# 分布式缓存服务

# 故障排除

**文档版本** 01

**发布日期** 2024-03-18

**华为云计算技术有限公司**

**商标声明**


和其他华为商标均为华为技术有限公司的商标。
本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

**注意**

您购买的产品、服务或特性等应受华为云计算技术有限公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为云计算技术有限公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。


# 华为云计算技术有限公司

地址： 贵州省贵安新区黔中大道交兴功路华为云数据中心 邮编：550029

网址： https://www.huaweicloud.com/

# 目 录



# 1 Redis 连接失败问题排查和解决

## 概述

本章节主要描述Redis连接过程出现的问题，以及解决方法。

## 问题分类

当您发现与Redis实例连接出现异常时，可以根据本文的内容，从以下几个方面进行排查。

- **Redis和ECS之间的连接问题**
- **公网连接Redis (仅Redis 3.0涉及)**
- **密码问题**
- **实例配置问题**
- **客户端连接问题**
- **带宽超限导致连接问题**
- **性能问题导致连接超时**

## Redis 和 ECS 之间的连接问题

客户端所在的ECS必须和Redis实例在同一个VPC内，并且需要确保ECS和Redis之间可以正常连接。

- 如果是Redis 3.0或企业版实例，Redis和ECS的安全组没有配置正确，连接失败。

  解决方法：配置ECS和Redis实例所在安全组规则，允许Redis实例被访问。具体配置，可以参考**配置安全组**。
- 如果是Redis 4.0/5.0/6.0基础版实例，开启了白名单功能，连接失败。

  如果实例开启了白名单，在使用客户端连接时，需要**确保客户端IP是否在白名单内**，如果不在白名单，会出现连接失败。具体配置操作，可以参考**配置白名单**。客户端IP如果有变化，需要将变化后的IP加入白名单。
- Redis实例和ECS不在同一个Region。

  解决方法：不支持跨Region访问，可以在ECS所在的Region创建Redis实例，创建时注意选择与ECS相同VPC，创建之后，使用**数据迁移**进行迁移，将原有Redis实例数据迁移到新实例中。

- Redis实例和ECS不在同一个VPC。

  不同的VPC，网络是不相通的，不在同一VPC下的ECS是无法访问Redis实例。可以通过创建VPC对等连接，将两个VPC的网络打通，实现跨VPC访问Redis实例。

  关于创建和使用VPC对等连接，请参考**VPC对等连接说明**。

## 公网连接 Redis (仅 Redis 3.0 涉及)

在进行公网访问时，请先仔细阅读**公网连接**章节，**检查实例是否满足公网访问的要求**。

- 连接时提示：Error: Connection reset by peer或者出现：远程主机强迫关闭一个现有的连接。
  - 原因1：安全组没有配置正确。

    解决方法：需要允许Redis实例被访问，具体配置操作和公网连接操作，请按照**公网连接**章节中的操作进行。
  - 原因2：查看Redis所在vpc子网是否被ACL关联，同时这个ACL出方向被限制了。若是，放开限制。
  - 原因3：开启了SSL加密传输，连接时没有安装配置Stunnel，直接使用了界面提示的IP地址进行连接。

    解决方法：开启SSL加密时，必须安装配置Stunnel客户端，具体操作，请按照**公网连接Redis实例（开启SSL加密）**执行。其中，请注意，**在连接Redis实例命令中，IP地址需要配置为Stunnel客户端地址和端口，不要使用控制台展示的Redis实例公网连接地址和端口**。
- 已经开启了公网访问的Redis实例，公网访问被关闭了，无法使用公网访问。

  原因：该Redis实例绑定的弹性公网IP被解绑，导致Redis实例公网被关闭。

  解决方法：在控制台重新开启实例的公网访问，绑定弹性公网IP，并重新连接。

## 密码问题

密码输入错误时，端口可以连接上，但鉴权认证会失败。如果忘记了密码，可以**重置密码**。

## 实例配置问题

连接Redis时存在拒绝连接，可登录分布式缓存服务控制台，进入实例详情页面，调整实例参数maxclients的配置，具体操作可参考**修改配置参数**。

## 客户端连接问题

- 在使用Redis-cli连接Cluster集群时，连接失败。

  解决方法：请检查连接命令是否加上**-c**，在连接Cluster集群节点时务必使用正确连接命令。
  - Cluster集群连接命令：

    ./redis-cli -h *{dcs_instance_address}* -p 6379 -a {password} -c
  - 单机、主备、Proxy集群连接命令：

    ./redis-cli -h *{dcs_instance_address}* -p 6379 -a {password}

  具体连接操作，请参考**Redis-cli连接**。
- 出现Read timed out或Could not get a resource from the pool。

解决方法：

- 排查是否使用了keys命令，keys命令会消耗大量资源，造成Redis阻塞。建议使用scan命令替代，且避免频繁执行。
- 排查实例是否是Redis 3.0，Redis 3.0底层用的是sata盘，当Redis数据持久化即AOF时，会触发偶现的磁盘性能问题，导致连接异常，可更换Redis实例为4.0及以上版本，其底层是ssd盘，磁盘性能更高，或若不需要持久化可关闭AOF。

- 出现unexpected end of stream错误，导致业务异常。

  解决方法：

  - Jedis连接池调优，建议参考**Jedis参数配置建议**进行配置连接池参数。
  - 排查是否大key较多，建议根据**优化大key**排查优化。

- 连接断开。

  解决方法：

  - 调整应用超时时间。
  - 优化业务，避免出现慢查询。
  - 建议使用scan命令替代keys命令。

- Jedis连接池问题，请参考**使用Jedis连接池报错如何处理？** 。

## 带宽超限导致连接问题

当实例已使用带宽达到实例规格最大带宽，可能会导致部分Redis连接超时现象。

您可以查看监控指标“流控次数”，统计周期内被流控的次数，确认带宽是否已经达到上限。

然后，检查实例是否有大Key和热Key，如果存在大Key或者单个Key负载过大，容易造成对于单个Key的操作占用带宽资源过高。大Key和热Key操作，请参考**分析实例大Key和热Key**。

Redis 4.0及之后版本的实例，支持通过控制台对Redis实例进行带宽的**临时扩容**（7天内有效），可用于临时解决业务流量高峰，带宽超限的问题。

## 性能问题导致连接超时

使用了keys等消耗资源的命令，导致CPU使用率超高；或者实例没有设置过期时间、没有清除已过期的Key，导致存储的数据过多，一直在内存中，内存使用率过高等，这些都容易出现访问缓慢、连接不上等情况。

- 建议客户改成scan命令或者禁用keys命令。
- 查看监控指标，并配置对应的告警。监控项和配置告警步骤，可查看**必须配置的监控告警**。

  例如，可以通过监控指标“内存利用率”和“已用内存”查看实例内存使用情况、“活跃的客户端数量”查看实例连接数是否达到上限等。

- 检查实例是否存在大Key和热Key。

  DCS控制台提供了大Key和热Key的分析功能，具体使用，请参考**分析Redis实例的大Key和热Key**。

# 2 Redis 实例 CPU 使用率高问题排查和解决

## 问题现象

Redis实例CPU使用率短时间内冲高。CPU过高可能会导致连接超时，影响业务。CPU过高也可能触发主备倒换。

## 可能原因

1. 客户的业务负载过重，QPS过高，导致CPU被用满，排查方法请参考**排查QPS是否过高**。
2. 使用了keys等消耗资源的命令，排查及处理措施请参考**查找并禁用高消耗命令**。
3. 发生Redis的持久化重写操作，排查及处理措施请参考**是否存在Redis的持久化重写操作**。

## 排查 QPS 是否过高

在分布式缓存服务控制台的缓存管理页面，单击实例进入实例详情界面，单击左侧的性能监控，进入性能监控页面，查询实例级别的每秒并发操作数（ QPS ）。

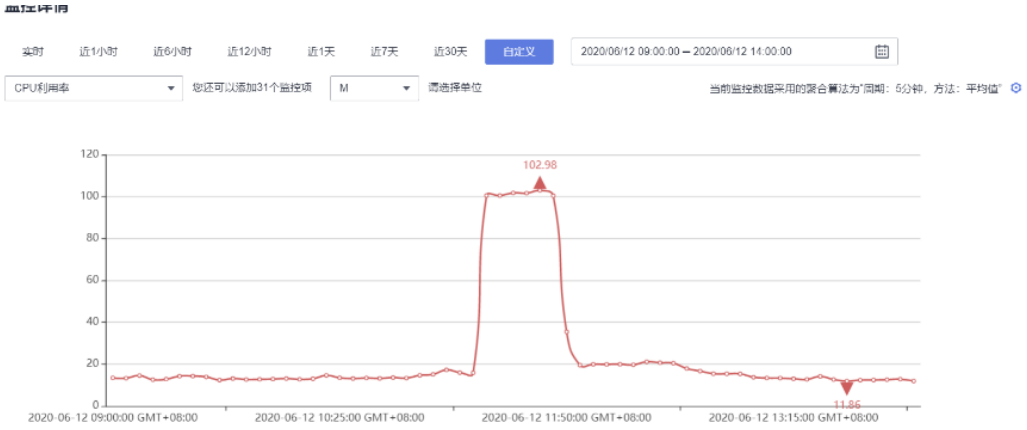
如果QPS过高，建议优化客户业务或者**变更实例规格**。不同实例规格支持的QPS请参考**实例规格**。

## 查找并禁用高消耗命令

使用了keys等消耗资源的命令，高消耗资源的命令即时间复杂度为O(N)或更高的命令，通常情况下，命令时间复杂度越高，在执行时消耗的资源越高，这会导致CPU使用率超高，容易触发主备倒换。关于各命令对应的时间复杂度信息请参见**Redis官网**。例如，使用了keys等消耗资源的命令，导致CPU超高，建议客户改成scan命令或者禁用keys命令。

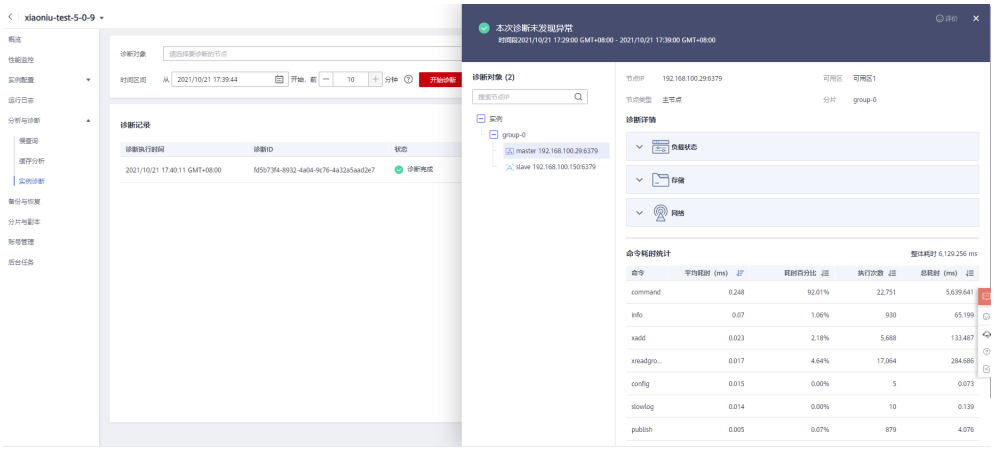
**步骤1** 通过性能监控功能，确认CPU使用率高的具体时间段。




**步骤2** 通过下述方法，找出高消耗的命令。

- 慢查询功能会记录执行超过指定时间阈值的命令，通过分析慢查询的语句和执行时长可帮助您找出高消耗命令，具体操参见**慢查询**。
- 通过实例诊断功能，选择CPU冲高的时间点进行诊断后，可以看到报告中的对应时间段命令的执行情况以及CPU耗时百分比，具体操作参见**实例诊断**。




**步骤3** 处理措施。

- 评估并禁用高风险命令和高消耗命令，例如**FLUSHALL**、**KEYS**、**HGETALL**等。
- 优化业务，例如避免频繁执行数据排序操作。
- **可选：**根据业务情况，选择下述方法对实例进行调整：
  - 调整实例为读写分离实例，对高消耗命令或应用进行分流。
  - 扩容实例增强实例处理能力。

**----结束**

## 是否存在 Redis 的持久化重写操作

对于主备和集群实例，华为云Redis实例默认开启AOF数据落盘，实例开启了AOF持久化功能后，华为云Redis会定期进行AofRewrite的磁盘整理，AOF磁盘持久化整理一般在以下2种场景执行：

- 数据量写入不大，AOF文件不大时，固定在每天的凌晨1-4点进行AOF持久化重写。所以容易出现这个时间点实例CPU使用率超高的现象。

- 数据量写入过大，AOF文件大小超过阈值（缓存实例容量的3-5倍）时，不论当前的所处的时间，会自动触发后台AOF持久化重写。

Redis的持久化重写操作（Bgsave或Bgrewriteaof）比较消耗CPU资源（请参考**为什么使用Fork执行Bgsave和Bgrewriteaof**），Bgsave和Bgrewriteaof会调用系统的Fork机制，造成CPU短暂时间冲高。

如果客户没有需要用到持久化功能，建议将该功能关闭（请根据实际业务慎重操作，关闭持久化功能会导致极端故障场景下恢复时，由于没有落盘造成的数据丢失）。关闭操作：在实例详情页面，选择“配置参数”页签，将“appendonly”修改为“no”。

# 3 Redis 实例内存使用率高问题排查和解决

## 问题现象

Redis可提供高效的数据库服务，当内存不足时，可能导致Key频繁被逐出、响应时间上升、QPS（每秒访问次数）不稳定等问题，进而影响业务运行。由于Redis自身运行机制（主从同步、延迟释放等），内存占用率可能出现略微超过100%的情况，此为正常情况，此时内存已经写满，用户需要考虑扩容，或者清理一些无用的数据。通常情况下，当内存使用率超过95%时需要及时关注。

## 排查原因

1. 查询指定时段的内存使用率信息，具体操作请参见**性能监控**。“内存利用率”指标持续接近100%。
2. 查询内存使用率超过95%的时间段内，“已逐出的键数量”和“命令最大时延”，均呈现显著上升趋势，表明存在内存不足的问题。

   建议客户登录控制台，参考**缓存分析**和**慢查询**，执行大Key扫描和慢查询。如果实例没有设置过期时间，会导致存储数据太多，内存被占满。
3. Redis实例如果内存满了但是key不多，可能原因是客户端缓冲区（output buffer）占用过多的内存空间。

   可以在Redis-cli客户端连接实例后，执行大key扫描命令：redis-cli --bigkeys，然后执行info，查看output buffer占用情况。

## 处理措施

1. DCS控制台提供了大Key和热Key的分析功能，你可参考**分析Redis实例大Key和热Key**减少大key和热key。

   **说明**

   Redis 3.0实例不支持热key分析，您可以通过**配置告警**的方式帮助您发现热key。
2. 执行**过期Key扫描**释放已经过期的Key，或手动清理一些不需要的Key，释放空间。
3. 其他优化建议：
   - **String类型数据的Value大小控制在10KB以内**。
   - **Hash、List、Set、Zset类数据结构，建议单Key中的元素不要超过5000个。**
   - Key的命名前缀为业务缩写，禁止包含特殊字符（比如空格、换行、单双引号以及其他转义字符）。

- Redis事务功能较弱，不建议过多使用。
- 短连接性能差，推荐使用带有连接池的客户端。
- 如果只是用于数据缓存，容忍数据丢失，建议关闭持久化（在实例参数配置中将appendonly参数修改为no即关闭AOF持久化）。
- 配置告警，便于提前发现大Key、热Key。
  - 参考**配置告警**配置节点级别的**内存利用率**监控指标的告警。

    如果某个节点存在大key，这个节点比其他节点内存使用率高很多，会触发告警，便于您发现潜在的大key。
  - 参考**配置告警**配置节点级别的**入网最大带宽**、**出网最大带宽**、**CPU利用率**监控指标的告警。

    如果某个节点存在热key，这个节点的带宽占用、CPU利用率都比其他节点高，该节点会容易触发告警，便于您发现潜在热key。

4. 如果实例内存使用率通过以上方式仍然很高，请考虑在业务低峰期扩大实例规格。具体操作请参见**变更实例规格**。

# 4 排查 Redis 实例带宽使用率高的问题

## 概述

Redis实例作为更靠近应用服务的数据层，通常会执行较多的数据存取并消耗网络带宽。不同的实例规格对应的最大带宽有所不同，当超过该规格的最大带宽时，会产生流控，流控会导致连接被丢弃，从业务角度可能会造成业务的延迟增大，客户端连接异常等问题。本节讲述如何排查Redis实例带宽使用率高的问题。

## 操作步骤

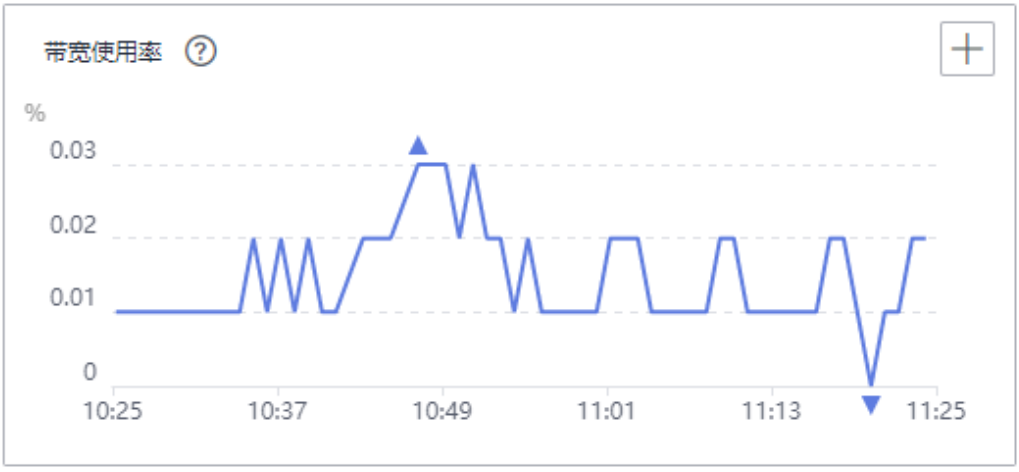
**步骤1** 查询带宽使用率。

查询实例在指定时段的带宽使用率。具体操作请参见**查看监控数据**。

通常来说，“网络瞬时输入流量”和“网络瞬时输出流量”快速上升，并持续大于实例最大带宽的80%时，需引起注意，可能流量不足。

需关注的监控指标为带宽使用率如下图。带宽使用率的计算公式：带宽使用率=（网络瞬时输入流量+网络瞬时输出流量）/（2*最大带宽限制）* 100%。

**图 4-1** 带宽使用率示例




其中，带宽使用率超过100%，不一定导致限流，有没有被流控需要看流控次数指标。

带宽使用率没有超过100%，也有可能有限流，因为带宽使用率是上报周期实时值，一个上报周期检查一次。流控检查是秒级的，有可能存在上报周期间隔期间，流量有秒级冲高，然后回落，待上报带宽使用率指标时已恢复正常。

**步骤2** 优化带宽使用率。

1. 当业务的访问量与预期带宽消耗不匹配，例如带宽使用率的增长趋势和QPS的增长趋势明显不一致（可结合网络瞬时输入流量和网络瞬时输出流量，分析业务是读业务和还是写业务导致的流量上涨）。对于单个节点带宽使用率上涨，您可以通过缓存分析功能，发现实例中存在的大Key，具体操作请参见**大key分析**。对大Key（通常大于10 KB）进行优化，例如将大Key拆分、减少对大Key的访问、删除不必要的大Key等。
2. 经过上述步骤优化后流量使用率依旧较高，可评估升级至更大内存的规格，以承载更大的网络流量。具体操作请参见**变更实例规格**。

   **说明**

   - 在正式升级实例的规格前，您可以先购买一个按需付费的实例，测试要升级到的目标规格是否能够满足业务的负载需求，测试完成后可将其释放。释放实例请参考**删除实例**。
   - 通过控制台对Redis实例进行**带宽临时扩容**，可临时解决业务流量高峰，带宽超限的问题。

**----结束**

# 5 使用 Jedis 连接池报错如何处理?

在使用Jedis连接池JedisPool模式下，比较常见的报错如下：

```
redis.clients.jedis.exceptions.JedisConnectionException: Could not get a resource from the pool
```

首先确认DCS缓存实例是正常运行中状态，然后按以下步骤进行排查。

**步骤1** 网络

1. 核对IP地址配置

   检查jedis客户端配置的ip地址是否与DCS缓存实例配置的子网地址一致，如果从公网访问，则检查是否与DCS缓存实例绑定的弹性ip地址一致，不一致则修改一致后重试。

2. 测试网络

   在客户端使用ping和Telnet小工具测试网络。

   - 如果ping不通：
     - VPC内访问Redis 3.0或企业版Redis时，要求客户端与DCS缓存实例的VPC相同，安全组相同或者DCS缓存实例的**安全组放开了6379端口访问**。
     - VPC内访问Redis 4.0/5.0/6.0基础版时，要求客户端与DCS缓存实例的VPC相同，Redis实例如果配置了白名单，需确保白名单中包含客户端IP，允许客户端访问，参考**配置白名单**。
     - 公网SSL方式访问Redis 3.0时，要求DCS缓存实例**安全组放开了36379端口访问**。
     - 公网直接访问（非SSL方式）Redis 3.0时，要求DCS缓存实例**安全组放开了6379端口访问**。
   - 如果IP地址可以ping通，telnet对应的端口不通，则尝试重启实例，如重启后仍未恢复，请联系技术支持。

**步骤2** 检查连接数是否超限

查看已建立的网络连接数是否超过JedisPool配置的上限。如果连接数接近配置的上限值，则建议重启服务观察。如果明显没有接近，排除连接数超限可能。

Unix/Linux系统使用：

**netstat -an | grep 6379 | grep ESTABLISHED | wc -l**

Windows系统使用：

**netstat -an | find "6379" | find "ESTABLISHED" /C**

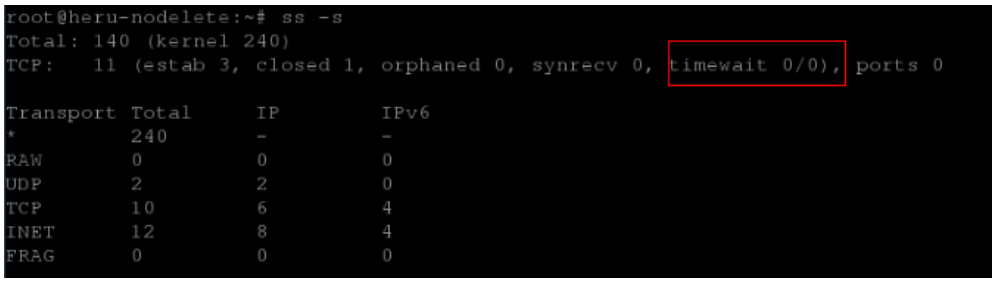
**步骤3** 检查JedisPool连接池代码

如果连接数接近配置的上限，请分析是业务并发原因，或是没有正确使用JedisPool所致。

对于JedisPool连接池的操作，每次调用**jedisPool.getResource()**方法之后，需要调用**jedisPool.returnResource()**或者**jedis.close()**进行释放，优先使用close()方法。

**步骤4** 客户端TIME_WAIT是否过多

通过**ss -s**查看**time wait**链接是否过多。




如果**TIME_WAIT**过多，可以调整内核参数（/etc/sysctl.conf）：

```
##当出现SYN等待队列溢出时，启用cookies来处理，可防范少量SYN攻击
net.ipv4.tcp_syncookies = 1
##允许将TIME-WAIT sockets重新用于新的TCP连接
net.ipv4.tcp_tw_reuse = 1
##开启TCP连接中TIME-WAIT sockets的快速回收
net.ipv4.tcp_tw_recycle = 1
##修改系統默认的TIMEOUT时间
net.ipv4.tcp_fin_timeout = 30
```

调整后重启生效：**/sbin/sysctl -p**

**步骤5** 无法解决问题

如果按照以上原因排查之后还有问题，可以通过抓包并将异常时间点、异常信息以及抓包文件发送给技术支持协助分析。

抓包可使用tcpdump工具，命令如下：

**tcpdump -i eth0 tcp and port 6379 -n -nn -s 74 -w dump.pcap**

Windows系统下还可以安装Wireshark工具抓包。

**说明**

公网访问时请将端口改成36379。

网卡名请改成实际的网卡名称。

**----结束**