DRBD 跨集群异地数据同步

方案验证报告

测试日期：2022年7月7日

目录

[概览 4](#_Toc108165310)

[一、环境信息 4](#_Toc108165311)

[1.1 研发高可用测试环境 CNP 集群（主） 4](#_Toc108165312)

[1.2 jinan-lab 租户侧 ICE 集群（备） 5](#_Toc108165313)

[1.3 网络拓扑 6](#_Toc108165314)

[二、主备集群方案配置步骤 8](#_Toc108165315)

[2.1 备集群配置本地存储 hostpath-provisioner 8](#_Toc108165316)

[2.2 备集群部署配置 es-tenant 8](#_Toc108165317)

[2.3 重建备集群 PVC 和 PV 8](#_Toc108165318)

[2.4 配置 drbd 资源 9](#_Toc108165319)

[2.5 主集群 es 从本地存储切换到 drbd 13](#_Toc108165320)

[三、主备同步性能测试 14](#_Toc108165321)

[4.1 主备间直通网络性能 14](#_Toc108165322)

[4.2 跨集群 DRBD 同步效率测试 14](#_Toc108165323)

[五、主备同步数据可靠性测试 17](#_Toc108165324)

[5.1 测试目标 17](#_Toc108165325)

[5.2 测试步骤 17](#_Toc108165326)

[5.3 测试结果 18](#_Toc108165327)

[六、主备增量数据同步测试 21](#_Toc108165328)

[6.1 drbd 纳管已有数据盘初始化同步测试 21](#_Toc108165329)

[6.2 主备集群网络中断期间 drbd 增量数据同步测试 24](#_Toc108165330)

[6.3 主备集群网络不稳定（延迟/拥塞）状态下数据同步测试 27](#_Toc108165331)

[七、ES 主备同步功能验证 29](#_Toc108165332)

[7.1 主集群正常关机，主备切换验证 29](#_Toc108165333)

[7.2 备集群临时变为主集群后主备角色回切验证 32](#_Toc108165334)

[八、问题记录 34](#_Toc108165335)

[九、术语说明 35](#_Toc108165336)

[9.1 DRBD 复制协议 35](#_Toc108165337)

[9.2 单主、双主模式 36](#_Toc108165338)

[9.3 脑裂 36](#_Toc108165339)

[9.4 DRBD 元数据 37](#_Toc108165340)

[9.5 连接状态信息 38](#_Toc108165341)

[9.6 资源角色 38](#_Toc108165342)

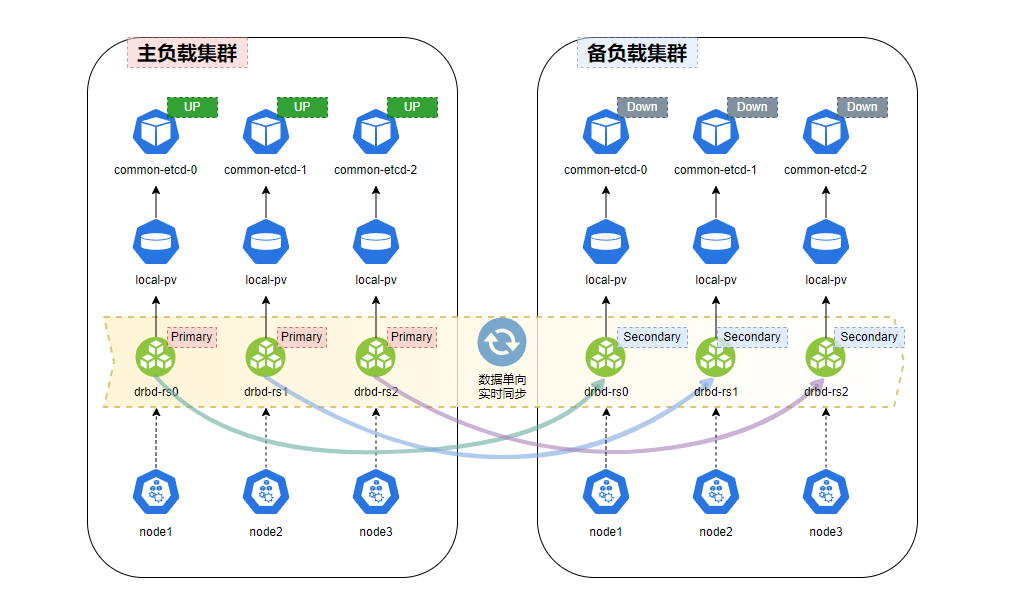
[9.7 磁盘状态 39](#_Toc108165343)

[9.8 连接信息 39](#_Toc108165344)

[十、参考资料 40](#_Toc108165345)

## 概览

架构参考



## 一、环境信息

### 1.1 研发高可用测试环境 CNP 集群（主）

* 物理机 IP（代理）： 10.110.21.42
* 虚机：
  + 100.200.8.136 master1
  + 100.200.8.137 master2
  + 100.200.8.142 master3
* 主集群已部署 es 公共组件，可正常读写数据，数据通过 hostpath-provisioner 提供动态 pv, 数据落在本地磁盘；
* 主集群未配置 drbd;

#### 1.1.1 主集群 DRBD 资源规划

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| DRBD 资源名称 | 端口号 | 节点 | IP | DBRD 设备名称 | 底层设备名称 | 对端地址 | 挂载点 |
| es0 | 7791 | master1 | 100.200.8.136 | /dev/drbd2 | /dev/loop2 | 10.110.26.199:27791 | /var/hpvolumes/ |
| es1 | 7792 | master2 | 100.200.8.137 | /dev/drbd2 | /dev/loop2 | 10.110.26.199:27792 | /var/hpvolumes/ |
| es2 | 7793 | master3 | 100.200.8.142 | /dev/drbd2 | /dev/loop2 | 10.110.26.199:7793 | /var/hpvolumes/ |

### 1.2 jinan-lab 租户侧 ICE 集群（备）

* EIP: 10.110.26.199 （带宽 50Mbps）
* 虚机：
  + 192.168.3.58 master1
  + 192.168.2.69 master2
  + 192.168.3.74 master3 (+EIP)
* 备集群未部署 es;
* 备集群未配置 drbd;

#### 1.2.1 备集群 DRBD 资源规划

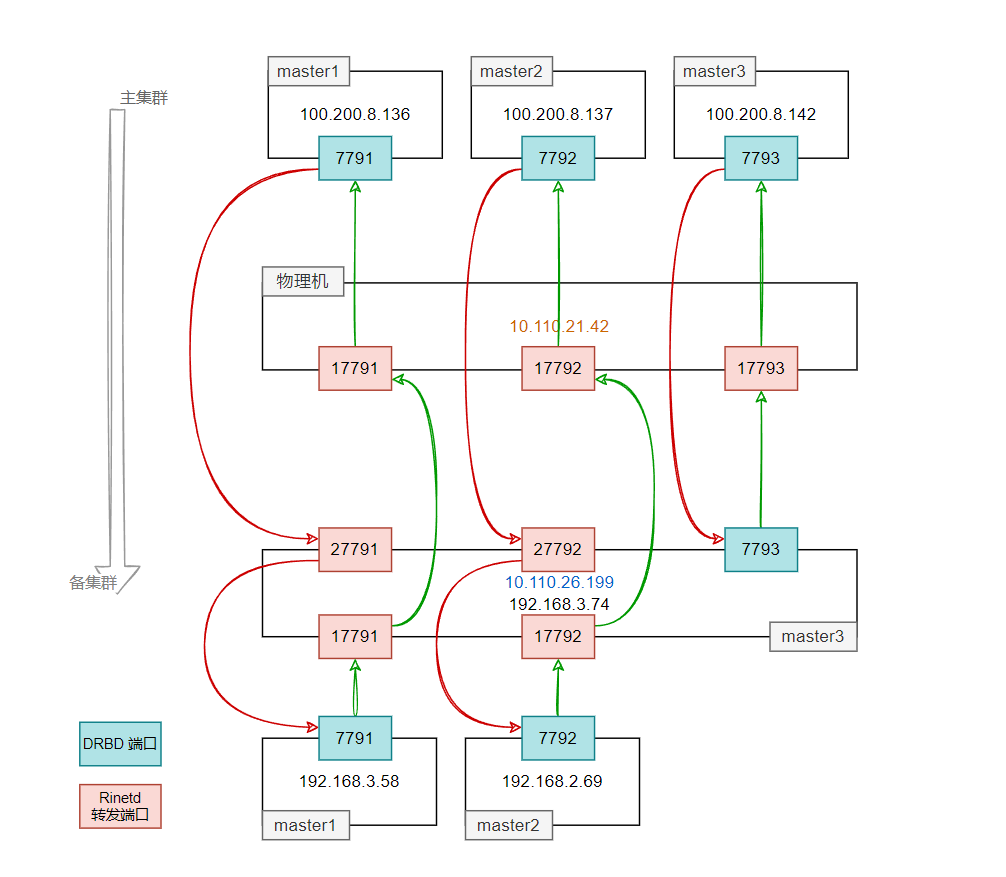
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| DRBD 资源名称 | 端口号 | 节点 | IP | DBRD 设备名称 | 底层设备名称 | 对端地址 | 挂载点 |
| es0 | 7791 | master1 | 192.168.3.58 | /dev/drbd2 | /dev/loop2 | 192.168.3.74:17791 | /var/hpvolumes/ |
| es1 | 7792 | master2 | 192.168.2.69 | /dev/drbd2 | /dev/loop2 | 192.168.3.74:17792 | /var/hpvolumes/ |
| es2 | 7793 | master3 | 192.168.3.74 | /dev/drbd2 | /dev/loop2 | 10.110.21.42:17793 | /var/hpvolumes/ |

### 1.3 网络拓扑

#### 1.3.1 主备测试集群网络条件

* 主集群位于研发环境物理机上置备的虚机，虚机内部网络可以通过物理机默认网关，与外部（备集群 EIP）直通；
* 外部网络无法直接访问主集群虚机；
* 备集群只有 EIP 所在节点能够访问外部网络（主集群所在物理机）；
* 外部无法访问备集群内其他节点；

因此基于 TCP 端口转发，设计了如下网络拓扑方案：



#### 1.3.2 代理转发配置

基于 rinetd 服务做 TCP 端口转发

##### 1.3.2.1 主集群代理转发配置：

rinetd on 10.110.21.42

# cat /etc/rinetd.conf

# bindadress bindport connectaddress connectport

0.0.0.0 17791 100.200.8.136 7791

0.0.0.0 17792 100.200.8.137 7792

0.0.0.0 17793 100.200.8.142 7793

##### 1.3.2.2 备集群代理转发配置：

rinetd on 10.110.26.199

# cat /etc/rinetd.conf

# bindadress bindport connectaddress connectport

0.0.0.0 17791 10.110.21.42 17791

0.0.0.0 17792 10.110.21.42 17792

0.0.0.0 27791 192.168.3.58 7791

0.0.0.0 27792 192.168.2.69 7792

## 二、主备集群方案配置步骤

### 2.1 备集群配置本地存储 hostpath-provisioner

采用组件化部署 hostpath-provisoner, sc 使用默认创建的 sc , 目录使用默认的目录 /var/hpvolumes

此处省略部署步骤。

### 2.2 备集群部署配置 es-tenant

# 部署 chart 包

helm install es-tenant -n common ./es-tenant

# 待 chart 包部署完成，缩容 es-tenant 副本数到 0

kubectl scale --replicas=0 sts/elasticsearch -n common

### 2.3 重建备集群 PVC 和 PV

由于主集群 es 数据使用 hostpath-provisioner 提供动态 pvc 自动生成的 pv ，数据落在 pv 的宿主机目录下，如 /var/hpvolumes/pvc-xxxx-xxxx-xxxx-xxxx ，

而 /var/hpvolumes 这个目录即数据盘的挂载点，

因此通过 drbd 纳管同步之后，实际同步的文件是 pv 的数据目录 pvc-xxxx-xxxx-xxxx-xxxx ，

所以，必须基于主集群的 pvc 和 pv 元数据克隆到备集群，保证 es 能够读写同样的 pv 数据目录。

#### 2.3.1 操作步骤：

1. 将主集群 PVC 和 PV 通过 kubectl get -oyaml 导出到本地文件；
2. 编辑并清理无关字段后（annotations, timestamp, resource version , status 等）；
3. 拷贝 yaml 文件到备集群，通过 kubectl apply -f 重建；（此处省略 yaml 样例）
4. 检查 pvc 、pv 状态是否正常（pvc=Bound, pv=Available）;

### 2.4 配置 drbd 资源

由于集群网络拓扑结构较特殊，目前需要手动配置 drbd 资源（通过组件化部署 drbd 时，comdeploy 中 resource 字段留空）。

1. 出于存量数据纳管需求，默认使用 loop 外置方式管理 drbd 元数据；
2. 备集群磁盘大小以及扇区数需要与主集群一致，若不一致可能 drbd 会拒绝同步，需要手动配置 drbd 可用的扇区数为较小一方的扇区数；
3. 默认所有资源均设置为备角色（secondary）；

#### 2.4.1 备集群配置 DRBD

备集群 master1 配置文件示例：

# cat /etc/drbd.d/cke\_es0.res

resource es0 {

net {

protocol A;

allow-two-primaries no;

}

options {

auto-promote no;

}

on master1 {

meta-disk /dev/loop0;

device /dev/drbd2;

disk /dev/loop2;

address 192.168.3.58:7791;

node-id 1;

}

on remote\_100.200.8.136 {

meta-disk /dev/loopx;

device /dev/drbd2;

disk /dev/loop2;

address 192.168.3.74:17791;

node-id 2;

}

}

#### 2.4.2 主集群配置 DRBD

主集群 master1 drbd 资源配置示例：

resource es0 {

net {

protocol A;

allow-two-primaries no;

}

options {

auto-promote no;

}

on master1 {

meta-disk /dev/loop3;

device /dev/drbd2;

disk /dev/loop2;

address 100.200.8.136:7791;

node-id 2;

}

on remote\_192.168.3.58 {

meta-disk /dev/loopx;

device /dev/drbd2;

disk /dev/loop2;

address 10.110.26.199:27791;

node-id 1;

}

}

#### 2.4.3 初始化元数据并启动首次同步

# 1. 初始化元数据

drbdadm create-md es0

# 2. 挂载底层数据盘

drbdadm attach es0

# 3. 重新建立连接

drbdadm disconnect es0

drbdadm connect es0

# 4. 登陆到对端节点，重建连接

drbdadm disconnect es0

drbdadm connect es0

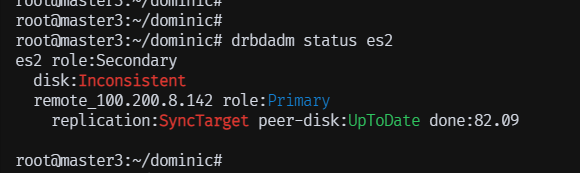
# 5. 查看 drbd 资源状态

drbdadm status es0

# 6. 查看 drbd 资源实时同步详情

drbdtop

drbd 资源状态示例 （正在同步）：



### 2.5 主集群 es 从本地存储切换到 drbd

1. 确认主集群 es 所在节点 drbd 资源状态正常，角色为 primary: drbdadm status
2. 缩容主集群 es 副本数为 0： kubectl scale sts elasticsearch -n common --replicas=0
3. 挂载 drbd 资源到临时挂载点： mount /dev/drbd2 /data1
4. 拷贝 es 数据到临时挂载点：cp -r /var/hpvolumes/\* /data1
5. 解除临时挂载点： umount /data1
6. 挂载 pv 目录到 drbd 资源： mount /dev/drbd2 /var/hpvolumes
7. 检查 es 数据目录是否存在：ls -l /var/hpvolumes/
8. 检查 drbd 数据同步状态；drbdadm status 或 drbdtop
9. 登陆主集群其他 es 节点，重复第 1~8 步；
10. 所有节点切换完成并且同步状态正常，执行 es 扩容到 3 副本：kubectl scale sts elasticsearch -n common --replicas=3
11. 等待 es 所有 pod 就绪，登入业务系统，查询 es 数据是否正常访问，且无丢失；（切换前的数据仍然在原始磁盘的 /var/hpvolumes 目录下，解除 drbd 挂载后仍可继续访问）

## 三、主备同步性能测试

### 4.1 主备间直通网络性能

#### 4.1.1 测试目标

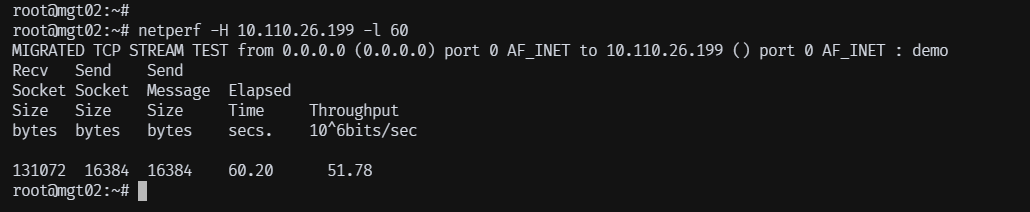
测试两个主备集群的（代理）节点间直连网络性能。

#### 4.1.2 测试步骤

1. 登陆主集群物理机节点 10.110.21.42 ；
2. 执行 `netperf -H 10.110.26.199 -l 60`

#### 4.1.3 测试结果

主备代理直连网络传输速率（tcp）约为 51Mb/s , 与 EIP 带宽（50Mbps）基本一致。



### 4.2 跨集群 DRBD 同步效率测试

#### 4.2.1 测试目标

验证 drbd 跨集群数据同步效率，是否有损耗，以及损耗是否在预期范围内（官方数据 3%）。

#### 4.2.2 测试步骤：

警告：以下测试步骤必须使用空数据盘进行测试！否则数据会丢失不可找回！

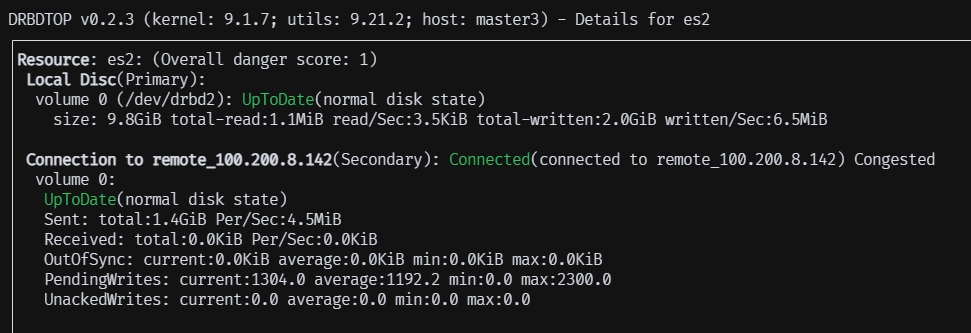
1. 默认 drbd 资源配置同步方式为 C (全同步)；
2. 拉起主集群 1 节点 drbd 资源，确保资源状态正常（Connected/UpToDate；
3. 确保对端备集群 drbd 资源为备角色，并确保资源状态正常（Connected/UpToDate）；
4. 新开窗口执行 drbdtop 进入交互式界面，监控资源实时状态；
5. 向主集群 drbd 挂载盘 写入2G测试文件: dd if=/dev/zero of=/var/hpvolumes/test bs=4k count=500000 oflag=direct
6. 查看 drbdtop 内实时传输状态；
7. 将 drbd 资源置为非纳管状态： drbdadm down es2；
8. 使用 同样的 dd 命令向 drbd 底层盘写入数据对比
9. 修改 drbd 配置文件中同步策略为 A(异步); 重复5~8；
10. 退出 drbdtop 重新进入（重置数据统计），删除刚刚的测试文件，观察数据统计；

#### 4.2.3 测试结论：

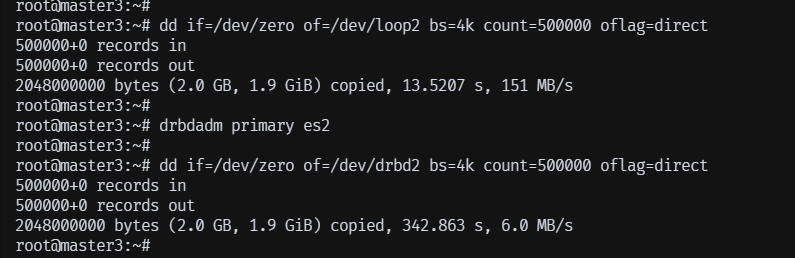
1. 基于 4.1 中网络直通性能测试数据计算，drbd 数据同步速率损耗为 2.3% ~ 4%，符合官方数据 3%；
2. drbd 吞吐量和延迟取 IO 系统和网络的最劣值，drbd 数据落盘速率即 drbd 同步速率；
3. 同步策略不影响 drbd 同步效率；
4. 文件删除操作仅产生微量数据差异；

#### 4.2.4 过程截图

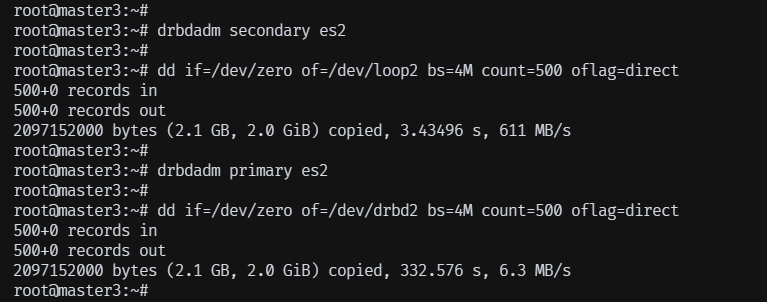
dd 写文件时的 drbd 实时同步状态：



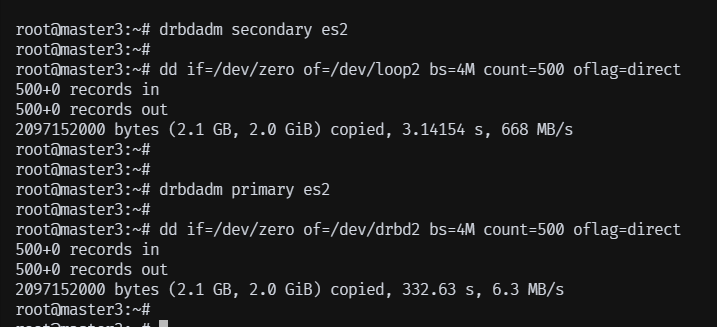
drbd 吞吐量/延迟对比（4K）：



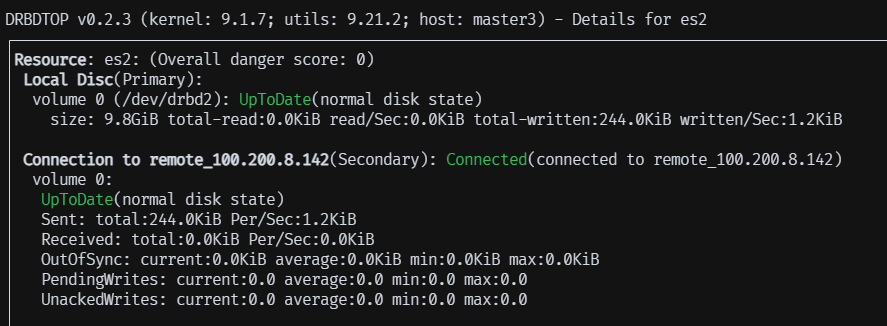
drbd 吞吐量/延迟对比（4M）：



异步模式下吞吐量/延迟对比（4M）:



删除文件时，数据同步监测：



## 五、主备同步数据可靠性测试

### 5.1 测试目标

测试 drbd 数据同步可靠性，即主节点正常关机以及异常宕机重启场景下，备节点切换为主节点时，存在多少数据量的丢失，以及数据完整性。

### 5.2 测试步骤

1. 主集群测试节点 dbrb 使用全同步策略（C），资源角色置为 primary;
2. 执行测试脚本，同时向 drbd 磁盘测试文件以及本地磁盘测试文件每隔10ms 持续写入时间戳数据；
3. 在新窗口中登陆主集群测试节点，通过 sysrq 模拟立即断电重启 （echo "b" > /proc/sysrq-trigger）；
4. 登陆备集群测试节点，切换 drbd 为主角色，挂载 drbd 盘到目录，查看同步的测试文件的最后时间戳记录；
5. 故障节点重启后，登陆查看本地磁盘测试文件最后记录的时间戳，并对比；
6. 将故障节点 drbd 断开连接（避免脑裂），设置为主角色并挂载目录，查看本地 drbd 内最后记录时间戳，并对比；
7. 恢复 drbd 主备状态；修改同步策略为异步（C），重复 2~6;
8. 断电后登陆备节点，等待主节点开机并恢复drbd连接，数据同步完成后，备节点切换主节点，检查数据；
9. 通过 reboot 命令优雅重启，重复 2~5；

数据写入脚本：

#!/bin/bash

while true; do

  t=$(date +%s.%N|cut -b 1-14)

  echo $t >>/data1/ha-test.log &

  echo $t >>/root/ha-test.log

  sleep 0.01

done

### 5.3 测试结果

#### 5.3.1 总体结论

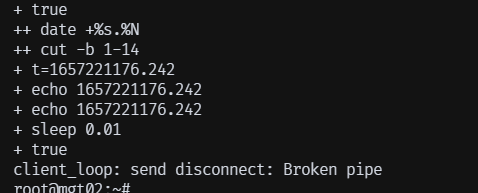
1. 优雅关机/重启，数据具备高可靠性（无丢失），处于内核空间的 drbd 资源（网络）断连要晚于用户空间业务进程终止，即业务侧进程停止后， drdb 仍有足够时间同步数据；
2. 异常断电重启，drbd 主备数据同步具备高可靠性（主备一致无差异）；
3. 异常断电重启，原始数据完整性无法保证：本地磁盘数据与 drbd 数据都会大概率出现文件损坏/脏数据问题；

#### 5.3.2 测试数据汇总

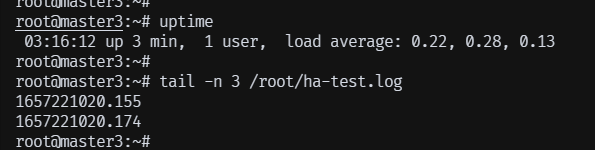
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 测试轮次 | 脚本最后数据产生时间  （控制台输出） | 本地磁盘最后记录时间 | 本地 drbd 盘最后记录时间 | 备节点 drbd 盘最后记录时间 | 本地磁盘数据损失量(秒) | 本地 drbd 盘数据损失量（秒） | 远程数据损失量（秒） |
| 优雅重启 | 1657072255.714 | 1657072255.714 | 1657072255.714 | 1657072255.714 | 0 | 0 | 0 |
| 异常断电第一次 | 1657221176.242 | 1657221020.174 | 1657221174.413 | 1657221174.413 | 156.068 | 1.829 | 0 |
| 异常断电第二次 | 1657221962.922 | 1657221960.901 | 1657221943.685 | 1657221943.685 | 2.021 | 19.237 | 0 |
| 异常断电（异步复制协议） | 1657222692.911 | 1657221960.901 | 1657221943.685 | 1657221943.685 | 732.010 | 749.226 | 0 |

#### 5.3.3 测试过程截图（部分）：

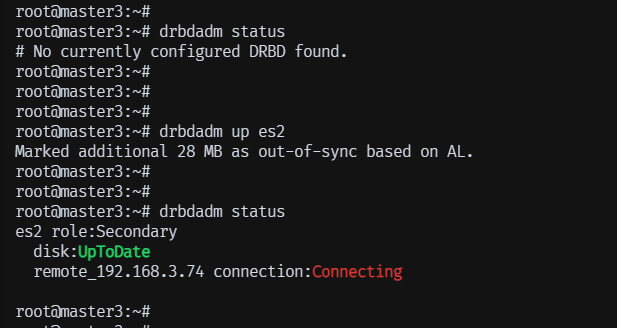
系统宕机前脚本最后执行记录：



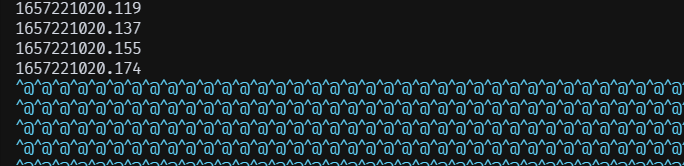
主节点本地文件最后数据：



主节点 从故障中重启后，drbd 启动时显示有未同步数据:



异常断电测试后，检查文件内容，发现不可读的脏数据：



## 六、主备增量数据同步测试

### 6.1 drbd 纳管已有数据盘初始化同步测试

#### 6.1.1 测试目标

当 drbd 基于已有存量数据的磁盘进行纳管时，

1. 测试首次初始化同步效率；
2. 验证同步未完成期间是否影响磁盘正常读写，以及读写效率是否有影响；

#### 6.1.2 测试步骤

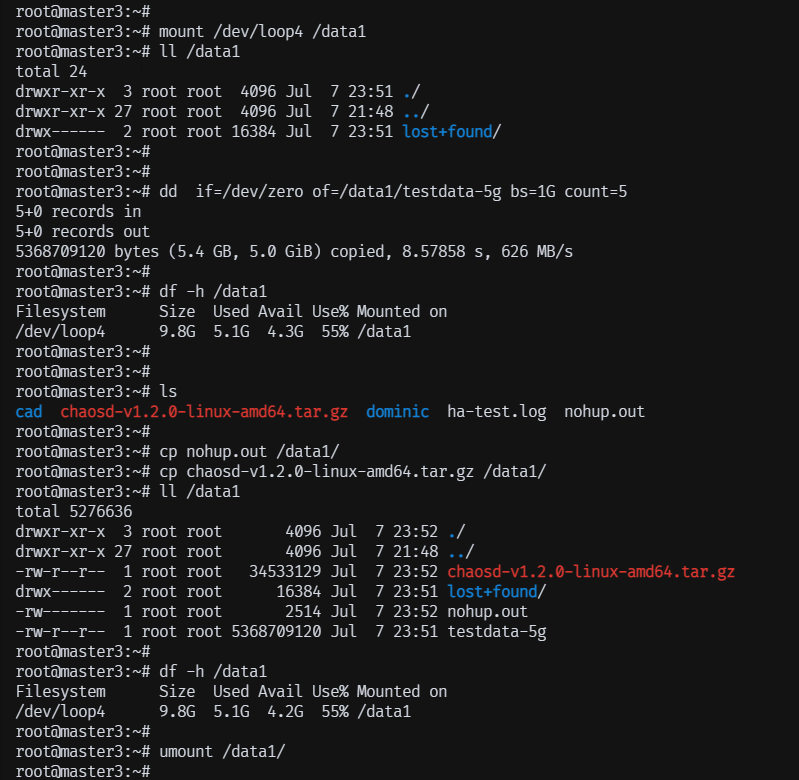
1. 通过 loop 设备虚拟一块大小为 20G 的磁盘作为数据盘： dd if=/dev/zero of=/tmp/test.disk bs=1G count=10; losetup /dev/loop4 /tmp/test.disk
2. 通过 loop 设备虚拟一块大小为 1G 的磁盘作为 drbd 元数据存储盘： dd if=/dev/zero of=/tmp/test bs=1G count=1; losetup /dev/loop5 /tmp/test.meta
3. 初始化虚拟数据盘 /dev/loop4 文件系统： mkfs.ext4 /dev/loop4
4. 直接挂载虚拟盘: mount /dev/loop4 /data1
5. 向虚拟盘中写入大文件（5G）： dd if=/dev/zero of=/data1/testdata-5g bs=1G count=5 ; 随后拷入任意个小文件；
6. 解除挂载：umount /data1
7. 创建 drbd 资源文件，配置使用 /dev/loop4 作为数据同步磁盘，/dev/loop5 作为外置元数据盘;
8. 登陆备集群某节点，重复 1~4 步，创建对端 drbd 资源；
9. 将第一个节点 drbd 置为主角色，并建立连接， 挂载 drbd 盘到本地目录： mount /dev/drbd3 /data1
10. 连接后查看初始同步状态以及同步速率：drbdtop
11. 初始同步过程中，继续向 drbd 盘中写入大文件测试吞吐量：dd if=/dev/zero of=/data1/testdata-3g bs=1G count=3 oflag=direct

#### 6.1.3 测试结论

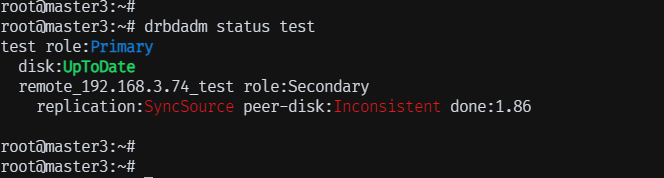
1. 初始化同步数据通常大于实际磁盘文件数据量，小于或等于磁盘容量；
2. 初始化同步效率约等于最大可用带宽；
3. 初始化同步期间，主节点文件读取正常，并可同时写入数据；
4. 初始化同步期间，主节点新数据写入速率与最大可用带宽一致（可满速写入）；

#### 6.1.4 测试过程截图

主集群置备测试资源：

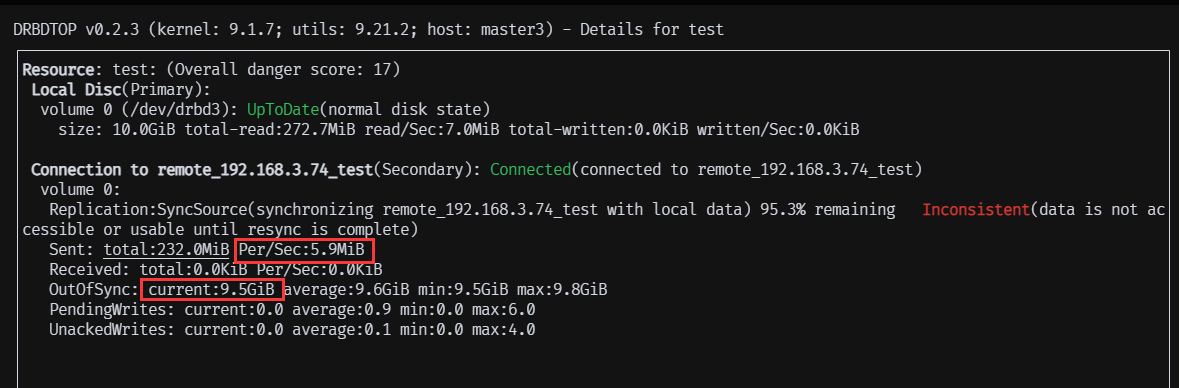


drbd 初始化同步状态：

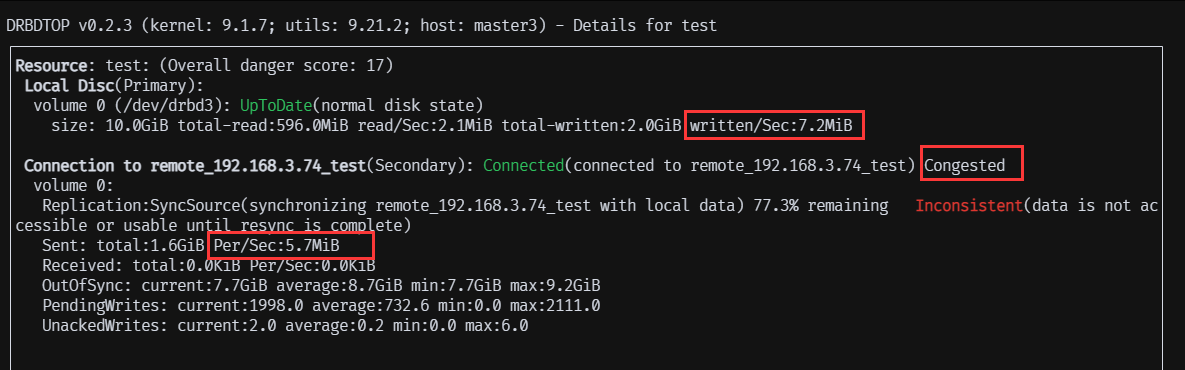


drbd 实时同步速率以及未同步数据量查看：

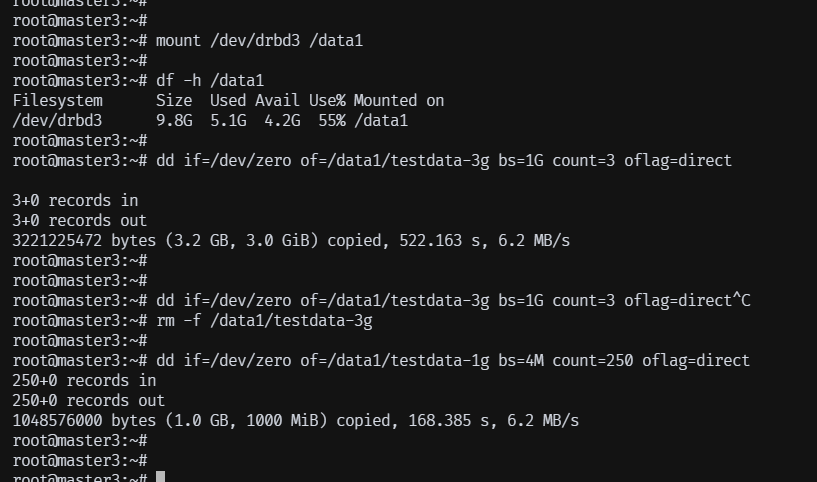
drbdtop命令查看



drbd 初始化同步期间，同时写入数据测试：



drbd 初始化同步期间，多次文件写入速率测试：



·

### 6.2 主备集群网络中断期间 drbd 增量数据同步测试

#### 6.2.1 测试目标

当主备集群网络中断时，

1. 验证全同步复制模式下（protocol A）是否影响主集群业务数据正常写入，以及写入效率是否有影响；
2. 验证网络中断期间，主集群写入的增量数据，在网络恢复后是否可以正常同步到备集群；

#### 6.2.2 测试步骤

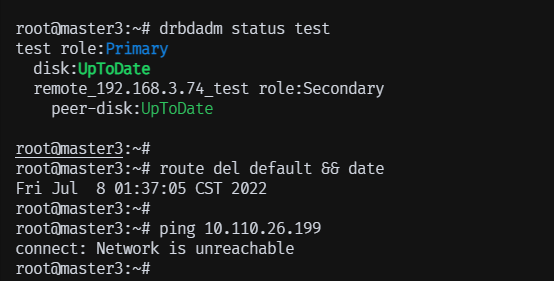
1. 基于 6.1 测试用例中置备的主备 drbd 资源，确认完成初始化同步，且 drbd 连接状态正常，可正常同步数据；
2. 中断主备之间的网络连接；（ 移除默认路由 route del default ）；
3. 观察 drbd 资源实时状态；记录各状态切换时间；
4. 使用 dd 命令向主节点 drbd 盘内写入 1G 测试数据; 并查看 drbd 资源状态；
5. 恢复网络连接；（增加默认路由 route add default gw 100.200.8.1）；
6. 观察 drbd 资源实时状态，并记录各状态切换时间；
7. 观察 drbd 恢复连接后，数据同步状态；

#### 6.2.3 测试结论

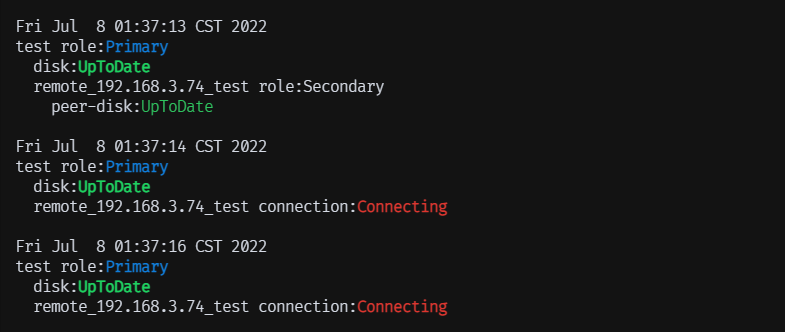
1. 网络断开约 8s 后，主节点 drbd 资源状态变为 connecting;
2. 网络断开期间，drbd 资源始终为 connecting 状态；
3. 断网期间，主节点 drbd 盘可正常读写文件，写入速率等同于本地磁盘 io；
4. 网络恢复约 23s 后，主节点 drbd 资源状态变为正在同步，并开始同步增量数据，同步速率接近最大可用带宽；

#### 6.2.4 测试过程截图

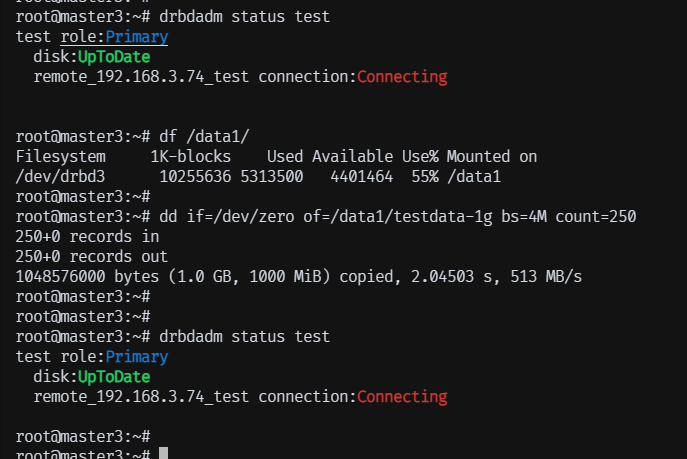
移除默认路由，使 drdb 网络中断,并记录时间:



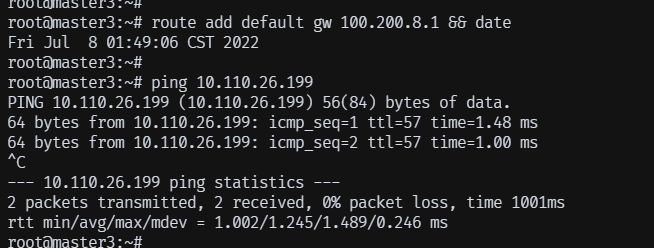
持续检测 drbd 状态，记录状态变化时间：



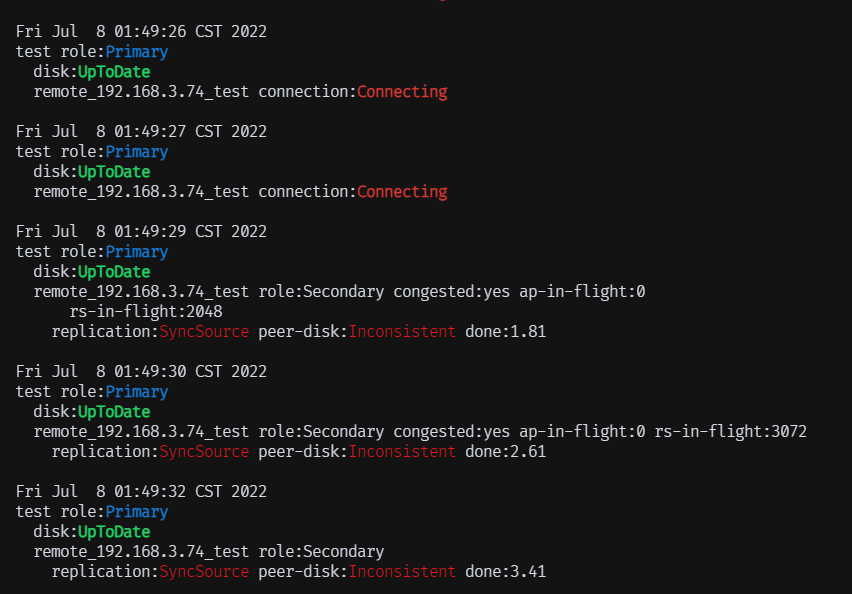
断网期间写入 1G 增量数据：



增加默认路由，恢复网络，并记录时间：



持续检测 drbd 状态并记录状态变更时间：



### 6.3 主备集群网络不稳定（延迟/拥塞）状态下数据同步测试

#### 6.3.1 测试目标

测试在主备集群网络拥塞、延迟等不稳定状态下，

1. 数据能否正常同步；
2. 主节点文件写入是否正常；
3. drbd 资源复制协议（A/C）是否影响 io 数据传输速率；

#### 6.3.2 测试步骤

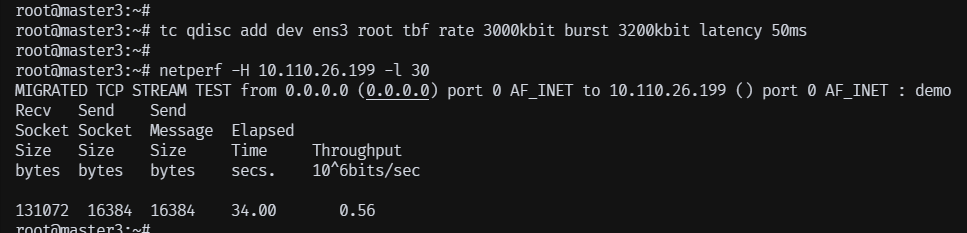
1. 正常状态下，测试主节点到对端代理节点的网络速度: netperf -H 10.110.26.199 -l 60
2. 正常状态下，测试主节点 drbd 文件写入速度；
3. 限制主节点网卡速度到 0.5Mbps (1%) ： tc qdisc add dev ens3 root tbf rate 3000kbit burst 3200kbit latency 50ms
4. 重复第 1、2 步 ，记录结果并对比；
5. 观察 drbd 资源状态；
6. 移除网卡限速并观察 drbd 资源状态；

#### 6.3.3 测试结论

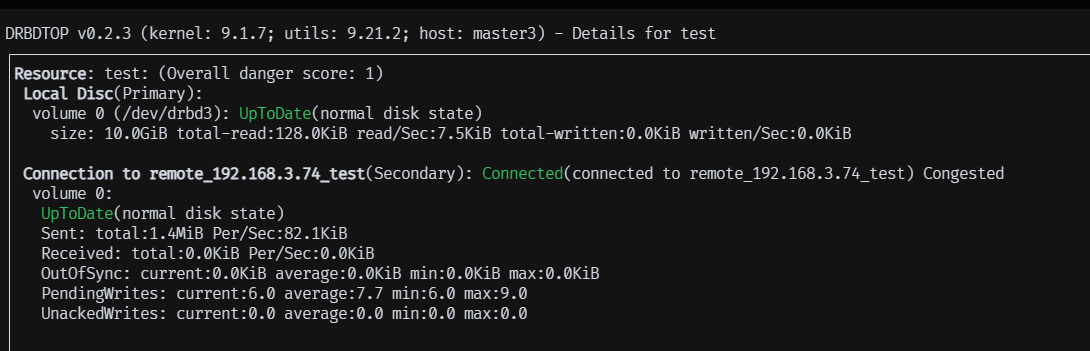
1. 限速条件下，drbd 读写正常；
2. drbd 数据盘 io 数据写入受限，吞吐量等于限速值（默认使用同步复制协议C）；
3. 限速条件下，异步复制协议（A）吞吐量高于限速值（200%）；

#### 6.3.4 测试过程截图

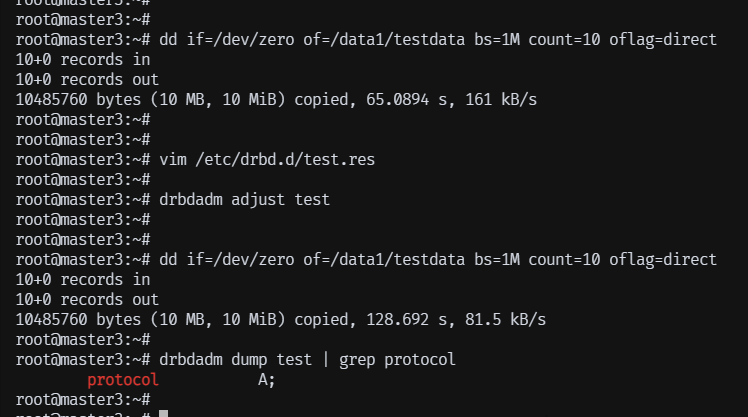
限速至 0.5Mbps (1%):



drbd 同步阻塞：



异步复制协议（C）与同步复制协议（A）对比：



## 七、ES 主备同步功能验证

验证 ElasticSearch 服务基于 drbd 跨集群数据同步能力。

### 7.1 主集群正常关机，主备切换验证

#### 7.1.1 测试目标

验证主集群正常关机时，主备切换能力，以及 es 服务数据可用性。

#### 7.1.2 测试步骤

1. 确认主集群 ES 集群运行正常，可以正常读写数据，备集群 es 副本数为0，drbd 资源状态正常，数据完全同步；
2. 通过测试脚本向主集群 ES 服务每隔 0.1s 持续写入当前时间戳数据；
3. 新开 ssh 窗口，登陆到主集群 es 所在的各个节点，执行关机操作 (shutdown -h now)；
4. 登陆备集群 es 所在节点，将 drbd 切换为主角色 (drbdadm primary es0)，并挂载 hostpath 的 pv 目录(mount /dev/drbd2 /var/hpvolumes) ；
5. 将备集群 es 副本数扩容为3 (kubectl scale sts elasticsearch --replicas=3 -n common)；
6. 等待 es 服务运行，查询最新测试数据，与第二步中主集群测试脚本关机时最后一次写入记录做比对；

##### 7.1.2.1 es 数据持续写入测试脚本：

#!/bin/sh

# chmod +x index\_sync.sh

# ./index\_sync.sh 2022-04-17

ES\_HOST="http://elasticsearch.common.svc.cluster.local:9200"

#ES\_HOST=$1

echo "es hosts is: $ES\_HOST"

genIndex() {

    echo "start to create index"

    if curl -s "${ES\_HOST}/\_cat/indices|grep testindex" >/dev/null; then

        curl -s -X DELETE ${ES\_HOST}/testindex >/dev/null

        echo "delete old index"

    fi

    curl -s -X PUT ${ES\_HOST}/testindex >/dev/null

    echo "create index ok"

}

writeData() {

    echo "start to write data"

    while true; do

        local ts=$(date +%s.%N | cut -b 1-15)

        local code=$(curl -s -o /dev/null -w %{http\_code} -H "Content-Type: application/json" -X POST -d "{\"ts\":$ts}" "${ES\_HOST}/testindex/\_doc/$ts")

        echo "written $ts : $code"

        sleep 0.1

    done

}

genIndex

writeData

##### 7.1.2.2 查询 es 最新一条数据：

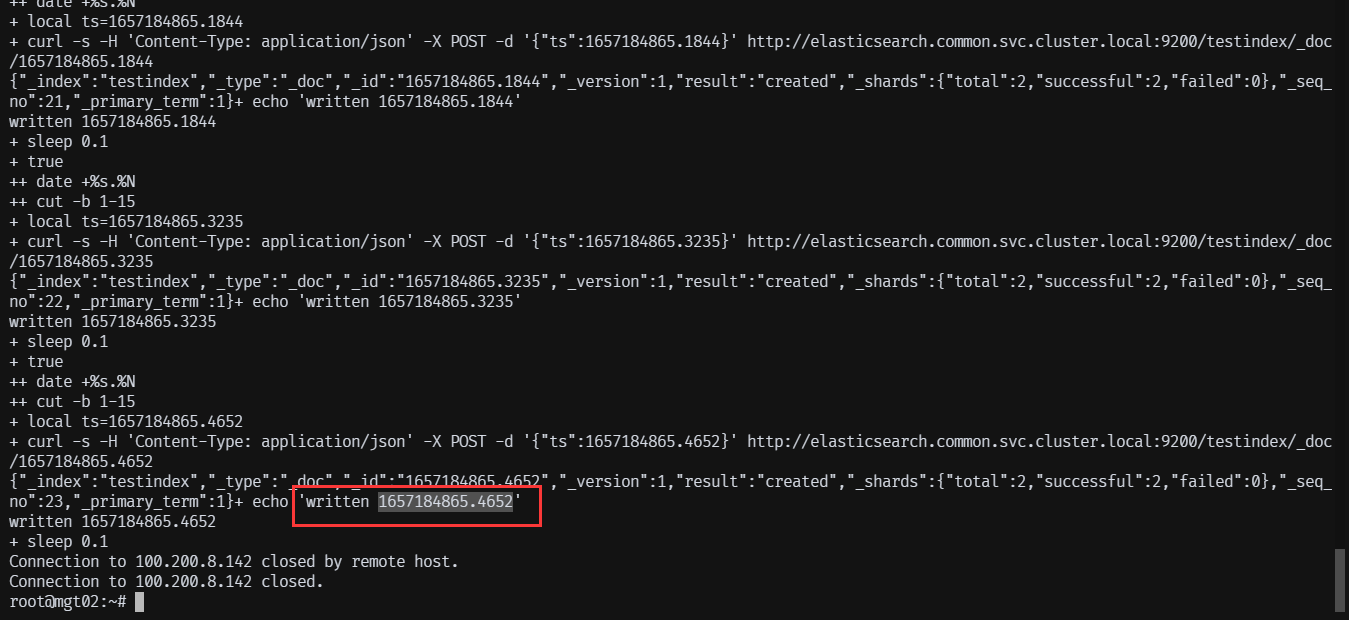
curl -H "Content-Type: application/json" -X GET -d '{"size":1,"sort":[{"\_id":{"order":"desc"}}]}' "http://elasticsearch.common.svc.cluster.local:9200/testindex/\_search?pretty"

#### 7.1.3 测试结果

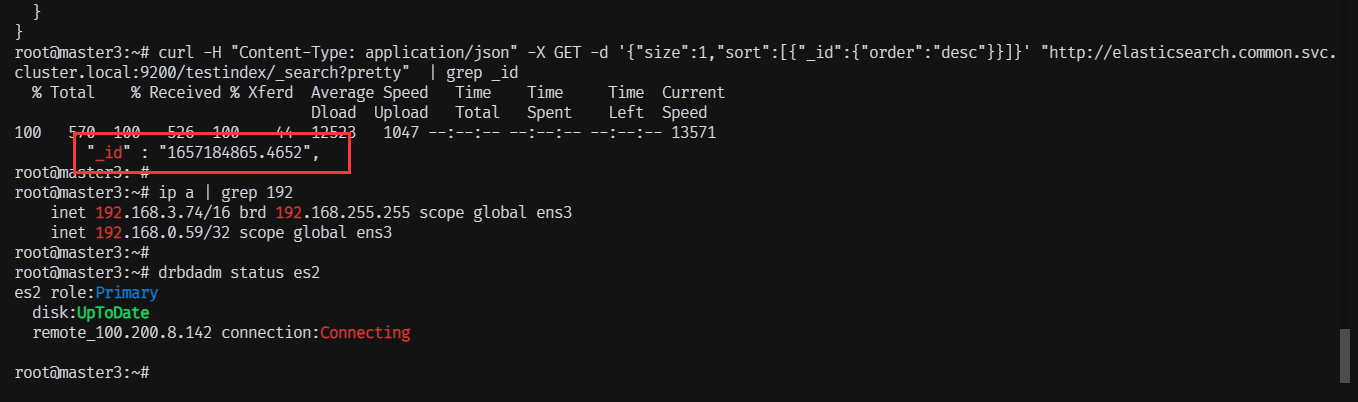
1. 正常关机，ES 数据可正常同步，主备集群切换后，备集群访问正常，能查询到最新数据；
2. 主备切换全程手动操作，主备切换全过程时间可在 10 分钟之内完成（包含备集群服务达到可用时间）；

#### 7.1.4 测试过程截图

主集群停机时，脚本程序最后写入数据截图



备集群切换 drbd 到主角色，挂载目录并扩容 es 等待 es 就绪后，查询最新数据



### 7.2 备集群临时变为主集群后主备角色回切验证

#### 7.2.1 测试目标

验证在主集群故障，备集群临时切换为主集群角色，当原主集群恢复时，主备角色回切方案。

#### 7.2.2 测试步骤

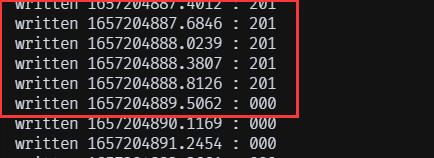
1. 基于 7.1 测试后集群状态进行验证，目前备集群为主角色， es 服务可用，读写正常；主集群节点为关机状态；
2. 备集群使用 7.1 中的测试脚本对 es 进行持续写入操作；
3. 主集群节点开机；
4. 将主集群 es 副本书缩容为 0；
5. 解除主集群 pv 数据目录挂载（umount /var/hpvolume）;
6. 将主集群各个节点 drbd 资源置为备角色（drbdadm secondary es0）；
7. 确认主备集群网络正常后，连接主备集群 drbd 资源，（drbdadm connect es0）；
8. 若超过 30s drbdadm status 状态仍未连接，检查内核日志是否有脑裂现象；如出现脑裂，需要确认是否放弃故障节点数据，以备集群为主，可执行 drbdadm disconnect es0;drbdadm connect es0 --discard-my-data 来重建连接；
9. 等待 drbd 三个节点资源建立连接，并确认数据完成同步（drbdadm status）；
10. 数据完成同步后，进入备集群，缩容 es 副本数为 0；
11. 备集群 es 缩容完成后，解除各节点 pv 数据目录挂载，并将 drbd 资源置为备角色（drbdadm secondary es0）；
12. 切换主集群所有节点的 drbd 为 primary 角色（drbdadm primary es0）；
13. 在主集群挂载 pv 目录到 drbd 盘；
14. 扩容主集群 es 为 3 副本；
15. 待 es 就绪后，查询数据，与第2步脚本最后一次成功写入的数据进行对比；

#### 7.2.3 测试结论

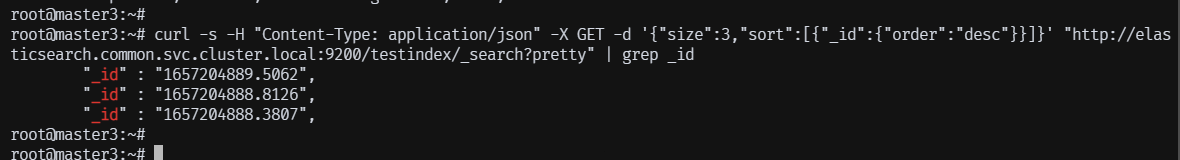
1. 主备集群回切方案验证通过，数据无丢失；
2. 全过程手动切换，除去虚机开机启动时间（不可控），全过程耗时基本可以做到10分钟之内；

#### 7.2.4 测试过程截图

备集群最后写入数据：



主集群恢复后读取数据：



## 八、问题记录

1. ICE 集群默认 hostname 与 cke node 不一致，drbd 现版本需要节点名称和 k8s 节点/ ckenode 名称一致，因此需要手动修改 ICE 集群 hostname 与 CKE 集群节点名称一致；
2. es-tenant 三个实例需要落到三个节点上，基于 node 做强制反亲和 （ICE 集群默认没有物理机反亲和标签，需要在集群节点添加 label physical\_machine\_ip=<node ip>）；
3. 目前使用的是 hostpath-provsioner 的动态 pv / pvc 做 etcd/es pod 存储方案，如果将 /var/hpvolumes 作为直接挂载目录，将无法直接对接两集群数据，因为在 /var/hpvolumes 里生成的 pv 目录是带有 uuid 的目录；解决方案： 基于主集群 pvc 和 pv yaml元数据导出在备集群克隆重建，确保 pv 目录一致；
4. drbd 资源进行首次同步，可能做全盘数据同步（而不是实际存储文件数据量），实际耗时取决于最大可用带宽和数据盘大小；
5. 当意外产生 drbd 脑裂时，即使已知数据冲突很小，当手动丢弃某一方数据进行重新同步时，需要重新同步的数据量可能很大；
6. 主集群 drbd 资源默认需要配置为默认主角色，即通常节点重启，应当自动设置为主角色，否则在单节点计划外重启场景下（无需切换主备），重启后若不自动恢复主角色，业务数据将无法正常读写；这种场景下，当需要主备切换时，备集群已经切换为主后，若原主集群已自动从故障中恢复启动，此时主集群业务以及 drbd 资源按照设定也会自动恢复，此时即产生双主问题，会产生数据脑裂现象。需要人工及时介入，停止主集群的业务，drbd 降级为备角色，并丢弃 drbd 主集群数据。待时机合适后再执行回切方案。
7. 在本例测试环境中，备集群所有流量全部通过单个 EIP 进出，即共享 EIP 带宽，三个节点 drbd 同时同步时受共享带宽限制，三节点同步速率总和为约等于可用 EIP 带宽；
8. ICE 集群租户侧集群节点 DNS 默认无 coredns 配置，无法解析 cluster svc 域名，测试时需要手动配置；

## 九、术语说明

### 9.1 DRBD 复制协议

DRBD支持三种不同的复制模式，允许三种程度的复制同步性。

* **Protocol A异步复制协议。**一旦本地磁盘写入完成，并且复制数据包已放置在本地TCP发送缓冲区中，则认为主节点上的本地写入操作已完成。在强制故障转移的情况下，可能会发生数据丢失。备用节点上的数据在故障转移后是一致的；但是，在崩溃之前执行的最新更新可能会丢失。Protocol A最常用于远程复制场景。当与DRBDProxy结合使用时，它是一个有效的灾难恢复解决方案。有关详细信息，请参见通过DRBD代理进行远程复制。
* **Protocol B内存同步（半同步）复制协议。**一旦本地磁盘写入发生，并且复制数据包已到达对等节点，则认为主节点上的本地写入操作已完成。通常，在强制故障转移的情况下不会丢失任何写操作。但是，如果同时在主备节点 同时 发生了电源故障，则主节点的数据存储可能会丢失最新完成的写入操作。
* **Protocol C同步复制协议。**只有在确认本地和远程磁盘写入之后，主节点上的本地写入操作才被视为完成。因此，单个节点的丢失不会导致任何数据丢失。数据丢失当然是不可避免的，即使采用这个复制协议，如果所有节点（例如:它们的存储子系统）同时遭到不可逆转的破坏, 数据也可能丢失。

### 9.2 单主、双主模式

#### 9.2.1 单一主(Single-primary)模式

在单一主模式下，资源在任何给定的时间仅在一个群集成员上处于主角色。由于可以保证在任何时候只有一个集群节点操作数据，因此这种模式可以用于任何传统的文件系统（ext3、ext4、XFS等）。在单一主模式下部署DRBD是高可用性（支持故障转移）集群的规范方法。

#### 9.2.2 双主(Dual-primary)模式

在双主模式下，资源在任何给定时间都处于两个群集节点上的主角色 。由于可以并发访问数据，因此此模式需要依赖使用分布式锁管理器的共享群集文件系统。示例包括GFS和OCFS2。在双主模式下部署DRBD是负载均衡集群的首选方法，这种集群需要从两个节点并发访问数据，例如需要实时迁移的虚拟化环境。此模式在默认情况下是禁用的，并且必须在DRBD的配置文件中显式启用。

### 9.3 脑裂

脑裂是指由于群集节点之间的所有网络链路的暂时故障，以及可能由于群集管理软件的干预或人为错误，两个节点在断开连接时都切换到主要角色的情况。这是一种潜在的有害状态，因为它意味着对数据的修改可能是在任一节点上进行的，而没有复制到对等节点。因此，在这种情况下，很可能已经创建了两个不同的数据集，这些数据集不能简单地合并。

### 9.4 DRBD 元数据

DRBD存储关于它保存在一个专用区域中的数据的各种信息。此元数据包括：

* DRBD设备的大小，
* 生成标识符（GI），
* 活动日志（AL）。
* 快速同步位图（详见快速同步位图）

此元数据可以存储在内部或外部。所使用的方法是基于每个资源可配置的。

#### 9.4.1 内部元数据

将资源配置为使用内部元数据意味着DRBD将其元数据存储在与实际生产数据相同的物理低级设备上。它通过在设备的末端留出一个区域来存储元数据。

**优势**

由于元数据与实际数据有着千丝万缕的联系，因此在硬盘出现故障时，管理员无需执行任何特殊操作。元数据与实际数据一起丢失，也一起还原。

**劣势**

如果较低级别的设备是单个物理硬盘（而不是RAID集），内部元数据可能会对写入吞吐量产生负面影响。应用程序的写请求的性能可能触发DRBD中的元数据的更新。如果元数据存储在硬盘的同一个磁盘上，则写入操作可能导致硬盘的写/读磁头的两个额外移动。

#### 9.4.2 外部元数据

外部元数据外部元数据简单地存储在一个独立的专用块设备上，与保存生产数据的设备不同。

**优势**

对于某些写操作，使用外部元数据会产生稍有改进的延迟行为。

**劣势**

元数据与实际生产数据并非密不可分。这意味着，在硬件故障仅破坏生产数据（而非DRBD元数据）的情况下，需要手动干预，以实现从幸存节点到随后更换的磁盘的完全数据同步。

### 9.5 连接状态信息

资源可能具有以下连接状态之一：

* **StandAlone** 没有可用的网络配置。资源尚未连接，或者已被管理性断开（使用 drbdadm disconnect ），或者由于身份验证失败或脑裂而断开其连接。
* **Disconnecting** 断开连接期间的临时状态。下一个状态是 StandAlone。
* **Unconnected** 临时状态，在尝试连接之前。可能的下一个状态： Connecting。
* **Timeout** 与对等方通信超时后的临时状态。下一个状态：Unconnected。
* **BrokenPipe** 与对等方的连接丢失后的临时状态。下一个状态： Unconnected 。
* **NetworkFailure** 与伙伴的连接丢失后的临时状态。下一个状态： Unconnected。
* **ProtocolError** 与伙伴的连接丢失后的临时状态。下一个状态：Unconnected。
* **TearDown** 临时状态。对等方正在关闭连接。下一个状态： Unconnected。
* **Connecting** 此节点正在等待，直到对等节点在网络上变为可见。
* **Connected** 已建立DRBD连接，数据镜像现在处于活动状态。这是正常状态。

### 9.6 资源角色

资源角色可以通过调用 drbdadm role 命令来观察资源的角色。您可以看到以下资源角色之一：

* **Primary** 资源当前处于主角色中，可以读取和写入。此角色仅在两个节点中的一个节点上发生，除非启用了dual primary mode
* **Secondary** 资源当前处于辅助角色中。它通常从其对等方接收更新（除非在断开连接模式下运行），但既不能读取也不能写入。此角色可能出现在一个或两个节点上。
* **Unknown** 资源的角色当前未知。本地资源角色从未具有此状态。它仅为对等方的资源角色显示，并且仅在断开连接模式下显示。

### 9.7 磁盘状态

磁盘状态可以通过调用 drbdadm dstate 命令来观察资源的磁盘状态。磁盘状态可以是以下之一：

* **Diskless** 没有为DRBD驱动程序分配本地块设备。这可能意味着资源从未连接到其备份设备，它已使用 drbdadmdetach 手动分离，或者在发生较低级别的I/O错误后自动分离。
* **Attaching** 读取元数据时的临时状态。
* **Detaching** 在分离并等待正在进行的IOs完成时的临时状态。
* **Failed** 本地块设备报告I/O失败后的瞬态。下一个状态：Diskless。
* **Negotiating** 在 已经Connected 的DRBD设备上执行附加操作时的瞬态。
* **Inconsistent** 数据不一致。在两个节点上（在初始完全同步之前）创建新资源时立即出现此状态。此外，在同步期间，在一个节点（同步目标）中可以找到此状态。
* **Outdated** 资源数据一致，但标记为过时。
* **DUnknown** 如果没有可用的网络连接，则此状态用于对等磁盘。
* **Consistent** 没有连接的节点的一致数据。建立连接后，决定数据是 UpToDate 还是 Outdated。
* **UpToDate** 数据的一致、最新状态。这是正常状态。

### 9.8 连接信息

连接信息数据

* **local** 显示网络协议栈、用于接受来自对等方的连接的本地地址和端口。
* **peer** 显示网络协议栈、对等方节点地址和用于连接的端口。
* **congested** 此标志指示数据连接的 TCP 发送缓冲区是否已填充 80% 以上。

## 十、参考资料

1. DRBD 9 user manual: <https://linbit.com/drbd-user-guide/drbd-guide-9_0-en/>