**Linux下Redis的安装和配置**

**一. 安装Redis**

Redis的安装其实相当简单，推荐的方式是下载redis的源码，并在本机编译后安装。

首次进入主文件夹的下载目录下，执行wget下载源码

**[root@localhost ~]$** wget http://download.redis.io/redis-stable.tar.gz

接下来解压之后，移动到/usr/redis目录下

**[root@localhost ~]$** tar -zxvf redis-stable.tar.gz

**[root@localhost ~]$** su mv redis-stable /usr/redis

然后进入redis目录，执行make命令，编译redis源码

**[root@localhost ~]#** cd /usr/redis/

**[root@localhost redis]#** make

编译完成之后，在src目录下有2个重要程序生成，一个是redis-server，另一个是redis-cli；接着进入src目录，执行make install，这时会把这些可执行程序拷贝到/usr/local/bin目录下，由于/usr/local/bin是在系统的环境变量$PATH下定义的，因此终端在任意位置就可以执行redis-server和redis-cli了。

**[root@localhost redis]#** cd src/

**[root@localhost src]#** make install

至此安装redis的工作就完成了。

我们来看看编译出来的几个程序分别是干什么的：

redis-server：顾名思义，redis服务

redis-cli：redis client，提供一个redis客户端，以供连接到redis服务，进行增删改查等操作

redis-sentinel：redis实例的监控管理、通知和实例失效备援服务

redis-benchmark：redis的性能测试工具

redis-check-aof：若以AOF方式产生日志，当意外发生时用来快速修复

redis-check-rdb：若以RDB方式产生日志，当意外发生时用来快速修复

安装完成之后，启动redis-server，并运行redis-cli进行测试

**[root@localhost ~]$** redis-server

**[root@localhost ~]$** redis-cli

**127.0.0.1:6379>** PING

PONG

**127.0.0.1:6379>**

如此说明redis服务已经正常工作，如果redis服务未启动，则运行redis-cli时会报Could not connect to Redis at 127.0.0.1:6379: Connection refused的错误。

**二. 配置自启动**

为了让redis-server能在系统启动时自动运行，需要将redis服务作为守护进程（daemon）来运行，我们回到/usr/redis/目录中找到一个redis.conf的文件，这个文件是redis服务运行时加载的配置，我们先观察一下其中的内容

**[root@localhost redis]$** vi redis.conf

此文件内容非常长，但是大部分是注释，我们重点关注其中的几个设置daemonize和pidfile：

其中daemonize默认值是false，pidfile默认值是pidfile /var/run/redis\_6379.pid

第一个表示是否daemon化，显然我们要把它改成daemonize yes；

第二个表示当服务以守护进程方式运行时，redis默认会把pid写入/var/run/redis\_6379.pid文件，服务运行中该文件就存在，服务一旦停止该文件就自动删除，因而可以用来判断redis是否正在运行。

还有一些比如说日志文件的存放位置等的信息可以自己设置。

保存后退出。

有了基本配置，redis还需要有一个管理启动、关闭、重启的一个脚本。redis源码里其实已经提供了一个初始化脚本，位置在/usr/redis/utils/redis\_init\_script。

**#!/bin/sh**

**REDISPORT=6379**

**EXEC=/usr/local/bin/redis-server**

**CLIEXEC=/usr/local/bin/redis-cli**

**PIDFILE=/var/run/redis.pid**

**CONF="/etc/redis/${REDISPORT}.conf"**

**case "$1" in**

**start)**

**if [ -f $PIDFILE ]**

**then**

**echo "$PIDFILE exists, process is already running or crashed"**

**else**

**echo "Starting Redis server..."**

**$EXEC $CONF**

**fi**

**;;**

**stop)**

**if [ ! -f $PIDFILE ]**

**then**

**echo "$PIDFILE does not exist, process is not running"**

**else**

**PID=$(cat $PIDFILE)**

**echo "Stopping ..."**

**$CLIEXEC -p $REDISPORT shutdown**

**while [ -x /proc/${PID} ]**

**do**

**echo "Waiting for Redis to shutdown ..."**

**sleep 1**

**done**

**echo "Redis stopped"**

**fi**

**;;**

**\*)**

**echo "Please use start or stop as first argument"**

**;;**

**esac**

脚本中指定了端口、server路径、cli路径、pidfile路径以及conf路径，上述标黄的地方都需要正确配置，多说一句，如果在安装时执行了make install，那么这里的脚本不需要做多大改动，因为make install把server和cli都拷到/usr/local/bin下面了。

另外看到这里conf的路径，我们需要把redis目录下的redis.conf文件拷贝到/etc/redis/6379.conf

**[root@localhost utils]#** cd /etc

**[root@localhost etc]#** mkdir redis

**[root@localhost etc]#** cp /usr/redis/redis.conf /etc/redis/6379.conf

接着将redis\_init\_script脚本拷贝到/etc/init.d/redisd

**[root@localhost etc]#** cp /usr/redis/utils/redis\_init\_script /etc/init.d/redisd

在/etc/init.d下的脚本都是可以在系统启动是自动启动的服务，而现在还缺一个系统启动时的配置：

**[root@localhost root]#** chkconfig redisd on

然后就会发现报了一个错误：服务 redisd 不支持 chkconfig ？

这是因为我们需要在redis\_init\_script的开头加一个小改动：

**#!/bin/sh**

**# chkconfig: 2345 90 10**

**# description: Redis is a persistent key-value database**

保存完重新拷贝到/etc/init.d/redisd后，再运行chkconfig就完成了。

一切就绪之后，可以执行以下命令检验service是否设置成功：

**[root@localhost root]**# service redisd start

**[root@localhost root]**# service redisd stop

等价于

**[root@localhost root]**# /etc/init.d/redisd start

**[root@localhost root]**# /etc/init.d/redisd stop

这里需要注意的是，启动和停止Redis用的不是同一个文件，启动时用的是Redis-server，停止时用的是redis-cli，所以，如果你的Redis设置了密码登录，则service redisd stop时会提示需要权限验证，这是为了方便，可以将密码设置在/etc/init.d/redisd文件中，或者修改该文件，读取输入的密码参数。