## 课程目标

1. 理解中间件概念
2. 理解消息队列软件Rabbitmq及其应用场景

*注意事项：请使用chrome、360浏览器进行相关操作。*

## 前置条件

准备一台服务器：Ubuntu Server 16.04.1 LTS 64位

## 帮助信息

RabbitMQ是实现了高级消息队列协议（AMQP）的开源消息代理软件（亦称面向消息的中间件）。RabbitMQ服务器是用[Erlang](https://baike.baidu.com/item/Erlang" \t "https://baike.baidu.com/item/rabbitmq/_blank)语言编写的，而集群和故障转移是构建在[开放电信平台](https://baike.baidu.com/item/%E5%BC%80%E6%94%BE%E7%94%B5%E4%BF%A1%E5%B9%B3%E5%8F%B0" \t "https://baike.baidu.com/item/rabbitmq/_blank)框架上的。所有主要的编程语言均有与代理接口通讯的客户端[库](https://baike.baidu.com/item/%E5%BA%93)。本实验通过python开发两个示例应用，分别演示RabbitMQ的工作模式和发布订阅模式。

工作模式（资源的争抢）

 ①生产者将消息发送给交换机

②交换机发送给绑定的后端队列

③一个队列被多个消费者同时监听，形成消息的争抢结构：根据消费者所在的系统的空闲、性能争抢队列中的消息

一些场景：抢红包



发布订阅（交换机类型为fanout）

注：图中未画消费者

①交换机定义类型为：fanout

②交换机绑定多个队列

③生产者将消息发送给交换机，交换机复制同步消息到后端所有的队列中

一些场景：邮件群发



本文代码来源：<https://www.cnblogs.com/xiajq/p/9536421.html>

## 实验步骤

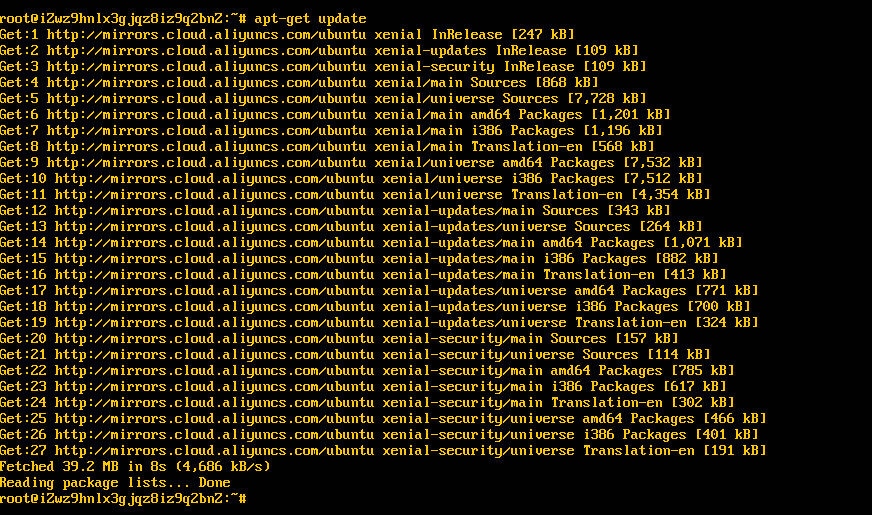
### 安装Rabbitmq

#### ****118479****

这里我们使用apt-get来安装，就简单的几条命令：

1. 更新软件包

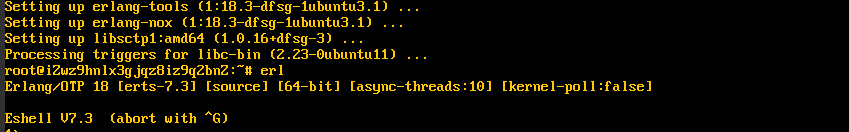
apt-get update



1. 由于rabbitMq需要erlang语言的支持，在安装rabbitMq之前需要安装erlang，执行命令：

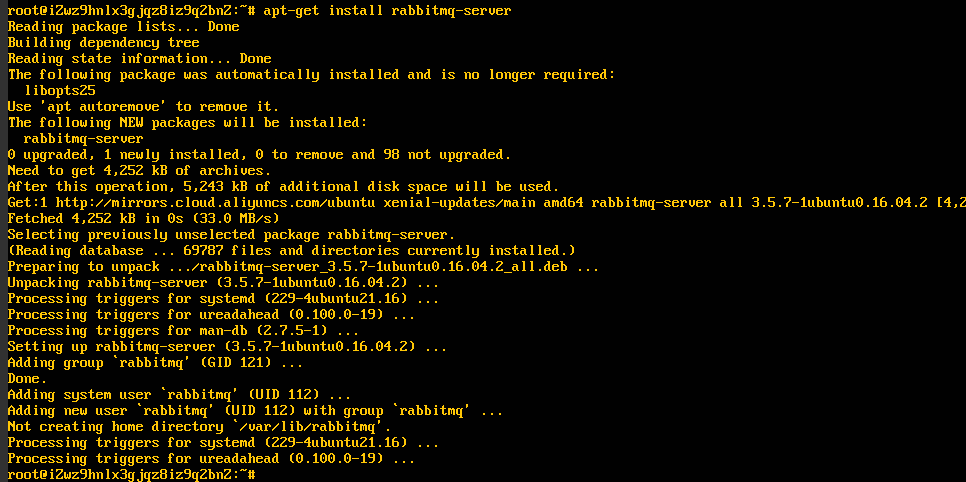
apt-get install erlang-nox # 安装erlang

erl # 查看relang语言版本，成功执行则说明relang安装成功



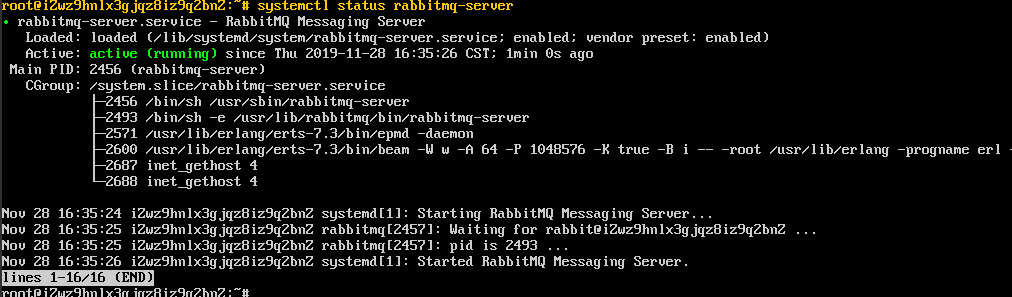
1. 安装 RabbitMQ

apt-get install rabbitmq-server #安装成功自动启动



1. 查看 RabbitMq状态

systemctl status rabbitmq-server #Active: active (running) 说明处于运行状态



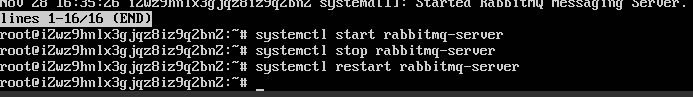
1. 启动、停止、重启

systemctl start rabbitmq-server # 启动

systemctl stop rabbitmq-server # 停止

systemctl restart rabbitmq-server # 重启

执行了上面的步骤，rabbitMq已经安装成功。

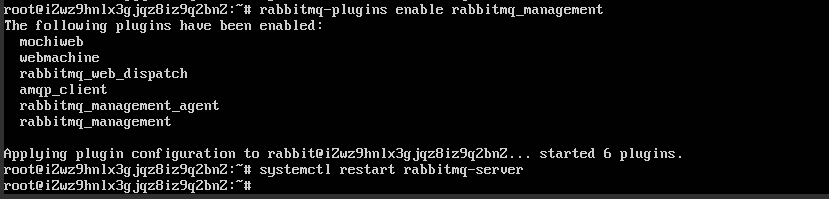


### 启用 web端可视化操作界面

1. 启用 RabbitMQ web 管理插件

rabbitmq-plugins enable rabbitmq\_management

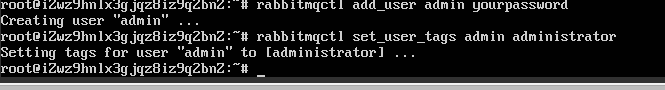
systemctl restart rabbitmq-server



1. 添加管理用户

rabbitmqctl add\_user admin yourpassword # 增加普通用户

rabbitmqctl set\_user\_tags admin administrator # 给普通用户分配管理员角色

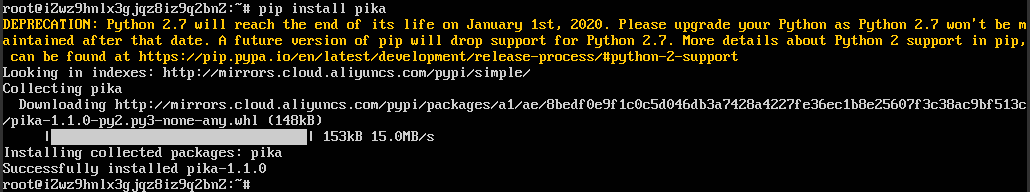


在你的浏览器上输入：http://119.23.200.239:15672/ 来访问你的rabbitmq监控页面。使用刚刚添加的新用户登录。

### 安装python pika包，用于实现软件开发。

1. 执行下列命令安装pika包

pip install pika



1. 复制下列代码文件到Ubuntu上



### 测试消息队列模式和发布订阅方式的区别

1. 队列模式执行任务，打开3个窗口，在其中两个窗口执行python consumer.py等待接收消息



在第三个窗口执行两次python producer.py，查看另外两个窗口是否接收到消息。



1. 发布订阅方式执行任务，打开3个窗口，在其中两个窗口执行python consumer.py等待接收消息，在第三个窗口执行两次python producer.py，查看另外两个窗口是否接收到消息。

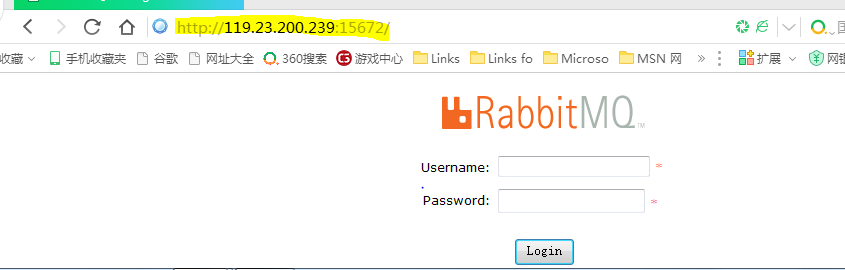


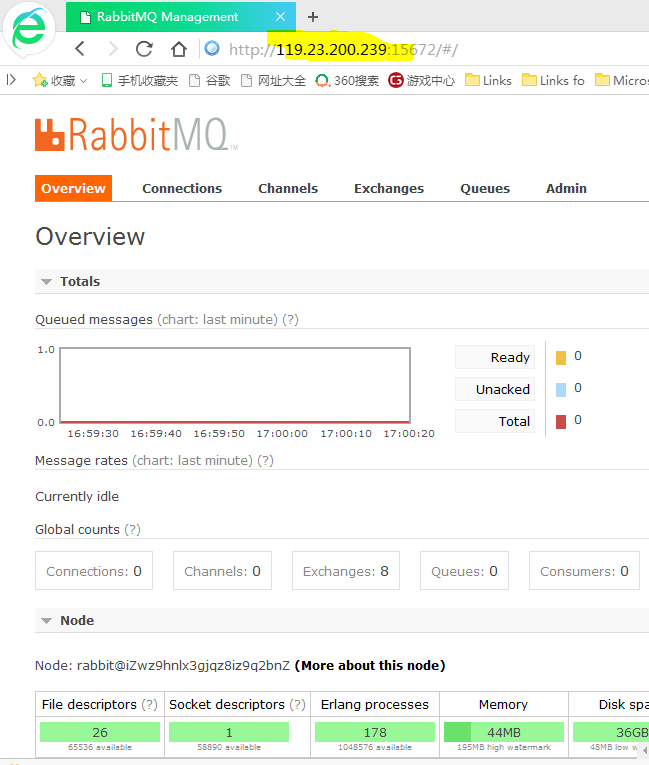


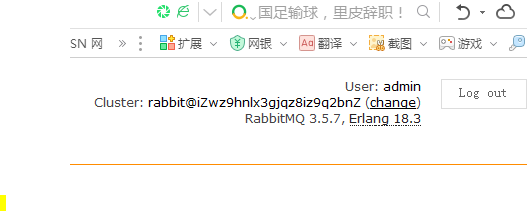
## 作业提交要求：

提交要求：相应结果进行截图，整合在一个word文档，提交到FTP。相应截图示例如下：

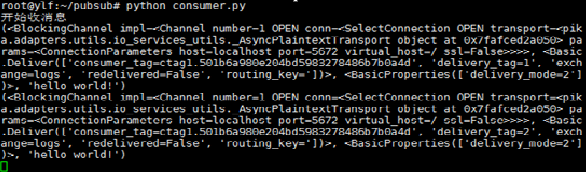
1. RabbitMQ的web管理界面截图



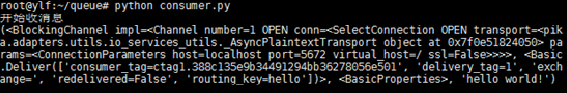




1. 发布订阅方式消费者执行结果截图



1. 队列方式消费者执行结果截图



## 练习

一、在一个复杂的分布式系统中，假设有n个组件，每个组件互相都需要进行通信。请问：

1. 组件之间的通信路径有几条？
2. 引入消息中间件技术简化通信过程，则通信路径变为几条？
3. RabbitMQ是一种什么软件（）
4. 系统软件 B.应用软件 C.中间件 D.数据库软件
5. Ubuntu系统用于安装软件的命令是（）

A.apt-get B.pip C.systemctl D.update

1. 使用阿里云ECS安装完RabbiMQ但是无法通过web界面登陆其端口15672的可能原因是什么?

判断：RabbitMQ发布订阅模式中，一个消息发布后只会被一个消费者获取到。

判断：RabbitMQ消息队列模式（资源争抢模式）中，一个消息发布后只会被一个消费者获取到。