Redis是一个开源的使用ANSI C语言编写、支持网络、可基于内存亦可持久化的日志型、Key-Value数据库，  
并提供多种语言的API。它支持丰富的数据类型，和高速的内存读写。正在逐步取代memcached的地位。

下面就演示下在centos6.9的32位下的安装配置过程。

**1、**[**安装**](https://www.cnblogs.com/zerotomax/p/7468833.html#install)

**2、**[**配置**](https://www.cnblogs.com/zerotomax/p/7468833.html#conf)

**3、**[**卸载**](https://www.cnblogs.com/zerotomax/p/7468833.html#uninstall)

**1、****安装**

　　①我们可以通过在官网下载tar.gz的安装包，或者通过wget的方式下载

[复制代码](javascript:void(0);)

**[**root@localhost data**]**# wget http://download.redis.io/releases/redis-4.0.1.tar.gz

--2017-09-03 09:51:27-- http://download.redis.io/releases/redis-4.0.1.tar.gz

正在解析主机 download.redis.io... 109.74.203.151

正在连接 download.redis.io|109.74.203.151|:80... 已连接。

已发出 HTTP 请求，正在等待回应... 200 OK

长度：1711660 (1.6M) **[**application/x-gzip**]**

正在保存至: “redis-4.0.1.tar.gz”

57% **[**=====================================> **]** 981,091 298K/s eta(英国中部时  
64% **[**==========================================> **]** 1,105,869 309K/s eta(英国中部时  
72% **[**===============================================> **]** 1,237,657 321K/s eta(英国中部时  
80% **[**====================================================> **]** 1,376,455 334K/s eta(英国中部时  
86% **[**========================================================> **]** 1,478,801 341K/s eta(英国中部时  
92% **[**============================================================> **]** 1,582,549 349K/s eta(英国中部时  
100%**[**==================================================================>**]** 1,711,660 380K/s in 4.6s

2017-09-03 09:51:32 (363 KB/s) - 已保存 “redis-4.0.1.tar.gz” **[**1711660/1711660**]**)

**[**root@localhost data**]**#

[复制代码](javascript:void(0);)

　　②解压

https://images.cnblogs.com/OutliningIndicators/ContractedBlock.gif View Code

　　③移动到/local/usr/redis目录下去，这个纯属个人爱好

[复制代码](javascript:void(0);)

**[**root@localhost data**]**# ll

总用量 1676

drwxrwxr-x. 6 root root 4096 7月 24 22:58 redis-4.0.1

-rw-r--r--. 1 root root 1711660 7月 24 22:59 redis-4.0.1.tar.gz

**[**root@localhost data**]**# mv redis-4.0.1 ../redis

**[**root@localhost data**]**# cd ..

**[**root@localhost local**]**# ll

总用量 44

drwxr-xr-x. 2 root root 4096 9月 3 09:43 bin

drwxr-xr-x. 2 root root 4096 9月 3 10:04 data

drwxr-xr-x. 2 root root 4096 9月 23 2011 etc

drwxr-xr-x. 2 root root 4096 9月 23 2011 games

drwxr-xr-x. 2 root root 4096 9月 23 2011 include

drwxr-xr-x. 2 root root 4096 9月 23 2011 lib

drwxr-xr-x. 2 root root 4096 9月 23 2011 libexec

drwxrwxr-x. 6 root root 4096 7月 24 22:58 redis

drwxr-xr-x. 2 root root 4096 9月 23 2011 sbin

drwxr-xr-x. 5 root root 4096 4月 1 04:48 share

drwxr-xr-x. 2 root root 4096 9月 23 2011 src

**[**root@localhost local**]**# cd redis

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

总用量 276

-rw-rw-r--. 1 root root 127778 7月 24 22:58 00-RELEASENOTES

-rw-rw-r--. 1 root root 53 7月 24 22:58 BUGS

-rw-rw-r--. 1 root root 1815 7月 24 22:58 CONTRIBUTING

-rw-rw-r--. 1 root root 1487 7月 24 22:58 COPYING

drwxrwxr-x. 6 root root 4096 7月 24 22:58 deps

-rw-rw-r--. 1 root root 11 7月 24 22:58 INSTALL

-rw-rw-r--. 1 root root 151 7月 24 22:58 Makefile

-rw-rw-r--. 1 root root 4223 7月 24 22:58 MANIFESTO

-rw-rw-r--. 1 root root 20530 7月 24 22:58 README.md

-rw-rw-r--. 1 root root 57764 7月 24 22:58 redis.conf

-rwxrwxr-x. 1 root root 271 7月 24 22:58 runtest

-rwxrwxr-x. 1 root root 280 7月 24 22:58 runtest-cluster

-rwxrwxr-x. 1 root root 281 7月 24 22:58 runtest-sentinel

-rw-rw-r--. 1 root root 7606 7月 24 22:58 sentinel.conf

drwxrwxr-x. 3 root root 4096 7月 24 22:58 src

drwxrwxr-x. 10 root root 4096 7月 24 22:58 tests

drwxrwxr-x. 8 root root 4096 7月 24 22:58 utils

**[**root@localhost redis**]**# cd ..

**[**root@localhost local**]**#

[复制代码](javascript:void(0);)

　　④接下来就是编译了，通过make命令，如果编译的时候报gcc命令找不到的话，可以通过下面的命令安装gcc命令，gcc是c的编译命令

yum install gcc-c++

　　下面就通过make来编译，make是自动编译，会根据Makefile中描述的内容来进行编译。

[复制代码](javascript:void(0);)

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

cd src && make all

make**[**1**]**: Entering directory `/usr/local/redis/src'

CC Makefile.dep

.

.

.

.

Hint: It's a good idea to run 'make test' ;)

make**[**1**]**: Leaving directory `/usr/local/redis/src'

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

[复制代码](javascript:void(0);)

　　⑤可以看到在src目录下生成了几个新的文件。

[复制代码](javascript:void(0);)

**[**root@localhost redis**]**# ll -tr src

总用量 44440

-rw-rw-r--. 1 root root 3779 7月 24 22:58 zmalloc.h

.

.

.

-rw-r--r--. 1 root root 56148 9月 3 10:11 rax.o

-rwxr-xr-x. 1 root root 7185836 9月 3 10:11 redis-server

-rwxr-xr-x. 1 root root 7185836 9月 3 10:11 redis-sentinel

-rw-r--r--. 1 root root 143800 9月 3 10:11 redis-cli.o

-rwxr-xr-x. 1 root root 5092431 9月 3 10:11 redis-cli

-rw-r--r--. 1 root root 44892 9月 3 10:11 redis-benchmark.o

-rwxr-xr-x. 1 root root 4985275 9月 3 10:11 redis-benchmark

-rwxr-xr-x. 1 root root 7185836 9月 3 10:11 redis-check-rdb

-rwxr-xr-x. 1 root root 7185836 9月 3 10:11 redis-check-aof

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

[复制代码](javascript:void(0);)

　　为了使用方便，我们需要将这个几个文件加到/usr/local/bin目录下去。这个目录在Path下面的话，就可以直接执行这几个命令了。

[复制代码](javascript:void(0);)

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

cd src && make install

make**[**1**]**: Entering directory `/usr/local/redis/src'

CC Makefile.dep

make**[**1**]**: Leaving directory `/usr/local/redis/src'

make**[**1**]**: Entering directory `/usr/local/redis/src'

Hint: It's a good idea to run 'make test' ;)

INSTALL install

INSTALL install

INSTALL install

INSTALL install

INSTALL install

make**[**1**]**: Leaving directory `/usr/local/redis/src'

**[**root@localhost redis**]**# cd ..

**[**root@localhost local**]**# ll

总用量 44

drwxr-xr-x. 2 root root 4096 9月 3 10:16 bin

drwxr-xr-x. 2 root root 4096 9月 3 10:04 data

drwxr-xr-x. 2 root root 4096 9月 23 2011 etc

drwxr-xr-x. 2 root root 4096 9月 23 2011 games

drwxr-xr-x. 2 root root 4096 9月 23 2011 include

drwxr-xr-x. 2 root root 4096 9月 23 2011 lib

drwxr-xr-x. 2 root root 4096 9月 23 2011 libexec

drwxrwxr-x. 6 root root 4096 7月 24 22:58 redis

drwxr-xr-x. 2 root root 4096 9月 23 2011 sbin

drwxr-xr-x. 5 root root 4096 4月 1 04:48 share

drwxr-xr-x. 2 root root 4096 9月 23 2011 src

**[**root@localhost local**]**# ll bin

总用量 30908

-rwxr-xr-x. 1 root root 4985275 9月 3 10:16 redis-benchmark

-rwxr-xr-x. 1 root root 7185836 9月 3 10:16 redis-check-aof

-rwxr-xr-x. 1 root root 7185836 9月 3 10:16 redis-check-rdb

-rwxr-xr-x. 1 root root 5092431 9月 3 10:16 redis-cli

lrwxrwxrwx. 1 root root 12 9月 3 10:16 redis-sentinel -> redis-server

-rwxr-xr-x. 1 root root 7185836 9月 3 10:16 redis-server

**[**root@localhost local**]**#

[复制代码](javascript:void(0);)

　　可以看到，这几个文件就已经被加载到bin目录下了

　　⑥下面启动服务器，来看看安装是否成功。使用redis-server命令。

[复制代码](javascript:void(0);)

**[**root@localhost local**]**# redis-server

9190:C 03 Sep 10:19:09.291 # oO0OoO0OoO0Oo Redis is starting oO0OoO0OoO0Oo

9190:C 03 Sep 10:19:09.292 # Redis version=4.0.1, bits=32, commit=00000000, modified=0, pid=9190, just started

9190:C 03 Sep 10:19:09.292 # Warning: no config file specified, using the default config. In order to specify a config file use redis-server /path/to/redis.conf

9190:M 03 Sep 10:19:09.295 \* Increased maximum number of open files to 10032 (it was originally set to 1024).

9190:M 03 Sep 10:19:09.312 # Warning: 32 bit instance detected but no memory limit set. Setting 3 GB maxmemory limit with 'noeviction' policy now.

\_.\_

\_.-``\_\_ ''-.\_

\_.-`` `. `\_. ''-.\_ Redis 4.0.1 (00000000/0) 32 bit

.-`` .-```. ```\/ \_.,\_ ''-.\_

( ' , .-` | `, ) Running in standalone mode

|`-.\_`-...-` \_\_...-.``-.\_|'` \_.-'| Port: 6379

| `-.\_ `.\_ / \_.-' | PID: 9190

`-.\_ `-.\_ `-./ \_.-' \_.-'

|`-.\_`-.\_ `-.\_\_.-' \_.-'\_.-'|

| `-.\_`-.\_ \_.-'\_.-' | http://redis.io

`-.\_ `-.\_`-.\_\_.-'\_.-' \_.-'

|`-.\_`-.\_ `-.\_\_.-' \_.-'\_.-'|

| `-.\_`-.\_ \_.-'\_.-' |

`-.\_ `-.\_`-.\_\_.-'\_.-' \_.-'

`-.\_ `-.\_\_.-' \_.-'

`-.\_ \_.-'

`-.\_\_.-'

9190:M 03 Sep 10:19:09.316 # WARNING: The TCP backlog setting of 511 cannot be enforced because /proc/sys/net/core/somaxconn is set to the lower value of 128.

9190:M 03 Sep 10:19:09.316 # Server initialized

9190:M 03 Sep 10:19:09.318 # WARNING overcommit\_memory is set to 0! Background save may fail under low memory condition. To fix this issue add 'vm.overcommit\_memory = 1' to /etc/sysctl.conf and then reboot or run the command 'sysctl vm.overcommit\_memory=1' for this to take effect.

9190:M 03 Sep 10:19:09.318 \* Ready to accept connections

[复制代码](javascript:void(0);)

　　看到这个界面的话，就表示安装成功了。

　　下面通过在启动一个linux客户端，通过server-cli shutdown来关闭服务器。

[复制代码](javascript:void(0);)

**[**root@localhost ~**]**# redis-cli shutdown

**[**root@localhost ~**]**#

下面是刚刚服务器启动的客户端的log

9190:M 03 Sep 10:20:45.566 # User requested shutdown...

9190:M 03 Sep 10:20:45.566 \* Saving the final RDB snapshot before exiting.

9190:M 03 Sep 10:20:45.631 \* DB saved on disk

9190:M 03 Sep 10:20:45.631 # Redis is now ready to exit, bye bye...

**[**root@localhost local**]**#

[复制代码](javascript:void(0);)

**2、****配置**

　　可以看到，前面在启动redis服务器后，都是在前台启动的，需要重新启动一个客户端来进行登陆操作。这样非常不方便，

所以，我们需要设置后台启动。

　　在redis.conf的配置文件里面。做如下的修改：

daemonize no

修改为：

daemonize yes

　　下面需要设置redis服务器开机自动启动：

　　要先让redis服务自动启动的话，首先需要在/etc/init.d目录下创建redis的启动脚本。

　　将redis安装目录下的utils/redis\_init\_script复制到/etc/init.d目录下，命名为redis（名字简单，使用方便）

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

**[**root@localhost utils**]**# ll /etc/init.d/redis

-rwxr-xr-x. 1 root root 1098 9月 3 13:02 /etc/init.d/redis

**[**root@localhost utils**]**#

　　继续编辑启动文件，修改其中指定的pid和配置文件。

PIDFILE=/var/run/redis\_${REDISPORT}.pid

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

修改为

PIDFILE=/var/redis/run/redis\_${REDISPORT}.pid

CONF="/etc/redis/redis.conf"

　　首先创建存放pid的目录为/var/redis/run

**[**root@localhost var**]**# pwd

/var

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

**[**root@localhost var**]**# cd redis

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

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

　　接下来，我们需要把配置文件复制一份到/etc/redis这个目录下面去

[复制代码](javascript:void(0);)

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

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

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

总用量 0

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

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

总用量 60

-rw-r--r--. 1 root root 57765 9月 3 13:12 redis.conf

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

/etc/redis

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

[复制代码](javascript:void(0);)

　　修改配置文件中的pid为前面配置的pid。

pidfile /var/run/redis\_6379.pid

修改为

pidfile /var/redis/run/redis\_6379.pid

　　至此为止，我们已经可以通过service redis start/stop来启动和关闭redis服务了。

　　最后只需要通过chkconfig redis on命令来设置开机启动即可。

　　如果提示redis 服务不支持 chkconfig的话，只需要在/etc/init.d/redis这个启动脚本的第二行后面加上下面的内容即可。

# chkconfig:2345 90 10

#

# description:Redis is a persistent key-value database

**[**root@localhost ~**]**# chkconfig redis on

redis 服务不支持 chkconfig

**[**root@localhost ~**]**# vi /etc/init.d/redis

**[**root@localhost ~**]**# chkconfig redis on

**[**root@localhost ~**]**#

**3、****卸载**

　卸载redis非常的简单，只需要简单的三步

1、停止redis服务器

　　首先，通过下面的命令查看redis服务是否在运行

**[**root@localhost ~**]**# ps aux|grep redis

root 2553 0.2 0.1 41964 1916 ? Ssl 09:38 0:00 redis-server 127.0.0.1:6379

root 2565 0.0 0.0 6048 780 pts/0 S+ 09:39 0:00 grep redis

**[**root@localhost ~**]**#

　　可以看到，在6379端口，有redis-server的监听

　　通过下面的命令停止redis服务器。

**[**root@localhost ~**]**# redis-cli shutdown

**[**root@localhost ~**]**# ps aux|grep redis

root 2575 0.0 0.0 6048 780 pts/0 S+ 09:41 0:00 grep redis

**[**root@localhost ~**]**#

　　可以看到，已经停止了redis服务了。

　　需要注意的是，由于我的redis命令都安装到/usr/local/bin目录下面了，并且添加到环境变量PATH里面了，所以可以直接运行。

2、删除make的时候生成的几个redisXXX的文件

[复制代码](javascript:void(0);)

**[**root@localhost local**]**# ll /usr/local/bin

总用量 30908

-rwxr-xr-x. 1 root root 4985307 9月 2 21:13 redis-benchmark

-rwxr-xr-x. 1 root root 7185872 9月 2 21:13 redis-check-aof

-rwxr-xr-x. 1 root root 7185872 9月 2 21:13 redis-check-rdb

-rwxr-xr-x. 1 root root 5092475 9月 2 21:13 redis-cli

lrwxrwxrwx. 1 root root 12 9月 2 21:13 redis-sentinel -> redis-server

-rwxr-xr-x. 1 root root 7185872 9月 2 21:13 redis-server

**[**root@localhost local**]**# rm -f /usr/local/bin/redis\*

**[**root@localhost local**]**# ll /usr/local/bin

总用量 0

**[**root@localhost local**]**#

[复制代码](javascript:void(0);)

3、顺便也删除掉解压后的文件目录和所以文件

[复制代码](javascript:void(0);)

**[**root@localhost local**]**# ll

总用量 40

drwxr-xr-x. 2 root root 4096 9月 3 09:43 bin

drwxr-xr-x. 2 root root 4096 9月 23 2011 etc

drwxr-xr-x. 2 root root 4096 9月 23 2011 games

drwxr-xr-x. 2 root root 4096 9月 23 2011 include

drwxr-xr-x. 2 root root 4096 9月 23 2011 lib

drwxr-xr-x. 2 root root 4096 9月 23 2011 libexec

drwxrwxr-x. 6 root root 4096 9月 2 21:11 redis

drwxr-xr-x. 2 root root 4096 9月 23 2011 sbin

drwxr-xr-x. 5 root root 4096 4月 1 04:48 share

drwxr-xr-x. 2 root root 4096 9月 23 2011 src

**[**root@localhost local**]**# rm -rf redis

**[**root@localhost local**]**# ll

总用量 36

drwxr-xr-x. 2 root root 4096 9月 3 09:43 bin

drwxr-xr-x. 2 root root 4096 9月 23 2011 etc

drwxr-xr-x. 2 root root 4096 9月 23 2011 games

drwxr-xr-x. 2 root root 4096 9月 23 2011 include

drwxr-xr-x. 2 root root 4096 9月 23 2011 lib

drwxr-xr-x. 2 root root 4096 9月 23 2011 libexec

drwxr-xr-x. 2 root root 4096 9月 23 2011 sbin

drwxr-xr-x. 5 root root 4096 4月 1 04:48 share

drwxr-xr-x. 2 root root 4096 9月 23 2011 src

**[**root@localhost local**]**#

[复制代码](javascript:void(0);)

这样，redis就卸载完成了。