

阿里云 EMQX 服务镜像说明

杭州映云科技有限公司

2018/8

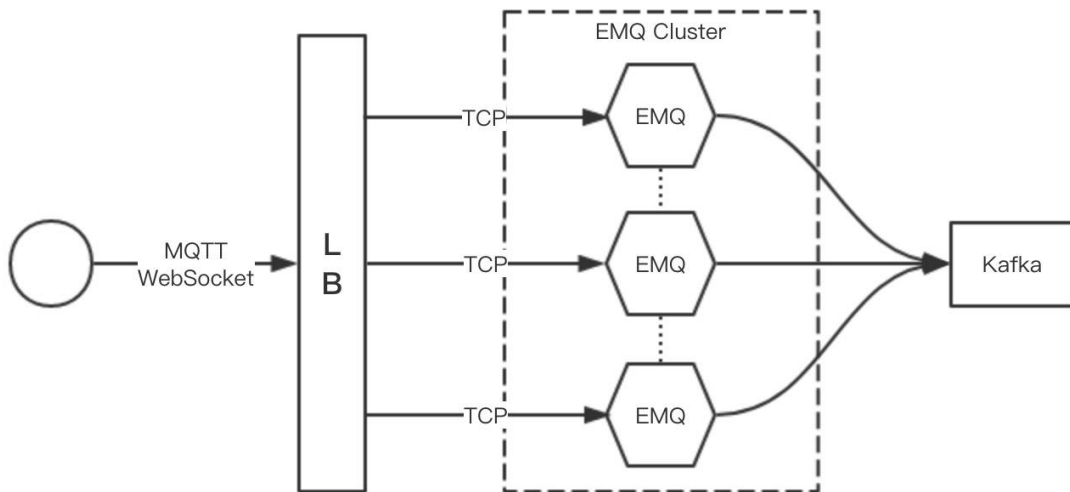
介绍.....	2
查看 EMQ 服务器状态.....	2
访问 EMQ 控制台.....	3
EMQX 插件服务配置.....	4
EMQ 安全组配置.....	10
验证测试.....	12
附录.....	13
EMQ 2.0 消息服务器默认占用的 TCP 端口.....	13
常用网址.....	13

介绍

EMQ IoT Hub 能够接入海量物联网设备，支持完整的 MQTT v3.1/3.1.1 协议栈，毫秒级消息时延，分布式集群能力，可通过 JWT 进行设备安全认证，并可持久化消息至 Kafka。

本产品已被全球 5000 家企业部署使用，包括数十家世界 500 强企业。

本版本能够支持的最大并发连接数量为 1 千线，如有其他容量需求，请联系 EMQ (www.emqtt.com)。部署架构如图：



查看 EMQ 服务器状态

缺省虚拟机启动之后，EMQTT 服务已经启动，如果用户可以通过远程登录到该虚拟机的话，可以通过下面的命令来查看 EMQ 的运行状态。

```
ssh root@$eip
```

```
ps -ef|grep emqx
```

可看到以下信息

```

[root@izm5e6i3z9nqfwbtiziw7lz ~]# ps -ef|grep emqx
root      1260      1  0 10:12 ?          00:00:00 /opt/emqx/erts-9.0/bin/epmd -dae
mon
root      1278      1  0 10:12 ?          00:00:00 /opt/emqx/erts-9.0/bin/run_erl -
daemon /data/root_erl_pipes/emqx@127.0.0.1/ /opt/emqx/log exec "/opt/emqx/bin/em
qx" "console"
root      1283    1278  0 10:12 pts/0      00:00:01 /opt/emqx/erts-9.0/bin/beam.smp
-W w -e 256000 -Q 256000 -P 256000 -A 32 -K true -zdbbl 8192 -- -root /opt/emqx
-progname opt/emqx/bin/emqx -- -home /root -- -boot /opt/emqx/releases/2.4.2/emq
x -mode embedded -boot_var ERTS_LIB_DIR /opt/emqx/erts-9.0/./lib -mnesia dir "d
ata/mnesia/emqx@127.0.0.1" -config /opt/emqx/data/configs/app.2018.08.10.10.12.1
6.config -kernel net_ticktime 60 -smp auto -setcookie emqxsecretcookie -proto_di
st inet_tcp -name emqx@127.0.0.1 -vm_args /opt/emqx/data/configs/vm.2018.08.10.1
0.12.16.args -- console
root      1712    1625  0 10:12 ?          00:00:00 /opt/emqx/lib/os_mon-2.4.2/priv/
bin/memsup
root      1713    1625  0 10:12 ?          00:00:00 /opt/emqx/lib/os_mon-2.4.2/priv/
bin/cpu_sup
root      1822    1805  0 10:14 pts/1      00:00:00 grep --color=auto emqx

```

访问 EMQ 控制台

EMQ 消息服务器启动后，会默认加载 Dashboard 插件，启动 Web 管理控制台。用户可通过 Web 控制台，查看服务器运行状态、统计数据、客户端(Client)、会话(Session)、主题(Topic)、订阅(Subscription)、插件(Plugin)。

控制台地址: <http://ECS公网IP:18083>，默认用户: admin，密码: public。请
注意: 访问控制台地址之前, 需要先按照下一章所示配置好阿里云的安全组配置, 该步骤的目的就是允许从本地访问远程部署好的 EMQ 服务。

EMQ Dashboard
控制台
emqx@192.168.0.2

监控

控制台

连接

会话

主题

订阅

管理

服务

监听器

工具

Websocket

HTTP接口

应用

应用

系统

用户

设置

系统信息

名称	版本	运行时间	系统时间
EMQ X Broker	2.4.2	1 hours, 11 minutes, 17 seconds	2018-06-07 02:56:59

节点信息(3)

节点名称	Erlang/OTP 版本	Erlang进程 (used/available)	CPU (1load/5load/15load)	内存 (used/total)	最大文件句柄数	状态
emqx@192.168.0.2	R20/9.1.5	362 / 2097152	0.12 / 0.09 / 0.02	89.62M / 118.75M	524288	Running
emqx@192.168.0.3	R20/9.1.5	362 / 2097152	0.00 / 0.02 / 0.02	89.59M / 118.75M	524288	Running
emqx@192.168.0.4	R20/9.1.5	362 / 2097152	0.00 / 0.00 / 0.00	89.58M / 118.75M	524288	Running

运行统计(3)

节点名称	连接数	连接峰值	主题数	主题峰值	保留消息数	最大保留消息数	会话数
emqx@192.168.0.2	0	0	0	0	7	7	0
emqx@192.168.0.3	0	0	0	0	7	7	0
emqx@192.168.0.4	0	0	0	0	7	7	0

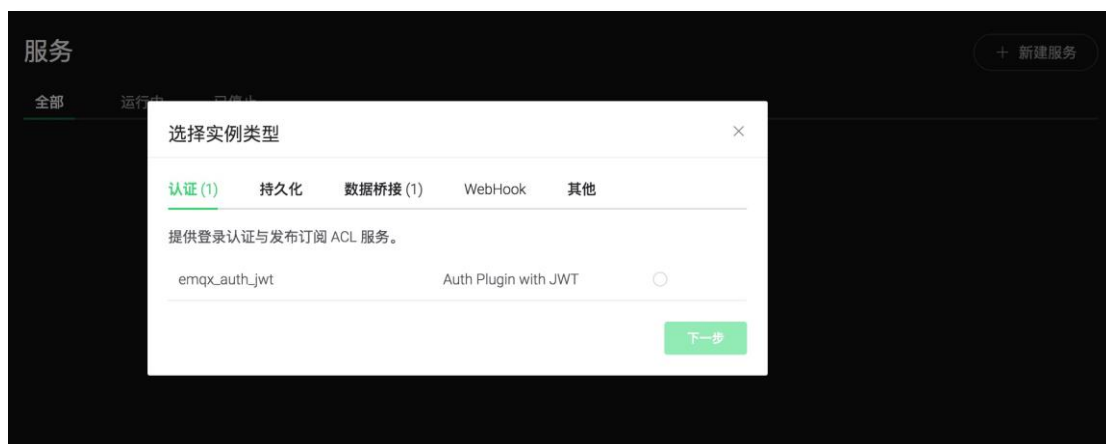
度量指标

MQTT报文		消息(数)		流量收发统计(字节)	
received	0	received	0	received	0
sent	0	sent	0	sent	0
connect	0	dropped	0		
connack	0	retained	7		
disconnect	0	qos0/received	0		
pingreq	0	qos0/sent	0		
pingresp	0	qos1/received	0		
publish/received	0	qos1/sent	0		
publish/sent	0	qos2/received	0		
puback/received	0	qos2/sent	0		
puback/sent	0	qos2/dropped	0		
puback/misssed	0				
pubcomp/received	0				
pubcomp/sent	0				
pubcomp/misssed	0				
pubrec/received	0				
pubrec/sent	0				
pubrec/misssed	0				
pubrel/received	0				
pubrel/sent	0				
pubrel/misssed	0				
subscribe	0				
suback	0				
unsubscribe	0				
unsuback	0				

EMQX 插件服务配置

Dashboard 可对 EMQX 插件内置服务进行管理 Dashboard -> 管理 -> 服务。EMQX 将提供认证、持久化、数据桥接、WebHook 等类型的服务，目前认证

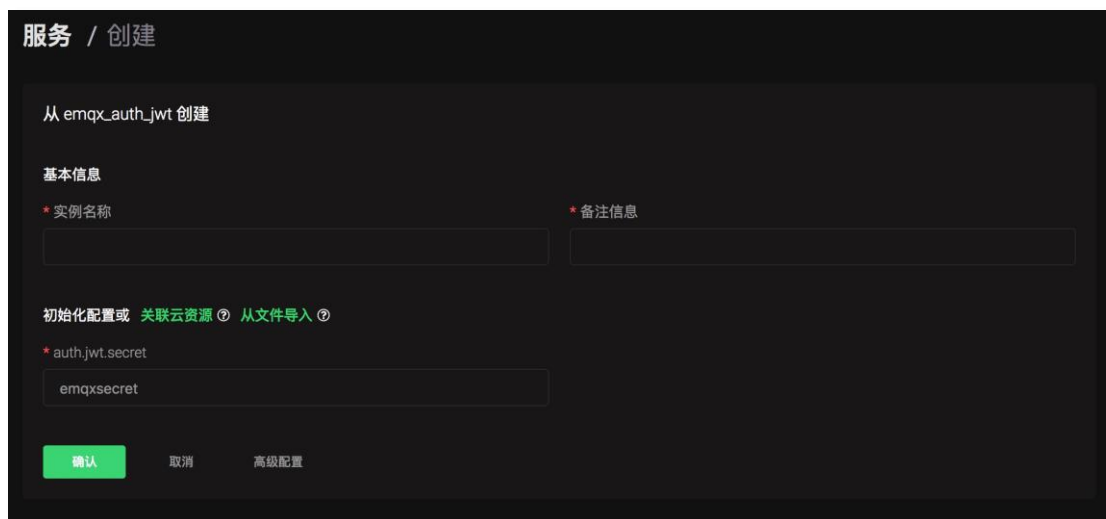
类型支持 emqx_auth_jwt、数据桥接支持 emqx_bridge_kafka，未来将陆续上线更多服务：



emqx_auth_jwt 服务

emqx_auth_jwt 认证插件支持的加密算法：HS256、HS384、HS512、RS256、RS384、RS512、ES256、ES384、ES512。

服务创建

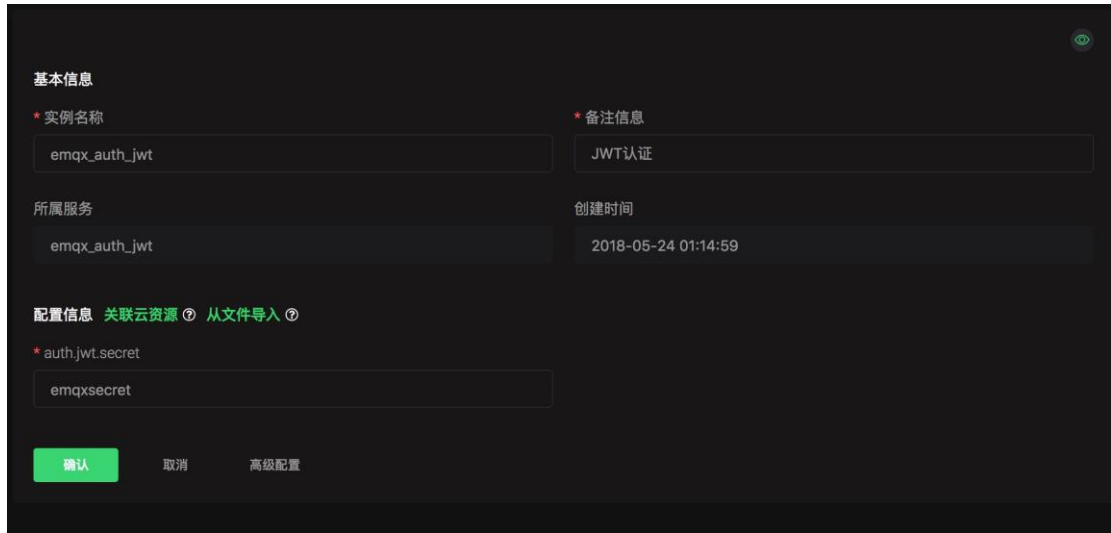


根据自己的需求填写 `基本信息` 和 `备注信息`，初始化配置有两种方式：

- 填写 `auth.jwt.secret`
- 从本地导入一个 json 文件格式的文本

服务编辑

点击编辑，修改 `emqx_auth_jwt` 参数配置：



基本信息

* 实例名称 `emqx_auth_jwt` * 备注信息 `JWT认证`

所属服务 `emqx_auth_jwt` 创建时间 `2018-05-24 01:14:59`

配置信息 [关联云资源](#) [从文件导入](#)

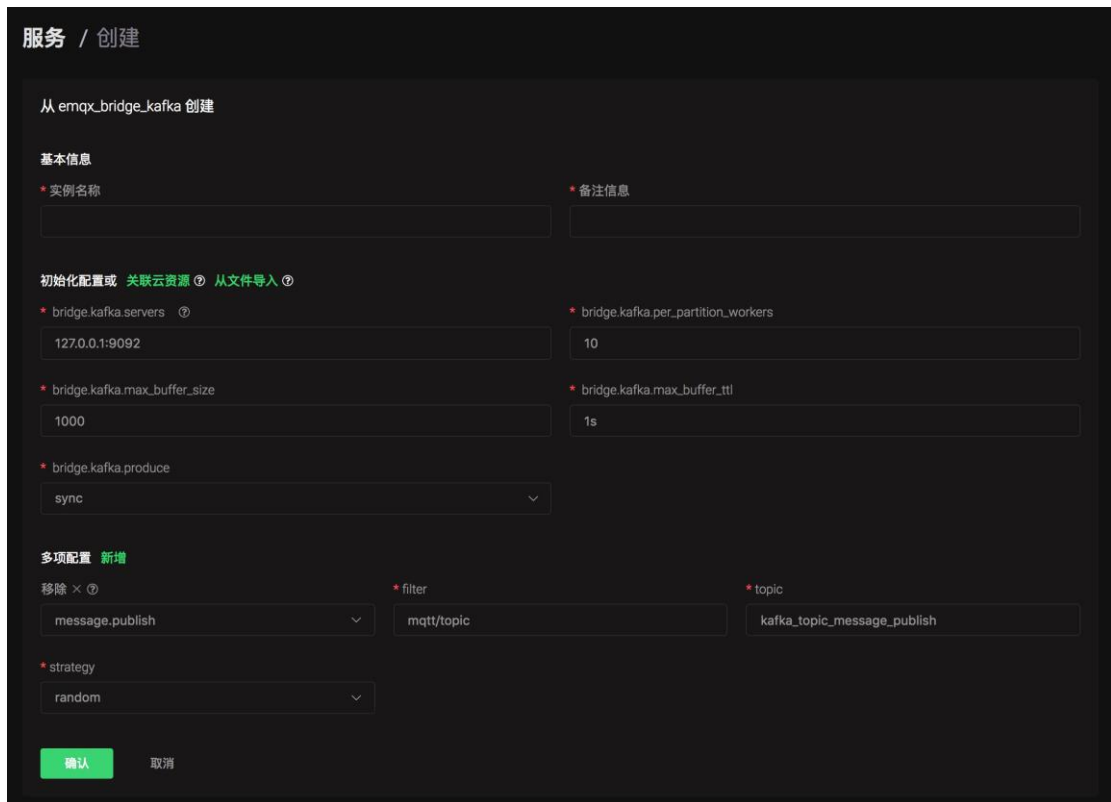
* `auth.jwt.secret` `emqxsecret`

[确认](#) [取消](#) [高级配置](#)

点击下载，可以将 `emqx_auth_jwt` 配置以 json 格式导出到本地，在您下次服务创建 -> 初始化配置时使用。

emqx_bridge_kafka 服务配置

服务创建



服务 / 创建

从 `emqx_bridge_kafka` 创建

基本信息

* 实例名称 * 备注信息

初始化配置或 [关联云资源](#) [从文件导入](#)

* `bridge.kafka.servers` `127.0.0.1:9092` * `bridge.kafka.per_partition_workers` `10`

* `bridge.kafka.max_buffer_size` `1000` * `bridge.kafka.max_buffer_ttl` `1s`

* `bridge.kafka.produce` `sync`

多项配置 [新增](#)

移除 × [filter](#) [topic](#)

`message.publish` `mqtt/topic` `kafka_topic_message_publish`

* `strategy` `random`

[确认](#) [取消](#)

实例名称 备注信息 和 初始化配置 根据自己需求填写，初始化配置提供三种方式：

- 填写 `bridge.kafka.servers` `bridge.kafka.partition_strategy`
`bridge.kafka.per_partition_workers` `bridge.kafka.max_buffer_size`
`bridge.kafka.max_buffer_ttl` `bridge.kafka.produce`
- 关联云资源
- 从本地导入一个 json 文件格式的文本

多项配置 可动态配置多个 MQTT 事件【连接事件、断开连接事件、消息发布事件、消息订阅事件、消息取消订阅事件、消息投递事件、消息应答事件】桥接到 Kafka。`filter` 对应 MQTT 主题过滤器，`topic` 对应 kafka 主题，`strategy` 支持 4 种策略：

- **random**: 随机，快速，但是不会维护 `partition_workers` 秩序；
- **strict_round_robin**: 消息桥接会转发到不同的 `partition workers`；
- **sticky_round_robin**: 当消息桥接数量达到 `bridge.kafka.max_buffer_size` 设置的值后，消息桥接另一个分区；
- **custom**: 可以对主题进行配置，可以根据 `partition Key` 来决定分区。

注意事项

多项配置需要根据实际业务情况进行新增，如果此项无数据，是不会有消息桥接到 kafkaServer。

服务编辑

点击编辑，修改 `emqx_bridge_kafka` 参数配置：

服务 / emqx_brige_kafka

基本信息

* 实例名称	emqx_brige_kafka	* 备注信息	MQTT消息桥接kafka服务器
所属服务	emqx_bridge_kafka	创建时间	2018-06-07 10:54:02

配置信息 [关联云资源](#) [从文件导入](#)

* bridge.kafka.servers	127.0.0.1:9092	* bridge.kafka.per_partition_workers	10
* bridge.kafka.max_buffer_size	1000	* bridge.kafka.max_buffer_ttl	1s
* bridge.kafka.produce	sync		

多项配置 [新增](#)

移除 × [帮助](#)

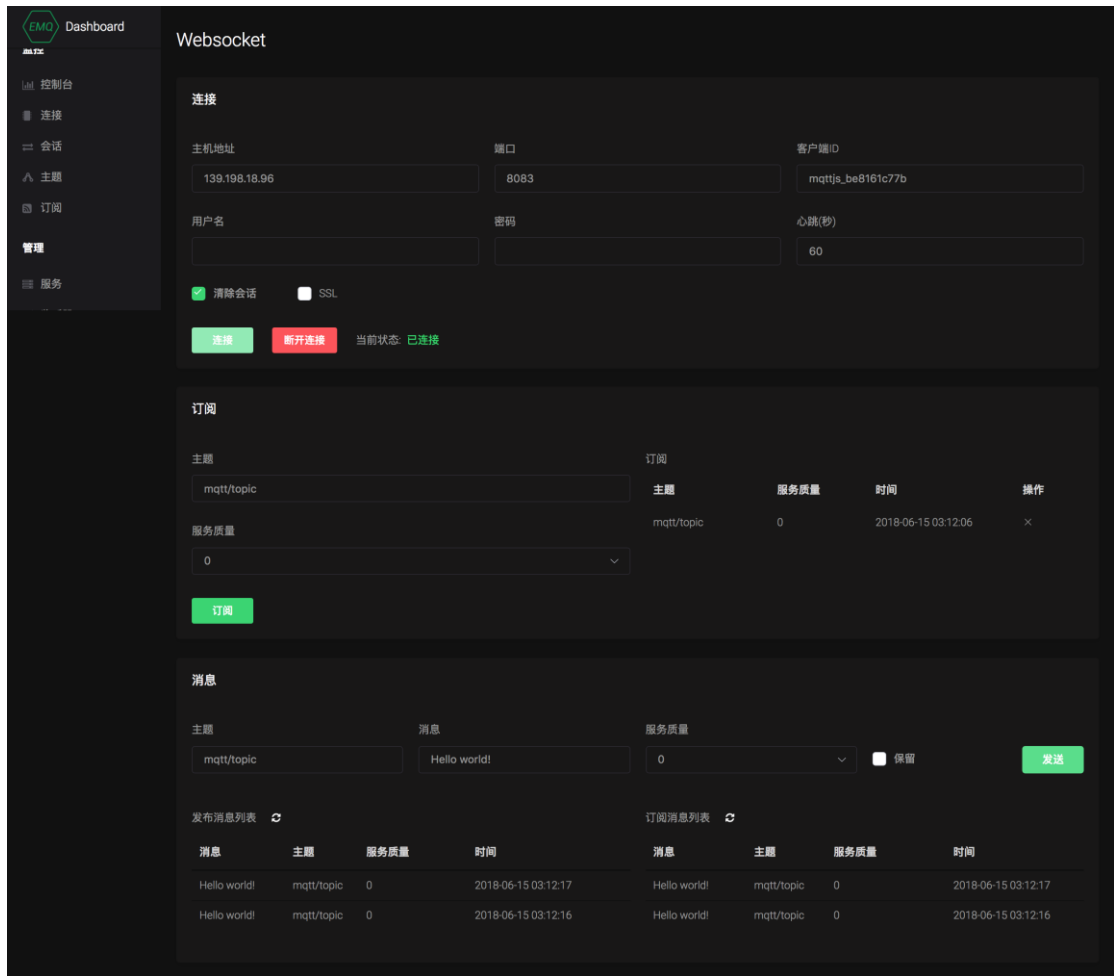
* message.publish	message.publish	* filter	mqtt/topic	* topic	kafka_topic_message_publish
* strategy	custom				

[确认](#) [取消](#)

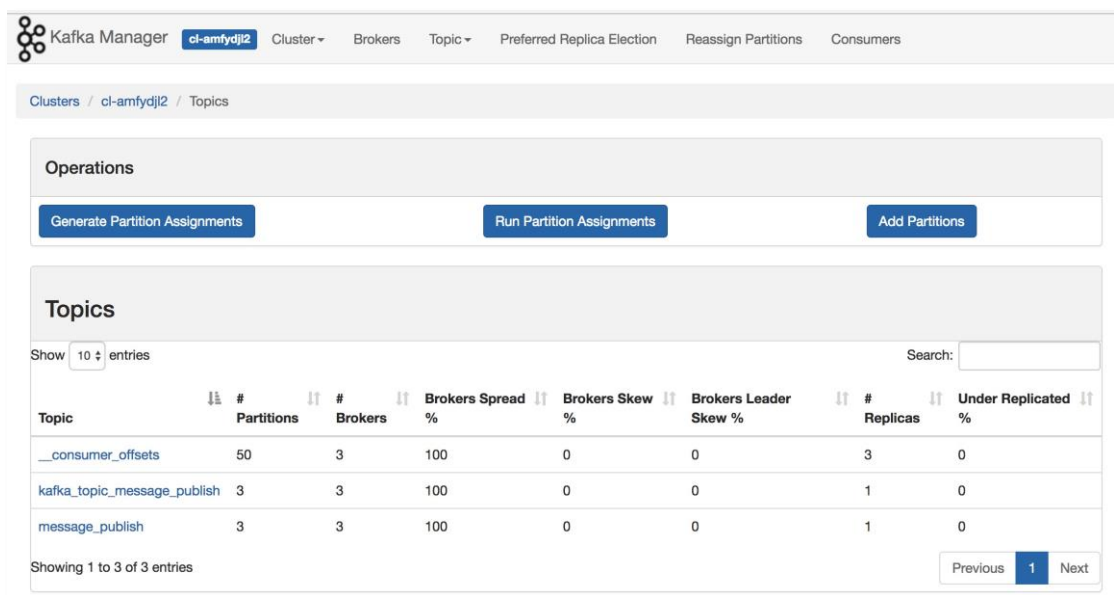
点击下载，可以将 `emqx_bridge_kafka` 配置以 json 格式导出到本地，在您下次服务创建 -> 初始化配置时使用.

MQTT 消息桥接 KafkaServer 示例

Dashboard 提供 MQTT WebSocket 客户端测试界面，测试将 mqtt 主题为 `mqtt/topic` 消息桥接到 kafka 中主题为 `kafka_topic_message_publish`:



从 Kafka-Manager 管理界面可查看 `kafka_topic_message_publish` 主题被创建:



接下来，可以从 kafka_topic_message_publish 此主题开始消费消息。

EMQ 安全组配置

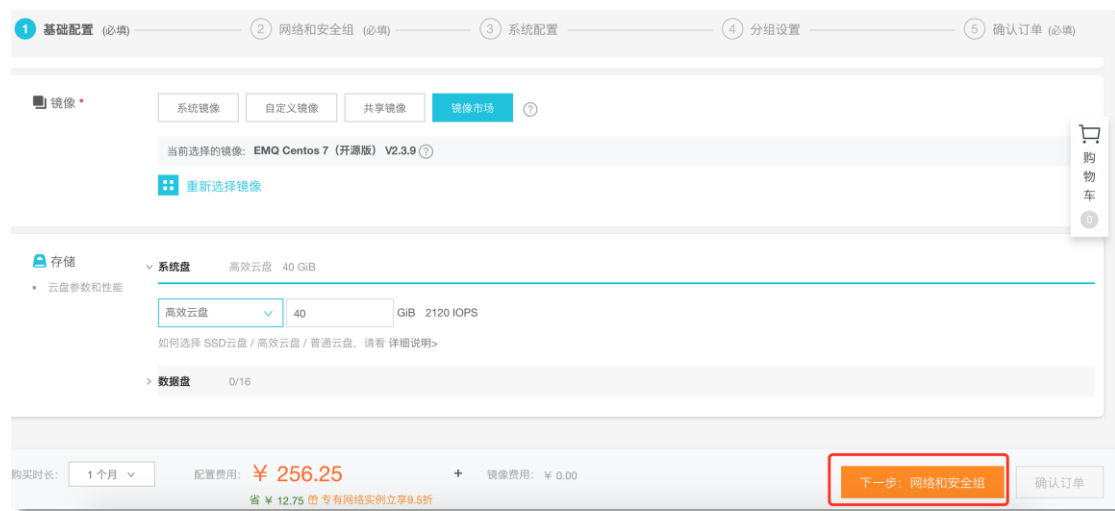
安全组相关介绍：

https://help.aliyun.com/document_detail/25833.html

下列是 EMQ 所需的 TCP 端口 1883、管理端口 18083 配置过程。

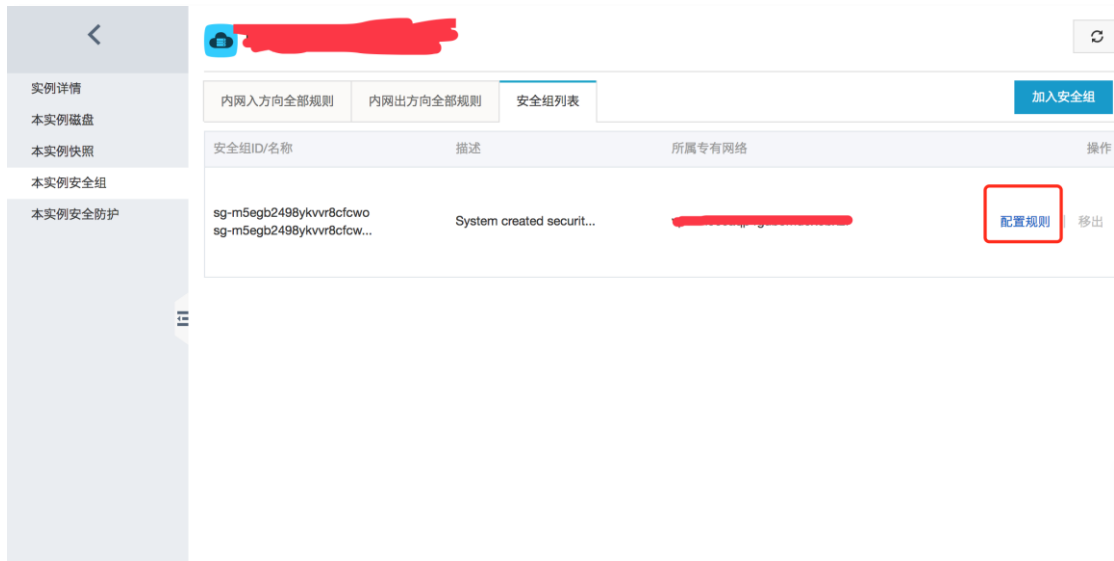
1. 新建增加安全组

点击“下一步：网络和安全组”



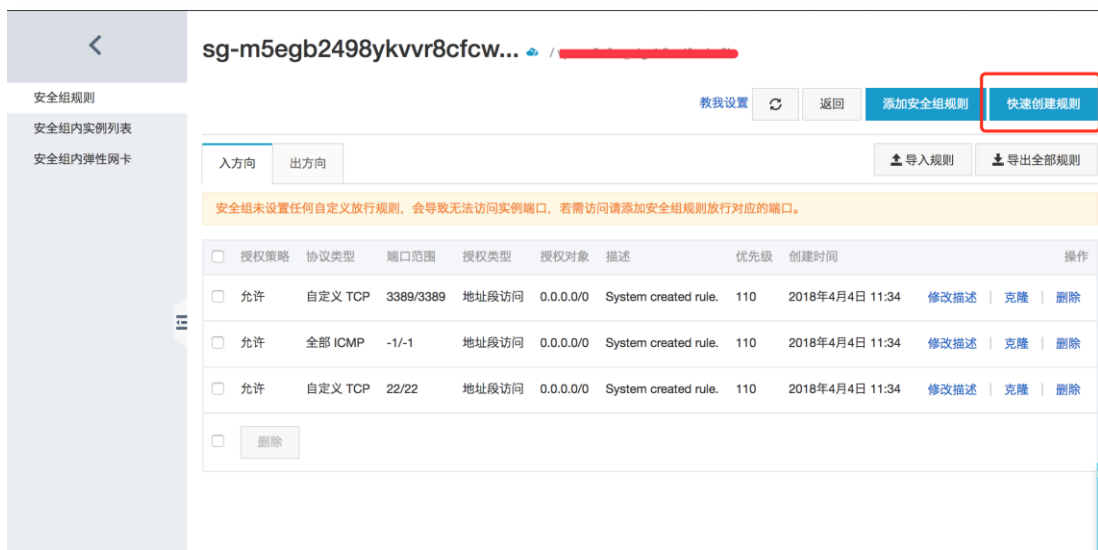
2. 配置安全组规则

点击安全组中的配置规则



3. 创建规则

点击其中的“快速创建规则”，创建相关安全组规则



4. 填写开放端口

按红框内进行正确填写，自定义端口里面选择 TCP，开放端口填写 1883/1883，点击确认完成。同样创建第二次，第二次端口填写 18083/18083，其它不变。

快速创建规则

网卡类型: 内网

规则方向: 入方向

授权策略: 允许

常用端口(TCP): SSH (22) telnet (23) HTTP (80) HTTPS (443) MS SQL (1433) Oracle (1521) MySQL (3306) RDP (3389) PostgreSQL (5432) Redis (6379)

自定义端口: TCP UDP 1883/1883

授权类型: 地址段访问

授权对象: 0.0.0.0/0

优先级: 1

描述: MQTT接入

长度为2-256个字符, 不能以http://或https://开头。

需要开放端口

教我设置

可以访问上述端口的网段全部开放填写0.0.0.0/0

5. 确认更改端口

确定后, 能在红框内看到相关的安全组列表。

sg-m5egb2498ykvvr8cfw... / vpc-m5e8aqp4gub8md0xobx2l

教我设置 返回 添加安全组规则 快速创建规则

安全组规则

安全组内实例列表

安全组内弹性网卡

入方向 出方向 导入规则 导出全部规则

授权策略	协议类型	端口范围	授权类型	授权对象	描述	优先级	创建时间	操作
<input type="checkbox"/> 允许	自定义 TCP	18083/18083	地址段访问	0.0.0.0/0	EMQ dashboard端口	1	2018年7月12日 15:58	修改描述 克隆 删除
<input type="checkbox"/> 允许	自定义 TCP	1883/1883	地址段访问	0.0.0.0/0	MQTT接入	1	2018年7月11日 11:19	修改描述 克隆 删除
<input type="checkbox"/> 允许	自定义 TCP	3389/3389	地址段访问	0.0.0.0/0	System created rule.	110	2018年4月4日 11:34	修改描述 克隆 删除
<input type="checkbox"/> 允许	全部 ICMP	-1/-1	地址段访问	0.0.0.0/0	System created rule.	110	2018年4月4日 11:34	修改描述 克隆 删除
<input type="checkbox"/> 允许	自定义 TCP	22/22	地址段访问	0.0.0.0/0	System created rule.	110	2018年4月4日 11:34	修改描述 克隆 删除

验证测试

参考我们的博客中的“客户端测试 – MQTTBox”部分来验证安装是否成功:

- 超级链接: <https://www.jianshu.com/p/e5cf0c1fd55c>

附录

EMQ 2.0 消息服务器默认占用的 TCP 端口

端口号	描述	是否建议公网开放
1883	MQTT 协议端口	是
18083	Dashboard 管理控制台端口	是
8883	MQTT/SSL 端口	否 (安全连接需要)
8083	MQTT/WebSocket 端口	否 (视情况而定)
8080	HTTP API 端口	否 (视情况而定)

更多详情请见: <http://emqtt.com/docs/v2/install.html#tcp>

常用网址

Github: <https://github.com/emqtt/>

在线文档: <http://emqtt.com/docs/v2/index.html>

博客: <https://www.jianshu.com/u/9cbcdf094d33>