

Kafka使用指南

1 商品说明

Apache Kafka是一个分布式、支持分区 (partition)、多副本 (replica) 的，基于 Zookeeper协调的分布式消息系统，它由服务器和客户端组成，这些服务器和客户端通过高性能 TCP 网络协议进行通信。Kafka可以部署在本地和云中的裸机硬件、虚拟机和容器环境上，Kafka集群具有高度可扩展性和容错性。

本商品通过鲲鹏服务器+ EulerOS2.0进行安装部署。

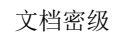
2 商品购买

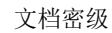
您可以在云商店搜索 “Kafka” 购买(推荐)，其中，地域、规格、推荐配置使用默认，购买方式根据您的需求选择按需/按月/按年，短期使用推荐按需，长期使用推荐按月/按年，确认配置后点击 “立即购买”。

下面以按需方式举例：

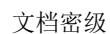
【自定义开通】和【模板配置开通】两种购买方式任选一种即可！

【自定义开通】方式：





第3页, 共10页



使用RFS模板直接部署

必填项填写后，点击 下一步



选择模板

参数配置

资源栈设置

配置确认

配置参数

请输入关键字搜索参数名称

按模板要求对部分资源加锁

参数名称	值	类型	描述
ECS实例名称		字符串	ECS实例的管理类名称。名称长度要求：名称要求长度为8到256位。名称至少包含一个大写字母、小写字母、数字和特殊字符（@%&*_~+!@!_!）。
系统盘大小	40	number	设置系统盘大小（至少40G，默认40G）。
数据盘大小	50	number	设置数据盘的大小。不需要数据盘，可设置值为0，可按照实际情况配置。默认值为50。
版本	请选择	字符串	选择版本
vpc IPv4网段	192.168.0.0/16	字符串	取值范围：19.0.0.0/8 to 19.255.255.0/24, 172.16.0.0/12 to 172.31.255.0/24, or 192.168.0.0/16 to 192.168.255.0/24
子网 IPv4网段	192.168.10.0/24	字符串	必须是CIDR格式，且在VPC的CIDR范围内。子网掩码不能大于24。
子网的网关	192.168.10.1	字符串	子网的网关。必须是子网范围内的合法IP地址
付费类型(不包含应用防火墙)	请选择	字符串	prePaid 预付费，即包年包月，postPaid 后付费，即按量付费
订购周期类型(不包含应用防火墙)	month	字符串	当chargingMode为prePaid时生效且为必填项。取值范围：month-月，year-年
订购周期数(不包含应用防火墙)	1	字符串	当chargingMode为prePaid时生效且为必填项。取值范围：periodType=month（周期类型为月）时，取值为1、6；periodType=year（周期类型为年）时，取值为1、3

上一步

下一步

IAM权限委托

hualicloud

请选择委托

如何创建委托?

选择权限委托可明确定义API权限在资源栈上可执行的所有操作（如创建、更新、删除资源栈等）。如果委托权限不足，有可能导致部署失败，生成执行计划操作失败。前往IAM创建委托

用户不配置权限委托系统采用当前登录用户的权限进行部署。权限委托有利于用户最小化授权对云资源的操作权限，可有效避免操作或者参数错误导致部署失败造成安全事故，推荐用户配置。

回滚设置

开启回滚设置当部署失败时，会自动回滚至系统所保留的上一个成功的资源状态。创建后可在资源栈详情页面修改。

删除保护

防止资源栈被意外删除。创建后可在资源栈详情页面修改。

上一步

下一步

创建直接计划后，点击 确定

立即创建资源栈

选择模板

参数配置

资源栈设置

配置确认

配置参数

参数名称	值	类型	描述
ECS实例名称	*****	字符串	名称长度为8到256位。名称至少包含一个大写字母、小写字母、数字和特殊字符（@%&*_~+!@!_!）中的三种！
系统盘大小	40	number	设置系统盘大小（至少40G，默认40G）。
数据盘大小	50	number	设置数据盘的大小。不需要数据盘，可设置值为0，可按照实际情况配置。默认值为50。
版本	v1.13.0	字符串	选择版本
vpc IPv4网段	192.168.0.0/16	字符串	取值范围：19.0.0.0/8 to 19.255.255.0/24, 172.16.0.0/12 to 172.31.255.0/24, or 192.168.0.0/16 to 192.168.255.0/24
子网 IPv4网段	192.168.10.0/24	字符串	必须是CIDR格式，且在VPC的CIDR范围内。子网掩码不能大于24。
子网的网关	192.168.10.1	字符串	子网的网关。必须是子网范围内的合法IP地址
付费类型(不包含应用防火墙)	postPaid	字符串	prePaid 预付费，即包年包月，postPaid 后付费，即按量付费
订购周期类型(不包含应用防火墙)	month	字符串	当chargingMode为prePaid时生效且为必填项。取值范围：month-月，year-年
订购周期数(不包含应用防火墙)	1	字符串	当chargingMode为prePaid时生效且为必填项。取值范围：periodType=month（周期类型为月）时，取值为1、6；periodType=year（周期类型为年）时，取值为1、3

上一步

创建执行计划

直接部署资源栈

资源栈设置

IAM权限委托

--

回滚

未开启

删除保护

未开启

操作须知：创建执行计划（免费）后可获取快速部署

创建执行计划

通过执行计划，可以动态地部署资源。

执行计划名称

executorPlan_20250324_1057_4eda

描述

请输入对执行计划的描述

8/255

确定

取消

点击 部署



基本信息 资源 输出 事件 模板 执行计划						
应用						
请输入关键字						
执行计划名称ID	状态	操作链接	创建时间	描述	操作	
executionPlan_20250324_1957_4e6a 18a83c49-7d29-4b69-b8ca-689e5c32e7	创建成功, 待部署	查看部署详情	2025/03/24 10:58:08 GMT+08:00	-	部署 部署	

如下图 “Apply required resource success.” 即为资源创建完成

基本信息 资源 输出 事件 模板 执行计划						
资源名称 请输入关键字						
事件时间	事件类型	事件描述	资源名称/类型	关联资源ID		
2025/03/24 11:00:06 GMT+08:00	日志	Apply required resource success.	-	-		
2025/03/24 11:00:01 GMT+08:00	生成完成	module.ecs.huaweicloud_compute_instance.ecs[0]: Creation complete after 57s [id=aa08d528-dc2e-4095-b388-fb333cecb44]	ecs ECS	aa08d528-dc2e-4095-b388-fb333cecb44		
2025/03/24 11:00:01 GMT+08:00	完成概要	Apply complete! Resources: 0 added, 0 changed, 0 destroyed.	-	-		
2025/03/24 10:59:54 GMT+08:00	正在生成	module.ecs.huaweicloud_compute_instance.ecs[0]: Still creating... [50s elapsed]	ecs ECS	-		
2025/03/24 10:59:44 GMT+08:00	正在生成	module.ecs.huaweicloud_compute_instance.ecs[0]: Still creating... [40s elapsed]	ecs ECS	-		
2025/03/24 10:59:34 GMT+08:00	正在生成	module.ecs.huaweicloud_compute_instance.ecs[0]: Still creating... [30s elapsed]	ecs ECS	-		
2025/03/24 10:59:24 GMT+08:00	正在生成	module.ecs.huaweicloud_compute_instance.ecs[0]: Still creating... [20s elapsed]	ecs ECS	-		
2025/03/24 10:59:14 GMT+08:00	正在生成	module.ecs.huaweicloud_compute_instance.ecs[0]: Still creating... [10s elapsed]	ecs ECS	-		
2025/03/24 10:59:04 GMT+08:00	生成完成	module.vpc.huaweicloud_vpc_subnet.subnet: Creation complete after 9s [id=4b3aceef-a475-4a7d-9e7c-000003b63763]	subnet Subnet	4b3aceef-a475-4a7d-9e7c-000003b63763		
2025/03/24 10:59:04 GMT+08:00	正在生成	module.ecs.huaweicloud_compute_instance.ecs[0]: Creating...	ecs ECS	-		

3 商品使用

3.1 准备工作

在使用ECS控制台配置前，需要您提前配置好安全组规则。

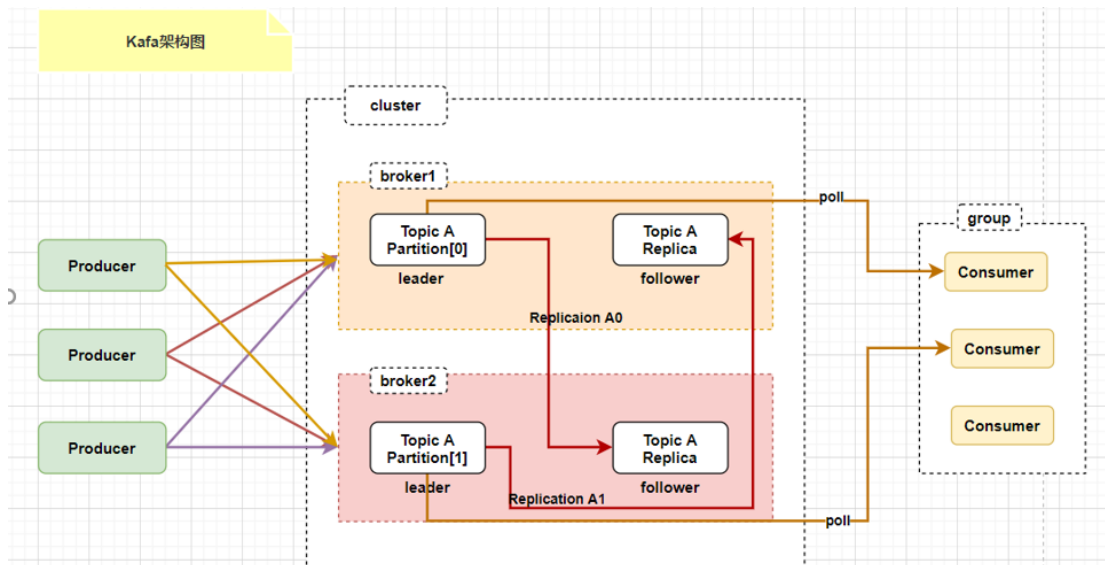
安全组规则的配置如下：

- 入方向规则放通端口见下面列表，源地址内必须包含您的客户端公网ip，否则无法访问
- 入方向规则放通CloudShell连接实例使用的端口22，以便在控制台登录调试。
- 出方向规则建议一键放通

端口说明：

组件	默认端口	说明
ZooKeeper	2181	Zookeeper 默认端口
ZooKeeper	2887、2888、2889	zk 集群之间进行数据/心跳同步的端口
ZooKeeper	3887、3888、3889	zk 集群之间进行领导者选举的端口
Kafka	9092	Kafka 的默认端口，用于接收和处理来自生产者和消费者的请求
SSH Client	22	SSH 远程访问主机端口

3.2 Kafka 的使用



架构图示例:

1. 修改 /etc/hosts 域名

将 192.168.0.103 hadoop1 改成自己本机的私有 ip x.x.x.x hadoop1

2. 执行/root 目录的 kafka.sh 命令，1.1.1.1 替换成自己使用的公网 ip

./kafka.sh 1.1.1.1

3、查看 jps

执行 kafka.sh 的命令有可能启动不了 zookeeper 或 kafka 服务，可以二次启动 kafka.sh 脚本

4、手工启动 zookeeper 服务和 kafka 命令（如果服务没启动，根据以下命令手工启动）

-- zookeeper 服务停止和启动（路径如有更改 换成本地实际使用路径）

```
/opt/software/zookeeper-3.5.7/bin/zkServer.sh stop
```

```
/opt/software/zookeeper-3.5.7/bin/zkServer.sh start /opt/software/zookeeper-3.5.7/conf/zoo.cfg #启动 zk
```

-- kafka 服务停止和启动

#停止 kafka 服务

```
/opt/software/kafka/bin/kafka-server-stop.sh
```

#后台启动kafka服务

```
/opt/software/kafka/bin/kafka-server-start.sh -daemon /opt/software/kafka/config/server.properties
```

其它kafka相关命令参考:

Topic (主题)

#创建 Topic

```
bin/kafka-topics.sh --create --bootstrap-server hadoop01:9092 --replication-factor 3 --partitions 2 --topic test
```

#查询所有 Topic 列表

```
bin/kafka-topics.sh --list --bootstrap-server hadoop01:9092
```

删除 Topic

```
bin/kafka-topics.sh --bootstrap-server hadoop01:9092 --delete --topic test
```

单个 Topic 扩容

```
bin/kafka-topics.sh --bootstrap-server hadoop01:9092 --alter --topic test --partitions 3
```

Producer (生产者)

发送消息

```
bin/kafka-console-producer.sh --broker-list hadoop01:9092 --topic test
```

```
[root@hadoop01 kafka]#  
[root@hadoop01 kafka]# bin/kafka-console-producer.sh --broker-list hadoop01:9092 --topic test  
>  
>This is a message  
>This is another message  
>
```

发送消息, 指定生产者参数 acks 为 -1, 同时启用 LZ4 的压缩算法:

```
bin/kafka-console-producer.sh --broker-list hadoop01:9092 --topic test --request-required-acks  
-1 --producer-property compression.type=lz4
```

Consumer (消费者)

消费消息 从头开始消费 (--from-beginning 参数表示从该主题最早的位移开始消费)

```
bin/kafka-console-consumer.sh --bootstrap-server hadoop01:9092 --topic test --from-  
beginning
```

```
[root@hadoop01 kafka]#  
[root@hadoop01 kafka]# bin/kafka-console-consumer.sh --bootstrap-server hadoop01:9092 --topic test --from-beginning  
  
I am a student.  
Hello, How are you?  
  
This is a message  
This is another message
```

指定消费者组

```
bin/kafka-console-consumer.sh --bootstrap-server hadoop01:9092 --topic test --from-  
beginning --group group01
```

从头开始消费 (--from-beginning 参数表示从该主题最早的位移开始消费)

```
bin/kafka-console-consumer.sh --bootstrap-server hadoop01:9092 --topic test --from-  
beginning --max-messages 100
```

说明: --max-messages 100 , 表示消费 100 条记录后, 停止消费。

Consumer_groups (消费者组)

查看消费者组的消费情况

```
bin/kafka-consumer-groups.sh --bootstrap-server hadoop01:9092 --describe --group group01
```

```
[root@hadoop01 kafka]#  
[root@hadoop01 kafka]# bin/kafka-consumer-groups.sh --bootstrap-server hadoop01:9092 --describe --group group01  
Consumer group 'group01' has no active members.  
  
GROUP      TOPIC      PARTITION  CURRENT-OFFSET  LOG-END-OFFSET  LAG      CONSUMER-ID     HOST     CLIENT-ID  
group01    test       0          0               0               0        -               -       -  
group01    test       1          3               3               0        -               -       -  
group01    test       2          3               3               0        -               -       -
```

查看所有消费者组提交的位移数据

对于 `__consumer_offsets` 而言，由于它保存了消费者组的位移数据，直接查看该主题消息是很方便的事情。下面的命令可以帮助我们直接查看消费者组提交的位移数据。

```
bin/kafka-console-consumer.sh --bootstrap-server hadoop01:9092 --topic __consumer_offsets -  
-formatter "kafka.coordinator.group.GroupMetadataManager$OffsetsMessageFormatter" --  
from-beginning
```

除了查看位移提交数据，我们还可以直接读取该主题消息，查看消费者组的状态信息。

```
bin/kafka-console-consumer.sh --bootstrap-server hadoop01:9092 --topic __consumer_offsets -  
-formatter  
"kafka.coordinator.group.GroupMetadataManager$GroupMetadataMessageFormatter" --from-  
beginning
```

3.3 参考文档

- [Kafka官网](#)