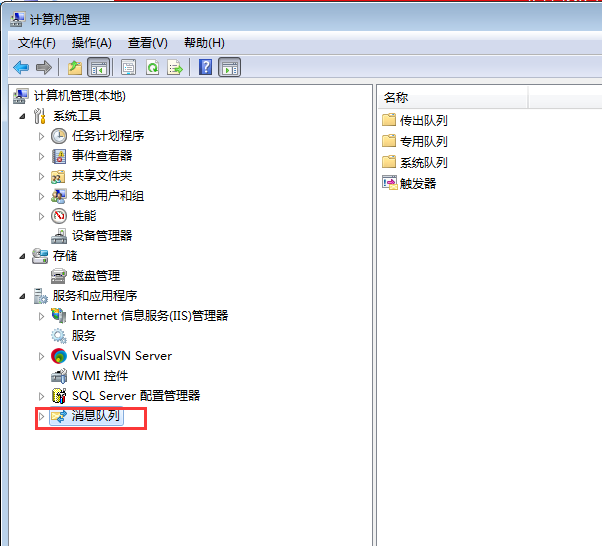
# 1.了解RabbitMQ

RabbitMq是MQ的一个产品为消息队列

## 1.1.我们接触过得第一个消息队列

我们接触过的第一个消息队列为windows自带的一个服务MSMQ

在“计算机”->右击“管理”



这是最原始的消息队列，message存放在文件系统中，不具有现在消息队列【集群，消息确认，内存化，高可用，镜像】等特性。是八九十年代的产品。

## 1.2.现在我们比较流行消息队列

ActiveMQ，ZeroMQ，RabbitMQ，RocketMQ

## 1.3．Rabbitmq本质是一个什么东西？

《1》 rabbitmq是用什么语言写的 => erlang

《2》 rabbitmq其实是遵循 amqp协议 的一个erlang代码实现。

## 1.4. 什么是Erlang

Erlang(['ə:læŋ])是一种通用的面向并发的编程语言，它由瑞典电信设备制造商爱立信所辖的CS-Lab开发，目的是创造一种可以应对大规模并发活动的编程语言和运行环境

## 1.5.RabbitMq遵循的协议

我们所知道的协议有：

http协议， 【request，response】

telnet协议： 经常用来查看某一台ip上指定端口是否是ping通的。。【远程登陆协议】

rabbitMq遵循的协议是amqp协议: 高级消息队列协议。

rabbitmq实现的ampq协议遵循的版本号：AMQP 0-9-1 Complete Reference Guide

协议的内容：

1. connection连接时遵循open,use,close协议，实现的方法open-ok,close,une-ok
2. channel 流通时遵循open,flow,close协议，构建在connection之上，在amqp中常作为长连接
3. exchange 交换机
4. queue 队列
5. basic 发布 和 获取 message中的一些设置
6. tx 事务处理
7. confirm 发布确认机制

amqp协议就像设计书一样。它的实现语言可以为c#，java

## 1.6. rabbitmq的元素

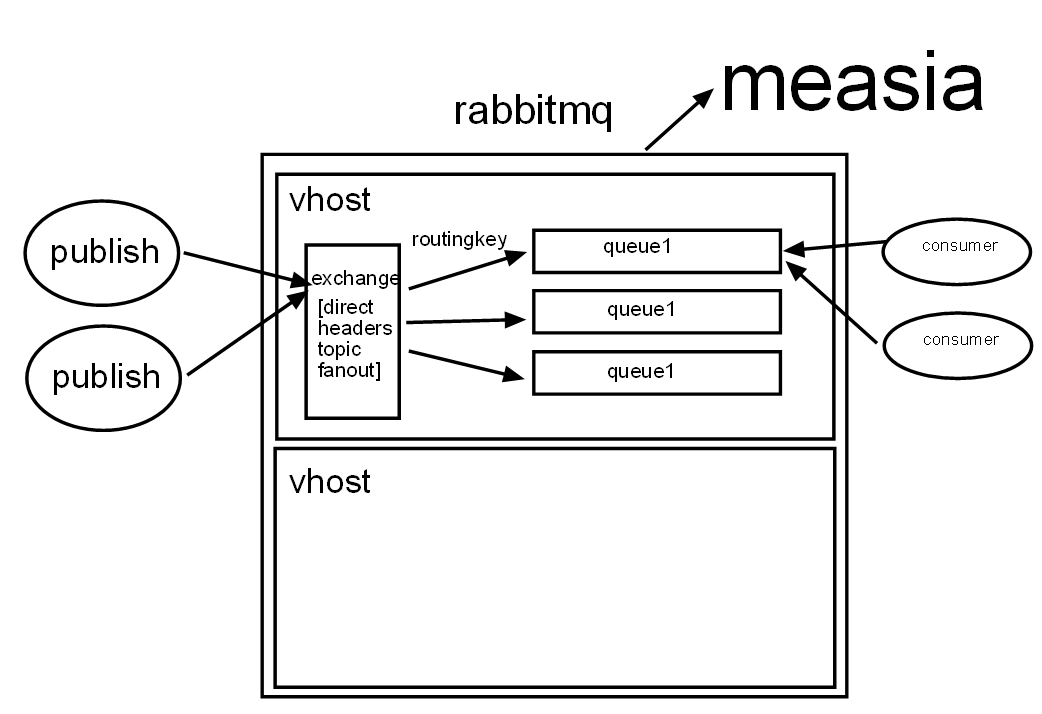
vhost： 避免了大家的命名冲突。

Exchange交换机包含的类型：direct（直连），headers（头部），topic（主题），fanout（分列）

Publicsh 发布者

queue 队列

Consumer 消费者



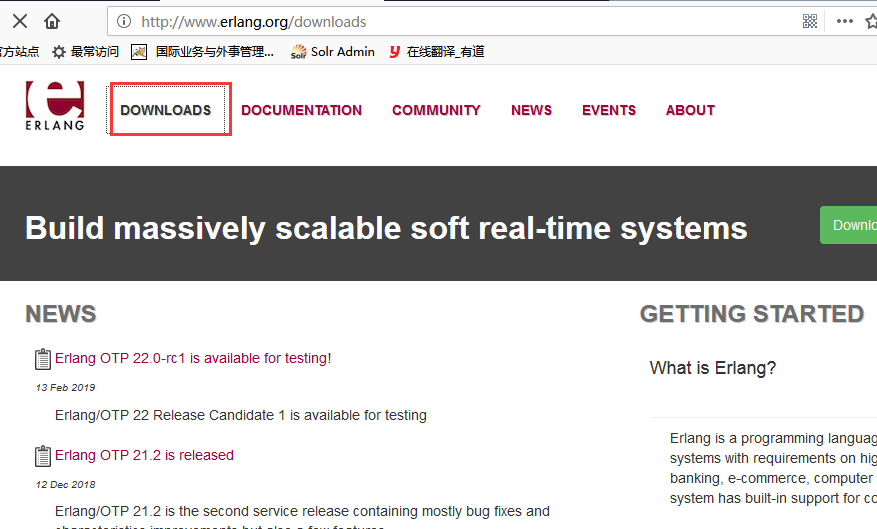
# 2.下载与安装

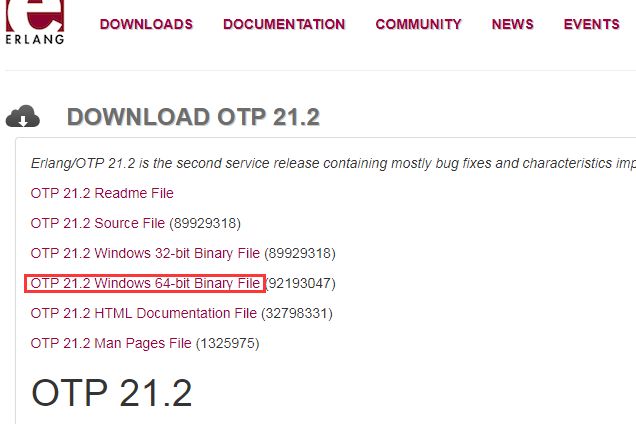
## 2.1.Window环境

### 2.1.1.下载

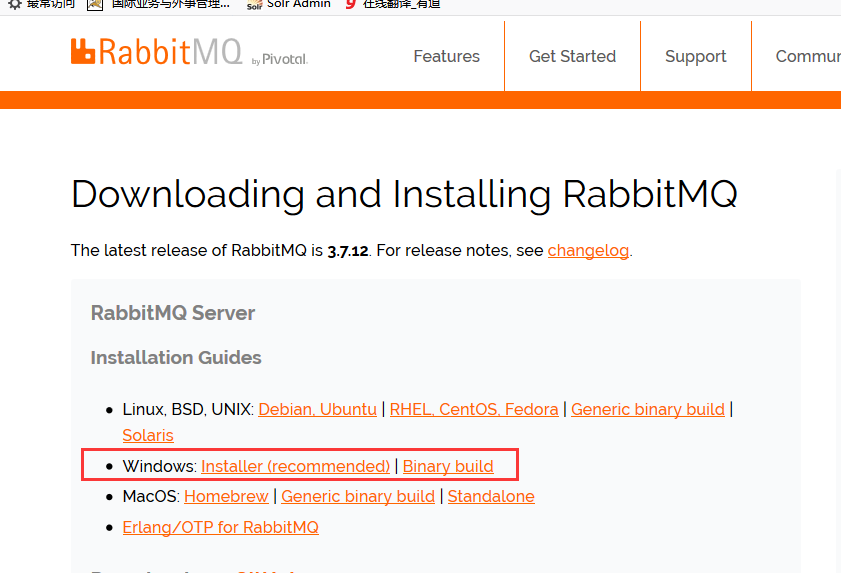
注意erlang的版本与radditMq版本匹配 可以先下载radditMq 安装时会提示安装erlang

由于rabbitmq的编写语言为erlang，所以先下载erlang。地址<http://www.erlang.org/>





下载rabbitmq 地址：<http://www.rabbitmq.com/download.html>



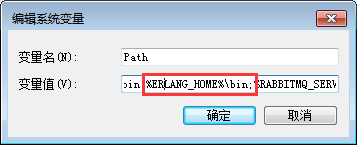
### 2.2.2. 安装erlang

《1》 安装 opt 机械化安装

### 2.2.3. 配置erlang环境变量

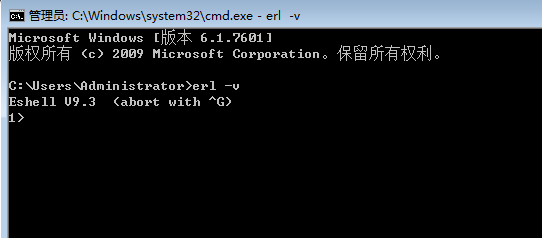
设置环境变量ERLANG\_HOME= D:\rabbitMq\erl9.3   
      添加到PATH  %ERLANG\_HOME%\bin;





测试

打开cmd 输入命令 erl –v 或erl –version



这样就是成功了

### 2.2.4.安装rabbitmq

《2》 安装 rabbitMq 机械化安装

如果安装错误 卸载去控制面板里去卸载

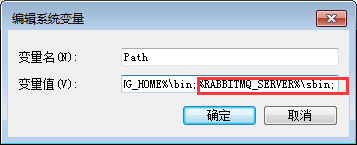
### 2.2.5.配置rabbitmq环境变量

设置环境变量的目的：

1. 在cmd环境中执行rabbitMq命令时不需要目录定位到sbin目录下。

设置环境变量RABBITMQ\_SERVER=D:\rabbitMq\RabbitMQ Server\rabbitmq\_server-3.7.4  
      添加到PATH %RABBITMQ\_SERVER%\sbin;

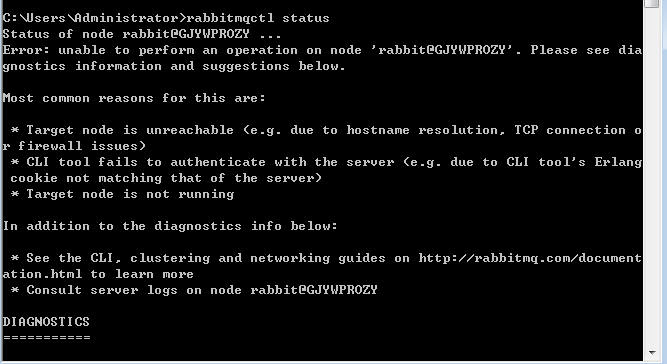


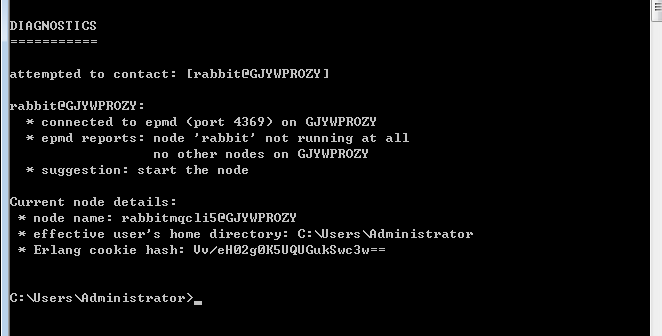


测试：

cmd  --->  rabbitmqctl status  查看环境变量是否配置正确

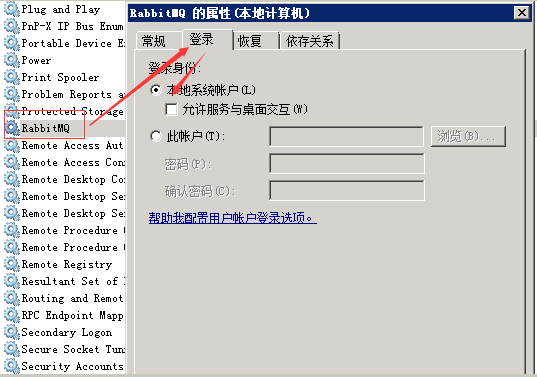
这时出现错误





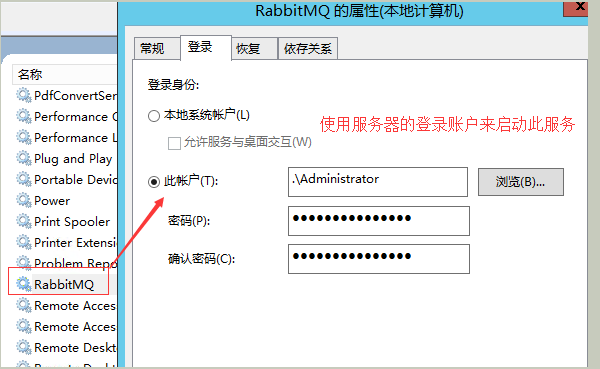
登录账号权限问题

后来看到一个网友说可能是登录账号的问题，我是用administrator账号登录服务器并安装rabbitmq的，rabbitmq也自动在C：\user\administrator下创建了一个.erlang.cookie文件，但是RabbitMQ服务启动的时候，执行者是local system account，如下图所示：

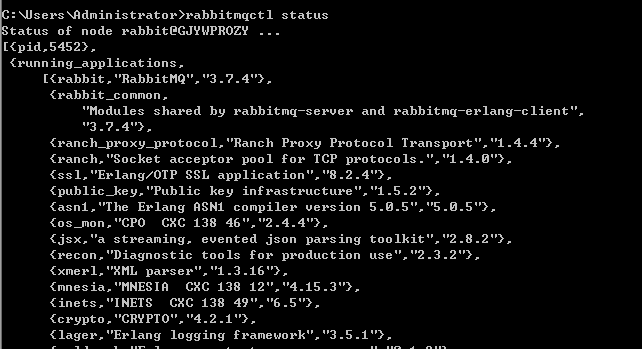


此时，我想到将RabbitMQ服务的启动账号改成administrator，

右击--属性—登录 注意密码与确认密码为 开机密码。（解屏密码）密码不能为空。



更改后重启服务，再次执行rabbitmqctl status，

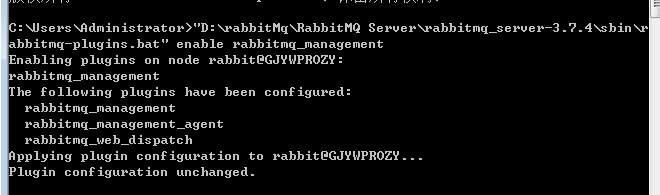


成功了！

### 2.2.6. RabbitMQ Web插件安装

《3》 安装web监控插件【可视化的去监视我们的rabbitmq】

输入命令：**rabbitmq-plugins.bat enable rabbitmq\_management**来执行安装

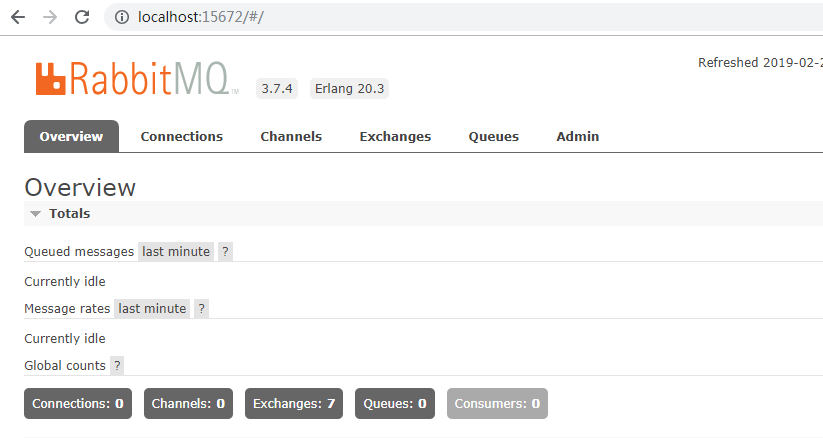


-- 至此安装过程均已完成，下面就可以使用刚创建的账户登录进行管理了

### 2.2.7.访问管理页面

管理页面：http://127.0.0.1:15672

默认登录用户：guest; 密码：guest



## 2.2. Linux环境

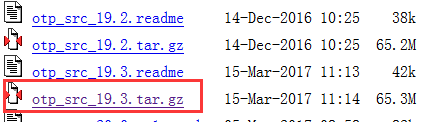
### 2.2.1.环境介绍

1、系统版本：centos 6.5

Erlang与rabbitMq版本对照（<http://www.rabbitmq.com/which-erlang.html>）

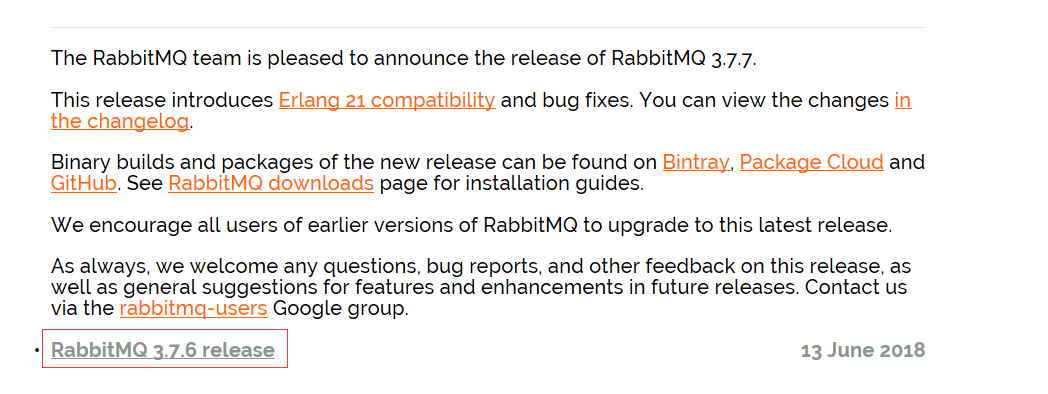
2、Erlang版本：otp\_src\_19.3.tar.gz

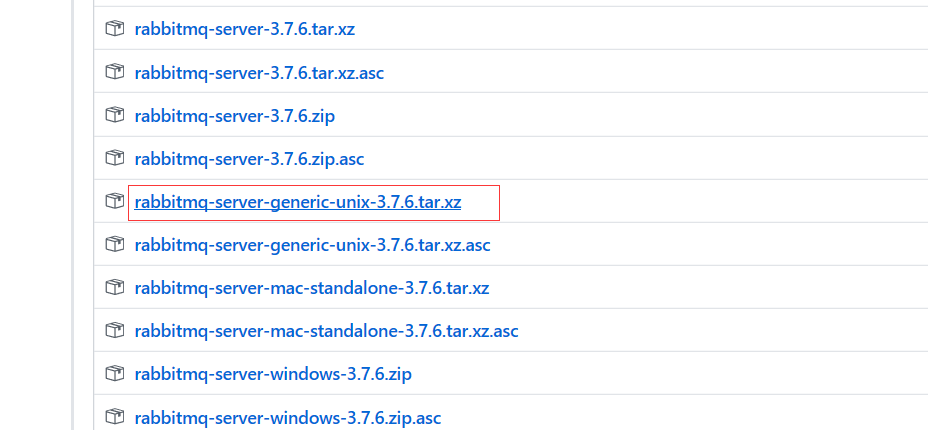
寻找资源（<http://erlang.org/download/>）



3、Rabbitmq版本：rabbitmq-server-generic-unix-3.7.6.tar.xz

寻找资源（<http://www.rabbitmq.com/news.html>）





路径为：

https://github.com/rabbitmq/rabbitmq-server/releases/tag/v3.7.6/rabbitmq-server-generic-unix-3.7.6.tar.xz

### 2.2.2.安装erlang

#### 2.2.2.1.安装依赖环境

依赖环境的安装-如果需要用编译安装erlang语言环境，需要安装C++编译。

**1、在有网的环境下执行**

|  |
| --- |
| yum -y install make gcc gcc-c++ kernel-devel m4 ncurses-devel openssl-devel unixODBC unixODBC-devel httpd python-simplejson |

**2、在没网的环境下执行（注意centos的版本为6.5）**

（1）安装gcc gcc-c++

下载下面rpm包：[http://vault.centos.org/6.5/os/x86\_64/Packages/](http://vault.centos.org/6.5/os/x86_64/Packages/" \t "_blank)

依次安装：

|  |
| --- |
| rpm -Uvh kernel-headers-2.6.32-431.el6.x86\_64.rpm  rpm -Uvh glibc-headers-2.12-1.132.el6.x86\_64.rpm  rpm -Uvh glibc-devel-2.12-1.132.el6.x86\_64.rpm  rpm -Uvh libgomp-4.4.7-4.el6.x86\_64.rpm  rpm -Uvh ppl-0.10.2-11.el6.x86\_64.rpm  rpm -Uvh cloog-ppl-0.15.7-1.2.el6.x86\_64.rpm  rpm -Uvh mpfr-2.4.1-6.el6.x86\_64.rpm  rpm -Uvh cpp-4.4.7-4.el6.x86\_64.rpm  rpm -Uvh gcc-4.4.7-4.el6.x86\_64.rpm  rpm -Uvh libstdc++-devel-4.4.7-4.el6.x86\_64.rpm  rpm -Uvh gcc-c++-4.4.7-4.el6.x86\_64.rpm  怕麻烦的话，可以这样一次过：**rpm -Uvh \*.rpm –nodeps –force** |

（2）安装Perl

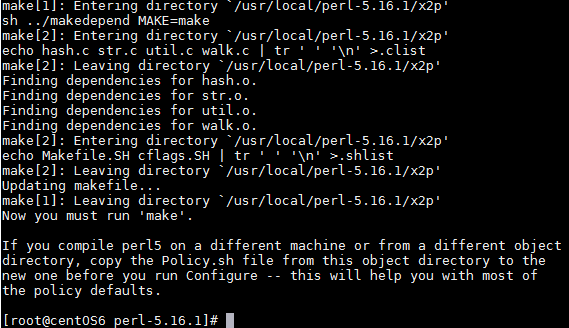
下载路径<http://www.cpan.org/src/5.0/perl-5.16.1.tar.gz>

上传到linux上

解压缩包：tar –zxvf perl-5.16.1.tar.gz –C /usr/local/

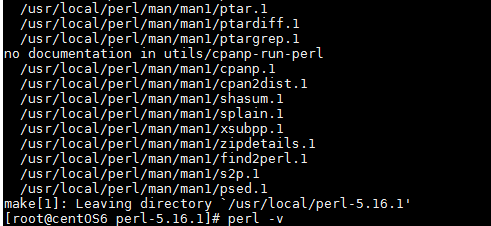
切换到解压包目录：cd perl-5.16.1

安装perl：./Configure -des -Dprefix=/usr/local/perl

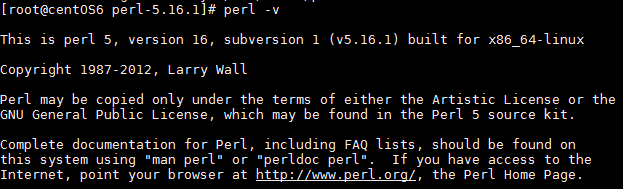


完成后输入命令：make && make test && make install

（注意：这需要很长一段时间进行安装）



完成后输入：perl -v  查看是否安装成功



（3）安装ncurses-devel

下载下面rpm包：[http://vault.centos.org/6.5/os/x86\_64/Packages/](http://vault.centos.org/6.5/os/x86_64/Packages/" \t "_blank)

依次安装：

|  |
| --- |
| rpm -Uvh [ncurses-libs-5.7-3.20090208.el6.x86\_64.rpm](http://vault.centos.org/6.5/os/x86_64/Packages/ncurses-libs-5.7-3.20090208.el6.x86_64.rpm) rpm -Uvh [ncurses-base-5.7-3.20090208.el6.x86\_64.rpm](http://vault.centos.org/6.5/os/x86_64/Packages/ncurses-base-5.7-3.20090208.el6.x86_64.rpm) rpm -Uvh [ncurses-devel-5.7-3.20090208.el6.x86\_64.rpm](http://vault.centos.org/6.5/os/x86_64/Packages/ncurses-devel-5.7-3.20090208.el6.x86_64.rpm)  怕麻烦的话，可以这样一次过：**rpm -Uvh \*.rpm –nodeps –force** |

#### 2.2.2.2.下载erlang文件

erlang语言环境的安装(rabbitMq是用erlang分布式语言开发的)

定义到一个位置进行下载/usr/local/rabbitmq/

方法一：有网的环境下

|  |
| --- |
| wget <http://erlang.org/download/otp_src_19.3.tar.gz> |

方法二：没网的环境

本地下载后上传到linux系统中

#### 2.2.2.3.运行安装erlang

解压erlang安装包 到/usr/local/rabbitmq/：

|  |
| --- |
| tar -zxvf otp\_src\_19.3.tar.gz –C /usr/local/rabbitmq/ |

进入erlang目录

|  |
| --- |
| cd otp\_src\_19.3 |

编译安装erlang语言环境 prefix=/usr/local/erlang 为安装目录

|  |
| --- |
| ./configure --prefix=/usr/local/erlang --enable-smp-support --enable-threads --enable-sctp --enable-kernel-poll --enable-hipe --with-ssl --without-javac |

erlang语言编译配置选项：

–prefix 指定安装目录

–enable-smp-support启用对称多处理支持（Symmetric Multi-Processing对称多处理结构的简称）

–enable-threads启用异步线程支持

–enable-sctp启用流控制协议支持（Stream Control Transmission Protocol，流控制传输协议）

–enable-kernel-poll启用Linux内核poll

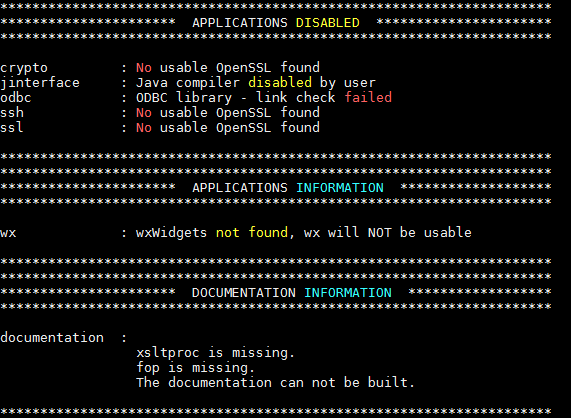
–enable-hipe启用高性能Erlang

–with-ssl 启用ssl包

–without-javac 不用java编译

出现的一些错误：

错误1：缺少OpenSSL



解决：

# yum list | grep ssl

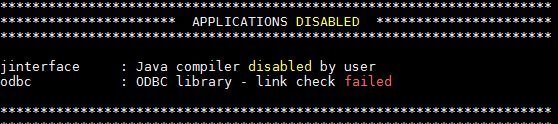
# yum install openssl-devel

# ./configure --prefix=/home/erlang --without-javac

重新执行命令

./configure --prefix=/usr/local/erlang --enable-smp-support --enable-threads --enable-sctp --enable-kernel-poll --enable-hipe --with-ssl --without-javac

错误2缺少ODBC



解决方法 :

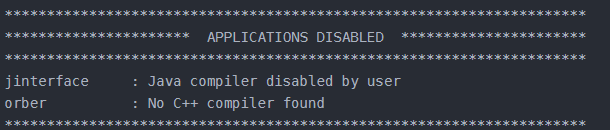
# yum list | grep ODBC

# yum install unixODBC-devel

重新执行命令

./configure --prefix=/usr/local/erlang --enable-smp-support --enable-threads --enable-sctp --enable-kernel-poll --enable-hipe --with-ssl --without-javac

错误3 gcc- c++



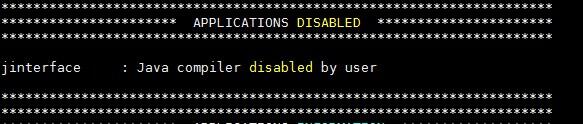
解决方法 :

# yum install gcc-c++

重新执行命令

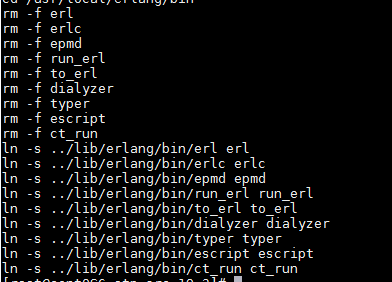
./configure --prefix=/usr/local/erlang --enable-smp-support --enable-threads --enable-sctp --enable-kernel-poll --enable-hipe --with-ssl --without-javac

展示下面算是成功



开始安装编译：

|  |
| --- |
| make && make install |

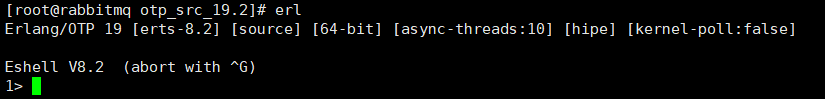


配置erlang环境变量

|  |
| --- |
| # vi /etc/profile  在末尾加入以下内容：  #set erlang environment  export PATH=$PATH:/usr/local/erlang/bin  :wq  使环境变量生效  # source /etc/profile |

测试erlang安装是否成功 会输出 erlang安装版本号

|  |
| --- |
| erl |



### 2.2.3.安装RabbitMQ Server

#### 2.2.3.1.下载安装rabbitmq

**1、在有网的环境下 下载安装rabbitmq**

|  |
| --- |
| cd /usr/local *//切换到计划安装RabbitMQ的目录，我这里放在/usr/local*  wget http://www.rabbitmq.com/releases/rabbitmq-server/v3.6.15/rabbitmq-server-3.6.15-1.el6.noarch.rpm *//下载RabbitMQ* |

**2、再没有网的时候，下载安装rabbitmq**

可以先下载到本地，在复制到centos中

解压压缩包：需要先安装xz包 执行命令：yum -y install xz

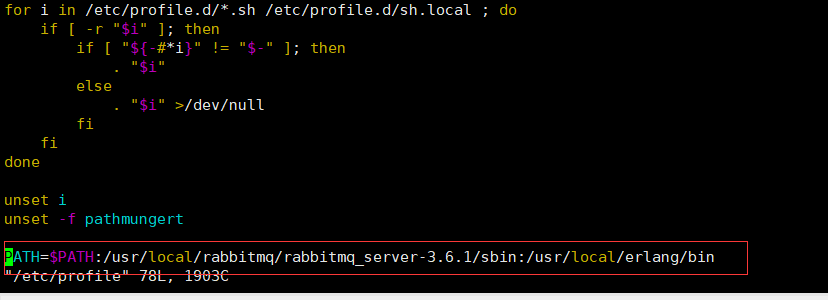
|  |
| --- |
| xz -d rabbitmq-server-generic-unix-3.7.6.tar.xz  tar -xvf rabbitmq-server-generic-unix-3.7.6.tar |

添加永久环境变量

|  |
| --- |
| vi /etc/profile  //将下面的写入进去  export PATH=$PATH: /usr/local/rabbitmq/rabbitmq\_server-3.7.6/sbin:/usr/local/erlang/bin |

环境变量生效

|  |
| --- |
| source  /etc/profile |

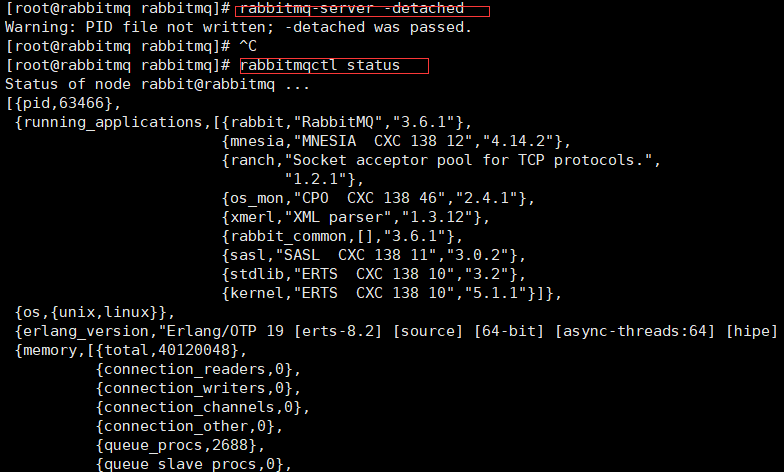


启动服务：

|  |
| --- |
| rabbitmq-server -detached *//启动rabbitmq，-detached代表后台守护进程方式启动。* |

查看状态，如果显示如下[截图](https://www.baidu.com/s?wd=%E6%88%AA%E5%9B%BE&tn=24004469_oem_dg&rsv_dl=gh_pl_sl_csd)说明安装成功：

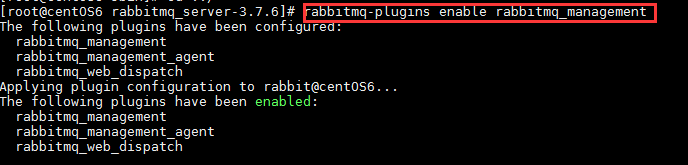
|  |
| --- |
| rabbitmqctl status |



## 2.3.安装Web管理界面插件

安装命令

rabbitmq-plugins enable rabbitmq\_management



## 2.4.设置RabbitMQ远程ip登录

这里我们以创建个admin帐号，密码123456为例，创建一个账号并支持远程ip访问。

1.创建账号

rabbitmqctl add\_user admin 123456



2.设置用户角色

rabbitmqctl set\_user\_tags admin administrator



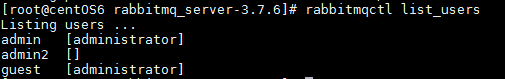
3.设置用户权限

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



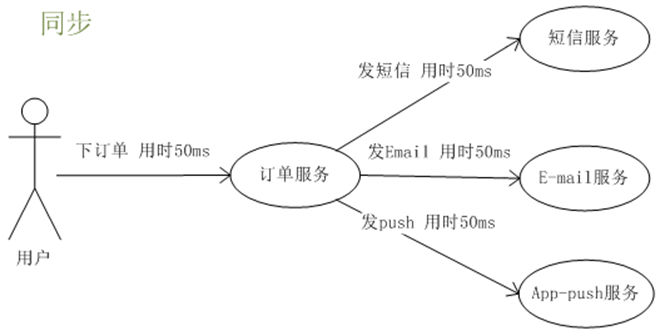
4.设置完成后可以查看当前用户和角色(需要开启服务)

rabbitmqctl list\_users

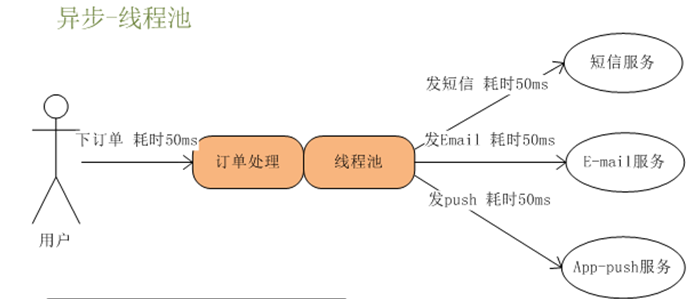


浏览器输入：serverip:15672。其中serverip是RabbitMQ-Server所在主机的ip。（注意：关闭防火墙）

# 3.为什么要使用RabbitMQ

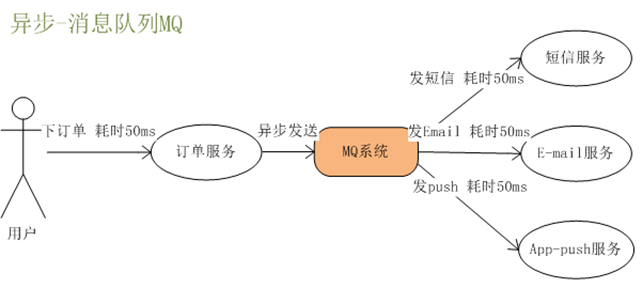


**当用户下订单到订单服务需要50ms，订单服务发短信需要50ms，发email需要50ms，发push需要50ms，总共下来需要耗时200ms，并且强耦合，为了解决这个为题以下有几套方案：**



**订单服务自己创建一个线程池，这样就可以节省时间。**

**缺点：自己实现线程池，并且强耦合**



**使用消息队列MQ，消息队列MQ是一套完整的系统，订单服务只需要将任务交给MQ系统，自己便可以脱身，发短信、发email、发push这些交给MQ系统去处理，自己只需要返回给客户一个等待信息，这样既节省了时间，也降低了耦合度。**

# **4.**消息队列基础知识

## 4.1、Provider

消息生产者，就是投递消息的程序。

## 4.2、Consumer

消息消费者，就是接受消息的程序。

## 4.3、没有使用消息队列时消息传递方式

Consumer

Provider

消息

## 4.4、使用消息队列后消息传递方式

Provider

消息队列

Consumer

消息 消息

## 4.5、什么是队列？

队列就像存放了商品的仓库或者商店，是生产商品的工厂和购买商品的用户之间的中转站

## 4.6、队列里存储了什么？

在rabbitMQ中，信息流从你的应用程序出发，来到Rabbitmq的队列，所有信息可以只存储在一个队列中。队列可以存储很多信息，因为它基本上是一个无限制的缓冲区，前提是你的机器有足够的存储空间。

## 4.7、队列和应用程序的关系？

多个生产者可以将消息发送到同一个队列中，多个消费者也可以只从同一个队列接收数据。

# 4.rabbitmq控制器-- rabbitmqctl

## 3.1. rabbitmqctl简介

Rabbitmqctl 可以管理rabbitmq的方方面面。Ui不如Rabbitmqctl管理的多。

Rabbitmqctl的位置在：RabbitMQ Server\rabbitmq\_server-3.7.4\sbin\ rabbitmqctl

Rabbitmqctl的命令格式：

rabbitmqctl [-n node] [-t timeout] [-q] {command} [command options...]

## 3.2.Rabbitmqctl管理的内容

参考资料：http://www.rabbitmq.com/rabbitmqctl.8.html

### 3.2.1.应用程序和集群的管理 Application and Cluster Management

《1》rabbitmqctl stop 停止erlang节点

《2》rabbitmqctl stop\_app 停止rabbitmq应用

《3》rabbitmqctl start\_app 开启rabbitmq应用

说明：erlang相当于一门语言，一种环境。单个Rabbitmq以及集群都是建立在这样的环境下。rabbitmqctl stop\_app与rabbitmqctl start\_app只是对rabbitmq应用进行操作，rabbitmqctl stop与 rabbitmqctl start 操作的是整个环境。

《4》rabbitmqctl reset 重置格式化rabbitmq（这样里面的交换机、队列都会不见）

说明：在执行rabbitmqctl reset时，需要先rabbitmqctl stop\_app（关闭rabbitmq应用）才能执行rabbitmqctl reset成功

1. rabbitmqctl force\_reset 强制格式化。

### 3.2.2.集群配置 【Cluster management】

RAM 内存节点 指数据保存在内存

DISK 磁盘节点 指数据保存在磁盘中

rabbitmqctl purge\_queue [-p vhost] {queue} 用于清空某一个队列中的内容

### 3.2.3.用户管理User management

1. rabbitmqctl add\_user {username} {password} 新增用户
2. rabbitmqctl set\_user\_tags {username} {tag ...} 设置user的角色
3. rabbitmqctl authenticate\_user {username} {password} 验证用户名，密码是否正确
4. rabbitmqctl list\_users 查看所有的用户

说明：用户角色可分为五类：超级管理员, 监控者, 策略制定者, 普通管理者以及其他。

(1) 超级管理员(administrator)

可登陆管理控制台(启用management plugin的情况下)，可查看所有的信息，并且可以对用户，策略(policy)进行操作。

(2) 监控者(monitoring)

可登陆管理控制台(启用management plugin的情况下)，同时可以查看rabbitmq节点的相关信息(进程数，内存使用情况，磁盘使用情况等)

(3) 策略制定者(policymaker)

可登陆管理控制台(启用management plugin的情况下), 同时可以对policy进行管理。

(4) 普通管理者(management)

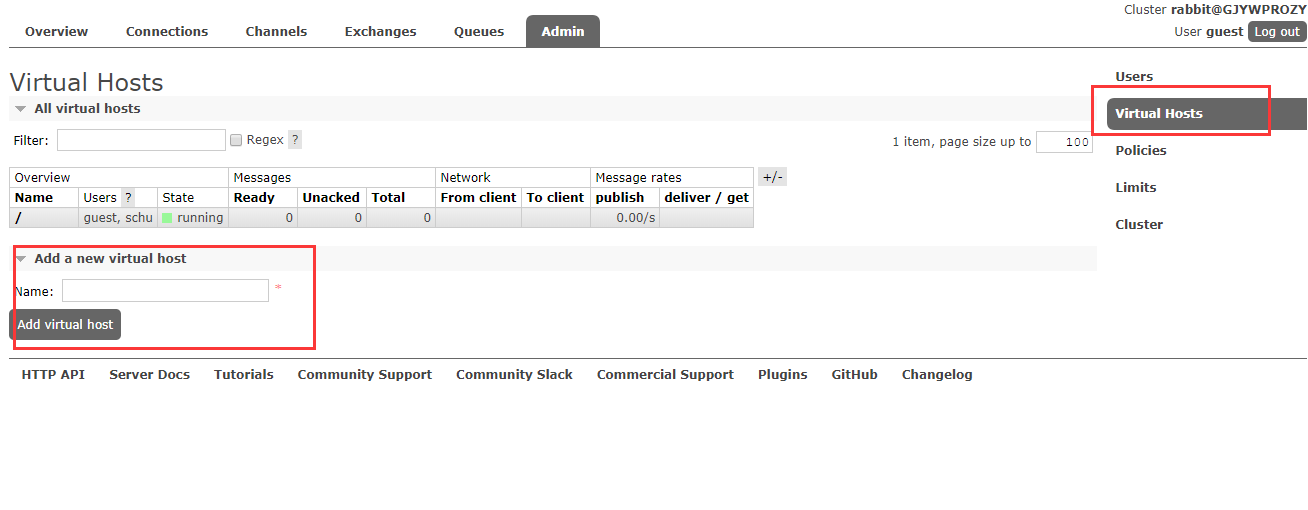
仅可登陆管理控制台(启用management plugin的情况下)，无法看到节点信息，也无法对策略进行管理。

(5) 其他的

无法登陆管理控制台，通常就是普通的生产者和消费者。

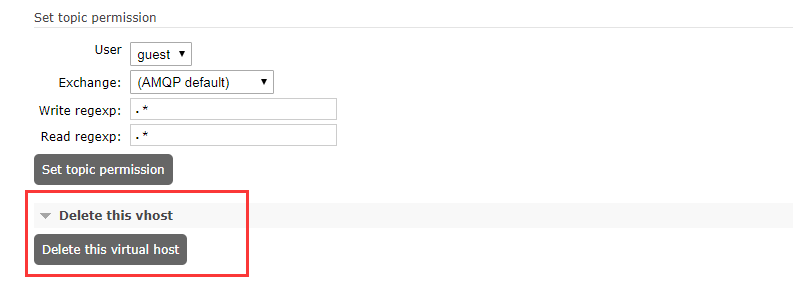
### 3.3.4.访问控制Access control

1. rabbitmqctl add\_vhosts {vhost} 添加虚拟机



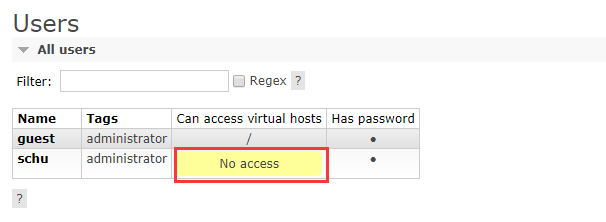
1. rabbitmqctl delete\_vhost {vhost} 删除虚拟机

进入单个虚拟机后，最下面有删除虚拟机

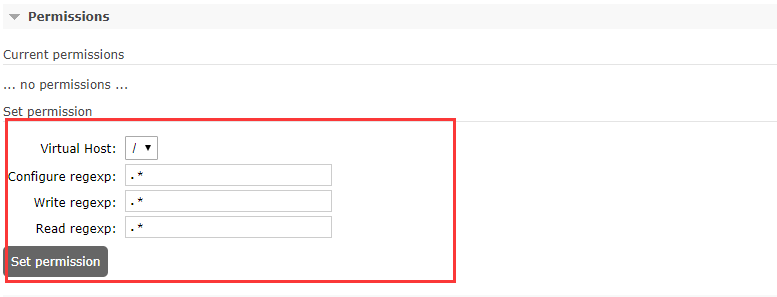


1. rabbitmqctl list\_vhosts name tracing 查看交换机
2. rabbitmqctl set\_permissions [–p vhost] {user} {conf} {write} {read} 设置用户访问虚拟机的权限

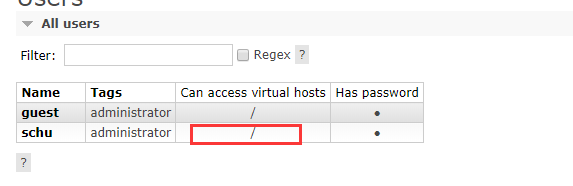
当用户没有访问虚拟机权限时：



设置权限：点击名称进入后设置



设置好后：



1. rabbitmqctl clear\_permissions [-p vhost] {username} 删除用户访问虚拟你的权限

### 3.3.5. 参数管理Parameter Management

Parameter Management【第三方插件比较多】

### 3.3.6. 政策的管理Policy Management

使用在对queue的全局设置，队列的镜像

### 3.3.7. 服务器状态 Server Status

《1》rabbitmqctl list\_queues [-p vhost] [queueinfoitem ...] 获取队列的信息

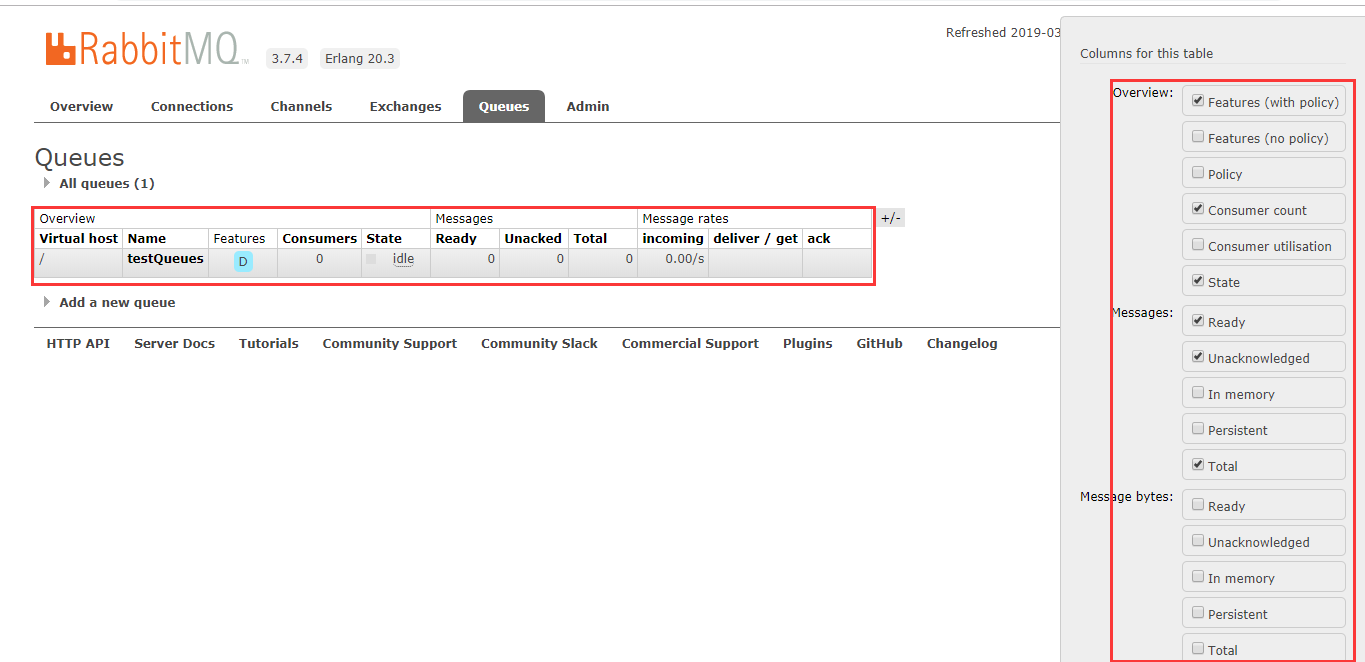
如：rabbitmqctl list\_queues name 获取对列的名称

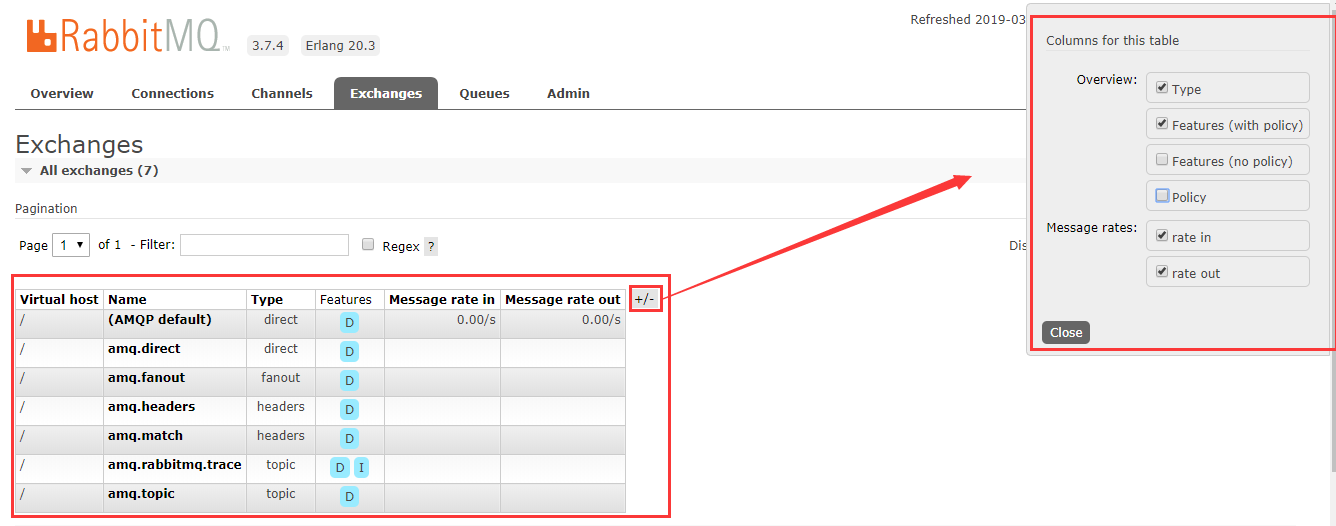
rabbitmqctl list\_queues name pid 获取对列的pid

rabbitmqctl list\_queues name durable 获取对列是否持久化

rabbitmqctl list\_queues name exclusive 获取对列是否排外（只能自己用，不能别人用）

图中这些信息都可以查到：

《2》 rabbitmqctl list\_exchanges [-p vhost] [exchangeinfoitem ...]



《3》 rabbitmqctl list\_bindings [-p vhost] [bindinginfoitem ...]

《4》 rabbitmqctl list\_connections [connectioninfoitem ...]

《5》rabbitmqctl list\_channels [channelinfoitem ...]

connections是长连接 ，channels是挂在connection之上的

《6》 rabbitmqctl status 状态

《7》 rabbitmqctl environment 配置文件

《8》 rabbitmqctl Miscellaneous 【混合一些命令】

# 4.rabbitmq的配置文件

1.