# RabbitMq

## 下载安装

1. 官网下载<http://www.rabbitmq.com/download.html>
2. 下载 <http://www.rabbitmq.com/releases/erlang/erlang-18.1-1.el6.x86_64.rpm>

# rpm -ihv erlang-18.1-1.el6.x86\_64.rpm

1. rpm -ihv rabbitmq-server-3.6.6-1.el7.noarch.rpm --force -–nodeps

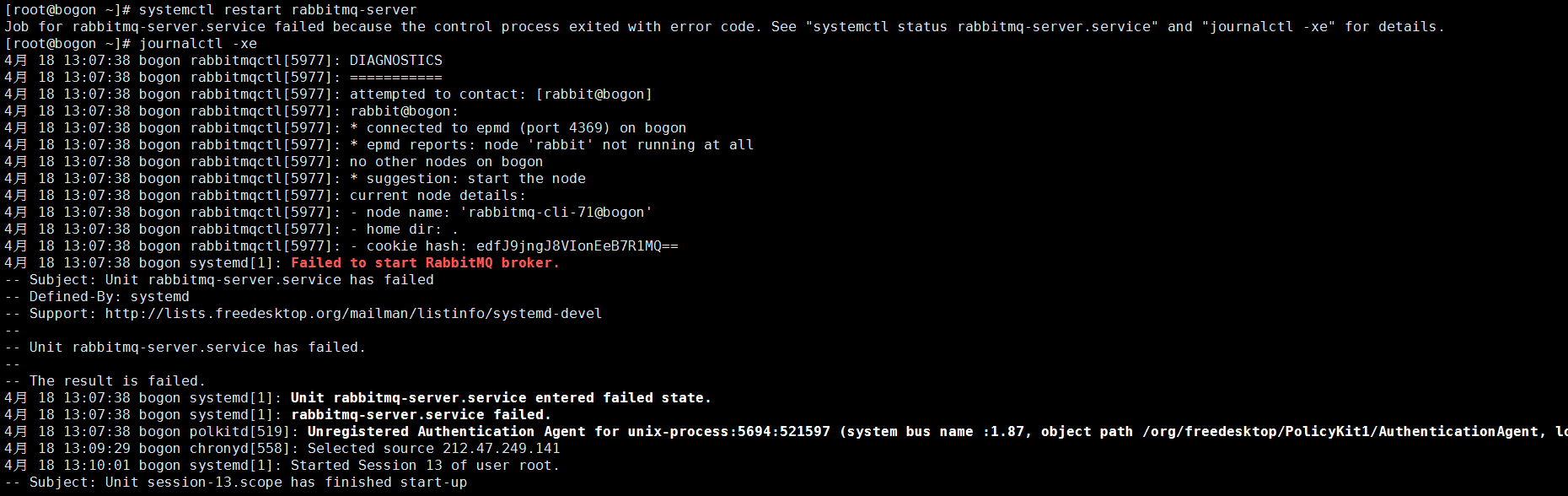
## 启动

### !!!!!!!!这是个非常蛋疼的问题

我一直用 systemctl 操作rabbitmq-server 一直报错

折腾了好几天

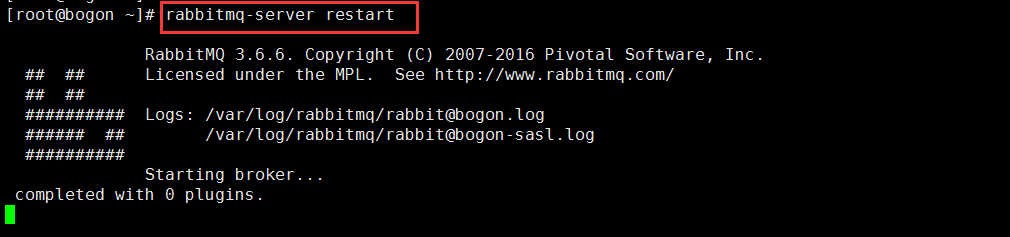
如果启动报错

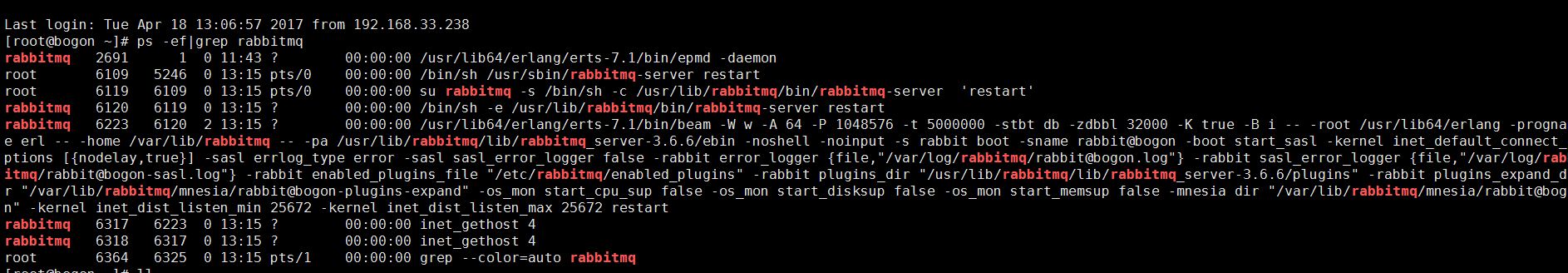


### 蛋疼问题解决

用 rabbitmq-server restart

启动成功





第一件事要创建用户，因为缺省的guest/guest用户只能在本地登录，所以先用命令行创建一个admin/admin123，并让他成为管理员。

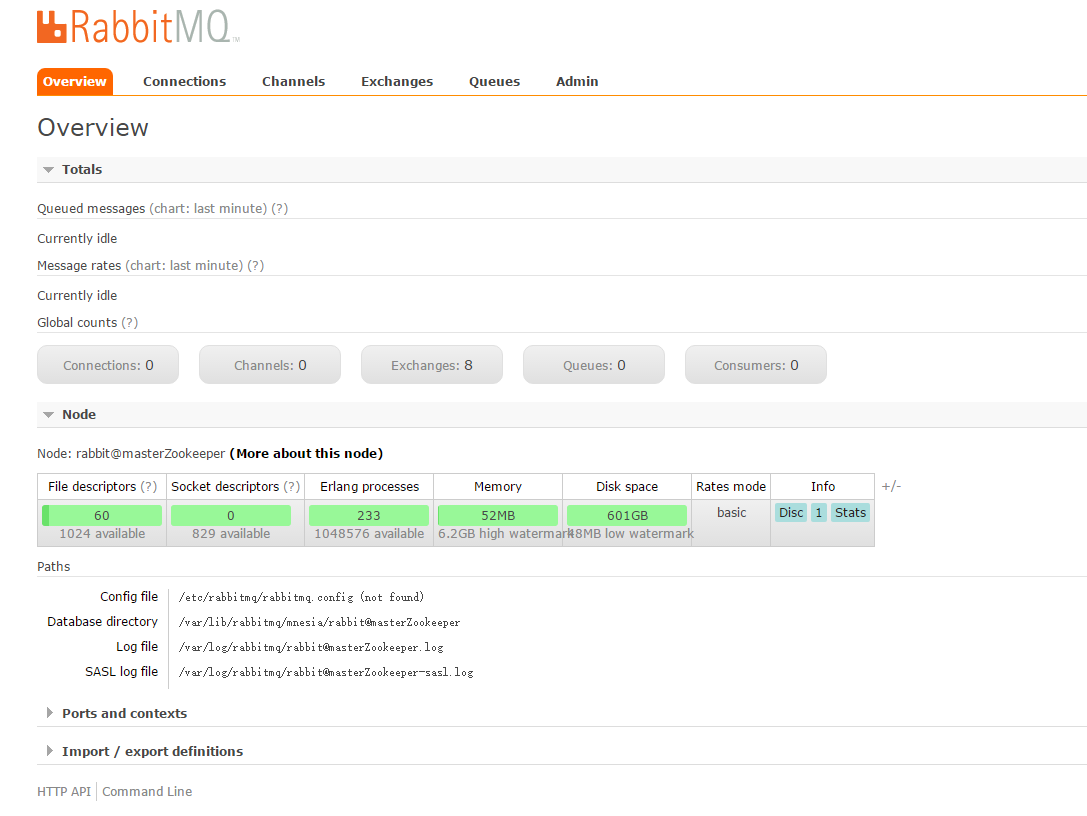
# ./rabbitmqctl add\_user admin admin123

# ./rabbitmqctl set\_user\_tags admin administrator

# ./rabbitmq-plugins enable rabbitmq\_management

记得开放服务器 15672端口

<http://192.168.1.107:15672/#/>



#创建用户

rabbitmqctl add\_user root 123456

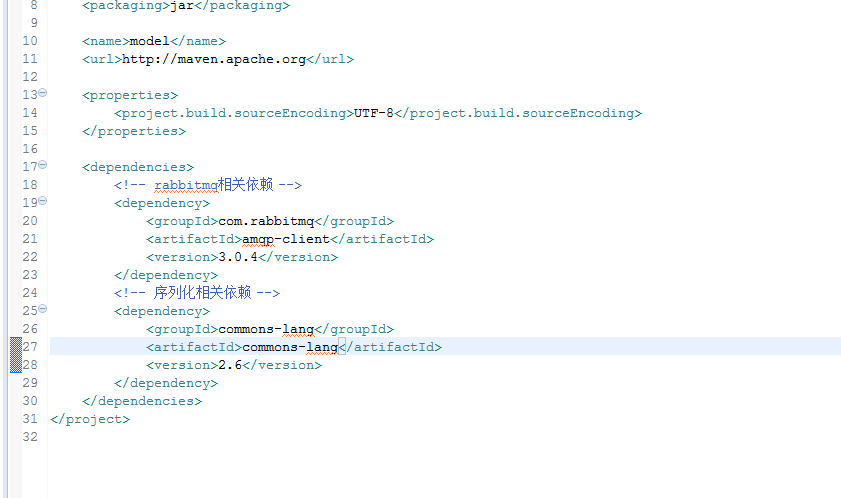
#设置tag

rabbitmqctl set\_user\_tags root administrator

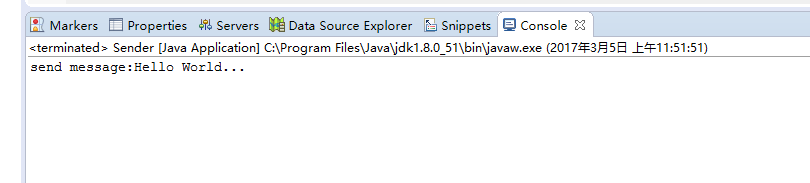
#赋权限

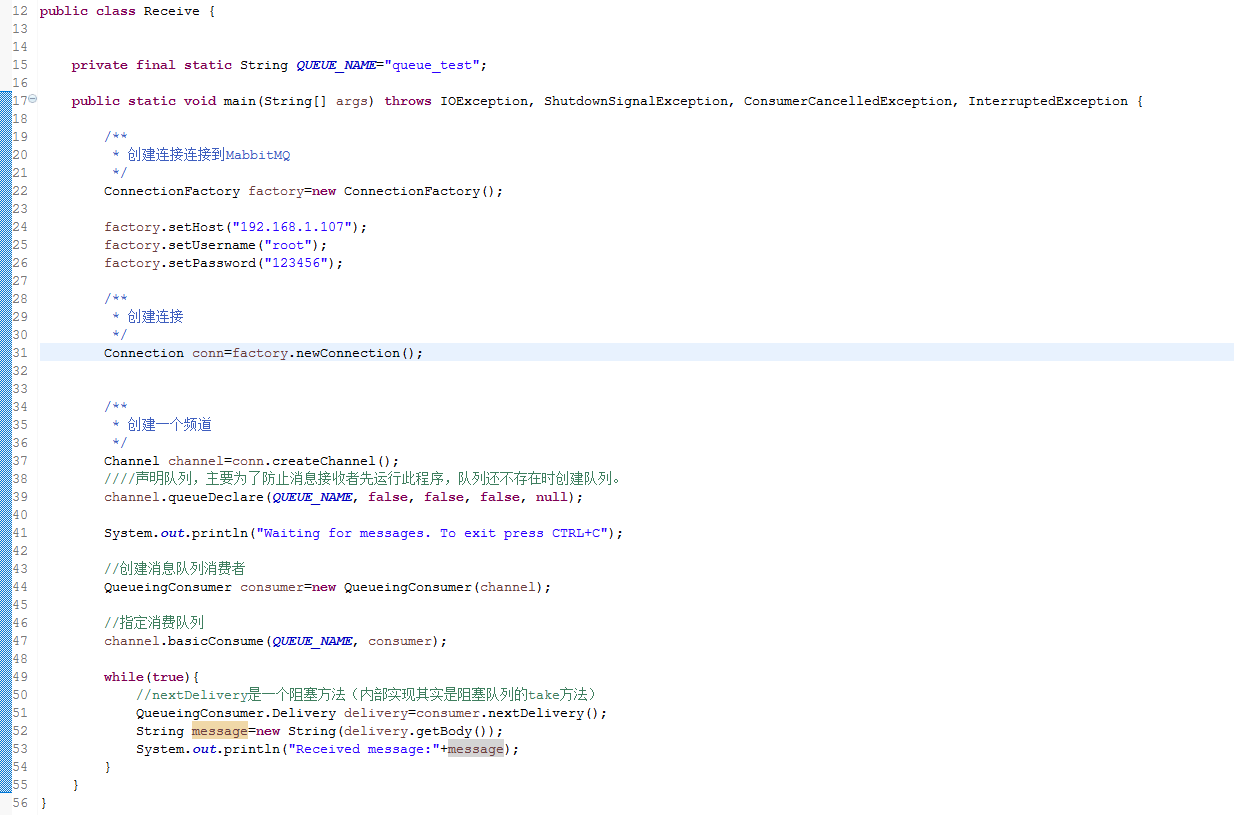
rabbitmqctl set\_permissions -p / root ".\*" ".\*" ".\*"

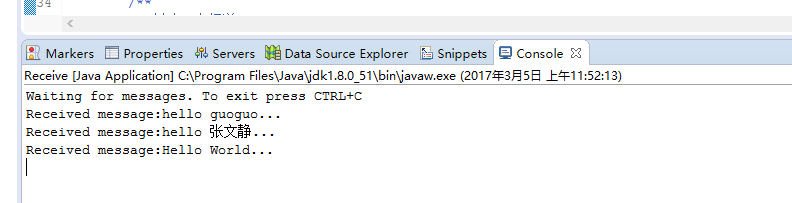
测试情况…











## ACK

acknowledgment 是 consumer 告诉 broker 当前消息是否成功 consume，至于 broker 如何处理 NACK，取决于 consumer 是否设置了 requeue：如果 requeue=False，那么 NACK 后 broker 是会删除消息的。看看 [RabbitMQ 官方的解释](https://link.zhihu.com/?target=https%3A//www.rabbitmq.com/confirms.html)。Consumer 做一个 ACK，是为了告诉 Broker 这条消息已经被成功处理了（transaction committed）。只要没收到 consumer 的 acknowledgment，broker 就会一直保存着这条消息（但不会 requeue，更不会分配给其他 consumer，直到当前 consumer 发生断开连接之类的异常）。RabbitMQ 之所以是 guaranteed delivery，这是一个关键。换言之，你的 consumer 代码必须能够处理各种异常，确保只要收到一条消息，最终一定能够执行一条 ACK / NACK（当然也没人阻止你设置 no\_ack=True，干脆不用 acknowledgment 机制，这个视业务需求而定）