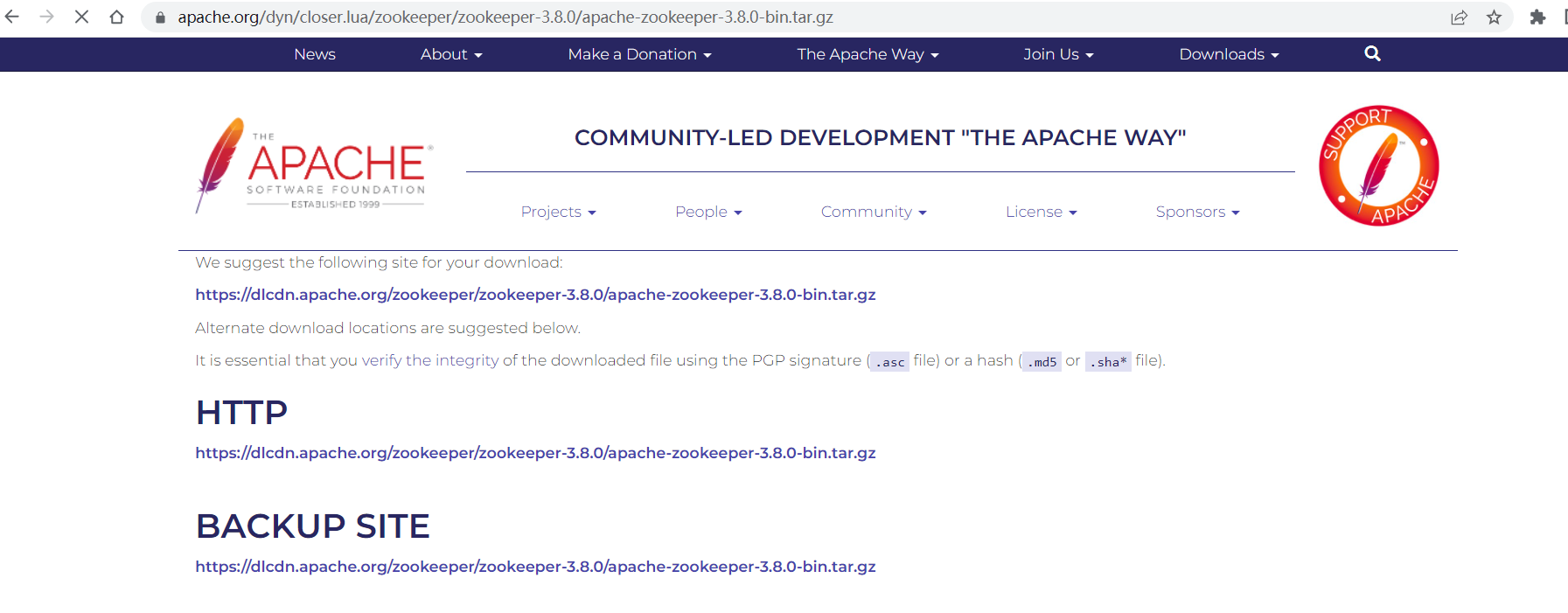
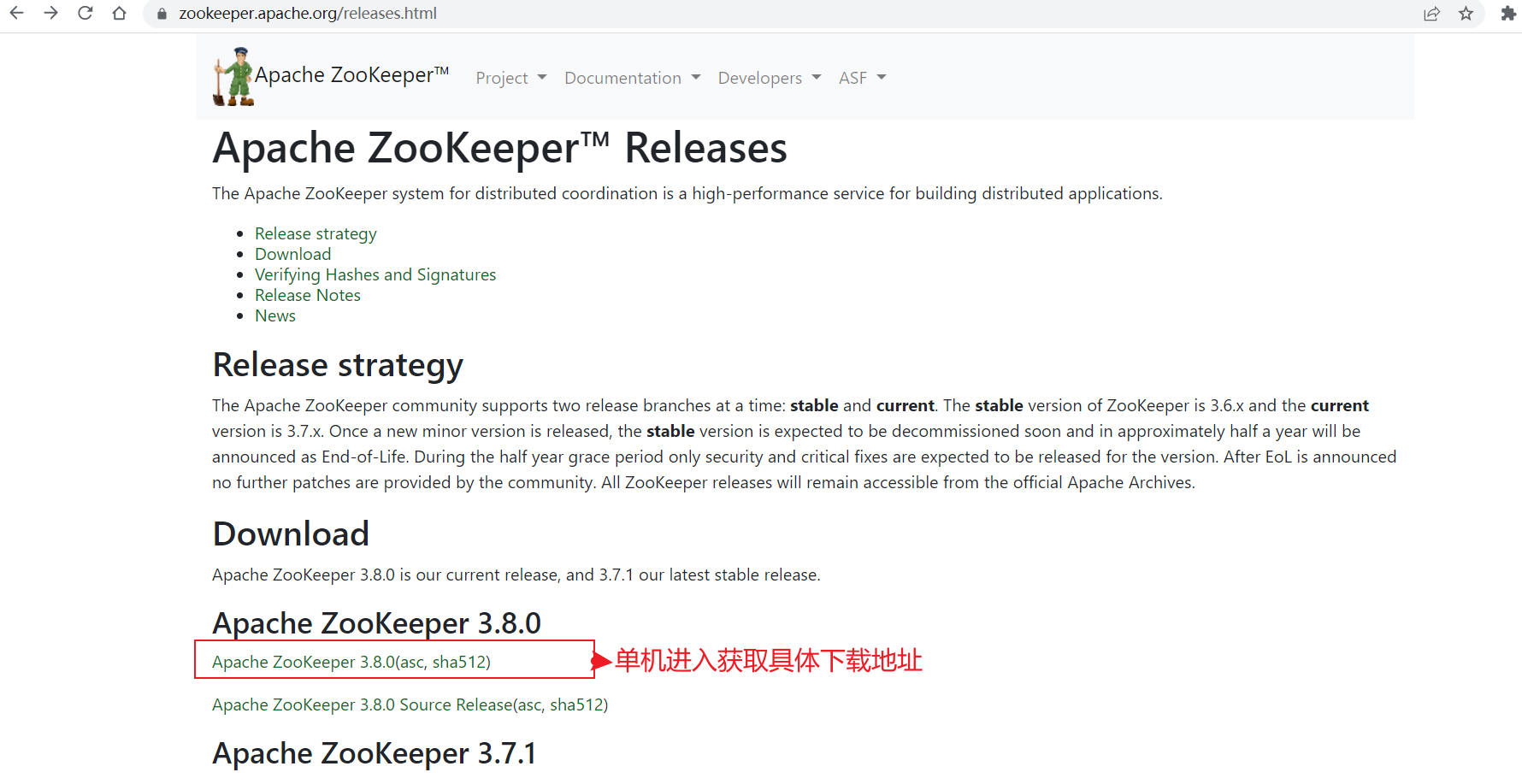
# **1 下载**

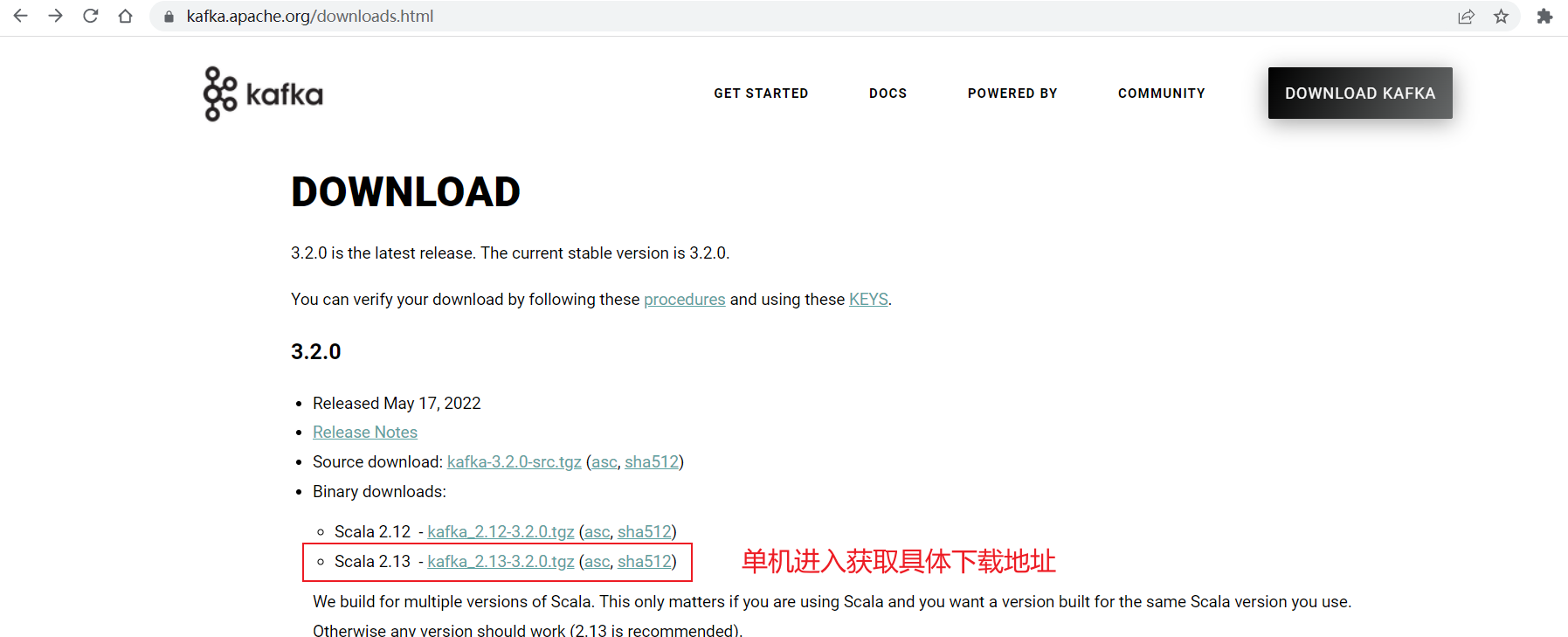
<https://zookeeper.apache.org/releases.html>

**<https://dlcdn.apache.org/zookeeper/zookeeper-3.8.0/apache-zookeeper-3.8.0-bin.tar.gz>**



<https://kafka.apache.org/downloads.html>

**<https://dlcdn.apache.org/kafka/3.2.0/kafka_2.13-3.2.0.tgz>**





# **2 部署**

**2.1 部署jdk1.8**

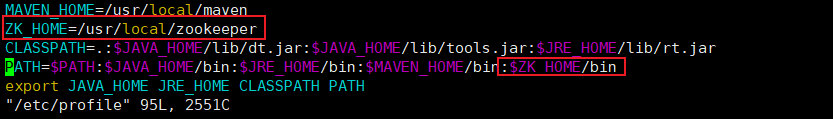
**2.2 部署zookeeoer**

#tar xf apache-zookeeper-3.8.0-bin.tar.gz -C /usr/local/

#cd /usr/local/ && ln -sv apache-zookeeper-3.8.0-bin zookeeper

#vim /etc/profile

ZK\_HOME=/usr/local/zookeeper



#source /etc/profile

|  |
| --- |
| sed -i.bak '$a export ZK\_HOME=/opt/zookeeper\nexport PATH=$PATH:$ZK\_HOME/bin' /etc/profile  source /etc/profile |

#mkdir -p /data/zookeeper

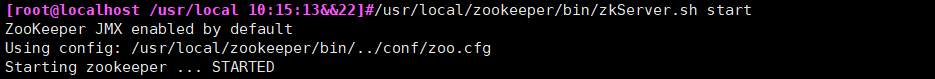
#cp /usr/local/zookeeper/conf/zoo\_sample.cfg /usr/local/zookeeper/conf/zoo.cfg

#vim /usr/local/zookeeper/conf/zoo.cfg

|  |
| --- |
| tickTime=2000  initLimit=10  syncLimit=5  dataDir=/data/zookeeper  clientPort=2181 |

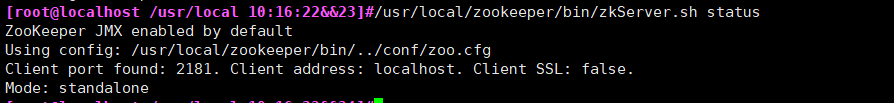
**2.3 启动zookeeper**

#/usr/local/zookeeper/bin/zkServer.sh start



**2.4 查看zookeeper状态**

#/usr/local/zookeeper/bin/zkServer.sh status



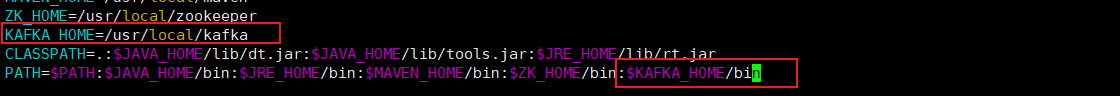
**2.5 部署kafka**

tar xf kafka\_2.13-3.1.0.tgz -C /usr/local

cd /usr/local/ && ln -sv kafka\_2.13-3.1.0 kafka

vim /etc/profile

KAFKA\_HOME=/usr/local/kafka



source /etc/profile

|  |
| --- |
| sed -i.bak '$a export KAFKA\_HOME=/opt/kafka\nexport PATH=$PATH:$KAFKA\_HOME/bin' /etc/profile  source /etc/profile |

mkdir -p /data/logs/kafka

vim /usr/local/kafka/config/server.properties

**阿里云服务器9092端口要对自身IP开放**

listeners=PLAINTEXT://172.16.8.19:9092 #阿里内网IP

advertised.listeners=PLAINTEXT://47.254.34.67:9092 #阿里外网IP

|  |
| --- |
| broker.id=0  listeners=PLAINTEXT://172.16.8.19:9092  advertised.listeners=PLAINTEXT://47.254.34.67:9092  num.network.threads=3  num.io.threads=8  socket.send.buffer.bytes=102400  socket.receive.buffer.bytes=102400  socket.request.max.bytes=104857600  log.dirs=/data/kafka/logs  num.partitions=1  num.recovery.threads.per.data.dir=1  offsets.topic.replication.factor=1  transaction.state.log.replication.factor=1  transaction.state.log.min.isr=1  log.retention.hours=168  log.segment.bytes=1073741824  log.retention.check.interval.ms=300000  zookeeper.connect=localhost:2181  zookeeper.connection.timeout.ms=18000  group.initial.rebalance.delay.ms=0 |
|  |

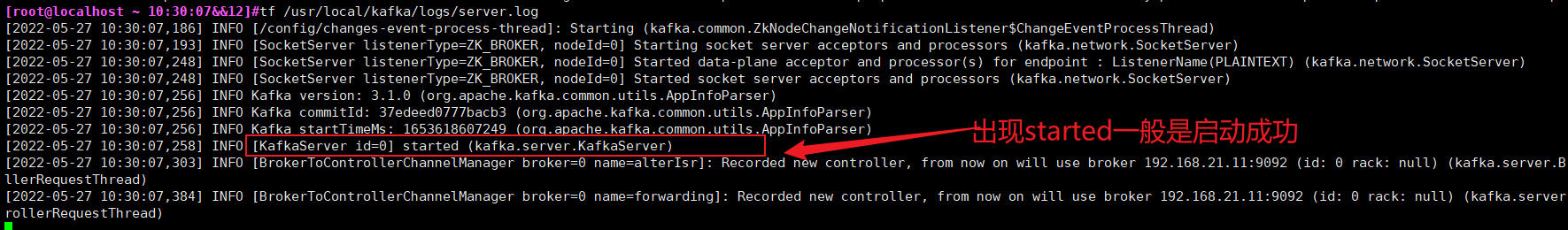
**2.6 启动kafka**

**前台启动**

/usr/local/kafka/bin/kafka-server-start.sh /usr/local/kafka/config/server.properties

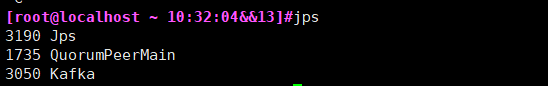
**后台启动**

/usr/local/kafka/bin/kafka-server-start.sh -daemon /usr/local/kafka/config/server.properties



**2.7验证进程**

jps



# **3 测试**

**3.1 测试创建topic**

创建名为kafkatest，partitions(分区)为3，replication(复制)为1的topic(主题),在任意机器操作即可

#kafka-topics.sh --create --bootstrap-server 192.168.21.11:9092 --replication-factor 1 --partitions 3 --topic kafkatest



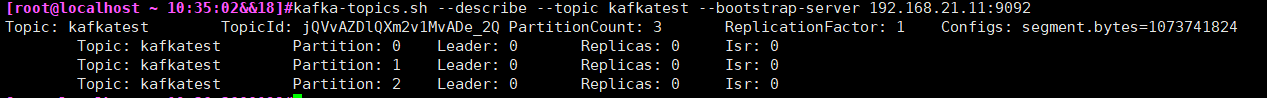
**3.2 获取所有的topic列表**

#kafka-topics.sh --list --bootstrap-server 192.168.21.11:9092



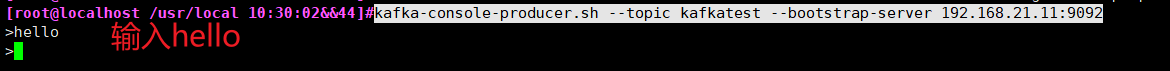
**3.3显示topic详细信息**

#kafka-topics.sh --describe --topic kafkatest --bootstrap-server 192.168.21.11:9092



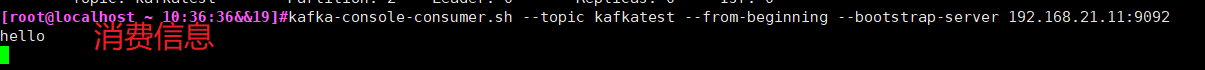
**3.4命令行发送消息**

kafka-console-producer.sh --topic kafkatest --bootstrap-server 192.168.21.11:9092



**3.5命令行消费消息**

kafka-console-consumer.sh --topic kafkatest --from-beginning --bootstrap-server 192.168.21.11:9092



**3.6删除topic**

kafka-topics.sh --delete --bootstrap-server 192.168.21.11:9092 --topic kafkatest

**3.7验证是否真的删除**

kafka-topics.sh --list --bootstrap-server 192.168.21.11:9092



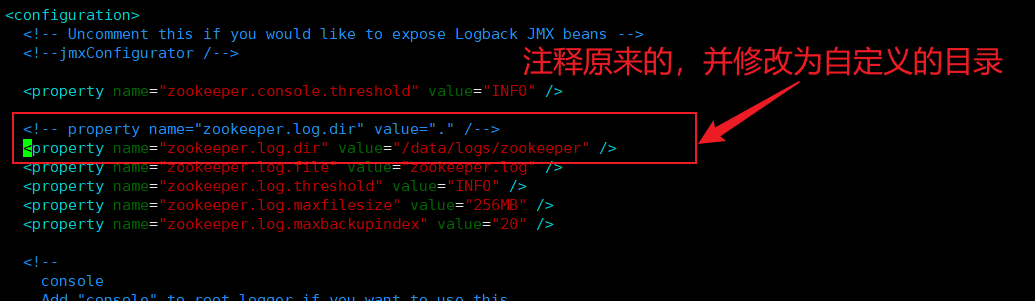
# **4 日志**

**4.1 修改zookeeper \*.out日志位置**

默认是在/usr/local/zookeeper/logs/



vim /usr/local/zookeeper/conf/logback.xml



vim /usr/local/zookeeper/bin/zkEnv.sh



重启zookeeper验证

/usr/local/zookeeper/bin/zkServer.sh restart



<https://wenku.baidu.com/view/d361884d26c52cc58bd63186bceb19e8b8f6ec11.html>