CentOS 6.5下PostgreSQL 9.4.3安装与配置

目录

[CentOS 6.5下PostgreSQL 9.4.3安装与配置 1](#_Toc496263614)

[一、 简介 1](#_Toc496263615)

[二、 系统环境 1](#_Toc496263616)

[三、 源码安装 2](#_Toc496263617)

[1. 下载PostgreSQL 源码包 2](#_Toc496263618)

[2. 解压源码包 2](#_Toc496263619)

[3. 进入解压后的目录 2](#_Toc496263620)

[4. 安装依赖包 2](#_Toc496263621)

[5. 增加用户设置密码 2](#_Toc496263622)

[6. 开始编译安装PostgreSQL 数据库。 2](#_Toc496263623)

[7. 设置环境变量 2](#_Toc496263624)

[8. 初始化数据库 3](#_Toc496263625)

[9. 系统服务 3](#_Toc496263626)

[10. 测试使用 9](#_Toc496263627)

1. 简介

PostgreSQL 是一种非常复杂的对象-关系型数据库管理系统（ORDBMS），也是目前功能最强大，特性最丰富和最复杂的自由软件数据库系统。有些特性甚至连商业数据库都不具备。这个起源于伯克利（BSD）的数据库研究计划目前已经衍生成一项国际开发项目，并且有非常广泛的用户。

1. 系统环境

系统平台：[CentOS](http://www.linuxidc.com/topicnews.aspx?tid=14) release 6.5 (Final)  
PostgreSQL 版本：PostgreSQL 9.4.3  
防火墙已关闭/iptables: Firewall is not running.  
SELINUX=disabled

1. 源码安装
2. 下载PostgreSQL 源码包  
   # wget <http://ftp.postgresql.org/pub/source/v9.4.3/postgresql-9.4.3.tar.bz>2
3. 解压源码包  
   # tar xjf postgresql-9.4.3.tar.bz2
4. 进入解压后的目录  
   # cd postgresql-9.4.3
5. 安装依赖包  
   yum -y install gcc\*  
   yum -y install readline-devel
6. 增加用户设置密码  
   [root@postgresql ~]# adduser postgres  
   [root@postgresql ~]# passwd postgres  
   Changing password for user postgres.  
   New password:   
   BAD PASSWORD: it is too simplistic/systematic  
   BAD PASSWORD: is too simple  
   Retype new password:   
   passwd: all authentication tokens updated successfully.
7. 开始编译安装PostgreSQL 数据库。  
   [root@postgresql ~]# cd postgresql-9.4.3  
   [root@postgresql postgresql-9.4.3]# ./configure --prefix=/home/postgres/pgsql  
      
   [root@postgresql postgresql-9.4.3]# gmake  
      
   [root@postgresql postgresql-9.4.3]# gmake install
8. 设置环境变量  
   [root@postgresql ~]# cd /home/postgres/  
   [root@postgresql postgres]# ls  
   pgsql  
   [root@postgresql postgres]# vi .bash\_profile  
   把 PATH=$PATH:$HOME/bin  
   改成 PATH=$PATH:$HOME/bin:/home/postgres/pgsql/bin  
   保存退出。  
   让环境变量生效：  
   在设置  
   [root@postgresql ~]# vi .bash\_profile  
   把 PATH=$PATH:$HOME/bin  
   改成 PATH=$PATH:$HOME/bin:/home/postgres/pgsql/bin  
   保存退出。  
   让环境变量生效：  
   [root@postgresql ~]# source .bash\_profile
9. 初始化数据库

8.1新建数据目录  
[root@postgresql ~]# mkdir /home/postgres/pgsql/data

8.2 更改权限  
[root@postgresql ~]# chown postgres:postgres /home/postgres/pgsql/data

8.3 切换到postgres用户  
[root@postgresql ~]# su – postgres

8.4 init db  
[postgres@postgresql ~]$ /home/postgres/pgsql/bin/initdb -D /home/postgres/pgsql/data  
到这里数据的初始化就完成

1. 系统服务

9.1回到root用户  
[postgres@postgresql ~]$ exit  
9.2复制安装目录下的linux文件到/etc/init.d/  
进入postgresql 的安装目录（即刚刚使用tar命令解压的目录）  
[root@postgresql ~]# cd postgresql-9.4.3  
[root@postgresql postgresql-9.4.3]# cp contrib/start-scripts/linux /etc/init.d/postgresql  
9.3修改/etc/init.d/postgresql  注意：红色是修改部分

[root@postgresql postgresql-9.4.3]# vi /etc/init.d/postgresql  
#! /bin/sh  
   
# chkconfig: 2345 98 02  
# description: PostgreSQL RDBMS  
   
# This is an example of a start/stop script for SysV-style init, such  
# as is used on Linux systems.  You should edit some of the variables  
# and maybe the 'echo' commands.  
#  
# Place this file at /etc/init.d/postgresql (or  
# /etc/rc.d/init.d/postgresql) and make symlinks to  
#  /etc/rc.d/rc0.d/K02postgresql  
#  /etc/rc.d/rc1.d/K02postgresql  
#  /etc/rc.d/rc2.d/K02postgresql  
#  /etc/rc.d/rc3.d/S98postgresql  
#  /etc/rc.d/rc4.d/S98postgresql  
#  /etc/rc.d/rc5.d/S98postgresql  
# Or, if you have chkconfig, simply:  
# chkconfig --add postgresql  
#  
# Proper init scripts on Linux systems normally require setting lock  
# and pid files under /var/run as well as reacting to network  
# settings, so you should treat this with care.  
   
# Original author:  Ryan Kirkpatrick <[pgsql@rkirkpat.net](mailto:pgsql@rkirkpat.net)>  
   
# contrib/start-scripts/linux  
   
## EDIT FROM HERE  
   
# Installation prefix  
prefix=/home/postgres/pgsql  
   
# Data directory  
PGDATA="/home/postgres/pgsql/data"  
# Who to run the postmaster as, usually "postgres".  (NOT "root")  
PGUSER=postgres  
   
# Where to keep a log file  
PGLOG="$PGDATA/serverlog"  
   
# It's often a good idea to protect the postmaster from being killed by the  
# OOM killer (which will tend to preferentially kill the postmaster because  
# of the way it accounts for shared memory).  Setting the OOM\_SCORE\_ADJ value  
# to -1000 will disable OOM kill altogether.  If you enable this, you probably  
# want to compile PostgreSQL with "-DLINUX\_OOM\_SCORE\_ADJ=0", so that  
# individual backends can still be killed by the OOM killer.  
#OOM\_SCORE\_ADJ=-1000  
# Older Linux kernels may not have /proc/self/oom\_score\_adj, but instead  
# /proc/self/oom\_adj, which works similarly except the disable value is -17.  
# For such a system, enable this and compile with "-DLINUX\_OOM\_ADJ=0".  
#OOM\_ADJ=-17  
   
## STOP EDITING HERE  
   
# The path that is to be used for the script  
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin  
   
# What to use to start up the postmaster.  (If you want the script to wait  
# until the server has started, you could use "pg\_ctl start -w" here.  
# But without -w, pg\_ctl adds no value.)  
DAEMON="$prefix/bin/postmaster"  
   
# What to use to shut down the postmaster  
PGCTL="$prefix/bin/pg\_ctl"  
   
set -e  
   
# Only start if we can find the postmaster.  
test -x $DAEMON ||  
{  
        echo "$DAEMON not found"  
        if [ "$1" = "stop" ]  
        then exit 0  
        else exit 5  
        fi  
}  
   
   
# Parse command line parameters.  
case $1 in  
  start)  
        echo -n "Starting PostgreSQL: "  
        test x"$OOM\_SCORE\_ADJ" != x && echo "$OOM\_SCORE\_ADJ" > /proc/self/oom\_score\_adj  
        test x"$OOM\_ADJ" != x && echo "$OOM\_ADJ" > /proc/self/oom\_adj  
        su - $PGUSER -c "$DAEMON -D '$PGDATA' &" >>$PGLOG 2>&1  
        echo "ok"  
        ;;  
  stop)  
        echo -n "Stopping PostgreSQL: "  
        su - $PGUSER -c "$PGCTL stop -D '$PGDATA' -s -m fast"  
        echo "ok"  
        ;;  
  restart)  
        echo -n "Restarting PostgreSQL: "  
        su - $PGUSER -c "$PGCTL stop -D '$PGDATA' -s -m fast -w"  
        test x"$OOM\_SCORE\_ADJ" != x && echo "$OOM\_SCORE\_ADJ" > /proc/self/oom\_score\_adj  
        test x"$OOM\_ADJ" != x && echo "$OOM\_ADJ" > /proc/self/oom\_adj  
        su - $PGUSER -c "$DAEMON -D '$PGDATA' &" >>$PGLOG 2>&1  
        echo "ok"  
        ;;  
  reload)  
        echo -n "Reload PostgreSQL: "  
        su - $PGUSER -c "$PGCTL reload -D '$PGDATA' -s"  
        echo "ok"  
        ;;  
status)  
        su - $PGUSER -c "$PGCTL status -D '$PGDATA'"  
        ;;  
  \*)  
        # Print help  
        echo "Usage: $0 {start|stop|restart|reload|status}" 1>&2  
        exit 1  
        ;;  
esac  
   
exit 0

9.4添加执行权限  
[root@postgresql postgresql-9.4.3]# chmod +x /etc/init.d/postgresql  
   
9.5启动数据库  
[root@postgresql postgresql-9.4.3]# /etc/init.d/postgresql start  
Starting PostgreSQL: ok  
9.6让数据库开机启动  
[root@postgresql postgresql-9.4.3]# chkconfig --add postgresql  
[root@postgresql postgresql-9.4.3]# chkconfig postgresql on  
9.7创建数据库操作历史记录文件   
[root@postgresql postgresql-9.4.3]# touch /home/postgres/pgsql/.pgsql\_history  
[root@postgresql postgresql-9.4.3]# chown postgres:postgres /home/postgres/pgsql/.pgsql\_history

1. 测试使用  
   [root@postgresql postgresql-9.4.3]# su - postgres  
   [postgres@postgresql ~]$ createdb test  
   [postgres@postgresql ~]$ psql test  
   psql (9.4.3)  
   Type "help" for help.  
   test=#  
      
   源码编译安装成功。