

centos 7安装rabbitmq

1. 准备工作

- centos 7及以上操作系统
- 下载[对应的erlang](#)
- 下载对应erlang版本的[rabbitmq安装包](#)
- 配置 centos host的文件

```
vim /etc/hosts
#新增一行
127.0.0.1 主机名
192.168.1.100 主机名  ##其中 192.168.1.100为centos的真实网络ip
```

这里一定要配置否则开启rabbitmq服务时会报无法连接host 超时的错误

- 关联组件包安装

```
#系统升级 防止不必要的错误发生
yum update
#安装 epel-release
yum install epel-release
#批量安装相关组件
yum install gcc gcc-c++ glibc-devel make ncurses-devel openssl-devel autoconf git wget
#安装java 1.8 的jdk 上传rpm 包安装 ( 这里不安装不知道是否有问题 , 以防万一 )
rpm -ivh jdk-1.8.0_1u***x86_64.rpm
```

2. 安装rpm包

- 上传erlang 安装包 并安装

```
rpm -ivh erlang-21.3.8.6-1.el7.x86_64.rpm
```

注意：这里的erlang 版本号一定要与待安装的rabbitmq匹配，具体版本对应信息请参考[官网说明](#)

- 导入密钥【可选】

```
rpm --import https://www.rabbitmq.com/rabbitmq-release-signing-key.asc
```

- 上传rabbitmq安装包并安装

```
rpm -ivh rabbitmq-server-3.7.15-1.el7.noarch.rpm
## 这里安装的是3.7.5版本 对应的erlang版本为>=20.3 <=22.0
## 如果报错 提示 socat 没有安装，执行下面的语句，否则不用执行
yum install socat #安装socat 仅仅有需要的时候安装
```

3. 操作命令

#####rabbitmq-server服务相关命令语句#####

```
#设置rabbitmq 为开机启动
systemctl enable rabbitmq-server.service
#查看rabbitmq 状态
systemctl status rabbitmq-server.service
#打开 rabbitmq 服务
systemctl start rabbitmq-server.service #centos7 及以上系统推荐
rabbitmq-server start #与上一条命令两者任选其一即可
#停止服务
systemctl stop rabbitmq-server.service
```

#####rabbitmq操作相关命令语句#####

```
#查看当前所有用户
rabbitmqctl list_users
#查看默认guest用户权限
rabbitmqctl list_user_permissions guest
#由于rabbitmq 默认的账号用户名和密码都是guest 为了安全起见要先删除guest账号
rabbitmqctl delete_user guest
#添加新用户
rabbitmqctl add_user <username> <password>
#设置用户tag,给新建的用户 administrator权限
rabbitmqctl set_user_tags <username> administrator
#赋予用户默认vhost的全部操作权限
#语法格式为 set_permissions [-p vhost] {user}{conf}{write}{read}
#conf、write、read 权限采用正则表达 ^$: 表示完全不匹配 ( 没有权限 ) .* : 所有权限
rabbitmqctl set_permissions -p / <username> ".*" ".*" ".*"
#查看新增用户权限是否配置成功
rabbitmqctl list_user_permissions <username>
```

更多的rabbitmqctl的使用，可以参考[官方说明文档](#)

```
#开启web管理接口
rabbitmq-plugins enable rabbitmq_management
#默认web管理接口是 15672
#rabbitmq-server 服务端口是25672
#关闭centos防火墙或者配置允许 15672和25672两个端口
```