request-ru@raidix.com
raidix.ru

RAIDIX 5.3.1

Инструкция по проверке системы RAIDIX 5.3.1

Редакция 1

2024

Все права защищены. ООО «Рэйдикс»,
2024

Содержание

Глава 1. Об этом документе....3
Что нового....3
Глава 2. Проверка системы в одноконтроллерном режиме....4
Глава 3. Проверка системы в двухконтроллерном режиме....5
Глава 4. Тестирование локальной производительности....6
Глава 5. Приложение. Скрипты локального нагрузочного тестирования....9
fio-test....9
fio-parser....10

Все права защищены. ООО «Рэйдикс», 2024

ГЛАВА 1. ОБ ЭТОМ ДОКУМЕНТЕ

Документ описывает процедуру проверки системы хранения данных на программном обеспечении RAIDIX 5.3.1.

В документе представлены инструкция для администраторов систем хранения данных по проверке системы после установки и инструкция для тестирования локальной производительности системы, предназначенная для использования партнёрами RAIDIX.

Что нового

Редакция	Изменения	Дата внесения изменения
1	Документ создан.	17.12.2025

Все права защищены. ООО «Рэйдикс», 2024

ГЛАВА 2. ПРОВЕРКА СИСТЕМЫ В ОДНОКОНТРОЛЛЕРНОМ РЕЖИМЕ

Проверка системы в одноконтроллерном режиме проводится после установки системы и применения лицензии в соответствии с инструкцией по установке и настройке системы.

1. В веб-интерфейсе откройте страницу **Мониторинг > Оповещения**. Просмотрите последние оповещения.
2. Откройте страницу **Мониторинг > Датчики** и откройте вкладку **Контроллер**. Убедитесь, что все датчики имеют статус OK .
3. Откройте страницу **Мониторинг > Датчики** и откройте вкладку **Корзина**. Убедитесь, что все датчики имеют статус OK .
4. Откройте страницу **Хранилище > Диски**. Убедитесь, что серийные номера, модели и ёмкость дисков отображаются верно. Для просмотра SMART-диагностики кликните в конце строки нужного диска и выберите **S.M.A.R.T.**. Повторите шаг на втором контроллере
5. Кликните в строке одного из дисков и убедитесь, что световой индикатор включился на соответствующем диске в корзине.
6. Откройте страницу **SAN > Адаптеры**. Убедитесь, что идентификатор, тип, статус подключения и скорость соединения установленных адаптеров отображены верно.
7. Откройте страницу **Мониторинг > Логи и метрики**. Нажмите **Собрать логи** и убедитесь, что начался процесс сбора логов, после которого появилась кнопка **Загрузить журнал событий**.
8. Откройте страницу **Настройки > Система**, затем выберите вкладку **Основной интерфейс**. Убедитесь, что параметры основного интерфейса соответствуют выбранным в процессе установки. При необходимости, настройки можно изменить в соответствии с «Руководством администратора RAIDIX 5.3.1».
9. Создайте iSCSI-таргет в соответствии с «Руководством администратора RAIDIX 5.3.1». Убедитесь, что таргет создан успешно.
10. Создайте RAID и LUN, следуя инструкциям, приведённым в «Руководстве администратора RAIDIX 5.3.1». Убедитесь, что RAID и LUN успешно создаются.
11. Создайте общие папки по протоколам NFS, SMB, FTP, AFP в соответствии с инструкциями в «Руководстве администратора RAIDIX 5.3.1». Убедитесь, что папки успешно создаются.
12. Удалите созданные общие папки в соответствии с инструкциями в «Руководстве администратора RAIDIX 5.3.1». Убедитесь, что папки успешно удаляются.
13. Удалите RAID и LUN в соответствии с инструкциями в «Руководстве администратора RAIDIX 5.3.1». Убедитесь, что RAID и LUN успешно удаляются.

Все права защищены. ООО «Рэйдикс», 2024

ГЛАВА 3. ПРОВЕРКА СИСТЕМЫ В ДВУХКОНТРОЛЛЕРНОМ РЕЖИМЕ

Проверка системы в DC-режиме проводится после установки системы, применения лицензии на обоих контроллерах и настройки двухконтроллерного режима.

1. В веб-интерфейсе откройте страницу **Мониторинг > Оповещения**. Просмотрите последние оповещения.
2. Откройте страницу **Мониторинг > Датчики** и откройте вкладку **Контроллер**. Убедитесь, что все датчики имеют статус OK . Повторите шаг на втором контроллере.
3. Откройте страницу **Мониторинг > Датчики** и откройте вкладку **Корзина**. Убедитесь, что все датчики имеют статус OK . Повторите шаг на втором контроллере.
4. Откройте страницу **Хранилище > Диски**. Убедитесь, что серийные номера, модели и ёмкость дисков отображаются верно. Для просмотра SMART-диагностики кликните в конце строки нужного диска и выберите **S.M.A.R.T.**. Повторите шаг на втором контроллере
5. Кликните в строке одного из дисков и убедитесь, что световой индикатор включился на соответствующем диске в корзине. Повторите шаг на втором контроллере.
6. Откройте страницу **SAN > Адаптеры**. Убедитесь, что идентификатор, тип, статус подключения и скорость соединения установленных адаптеров отображены верно. Повторите шаг на втором контроллере.
7. Откройте страницу **Мониторинг > Логи и метрики**. Нажмите **Собрать логи** и убедитесь, что начался процесс сбора логов, после которого появилась кнопка **Загрузить журнал событий**. Повторите шаг на втором контроллере.
8. Откройте страницу **Настройки > Система**, затем выберите вкладку **Основной интерфейс**. Убедитесь, что параметры основного интерфейса соответствуют выбранным в процессе установки. При необходимости, настройки можно изменить в соответствии с «Руководством администратора RAIDIX 5.3.1». Повторите шаг на втором контроллере.
9. Создайте iSCSI-таргет в соответствии с «Руководством администратора RAIDIX 5.3.1». Убедитесь, что таргет создан успешно. Повторите шаг на втором контроллере.
10. Создайте RAID и LUN, следуя инструкциям, приведённым в «Руководстве администратора RAIDIX 5.3.1». Убедитесь, что RAID и LUN успешно создаются. Повторите шаг на втором контроллере.
11. Создайте общие папки по протоколам NFS, SMB, FTP, AFP в соответствии с инструкциями в «Руководстве администратора RAIDIX 5.3.1». Убедитесь, что папки успешно создаются. Повторите шаг на втором контроллере.
12. Удалите созданные общие папки в соответствии с инструкциями в «Руководстве администратора RAIDIX 5.3.1». Убедитесь, что папки успешно удаляются. Повторите шаг на втором контроллере.
13. Удалите RAID и LUN в соответствии с инструкциями в «Руководстве администратора RAIDIX 5.3.1». Убедитесь, что RAID и LUN успешно удаляются. Повторите шаг на втором контроллере.

Все права защищены. ООО «Рэйдикс», 2024

ГЛАВА 4. ТЕСТИРОВАНИЕ ЛОКАЛЬНОЙ ПРОИЗВОДИТЕЛЬНОСТИ

Обзор

> В DC-системе запускайте тесты на контроллере, на котором активен RAID с тестируемым LUN.

Для тестирования используются следующие паттерны:

- Пять минут последовательного чтения и/или записи с размером блока 4K и глубиной очереди 64.
- Пять минут последовательного чтения и/или записи с размером блока 1M и глубиной очереди 64.
- Пять минут случайного чтения и/или записи с размером блока 4K и глубиной очереди 64.
- Пять минут случайного чтения и/или записи с размером блока 1M и глубиной очереди 64.

Тестирование производится при помощи бенчмарка «Flexible I/O Tester» (далее – FIO) последоватльным запуском двух скриптов: fio-test и fio-parser (см. листинг скриптов в главе *Приложение. Скрипты локального нагрузочного тестирования (стр. 9)*).

Подробное описание параметров см. на *официальном сайте fio*.

> В скрипте fio-test не меняйте порядок параметров, т.к. от него зависит работа скрипта fio-parser.

Для тестирования используются следующие параметры fio:

- blocksize => ["4k", "1m"]

  прочие возможные значения ["4k", "8k", "16k", "32k", "64k", "128k", "256k", "512k", "1m"]

- rw => ["randrw", "rw"]

  прочие возможные значения ["read", "write", "randread", "randwrite", “rw”, “randrw”]

- rwmixread => ["0", "30", "50", "70", "100"]

  прочие возможные значения ["0", "5", "10", "15", "20", "25", "30", "35", "40", "45", "50", "55", "60", "65", "70", "75", "80", "85", "90", "95", "100"]

- direct => [1]
- ioengine => ["libaio"]
- iodepth=> ["64"]

  прочие значения ["1", "4", "8", "32", "64"]

- runtime => [60]
- numjobs => ["32"]

  прочие значения ["1", "8", "32"]

- bwavgtime => [100000]

Тестирование IOPS, времени отклика и пропускной способности

> Все данные на тестируемом LUN будут удалены.

Чтобы локально протестировать производительность LUN рекомендуемым способом:

Все права защищены. ООО «Рэйдикс», 2024

1. Создайте LUN с требуемыми характеристиками.
2. Убедитесь, что логирование отключено:

   ```
   $ rdcli param driver show | grep log_level
   ```

   Значение log_level должно быть 0.

   Чтобы отключить логирование, выполните

   ```
   $ rdcli param driver modify -l 0
   ```

3. Узнайте имя устройства LUN:

   ```
   $ lsblk
   ```

   Устройство LUN будет иметь имя вида lrvm_<lun_name>.

4. Запустите fio-test:

   ```
   $ ./fio-test </dev/lun_for_test> > out.txt
   ```

   Продолжительность тестирования - 20 минут.

5. Запустите fio-parser:

   ```
   $ ./fio-parser out.txt > out.csv
   ```

Аппаратное тестирование

> **!** Все данные на тестируемых дисках будут удалены, в том числе метаданные RAID.

- Тестирование дисков, подключенных через встроенную корзину.
- Тестирование дисков, подключенных через внешнюю корзину.

Для аппаратного тестирования дисков используйте следующую конфигурацию fio:

```
[global]
ioengine=libaio
direct=1
group_reporting

numjobs=32
iodepth=1
bs=1M
runtime=1800s
ramp_time=5s
rw=randrw
offset_increment=10G

rwmixread=50
rwmixwrite=50
```

Для добавления большого числа устройств для тестирования в файл конфигурации вы можете использовать следующий способ:

```
$ echo -e "\n" >> fio.cfg; counter=0; for disk in `lsblk -r|grep "3.7T"|awk '{ print $1 }'`; do echo \[disk$counter\];
 ((counter=counter+1)); echo filename=/dev/$disk; done >> fio.cfg
```

В этом примере файл конфигурации называется «fio.cfg» и в него добавляются все диски из `lsblk` с размером *3.7 T*.

Интерпретация результатов

Значения столбцов:

Все права защищены. ООО «Рэйдикс», 2024

- blocksize – размер блока;
- iodepth – глубина очереди;
- numjobs – количество потоков;
- rw – тип нагрузки (последовательные или случайные чтение/запись);
- rwmixread – соотношение количества команд чтения к количеству команд записи (при 0 только запись, при 100 только чтение, другие значения для смешанной нагрузки);
- r_iops – IOPS при чтении;
- r_maxlat_usec – максимальная задержка в микросекундах при чтении;
- r_avglat_usec – средняя задержка в микросекундах при чтении;
- w_iops – IOPS при записи;
- w_maxlat_usec – максимальная задержка в микросекундах при записи;
- w_avglat_usec – средняя задержка в микросекундах при записи;
- r_aggrb_MB – средняя пропускная способность потоков в МБ при чтении;
- w_aggrb_MB – средняя пропускная способность потоков в МБ при записи.

Все права защищены. ООО «Рэйдикс», 2024

ГЛАВА 5. ПРИЛОЖЕНИЕ. СКРИПТЫ ЛОКАЛЬНОГО НАГРУЗОЧНОГО ТЕСТИРОВАНИЯ

В главе представлены листинги скриптов для рекомендуемого локального тестирования производительности fio-test и fio-parser.

Для использования скриптов необходимы права root.

Для выполнения скрипта ему необходимо задать разрешение на исполнение, например, так:

```
$ chmod +x <script_name>
```

Для файлов скриптов, созданных на ОС Windows, после их копирования на систему с ОС Linux необходимо удалить символы «0x0D». Эта необходимость обусловлена тем, что в ОС разных семейств перевод строки в текстовом файле кодируется разными последовательностями управляющих символов ASCII.

fio-test

```
#!/usr/bin/perl
use strict ;
use warnings ;
use feature qw(say switch state) ;
use Data::Dumper;
my %global = (
# Don't change the parameter sequence. If the sequence is changed, the parser will not work.
#    blocksize => [ "4k", "8k", "16k", "32k", "64k", "128k", "256k", "512k", "1m" ],
    blocksize => [ "4k", "1m" ],
# mixed workflows
#    rwmixread => [ "0", "5", "10", "15", "20", "25", "30", "35", "40", "45", "50", "55", "60", "65", "70", "75", "80",
 "85", "90", "95", "100" ],
    rw        => [ "rw", "randrw" ],
    rwmixread => [ "0", "30", "50", "70", "100" ],
#direct
    direct    => [ 1 ],
#    size      => [ "1T" ],
    ioengine  => [ "libaio" ],
#    iodepth   => [ "1", "4", "8", "32", "64" ],
    iodepth   => [ "64" ],
    runtime   => [ 60 ],
#    numjobs   => [ "1", "8", "32" ],
    numjobs   => [ "32" ],
    bwavgtime => [ 100000 ],
#    write_bw_log => [ "results" ],
#    write_iops_log => [ "results" ],
#    write_lat_log => [ "results"],
);

my @config_name = qw( blocksize rw iodepth numjobs );

my $device1 = $ARGV[0];
#my $device2 = $ARGV[1];
#my $device3 = $ARGV[2];
#my $device4 = $ARGV[3];


sub cartesian {
    my @C = map { [ $_ ] } @{ shift @_ };
    foreach (@_) {
        my @A = @$_;
        @C = map { my $n = $_; map { [ $n, @$_ ] } @C } @A;
    }
    return @C;
}

my @keys = sort keys %global ;
my @global = map { $global{$_} } @keys ;

foreach ( cartesian( @global ) ) {
```


Все права защищены. ООО «Рэйдикс», 2024

```
    my $config .= "[global]\n" ;
    my $parameters .= "" ;
    my $template = "";
    my @params = reverse @$_ ;
    foreach (@keys) {
        $template = shift( @params ) ;

        $config .= "$_=" ;
        $config .= $template ;
        $config .= "\n" ;

            if ($_ ~~ @config_name) {
              $parameters .= "$_-" ;
              $parameters .= $template ;
              $parameters .= "_" ;
            }
    }
    $config .= "group_reporting\n" ;
    $config .= "[job]\n" ;
    $config .= "filename=$device1\n" ;
#    $config .= "filename=$device2\n" ;
#    $config .= "filename=$device3\n" ;
#    $config .= "filename=$device4\n" ;
    print "-----------------\n" ;
    print "Start new test:\n" ;
    print "-----------------\n" ;
    my $time = localtime;
    print "$time\n";
    print $config ;
    print "-----------------\n" ;
#    system( "echo \"$config\" | fio --output-format=normal --output=results_$parameters.txt -" ) ;
    system( "echo \"$config\" | fio --output-format=normal -" ) ;
    sleep 1 ;
}
```

fio-parser

```
#!/usr/bin/perl

use strict ;
use warnings ;
use Fcntl qw(O_RDONLY LOCK_EX LOCK_UN F_GETFD F_SETFD FD_CLOEXEC) ;
use feature qw(say switch state) ;
use threads ;
use Data::Dumper;
use feature "switch";

my $ss=0;
my $avg_lat=0;
my $max_lat=0;
my $IOps=0;
my $var=0;
my $BW=0;

sub to_kbps {
    $_ = shift ;
    my ($val) = /([0-9.]+)/ ;
    return (/KB\/s/)? $val / 1000 : $val ;
}
print "sep=;\n";
print
 "blocksize;iodepth;numjobs;rw;rwmixread;r_iops;r_maxlat_usec;r_avglat_usec;w_iops;w_maxlat_usec;w_avglat_usec;r_aggrb_
MB;w_aggrb_MB;\n" ;
while( <> ) {
    if ( /^blocksize=\s*([^ ,\n]+)/ ) { print "$1;"; next }
    if ( /^iodepth=\s*([^ ,\n]+)/ ) { print "$1;"; next }
    if ( /^numjobs=\s*([^ ,\n]+)/ ) { print "$1;"; next }
    if ( /^rw=\s*([^ ,\n]+)/ )        { print "$1;"; next }
    if ( /^rwmixread=\s*([^ ,\n]+)/ ) {
        $ss=$1;
        if ( $ss==0 ) { print "$ss;0;0;0;";  next }
        else { print "$ss;"; next }
    }
    if ( /IOPS=\s*([^ ,\n]+)/ ) {
        $IOps = $1;
        if ( $IOps =~ /[.][0-9][k]/ ) {
            $IOps =~ tr/k/0/;
            $IOps = $IOps*1000;
            print "$IOps;"; next }
        if ( $IOps =~ /[k]/ ) {
            $IOps =~ tr/k/0/;
            $IOps = $IOps*100;
```


Все права защищены. ООО «Рэйдикс», 2024

```
            print "$IOps;"; next }
        else { print "$IOps;"; next }
    }
#    if ( /bw.*min=\s*([^ ,\n]+), max=\s*([^ ,\n]+).*avg=\s*([^ ,\n]+), stdev=\s*([^ ,\n]+)/ )
#        { print "$1;$2;$3;$4;"; next }
#    if ( /[^cs]lat.*\(([u,m]).*max=\s*([^ ,\n]+).*avg=\s*([^ ,\n]+)/) { print "$1_$2;$1_$3;"; next }
    if ( /[^cs]lat.*\(([u,m]).*max=\s*([^ ,\n]+).*avg=\s*([^ ,\n]+)/) {
        if ( $1 eq "u" ) {
            $max_lat=$2;
            $avg_lat=$3;
            given ($max_lat) {
              when ( /[.][0-9][k]/ ) {
                $max_lat =~ tr/k/0/;
                $max_lat = $max_lat*1000;
                print "$max_lat;";
              }
              when ( /[k]/ ) {
                $max_lat =~ tr/k/0/;
                $max_lat = $max_lat*100;
                print "$max_lat;";
              }
              default                { print "$max_lat;" }
            }
            given ($avg_lat) {
              when ( /[.][0-9][k]/ ) {
                  $avg_lat =~ tr/k/0/;
                  $avg_lat = $max_lat*1000;
                  print "$avg_lat;";
              }
              when ( /[k]/ ) {
                  $avg_lat =~ tr/k/0/;
                  $avg_lat = $avg_lat*100;
                  print "$avg_lat;";
              }
              default                { print "$avg_lat;" }
            }
            #print "$2;$3;";
             next }
        elsif ( $1 eq "m" ) {
          $max_lat=$2*1000;
          $avg_lat=$3*1000;
          print "$max_lat;$avg_lat;"; next
          }
        else {
          print "not_usec_or_msec_$max_lat;not_usec_or_msec_$avg_lat;"; next
          }
         }
    if ( /(READ): bw=[^\(]+\(([\d+(\.\d{1,2})?+M+B+G+B+K+B]+)/ ) {
        if ( $ss==100 ) {
            $BW=$2;
            given ($BW) {
                when ( /(MB)/ ) {
                    $BW =~ s/.{2}$//;
                    print "0;0;0;$BW;0\n";
                }
                when ( /(GB)/ ) {
                    $BW =~ s/.{2}$//;
                    $BW = $BW*1000;
                    print "0;0;0;$BW;0\n";
                }
                default                { print "0;0;0;when_other_$2;0\n" }
              }
             #$BW =~ s/.{2}$//;
             #print "0;0;0;0;0;0;0;when1_MB_$2;0\n";
             next }
        else {
            $BW=$2;
            given ($BW) {
                when ( /(MB)/ ) {
                    $BW =~ s/.{2}$//;
                    print "$BW;";
                }
                when ( /(GB)/ ) {
                     $BW =~ s/.{2}$//;
                     $BW = $BW*1000;
                     print "$BW;";
                }
                default                { print "when_other_$2;" }
              }
             #$BW =~ s/.{2}$//;
             #print "$2;" ;
             next }
        }
    if ( /(WRITE): bw=[^\(]+\(([\d+(\.\d{1,2})?+M+B+G+B+K+B]+)/ ) {
       if ( $ss==0 ) {
            $BW=$2;
            given ($BW) {
```


Все права защищены. ООО «Рэйдикс», 2024

```
                when ( /(MB)/ ) {
                    $BW =~ s/.{2}$//;
                    print "0;$BW;\n";
                }
                when ( /(GB)/ ) {
                    $BW =~ s/.{2}$//;
                    $BW = $BW*1000;
                    print "0;$BW;\n";
                }
                default                { print "0;when_other_$2;\n" }
            }
            #print "0;$2;\n" ;
            next }
        else {
            $BW=$2;
            given ($BW) {
                when ( /(MB)/ ) {
                    $BW =~ s/.{2}$//;
                    print "$BW;\n";
                }
                when ( /(GB)/ ) {
                    $BW =~ s/.{2}$//;
                    $BW = $BW*1000;
                    print "$BW;\n";
                }
                default                { print "when_other_$2;\n" }
            }
            #print "$2;\n" ;
            next }
        }
    }
print
 "blocksize;iodepth;numjobs;rw;rwmixread;r_iops;r_maxlat_usec;r_avglat_usec;w_iops;w_maxlat_usec;w_avglat_usec;r_aggrb_
MB;w_aggrb_MB;\n" ;
```


Все права защищены. ООО «Рэйдикс», 2024