# RabbitMQ

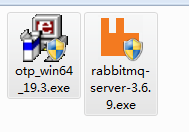
## 安装

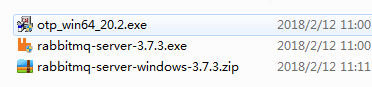
RabbitMQ是在AMQP（ActiveMQ是JMS）基础上的完整、可复用的企业消息系统。RabbitMQ是由ERLang实现的，ERLang是通用的面向开发的编程语言，Erlang是并发的、分布式的、健壮的、软实时的语言。

Rabbitmq单个消息的大小限制是2GB，并且是不可更改的

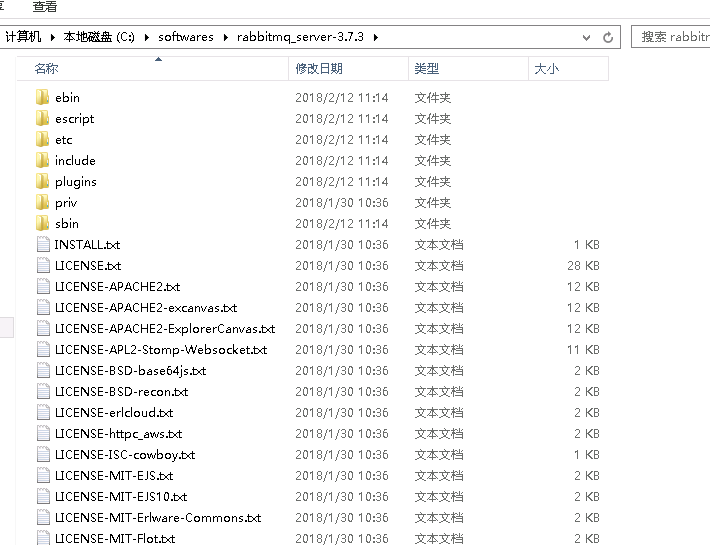
Message size is limited to 2GB

安装RabbitMQ，必须先安装ERLang，再安装RabbitMQ。





提供了两种版本:exe zip 其中zip版本解压可以直接使用，zip解压后如下：



## 安装

检查机器是64位还是32位，下载相应版本的ERLang （64位机器可以安装32位程序，32位机器只能安装32位）

Erlang的官网: <http://www.erlang.org/downloads>

RabbitMQ的官网: https://www.rabbitmq.com

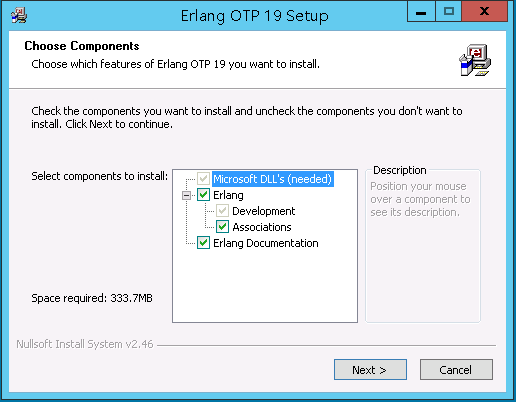
在[https://www.rabbitmq.com/which-erlang.html 查看Erlang和rabbitmq](https://www.rabbitmq.com/which-erlang.html%20查看Erlang和rabbitmq)对应版本

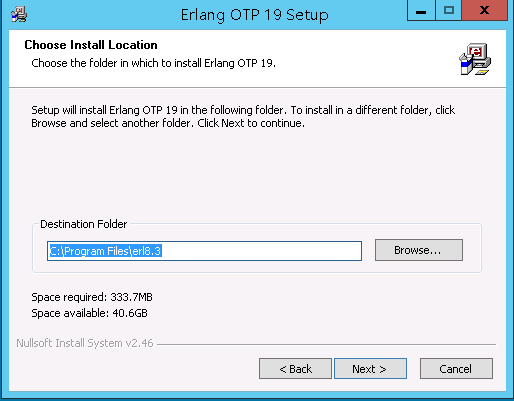
安装步骤：https://www.rabbitmq.com/install-windows-manual.html

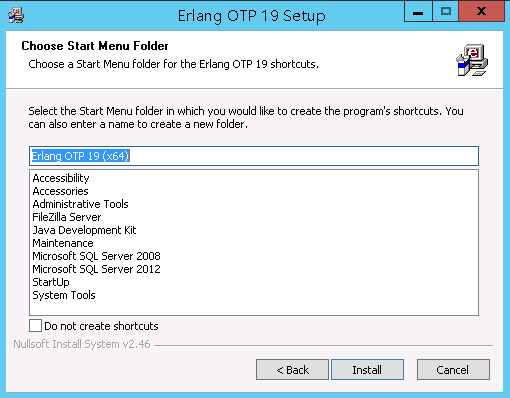
### 安装ERLang

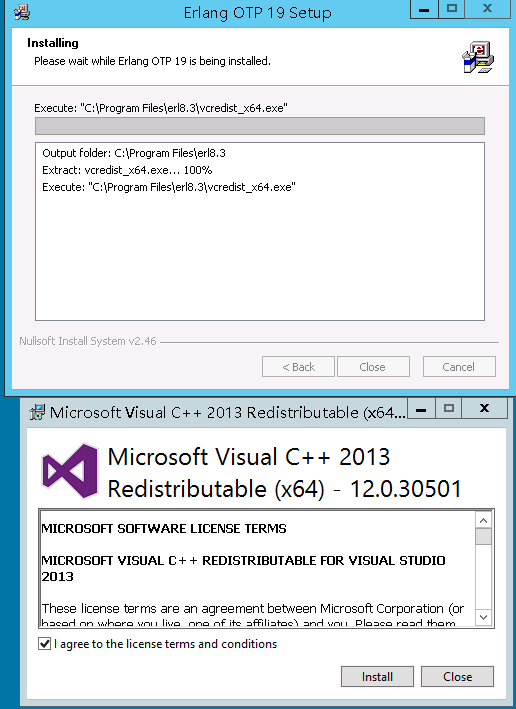
以管理员运行ERLang安装文件，推荐使用64位版本，. **Important:** you mustrun the Erlang installer using an administrative account otherwise a registry key expected by the RabbitMQ installer will not be present.

以管理员权限运行ERLang安装包。



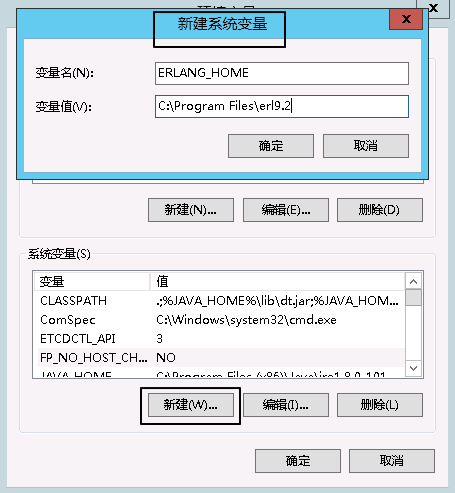




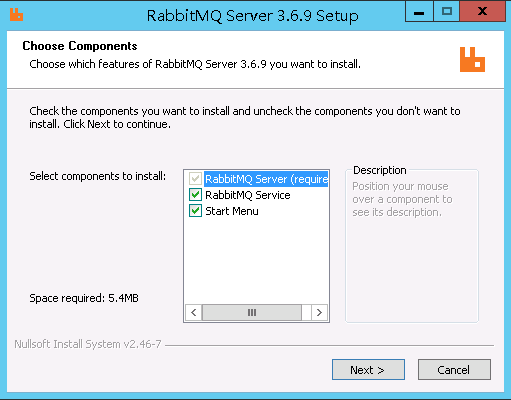


### 设置环境变量ERLANG\_HOME

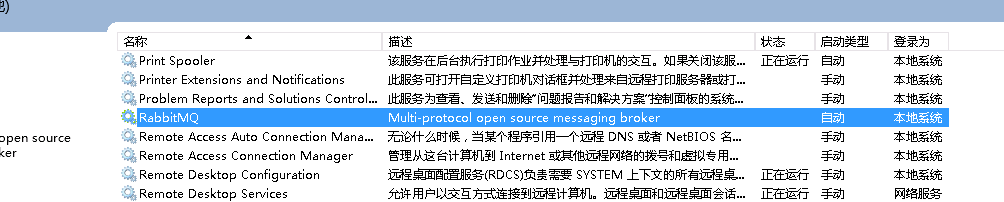
设置系统变量  ERLANG\_HOME ,设置为包含 bin\erl.exe的目录。



### 安装RabbitMQ EXE版本



安装完成后，RabbitMQ以系统服务的形式存在，启动该服务



此时RabbitMQ以默认配置运行。The RabbitMQ service starts automatically. You can stop/reinstall/start the RabbitMQ service from the Start Menu.

RabbitMQ使用如下端口：

* 4369: [epmd](http://erlang.org/doc/man/epmd.html), a peer discovery service used by RabbitMQ nodes and CLI tools
* 5672, 5671: used by AMQP 0-9-1 and 1.0 clients without and with TLS
* 25672: used by Erlang distribution for inter-node and CLI tools communication and is allocated from a dynamic range (limited to a single port by default, computed as AMQP port + 20000). See [networking guide](http://www.rabbitmq.com/networking.html) for details.
* 15672: [HTTP API](http://www.rabbitmq.com/management.html) clients and [rabbitmqadmin](http://www.rabbitmq.com/management-cli.html) (only if the [management plugin](http://www.rabbitmq.com/management.html) is enabled)

### 同步Erlang Cookies

The Erlang cookie 指的是 .erlang.cookie

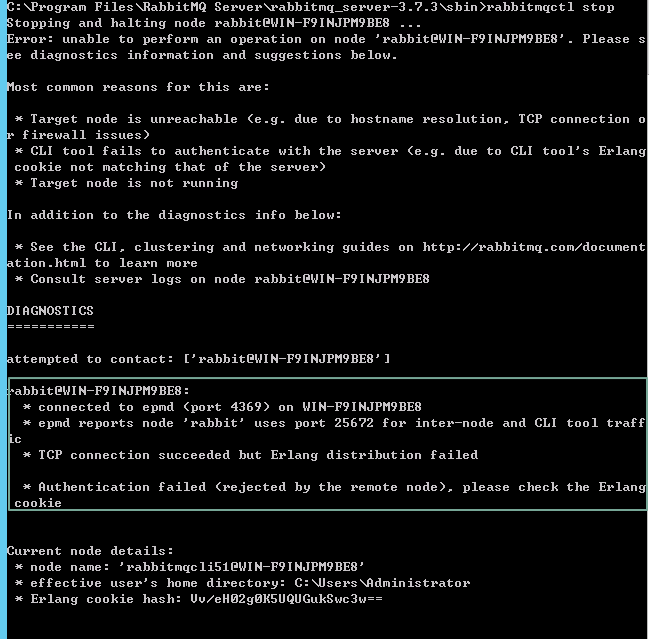
Erlang cookie文件是erlang实现分布式的必要文件，erlang分布式的每个节点要保持相同的.erlang.cookie文件，保证文件的权限是400。400表示拥有者有读的权限，没有其它权限。

在windows下将C:\Windows\System32\config\systemprofile目录下的.erlang.cookie与 C:\Users\%USERNAME%\目录下的.erlang.cookie内容同步，

以C:\Windows\System32\config\systemprofile目录下的内容为准。

rmq集群中所有的节点必须有相同的erlang cookie文件。

该同步用于解决如下问题:



### RabbitMQ配置

RabbitMQ可以通过环境变量和rabbitmq-env.conf配置，如果两者都没有将使用RabbitMQ的默认值。

优先级: 环境变量>rabbitmq-env.conf>默认值

### 安装RabbitMQ zip版本

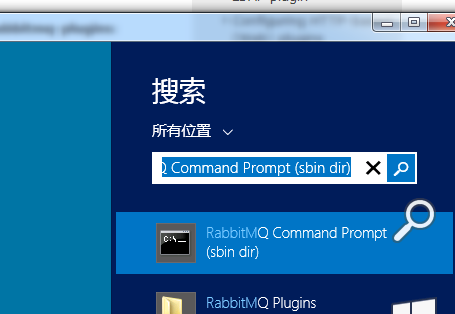
Download rabbitmq-server-windows-3.7.3.zip from the link above.

From the zip file, extract the folder named rabbitmq\_server-3.7.3 into C:\Program Files\RabbitMQ (or somewhere suitable for application files).

解压可用

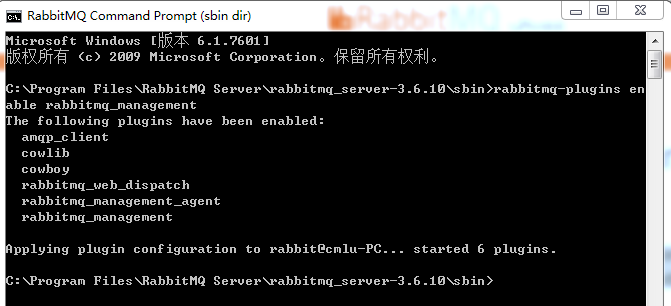
## 配置管理界面Management Plugin

启动RabbitMQ命令行

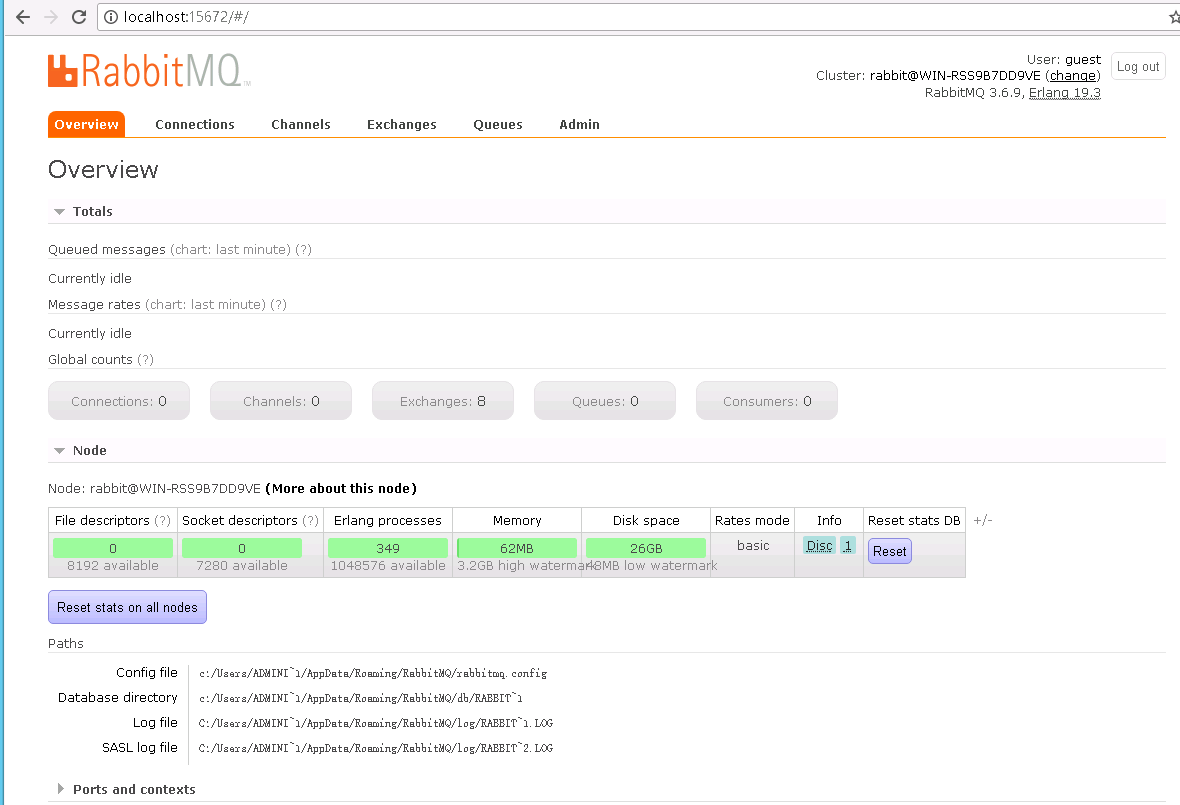


执行如下命令：

rabbitmq-plugins enable rabbitmq\_management

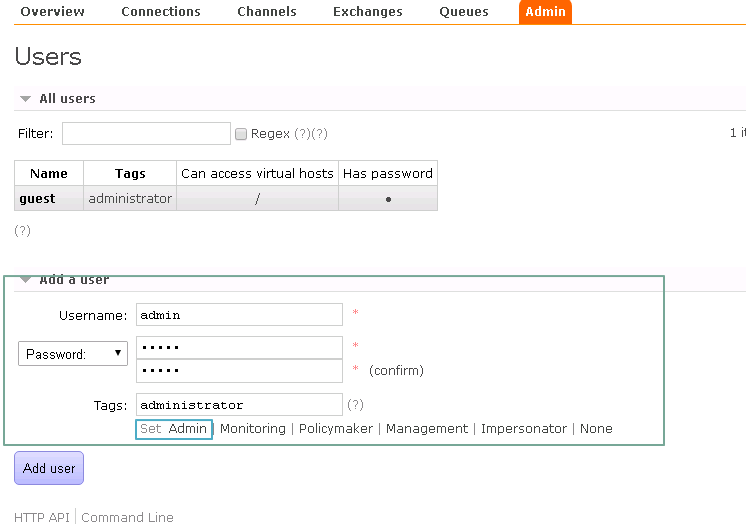


WebUi的地址：[http://*server-name*:15672/](http://server-name:15672/) 默认用户名密码：guest guest 该用户有管理员权限

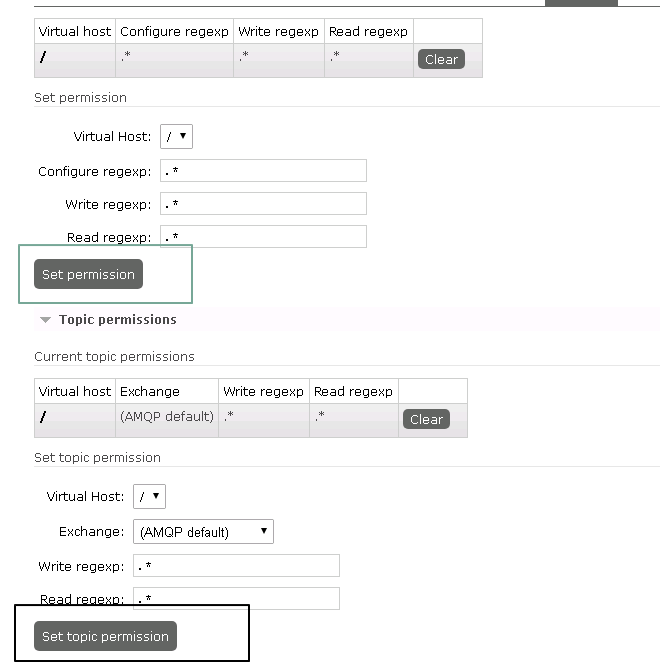


默认guest只能本地localhost访问。

添加新的管理员帐号admin/admin

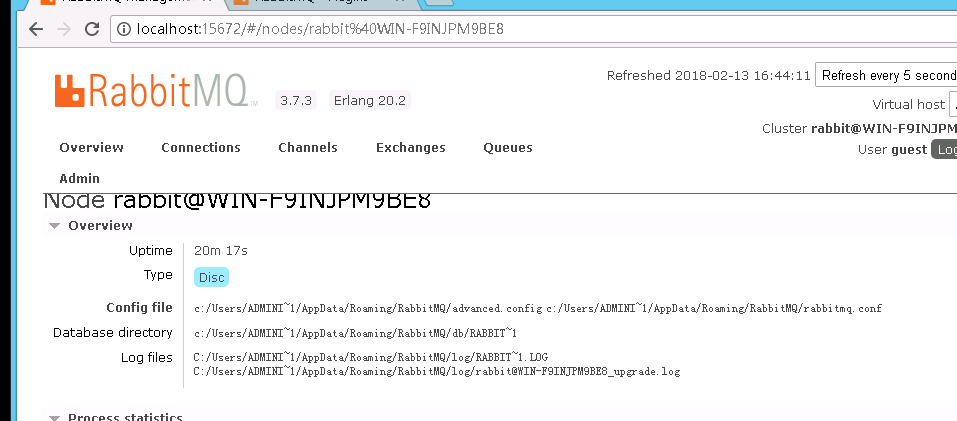


设置权限



.\*表示所有的权限。

通过managed ui查看配置文件的位置



默认rabbitmq的logs和data目录放在C:\Users\%USERNAME%\AppData\Roaming目录下，日志总是append的。

WebUI使用Http Api来实现，API的地址： http://*server-host*:15672/api/

使用rabbitmq-plugins list查看安装了哪些插件

To enable plugins, use the [rabbitmq-plugins](http://www.rabbitmq.com/man/rabbitmq-plugins.8.html) tool:

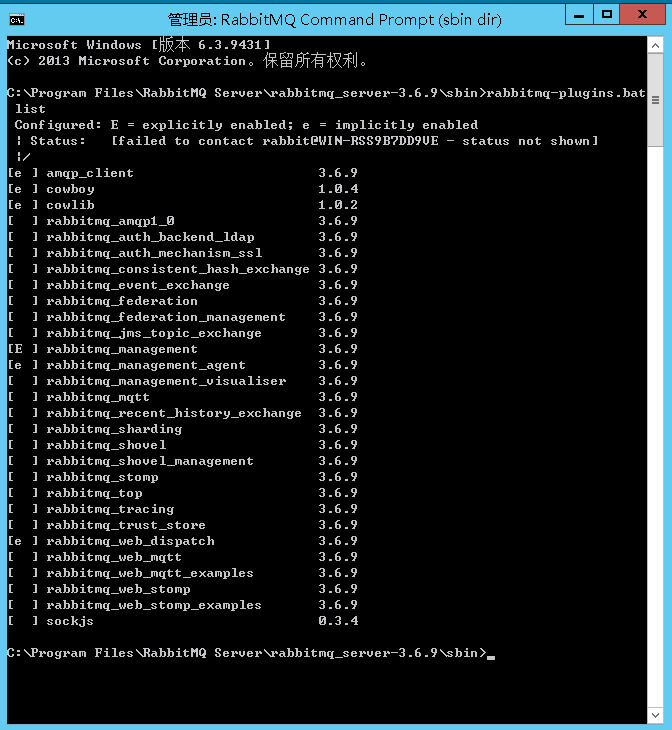
rabbitmq-plugins enable *plugin-name*

And to disable plugins again, use:

rabbitmq-plugins disable *plugin-name*

You can see a list of which plugins are enabled with:

rabbitmq-plugins list



### 注意事项

RabbitMQ的安装路径不能包含non-ascii字符。

安装erlang和rabbitmq时使用管理员权限

Copy the file .erlang.cookie manually from %SystemRoot% or %SystemRoot%\system32\config\systemprofile to %HOMEDRIVE%%HOMEPATH%.

### RabbitMq集群

集群中所有的重复所有的除队列以外的数据，队列只存在一个节点上，但是对所有的节点可见。

Hostname resolution: RabbitMQ 节点使用domain name,因此集群中所有的节点需要根据domain name翻译出ip,有两种解决方案:

1)DNS recoreds 2)local host files (c:\windows\system32\drivers\etc\hosts)

Hosts文件作用，当访问一个网站时，首先查找hosts获取IP，如果找不到再去查询dns服务器

集群可以以如下几种方式建立:

1. 在配置文件中列出集群节点
2. 使用rabbitmqctl建立

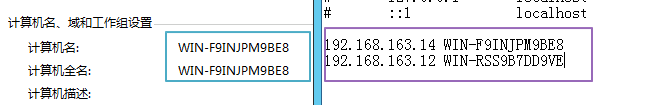
DISK and RAM node: ram节点将数据(除队列消息,队列索引和集群其它节点信息外)存放在ram中，in well over 90% of cases you want all your nodes to be disk nodes; RAM nodes are a special case that can be used to improve the performance. Ram节点启动时从集群中同步信息，集群中至少有一个Disk节点

#### 使用rabbitmqctl构建集群

Rmq node和cli工具使用cookie来判断他们是否可以彼此通信。两个节点必须有相同的cookie来彼此通信。Cookie是一个最多255的字符串。确保机器上rabbitmq的版本一致

第一步:修改hosts

将集群中所有节点的IP和机器名写入hosts文件

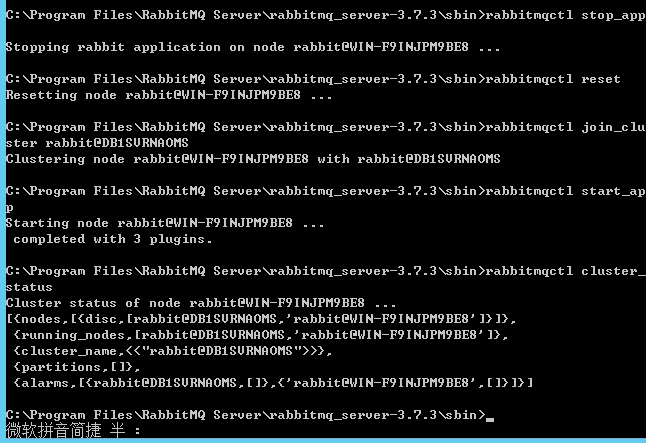


第二步:同步所有节点的cookie

停止rabbitmq，替换cookie文件，选取第一个集群节点为基准，替换其它节点的cookie文件

第三步:一次将节点rabbit2,rabbit3…….. 添加到集群中的第一个节点rabbit1(该节点的名字将成为集群的名字)

在每个要添加的节点上依次运行stop\_app(停止rabbitmq节点),reset(重置为初始状态),join\_cluster rabbit1@host(加入集群),start\_app(启动节点)



注意事项:

当整个集群停止时，the last node to go down must be the first node to be brought online.  If this doesn't happen, the nodes will wait 30 seconds for the last disc node to come back online, and fail afterwards. If the last node to go offline cannot be brought back up, it can be removed from the cluster using the forget\_cluster\_node command.

If all cluster nodes stop in a simultaneous and uncontrolled manner (for example with a power cut) you can be left with a situation in which all nodes think that some other node stopped after them. In this case you can use the force\_boot command on one node to make it bootable again - consult the rabbitmqctl manpage for more information.

Rabbitmqctl force\_boot

### Rabbitmqctl命令

参考文章: http://www.rabbitmq.com/rabbitmqctl.8.html

Rabbitmqctl:用来管理RabbitMQ broker的命令行工具

|  |  |
| --- | --- |
| **rabbitmqctl** | [**-q**] [**-l**] [**-n** *node*] [**-t** *timeout*] *command* [*command\_options*] |

命令:

Rabbitmqctl [-q] [-l] [-n node] [-t timeout] command [command\_options]

#### [OPTIONS](http://www.rabbitmq.com/rabbitmqctl.8.html#OPTIONS)

-n node

指定操作的节点名，默认时rabbit@servername, servername指的是计算机名。

-q, --quiet

Quiet output mode

--dry-run

不运行命令，only print information message

-t timeout, --timeout timeout

Operation timeout in seconds. Only applicable to “list” commands. Default is **infinity**.

-l, --longnames

Use longname for erlang distribution

--erlang-cookie cookie

Erlang distribution cookie.

#### [COMMANDS](http://www.rabbitmq.com/rabbitmqctl.8.html#COMMANDS)

help [-l] [command\_name]

列出命令的用法

force\_reset

强制mq节点设置为他的初始状态 For **reset** and **force\_reset** to succeed the RabbitMQ application must have been stopped, e.g. with **stop\_app**.

reset

设置rabbitmq到初始状态

rotate\_logs

指导rabbitmq node进行rotate log操作

shutdown

关闭rabbitmq运行的erlang进程

start\_app

启动RabbitMQ程序

stop [pid\_file]

停止Erlang node on which RabbitMQ is running

stop\_app

停止rabbitmq node，但是leave the erlang node run

wait pid\_file, wait –pid pid

等待rabbitmq application to start

集群管理命令

join\_cluster clusternode [--ram]

clusternode: node to cluster with

--ram 如果提供表示该节点是ram节点，否则是disc节点

将节点变成clusternode节点所在集群的一个节点。Before clustering, 当前节点需要reset，然后stop\_app。集群至少有一个disk node,通常不止一个。

cluster\_status

显示集群的所有节点状态

change\_cluster\_node\_type type

改变当前节点的类型 type必须是disk或者ram

forget\_cluster\_node [--offline]

--offline enable node removal from an offline node.  This is only useful in the situation where all the nodes are offline and the last node to go down cannot be brought online

set\_cluster\_name name

设置集群的名字，默认集群的名字是集群中第一个节点的名字

list\_users

列出集群中所有的用户名

### rabbitmq-server命令

http://www.rabbitmq.com/rabbitmq-server.8.html

RABBITMQ\_NODENAME:节点名,默认是rabbit, 每个节点加机器名的组合必须唯一。

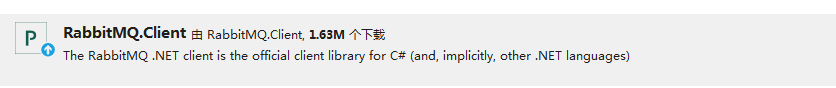
#### [OPTIONS](http://www.rabbitmq.com/rabbitmq-server.8.html#OPTIONS)

--detached

在后台启动rabbitmq节点

## .NET开发

通过Nuget添加RabbitMQ.Client



队列：多个生产者可以发送消息到一个队列，多个消费者可以从一个队列中取消息。

## Server配置