

## 搭建 mcollective 高可用，使 puppet 架构更加安全、稳定

存在这样一种场景，当你的 puppet 基于 mcollective 环境搭建完成之后，需要考虑 MQ 的高可用，否则，MQ 挂掉之后就不能用 mco 命令进行推送了哦。如何做 MQ 的高可用呢，其实有两种方法：方法一：两台 MQ 做集群，通过复制队列信息进行同步，节点访问可通过浮动 IP 进行。方法二：两台 MQ 独立，在 MC Server 端做 failover，通过 rabbitmq 的 plugins 参数实现，可设置自动检测，切换时间等等。

## 一、配置 Rabbitmq

安装（略），可参考 <http://kisspuppet.com/2013/11/10/mcollective-middleware/>或 <http://rsyslog.org/2013/11/10/mcollective-middleware/>

### 1. 开启插件 rabbitmq\_stomp

```
[root@linuxmaster1poc ~]# rabbitmq-plugins enable rabbitmq_stomp
The following plugins have been enabled:
    rabbitmq_stomp
Plugin configuration has changed. Restart RabbitMQ for changes to take effect.
```

### 2. 添加 tcp 监听端口和范围

```
[root@linuxmaster1poc ~]# vim /etc/rabbitmq/rabbitmq.config

[
  {rabbitmq_stomp, [{tcp_listeners, [61613]}}]
].
```

备注：可参考 <http://www.rabbitmq.com/stomp.html>

### 3. 创建账户并设置权限

如果你以前配置过，建议将配置清空

```
[root@linuxmaster1poc ~]# rabbitmqctl stop_app
Stopping node rabbit@linuxmaster1poc ...
...done.
[root@linuxmaster1poc ~]# rabbitmqctl reset
Resetting node rabbit@linuxmaster1poc ...
...done.
```

```
[root@linuxmaster1poc ~]# rabbitmqctl start_app
Starting node rabbit@linuxmaster1poc ...
...done.
```

删除默认用户 `guest`，添加三个用户（`webadmin-http` 访问用，`admin`--管理员，`mc_rabbitmq--mcollective` 链接用）

```
[root@linuxmaster1poc ~]# rabbitmqctl list_users
Listing users ...
guest [administrator]
...done.
[root@linuxmaster1poc ~]# rabbitmqctl delete_user guest
Deleting user "guest" ...
...done.
[root@linuxmaster1poc ~]# rabbitmqctl add_user mc_rabbitmq 123.com
Creating user "mc_rabbitmq" ...
...done.
[root@linuxmaster1poc ~]# rabbitmqctl add_user admin password=123.com
Creating user "admin" ...
...done.
[root@linuxmaster1poc ~]# rabbitmqctl add_user web_admin 123.com
Creating user "web_admin" ...
...done.
```

设置用户的角色

```
[root@linuxmaster1poc ~]# rabbitmqctl set_user_tags admin administrator
Setting tags for user "admin" to [administrator] ...
...done.

[root@linuxmaster1poc ~]# rabbitmqctl set_user_tags web_admin monitoring
Setting tags for user "web_admin" to [monitoring] ...
...done.
```

创建虚拟主机组

```
[root@linuxmaster1poc ~]# rabbitmqctl add_vhost /mcollective
Creating vhost "/mcollective" ...
...done.
```

设置用户访问虚拟主机组的权限

```
[root@linuxmaster1poc ~]# rabbitmqctl set_permissions -p "/mcollective" mc_rabbitmq
".*" ".*" ".*"
```

```
Setting permissions for user "mc_rabbitmq" in vhost "/mcollective" ...
...done.
[root@linuxmaster1poc ~]# rabbitmqctl set_permissions -p "/mcollective" admin ".*"
".*" ".*"
Setting permissions for user "admin" in vhost "/mcollective" ...
...done.
[root@linuxmaster1poc ~]# rabbitmqctl set_permissions -p "/mcollective" web_admin
".*" ".*" ".*"
Setting permissions for user "web_admin" in vhost "/mcollective" ...
...done.
```

重启 rabbitmq-server 服务

```
[root@linuxmaster1poc ~]# /etc/init.d/rabbitmq-server restart
Restarting rabbitmq-server: SUCCESS
rabbitmq-server.
```

查看用户以及角色是否创建成功

```
[root@linuxmaster1poc ~]# rabbitmqctl list_users
Listing users ...
admin    [administrator]
mc_rabbitmq []
web_admin [monitoring]
...done.
```

查看虚拟主机组“/mcollective”中所有用户的权限

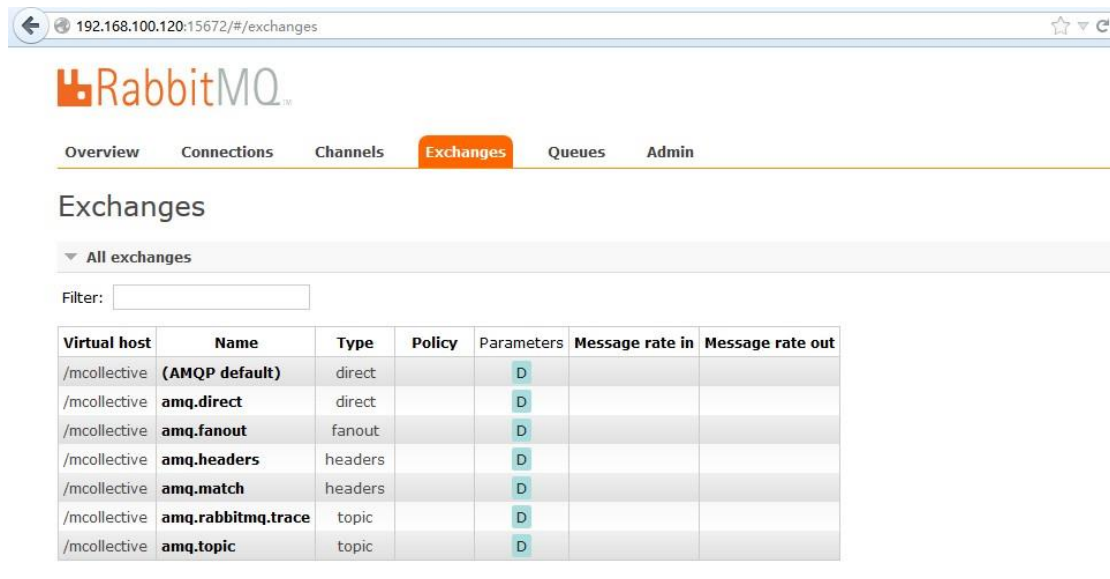
```
[root@linuxmaster1poc ~]# rabbitmqctl list_permissions -p "/mcollective"
Listing permissions in vhost "/mcollective" ...
admin    .* .* .*
mc_rabbitmq .* .* .*
web_admin .* .* .*
...done.
[root@linuxmaster1poc ~]#
```

## 4、登录 <http://192.168.100.120:15672/> 设置虚拟主机“/mcollective”的 exchanges

---

默认配置

```
[root@linuxmaster1poc ~]# rabbitmqctl list_exchanges -p "/mcollective"
Listing exchanges ...
    direct
amq.direct  direct
amq.fanout  fanout
amq.headers headers
amq.match   headers
amq.rabbitmq.trace topic
amq.topic   topic
...done.
```



RabbitMQ

Overview Connections Channels **Exchanges** Queues Admin

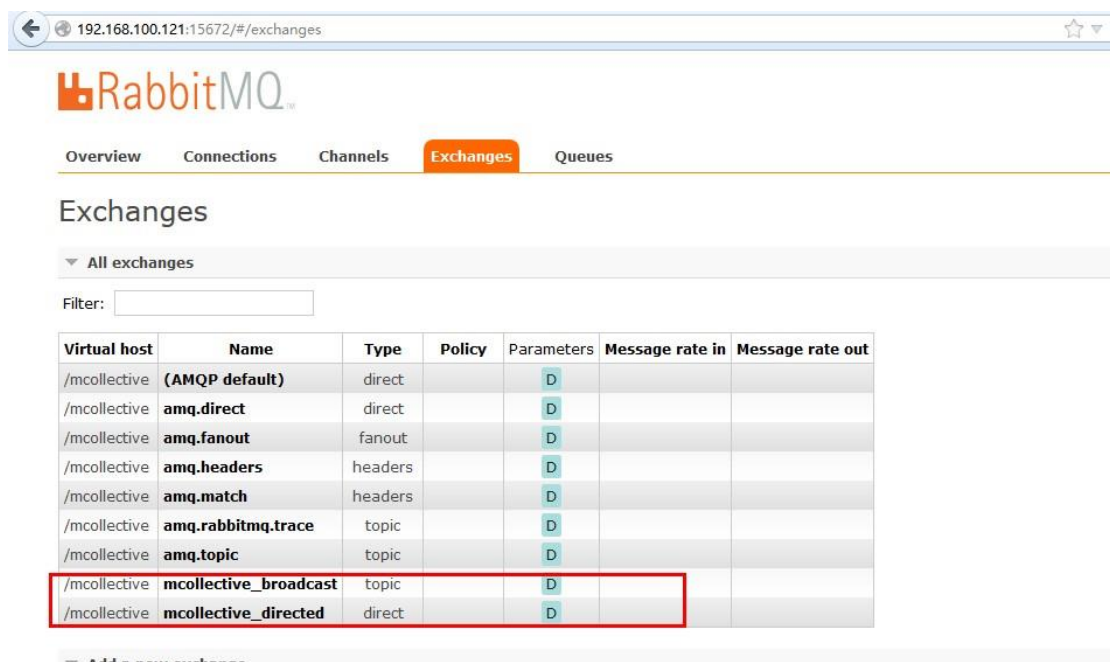
### Exchanges

▼ All exchanges

Filter:

Virtual host	Name	Type	Policy	Parameters	Message rate in	Message rate out
/mcollective	(AMQP default)	direct		D		
/mcollective	amq.direct	direct		D		
/mcollective	amq.fanout	fanout		D		
/mcollective	amq.headers	headers		D		
/mcollective	amq.match	headers		D		
/mcollective	amq.rabbitmq.trace	topic		D		
/mcollective	amq.topic	topic		D		

设置后更新配置



RabbitMQ

Overview Connections Channels **Exchanges** Queues

### Exchanges

▼ All exchanges

Filter:

Virtual host	Name	Type	Policy	Parameters	Message rate in	Message rate out
/mcollective	(AMQP default)	direct		D		
/mcollective	amq.direct	direct		D		
/mcollective	amq.fanout	fanout		D		
/mcollective	amq.headers	headers		D		
/mcollective	amq.match	headers		D		
/mcollective	amq.rabbitmq.trace	topic		D		
/mcollective	amq.topic	topic		D		
/mcollective	mcollective_broadcast	topic		D		
/mcollective	mcollective_directed	direct		D		

▼ Add a new exchange

```
[root@linuxmaster1poc ~]# rabbitmqctl list_exchanges -p "/mcollective"
Listing exchanges ...
    direct
amq.direct  direct
amq.fanout  fanout
amq.headers headers
amq.match   headers
amq.rabbitmq.trace topic
amq.topic   topic
mcollective_broadcast topic
mcollective_directed direct
...done.
```

备注：可参考官网设置 <https://www.rabbitmq.com/man/rabbitmqctl.1.man.html>

## 二、 配置 MCollective:

### 1. 配置 mcollective client 端

```
[root@linuxmaster1poc testing]# cat /etc/mcollective/client.cfg
topicprefix = /topic/
main_collective = mcollective
collectives = mcollective
libdir = /usr/libexec/mcollective
logger_type = console
#loglevel = debug
loglevel = warn

# Plugins
securityprovider = psk
plugin.psk = a36cd839414370e10fd281b8a38a4f48

direct_addressing = 1
connector = rabbitmq
plugin.rabbitmq.vhost = /mcollective #虚拟主机
plugin.rabbitmq.pool.size = 2 #设置地址池里有两个 mq

plugin.rabbitmq.initial_reconnect_delay = 0.01
plugin.rabbitmq.max_reconnect_delay = 30.0 #重连时间
plugin.rabbitmq.use_exponential_back_off = true
plugin.rabbitmq.back_off_multiplier = 2
plugin.rabbitmq.max_reconnect_attempts = 0
plugin.rabbitmq.randomize = false
```

```
plugin.rabbitmq.timeout = -1

plugin.rabbitmq.pool.1.host = 192.168.100.120
plugin.rabbitmq.pool.1.port = 61613
plugin.rabbitmq.pool.1.user = mc_rabbitmq
plugin.rabbitmq.pool.1.password = 123.com
plugin.rabbitmq.pool.1.ssl = false

plugin.rabbitmq.pool.2.host = 192.168.100.121
plugin.rabbitmq.pool.2.port = 61613
plugin.rabbitmq.pool.2.user = mc_rabbitmq
plugin.rabbitmq.pool.2.password = 123.com
plugin.rabbitmq.pool.2.ssl = false

# Facts
factsource = yaml
plugin.yaml = /etc/mcollective/facts.yaml
```

## 2. 配置 mcollective server 端

---

```
[root@linux57poc tmp]# cat /etc/mcollective/server.cfg
# --Global--
topicprefix = /topic/
main_collective = mcollective
collectives = mcollective
libdir = /usr/libexec/mcollective
logfile = /var/log/puppet/mcollective.log
loglevel = info
daemonize = 1

# --rabbitmq Plugins--
securityprovider = psk
plugin.psk = a36cd839414370e10fd281b8a38a4f48

direct_addressing = 1
connector = rabbitmq
plugin.rabbitmq.vhost = /mcollective
plugin.rabbitmq.pool.size = 2
plugin.rabbitmq.initial_reconnect_delay = 0.01
plugin.rabbitmq.max_reconnect_delay = 30.0
plugin.rabbitmq.use_exponential_back_off = true
plugin.rabbitmq.back_off_multiplier = 2
plugin.rabbitmq.max_reconnect_attempts = 0
```

```
plugin.rabbitmq.randomize = false
plugin.rabbitmq.timeout = -1

plugin.rabbitmq.pool.1.host = 192.168.100.120
plugin.rabbitmq.pool.1.port = 61613
plugin.rabbitmq.pool.1.user = mc_rabbitmq
plugin.rabbitmq.pool.1.password = 123.com
plugin.rabbitmq.pool.1.ssl = false

plugin.rabbitmq.pool.2.host = 192.168.100.121
plugin.rabbitmq.pool.2.port = 61613
plugin.rabbitmq.pool.2.user = mc_rabbitmq
plugin.rabbitmq.pool.2.password = 123.com
plugin.rabbitmq.pool.2.ssl = false

# --Puppet provider specific options--
plugin.service.provider = puppet
plugin.service.puppet.hasstatus = true
plugin.service.puppet.hasrestart = true

plugin.puppet.command = puppet agent
plugin.puppet.splay = true
plugin.puppet.splaylimit = 30
plugin.puppet.config = /etc/puppet/puppet.conf

# --Facts--
factssource = yaml
##factssource = facter
plugin.yaml = /etc/mcollective/facts.yaml
```

## 三、高可用测试

**特别注意：** 节点 mcollective 的 server.cfg 中 pool 是有优先级的，默认数字小的生效，这点需要注意，也就是说当所有节点都连接在 MQ2 上的时候，启动 MQ1，mco 命令是无法使用的，因为它在运行的时候连接的是 MQ1，而所有节点都连接在 MQ2 上。

### 1. 停止 MQ1，查看切换状态

#### 1.1 先看当前的节点连接状态

```
[root@linuxmaster1poc ~]# mco ping    #查看连接的节点
linux57poc                            time=69.46 ms
linux58poc                            time=70.05 ms
```

```
linux64poc                                time=70.59 ms

---- ping statistics ----
3 replies max: 70.59 min: 69.46 avg: 70.03
[root@linuxmaster1poc ~]# mco shell "lsof -i:61613" #查看所有节点监听的端口情况，可以看到目前都连接在 linuxmaster1poc 上。
Do you really want to send this command unfiltered? (y/n): y
Discovering hosts using the mc method for 2 second(s) .... 3
Host: linux64poc
Statuscode: 0
Output:
COMMAND  PID USER  FD  TYPE DEVICE SIZE/OFF NODE NAME
ruby     36625 root   6u  IPv4  27771      0t0  TCP
linux64poc:40493->linuxmaster1poc:61613 (ESTABLISHED)
Host: linux58poc
Statuscode: 0
Output:
COMMAND  PID USER  FD  TYPE DEVICE SIZE/OFF NODE NAME
ruby     11060 root   6u  IPv4  34046      0t0  TCP
linux58poc:36295->linuxmaster1poc:61613 (ESTABLISHED)
Host: linux57poc
Statuscode: 0
Output:
COMMAND  PID USER  FD  TYPE DEVICE SIZE NODE NAME
ruby     18076 root   6u  IPv4  1351365      TCP
linux57poc:24698->linuxmaster1poc:61613 (ESTABLISHED)

[root@linuxmaster1poc ~]# /etc/init.d/rabbitmq-server stop
Stopping rabbitmq-server: rabbitmq-server.
```

## 1.2 再次运行 mco 查看切换状态

```
[root@linuxmaster1poc ~]# mco ping
linux58poc                                time=73.54 ms
linux64poc                                time=74.61 ms
linux57poc                                time=75.39 ms

---- ping statistics ----
3 replies max: 75.39 min: 73.54 avg: 74.51
[root@linuxmaster1poc ~]# mco shell "lsof -i:61613"
Do you really want to send this command unfiltered? (y/n): y
Discovering hosts using the mc method for 2 second(s) .... 3
```



```
Host: linux58poc
Statuscode: 0
Output:
COMMAND  PID USER  FD   TYPE DEVICE SIZE/OFF NODE NAME
ruby     11060 root   6u   IPv4  34046      0t0  TCP
linux58poc:36295->linuxmaster1poc:61613 (CLOSE_WAIT)
ruby     11060 root   9u   IPv4  34137      0t0  TCP
linux58poc:47200->linuxmaster2poc:61613 (ESTABLISHED)
Host: linux64poc
Statuscode: 0
Output:
COMMAND  PID USER  FD   TYPE DEVICE SIZE/OFF NODE NAME
ruby     36625 root   6u   IPv4  27771      0t0  TCP
linux64poc:40493->linuxmaster1poc:61613 (CLOSE_WAIT)
ruby     36625 root   8u   IPv4  27877      0t0  TCP
linux64poc:37472->linuxmaster2poc:61613 (ESTABLISHED)
Host: linux57poc
Statuscode: 0
Output:
COMMAND  PID USER  FD   TYPE DEVICE SIZE NODE NAME
ruby     18076 root   9u   IPv4 1351484      TCP
linux57poc:9309->linuxmaster2poc:61613 (ESTABLISHED)
```

通过日志查看

```
[root@linuxmaster1poc ~]# mco shell "lsof -i:61613"
Do you really want to send this command unfiltered? (y/n): y
Discovering hosts using the mc method for 2 second(s) .... 3
Host: linux58poc
Statuscode: 0
Output:
COMMAND  PID USER  FD   TYPE DEVICE SIZE/OFF NODE NAME
ruby     11428 root   6u   IPv4  34283      0t0  TCP
linux58poc:36300->linuxmaster1poc:61613 (CLOSE_WAIT)
ruby     11428 root   8u   IPv4  34338      0t0  TCP
linux58poc:47205->linuxmaster2poc:61613 (ESTABLISHED)
Host: linux57poc
Statuscode: 0
Output:
COMMAND  PID USER  FD   TYPE DEVICE SIZE NODE NAME
ruby     18447 root   6u   IPv4 1351559      TCP
linux57poc:59343->linuxmaster1poc:61613 (CLOSE_WAIT)
ruby     18447 root   8u   IPv4 1351622      TCP
linux57poc:29757->linuxmaster2poc:61613 (ESTABLISHED)
```

```
Host: linux64poc
Statuscode: 0
Output:
COMMAND  PID USER  FD   TYPE DEVICE SIZE/OFF NODE NAME
ruby    37054 root   4u   IPv4  28036      0t0  TCP
linux64poc:37476->linuxmaster2poc:61613 (ESTABLISHED)
ruby    37054 root   6u   IPv4  27990      0t0  TCP
linux64poc:40497->linuxmaster1poc:61613 (CLOSE_WAIT)
```

总结：可以看到之前的连接已经变成 CLOSE\_WAIT，新的连接被建立

## 2. 停止 MQ2，启动 MQ1 查看切换状态

```
[root@linuxmaster2poc rabbitmq]# /etc/init.d/rabbitmq-server stop
Stopping rabbitmq-server: rabbitmq-server.
```

```
[root@linux57poc service]# lsof -i:61613
COMMAND  PID USER  FD   TYPE DEVICE SIZE NODE NAME
ruby    18447 root   6u   IPv4 1351559      TCP
linux57poc:59343->linuxmaster1poc:61613 (CLOSE_WAIT)
ruby    18447 root   8u   IPv4 1351622      TCP
linux57poc:29757->linuxmaster2poc:61613 (CLOSE_WAIT)
[root@linux58poc ~]# lsof -i:61613
COMMAND  PID USER  FD   TYPE DEVICE SIZE/OFF NODE NAME
ruby    11428 root   6u   IPv4  34283      0t0  TCP
linux58poc:36300->linuxmaster1poc:61613 (CLOSE_WAIT)
ruby    11428 root   8u   IPv4  34338      0t0  TCP
linux58poc:47205->linuxmaster2poc:61613 (CLOSE_WAIT)
[root@linux64poc ~]# lsof -i:61613
COMMAND  PID USER  FD   TYPE DEVICE SIZE/OFF NODE NAME
ruby    37054 root   4u   IPv4  28036      0t0  TCP
linux64poc:37476->linuxmaster2poc:61613 (CLOSE_WAIT)
ruby    37054 root   6u   IPv4  27990      0t0  TCP
linux64poc:40497->linuxmaster1poc:61613 (CLOSE_WAIT)
```

```
[root@linuxmaster1poc ~]# /etc/init.d/rabbitmq-server start
Starting rabbitmq-server: SUCCESS
rabbitmq-server.
```

根据 `plugin.rabbitmq.maxreconnectdelay = 30.0`，需要过最多 30 秒，mcollective 服务端会重新建立连接请求

```
[root@linuxmaster1poc ~]# tailf /var/log/rabbitmq/rabbit@linuxmaster1poc.log
=INFO REPORT==== 24-Dec-2013::11:00:45 ==
```

```
accepting STOMP connection <0.332.0> (192.168.100.126:36316 ->
192.168.100.120:61613)
```

```
=INFO REPORT==== 24-Dec-2013::11:00:45 ===
```

```
accepting STOMP connection <0.348.0> (192.168.100.125:18945 ->
192.168.100.120:61613)
```

```
=INFO REPORT==== 24-Dec-2013::11:00:45 ===
```

```
accepting STOMP connection <0.382.0> (192.168.100.127:40513 ->
192.168.100.120:61613)
```

```
[root@linuxmaster1poc ~]# mco ping
```

```
linux58poc                time=70.60 ms
linux57poc                time=71.32 ms
linux64poc                time=111.56 ms
```

```
---- ping statistics ----
```

```
3 replies max: 111.56 min: 70.60 avg: 84.49
```

```
[root@linuxmaster1poc ~]# mco shell "lsof -i:61613"
```

```
Do you really want to send this command unfiltered? (y/n): y
```

```
Discovering hosts using the mc method for 2 second(s) .... 3
```

```
Host: linux58poc
```

```
Statuscode: 0
```

```
Output:
```

```
COMMAND  PID USER  FD   TYPE DEVICE SIZE/OFF NODE NAME
ruby     11428 root   6u   IPv4  34283      0t0  TCP
linux58poc:36300->linuxmaster1poc:61613 (CLOSE_WAIT)
ruby     11428 root   8u   IPv4  34338      0t0  TCP
linux58poc:47205->linuxmaster2poc:61613 (CLOSE_WAIT)
ruby     11428 root  10u  IPv4  34444      0t0  TCP
linux58poc:36316->linuxmaster1poc:61613 (ESTABLISHED)
```

```
Host: linux57poc
```

```
Statuscode: 0
```

```
Output:
```

```
COMMAND  PID USER  FD   TYPE DEVICE SIZE NODE NAME
ruby     18447 root  10u  IPv4 1351723      TCP
linux57poc:18945->linuxmaster1poc:61613 (ESTABLISHED)
```

```
Host: linux64poc
```

```
Statuscode: 0
```

```
Output:
```

```
COMMAND  PID USER  FD   TYPE DEVICE SIZE/OFF NODE NAME
```

```
ruby    37054 root    4u  IPv4  28036      0t0  TCP
linux64poc:37476->linuxmaster2poc:61613 (CLOSE_WAIT)
ruby    37054 root    6u  IPv4  27990      0t0  TCP
linux64poc:40497->linuxmaster1poc:61613 (CLOSE_WAIT)
ruby    37054 root    9u  IPv4  28206      0t0  TCP
linux64poc:40513->linuxmaster1poc:61613
```

---

为了能够和大家更好的交流和学习 Puppet，本人 2014 年又新开辟了微信公众号进行交流学习，目前已经有 300 多人同时收听，喜欢 Puppet 的大神们可自行加入哦。

如果你有好的有关 Puppet 的咨询也可以给我投稿，投稿邮箱：  
[admin@kisspuppet.com](mailto:admin@kisspuppet.com)

微信公众号：“**puppet2014**”，可搜索加入，也可以扫描以下二维码

