**奇安信网神RDS V4.0**

**安装手册**

**网神信息技术（北京）股份有限公司**

**2019年05月**

目录

[1. 前言 4](#_Toc11322636)

[2. 机器准备 4](#_Toc11322637)

[3. 准备配置库 5](#_Toc11322638)

[3.1. 设置配置库的linux系统 5](#_Toc11322639)

[3.1.1 查看linux 系统的发行版 5](#_Toc11322640)

[3.1.2 查看配置库机器的网卡 6](#_Toc11322641)

[3.1.3 查看防火墙设置 7](#_Toc11322642)

[3.1.4 防火墙添加一个临时规则 9](#_Toc11322643)

[3.1.5 将系统语言设置为英文 10](#_Toc11322644)

[3.1.6 修改系统时间为上海时间 并对时间进行同步 10](#_Toc11322645)

[3.1.7 修改系统默认java版本(这部分可以跳过) 11](#_Toc11322646)

[3.1.8 修改系统内核参数 12](#_Toc11322647)

[3.1.9 修改 limits.conf 13](#_Toc11322648)

[3.1.10 修改pam 文件 13](#_Toc11322649)

[3.1.11 增加用户和组 13](#_Toc11322650)

[3.1.12 修改oracle用户配置文件 14](#_Toc11322651)

[3.1.13 创建文件夹 并赋予访问权限 14](#_Toc11322652)

[3.1.14 安装依赖库 15](#_Toc11322653)

[3.2. 静默安装 Oracle( RDS安装可以跳过) 15](#_Toc11322654)

[3.2.1 复制安装文件 15](#_Toc11322655)

[3.2.2 静默安装 15](#_Toc11322656)

[3.2.3 手动创建监听文件 16](#_Toc11322657)

[3.2.4 dbca 创建数据库 17](#_Toc11322658)

[3.2.5 安装 rlwrap 19](#_Toc11322659)

[3.2.6 设置为开机自启动 19](#_Toc11322660)

[3.3. 创建用户,并指定独立的表空间 20](#_Toc11322661)

[4. 安装FTP 环境 22](#_Toc11322662)

[5. 配置主机docker环境 23](#_Toc11322663)

[5.1. 离线安装docker 23](#_Toc11322664)

[5.1.1 检查日志采集软件主机的linux 环境 23](#_Toc11322665)

[5.1.2 离线安装docker环境 24](#_Toc11322666)

[5.2. 导入镜像文件 25](#_Toc11322667)

[5.3. 创建日志采集软件相关文件夹 25](#_Toc11322668)

[5.4. 启动容器 26](#_Toc11322669)

[5.4.1 通过镜像生成容器 26](#_Toc11322670)

[5.4.2 将数据库结构写入配置库 27](#_Toc11322671)

[5.4.3 修改client端的配置信息 28](#_Toc11322672)

[5.4.4 启动生成的容器 29](#_Toc11322673)

[5.4.5 配置route 30](#_Toc11322674)

[6. 认证, 输入备案信息 30](#_Toc11322675)

# 前言

本文档是奇安信网神实时数据同步系统安装文档, 旨在帮助运维人员安装数据同步系统。

# 机器准备

|  |  |  |  |
| --- | --- | --- | --- |
|  | 操作系统 | 主机名 | 端口 |
| 数据采集软件(主机) | Centos 6.6 final x86-64 | machine01 | 开放18000、8000、8181端口 |
| 数据采集软件(备机) | Centos 6.6 final x86-64 | machine02 | 开放18000、8000、8181端口 |
| 配置库 | oracle单节点 | machine03 | 开放1521端口 |
| FTP 服务器 | Centos 6.6 final x86-64 | machine04 | 开放21端口 |
| 大数据机器(节点1) | Centos 6.6 final x86-64 | machine05 | 开放6808端口 |
| 大数据机器(节点2) | Centos 6.6 final x86-64 | machine06 | 开放6808端口 |

备注: 大数据机器为无锡所大数据平台使用, 另外单独安装, 本文档不做说明。

# 准备配置库

## 设置配置库的linux系统

### 3.1.1 查看linux 系统的发行版

查看操作系统版本

|  |
| --- |
| **[root@cent03 etc]#** cat /etc/centos-release |
| *CentOS release 6.6 (Final)* |

查看系统发行版

|  |
| --- |
| **[root@cent03 etc]#** uname –a |
| *Linux cent03 2.6.32-504.el6.x86\_64 #1 SMP Wed Oct 15 04:27:16 UTC 2014 x86\_64 x86\_64 x86\_64 GNU/Linux* |
| **[root@cent03 etc]#** lsb\_release –a |
| *LSB Version: :base-4.0-amd64:base-4.0-noarch:core-4.0-amd64:core-4.0-noarch:graphics-4.0-amd64:graphics-4.0-noarch:printing-4.0-amd64:printing-4.0-noarch*  *Distributor ID: CentOS*  *Description: CentOS release 6.6 (Final)*  *Release: 6.6*  *Codename: Final* |

查看yum 缓存的位置

|  |
| --- |
| **[root@cent04 ~]**# vim /etc/yum.conf |
| *[main] cachedir=/var/cache/yum/$basearch/$releasever keepcache=0 debuglevel=2 logfile=/var/log/yum.log exactarch=1 obsoletes=1 gpgcheck=1 plugins=1 installonly\_limit=5 bugtracker\_url=http://bugs.centos.org/set\_project.php?project\_id=19&ref=http://bugs.centos.org/bug\_report\_page.php?category=yum distroverpkg=centos-release* |

查看cpu 物理个数

|  |
| --- |
| **[root@cent04 ~]#** grep 'physical id' /proc/cpuinfo | sort -u | wc -l |

查看cpu 核心数量

|  |
| --- |
| **[root@cent04 ~]#** grep 'core id' /proc/cpuinfo | sort -u | wc -l |

查看cpu 线程数

|  |
| --- |
| **[root@cent04 ~]#** grep 'processor' /proc/cpuinfo | sort -u | wc -l |

查看cpu 详细信息

|  |
| --- |
| **[root@cent04 ~]#** cat /proc/cpuinfo |

### 3.1.2 查看配置库机器的网卡

查看hosts 文件是否已经正确配置

|  |
| --- |
| **[root@cent03 etc]#** vim /etc/hosts |
| *192.168.111.149 cent01*  *192.168.111.152 cent02*  *192.168.111.155 cent03*  *192.168.111.156 cent04* |

查看网卡设置

|  |
| --- |
| **[root@cent03 ~]#** cd /etc/sysconfig/network-scripts/ |
| **[root@cent03 network-scripts]#** vim /etc/sysconfig/network-scripts/ifcfg-eth0 |
| *DEVICE=eth0*  *HWADDR=00:0C:29:58:ED:71*  *TYPE=Ethernet*  *UUID=70d9b97a-3285-44a5-bd7d-da50d0ab5c31*  *ONBOOT=yes*  *NM\_CONTROLLED=yes*  *BOOTPROTO=dhcp*  *IPADDR=192.168.111.155*  *GATEWAY=192.168.111.2* |

确认网卡已经设置为开机自启动,并且IP和网关已经正确配置

查看机器 hostname

|  |
| --- |
| **[root@cent03 oracle]#** vim /etc/sysconfig/network |
| *NETWORKING=yes*  *HOSTNAME=cent03* |

### 3.1.3 查看防火墙设置

确认1521端口已经开放访问权限

关闭iptables和防火墙 (不建议)

|  |
| --- |
| **[root@cent03 oracle]#** vim /etc/sysconfig/selinux |
| *SELINUX=disabled*  *# SELINUXTYPE= can take one of these two values:*  *# targeted - Targeted processes are protected,*  *# mls - Multi Level Security protection.*  *SELINUXTYPE=targeted* |

查看防火墙状态

|  |
| --- |
| **[root@cent03 oracle]#** service iptables status |
| *Table: filter Chain INPUT (policy ACCEPT) num target prot opt source destination  1 ACCEPT all -- 0.0.0.0/0 0.0.0.0/0 state RELATED,ESTABLISHED  2 ACCEPT icmp -- 0.0.0.0/0 0.0.0.0/0  3 ACCEPT all -- 0.0.0.0/0 0.0.0.0/0  4 ACCEPT tcp -- 0.0.0.0/0 0.0.0.0/0 state NEW tcp dpt:22  5 REJECT all -- 0.0.0.0/0 0.0.0.0/0 reject-with icmp-host-prohibited   Chain FORWARD (policy ACCEPT) num target prot opt source destination  1 REJECT all -- 0.0.0.0/0 0.0.0.0/0 reject-with icmp-host-prohibited   Chain OUTPUT (policy ACCEPT) num target prot opt source destination* |

关闭防火墙

|  |
| --- |
| **[root@cent04 ~]**# service iptables stop |
| *iptables: Setting chains to policy ACCEPT: filter [ OK ] iptables: Flushing firewall rules: [ OK ] iptables: Unloading modules:* |

另一种临时关闭防火墙的方法

|  |
| --- |
| **[root@cent04 ~]**# /etc/init.d/iptables status |
| **[root@cent04 ~]**# /etc/init.d/iptables stop |

永久关闭防火墙的方法

|  |
| --- |
| **[root@cent04 ~]**# chkconfig --list iptables |
| **[root@cent04 ~]**# chkconfig iptables off |

查看防火墙运行状态

|  |
| --- |
| **[root@cent04 ~]**# chkconfig --list iptables |
| *iptables 0:off 1:off 2:on 3:on 4:on 5:on 6:off* |

### 3.1.4 防火墙添加一个临时规则

查看宿主机iptables的状态

|  |
| --- |
| **[root@cent04 ~]**# service iptales status |
|  |
| **[root@cent04 ~]**# service iptales start |
|  |
| **[root@cent04 ~]**# iptables -nvL ; iptables -t nat -nvL |
|  |

添加一个临时规则

|  |
| --- |
| **[root@cent04 ~]**# iptables -t nat -F POSTROUTING; iptables -t nat -A POSTROUTING -s 172.16.0.0/16 ! -o docker0 -j MASQUERADE |
| **[root@cent04 ~]**# iptables -nvL ; iptables -t nat -nvL |
|  |

查看容器里面的路由

|  |
| --- |
| **[root@qssec ~]**# ifconfig ; route -n |
|  |
| **[root@qssec ~]**# route add default gw 172.17.42.1 |
|  |

查看宿主机状态

|  |
| --- |
| **[root@qssec ~]**# /proc/sys/net/ipv4/ip\_forward |
|  |
| **[root@qssec ~]**# echo 1 >/proc/sys/net/ipv4/ip\_forward |
|  |

重启docker 服务

|  |
| --- |
| **[root@qssec ~]**# docker stop qssec |
|  |
| **[root@qssec ~]**# service docker restart |
|  |

进入容器, 查看是否可以telnet 外部接口

|  |
| --- |
| **[root@qssec ~]**# telnet 192.168.111.137 1521 |
|  |

### 3.1.5 将系统语言设置为英文

|  |
| --- |
| **[root@cent03 network-scripts]#** cd /etc/sysconfig/ |
| **[root@cent03 sysconfig]#** vim /etc/sysconfig/i18n |
| *LANG="en\_US.UTF-8"* |
| **[root@cent03 sysconfig]#** source /etc/sysconfig/i18n |

确认当前系统语言已经设置为英文

### 3.1.6 修改系统时间为上海时间 并对时间进行同步

|  |
| --- |
| **[root@cent03 sysconfig]#** date |
| **[root@cent03 sysconfig]#** cd /etc/sysconfig |
| **[root@cent03 sysconfig]#** vim /etc/sysconfig/clock |
| *ZONE="Asia/Shanghai"*  *UTC=false*  *ARC=false* |

然后对时间进行同步

|  |
| --- |
| **[root@cent03 sysconfig]#** ln -sf /usr/share/zoneinfo/Asia/Shanghai /etc/localtime |
| **[root@cent03 sysconfig]#** date -s '2018-01-02 03:00:00' |
| **[root@cent03 sysconfig]#** clock -w |
| **[root@cent03 sysconfig]#** hwclock |

然后查看当前时间是否已经同步

|  |
| --- |
| **[root@cent03 sysconfig]#** date |

Docker容器和外面的宿主机时间进行同步

|  |
| --- |
| **[root@qssec qsrds]#** echo "Asia/Shanghai" > /etc/timezone && \ |
| **[root@qssec qsrds]#** ln -sf /usr/share/zoneinfo/Asia/Shanghai /etc/localtime |

### 3.1.7 修改系统默认java版本(这部分可以跳过)

|  |
| --- |
| **[root@cent04 2]**# rpm -qa|grep gcj |
| **[root@cent04 2]**# rpm -qa|grep jdk |
| *java-1.6.0-openjdk-1.6.0.0-11.1.13.4.el6.x86\_64 java-1.7.0-openjdk-1.7.0.65-2.5.1.2.el6\_5.x86\_64* |

然后卸载系统自带的jdk

|  |
| --- |
| *yum -y remove java java-1.6.0-openjdk-1.6.0.0-11.1.13.4.el6.x86\_64 yum -y remove java java-1.7.0-openjdk-1.7.0.65-2.5.1.2.el6\_5.x86\_64* |

上传压缩包,并进行解压缩

|  |
| --- |
| **[root@cent04 2]**# mv jdk1.7.0\_80/ /usr/share/ |
| **[root@cent04 2]**# cd /usr/share/jdk1.7.0\_80/ |

加载修改完成的环境变量

|  |
| --- |
| **[root@cent03 etc]#** vim /etc/profile |
| *export JAVA\_HOME=/usr/share/jdk1.7.0\_80*  *export PATH=$JAVA\_HOME/bin:$PATH*  *export CLASSPATH=.:$JAVA\_HOME/lib/dt.jar:$JAVA\_HOME/lib/tools.jar*  *if [ $USER == "oracle" ]; then*  *if [ $SHELL == "/bin/ksh" ]; then*  *ulimit -p 16384*  *ulimit -n 65536*  *else*  *ulimit -u 16384 -n 65536*  *fi*  *fi* |
| **[root@cent03 etc]#** source /etc/profile |

设置默认java版本

|  |
| --- |
| *alternatives --install /usr/bin/java java /usr/share/jdk1.7.0\_80/bin/java 500* |

或者按如下方式进行设置

|  |
| --- |
| **[root@cent04 jdk1.7.0\_80]**# alternatives --config java |
| *There is 1 program that provides 'java'.   Selection Command ----------------------------------------------- \*+ 1 /usr/share/jdk1.7.0\_80/bin/java  Enter to keep the current selection[+], or type selection number: 1* |

### 3.1.8 修改系统内核参数

|  |
| --- |
| **[root@cent03 etc]#** vim /etc/sysctl.conf |
| *fs.aio-max-nr = 1048576*  *fs.file-max = 6815744*  *kernel.shmall = 2097152*  *kernel.shmmax = 536870912*  *kernel.shmmni = 4096*  *kernel.sem = 250 32000 100 128*  *net.ipv4.ip\_local\_port\_range = 9000 65500*  *net.core.rmem\_default=4194304*  *net.core.rmem\_max=4194304*  *net.core.wmem\_default=262144*  *net.core.wmem\_max=1048586* |
| **[root@cent03 etc]#** sysctl -p |

### 3.1.9 修改 limits.conf

|  |
| --- |
| **[root@cent03 security]#** vim /etc/security/limits.conf |
| *oracle soft nproc 2047*  *oracle hard nproc 16384*  *oracle soft nofile 1024*  *oracle hard nofile 65536*  *oracle soft stack 10240*  *root soft nofile 65535*  *root hard nofile 65535* |

### 3.1.10 修改pam 文件

|  |
| --- |
| **[root@cent03 pam.d]#** vim /etc/pam.d/login |
| *session required /lib64/security/pam\_limits.so* |

### 3.1.11 增加用户和组

|  |
| --- |
| **[root@cent04 jdk1.7.0\_80]**# groupadd oinstall |
| **[root@cent04 jdk1.7.0\_80]**# groupadd dba |
| **[root@cent04 jdk1.7.0\_80]**# groupadd oper |
| **[root@cent04 jdk1.7.0\_80]**# useradd -g oinstall -G dba,oper oracle |
| **[root@cent04 jdk1.7.0\_80]**# passwd oracle |

### 3.1.12 修改oracle用户配置文件

|  |
| --- |
| **[root@cent03 pam.d]#** su - oracle |
| **[oracle@cent03 ~]$** vim ~/.bash\_profile |
| *export ORACLE\_BASE=/u01/app/oracle export ORACLE\_HOME=$ORACLE\_BASE/product/11.2.4/db\_1 export ORACLE\_SID=orcl155 export NLS\_LANG=AMERICAN\_AMERICA.ZHS16GBK export LD\_LIBRARY\_PATH=$LD\_LIBRARY\_PATH:$ORACLE\_HOME/lib:$OGG\_HOME:/lib:/usr/lib  PATH=$PATH:$HOME/.local/bin:$HOME/bin:$ORACLE\_HOME/bin:/usr/sbin:$PATH:$OGG\_HOME export PATH   stty erase ^? alias sqlplus="rlwrap sqlplus" alias rman="rlwrap rman"    #PATH=$PATH:$HOME/.local/bin:$HOME/bin:$ORACLE\_HOME/bin:/usr/sbin:$PATH:$OGG\_HOME #export PATH* |
| **[oracle@cent03 ~]$** source ~/.bash\_profile |

### 3.1.13 创建文件夹 并赋予访问权限

|  |
| --- |
| **[root@cent03 pam.d]#** mkdir -p /u01/app/oracle |
| **[root@cent03 pam.d]#** cd /u01/app/oracle |
| **[root@cent03 oracle]#** mkdir -p /u01/app/oracle/product/11.2.4/db\_1 |
| **[root@cent03 oracle]#** mkdir -p /u01/app/oraInventory |
| **[root@cent03 oracle]#** chown -R oracle:oinstall /u01 |

### 3.1.14 安装依赖库

将依赖包上传到指定位置

|  |
| --- |
| rpm -ivh \*.rpm --force --nodeps |

安装完成后重启机器, **用oracle用户登陆**(必需,否则环境变量的设置会有问题)

## 静默安装 Oracle( RDS安装可以跳过)

### 3.2.1 复制安装文件

上传安装包,并解压缩

|  |
| --- |
| **[root@cent04 2]**# unzip p13390677\_112040\_Linux-x86-64\_1of7.zip |
| **[root@cent04 2]**# unzip p13390677\_112040\_Linux-x86-64\_2of7.zip |

移动到oracle 用户, 并修改所属组

|  |
| --- |
| **[root@cent04 2]**# mv database/ /home/oracle/ |
| **[root@cent04 2]**# cd /home/oracle |
| **[root@cent04 oracle]**# chown -R oracle:oinstall database/ |

### 3.2.2 静默安装

用root用户登陆

|  |
| --- |
| **[root@cent04 oracle]**# xhost + |
| **[root@cent04 oracle]**# export LANG=US\_en |
| **[root@cent03 home]#** su - oracle |
| **[oracle@cent03 database]$** vim runInstaller |
| *if [ `$UNAME` = "Linux" ]; then*  *if [ -e $GETCONF ]; then*  *value=`$GETCONF LONG\_BIT`*  *if [ $value != 64 ]; then*  *echo "\"You are attempting to install 64-bit Oracle on a 32-bit operating system. This is not supported and will not work.\"";*  *#exit 126; #corresponding to the exit code of oui*  *fi*  *fi*  *fi* |

修改db.rsp 文件

|  |
| --- |
| *ORACLE\_HOSTNAME=cent03*  *INVENTORY\_LOCATION=/u01/app/oraInventory*  *ORACLE\_HOME=/u01/app/oracle/product/11.2.4/db\_1*  *ORACLE\_BASE=/u01/app/oracle* |

开始执行静默安装 路径必须使用绝对路径

|  |
| --- |
| **[oracle@cent03 database]$** ./runInstaller -ignorePrereq -silent -responseFile /home/oracle/database/db.rsp |

root 用户执行两个命令

|  |
| --- |
| **[root@cent03 home]#** /u01/app/oraInventory/orainstRoot.sh |
| **[root@cent03 home]#** /u01/app/oracle/product/11.2.4/db\_1/root.sh |

### 3.2.3 手动创建监听文件

|  |
| --- |
| **[oracle@cent04 admin]**$ pwd |
| **[oracle@cent04 admin]**$ /u01/app/oracle/product/11.2.4/db\_1/network/admin |
| **[oracle@cent04 admin]**$ vim listener.ora |
| *LISTENER =  (DESCRIPTION\_LIST =  (DESCRIPTION =  (ADDRESS = (PROTOCOL = TCP)(HOST = cent04)(PORT = 1521))  (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1521))  )  )  ADR\_BASE\_LISTENER = /u01/app/oracle* |
| **[oracle@cent04 admin]**$ vim tnsnames.ora |
| *ORCL156 =  (DESCRIPTION =  (ADDRESS\_LIST =  (ADDRESS = (PROTOCOL = TCP)(HOST = cent04)(PORT = 1521))  )  (CONNECT\_DATA =  (SERVICE\_NAME = orcl156)  )  )* |

### 3.2.4 dbca 创建数据库

修改本地的配置文件

|  |
| --- |
| **[oracle@cent04 admin]**$ vim /home/oracle/database/response/dbca.rsp |
| *[GENERAL] RESPONSEFILE\_VERSION = "11.2.0" OPERATION\_TYPE = "createDatabase" [CREATEDATABASE] GDBNAME = "nifa" SID = "nifa" TEMPLATENAME = "General\_Purpose.dbc" STORAGETYPE=FS DATAFILEDESTINATION =/oracle/app/oradata RECOVERYAREADESTINATION=/oracle/app/flash\_recovery\_area CHARACTERSET = "AL32UTF8" NATIONALCHARACTERSET= "AL32UTF8" LISTENERS=LISTENER TOTALMEMORY = "7000" SYSPASSWORD = "oracle" SYSTEMPASSWORD = "oracle"* |

执行安装命令

|  |
| --- |
| **[oracle@cent04 admin]**$ $ORACLE\_HOME/bin/dbca -silent -responseFile /home/oracle/database/response/dbca.rsp |
| *Copying database files 1% complete 3% complete 11% complete 18% complete 26% complete 37% complete Creating and starting Oracle instance 40% complete 45% complete 50% complete 55% complete 56% complete 57% complete 60% complete 62% complete Completing Database Creation 66% complete 70% complete 73% complete 85% complete 96% complete 100% complete Look at the log file "/u01/app/oracle/cfgtoollogs/dbca/orcl156/orcl156.log" for further details.* |

### 3.2.5 安装 rlwrap

上传文件 并解压缩

|  |
| --- |
| **[root@cent04 rlwrap-0.42]**# ./configure |
| **[root@cent04 rlwrap-0.42]**# make && make install |

### 3.2.6 设置为开机自启动

设置为开机自启动

|  |
| --- |
| **[oracle@cent04 admin]**$ vim /etc/oratab |
| *orcl156:/u01/app/oracle/product/11.2.4/db\_1:Y* |

修改自启动的选项

|  |
| --- |
| **[oracle@cent04 ~]**$ vim $ORACLE\_HOME/bin/dbstart |
| *# Set path if path not set (if called from /etc/rc)*  *SAVE\_PATH=/bin:/usr/bin:/etc:${PATH} ; export PATH*  *SAVE\_LLP=$LD\_LIBRARY\_PATH*  *# First argument is used to bring up Oracle Net Listener*  *ORACLE\_HOME\_LISTNER=$ORACLE\_HOME* |

|  |
| --- |
| **[oracle@cent04 ~]**$ vim $ORACLE\_HOME/bin/dbshut |
| *# Set path if path not set (if called from /etc/rc)*  *SAVE\_PATH=/bin:/usr/bin:/etc:${PATH} ; export PATH*  *SAVE\_LLP=$LD\_LIBRARY\_PATH*  *# The this to bring down Oracle Net Listener*  *ORACLE\_HOME\_LISTNER=$ORACLE\_HOME* |

加入开机自启动 先备份/etc/rc.local文件

|  |
| --- |
| **[root@cent04 rlwrap-0.42]**# vim /etc/rc.local |
| *#!/bin/sh*  *#*  *# This script will be executed \*after\* all the other init scripts.*  *# You can put your own initialization stuff in here if you don't*  *# want to do the full Sys V style init stuff.*  *touch /var/lock/subsys/local*  *su - oracle -c "/u01/app/oracle/product/11.2.4/db\_1/bin/dbstart"*  *su - oracle -c "/u01/app/oracle/product/11.2.4/db\_1/bin/emctl start dbconsole"* |

配置完成后重启, 查看数据库是否启动完成

## 创建用户,并指定独立的表空间

创建表空间

|  |
| --- |
| **[root@cent03 ~]#** su - oracle |
| **[oracle@cent03 ~]**$ sqlplus / as sysdba |

创建用户

|  |
| --- |
| **[oracle@cent03 ~]**$ sqlplus / as sysdba |
| **SQL>** create user rds\_admin identified by \*\*\*\*\*\*; |
| **SQL>** grant connect,resource to rds\_admin; |

# 安装FTP 环境

# 配置主机docker环境

## 离线安装docker

### 5.1.1 检查日志采集软件主机的linux 环境

检查linux的发行版

|  |
| --- |
| **[root@cent01 ~]#** cat /etc/centos-release |
| *CentOS release 6.6 (Final)* |
| **[root@cent01 ~]#** uname -a |
| *Linux cent01 2.6.32-504.el6.x86\_64 #1 SMP Wed Oct 15 04:27:16 UTC 2014 x86\_64 x86\_64 x86\_64 GNU/Linux* |
| **[root@cent01 ~]#** lsb\_release -a |
| *LSB Version: :base-4.0-amd64:base-4.0-noarch:core-4.0-amd64:core-4.0-noarch:graphics-4.0-amd64:graphics-4.0-noarch:printing-4.0-amd64:printing-4.0-noarch*  *Distributor ID: CentOS*  *Description: CentOS release 6.6 (Final)*  *Release: 6.6*  *Codename: Final* |

查看hosts 文件按是否已经正确配置

|  |
| --- |
| **[root@cent01 ~]#** cat /etc/hosts |
| *192.168.111.149 cent01*  *192.168.111.152 cent02*  *192.168.111.155 cent03*  *192.168.111.156 cent04* |

查看网卡是否已经正确配置

|  |
| --- |
| **[root@cent03 network-scripts]#** vim /etc/sysconfig/network-scripts/ifcfg-eth0 |
| *DEVICE=eth0*  *HWADDR=00:0C:29:E3:21:93*  *TYPE=Ethernet*  *UUID=853d0759-ca5b-49c5-83a3-a7fee5bd3667*  *ONBOOT=yes*  *NM\_CONTROLLED=yes*  *BOOTPROTO=dhcp* |

查看系统时间是否已经正确设置

|  |
| --- |
| **[root@cent01 ~]#** date |
| *Tue Nov 6 14:15:59 CST 2018* |

### 5.1.2 离线安装docker环境

上传压缩包到指定位置, 并解压缩

|  |
| --- |
| **[root@cent01 get]#** ls |
| *device-mapper-1.02.117-12.el6.x86\_64.rpm device-mapper-libs-1.02.117-12.el6.x86\_64.rpm libcgroup-0.40.rc1-23.el6.x86\_64.rpm*  *device-mapper-event-1.02.117-12.el6.x86\_64.rpm device-mapper-persistent-data-0.6.2-0.1.rc7.el6.x86\_64.rpm*  *device-mapper-event-libs-1.02.117-12.el6.x86\_64.rpm docker-engine-1.7.1-1.el6.x86\_64.rpm* |
| **[root@cent01 get]#** rpm -ivh \* --force --nodeps |
| **[root@cent01 get]#** service docker start |
| *Starting docker: [ OK ]* |
| **[root@cent01 get]#** docker ps -a |

## 导入镜像文件

创建文件夹,用于存放镜像文件

|  |
| --- |
| **[root@cent01 home]#** mkdir -p /home/newdata |
| **[root@cent01 home]#** cd /home/newdata |

将当前容器导出镜像文件

|  |
| --- |
| **[root@cent01 home]#** cd /home/newdata |
| **[root@cent01 home]#** docker export -o icarus.tar qssec |
| **[root@cent01 home]#** tar -zcvf icarusXXXX.tar.gz icarus.tar |

上传镜像文件到指定位置, 查看镜像文件

|  |
| --- |
| **[root@cent01 newdata]#** cat /opt/newdata/icarus.tar |docker import - wangshen:latest |
| **[root@cent01 newdata]#** docker images -a |
| *[root@cent01 newdata]# docker images -a*  *REPOSITORY TAG IMAGE ID CREATED VIRTUAL SIZE*  *icarus latest 1a084f73bc6e 4 days ago 2.395 GB* |

## 创建日志采集软件相关文件夹

|  |
| --- |
| **[root@cent01 newdata]#**  mkdir -p /opt/newdata/rds  mkdir -p /opt/newdata/log  mkdir -p /opt/newdata/log/supervisor  mkdir -p /opt/newdata/log/qs\_log  mkdir -p /opt/newdata/log/nginx  mkdir -p /opt/newdata/log/redis  mkdir -p /opt/newdata/xml/  mkdir -p /opt/newdata/xml/pre  mkdir -p /opt/newdata/xml/back  mkdir -p /opt/newdata/xml/uploaded  mkdir -p /opt/newdata/temp  mkdir -p /opt/newdata/tmp |

创建一个run.sh 文件,并赋予执行权限

|  |
| --- |
| **[root@cent01 newdata]#** mkdir -p /opt/newdata/run.sh |
| *#!/bin/bash*  *source /root/.bashrc*  *rpcbind*  *python /usr/local/project/qsrds/master\_start.py*  */usr/sbin/crond*  *while [[ true ]]; do*  *sleep 1*  *done* |
| **[root@cent01 newdata]#** chmod 755 /opt/newdata/run.sh |

## 启动容器

### 5.4.1 通过镜像生成容器

|  |
| --- |
| **[root@cent01 newdata]#** cd /opt/newdata |
| **[root@cent01 newdata]#** docker run --privileged -h="wangshen2" --name wangshen2 -d --restart=always -p 18001:18000 -p 8001:8000 -p 8182:8181 -v /opt/newdata/:/data/ -v /opt/newdata/log/:/var/log/ -v /opt/newdata/tmp:/tmp/ --mac-address 50:98:B8:D2:EF:00 icarus:latest /bin/bash /data/run.sh |
| **[root@cent01 newdata]#** docker ps -a |
| *CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES*  *7af95800af77 icarus:latest "/bin/bash /data/run 4 days ago Up 16 minutes 0.0.0.0:8000->8000/tcp, 0.0.0.0:8181->8181/tcp, 0.0.0.0:18000->18000/tcp qssec* |
| **[root@cent01 newdata]#** docker exec -it qssec /bin/bash |
| **[root@qssec /]#** /usr/bin/supervisord -c /etc/supervisord.conf |
| **[root@qssec /]#** supervisorctl stop all |
| **[root@qssec /]#** supervisorctl status |
| **[root@qssec /]#** /usr/local/bin/redis-server & |
| **[root@qssec /]#** /usr/local/nginx/sbin/nginx |

### 5.4.2 将数据库结构写入配置库

登入安装Oracle数据库机器切换至安装Oracle的用户

例：su – oracle 默认安装Oracle数据库为oracle用户

进入数据库

sqlplus / as sysdba

创建表空间

 create tablespace rds\_peizhi datafile ‘/u01/app/oracle/oradata/orcl64/peizhi.dbf’ size 500M; 指定路径及文件需要手动创建 size大小根据实际情况填写

创建用户

create user rds\_peizhi identified by rds\_peizhi default tablespace rds\_peizhi;

给用户授权

grant connect,resource to rds\_peizhi;

grant select any dictionary to rds\_qiushi;

登入安装rds的机器

修改 local\_settings.py 文件, 修改为当前数据库的配置

|  |
| --- |
| **[root@qssec /]#** vim /usr/local/project/qsrds/qsrds/local\_settings.py |
| *# coding=utf-8*  *DATABASES = {*  *'default': {*  *'ENGINE': 'django.db.backends.oracle',*  *'NAME': 'orcl155',*  *'USER': 'rds\_admin',*  *'PASSWORD': '\*\*\*\*\*\*',*  *'HOST': '192.168.111.155',*  *'PORT': '1521',*  *#'CONN\_MAX\_AGE': 10 \* 60,*  *'OPTIONS': {'threaded': False}*  *}*  *}* |

通过测试脚本 test.py ,确认配置库可以正常连接

|  |
| --- |
| *import cx\_Oracle*  *resource\_ip = '192.168.111.156' resource\_port = 1521 sid = 'orcl156' resource\_user = 'rds\_big' resource\_password = 'wuxi2018'  try:  conn\_str = '%s:%s/%s' % (resource\_ip, resource\_port, sid)  if resource\_user.upper() == "SYS":  conn = cx\_Oracle.connect(resource\_user, resource\_password, conn\_str, cx\_Oracle.SYSDBA)   else:  conn = cx\_Oracle.connect(resource\_user, resource\_password, conn\_str)  except cx\_Oracle.DatabaseError as e:  # 加入日志信息  error, = e.args  print error  return error.code, str(error)* |

配置完成后, 建立表结构,并创建登陆用户

|  |
| --- |
| **[root@qssec /]#** cd /usr/local/project/qsrds/ |
| **[root@qssec /]#** python manage.py migrate |
| **[root@qssec /]#** python manage.py createsuperuser |
| *Username: xiaoming*  *Email address:*  *Password:*  *Password (again):*  *Superuser created successfully.* |

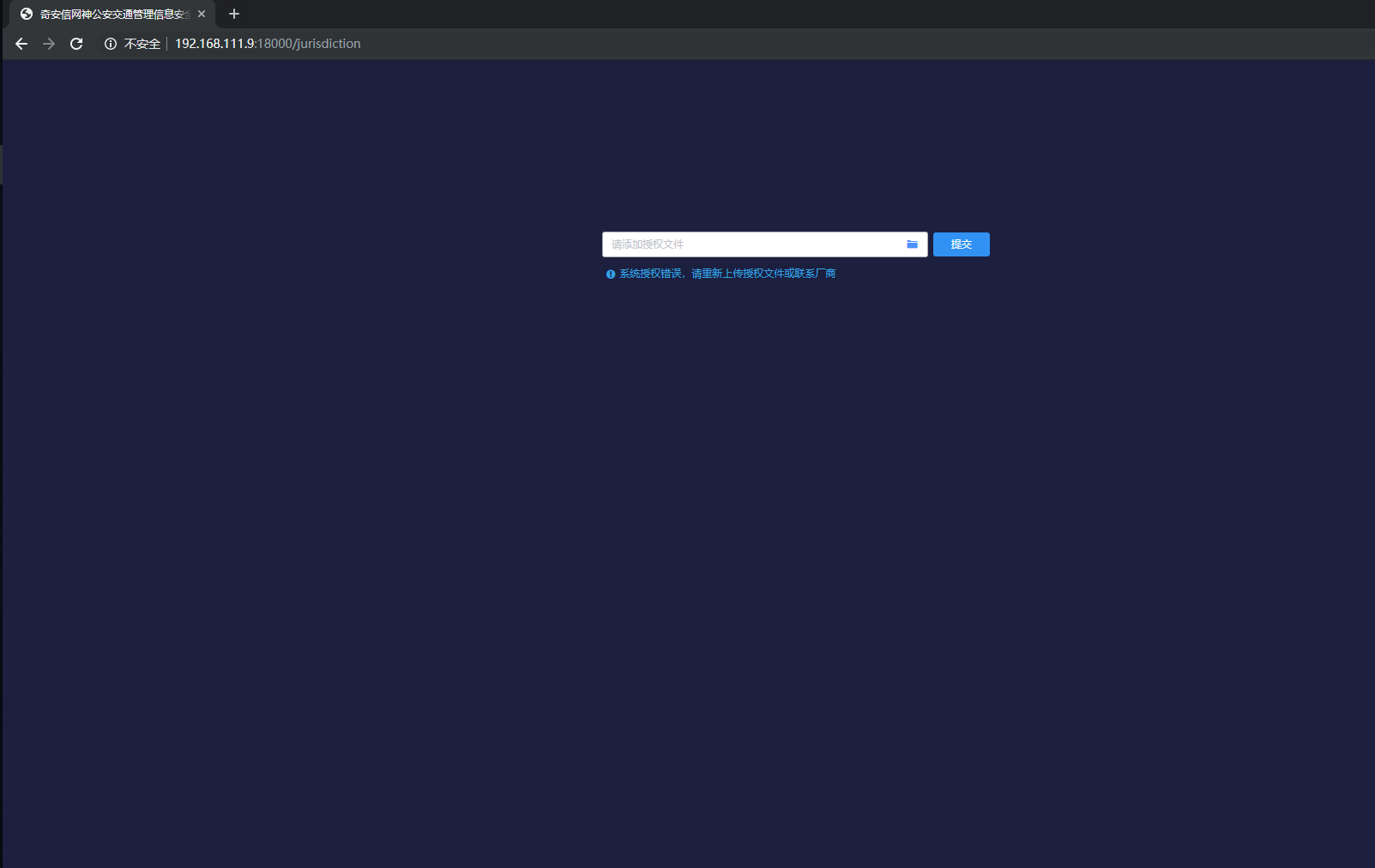
### 5.4.3 修改client端的配置信息

|  |
| --- |
| **[root@qssec /]#** cd /root/rds |
|  |
| *pid-file = /var/run/qs-client.pid*  *#lock file = /var/run/qs-agent.lock*  *log-file = /var/log/qs-client.log*  *[192.168.111.156-orcl156]*  *server-ip = 192.168.111.156*  *instance-name = orcl156*  *passwd = QssecK2b9HKTEbS5vNbto*  *auth-user = oracle*  *auth-user = oracle*  *arch-block = archblock*  *onli-block = onliblock*  *sync-local-path = /data/rds/*  *archlog-name = archivelog*  *onlilog-name = onlinelog*  *arch-fs-type = 0*  *onli-fs-type = 0*  *compress = 0*  *encrypt = 0*  *arch-switch = 1*  *onli-switch = 1*  *server-port = 8730* |
|  |

### 5.4.4 启动生成的容器

|  |
| --- |
| **[root@qssec /]#** supervisorctl start qsrds\_runserver |
| *qsrds\_runserver RUNNING pid 6726, uptime 0:00:02* |

能看到如下页面, 则容器创建完成



### 5.4.5 配置route

1 宿主机配置

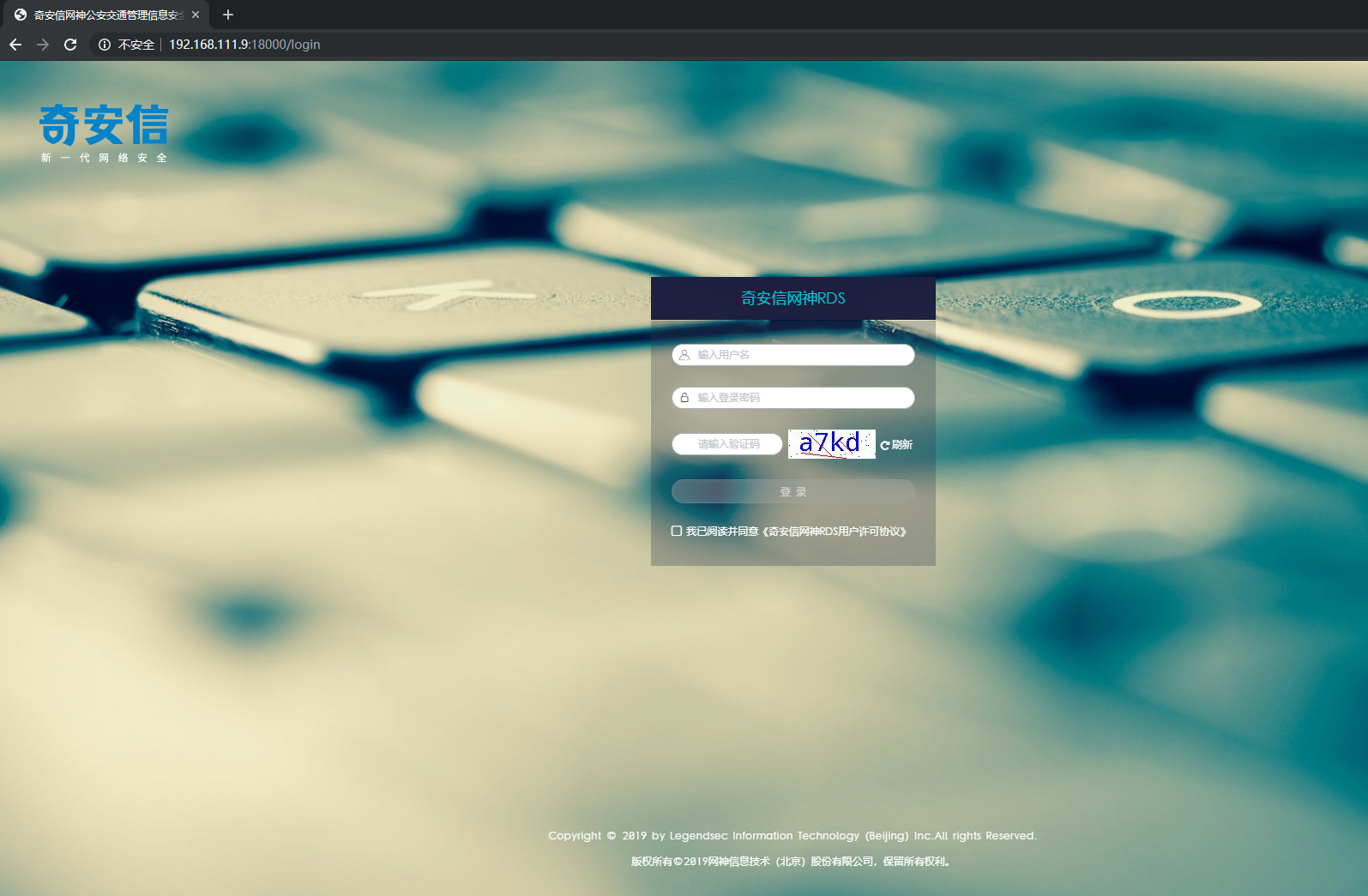
2 容器配置

# 认证, 输入备案信息

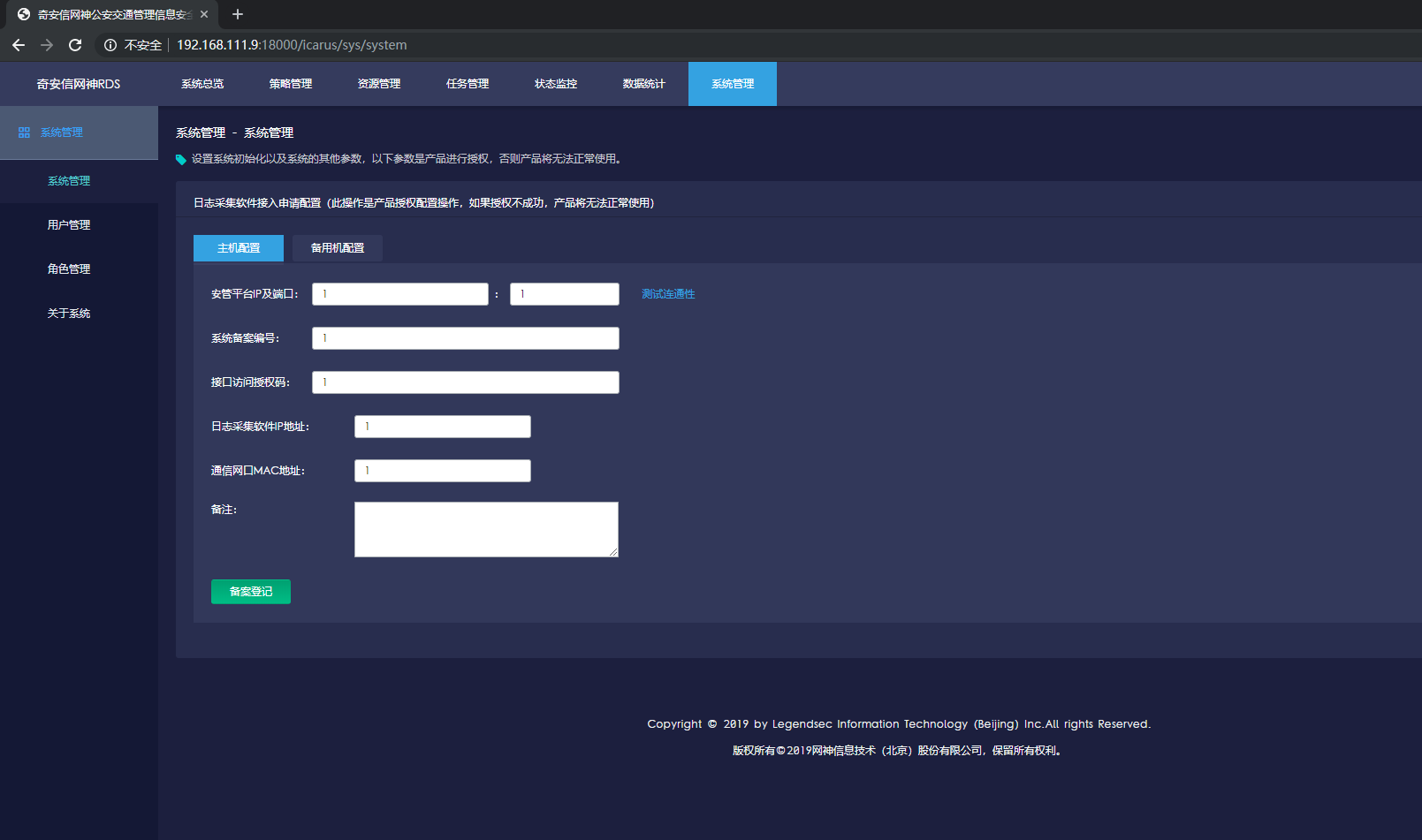
在docker 容器里面,获取当前机器的 uuid

|  |
| --- |
| **[root@qssec qsrds]#** dmidecode -s system-uuid |
| *4F9E4D56-60E6-74C9-F868-5E821AE32193* |

通过uuid , 生成license, 并通过前端页面进行上传,然后重新登陆前端页面



输入登陆信息 ,admin/\*\*\*\*\*\*, 从侧边栏选择: 系统管理 – 系统管理



输入api 接口的相关配置信息, 并进行保存, 安装工作完成。

异常处理

|  |  |
| --- | --- |
| 现象 | api 注册异常 |
| 处理方法 | *curl C:\Users\qssec\AppData\Local\Temp\%W@GJ$ACOF(TYDYECOKVDYB.png10.94.86.215:6808/bigweb/restSafeAccess.ews?method=queryRdsOut* |
| 结果 | 查看网络的连通性 |