**Redis安装部署**

下载Redis 源码包，解压

Shell > yum –y install gcc-c++

Shell > cd redis-4.0.9

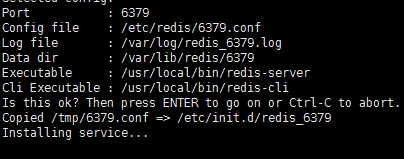
Shell > cd src

Shell > make

Shell > make install

Shell > cd ..

Shell > ./utils/install\_server.sh (一路回车，按默认配置)



Shell > vim /etc/redis/6379.conf (主配置文件)

Shell > cd /usr/local/reids/redis-4.0.9/src

启动Redis有三种方法

1. 默认配置
2. 运行启动
3. 配置文件启动

Shell > ./redis-server /etc/redis/6379.conf (以配置文件启动Redis服务)

Shell > ps -ef|grep redis (查看Redis是否启动)

Shell > redis-server --version （查看Redis版本信息）

Shell > Redis-cli –v

Redis命令行客户端

第一种是交互式方式：redis-cli -h {host} -p {port}

第二种是命令方式： redis-cli -h {host} -p {port} {command}

停止Redis服务

redis-cli shutdown

这里有三点需要注意一下：

1. Redis关闭的过程：断开与客户端的连接、 持久化文件生成， 是一种相对优雅的关闭方式。
2. 除了可以通过 shutdown 命令关闭 Redis 服务以外， 还可以通过 kill 进程号的方 式关闭掉 Redis, 但是不要粗暴地使用 K过1 -9 强制杀死 Redis 服务， 不但不会做持久化操作， 还会造成缓冲区等资源不能被优雅关闭， 极端情况会造成 AOF 和复制丢失数据的情况。
3. shutdown 还有一个参数， 代表是否在关闭 Redis 前， 生成持久化文件： redis-cli shutdown nosave |save

Redis4.0版本介绍：

Redis 4.0

可能出乎很多人的意料， Redis 3.2 之后的版本是4.0, 而不是3.4 、3.6、3.8。一般这种重大版本号的升级也意味着软件或者工具本身发 生了重大变革

下面列出Redis 4.0 的新特性：

1. 提供了模块系统， 方便第三方开发者拓展Redis的功能， 更多模块详见： <http://redismodules.com>。
2. PSYNC 2.0: 优化了之前版本中， 主从节点切换必然引起全量复制的问题。
3. 提供了新的缓存剔除算法： LFU(Last Frequently Used), 并对已有算法进行 了优化。
4. 提供了非阻塞 del 和flushall/flushdb 功能， 有效解决删除bigkey可能造成的Redis阻塞。
5. 提供了RDB-AOF混合待久化格式， 充分利用了AOF 和RDB各自优势。
6. 提供memory命令， 实现对内存更为全面的监控统计。
7. 提供了交互数据库功能， 实现Redis内部数据库之间的数据置换。
8. Redis Cluster兼容NAT和Docker。

Redis的几点总结：

1. Redis的8个特性：速度快、基于键值对的数据结构服务器、功能丰富、简单稳定、 客户端语言多、持久化、主从复制、支持高可用和分布式
2. Redis并不是万金油， 有些场景不适合使用Redis进行开发
3. 开发运维结合以及阅读源码是用好Redis的重要方法。
4. 生产环境中使用配置文件启动Redis 。
5. 生产环境选取稳定版本的Redis。
6. Redis 3.0是重要的里程碑， 发布了Redis官方的分布式实现Redis Cluster。

全局命令：

Keys \* 将所有的键输出，keys命令会遍历所有的键，当 Redis 保存了大量键时线上环境禁止使用

dbsize dbsize 命令会返回当前数据库中键的总数。不会遍历所有键， 而是直接获取 Redis 内置的键总数变量

检查键是否存在

exists key

如果键存在则返回1' 不存在则返回0:

127.0.0.1:6379> exists java

(integer) 1

127.0.0.1:6379> exists not\_exist\_key

(integer) 0

删除键

del key [key ...

del 是一个通用命令，无论值是什么数据结构类型， del 命令都可以将其删除

127.0.0.1:6379> del java

(integer} 1

127.0.0.1:6379> exists java

(integer} 0