УТВЕРЖДЕН
ru.milofon.00001-01 34 01-ЛУ

РУКОВОДСТВО ПО ВОССТАНОВЛЕНИЮ РАБОТЫ ОТКАЗОУСТОЙЧИВОГО КЛАСТЕРА

Руководство администратора

ru.milofon.00001-01 34 01

Листов 26

Инв. № подл.		Подп. и дата		Взам. инв. №		Инв. № дубл.		Подп. и дата	

АННОТАЦИЯ

Настоящий документ содержит информацию, предназначенную для администраторов отказоустойчивого кластера, по решению различных проблем и ошибочных ситуаций, возможных в процессе эксплуатации отказоустойчивого кластера. Документ не рассматривает все возможные ошибки, которые могут возникнуть в процессе эксплуатации, но дает представление об общем алгоритме восстановительных работ в случае ошибок и нештатных ситуаций для приведения кластера в рабочее состояние.

СОДЕРЖАНИЕ

1. КОНФИГУРАЦИЯ КЛАСТЕРА

1.1. Схема кластера

В настоящем документе описывается схема кластера состоящего из двух узлов: основной узел и резервный. Оба узла представляют собой виртуальные машины Vmware (в общем случае это могут быть не виртуальные машины или виртуальные на другой платформе виртуализации).

На каждом узле настроены:

1) Сервер приложений (tomcat)
2) СУБД Ред База данных
3) Компоненты кластера (corosync, pacemaker, pcs - командная утилита для управления pacemaker, настройки и тп...)

1.2. Основной узел

На основном узле настроены:

- Основная БД в режиме репликации

1.3. Резервный узел

На резервном узле настроены:

- Резервная БД в режиме репликации

2. ШТАТНЫЙ РЕЖИМ РАБОТЫ

2.1. Настройки и статус кластера

2.1.1. Посмотреть текущий статус кластера возможно с любого (рабочего) узла кластера командой:

```
> pcs status

Cluster name: rosatom
Stack: corosync
Current DC: gkrs-s-tsmev1n3 (version 1.1.18-11.el7-2b07d5c5a9) - partition with quorum
Last updated: Thu Nov 15 15:23:38 2018
Last change: Thu Nov 15 15:18:40 2018 by root via cibadmin on gkrs-s-tsmev1n3

2 nodes configured
6 resources configured

Online: [ gkrs-s-tsmev1n3 gkrs-s-tsmev1n4 ]

Full list of resources:

 virtual_ip  (ocf::heartbeat:IPaddr2):  Started gkrs-s-tsmev1n3
 vmfence_gkrs-s-tsmev1n4  (stonith:fence_vmware_soap):  Started gkrs-s-tsmev1n3
 vmfence_gkrs-s-tsmev1n3  (stonith:fence_vmware_soap):  Started gkrs-s-tsmev1n4
 Master/Slave Set: rdb-master [rdb]
     Masters: [ gkrs-s-tsmev1n3 ]
     Slaves: [ gkrs-s-tsmev1n4 ]
 tomcat7  (ocf::heartbeat:tomcat):  Started gkrs-s-tsmev1n3

Daemon Status:
  corosync: active/enabled
  pacemaker: active/enabled
  pcsd: active/enabled
```

2.1.2. Список «Online» отображает список узлов доступных в кластере. В примере выше это два узла gkrs-s-tsmev1n3 и gkrs-s-tsmev1n4.

В случае если один из узлов выйдет из строя, то статус изменится на примерно такой вид:

```
...
Online: [ gkrs-s-tsmev1n4 ]
OFFLINE: [ gkrs-s-tsmev1n3 ]
...
```

В данном случае узел gkrs-s-tsmev1n4 находится в списке «OFFLINE», который отображает список недоступных узлов.

2.1.3. Список ресурсов кластера отображается в списке «Full list of resources», в нашем примере это ресурсы virtual_ip, vmfence_gkrs-s-tsmev1n4, vmfence_gkrs-s-tsmev1n3, rdb, tomcat7. Также в списке «Daemon Status» отображается информация со статусами сервисов кластера. В штатном режиме работы это список:

```
Daemon Status:
  corosync: active/enabled
  pacemaker: active/enabled
  pcsd: active/enabled
```

2.1.4. Чтобы определить основной узел системы нужно вызывать команду:

```
> pcs resource
...
Master/Slave Set: rdb-master [rdb]
     Masters: [ gkrs-s-tsmev1n3 ]
     Slaves: [ gkrs-s-tsmev1n4 ]
...
```

Запись *Masters: [gkrs-s-tsmev1n3]* указывает что узел gkrs-s-tsmev1n3 в текущий момент является основным.

2.2. Логи кластера

Состояние кластера непрерывно выводится в лог-файл `/var/log/cluster/corosync.log`.

Посмотреть логи кластера возможно в режиме реального времени командой:

```
> tail -f /var/log/cluster/corosync.log
...
```

Примечание. Логи различаются для основного и второстепенного узлов.

2.3. Логи на основном узле

Ниже приведена примерная картина логов и характерные признаки рабочего состояния основного узла кластера:

```
> tail -f -n 10 /var/log/cluster/corosync.log
__main__.py(rdb)[11574]:  %(levelname)s: Monitor: rdb (master), RDB: ROLE.MASTER
__main__.py(rdb)[11644]:  %(levelname)s: Monitor: rdb (master), RDB: ROLE.MASTER
__main__.py(rdb)[11702]:  %(levelname)s: Monitor: rdb (master), RDB: ROLE.MASTER
...
```

В данном случае ключевое значение играет выполняемое в режиме реального времени получение роли БД в схеме репликации БД и отображение результата в логе *RDB: ROLE.MASTER*. Что означает что текущая БД на данном компьютере (узле) является мастер-базой и с нее происходит копия в БД реплику, которая находится на резервном узле.

2.4. Логи на резервном узле

Ниже приведена примерная картина логов и характерные признаки рабочего состояния резервного узла кластера:

```
> tail -f -n 10 /var/log/cluster/corosync.log
__main__.py(rdb)[10976]:  %(levelname)s: Monitor: rdb (slave), RDB: ROLE.SLAVE
__main__.py(rdb)[10986]:  %(levelname)s: Monitor: rdb (slave), RDB: ROLE.SLAVE
__main__.py(rdb)[10996]:  %(levelname)s: Monitor: rdb (slave), RDB: ROLE.SLAVE
...
```

В данном примере статус БД - *RDB: ROLE.SLAVE*, что означает что БД находится в режиме реплики.

2.5. Виртуальный IP-Адрес

2.5.1. Среди прочих ресурсов кластера, в примере выше, был указал virtual_ip. Это виртуальный IP-адрес - специальный ресурс кластера, представляет собой дополнительный IP-адрес, по которому доступны ресурсы кластера. Кластер сконфигурирован таким образом, чтобы этот ресурс принадлежал всегда основному узлу. В случае нештатной ситуации в работе кластера, если будет происходить миграция кластера на резервный, резервному узлу будет назначен дополнительный фиксированный IP-адрес, ровно такой, какой назначен в штатной ситуации основному узлу.

2.5.2. Диагностируется штатное состояние этого ресурса командой:

```
> ip a
```

При этом выводятся все сконфигурированные сетевые интерфейсы узла и значение этого IP-адреса:

```
2: ens160: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP qlen 1000
    link/ether 00:50:56:bc:f1:71 brd ff:ff:ff:ff:ff:ff
    inet 93.x8x.xx.x4/25 brd 93.x8x.xx.127 scope global ens160
       valid_lft forever preferred_lft forever
    inet 93.x8x.xx.x6/32 brd 93.x8x.xx.127 scope global ens160
       valid_lft forever preferred_lft forever
```

В данном случае IP-адрес 93.x8x.xx.x6 является статическим дополнительным IP-адресом кластера, помимо основного 93.x8x.xx.x4.

2.5.3. В кластере статус ресурса проверяется командой:

```
> pcs status resources
...
virtual_ip  (ocf::heartbeat:IPaddr2):  Started gkrs-s-tsmev1n3
...
```

Что означает, что ресурс работает на узле gkrs-s-tsmev1n3, в данный момент основной узел.

2.5.4. Чтобы посмотреть настройки ресурса используется команда:

```
> pcs status resources virtual_ip
```

или

```
> pcs resource show virtual_ip
...
Attributes: cidr_netmask=32 ip=93.x8x.xx.x6
...
```

2.6. Остальные ресурсы

В штатном режиме работы работают корректно все остальные ресурсы системы.

3. АВАРИЙНЫЙ РЕЖИМ РАБОТЫ

3.1. Аварийный режим работы

Основную работу по предоставлению ресурсов кластера выполняет основной узел. В случае любой нештатной ситуации (ошибки) работа переключается в аварийный режим работы кластера, при этом:

1) В процессе миграции происходит переключение работы кластера на один узел. Начинает работать или основной узел (в случае выхода из строя резервного) или резервный.

2) Узел с ошибкой попадает в бан.

3) БД переключается в режим асинхронной репликации.

4) В случае выхода из строя основного узла, сервер приложений запускается на резервном узле.

5) В случае выхода из строя основного узла, динамический IP-Адрес переключается на резервный узел.

6) В некоторых случаях отрабатывает команда stonith кластера, которая вызывает перезагрузку нерабочего узла кластера.

3.2. Ручной переход в аварийный режим работы

3.2.1. В некоторых случаях может произойти нештатная ситуация, в которой может потребоваться вручную перейти в режим аварийной работы. Для этого принудительно выключаем узел из кластера командой:

```
> pcs cluster stop [node-name]
```

Соответственно, если команда вызывается на одном из узлов без указания имени узла - то кластер останавливается на текущем узле, где выполняется команда. Также возможна остановка узла с другого узла по имени.

3.2.2. После установки аварийного режима работы, необходимо проверить доступность ресурсов кластера и при первой же возможности восстановить штатный режим работы кластера с двумя (или более) узлами.

3.3. Диагностика аварийного режима работы

3.3.1. Аварийный режим можно диагностировать проверкой статуса кластера:

```
> pcs status
cluster name: rosatom
```

```
...
Online: [ gkrs-s-tsmev1n4 ]
OFFLINE: [ gkrs-s-tsmev1n3 ]

Full list of resources:
 virtual_ip  (ocf::heartbeat:IPaddr2):  Started gkrs-s-tsmev1n4
 Master/Slave Set: rdb-master [rdb]
     Masters: [ gkrs-s-tsmev1n4 ]
     Stopped: [ gkrs-s-tsmev1n3 ]
 tomcat7  (ocf::heartbeat:tomcat):  Started gkrs-s-tsmev1n4
...
```

В данном примере по группе «OFFLINE» мы понимаем что один узел недоступен кластеру. Также БД переключилась в асинхронный режим репликации, это видно по статусу ресурса rdb «Stopped» с указанием машины на которой произошел сбой. Кроме того сработала миграция системы и теперь основной узел это gkrs-s-tsmev1n4, все ресурсы мигрировали на данный узел.

3.3.2. Дополнительно аварийный режим можно проверить анализом логов кластера (на основном узле):

```
> tail -f /var/log/cluster/corosync.log
...
[25259] node1.clister corosyncnotice  [TOTEM ] A new membership (192.168.56.11:48156) was formed. Members
↪  left: 2
[25259] node1.clister corosyncnotice  [QUORUM] Members[1]: 1
[25259] node1.clister corosyncnotice  [MAIN  ] Completed service synchronization, ready to provide
↪  service.
...
__main__.py(rdb)[3583]:  %(levelname)s: Monitor: rdb (master), RDB: ROLE.SA
...
```

В данном случае по статусу БД *ROLE.SA* можно с уверенностью говорить о переключении БД в режим асинхронной репликации. Штатный режим репликации не работает.

3.3.3. Также признаком нештатной ситуации и аварийной работы кластера являются проблемы с доступом к любому ресурсу кластера.

Примечание. В некоторых случаях возможны такие ситуации, когда аварийный режим работы не диагностируется проверкой статуса кластера, а также логами кластера.

4. ОШИБКА СЕРВИСА СУБД НА РЕЗЕРВНОМ УЗЛЕ

4.1. Диагностика

4.1.1. В некоторых случаях возможна остановка или поломка сервиса СУБД на одном из узлов. В результате не работает репликация БД и СУБД переходит в нештатный режим работы. Не работает сохранение транзакций, существующие подключения к БД не обрываются, однако подключиться к БД при этом не удастся (соответственно новые пользователи войти в систему не смогут).

4.1.2. Со стороны кластера, ситуация выглядит как штатный режим работы – обычными методами диагностики аварийной работы (см подразд. 3.3) проблемы в работе кластера не обнаруживаются.

Также эта ситуация возможна в случае поломки режима репликации БД (в результате любой ошибки).

4.1.3. Статус кластера при этом:

```
> pcs resource
...
virtual_ip  (ocf::heartbeat:IPaddr2):  Started gkrs-s-tsmev1n4
 tomcat7  (ocf::heartbeat:tomcat):  Started gkrs-s-tsmev1n4
 Master/Slave Set: rdb-master [rdb]
     Masters: [ gkrs-s-tsmev1n4 ]
     Slaves: [ gkrs-s-tsmev1n3 ]
...
```

4.1.4. В логах ситуация тоже нормальная (на обоих узлах):

```
gkrs-s-tsmev1n4> tail -f /var/log/cluster/corosync.log
  __main__.py(rdb)[12955]:  %(levelname)s: Monitor: rdb (master), RDB: ROLE.MASTER
__main__.py(rdb)[12968]:  %(levelname)s: Monitor: rdb (master), RDB: ROLE.MASTER
__main__.py(rdb)[13026]:  %(levelname)s: Monitor: rdb (master), RDB: ROLE.MASTER
...

gkrs-s-tsmev1n3> tail -f /var/log/cluster/corosync.log
__main__.py(rdb)[22285]:  %(levelname)s: Monitor: rdb (slave), RDB: ROLE.SLAVE
__main__.py(rdb)[22295]:  %(levelname)s: Monitor: rdb (slave), RDB: ROLE.SLAVE
__main__.py(rdb)[22305]:  %(levelname)s: Monitor: rdb (slave), RDB: ROLE.SLAVE
...
```

4.2. Ремонт

В общем случае необходимо для восстановления системы:

4.2.1. проверить доступность сервиса Ред Базы Данных:

```
gkrs-s-tsmev1n4> ps -aux | grep firebird
firebird  9660  0.0  0.0  32232   468 ?        S    11:48   0:00 /opt/RedDatabase/bin/rdbguard
↪  -guardpidfile /var/run/firebird/firebird.pid -daemon -forever
firebird 11950  0.0  1.6 226228  8316 ?        Sl   11:54   0:00 /opt/RedDatabase/bin/rdbserver

gkrs-s-tsmev1n3> ps -aux | grep firebird
root     23128  0.0  0.1  10676   904 pts/0    S+   12:03   0:00 grep --color=auto firebird
```

В данном случае сервис работает на основном узле, однако не работает на резервном узле.

4.2.2. Необходимо разобраться в причинах остановки сервиса, это можно сделать, например, с помощью лог-файлов `/opt/RedDatabase/firebird.log`, `/var/log/messages`.

4.2.3. В некоторых случаях, может потребоваться просто запустить сервис Ред Базы Данных:

```
gkrs-s-tsmev1n3> service firebird start
```

4.2.4. Если проблемы продолжаются, и оперативно устранить проблему не удается, необходимо принудительно остановить узел см. подразд. 3.2, на котором происходит ошибка БД. И в дальнейшем восстанавливать работу БД, как указано в разд. 6:

```
gkrs-s-tsmev1n3> pcs cluster stop
```

или

```
gkrs-s-tsmev1n4> pcs cluster stop gkrs-s-tsmev1n3
```

При этом происходит переход кластера в аварийный режим работы и появляется время для устранения ошибок БД.

4.2.5. После миграции на основной узел логи pacemaker'а на основном узле выглядят так:

```
gkrs-s-tsmev1n3> tail -f /var/log/cluster/corosync.log
[25259] node1.clister corosyncnotice  [TOTEM ] A new membership (192.168.56.11:48156) was formed. Members
↪  left: 2
[25259] node1.clister corosyncnotice  [QUORUM] Members[1]: 1
[25259] node1.clister corosyncnotice  [MAIN  ] Completed service synchronization, ready to provide
↪  service.
...
__main__.py(rdb)[3583]:  %(levelname)s: Monitor: rdb (master), RDB: ROLE.SA
```

То есть БД переключилась в режим «Standalone» и способна к дальнейшей работе. При этом репликация, как было указано в подразд. 3.1, переключается в асинхронный режим и см. разд. 6 как восстановить БД.

Примечание. Здесь нужно заметить, что соединения, ранее установленные к БД, после восстановления репликации БД, могут не заработать и может потребоваться выйти из системы и войти в систему снова.

5. ОШИБКА СЕРВИСА СУБД НА ОСНОВНОМ УЗЛЕ

5.1. Ситуация по большому случаю аналогична ситуации нештатной работе сервиса СУБД на резервном узле см. разд. 4.

5.2. В случае невозможности быстро восстановить работу сервиса, необходимо отключить проблемный узел (в данном случае основной) от кластера, аналогично тому как отключается резервный см. п. 4.2.4, далее продолжать работу по восстановлению согласно разд. 6.

6. ВОССТАНОВЛЕНИЕ ШТАТНОГО РЕЖИМА РАБОТЫ КЛАСТЕРА

Данная глава посвящена проблемам восстановления штатного режима работы кластера и решение последствий миграции кластера на один узел, а также устранению последствий работы в аварийном режиме.

6.1. Анализ ситуации

6.1.1. Произошла миграция в кластере и работает только один узел. Ситуация диагностируется следующей командой:

```
> pcs status
virtual_ip  (ocf::heartbeat:IPaddr2):  Started gkrs-s-tsmev1n4
 tomcat7  (ocf::heartbeat:tomcat):  Started gkrs-s-tsmev1n4
 Master/Slave Set: rdb-master [rdb]
     Masters: [ gkrs-s-tsmev1n4 ]
     Stopped: [ gkrs-s-tsmev1n3 ]
```

Резервный узел не работает в кластере, на это указывает группа «Stopped».

6.1.2. Произошла миграция на основной/резервный узел. БД работает в режиме «Standalone», что диагностируется логами pacemaker'а на проблемном узле:

```
> tail -f /var/log/cluster/corosync.log
__main__.py(rdb)[22183]:  %(levelname)s: Monitor: rdb (master), RDB: ROLE.SA
__main__.py(rdb)[22238]:  %(levelname)s: Monitor: rdb (master), RDB: ROLE.SA
__main__.py(rdb)[22249]:  %(levelname)s: Monitor: rdb (master), RDB: ROLE.SA
__main__.py(rdb)[22305]:  %(levelname)s: Monitor: rdb (master), RDB: ROLE.SA
...
```

В логах отображается статус работы ресурса RDB как ROLE.SA.

6.1.3. Сменились настройки репликации БД:

```
> cat /opt/RedDatabase/replication.conf
database = "/opt/DB/shuffle-gate.fdb"
{
  buffer_size = 1048576
  disable_on_error = false
  compress_records = false
  master_priority = false
  exclude_without_pk = false
  log_directory = "/opt/DB/logs"
  log_file_prefix = "rdb"
  log_segment_size = 16777216
  log_segment_count = 8
  log_archive_directory = "/opt/DB/arch"
  log_archive_command = "test ! -f $(archpathname) && cp $(logpathname) $(archpathname)"
  log_archive_timeout = 60
  log_group_flush_delay = 0
}
```

В текущий момент БД переключена в режим асинхронной репликации.

6.1.4. В асинхронном режиме, БД продолжает работать в режиме репликации, однако все изменения фиксируются в файлы на диске. Согласно настройкам, изменения сохраняются в папки:

/opt/DB/logs

/opt/DB/arch

Примечание. Поскольку в системе постоянно отрабатывают различные фоновые службы, транзакции закрываются и открываются, происходит удаление/добавление/изменение строк, то в данном режиме ожидать ситуации завершения процесса создания логов (вроде как оптимального времени для ремонта кластера) с репликации нецелесообразно.

БД необходимо остановить.

6.2. Приостановка работы кластера

Перед дальнейшим ремонтом БД целесообразно отключить ресурсы кластера связанные с БД, чтобы избежать дополнительных ошибок.

1. Отключить ресурс RDB в кластере:

```
> pcs resource disable rdb
```

2. Проверяем что отключение состоялось:

```
> pcs status
...
Full list of resources:

 virtual_ip  (ocf::heartbeat:IPaddr2):  Started gkrs-s-tsmev1n4
 tomcat7  (ocf::heartbeat:tomcat):  Started gkrs-s-tsmev1n4
 Master/Slave Set: rdb-master [rdb]
     Stopped (disabled): [ gkrs-s-tsmev1n3 gkrs-s-tsmev1n4 ]
...
```

Группа «Stopped» отображает отключенные ресурсы, в данном случае весь ресурс RDB.

6.3. Ремонт реплики

6.3.1. Для предотвращения в дальнешем обращения к БД (и возможного ее изменения). Блокируем мастер-БД для использования:

```
> /opt/RedDatabase/bin/gfix -shut full -force 30 /opt/DB/shuffle-gate.fdb
```

6.3.2. При этом может возникнуть ситуация блокировки БД (в том случае если в текущий момент происходит работа с БД):

```
I/O error during "lock" operation for file "/opt/DB/shuffle-gate.fdb"
-Database already opened with engine instance, incompatible with current
```

В этом случае, можно завершить процессы использующие файл БД:

```
fuser -s -k -SIGKILL /opt/DB/shuffle-gate.fdb
```

6.3.3. п. 6.3.1 можно заменить полной остановкой сервиса СУБД:

```
service firebird stop
```

Примечание. Это имеет смысл делать только в том случае, если в кластере обслуживается единственная БД. Для дополнительного узла это целесообразно делать всегда, поскольку он выключен из работы кластера.

6.3.4. После остановки доступа к БД в п. 6.3.1 – п. 6.3.3, создание файлов логов данных в режиме асинхронной репликации прекратится. Сами файлы логов могут быть найдены в папках `/opt/DB/logs` и `/opt/DB/arch` (настройки куда сохраняются репликационные логи указаны в файле `/opt/RedDatabase/replication.conf`), см пример в подразд. 6.1.

```
> ls /opt/DB/logs
rdb.log-000
> ls /opt/DB/arch/
rdb.arch-000000001  rdb.arch-000000002  rdb.arch-000000003  rdb.arch-000000004
```

6.3.5. Состояние данных асинхронной репликации можно получить командой (на основном узле):

```
> /opt/RedDatabase/bin/fblogmgr -D /opt/DB/shuffle-gate.fdb -U sysdba -P masterkey
  Log status:
  Current sequence: 5
  Last modified: 2018-11-15 10:55:28
  Active segment: rdb.log-000, size: 180
  Total log size: 180 bytes in 1 segments
  Free segments: 0, full segments: 0, archived segments: 0
```

Примечание. Это возможно сделать только на включенной БД (соотв. если ранее использовалась команда gfix –shut из п. 6.3.1, необходимо сделать копию БД настроить файл replication.conf, как в примере выше). Это неактуально в том случае, если просто был остановлен сервис Ред Базы Данных командой из п. 6.3.3.

6.4. Восстановление реплики БД (логи)

1. Далее нужно сделать принудительное архивирование транзакций репликации командой:

```
> /opt/RedDatabase/bin/fblogmgr -A all -F -D /opt/DB/shuffle-gate.fdb -U sysdba -P masterkey
 No suitable segment found for archiving
```

2. БД второстепенная должна быть настроена в режиме реплики (если это не так, то необходимо настроить БД реплики таким образом):

```
> cat /opt/RedDatabase/replication.conf
replica = "/opt/DB/shuffle-gate.fdb"
{
  log_directory = "/opt/DB/slave_logs"
  master_database = "sysdba:masterkey@gkrs-s-tsmev1n4: /opt/DB/shuffle-gate.fdb"
}
```

Папка для репликационных логов в данном случае `/opt/DB/slave_logs`.

Примечание. Если папка отсуствует ее необходимо создать в системе.

3. Копируем файлы репликации на второй узел командой и в папку указанную в настройках реплики см. подп. 2:

```
> scp /opt/DB/arch/* gkrs-s-tsmev1n3:/opt/DB/slave_logs
  rdb.arch-000000001        100%  205    20.4KB/s   00:00
  rdb.arch-000000002        100% 3238   703.0KB/s   00:00
  ...
```

4. Теперь необходимо применить скопированные логи на репликационную БД:

```
> /opt/RedDatabase/bin/fbreplmgr -V -A /opt/DB/shuffle-gate.fdb
Connect to the slave database and apply logs... OK
```

5. Дополнительно можно проверить статус репликации:

```
> /opt/RedDatabase/bin/fbrepldiff -D /tmp/shuffle-gate.fdb -R /opt/DB/shuffle-gate.fdb -U masterkey -P
↪  password
  Summary: no differences found
```

БД мастера при этом предварительно скопирована в папку /tmp.

6.5. Восстановление реплики (копирование)

6.5.1. Альтернативный вариант восстановления реплики возможен просто копированием и заменой БД реплики БД мастером. Файлы логов при этом варианте удаляются. Обе БД должны быть остановлены согл. п. 6.3.1 или сервис Ред Базы Данных на обоих машинах должен быть остановлен согл. п. 6.3.3.

6.5.2. Еще один альтернативный вариант - создать копию БД мастера с помощью утилиты `gbak` из комплекта Ред База Данных.

6.5.3. Рассмотрим вариант п. 6.5.1 подробнее. Копируем БД мастера поверх БД реплики:

```
scp /opt/DB/shuffle-gate.fdb gkrs-s-tsmev1n3:/opt/DB/shuffle-gate.fdb
```

Примечание. Внимание. Будьте особенно внимательны в таком случае и не удалите случайно БД мастера, перепутав ее с БД реплики. Поскольку БД реплика находилась в нерабочем режиме и не содержит часть данных, которые есть у БД мастера. Вы можете потерять часть данных.

6.5.4. Удаляем логи БД на основном узле - они не нужны:

```
> rm -f /opt/DB/logs/*
> rm -f /opt/DB/arch/*
```

6.6. Включение реплики

Данный пункт применим только в том случае, если использовался метод копирования мастер БД см. подразд. 6.5. Также этот вариант работ рекомендован в тех случаях, когда необходимо восстановить репликационный механизм БД.

6.6.1. Включаем режим реплики на копии мастер БД.

Поскольку реплика представляет собой теперь копию БД мастера, мы также потеряли настройки реплики в непосредственно самой БД. Для включения режима реплики необходимо определить GUID мастер БД:

```
> /opt/RedDatabase/bin/gstat -h /opt/DB/shuffle-gate.fdb
...
Database GUID:  {31548829-7F7B-4CA3-79B7-5A716B75A7D2}
...
```

Данный GUID используется далее в настройках репликации.

Примечание. В нашем случае, после копирования, GUID мастер БД и реплики совпадают. Поэтому определять GUID можно на БД реплике.

6.6.2. Устанавливаем GUID в качестве GUID мастер БД в настройках БД реплики. Делается это командой:

```
> /opt/RedDatabase/bin/gfix -replica {31548829-7F7B-4CA3-79B7-5A716B75A7D2} -user sysdba -pass masterkey
↪  /opt/DB/shuffle-gate.fdb
```

6.6.3. Проверяем теперь статус репликационной настройки в БД реплике:

```
/opt/RedDatabase/bin/gstat -h /opt/DB/shuffle-gate.fdb
...
Variable header data:
  Database GUID:  {31548829-7F7B-4CA3-79B7-5A716B75A7D2}
  Replication master GUID:  {31548829-7F7B-4CA3-79B7-5A716B75A7D2}
  *END*
...
```

Для нас здесь важен параметр «Replication master GUID». Видим, что GUID мастер БД установлен. Теперь репликационная БД настроена для работы. Совпадение GUID мастер-БД и реплики-БД не принципиально.

Примечание. Если происходит восстановление мастер БД из реплики, то этот параметр нужно наоборот снять для БД мастера командой:

```
> /opt/RedDatabase/bin/gfix -replica {} -user sysdba -pass masterkey /opt/DB/shuffle-gate.fdb
```

6.7. Запуск репликации

Теперь когда БД подготовлены, необходимо подготовить настройки репликации и запустить репликационный режим Ред Базы Данных.

6.7.1. Для этого надо произвести настройку репликационного файла конфигурации `replication.conf` для мастер БД:

```
> cat /opt/RedDatabase/replication.conf
database = "/opt/DB/shuffle-gate.fdb"
{
    replica_database = "sysdba:masterkey@gkrs-s-tsmev1n4:/opt/DB/shuffle-gate.fdb"
  exclude_without_pk = true
}
```

6.7.2. Проверяем настройки репликации в файле конфигурации для БД реплики:

```
> cat /opt/RedDatabase/replication.conf
database = "/opt/DB/shuffle-gate.fdb"
{
}
```

6.7.3. Разблокируем мастер-БД командой (если была ранее заблокирована):

```
> /opt/RedDatabase/bin/gfix -online /opt/DB/shuffle-gate.fdb
```

6.7.4. Запускаем сервис Ред Базы Данных командой на второстепенном узле и на основном узле (если был сервис остановлен):

```
> service firebird start
```

6.8. Проверка репликации

6.8.1. Основной признак работы репликации и который быстрее всего проверяется – это что к БД мастеру возможно подключение и происходит запись в БД. То есть в случае работы основной системы, достаточно просто войти в систему и например зарегистрироваться на странице авторизации.

6.8.2. Альтернативно, возможно получить статус о работе репликации с помощью утилиты `isql` из комплекта Ред Базы Данных:

```
> isql -u sysdba /opt/DB/shuffle-gate.fdb

SQL> select MON$TYPE from MON$REPLICATION;
MON$TYPE
========
       1

SQL> quit;
```

6.8.3. В случае если репликация не работает, но мастер корректно настроен на репликацию, согласно п. 6.7.1, будет ошибка уже на этапе подключения:

```
> isql -u sysdba /opt/DB/shuffle-gate.fdb
Statement failed, SQLSTATE = 08006
Replication error
-Unable to complete network request to host "gkrs-s-tsmev1n4".
-Failed to establish a connection.
Use CONNECT or CREATE DATABASE to specify a database
```

6.9. Запуск кластера

Переходим к этапу запуска непосредственно кластера.

6.9.1. При поломке кластера, проблемный узел блокируется, о чем можно узнать из статуса блокировок:

```
> pcs constraint
 Location Constraints:
  Resource: rdb
    Disabled on: gkrs-s-tsmev1n3 (score:-INFINITY) (role: Started)
  Resource: rdb-master
    Enabled on: gkrs-s-tsmev1n4 (score:50)
...
```

В данном примере «Disabled on... score:-INFINITY» указывает на блокировку узла.

6.9.2. Для очистки блокировки используем команду:

```
> pcs resource clear rdb
```

6.9.3. Теперь проверка статуса блокировок показывает что блокировка снята.

```
> pcs constraint
 Location Constraints:
  Resource: rdb-master
    Enabled on: gkrs-s-tsmev1n4 (score:50)
 Ordering Constraints:
...
```

В примере, блокировка узла gkrs-s-tsmev1n3 снята.

6.9.4. Включаем ресурс rdb (он был ранее заблокирован перед началом ремонта см. подп. 1):

```
> pcs resource enable rdb
```

6.9.5. Запускаем кластер:

```
> pcs cluster start --all
gkrs-s-tsmev1n3: Starting Cluster...
gkrs-s-tsmev1n4: Starting Cluster...
```

6.10. Диагностика ситуации

После окончания работ по переводу режима работы кластера из аварийного состояния в штатный, следует проверить состояние кластера согласно разд. 2.

6.11. Скрипт для автоматического восстановления

6.11.1. Для нужд автоматического создания БД реплики настроен специальный bash-скрипт. Который доступен для запуска в системе:

```
> ls /opt/DB/sync.sh
/opt/DB/sync.sh
```

6.11.2. Данный скрипт реализует след. сценарий обновления БД реплики:

1) Отключение всех ресурсов кластера
2) Кластер останавливается целиком
3) БД реплики заменяется БД мастером
4) Кластер запускается

6.11.3. Алгоритм использования данного срипта может быть такой:

1) Отключение сервисов Ред Базы Данных для основного и второстепенного узла (для исключения доступа к БД):

```
> service firebird stop
```

2) Запуск скрипта:

```
> /opt/DB/sync.sh
```

3) Включение ресурсов кластера

```
> pcs resource enable tomcat7
> pcs resource enable rdb
> pcs resource enable virtual_ip
```

4) Запуск сервисов Ред Базы Данных на обоих узлах

```
> service firebird start
```

П р и м е ч а н и е. Использование данного скрипта строго ограничено ситуацией, когда мастер БД находится на машине gkrs-s-tsmev1n3. В скрипте прописаны фиксированные IP-Адреса. Рекомендуется перед использованием скрипта ознакомиться с кодом скрипта, во избежание непредвиденных ситуаций и возможных потерь данных.

7. ОТКАЗ СЕТЕВОГО ОБОРУДОВАНИЯ

7.1. При отказе сетевого оборудования сработает система защиты от «Split Brain» STONITH.

7.2. В результате один из серверов остановит свою работу. ОУК перейдет в режим работы «Без резервирования». Это происходит в том случае, если один из узлов станет недоступным, тогда кластер переходит в «Аварийный режим» работы.

7.3. Для решения этой задачи на кластере настроен механизм «fencing» посредством модуля pacemaker stonith:fence_vmware_soap. Проблемный узел будет перезагружен средствами гипервизора. При этом проблемный узел будет «забанен» в кластере. А работа кластера будет продолжена на одном узле.

7.4. Настройки кластера:

```
> pcs stonith
 vmfence_gkrs-s-tsmev1n4  (stonith:fence_vmware_soap):  Stopped
 vmfence_gkrs-s-tsmev1n3  (stonith:fence_vmware_soap):  Started gkrs-s-tsmev1n4
```

Настроены два ресурса один из них настроен на основном узле ОУК, а второй настроен для выполнения на резервном узле. Соотвественно, каждый из них следит за доступностью второго узла, и в случае если второй узел станет недоступным, отправится команда «stonith» для перезагрузки недоступного узла средствами гипервизора Vmware.

7.5. В случае, если это произойдет, то кластер перейдет в режим аварийной работы см. подразд. 3.1.

7.6. Для устранения отказа и перевода работы ОУК в «Штатный режим» необходимо устранить возникшую проблему в сетевом оборудовании. Также необходимо перевести работу кластера в штатный режим работы, см. разд. 6.

7.7. По результатам восстановительных работ оба узла должны быть в статусе «Доступен» (Online):

```
> pcs status nodes
Pacemaker Nodes:
 Online: gkrs-s-tsmev1n3 gkrs-s-tsmev1n4
...
```

7.8. Если один из узлов или оба узла не доступны определите причину по журналам ОУК (файлы в директориях *var/log/cluster* и файл /var/log/messages).

7.9. В некоторых случаях может потребоваться принудительно выполнить команду «stonith», это возможно сделать командой с рабочего узла кластера:

```
> pcs stonith fence <node>
```

где node - имя второго (проблемного) узла.

8. ОТСУТСТВИЕ СВОБОДНОГО МЕСТА ИЛИ ИНДЕКСНЫХ ДЕСКРИПТОРОВ ЛОКАЛЬНОЙ ФС НА ОДНОМ ИЗ УЗЛОВ ОУК

При отсутствии свободного места или индексных дескрипторов на корневом разделе любого из узлов кластера происходит частичный или полный отказ ОС и служб, использующих пространство корневого раздела.

Для устранения отказа необходимо удалить или переместить на свободные разделы неиспользуемые или устаревшие данных.

Определить какие службы находятся в не штатном режиме. Восстановить работу этих служб.

9. ОТКАЗ СЕРВЕРА ПРИЛОЖЕНИЙ

9.1. В процессе эксплуатации ОУК могут возникнуть нештатные ситуации в работе сервера приложений. Сервер приложений в ОУК запускается всегда на основном узле кластера. При этом возможны две ситуации:

- Сервер приложений стал недоступен (упал),
- Сервер приложений не работает или работает некорректно, однако процесс сервера приложений не завершался.

9.2. Во всех случаях, требуется изучить журналы сервера приложений, файлы `ncore.log` и `catalina.out`:

```
> tail -n 100 /opt/tomcat/logs/catalina.out
...
> tail -n 100 /opt/tomcat/logs/ncore.log
...
```

9.3. Первая ситуация, штатная для кластера. В этом случае кластер перейдет в аварийный режим работы и потребуется только восстановить штатный режим работы кластера см. разд. 6.

9.4. Вторая ситуация, теоретически возможна. Сервер приложений работает, однако пользователи жалуются на некорректную работу сервера. В таком случае рекомендуется прибегнуть к ручному приведению ОУК в аварийный режим, отключив основной узел, см. подразд. 3.2.

9.5. Далее восстановить работу кластера в штатный режим работы, согл. разд. 6.

ПЕРЕЧЕНЬ ТЕРМИНОВ

Термин	Определение
1. **IP-адрес**	Уникальный сетевой адрес узла в компьютерной сети, построенной на основе стека протоколов TCP/IP.
2. **Pacemaker**	Программа с окрытым исходным кодом, ресурсный менеджер, предназначенная для создания отказоустойчивых кластеров.
3. **STONITH**	(англ. «Shoot The Other Node In The Head») техника защиты узлов, при которой сбойный узел кластера изолируется от остальных узлов (например узел перезагружается или выключается и тп...).
4. **База данных**	Кратко БД. Имеется ввиду файл с данными СУБД «Ред База Данных».
5. **Бан**	Состояние одного или нескольких узлов кластера, когда они специальной настройкой выключены из работы кластера и вернуться работу не смогут до тех пор пока не будут разбанены.
6. **Виртуализация**	Предоставление набора вычислительных ресурсов или их логического объединения, абстрагированное от аппаратной реализации, и обеспечивающее при этом логическую изоляцию друг от друга вычислительных процессов, выполняемых на одном физическом ресурсе.
7. **Лог-файл**	Файл с записями о событиях (и состояниях) кластера в хронологическом порядке.
8. **Миграция**	Специальный режим работы кластера, при котором работа переходит с основного узла кластера к резервному см. Узел.
9. **Отказоустойчивый кластер**	(англ. High-Availability cluster, HA cluster — кластер высокой доступности) — кластер (группа серверов), спроектированный в соответствии с методиками обеспечения высокой доступности и гарантирующий минимальное время простоя за счёт аппаратной избыточности.

Термин	Определение
10. **Репликация БД**	Механизм создания копии БД. В данном документе имеется ввиду специальный режим работы СУБД 'Ред База Данных' при котором в режиме реального времени создается копия данных основной БД (с которой создается копия БД, иначе мастер-база, master) в репликационной БД (иначе реплика, slave).
11. **Сервер приложений**	(англ. application server) — это программная платформа (фреймворк), предназначенная для эффективного исполнения процедур (программ, скриптов), на которых построены приложения... Для веб-приложений основная задача компонентов сервера — обеспечивать создание динамических страниц.
12. **Система управления базами данных**	Кратко СУБД. Совокупность языковых и программных средств, предназначенных для создания, ведения и совместного использования БД многими пользователями. В данном документе 'Ред База Данных' имеется ввиду конкретная реализация СУБД от компании 'Ред Софт'.
13. **Узел**	(англ. node) Имеется ввиду узел кластера. Один из компьютеров кластера. В данном документе два узла основной и второстепенный. Основной узел используется для работы кластера, в то время как второстепенный/резервный узел используется в качестве запасного и в случае проблем с основным узлом работа системы перемещяется (мигрирует) на резервный.

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

Сокращение	Расшифровка
БД	База данных.
ОС	Операционная система.
ОУК	Отказоустойчивый кластер.
СУБД	Система управления базами данных.
ФС	Файловая система.