跨级建云 – Redis灾备

1. 概述
2. 当前Redis分布及数据存储



* 1. CMDP负责写会议即时数据。
  2. CSS负责读会议数据，写一些自身逻辑需要的会议信息、license信息
  3. CMC负责读会议数据(从Redis)，写会议临时数据（点名）、用户账户信息、数据库二级缓存信息
     1. 会议临时数据灾备后允许用户操作
     2. 用户账户信息灾备后可让用户重新登录
     3. 二级缓存信息灾备后，无法读取会重新读取数据库并加载到Redis
  4. 会议数据需要主备同步，会管Redis数据可以重新构建。

1. Redis相关知识
   1. 可以进行一主多从，不能一从多主(后设置后生效)
   2. 可以链路式主从、树形主从，但不能环
   3. 不支持主主
   4. 从服务器可以写，不影响主
   5. 从服务器重启需要重新执行同步命令，主服务器重启会自动同步数据
2. 实际产生灾备场景
   1. 断电
   2. 路由器故障、网络中断
   3. 拔板卡(管理板卡、资源板卡)
3. 方案设计
4. 模型



* 1. 为每一个灾备平台域分配一个备Redis。
  2. Redis过Haproxy，端口区分不同Redis。
  3. 为保证读写分离和效率，支持读写端口分离（由业务保证只读）
  4. 跨域走两层Haproxy反向代理。
  5. 灾备发生后，PMS动态控制Haproxy代理节点，重置到本地Redis，业务无需感知服务变化。
  6. 灾备发生后，重置本地备Redis为主，防止数据回写。
  7. 需要封装Jedis以及RedisClient，支持连接池，业务根据用户所属平台域动态切换Redis。也可以业务使用多client策略。
  8. 灾备发生的时候，PMS记录托管时间，托管期间该平台域是否重启异常等，若没有，就直接同步备平台数据给主平台域（slaveof）。若根本没托管过，就以主平台域本地数据为主。
  9. 预留6381-6400作为备Redis端口，原来的会管Redis考虑合并到主Redis，分库使用。

1. PMS协商（主平台启动）



* 1. Redis本身为单线程，且在同步数据的时候会将请求缓存起来，所以可以在回复ok后执行下一道命令，无需等待同步结束。
  2. 备PMS回复内容中携带托管主平台域的时间，若没有托管过，就无需从备平台域同步数据。（消息机制走服务发现还是MQ再议，建议MQ）
  3. 数据恢复后，需要重新指定备平台域Redis为备。
  4. 若无需同步或者连接不上备PMS，以当前Redis的aof数据为主（自动加载）

1. PMS协商（备平台启动）
   1. 若主正常，直接flushall，slaveof同步数据
   2. 若主异常，加载aof文件，广播当前平台域状态以及托管状态。
      1. CSS会再次恢复会议，包含主平台域和自身平台域的数据。
   3. 若正在恢复当中主恢复了，走主平台恢复逻辑。待数据同步后，主CSS只处理待恢复的会议，不处理恢复中、已经恢复的会议。备CSS停止恢复主平台域会议。
2. 数据变化



* 1. 正常情况主备Redis数据一致
  2. 灾备发生后备平台域通过Haproxy动态调整节点进行写入
  3. 灾备恢复时，主平台域拉取备Redis数据，会议实际控制已经变为备平台域，也有可能某些会议还未恢复，会议实际控制还未切换，主平台域可以继续恢复。
  4. CSS若发现会议实际归属为当前平台域，则走正常恢复逻辑，若实际归属为其它平台域，不再恢复。

1. 模块波及

|  |  |
| --- | --- |
| **模块** | **波及** |
| CMC | 根据用户域所属平台域使用Redis |
| CSS | 根据用户域所属平台域使用Redis |
| CMDP | 根据用户域所属平台域使用Redis |
| PMS | 控制灾备处理流程  协商主从复制流程（Redis命令）  控制Haproxy切换Redis节点（动态加载） |
| 部署 | 根据灾备情况启动备Redis服务器  Haproxy中加入Redis反向代理设置  提供查询Redis和平台域绑定关系 |