Руководство по эксплуатации
распределенной системы хранения данных

Р-Хранилище

с командной строкой

ООО «Р-Платформа»
ОГРН 1167746349858, ИНН 9715253528
Россия, г. Москва,
Отрадная улица, 2Б, стр. 9, 3 этаж
Тел.: 8-800-700-7460
www.rosplatforma.ru

Содержание

Глава 1

Введение

В главе представлена основная информация о программном комплексе «Распределенная система хранения данных «Р-Хранилище» (далее ПК Р-Хранилище).

О данном руководстве

Данное руководство предназначено для двух целей:

- Если вы управляете ПК Р-Хранилище с помощью веб-панели, данное руководство дополняет документацию по панели управления (см. *Руководство по эксплуатации ПК Р-Хранилище с графическим интерфейсом*). Ознакомьтесь с данным руководством, чтобы узнать, как использовать инструменты командной строки для выполнения задач, безопасных для ПК Р-Хранилище с графическим интерфейсом. Однако следует пропустить главы **Установка кластера ПК Р-Хранилище** (стр. 9) и **Настройка кластеров ПК Р-Хранилище** (стр. 23), так как команды, используемые в этих главах, не являются безопасными для ПК Р-Хранилище с графическим интерфейсом.
- Если вы управляете ПК Р-Хранилище через командную строку (т.е. панель управления не установлена), то используйте данное руководство. Ознакомьтесь с ним для того, чтобы узнать, как выполнить все задачи, связанные с управлением кластерами ПК Р-Хранилище, от начальной настройки до мониторинга, оптимизации и устранения неисправностей.

Как правило, рекомендуется управлять ПК Р-Хранилище через панель управления. Если она установлена, то инструменты командной строки можно считать вторичным способом управления и использовать их с осторожностью.

Если панель управления установлена, не следует выполнять следующие действия при помощи инструментов командной строки:

- задавать пути к службам по своему усмотрению ПК Р-Хранилище, в частности:
  - кластеры S3 необходимо создавать только в `/mnt/vstorage/vols/s3`,
  - iSCSI-таргеты необходимо создавать только в `/mnt/vstorage/vols/iscsi`,
- монтировать кластеры или изменять параметры монтирования кластера,
- настраивать брандмауэр с помощью `firewall-cmd`,
- переименовывать сетевые соединения,
- управлять серверами метаданных и фрагментов,

- управлять разделами, логическими томами (LVM) или программными RAID-массивами,
- изменять файлы в директориях `/mnt/vstorage/vols` и `/mnt/vstorage/webcp/backup`,
- настраивать репликацию или избыточное кодирование для root кластера.

О ПК Р-Хранилище

ПК Р-Хранилище позволяет быстро и легко преобразовать недорогое стандартное сетевое оборудование и оборудование для хранения данных в сетевое хранилище с многоуровневой системой защиты, такое как SAN (Storage Area Network) и NAS (Network Attached Storage).

Решение ПК Р-Хранилище оптимизировано для хранения больших объемов данных, оно также обеспечивает репликацию, высокую доступность и самовосстановление данных. ПК Р-Хранилище дает возможность безопасного хранения и запуска виртуальных машин и контейнеров ПК Р-Виртуализация, их живую миграцию между физическими хостами, а также отказоустойчивость установок ПК Р-Виртуализация и т.д.

Глава 2

Установка кластера ПК Р-Хранилище

В данной главе представлена информация по установке кластера ПК Р-Хранилище. Сначала дается краткий обзор установки ПК Р-Хранилище, а затем подробно описывается каждый ее шаг.

Обзор установки

Установка кластера ПК Р-Хранилище состоит из следующих шагов:

1. Настройка обнаружения кластера ПК Р-Хранилище (стр. 9). На этом шаге необходимо определить способ обнаружения кластера и преобразования его имени в IP-адреса серверов метаданных.
2. Проверка сброса данных на диск (стр. 13). На этом шаге нужно убедиться, что все устройства хранения (жесткие диски, твердотельные накопители, RAID-массивы и др.), которые будут включены в кластер, могут сбросить данные на диск в случае отключения питания сервера.
3. Подготовка дисков для ПК Р-Хранилище (стр. 15). На этом шаге при необходимости нужно подготовить дополнительные (второй, третий и т.д.) диски.
4. Установка серверов метаданных (стр. 16). На этом шаге создаются и настраиваются серверы метаданных для хранения метаданных о серверах фрагментов и фрагментов данных.
5. Установка серверов фрагментов (стр. 18). На этом шаге создаются и настраиваются серверы фрагментов для хранения содержимого виртуальных машин и контейнеров во фрагментах данных.
6. Установка клиентов (стр. 21). На этом шаге создаются и настраиваются клиенты, с которых осуществляется доступ к кластеру ПК Р-Хранилище и запуск виртуальных машин и контейнеров.

Примечание: Установить ПК Р-Хранилище также можно с помощью установщика ПК Р-Виртуализация. Он автоматически настраивает систему как сервер метаданных, сервер фрагментов или клиент. Для получения подробной информации см. *Руководство по установке ПК Р-Виртуализация.*

Настройка обнаружения кластера

Обнаружение кластера ПК Р-Хранилище – это:

- Процесс обнаружения всех доступных кластеров в сети. Каждый кластер ПК Р-Хранилище идентифицируется по уникальному имени. Все инструменты кластера используют данное имя при выполнении определенных операций с кластером или при мониторинге его состояния и статуса.
- Процесс преобразования обнаруженных имен кластеров в сетевые адреса серверов метаданных. Серверы метаданных являются ключевыми компонентами любого кластера, поэтому все инструменты кластера должны быть способны обнаружить их IP-адреса.

Для того чтобы настроить обнаружение кластера в сети, можно использовать один из следующих способов:

- (Рекомендуется) **Записи DNS** (стр. 10),
- **Zeroconf** (стр. 12).

Также можно указать информацию о серверах метаданных вручную при установке и настройке кластера (стр. 12).

Примечание: Чтобы проверить, может ли физический сервер обнаружить кластер, используйте команду `vstorage discover`.

Использование записей DNS

При настройке обнаружения кластера рекомендуется использовать специальные записи DNS. Данный процесс состоит из следующих шагов:

1. Объявление информации о запущенных серверах метаданных в кластере, чтобы при необходимости серверы фрагментов, клиенты и новые серверы метаданных могли ее автоматически получить. Данная процедура осуществляется с помощью DNS SRV-записей.
2. Определение DNS TXT-записей или включение передачи зоны DNS, чтобы при необходимости иметь возможность обнаружения уникальных имен доступных кластеров.

Объявление информации о серверах метаданных

Для объявления информации о запущенных серверах метаданных можно использовать SRV-записи. Служебное поле SRV-записи, указывающее на сервер метаданных, должно иметь следующий формат:

```
_pstorage._tcp.CLUSTER_NAME
```

где

- `_pstorage` является символьным именем для ПК Р-Хранилище;
- `_tcp` указывает на то, что ПК Р-Хранилище использует протокол TCP для передачи данных в кластере;
- `CLUSTER_NAME` является именем кластера ПК Р-Хранилище, описанного в записи.

В примере ниже показан файл зоны DNS, который содержит записи для трех серверов метаданных, прослушивающих порт 2510 по умолчанию и настроенных для кластера `stor1`:

```
$ORIGIN stor.test.
$TTL   1H
@      IN SOA ns rname.invalid. (1995032001 5H 10M 1D 3H)
NS @
A      192.168.100.1 s1
A      192.168.100.1 s2
A      192.168.100.2 s3
A      192.168.100.3
; SERVICE SECTION
; MDS for the 'stor1' cluster runs on s1.stor.test and listens on port 2510
_pstorage._tcp.stor1SRV 0 1 2510  s1
; MDS for the 'stor1' cluster runs on s2.stor.test and listens on port 2510
_pstorage._tcp.stor1 SRV 0 1 2510 s2
; MDS for the 'stor1' cluster runs on s3.stor.test and listens on port 2510
_pstorage._tcp.stor1SRV 0 1 2510  s3
; eof
```

После настройки DNS SRV-записей для кластера `stor1` их можно отобразить в виде списка с помощью следующего SRV запроса:

```
# host -t SRV _pstorage._tcp.stor1
_pstorage._tcp.stor1.stor.test has SRV record 0 1 2510 s1.stor.test.
_pstorage._tcp.stor1.stor.test has SRV record 0 1 2510 s2.stor.test.
_pstorage._tcp.stor1.stor.test has SRV record 0 1 2510 s3.stor.test.
```

Обнаружение имен кластеров

Самым простым и надежным способом обнаружения имен кластеров в сети является указание всех имен кластеров в `pstorage_clusters` TXT-записях файлов зоны DNS. Ниже дается образец допустимых форматов TXT-записей для кластеров ПК Р-Хранилище:

```
pstorage_clusters   300    IN     TXT    "cluster1,cluster2" "cluster3,cluster4"
pstorage_clusters   300    IN     TXT    "cluster5"
pstorage_clusters   300    IN     TXT    "cluster6" "cluster7"
```

Другим способом обнаружения имен кластеров в сети является использование передачи зоны DNS. После включения передачи зоны DNS инструменты кластера смогут извлечь все DNS SRV-записи из файлов фоны DNS, а затем имена кластеров из этих записей.

После настройки обнаружения кластера с помощью DNS TXT-записей или передачи зоны DNS можно выполнить команду `vstorage discover` на любом сервере кластера, чтобы обнаружить имена всех кластеров в сети:

```
# vstorage discover
02-10-12 13:16:46.233 Discovering using DNS TXT records: OK
02-10-12 13:16:46.308 Discovering using DNS zone transfer: FAIL
stor1
stor2
stor3
```

В примере выше вывод `vstorage` показывает, что:

- Имена кластеров обнаружены с помощью DNS TXT-записей.
- В сети в данный момент установлены три кластера с именами `stor1`, `stor2` и `stor3`.

Установка Zeroconf

Внимание: Обнаружение с помощью Zeroconf не работает, если службы запущены в виртуальных машинах или контейнерах.

Zeroconf является другим методом обнаружения имен кластеров и преобразования их в IP-адреса запущенных серверов метаданных. Данный метод не требует специальной настройки с вашей стороны, необходимо только убедиться, что в сети поддерживается и включено многоадресное вещание.

Примечание: Чтобы проверить, может ли физический сервер обнаружить кластер, используйте команду `vstorage discover`.

Указание серверов метаданных вручную

Если невозможно настроить записи DNS в сети, то необходимо каждый раз вручную указывать IP-адреса для всех запущенных серверов метаданных в кластере при выполнении следующих действий:

- Установка нового сервера метаданных в кластере (кроме первого сервера метаданных). Для получения подробной информации см. **Добавление серверов метаданных** (стр. 23).
- Установка нового сервера фрагментов в кластере. Для получения подробной информации см. **Установка серверов фрагментов** (стр. 18).
- Установка нового клиента в кластере. Для получения подробной информации см. **Установка клиентов** (стр. 21).

Чтобы вручную указать IP-адрес для сервера метаданных, нужно создать файл `bs.list` в директории `/etc/vstorage/clusters/Cluster_Name` (создайте данную директорию, если она не существует) на том сервере, который настраивается для кластера, и указать в нем IP-адрес и порт, которые будут использоваться для соединения с сервером метаданных. Например:

```
# echo "10.30.100.101:2510" >> /etc/vstorage/clusters/stor1/bs.list
# echo "10.30.100.102:2510" >> /etc/vstorage/clusters/stor1/bs.list
```

Данная команда:

1. Предполагает, что осуществляется настройка обнаружения для кластера `stor1` (таким образом, имя директории `/etc/vstorage/clusters/stor1`).
2. Создает файл `/etc/vstorage/clusters/stor1/bs.list` на сервере, если он не существовал ранее.
3. Добавляет информацию о двух серверах метаданных с IP-адресами 10.30.100.101 и 10.30.100.102 в файл `bs.list`.

Проверка сброса данных на диск

Перед созданием кластера рекомендуется проверить, что все устройства хранения (жесткие диски, твердотельные накопители, RAID-массивы и др.), которые будут включены в кластер, могут сбросить данные на диск в случае отключения питания сервера. Данная процедура поможет обнаружить возможные проблемы с устройствами, которые могут потерять хранящиеся в их кэше данные из-за сбоя питания.

В состав ПК Р-Хранилище входит утилита `vstorage-hwflush-check`, чтобы проверять, как устройство хранения сбрасывает данные на диск в экстренных случаях (например, при отключении питания). Утилита имеет клиентскую и серверную части:

- **Клиентская часть** непрерывно записывает блоки данных на устройство хранения. Записав блок данных, клиент увеличивает значение счетчика и отправляет его серверной части.
- **Серверная часть** следит за значениями счетчика, приходящими от клиентской части, и знает, какое значение должно прийти следующим. Если пришедшее от клиента значение счетчика меньше значения, полученного ранее (например, из-за отключения питания и невыполнения сброса кэшированных данных на диск), серверная часть сообщает об ошибке.

Чтобы проверить, что устройство хранения сможет выполнить сброс данных на диск при отключении питания, выполните следующие действия:

Со стороны серверной части:

1 На компьютере с ПК Р-Виртуализация установите инструмент `vstorage-hwflush-check`. Данный инструмент входит в пакет `vstorage-ctl`, и его можно установить с помощью следующей команды:

```
# yum install vstorage-ctl
```

2 Запустите серверную часть `vstorage-hwflush-check`:

```
# vstorage-hwflush-check -l
```

Со стороны клиента:

1 На компьютере с устройством хранения, которое нужно проверить, установите инструмент `vstorage- hwflush-check`:

```
# yum install vstorage-ctl
```

2 Запустите клиентскую часть `vstorage-hwflush-check`, например:

```
# vstorage-hwflush-check -s vstorage1.example.com -d /vstorage/stor1-ssd/test -t 50
```

где

- `-s vstorage1.example.com` является именем хоста компьютера, на котором запущена серверная часть `vstorage- hwflush-check`.
- `-d /vstorage/stor1-ssd/test` определяет директорию, которая будет использоваться при проверке сброса данных на диск. При выполнении этой

команды клиент создает файл в данной директории и записывает в него блоки данных.

- `-t 50` устанавливает число потоков для клиента, чтобы записывать данные на диск. Каждый поток имеет свой файл и счетчик.

При запуске клиентской части также можно указать другие параметры. Для получения подробной информации о доступных параметрах см. man-страницу по `vstorage-hwflush-check`.

3 Подождите 10-15 секунд и выключите компьютер, на котором запущен клиент, а затем включите его снова.

> **Примечание**: При нажатии кнопки **Reset** компьютер не выключается, поэтому необходимо нажать кнопку **Power** или отключить кабель питания.

4 Запустите клиент с помощью той же команды, что и при его запуске в первый раз:

```
# vstorage-hwflush-check -s vstorage1.example.com -d /vstorage/stor1-ssd/test -t 50
```

После включения клиент читает все записанные данные, проверяет их и перезапускает проверку от последнего действительного значения счетчика. Затем он отправляет данное значение счетчика серверной части, и она сравнивает его с предыдущим значением.

Вывод может иметь следующий вид:

```
id<N>:<counter_on_disk> -> <counter_on_server>
```

- Если значение счетчика на диске меньше значения счетчика на сервере, это означает, что устройству хранения не удалось сбросить данные на диск. Не рекомендуется использовать устройство хранения в рабочей среде – особенно для сервера фрагментов или журналов – так как есть риск потери данных.
- Если значение счетчика на диске больше значения счетчика на сервере, значит, устройство хранения сбросило данные на диск, но клиенту не удалось сообщить об этом серверу. Сетевое соединение может быть слишком медленным, или устройство хранения – слишком быстрым для заданного числа загруженных потоков. Для устранения данной проблемы можно увеличить их количество. Данное устройство хранения рекомендуется использовать в рабочей среде.
- Если значения счетчиков равны, значит, устройство хранения сбросило данные на диск, и клиент сообщил об этом серверу. Данное устройство хранения рекомендуется использовать в рабочей среде.

Для большей верности повторите данную процедуру несколько раз. После проверки первого устройства хранения, выполните проверку остальных устройств хранения, которые будут использоваться в кластере. Необходимо проверить следующие устройства:

- SSD-диски, используемые для кэширования клиента и журналирования серверов фрагментов,
- диски, используемые для журналов сервера метаданных,
- диски, используемые для серверов фрагментов.

Подготовка дисков для ПК Р-Хранилище

Каждый сервер фрагментов представляет собой службу, которая использует один физический диск в кластере. Не смотря на то, что диск должен использоваться только одной службой сервера фрагментов, технически его можно использовать для различных целей. Например, создать небольшой раздел для операционной системы, а остальное место на диске предоставить ПК Р-Хранилище. Если диск уже разделен, то можно пропустить этот шаг и перейти к созданию сервера фрагментов. В противном случае, следуйте инструкциям из данного раздела, чтобы подготовить диск к использованию в ПК Р-Хранилище.

Новые диски, подключенные к физическому серверу и обнаруженные им, должны быть подготовлены к использованию в кластере ПК Р-Хранилище с помощью инструмента `/usr/libexec/vstorage/prepare_vstorage_drive`. Инструмент делает следующее:

1. Удаляет существующие разделы с диска.
2. Создает и форматирует необходимые разделы.

После данной процедуры самостоятельно добавьте новые разделы в `/etc/fstab`.

Примечания:

1. Если диск не должен быть загрузочным, запустите инструмент с параметром `--noboot`, чтобы пропустить установку загрузчика операционной системы GRUB.

2. Для SSD-дисков используйте параметр `--ssd`.

3. Для пропуска запросов подтверждения используйте параметр `-y`.

Подготовка дисков к использованию в качестве серверов фрагментов

1. Чтобы подготовить HDD- или SSD-диск к использованию в роли сервера фрагментов, запустите инструмент, используя имя диска в качестве параметра. Например:

```
# /usr/libexec/vstorage/prepare_vstorage_drive /dev/sdb
ALL data on /dev/sdb will be completely destroyed. Are you sure to continue? [y]
y
Zeroing out beginning and end of /dev/sdb...
Partitioning /dev/sdb...
Waiting for kernel...
Formatting /dev/sdb1 partition...
Done!
```

2. Узнайте UUID раздела:

```
# ls -al /dev/disk/by-uuid/ | grep sdb1
lrwxrwxrwx 1 root root 10 Jun 19 02:41 f3fbcbb8-4224-4a6a-89ed-3c55bbc073e0 -> \
../../sdb1
```

3. Добавьте новый раздел в `/etc/fstab` по UUID.

   - Для `vzkernel` 2.6.32-042stab108.8 или более новой версии, рекомендуется использовать параметр монтирования `lazytime`. Например:

```
UUID=f3fbcbb8-4224-4a6a-89ed-3c55bbc073e0 /vstorage/stor1-cs1 ext4 \
    defaults,lazytime 1 2
```

- Для старых версий `vzkernel` используйте параметр монтирования `defaults`. Например:

```
UUID=f3fbcbb8-4224-4a6a-89ed-3c55bbc073e0 /vstorage/stor1-cs1 ext4 \
    defaults 1 2
```

4 Монтируйте раздел к `/vstorage/<cluster>-cs<N>`, где `<cluster>` является именем кластера, а `<N>` - первым неиспользованным порядковым номером сервера фрагментов.

Примечание: Если `/vstorage/<cluster>-cs<N>` не существует, необходимо его создать.

Подготовка SSD-дисков для журналирования или кэширования

1 Чтобы подготовить SSD-диск для журналирования или кэширования, запустите инструмент с двумя параметрами: `--ssd` и именем диска. Например:

```
# /usr/libexec/vstorage/prepare_vstorage_drive /dev/sdb --ssd
ALL data on /dev/sdb will be completely destroyed. Are you sure to continue? [y]
y
Zeroing out beginning and end of /dev/sdb...
Partitioning /dev/sdb...
Waiting for kernel...
Formatting /dev/sdb1 partition...
Done!
```

2 Узнайте UUID раздела:

```
# ls -al /dev/disk/by-uuid/ | grep sdb1
lrwxrwxrwx 1 root root 10 Jun 19 02:41 f3fbcbb8-4224-4a6a-89ed-3c55bbc073e0 -> \
../../sdb1
```

3 Добавьте новый раздел в `/etc/fstab` по UUID.

- Для `vzkernel` 2.6.32-042stab108.8 или более новой версии, рекомендуется использовать параметр монтирования `lazytime`. Например:

```
UUID=f3fbcbb8-4224-4a6a-89ed-3c55bbc073e0 /vstorage/stor1-ssd1 ext4 \
    defaults,lazytime 1 2
```

- Для старых версий `vzkernel` используйте параметр монтирования `defaults`. Например:

```
UUID=f3fbcbb8-4224-4a6a-89ed-3c55bbc073e0 /vstorage/stor1-ssd1 ext4 \
    defaults 1 2
```

4 Монтируйте раздел к `/vstorage/<cluster>-cs<N>`, где `<cluster>` является именем кластера, а `<N>` - первым неиспользованным порядковым номером SSD-диска.

Примечание: Если `/vstorage/<cluster>-cs<N>` не существует, необходимо его создать.

Установка первого сервера метаданных

Установка первого сервера метаданных является первым шагом в создании кластера ПК Р-Хранилище. Для обеспечения высокой доступности позже можно добавить в кластер больше серверов метаданных, как описано в разделе **Настройка серверов метаданных** (стр. 23).

Процесс установки сервера метаданных (или *master сервера метаданных*) состоит из двух этапов:

1 Подготовка к созданию сервера метаданных (стр. 17).

2 Создание сервера метаданных (стр. 17).

Этап 1: Подготовка к созданию первого сервера метаданных

При подготовке к созданию первого сервера метаданных выполните следующие действия:

1 Выберите имя для кластера, которое будет уникально идентифицировать его среди остальных кластеров в сети. Имя может содержать символы a-z, A-Z, 0-9, тире (-) и подчеркивание (_). В примерах данного руководства используется имя кластера `stor1`.

> **Примечание**: При выборе имени для кластера убедитесь, что оно уникально в сети. Не рекомендуется использовать имена, которые были раньше назначены другим кластерам в сети, даже если эти кластеры уже не существуют. Данное правило поможет избежать возможных проблем со службами предыдущих кластеров, которые могут быть все еще запущены и попытаются работать в новом кластере. Хотя подобные попытки и не будут успешными, они могут значительно усложнить работу администратору кластера.

2 Войдите в компьютер, который будет настроен как сервер метаданных, в роли пользователя root или пользователя с привилегиями root.

3 Загрузите и установите на компьютер следующие пакеты RPM: `vstorage-ctl`, `vstorage-libs-shared` и `vstorage-metadata-server`. Пакеты доступны в удаленном хранилище ПК Р-Виртуализация (автоматически настраивается при установке ПК Р-Виртуализация), и их можно установить с помощью следующей команды:

```
# yum install vstorage-metadata-server
```

4 Убедитесь, что в сети настроено обнаружение кластера. Для получения подробной информации см. **Настройка обнаружения кластера** (стр. 9).

После выполнения перечисленных действий можно приступить к созданию сервера метаданных.

Этап 2: Создание первого сервера метаданных

Для создания первого сервера метаданных используйте команду `vstorage make-mds`, например:

```
# vstorage -c stor1 make-mds -I -a 10.30.100.101 -r /vstorage/stor1-mds -p
```

Данная команда:

1 Просит ввести пароль, который будет использоваться для идентификации по паролю в кластере. Идентификация по паролю повышает безопасность, так как каждый сервер проходит идентификацию до включения его в кластер. Указываемый пароль

зашифровывается и сохраняется на сервере метаданных в файле `/etc/vstorage/clusters/stor1/auth_digest.key`.

2 Создает кластер ПК Р-Хранилище с именем `stor1` (параметр `-I` говорит `vstorage` создать новый кластер).

3 Создает сервер метаданных и настраивает IP-адрес 10.30.100.101 для соединения с данным сервером. По умолчанию ПК Р-Хранилище использует порты 2510 и 2511 для соединения с серверами метаданных. При необходимости можно заменить стандартные порты своими собственными, зарезервировав два незанятых последовательных порта и указав первый из них после IP-адреса сервера метаданных (например, `-a 10.30.100.101:4110`, если выбраны порты 4110 и 4111).

   В примере выше замените 10.30.100.101 IP-адресом своего сервера метаданных. Указываемый IP-адрес должен быть (1) статическим (или в случае использования DHCP, преобразованный в MAC-адрес сервера метаданных) и (2) выбранным из диапазона IP-адресов в сети BackNet, которая выделена кластеру ПК Р-Хранилище. Для получения подробной информации см. **Сетевые требования** в *Руководстве по установке ПК Р-Виртуализация*.

4 Создает журнал в директории `/vstorage/stor1-mds` на сервере метаданных и добавляет в него информацию о кластере `stor1`. При выборе директории для хранения журнала убедитесь, чтобы в разделе, в котором будет находиться данная директория, было как минимум 10 ГБ свободного дискового пространства.

После создания сервера метаданных запустите службу управления сервером метаданных (`vstorage-mdsd`) и настройте ее таким образом, чтобы она запускалась при загрузке сервера:

```
# systemctl start vstorage-mdsd.target
```

Для получения информации о включении дополнительных серверов метаданных в кластер ПК Р-Хранилище см. **Настройка серверов метаданных** (стр. 23).

Установка серверов фрагментов

Сервер фрагментов хранит данные виртуальных машин и контейнеров и обрабатывает запросы к ним. Все данные делятся на фрагменты и могут храниться в кластере ПК Р-Хранилище во многих копиях, которые называются *репликами*.

Изначально любой кластер настраивается таким образом, что для каждого фрагмента данных создается одна реплика. Одной реплики достаточно для оценки функциональности ПК Р-Хранилище при использовании только одного сервера. Однако в рабочей среде для обеспечения высокой доступности данных необходимо настроить кластер таким образом, чтобы для каждого фрагмента данных создавалось, по крайней мере, три реплики, что, в свою очередь, требует установки в кластере как минимум трех серверов фрагментов. Изменить стандартные параметры репликации можно с помощью утилиты `vstorage`. Для получения подробной информации см. **Настройка параметров репликации** (стр. 29).

Примечания:

1. Использование общего JBOD-массива на нескольких серверах со службами сервера фрагментов может стать единой точкой отказа и привести к недоступности кластера, если все реплики данных будут выделены и храниться в отказавшем JBOD-массиве. Для получения дополнительной информации см. **Настройка областей отказов** (стр. 31).

2. Не рекомендуется устанавливать серверы фрагментов на диски, которые уже используются в других нагрузках ввода-вывода, например, для системы или области подкачки. Использование общих дисков для сервера фрагментов и других источников ввода-вывода приведет к снижению производительности и увеличению времени задержки ввода-вывода.

Процесс установки сервера фрагментов состоит из двух этапов:

1 Подготовка к созданию сервера фрагментов (стр. 19).

2 Создание сервера фрагментов (стр. 20).

Этап 1: Подготовка к созданию сервера фрагментов

При подготовке к созданию сервера фрагментов выполните следующие действия:

1 Войдите в компьютер, который будет использоваться в качестве сервера фрагментов, в роли пользователя root или пользователя с привилегиями root.

2 Загрузите и установите следующие пакеты RPM: `vstorage-ctl`, `vstorage-libs-shared` и `vstorage-chunk-server`. Пакеты доступны в удаленном хранилище ПК Р-Виртуализация (данное хранилище автоматически настраивается для вашей системы при установке ПК Р-Виртуализация), и их можно установить с помощью следующей команды:

```
# yum install vstorage-chunk-server
```

3 Убедитесь, что для сервера настроено обнаружение кластера. Для получения подробной информации см. **Настройка обнаружения кластера** (стр. 9).

4 Выполните идентификацию сервера в кластере. Данная процедура необходима, только если сервер, на который устанавливается сервер фрагментов, никогда раньше не был идентифицирован в кластере. Например, можно пропустить этот шаг, если сервер фрагментов устанавливается на тот же сервер, где установлен первый сервер метаданных. В противном случае, для идентификации сервера в кластере выполните следующую команду:

```
# vstorage -c stor1 auth-node
Please enter password for cluster:
```

Во время исполнения команда попросит ввести пароль для проверки сервера. Введите пароль, который вы указали при установке первого сервера метаданных и нажмите **Enter**. Затем `vstorage` сравнивает введенный пароль с тем, который хранится на сервере метаданных, и если они совпадают, успешно идентифицирует сервер.

5 Если диск, на котором создается сервер фрагментов, не был подготовлен для ПК Р-Хранилище, выполните действия, описанные в разделе **Подготовка дисков для ПК Р-Хранилище** (стр. 15).

6 Смонтируйте подготовленный диск.

Этап 2: Создание сервера фрагментов

Примечание: Для крупных кластеров (см. **Рекомендуемая конфигурация** в *Руководстве по установке ПК Р-Виртуализация*) очень важно правильно настроить область отказов для обеспечения высокой доступности данных. Для получения дополнительной информации см. **Настройка областей отказов** (стр. 31).

Для создания сервера фрагментов используйте команду `vstorage make-cs`, например:

```
# vstorage -c stor1 make-cs -r /vstorage/stor1-cs
```

Данная команда:

1 Создает директорию `/vstorage/stor1-cs`, если она не существует, и настраивает ее для хранения фрагментов данных.

2 Настраивает диск как сервер фрагментов и добавляет его к кластеру `stor1`.

3 Назначает серверу фрагментов уровень хранения по умолчанию. Уровни хранения позволяют держать данные различных типов на разных серверах фрагментов. Чтобы назначить серверу фрагментов определенному уровню, нужно добавить к команде параметр `-t <уровень>`, например, `-t 1` для назначения серверу фрагментов уровню 1. Для получения подробной информации см. **Настройка уровней хранения** (стр. 34).

После создания сервера фрагментов запустите службу управления фрагментами (`vstorage-csd`) и настройте таким образом, чтобы она запускалась при загрузке сервера:

```
# systemctl start vstorage-csd.target
```

После установки первого сервера фрагментов перейдите к созданию остальных серверов фрагментов.

Создание UUID хостов для серверов фрагментов

ПК Р-Хранилище различает хосты, на которых запущены службы сервера фрагментов, по универсально уникальным идентификаторам (UUID), генерируемых в процессе установки. Если вы собираетесь создать новые хосты из образа типа golden, включающего ОС и пакеты ПК Р-Хранилище, то вам потребуется сгенерировать новые UUID хостов вместо того, который был наследован от образа типа golden.

Чтобы создать сервер фрагментов на копии хоста, выполните следующие действия:

1 Убедитесь, что образ типа golden не содержит серверов метаданных, серверов фрагментов или клиентов.

2 Разверните образ типа golden на чистом хосте.

3 Сгенерируйте новый UUID для хоста, чтобы заменить им идентификатор, который был наследован от образа типа golden:

```
# /usr/bin/uuidgen -r | tr '-' ' ' | awk '{print $1$2$3}' > /etc/vstorage/host_id
```

Примечание: Для получения подробной информации об утилите `uuidgen` см. ее man-страницу.

4 Создайте сервер фрагментов на хосте.

Установка клиентов

Процесс установки клиента состоит из трех этапов:

1 Подготовка к монтированию кластера ПК Р-Хранилище к клиенту (стр. 21).

2 Монтирование кластера (стр. 22).

3 Настройка виртуальных машин и контейнеров, которые будут храниться в кластере (стр. 22).

Этап 1: Подготовка к монтированию кластера

Для подготовки монтирования кластера ПК Р-Хранилище к клиенту выполните следующие действия:

1 Войдите в компьютер, который будет использоваться в качестве клиента, в роли пользователя root или пользователя с привилегиями root.

2 Загрузите и установите следующие пакеты RPM: `vstorage-libs-shared` и `vstorage-client`. Пакеты доступны в удаленном хранилище ПК Р-Виртуализация (данное хранилище автоматически настраивается для вашей системы при установке ПК Р-Виртуализация), и их можно установить с помощью следующей команды:

```
# yum install vstorage-client
```

3 Создайте директорию, к которой будет монтирован кластер ПК Р-Хранилище, например:

```
# mkdir -p /vstorage/stor1
```

4 Убедитесь, что в сети настроено обнаружение кластера. Для получения подробной информации см. **Настройка обнаружения кластера** (стр. 9).

5 Выполните идентификацию сервера в кластере. Данная процедура необходима, только если сервер, на который устанавливается клиент, никогда раньше не был идентифицирован в кластере.

Например, можно пропустить этот шаг, если клиент устанавливается на тот же сервер, где установлен первый сервер метаданных или один из серверов фрагментов. В противном случае, для идентификации сервера в кластере выполните следующую команду:

```
# vstorage -c stor1 auth-node
Please enter password for cluster:
```

Во время исполнения команда попросит ввести пароль для проверки сервера. Введите пароль, который вы указали при установке первого сервера метаданных и нажмите **Enter**. Затем `vstorage` сравнивает введенный пароль с тем, который хранится на сервере метаданных, и если они совпадают, успешно идентифицирует сервер.

Этап 2: Монтирование кластера

Далее необходимо монтировать кластер, чтобы он стал доступен клиенту, с помощью команды `vstorage-mount`. Например, если имя кластера ПК Р-Хранилище `stor1`, то для его монтирования к директории `/vstorage/stor1` на клиенте нужно выполнить следующую команду:

```
# vstorage-mount -c stor1 /vstorage/stor1
```

Также можно настроить клиент таким образом, чтобы при загрузке клиента кластер автоматического монтировался к директории `/vstorage/stor1`. Чтобы выполнить данную процедуру, добавьте к файлу `/etc/fstab` следующую строку:

```
vstorage://stor1 /vstorage/stor1 fuse.vstorage rw,nosuid,nodev 0 0
```

Примечание: Если кластер не используется для виртуализации, его можно монтировать при помощи параметра `--fail-on-nospace`. Если в кластере не хватает свободного дискового пространства, будет получено сообщение об ошибке `ERR_NO_SPACE`.

Этап 3: Настройка виртуальных машин и контейнеров

Чтобы настроить сервер с ПК Р-Виртуализация для хранения его виртуальных машин и контейнеров в кластере, выполните следующие действия:

1 Зайдите на сервер в роли пользователя root.

2 Настройте контейнеры для использования в кластере:

1. Проверьте путь к частной зоне контейнера в файле the `/etc/vz/vz.conf`. По умолчанию путь должен быть следующим:

```
VE_PRIVATE=/vz/private/$VEID
```

2. Создайте символьную ссылку от частной зоны контейнера к директории в кластере, где будут храниться контейнеры. Если путь к ней `/vstorage/stor1/private`, создайте эту директорию и выполните следующую команду:

```
# ln -s /vstorage/stor1/private/ /vz/private
```

Примечание: Если директория `/vz/private` уже существует на сервере, удалите ее перед исполнением команды `ln -s`.

3 Настройте виртуальные машины для использования в кластере:

1. Проверьте расположение файлов виртуальной машины по умолчанию:

```
# prlsrvctl info | grep "VM home"
VM home: /vz/vmprivate
```

2. Создайте символьную ссылку от `/vz/vmprivate` к директории в кластере, где будут храниться виртуальные машины. Например, чтобы создать ссылку на директорию `/vstorage/stor1/vmprivate`, выполните следующую команду:

```
# ln -s /vstorage/stor1/vmprivate/ /vz/vmprivate
```

Примечание: Если директория `/vz/vmprivate` уже существует на сервере, удалите ее перед исполнением команды `ln -s`.

Глава 3

Настройка кластеров ПК Р-Хранилище

В данной главе описаны способы настройки кластера ПК Р-Хранилище после его создания.

Настройка серверов метаданных

Для работы кластера ПК Р-Хранилище в нем должно быть запущено и работать большинство серверов метаданных. Для обеспечения высокой доступности кластера необходимо установить в нем, по крайней мере, три сервера метаданных, что позволит пережить потерю одного сервера метаданных. Настройка пяти серверов метаданных обеспечит исправную работу кластера, даже если выключатся два сервера метаданных.

Примечания:

1. При удалении и создании дополнительных серверов метаданных необходимо убедиться, что большинство серверов метаданных в кластере запущено.

2. Неработающие серверы метаданных рекомендуется удалить из кластера как можно скорее (например, сразу после замещения неисправного сервера новым), чтобы все серверы метаданных были запущены и в рабочем состоянии находилось большинство из них. Например, если в кластере работают 3 сервера метаданных, 1 сервер метаданных выходит из строя и на смену ему в кластер добавляется новый сервер метаданных, то общее число серверов метаданных становится 4, один из которых выключен. При сбое еще одного сервера метаданных только 2 сервера метаданных останутся запущенными, и кластер станет недоступен, так как большинство (3 работающих сервера метаданных) не достигнуто.

В данном разделе объясняется, как:

- добавить в кластер новые серверы метаданных (стр. 23)
- удалить из кластера существующие серверы метаданных (стр. 25).

Добавление серверов метаданных

Процедура установки первого сервера метаданных описана в разделе **Установка первого сервера метаданных** (стр. 16). Чтобы настроить второй и все последующие серверы метаданных для кластера, выполните следующие действия:

1 Убедитесь, что вы помните точное имя кластера ПК Р-Хранилище, в который будет добавлен новый сервер метаданных. На примерах ниже используется имя кластера `stor1`.

2 Войдите в компьютер, который будет настроен как сервер метаданных и добавлен в кластер, в роли пользователя root или пользователя с привилегиями root.

3 Загрузите и установите на компьютер следующие пакеты RPM: `vstorage-ctl`, `vstorage-libs-shared` и `vstorage-metadata-server`. Данные пакеты можно установить с помощью следующей команды:

```
# yum install vstorage-metadata-server
```

4 Убедитесь, что в сети настроено обнаружение кластера. Для получения подробной информации см. **Настройка обнаружения кластера** (стр. 9).

5 Идентифицируйте сервер в кластере. Данная процедура необходима, только если физический сервер, на который устанавливается сервер метаданных, никогда раньше не был идентифицирован в кластере. Например, можно пропустить этот шаг, если данный сервер уже настроен как сервер фрагментов или клиент. В противном случае, для идентификации сервера в кластере выполните следующую команду:

```
# vstorage -c stor1 auth-node
Please enter password for cluster:
```

Во время исполнения команда попросит ввести пароль для проверки сервера. Введите пароль, который вы указали при установке первого сервера метаданных и нажмите **Enter**. Затем `vstorage` сравнивает введенный пароль с тем, который хранится на сервере метаданных, и если они совпадают, успешно идентифицирует сервер.

6 Создайте сервер метаданных и добавьте его в кластер с помощью следующей команды:

```
# vstorage -c stor1 make-mds -a 10.30.100.102 -r /vstorage/stor1-mds
```

где

- `stor1` является именем кластера, в который добавляется сервер метаданных.
- 10.30.100.102 является IP-адресом нового сервера метаданных.

  В примере выше замените 10.30.100.101 IP-адресом своего сервера метаданных. Указываемый IP-адрес должен быть (1) статическим (или в случае использования DHCP, преобразованный в MAC-адрес сервера метаданных) и (2) выбранным из диапазона IP-адресов в сети BackNet, которая выделена кластеру ПК Р-Хранилище. Для получения подробной информации см. **Сетевые требования** в *Руководстве по установке ПК Р-Виртуализация*.
- `/vstorage/stor1-mds` является путем к журналу, в котором будет храниться информация о кластере `stor1`. При выборе директории для хранения журнала убедитесь, чтобы в разделе, в котором будет находиться данная директория, было как минимум 10 ГБ свободного дискового пространства.

Если в сети не настроено обнаружение кластера с помощью записей DNS или Zeroconf, необходимо дополнительно использовать параметр `-b` и указать IP-адреса первого сервера метаданных (и всех остальных серверов метаданных, если в кластере их больше одного) при выполнении следующей команды:

```
# vstorage -c stor1 make-mds -a 10.30.100.102:2510 -r /vstorage/stor1-mds -b \
10.30.100.101
```

7 Запустите службу управления сервером метаданных (`vstorage-mdsd`) и настройте ее таким образом, чтобы она запускалась при загрузке сервера метаданных:

```
# systemctl start vstorage-mdsd.target
```

Для получения инструкций по проверке, успешно ли прошла настройка сервера метаданных для кластера, см. **Мониторинг кластеров ПК Р-Хранилище** (стр. 84).

Удаление серверов метаданных

Если требуется удалить сервер метаданных из кластера ПК Р-Хранилище (например, для обновления сервера или выполнения некоторых задач обслуживания), выполните следующие действия:

1. Настройте новый сервер метаданных для замены сервера, который будет удален из кластера. Для получения инструкций см. **Создание дополнительных серверов метаданных** (стр. 23).
2. Узнайте порядковый номер сервера метаданных, чтобы его удалить, введя следующую команду на некоторых серверах метаданных или клиентах:

```
# vstorage -c stor1 top
```

   Данная команда отображает подробную информацию о кластере `stor1`. Найдите секцию с информацией о серверах метаданных, например:

```
...
MDSID   STATUS   %CTIME   COMMITS   %CPU   MEM   UPTIME   HOST
M   1    avail     0.0%       0/s   0.0%   64m   17d 6h   10.30.17.38
    2    avail     0.0%       0/s   0.0%   50m   12d 3h   10.30.45.12
    3    avail     0.0%       0/s   0.0%   57m    7d 1h   10.30.10.15
...
```

   Порядковый номер показан в колонке **MDSID**. В вышеприведенном примере в кластере `stor1` настроено три сервера метаданных с порядковыми номерами 1, 2 и 3.

3. Удалите сервер метаданных из кластера. Например, чтобы удалить сервер метаданных с порядковым номером 3, введите команду:

```
# vstorage -c stor1 rm-mds 3
```

Настройка серверов фрагментов

В данном разделе объясняется, как можно выполнить следующие задачи:

- увеличить дисковое пространство в кластере, добавив в него новые серверы фрагментов (стр. 25);
- удалить серверы фрагментов из кластера для выполнения с ними задач обслуживания (стр. 26).

Добавление серверов фрагментов для увеличения дискового пространства

Увеличить размер дискового пространства в кластере ПК Р-Хранилище можно, просто добавив в него новые серверы фрагментов. Для получения подробной информации см. **Установка серверов фрагментов** (стр. 18).

Примечание: ПК Р-Хранилище можно настроить для поддержки как минимум 8 ПБ свободного дискового пространства, что означает, до 24 ПБ физического дискового пространства в случае зеркалирования с 3 копиями.

Удаление серверов фрагментов

Если требуется удалить сервер фрагментов из кластера (например, для выполнения некоторых задач обслуживания), выполните следующие действия:

1 Убедитесь, что

- количество серверов фрагментов, настроенных для кластера, достаточно для хранения необходимого числа фрагментов данных (а именно, равно или превышает текущее значение репликации);
- на серверах фрагментов достаточно свободного дискового пространства для хранения фрагментов данных. Для получения подробной информации см. **Мониторинг серверов фрагментов** (стр. 87).

2 Узнайте порядковый номер сервера фрагментов, чтобы его удалить, введя следующую команду на некоторых серверах кластера:

```
# vstorage -c stor1 top
```

Данная команда отображает подробную информацию о кластере `stor1`. Найдите секцию с информацией о серверах фрагментов, например:

```
...
CSID   STATUS   SPACE   FREE  REPLICAS  IOWAIT  IOLAT(ms)  QDEPTH   HOST
1025   active   105GB   88GB        40      0%        0/0     0.0   10.30.17.38
1026   active   105GB   88GB        40      0%        0/0     0.0   10.30.18.40
1027   active   105GB   99GB        40      0%        0/0     0.0   10.30.21.30
1028   active   107GB  101GB        40      0%        0/0     0.0   10.30.16.38
...
```

Порядковый номер серверов фрагментов показан в колонке **CSID**. В вышеприведенном примере в кластере `stor1` настроено четыре серверы фрагментов с порядковыми номерами 1025, 1026, 1027 и 1028.

3 Удалите сервер фрагментов из кластера. Например, чтобы удалить сервер фрагментов с порядковым номером 1028, введите команду:

```
# vstorage -c stor1 rm-cs --wait 1028
```

Сразу после инициации операции удаления кластер начинает реплицировать фрагменты данных, хранившиеся на удаленном сервере, и помещает их на оставшиеся серверы фрагментов. Если указан параметр `--wait`, команда будет ждать завершения операции (что может занять длительное время).

Примечания:

1. Если диск сервера фрагментов исчез из оперативной системы и сервер фрагментов не имеет доступа к его хранилищу, фрагменты данных, которые должны быть реплицированы, будут недоступны и удаление сервера фрагментов не будет завершено. В этом случае придется принудительно удалить сервер фрагментов из кластера с помощью параметра `-f`. **Внимание:**

Данную операцию рекомендуется выполнять, только если сервер фрагментов безвозвратно потерян. Не следует удалять активные серверы фрагментов с помощью параметра `-f`.

2. Во время операции удаления можно отменить ее, используя команду `vstorage -c stor1 rm-cs -- cancel 1028`. Данная команда может быть полезна, например, если вы указали неверный ID сервера фрагмента для удаления.

Чтобы добавить удаленный сервер фрагментов обратно в кластер, нужно установить его с нуля, следуя инструкции в разделе **Установка серверов фрагментов** (стр. 18).

Настройка клиентов

В данном разделе объясняется, как можно выполнить следующие задачи:

- добавить в кластеры новые клиенты (стр. 27);
- обновить клиентов (стр. 27);
- удалить клиентов из кластеров (стр. 27).

Добавление клиентов

Процесс включения дополнительных клиентов в кластер ПК Р-Хранилище аналогичен процессу установки первого клиента и подробно описан в разделе **Установка клиентов** (стр. 21).

После настройки клиента можно управлять им с помощью различных команд. Например, можно контролировать состояние и статус кластера, используя команду `vstorage top`. Для получения дополнительной информации по мониторингу кластеров ПК Р-Хранилище см. **Мониторинг кластеров ПК Р-Хранилище** (стр. 84).

Обновление клиентов

Процесс обновления программного обеспечения клиентов аналогичен процессу обновления ПО на автономных серверах, за исключением обновления пакета `vstorage-client`. При обновлении данного пакета нужно обратить внимание на следующее:

- Если ни один кластер не подключен к клиенту, то клиент сразу начинает использовать обновленный пакет.
- Если хоты бы один кластер подключен к клиенту, обновленный пакет устанавливается, но клиент начинает его использовать только после повторного подключения кластера или перезагрузке клиента.

Удаление клиентов

Удаление клиента из кластера ПК Р-Хранилище означает размонтирование директории на клиенте, в которую монтирован кластер. Например, если кластер монтирован в директорию `/vstorage/stor1`, то можно размонтировать его следующим образом:

1 Убедитесь, что все виртуальные машины и контейнеры в кластере остановлены.

2 Размонтируйте кластер:

```
# umount /vstorage/stor1
```

3 Если кластер настроен на автоматическое монтирование при загрузке клиента, введите комментарий в записи кластера в файле `/etc/fstab` на клиенте:

```
# vstorage://stor1 /vstorage/stor1 fuse.vstorage rw,nosuid,nodev 0 0
```

Настройка высокой доступности

Высокая доступность обеспечивает работу виртуальных машин, контейнеров и целей iSCSI даже при отказе физического сервера, на котором они находятся. Доступны три режима:

- DRS (по умолчанию). В данном режиме виртуальные машины и контейнеры, которые были запущены на отказавшем сервере, перемещаются на исправные серверы на основе доступной свободной памяти и количества виртуальных сред, разрешенного лицензией. Данный режим может использоваться для серверов, на которых запущена служба `pdrs`.
- Round-robin (запасной вариант по умолчанию). В данном режиме виртуальные машины, контейнеры и цели iSCSI с отказавшего сервера перемещаются на исправные серверы в порядке круговой очереди.
- Spare. В данном режиме виртуальные машины и контейнеры с отказавшего сервера перемещаются на свободный сервер — пустой сервер, имеющий достаточно ресурсов и подходящую лицензию, позволяющие разместить все виртуальные среды с любого сервера в кластере. Такой сервер требуется для работы высокой доступности в данном режиме.

Для получения информации по настройке высокой доступности и переключению между ее режимами см. раздел **Управление кластерами высокой доступности** в *Руководстве пользователя по ПК Р-Виртуализация*.

Управление параметрами кластера

В данном разделе дается определение параметрам кластера и объясняется, как их настраивать с помощью утилиты `vstorage`.

Обзор параметров кластера

Параметры кластера влияют на создание, размещение и управление репликами фрагментов данных в кластере ПК Р-Хранилище. Все параметры можно разделить на три основные группы: параметры репликации, кодирования, размещения.

Каждый параметр кластера кратко описан в таблице ниже. Для получения дополнительной информации по параметрам и их настройке см. следующие подразделы.

Параметр	Описание
Параметры репликации	
Normal Replicas	Количество создаваемых реплик для фрагмента данных, от 1 до 15. Рекомендуется: 3.
Minimum Replicas	Минимальное число создаваемых реплик для фрагмента данных, от 1 до 15. Рекомендуется: 2. Данный параметр не является обязательным.
Параметры размещения	
Failure Domain	Политика размещения реплик, может быть `room`, `row`, `rack`, `host` (по умолчанию) или `disk` (сервер фрагментов).
Tier	Уровни хранения, от 0 до 3 (0 по умолчанию).

Настройка параметров репликации

Параметры репликации кластера определяют следующее:

- Нормальное число реплик для фрагмента данных. При создании нового фрагмента данных ПК Р-Хранилище автоматически реплицирует его до тех пор, пока не будет создано нормальное число реплик.
- Минимальное число реплик для фрагмента данных (необязательно). В течение жизненного цикла фрагмента данных число его реплик может изменяться. Например, в случае выхода из строя компонента кластера, число копий фрагмента может даже иногда уходить ниже заданного минимума. В таком случае все операции записи в реплики приостанавливаются до тех пор, пока нормальное число реплик не будет соответствовать указанному значению. Кластер постоянно пытается поддерживать количество реплик в нормальном значении путем фоновой репликации

Проверить, какие параметры репликации применяются к кластеру, можно при помощи команды `vstorage get-attr`. Например, если кластер монтирован к директории `/vstorage/stor1`, можно ввести следующую команду:

```
# vstorage get-attr /vstorage/stor1
connected to MDS#1
File: '/vstorage/stor1'
Attributes:
...
replicas=1:1
...
```

На примере выше нормальное и минимальное число реплик равно 1.

Изначально любой кластер настроен таким образом, что для каждого фрагмента данных создается одна реплика. Одной реплики достаточно для оценки функциональности ПК Р-Хранилище при использовании только одного сервера. Однако в рабочей среде для обеспечения высокой доступности данных рекомендуется:

- настроить, чтобы для каждого фрагмента данных создавалось, по крайней мере, три реплики;
- задать минимальное число реплик равное 2.

Для подобной конфигурации необходима установка как минимум трех серверов фрагментов в кластере.

Чтобы текущие параметры репликации применялись ко всем виртуальным машинам и контейнерам в кластере, нужно запустить команду `vstorage set-attr` в той директории, к которой монтирован кластер. Например, чтобы задать рекомендуемые значения репликации для кластера `stor1`, монтированного к `/vstorage/stor1`, укажите нормальное число реплик для кластера равное 3:

```
# vstorage set-attr -R /vstorage/stor1 replicas=3
```

Минимальное число реплик будет автоматически задано равное 2 по умолчанию.

Примечание: Для получения информации о том, как правильно вычислить минимальное число реплик, см. man-страницу по `vstorage set-attr`.

Применять параметры репликации можно не только к целому кластеру, но и к отдельным директориям и файлам. Например:

```
# vstorage set-attr -R /vstorage/stor1/private/MyCT replicas=3
```

Настройка параметров кодирования

Для обеспечения избыточности данных вместо репликации ПК Р-Виртуализация может использовать избыточное кодирование. С помощью него ПК Р-Виртуализация разбивает входящий поток данных на фрагменты определенного размера, затем разделяет каждый фрагмент на определенное число (M) блоков размером 1 МБ и создает определенное число (N) блоков четности для избыточности. Все блоки распределяются между серверами M+N, то есть, по одному блоку на каждый сервер. Блоки хранятся на серверах в обычных фрагментах, которые не реплицируются, так как избыточность уже достигнута. Кластер может пережить отказ серверов N без потери данных.

Значения M и N указаны в именах режимов кодирования. Например, в режиме 5+2 входящий трафик разбивается на фрагменты размером 5 МБ, каждый фрагмент разделяется на пять блоков размером 1 МБ, и еще два блока четности размером 1 МБ добавляются для избыточности. Дополнительно, если N равно 2, данные кодируются с помощью схемы RAID6, а если N больше 2, то используются коды избыточности Рида-Соломона.

Следует использовать следующие режимы избыточного кодирования (M+N):

- 1+0,
- 3+2,
- 5+2,
- 7+2,
- 17+3.

Кодирование настраивается для директорий. Например:

```
# vstorage set-attr -R /vstorage/stor1 encoding=5+2
```

После включения кодирования режим избыточности невозможно сменить обратно на репликацию. Однако можно переключаться между разными режимами кодирования для одной директории.

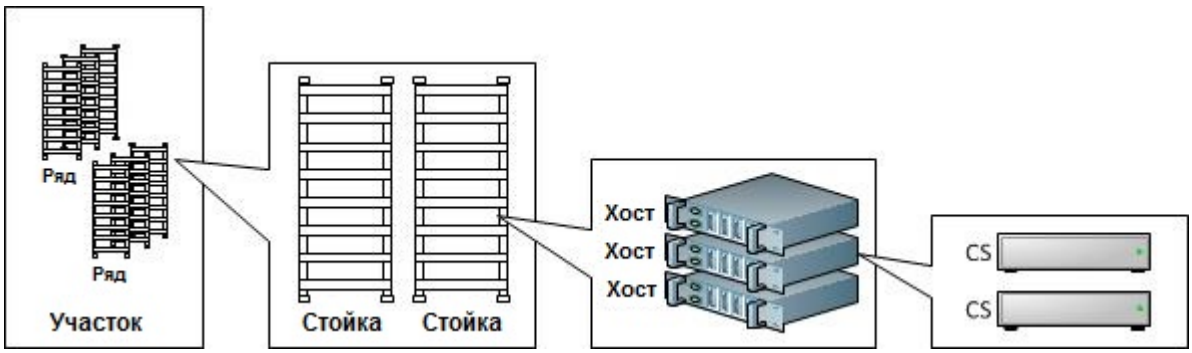
Настройка областей отказов

Область отказов представляет собой набор служб, которые могут отказать коррелированным образом. В связи с коррелированными отказами необходимо, чтобы реплики данных находились в разных областях отказа для обеспечения доступности данных. Примеры областей отказов включают:

- Один диск (самая малая возможная область отказов). В этом случае ПК Р-Хранилище никогда не помещает больше одной реплики данных на один диск или сервер фрагментов.
- Один хост с запущенными службами сервера фрагментов. При отказе такого хоста (например, из-за отключения питания или потери сетевого соединения) все службы сервера фрагментов, находящиеся на нем, сразу становятся недоступны. По этой причине ПК Р-Хранилище настраивается по умолчанию таким образом, чтобы больше одной реплики фрагментов данных не хранилось на одном хосте (см. **Определение областей отказов** ниже).
- Крупномасштабные кластерные установки с большим числом стоек имеют дополнительные точки отказа в виде коммутаторов и элементов питания для каждой стойки. В данном случае необходимо настроить ПК Р-Хранилище для хранения реплик данных во всех подобных областях отказов для обеспечения доступности данных и предотвращения крупных коррелированных отказов одной области.

Топология областей отказов

Каждому компоненту службы ПК Р-Хранилище назначена топологическая информация. Топологические пути определяют логическое дерево физического расположения компонентов, состоящего из 5 идентификаторов: `room.row.rack.host_ID.cs_ID`.



Первые три компонента топологического пути `room.row.rack` можно настроить в файлах конфигурации `/etc/vstorage/location` (для дополнительной информации см. man-страницу по `vstorage-config-files`). Последние два компонента `host_ID.cs_ID` генерируются автоматически:

- `host_ID` является уникальным, случайно генерируемым идентификатором хоста, который создается во время установки и хранится в `/etc/vstorage/host_id`;
- `cs_ID` является уникальным идентификатором службы, который генерируется при создании сервера фрагментов.

Примечание: Для просмотра текущей топологии служб и дискового пространства, доступного для каждого местоположения введите команду `vstorage top` и нажмите **w**.

Определение областей отказов

Основываясь на уровнях иерархии, описанных выше, можно использовать команду `vstorage set-attr`, чтобы определить области отказов для правильного выделения реплики файлов:

```
# vstorage -c <cluster_name> set-attr -R -p /failure-domain=<disk|host|rack|row|room>
```

где

- `disk` означает, что только 1 реплика выделяется для каждого диска или сервера фрагментов (не рекомендуется),
- `host` означает, что только 1 реплика выделяется для каждого хоста (по умолчанию),
- `rack` означает, что только 1 реплика выделяется для каждой стойки,
- `row` означает, что только 1 реплика выделяется для каждого ряда,
- `room` означает, что только 1 реплика выделяется для каждого участка.

Рекомендуется использовать одинаковую конфигурацию для всех файлов в кластере для упрощения анализа и меньшей вероятности возникновения ошибок.

В качестве примера для настройки установки с 5-ю стойками выполните следующие действия:

1. Назначьте 0.0.1 для `/etc/vstorage/location` на всех хостах из первой стойки, 0.0.2 на всех хостах из второй стойки и т.д.
2. Создайте 5 серверов метаданных: 1 на любом хосте из первой стойки, 1 на любом хосте из второй стойки и т.д.
3. Настройте ПК Р-Хранилище таким образом, чтобы на каждой стойке создавалась только 1 реплика (предполагая, что имя кластера `stor1`):

```
# vstorage -c stor1 set-attr -R -p /failure-domain=rack
```

4. Создайте службы сервера фрагментов с помощью команды `vstorage-make-cs`, как описано в разделе **Установка серверов фрагментов** (стр. 18).

Изменение расположения хоста

Как только хост запускает службы в кластере, их топология запоминается на сервер метаданных и не может быть изменена. Все новые службы, создаваемые на данном

хосте, будут использовать эту кэшированную информацию. Если требуется изменить информацию о местоположении хоста, выполните следующие действия:

1. Завершите работу служб сервера фрагментов и клиента, запущенных на хосте, и удалите их с помощью команд `vstorage-rm-cs` и `umount`, как описано в разделах **Удаление серверов фрагментов** (стр. 26) и **Удаление клиентов** (стр. 27) соответственно.
2. Задайте `/etc/vstorage/host_id` другой уникальный ID, например, сгенерированный при помощи `/dev/urandom`.
3. Настройте `/etc/vstorage/location` в соответствии с требованиями.
4. Создайте заново службы сервера фрагментов и клиента: монтируйте кластер и создайте новые службы сервера фрагментов с помощью команды `vstorage-make-cs` ,как описано в разделах **Установка серверов фрагментов** (стр. 18) и **Установка клиентов** (стр. 21).

Рекомендации по областям отказов

Важно: Не рекомендуется использовать область отказа `disk` одновременно с журналирующими SSD-дисками. В этом случае, некоторое число реплик может оказаться на дисках, обслуживаемых одним и тем же журналирующим SSD-диском. При отказе данного SSD-диска все реплики, которые зависят от находящихся на нем журналов, будут потеряны. В результате ваши данные могут быть потеряны.

- Рекомендуется иметь, по крайней мере, 5 областей отказов, настроенных в рабочей среде (хосты, стойки и др.) и свободное дисковое пространство в каждой области отказов, чтобы в случае отказа области, ее можно было восстановить.
- При создании служб сервера метаданных необходимо вручную настроить топологию и области отказов. То есть в установках с большим количеством стоек серверы метаданных должны создаваться в разных стойках (всего 5 серверов метаданных).
- Для установок с большим количеством стоек рекомендуется как минимум 3 реплики.
- Полное отсутствие распределения данных на дисках более критично для огромных областей отказов. Например, если в области есть 5 стоек с 10 ТБ, 20 ТБ, 30 ТБ, 100 ТБ и 100 ТБ общего дискового пространства, то выделить (10+20+30+100+100)/3 = 86 ТБ данных в 3 репликах будет невозможно. Можно будет выделить только 60 ТБ, так как место в малообъемных стойках закончится быстрее и для выделения данных не будет 3 доступных областей, в то время как в крупных стойках (с 100ТБ) останется свободное дисковое пространство.
- При отказе и выключении крупной области отказов ПК Р-Хранилище по умолчанию не выполняет восстановление данных, так как репликация большого объема данных может занять больше времени, чем восстановление области отказов. Такое поведение задается глобальным параметром `mds.wd.max_offline_cs_hosts` (настраиваемым с помощью `vstorage-config`), контролирующим количество отказавших хостов, которое считается обычным отказом и будет восстановлено в автоматическом режиме.
- Области отказов должны иметь похожую производительность ввода-вывода во избежание дисбаланса. Например, не следует создавать установки, где в параметре `failure-domain` задано `rack`, все, кроме одной, стойки имеют по 10 серверов, а одна

стойка имеет только 1 сервер. ПК Р-Хранилище придется непрерывно сохранять реплику на данный сервер, что приведет к ухудшению общей производительности.

- В зависимости от глобального параметра `mds.alloc.strict_failure_domain` (настраиваемого с помощью `vstorage-config`) политика областей может быть жесткой (по умолчанию) или консультативной. Настраивать данный параметр настоятельно не рекомендуется, если вы не уверены в правильности своих действий.

Использование уровней хранения

В данном разделе описываются уровни хранения, используемые в кластерах ПК Р-Хранилище, а также их настройка и мониторинг.

Что такое уровни хранения

Уровни хранения представляют собой способ организации дискового пространства. Они могут использоваться для хранения различных категорий данных на разных серверах фрагментов. Например, можно использовать высокоскоростные твердотельные накопители для хранения данных, отвечающих за производительность, вместо кэширования операций кластера.

Настройка уровней хранения

Чтобы назначить уровню хранения дисковое пространство, выполните следующие действия:

1 Назначьте одному уровню все серверы фрагментов с SSD-дисками. Данную процедуру можно выполнить при установке сервера фрагментов; см. **Создание сервера фрагментов** (стр. 20) для получения подробной информации.

> **Примечание:** Для получения информации о рекомендуемых SSD-дисках, см. **Использование SSD-дисков** (стр. 110).

2 Назначьте уровню 1 директории и файлы, к которым происходит частое обращение, с помощью команды `vstorage set-attr`. Например:

```
# vstorage set-attr -R /vstorage/stor1/private/MyCT tier=1
```

Данная команда рекурсивно назначает директорию `/vstorage/stor1/private/MyCT` и ее содержимое уровню 1.

При назначении уровню дискового пространства нужно иметь в виду, что диски с более высокой скоростью нужно назначать более высоким уровням. Например, уровень 0 можно использовать для резервных копий и других «холодных» данных (сервер фрагментов без SSD-журналов), уровень 1 – для виртуальных сред: много «холодных» данных, но быстрая произвольная запись (сервер фрагментов с SSD-журналами), уровень 2 – для «горячих» данных (сервер фрагментов на SSD-диске), журналов, кэша, отдельных дисков виртуальных машин и т.п.

Данная рекомендация относится к работе ПК Р-Хранилище с дисковым пространством. Если на уровне хранения закончилось свободное место, ПК Р-Хранилище попробует временно использовать более низкие уровни, а если они также заполнены, то более

высокий уровень. Если позже добавить дисковое пространство исходному уровню, данные, которые временно хранятся в другом месте, будут перемещены на исходный уровень, где и должны были храниться изначально.

Например, если уровень 2 заполнен, то при попытке записать на него данные ПК Р-Хранилище попробует записать их на уровень 1, затем на уровень 0, а потом на уровень 3. Если позже добавить дополнительное дисковое пространство к уровню 2, то эти данные, в данный момент хранящиеся на уровне 1, либо 0, либо 3, будут перемещены обратно на уровень 2, где они должны были храниться с самого начала.

Распределение данных

Чтобы повысить производительность ввода-вывода серверов фрагментов в кластере, ПК Р-Хранилище автоматически распределяет загрузку серверов фрагментов, перемещая «горячие» фрагменты данных с «горячих» серверов фрагментов на «холодные».

Сервер фрагментов считается «горячим», если глубина его очереди запросов превосходит среднее значение в кластере на 40% или более (см. пример ниже). В отношении фрагментов данных, «горячие» означает «часто запрашиваемые».

Глубина очереди запросов серверов фрагментов обозначается параметром `QDEPTH` в выводе команд `vstorage top` и `vstorage stat`. Например:

```
...
IO QDEPTH: 0.1 aver, 1.0 max; 1 out of 1 hot CS balanced    46 sec ago
...
CSID STATUS SPACE  AVAIL REPLICAS UNIQUE IOWAIT IOLAT(ms)QDEPTH HOST      BUILD_VERSION
1025 active 1007.3 156.8G    7142      0    10%    1/117    0.3 10.31.240.167 6.0.11-10
1026 active 1007.3 156.8G    7267      0    11%    0/225    0.1 10.31.240.167 6.0.11-10
1027 active 1007.3 156.8G    7151      0     2%     0/10    0.1 10.31.240.167 6.0.11-10
1028 active 1007.3 156.8G    7285      0    13%    1/141    1.0 10.31.240.167 6.0.11-10
...
```

Строка `IO QDEPTH` отображает среднее и максимальное значение глубины очереди запросов во всем кластере за последние 60 секунд. Колонка `QDEPTH` показывает средние значения глубины очереди запросов для каждого сервера фрагментов за последние 5 секунд.

Каждые 60 секунд самый «горячий» фрагмент данных переносится с «горячего» сервера фрагментов на сервер с более короткой очередью запросов.

Мониторинг уровней хранения

Мониторинг дискового пространства, назначенного каждому уровню хранения, можно осуществить при помощи утилиты `top` в режиме расширенного вывода (включается нажатием **v**). Пример вывода приведен ниже:

```
Cluster 'tiers': healthy
Space: [OK] allocatable 3.3TB of 3.5TB, 3.5TB total, 3.5TB free
 tier 0: allocatable 1.6TB of 1.7TB, 1.7TB total, 1.7TB free
 tier 1: allocatable 866GB of 912GB, 912GB total, 912GB free
 tier 3: allocatable 866GB of 912GB, 912GB total, 912GB free
MDS nodes: 1 of 1, epoch uptime:  3 min
CS nodes:  4 of 4 (4 avail, 0 inactive, 0 offline)
Replication:  1 norm,  1 limit
Chunks: [OK] 0 healthy,  0 standby,  0 degraded,  0 urgent,
             0 blocked,  0 pending,  0 offline,  0 replicating,
             0 overcommitted,  0 deleting,  0 void
FS:  0B in 1 files, 0 inodes,  0 file maps,  0 chunks,  0 chunk
IO:       read     0B/s (  0ops/s), write     0B/s (  0ops/s)
IO total: read       0B (    0ops), write       0B (    0ops)
Repl IO:  read     0B/s, write:     0B/s
Sync rate:   0ops/s, datasync rate:   0ops/s
```

Изменение сети кластера ПК Р-Хранилище

Перед перемещением кластера в новую сеть следует иметь в виду следующее:

- При смене сети кластер находится в нерабочем состоянии в течение краткого периода времени, когда половина серверов метаданных недоступны.
- Перед сменой сети кластера настоятельно рекомендуется сделать резервные копии всех хранилищ метаданных.

Чтобы поменять сеть кластера ПК Р-Хранилище, выполните следующие действия на каждом сервере кластера, на котором запущена служба сервера метаданных:

1 Остановите службу сервера метаданных:

```
# systemctl stop vstorage-mdsd.target
```

2 Укажите новые IP-адреса для всех серверов метаданных в кластере командой `vstorage configure-mds -r <MDS_repo> -n <MDS_ID@new_IP_address> \ [:port] [-n...]`, где:

- `<MDS_repo>` является путем к хранилищу метаданных на данном сервере;
- каждая пара `<MDS_ID@new_IP_address>` является идентификатором сервера метаданных и соответствующим новым IP-адресом. Например, для кластера с 5 серверами метаданных:

```
# vstorage -c stor1 configure-mds -r /vstorage/stor1-cs1/mds/data -n 1@10.10.20.1 \
-n 2@10.10.20.2 -n 3@10.10.20.3 -n 4@10.10.20.4 -n 5@10.10.20.5
```

Примечания:

1. Узнать идентификатор сервера метаданных и путь к хранилищу метаданных можно при помощи команды `vstorage list-services -M`.

2. Если порт не указывается, то по умолчанию будет использоваться порт 2510.

3 Запустите службу сервера метаданных:

```
# systemctl start vstorage-mdsd.target
```

Включение онлайн сжатия для виртуальных машин

Онлайн сжатие виртуальных машин в ПК Р-Хранилище в режиме репликации позволяет высвободить дисковое пространство, которое больше не используется, с помощью флага FALLOC_FL_PUNCH_HOLE. При онлайн сжатии происходит запуск команды TRIM внутри гостевой ОС. В гостевой ОС Windows данная функция включена по умолчанию, а в Linux guests она включается при установке гостевых инструментов.

Примечание: Онлайн сжатие работает по умолчанию, пока не поставлен флаг discard для unmap для дисков виртуальной машины.

Чтобы включить онлайн сжатие для кластера ПК Р-Хранилище, выполните следующие действия:

1 Обновите серверы кластера до последней версии ПК Р-Виртуализация.

2 Перезагрузите обновленные серверы по одному.

3 Выполните следующую команду на любом сервере кластера:

```
# vstorage set-config "gen.do_punch_hole=1"
```

Внимание: Выполнение команды перед обновлением всех серверов фрагментов приведет к потере данных!

Примечание: Чтобы высвободить неиспользуемое пространство, накопившееся до включения онлайн сжатия (например, от виртуальных сред, которые были созданы в ПК Р-Виртуализация более ранних версий), создайте файл внутри виртуальной машины с размером неиспользуемого пространства, и затем удалите его.

Управление лицензиями ПК Р-Хранилище

В данном разделе описывается процесс управления лицензиями ПК Р-Хранилище, а именно, как можно выполнить следующие действия:

- установить новую лицензию для кластера ПК Р-Хранилище (стр. 37).
- просмотреть содержимое установленной лицензии (стр. 38).
- проверить статус лицензии (стр. 39).

Установка лицензии

Кроме установки лицензий ПК Р-Виртуализация на все клиенты в кластере, необходимо также установить отдельную лицензию ПК Р-Хранилище. Для одного кластера требуется одна лицензия. Лицензию можно установить с любого сервера кластера: сервера метаданных, сервера фрагментов или клиента.

Для установки лицензии введите команду `vstorage load-license` и полный путь к файлу лицензии. Например:

```
# vstorage -c stor1 load-license -f /etc/storlicense
```

Просмотр содержимого лицензии

С помощью команды `vstorage view-license` можно просмотреть информацию о лицензиях, которые в данный момент установлены в кластере. Во время выполнения утилита обрабатывает лицензию и отображает на экране ее содержимое. Пример вывода `vstorage view-license` приводится ниже:

```
# vstorage -c stor1 view-license
HWID: XXXX.XXXX.XXXX.XXXX.XXXX.XXXX.XXXX.XXXX
RSTOR
status="ACTIVE"
version=1.0
expiration="08/24/2012 19:59:59"
graceperiod=3600
key_number="PCSS.XXXXXXXX.XXXX"
platform="Linux"
product="RSTOR"
gracecapacity=5
autorecovery=0
autorebalance=0
snapshots=1
capacity=500
replicas=5
```

Основные параметры лицензии показаны в таблице ниже.

Параметр	Описание
HWID	Идентификатор кластера.
status	Статус лицензии. Для получения дополнительной информации см. **Проверка статуса лицензии** (стр. 39).
version	Версия ПК Р-Хранилище, для которой была выпущена лицензия.
expiration	Дата и время окончания срока действия лицензии.
graceperiod	Период времени, в секундах, в течение которого ПК Р-Хранилище будет работать после окончания срока действия.
key_number	Номер ключа, под которым зарегистрирована лицензия на сервере ПК Р-Виртуализация Key Authentication.
platform	Операционная система, с которой совместима лицензия.
product	Продукт, для которого была выпущена лицензия.
gracecapacity	Дополнительный размер дискового пространства, который могут занимать фрагменты данных в кластере, процентное отношение к значению параметра **capacity**. Например, если значение параметра **capacity** 1 ТБ, а значение **gracecapacity** 5%, то фрагменты данных могут использовать на 50 ГБ больше максимальной емкости.
capacity	Общий размер дискового пространства, в ГБ, который могут занимать фрагменты данных в кластере. Для просмотра дискового пространства, используемого фрагментами данных,

	используйте команду vstorage top, нажмите клавишу **v** и проверьте поле **FS**. Для получения информации см. **Использование дискового пространства** (стр. 88).
replicas	Максимальное число реплик, которое могут иметь фрагменты данных.
autorecovery	Показывает, включена (1) или отключена (0) функция автоматического восстановления.
autorebalance	Показывает, включена (1) или отключена (0) функция автоматического перераспределения.
snapshots	Показывает, включена (1) или отключена (0) функция снапшотов.

Проверка статуса лицензии

Проверить статус лицензии можно одним из следующих способов:

- С помощью `vstorage view-license`, например:

```
# vstorage -c stor1 view-license | grep status
status="ACTIVE"
```

- С помощью `vstorage stat` или `vstorage top`, например:

```
# vstorage -c stor1 stat | grep License
connected to MDS#1
License: STORS.XXXXXXXX.XXXX is ACTIVE
```

В таблице ниже показаны все возможные статусы лицензии.

Статус	Описание
ACTIVE	Лицензия действительна и успешно установлена в кластер.
VALID	Лицензия действительна и может быть установлена в кластер.
EXPIRED	Срок действия лицензии истек.
GRACED	Лицензии предоставлен период отсрочки, или фрагменты данных используют дисковое пространство из дополнительной емкости.
INVALID	Лицензия недействительна (например, если срок ее действия еще не начался).

Завершение работы кластеров ПК Р-Хранилище

Для полного завершения работы кластера ПК Р-Хранилище выполните следующие действия:

1 Завершите работу всех клиентов в кластере. Для этого на каждом клиенте:

1. Остановите все запущенные контейнеры и виртуальные машины.
2. Размонтируйте файловую систему кластера с помощью команды `umount`. Например, если кластер монтирован в директорию `/vstorage/stor1` на клиенте, можно размонтировать его следующим образом:

```
# umount /vstorage/stor1
```

3. Отключите автоматическое монтирование кластера, удалив запись кластера из файла `/etc/fstab`.

2 Остановите службу shaman и отключите ее автоматический запуск:

```
# systemctl stop shaman.service
# systemctl disable shaman.service
```

3 Остановите все серверы метаданных и отключите их автоматический запуск:

```
# systemctl stop vstorage-mdsd.target
# systemctl disable vstorage-mdsd.target
```

4 Остановите все серверы фрагментов и отключите их автоматический запуск:

```
# systemctl stop vstorage-csd.target
# systemctl disable vstorage-csd.target
```

Глава 4

Экспорт данных кластера ПК Р-Хранилище

Обычно доступ к виртуальным машинам и контейнерам осуществляется с клиентов, на которых установлен продукт ПК Р-Виртуализация при помощи стандартной утилиты для командной строки, такой как `prlctl`. Другим способом удаленного доступа к данным кластеров ПК Р-Хранилище является их экспорт в виде

- NFS (на основе ploop),
- образов через iSCSI или
- объектного хранилища типа S3.

Подробное описание данных методов можно найти далее в этой главе.

Доступ к кластерам ПК Р-Хранилище через NFS

Для доступа к кластеру ПК Р-Хранилище через NFS необходимо выполнить следующие действия:

1. Создайте и монтируйте ploop с файловой системой ext4.
2. Установите хранилище NFS с помощью стандартной команды `exportfs` или файла `/etc/exports`.
3. Войдите в общую папку NFS на удаленном компьютере.

Подробное описание данных шагов см. ниже.

Подготовка ploop

Виртуальные блочные устройства ploop позволяют подключить любой файл ПК Р-Хранилище в виде блочного устройства и отформатировать его в стандартную файловую систему, такую как ext4. Так как решение ПК Р-Хранилище не оптимизировано для файлов небольшого размера и не использует POSIX-совместимую файловую систему, ploop с ext4 можно использовать только для вышеуказанных функций.

Для подготовки ploop выполните следующие действия:

1. Загрузите необходимые модули:

```
# modprobe ploop pfmt_ploop1 pio_kaio
```

2 Создайте ploop:

```
# mkdir /vstorage/ploop0
# ploop init -s 1g -t ext4 /vstorage/ploop0/img0
```

Данная команда создает plop объемом 1 ГБ с файловой системой ext4.

3 Монтируйте ploop:

```
# mkdir /mnt/ploop0
# ploop mount -m /mnt/ploop0 /vstorage/ploop0/DiskDescriptor.xml
```

4 (Необязательно) Установите непрерывное монтирование ploop с помощью `/etc/fstab`:

```
# cat >> /etc/fstab <<EOF
/vstorage/ploop0/DiskDescriptor.xml            /mnt/ploop0   ploop         defaults 0 0
EOF
```

Установка хранилища NFS

Примечание: Для данного примера можно предположить, что:

1. IP-адрес исходного компьютера 192.168.0.1, а IP-адрес конечного компьютера 192.168.0.2.

2. Для установки хранилища NFS используется команда `exportfs`

3. На конечном компьютере хранилище NFS монтировано к директории `/nfsshare`.

Чтобы предоставить доступ к созданной файловой системе через NFS, выполните следующие действия:

1 Убедитесь, что служба `nfsd` запущена на исходном компьютере.

2 На исходном компьютере введите команду `exportfs`:

```
# exportfs 192.168.0.2:/mnt/ploop0
```

3 На удаленном компьютере монтируйте путь к хранилищу следующим образом:

```
# mount 192.168.0.1:/mnt/ploop0 /nfsshare
```

Теперь у вас есть доступ к содержимому общей файловой системы ext4 на удаленном компьютере.

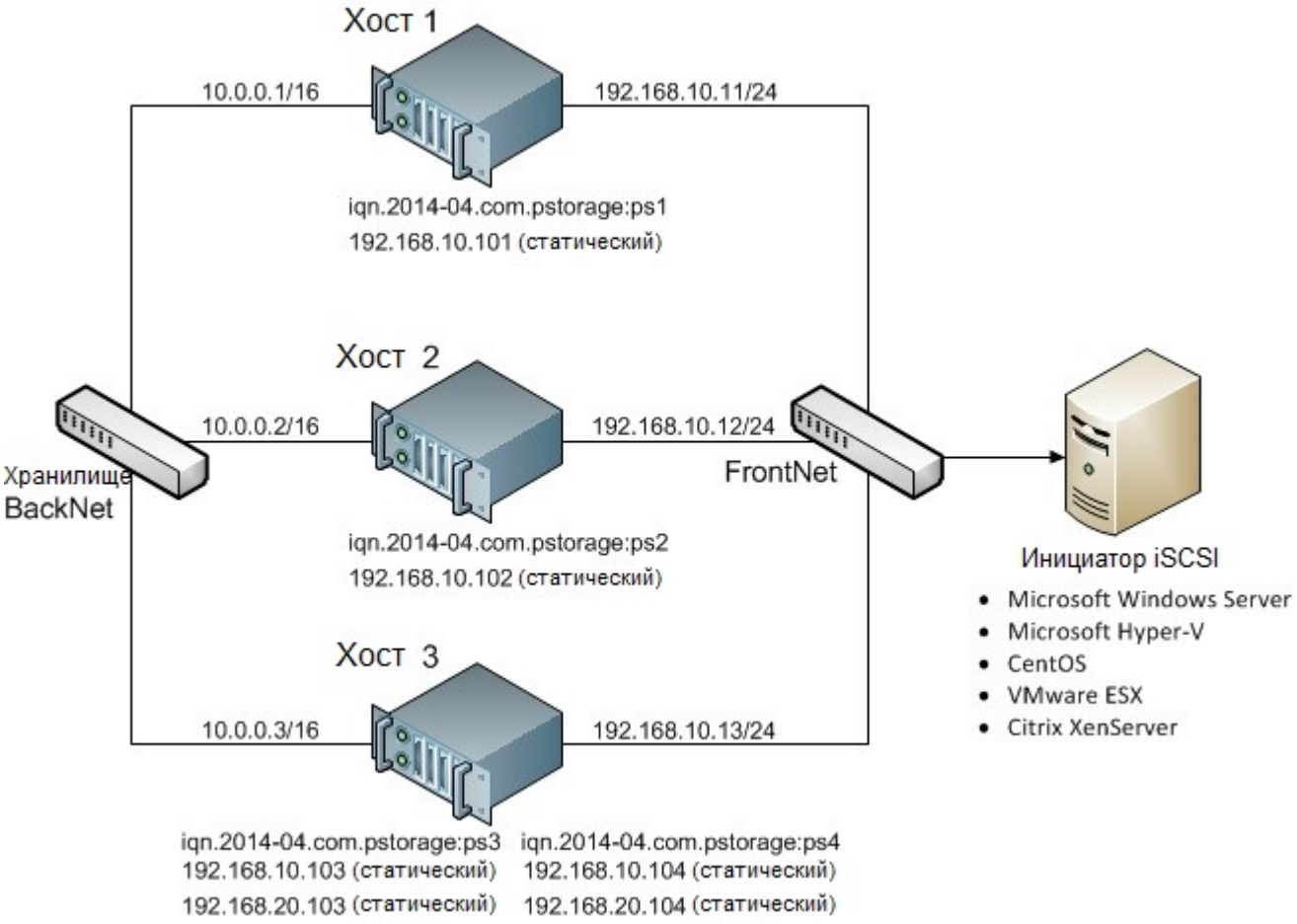
Доступ к кластерам ПК Р-Хранилище через iSCSI

ПК Р-Хранилище позволяет экспортировать дисковое пространство кластера за границами ПК Р-Хранилище в операционные системы и сторонние решения виртуализации. С помощью выделенных инструментов `vstorage-iscsi` можно экспортировать дисковое пространство ПК Р-Хранилище в качестве блочных устройств LUN через iSCSI, как это осуществляется в SAN.

В ПК Р-Хранилище можно создать и запустить любое количество целей iSCSI на каждом сервере кластера. В свою очередь, каждая цель iSCSI может иметь любое количество

LUN (виртуальных дисков). В любой заданный момент каждая цель iSCSI запускается на одном физическом сервере. Благодаря высокой доступности при отказе сервера цели iSCSI, находящиеся на нем, перемещаются и запускаются заново на работающем сервере.

На рисунке ниже показана типичная сеть, настроенная для экспорта дискового пространства ПК Р-Хранилище через iSCSI.



В данном примере в кластере ПК Р-Хранилище работают три физических сервера ПК Р-Виртуализация. На двух серверах находится по одной цели iSCSI, а на третьем сервере – две цели iSCSI. Каждый сервер имеет статический и динамический IP-адреса, назначенные из хранилища BackNet (которое было создано вместе с кластером ПК Р-Хранилище) и FrontNet. Каждая цель iSCSI имеет статический IP-адрес, назначенный из FrontNet.

Подготовка к работе с целями iSCSI ПК Р-Хранилище

На каждом физическом сервере ПК Р-Виртуализация, на котором необходимо создать и запустить цели iSCSI, выполните следующие действия:

1. Убедитесь, что на физическом сервере установлены пакеты `vstorage-iscsi` и `pstorage-scsi-target-utils`.
2. Убедитесь, что у физического сервера есть доступ к кластеру ПК Р-Хранилище в роли клиента и запись в `/etc/fstab`. Для получения дополнительной информации см. **Установка клиентов** (стр. 21).
3. Создайте директорию в кластере ПК Р-Хранилище, в которой будут храниться цели iSCSI и их конфигурация. Например, `/vstorage/stor1/iscsi`.
4. Укажите для переменной ISCSI_ROOT в `/etc/vstorage/iscsi/config` директорию из предыдущего шага. Например:

```
ISCSI_ROOT=/vstorage/stor1/iscsi
```

5. Включите поддержку высокой доступности для физического сервера. Для получения подробной информации см. *Руководство пользователя по ПК Р-Виртуализация*.

Теперь можно создать и запустить цели iSCSI в кластере ПК Р-Хранилище.

Создание и запуск целей iSCSI ПК Р-Хранилище

Примечания:

1. Каждой цели iSCSI должен быть назначен, по крайней мере, один уникальный IP-адрес из статического пула FrontNet.

2. Имя каждой цели iSCSI должно быть уникальным в кластере ПК Р-Хранилище.

3. Цели iSCSI ПК Р-Хранилище поддерживают постоянное резервирование (persistent reservation), чтобы инициаторы iSCSI могли получить монопольный доступ к LUN указанных целей.

Для того чтобы создать и запустить цель `test1`, которая имеет объем 100 ГБ, LUN 1 и IP-адрес 192.168.10.100, выполните следующие команды:

```
# vstorage-iscsi create -n test1 -a 192.168.10.100
IQN: iqn.2014-04.com.vstorage:test1
# vstorage-iscsi lun-add -t iqn.2014-04.com.vstorage:test1 -l 1 -s 100G
# vstorage-iscsi start -t iqn.2014-04.com.vstorage:test1
```

Примечания:

1. Если необходимо изменить IP-адрес цели, остановите цель, как описано в подразделе **Остановка целей iSCSI ПК Р-Хранилище** (стр. 46), а затем введите команду `vstorage-iscsi set -t <target_name> -a <new_IP_address>`.

2. Если необходимо увеличить размер LUN, остановите цель, как описано в подразделе **Остановка целей iSCSI ПК Р-Хранилище** (стр. 46), а затем введите команду `vstorage-iscsi lun-grow -t <target_name> -l <lun_ID> -s <new_size>`.

Проверить, работает цель или нет, можно с помощью команды `vstorage-iscsi list` с именем цели в качестве параметра. Например:

```
[root@dhcp-10-30-24-73 ~]# vstorage-iscsi list -t iqn.2014-04.com.vstorage:test1
Target iqn.2014-04.com.vstorage:test1:
 Portals:    192.168.10.100
 Status:     running
 Registered: yes
 Host:       fefacc38a2f140ca
 LUN:  1, Size:     102400M, Used:      1M, Online: Yes
```

Для получения информации о выводе команды см. **Вывод списка целей iSCSI ПК Р-Хранилище** (стр. 45).

Теперь инициаторы iSCSI имеют доступ к цели `iqn.2014-04.com.vstorage:test1` через портал 192.168.10.100.

Советы по оптимизации производительности

- Цели iSCSI должны быть равномерно распределены на физических серверах кластера. Например, производительность 10 физических серверов с 1 целью iSCSI на каждом сервере будет выше производительности одного физического сервера с 10 целями iSCSI.
- Больше LUN на меньшем количестве целей iSCSI будут работать лучше, чем меньше LUN на большем количестве целей iSCSI.

Вывод списка целей iSCSI ПК Р-Хранилище

С помощью команды `vstorage-iscsi list` можно вывести список всех целей iSCSI, зарегистрированных на сервере ПК Р-Хранилище, или отобразить подробную информацию об определенной цели iSCSI на сервере ПК Р-Хранилище.

Для отображения в виде списка всех целей iSCSI, зарегистрированных на сервере ПК Р-Хранилище, введите следующую команду:

```
# vstorage-iscsi list
IQN                              STATUS  LUNs  HOST                PORTAL(s)
iqn.2014-04.com.vstorage:test1   running 1     fefacc38a2f140ca    192.168.10.100
iqn.2014-04.com.vstorage:test2   running 1     fefacc38a2f140ca    192.168.10.101
iqn.2014-04.com.vstorage:test3   stopped 1     fefacc38a2f140ca    192.168.10.102
iqn.2014-04.com.vstorage:test4   stopped 0     fefacc38a2f140ca    192.168.10.103
```

Для отображения подробной информации о цели iSCSI, зарегистрированной на сервере ПК Р-Хранилище, выполните команду `vstorage-iscsi list` с именем цели в качестве параметра. Например:

```
# vstorage-iscsi list -t iqn.2014-04.com.vstorage:test1
Target iqn.2014-04.com.vstorage:test1:
 Portals:    192.168.10.100
 Status:     running
 Registered: yes
 Host:       fefacc38a2f140ca
LUN:   1, Size:     102400M, Used:      1M, Online: Yes
```

Вывод команды, приведенный выше, показывает следующую информацию:

Элемент	Описание
Target	Уникальное буквенно-цифровое имя цели iSCSI.

Portals	IP-адрес(а) цели.
Status	Текущее состояние цели. • `running`: цель запущена и готова к использованию (для локальных целей); • `stopped`: цель остановлена (для локальных целей); • `service failed`: служба iSCSI отключена (для локальных целей); • `remote`: цель зарегистрирована на другом сервере. • `unregistered`: цель не зарегистрирована ни на одном сервере в кластере ПК Р-Хранилище.
Registered	Зарегистрирована или нет цель на хосте, идентификатор которого отображается в записи **Host**.
Host	Идентификатор физического сервера ПК Р-Хранилище.
LUN	Целочисленный номер виртуального диска в пределах цели.
Size	Логический размер виртуального диска (максимум 16 ТБ).
Used	Физический размер виртуального диска. Физический размер может быть меньше логического из-за расширяемого формата виртуального диска.
Online	• `Yes`: LUN виден инициаторам iSCSI и может быть ими монтирован. • `No`: не виден инициаторам iSCSI и не может быть ими монтирован.

Перемещение целей iSCSI ПК Р-Хранилище между серверами ПК Р-Хранилище

Остановленные цели iSCSI можно перемещать между серверами ПК Р-Хранилище. После того, как цель iSCSI перемещена, ее можно запустить и управлять ей на конечном сервере. На исходном сервере можно только удалить перемещенную цель с помощью параметра `--force` (для получения дополнительной информации см. **Удаление целей iSCSI ПК Р-Хранилище** (стр. 47)).

Для перемещения цели iSCSI выполните следующие действия:

1. Убедитесь, что цель остановлена. Для получения дополнительной информации см. **Остановка целей iSCSI ПК Р-Хранилище** (стр. 46).
2. Отмените регистрацию цели на текущем сервере, используя команду `vstorage-iscsi unregister`. Например:

```
# vstorage-iscsi unregister -t iqn.2014-04.com.vstorage:test1
```

3. Зарегистрируйте цель на новом сервере с помощью команды `vstorage-iscsi register`. Например:

```
# vstorage-iscsi register -t iqn.2014-04.com.vstorage:test1
```

Остановка целей iSCSI ПК Р-Хранилище

Для того чтобы остановить цель iSCSI ПК Р-Хранилище, которая не соединена с инициаторами, используйте команду `vstorage-iscsi stop`. Например, для цели `iqn.2014-04.com.vstorage:test1`:

```
# vstorage-iscsi stop -t iqn.2014-04.com.vstorage:test1
```

Если один или несколько инициаторов iSCSI все еще подключены к цели, то появится сообщение об этом:

```
# vstorage-iscsi stop -t iqn.2014-04.com.vstorage:test1
initiators still connected
 Initiator:  iqn.1994-05.com.redhat:c678b9f6f0 (192.168.30.100)
Unable stop target iqn.2014-04.com.vstorage:test1
```

В этом случае отсоедините инициатор iSCSI, следуя инструкции в руководстве продукта, и повторно введите команду `vstorage-iscsi stop`.

Для принудительной остановки цели, к которой все еще подключен один или несколько инициаторов, добавьте параметр `-f` к команде выше. Например:

```
# vstorage-iscsi stop -t iqn.2014-04.com.vstorage:test1 -f
```

Разрыв iSCSI-соединения подобным способом может привести к ошибкам ввода-вывода на стороне инициатора iSCSI.

Удаление целей iSCSI ПК Р-Хранилище

Удалить цели iSCSI ПК Р-Хранилище можно при помощи команды `vstorage-iscsi delete`. При удалении цели iSCSI ПК Р-Хранилище вместе с ним также будут удалены все LUN.

Для удаления цели iSCSI ПК Р-Хранилище выполните следующие действия:

1. Убедитесь, что цель остановлена. Для получения дополнительной информации см. **Остановка целей iSCSI ПК Р-Хранилище** (стр. 46).
2. Выполните команду `vstorage-iscsi delete` с именем цели в качестве параметра. Например:

```
# vstorage-iscsi delete -t iqn.2014-04.com.vstorage:test1
```

Чтобы удалить остановленную цель iSCSI, зарегистрированный на другом хосте, добавьте параметр `--force` к команде `vstorage-iscsi delete`. Например:

```
# vstorage-iscsi delete -t iqn.2014-04.com.vstorage:test1 --force
```

Доступ к целям iSCSI ПК Р-Хранилище из операционных систем и сторонних решений виртуализации

В данном подразделе описываются способы подключения целей iSCSI ПК Р-Хранилище к ряду операционных систем и сторонних решений виртуализации.

Доступ к целям iSCSI ПК Р-Хранилище из CentOS 6.5

1. Убедитесь, что установлен пакет `iscsi-initiator-utils`.
2. Найдите нужную цель по ее IP-адресу. Например:

```
# iscsiadm --mode discovery --type sendtargets --portal 192.168.10.100
```

3 Перезапустите службу `iscsid`, чтобы проверить недавно подключенные устройства:

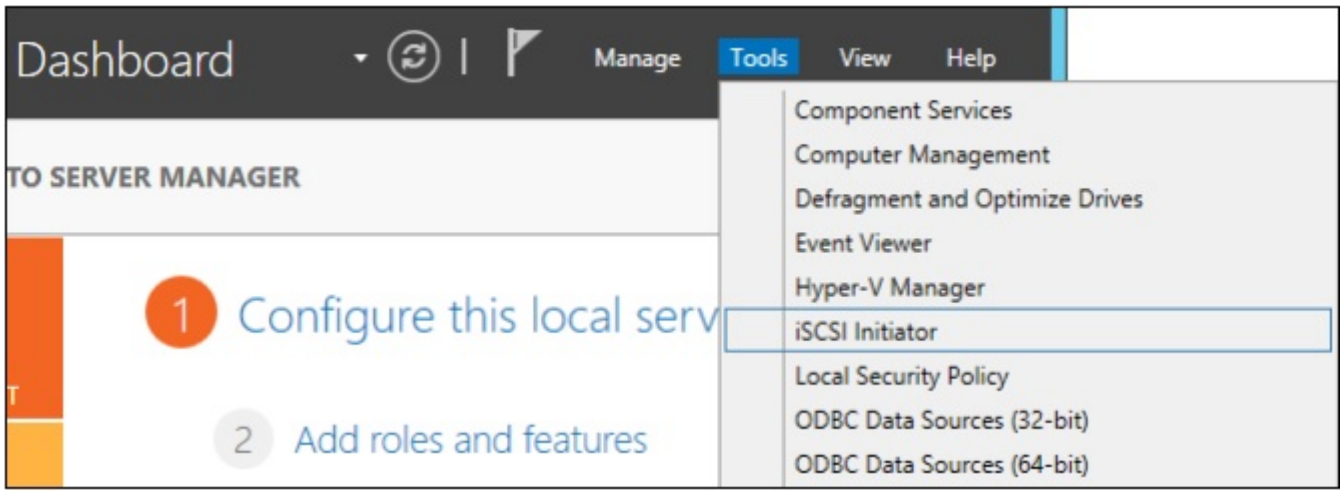
```
# service iscsi restart
```

Чтобы проверить, появился ли новый диск в системе, используйте `fdisk`, `parted` или похожие инструменты.

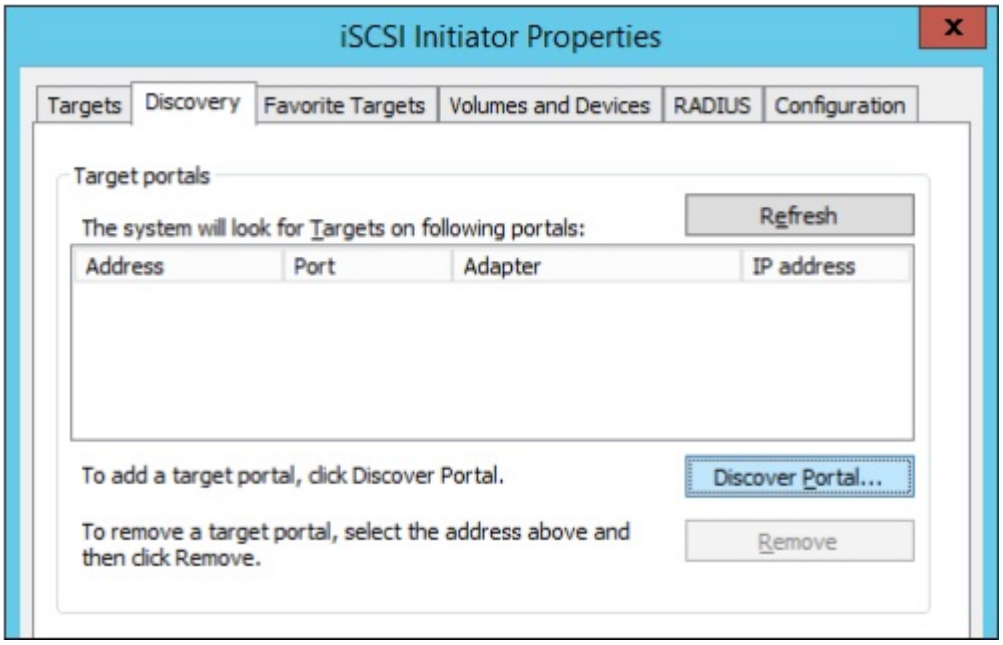
Для получения дополнительной информации см. *Red Hat Enterprise Linux Storage Administration Guide*.

Доступ к целям iSCSI ПК Р-Хранилище из Microsoft Windows Server 2012 R2

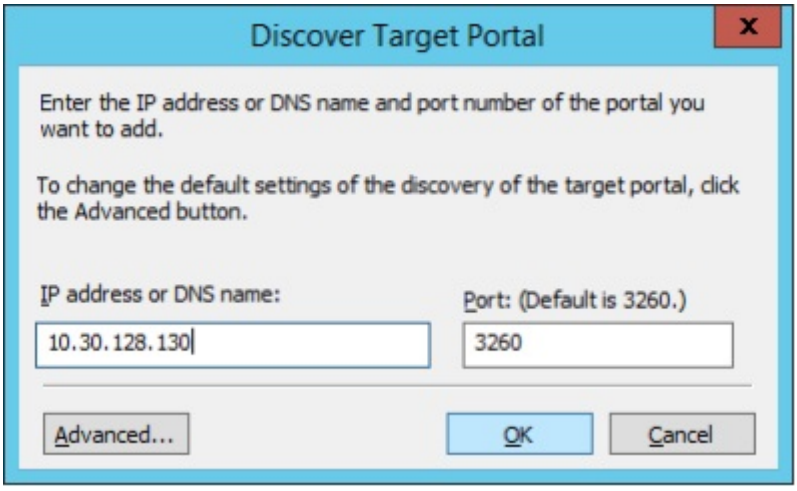
1 В **Server Manager Dashboard** щелкните меню **Tools** на панели инструментов и выберите **iSCSI Initiator**.



2 В окне **iSCSI Initiator Properties** перейдите на вкладку **Discovery** и щелкните **Discover Portal...**.

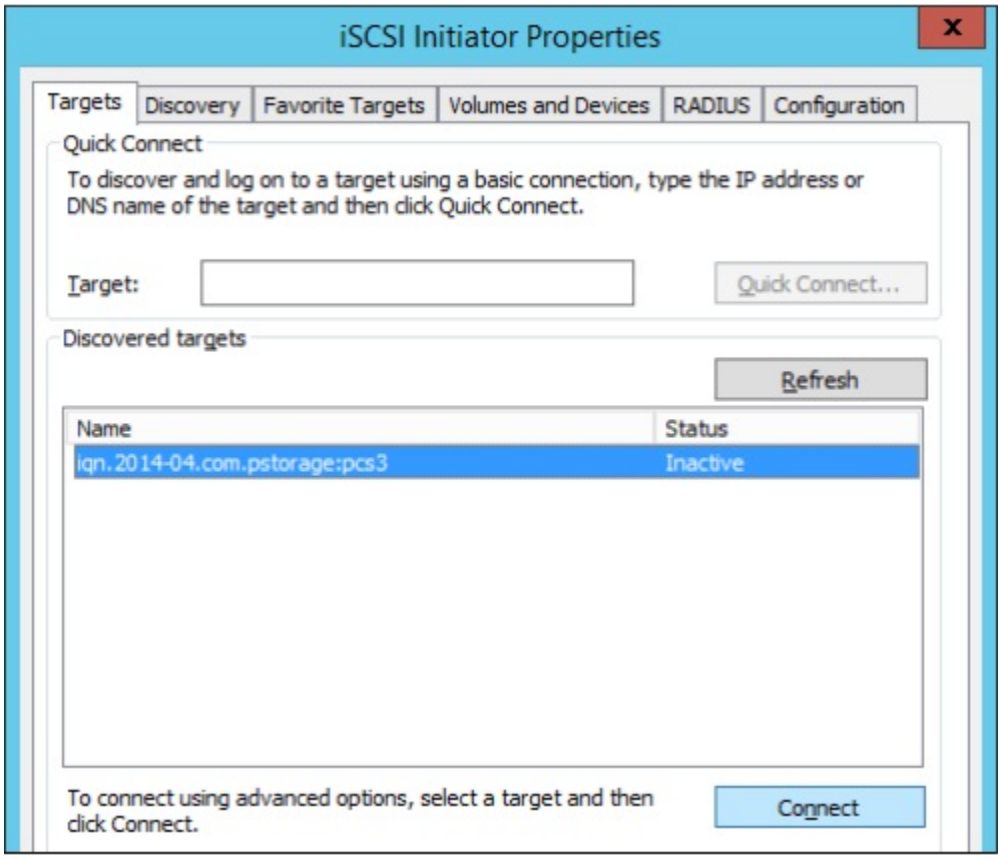


3 В окне **Discover Target Portal** введите IP-адрес портала и щелкните **OK**.

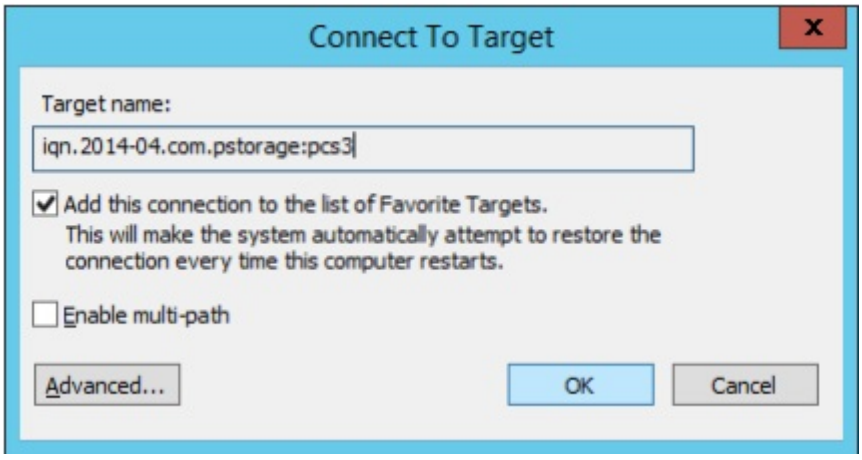


Добавленный портал появится в секции **Target portals**.

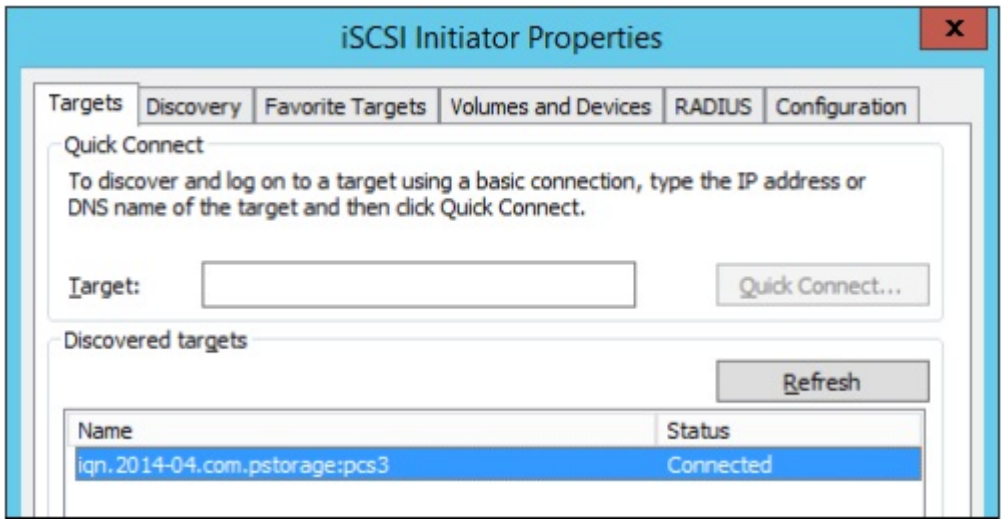
4 На вкладке **iSCSI Initiator Properties** > **Targets** выберите новую цель в секции **Discovered targets** и щелкните **Connect**.



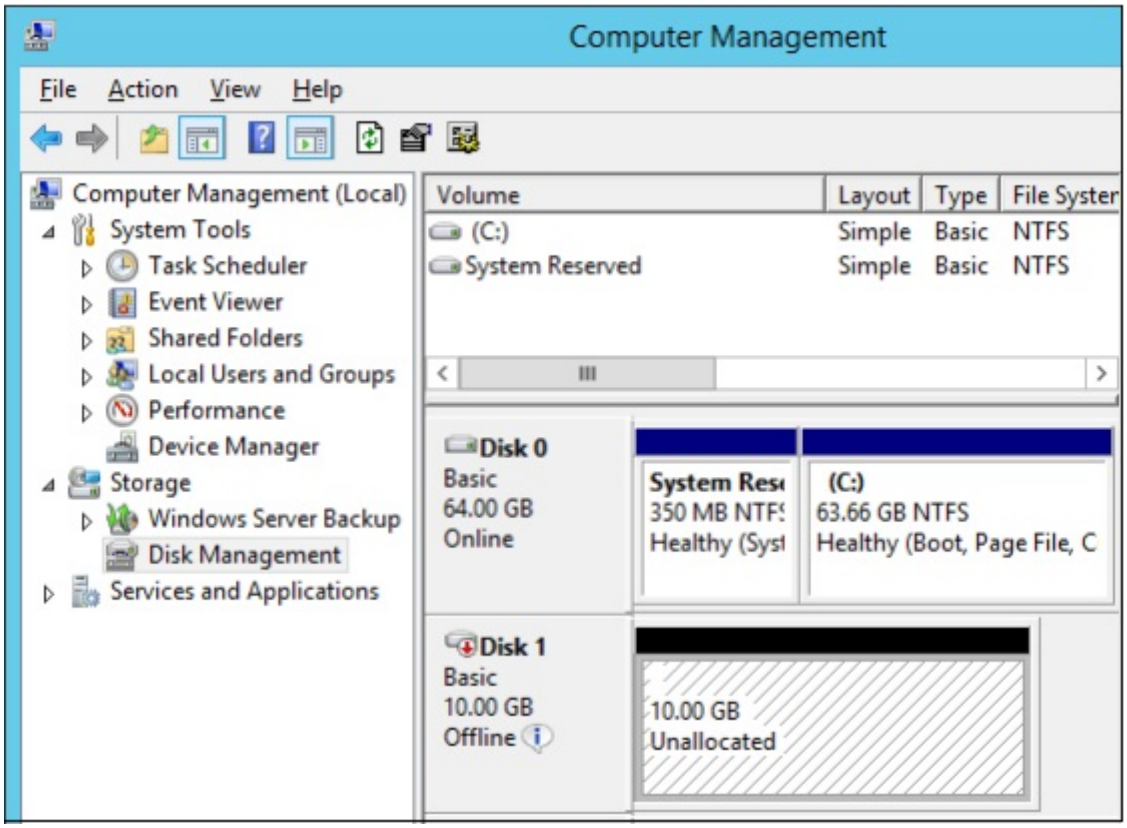
5 В окне **Connect to Target** щелкните **OK**.



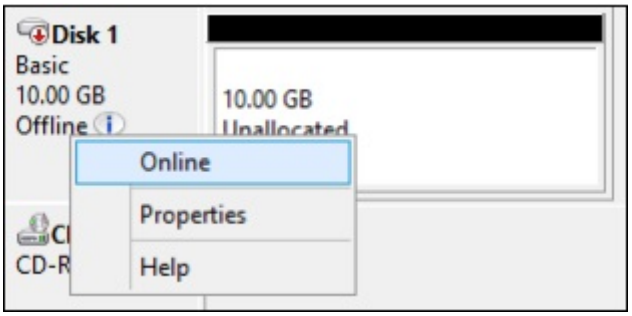
6 Статус цели **Inactive** изменится на **Connected**.



7 Новый диск появится **Server Manager Dashboard** > **Computer Management** > **Storage** > **Disk Management**.

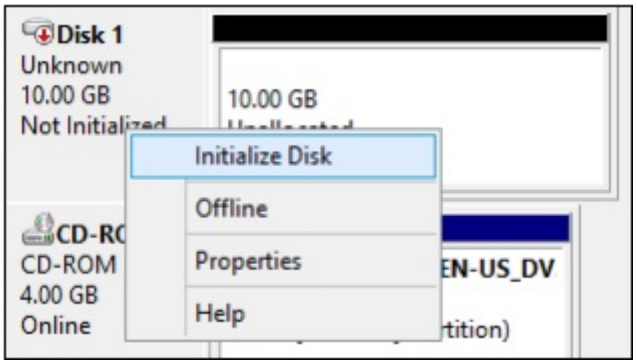


8 Щелкните правой кнопкой мыши по секции с информацией о диске и выберите **Online**.

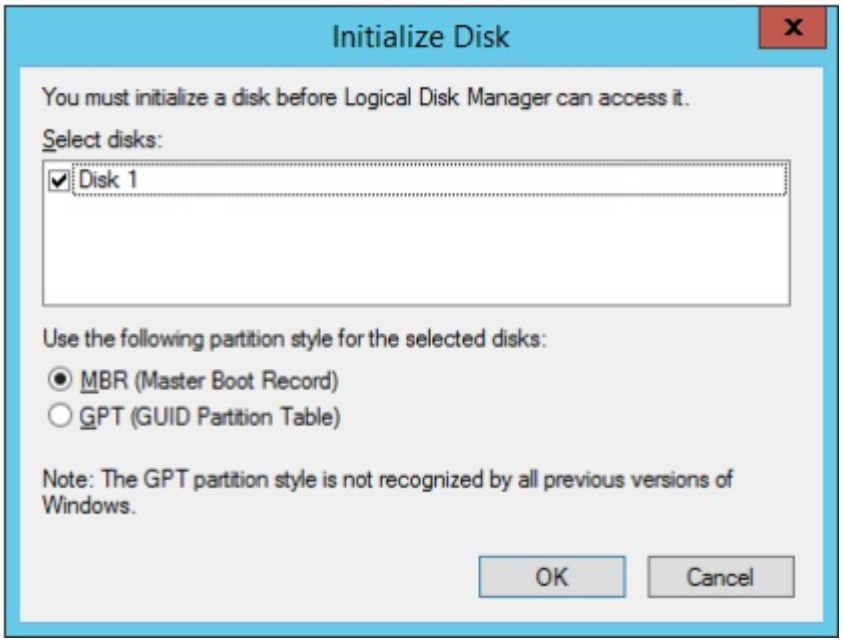


Статус диска изменится на **Online**.

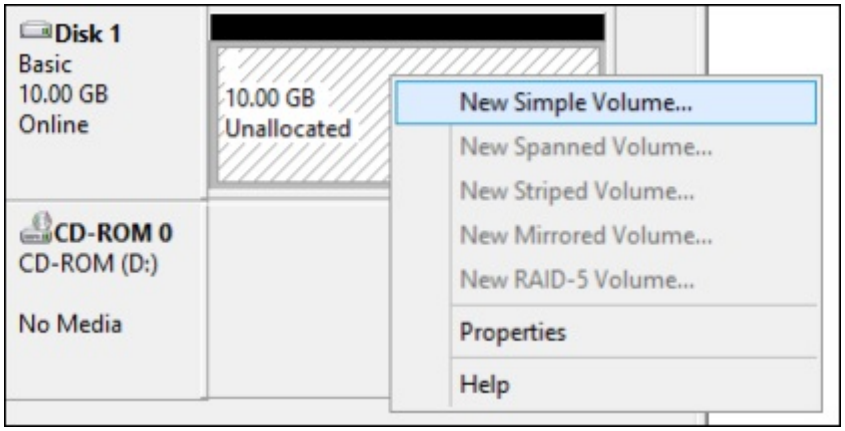
9 Щелкните правой кнопкой мыши по секции с информацией о диске и выберите **Initialize Disk**.



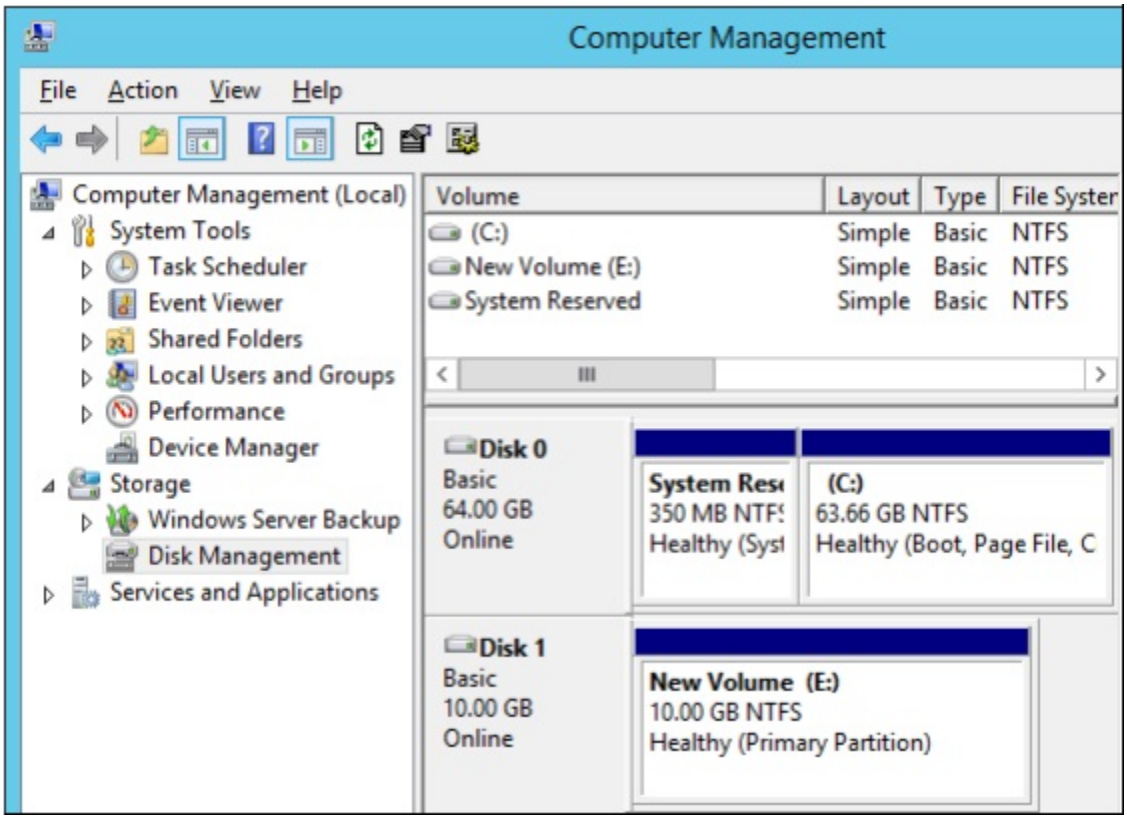
10 В окне **Initialize Disk** щелкните **OK**.



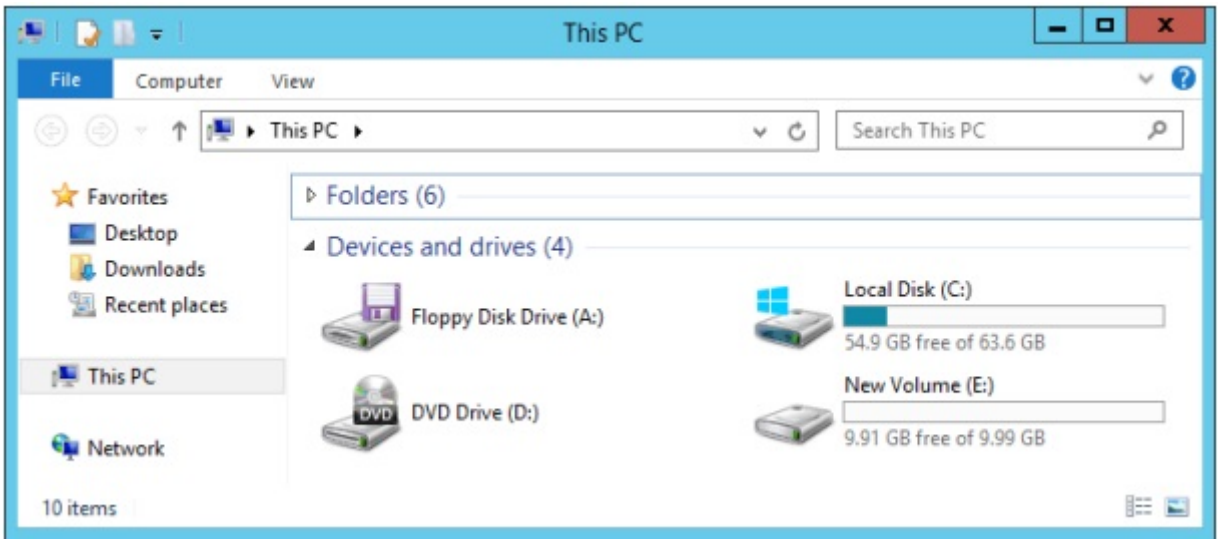
11 Щелкните правой кнопкой мыши по секции с информацией о диске, выберите **New Simple Volume...** и следуйте инструкциям помощника, чтобы отформатировать новый диск в NTFS.



12 Состояние диска изменится на **Healthy**.



13 Новый диск появится в Windows Explorer.



Доступ к целям iSCSI ПК Р-Хранилище из VMware ESXi

1 В **vSphere Client** перейдите на вкладку **Configuration** и щелкните **Storage Adapters** в секции **Hardware**.

2 Если нет ни одного программного адаптера iSCSI, то его можно добавить, щелкнув правой кнопкой мыши в секции **Storage Adapters** и выбрав **Add Software iSCSI Adapter...**.

3 Откройте свойства программного адаптера iSCSI, перейдите на вкладку **Static Discovery** и щелкните **Add...**.

4 В окне **Add Static Target Server** введите IP-адрес и имя цели.

5 Закройте окно свойств программного адаптера iSCSI и снова выполните проверку адаптера.

6 Добавленная цель iSCSI появится в секции **Details** настроенного программного адаптера iSCSI.

Для получения дополнительной информации см. *VMware vSphere Storage Guide*.

Доступ к целям iSCSI ПК Р-Хранилище из Citrix XenServer 6.2

1 В XenCenter перейдите на вкладку **Storage** и щелкните **New SR...**.

2 В окне **New Storage Repository**:

   1. В секции **Type** выберите пункт **Software iSCSI**.
   2. В секции **Name** укажите новое имя или оставьте имя по умолчанию.
   3. В секции **Location** введите IP-адрес цели в поле **Target Host**, щелкните **Discover IQNs** и выберите нужную цель, затем щелкните **Discover LUNs** и выберите нужный LUN.

3 Щелкните **Finish**, чтобы начать форматирование диска.

Новый репозиторий появится в XenCenter.

Для получения дополнительной информации см. документацию для XenCenter по ссылке http://support.citrix.com/proddocs/topic/xencenter-62/xs-xc-storage.html.

Доступ к целям iSCSI ПК P-Хранилище из Microsoft Hyper-V

Примечание: Имя целей, которые будут монтированы, не должно содержать символы подчеркивания.

1 Убедитесь, что запущена служба Microsoft iSCSI Initiator `MSiSCSI`.

2 Найдите новый целевой портал. Например, для портала 192.168.10.100, выполните:

```
PS C:\Users\Administrator>new-iscsitargetportal -targetportaladdress 192.168.10.100
Initiator Instance Name    :
Initiator Portal Address   :
IsDataDigest               : False
IsHeaderDigest             : False
TargetPortalAddress        : 192.168.10.100
TargetPortalPortNumber     : 3260
PSComputerName             :
```

3 Подключитесь к нужной цели. Например, для цели `iqn.2014-03.com.vstorage:test1`:

```
PS C:\Users\Administrator> connect-iscsitarget
cmdlet Connect-IscsiTarget at command pipeline position 1
Supply values for the following parameters:
NodeAddress: iqn.2014-04.com.vstorage:test1
AuthenticationType         : NONE
InitiatorInstanceName      : ROOT\ISCSIPRT\0000_0
InitiatorNodeAddress       : iqn.1991-05.com.microsoft:win-l2dj7g36n7e.sw.swsoft.com
InitiatorPortalAddress     : 0.0.0.0
InitiatorSideIdentifier    : 400001370000
IsConnected                : True
IsDataDigest               : False
IsDiscovered               : True
IsHeaderDigest             : False
IsPersistent               : False
NumberOfConnections        : 1
SessionIdentifier          : ffffe00000b5e020-4000013700000005
TargetNodeAddress          : iqn.2014-04.com.vstorage:test1
TargetSideIdentifier       : 0001
PSComputerName             :
```

4 Чтобы проверить, что диск был подключен, выполните:

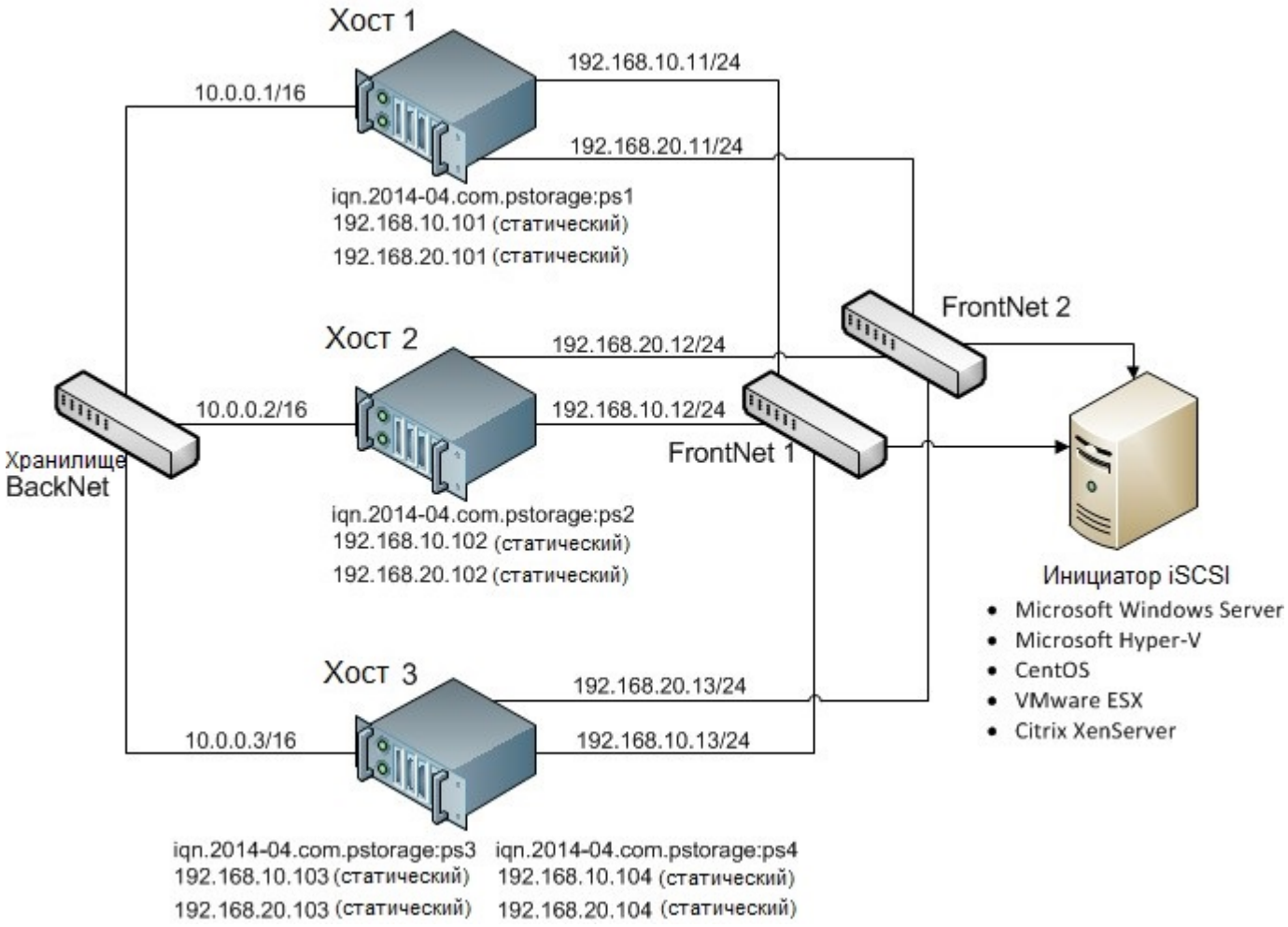
```
PS C:\Users\Administrator> get-disk
Number Friendly Name                           OperationalStatus
Total Size Partition Style
------ -------------                           ----------------                    --
-------- --------------
1      IET VIRTUAL-DISK SCSI Disk Device       Offline
100 GB RAW
<...>
```

Смонтированный диск можно инициализировать для использования в Microsoft Hyper-V.

Для получения дополнительной информации см. iSCSI Cmdlets in Windows PowerShell (http://technet.microsoft.com/en-us/library/hh826099.aspx).

Настройка многопутевого ввода-вывода для целей iSCSI ПК Р-Хранилище

Техника многопутевого ввода-вывода используется для улучшения отказоустойчивости и производительности при помощи создания нескольких путей для одной цели iSCSI. На рисунке ниже показана типичная сеть с включенным многопутевым вводом-выводом, настроенная для экспорта дискового пространства ПК Р-Хранилище через iSCSI.



В данном примере в кластере ПК Р-Хранилище работают три физических сервера ПК Р-Виртуализация. На двух серверах находится по одной цели iSCSI, а на третьем сервере – две цели iSCSI. Каждый сервер имеет статический и динамический IP-адреса, назначенные из FrontNet 1 и статический и динамический IP-адреса, назначенные из FrontNet 2. В свою очередь, каждой цели iSCSI назначен статический IP-адрес из FrontNet 1 и статический IP-адрес из FrontNet 2. Если одна из сетей FrontNet откажет, цели iSCSI будут доступны по другой сети.

Для того чтобы включить многопутевой ввод-вывод для цели iSCSI ПК Р-Хранилище, необходимо назначить ему несколько IP-адресов из разных сетей с помощью параметра `-a`. Например, для сервера, подключенного к двум сетям 192.168.10.0/24 и 192.168.20.0/24, выполните следующую команду:

```
# vstorage-iscsi create -n ps1 -a 192.168.10.101 -a 192.168.20.101
```

Управление учетными записями CHAP для целей iSCSI ПК Р-Хранилище

ПК Р-Хранилище позволяет ограничить доступ к целям iSCSI при помощи проверки подлинности CHAP.

Для использования проверки подлинности CHAP необходимо:

1. Создать учетную запись CHAP.
2. Создать цель iSCSI, привязанную к данной учетной записи CHAP.

Данные шаги подробно описываются ниже.

Создание учетных записей CHAP для целей iSCSI ПК Р-Хранилище

Создать учетную запись CHAP можно с помощью команды `vstorage-iscsi account-create`. Например, для создания учетной записи CHAP `user1`:

```
# vstorage-iscsi account-create -u user1
Enter password:
Verify password:
```

Создание целей iSCSI ПК Р-Хранилище, привязанных к учетным записям CHAP

Чтобы создать цель iSCSI ПК Р-Хранилище, привязанную к учетной записи CHAP, используйте команду `vstorage- iscsi create` с дополнительным парамером `-u`. Например, для создания цели, привязанной к учетной записи CHAP `user1`:

```
# vstorage-iscsi create -n test1 -a 192.168.10.100 -u user1
IQN: iqn.2014-04.com.vstorage:test1
```

Смена пароля от учетной записи CHAP

Для смены пароля от учетной записи CHAP выполните команду `vstorage-iscsi account-set`. Например, для смены пароля от учетной записи CHAP `user1`:

```
# vstorage-iscsi account-set -u user1
Enter password:
Verify password:
```

Смена пароля произойдет после перезагрузки цели.

Вывод списка учетных записей CHAP и назначенных им целей iSCSI ПК Р-Хранилище

Чтобы отобразить список всех существующих учетных записей CHAP, используйте команду `vstorage-iscsi account-list`. Например:

```
# vstorage-iscsi account-list
user1
```

Вывести список целей iSCSI ПК Р-Хранилище, назначенных определенной учетной записи CHAP, можно с помощью команды `vstorage-iscsi account-list` с параметром `-u`. Например, для отображения списка целей iSCSI, назначенных учетной записи CHAP `user1`:

```
# vstorage-iscsi account-list -u user1
iqn.2014-04.com.vstorage:test1
```

Управление снапшотами LUN

Создавать снапшоты LUN и управлять ими можно так же, как и снапшотами виртуальных машин. Для того чтобы создать снапшот целой цели, необходимо создать снапшоты каждого LUN внутри него.

Создание снапшотов LUN

Для создания снапшота LUN в цели iSCSI используйте команду `vstorage-iscsi snapshot-create`. Например, для LUN 1 в цели `iqn.2014-04.com.vstorage:test1`:

```
# vstorage-iscsi snapshot-create -t iqn.2014-04.com.vstorage:test1 -l 1
Snapshot a1f54314-bc06-40c6-a587-965feb9d85bb successfully created.
```

Примечание: Чтобы самостоятельно сгенерировать UUID, используйте `uuidgen`.

Вывод списка снапшотов LUN

Чтобы вывести список снапшотов для указанного LUN, используйте команду `vstorage-iscsi snapshot-list`. Например, для LUN 1 в цели `iqn.2014-04.com.vstorage:test1`:

```
# vstorage-iscsi snapshot-list -t iqn.2014-04.com.vstorage:stor4 -l 1
CREATED             C UUID                                PARENT_UUID
2014-04-11 13:16:51   a1f54314-bc06-40c6-a587-{...}  00000000-0000-0000-0000-{...}
2014-04-11 13:16:57 * 9c98b442-7482-4fd0-9c45-{...}  a1f54314-bc06-40c6-a587-{...}
```

В выводе команды, приведенном выше, звездочка (`*`) в колонке **C** указывает на текущий снапшот, а колонка `PARENT_UUID` показывает зависимость или историю снапшота.

Переключение между снапшотами LUN

Переключиться на указанный снапшот LUN можно с помощью команды `vstorage-iscsi snapshot-switch`. Например:

```
# vstorage-iscsi snapshot-switch -u a1f54314-bc06-40c6-a587-965feb9d85bb
```

После переключения текущий снапшот LUN будет удален.

Примечание: Переключение между снапшотами возможно только при выключенном LUN.

Просмотр подробной информации о снапшоте LUN

Для просмотра подробной информации об указанном снапшоте используйте команду `vstorage-iscsi snapshot-info`. Например:

```
# vstorage-iscsi snapshot-info -u 9c98b442-7482-4fd0-9c45-9259374ca84e
Target: iqn.2014-04.com.vstorage:stor4
LUN: 1
Created: 2014-04-11 13:16:57
Parent: 00000000-0000-0000-0000-000000000000}
{a1f54314-bc06-40c6-a587-965feb9d85bb}
{9c98b442-7482-4fd0-9c45-9259374ca84e
Description: None
```

Удаление снапшотов LUN

Чтобы удалить определенный снапшот LUN, выполните команду `vstorage-iscsi snapshot-delete`. Например:

```
# vstorage-iscsi snapshot-delete -u a1f54314-bc06-40c6-a587-965feb9d85bb
```

Если снапшот не имеет дочерних элементов, он будет удален. Если снапшот имеет хотя бы один дочерний элемент, он перейдет в данный дочерний элемент.

Примечания:

1. Удаление возможно только для выключенных снапшотов.

2. Удаление снапшота, который имеет несколько дочерних элементов, на текущий момент не поддерживается.

Доступ к кластерам ПК Р-Хранилище через объектное хранилище типа S3

ПК Р-Хранилище может осуществлять экспорт данных при помощи API, который совместим с Amazon S3 и позволяет провайдерам:

- запускать службы на базе S3 в своих собственных инфраструктурах ПК Р-Хранилище,
- предоставлять клиентам хранилище на базе S3 в виде услуги вместе с ПК Р-Хранилище.

Поддержка S3 расширяет функционал ПК Р-Хранилище и требует наличия рабочего кластера ПК Р-Хранилище.

Об объектном хранилище

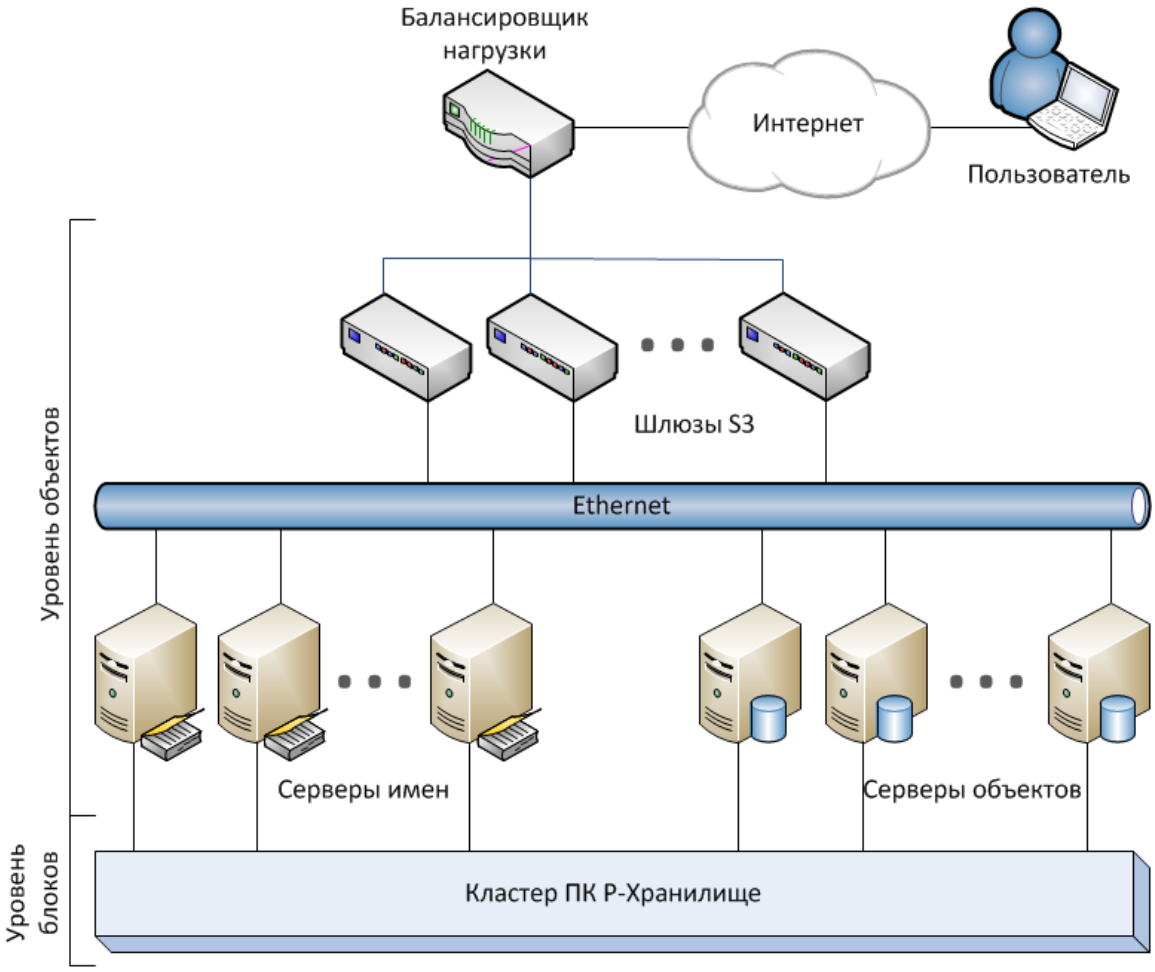
Объектное хранилище представляет собой архитектуру хранилища, позволяющую управлять данными в виде объектов (как в хранилищах типа «ключ-значение»), в отличие от файлов в файловых системах или блоков в блочных хранилищах. Кроме данных, у каждого объекта есть метаданные, которые его описывают, а также уникальный идентификатор, позволяющий находить объект в хранилище. Объектное хранилище оптимизировано для хранения миллиардов объектов, в частности для хранилища приложений, размещения статического веб-контента, служб онлайн хранилищ, большого объема данных и резервных копий. Все данные способы использования возможны благодаря совокупности высокой масштабируемости, а также доступности и непротиворечивости данных.

Главным отличием объектного хранилища от других типов хранилищ является невозможность изменения частей объекта, поэтому, если объект изменяется, вместо него создается другая версия. Данный подход очень важен для поддержания доступности и непротиворечивости данных. Изменение объекта целиком исключает возникновение конфликтов. Таким образом, объект с последней временной отметкой считается текущей версией. В результате, объекты всегда непротиворечивы.

Другой особенностью объектного хранилища является согласованность в конечном счете. Согласованность в конечном счете не гарантирует, что чтение записей должно возвращаться в новое состояние после завершения записи. Читатели могут наблюдать старое состояние неопределенный промежуток времени, пока запись не будет перенесена во все реплики (копии). Это важно для обеспечения доступности данных, так как территориально удаленные центры обработки данных не всегда могут выполнять синхронное обновление данных (например, из-за неполадок сети) и самообновление может происходить медленно из-за того, что ожидание подтверждения от всех реплик данных, находящихся на больших расстояниях, может занять сотни миллисекунд. Таким образом, согласованность в конечном счете помогает скрывать задержки обмена данными при записи за счет возможного устаревшего состояния данных, которое видят читатели. Однако во многих сценариях использования это допускается.

Инфраструктура объектного хранилища

Инфраструктура объектного хранилища состоит из следующих сущностей: серверы объектов, серверы имен, шлюзы S3 и хранилище блочного уровня.



- На сервере объектов хранятся актуальные данные объектов, полученные от шлюза S3. Сервер объектов хранит свои собственные данные в ПК Р-Хранилище со встроенной функцией высокой доступности.
- На сервере имен хранятся метаданные об объектах, полученные от шлюза S3. Метаданные включают имя объекта, его размер, список контроля доступа (ACL), расположение, владельца и др. Сервер имен также хранит свои собственные данные в ПК Р-Хранилище со встроенной функцией высокой доступности.
- Шлюз S3 является прокси-сервером между службами объектного хранилища и пользователями. Он получает и обрабатывает запросы протокола Amazon S3 и использует веб-сервер nginx для внешних соединений. Шлюз S3 выполняет аутентификацию пользователей S3 и проверку ACL. У него нет своих данных (т.е. он не сохраняет данные о запросах).
- Хранилище блочного уровня представляет собой кластер ПК Р-Хранилище с высокой доступностью служб и данных. Так как все службы объектного хранилища запускаются на хостах, то для него не нужны виртуальные среды (или соответствующие лицензии).

Обзор объектного хранилища

В терминах объектного хранилища S3, файл является объектом. На серверах объектов каждый объект, загруженный с помощью API S3, хранится в виде пары сущностей:

- Имена объектов и связанные с объектами метаданные хранятся на сервере имен. Имя объекта в хранилище определяется параметрами на основе запроса и свойствами корзины следующим образом:
  - Если управление версиями корзин отключено, имя объекта в хранилище содержит имя корзины и имя объекта, взятое из запроса S3.
  - Если управление версиями корзин включено, имя объекта также включает в себя список версий объекта.
- Данные объектов хранятся на сервере объектов. Часть имени объекта, содержащая директорию, определяет сервер имен для его хранения, а полное имя объекта определяет сервер объектов для хранения данных объекта.

Многопоточная загрузка

Имя многопоточной загрузки определяется способом, похожим на имя объекта, но объект, который ему соответствует, содержит таблицу, а не содержимое файла. Таблица состоит из индексных номеров частей и их смещения в файле. Это позволяет одновременно загружать части многопоточной загрузки (рекомендуется для файлов большого объема). Максимальное количество частей равно 10,000.

Взаимодействие хранилища S3 с кластером ПК Р-Хранилище

Для кластера хранилища S3 требуется наличие рабочего кластера ПК Р-Хранилище на каждом сервере кластера S3. ПК Р-Хранилище обеспечивает общий доступ к данным, непротиворечивость и доступность данных, оптимальную производительность для произвольных операций ввода-вывода, а также высокую доступность для служб хранилища. В терминах хранилища, данные S3 являются набором файлов (см. **Сервер объектов** (стр. 64)), которые уровень файловой системы ПК Р-Хранилище (vstorage-mount) никак не интерпретирует.

Компоненты объектного хранилища

В данном подразделе описываются компоненты хранилища S3—шлюзы, серверы объектов и имен—а также инструменты управления S3 и служебные корзины.

Шлюз

Шлюз выполняет следующие функции:

- Получает запросы S3 от веб-сервера (через nginx и FastCGI).
- Анализирует пакеты S3 и проверяет правильность запросов S3.
- Аутентифицирует пользователей S3.

- Проверяет правильность прав доступа к корзинам и объектам при помощи ACL.
- Собирает статистику по числу различных запросов, а также по объему полученных или переданных данных.
- Определяет пути к серверам имен и объектов, на которых хранятся данные объекта.
- Запрашивает имена и связанные с объектом метаданные у сервера имен.
- Получает ссылки на объекты, хранящиеся на серверах объектов, запрашивая имя у сервера имен.
- Кэширует метаданные и ACL объектов S3, полученные от серверов имен, а также данные, которые необходимы для аутентификации пользователей и хранятся на серверах имен.
- Работает в роли прокси-сервера при чтении из или записи в данные объекта, хранящегося на сервере объектов. Только запрошенные данные передаются в процессе чтения или записи. Например, если пользователь запрашивает прочитать 10МБ из объекта размером 1ТБ, только данные 10МБ будут прочитаны с сервера объектов.

Шлюз S3 состоит из анализатора входящих запросов, зависимого от типа и асинхронного обработчика этих запросов и асинхронного обработчика прерванных запросов, нуждающихся в завершении (сложные операции, такие как создание или удаление корзины). В долгосрочно выделенной памяти шлюза не хранятся данные о его состоянии. Вместо этого, он хранит все данные, необходимые для хранилища S3, в самом объектном хранилище (на серверах имен и объектов).

Сервер имен

Сервер имен выполняет следующие функции:

- Хранит имена и метаданные объектов.
- Предоставляет API для вставки, удаления, отображения списка имен объектов и изменения метаданных объектов.

Сервер имен состоит из данных (т.е. метаданных объектов), журнала изменений объектов, асинхронной программы очистки памяти и асинхронных обработчиков входящих запросов от разных компонентов системы.

Данные хранятся в виде B-дерева, где каждому имени объекта соответствует структура метаданных этого объекта. Метаданные объекта S3 состоят из трех частей: информации об объекте, пользовательских заголовков (необязательно) и списка контроля доступа для объекта. Файлы хранятся в соответствующей директории на базе общего хранилища (т.е. ПК Р-Хранилища).

Сервер имен отвечает за подмножество пространства имен для объектов кластера S3. Каждая копия сервера имен представляет собой процесс в пространстве пользователя, который запущен одновременно с другими процессами и может использовать до одного ядра процессора. Оптимальное количество серверов имен на один сервер - от 4 до 10. Рекомендуется начать с создания 10 копий на каждом сервере при создании кластера, чтобы в дальнейшем упростить масштабируемость. Если на сервере есть ядра ЦП, не используемые другими службами хранилища, вы можете создать больше серверов имен, чтобы задействовать эти ядра ЦП.

Сервер объектов

Сервер объектов выполняет следующие функции:

- Хранит данные объектов в пулах (контейнерах данных).
- Предоставляет API для создания, чтения (включая частичное чтение), удаления объектов и записи в них.

Сервер объектов состоит из:

- информации о блоках объектов, хранящихся на данном сервере объектов,
- контейнеров с данными объектов,
- асинхронной программы очистки памяти, которая освобождает секции контейнеров после операций удаления объектов.

Блоки данных объектов хранятся в пулах. Хранилище использует 12 пулов с размером блоков, равным 2^x, в диапазоне от 4 КБ до 8 МБ. Пул представляет собой обычный файл в блочном хранилище, состоящий из блоков фиксированного размера (регионов). Другими словами, каждый пул является файлом очень большого размера, который предназначен для хранения объектов определенного размера: первый пул для объектов размером 4 КБ, второй пул для объектов размером 8 КБ и т.д.

Каждый пул состоит из блока с информацией о системе и регионов с фиксированным размером данных. Каждый регион содержит маску свободного/измененного бита. Данные региона хранятся в том же файле с B-деревом объекта. Это обеспечивает атомарность в процессе выделения и освобождения блока. Каждый блок в регионе содержит заголовок и данные объекта. Заголовок включает в себя идентификатор объекта, которому принадлежат данные. Идентификатор нужен для алгоритма дефрагментации на уровне пула, который не имеет доступа к B-дереву объекта. Пул для хранения объекта выбирается в зависимости от размера объекта.

Например, объект размером 30 КБ будет помещен в пул для объектов размером 32 КБ и будет занимать один объект размером 32 КБ. Объект размером 129 КБ будет разделен на две части: одну часть в 128 КБ и одну - в 1 КБ. Первая часть будет помещена в пул для объектов размером 128 КБ, а вторая – в пул для объектов размером 4 КБ. В случае с небольшими объектами затраты ресурсов могут быть значительными, так как даже объект размером 1 Б будет занимать блок размером 4 КБ. Дополнительно около 4 КБ метаданных для каждого объекта будут храниться на сервере имен. Однако данный подход позволяет получить максимальную производительность и устраняет фрагментацию свободного пространства. Также чем больше размер объекта, тем меньше потребление ресурсов. И наконец, когда объект удаляется, блок пула помечается как свободный и может быть использован для хранения новых объектов.

Объекты, состоящие из нескольких частей, хранятся по частям (каждая часть хранится в качестве отдельного объекта), и эти части можно хранить на разных серверах объектов.

Инструменты управления S3

Объектное хранилище имеет два инструмента:

- `ostor-ctl` для настройки компонентов хранилища и
- `ostor-s3-admin` для управления пользователями, данное приложение позволяет создавать, редактировать и удалять учетные записи пользователей S3, а также управлять ключами доступа (создавать и удалять пары идентификаторов ключей доступа S3 и секретных ключей S3).

Служебная корзина

В служебной корзине хранится информация о службах и промежуточная информация, необходимая для хранилища S3. Доступ к данной корзине есть только у администратора S3 (для системного администратора потребуется создать ключи доступа с помощью инструмента `ostor-s3-admin`).

Обмен данными

В объектном хранилище Р-Хранилище каждая служба имеет 64-битный уникальный идентификатор. Также у каждого объекта есть уникальное имя. Часть имени объекта, содержащая директорию, определяет сервер имен для его хранения, а полное имя объекта определяет сервер объектов для хранения данных объекта. Списки серверов имен и объектов хранятся в директории `vstorage` кластера, предназначенной для данных объектного хранилища и доступной для любого пользователя, у которого есть доступ к кластеру. Данная директория включает в себя поддиректории, которые соответствуют службам на серверах имен и объектов. Имена поддиректорий представляют собой идентификаторы служб в шестнадцатеричной форме. В каждой поддиректории службы есть файл, содержащий идентификатор хоста, на котором запущена служба. Таким образом, при помощи шлюза компонент системы, имеющий доступ к кластеру, может найти идентификатор службы, обнаружить ее хост и отправить ей запрос.

Шлюз S3 осуществляет обмен данными со следующими компонентами:

- Клиентами через веб-сервер. Шлюз получает запросы S3 от пользователей и отвечает на них.
- Серверами имен. Шлюз создает, удаляет, изменяет имена, которые соответствуют корзинам или объектам S3, проверяет их наличие и запрашивает множества имен из списков корзин.
- Серверами объектов в хранилище. Шлюз отправляет запросы, изменяющие данные, на серверы объектов и имен.

Кэширование данных

Для обеспечения эффективного использования данных в объектном хранилище все шлюзы, серверы имен и объектов кэшируют данные, которые они хранят. Серверы имен и объектов кэшируют B-деревья.

Шлюзы хранят и кэшируют следующие данные, полученные от серверов имен:

- Списки пар идентификаторов и адресов электронной почты пользователей.
- Данные, необходимые для аутентификации пользователей: идентификаторы ключей доступа и секретных ключей. Для получения дополнительной информации см. документацию Amazon S3.
- Метаданные и списки контроля доступа корзин. Метаданные содержат время создания, идентификатор текущей версии и передают их на сервер имен, чтобы проверить, что шлюз хранит последнюю версию метаданных.

Операции с объектами

В данном разделе описываются процессы операций хранилища S3: запросы операций; операции создания, чтения и удаления.

Запросы операций

Чтобы создать, удалить, прочитать объект или изменить его данные, объектное хранилище S3 должно сначала запросить одну из этих операций, а затем выполнить их. Весь процесс запрашивания и выполнения операции состоит из следующих частей:

1. Запрашивание данных для аутентификации пользователя. Они будут храниться на сервере имен в специальном формате (см. **Служебная корзина** (стр. 65)). Для получения данных (идентификатора, адреса электронной почты, ключей доступа) отправляется запрос с кодом операции поиска на подходящий сервер имен.
2. Аутентификация пользователя.
3. Запрашивание метаданных корзины и объекта. Для их получения отправляется другой запрос с кодом операции поиска на сервер имен, где хранятся имена объектов и корзин.
4. Проверка прав доступа пользователя к корзинам и объектам.

5 Выполнение запрошенной операции: создание, редактирование или чтение данных или удаление объекта.

Операция создания

Для создания объекта шлюз отправляет следующие запросы:

1 Запрос с кодом операции защиты на сервер имен. Он создает защиту с таймером для проверки, которая будет выполнена после определенного периода времени, чтобы выяснить, был ли действительно создан объект с данными. Если объект не был создан, операция создания завершится с ошибкой, и защита отправит запрос на сервер объектов на удаление данных объекта, если они были записаны. После этого защита удаляется.

2 Запрос с кодом операции создания на сервер объектов и последующие сообщения фиксированного размера, содержащие данные объекта. Последнее сообщение включает в себя флаг окончания передачи данных.

3 Другой запрос с кодом операции создания на сервер имен. Сервер проверяет, есть ли соответствующая защита и, если ее нет, операция завершается с ошибкой. В противном случае сервер создает имя и отправляет на шлюз подтверждение успешно выполненной операции создания.

Операция чтения

Для выполнения запроса S3 на чтение шлюз определяет идентификатор подходящего сервера имен на основе имени директории и идентификатор соответствующего сервера объектов на основе полного имени объекта. Чтобы выполнить операцию чтения, шлюз отправляет следующие запросы:

1 Запрос с кодом операции чтения на подходящий сервер имен. Ответ на него содержит ссылку на объект.

2 Запрос на подходящий сервер объектов с кодом операции чтения и ссылкой на объект, полученной от сервера имен.

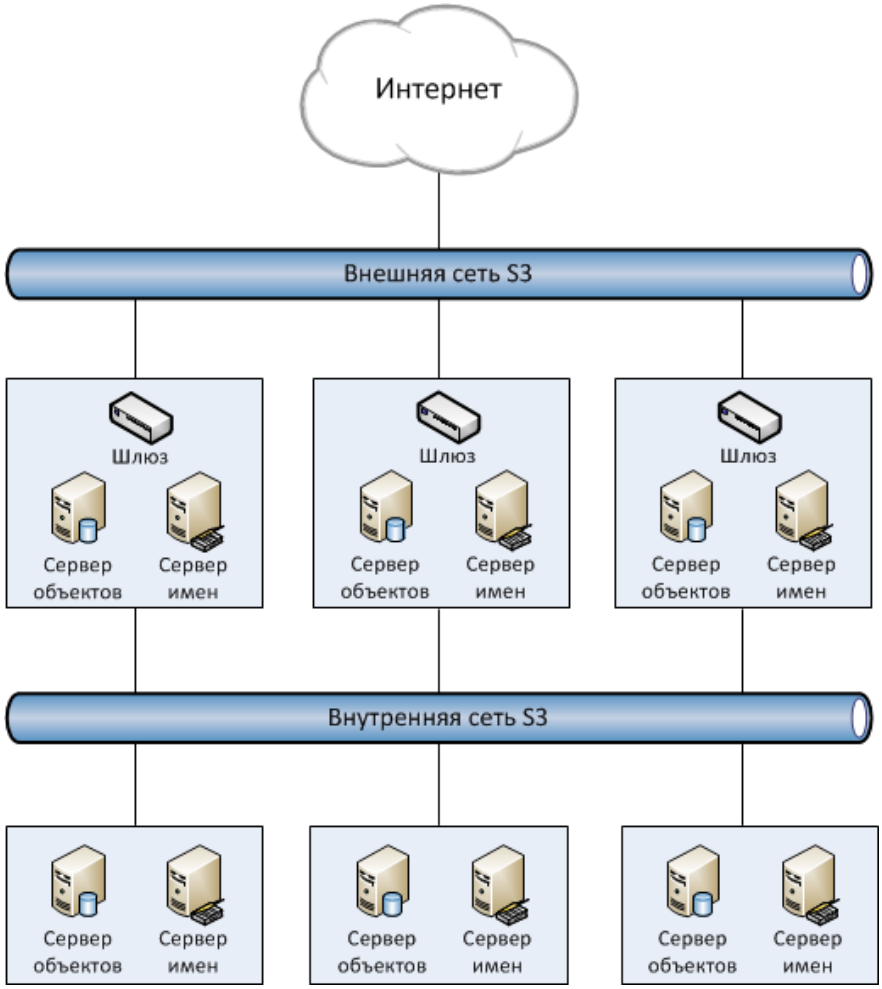
Для выполнения запроса сервер объектов передает на шлюз сообщения фиксированного размера с данными объекта. Последнее сообщение включает в себя флаг окончания передачи данных.

Операция удаления

Чтобы удалить объект (и его имя) из хранилища, шлюз определяет идентификатор сервера имен на основе части имени, содержащей директорию, и отправляет запрос с кодом операции удаления на сервер. В свою очередь, сервер имен удаляет имя из своих структур и отправляет ответ. После некоторого времени программа очистки памяти удаляет соответствующий объект из хранилища.

Создание объектного хранилища

В данном разделе описывается, как развернуть объектное хранилище поверх готового кластера ПК Р-Хранилище. В результате у вас получится создать установку, как показано на рисунке ниже. Следует отметить, что службы объектного хранилища могут быть запущены не на всех серверах кластера. Выбор должен быть основан на загрузке и аппаратной конфигурации кластера.



Для установки служб объектного хранилища необходимо выполнить следующие действия:

1 Составьте план сети S3. Как и для кластера ПК Р-Хранилище, для кластера объектного хранилища требуется две сети:

- Внутренняя сеть, в которой будут взаимодействовать сервер имен, сервер объектов и шлюз. Данные службы будут генерировать трафик, по объему похожий на весь (входящий и исходящий) трафик пользователей S3. Если объем трафика не будет большим, то можно использовать одну внутреннюю сеть для объектного хранилища и для ПК Р-Хранилище. Однако если вы предполагаете, что объем трафика будет значительным, то следует настроить трафик S3 через сеть данных пользователей (т.е. сеть центра обработки данных). После выбора сети для трафика S3 нужно определить, какие IP-адреса можно использовать для добавления серверов кластера.
- Внешняя (общедоступная) сеть, через которую конечные пользователи будут иметь доступ к хранилищу S3. В этой сети должны быть открыты стандартные порты HTTP и HTTPS.

Кластер объектного хранилища почти не зависит от блочного хранилища (как и все точки доступа, включая виртуальные среды и iSCSI). Серверы объектов и имен хранят свои данные в кластере ПК Р-Хранилище так же, как и виртуальные среды, iSCSI и другие службы. Таким образом, службы сервера объектов и имен зависят от vstorage-mount (клиента) и могут быть запущены, только когда кластер монтирован. В отличие от них, шлюз не имеет своих данных, поэтому он не зависит от vstorage-mount и теоретически может быть запущен даже на серверах, где кластер ПК Р-Хранилище не монтирован. Однако для простоты рекомендуется создание шлюзов на серверах с серверами имен и объектов.

Серверы имен и объектов также используют стандартные способы высокой доступности ПК Р-Хранилище (т.е. службу `shaman`). Как виртуальные среды и iSCSI, серверы объектов и имен подписаны на события кластера, относящиеся к высокой доступности. Однако, в отличие от других служб, компонентами кластера S3 невозможно управлять (отслеживать и перемещать между серверами) с помощью `shaman`. Вместо этого, управление осуществляется при помощи службы конфигурации S3, которая подписана на события кластера, относящиеся к высокой доступности, и получает уведомления от `shaman` о работоспособности серверов. По этой причине компоненты кластера S3 не отображаются в выводе команды `shaman top`.

Службы шлюза, не сохраняющие данные о своем состоянии, никогда не перемещаются, и их высокой доступностью нельзя управлять при помощи кластера ПК Р-Хранилище. Вместо этого, при необходимости создается новая служба шлюза.

2 Удостоверьтесь, что каждый сервер, на котором будут запущены службы сервера объектов и имен, находится в кластере высокой доступности. Вы можете добавить серверы в кластер высокой доступности с помощью команды `shaman join`.

3 Установите пакет `vstorage-ostor` на каждый сервер кластера.

```
# yum install vstorage-ostor
```

4 Создайте конфигурацию кластера на одном из серверов кластера, где будут запущены службы объектного хранилища. Рекомендуется создавать конфигурации с 10 службами сервера имен и10 службами сервера объектов для каждого сервера. Например, для кластера из пяти серверов понадобятся 50 служб серверов имен и 50 служб серверов объектов. Выполните следующую команду на первом сервере кластера:

```
# ostor-ctl create -r /var/lib/ostor/configuration -n <IP_addr>
```

где <*IP_addr*> является IP-адресом сервера (который относится к внутренней сети S3), который будет прослушивать служба конфигурации,

Вам будет предложено ввести и подтвердить пароль для нового объектного хранилища (он может совпадать с паролем для кластера ПК Р-Хранилище). Данный пароль потребуется для добавления новых серверов.

Служба конфигурации будет хранить конфигурацию кластера локально в `/var/lib/ostor/configuration`. Дополнительно, <*IP_addr*> будет храниться в `/<storage_mount>/<ostor_dir>/control/name` (<*ostor_dir*> - это директория в кластере, в которой хранятся файлы службы объектного хранилища). При отказе первой службы конфигурации (и если команда `ostor-ctl get-config` перестанет работать) замените IP-адрес в `/<storage_mount>/<ostor_dir>/control/name` на IP-адрес сервера с запущенной службой конфигурации (которая создается в шаге 6).

5 Запустите службу конфигурвции.

```
# systemctl start ostor-cfgd.service
# systemctl enable ostor-cfgd.service
```

6 Добавьте как минимум еще две службы конфигурации для избыточности (чтобы вместе иметь как минимум три службы). Служба конфигурации нужна только для добавления в кластер S3 или удаления из него серверов, она не влияет на работу служб S3 и их высокую доступность. Поэтому отказ службы конфигурации не является критически важным для кластера S3. Однако подобный отказ нежелателен, и рекомендуется создать несколько служб конфигурации, чтобы по крайней мере одна служба всегда была запущена.

Чтобы добавить службу конфигурации, выполните следующие команды на сервере, на котором будут запущены службы объектного хранилища. Повторите их, чтобы создать требуемое число служб конфигурации.

```
# ostor-ctl join -n <remote_IP_addr> -a <local_IP_addr>
# systemctl start ostor-cfgd.service
# systemctl enable ostor-cfgd.service
```

где <*remote_IP_addr*> является <*IP_addr*> из шага 4.

Каждая добавленная служба конфигурации будет хранить конфигурацию кластера локально в `/var/lib/ostor/configuration`.

7 Инициализируйте новое объектное хранилище на первом сервере. В разделе root кластера будет создана директория `<ostor_dir>`.

```
# ostor-ctl init-storage -n <IP_addr> -s <cluster_mount_point>
```

Вам нужно будет предоставить IP-адрес и пароль для объектного хранилища, указанные в шаге 3.

8 Добавьте к DNS общедоступные IP-адреса серверов, на которых будут запущены службы шлюза. Вы можете настроить DNS, чтобы открыть доступ к вашему объектному хранилищу по имени хоста и чтобы конечная точка S3 могла получать запросы REST API со стилем виртуального хостинга с URI наподобие http://bucketname.s3.example.com/objectname.

После настройки DNS убедитесь, что преобразователь DNS для конечной точки S3 работает на клиентских машинах.

Примечание: Только к корзинам с именами, совместимыми с DNS, можно получить доступ при помощи запросов со стилем виртуального хостинга. Для получения дополнительной информации см. **Политики именования корзин и ключей** (стр. 81).

Ниже приведен пример конфигурационного файла зон DNS для сервера BIND DNS:

```
;$Id$
$TTL 1h  @  IN   SOA    ns.example.com. s3.example.com. (
                        2013052112      ; serial
                        1h      ; refresh
                        30m     ; retry
                        7d      ; expiration
                        1h )    ; minimum
                NS      ns.example.com.
$ORIGIN s3.example.com
h1 IN   A 10.29.1.95
        A 10.29.0.142
        A 10.29.0.137
* IN    CNAME    @
```

Данная конфигурация указывает DNS перенаправлять все запросы с URI http://s3.example.com и их дочерние домены (http://*.s3.example.com/*) на одну из конечных точек, перечисленных в записи ресурса h1 (10.29.1.95, 10.29.0.142 или 10.29.0.137) циклическим (круговым) способом.

9 Добавьте в конфигурацию серверы, на которых будут запущены службы объектного хранилища.

Примечание: Добавление серверов в существующие кластеры осуществляется так же путем выполнения шагов 8-12.

Для этого выполните команду `ostor-ctl add-host` на каждом подобном сервере:

```
# ostor-ctl add-host -r /var/lib/ostor/configuration --hostname <name> --roles OBJ
```

Вам нужно будет предоставить пароль для объектного хранилища, указанный в шаге 3.

Примечание: Если вы хотите, чтобы служба агента объектного хранилища прослушивала внутренний IP-адрес, добавьте опцию `-H <internal_IP_address>` к команде выше.

10 Создайте новый том S3 с необходимым числом служб серверов имен и объектов:

```
# ostor-ctl add-vol --type OBJ -s <cluster_mount_point> --os-count <OS_num> \
--ns-count <NS_num> --vstorage-attr "failure-domain=host,tier=0,replicas=3"
```

где

- *<NS_num>* и *<OS_num>* являются числом служб серверов имен и объектов, а
- `failure-domain=host`, `tier=0`, `replicas=3` – параметрами, задающими для тома область отказа, уровень и режим избыточности (для получения подробной информации см. **Обзор параметров кластера** (стр. 28**)**).

В выводе команды будет указан идентификатор созданного тома. Он вам понадобится в следующем шаге.

11 Создайте копии шлюза S3 на выбранных серверах, имеющих доступ к сети Интернет и внешние IP-адреса. Рекомендуется создать 4 шлюза на каждом сервере.

Примечание: В целях безопасности, убедитесь, что только `nginx` имеет доступ к внешней сети и что шлюзы S3 прослушивают только внутренние IP-адреса.

```
# ostor-ctl add-s3gw -a <internal_IP_address>:<port> -V <volume_ID>
```

где

- *<internal_IP_address>* является внутренним IP-адресом сервера со шлюзом,
- *<port>* (обязательный параметр) - неиспользуемый порт, уникальный для каждой копии шлюза на сервере,
- <volume_ID> - идентификатор тома, который был создан в предыдущем шаге (также его можно узнать с помощью `ostor-ctl get-config`).

Например:

```
# ostor-ctl add-s3gw -a 127.0.0.1:9001 -V 0100000000000001
# ostor-ctl add-s3gw -a 127.0.0.1:9002 -V 0100000000000001
# ostor-ctl add-s3gw -a 127.0.0.1:9003 -V 0100000000000001
# ostor-ctl add-s3gw -a 127.0.0.1:9004 -V 0100000000000001
```

12 Запустите агента объектного хранилища на каждом сервере кластера, добавленном в конфигурацию объектного хранилища.

```
# systemctl start ostor-agentd.service
# systemctl enable ostor-agentd.service
```

13 Убедитесь, что службы серверов имен и объектов привязаны к серверам.

По умолчанию агенты попробуют автоматически привязать службы серверов имен и объектов к серверам круговым способом. Однако, если новый хост был добавлен в конфигурацию или текущая конфигурация не оптимизирована, то необходима привязка вручную (для получения подробной информации см. **Привязка служб к серверам вручную** (стр. 73)).

Вы можете проверить текущую конфигурацию привязки с помощью команды `ostor-ctl agent-status`. Например:

```
# ostor-ctl agent-status
TYPE     SVC_ID               STATUS          UPTIME  HOST_ID            ADDRS
S3GW     8000000000000009     ACTIVE             527  fcbf5602197245da   127.0.0.1:9090
S3GW     8000000000000008     ACTIVE             536  4f0038db65274507   127.0.0.1:9090
S3GW     8000000000000007     ACTIVE             572  958e982fcc794e58   127.0.0.1:9090
OS       1000000000000005     ACTIVE             452  4f0038db65274507
10.30.29.124:39746
OS       1000000000000004     ACTIVE             647  fcbf5602197245da
10.30.27.69:56363
OS       1000000000000003     ACTIVE             452  4f0038db65274507
10.30.29.124:52831
NS       0800000000000002     ACTIVE             647  fcbf5602197245da
10.30.27.69:56463
NS       0800000000000001     ACTIVE             452  4f0038db65274507
10.30.29.124:53044
NS       0800000000000000     ACTIVE             647  fcbf5602197245da
10.30.27.69:37876
```

14 На каждом сервере с копиями шлюза установите `nginx` для обслуживания запросов S3 от конечных пользователей:

```
# yum install nginx
```

Используйте инструмент `ostor-configure-nginx`, чтобы настроить `nginx` для S3. Существуют две опции конфигурации:

- `create` для создания новой конфигурации и

- `update` для обновления существующей конфигурации; используйте ее после изменения конфигурации шлюза.

Для начальной конфигурации `nginx` используйте `create`. Например, для HTTP:

```
# ostor-configure-nginx create -D s3.mydomain.com -p 80
```

где `s3.mydomain.com` является доменом конечной точки S3, а `80` – портом, который будет прослушивать `nginx`.

Чтобы настроить HTTP с SSL-сертификатом для домена конечной точки S3 и его дочерних доменов, укажите сертификат и ключ. Например:

```
# ostor-configure-nginx create -D s3.mydomain.com -p 443 **ssl **ssl-cert file.cert
**ssl-key file.key
```

Будет создан файл конфигурации `/etc/nginx/conf.d/ostor-s3.conf`. Он будет заниматься перенаправлением FastCGI на локальные копии шлюзов.

15 Запустите `nginx`:

```
# systemctl start nginx.service
# systemctl enable nginx.service
```

16 Добавьте в кластер S3 серверы, которые находятся в кластере высокой доступности, но на которых не будут запущены службы S3. Другими словами, убедитесь, что все серверы из кластера высокой доступности также включены в кластер S3. Это нужно для правильной работы высокой доступности.

```
# ostor-ctl add-host -n <IP_addr>
# systemctl start ostor-agentd.service
# systemctl enable ostor-agentd.service
```

Объектное хранилище развернуто. Теперь вы можете добавить пользователей S3, используя инструмент `ostor-s3-admin`, как описано в разделе **Создание пользователей S3** (стр. 75).

Чтобы проверить, что установка завершена успешно, или посмотреть статус объектного хранилища, используйте команду `ostor-ctl get-config`. Например:

```
# ostor-ctl get-config
07-08-15 11:58:45.470 Use configuration service 'ostor'
SVC_ID             TYPE  URI
8000000000000006   S3GW  svc://1039c0dc90d64607/?address=127.0.0.1:9000
0800000000000000     NS  vstorage://cluster1/ostor/services/0800000000000000
1000000000000001     OS  vstorage://cluster1/ostor/services/1000000000000001
1000000000000002     OS  vstorage://cluster1/ostor/services/1000000000000002
1000000000000003     OS  vstorage://cluster1/ostor/services/1000000000000003
1000000000000004     OS  vstorage://cluster1/ostor/services/1000000000000004
8000000000000009   S3GW  svc://7a1789d20d9f4490/?address=127.0.0.1:9000
800000000000000c   S3GW  svc://7a1789d20d9f4490/?address=127.0.0.1:9090
```

Привязка служб к серверам вручную

При развертывании объектного хранилища вы можете вручную привязать службы к серверам с помощью команды `ostor-ctl bind`. Для привязки необходимо указать идентификатор целевого сервера и идентификатор одной или более служб. Например, команда:

```
# ostor-ctl bind -H 4f0038db65274507 -S 0800000000000001 -S 1000000000000003 -S
1000000000000005
```

привязывает службы с идентификаторами 800000000000001, 1000000000000003 и 1000000000000005 к хосту с идентификатором 4f0038db65274507.

Службу можно привязать только к хосту, соединенному с общим хранилищем, в котором хранятся все данные этой службы. Другими словами, имя кластера в URI службы должно совпадать с именем кластера в URI хоста.

Например, в конфигурации с двумя общими хранилищами stor1 и stor2 (см. ниже) службы с URI, начинающимися с vstorage://stor1, можно привязать только к хостам host510 и host511, а службы с URI, начинающимися с vstorage://stor2, можно привязать только к хостам host512 и host513.

```
# ostor-ctl get-config
SVC_ID              TYPE  URI
0800000000000000      NS  vstorage://stor1/s3-data/services/0800000000000000
0800000000000001      NS  vstorage://stor1/s3-data/services/0800000000000001
0800000000000002      NS  vstorage://stor2/s3-data/services/0800000000000002
1000000000000003      OS  vstorage://stor1/s3-data/services/1000000000000003
1000000000000004      OS  vstorage://stor2/s3-data/services/1000000000000004
1000000000000005      OS  vstorage://stor1/s3-data/services/1000000000000005
HOST_ID             HOSTNAME      URI
0fcbf5602197245da   host510:2530  vstorage://stor1/s3-data
4f0038db65274507    host511:2530  vstorage://stor1/s3-data
958e982fcc794e58    host512:2530  vstorage://stor2/s3-data
953e976abc773451    host513:2530  vstorage://stor2/s3-data
```

Управление пользователями S3

Понятие пользователя S3 является одним из основных понятий объектного хранилища вместе с понятиями объекта и корзины (контейнера для хранения объектов). Протокол Amazon S3 использует модель разрешений на базе списков контроля доступа (ACL), где каждой корзине и каждому объекту назначен список ACL, в котором перечислены все пользователи с правом доступа к данному ресурсу, а также тип доступа (чтение, запись, чтение ACL, запись ACL). Список пользователей включает владельца сущности, который назначается каждому объекту и каждой корзине при создании. Владелец сущности имеет дополнительные права, по сравнению с другими пользователями, например, только владелец корзины может ее удалить.

Модель пользователей и политик доступа, реализованная в объектном хранилище Р-Хранилище, соответствуют модели пользователей и политик доступа Amazon S3.

Сценарии управления пользователями в объектном хранилище Р-Хранилище во многом основаны на управлении пользователями в Amazon Web Services и включают следующие операции: создание, запрос и удаление пользователей, а также генерация и аннулирование пар ключей доступа для пользователей.

Пользователями можно управлять при помощи инструмента ostor-s3-admin. Для этого нужно знать идентификатор тома, в котором хранятся пользователи. Идентификатор можно узнать, выполнив команду ostor-ctl get-config. Например:

```
# ostor-ctl get-config -n 10.94.97.195
VOL_ID              TYPE     STATE
```

```
0100000000000002   OBJ     READY
...
```

Примечание: Инструмент `ostor-s3-admin` предназначен для использования администраторами объектного хранилища, поэтому данные команды не включают проверки аутентификации и идентификации.

Создание пользователей S3

Вы можете сгенерировать уникальный идентификатор для любого пользователя S3, а также пару ключей доступа (идентификатор ключа доступа S3, секретный ключ доступа S3), используя команду `ostor-s3-admin create-user`. Для команды необходимо указать адрес электронной почты пользователя. Например:

```
# ostor-s3-admin create-user -e user@email.com -V 0100000000000002
UserEmail:user@email.com
UserId:a49e12a226bd760f
KeyPair[0]:S3AccessKeyId:a49e12a226bd760fGHQ7
KeyPair[0]:S3SecretAccessKey:HSDu2DA00JNGjnRcAhLKfhrvlymzOVdLPsCK2dcq
Flags:none
```

Идентификатор пользователя S3 представляет собой 16-значную шестнадцатеричную строку. Сгенерированная пара ключей доступа используется для подписи запросов к объектному хранилищу S3 в соответствии со схемой аутентификации Amazon S3 Signature Version 2.

Просмотр списка пользователей S3

Вы можете отобразить всех пользователей объектного хранилища в виде списка с помощью команды `ostor-s3-admin query-users`. Информация о каждом пользователе может занимать одну или несколько последовательных строк в таблице. Дополнительные строки используются для вывода списка пар ключей доступа S3, относящихся к пользователю. Если у пользователя нет активных пар ключей, то в соответствующих ячейках отображается знак минуса. Например:

```
# ostor-s3-admin query-users -V 0100000000000002
      S3 USER ID      S3 ACCESS KEY ID              S3 SECRET ACCESS KEY  S3 USER EMAIL
bf0b3b15eb7c9019  bf0b3b15eb7c9019I36Y                              ***  user2@abc.com
d866d9d114cc3d20  d866d9d114cc3d20G456                              ***  user1@abc.com
                  d866d9d114cc3d20D8EW                              ***
e86d1c19e616455                     -                                -  user3@abc.com
```

Чтобы получить список в XML, используйте опцию `-X`; для вывода секретных ключей используйте опцию `-a`. Например:

```
# ostor-s3-admin query-users -V 0100000000000002 -a -X
<?xml version="1.0" encoding="UTF-
8"?><QueryUsersResult><Users><User><Id>a49e12a226bd760f</Id><Email>user@email.com</Emai
l><Keys><OwnerId>0000000000000000</OwnerId><KeyPair><S3AccessKeyId>a49e12a226bd760fGHQ7
</S3AccessKeyId><S3SecretAccessKey>HSDu2DA00JNGjnRcAhLKfhrvlymzOVdLPsCK2dcq</S3SecretAc
cessKey></KeyPair></Keys></User><User><Id>d7c53fc1f931661f</Id><Email>user@email.com</E
mail><Keys><OwnerId>0000000000000000</OwnerId><KeyPair><S3AccessKeyId>d7c53fc1f931661fZ
LIV</S3AccessKeyId><S3SecretAccessKey>JL7gt1OH873zR0Fzv8Oh9ZuA6JtCVnkgV7lET6ET</S3Secre
tAccessKey></KeyPair></Keys></User></Users></QueryUsersResult>
```

Запрос информации о пользователе S3

Чтобы отобразить информацию о конкретном пользователе, используйте команду `ostor-s3-admin query-user-info`. Для команды необходимо указать адрес электронной почты пользователя (`-e`) или идентификатор S3 (`-i`). Например:

```
# ostor-s3-admin query-user-info -e user@email.com -V 0100000000000002
Query user: user id=d866d9d114cc3d20, user email=user@email.com
Key pair[0]: access key id=d866d9d114cc3d20G456,
secret access key=5EAne6PLL1jxprouRqq8hmfONMfgrJcOwbowCoTt
Key pair[1]: access key id=d866d9d114cc3d20D8EW,
secret access key=83tTsNAuuRyoBBqhxMFqHAC60dhKHtTCCkQe54zu
```

Отключение пользователей S3

Вы можете отключить пользователя, используя команду `ostor-s3-admin disable-user`. Для команды необходимо указать адрес электронной почты пользователя (`-e`) или идентификатор S3 (`-i`). Например:

```
# ostor-s3-admin disable-user -e user@email.com -V 0100000000000002
```

Удаление пользователей S3

Вы можете удалить существующих пользователей объектного хранилища с помощью команды `ostor-s3-admin delete-user`. Пользователей, у которых есть корзины, невозможно удалить, поэтому следует сначала удалить корзины пользователя. Для команды необходимо указать адрес электронной почты пользователя (`-e`) или идентификатор S3 (`-i`). Например:

```
# ostor-s3-admin delete-user -i bf0b3b15eb7c9019 -V 0100000000000002
Deleted user: user id=bf0b3b15eb7c9019
```

Генерация пар ключей доступа для пользователей S3

Вы можете сгенерировать новую пару ключей доступа для определенного пользователя, используя команду `ostor-s3-admin gen-access-key`. Каждому пользователю разрешено иметь максимум 2 активные пары ключей доступа (аналогично Amazon Web Services). Для команды необходимо указать адрес электронной почты пользователя (`-e`) или идентификатор S3 (`-i`). Например:

```
# ostor-s3-admin gen-access-key -e user@email.com -V 0100000000000002
Generate access key: user id=d866d9d114cc3d20, access key id=d866d9d114cc3d20D8EW,
secret access key=83tTsNAuuRyoBBqhxMFqHAC60dhKHtTCCkQe54zu
```

Рекомендуется периодически аннулировать старые пары ключей доступа и генерировать новые.

Аннулирование пар ключей доступа для пользователей S3

Чтобы аннулировать указанную пару ключей доступа для указанного пользователя, используйте команду `ostor-s3-admin revoke-access-key`. Для команды необходимо указать ключ доступа в паре ключей, которую вы хотите удалить, а также адрес электронной почты пользователя или идентификатор S3. Например:

```
# ostor-s3-admin revoke-access-key -e user@email.com -k de86d1c19e616455YIPU -V
0100000000000002
Revoke access key: user id=de86d1c19e616455, access key id=de86d1c19e616455YIPU
```

Рекомендуется периодически аннулировать старые пары ключей доступа и генерировать новые.

Управление корзинами объектного хранилища

Все объекты в хранилище наподобие Amazon S3 хранятся в контейнерах, называемых корзинами. Корзины идентифицируются по именам, которые уникальны в одном объектном хранилище, таким образом, пользователь S3 объектного хранилища не может создать корзину с тем же именем, которое имеет другая корзина из этого объектного хранилища. Корзины используются, чтобы:

- сгруппировать и изолировать объекты от объектов из других корзин,
- предоставить механизмы управления ACL для объектов в них,
- установить политики доступа для отдельных корзин, например, управление версиями в корзине.

Корзинами можно управлять при помощи инструмента `ostor-s3-admin`, а также сторонних браузеров S3, например, CyberDuck или DragonDisk. Инструмент `ostor-s3-admin` предназначен для использования администраторами объектного хранилища, поэтому данные команды не включают проверки аутентификации и идентификации. Рекомендуется сначала использовать стандартные команды API Amazon S3.

Для управления корзинами через инструменты командной строки нужно знать идентификатор тома, в котором хранятся пользователи. Идентификатор можно узнать, выполнив команду `ostor-ctl get-config`. Например:

```
# ostor-ctl get-config -n 10.94.97.195
VOL_ID              TYPE     STATE
0100000000000002    OBJ     READY
...
```

Внимание: Команды для выполнения операций изменения и удаления корзин не входят в стандартный API S3 и могут нарушить интеграцию с внешними биллинговыми системами и системами бухгалтерского учета. Их использование должно быть хорошо обоснованно и обдуманно.

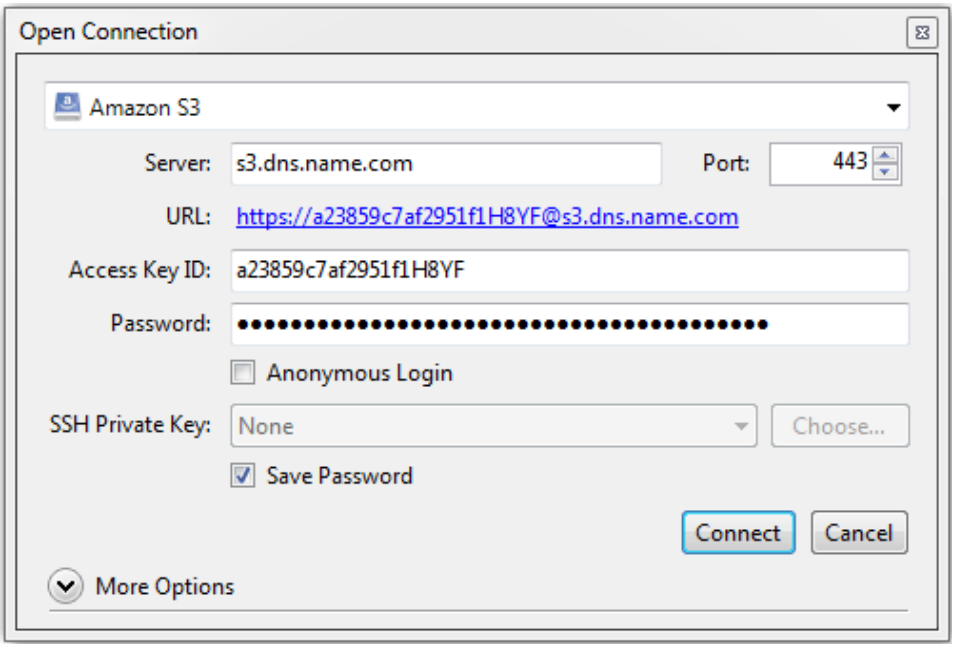
Управление корзинами с помощью CyberDuck

Создание корзин

Чтобы создать новую корзину S3 при помощи CyberDuck, необходимо выполнить следующие действия:

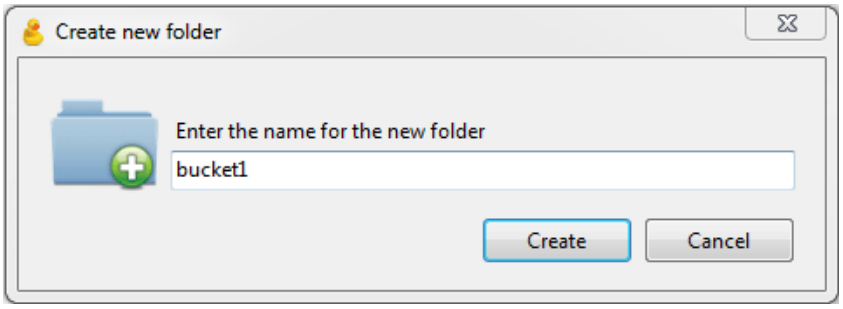
1. Щелкните **Open Connection**.
2. Укажите следующие параметры:

- Внешнее доменное имя для конечной точки S3, которое было указано при создании кластера S3.
- Идентификатор ключа доступа и секретный ключ доступа пользователя объектного хранилища (см. **Создание пользователей S3** (стр. 75)).



По умолчанию соединение устанавливается по HTTPS. Для использования CyberDuck поверх HTTP нужно установить специальный профиль S3 по ссылке https://trac.cyberduck.io/wiki/help/en/howto/s3.

3 Когда соединение будет установлено, щелкните **File** > **New Folder**.



4 Укажите имя новой корзины и щелкните **Create**.

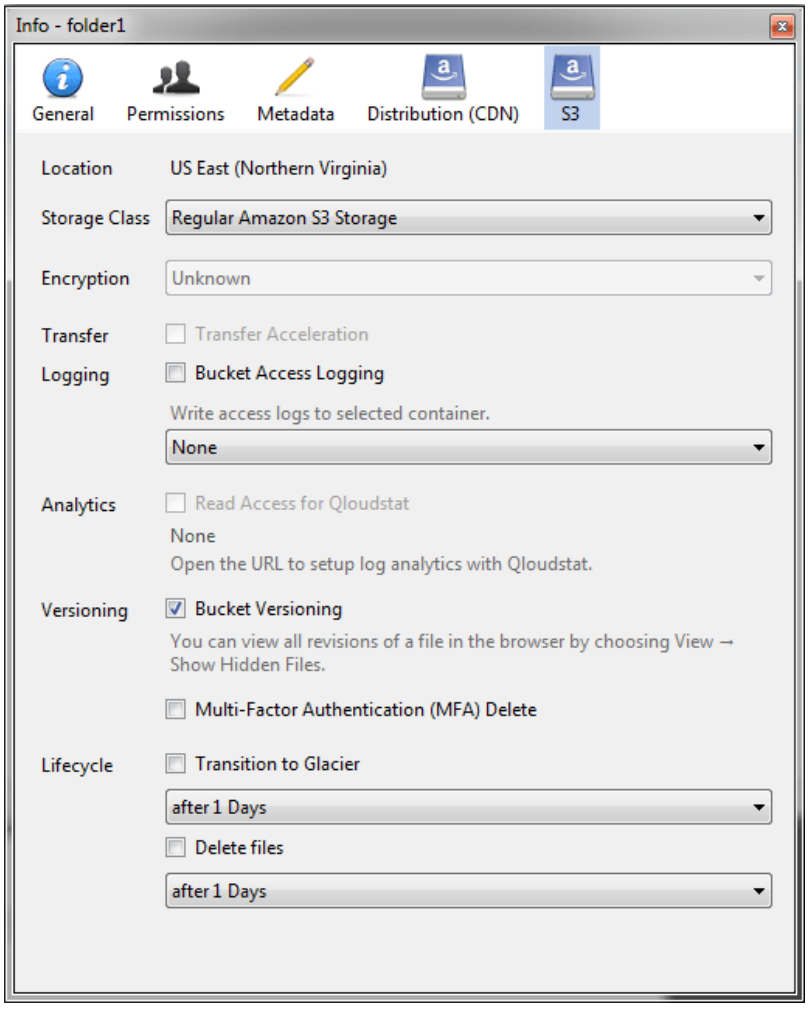
Примечание: Рекомендуется использовать имена корзин, которые соответствуют общепринятым нормам именования DNS. Для получения подробной информации по именованию корзин см. **Политики именования корзин и ключей** (стр. 81).

Новая корзина появится в CyberDuck, где можно будет управлять ею и загружать в нее файлы.

Управление версиями корзин

Управление версиями представляет собой способ хранения нескольких вариантов объекта в одной корзине. Управление версиями можно использовать для сохранения, извлечения и восстановления каждой версии каждого объекта, который хранится в корзине S3. При помощи управления версиями можно легко восстановить объект после непреднамеренных действий пользователя или отказа приложения. Для получения дополнительной информации по управлению версиями корзин см. Using Versioning.

Управление версиями корзин отключено по умолчанию. Вы можете его включить в стороннем браузере S3, поставив галочку в свойствах корзины. Например:



Просмотр содержимого корзины

В веб-браузере можно просмотреть содержимое корзины. Для этого пройдите по ссылке, состоящей из внешнего доменного имени для конечной точки S3, которое было указано при создании кластера S3, и имени корзины. Например, `mys3storage.example.com/mybucket`.

Примечание: Вы также можете скопировать ссылку на содержимое корзины, щелкнув правой кнопкой мыши по корзине в CyberDuck и выбрав **Copy URL**.

Управление корзинами через командную строку

Просмотр списка корзин объектного хранилища

Вы можете просмотреть список всех корзин в объектном хранилище S3 с помощью команды `ostor-s3-admin list-all-buckets`. В выводе команды для каждой корзины указан владелец, дата создания, статус версий и общий размер (размер всех объектов, хранящихся в корзине, а также размер всех незаконченных многопоточных загрузок для данной корзины). Например:

```
# ostor-s3-admin list-all-buckets -V 0100000000000002
Total 3 buckets
BUCKET                OWNER             CREATION_DATE  VERSIONING     TOTAL SIZE, BYTES
bucket1    968d1a79968d1a79  2015-08-18T09:32:35.000Z        none                  1024
bucket2    968d1a79968d1a79  2015-08-18T09:18:20.000Z     enabled                     0
bucket3    968d1a79968d1a79  2015-08-18T09:22:15.000Z   suspended               1024000
```

Чтобы отобразить список в XML, используйте опцию `-X`. Например:

```
# ostor-s3-admin list-all-buckets -X
<?xml version="1.0" encoding="UTF-
8"?><ListBucketsResult><Buckets><Bucket><Name>bucker2</Name><Owner>d7c53fc1f931661f</Ow
ner><CreationDate>2017-04-
03T17:11:44.000Z</CreationDate><Versioning>none</Versioning><Notary>off</Notary><TotalS
ize>0</TotalSize></Bucket><Bucket><Name>bucket1</Name><Owner>d7c53fc1f931661f</Owner><C
reationDate>2017-04-
03T17:11:33.000Z</CreationDate><Versioning>none</Versioning><Notary>off</Notary><TotalS
ize>0</TotalSize></Bucket></Buckets></ListBucketsResult>
```

Чтобы отфильтровать корзины по пользователю, которому они принадлежат, используйте опцию `-i`. Например:

```
# ostor-s3-admin list-all-buckets -i d7c53fc1f931661f
BUCKET   OWNER             CREATION_DATE             VERSIONING  TOTAL_SIZE NOTARY
NOTARY_PROVIDER
bucker2  d7c53fc1f931661f  2017-04-03T17:11:44.000Z  none        0          off    0
```

Запрос информации о корзине объектного хранилища

Вы можете запросить метаданные и ACL для корзины, используя команду `ostor-s3-admin query-bucket-info`. Например, для `bucket1`:

```
# ostor-s3-admin query-bucket-info -b bucket1 -V 0100000000000002
BUCKET   OWNER             CREATION_DATE             VERSIONING  TOTAL_SIZE
bucket1  d339edcf885eeafc  2017-12-21T12:42:46.000Z  none        0

ACL: d339edcf885eeafc: FULL_CONTROL
```

Смена владельцев корзины объектного хранилища

Вы можете передать владение корзиной определенному пользователю при помощи команды `ostor-s3-admin change-bucket-owner`. Например, чтобы сделать пользователя с идентификатором `bf0b3b15eb7c9019` владельцем корзины `bucket1`, выполните следующую команду:

```
# ostor-s3-admin change-bucket-owner -b bucket1 -i bf0b3b15eb7c9019 -V 0100000000000002
Changed owner of the bucket bucket1. New owner bf0b3b15eb7c9019
```

Удаление корзин объектного хранилища

Чтобы удалить указанную корзину, используйте команду `ostor-s3-admin delete-bucket`. Удаление корзины подразумевает удаление всех объектов, находящихся в ней (включая их старые версии), а также все незаконченные многопоточные загрузки для данной корзины. Например:

```
# ostor-s3-admin delete-bucket -b bucket1 -V 0100000000000002
```

Рекомендации по использованию объектного хранилища

В данном разделе предлагаются рекомендации по использованию различных функций объектного хранилища Р-Хранилище. Данные рекомендации помогут вам включить дополнительные функции или улучшить производительность ПК Р-Хранилище.

Политики именования корзин и ключей

Следует использовать имена корзин, которые соответствуют принятым нормам именования DNS:

- содержат от 3 до 63 символов,
- начинаются и заканчиваются со строчной буквы или числа,
- могут содержать строчные буквы, числа, точки (.), тире (-) и подчеркивания (_),
- могут быть последовательностью частей имен (описанных ранее), разделенных точками.

Ключ объекта может быть цепочкой любых символов в кодировке UTF-8 и размером до 1024 байт.

Повышение производительности PUT-операций

В объектное хранилище можно загружать объекты размером до 5 ГБ с помощью одного PUT-запроса и объекты размером до 5 ТБ, используя многопоточную загрузку. Производительность загрузки можно повысить путем разделения больших объектов на части и их параллельной загрузки при помощи API многопоточной загрузки. Данный подход разделяет нагрузку между разными службами сервера объектов.

Рекомендуется использовать многопоточную загрузку для объектов размером более 5 МБ.

Приложения

В данном разделе представлена справочная информация, относящаяся к объектному хранилищу Р-Хранилище.

Приложение А: Поддерживаемые REST-операции Amazon S3

На текущий момент реализация ПК Р-Хранилище протокола Amazon S3 поддерживает следующие REST-операции Amazon S3:

Операции со службами:

- GET Service

Операции с корзинами:

- DELETE Bucket
- GET Bucket (List Objects)
- GET Bucket acl
- GET Bucket location
- GET Bucket Object versions
- GET Bucket versioning
- HEAD Bucket
- List Multipart Uploads
- PUT Bucket
- PUT Bucket acl
- PUT Bucket versioning

Операции с объектами:

- DELETE Object
- DELETE Multiple Objects
- GET Object
- GET Object ACL
- HEAD Object
- POST Object
- PUT Object
- PUT Object - Copy
- PUT Object acl
- Initiate Multipart Upload

- Upload Part
- Complete Multipart Upload
- Abort Multipart Upload
- List Parts

Примечание: Для получения полного списка REST-операций Amazon S3 см. документацию по *Amazon S3 REST API*.

Приложение Б: Поддерживаемые заголовки запросов Amazon

На текущий момент реализация ПК Р-Хранилище протокола Amazon S3 поддерживает следующие заголовки REST-запросов Amazon S3:

- x-amz-acl
- x-amz-delete-marker
- x-amz-grant-full-control
- x-amz-grant-read-acp
- x-amz-grant-read
- x-amz-grant-write
- x-amz-grant-write-acp
- x-amz-meta-**
- x-amz-version-id
- x-amz-copy-source
- x-amz-metadata-directive
- x-amz-copy-source-version-id

Приложение В: Поддерживаемые схемы аутентификации

На текущий момент реализация ПК Р-Хранилище протокола Amazon S3 поддерживает следующие схемы аутентификации:

- Signature Version 2
- Signature Version 4

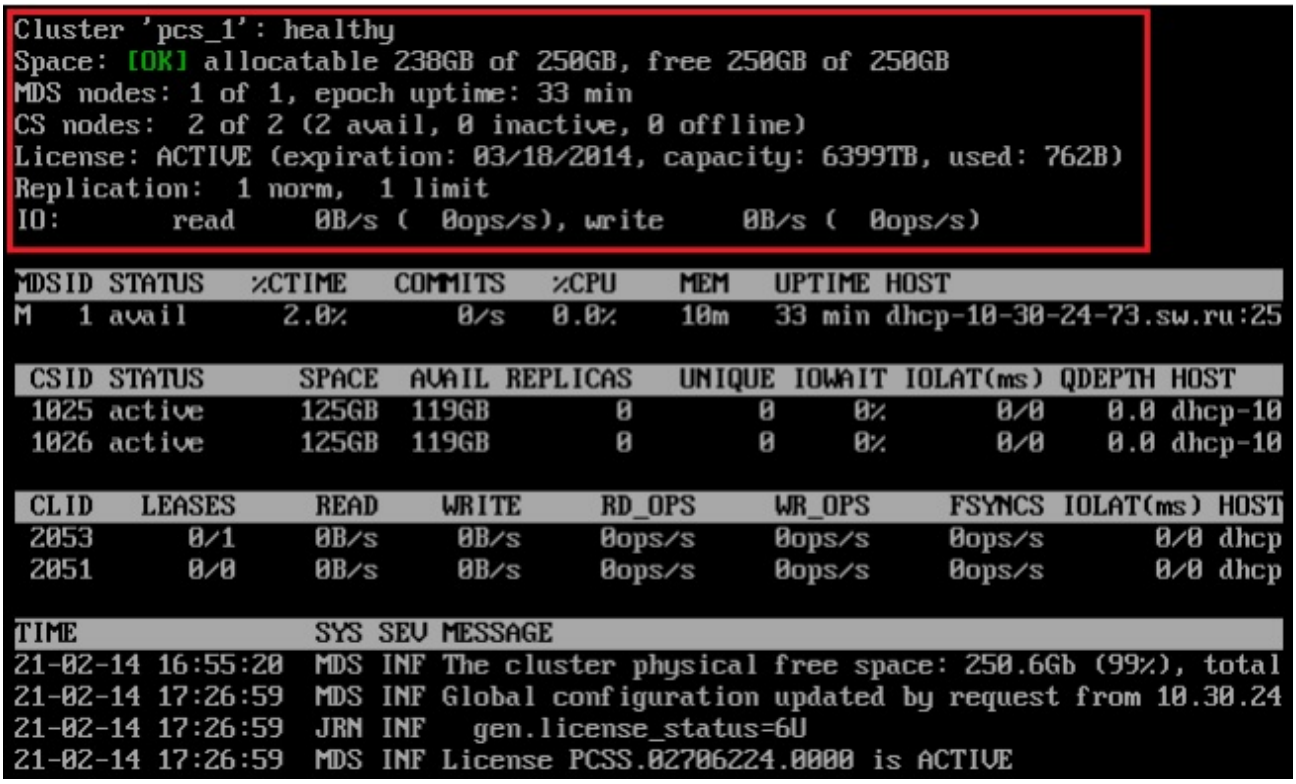
Глава 5

Мониторинг кластеров ПК Р-Хранилище

Мониторинг кластера ПК Р-Хранилище очень важен, так как он позволяет проверить статус и состояние всех компьютеров в кластере и действовать по необходимости.

Мониторинг основных параметров кластера

Мониторинг основных параметров позволяет получить подробную информацию по всем компонентам кластера ПК Р-Хранилище, его общее состояние и статус. Для отображения данной информации можно использовать команду `vstorage -c <cluster_name> top`, например:



Вывод команды, приведенный выше, показывает подробную информацию о кластере `stor1`. Основные параметры (выделенным красным) объясняются в таблице ниже:

Параметр	Описание
Cluster	Общий статус кластера:

	• `healthy`. Все серверы фрагментов в кластере работают. • `unknown`. Недостаточно информации о состоянии кластера (например, из-за того, что master-сервер метаданных был выбран давно). • `degraded`. Некоторые серверы фрагментов в кластере не работают. • `failure`. Большое число серверов фрагментов в кластере не работает; автоматическая репликация отключена. • `SMART warning`. Один или несколько физических дисков, подключенных к серверам кластера, находится в предотказном состоянии. Для получения дополнительной информации см. **Мониторинг физических дисков** (стр. 93).
Space	Размер дискового пространства в кластере: • `free`. Свободное дисковое пространство в кластере. • `allocatable`. Размер логического дискового пространства, доступного для клиентов. Логически доступное дисковое пространство вычисляется, исходя из текущих параметров репликации и свободного дискового пространства на серверах фрагментов. Его размер может быть ограничен лицензией. **Примечание**: Для получения дополнительной информации о мониторинге и использовании дискового пространства см. **Использование дискового пространства** (стр. 88).
MDS nodes	Число работающих серверов метаданных в сравнении с общим числом серверов метаданных, настроенных для кластера.
epoch time	Время, прошедшее с выбора master-сервера метаданных.
CS nodes	Число работающих серверов фрагментов в сравнении с общим числом серверов фрагментов, настроенных для кластера. Информация в круглых скобках показывает количество • Активных серверов фрагментов (`avail`), которые в данный момент запущены и работают в кластере. • Неактивных серверов фрагментов (`inactive`), которые временно не работают. Сервер фрагментов отмечается как **inactive** в течение первых 5 минут неактивности. • Выключенных серверов фрагментов (`offline`), которые не работают более 5 минут. У сервера фрагментов изменяется статус на `offline` после 5 минут неактивности. После изменения статуса на `offline` кластер начинает реплицировать данные, чтобы сохранить те фрагменты, которые хранились на выключенном сервере фрагментов.
License	Номер ключа, под которым зарегистрирована лицензия на сервере ПК Р-Виртуализация Key Authentication, и статус лицензии. Для получения дополнительной информации см. **Управление лицензиями ПК Р-Хранилище** (стр. 37).
Replication	Настройки репликации. Нормальное число реплик фрагментов и ограничение, достигнув которое фрагмент блокируется до восстановления.
IO	Дисковый ввод-вывод в кластере: • Скорость операций чтения и записи, в байтах в секунду. • Количество операций чтения и записи в секунду.

Мониторинг серверов метаданных

Серверы метаданных являются основным компонентом любого кластера ПК Р-Хранилище, поэтому очень важно контролировать их состояние и статус. Для мониторинга серверов метаданных используйте команду `vstorage -c <cluster_name> top`, например:

```
Cluster 'pcs_1': healthy
Space: [OK] allocatable 238GB of 250GB, free 250GB of 250GB
MDS nodes: 1 of 1, epoch uptime: 33 min
CS nodes:  2 of 2 (2 avail, 0 inactive, 0 offline)
License: ACTIVE (expiration: 03/18/2014, capacity: 6399TB, used: 762B)
Replication:  1 norm,  1 limit
IO:       read     0B/s (  0ops/s), write     0B/s (  0ops/s)

MDSID STATUS   %CTIME   COMMITS   %CPU    MEM   UPTIME HOST
M   1 avail      2.0%       0/s   0.0%    10m   33 min dhcp-10-30-24-73.sw.ru:25

 CSID STATUS      SPACE  AVAIL REPLICAS   UNIQUE IOWAIT IOLAT(ms) QDEPTH HOST
 1025 active      125GB  119GB        0        0     0%       0/0    0.0 dhcp-10
 1026 active      125GB  119GB        0        0     0%       0/0    0.0 dhcp-10

 CLID   LEASES     READ    WRITE    RD_OPS    WR_OPS    FSYNCS IOLAT(ms) HOST
 2053      0/1     0B/s     0B/s    0ops/s    0ops/s    0ops/s       0/0 dhcp
 2051      0/0     0B/s     0B/s    0ops/s    0ops/s    0ops/s       0/0 dhcp

TIME                SYS SEV MESSAGE
21-02-14 16:55:20  MDS INF The cluster physical free space: 250.6Gb (99%), total
21-02-14 17:26:59  MDS INF Global configuration updated by request from 10.30.24
21-02-14 17:26:59  JRN INF   gen.license_status=6U
21-02-14 17:26:59  MDS INF License PCSS.02706224.0000 is ACTIVE
```

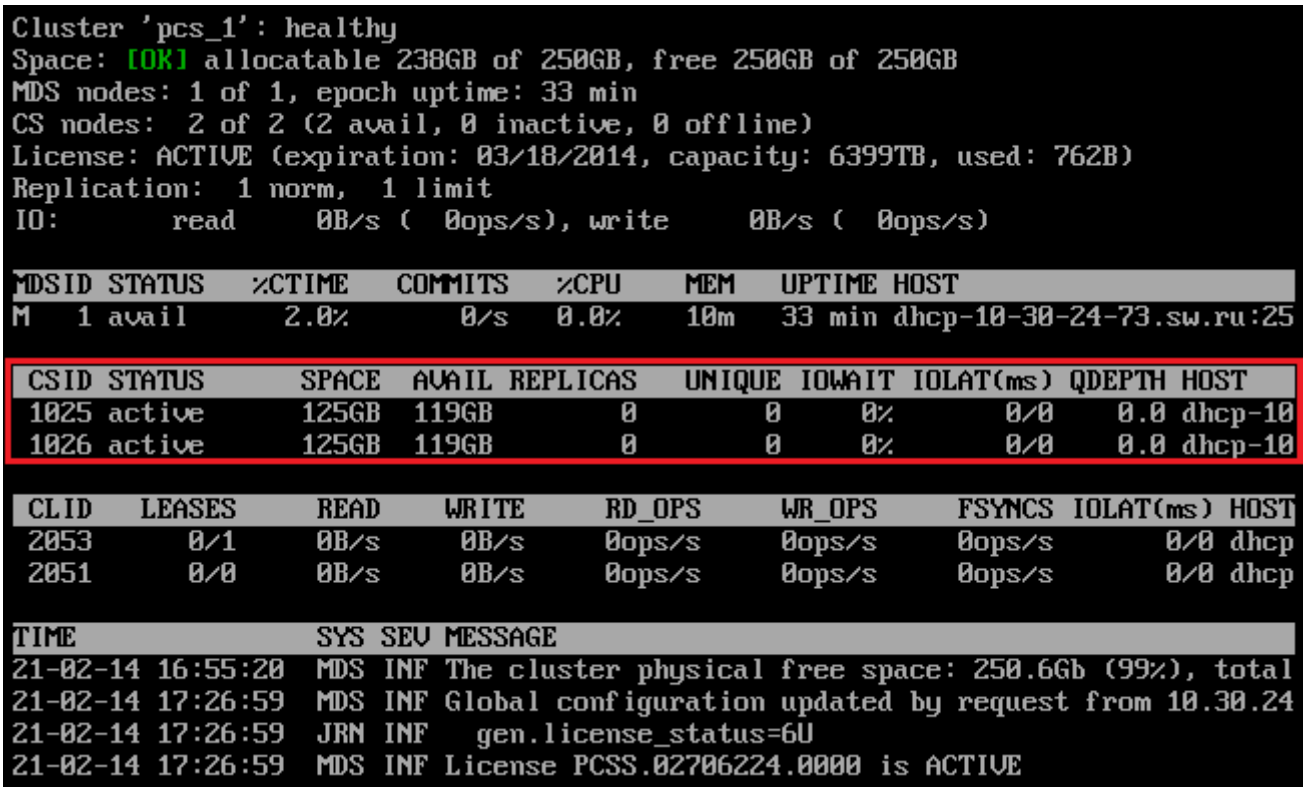
Вывод команды, приведенный выше, показывает подробную информацию о кластере `stor1`. Контролируемые параметры для серверов метаданных (выделенные красным) объясняются в таблице ниже:

Параметр	Описание
MDSID	Идентификатор (ID) сервера метаданных. Если перед ID стоит буква "M", значит, данный сервер является master-сервером метаданных.
STATUS	Статус сервера метаданных.
%CTIME	Общее время, затраченное сервером метаданных на ведение локального журнала.
COMMITS	Поток транзакций локального журнала.
%CPU	Продолжительность работы сервера метаданных.
MEM	Размер физической памяти, используемой сервером метаданных.

UPTIME	Время, прошедшее с последнего запуска сервера метаданных.
HOST	Имя хоста или IP-адрес сервера метаданных.

Мониторинг серверов фрагментов

Мониторинг серверов фрагментов позволяет следить за дисковым пространством, доступным в кластере ПК Р-Хранилище. Для мониторинга серверов фрагментов можно использовать команду `vstorage -c <cluster_name> top`, например:



Вывод команды, приведенный выше, показывает подробную информацию о кластере `stor1`. Контролируемые параметры для серверов фрагментов (выделенные красным) объясняются в таблице ниже:

Параметр	Описание
CSID	Идентификатор (ID) сервера фрагментов.
STATUS	Статус сервера фрагментов: • `active`. Сервер фрагментов запущен и работает. • `inactive`. Сервер фрагментов временно недоступен. Сервер фрагментов отмечается как **inactive** в течение первых 5 минут неактивности. • `offline`. Сервер фрагментов неактивен более 5 минут. После изменения статуса сервера на `offline` кластер начинает реплицировать данные, чтобы сохранить те фрагменты, которые хранились на выключенном сервере

	фрагментов. • `dropped`. Сервер фрагментов был удален администратором.
SPACE	Общий размер дискового пространства на сервере фрагментов.
FREE	Свободное дисковое пространство на сервере фрагментов.
REPLICAS	Количество реплик, которое хранится на сервере фрагментов.
IOWAIT	Процентное отношение времени, затраченное на ожидание обработки операций ввода-вывода.
IOLAT	Среднее/максимальное время, в миллисекундах, которое потребовалось клиенту для завершения одной операции ввода-вывода в течение последних 20 секунд.
QDEPTH	Средняя глубина дисковой очереди на сервере фрагментов.
HOST	Имя хоста или IP-адрес сервера фрагментов.
FLAGS	Для активных серверов фрагментов могут отображаться следующие флаги: • `J`: Сервер фрагментов использует журнал операций записи. • `C`: Для сервера фрагментов включено контрольное суммирование. Контрольное суммирование позволяет определить, когда стороннее приложение изменяет данные на диске. • `D`: Прямой ввод-вывод, нормальное состояние для сервера фрагментов без журнала. • `c`: Журнал сервера фрагментов пуст, в журнале SSD-диска нет записей для подтверждения на жестком диске, на котором находится сервер фрагментов. **Примечание:** Флаги, которые могут отображаться для отказавших серверов фрагментов, описаны в разделе **Отказавшие серверы фрагментов** (стр. 124).

Использование дискового пространства

Информацию по использованию дискового пространства в кластере можно получить при помощи команды `vstorage top`. Данная команда отображает следующую информацию, связанную с диском: общее дисковое пространство, свободное и логически доступное дисковое пространство. Например:

```
# vstorage -c stor1 top
connected to MDS#1
Cluster 'stor1': healthy
Space: [OK] allocatable 180GB of 200GB, free 1.6TB of 1.7TB
...
```

В данном выводе команды:

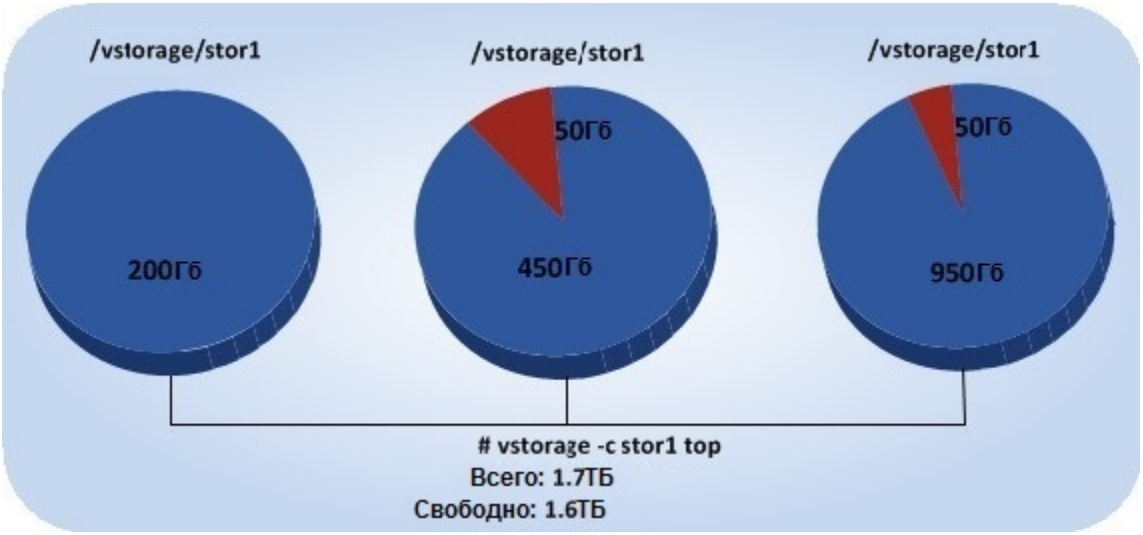
- 1.7 ТБ является размером общего дискового пространства в кластере `stor1`. Размер общего дискового пространства рассчитывается на основе используемого и свободного дискового пространства на всех разделах в кластере. Используемое дисковое пространство включает пространство, занимаемое всеми фрагментами данных и их репликами, а также любыми другими файлами, которые хранятся в разделах кластера.

  Например: Есть раздел диска объемом 100 ГБ, 20 ГБ из которых занимают некоторые файлы. Установка сервера фрагментов в данном разделе добавит 100 ГБ к общему

размеру дискового пространства в кластере, но только 80 ГБ этого дискового пространства будет свободно и доступно для хранения фрагментов данных.

- 1.6 ТБ является размером свободного дискового пространства в кластере `stor1`. Размер свободного дискового пространства вычисляется путем вычитания дискового пространства, занимаемого фрагментами данных и другими файлами в разделах кластера, из размера общего дискового пространства.

  Например, если размер свободного дискового пространства 1.6 ТБ, а размер общего дискового пространства 1.7 ТБ, значит, около 100 ГБ в разделах кластера занимают файлы.



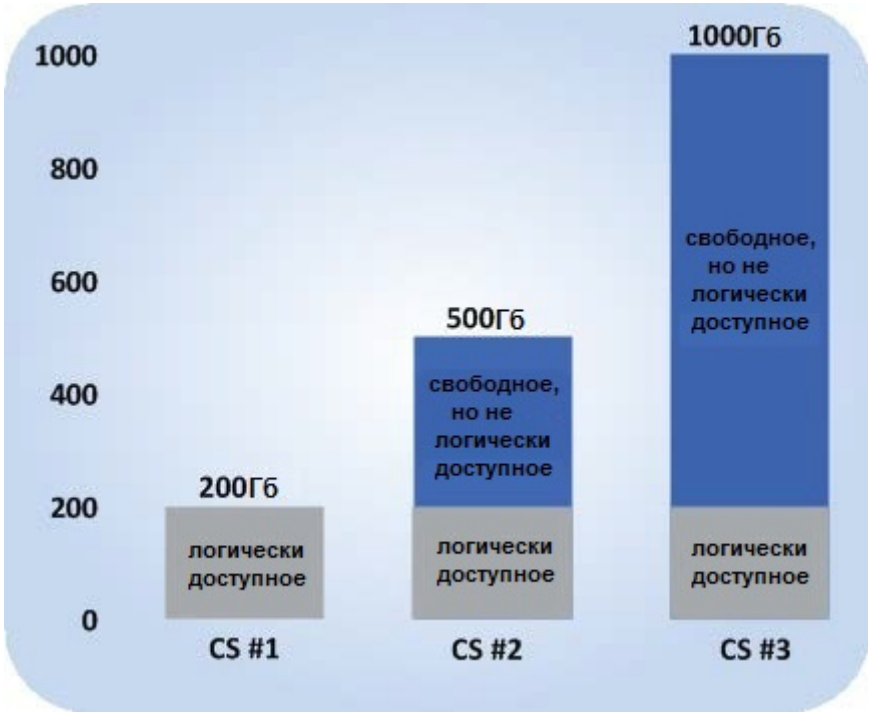
- логически доступные 180 ГБ из 200 ГБ являются размером свободного дискового пространства, которое может быть использовано для хранения фрагментов данных. Для получения подробной информации см. **Понимание логически доступного дискового пространства** ниже.

Понимание логически доступного дискового пространства

При мониторинге информации по использованию дискового пространства в кластере также следует обратить внимание на дисковое пространство, которое отмечается утилитой `vstorage top` как *allocatable*. Логически доступное дисковое пространство представляет собой дисковое пространство, которое свободно и может быть использовано для хранения данных пользователей. Когда данное дисковое пространство заканчивается, в кластер больше невозможно записать данные.

Для лучшего понимания, как вычисляется логически доступное дисковое пространство, можно рассмотреть следующий пример:

- В кластере есть 3 сервера фрагментов: у первого сервера фрагментов 200 ГБ дискового пространства, у второго — 500 ГБ, а у третьего — 1 ТБ.
- Нормальное число реплик в кластере по умолчанию равно 3, значит, для каждого фрагмента данных должно храниться по 3 реплики на разных серверах фрагментов.



В этом примере доступное дисковое пространство будет равно 200 ГБ, так как оно задается по размеру дискового пространства на наименьшем сервере фрагментов:

```
# vstorage -c stor1 top
connected to MDS#1
Cluster 'stor1': healthy
Space: [OK] allocatable 180GB of 200GB, free 1.6TB of 1.7TB
...
```

Данный расчет объясняется тем, что в конфигурации кластера задано, чтобы на каждом сервере хранилась одна реплика для каждого фрагмента данных. Таким образом, когда место на наименьшем сервере фрагментов (200 ГБ) закончится, в кластере невозможно будет создать ни одного фрагмента, пока не будет добавлен новый сервер фрагментов или снижено нормальное число репликаций.

Если изменить нормальное число репликаций на 2, то команда `vstorage top` сообщит о доступном дисковом пространства в 700 ГБ:

```
# vstorage set-attr -R /vstorage/stor1 replicas=2:1
# vstorage -c stor1 top
connected to MDS#1
Cluster 'stor1': healthy
Space: [OK] allocatable 680GB of 700GB, free 1.6TB of 1.7TB
...
```

Размер доступного дискового пространства увеличился, так как теперь для каждого фрагмента данных создаются 2 реплики и новые фрагменты данных могут быть созданы, даже если на наименьшем сервере фрагментов закончится дисковое пространство (в этом случае реплики будут храниться на более крупном сервере фрагментов).

Примечание: Логически доступное дисковое пространство может быть ограничено лицензией.

Просмотр дискового пространства, занимаемого фрагментами данных

Просмотреть общий размер дискового пространства, занимаемого всеми данными пользователей в кластере, можно, выполнив команду `vstorage top` и нажав клавишу **v**. Вывод команды должен выглядеть следующим образом:

```
# vstorage -c stor1 top
Cluster 'stor1': healthy
Space: [OK] allocatable 180GB of 200GB, free 1.6TB of 1.7TB
MDS nodes:    1 of 1, epoch uptime: 2d   4h
CS nodes:     3 of 3 (3 avail, 0 inactive, 0 offline)
Replication:  2 norm,       1 limit,      4 max
Chunks: [OK] 38 (100%) healthy,   0 (0%) degraded,    0 (0%) urgent,
       0 (0%) blocked,     0 (0%) offline,     0 (0%) replicating,
       0 (0%) overcommitted,      0 (0%) deleting,    0 (0%) void
FS:    1GB in 51 files, 51 inodes,       23 file maps, 38 chunks,    76 chunk replicas
...
```

Примечание: В поле **FS** отображается размер всех данных пользователей в кластере без учета реплик.

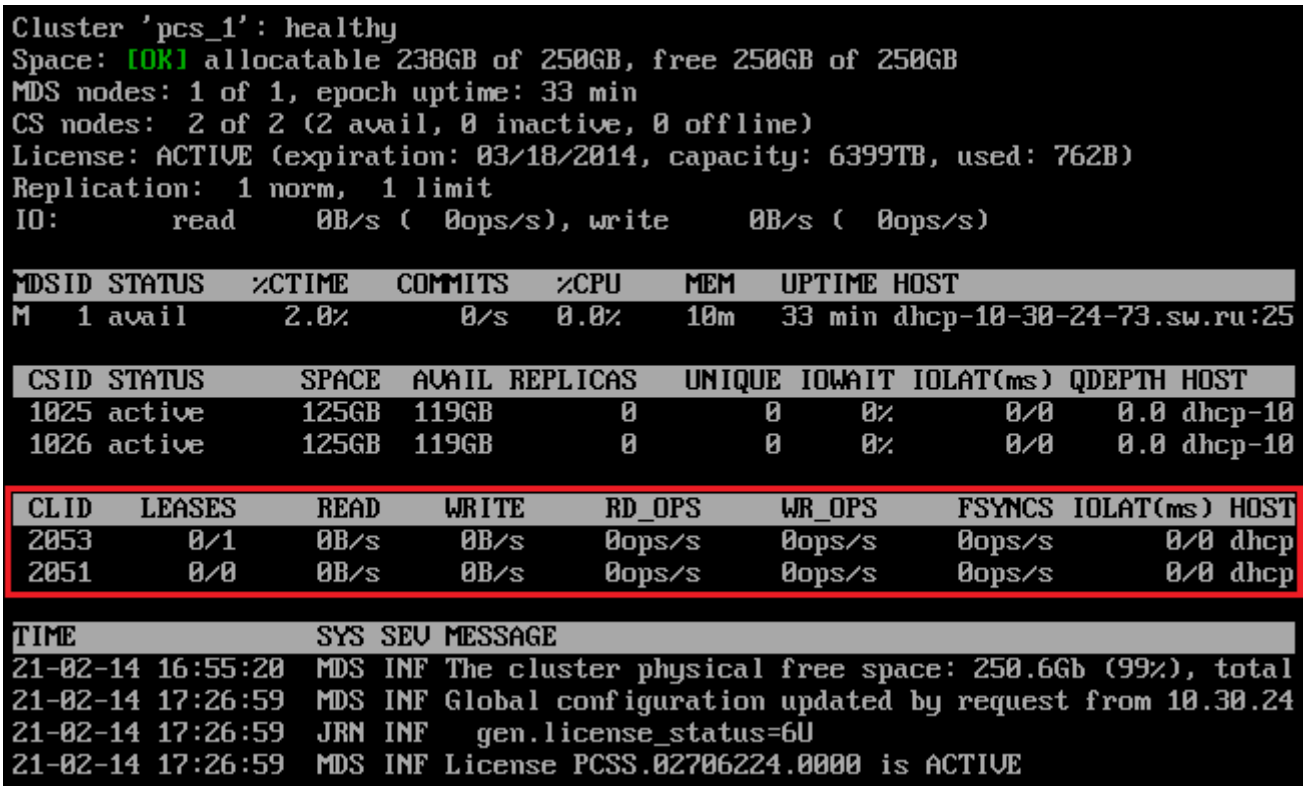
Изучение статусов фрагментов

В таблице ниже представлены все возможные статусы фрагментов.

Статус	Описание
healthy	Процентное отношение фрагментов, которые имеют достаточное число активных реплик.
replicating	Процентное отношение фрагментов, для которых создаются реплики.
offline	Процентное отношение фрагментов, которые не имеют активных реплик.
void	Процентное отношение фрагментов, которые имеют одну или несколько реплик с неясным статусом.
pending	Процентное отношение фрагментов, которые должны быть реплицированы в первую очередь, так как все операции с данными фрагментами приостановлены и клиент ожидает завершения репликации.
blocked	Процентное отношение фрагментов, число реплик у которых равно или меньше минимального числа реплик. Операции записи для данных фрагментов запрещены.
urgent	Процентное отношение фрагментов, число реплик у которых приближается к минимальному числу реплик.
degraded	Процентное отношение фрагментов, которые не имеют достаточного числа активных реплик.
standby	Процентное отношение фрагментов, которые имеют одну или несколько реплик в состоянии ожидания. Реплика имеет статус ожидания, если она неактивна более 5 минут.
overcommitted	Процентное отношение фрагментов, которые превысили число реплик.

Мониторинг клиентов

Путем мониторинга клиентов можно проверить статус и состояние серверов, используемых для входа в виртуальные машины и контейнеры ПК Р-Виртуализация. Для мониторинга клиентов можно использовать команду `vstorage -c <cluster_name> top`, например:



Вывод команды, приведенный выше, показывает подробную информацию о кластере `stor1`. Контролируемые параметры для клиентов (выделенные красным) объясняются в таблице ниже:

Параметр	Описание
CLID	Идентификатор (ID) клиента.
LEASES	Среднее число файлов, открытых клиентом для чтения/записи и еще не закрытых, в течение последних 20 секунд.
READ	Средняя скорость, в байтах в секунду, с которой клиент читает данные, в течение последних 20 секунд.
WRITE	Средняя скорость, в байтах в секунду, с которой клиент пишет данные, в течение последних 20 секунд.
RD_OPS	Среднее число операций чтения в секунду, выполненных клиентом в течение последних 20 секунд.

WR_OPS	Среднее число операций записи в секунду, выполненных клиентом в течение последних 20 секунд.
FSYNCS	Среднее число операций синхронизации в секунду, выполненных клиентом в течение последних 20 секунд.
IOLAT	Среднее/максимальное время, в миллисекундах, которое потребовалось клиенту для завершения одной операции ввода-вывода в течение последних 20 секунд.
HOST	Имя хоста или IP-адрес клиента.

Мониторинг физических дисков

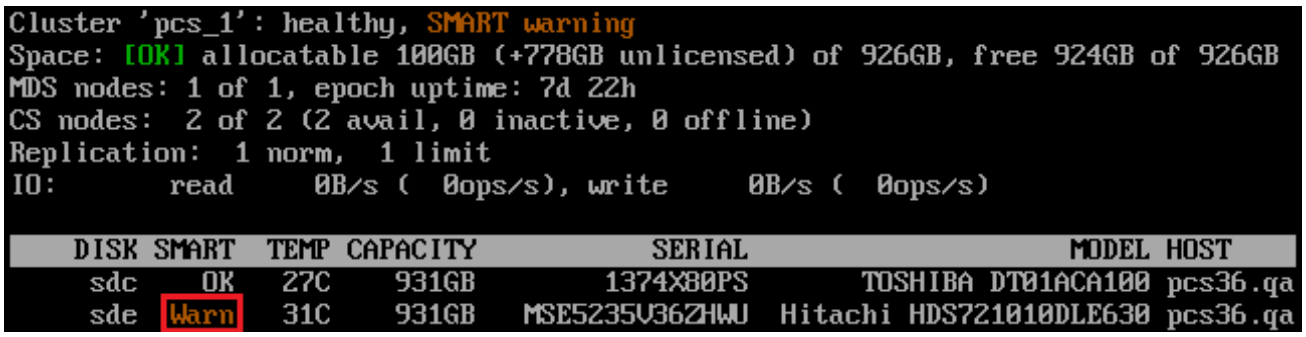
Статус S.M.A.R.T. физических дисков контролируется при помощи инструмента `smartctl`, устанавливаемого вместе с ПК Р-Виртуализация. Инструмент запускается каждые 10 минут в виде задачи демона `cron`, который также добавляется при установке ПК Р-Виртуализация. Инструмент `smartctl` опрашивает все физические диски, подключенные к физическим серверам в кластере, включая кэширующие и журналирующие SSD-диски, и отправляет результаты серверу метаданных.

Примечание: Для работы инструмента необходимо включить функцию S.M.A.R.T. в BIOS сервера.

Можно посмотреть результаты опроса дисков за последние 10 минут в выводе команды `vstorage top`. Например:

```
Cluster 'pcs_1': healthy, SMART warning
Space: [OK] allocatable 100GB (+778GB unlicensed) of 926GB, free 924GB of 926GB
MDS nodes: 1 of 1, epoch uptime: 7d 22h
CS nodes:  2 of 2 (2 avail, 0 inactive, 0 offline)
Replication:  1 norm,  1 limit
IO:       read     0B/s (  0ops/s), write     0B/s (  0ops/s)

MDSID STATUS   %CTIME   COMMITS   %CPU    MEM   UPTIME HOST
M   1 avail      0.0%       0/s   0.0%    48m   7d 22h pcs36.qa.sw.ru:2510

 CSID STATUS      SPACE  AVAIL REPLICAS   UNIQUE IOWAIT IOLAT(ms) QDEPTH HOST
 1025 active      9.1GB  7.1GB        0        0     0%       0/0    0.0 pcs36.q
 1026 active      916GB  870GB        0        0     0%       0/0    0.0 pcs36.q

 CLID   LEASES     READ    WRITE     RD_OPS     WR_OPS     FSYNCS IOLAT(ms) HOST

TIME                SYS SEV MESSAGE
01-07-14 16:42:19  MON WRN CS#1026 was stopped
01-07-14 16:42:26  JRN INF MDS#1 at 10.29.2.16:2510 became master
01-07-14 16:42:26  MDS WRN License not installed, please add license using comma
01-07-14 16:42:29  MON WRN MDS#1 was stopped
01-07-14 16:42:44  MDS INF CS#1025, CS#1026 are active
01-07-14 16:42:53  MDS INF The cluster is healthy with 2 active CS
01-07-14 16:42:53  MDS INF The cluster physical free space: 925.0Gb (99%), total
```

Если в главной таблице отображается **SMART warning**, значит, согласно S.M.A.R.T, один из физических дисков находится в предотказном состоянии. Нажмите клавишу **d**, чтобы переключить на таблицу дисков для просмотра подробной информации. Например:



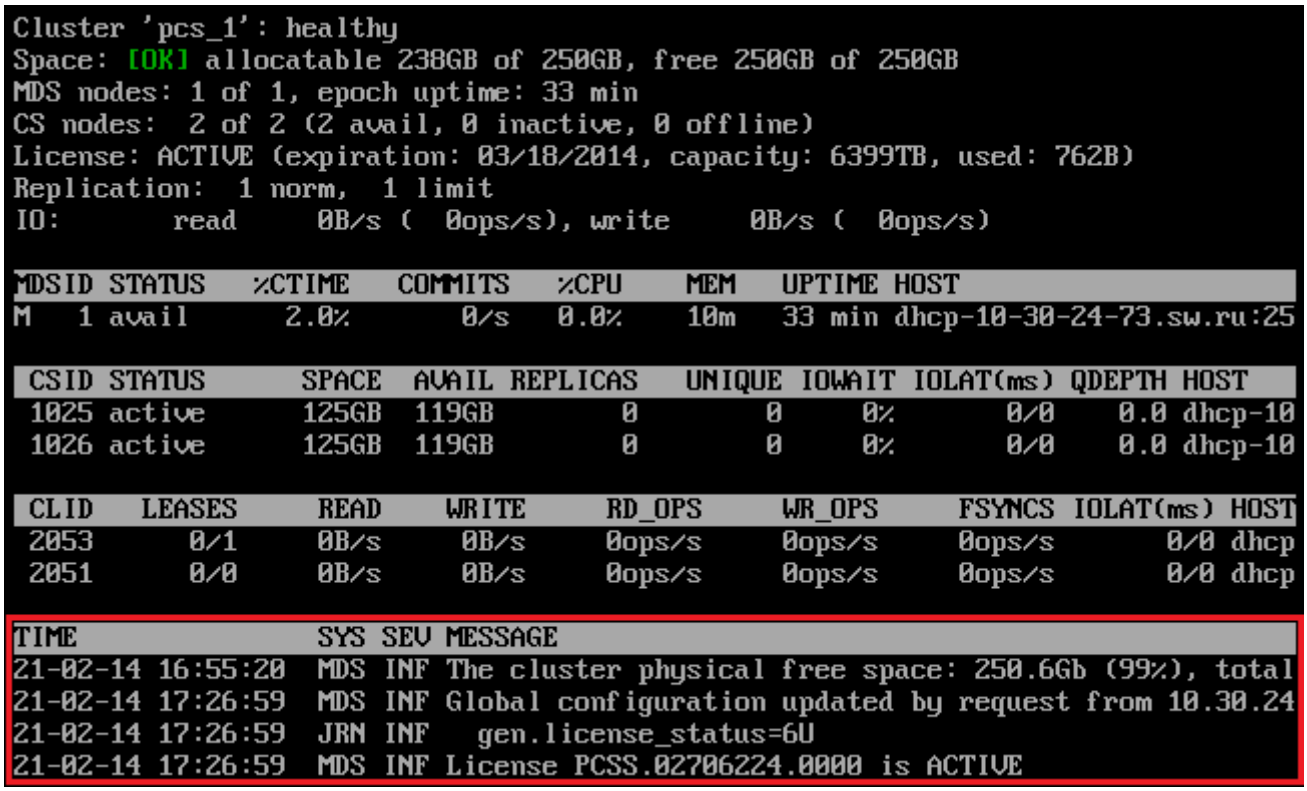
В таблице дисков отображаются следующие параметры:

Параметр	Описание
DISK	Имя диска, назначенное операционной системой.
SMART	Статус S.M.A.R.T. диска: • `OK`: Диск исправен. • `Warn`: Диск находится в предотказном состоянии. Предотказное состояние означает, что по крайней мере одна из следующих переменных S.M.A.R.T. не равна нулю: • `Reallocated Sector Count` • `Reallocated Event Count` • `Current Pending Sector Count` • `Offline Uncorrectable`
TEMP	Температура диска по Цельсию.
CAPACITY	Емкость диска.
SERIAL	Серийный номер диска.
MODEL	Модель диска.
HOST	Адрес хоста диска.

Примечание: Чтобы отключить мониторинг дисков S.M.A.R.T., необходимо удалить соответствующую задачу `cron`.

Мониторинг журналов событий

Для мониторинга важных событий в кластере ПК Р-Хранилище можно использовать утилиту `vstorage -c <cluster_name> top`, например:



Вывод команды, приведенный выше, показывает последние события в кластере `stor1`. Информация о событиях (выделенная красным) отображается в виде таблицы со следующими колонками:

Колонка	Описание
TIME	Время начала события.
SYS	Компонент кластера, на котором произошло событие (например, `MDS` для сервера метаданных или `JRN` для локального журнала).
SEV	Серьезность события.
MESSAGE	Описание события.

Изучение основных событий

В таблице ниже описываются основные события, отображаемые при запуске утилиты `vstorage top`.

Событие	Серьезность	Описание
MDS#<*N*> (<*addr*>:<*port*>) lags behind for more than 1000 rounds	JRN-ошибка	Генерируется master-сервером метаданных при обнаружении устаревшего `MDS#<N>`. Данное сообщение может указывать на то, что у сервера метаданных очень низкая скорость и он отстает в работе.

MDS#<*N*> (<*addr*>:<*port*>) didn't accept commits for *M* sec	JRN-ошибка	Генерируется master-сервером метаданных, если `MDS#<N>` не принимает операции подтверждения в течение `M` секунд. `MDS#<N>` помечается как `stale`. Данное сообщение может указывать на то, что служба сервера метаданных на `MDS#<N>` столкнулась с проблемой. Проблема может быть критической и должна быть устранена, как можно скорее.
MDS#<*N*> (<*addr*>:<*port*>) state is outdated and will do a full resync	JRN-ошибка	Генерируется master-сервером метаданных перед повторной полной синхронизаций `MDS#<N>`. `MDS#<N>` помечается как `stale`. Данное сообщение может указывать на то, что у сервера метаданных была очень низкая скорость или он был отключен в течение такого длительного периода времени, что он уже не управляет состоянием метаданных и должен быть повторно синхронизирован. Проблема может быть критической и должна быть устранена, как можно скорее.
MDS#<*N*> at <*addr*>:<*port*> became master	JRN-сообщение	Генерируется каждый раз при выборе нового master-сервера метаданных в кластере. Частая смена master-серверов метаданных может указывать на ненадежное сетевое соединение и может влиять на работу кластера.
The cluster is healthy with *N* active CS	MDS-сообщение	Генерируется при изменении статуса кластера на `healthy` или при выборе нового master-сервера метаданных. Данное сообщение указывает на то, что все серверы фрагментов в кластере активны и число реплик соответствует заданным параметрам кластера.
The cluster is degraded with *N* active, *M* inactive, *K* offline CS	MDS-предупреждение	Генерируется при изменении статуса кластера на `degraded` или при выборе нового master-сервера метаданных. Данное сообщение указывает на то, что некоторые серверы фрагментов в кластере • неактивны (не оправляют сообщений о регистрации) или • выключены (неактивны в течение периода времени больше, чем `mds.wd.offline_tout = 5min` (по умолчанию)).
The cluster failed with *N* active, M inactive, *K* offline CS (mds.wd.max_offline_cs=<*n*>)	MDS-ошибка	Генерируется при изменении статуса кластера на `failed` или при выборе нового master-сервера метаданных. Данное сообщение указывает на то, что число выключенных серверов фрагментов превышает `mds.wd.max_offline_cs` (2 по умолчанию). При отказе кластера автоматическая репликация больше не планируется. Таким образом, администратор кластера должен принять некоторые меры для восстановления отказавших серверов фрагментов или увеличения значения параметра `mds.wd.max_offline_cs`. Значение данного параметра равное `0` полностью отключает режим отказа.
The cluster is filled up to <*N*>%	MDS-сообщение/ предупреждение	Показывает текущее использование дискового пространства в кластере. Предупреждающее сообщение генерируется, если использование дискового пространства равно или превышает 80%. Рекомендуется иметь свободное дисковое пространство для реплик данных в случае отказа одного из серверов фрагментов.

Replication started, *N* chunks are queued	MDS-сообщение	Генерируется, когда кластер запускает автоматическую репликацию данных для восстановления недостающих реплик.
Replication completed	MDS-сообщение	Генерируется по завершению автоматической репликации данных.
CS#<*N*> has reported hard error on '*path*'	MDS-предупреждение	Генерируется при обнаружении сервером фрагментов CS#<N> повреждение данных на диске. Рекомендуется заменить серверы фрагментов с поврежденными дисками на новые как можно скорее и проверить оборудование на наличие ошибок.
CS#<*N*> has not registered during the last *T* sec and is marked as inactive/offline	MDS-предупреждение	Генерируется, когда сервер фрагментов CS#<N> недоступен в течение некоторого времени. В этом случае сервер фрагментов сначала отмечается как `inactive`. Спустя 5 минут статус меняется на `offline`, тем самым, запуская автоматическую репликацию данных для восстановления реплик, которые хранились на выключенном сервере фрагментов.
Failed to allocate *N* replicas for '*path*' by request from <*addr*>:<*port*> *-K* out of *M* chunks servers are available	MDS-предупреждение	Генерируется, когда кластер не может выделить реплики фрагментов, например, когда у него закончилось дисковое пространство.
Failed to allocate *N* replicas for '*path*' by request from <*addr*>:<*port*> since only *K* chunk servers are registered	MDS-предупреждение	Генерируется, когда кластер не может выделить реплики фрагментов из-за недостаточного числа серверов фрагментов, зарегистрированных в кластере.

Мониторинг статуса параметров репликации

При настройке параметров репликации следует иметь в виду, что новые настройки применяются не сразу. Например, увеличение стандартного параметра репликации для фрагментов данных может занять некоторое время, в зависимости от нового значения параметра и числа фрагментов данных в кластере.

Чтобы проверить, успешно ли применены новые параметры репликации в кластере:

1. Выполните команду `vstorage -c <cluster_name> top`.
2. Нажмите клавишу **v**, чтобы отобразить дополнительную информацию о кластере. Вывод команды должен выглядеть следующим образом:

```
# vstorage -c stor1 top
connected to MDS#1
Cluster 'stor1': healthy
Space: [OK] allocatable 200GB of 211GB, free 211GB of 211GB
MDS nodes: 1 of 1, epoch uptime: 2h 21m
CS nodes:    2 of 2 (2 avail, 0 inactive, 0 offline)
License: STORS.02444715.0000 is ACTIVE, 6399TB capacity
Replication:  3 norm,       2 limit
Chunks: [OK] 431 (100%) healthy,  0 (0%) degraded,    0 (0%) urgent,
    0 (0%) blocked, 0 (0%) offline,     0 (0%) replicating,
    0 (0%) overcommitted,  0 (0%) deleting,    0 (0%) void
...
```

3. Проверьте поле `Chunks`:

- При уменьшении значений параметров репликации проверьте фрагменты со статусами `overcommitted` и `deleting`. Если процесс репликации завершен, в выводе не должно быть фрагментов с данными статусами.
- При увеличении значений параметров репликации проверьте фрагменты со статусами `blocked` и `urgent`. Если процесс репликации завершен, в выводе не должно быть фрагментов с данными статусами. Также если процесс еще не завершен, значение параметра `healthy` будет меньше 100%.

Примечание: Для получения дополнительной информации по статусам фрагментов см. **Изучение статусов фрагментов** (стр. 91).

Глава 6

Управление безопасностью кластера

В данной главе описываются некоторые ситуации, затрагивающие безопасность кластера.

Ограничения по безопасности

В данном разделе описываются ограничения по безопасности, которые следует иметь в виду при развертывании кластера ПК Р-Хранилище.

Анализ трафика

ПК Р-Хранилище не защищает от перехвата и последующего анализа трафика. Любой, у кого есть доступ к сети, может перехватить и обработать данные, отправляемые или получаемые по этой сети.

Для получения информации о том, как обезопасить данные, см. **Обеспечение безопасной передачи данных между серверами в кластере** (стр. 99).

Отсутствие пользователей и групп

ПК Р-Хранилище не использует такие понятия, как пользователи и группы, и не предоставляет определенным пользователям и группам доступ к определенным частям кластера. Любой, у кого есть право доступа к кластеру, имеет доступ ко всем данным в нем.

Незашифрованные данные на дисках

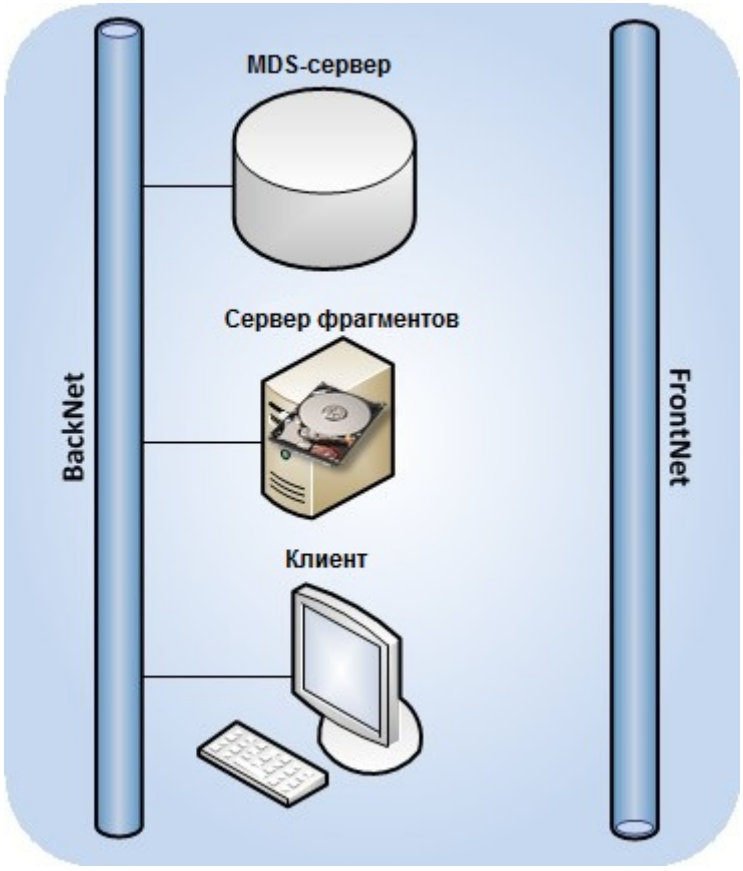
ПК Р-Хранилище не шифрует данные, хранящиеся в кластере. Получив доступ к физическому диску, злоумышленники могут сразу видеть все данные на нем.

Обеспечение безопасной передачи данных между серверами в кластере

Кластер ПК Р-Хранилище включает три типа серверов:

- серверы метаданных;
- серверы фрагментов;
- клиенты.

Во время операции кластера серверы передают друг другу данные. Для обеспечения безопасной передачи данных необходимо, чтобы все серверы находились в изолированной частной сети — BackNet. На рисунке ниже изображен пример конфигурации кластера, в котором все серверы установлены в сети BackNet.



Процесс создания подобной конфигурации может быть описан следующим образом:

1 Установка сервера метаданных и указание одного из его IP-адресов:

```
# vstorage -c Cluster-Name make-mds -I -a MDS-IP-Address -r Journal-Directory -p
```

Указанный адрес будет использоваться позже для подключения сервера метаданных к другим серверам в кластере и передачи данных между ними.

2 Установка сервера фрагментов:

```
# vstorage -c Cluster-Name make-cs -r CS-Directory
```

Созданный сервер фрагментов подключается к серверу метаданных и выполняет привязку к используемому IP-адресу для установления соединения. Если у сервера фрагментов несколько сетевых карт, можно назначить серверу фрагментов IP-адрес определенной сетевой карты, чтобы передача данных между сервером фрагментов и сервером метаданных осуществлялась через данный IP-адрес.

Для того чтобы выполнить привязку сервера фрагментов к выбранному IP-адресу, нужно при создании сервера фрагментов использовать параметр `-a` с командой `vstorage make-cs`:

```
# vstorage make-cs -r CS-Directory -a Custom-IP-Address
```

Примечание: Для обеспечения безопасности кластера выбранный IP-адрес должен принадлежать сети BackNet.

3 Монтирование кластера к клиенту:

```
# vstorage-mount -c Cluster-Name Mount-Directory
```

После монтирования кластера клиент подключается к IP-адресам сервера метаданных и сервера фрагментов.

Данный пример конфигурации кластера обеспечивает высокий уровень безопасности для передачи данных между серверами, так как сервер метаданных, сервер фрагментов и клиент находятся в изолированной сети BackNet.

Методы обнаружения кластера

При выборе метода обнаружения кластера следует обратить внимание на следующее:

- Рекомендуется настраивать автоматическое обнаружение кластера ПК Р-Хранилище с помощью записей DNS. Для получения дополнительной информации см. **Использование записей DNS** (стр. 10).
- Автоматическое обнаружение с помощью Zeroconf можно использовать для тестирования и оценки функций ПК Р-Хранилище. Данный метод обнаружения не рекомендуется использовать в рабочей среде из соображений безопасности. Злоумышленники могут устроить DoS-атаки на службу Zeroconf, даже если для идентификации в сети используются сертификаты безопасности.

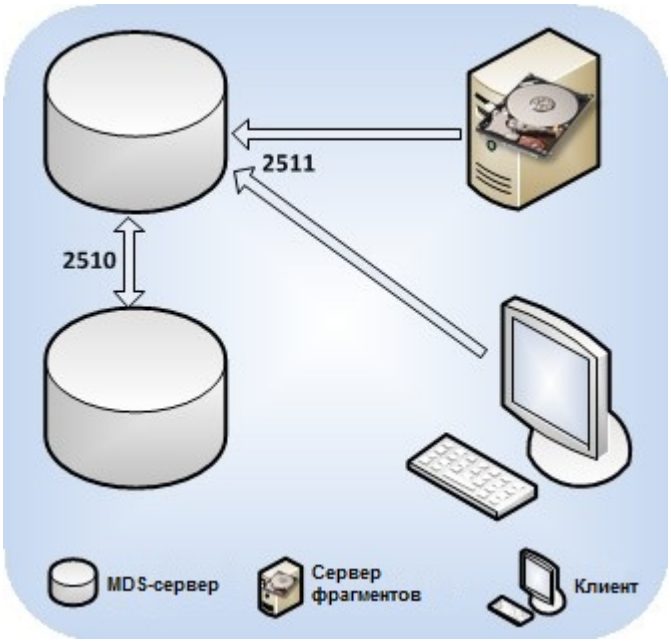
Порты ПК Р-Хранилище

В данном разделе перечислены порты, которые должны быть открыты на серверах в кластере ПК Р-Хранилище в дополнение к портам, используемым ПК Р-Виртуализация и ПК Р-Управление.

Серверы метаданных

На сервере метаданных должны быть открыты следующие порты:

- **Порты прослушивания**: 2510 для входящих соединений от других серверов метаданных и 2511 для входящих соединений от серверов фрагментов и клиентов.
- **Исходящие порты**: 2510 для исходящих соединений к другим серверам метаданных.



По умолчанию ПК Р-Хранилище использует порт 2510 для передачи данных между серверами метаданных и порт 2511 для входящих соединений от серверов фрагментов и клиентов. При создании серверов метаданных можно заменить порты, используемые по умолчанию, следующим образом:

1 Зарезервируйте два неиспользуемых последовательных порта.

Все серверы метаданных, которые будут установлены в кластере, должны иметь одинаковые порты.

2 Выполните команду `vstorage make-mds`, чтобы создать сервер метаданных и указать нижний порт после IP-адреса сервера.

Например, чтобы использовать порты 4110 и 4111 для передачи данных между серверами метаданных в кластере `stor1`, введите следующую команду:

```
# vstorage -c stor1 make-mds -I -a 10.30.100.101:4110 -r /vstorage/stor1-mds -p
```

После указания портов 4110 и 4111 выполните следующие действия:

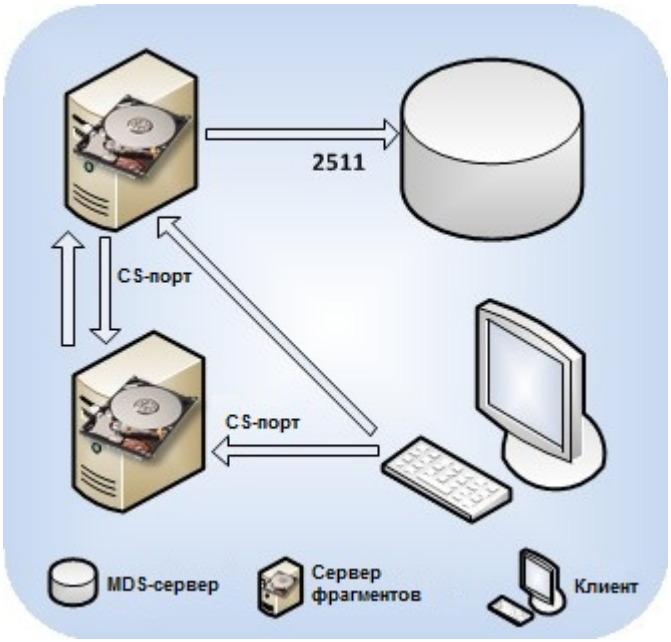
- На каждом сервере метаданных с выбранными портами откройте порты 4110 и 4111 для входящего трафика и порт 4110 для исходящего трафика.
- На всех серверах фрагментов и клиентах в кластере откройте порт 4111 для исходящего трафика.

Серверы фрагментов

На сервере фрагментов должны быть открыты следующие порты:

- **Порты прослушивания**: произвольно выбранный порт для входящих соединений от клиентов и других серверов фрагментов.

- **Исходящие порты**: 2511 для исходящих соединений к серверам метаданных и произвольно выбранный порт для исходящих соединений к другим серверам фрагментов.



Служба управления сервером фрагментов требует:

- Порт для обмена данными с серверами метаданных (либо порт 2511 по умолчанию, либо порт, выбранный самостоятельно).
- Порт для обмена данными с серверами фрагментов и клиентами.

  По умолчанию служба выполняет привязку к любому доступному порту. Указать порт вручную можно с помощью параметра `-a` для команды `vstorage make-cs` при создании сервера фрагментов. Например, чтобы служба управления использовала порт 3000, можно ввести следующую команду:

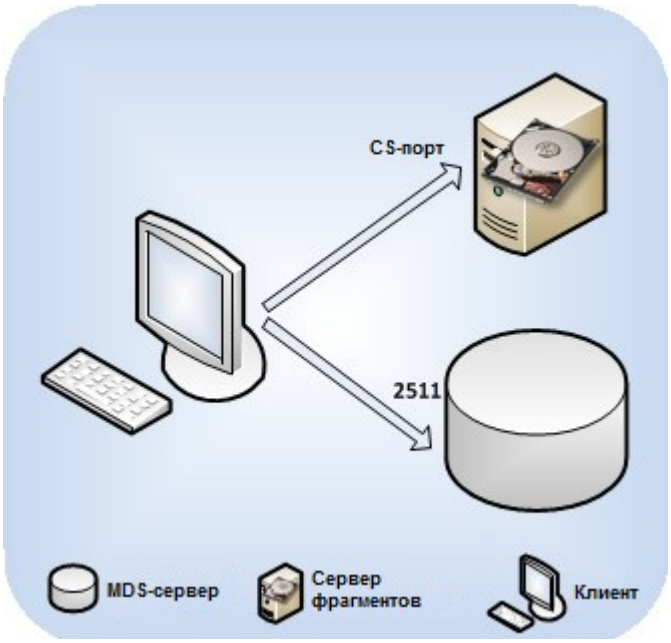
```
# vstorage make-cs -r /vstorage/stor1-cs -a 132.132.1.101:3000
```

  После указания специального порта необходимо открыть его для исходящего трафика на всех клиентах и остальных серверах фрагментов в кластере.

Клиенты

На клиенте должны быть открыты следующие порты:

- **Порты прослушивания**: нет.
- **Исходящие порты**: 2511 для исходящих соединений к серверам метаданных и порт, настроенный в качестве порта прослушивания на серверах фрагментов.



По умолчанию ПК P-Хранилище автоматически открывает на клиенте следующие исходящие порты:

- Для обмена данными с серверами метаданных: порт 2511 по умолчанию.
- Для обмена данными с серверами фрагментов: порт, настроенный в качестве порта прослушивания на серверах фрагментов.

При указании специальных портов для серверов метаданных и серверов фрагментов необходимо открыть эти порты на клиенте для исходящего трафика. Например, при настройке порта 4111 на серверах метаданных и порта 3000 на серверах фрагментов для обмена данными с клиентами нужно открыть исходящие порты 2511 и 3000 на клиенте.

Идентификация по паролю

ПК P-Хранилище использует идентификацию по паролю для повышения безопасности в кластерах. Идентификация по паролю является обязательной, то есть необходимо пройти процедуру идентификации перед добавлением нового сервера к кластеру.

Идентификация по паролю работает следующим образом:

1. Указывается пароль для идентификации при создании первого сервера метаданных в кластере. Указываемый пароль зашифровывается и сохраняется на сервере в файле `/etc/vstorage/clusters/stor1/auth_digest.key`.
2. Новые серверы метаданных, серверы фрагментов или клиенты добавляются к кластеру, и для их идентификации используется команда `vstorage auth-node`. При идентификации используется пароль, указанный при создании первого сервера метаданных.

3 ПК Р-Хранилище сравнивает введенный пароль и пароль, хранимый на первом сервере метаданных, и при их совпадении успешно идентифицирует сервер.

Для каждого физического сервера идентификация является разовой процедурой. Как только сервер идентифицирован в кластере (например, при его настройке в роли сервера метаданных), на нем создается файл `/etc/vstorage/clusters/stor1/auth_digest.key`. При смене роли сервера в кластере (например, на роль сервера фрагментов), кластер обнаруживает файл `auth_digest.key` и не требует повторной идентификации сервера.

Установка через серверы PXE

Kickstart-файлы, используемые для выполнения автоматической установки ПК Р-Виртуализация и ПК Р-Хранилище по сети, содержат пароль для идентификации кластера в виде простого текста. Злоумышленники могут перехватить пароль и получить доступ к кластеру. Для обеспечения безопасности системы следует выполнить одно из следующих действий:

- Физически изолируйте сеть, в которой находится сервер PXE, от других (потенциально ненадежных) сетей.
- Установите ПК Р-Виртуализация через сервер PXE, но не создавайте кластер ПК Р-Хранилище. По окончанию установки вручную разверните кластер в сети, используя механизм идентификации по паролю в ПК Р-Хранилище.

Глава 7

Повышение производительности кластера

В данной главе описывают рекомендуемые конфигурации для кластеров ПК Р-Хранилище и способы их настройки для повышения производительности.

Примечания:

1. Для получения информации о типичных проблемах, которые могут влиять на производительность кластера, также см. **Устранение неисправностей** (стр. 121).

2. Дополнительно можно обновить физические серверы, входящие в кластер, как описано в *Руководстве пользователя по ПК Р-Виртуализация*.

Возможные конфигурации дисковых накопителей

Если серверы, которые будут включены в кластер ПК Р-Хранилище, имеют несколько дисковых накопителей, можно выбрать одну из следующих конфигураций для кластера:

1 **(Рекомендуется с SSD-дисками) Нет локальных RAID-массивов**

Сервер фрагментов установлен на каждом жестком диске, и каждый фрагмент данных имеет две или более реплики. Данная конфигурация обеспечивает независимость от отказов диска, а репликация повышает надежность кластера на уровень RAID1 (зеркалирование без контроля четности и чередования). Для журналирования серверов фрагментов настоятельно рекомендуется использовать твердотельные накопители, что сократит время задержки подтверждения операций записи (например, для баз данных).

2 **(Рекомендуется) Отдельные транзитные диски, подключенные к аппаратному RAID-контроллеру (без уровней RAID 0/1/10/5/6), с наличием или отсутствием SSD-дисков**

Данная рекомендуемая конфигурация похожа на конфигурацию в п. 1, но имеет более высокую скорость. Она включает отдельный диск, подключенный к аппаратному RAID-контроллеру с кэш-памятью и батареей резервного питания (BBU). Кэш отложенной записи RAID-массива существенно повышает скорость выполнения операций записи в произвольном порядке, а также производительность базы данных. Использование твердотельных накопителей в дальнейшем будет оптимизировать операции ввода-вывода в произвольном порядке, особенно операции чтения.

3 **(Не рекомендуется) Локальный RAID0 с чередованием, фрагменты данных с двумя и более репликами, с или без SSD-дисков**

Данная конфигурация не обеспечивает высокую надежность кластера, так как один отказ диска приведет к отказу целого RAID0, и кластеру придется реплицировать больше данных при каждом подобном отказе. Однако данная проблема считается незначительной по той причине, что кластеры ПК Р-Хранилище выполняют параллельное восстановление от нескольких серверов, что занимает меньше времени, чем повторное создание традиционного RAID1.

Использование твердотельных накопителей для кэширования и журналирования в дальнейшем повысит общую производительность кластера и обеспечит контрольное суммирование и скраббинг данных для повышения его надежности.

4 **(Не рекомендуется) Локальный RAID1 с зеркалированием, фрагменты данных с двумя и более репликами**

Если для каждого фрагмента данных создается только одна реплика, то такая конфигурация не обеспечивает высокую доступность кластера в случае отказа одного из серверов. Кроме того, подобная конфигурация не экономит дисковое пространство, так как она является аналогом зеркалирования кластера, локальный RAID-массив просто дублирует число реплик данных и сохраняет сетевой трафик кластера.

5 **(Настоятельно не рекомендуется) Локальный RAID1, 5 или 6, фрагменты данных с двумя и более репликами**

Не следует запускать ПК Р-Хранилище в RAID-массивах избыточного типа, таких как 1, 5 или 6, через локальный накопитель. В этом случае одна операция записи может влиять на значительное число жестких дисков, что приведет к очень низкой производительности. Например, для 3 реплик ПК Р-Хранилище и RAID5 на серверах, которые имеют по 5 жестких дисков, одна операция записи может привести к 15 операциям ввода-вывода.

Проведение проверки для оценки производительности

При проверке производительности кластера ПК Р-Хранилище и ее сравнении с другими установками:

- Следует сравнивать конфигурации с похожим уровнем избыточности. Например, неверно сравнивать производительность кластера с двумя и тремя репликами на каждый фрагмент данных с автономным сервером, который не использует избыточность данных, как RAID1, 10 или 5.
- Следует учитывать использование интерфейсов файловой системы, например, что монтирование кластера ПК Р-Хранилище с использованием интерфейса FUSE обеспечивает удобное отображение кластера, но снижает его производительность. Таким образом, следует проводить проверку из контейнеров и виртуальных машин.

- Следует иметь в виду, что число реплик фрагментов влияет на производительность кластера: кластеры с двумя репликами имеют более высокую скорость, чем кластеры с тремя репликами.

Использование гигабитной и 10-гигабитной сети Ethernet

Гигабитные сети Ethernet могут передавать данные со скоростью 110-120 Мб/с, что почти соответствует производительности одного диска для последовательного ввода-вывода. Так как несколько дисков на одном сервере могут иметь более высокую пропускную способность, чем один гигабитный канал Ethernet, использование сети может стать узким местом.

Однако в реальных приложениях и виртуальных средах последовательный ввод-вывод используется нечасто (в основном, для резервного копирования), и большинство операций ввода-вывода выполняются в произвольном порядке. Таким образом, обычная пропускная способность жесткого диска значительно ниже, около 10-20 Мб/с, согласно статистике, собранной рядом крупных провайдеров хостинга от сотен серверов.

Исходя из данных двух наблюдений, рекомендуется использовать одну из следующих сетевых конфигураций (или лучше):

- Один гигабитный канал на каждые два жестких диска на физическом сервере. Однако если сервер имеет 1 или 2 жестких диска, для высокой надежности рекомендуется использовать два агрегированных сетевых адаптера (см. **Агрегирование каналов сети** (стр. 109)).
- 10-гигабитный канал на каждый физический сервер для повышения производительности.

В таблице ниже показано, как вышеприведенные рекомендации могут применяться к физическому серверу с 1-6 жесткими дисками:

Жесткие диски	Гигабитные каналы	10-гигабитные каналы
1	1 (2 для высокой доступности)	1 (2 для высокой доступности)
2	1 (2 для высокой доступности)	1 (2 для высокой доступности)
3	2	1 (2 для высокой доступности)
4	2	1 (2 для высокой доступности)
5	3	1 (2 для высокой доступности)
6	3	1 (2 для высокой доступности)

Примечания:

1. Для повышения производительности последовательного ввода-вывода рекомендуется использовать один гигабитный канал на каждый жесткий диск или один 10-гигабитный канал на каждый физический сервер.

2. Не рекомендуется настраивать гигабитные сетевые адаптеры для использования нестандартных MTU (например, 9000- байтных jumbo-кадров). Подобные настройки требуют смены конфигурации и часто приводят к ошибкам администратора. 10-гигабитные сетевые адаптеры, с другой стороны, необходимо настроить для использования jumbo-кадров, чтобы значительно повысить производительность.

3. Для максимальной эффективности рекомендуется использовать режим агрегирования `balance-xor` с параметром `xmit_hash_policy=layer3+4`. Если вы хотите использовать режим агрегирования `802.3ad`, также настройте параметр контроллера `xmit_hash_policy=layer3+4`.

Агрегирование сетевых адаптеров

Агрегирование нескольких сетевых интерфейсов дает следующие преимущества:

1. Высокую доступность сети. При отказе одного интерфейса трафик будет автоматически направлен к работающим интерфейсам.
2. Высокую производительность сети. Например, два объединенных гигабитных интерфейса будут передавать данные со скоростью 1.7 Гбит/с или 200 Мб/с. Необходимое количество объединенных сетевых интерфейсов хранения данных может зависеть от количества накопителей на физическом сервере. Например, пропускная способность жесткого диска с вращательным движением может достигать до 1 Гбит/с.

Для того чтобы настроить агрегирование интерфейса, выполните следующие действия:

1. Создайте файл `/etc/modprobe.d/bonding.conf`, содержащий строку:

```
alias bond0 bonding
```

2. Создайте файл `/etc/sysconfig/network-scripts/ifcfg-bond0`, содержащий строки:

```
DEVICE=bond0
ONBOOT=yes
BOOTPROTO=none
IPV6INIT=no
USERCTL=no
BONDING_OPTS="mode=balance-xor xmit_hash_policy=layer3+4 miimon=300 downdelay=300
updelay=300"
NAME="Storage net0"
NM_CONTROLLED=yes
IPADDR=xxx.xxx.xxx.xxx
PREFIX=24
```

Примечания:

1. Следует проверить, чтобы в строках `IPADDR` и `PREFIX` были введены правильные значения.

2. Рекомендуется использовать режим `balance-xor`, так как он обеспечивает отказоустойчивость и повышает производительность.

3 Убедитесь, что файл конфигурации каждого интерфейса Ethernet, который будет объединен (например, `/etc/sysconfig/network-scripts/ifcfg-eth0`), содержит следующие строки:

```
DEVICE="eth0"
BOOTPROTO=none
NM_CONTROLLED="yes"
ONBOOT="yes"
TYPE="Ethernet"
HWADDR=xx:xx:xx:xx:xx:xx
MASTER=bond0
SLAVE=yes
USERCTL=no
```

4 Поднимите интерфейс `bond0`:

```
# ifup bond0
```

5 Используйте вывод `dmesg` для проверки, что `bond0` и его slave-интерфейсы Ethernet работают и каналы готовы.

Примечание: Дополнительную информацию об агрегировании каналов сети см. в *Red Hat Enterprise Linux Deployment Guide* and *Linux Ethernet Bonding Driver HOWTO*.

Использование SSD-дисков

ПК Р-Хранилище поддерживает SSD-диски, отформатированные в файловую систему ext4 и подмонтированные с включенной поддержкой TRIM.

Примечание: Сценарий использования SSD-дисков ПК Р-Хранилище не генерирует команды TRIM. Также современные диски, например, Intel SSD DC S3700, не требуют TRIM.

ПК Р-Хранилище поддерживает использование SSD-дисков не только для хранения фрагментов данных, но и для таких целей, как:

- **Журналирование операций записи** (стр. 112). Можно подключить SSD-диск к серверу фрагментов и настроить его для хранения журнала операций записи, тем самым повышая производительность операций записи в кластере в 2 и более раз.
- **Кэширование данных** (р. 115). Можно подключить SSD-диск к клиенту и настроить его для хранения локального кэша часто используемых данных, тем самым повышая производительность целого кластера в 10 и более раз.

Место на твердотельных накопителях должно быть разделено между журналами и кэшем считывания в соответствии с загрузкой, т.е. число операций записи относительно числу операций чтения. Размер кэша считывания также зависит от того, насколько высоки требования приложений. При возникновении сомнений рекомендуется разделить место на SSD-диске поровну. Например, при наличии SSD-диска емкостью 100 ГБ и четырех серверов фрагментов на четырех жестких дисках емкостью 1ТБ, следует разделить пространство SSD-диска следующим образом:

- 20 ГБ резервируется для контрольных сумм и крайних нужд, а также для того, чтобы препятствовать полному заполнению SSD-диска (что приведет к понижению его производительности);
- 40 ГБ для кэша считывания,
- 40 ГБ для журналов, т.е. 10 ГБ на каждый жесткий диск/сервер фрагментов.

Для контрольных сумм необходимо 4 байта для каждой страницы размером 4 КБ (отношение 1:1000). Например, для хранения 4 ТБ понадобится 4 ГБ дискового пространства для контрольных сумм.

Чтобы понять, сколько дискового пространства нужно выделить для журналов и кэша в конфигурации отдельного кластера, запустите команду `vstorage advise-configuration`. Команда использует параметры кластера в качестве вводных данных и в выводе предлагает варианты по оптимизации производительности кластера, установке хоста и монтированию кластера через `/etc/fstab` (примеры см. в разделах ниже).

В целом, оптимизация для журналов означает, что около 70% места на SSD-диске должно использоваться для журналов и около 30% - для кэша; оптимизация для кэша предполагает обратное соотношение. Если оптимизация для журналов или кэша не является целью, то можно использовать равные части дискового пространства на SSD-диске для журналов и кэша. В каждом из этих случаев некоторое место должно быть зарезервировано для контрольных сумм и т.п. (команда `vstorage advise-configuration` предлагает размер места на диске, который нужно зарезервировать).

Таблица ниже может помочь рассчитать количество SSD-дисков, необходимых для кластера.

Тип SSD-диска	Количество SSD-дисков
Intel SSD 320 Series, Intel SSD 710 Series, Kingston SSDNow E Enterprise Series или другие SSD-модели SATA с 3 Гбит/с, которые обеспечивают скорость последовательной записи произвольных данных 150-200 Мб/с.	1 SSD-диск на каждые 3 HDD-диска
Intel SSD DC S3700 Series, Samsung SM1625 Enterprise Series или другие SSD-модели SATA с 3 Гбит/с, которые обеспечивают скорость последовательной записи произвольных данных как минимум 300 Мб/с.	1 SSD-диск на каждые 5-6 HDD-дисков

В разделах ниже представлена подробная информация по настройке SSD-дисков для журналирования и кэширования данных.

Примечания:

1. Не все твердотельные накопители придерживаются семантики сброса данных на диск и выполняют подтверждение данных в соответствии с протоколом, что может привести к произвольной потере или повреждению данных в случае отключения питания. Следует всегда проверять SSD-диски с помощью инструмента `vstorage-hwflush-check` (для получения дополнительной информации см. **Проверка сброса данных на диск** (стр. 13)).

2. Рекомендуется использовать диски Intel SSD DC S3700. Также можно использовать Samsung SM1625, Intel SSD 710, Kingston SSDNow E или любой другой SSD-диск с поддержкой защиты данных от потери питания. Некоторые названия подобных технологий: Enhanced Power Loss Data Protection (Intel), Cache Power Protection (Samsung), Power-Failure Support (Kingston),

Complete Power Fail Protection (OCZ). Для получения дополнительной информации см. **SSD-диски игнорируют сброс данных на диск** (стр. 124).

3. Минимальный рекомендуемый размер SSD-диска составляет 30 ГБ.

Настройка SSD-дисков для журналирования

Использование SSD-дисков для журналирования операций записи может помочь сократить время задержки операций записи, таким образом, повышая общую производительность кластера.

Определить, сколько места на SSD-диске потребуется для журналов, можно с помощью команды `vstorage advise-configuration`. Например:

```
# vstorage -c stor1 advise-configuration -w --cs /vstorage/stor1-cs1 --cs \
/vstorage/stor1-cs2 --cs /vstorage/stor1-cs3 --cs /vstorage/stor1-cs4 --ssd \
/vstorage/stor1-ssd -m /vstorage/stor1
You have the following setup:
CS on /vstorage/stor1-cs4 -- Total disk space 1007.3GB
CS on /vstorage/stor1-cs3 -- Total disk space 1007.3GB
CS on /vstorage/stor1-cs2 -- Total disk space 1007.3GB
CS on /vstorage/stor1-cs1 -- Total disk space 1007.3GB
SSD on /vstorage/stor1-ssd -- Total disk space 251.8GB
Proposed server configuration optimized for writes:
 - 155.9GB (61%) for CS journals, 66.8GB (26%) for mount read cache on /vstorage/stor1-
ssd,29.1GB (11%) reserved (including 3.9GB checksums for 3.9TB of data)
 - CS journal sizes:
38.9GB for /vstorage/stor1-cs4 at /vstorage/stor1-ssd
38.9GB for /vstorage/stor1-cs3 at /vstorage/stor1-ssd
38.9GB for /vstorage/stor1-cs2 at /vstorage/stor1-ssd
38.9GB for /vstorage/stor1-cs1 at /vstorage/stor1-ssd
How to setup the node:
vstorage -c stor1 make-cs -r /vstorage/stor1-cs4/cs -j /vstorage/stor1-ssd/cs4-stor1-
journal -s 39914
vstorage -c stor1 make-cs -r /vstorage/stor1-cs3/cs -j /vstorage/stor1-ssd/cs3-stor1-
journal -s 39914
vstorage -c stor1 make-cs -r /vstorage/stor1-cs2/cs -j /vstorage/stor1-ssd/cs2-stor1-
journal -s 39914
vstorage -c stor1 make-cs -r /vstorage/stor1-cs1/cs -j /vstorage/stor1-ssd/cs1-stor1-
journal -s 39914
vstorage-mount -c stor1 /vstorage/stor1 -C /vstorage/stor1-ssd/vstorage-stor1-cache -R
68424
Mount option for automatic cluster mount from /etc/fstab:
vstorage://stor1 /vstorage/stor1 fuse.vstorage cache=/vstorage/stor1-ssd/vstorage-
stor1-cache,cachesize=68424 0 0
```

В данном примере предполагается выделить 61% места на SSD-диске для журналов серверов фрагментов, чтобы получить оптимальную производительность кластера.

Примечания:

1. Если на одном хосте находится несколько серверов фрагментов, следует создать отдельный журнал на SSD-диске для каждого сервера фрагментов, убедившись, что на SSD-диске

достаточно места для всех журналов. Для изменения размера существующих журналов серверов фрагментов можно использовать команду `vstorage configure-cs`.

2. При определении размера журнала без помощи команды `vstorage advise-configuration` следует убедиться, что на SSD-диске есть 1 ГБ на каждый жесткий диск емкостью 1 ТБ для вычисления контрольных сумм.

Установка сервера фрагментов с журналом на SSD-диске

Для того чтобы установить сервер фрагментов с журналом на SSD-диске, выполните следующие действия:

1. Зайдите на сервер, который будет настроен как сервер фрагментов, в роли пользователя root или пользователя с привилегиями root. Сервер должен иметь, по крайней мере, один жесткий диск (HDD) и один твердотельный накопитель (SSD).
2. Загрузите и установите следующие пакеты RPM: `vstorage-ctl`, `vstorage-libs-shared` и `vstorage-chunk-server`.

   Пакеты доступны в удаленном хранилище ПК Р-Виртуализация (данное хранилище автоматически настраивается для вашей системы при установке ПК Р-Виртуализация), и их можно установить с помощью следующей команды:

```
# yum install vstorage-chunk-server
```

3. Убедитесь, что в сети настроено обнаружение кластера. Для получения подробной информации см. **Настройка обнаружения кластера** (стр. 9).
4. Выполните идентификацию сервера в кластере. Данная процедура необходима, только если данный сервер никогда раньше не был идентифицирован в кластере:

```
# vstorage -c stor1 auth-node
```

5. При необходимости подготовьте SSD-диск, как описано в разделе **Подготовка дисков для ПК Р-Хранилище** (стр. 15).
6. Создайте конфигурацию, хранилище и журнал сервера фрагментов, например:

```
# vstorage -c stor1 make-cs -r /vstorage/stor1-cs -j /ssd/stor1/cs1 -s 30720
```

   Данная команда

   - Создает директорию `/vstorage/stor1-cs` на жестком диске компьютера и настраивает ее для хранения фрагментов данных.
   - Настраивает компьютер как сервер фрагментов и добавляет его к кластеру `stor1`.
   - Создает журнал в директории `/ssd/stor1/cs1` на SSD-диске и выделяет 30 ГБ дискового пространства данному журналу.

     Примечание: При выборе директории для журнала и определении ее размера следует выделить необходимое дисковое пространство журналу и убедиться, что на SSD-диске есть 1 ГБ на каждый жесткий диск емкостью 1 ТБ для вычисления контрольных сумм.

7. Запустите службу управления сервером фрагментов (`vstorage-csd`) и настройте ее таким образом, чтобы она автоматически запускалась при загрузке сервера фрагментов:

```
# systemctl start vstorage-csd.target
```

Добавление, удаление и настройка журналов сервера фрагментов в кластерах ПК Р-Хранилище

Примечание: Узнать путь к хранилищу сервера фрагментов можно с помощью команды `vstorage list-services -C`.

Добавление журналов к серверу фрагментов

Чтобы добавить новый журнал к серверу фрагментов, используйте команду `vstorage configure-cs -a -s`. Например, чтобы добавить журнал размером 2048 МБ к серверу фрагментов CS#1 и поместить его в директорию на SSD-диске, монтированному к `/ssd`, введите:

```
# vstorage -c stor1 configure-cs -r /vstorage/stor1-cs1/data -a /ssd/stor1-cs1-journal
–s 2048
```

Удаление журналов с сервера фрагментов

Чтобы удалить журнал с сервера фрагментов, используйте команду `vstorage configure-cs -d`. Например:

```
# vstorage -c stor1 configure-cs -r /vstorage/stor1-cs1/data -d
```

Перемещение журналов сервера фрагментов

Чтобы изменить директорию журнала сервера фрагментов, выполните следующие действия, используя команды, приведенные выше:

1. Удалите существующий журнал.
2. Добавьте новый журнал нужного размера в соответствующую директорию.

Изменение размера журналов сервера фрагментов

Изменить размер журнала сервера фрагментов можно с помощью команды `vstorage configure-cs -s`. Например, для изменения размера журнала сервера фрагментов на 4096 МБ введите:

```
# vstorage -c stor_1 configure-cs -r /vstorage/stor_1-cs1/data -s 4096
```

Отключение контрольного суммирования

Использование контрольного суммирования обеспечивает высокую надежность и целостность всех данных в кластере. Если контрольное суммирование включено, ПК Р-Хранилище генерирует контрольные суммы при каждом изменении данных в кластере. При последующем чтении этих данных контрольная сумма вычисляется еще раз и сравнивается со значением контрольной суммы, сгенерированной ранее.

По умолчанию контрольное суммирование данных автоматически включено для всех новых серверов фрагментов. При необходимости данную функцию можно отключить при создании сервера фрагментов с помощью параметра `-S`, например:

```
# vstorage -c stor1 make-cs -r /vstorage/stor1-cs -j /ssd/stor1/cs1 -s 30720 -S
```

Настройка скраббинга данных

Скраббинг данных (data scrubbing) представляет собой процесс проверки фрагментов данных на долговечность, а их содержимого на читаемость и правильность. По умолчанию ПК Р-Хранилище проверяет два фрагмента данных в минуту на каждом сервере фрагментов в кластере. При необходимости можно изменить это значение, используя утилиту `vstorage`, например:

```
# vstorage -c stor1 set-config mds.wd.verify_chunks=3
```

Данная команда задает количество фрагментов для проверки на каждом сервере фрагментов в кластере `stor1` равное 3.

Настройка SSD-дисков для кэширования данных

Другим способом повышения общей производительности кластера является создание локального кэша на SSD-диске клиента. После создания кэша все данные кластера, к которым происходило обращение два или более раз, будут помещены в данный кэш.

В таблице ниже перечислены основные характеристики локального кэша:

Характеристика	Описание
Малое время обращения	К данным в локальном кэше обращение происходит во много раз быстрее (до 10 и более раз) по сравнению с обращением к тем же данным, хранящимся на серверах фрагментов в кластере.
Нет потребления пропускной способности сети	Пропускная способность сети кластера не потребляется, так как обращение к данным происходит локально.
Специальный кэш загрузки	Локальный кэш использует специальный кэш загрузки для хранения данных небольшого объема при открытии файлов, тем самым значительно ускоряя процесс запуска виртуальных машин и контейнеров.
Сохранность кэша	Локальный кэш является постоянным и сохраняется при корректном выключении системы, но удаляется в случае отказа системы.
Последовательная фильтрация обращений	Кэшируются только данные, к которым происходило обращение в произвольном порядке. Приложения резервного копирования данных могут создавать огромное число последовательных операций ввода-вывода. Препятствие записи таких операций в кэш помогает избежать чрезмерной загрузки кэша.

Чтобы определить, сколько места на SSD-диске нужно выделить для кэша, используйте команду `vstorage advise-configuration` с параметром `-r`. Например:

```
# vstorage -c stor1 advise-configuration -r --cs /vstorage/stor1-cs1 --cs \
/vstorage/stor1-cs2 --cs /vstorage/stor1-cs3 --cs /vstorage/stor1-cs4 --ssd \
/vstorage/stor1-ssd -m /vstorage/stor1
You have the following setup:
CS on /vstorage/stor1-cs1 -- Total disk space 1007.3GB
CS on /vstorage/stor1-cs2 -- Total disk space 1007.3GB
CS on /vstorage/stor1-cs3 -- Total disk space 1007.3GB
CS on /vstorage/stor1-cs4 -- Total disk space 1007.3GB
SSD on /vstorage/stor1-ssd -- Total disk space 251.8GB
Proposed server configuration optimized for reads:
```

```
 - 66.8GB (26%) for CS journals, 155.9GB (61%) for mount read cache on /vstorage/stor1-
ssd,29.1GB (11%) reserved (including 3.9GB checksums for 3.9TB of data)
 - CS journal sizes:
16.7GB for /vstorage/stor1-cs4 at /vstorage/stor1-ssd
16.7GB for /vstorage/stor1-cs3 at /vstorage/stor1-ssd
16.7GB for /vstorage/stor1-cs2 at /vstorage/stor1-ssd
16.7GB for /vstorage/stor1-cs1 at /vstorage/stor1-ssd
How to setup the node:
vstorage -c stor1 make-cs -r /vstorage/stor1-cs4/cs -j /vstorage/stor1-ssd/cs4-stor1-
journal -s 17106
vstorage -c stor1 make-cs -r /vstorage/stor1-cs3/cs -j /vstorage/stor1-ssd/cs3-stor1-
journal -s 17106
vstorage -c stor1 make-cs -r /vstorage/stor1-cs2/cs -j /vstorage/stor1-ssd/cs2-stor1-
journal -s 17106
vstorage -c stor1 make-cs -r /vstorage/stor1-cs1/cs -j /vstorage/stor1-ssd/cs1-stor1-
journal -s 17106
vstorage-mount -c stor1 /vstorage/stor1 -C /vstorage/stor1-ssd/vstorage-stor1-cache -R
159658
Mount option for automatic cluster mount from /etc/fstab:
vstorage://stor1 /vstorage/stor1 fuse.vstorage cache=/vstorage/stor1-ssd/vstorage-
stor1-cache,cachesize=159658 0 0
```

В данном примере предполагается выделить 61% места на SSD-диске для кэша, чтобы получить оптимальную производительность кластера.

Создание локального кэша

Примечание: В отличие от директорий, используемых на большинстве этапов конфигурации ПК Р-Хранилище, локальный кэш на SSD-диске представляет собой файл. Следует убедиться в правильности путей, указанных для команды `vstorage-mount –C` и параметра `cache` в соответствующей записи `/etc/fstab`.

Создание кэша происходит при монтировании кластера ПК Р-Хранилище к клиенту. Данный процесс состоит из двух шагов:

1. Подготовка SSD-диска при необходимости, как описано в разделе **Подготовка дисков для ПК Р-Хранилище** (стр. 15).
2. Использование команды `vstorage-mount` для монтирования кластера и создания кэша.

Например, чтобы создать локальный кэш размером 64 ГБ для кластера `stor1` и сохранить его в файл `/mnt/ssd/vstorage-cache-for-cluster-stor1`, нужно ввести команду:

```
# vstorage-mount -c stor1 /vstorage/stor1 -C /mnt/ssd/vstorage-cache-for-cluster-stor1
-R 64000
```

Если размер кэша не указывается, `vstorage-mount` автоматически вычисляет его по следующей формуле:

```
SSD_free_space - 10 GB - SSD_total_space/10
```

Таким образом, если общая емкость SSD-диска 100 ГБ и свободное дисковое пространство составляет 80 ГБ, команда создаст локальный кэш размером 60 ГБ.

Примечания:

1. Локальный кэш не создается, если итоговый размер кэша меньше 1 ГБ.

2. Если SSD-диск будет также настроен для журналирования, то сначала следует создать журнал, чтобы зарезервировать для него дисковое пространство, а затем уже создавать локальный кэш. Для получения дополнительной информации см. **Настройка SSD-дисков для журналирования** (стр. 112).

Настройка автоматического создания кэша

Можно автоматизировать процесс создания локального кэша, чтобы он автоматически создавался при каждой загрузке клиента. Для автоматизации создания кэша нужно добавить информацию о кэше в файл `/etc/fstab` на клиенте.

Например, чтобы (1) кэш автоматически создавался с именем `vstorage-cache-for-cluster-stor1` и размером 64 ГБ, (2) он хранился в директории `/mnt/ssd` на клиенте и (3) отключить контрольное суммирование для данных в локальном кэше, необходимо указать следующие параметры в `/etc/fstab` через запятую:

- `cache=<path>`. Задает полный путь к файлу локального кэша.
- `cachesize=<size>`. Указывает размер локального кэша, в мегабайтах.
- `cachechksum=n`. Отключает контрольное суммирование для данных; по умолчанию контрольное суммирование включено.

После указания вышеперечисленных параметров файл `fstab` должен выглядеть следующим образом:

```
vstorage://stor1 /vstorage/stor1 fuse.vstorage cache=/mnt/ssd/vstorage-cache-for-
cluster- stor1,cachesize=64000,cachechksum=n 0 0
```

Для получения дополнительной информации о параметрах, которые можно использовать для создания и настройки локального кэша, см. man-страницу по `vstorage-mount`.

Отключение контрольного суммирования

Для обеспечения высокой надежности и целостности данных команда `vstorage-mount` автоматически включает контрольное суммирование для данных в локальном кэше. При необходимости можно отключить контрольное суммирование данных с помощью параметра `-S` для команды `vstorage-mount`:

```
# vstorage-mount -c stor1 /vstorage/stor1 -C /mnt/ssd/vstorage-cache-for-cluster-stor1
-R 64000 -S
```

Получение информации о кэше

Проверить состояние кэша для монтированного кластера и посмотреть его текущие параметры можно при помощи следующей команды:

```
# cat /vstorage/stor1/.vstorage.info/read_cache_info
path               : /mnt/ssd/vstorage-cache-for-cluster-stor1
main   size (Mb)   : 56000
```

```
boot   size (Mb)     : 8000
block  size (Kb)     : 64 checksum : enabled
```

Если кэш не существует, вывод команды пуст. В противном случае, команда выводит:

- путь к файлу кэша;
- размер основного кэша и кэша загрузки;
- размер блока;
- статус контрольной суммы.

Повышение производительности жесткого диска большой емкости

В отличие от более старых дисков с 512-байтными секторами, многие современные жесткие диски (емкостью от 3 ТБ) используют физические секторы размером 4 КБ, что в определенных случаях может значительно снизить производительность системы (в 3-4 раза) из-за лишних циклов «чтение-изменение-запись» (RMW), необходимых для выравнивания исходного запроса на запись. Когда операционная система посылает невыровненный запрос на запись, жесткий диск должен выровнять начало и конец этого запроса по границам 4-килобайтных секторов. Для этой процедуры жесткий диск считывает начальные и конечные диапазоны запроса, чтобы определить четное число секторов для изменения. Например, для запроса на запись блока размером 4 КБ со смещением в 2 КБ жесткий диск прочитает диапазоны 0-2 КБ и 6-8 КБ, чтобы изменить целый диапазон данных 0-8 КБ.

Типичными причинами низкой производительности жестких дисков с 4-килобатными секторами являются:

1. Файловая система гостевой ОС не выровнена по границе 4-килобатных секторов. Команда ПК Р-Хранилище `make-cs` пытается заранее определить подобные проблемы и отправить отчет о них администратору. Не рекомендуется использовать утилиту `fdisk` для организации разделов жестких дисков. Вместо нее следует использовать `parted`.
2. Записи (например, размером 1 КБ), выполненные гостевой ОС, не выровнены. Многие устаревшие операционные системы, такие как Microsoft Windows XP и Windows Server 2003 или Red Hat Enterprise Linux 5.x, по умолчанию имеют невыровненные разделы и создают невыровненные шаблоны ввода-вывода, у которых очень низкая скорость как в ПК Р-Хранилище, так и на жестких дисках с 4-килобатными секторами. При запуске подобных устаревших ОС следует иметь в виду:
   - Использование жестких дисков малой емкости с 512-байтными секторами или SSD-журналирования для служб сервера фрагментов, что поможет несколько повысить производительность.
   - Правильное выравнивание разделов ОС.

Для проверки наличия невыровненных операций записи в кластере, выполните следующие действия:

1 Введите команду `vstorage top` или `stat`. Например:

```
# vstorage -c stor1 top
```

2 Нажмите **i** для отображения колонок `RMW` и `JRMW` в части сервера фрагментов вверху вывода.

3 Проверьте переменные `RMW` или `JRMW`, которые объясняются ниже.

- При использовании SSD-журналирования переменная `RMW` показывает число запросов, которые приводят к циклам «чтение-изменение-запись», а переменная `JRMW` показывает число циклов «чтение-изменение-запись», пропущенных из-за использования SSD-журналов.
- Когда SSD-журналирование не используется, переменная `JRMW` показывает число невыровненных запросов, которые могут сгенерировать циклы «чтение-изменение-запись» на данном жестком диске.

Повышение производительности виртуальных дисков

Виртуальные диски в виртуальных машинах с устаревшими операционными системами, такими как Windows Server 2003, Windows XP, Windows 2000, CentOS 5 или RHEL 5, могут работать медленнее из-за неправильного выравнивания разделов. Решения данной проблемы см. в разделе **Выравнивание дисков и разделов в виртуальных машинах** в *Руководстве пользователя по ПК Р-Виртуализация*.

Отключение распределения данных между уровнями

Если на уровне хранения закончилось свободное место, ПК Р-Хранилище попробует временно использовать нижние уровни. Если самый нижний уровень также оказывается полностью заполненным, ПК Р-Хранилище попытается использовать более высокий уровень. Если позже добавить дисковое пространство исходному уровню, данные, которые временно хранятся в другом месте, будут перемещены на исходный уровень, где и должны были храниться изначально.

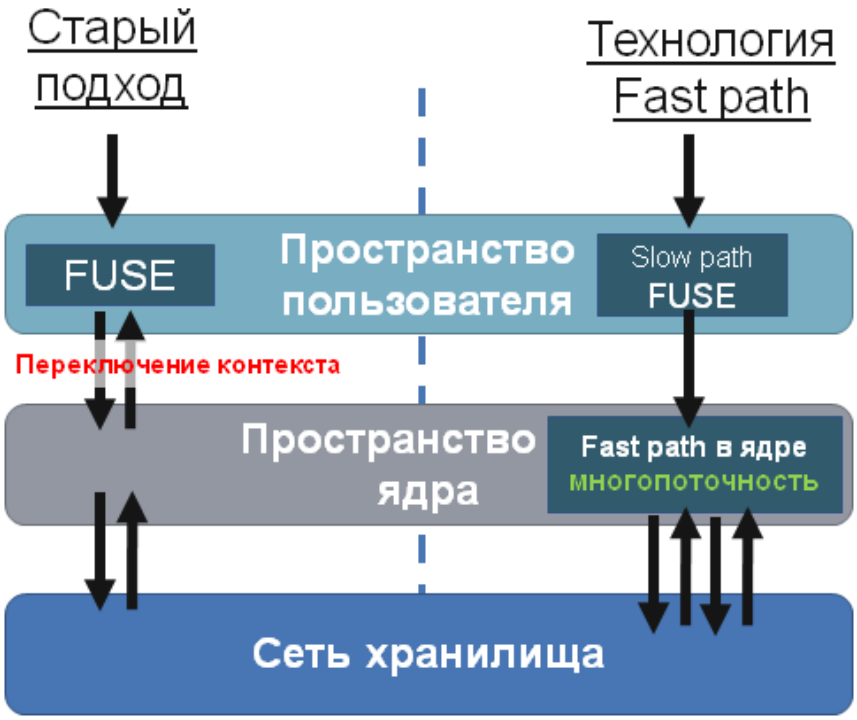
Не рекомендуется смешивать загрузки разных уровней из-за возможного снижения производительность кластера. Чтобы этого избежать, можно отключить автоматическую миграцию данных между уровнями после проверки, чтобы на каждом уровне было достаточно свободного пространства. На любом сервере кластера выполните следующую команду:

```
# vstorage -c cluster1 set-config mds.alloc.strict_tier=1
```

Включение технологии fast path

Внимание: В данный момент эта функция является экспериментальной из-за недостаточной статистики промышленного использования.

Технология fast path может значительно повысить скорость чтения ПК Р-Хранилище, если скорость операций ввода-вывода сервера является узким местом. В предыдущих версиях ПК Р-Хранилище скорость могла быть ограничена тем, что операции ввода-вывода сервера обрабатывались в одном потоке в пользовательском пространстве. В текущей версии ПК Р-Хранилище операции ввода-вывода обрабатываются с использованием многопоточности в ядре, что исключает ненужные переключения контекста и улучшает производительность.



Если скорость операций ввода-вывода сервера является проблемой, то включение технологии fast path может увеличить максимальную скорость чтения сервера в некоторых случаях до трех раз.

Обычно операции ввода-вывода становятся узким местом в кластерах, где кэш хранится на SSD-дисках или все диски являются SSD.

По умолчанию технология fast path отключена. Чтобы ее включить, нужно добавить `params='kdirect=pcs'` к опциям монтирования кластера в `/etc/fstab`, например:

```
vstorage://c1ddd1c28   /vz   fuse.vstorage   defaults,params='kdirect=pcs'   0   0
```

и перезапустить сервер.

Глава 8

Приложения

Приложение А – Устранение неисправностей

В данном разделе описываются типичные проблемы, с которыми можно столкнуться при работе с кластерами ПК Р-Хранилище, и способы их решения. Основным инструментом для решения проблем кластера и обнаружения аппаратных неисправностей является команда `vstorage top`.

Отправка отчета об ошибке службе технической поддержки

При возникновении проблемы в кластере, которую не получается решить самостоятельно, можно использовать команду `vstorage make-report` для составления подробного отчета о кластере. Затем данный отчет может быть отправлен в службу технической поддержки, которая обстоятельно изучит проблему и предпримет все возможные меры для ее устранения в кратчайшие сроки.

Для составления отчета выполните следующие действия:

1 Настройте SSH-доступ без пароля для пользователя root с того сервера, где будет выполнена команда `vstorage make-report`, для всех серверов в кластере.

Самым простым способом осуществления данной процедуры является создание SSH-ключа с помощью команды `ssh-keygen` и использование `ssh-copy-id` , чтобы настроить все серверы доверять данному ключу. Для получения подробной информации см. man-страницы по `ssh-keygen` и `ssh-copy-id`.

2 Введите команду `vstorage make-report` для составления отчета:

```
# vstorage -c stor1 make-report
The report is created and saved to vstorage-report-20121023-90630.tgz
```

Команда собирает информацию, относящуюся к кластеру на всех его серверах, и сохраняет ее в файл в текущем каталоге. Узнать точное имя файла можно, проверив вывод `vstorage` (`vstorage-report-20121023-90630.tgz` в примере выше).

При необходимости можно сохранить отчет в файл с определенным именем и поместить его в определенный каталог, добавив параметр `-f` к команде и указав желаемое имя файла (с расширением `.tgz`) и местоположение, например:

```
# vstorage -c stor1 make-report -f /home/reportSTOR1.tgz
```

Когда отчет готов, его можно оправить в службу технической поддержки. Например, для этого можно использовать инструмент Support Request Tracker. Для получения

дополнительной информации по его использованию см. *Руководство пользователя по ПК Р-Виртуализация*.

Примечание: Отчет содержит только информацию, относящуюся к настройкам и конфигурации кластера и не содержит персональной информации.

Закончилось свободное место на диске

Когда в кластере ПК Р-Хранилище остается очень мало свободного дискового пространства, очень важно увеличить его в кратчайшие сроки путем добавления новых серверов фрагментов или удаления ненужных данных. При 95% занятости дискового пространства кластера выделение новых фрагментов данных становится невозможным, а подобные запросы блокируются до тех пор, пока кластер не сможет удовлетворить их требованиям. В результате, ввод-вывод пользовательских данных также блокируется, «замораживая» контейнеры и виртуальные машины.

Примечание: Настоятельно рекомендуется иметь как минимум 10% свободного дискового пространства для восстановления в случае отказа машин. Также следует контролировать журнал использования дискового пространства, например, используя команду `vstorage top` или `vstorage get-event` (для получения дополнительной информации см. **Мониторинг кластеров ПК Р-Хранилище** (стр. 84)).

Симптомы

1. Зависающий ввод-вывод или не отвечающая точка монтирования, сообщения `dmesg` о зависании ввода-вывода, замороженное состояние контейнеров и виртуальных машин.
2. `vstorage top` и `vstorage get-event` выводят следующее сообщение об ошибке "`Failed to allocate X replicas at tier Y since only Z chunk servers are available for allocation`".

Решения

1. Удалите ненужные данные для освобождения дискового пространства.

   Примечание: Также может удивить, что как только очереди ввода-вывода в ядре оказываются заполнены блокированным вводом-выводом, точка монтирования на клиенте может зависнуть, отвечая всем одновременно, и быть не в состоянии обработать даже такие запросы, как создание списка файлов. В подобном случае можно добавить дополнительную точку монтирования для выведения списка, обращения и удаления ненужных данных.

2. Добавьте новые серверы фрагментов на неиспользуемых дисках (см. **Установка серверов фрагментов** (стр. 18)).

Если решения, приведенные выше, невозможно осуществить, следует использовать одно из следующих *временных* обходных решений:

1. Уменьшите число реплик для наименее важных данных пользователя (см. **Настройка параметров репликации** (стр. 29)). Не забудьте позже вернуть исходные настройки.
2. Уменьшить размер зарезервированного дискового пространства для выделения. Например, для кластера `stor1`:

```
# vstorage -c stor1 set-config mds.alloc.fill_margin=2
```

где `mds.alloc.fill_margin` является процентным отношением зарезервированного дискового пространства для нужд операций сервера фрагментов (по умолчанию равно 5). Не забудьте позже вернуть исходные настройки.

Низкая производительность записи

Известно, что некоторые сетевые адаптеры, такие как RTL8111/8168B, не могут осуществить передачу данных по сети с использованием полной пропускной способности и дуплексного режима, что может привести к снижению производительности операций записи.

Таким образом, перед созданием кластера ПК Р-Хранилище настоятельно рекомендуется проверить сеть на поддержку дуплексного режима. Можно использовать утилиту `netperf`, чтобы одновременно генерировать входящий и исходящий трафик. Например, в гигабитных сетях должно постоянно передаваться около 2 Гбит/с общего трафика (1 Гбит/с для входящего и 1 Гбит/с для исходящего трафика).

Низкая производительность дискового ввода-вывода

Во многих установках BIOS режим AHCI по умолчанию настроен для работы с включенным параметр**ом Legacy**. С данным параметром серверы работают с устройствами SATA через режим совместимости IDE, что может снижать производительность кластера в 2 раза, чем предполагалось. Проверить, включен или нет данный параметр, можно с помощью команды `hdparm`, например:

```
# hdparm -i /dev/sda
...
 PIO modes:  pio0 pio1 pio2 pio3 *pio4
 DMA modes:  mdma0 mdma1 mdma2
 UDMA modes: udma0 udma1 udma2 udma3 udma4 udma5 udma6
```

Звездочка (`*`) перед `pio4` в поле `PIO modes` означает, что жесткий диск `/dev/sda` в данный момент работает в режиме совместимости.

Для решения данной проблемы и повышения производительности кластера следует всегда включать параметр **AHCI** в настройках BIOS.

Кэш аппаратного RAID контролера и записей на диск

Важно, чтобы все жесткие диски выполняли команду сброса данных на диск и кэшировали данные до завершения команды. Однако не все RAID контроллеры и диски осуществляют данную процедуру, что может привести к противоречивости данных и повреждению файловой системы в случае отказа питания.

Некоторые 3ware RAID контроллеры не отключают кэш записей и не отправляют команды сброса данных на диск. В итоге, может повредиться файловая система, даже не смотря на то, что сам RAID контроллер имеет батарею. Для решения данной проблемы следует отключить кэш записей на всех дисках в RAID-массиве.

Также перед созданием кластера ПК Р-Хранилище необходимо убедиться, что заданная конфигурация прошла тщательную проверку на согласованность. Для получения дополнительной информации см. **SSD-диски игнорируют сброс данных на диск** (стр. 124).

SSD-диски игнорируют сброс данных на диск

Большинство SSD-дисков десктопного уровня могут игнорировать сброс данных на диск и обманывать операционные системы, сообщая, что данные были записаны. Примерами подобных дисков являются OCZ Vertex 3 и Intel X25-E, X-25-M G2, которые небезопасны для подтверждения данных. Такие диски не следует использовать с базами данных, так как они могут повредить файловую систему в случае отказа питания.

SSD-диски Intel третьего поколения (S3700 и 710 серии) не имеют данных проблем, так как содержат конденсатор, обеспечивающий аварийное аккумуляторное питание для сброса кэша на диск при отключении питания.

Следует использовать SSD-диски с осторожностью и только диски серверного уровня, которые подчиняются правилам сброса данных на диск. Дополнительную информацию по данной проблеме можно найти в статье о PostreSQL по ссылке http://www.postgresql.org/docs/current/static/wal-reliability.html.

Кластер не может создать достаточное число реплик

Иногда кластер не создает необходимое число фрагментов данных, даже если в кластере достаточно серверов фрагментов.

Данная проблема может возникать при создании новых серверов фрагментов путем копирования уже существующего сервера фрагментов (например, при установке сервера фрагментов в виртуальной машине и последующем клонировании этой машины). В этом случае все копированные серверы фрагментов имеют одинаковый UUID, то есть UUID исходного сервера. Кластер имеет сведения, что все серверы фрагментов находятся на исходном хосте, и не может выделить новые фрагменты данных.

Для решения данной проблемы следует сгенерировать новый UUID для клонированного сервера фрагментов с помощью выполнения следующей команды на целевом хосте:

```
# /usr/bin/uuidgen -r | tr '-' ' ' | awk '{print $1$2$3}' > /etc/vstorage/host_id
```

Для получения дополнительной информации по утилите `uuidgen` см. ее man-страницу.

Отказавшие серверы фрагментов

При отказе сервера фрагментов в кластере Р-Хранилище необходимо установить причину отказа и выбрать правильный способ решения проблемы.

Следует выполнить следующие действия:

1 Введите команду `vstorage top`. Например:

```
# vstorage -c stor1 top
```

2 Нажмите **i**, чтобы перейти к колонке FLAGS в секции серверов фрагментов, и найдите флаги, относящиеся к соответствующим отказавшим серверам фрагментов.

3 Найдите показанные флаги в таблице ниже для установления причины отказа и выбора способа решения проблемы.

Флаг	Проблема	Способ решения
H	Ошибка ввода-вывода. Диск, на котором запущен сервер фрагментов, неисправен.	Выполните проверку диска на наличие ошибок. Если диск неисправен, замените его и создайте заново сервер фрагментов, как описано в **Замена дисков, используемых в роли серверов фрагментов** (стр. 125). В противном случае, обратитесь в службу технической поддержки.
h	Несоответствие контрольных сумм фрагментов. Либо поврежден фрагмент, либо диск, на котором хранятся фрагменты, неисправен.	Выполните проверку диска на наличие ошибок. Если диск неисправен, замените его и создайте заново сервер фрагментов, как описано в **Замена дисков, используемых в роли серверов фрагментов** (стр. 125). В противном случае, обратитесь в службу технической поддержки.
S	Журнал сервера фрагментов, хранящийся на SSD-диске с журналированием, недоступен. Либо поврежден журнал, либо SSD-диск с журналированием неисправен.	Выполните проверку SSD-диска с журналированием на наличие ошибок. Если диск неисправен, замените его, как описано в **Отказавшие SSD-диски с журналированием записей** (стр. 126).
s	Контрольные суммы фрагментов, хранящиеся на SSD-диске с кэшированием, недоступны. Либо повреждены контрольные суммы, либо SSD-диск с кэшированием неисправен.	Выполните проверку SSD-диска с кэшированием на наличие ошибок. Если диск неисправен, замените его, как описано в **Отказавшие SSD-диски с кэшированием данных** (стр. 126).
R	Путь к хранилищу фрагментов недействителен при запуске сервера фрагментов. Диск, на котором запущен сервер фрагментов, не подключен или не монтирован.	Убедитесь, что диск подключен и монтирован правильным образом. Проверьте, чтобы дисковая запись в `/etc/fstab` была верной.
T	Превышено время ожидания для запроса ввода-вывода. Диск может быть недоступен по некоторым причинам и необязательно неисправен.	Убедитесь, что диск подключен и проверьте вывод `dmesg` для сообщений о превышении времени ожидания для запросов ввода-вывода, чтобы установить причину недоступности диска.

Замена дисков, используемых в роли серверов фрагментов

Чтобы осуществить замену HDD- или SSD-диска, используемого в роли сервера фрагментов, выполните следующие действия:

1 Удалите отказавший сервер фрагментов из кластера, как описано в подразделе **Удаление серверов фрагментов** (стр. 26).

Примечание: Не следует отключать неисправный диск до удаления сервера фрагментов.

2 Замените неисправный диск новым.

Если включена горячая замена, новый сервер фрагментов будет создан автоматически в течение одной минуты. В противном случае, выполните следующие действия:

1 Подготовьте SSD-диск, как описано в подразделе **Подготовка дисков для ПК Р-Хранилище** (стр. 15).

2 Создайте новый сервер фрагментов, следуя инструкции в подразделе **Создание серверов фрагментов** (стр. 18).

Отказавшие SSD-диски с журналированием записей

При неисправности SSD-диска, используемого для хранения журналов записей, произойдет отказ всех серверов фрагментов, у которых есть журналы на данном SSD-диске. Кластер будет продолжать функционировать и создавать реплики, чтобы восполнить потери. Если необходимо создать те же журналы записей на новом SSD-диске, выполните следующие действия:

1 Удалите отказавшие серверы фрагментов, как описано в подразделе **Удаление серверов фрагментов** (стр. 26).

2 Подготовьте SSD-диск, как описано в подразделе **Подготовка дисков для ПК Р-Хранилище** (стр. 15).

3 Создайте новые серверы фрагментов, которые будут иметь журналы записей на новом SSD-диске, следуя инструкции в подразделе **Настройка SSD-дисков для журналирования записей** (стр. 112).

Отказавшие SSD-диски с кэшированием данных

При неисправности SSD-диска, используемого для хранения кэша считывания, произойдет потеря кэша, и клиент продолжит работать без него. Если необходимо создать тот же кэш считывания на новом SSD-диске, выполните следующие действия:

1 Остановите все виртуальные среды, запущенные на клиенте, или мигрируйте их на другой сервер.

2 Размонтируйте кластер. Например:

```
# umount /vstorage/stor1
```

3 Подготовьте SSD-диск, как описано в подразделе **Подготовка дисков для ПК Р-Хранилище** (стр. 15).

4 Создайте кэш и монтируйте кластер, следуя инструкции в подразделе **Настройка SSD-дисков для кэширования данных** (стр. 115).

Отказавшие серверы метаданных

При неисправности диска, на котором находится сервер метаданных, следует заменить его следующим образом:

1 Удалите отказавший сервер метаданных, как описано в подразделе **Удаление серверов метаданных** (стр. 25).

2 Создайте новый сервер метаданных, следуя инструкции в подразделе **Добавление серверов метаданных** (стр. 23).

Приложение Б – Часто задаваемые вопросы

В данном приложении перечислены самые часто задаваемые вопросы о кластерах ПК Р-Хранилище.

Общие

Можно ли использовать директорию /pstorage в новых установках?

Да. Для совместимости в новых установках директория `/pstorage` остается в качестве символьной ссылки на новую директорию `/vstorage`.

Нужно ли покупать дополнительные устройства хранения для ПК Р-Хранилище?

Нет. ПК Р-Хранилище не требует внешних устройств хранения, которые обычно используются в SAN путем конвертирования систем хранения данных с локальным подключением от нескольких серверов в общую систему хранения.

Какие аппаратные требования для ПК Р-Хранилище?

ПК Р-Хранилище не требует специального оборудования и может работать на бюджетных компьютерах с традиционными дисками SATA и гигабитными сетями. Некоторые жесткие диски и RAID контроллеры, правда, игнорируют команду FLUSH для имитации лучшей производительности, и их не следует использовать в кластерах, так как их использование может привести к повреждениям файловой системы и журналов. Данная рекомендация относится, в первую очередь, к RAID контроллерам и SSD-дискам. Чтобы удостовериться в использовании надежного оборудования, следует обратиться к справочной документации по жесткому диску.

Для получения дополнительной информации *Руководство по установке ПК Р-Виртуализация*.

Сколько требуется серверов для развертывания кластера ПК Р-Хранилище?

Чтобы создать кластер ПК Р-Хранилище, требуется только один физический сервер. Однако для обеспечения высокой доступности данных рекомендуется настроить кластер

таким образом, чтобы для каждого фрагмента данных создавалось как минимум по 3 реплики. Для данной рекомендации требуется, по крайней мере, 3 запущенных сервера — и как минимум 5 серверов в целом — в кластере. Для получения дополнительной информации см. *Руководство по установке ПК Р-Виртуализация* и **Настройка параметров репликации** (стр. 29).

Можно ли объединить физические серверы с разными поддерживаемыми операционными системами в один кластер ПК Р-Хранилище?

Да. Можно создать кластеры ПК Р-Хранилище, состоящие из физических серверов, на которых запущены любые поддерживаемые операционные системы. Например, можно установить серверы метаданных на физические серверы с Ubuntu 14.04, серверы фрагментов – на физические серверы с Red Hat Enterprise Linux 7, а клиенты – на компьютеры с CentOS 7.

Примечание: Текущая автономная версия ПК Р-Хранилище не поддерживает ПК Р-Виртуализация.

Масштабируемость и производительность

Сколько серверов можно добавить к кластеру ПК Р-Хранилище?

Ограничения по количеству серверов, которые можно добавить к кластеру ПК Р-Хранилище, нет. Однако рекомендуется ограничить количество серверов в кластере до одной стойки, чтобы предотвратить возможное снижение производительности из-за передачи данных между стойками.

Сколько дискового пространства может иметь кластер ПК Р-Хранилище?

Кластер ПК Р-Хранилище может поддерживать до 8 ПБ доступного дискового пространства, что составляет до 24 ПБ физического дискового пространства при хранении трех реплик для каждого фрагмента данных.

Можно ли добавить серверы к уже существующему кластеру ПК Р-Хранилище?

Да, можно динамически добавлять серверы к кластеру ПК Р-Хранилище и удалять их для повышения его емкости или выключения серверов в течение обслуживания. Для получения дополнительной информации см. **Настройка серверов фрагментов** (стр. 25).

Какова ожидаемая производительность кластера ПК Р-Хранилище?

Производительность зависит от скорости работы сети и жестких дисков, используемых в кластере. В общих чертах, производительность должна быть той же, что у системы хранения данных с локальным подключением, или лучше. Также можно кэшировать данные на SSD-диск для повышения производительности бюджетного оборудования. Для получения дополнительной информации см. **Использование SSD-дисков** (стр. 110).

Какова производительность кластера с гигабитной сетью Ethernet?

Максимальная скорость гигабитной сети примерно равна скорости одного диска с **вращением**. Однако в большинстве рабочих нагрузок превалирует доступ ввода-вывода в произвольном порядке, и сеть не является узким местом. Исследование крупных провайдеров хостинга доказало, что средняя производительность ввода-вывода редко превышает 20 Мб/с из-за случайного распределения. Виртуализация также добавляет произвольное распределение, так как разные независимые среды одновременно осуществляют доступ ввода-вывода. Тем не менее, использование 10-гигабитной сети Ethernet обеспечит лучшую производительность и рекомендуется для рабочей среды.

Повысится ли общая производительность кластера при добавлении к нему новых серверов фрагментов?

Да. Так как данные распределяются между всеми жесткими дисками в кластере, у приложений, осуществляющих ввод-вывод в произвольном порядке, увеличивается значение операций ввода-вывода в секунду при добавлении серверов к кластеру. Даже одна **машина клиента** может получить ощутимую выгоду от увеличения количества серверов фрагментов, и иметь производительность, намного превосходящую ту, что имеют системы хранения данных с локальным подключением.

Зависит ли производительность от числа реплик фрагментов?

Каждая дополнительная реплика снижает производительность записи примерно на 10%, но в то же время может повысить производительность считывания, так как у кластера ПК Р-Хранилище появляется больше параметров для выбора сервера с более высокой скоростью.

Доступность

Как ПК Р-Хранилище осуществляет защиту данных?

ПК Р-Хранилище защищает от потери данных и временной недоступности с помощью создания копий данных (реплик) и хранения их на разных серверах. Для обеспечения дополнительной надежности можно настроить ПК Р-Хранилище для вычисления контрольных сумм данных пользователя и их проверки при необходимости.

Что происходит при потере диска или если сервер недоступен?

ПК Р-Хранилище автоматически восстанавливается после системного сбоя на указанный уровень избыточности путем осуществления репликации данных на запущенных серверах. Пользователи имеют доступ к данным во время процесса восстановления.

Как быстро осуществляется восстановление ПК Р-Хранилище после системного сбоя?

Так как ПК Р-Хранилище восстанавливается после системного сбоя, используя все доступные жесткие диски в кластере, процесс восстановления занимает намного меньше времени, чем у обычных RAID-массивов с локальным подключением. Таким образом, значительно повышается надежность системы хранения, так как возможность потери единственной оставшейся копии данных в течение периода восстановления очень мала.

Можно ли изменять настройки избыточности на лету?

Да, в любой момент можно изменить число копий данных, и ПК Р-Хранилище будет применять новые настройки, создавая новые копии или удаляя ненужные. Для получения дополнительной информации по настройке параметров репликации см. **Настройка параметров репликации** (стр. 29).

Нужно ли еще использовать локальные RAID-массивы?

Нет, ПК Р-Хранилище обеспечивает ту же встроенную избыточность данных, что и зеркальный дисковый массив RAID1 с множеством копий. Однако для повышения последовательной производительности можно использовать локальный массив с чередованием RAID0, экспортированный в кластер ПК Р-Хранилище. Для получения дополнительной информации по использованию RAID-массивов см. **Возможные конфигурации дисковых накопителей** (стр. 106).

Имеет ли ПК Р-Хранилище уровни избыточности схожие с RAID5?

Нет. Для создания надежной программной системы RAID5 также необходимо использовать специальные аппаратные возможности, такие как резервные батареи питания. В будущем, возможно, ПК Р-Хранилище будет обеспечивать избыточность уровня RAID5 для данных с доступом только для чтения, например, для резервных копий.

Сколько рекомендуется иметь копий данных?

Рекомендуется настроить ПК Р-Хранилище для создания 2 или 3 копий, что позволяет кластеру пережить потерю одновременно одного или двух жестких дисков.

Работа кластера

Как узнать, что новые параметры репликации были успешно применены к кластеру?

Для проверки завершения процесса репликации введите команду `vstorage top`, нажмите клавишу **v** и посмотрите информацию в поле `Chunks` в выводе команды:

- При уменьшении значений параметров репликации в выводе не должно быть фрагментов со статусами `overcommitted` и `deleting`.
- При увеличении значений параметров репликации в выводе не должно быть фрагментов со статусами `blocked` и `urgent`. Также значение параметра `healthy` должно быть 100%.

Для получения дополнительной информации см. **Мониторинг статуса параметров репликации** (стр. 97).

Как завершить работу кластера?

Для завершения работы кластеры ПК Р-Хранилище выполните следующие действия:

1 Остановите все клиенты.

2 Остановите все серверы метаданных.

3 Остановите все серверы фрагментов.

Для получения дополнительной информации см. **Завершение работы кластеров ПК Р-Хранилище** (стр. 39).

Какой инструмент следует использовать для мониторинга статуса и состояния кластера?

Мониторинг статуса и состояния кластера можно осуществлять при помощи команды `vstorage top`. Для получения дополнительной информации см. **Мониторинг кластеров ПК Р-Хранилище** (стр. 84).

Чтобы посмотреть общий размер дискового пространства, занимаемого всеми данными пользователя, введите команду `vstorage top`, нажмите клавишу **v** и посмотрите информацию в поле FS в выводе команды. В поле FS отображается общий размер используемого дискового пространства в кластере и количество файлов, в которых хранятся данные. Для получения дополнительной информации см. **Использование дискового пространства** (стр. 88).

Как настроить сервер ПК Р-Виртуализация для кластера?

Для подготовки сервера с ПК Р-Виртуализация для работы в кластере необходимо указать, чтобы сервер хранил контейнеры и виртуальные машины в кластере, а не на локальном диске. Для получения дополнительной информации см. **Настройка виртуальных машин и контейнеров** (стр. 22).

Почему vmstat/top и vstorage stat показывают разное время ввода-вывода?

Утилиты `vstorage` и `vmstat/top` используют разные методы для вычисления процентного отношения времени процессора, затраченного на ожидание дискового ввода-вывода (`wa%` в `top`, `wa` в `vmstat` и `IOWAIT` в `vstorage`). Утилиты `vmstat` и `top` отмечают, что ЦП находится в состоянии ожидания, только если на этом ЦП есть ожидающий выполнения запрос ввода-вывода, а утилита `vstorage` отмечает, что ЦП находится в состоянии ожидания вне зависимости от числа запросов ввода-вывода, ожидающих выполнения. Таким образом, `vstorage` может показать более высокие значения ввода-вывода. Например, для системы с 4 ЦП и одним потоком, осуществляющим ввод-вывод, `vstorage` покажет более 90% времени ожидания ввода-вывода, а `vmstat` и `top` – не более 25% времени ввода-вывода.

Как номер уровня влияет на работу ПК Р-Хранилище?

При назначении уровню дискового пространства нужно иметь в виду, что диски с более высокой скоростью нужно назначать более высоким уровням. Например, уровень 0 можно использовать для резервных копий и других «холодных» данных (сервер фрагментов без SSD-журналов), уровень 1 – для виртуальных сред: много «холодных» данных, но быстрая произвольная запись (сервер фрагментов с SSD-журналами), уровень 2 – для

«горячих» данных (сервер фрагментов на SSD-диске), журналов, кэша, отдельных дисков виртуальных машин и т.п.

Данная рекомендация связана с тем, как ПК Р-Хранилище работает с дисковым пространством. Если на уровне хранения заканчивается свободное пространство, ПК Р-Хранилище пробует временно использовать более низкий уровень, а если и они заполнены, то более высокий уровень. Если позже добавить дисковое пространство исходному уровню, данные, которые временно хранятся в другом месте, будут перемещены на исходный уровень, где и должны были храниться изначально.

Например, если уровень 2 заполнен, то при попытке записать на него данные ПК Р-Хранилище попробует записать их на уровень 1, затем на уровень 0, а потом на уровень 3. Если позже добавить дополнительное дисковое пространство к уровню 2, то эти данные, в данный момент хранящиеся на уровне 1, либо 0, либо 3, будут перемещены обратно на уровень 2, где они должны были храниться с самого начала.