

RabbitMQ-高级-集群

飞哥 VIP 分类: 学习笔记 创建时间: 2021/03/06 00:07 字体 皮肤

01、RabbitMQ 集群

RabbitMQ这款消息队列中间件产品本身是基于Erlang编写，Erlang语言天生具备分布式特性（通过同步Erlang cookie来实现）。因此，RabbitMQ天然支持Clustering。这使得RabbitMQ本身不需要像ActiveMQ、Kafka那实现HA方案和保存集群的元数据。集群是保证可靠性的一种方式，同时可以通过水平扩展以达到增加消息吞吐量在实际使用过程中多采取多机多实例部署方式，为了便于同学们练习搭建，有时候你不得不在一台机器上去搭建-要针对单机多实例这种方式来进行开展。

主要参考官方文档：<https://www.rabbitmq.com/clustering.html>

02、集群搭建

配置的前提是你的rabbitmq可以运行起来，比如” ps aux|grep rabbitmq” 你能看到相关进程，又比如运行 “r看到类似如下信息，而不报错：

执行下面命令进行查看：

1. ps aux|grep rabbitmq

```
[root@localhost ~]# ps aux|grep rabbitmq
root      54069  0.0  0.2 241336  4652 pts/0    S    18:52   0:00 sudo RABBITMQ_NODE_PORT=5672 RABBITMQ_NODENAME=rabbit-1 rabbitmq-server start
root      54071  0.0  0.0 144416  1556 pts/0    S    18:52   0:00 /sbin/runuser -u rabbitmq -- /usr/lib/rabbitmq/bin/rabbitmq-server start
rabbitmq  54079  0.0  0.0 113280  1524 pts/0    S    18:52   0:00 /bin/sh /usr/lib/rabbitmq/bin/rabbitmq-server start
rabbitmq  54086  0.5  4.4 1821576 82212 pts/0    SL   18:52   0:09 /usr/lib64/erlang/erts-11.1.7/bin/beam.smp -W w -MBas ageffcbf -MHas ageffcbf -MBmb
l 128000 -sbwt none -sbwtdcpu none -sbwtdio none -B i -- -root /usr/lib64/erlang -progname erl -- -home /var/lib/rabbitmq -- -pa -noshell -noinput -s
rs [] start
rabbitmq  54093  0.0  0.0   4356   540 ?        Ss   18:52   0:00 erl_child_setup 1024
rabbitmq  54128  0.0  0.0   11760   608 ?        S    18:52   0:00 /usr/lib64/erlang/erts-11.1.7/bin/epmd -daemon
rabbitmq  54145  0.0  0.0   11592   456 ?        Ss   18:52   0:00 inet_gethost 4
rabbitmq  54146  0.0  0.0   13716   700 ?        S    18:52   0:00 inet_gethost 4
```

或者

1. systemctl status rabbitmq-server

注意：确保RabbitMQ可以运行的，确保完成之后，把单机版的RabbitMQ服务停止，后台看不到RabbitMQ的

03、单机多实例搭建

场景：假设有两个rabbitmq节点，分别为rabbit-1, rabbit-2，rabbit-1作为主节点，rabbit-2作为从节点。
启动命令：RABBITMQ_NODE_PORT=5672 RABBITMQ_NODENAME=rabbit-1 rabbitmq-server -detach
结束命令：rabbitmqctl -n rabbit-1 stop

03-1、第一步：启动第一个节点rabbit-1

01、RabbitMQ 集群

02、集群搭建

03、单机多实例搭建

03-1、第一步：启动第一个节点rabbit-1

03-2、启动第二个节点rabbit-2

03-3、验证启动 “ps aux|grep rabbitmq”

03-4、rabbit-1操作作为主节点

03-5、rabbit2操作作为从节点

03-6、验证集群状态

03-7、Web监控

03-8、小结

```
1. > sudo RABBITMQ_NODE_PORT=5672 RABBITMQ_NODENAME=rabbit-1 rabbitmq-server start &
2.
3. ....省略.....
4. ##### Logs: /var/log/rabbitmq/rabbit-1.log
5. ##### ## /var/log/rabbitmq/rabbit-1-sasl.log
6. #####
7. Starting broker...
8. completed with 7 plugins.
```

至此节点rabbit-1启动完成。

03-2、启动第二个节点rabbit-2

注意：web管理插件端口占用,所以还要指定其web插件占用的端口号

RABBITMQ_SERVER_START_ARGS=" -rabbitmq_management listener [{port,15673}]"

```
1. sudo RABBITMQ_NODE_PORT=5673 RABBITMQ_SERVER_START_ARGS="-rabbitmq_management li
  [{port,15673}]" RABBITMQ_NODENAME=rabbit-2 rabbitmq-server start &
2.
3. ....省略.....
4. ##### Logs: /var/log/rabbitmq/rabbit-2.log
5. ##### ## /var/log/rabbitmq/rabbit-2-sasl.log
6. #####
7. Starting broker...
8. completed with 7 plugins.
```

至此节点rabbit-2启动完成

03-3、验证启动 “ps aux|grep rabbitmq”

```
1. rabbitmq 2022 2.7 0.4 5349380 77020 ? S1 11:03 0:06 /usr/lib/erlang
  9.2/bin/beam.smp -W w -A 128 -P 1048576 -t 5000000 -stbt db -zdbbl 128000 -K tru
  /usr/lib/erlang -progname erl -- -home /var/lib/rabbitmq -- -pa
  /usr/lib/rabbitmq/lib/rabbitmq_server-3.6.15/ebin -noshell -noinput -s rabbit bc
  -boot start_sasl -kernel inet_default_connect_options [{nodelay,true}] -rabbit t
  [{"auto",5672}] -sasl errlog_type error -sasl sasl_error_logger false -rabbit er
  {file,"/var/log/rabbitmq/rabbit-1.log"} -rabbit sasl_error_logger {file,"/var/lc
  1-sasl.log"} -rabbit enabled_plugins_file "/etc/rabbitmq/enabled_plugins" -rabbi
  "/usr/lib/rabbitmq/plugins:/usr/lib/rabbitmq/lib/rabbitmq_server-3.6.15/plugins"
  plugins_expand_dir "/var/lib/rabbitmq/mnesia/rabbit-1-plugins-expand" -os_mon st
  os_mon start_disksup false -os_mon start_memsup false -mnesia dir "/var/lib/rabb
  1" -kernel inet_dist_listen_min 25672 -kernel inet_dist_listen_max 25672 start
2.
3. rabbitmq 2402 4.2 0.4 5352196 77196 ? S1 11:05 0:05 /usr/lib/erlang
  9.2/bin/beam.smp -W w -A 128 -P 1048576 -t 5000000 -stbt db -zdbbl 128000 -K tru
  /usr/lib/erlang -progname erl -- -home /var/lib/rabbitmq -- -pa
  /usr/lib/rabbitmq/lib/rabbitmq_server-3.6.15/ebin -noshell -noinput -s rabbit bc
  -boot start_sasl -kernel inet_default_connect_options [{nodelay,true}] -rabbit t
  [{"auto",5673}] -sasl errlog_type error -sasl sasl_error_logger false -rabbit er
  {file,"/var/log/rabbitmq/rabbit-2.log"} -rabbit sasl_error_logger {file,"/var/lc
  2-sasl.log"} -rabbit enabled_plugins_file "/etc/rabbitmq/enabled_plugins" -rabbi
  "/usr/lib/rabbitmq/plugins:/usr/lib/rabbitmq/lib/rabbitmq_server-3.6.15/plugins"
  plugins_expand_dir "/var/lib/rabbitmq/mnesia/rabbit-2-plugins-expand" -os_mon st
  os_mon start_disksup false -os_mon start_memsup false -mnesia dir "/var/lib/rabb
  2" -rabbitmq_management listener [{port,15673}] -kernel inet_dist_listen_min 256
  inet_dist_listen_max 25673 start
```

03-4、rabbit-1操作作为主节点

[01、RabbitMQ 集群](#)

[02、集群搭建](#)

[03、单机多实例搭建](#)



[03-1、第一步：启动第一个节点rabbit-1](#)

[03-2、启动第二个节点rabbit-2](#)

[03-3、验证启动 “ps aux|grep rabbitmq”](#)

[03-4、rabbit-1操作作为主节点](#)

[03-5、rabbit2操作作为从节点](#)

[03-6、验证集群状态](#)

[03-7、Web监控](#)

[03-8、小结](#)



- 1. #停止应用
- 2. > sudo rabbitmqctl -n rabbit-1 stop_app
- 3. #目的是清除节点上的历史数据（如果不清除，无法将节点加入到集群）
- 4. > sudo rabbitmqctl -n rabbit-1 reset
- 5. #启动应用
- 6. > sudo rabbitmqctl -n rabbit-1 start_app

[01、RabbitMQ 集群](#)

[02、集群搭建](#)

[03、单机多实例搭建](#)

[03-1、第一步：启动第一个节点rabbit-1](#)

[03-2、启动第二个节点rabbit-2](#)

[03-3、验证启动“ps aux|grep rabbitmq”](#)

[03-4、rabbit-1操作作为主节点](#)

[03-5、rabbit2操作作为从节点](#)

[03-6、验证集群状态](#)

[03-7、Web监控](#)

[03-8、小结](#)

03-5、rabbit2操作作为从节点

- 1. # 停止应用
- 2. > sudo rabbitmqctl -n rabbit-2 stop_app
- 3. # 目的是清除节点上的历史数据（如果不清除，无法将节点加入到集群）
- 4. > sudo rabbitmqctl -n rabbit-2 reset
- 5. # 将rabbit2节点加入到rabbit1（主节点）集群当中【Server-node服务器的主机名】
- 6. > sudo rabbitmqctl -n rabbit-2 join_cluster rabbit-1@'Server-node'
- 7. # 启动应用
- 8. > sudo rabbitmqctl -n rabbit-2 start_app

03-6、验证集群状态

- 1. > sudo rabbitmqctl cluster_status -n rabbit-1
- 2.
- 3. //集群有两个节点：rabbit-1@Server-node、rabbit-2@Server-node
- 4. [{nodes,[{disc,['rabbit-1@Server-node','rabbit-2@Server-node']}]},
- 5. {running_nodes,['rabbit-2@Server-node','rabbit-1@Server-node']},
- 6. {cluster_name,<<"rabbit-1@Server-node.localdomain">>},
- 7. {partitions,[]},
- 8. {alarms,[{'rabbit-2@Server-noc < ,{'rabbit-1@Server-node',[]}]}} >

03-7、Web监控

Connections: 0

Channels: 0

Exchanges: 7

Queues: 0

Consumers: 0

▼ Nodes

Name	File descriptors ?	Socket descriptors ?	Erlang processes	Memory ?	Disk space	Uptime	Info
rabbit-1@localhost	<div>35</div> <div>1024 available</div>	<div>0</div> <div>832 available</div>	<div>440</div> <div>1048576 available</div>	<div>78 MiB</div> <div>728 MiB high watermark</div> <div>14 MiB low watermark</div>	<div>14 GiB</div>	21m 21s	<div>basic</div> <div>disc</div> <div>1</div> <div>rss</div>
rabbit-2@localhost	<div>35</div> <div>1024 available</div>	<div>0</div> <div>832 available</div>	<div>440</div> <div>1048576 available</div>	<div>80 MiB</div> <div>728 MiB high watermark</div> <div>14 MiB low watermark</div>	<div>14 GiB</div>	20m 58s	<div>basic</div> <div>disc</div> <div>1</div> <div>rss</div>

注意在访问的时候：web结面的管理需要给15672 node-1 和15673的node-2 设置用户名和密码。如下：

- 1. rabbitmqctl -n rabbit-1 add_user admin admin
- 2. rabbitmqctl -n rabbit-1 set_user_tags admin administrator
- 3. rabbitmqctl -n rabbit-1 set_permissions -p / admin ".*" ".*" ".*"
- 4.
- 5. rabbitmqctl -n rabbit-2 add_user admin admin
- 6. rabbitmqctl -n rabbit-2 set_user_tags admin administrator
- 7. rabbitmqctl -n rabbit-2 set_permissions -p / admin ".*" ".*" ".*"

03-8、小结



Tips:

如果采用多机部署方式，需读取其中一个节点的cookie, 并复制到其他节点（节点之间通过cookie确定相互是否可通信）。 cookie存放在/var/lib/rabbitmq/.erlang.cookie。

例如：主机名分别为rabbit-1、rabbit-2

1、逐个启动各节点

2、配置各节点的hosts文件(vim /etc/hosts)

ip1: rabbit-1

ip2: rabbit-2

其它步骤雷同单机部署方式

[01、RabbitMQ 集群](#)

[02、集群搭建](#)

[03、单机多实例搭建](#) ▾



[03-1、第一步：启动第一个节点rabbit-1](#)

[03-2、启动第二个节点rabbit-2](#)

[03-3、验证启动 “ps aux|grep rabbitmq”](#)

[03-4、rabbit-1操作作为主节点](#)

[03-5、rabbit2操作为从节点](#)

[03-6、验证集群状态](#)

[03-7、Web监控](#)

[03-8、小结](#)

[关于我们](#) | [加入我们](#) | [联系我们](#) | [帮助中心](#)

Copyright © 广东学相伴网络科技有限公司 粤ICP备 - 2020109190号

