**Redis**

redis是目前公认的速度最快的基于内存的键值对数据库，但redis的缺点也非常明显，仅提供最基本的hash set, list, sorted set等基于数据类型，不分表，没有schema，没有索引，没有外键，缺少int/date等基本数据类型，多条件查询需要通过集合内联(sinter,zinterstore)和连接间接实现，操作不便，开发效率低，可维护性不佳； 因此一般不将其视为完整的数据库单独使用，很多网站将redis作为高速缓存和session状态存储层，然后再与其他数据库搭配使用。

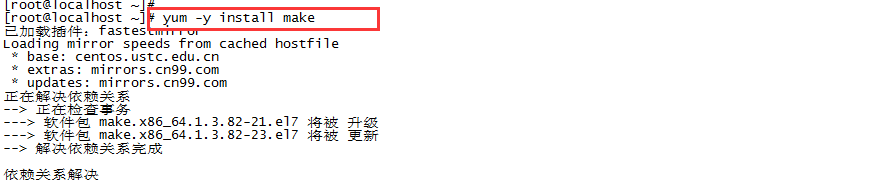
**安装步骤：**

**1，安装Make ，GCC进行编译**

yum install gcc

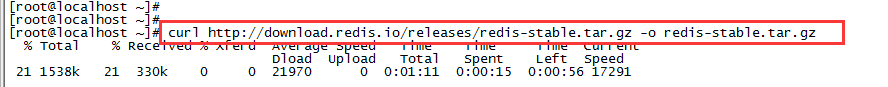
yum install make





**2，下载redis最新版本**

curl http://download.redis.io/releases/redis-stable.tar.gz -o redis-stable.tar.gz



**3，解压**

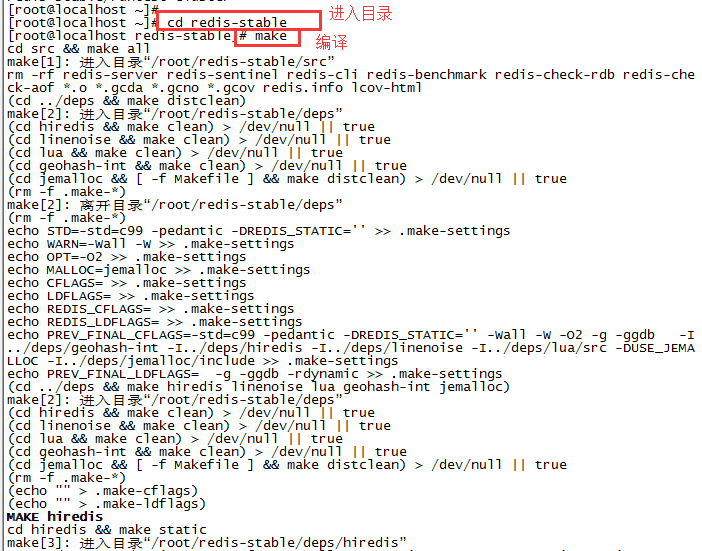
tar zxvf redis-stable.tar.gz



**4，进入redis目录，执行make编译**

cd redis-stable

make





**5，安装**

进入源文件的目录。

cd src

make install

{命令make install相当于以下操作：

复制 Redis 的服务器和客户端到 /usr/local/bin。

cp redis-server redis-cli /usr/local/bin

最好也把 sentinel，benchmark 和 check 复制过去。

cp redis-sentinel redis-benchmark redis-check-aof redis-check-rdb /usr/local/bin

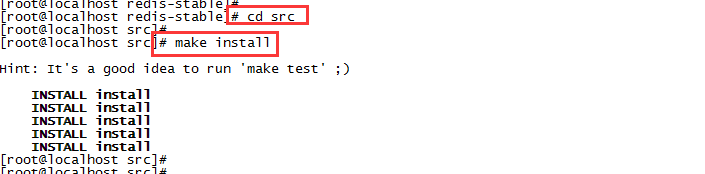
}

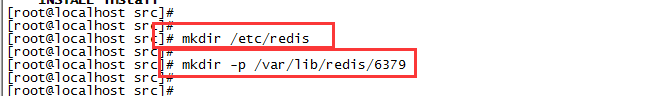
创建redis 配置文件夹。

mkdir /etc/redis

在/var/lib/redis 下创建有效的保存数据的目录

mkdir -p /var/lib/redis/6379





**6，系统参数配置**

为了让 redis 正常工作需要配置一些内核参数。

配置 vm.overcommit\_memory 为1，这可以避免数据被截断，详情见此。

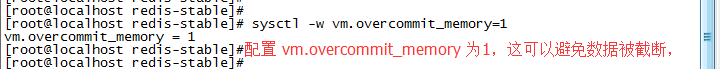
sysctl -w vm.overcommit\_memory=1

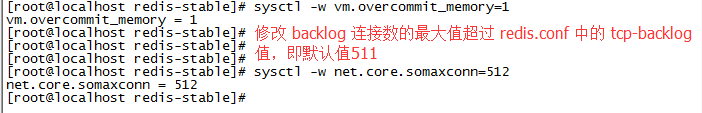
修改 backlog 连接数的最大值超过 redis.conf 中的 tcp-backlog 值，即默认值511。你可以在kernel.org 找到更多有关基于 sysctl 的 ip 网 络隧道的信息。

sysctl -w net.core.somaxconn=512

取消对透明巨页内存（transparent huge pages）的支持，因为这会造成 redis 使用过程产生延时和内存访问问题。

echo never >/sys/kernel/mm/transparent\_hugepage/enabled





**7，修改redis配置文件redis.conf**

修改位置参考http://blog.csdn.net/zh521zh/article/details/52101926

redis.conf 是 redis 的配置文件，然而你会看到这个文件的名字是 6379.conf ，而这个数字就是 redis 监听的网络端口。如果你想要运行超过一个的 redis 实例，推荐用这样的名字。

复制示例的 redis.conf 到 /etc/redis/6379.conf。

cp redis.conf /etc/redis/6379.conf （需要在redis-stable目录下面执行，请看运行截图）

现在编辑这个文件并且配置参数。

vi /etc/redis/6379.conf

**daemonize**

设置 daemonize 为 yes。

daemonize yes

**pidfile**

设置 pidfile 为 /var/run/redis\_6379.pid。

pidfile /var/run/redis\_6379.pid

**port**

如果不准备用默认端口，可以修改。

port 6379

**loglevel**

设置日志级别。

loglevel notice

**logfile**

修改日志文件路径。

logfile /var/log/redis\_6379.log

**dir**

设置目录为 /var/lib/redis/6379

dir /var/lib/redis/6379

**快照**

默认情况下，redis 会周期性的将数据集转储到我们设置的目录下的 dump.rdb 文件。你可以使用 save 命令配置转储的频率，它的第一个参数是以秒为单位的时间帧，第二个参数是在数据文件上进行修改的数量。

每隔15分钟并且最少修改过一次键。

save 900 1

每隔5分钟并且最少修改过10次键。

save 300 10

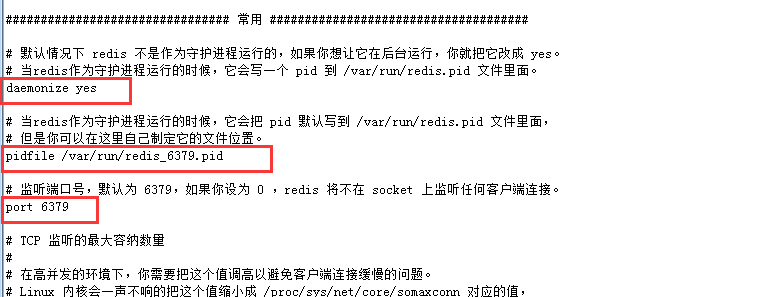
每隔1分钟并且最少修改过10000次键。

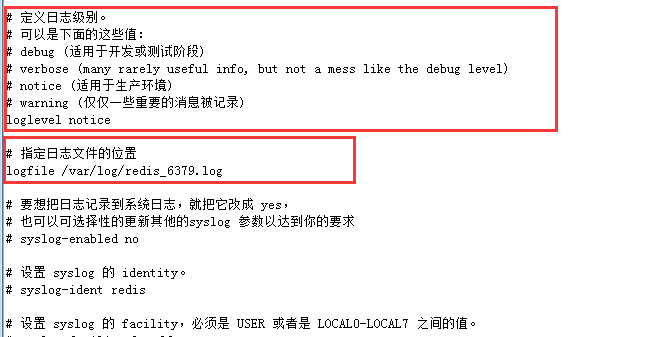
save 60 10000

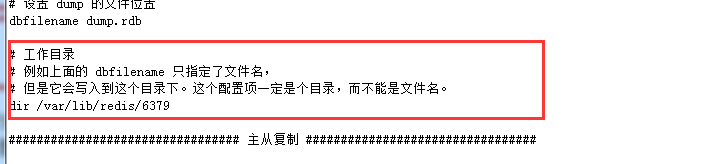
文件 /var/lib/redis/6379/dump.rdb 包含了从上次保存以来内存里数据集的转储数据。因为它先创建临时文件然后替换之前的转储文件，这里不存在数据破坏的问题，你不用担心，可以直接复制这个文件。

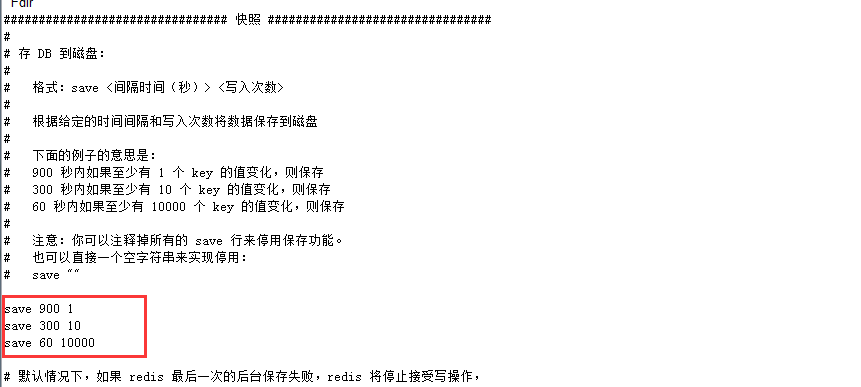
运行截图：











**8，创建服务**

你可以使用 systemd 将 redis 添加到系统开机启动列表。

进入redis-stable目录复制示例的 init\_script 文件到 /etc/init.d，注意脚本名所代表的端口号。

cp utils/redis\_init\_script /etc/init.d/redis\_6379

现在我们要使用 systemd，所以在 /etc/systemd/system 下创建一个单位文件名字为 redis\_6379.service。

vi /etc/systemd/system/redis\_6379.service

填写下面的内容，详情可见 systemd.service。

[Unit]

Description=Redis on port 6379

[Service]

Type=forking

ExecStart=/etc/init.d/redis\_6379 start

ExecStop=/etc/init.d/redis\_6379 stop

[Install]

WantedBy=multi-user.target

(解释：

Description:描述服务

After:描述服务类别

[Service]服务运行参数的设置

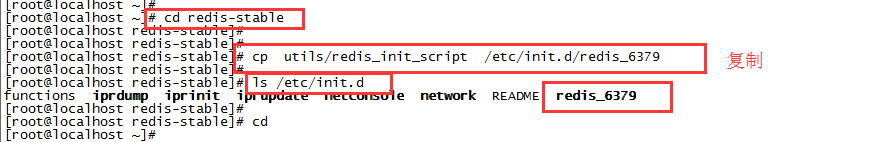
Type=forking是后台运行的形式

ExecStart为服务的具体运行命令

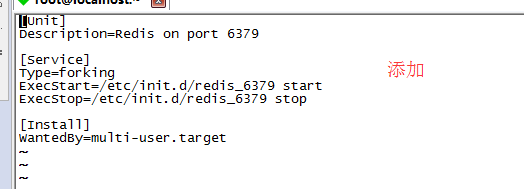
ExecStop为停止命令

[Install]服务安装的相关设置，可设置为多用户)

运行截图：







**9，Redis的查看，开启和关闭。**

查看： ps -ef|grep redis

开启： systemctl start redis\_6379 或 /etc/init.d/redis\_6379 start

关闭： systemctl stop redis\_6379 或 /etc/init.d/redis\_6379 stop



**10，Redis的开机自启动**

刚刚配置的服务需要让systemctl能识别，就必须刷新配置

systemctl daemon-reload

如果没有权限可以使用sudo

sudo systemctl daemon-reload

redis服务加入开机启动

systemctl enable redis\_6379

//禁止开机启动

//systemctl disable redis\_6379

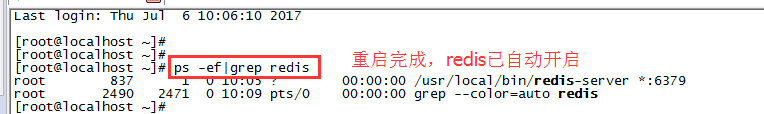
重启电脑，查看redis是否自动开启

reboot

ps -ef|grep redis







**11，运行测试**

cd redis-stable

./src/redis-cli

set too bar

get too

