Rabbit MQ高可用服务搭建

- 节点配置

节点	ip	节点类型
node1	10.2.40.220	磁盘节点
node2	10.2.40.222	内存节点

- 系统版本

  Centos-7

- 负载均衡

  HAProxy

- 学习资料参考 https://www.cnblogs.com/xishuai/p/centos-rabbitmq-cluster-and-haproxy.html
- hyproxy https://www.cnblogs.com/xiangsikai/p/8915609.html

系统环境配置

- 修改主机名

```
#在node1执行如下命令
echo "node1" >> /etc/hostname
#在node12执行如下命令
echo "node2" >> /etc/hostname
#重启服务器使生效
reboot
## 查看是否设置成功
hostnamectl status
```

- 修改hots文件添加如下内容

```
# node1执行
cp /etc/hosts /etc/hosts.back
cat >> /etc/hosts << EOF
10.2.40.220 node1
10.2.40.222 node2
127.0.0.1   node1
::1         node1
EOF
# node2执行
cp /etc/hosts /etc/hosts.back
cat >> /etc/hosts << EOF
10.2.40.220 node1
10.2.40.222 node2
127.0.0.1   node2
::1         node2
EOF
```

单机版搭建

- 更新yum

```
yum -y update
```

- 安装Erlang

```
#添加yum源
cat > /etc/yum.repos.d/rabbitmq-erlang.repo << EOF
[rabbitmq-erlang]
name=rabbitmq-erlang
baseurl=https://dl.bintray.com/rabbitmq/rpm/erlang/20/el/7
gpgcheck=1
gpgkey=https://dl.bintray.com/rabbitmq/Keys/rabbitmq-release-signing-key.asc
repo_gpgcheck=0
enabled=1
EOF
# 进行安装
yum -y install erlang socat
```

- 安装RabbitMQ

```
mkdir -p rabbitMQ && cd rabbitMQ
wget https://www.rabbitmq.com/releases/rabbitmq-server/v3.6.10/rabbitmq-server-
3.6.10-1.el7.noarch.rpm

rpm --import https://www.rabbitmq.com/rabbitmq-release-signing-key.asc
rpm -Uvh rabbitmq-server-3.6.10-1.el7.noarch.rpm
#若报错，说明erlang安装失败，需要执行以下命令安装
错误：依赖检测失败：
        erlang >= R16B-03 被 rabbitmq-server-3.6.10-1.el7.noarch 需要
wget http://www.rabbitmq.com/releases/erlang/erlang-19.0.4-
1.el7.centos.x86_64.rpm
rpm -ivh erlang-19.0.4-1.el7.centos.x86_64.rpm
```

- 启动MQ

```
systemctl start rabbitmq-server
# 添加到开机项
systemctl enable rabbitmq-server
```

```
[root@localhost rabbitMQ]# systemctl status rabbitmq-server
● rabbitmq-server.service - RabbitMQ broker
   Loaded: loaded (/usr/lib/systemd/system/rabbitmq-server.service; enabled;
vendor preset: disabled)
   Active: active (running) since 一 2019-09-02 22:37:20 CST; 3min 30s ago
 Main PID: 135926 (beam.smp)
   Status: "Initialized"
   CGroup: /system.slice/rabbitmq-server.service
           ├─135926 /usr/lib64/erlang/erts-8.0.3/bin/beam.smp -W w -A 1024 -P
1048576 -t 5000000 -stbt db ...
```

```
           ├─136067 /usr/lib64/erlang/erts-8.0.3/bin/epmd -daemon
           ├─137299 erl_child_setup 1024
           ├─137387 inet_gethost 4
           └─137388 inet_gethost 4

9月 02 22:37:19 node1 rabbitmq-server[135926]: RabbitMQ 3.6.10. Copyright (C)
2007-2017 Pivotal Softw...Inc.
9月 02 22:37:19 node1 rabbitmq-server[135926]: ##  ##      Licensed under the
MPL.  See http://www.ra...com/
9月 02 22:37:19 node1 rabbitmq-server[135926]: ##  ##
9月 02 22:37:19 node1 rabbitmq-server[135926]: ##########  Logs:
/var/log/rabbitmq/rabbit@node1.log
9月 02 22:37:19 node1 rabbitmq-server[135926]: ######  ##
/var/log/rabbitmq/rabbit@node1-sasl.log
9月 02 22:37:19 node1 rabbitmq-server[135926]: ##########
9月 02 22:37:19 node1 rabbitmq-server[135926]: Starting broker...
9月 02 22:37:20 node1 rabbitmq-server[135926]: systemd unit for activation
check: "rabbitmq-server.service"
9月 02 22:37:20 node1 systemd[1]: Started RabbitMQ broker.
9月 02 22:37:20 node1 rabbitmq-server[135926]: completed with 0 plugins.
Hint: Some lines were ellipsized, use -l to show in full.
```

- 添加控制台

```
rabbitmq-plugins enable rabbitmq_management
```

- 卸载 MQ

```
[root@node1 ~]# rpm -e rabbitmq-server-3.6.10-1.el7.noarch
[root@node1 ~]# rm -rf /var/lib/rabbitmq/     //清除rabbitmq配置文件
```

- 添加用户

```
#添加用户
#rabbitmqctl add_user 用户名 密码
rabbitmqctl add_user admin P@ssw0rd
#设置用户角色
#rabbitmqctl set_user_tags admin 角色名称（支持同时设置多个角色）
rabbitmqctl set_user_tags admin administrator
#设置用户权限
#rabbitmqctl set_permissions -p 虚拟主机名称 用户名 <conf> <write> <read>
rabbitmqctl set_permissions -p / admin ".*" ".*" ".*"
```

高可用集群搭建

- 查找`.erlang.cookie`文件

```
find / -name ".erlang.cookie"
[root@node1 cluster]# find / -name ".erlang.cookie"
/var/lib/docker/overlay2/6f1426376c862a7e3ffdacc578fc38e837ffc7c07fd78a0c0a84b21
a23665179/diff/root/.erlang.cookie
/var/lib/docker/volumes/f82e9cd8b233c5ad01d2758c5e107e946647afe9da240654cd0b1b9b
af1d5a9e/_data/.erlang.cookie
/var/lib/rabbitmq/.erlang.cookie
```

- 停止节点，

```
rabbitmqctl stop
rabbitmq-server -detached

[root@node1 cluster]# rabbitmqctl stop
Stopping and halting node rabbit@node1
```

- 将node1节点.erlang.cookie复制到node2对应路径下面

```
scp /var/lib/rabbitmq/.erlang.cookie root@node2:/var/lib/rabbitmq

[root@node1 cluster]# scp /var/lib/rabbitmq/.erlang.cookie
root@node2:/var/lib/rabbitmq
The authenticity of host 'node2 (10.2.40.222)' can't be established.
ECDSA key fingerprint is SHA256:mg9x0L3waADzWa4b4x0E11sBrTiOkrELAW/yORHU2UM.
ECDSA key fingerprint is MD5:58:7a:2a:20:d2:0d:01:bb:7b:6c:fa:21:4e:b1:d9:12.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'node2,10.2.40.222' (ECDSA) to the list of known
hosts.
root@node2's password:
.erlang.cookie                                                100%   20
 48.4KB/s   00:00
```

- 以后台形式重新启动

```
[root@node1 cluster]#  rabbitmq-server -detached
Warning: PID file not written; -detached was passed.
```

- 设置内存节点

```
#集群默认时磁盘节点，所以我们将node2设置为内存节点
#在node2执行以下命令
[root@node2 rabbitmq]# rabbitmqctl stop_app
[root@node2 rabbitmq]# rabbitmqctl reset
[root@node2 rabbitmq]# rabbitmqctl join_cluster --ram rabbit@node1
[root@node2 rabbitmq]# rabbitmqctl start_app
[root@node2 rabbitmq]# rabbitmqctl cluster_status
Cluster status of node rabbit@node2
[{nodes,[{disc,[rabbit@node1]},{ram,[rabbit@node2]}]},
 {running_nodes,[rabbit@node1,rabbit@node2]},
 {cluster_name,<<"rabbit@node1">>},
 {partitions,[]},
 {alarms,[{rabbit@node1,[]},{rabbit@node2,[]}]}]
```

 1567441233333

搭建 HAProxy 负载均衡

因为 RabbitMQ 本身不提供负载均衡，搭建 HAProxy，用作 RabbitMQ 集群的负载均衡。

在node1安装

```
tar -zxvf haproxy-1.7.8.tar.gz
mv haproxy-1.7.8 haproxy
cd haproxy
make TARGET=linux2628 && make install

#TARGET=linux310，内核版本，使用uname -r查看内核，如：3.10.0-514.el7，此时该参数就为linux310；kernel 大于2.6.28的可以用：TARGET=linux2628；
```

```
[root@node1 haproxy]# make install
install -d "/usr/local/sbin"
install haproxy  "/usr/local/sbin"
install -d "/usr/local/share/man"/man1
install -m 644 doc/haproxy.1 "/usr/local/share/man"/man1
install -d "/usr/local/doc/haproxy"
for x in configuration management architecture cookie-options lua WURFL-device-detection proxy-protocol linux-syn-cookies network-namespaces DeviceAtlas-device-detection 51Degrees-device-detection netscaler-client-ip-insertion-protocol close-options SPOE intro; do \
        install -m 644 doc/$x.txt "/usr/local/doc/haproxy" ; \
done
```

- 复制haproxy启动服务到指定目录下

```
cp /usr/local/sbin/haproxy sbin/
```

- 添加启动脚本到系统服务目录内，并给脚本添加启动权限

```
#创建目录
mkdir /etc/haproxy
cat > /etc/haproxy/haproxy.cfg << EOF
global
```

```
    log     127.0.0.1  local0 info
    log     127.0.0.1  local1 notice
    daemon
    maxconn 4096

defaults
    log     global
    mode    tcp
    option  tcplog
    option  dontlognull
    retries 3
    option  abortonclose
    maxconn 4096
    timeout connect  5000ms
    timeout client  3000ms
    timeout server  3000ms
    balance roundrobin

listen private_monitoring
    bind    0.0.0.0:8100
    mode    http
    option  httplog
    stats   refresh  5s
    stats   uri  /stats
    stats   realm   Haproxy
    stats   auth  admin:admin

listen rabbitmq_admin
    bind    0.0.0.0:8102
    server  node1 node1:15672
    server  node2 node2:15672

listen rabbitmq_cluster
    bind    0.0.0.0:8101
    mode    tcp
    option  tcplog
    balance roundrobin
    timeout client  3h
    timeout server  3h
    server  node1  node1:5672  check  inter  5000  rise  2  fall  3
    server  node2  node2:5672  check  inter  5000  rise  2  fall  3
EOF
```

- 启动hyproxy

```
haproxy -f /etc/haproxy/haproxy.cfg

http://node1:8100/stats：HAProxy 负载均衡信息地址，账号密码：admin/admin。
http://node1:8101：RabbitMQ Server 服务地址（基于负载均衡）。
http://node1:8102：RabbitMQ Server Web 管理界面（基于负载均衡）。
```