

rabbitmq基础环境搭建及高可用集群环境搭建

方法

1)安装GCC GCC-C++ Openssl等模块,安装过就不需要安装了

```
yum -y install make gcc gcc-c++ kernel-devel m4 ncurses-devel openssl-devel
```

一、安装Erlang

安装一:

```
wget https://github.com/rabbitmq/erlang-rpm/releases/download/v20.3.8.22/erlang-20.3.8.22-1.el7.x86_64.rpm
```

或者

```
wget https://github.com/qin314752/linux/raw/mq/erlang-20.3.8.22-1.el7.x86_64.rpm
```

```
rpm -Uvh erlang-20.3.8.22-1.el7.x86_64.rpm
```

```
[root@localhost local]# rpm -Uvh erlang-20.3.8.22-1.el7.x86_64.rpm
警告: erlang-20.3.8.22-1.el7.x86_64.rpm: 头V4 RSA/SHA1 Signature, 密钥 ID 6026d
准备中... ##### [100%]
正在升级/安装...
  1:erlang-20.3.8.22-1.el7 ##### [100%]
[root@localhost local]# yum install erlang
```

配置环境变量

更新环境变量

```
echo 'export PATH="/usr/lib64/erlang/bin:$PATH" ' >> /etc/profile
```

```
source /etc/profile
```

```
cat /etc/profile
```

erl 如果无显示 加入上面的环境变量

```
[root@localhost erlang-rpm-20.3.8.22]# erl
Erlang/OTP 20 [erts-9.3.3.11] [source] [64-bit] [smp:4:4] [ds:4:4:10] [async-th
reads:10] [hipe] [kernel-poll:false]

Eshell V9.3.3.11 (abort with ^G)
```

安装完

```

[root@localhost rabbitmq]# ll
总用量 4
-rw-r--r--. 1 root root 1952 Jul 31 16:25 erlang-solutions-1.0-1.noarch.rpm
[root@localhost rabbitmq]# rpm -Uvh erlang-solutions-1.0-1.noarch.rpm
错误: 依赖检测失败.
epel-release 被 erlang-solutions-1.0-1.noarch 需要
[root@localhost rabbitmq]# Install epel-release rpm

```

#执行第二句话的时候可能出现。rpm -Uvh erlang-solutions-1.0-1.noarch.rpm

错误: 依赖检测失败:

epel-release 被 erlang-solutions-1.0-1.noarch 需要

#这时候安装yum install epel-release

#然后在执行 yum install erlang

卸载 yum remove remove erlang

[root@192 ~]# erl -version

二、安装RabbitMQ Server

wget https://www.rabbitmq.com/releases/rabbitmq-server/v3.6.15/rabbitmq-server-3.6.15-1.el7.noarch.rpm

或者

wget https://github.com/qin314752/linux/raw/mq/rabbitmq-server-3.6.15-1.el7.noarch.rpm

#导入签名

rpm --import https://dl.bintray.com/rabbitmq/Keys/rabbitmq-release-signing-key.asc

或

rpm --import https://www.rabbitmq.com/rabbitmq-release-signing-key.asc

安装

yum install rabbitmq-server-3.6.15-1.el7.noarch.rpm

设置开机启动

chkconfig rabbitmq-server on

服务状态

service rabbitmq-server status

```
[root@localhost ~]# service rabbitmq-server status
Redirecting to /bin/systemctl status rabbitmq-server.service
● rabbitmq-server.service - RabbitMQ broker
   Loaded: loaded (/usr/lib/systemd/system/rabbitmq-server.service; enabled; vendor preset: disabled)
   Active: inactive (dead)
[root@localhost ~]#
```

启动服务

service rabbitmq-server start

```
[root@localhost ~]# service rabbitmq-server status
Redirecting to /bin/systemctl status rabbitmq-server.service
● rabbitmq-server.service - RabbitMQ broker
   Loaded: loaded (/usr/lib/systemd/system/rabbitmq-server.service; enabled; vendor preset: disabled)
   Active: inactive (dead)
[root@localhost ~]# service rabbitmq-server start
Redirecting to /bin/systemctl start rabbitmq-server.service
[root@localhost ~]# service rabbitmq-server status
Redirecting to /bin/systemctl status rabbitmq-server.service
● rabbitmq-server.service - RabbitMQ broker
   Loaded: loaded (/usr/lib/systemd/system/rabbitmq-server.service; enabled; vendor preset: disabled)
   Active: active (running) since 2019-10-23 17:11:21 CST; 6s ago
 Main PID: 25899 (beam.smp)
  Status: "Initialized"
   CGroup: /system.slice/rabbitmq-server.service
           └─25899 /usr/lib64/erlang/erts-9.3.3.11/bin/beam.smp -W w -A 64 -P 1048576 -t 5000000 -stbt d
             └─26040 /usr/lib64/erlang/erts-9.3.3.11/bin/epmd -daemon
               └─26176 erl_child_setup 1024
                 └─26199 inet_gethost 4
                   └─26200 inet_gethost 4
```

#停止服务

service rabbitmq-server stop

配置RabbitMQ

4.1 找到配置文件

```
Copyright (c) 2007-2018 Pivotal Software, Inc.  
Licensed under the MPL. See http://www.rabbitmq.com/  
2018-08-04 16:17:56.146 [info] <0.207.0>  
node           : rabbit@yky  
home dir       : /var/lib/rabbitmq  
config file(s) : (none)  
cookie hash    : xy7BSDdtG0wsW5CEeaA/QA==  
log(s)         : /var/log/rabbitmq/rabbit@yky.log  
                : /var/log/rabbitmq/rabbit@yky_upgrade.log  
database dir   : /var/lib/rabbitmq/mnesia/rabbit@yky  
2018-08-04 16:17:58.423 [info] <0.215.0> Memory high watermark set  
total
```

启动RabbitMQ后，会在 /var/log/rabbitmq 目录下生成运行日志，在日志的最上方，我们可以看到配置文件的信息：

然而，发现刚开始的时候并没有任何配置文件。只好照着官方的介绍，到

/usr/share/doc/rabbitmq-server-3.7.7/ 目录下复制一份模板到 /etc/rabbitmq 目录下进

行修改

```
cp /usr/share/doc/rabbitmq-server-3.6.15/rabbitmq.config.example  
/etc/rabbitmq/rabbitmq.config
```

这里值得提一下，从3.7.0开始，rabbitmq的配置文件引入了一个新的配置文件 rabbitmq.conf（rabbitmq.conf配置模板），该文件使用 key= value的格式进行配置，使配置文件看起来更加简洁易懂。当两个配置文件都存在的时候，rabbitmq只会读取旧版的配置文件。另外，官方还提到在 /usr/share/doc/rabbitmq-server-3.7.7/ 目录下，还有一个名为 advanced.config.example 的配置文件模板，用来配置rabbitmq的一些高级属性，但是我并没有找到，后面发现他已经整合到 rabbitmq.conf.example的文件末尾了。

虽然新版配置文件很简洁，但在配置LDAP Support时需要比较复杂的数据结构，所以我们将该配置放在advanced.config配置。rabbitmq在启动的时候能同时加载这两个配置文件。

主要端口说明

4369 -- erlang发现口
5672 --client端通信口
15672 -- 管理界面ui端口
25672 -- server间内部通信口

配置端口放开：

```
#vim /etc/sysconfig/iptables
```

```
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
-A INPUT -p icmp -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 22 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 5672 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 15672 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 25672 -j ACCEPT
-A INPUT -p tcp -m state --state NEW -m tcp --dport 4369 -j ACCEPT
-A INPUT -j REJECT --reject-with icmp-host-prohibited
-A FORWARD -j REJECT --reject-with icmp-host-prohibited
COMMIT
```

```
/sbin/iptables -I INPUT -p tcp --dport 端口 -j ACCEPT
```

举例说明

1. 我们访问RabbitMQ管理界面可以访问如下: <http://localhost:15672/>
2. 我们连接RabbitMQ的话就要用client端通信口: server:
amqp://guest:guest@localhost:5672/
3. 如果用其它协议, 还有其它口点击查看文档说明_
<http://www.rabbitmq.com/networking.html>

四、安装Web管理界面插件 (具体步骤, 参考下图)

安装

`rabbitmq-plugins enable rabbitmq_management`

2) 设置RabbitMQ远程ip登录

创建账号, 密码 (标准: `rabbitmqctl add_user 账号 密码`)

`rabbitmqctl add_user 账号 密码`

3) 设置用户角色

`rabbitmqctl set_user_tags 账号 administrator`

4) 设置用户权限

`rabbitmqctl set_permissions -p "/" 账号 ".*" ".*" ".*"`

5) 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`

```
[root@localhost /]# rabbitmqctl list_users
Listing users ...
guest    [administrator]
qin      [administrator]
[root@localhost /]#
```

交换器列表

`rabbitmqctl list_exchanges` 能够列出服务器上所有的交换器:

```
[root@localhost rabbitmq]# rabbitmqctl list_exchanges
Listing exchanges ...
      direct
amq.direct      direct
amq.fanout       fanout
amq.headers      headers
amq.match        headers
amq.rabbitmq.log  topic
amq.rabbitmq.trace topic
amq.topic        topic
ex1             direct
exchange 1      direct
logs            fanout
[root@localhost rabbitmq]#
```

扇形交换机


这个列表中有一些叫做amq.*的交换器。这些都是默认创建的，不过这时候你还不需要使用他们。

直连交换机 (**direct**)，主题交换机 (**topic**)，头交换机 (**headers**)，扇型交换机 (**fanout**)。我们在这里主要说明最后一个 —— 扇型交换机 (fanout)

访问: <http://192.168.3.30:15672/>

192.168.3.30:15672

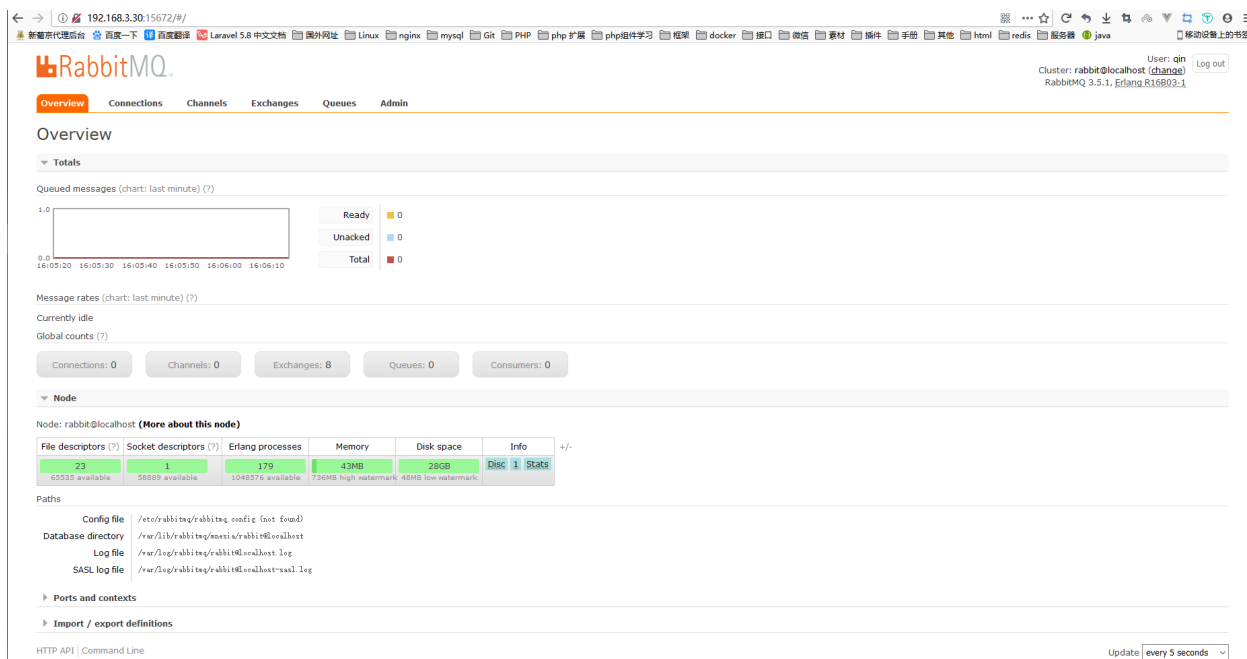
百度一下 百度翻译 Laravel 5.8 中文文档 国外网址 Linux nginx mysql Git PHP php 扩展 php组件学习 框架 docker 接口 微信 素材

 RabbitMQ

Username:

Password:

Login



至此算是成功了！！！！

注意：如果你在安装Web管理界面插件时，出现如下图错误时，说明你的（可参考：[rabbitmq与erlang之间的版本对应关系](#)）不对，导致报错。

```
centos7/systemd/rabbitmq.service
[root@localhost rabbitmq]# rabbitmq-plugins enable rabbitmq_management
=SUPERVISOR REPORT==== 30-Oct-2018::14:46:24.797991 ===
  supervisor: {local,'Elixir.Logger.Supervisor'}
  errorContext: start_error
  reason: noprocs
  offender: [{pid,undefined},
             {id,'Elixir.Logger.ErrorHandler'},
             {mfargs,
              {'Elixir.Logger.Watcher',watcher,
               [error_logger,'Elixir.Logger.ErrorHandler',
                {true,false,500}]}},
             {restart_type,permanent},
             {shutdown,5000},
             {child_type,worker}}]
=CRASH REPORT==== 30-Oct-2018::14:46:24.807705 ===
  crasher:
    initial call: application_master:init/4
    pid: <0.81.0>
    registered_name: []
    exception exit: {{shutdown,
                          {failed_to_start_child,'Elixir.Logger.ErrorHandler',
                           noprocs}},
                     {'Elixir.Logger.App',start,[normal,[]]}}
    in function application_master:init/4 (application_master.erl, line 136)
https://blog.csdn.net/qq_37495786
```

php使用安装RabbitMQ 需要安装php-amqp扩展

安装php-amqp扩展前必须安装rabbitmq-c

1.安装rabbitmq-c

```
wget https://github.com/alanxz/rabbitmq-c/releases/download/v0.7.1/rabbitmq-c-0.7.1.tar.gz
```

```
tar xzf rabbitmq-c-0.7.1.tar.gz && mv rabbitmq-c-0.7.1 rabbitmq-c && mv rabbitmq-c /usr/local/ && cd /usr/local/rabbitmq-c
```

```
./configure --prefix=/usr/local/rabbitmq-c  
make && make install
```

安装php-amqp扩展 (我用的php7.2版本)

```
wget https://pecl.php.net/get/amqp-1.9.4.tgz
```

```
tar -zxvf amqp-1.9.4.tgz && mv amqp-1.9.4 php-amqp && mv php-amqp /usr/local/ && cd /usr/local/php-amqp
```

```
phpize
```

```
./configure --with-php-config=/usr/local/php/bin/php-config --with-amqp --with-librabbitmq-dir=/usr/local/rabbitmq-c
```

```
make && make install
```

```
echo 'extension=/usr/local/php/lib/php/extensions/no-debug-non-zts-20170718/amqp.so' >> /usr/local/php/etc/php.ini
```