Step 1.

First make sure that all your system packages are up-to-date by running these following apt-get commands in the terminal.

1:sudo apt-get update

2:sudo apt-get upgrade

Step 2. Installing Erlang.

wget http://packages.erlang-solutions.com/ubuntu/erlang\_solutions.asc

sudo apt-key add erlang\_solutions.asc

sudo apt-get update

sudo apt-get install erlang

sudo apt-get install erlang-nox

Step 3. Installing RabbitMQ.

First, Enable RabbitMQ application repository:

echo "deb http://www.rabbitmq.com/debian/ testing main" >> /etc/apt/sources.list

After the repository is added, we will add the RabbitMQ public key to our trusted key list to avoid any warnings about unsigned packages:

wget https://www.rabbitmq.com/rabbitmq-signing-key-public.asc

sudo apt-key add rabbitmq-signing-key-public.asc

Now we just need to run an update, and install the rabbitmq-server from our newly added package:

sudo apt-get install rabbitmq-server

To start, stop, restart and check the RabbitMQ status, use the following:

|  |  |
| --- | --- |
| 3  4  5  6  7  8  9  10  11  12  13  14 | # To automatic enable boot service:  systemctl enable rabbitmq-server    # To start the service:  systemctl start rabbitmq-server    # To stop the service:  systemctl stop rabbitmq-server    # To restart the service:  systemctl restart rabbitmq-server    # To check the status:  systemctl status rabbitmq-server |

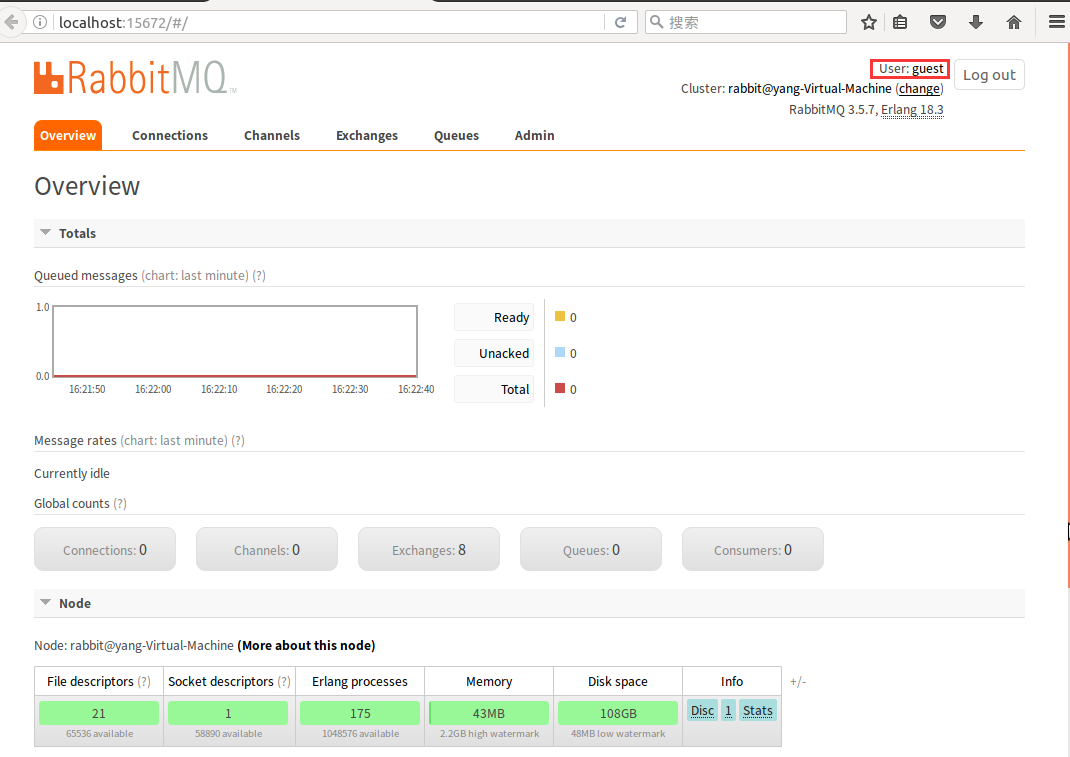
Step 4. Access RabbitMQ management console.

To manage your RabbitMQ server, you can use the rabbitmq-management plugin. This plugin allows you to manage and monitor your RabbitMQ server in a variety of ways, such as listing and deleting exchanges, queues, bindings and many more. To install the plugin, use the following command:

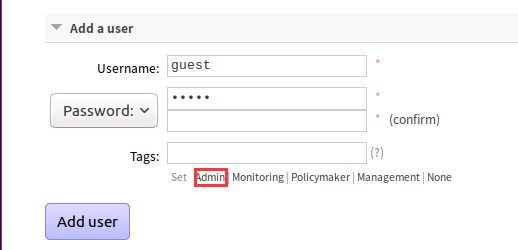
sudo rabbitmq-plugins enable rabbitmq\_management

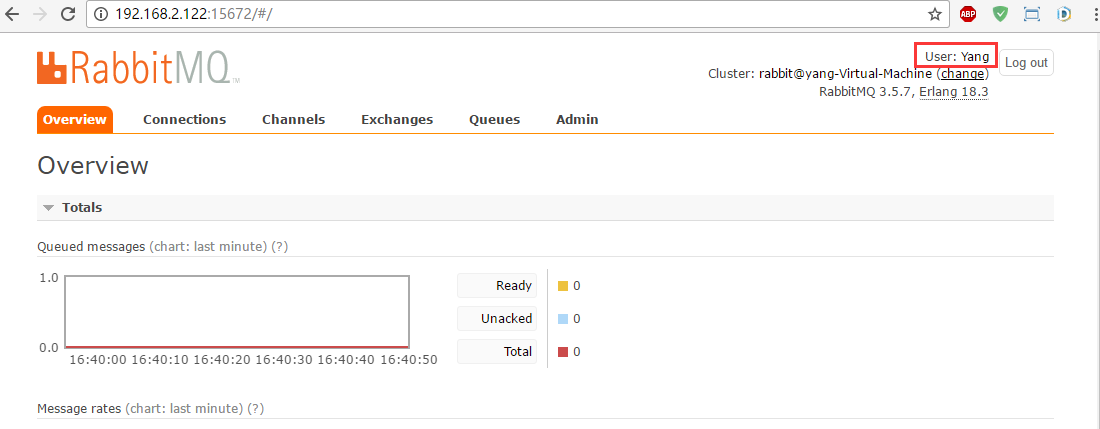
http://yourdomain.com:15672 or <http://server-ip:15672>

默认用记名密都是 guest ，但它只能在本地访问（安装RabbitMQ 的机器访问）。



Admin页签下面可以增加用户并设置为管理员就可以远程访问了。





/\*

未经处理的异常: RabbitMQ.Client.Exceptions.BrokerUnreachableException: None of the specified endpoints were reachable ---> RabbitMQ.Client.Exceptions.AuthenticationFailureException: ACCESS\_REFUSED - Login was refused using authentication mechanism PLAIN. For details see the broker logfile.

在 RabbitMQ.Client.Framing.Impl.Connection.StartAndTune()

在 RabbitMQ.Client.Framing.Impl.Connection.Open(Boolean insist)

在 RabbitMQ.Client.Framing.Impl.Connection..ctor(IConnectionFactory factory, Boolean insist, IFrameHandler frameHandler, String clientProvidedName)

在 RabbitMQ.Client.Framing.Impl.ProtocolBase.CreateConnection(IConnectionFactory factory, Boolean insist, IFrameHandler frameHandler, String clientProvidedName)

在 RabbitMQ.Client.ConnectionFactory.CreateConnection(IEndpointResolver endpointResolver, String clientProvidedName)

--- 内部异常堆栈跟踪的结尾 ---

在 RabbitMQ.Client.ConnectionFactory.CreateConnection(IEndpointResolver endpointResolver, String clientProvidedName)

在 RabbitMQ.Client.ConnectionFactory.CreateConnection()

在 RabbitMQ.Producer.Producer.Main(String[] args) 位置 E:\GitlabCIDemo\RabbitMQ\RabbitMQ.Producer\Producer.cs:行号 15

\*/

代码没有写默认用户名称密码，它会用默认的guest访问。但guest只能在安装RabbitMQ的机器上访问。我通过本机(win10 )访问ubuntu 16.04 ,所以，不行。

按上面的方法添加一个新的admin Yang，后仍然报错。

/\*

未经处理的异常: RabbitMQ.Client.Exceptions.BrokerUnreachableException: None of the specified endpoints were reachable ---> RabbitMQ.Client.Exceptions.OperationInterruptedException: The AMQP operation was interrupted: AMQP close-reason, initiated by Library, code=541, text="Unexpected Exception", classId=0, methodId=0, cause=System.IO.IOException: 无法从传输连接中读取数据: 远程主机强迫关闭了一个现有的连接。。 ---> System.Net.Sockets.SocketException: 远程主机强迫关闭了一个现有的连接。

在 System.Net.Sockets.Socket.Receive(Byte[] buffer, Int32 offset, Int32 size, SocketFlags socketFlags)

在 System.Net.Sockets.NetworkStream.Read(Byte[] buffer, Int32 offset, Int32 size)

--- 内部异常堆栈跟踪的结尾 ---

在 RabbitMQ.Client.Impl.Frame.ReadFrom(NetworkBinaryReader reader)

在 RabbitMQ.Client.Impl.SocketFrameHandler.ReadFrame()

在 RabbitMQ.Client.Framing.Impl.Connection.MainLoopIteration()

在 RabbitMQ.Client.Framing.Impl.Connection.MainLoop()

在 RabbitMQ.Client.Impl.SimpleBlockingRpcContinuation.GetReply(TimeSpan timeout)

在 RabbitMQ.Client.Impl.ModelBase.ConnectionOpen(String virtualHost, String capabilities, Boolean insist)

在 RabbitMQ.Client.Framing.Impl.Connection.Open(Boolean insist)

在 RabbitMQ.Client.Framing.Impl.Connection..ctor(IConnectionFactory factory, Boolean insist, IFrameHandler frameHandler, String clientProvidedName)

在 RabbitMQ.Client.Framing.Impl.ProtocolBase.CreateConnection(IConnectionFactory factory, Boolean insist, IFrameHandler frameHandler, String clientProvidedName)

在 RabbitMQ.Client.ConnectionFactory.CreateConnection(IEndpointResolver endpointResolver, String clientProvidedName)

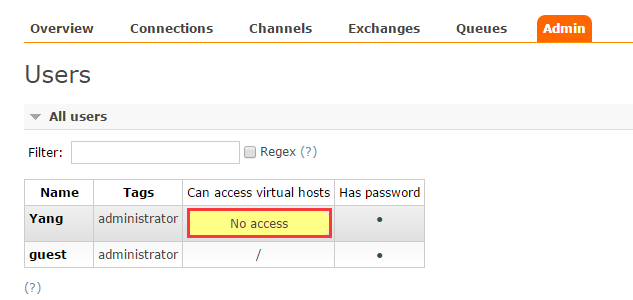
--- 内部异常堆栈跟踪的结尾 ---

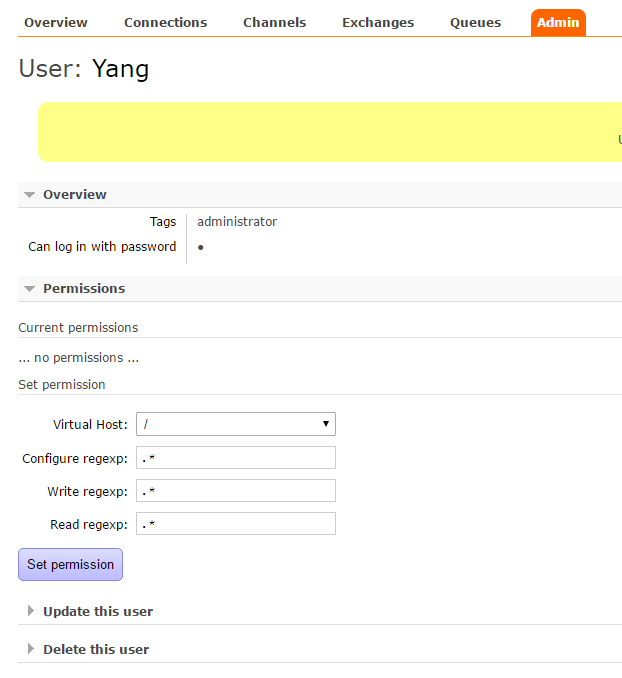
在 RabbitMQ.Client.ConnectionFactory.CreateConnection(IEndpointResolver endpointResolver, String clientProvidedName)

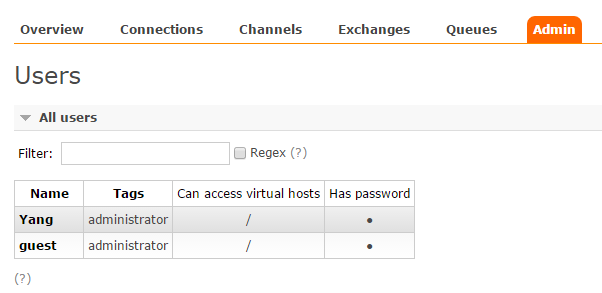
在 RabbitMQ.Client.ConnectionFactory.CreateConnection()

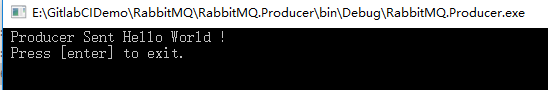
在 RabbitMQ.Producer.Producer.Main(String[] args) 位置 E:\GitlabCIDemo\RabbitMQ\RabbitMQ.Producer\Producer.cs:行号 21

\*/

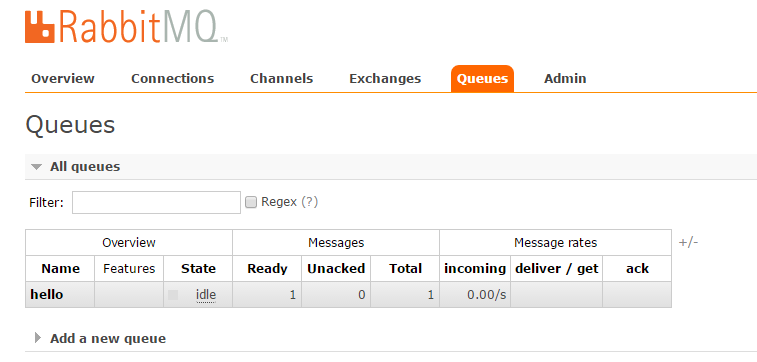




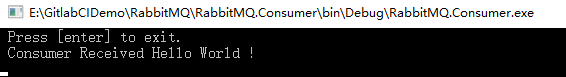


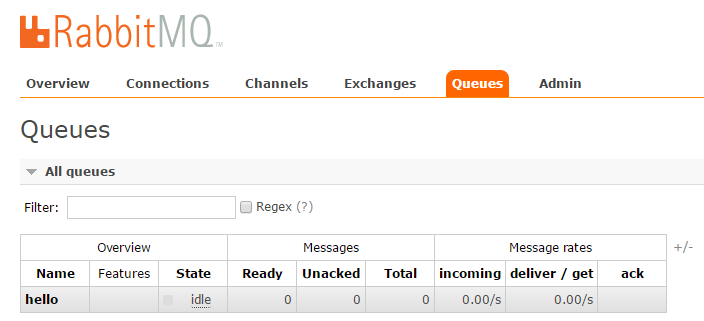


Queues中有一个hello的队列了

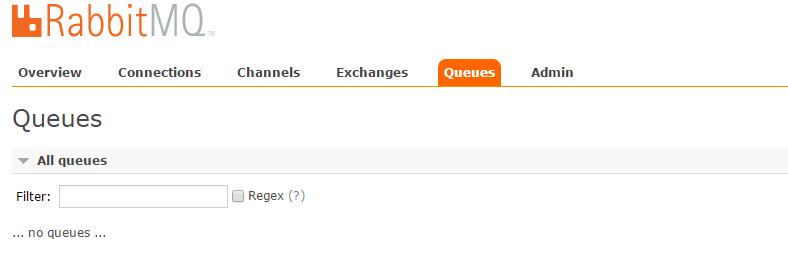


开启RabbitMQ. Consumer,用它来消费刚才那条信息，这样队列中的消息条数就变成了0(Ready和Total)





虚拟机或RabbitMQ重启之后 Queues丢失



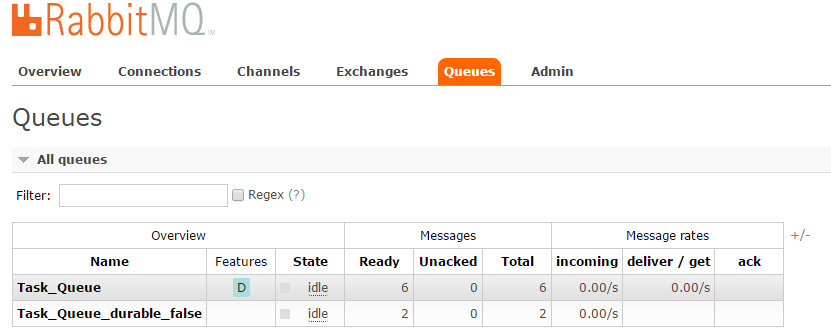
参考文档:

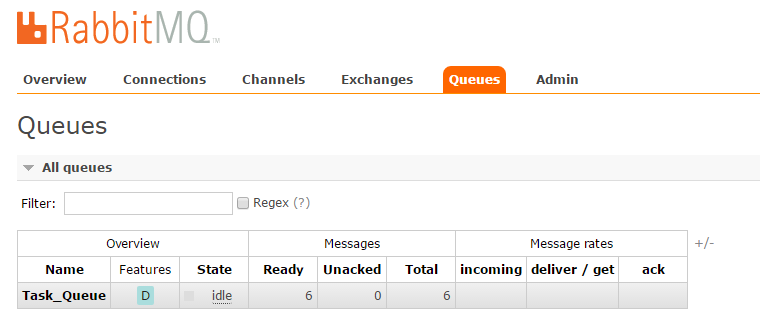
<http://idroot.net/linux/install-rabbitmq-ubuntu-16-04/>

<http://www.rabbitmq.com/tutorials/tutorial-one-dotnet.html>

## 2 [Work queues](http://www.rabbitmq.com/tutorials/tutorial-two-python.html)

## Durable



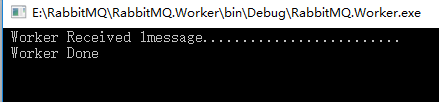


## Fair dispatch

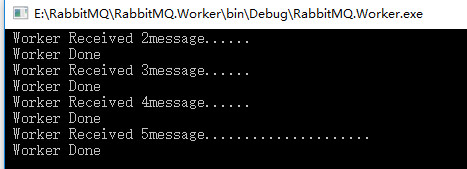
**Tasker**



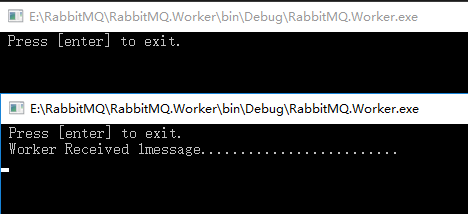
**Consumer1**

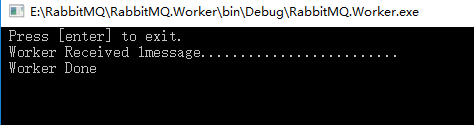


**Consumer2**



## Message acknowledgment





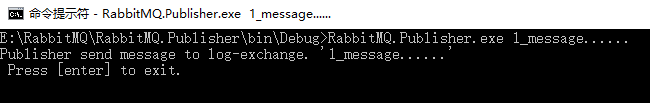
<http://www.rabbitmq.com/tutorials/tutorial-two-dotnet.html>

## Publish/Subscribe

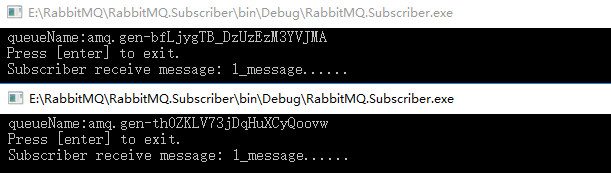
Exchange log



Publisher publish a message



Subscriber two subscriber receive the same message at the same time



<http://www.rabbitmq.com/tutorials/tutorial-three-dotnet.html>