平台-应用发布平台安装文档

## 一 客户环境

### 1.1服务端环境

#### 1.1.1硬件环境

服务器：（基本要求）4核CPU，内存8G以上，硬盘256G

网卡：1000M Ethernet

#### 1.1.2软件环境

操作系统：centos7.2

数据库：5.5.47-MariaDB

JDK版本：1.8.0\_65

Tomcat：8.0.30

### 1.2客户机环境

#### 1.2.1硬件环境

Cpu：双核

内存：2G以上

硬盘：256G以上

网卡：100M/1000M Ethernet

#### 1.2.2软件环境

操作系统：WINDOWS 系统皆可

浏览器：chrome 45+，Firefox 30+，IE10+

### 1.3 容器后台环境

#### 1.3.1硬件环境

Cpu：双核/单核

内存：2G以上

硬盘：256G以上

网卡：100M/1000M Ethernet

#### 1.3.2软件环境

操作系统：centos7.2

Docker版本：1.11

Swarm版本：swarm version 1.0.0 (HEAD)

Registry版本：docker-distribution-2.3.0

Nginx版本：1.10.0

## 二 工具安装

本平台使用的工具包主要是：jdk 1.8.0\_65，5.5.47-MariaD，Tomcat：8.0.30

，下面采用图文的方式介绍具体的安装过程

### 2.1 JDK 1.8.0\_65安装详解

安装过程说明

#### 2.1.1下载JDK1.8

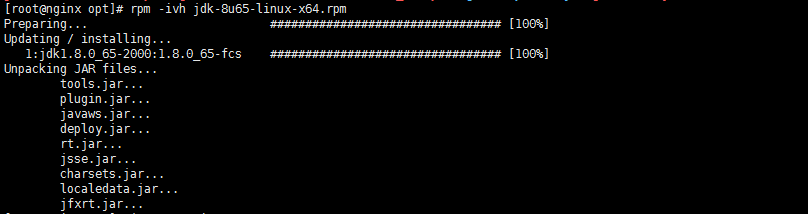
下载地址为：

#### <http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

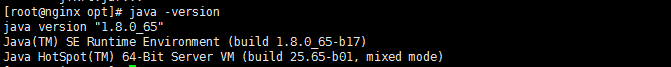
已下载好的存放位置：\\192.168.1.14\public\中移POC\搭建环境依赖包\jdk+tomcat

#### 2.1.2安装JDK1.8

进入jdk rpm包目录，运行rpm –ivh jdk-8u65-linux-x64.rpm



#### 2.1.3 测试java



### 2.2 MariaDB安装详解

CentOS7的yum源中默认好像是没有mysql的。为了解决这个问题，我们要先下载MariaDB

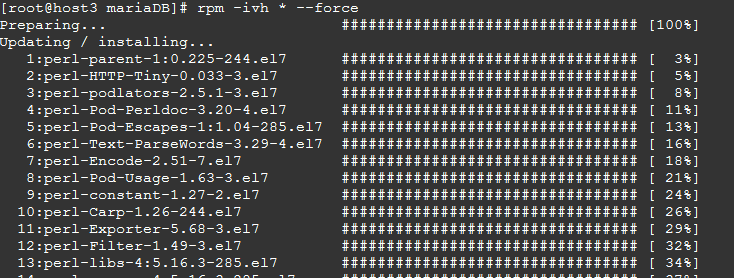
#### 2.2.1准备mariadb依赖包

依赖包位置：\\192.168.1.14\public\中移POC\搭建环境依赖包\mariaDB

#### 2.2.2安装

进入mariadb rpm包所在目录

rpm –ivh \* --force



#### 2.2.4启动

[root@host3 mariaDB]# systemctl start mariadb

[root@host3 mariaDB]# systemctl enable mariadb

#### 2.2.5设置密码为root权限

[root@host3 mariaDB]# mysql -uroot

Welcome to the MariaDB monitor. Commands end with ; or \g.

Your MariaDB connection id is 2

Server version: 5.5.47-MariaDB MariaDB Server

Copyright (c) 2000, 2015, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> use mysql

Reading table information for completion of table and column names

You can turn off this feature to get a quicker startup with -A

Database changed

MariaDB [mysql]> GRANT ALL PRIVILEGES ON \*.\* TO 'root'@'%' IDENTIFIED BY 'password' WITH GRANT OPTION;

Query OK, 0 rows affected (0.00 sec)

MariaDB [mysql]> GRANT ALL PRIVILEGES ON \*.\* TO root@localhost IDENTIFIED BY 'password' WITH GRANT OPTION;

Query OK, 0 rows affected (0.01 sec)

MariaDB [mysql]> FLUSH PRIVILEGES;

Query OK, 0 rows affected (0.00 sec)

MariaDB [mysql]> exit

测试

[root@host3 mariaDB]# mysql -uroot -ppassword

Welcome to the MariaDB monitor. Commands end with ; or \g.

Your MariaDB connection id is 3

Server version: 5.5.47-MariaDB MariaDB Server

Copyright (c) 2000, 2015, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

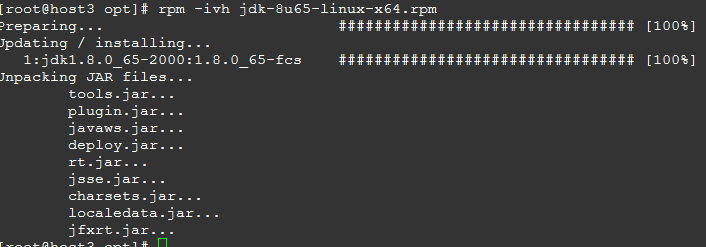
MariaDB [(none)]> exit

### 2.3 Tomcat安装详解

用winscp工具copy到发布平台的主机上面。Tomcat安装包位置：

\\192.168.1.14\public\中移POC\搭建环境依赖包\jdk+tomcat

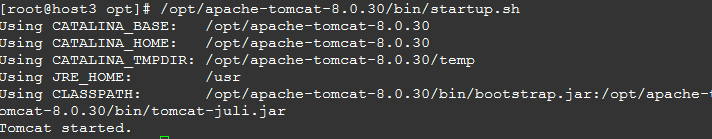
#### 2.3.1安装 jdk



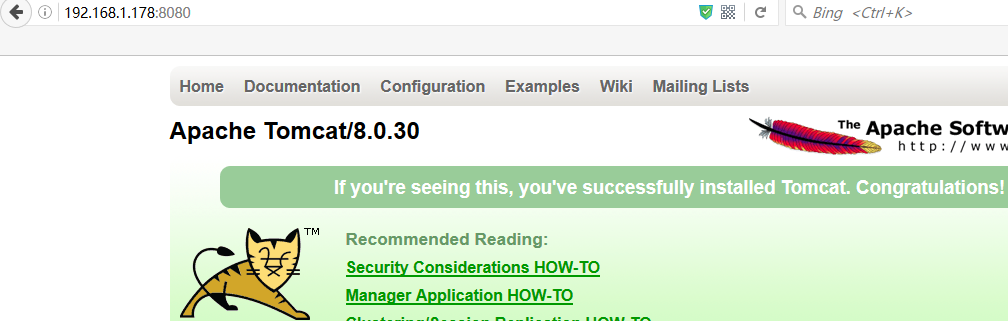
#### 2.3.2解压tomcat

[root@host3 opt]# tar -zxvf apache-tomcat-8.0.30.tar.gz

#### 2.3.3启动tomcat



测试，看到如下画面说明以及安装成功。



如果访问不到，可能是防火墙的问题。systemctl stop firewalld

### 2.4 容器组件安装详解

提前准备好docker 容器所需要的安装包

1. 所有主机需安装docker engine 以及监控软件
2. 仓库主机额外安装docker-distribution
3. Swarm主机额外安装 swarm

4.所有的docker主机都需要安装swarm

#### 2.4.1Docker engine的安装

包括docker主机，swarm主机以及仓库主机都需要安装。

安装包为

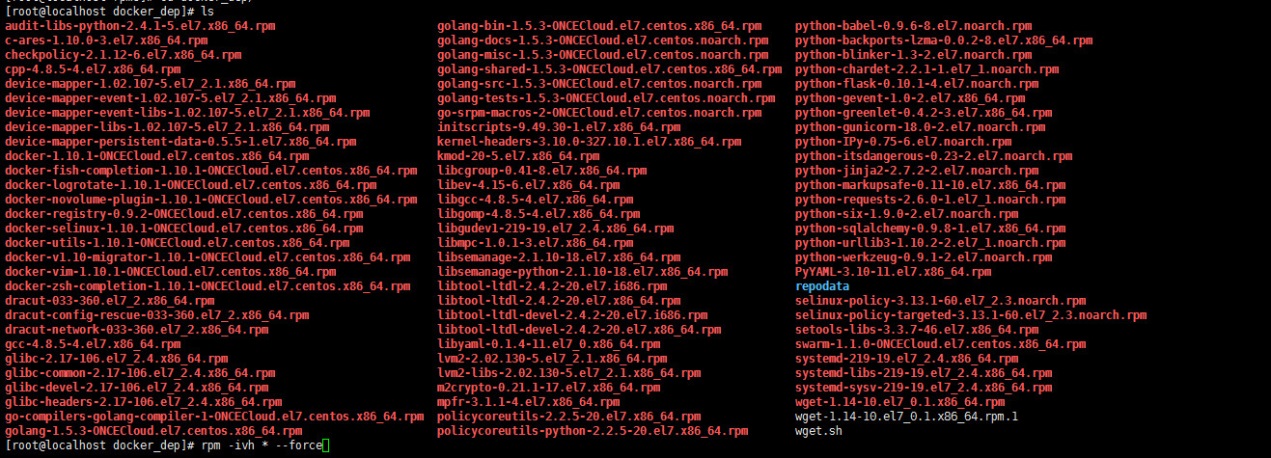
docker-engine-1.11.1-1.el7.centos.x86\_64.rpm

docker-engine-selinux-1.11.1-1.el7.centos.noarch.rpm

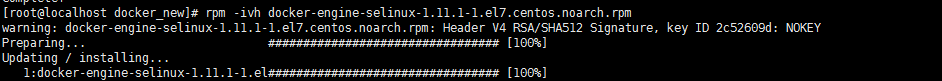
依赖包为docker\_dep中的包

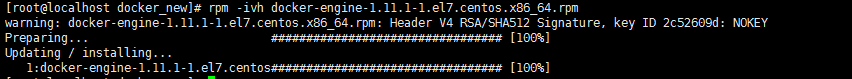
依赖包安装包存放位置：\\192.168.1.14\public\中移POC\搭建环境依赖包\docker

1. 安装依赖包

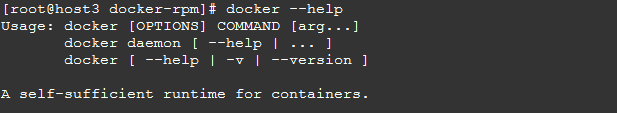


1. 安装docker selinux 以及docker engine



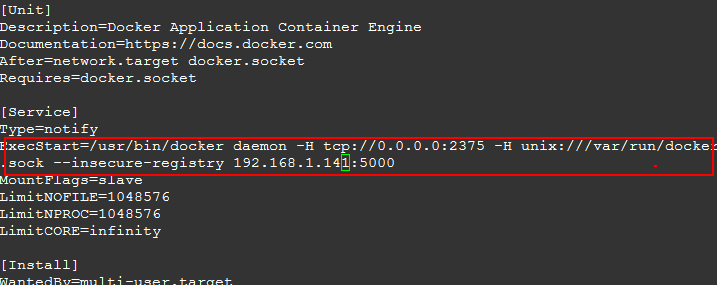


1. 测试安装成功



1. 修改启动参数

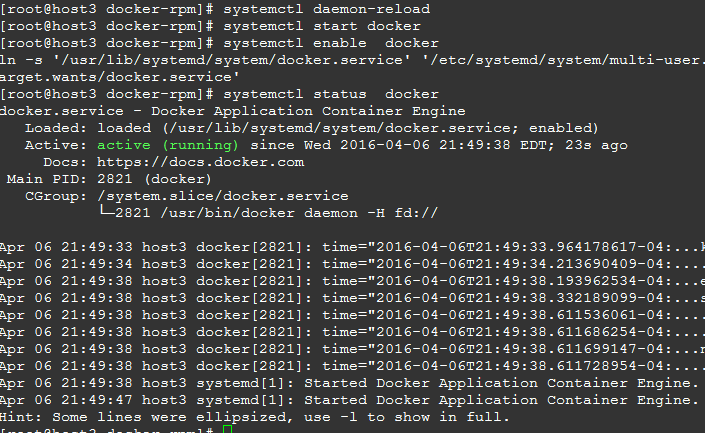
/usr/lib/systemd/system/docker.service



-H tcp://0.0.0.0:2375 -H unix:///var/run/docker.sock --insecure-registry

--insecure-registry 的ip为仓库主机的ip

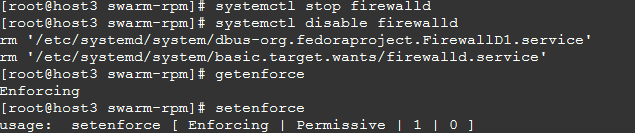
1. 启动docker 守护进程并且设置开机自动启动



1. 测试命令

docker info

1. 为了测试方便，关闭主机防火墙以及selinux



/etc/selinux/config 设置 SELINUX=disabled

#### 2.4.2swarm主机的安装

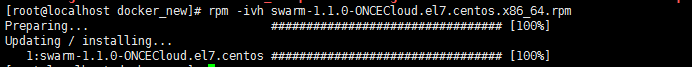
swarm是用来管理调度docker集群的来自docker官方的工具，swarm主机需要安装net-tools工具，用户自动获取集群端口

安装包为

swarm-1.1.0-ONCECloud.el7.centos.x86\_64.rpm

安装包存放位置：\\192.168.1.14\public\中移POC\搭建环境依赖包\swarm

swarm没有依赖包，直接安装就可以，安装完以后不需要启动swarm



#### 2.4.3仓库主机的安装

仓库主机是用来管理以及存放docker 镜像,仓库主机需要安装unzip命令

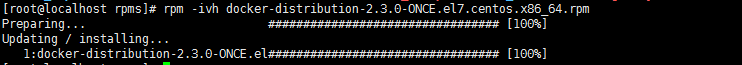
安装包为

docker-distribution-2.3.0-ONCE.el7.centos.x86\_64.rpm

默认是5000端口

安装包存放位置：\\192.168.1.14\public\中移POC\搭建环境依赖包\registry

1. docker-distribution的安装



1. 启动仓库进程



1. 测试

[root@localhost]# curl http://192.168.1.169:5000/v2/\_catalog

{"repositories":[]}

1. mkdir /home/sgcc

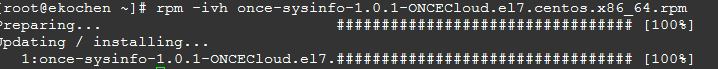
#### 2.4.4监控工具的安装

安装包为

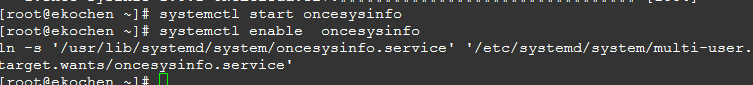
once-sysinfo-1.0.1-ONCECloud.el7.centos.x86\_64.rpm

安装包存放位置：\\192.168.1.14\public\中移POC\搭建环境依赖包\once-sysinfo

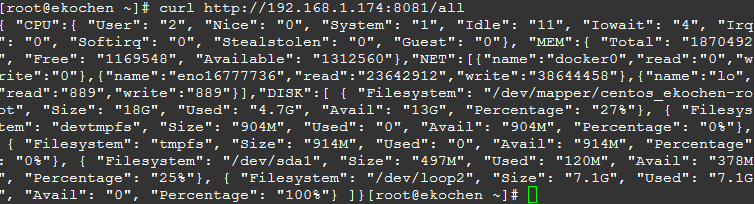
1. 安装



1. 启动 服务



1. 测试(ip为监控工具安装的机器ip地址，默认端口8081)



#### 2.4.5Nginx安装

Nginx用来管理发布应用的负载均衡

安装包为

nginx-filesystem-1.10.0-cmbc.el7.centos.noarch.rpm

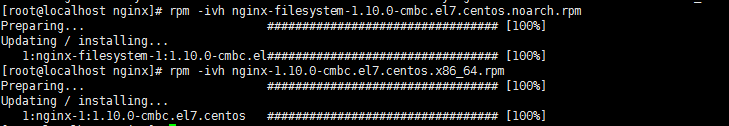
nginx-1.10.0-cmbc.el7.centos.x86\_64.rpm

依赖包为nginx\_dep中的包

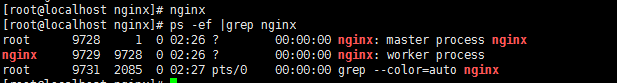
安装包位置：

\\192.168.1.14\public\中移POC\搭建环境依赖包\nginx

1. 安装



1. 启动



### 2.5 zookeeper安装详解

用于swarm集群的服务注册,zookeeper机器需要jdk环境

准备至少三台机器做zookeeper的集群，也可以在一台机器上做伪集群

安装包

zookeeper-3.4.8.tar.gz

安装包位置：\\192.168.1.14\public\中移POC\搭建环境依赖包\zookeeper

#### 2.5.1解压zookeeper包

把zookeeper的包放到任意一个目录，进行解压（如果在一台机器上搭建伪集群，需要copy到三个目录下）

tar -axvf zookeeper-3.4.8.tar.gz

#### 2.5.2设置存储目录

为每个zk创建zk的dataDir目录（/home/zkData/server-1,其他分别为server-2/3），并新建myid文件，里面写上对应的数字，如server-1里面就写1，其他分别为2,3

#### 2.5.3设置集群配置文件

以server-1为例，其他操作相同，

进入server-1的conf目录，新建一个zoo.cfg的文件，或者把zoo\_sample.cfg文件改成zoo.cfg

tickTime=2000

dataDir=/home/zkData/server-1 //步骤2.5.2中的dataDir目录，server2和3做相应的修改

initLimit=10

syncLimit=5

clientPort=2181 //监听端口, 如果是一台机器上的伪集群，修改下监听端口，如2182,2183

不同的机器无需修改

server.1=192.168.1.168:2888:3888 //192.168.1.168为zookeeper服务器的ip地址 server.2=192.168.1.188:2889:3889

server.3=192.168.1.88:2890:3890

#### 2.5.4启动zookeeper

进入bin目录，启动zk，如进入server-1/bin(需要三个都启动)

sh zkServer.sh start

#### 2.5.5查看zk状态

sh zkServer.sh status

如果启动异常，可以在bin\zookeeper.out里面查看错误日志

cat zookeeper.cou

### 2.6 sendmail邮件服务器搭建详解

\\192.168.1.14\public\中移POC\搭建环境依赖包\邮箱服务器，安装sendmail-8.14.7-4.el7.x86\_64、sendmail-cf-8.14.7-4.el7.noarch、dovecot.x86\_64、cyrus-sasl-2.1.26-19.2.el7.x86\_64、mailx

建议：该测试邮箱服务器最好和web平台搭建在一台机器上

#### 2.6.1安装邮件服务相关软件包

1. dovecot.x86\_64 : Secure imap and pop3 server

2. sendmail-8.14.7-4.el7.x86\_64 : A widely used Mail Transport Agent (MTA)

3. sendmail-cf-8.14.7-4.el7.noarch : The files needed to reconfigure Sendmail

4. cyrus-sasl-2.1.26-19.2.el7.x86\_64 : The Cyrus SASL library

5. mailx: 将邮件发往本地的MTA(Mail Transfer Agent)

#### 2.6.2修改配置文件

1.修改vi /etc/mail/sendmail.mc （把原来的文件内容中包含下面内容的注释的释放出来即可，不要全量覆盖）

# egrep -v '^dnl' /etc/mail/sendmail.mc

divert(-1)dnl

include(`/usr/share/sendmail-cf/m4/cf.m4')dnl

VERSIONID(`setup for linux')dnl

OSTYPE(`linux')dnl

define(`confDEF\_USER\_ID', ``8:12'')dnl

define(`confTO\_CONNECT', `1m')dnl

define(`confTRY\_NULL\_MX\_LIST', `True')dnl

define(`confDONT\_PROBE\_INTERFACES', `True')dnl

define(`PROCMAIL\_MAILER\_PATH', `/usr/bin/procmail')dnl

define(`ALIAS\_FILE', `/etc/aliases')dnl

define(`STATUS\_FILE', `/var/log/mail/statistics')dnl

define(`UUCP\_MAILER\_MAX', `2000000')dnl

define(`confUSERDB\_SPEC', `/etc/mail/userdb.db')dnl

define(`confPRIVACY\_FLAGS', `authwarnings,novrfy,noexpn,restrictqrun')dnl

define(`confAUTH\_OPTIONS', `A')dnl

TRUST\_AUTH\_MECH(`EXTERNAL DIGEST-MD5 CRAM-MD5 LOGIN PLAIN')dnl

define(`confAUTH\_MECHANISMS', `EXTERNAL GSSAPI DIGEST-MD5 CRAM-MD5 LOGIN PLAIN')dnl

define(`confTO\_IDENT', `0')dnl

FEATURE(`no\_default\_msa', `dnl')dnl

FEATURE(`smrsh', `/usr/sbin/smrsh')dnl

FEATURE(`mailertable', `hash -o /etc/mail/mailertable.db')dnl

FEATURE(`virtusertable', `hash -o /etc/mail/virtusertable.db')dnl

FEATURE(redirect)dnl

FEATURE(always\_add\_domain)dnl

FEATURE(use\_cw\_file)dnl

FEATURE(use\_ct\_file)dnl

FEATURE(local\_procmail, `', `procmail -t -Y -a $h -d $u')dnl

FEATURE(`access\_db', `hash -T<TMPF> -o /etc/mail/access.db')dnl

FEATURE(`blacklist\_recipients')dnl

EXPOSED\_USER(`root')dnl

DAEMON\_OPTIONS(`Port=smtp,Addr=0.0.0.0, Name=MTA')dnl

FEATURE(`accept\_unresolvable\_domains')dnl

LOCAL\_DOMAIN(`localhost.localdomain')dnl

MAILER(smtp)dnl

MAILER(procmail)dnl

2. 生成新的配置文件

# m4 sendmail.mc > sendmail.cf //生成配置

# vi /etc/mail/local-host-names //加入邮件域名

bocloudtest.com

#### 2.6.3修改主机名称

# vi /etc/sysconfig/network

HOSTNAME=bocloudtest.com

# vi /etc/hostname

bocloudtest.com

# vi /etc/hosts

127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4

::1 localhost localhost.localdomain localhost6 localhost6.localdomain6

* + - 1. bocloudtest.com //10.254.10.18为服务器ip地址

#### 2.6.4加入域名REPLAY

//允许发邮件到bocloud.com

# vi /etc/mail/access

Connect:localhost.localdomain RELAY

Connect:localhost RELAY

Connect:127.0.0.1 RELAY

Connect:bocloudtest.com RELAY //加入该行

# makemap hash /etc/mail/access.db < /etc/mail/access

# db\_dump -p /etc/mail/access.db

# db\_dump -p /etc/mail/access.db

......

connect:bocloudtest.com

RELAY

connect:localhost

RELAY

DATA=END

# grep '^protocols' /etc/dovecot/dovecot.conf

vi /etc/dovecot/dovecot.conf

protocols = imap imaps pop3 pop3s lmtp

2.6.5测试SMTP验证  
 # useradd tester  
 # passwd tester//添加邮箱帐号

# saslauthd -v

saslauthd 2.1.26

authentication mechanisms: getpwent kerberos5 pam rimap shadow ldap httpform

# saslpasswd2 -u bocloudtest.com -c tester

# sasldblistusers2

tester@bocloudtest.com: userPassword

删除用户  
 # saslpasswd2 -d [mailtest@bocloudtest.com](mailto:mailtest@bocloudtest.com)

1. Check an start sendmail, dovecot, saslauthd

# systemctl status sendmail //如果没有启动，则先启动 systemctl start sendmail

# lsof -i:25 //如果没有lsof命令，需要yum install lsof

COMMAND PID USER FD TYPE DEVICE SIZE/OFF NODE NAME

sendmail 18958 root 4u IPv4 10452762 0t0 TCP \*:smtp (LISTEN)

# systemctl status saslauthd //如果没有启动，则先启动systemctl start saslauthd

● saslauthd.service - SASL authentication daemon.

Loaded: loaded (/usr/lib/systemd/system/saslauthd.service; enabled; vendor preset: disabled)

Active: active (running) since Tue 2016-06-21 17:16:26 CST; 39min ago

# systemctl status dovecot //如果没有启动，则先启动systemctl start dovecot

● dovecot.service - Dovecot IMAP/POP3 email server

Loaded: loaded (/usr/lib/systemd/system/dovecot.service; enabled; vendor preset: disabled)

Active: active (running) since Tue 2016-06-21 17:15:46 CST; 40min ago

# lsof -i:143

COMMAND PID USER FD TYPE DEVICE SIZE/OFF NODE NAME

dovecot 19215 root 37u IPv4 10462244 0t0 TCP \*:imap (LISTEN)

dovecot 19215 root 38u IPv6 10462245 0t0 TCP \*:imap (LISTEN)

1. 测试发送邮件

# mail -s 'Test mail for tester' [tester@bocloudtest.com](mailto:tester@bocloudtest.com) //输入该命令后，需要手动输入邮件内容，以英文“.”结束输入内容

===============

Welcome

===============

Hello Tester,

.

EOT

1. 测试接收邮件

# mail -u tester //如果该账户下没有，则修改为mail –u root

Heirloom Mail version 12.5 7/5/10. Type ? for help.

"/var/mail/tester": 3 messages 1 new

1 root Tue Jun 21 18:29 21/731 "YYYY"

2 root Tue Jun 21 18:30 21/751 "YYYY2"

>N 3 root Tue Jun 21 18:34 23/800 "Test mail for tester"

& 3

Message 3:

From root@bocloudtest.com Tue Jun 21 18:34:16 2016

Return-Path: <root@bocloudtest.com>

From: root <root@bocloudtest.com>

Date: Tue, 21 Jun 2016 18:34:16 +0800

To: tester@bocloudtest.com

Subject: Test mail for tester

User-Agent: Heirloom mailx 12.5 7/5/10

Content-Type: text/plain; charset=us-ascii

Status: R

===============

Welcome

===============

Hello Tester,

& q //q为结束查看符

## 三 应用部署

前提条件：准备平台-应用发布平台必要的应用程序包ROOT.war和数据库初始化的poc.sql文件

### 3.1导入初始化数据库

把数据库初始化的sql文件上传到mysql数据库所在的物理服务器目录下，例如/opt/

下面示例均已/opt目录为代表

通过工具(如：xftp )把poc.sql上传到数据库服务器，

上传完成后登陆mysql

[root@admin~]mysql –u username –p password

登陆成功以后,新建数据库poc

mysql>create database poc;

数据库建立完成以后，导入原始的sql文件

mysql>source /opt/poc.sql

等待导入完成，数据库初始化完成。

### 3.2部署应用程序

关闭tomcat

把应用程序包ROOT.war上传到Tomcat的webapps,如/opt/apache-tomcat-8.0.30/webapps/

通过工具(如：xftp )把ROOT.war上传到到Tomcat的webaps目录下

启动tomcat

[root@admin]cd /opt/apache-tomcat-8.0.30/bin/

[root@admin]./startup.sh

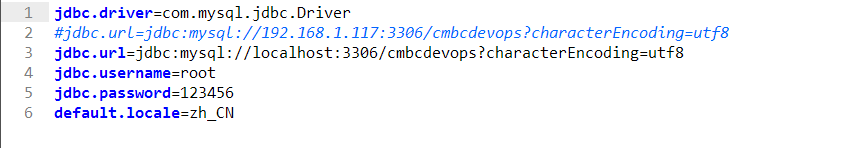
备注：由于未修改配置文件，启动时会报错误日志，建议在把war包上传到webapps目录后，先解压（unzip ROOT.war –d ROOT ,解压完成后，删除war包），再按照3.3中的步骤修改下配置文件再启动

### 3.3修改应用程序配置文件

修改应用程序的配置文件

[root@admin] /opt/apache-tomcat-8.0.30/webapps/ROOT/WEB-INF/classes/config

#### 3.3.1修改数据库配置文件（jdbc.properties）



参数说明

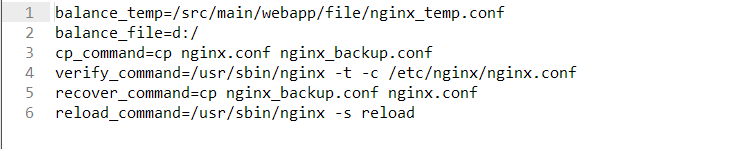
jdbc.driver mysql中java连接的数据库驱动

jdbc.url代表 mysql数据库的连接地址，通用格式表示如下语句所示：jdbc:mysql://ip:port/dbname?characterEncoding=utf8

jdbc.username 连接数据库用户名

jdbc.password 连接数据库密码

#### 3.3.2修改负载均衡nginx的配置文件(nginx.config)



该文件只需要修改nginx配置文件生成临时balance\_file（配置文件临时的位置）和temp文件路径

参数说明：

balance\_temp表示nginx配置文件的模板，模板存放在：

[\\192.168.1.14\public\中移POC\搭建环境依赖包\nginx](file:///\\192.168.1.14\public\中移POC\搭建环境依赖包\nginx)中的nginx\_temp.conf

balance\_file表示nginx临时配置文件的位置

cp\_command 表示备份nginx原始配置文件

verify\_command 表示校验新生成的nginx配置文件

recover\_command 表示回滚nginx配置文件

reload\_command 表示重新加载配置文件的命令

#### 3.3.3修改邮件发送的配置文件（email.properties）

|  |
| --- |
| **email.host**=smtp.sina.com.cn  **email.from**=zhansan@sina.com  **email.account**=zhansan@sina.com  **email.password**=123456 |

参数内容说明：

email.host：邮件smtp服务器的地址；步骤2.6中设置，如bocloudtest.com

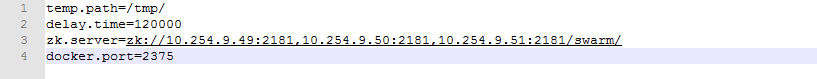
email.from：发件人的邮箱地址；如tester@bocloudtest.com

email.account：管理员的邮箱账号地址；如tester

email.password：登陆到邮箱的密码。如userPassword

**邮箱也可以配163的邮箱服务器信息以及账户**

#### 3.3.4修改服务注册配置文件（system.properties）



参数内容说明：

temp.path:配置文件路径，不需要更改

delay.time:延迟时间，不需要更改

zk.server：zookeeper集群信息，把ip和端口更改为搭建的zk集群的ip地址和端口

docker.port:docker端口，无需修改

#### 3.3.5启动Tomcat

修改配置文件以后，重新启动Tomcat应用

[root@admin] cd /opt/apache-tomcat-8.0.30/bin/

[root@admin] ./startup.sh

部署完成。

#### 3.4.6部署结果验证

当部署完成后，访问地址<http://localhost:8080>,出现下图所示界面表示部署成功



初始化用户名：admin

初始化密码：123456

登录成功后界面如下图所示：

