags:

Number Start End Size Type File system Flags

1 1049kB 53.7GB 53.7GB primary ext4 boot

Model: Virtio Block Device (virtblk)

Disk /dev/vdb: 107GB

Sector size (logical/physical): 512B/512B

Partition Table: loop

Disk Flags:

Number Start End Size File system Flags

1 0.00B 107GB 107GB ext4

阿里云修改新服务器root密码：

