Утвержден

РДЦП.10001-02-УД

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

ПРОГРАММНЫЙ КОМПЛЕКС «СРЕДСТВА ВИРТУАЛИЗАЦИИ «БРЕСТ»

Руководство администратора. Часть 1

РДЦП.10001-02 95 01-1

Листов 90

2022

Литера $О_1$

АННОТАЦИЯ

Настоящий документ является руководством администратора программного изделия «Программный комплекс «Средства виртуализации «Брест» (ПК СВ «Брест») РДЦП.10001-02 (далее — ПК СВ).

В документе приведено описание порядка развертывания и настройки ПК СВ с учетом особенностей операционной системы специального назначения «Astra Linux Special Edition» РУСБ.10015-01 (далее по тексту — ОС СН), под управлением которой функционирует ПК СВ.

Документ предназначен для использования совместно с эксплуатационными документами согласно ведомости РДЦП.10001-02 20 01 «Программный комплекс «Средства виртуализации «Брест». Ведомость эксплуатационных документов».

Руководство администратора состоит из двух частей:

- РДЦП.10001-02 95 01-1 «Программный комплекс «Средства виртуализации «Брест». Руководство администратора. Часть 1»;
- РДЦП.10001-02 95 01-2 «Программный комплекс «Средства виртуализации «Брест». Руководство администратора. Часть 2».

В первой части руководства описан порядок развертывания и первичной настройки ПК СВ.

Во второй части руководства представлен порядок использования среды виртуализации, обеспечения отказоустойчивости и масштабирования развернутого ПК СВ.

Дополнительная информация о порядке эксплуатации, а также варианты реализации отдельных решений приведены на официальном сайте `wiki.astralinux.ru/docs`.

СОДЕРЖАНИЕ

1. ОБЩИЕ СВЕДЕНИЯ

1.1. Обзор архитектуры

Основными программными компонентами ПК СВ являются (см. рис. 1):

- узел виртуализации — сервис, предоставляющий необходимые вычислительные ресурсы для виртуальных машин;
- фронтальная машина — сервис, обеспечивающий управление узлами виртуализации и виртуальными машинами. Также предоставляет веб-интерфейс администратора ПК СВ;
- облачное хранилище данных — система, предназначенная для хранения образов дисков виртуальных машин. Может быть построена на базе следующих технологий хранения:
    - файловой технологии хранения (с использованием локальной файловой системы или кластерной файловой системы, например, ocfs2 или nfs),
    - блочной технологии хранения с использованием LVM,
    - технологии хранения Ceph;
- контроллер домена — сервис, обеспечивающий аутентификацию пользователей в рамках единого пространства пользователей (не используется в сервисном режиме работы ПК СВ).

Примечание. В ПК СВ в качестве службы управления единым пространством пользователей используется FreeIPA. Если на объекте эксплуатации уже имеется настроенный домен FreeIPA, то разворачивать дополнительный контроллер домена нет необходимости. Все серверы вводятся в существующий домен.

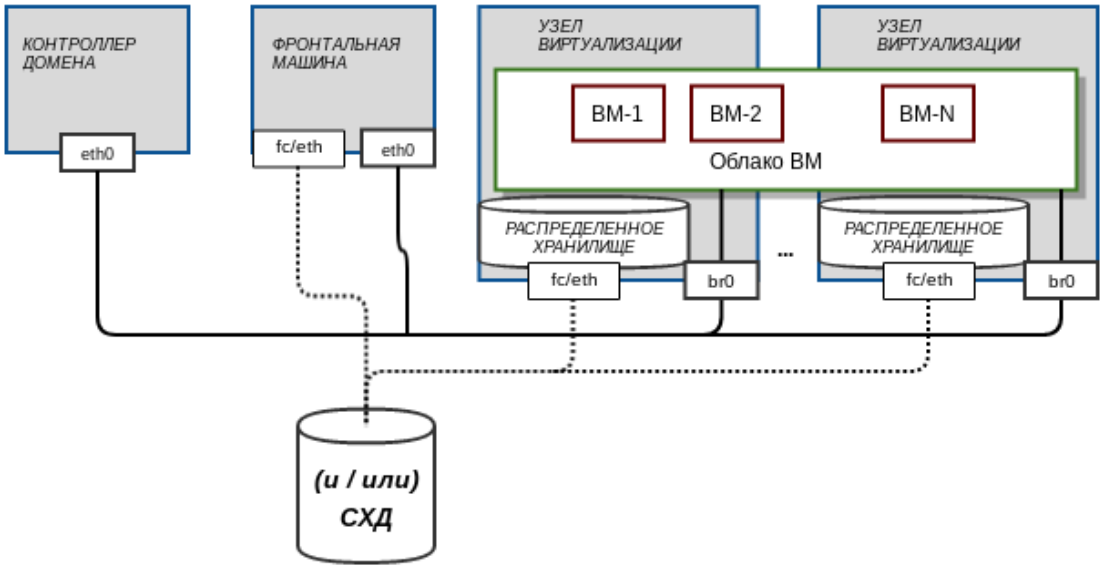

Рис. 1

ПК СВ может быть развернут как на группе физических серверов, так и на виртуальных машинах в пределах одного сервера или рабочей станции для тестирования. Для объединения физических серверов, обеспечения выполнения операций управления и поддержки облачных сетей для виртуальных машин используется физическая сеть.

Примечание. Допускается разворачивать несколько программных компонент ПК СВ на одном физическом сервере.

ВНИМАНИЕ! В связи с особенностью функционирования домена FreeIPA, конфигурация, при которой разворачиваются сервисы контроллера домена и фронтальной машины на одном сервере, недопустима.

1.2. Последовательность развертывания ПК СВ

Для развертывания ПК СВ необходимо выполнить следующие действия:

1) на серверы установить ОС СН. Процесс установки ОС СН описан в документе «Операционная система специального назначения «Astra Linux Special Edition» РУСБ.10015-01. Руководство по установке» (файл `OS-inst-help.pdf`, размещенный на установочном носителе в директории `install-doc`). При этом следует учитывать следующие особенности установки:

- на странице **Выбор программного обеспечения** дополнительно выбрать пункт `Средства Виртуализации`;

  Примечание. Пункт `Графический интерфейс Fly` допускается не выбирать для установки.

- на странице **Дополнительные настройки ОС**:
  - если планируется функционирование ПК СВ в дискреционном режиме, выбрать максимальный уровень защищенности («Смоленск») или усиленный уровень защищенности («Воронеж»),
  - если планируется функционирование ПК СВ в сервисном режиме, выбрать базовый уровень защищенности («Орел»);
- на странице **Дополнительные настройки ОС** дополнительно выбрать пункт `Запрет автонастройки сети`.

2) установить оперативное обновление (БЮЛЛЕТЕНЬ № 2022-0819SE17) в соответствии с указаниями, представленными в бюллетене;

3) установить оперативное обновление (БЮЛЛЕТЕНЬ № 2022-1011SE17MD) в соответствии с указаниями, представленными в бюллетене;

4) после применения оперативных обновлений установить ядро `linux-5.10-generic` или `linux-5.15-generic` (предпочтительно);

5) настроить физическую сеть, объединяющую серверы. Порядок настройки сети

представлен в документе РУСБ.10015-01 95 01-1 «Операционная система специального назначения «Astra Linux Special Edition». Руководство администратора. Часть 1» из комплекта поставки ОС СН;

6) установить и настроить сервис контроллера домена (не выполняется, если планируется функционирование ПК СВ в сервисном режиме). Порядок установки и настройки контроллера домена представлен в документе РУСБ.10015-01 95 01-1 «Операционная система специального назначения «Astra Linux Special Edition». Руководство администратора. Часть 1» из комплекта поставки ОС СН;

Примечание. Если сервис контроллера домена устанавливается на отдельном сервере, то при установке ОС СН на этот сервер также необходимо выбрать максимальный уровень защищенности («Смоленск») или усиленный уровень защищенности («Воронеж»).

7) установить сервис фронтальной машины на необходимое количество серверов (см. 2.2);

8) настроить подключение к веб-интерфейсу ПК СВ (см. 2.3);

9) установить сервис узла виртуализации на необходимое количество серверов (см. 2.4);

10) настроить облачное хранилище данных (см. 3);

11) настроить облачную сеть (см. 4);

12) выполнить дополнительное конфигурирование ПК СВ (см. 5);

13) при необходимости выполнить дополнительную настройку систем мониторинга и регистрации событий (см. 6);

14) если требуется обеспечить отказоустойчивость системы управления, настроить взаимодействие фронтальных машин по алгоритму RAFT (см. 2.6).

Примечание. Несколько экземпляров ПК СВ могут быть объединены в единый центр обработки и хранения данных (ЦОХД), называемый «федерация». Подробная информация представлена в 7.

2. УСТАНОВКА СЕРВИСОВ ПК СВ

ВНИМАНИЕ! Действия по установке и настройке сервисов ПК СВ выполняются в ОС СН под учетной записью администратора с высоким уровнем целостности.

2.1. Подготовка к установке сервисов ПК СВ

Перед установкой сервисов фронтальной машины и/или узла виртуализации на сервере необходимо выполнить следующие действия:

1) настроить доступ к репозиториям:
   - основному репозиторию (репозиторию установочного диска, main);
   - оперативному обновлению основного репозитория (бюллетень № 2022-0819SE17);
   - оперативному обновлению основного репозитория (бюллетень № 2022-1011SE17MD).

При этом может быть указан сетевой репозиторий или копия репозитория в локальной файловой системе (ФС). Для того чтобы подключить репозиторий ОС СН, следует в файле `/etc/apt/sources.list` добавить строку вида:

```
deb <путь_к_репозиторию> 1.7_x86-64 main contrib non-free
```

Примеры:

1. копия репозитория в локальной файловой системе

```
deb file:/srv/repo/main/ 1.7_x86-64 main contrib non-free
deb file:/srv/repo/0819SE17/ 1.7_x86-64 main contrib non-free
deb file:/srv/repo/1011SE17MD/ 1.7_x86-64 main contrib non-free
```

где `/srv/repo/main/` — каталог, в котором размещены файлы установочного диска ОС СН;

`/srv/repo/0819SE17/` — каталог, в котором размещены файлы оперативного обновления основного репозитория (бюллетень № 2022-0819SE17);

`/srv/repo/1011SE17MD/` — каталог, в котором размещены файлы оперативного обновления основного репозитория (бюллетень № 2022-1011SE17MD).

2. интернет-репозитории ОС СН:

```
# Репозиторий файлов установочного диска ОС~СН (основной репозиторий)
deb https://dl.astralinux.ru/astra/stable/1.7_x86-64/repository-main \
      1.7_x86-64 main contrib non-free

# Репозиторий оперативного обновления основного репозитория
# (бюллетень 2022-0819SE17)
deb https://dl.astralinux.ru/astra/frozen/1.7_x86-64/1.7.2/repository-update\
```

```
      1.7_x86-64 main contrib non-free

# Репозиторий оперативного обновления основного репозитория
# (бюллетень 2022-1011SE17MD)
deb https://dl.astralinux.ru/astra/frozen/1.7_x86-64/1.7.2/uu/1/\
      repository-update 1.7_x86-64 main contrib non-free
```

2) настроить доступ к репозиторию ПК СВ, при этом может быть указан сетевой репозиторий или копия репозитория в локальной ФС. Для того чтобы подключить репозиторий ПК СВ, следует в файле `/etc/apt/sources.list` добавить строку вида:

```
deb <путь_к_репозиторию> brest main non-free
```

Пример

```
deb file:/srv/repo/brest/ brest main non-free
```

где `/srv/repo/brest/` — каталог, в котором размещены файлы установочного диска ПК СВ.

3) выполнить повторную синхронизацию файлов описаний пакетов с их источником командой:

```
sudo apt update
```

4) выполнить обновление пакетов командой:

```
sudo astra-update -A -r
```

5) ввести сервер в домен FreeIPA (не выполняется если планируется функционирование ПК СВ в сервисном режиме). Порядок ввода сервера в домен представлен в документе РУСБ.10015-01 95 01-1 «Операционная система специального назначения «Astra Linux Special Edition». Руководство администратора. Часть 1» из комплекта поставки ОС СН.

2.2. Установка и инициализация сервиса фронтальной машины

В данном подразделе представлен порядок установки и инициализации сервиса фронтальной машины для дискреционного режима работы ПК СВ. Особенности установки и инициализации сервисного режима работы ПК СВ представлены в 2.7.

Примечание. Процесс инициализации сервиса фронтальной машины с помощью плейбука Ansible описан в 2.5.

Для установки и инициализации сервиса фронтальной машины необходимо выполнить следующие действия:

1) на сервере с ролью фронтальной машины установить пакет `brestcloud-ipa` командой:

```
sudo apt install brestcloud-ipa
```

В открывшемся окне **ЛИЦЕНЗИОННОЕ СОГЛАШЕНИЕ** нажать кнопку **[Принять]**;

2) перезагрузить сервер;

3) выполнить инициализацию сервиса фронтальной машины командой:

```
sudo brestcloud-configure
```

В процессе инициализации сервиса фронтальной машины необходимо:

а) указать имя администратора домена (`имя администратора ipa-сервера`), заданное во время выполнения действий по установке и настройке сервиса контроллера домена, и нажать клавишу **<Enter>**,

б) ввести пароль администратора домена, заданный во время выполнения действий по установке и настройке сервиса контроллера домена, и нажать клавишу **<Enter>**),

в) задать имя учетной записи администратора ПК СВ (имя `brestadmin` зарезервировано и не может быть использовано),

г) задать пароль учетной записи администратора ПК СВ.

ВНИМАНИЕ! Пароль администратора ПК СВ должен удовлетворять следующим требованиям сложности:

- быть длиной не менее 8 символов,
- пароль должен содержать символы из не менее чем 3-х групп:
  - латинские буквы в нижнем регистре,
  - латинские буквы в верхнем регистре,
  - цифры,
  - служебные символы.

Об успешной инициализации сервиса фронтальной машины будет свидетельствовать следующая надпись:

```
Настройка прошла успешно!
```

2.3. Настройка браузера Mozilla Firefox и подключение к веб-интерфейсу ПК СВ

В данном подразделе представлен порядок настройки браузера Mozilla Firefox для подключения к веб-интерфейсу ПК СВ, функционирующему в дискреционном режиме. Особенности настройки браузера Mozilla Firefox для сервисного режима работы ПК СВ представлены в 2.7.

Управление ПК СВ осуществляется с помощью веб-интерфейса по адресу `https://<полное_доменное_имя>/`,

где `<полное_доменное_имя>` — полное доменное имя сервера, на котором развернут сервис фронтальной машины.

Примечание. Подключение к веб-интерфейсу можно осуществить с любого

компьютера, имеющего сетевой доступ к серверу, на котором развернут сервис фронтальной машины.

Чтобы настроить браузер для использования самоподписанных ssl-сертификатов, необходимо выполнить следующие действия:

1) установить браузер Mozilla Firefox (если при установке ОС СН не был выбран пункт **Средства работы в сети**) командой:

```
sudo apt install firefox
```

2) запустить браузер, например, с использованием графического интерфейса: «Пуск — Сеть — Веб-браузер Firefox»;

3) в адресную строку ввести `about:config` и нажать клавишу **<Enter>**;

4) на открывшейся странице **Расширенные настройки** в поле поиска ввести следующее слово: `negotiate`;

5) для параметров `network.negotiate-auth.trusted-uris` и `network.negotiate-auth.delegation-uris` установить значение: «`http://, https://`»;

6) добавить в исключение самоподписной ssl сертифкат, для этого:

а) перейти по адресу: `https://<полное_доменное_имя>:2616`,

где `<полное_доменное_имя>` — полное доменное имя сервера, на котором развернут сервис фронтальной машины,

б) на открывшейся странице с предупреждением нажать на кнопку **[Дополнительно]**, а затем — на кнопку **[Принять риск и продолжить]**,

в) на открывшейся странице **Open Nebula** вводить ничего не нужно;

7) аналогичным образом добавить в исключение самоподписной ssl сертифкат для портов 443 и 29876 (используется для подключения к удаленному рабочему столу ВМ). Открывшуюся страницу с сообщением об ошибке можно закрыть;

8) перейти к веб-интерфейсу ПК СВ по адресу: `https://<полное_доменное_имя>`;

9) на открывшейся странице с предупреждением нажать на кнопку **[Дополнительно]**, а затем — на кнопку **[Принять риск и продолжить]**;

10) в открывшемся окне авторизации ввести имя и пароль доменной учетной записи, или аутентификационные параметры администратора ПК СВ, заданные во время выполнения действий по инициализации сервиса фронтальной машины (см. 2.2) и нажать кнопку **[Войти]**;

11) на открывшейся странице **Брест** нажать на кнопку **[Войти]**.

2.4. Установка и инициализация сервиса узла виртуализации

В данном подразделе представлен порядок установки и инициализации сервиса узла виртуализации для дискреционного режима работы ПК СВ.

Особенности установки и инициализации сервисного режима работы ПК СВ представлены в 2.7.

Примечание. Процесс инициализации сервиса узла виртуализации с помощью плейбуков Ansible описан в 2.5.

Для установки и инициализации сервиса узла виртуализации необходимо выполнить следующие действия:

1) установить пакет `ipa-libvirt-qemu` командой:

```
sudo apt install ipa-libvirt-qemu
```

В открывшемся окне **ЛИЦЕНЗИОННОЕ СОГЛАШЕНИЕ** нажать кнопку **[Принять]**;

2) перезагрузить сервер;

3) выполнить инициализацию сервиса узла виртуализации командой:

```
sudo ipa-libvirt-qemu-configure
```

В процессе инициализации сервиса узла виртуализации необходимо:

а) указать имя администратора домена (`имя администратора ipa-сервера`), заданное во время выполнения действий по установке и настройке сервиса контроллера домена, и нажать клавишу **<Enter>**),

б) ввести пароль администратора домена, заданный во время выполнения действий по установке и настройке сервиса контроллера домена, и нажать клавишу **<Enter>**),

в) ввести полное доменное имя фронтальной машины и нажать клавишу **<Enter>**),

г) ввести имя локального администратора фронтальной машины,

д) ввести пароль локального администратора фронтальной машины.

Об успешной инициализации сервиса узла виртуализации будет свидетельствовать следующая надпись:

```
Настройка прошла успешно!
```

2.5. Инициализация сервисов ПК СВ с помощью плейбуков Ansible

В домене FreeIPA с помощью программного средства Ansible возможно удаленно инициировать сервис фронтальной машины и узла виртуализации.

Примечание. На серверах предварительно должны быть установлены пакеты `brestcloud-ipa` (для сервиса фронтальной машины) или `ipa-libvirt-qemu` (для сервиса узла виртуализации).

Для того чтобы инициировать сервис фронтальной машины и узла виртуализации,

необходимо на машине, с которой будет производиться настройка, от имени администратора с высоким уровнем целостности выполнить следующие шаги:

1) установить Ansible командой:

```
sudo apt install brest-ansible
```

2) скопировать каталог с плейбуками Ansible в домашний каталог, выполнив команду:

```
cp -r /var/lib/brest-ansible $HOME
```

3) скорректировать файл `~/brest-ansible/inventory.ini`;

Пример

```
[all:vars]
### FreeIPA
freeipa_server_fqdn="astral.m.dom"
freeipa_admin="admin"
freeipa_admin_pass="Asdf1234"

### Брест
freeipa_brestadmin="brestchief"
freeipa_brestadmin_pass="Asdf1234"

[brest-front]
brest_front ansible_host='10.10.10.108' ansible_user='toor'
ansible_password='qwertyl23' ansible_become_pass='{{ ansible_password }}'

[brest-nodes]
brest_node_l ansible_host='10.10.10.103' ansible_user='toor'
ansible_password='qwertyl23' ansible_become_pass='{{ ansible_password }}'
```

где `freeipa_server_fqdn` — полное доменное имя контроллера домена,

`freeipa_admin` — имя администратора домена,

`freeipa_admin_pass` — пароль администратора домена,

`freeipa_brestadmin` — имя доменной учетной записи администратора ПК СВ (имя `brestadmin` зарезервировано и не может быть использовано),

`freeipa_brestadmin_pass` — пароль администратора ПК СВ.

ВНИМАНИЕ! Пароль администратора ПК СВ должен удовлетворять следующим требованиям сложности:

- быть длиной не менее 8 символов,
- пароль должен содержать символы из не менее чем 3-х групп:
  - латинские буквы в нижнем регистре,
  - латинские буквы в верхнем регистре,
  - цифры,
  - служебные символы;

`[brest-front]` — группа для описания серверов с ролью фронтальной машины,

`[brest-nodes]` — группа для описания серверов с ролью узла виртуализации,

`ansible_host` — IP-адрес или полное доменное имя сервера (с ролью фронтальной машины или узла виртуализации),

`ansible_user` — имя локального администратора сервера (с ролью фронтальной машины или узла виртуализации),

`ansible_password` — пароль локального администратора сервера (с ролью фронтальной машины или узла виртуализации),

`ansible_become_pass` — пароль для команды sudo. Если совпадает с паролем администратора или пароль для sudo не требуется, оставить без изменений значение «`{{ ansible_password }}`»;

4) перейти в каталог с плейбуками:

```
cd ~/brest-ansible
```

5) запустить плейбук конфигурирования:

- команда для инициализации сервиса фронтальной машины:

```
ansible-playbook brestcloud_ipa_configure.yml
```

- команда для инициализации сервиса узла виртуализации:

```
ansible-playbook brestcloud_ipa_kvm_nodes.yml
```

Об успешной инициализации сервиса будет свидетельствовать следующая надпись:

```
Настройка прошла успешно!
```

2.6. Алгоритм Raft

2.6.1. Общие сведения

Для обеспечения отказоустойчивости сервиса фронтальной машины в ПК СВ применяется технология Raft.

Алгоритм Raft позволяет объединять несколько экземпляров фронтальной машины в зону, конфигурацию которой можно менять (добавлять и удалять экземпляры фронтальной машины), не прерывая работу ПК СВ. Для этой зоны выделяется плавающий (способный при необходимости переходить от одного экземпляра к другому) IP-адрес. Из доступных экземпляров выбирается лидер, которому присваивается ранее выделенный IP-адрес. Лидер обслуживает все входящие запросы. Все изменения на лидере синхронизируются с остальными экземплярами фронтальной машины в зоне. Если работа лидера прерывается на 100 миллисекунд, то выбирается новый лидер из числа исправных экземпляров. Выделенный для зоны IP-адрес присваивается новому лидеру. Таким образом обеспечивается высокая доступность фронтальной машины.

Для работы Raft должны быть соблюдены следующие требования:

1) настроено нечетное количество (рекомендуется 3 или 5) экземпляров фронтальной машины (см. 2.2), при этом на всех экземплярах нужно указать одинаковое имя учетной записи администратора ПК СВ;

2) ни на одном из экземпляров не развернут сервис `apache2` в режиме «AstraMode off»;

3) выделен IP для настройки плавающего IP-адреса кластера;

4) настроен беспарольный доступ для пользователя `root` между всеми экземплярами фронтальной машины;

5) настроено общее облачное хранилище для образов дисков и файлов.

Чтобы настроить беспарольный доступ для пользователя `root` необходимо выполнить следующие действия:

1) на каждом экземпляре фронтальной машины создать ssh-ключ от имени пользователя `root` командой `sudo ssh-keygen`. Для всех параметров оставлять значения по умолчанию (сразу нажимать клавишу **<Enter>**);

2) на каждом экземпляре фронтальной машины выполнить обмен ключами командами:

```
KEY=$(sudo cat /root/.ssh/id_rsa.pub)
ssh <local-admin>@<front-1-hostname> \
"sudo bash -c \"echo $KEY >> /root/.ssh/authorized_keys\""
ssh <local-admin>@<front-2-hostname> \
"sudo bash -c \"echo $KEY >> /root/.ssh/authorized_keys\""
...
```

```
ssh <local-admin>@<front-N-hostname> \
"sudo bash -c \"echo $KEY >> /root/.ssh/authorized_keys\""
```

где <front-N-hostname> — имя (hostname) N-го экземпляра фронтальной машины. Допускается вместо имен указывать IP-адреса;

<local-admin> — имя локального администратора сервера, заданное при установке ОС;

а) при появлении приглашения для ввода вида:

```
Are you sure you want to continue connecting (yes/no)?
```

ввести yes и нажать клавишу **<Enter>**,

б) ввести пароль локального администратора сервера, заданный при установке ОС;

3) проверить обмен ключами, для этого:

а) в терминале первого экземпляра фронтальной машины выполнить вход по ssh на другой экземпляр командой:

```
sudo ssh <front-N-hostname>
```

б) выполнить вход по ssh на первый экземпляр фронтальной машины командой:

```
sudo ssh <front-1-hostname>
```

где <front-N-hostname> — имя (hostname) N-го экземпляра фронтальной машины. Допускается вместо имен указывать IP-адреса.

Настройка считается успешно завершенной, если после выполнения команды был осуществлен вход без пароля;

в) последовательно закрыть сессии ssh командами:

```
exit
exit
```

4) аналогичным образом проверить беспарольный доступ на остальных экземплярах фронтальной машины.

2.6.2. Настройка зоны объединения экземпляров фронтальных машин

Для автоматической настройки зоны, объединяющей несколько экземпляров фронтальной машины, можно воспользоваться скриптом brestcloud-raft-configure, который запускается на одной из фронтальных машин от имени администратора командой:

```
sudo brestcloud-raft-configure
```

В процессе работы мастера настройки объединения экземпляров фронтальных машин необходимо:

1) указать количество экземпляров фронтальных машин;

2) указать сетевой интерфейс, на который будет назначен плавающий IP-адрес (обычно указывается интерфейс, на котором настроен статический IP-адрес);

3) указать плавающий IP-адрес;

4) последовательно указать полные доменные имена экземпляров фронтальных машин;

5) указать короткое плавающее имя (эта A-запись будет зафиксирована в DNS FreeIPA).

После завершения работы мастера настройки объединения экземпляров фронтальных машин необходимо выполнить настройку браузера Mozilla Firefox для подключения к веб-интерфейсу ПК СВ в соответствии с 2.3.

2.6.3. Вывод из зоны объединения экземпляра фронтальной машины

Для вывода из зоны объединения экземпляра фронтальной машины (ноды) необходимо на лидере выполнить команду:

```
sudo onezone server-del <идентификатор_зоны> <идентификатор_удаляемого_экземпляра>
```

2.6.4. Ввод экземпляра фронтальной машины в зону объединения

Для ввода экземпляра фронтальной машины в зону объединения необходимо выполнить следующие действия:

1) на новом сервере установить и настроить сервис фронтальной машины (см. 2.2), при этом нужно указать существующее имя учетной записи администратора ПК СВ, проверить, что сервер находится в одиночном режиме командой:

```
sudo onezone show 0
```

Пример вывода после выполнения команды:

```
*ZONE 0 INFORMATION *
ID : 0
NAME : OpenNebula

ZONE TEMPLATE
ENDPOINT="http://localhost:2633/RPC2"
```

2) на новом сервере настроить беспарольный доступ для пользователя `root` на все и со всех экземпляров фронтальной машины;

3) на лидере сделать бекап базы и скопировать на новый сервер командами:

```
sudo onedb backup /tmp/db.backup -f -t postgresql -S localhost \
-u oneadmin -p "<пароль_БД_лидера>" -d opennebula
sudo scp /tmp/db.backup <new-front-hostname>:/tmp
```

где <пароль_БД_лидера> — взят из файла `/etc/one/one.d/db.conf`,

<new-front-hostname> — имя (hostname) нового сервера. Допускается вместо имен указывать IP-адреса;

4) на новом сервере остановить службу Брест и восстановить БД командами:

```
sudo systemctl stop opennebula
sudo onedb restore -f /tmp/db.backup -t postgresql -S localhost \
```

```
-u oneadmin -p "<пароль_БД_нового_сервера>" -d opennebula
```

где <пароль_БД_нового_сервера> — взят из файла /etc/one/one.d/db.conf;

5) на лидере скопировать директорию .one на новый сервер командой:

```
sudo scp -r /var/lib/one/.one/ <new-front-hostname>:/var/lib/one/
```

где <new-front-hostname> — имя (hostname) нового сервера. Допускается вместо имен указывать IP-адреса;

6) на лидере добавить новый сервер командой:

```
sudo onezone server-add <идентификатор_зоны> \
--name <полное_доменное_имя_нового_сервера> --rpc \
http://<полное_доменное_имя_нового_сервера>:2633/RPC2
```

7) с лидера скопировать файл конфигурации raft на новый сервер командой:

```
sudo scp /etc/one/one.d/raft.conf <new-front-hostname>:/etc/one/one.d/
```

где <new-front-hostname> — имя (hostname) нового сервера. Допускается вместо имен указывать IP-адреса;

8) на новом сервере скорректировать параметр FEDERATION[SERVER_ID] в файле /etc/one/one.d/raft.conf, указав <идентификатор_нового_сервера>. Значение параметра <идентификатор_нового_сервера> (ID) можно получить, выполнив команду sudo onezone show 0;

9) на новом сервере запустить службу Брест и перезапустить сервис веб-интерфейса командами:

```
sudo systemctl start opennebula
sudo systemctl restart opennebula-sunstone
```

10) с лидера скопировать сертификаты короткого плавающего имени командой:

```
sudo scp /etc/one/ssl/<короткое_плавающее_имя>.* \
<new-front-hostname>:/etc/one/ssl/
```

где <new-front-hostname> — имя (hostname) нового сервера. Допускается вместо имен указывать IP-адреса;

11) с лидера скопировать файлы конфигурации apache на новый сервер командами:

```
sudo scp /etc/apache2/sites-available/ipa-one-apache2-float.conf \
<new-front-hostname>:/etc/apache2/sites-available/
sudo scp /etc/apache2/apache2.<короткое_плавающее_имя>.keytab \
<new-front-hostname>:/etc/apache2/
```

где <new-front-hostname> — имя (hostname) нового сервера. Допускается вместо имен указывать IP-адреса;

12) на новом сервере применить файлы конфигурации apache командами:

```
sudo ktutil << EOF
rkt /etc/apache2/apache2.<короткое плавающее имя>.keytab
wkt /etc/apache2/apache2.keytab
q
EOF
sudo a2ensite ipa-one-apache2-float.conf
sudo systemctl restart apache2
```

13) проверить корректность новой конфигурации командой:

```
sudo onezone show 0
```

2.7. Особенности установки и инициализации сервисного режима работы ПК СВ

Для установки и инициализации сервисного режима работы ПК СВ необходимо выполнить следующие действия:

1) установить сервис фронтальной машины, для этого:

а) на сервере с ролью фронтальной машины установить пакет `brestcloud-base` командой:

```
sudo apt install brestcloud-base
```

В открывшемся окне **ЛИЦЕНЗИОННОЕ СОГЛАШЕНИЕ** нажать кнопку **[Принять]**;

Примечание. Во время установки пакета `brestcloud-base` автоматически будет выполнена инициализация сервисов фронтальной машины и узла виртуализации.

б) на сервере с ролью фронтальной машины назначить пароль локальному пользователю `brestadmin` (пользователь создается автоматически) командой:

```
sudo passwd brestadmin
```

в) перезагрузить сервер с ролью фронтальной машины;

2) установить сервис узла виртуализации на необходимое количество серверов, для этого:

а) на сервере с ролью узла виртуализации установить пакет `opennebula-node-kvm` командой:

```
sudo apt install opennebula-node-kvm
```

В открывшемся окне **ЛИЦЕНЗИОННОЕ СОГЛАШЕНИЕ** нажать кнопку **[Принять]**.

б) перезагрузить сервер с ролью узла виртуализации;

Примечание. На сервере с ролью фронтальной машины сервис узла виртуализации устанавливается и инициируется автоматически при установке пакета

`brestcloud-base`.

3) настроить подключение к веб-интерфейсу ПК СВ в браузере Mozilla Firefox на сервере с ролью фронтальной машины (или на любой другой машине, имеющей сетевой доступ к этому серверу), для этого:

а) установить браузер Mozilla Firefox (если при установке ОС СН не был выбран пункт **Средства работы в сети**) командой:

```
sudo apt install firefox
```

б) запустить браузер, например, с использованием графического интерфейса: «Пуск — Сеть — Веб-браузер Firefox»;

в) добавить в исключение самоподписной ssl сертификат, для этого:

- перейти по адресу: `https://<полное_доменное_имя>:2616`,

  где `<полное_доменное_имя>` — полное доменное имя сервера, на котором развернут сервис фронтальной машины.

- на открывшейся странице с предупреждением нажать на кнопку **[Дополнительно]**, а затем — на кнопку **[Принять риск и продолжить]**,
- на открывшейся странице **Open Nebula** вводить ничего не нужно;

г) аналогичным образом добавить в исключение самоподписной ssl сертифкат для портов 443 и 29876 (используется для подключения к удаленному рабочему столу ВМ). Открывшуюся страницу с сообщением об ошибке можно закрыть;

д) перейти к веб-интерфейсу ПК СВ по адресу:

`https://<полное_доменное_имя>`;

е) на открывшейся странице с предупреждением нажать на кнопку **[Дополнительно]**, а затем — на кнопку **[Принять риск и продолжить]** (дважды);

ж) на открывшейся странице **Брест**:

- в поле **Логин** ввести `brestadmin`,
- в поле **Пароль** ввести пароль локального пользователя `brestadmin`, который был задан во время выполнения действий по установке и инициализации сервиса фронтальной машины,
- нажать на кнопку **[Войти]**;

4) зарегистрировать узел виртуализации в веб-интерфейсе ПК СВ, для этого:

а) на сервере с ролью фронтальной машины в файле `/etc/hosts` добавить строку вида:

```
<IP-адрес_узла_виртуализации> <имя_узла_виртуализации>
```

где <IP-адрес_узла_виртуализации> — IP-адрес сервера с ролью узла виртуализации,

<имя_узла_виртуализации> — сетевое имя сервера с ролью узла виртуализации;

б) перейти к веб-интерфейсу ПК СВ по адресу:

https://<полное_доменное_имя>;

в) в веб-интерфейсе ПК СВ в меню слева выбрать пункт меню «Инфраструктура — Узлы» и на открывшейся странице **Узлы** нажать на кнопку **[+]**;

г) на открывшейся странице **Создать узел**:

- в поле **Имя хоста** указать сетевое имя сервера с ролью узла виртуализации,
- в поле **Логин администратора** ввести имя локального администратора сервера с ролью узла виртуализации,
- в поле **Пароль администратора** ввести пароль локального администратора сервера с ролью узла виртуализации,
- нажать на кнопку **[Создать]**;

д) на открывшейся странице **Узлы** появится запись о зарегистрированном узле виртуализации. Необходимо дождаться пока в столбце **Статус** для этого узла виртуализации значение Инициализация не изменится на ВКЛ. Для обновления значения статуса можно воспользоваться кнопкой **[Обновить]**.

3. НАСТРОЙКА ОБЛАЧНОГО ХРАНИЛИЩА

3.1. Общие сведения

В ПК СВ для развертывания ВМ используется два типа облачного хранилища данных:

- хранилище образов (Images Datastore) — предназначено для хранения всех зарегистрированных образов, которые могут использоваться для создания ВМ или хранения пользовательских данных;
- системное хранилище (System Datastore) — используется для хранения дисков виртуальных машин, работающих в текущий момент.

Кроме того, отдельно выделяют хранилище файлов и ядер (Files & Kernels Datastore), которое используется для хранения обычных файлов. Такими файлами могут быть ядра виртуальных машин (kernels), временные диски (ramdisks) или контекстные файлы. Например, в хранилище файлов и ядер можно поместить определенный инит-скрипт и указать его в контекстуализации для ВМ. Тогда при загрузке ОС этой ВМ будет выполняться указанный инит-скрипт. Для передачи файлов из хранилища файлов в runtime-директорию ВМ используется метод передачи ssh (устанавливается по умолчанию). Процесс настройки хранилища файлов и ядер описан в 3.8.

На рис. 2 представлена упрощенная схема взаимодействия облачных хранилищ данных.

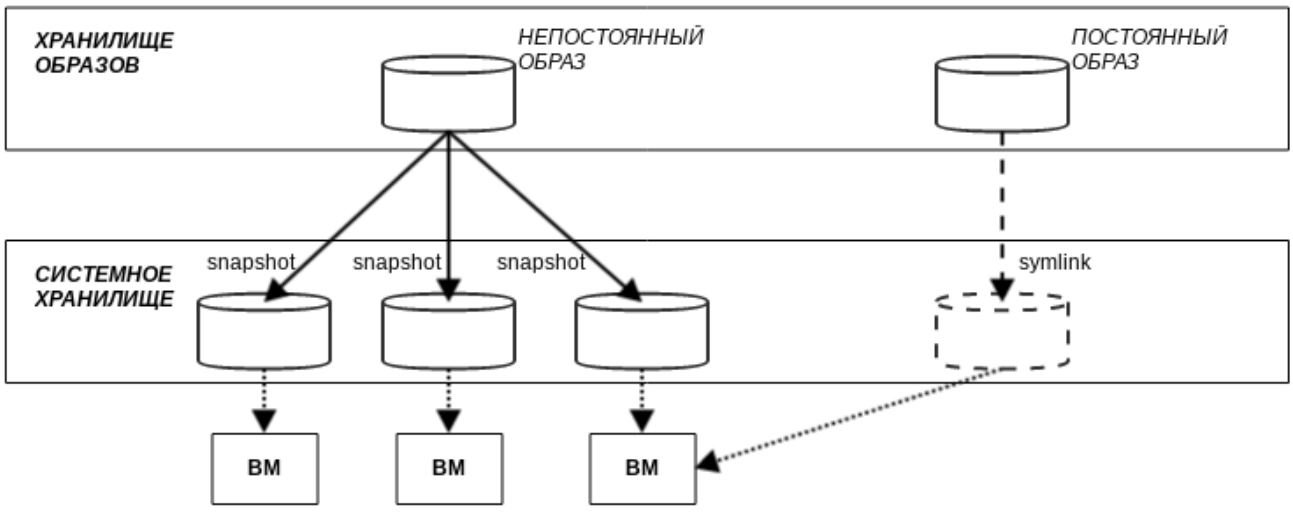

Рис. 2

В зависимости от назначения выделяют два типа образов:

- непостоянный образ, предназначен для создания системного диска ВМ. При создании ВМ такой образ, условно говоря, «копируется» из хранилища образов в системное хранилище. После удаления ВМ «копия» образа в системном хранилище также удаляется;
- постоянный образ, предназначен для хранения пользовательских данных (БД, файловый репозиторий и т.д.). При создании ВМ такой образ, условно говоря, «перемещается» из хранилища образов в системное хранилище. После удаления ВМ постоянный образ «перемещается» обратно в хранилище образов. Таким образом, изменения, внесенные во время работы ВМ, будут сохранены.

Образы дисков передаются («копируются» и «перемещаются») между хранилищем образов и системным хранилищем с помощью драйверов программного обеспечения (ПО) Transfer Manager (TM). Эти драйверы представляют собой специальные элементы ПО, которые выполняют низкоуровневые операции хранения.

Для построения облачного хранилища данных используются следующие базовые технологии хранения:

- Filesystem — файловая технология хранения (см. 3.3);

ВНИМАНИЕ! Сетевая файловая система (Network File System — NFS) не поддерживает файловые атрибуты безопасности, поэтому использование данной ФС при построении облачного хранилища, функционирующего в мандатном контексте, недопустимо.

- LVM — блочная технология хранения с использованием LVM (logical volume manager — менеджер логических томов) — см. 3.4;
- Ceph — программно-определяемая технология хранения Ceph (см. 3.5);
- Raw Device Mapping — прямое подключение к ВМ существующих блочных устройств, используется только для организации хранилища образов (см. 3.6);
- iSCSI-Libvirt — прямое подключение к ВМ существующих устройств iSCSI, используется только для организации хранилища образов (см. 3.7).

В таблице 1 приведено описание доступных методов передачи данных (драйверов) для используемых базовых технологий хранения.

Таблица 1

Технологии хранения	Методы передачи данных между хранилищем образов и системным хранилищем
Filesystem	ssh — образы копируются с помощью ssh-протокола; shared — образы экспортируются в соответствующий каталог системного хранилища на узле виртуализации; qcow2 — аналогично shared, но для образов формата `qcow2`. Образы создаются и передаются с помощью команды `qemu-img` с использованием оригинального образа в качестве опорного файла
Ceph	ceph — все образы экспортируются в Ceph-пулы; ssh — rbd-файл, ассоциируемый с образом, экспортируется в файл локальной файловой системы узла виртуализации
LVM	fs_lvm — образы хранятся как обычные файлы, при создании ВМ они выгружаются в логические тома (LV); lvm_lvm — создаются отдельные группы LVM-томов для хранилища образов и системного хранилища; lvm_thin — создаются отдельные группы LVM-томов для хранилища образов и системного хранилища, но системное хранилище организуется индивидуально для каждого узла виртуализации
Raw Devices	dev — образы представляют собой существующие блочные устройства в узлах
iSCSI libvirt	iscsi — образы представляют собой компоненты iSCSI target

По умолчанию после инициализации сервисов ПК СВ (см. 2) облачные хранилища настроены на использование локальной файловой системы (каталоги `/var/lib/one/datastores/<идентификатор_хранилища>`). При этом в качестве метода передачи данных между хранилищем образов и системным хранилищем установлен `ssh`.

Идентификаторы и наименования облачных хранилищ, созданных по умолчанию во время инициализации сервисов ПК СВ, приведены в таблице 2.

Таблица 2

Идентификатор	Наименование	Описание
0	`system`	системное хранилище
1	`default`	хранилище образов
2	`files`	хранилище файлов и ядер

Примечание. Стандартный путь для хранилищ `/var/lib/one/datastores` можно изменить в конфигурационном файле `/etc/one/oned.conf` через параметр настройки `DATASTORE_LOCATION` (см. 5.3).

3.2. Создание облачных хранилищ

Для создания облачного хранилища необходимо выполнить следующую последовательность действий:

1) подготовить систему хранения данных в соответствии с выбранной технологией хранения;

2) в ПК СВ создать логическую сущность хранилища (зарегистрировать), указав его имя, тип и метод передачи данных. После создания логической сущности хранилища будет создан каталог с идентификатором хранилища (по умолчанию `/var/lib/one/datastores/<идентификатор_хранилища>`);

3) на фронтальной машине и узлах виртуализации смонтировать подготовленную систему хранения данных в каталог хранилища.

ВНИМАНИЕ! При использовании файловой технологии хранения (например, кластерной файловой системы ocfs2), после добавления записи об автоматическом монтировании в файле `/etc/fstab` и перезагрузки ОС, необходимо назначить на каталог этого хранилища владельца `oneadmin`. В противном случае при перезагрузке ОС владелец меняется на `root` и использование хранилища будет не доступно.

Подробнее процесс создания облачных хранилищ, построенных на базе различных технологий хранения, описан в 3.3–3.7

3.3. Облачные хранилища на базе файловой технологии хранения

Файловая технология хранения позволяет хранить образы ВМ в виде файла.

Рекомендуется иметь несколько хранилищ, построенных на базе файловой технологии хранения и с применением различных методов передачи данных, для:

- распределения операций ввода-вывода между серверами хранения данных;
- обеспечение непрерывности обслуживания.

3.3.1. Схема облачного хранилища

Образы сохраняются в соответствующий каталог хранилища (по умолчанию `/var/lib/one/datastores/<идентификатор_хранилища>`). Для каждой рабочей ВМ создается каталог с названием по идентификационному номеру ВМ в соответствующем системном хранилище. В данных каталогах содержатся диски ВМ и дополнительные файлы, например, файлы контрольных точек или файлы снимков.

3.3.1.1. Методы передачи Shared и Qcow2

Метод совместной передачи (shared transfer driver) предполагает, что на всех узлах виртуализации установлена и настроена распределенная файловая система, например, NFS.

Примечание. Особенности настройки NFS представлены в 3.3.3.4.

Все файловые операции (`ln, cp` и т.д.) выполняются на узле виртуализации (см. рис. 3).

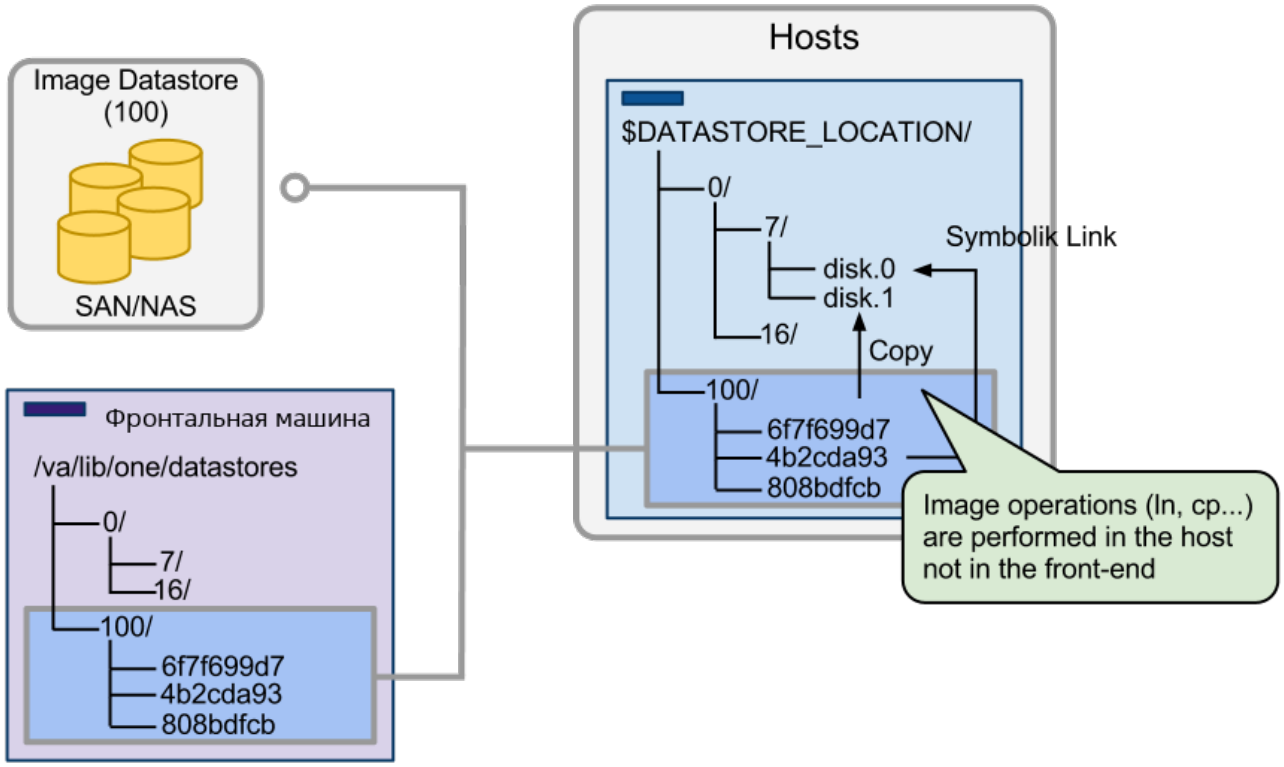

Рис. 3

Данный метод передачи сокращает время развертывания ВМ и обеспечивает возможность динамического перемещения. Однако возможно снижение производительности виртуальных машин, если службы виртуализации оказывают интенсивные нагрузки на диск. Это ограничение можно преодолеть путем:

- использования серверов с различными файловыми системами для хранилищ образов с распределением фактической пропускной способности подсистемы ввода-вывода;
- использования дополнительного системного хранилища, настроенного на применение метода передачи ssh, при котором образы копируются локально на каждый узел;
- дополнительной настройки или улучшения серверов файловых систем.

3.3.1.2. Метод передачи SSH

Метод передачи ssh использует локальную файловую систему узлов для размещения образов работающих виртуальных машин. Таким образом все файловые операции выполняются локально, но образы всегда необходимо копировать на узлы. Данный драйвер также не допускает использование динамических перемещений между узлами (см. рис. 4).

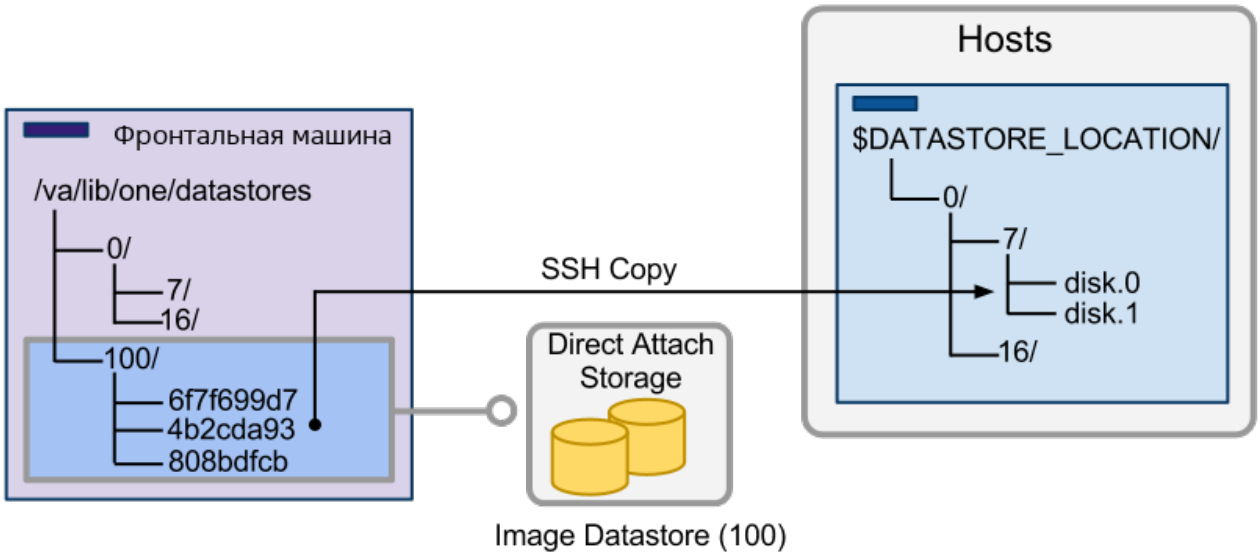

Рис. 4

3.3.2. Общие настройки ПК СВ

После того как будет настроена система хранения данных, построенная на базе файловой технологии хранения, настройка ПК СВ выполняется в два этапа:

- регистрация системного хранилища;
- регистрация хранилище образов.

3.3.2.1. Регистрация системного хранилища

При регистрации нового системного хранилища необходимо указать его имя, тип и метод передачи данных в соответствии с таблицей 3.

Таблица 3

Параметр	Значение
`NAME`	Имя хранилища
`TYPE`	`SYSTEM_DS`
`TM_MAD`	`shared` — для режима совместной передачи; `qcow2` — для режима передачи qcow2; `ssh` — для режима передачи ssh

Пример

Регистрация системного хранилища, в котором используется драйвер совместной передачи:

1) создать файл `systemds.txt` со следующим содержанием:

```
NAME = nfs_system
TYPE = SYSTEM_DS
TM_MAD = shared
```

2) выполнить команду:

```
onedatastore create systemds.txt
```

Пример вывода после выполнения команды:

```
ID: 100
```

3.3.2.2. Регистрация хранилища образов

При регистрации нового хранилища образов необходимо указать его имя, тип (IMAGE_DS), базовую технологию хранения (fs) и метод передачи данных в соответствии с таблицей 4.

Таблица 4

Параметр	Значение
NAME	Имя хранилища
TYPE	IMAGE_DS
DS_MAD	fs
TM_MAD	shared — для режима совместной передачи; qcow2 — для режима передачи qcow2; ssh — для режима передачи ssh

Пример

Регистрация хранилища образов, в котором используется драйвер совместной передачи:

1) создать файл imageds.txt со следующим содержанием:

```
NAME = nfs_images
TYPE = IMAGE_DS
DS_MAD = fs
TM_MAD = shared
```

2) выполнить команду:

```
onedatastore create imageds.txt
```

Пример вывода после выполнения команды:

```
ID: 101
```

ВНИМАНИЕ! Необходимо использовать одинаковый метод передачи данных (параметр TM_MAD) для системного хранилища и для хранилища образов.

3.3.3. Настройка фронтальной машины и узла виртуализации

3.3.3.1. Методы передачи Shared и Qcow2

Смонтировать подготовленную систему хранения данных в каталог хранилища (по умолчанию /var/lib/one/datastores/<идентификатор_хранилища>). Если все хранилища одного типа, можно смонтировать весь каталог /var/lib/one/datastores.

ВНИМАНИЕ! Для фронтальной машины необходимо смонтировать только хранилища образов.

3.3.3.2. Особенности метода передачи qcow2

Метод передачи qcow2 является разновидностью метода совместной передачи, ориентированным на работу с образами дисков формата `qcow2`. Образы создаются и передаются с помощью команды `qemu-img` с использованием оригинального образа в качестве опорного файла. Стандартные параметры команды `qemu-img` можно скорректировать, указав необходимые значения в конфигурационном файле `/etc/one/tmrc` (переменная `QCOW2_OPTIONS`).

3.3.3.3. Метод передачи SSH

Смонтировать подготовленный дисковый ресурс (локальное дисковое устройство или дисковый ресурс SAN/NAS-сервера) в каталог хранилища (по умолчанию `/var/lib/one/datastores/<идентификатор_хранилища>`). Кроме того, необходимо убедиться в том, что на смонтированном дисковом ресурсе достаточно места для хранения образов и дисков виртуальных машин, которые находятся в состоянии «остановлена» и «не размещена».

ВНИМАНИЕ! Необходимо убедиться в том, что все узлы, включая фронтальную машину, могут осуществлять ssh-передачу на любой другой узел, включая самих себя. В противном случае перемещения не будут выполняться.

3.3.3.4. Особенности использования NFS

ВНИМАНИЕ! Сетевая файловая система (Network File System — NFS) не поддерживает файловые атрибуты безопасности, поэтому использование данной ФС при построении облачного хранилища, функционирующего в мандатном контексте, недопустимо.

В сервисном режиме функционирования ПК СВ при использовании NFS на каждом узле виртуализации следует установить уровень целостности, назначаемый по умолчанию для ВМ, равным 0. Для этого необходимо выполнить следующие действия:

1) остановить сервис `libvirtd` командой:

```
sudo systemctl stop libvirtd.service
```

2) в конфигурационном файле `/etc/libvirt/libvirtd.conf`, установить значение параметра `ilev_vm` равное 0:

```
ilev_vm = 0
```

3) запустить сервис `libvirtd` командой:

```
sudo systemctl start libvirtd.service
```

Примечание. Для монтирования NFS-томов рекомендуются установить следующие параметры: `soft, intr, rsize=32768, wsize=32768, no_root_squash`.

3.4. Облачные хранилища LVM

Блочная технология хранения с использованием LVM обеспечивает возможность использования LVM-томов вместо обычных файлов для хранения образов. При использовании

данного типа хранилища отсутствует необходимость в организации файловой системы.

Примечание. Для хранилища LVM не требуется настройка кластерного управления логическими томами (CLVM) в кластере. Драйверы обновляют метаданные LVM каждый раз, когда образ требуется в другом узле.

3.4.1. Драйвер хранилища FS_LVM

3.4.1.1. Общие сведения

Образы хранятся как обычные файлы, по умолчанию установлен следующий путь размещения в хранилище образов: `/var/lib/one/datastores/<идентификатор_образа>`, но при создании ВМ они выгружаются в логические тома (LV). Виртуальные машины запускаются из LV на узле (см. рис. 5).

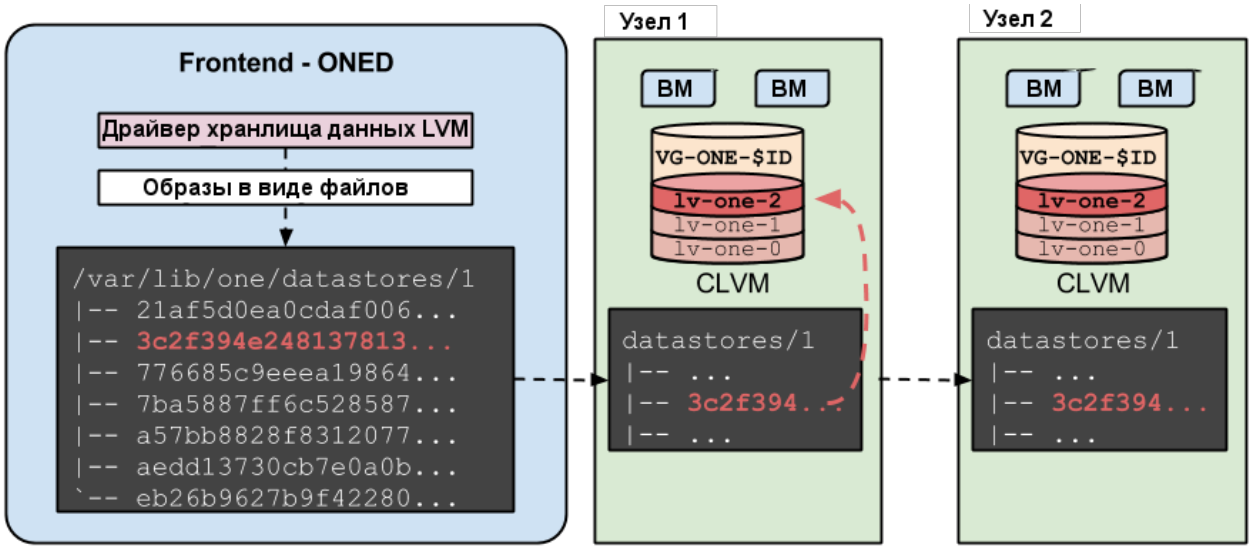

Рис. 5

Драйвер хранилища FS_LVM рекомендуется применять при наличии СХД высшего класса (high-end). В этом случае LUN можно экспортировать на все узлы.

ВНИМАНИЕ! Для хранения образов в виде файлов используется распределенная файловая система, такая как, например, NFS. При этом необходимо выполнить настройку каталогов облачных хранилищ и точек монтирования так же, как и при использовании метода совместной передачи (shared) в облачном хранилище, построенном на базе файловой технологии хранения (см. 3.3). Рекомендуется сначала выполнить развертывание системного хранилища, построенного на базе файловой технологии хранения. А затем, убедившись в его корректной работе, заменить его на системное хранилище LVM.

Примечание. Особенности настройки NFS представлены в 3.3.3.4.

3.4.1.2. Общие настройки ПК СВ

Настройка ПК СВ выполняется в два этапа:

- регистрация системного хранилища;
- регистрация хранилища образов.

При регистрации нового системного хранилища необходимо задать значения параметров в соответствии с таблицей 5

Таблица 5

Параметр	Значение
NAME	Имя хранилища
TYPE	SYSTEM_DS
TM_MAD	fs_lvm
BRIDGE_LIST	Список узлов, разделенных пробелами, через которые осуществляется доступ к системе хранения данных (SAN). Не используется, если фронтальная машина имеет прямой доступ к системе хранения данных.

Пример

Создание хранилища LVM с использованием конфигурационного файла:

1) создать файл `systemds.txt` со следующим содержанием:

```
NAME = lvm_system
TYPE = SYSTEM_DS
TM_MAD = fs_lvm
BRIDGE_LIST = "NODE1 NODE2"
```

2) выполнить команду:

```
onedatastore create systemds.txt
```

Пример вывода после выполнения команды:

```
ID: 100
```

При регистрации нового хранилища образов необходимо задать значения параметров в соответствии с таблицей 8

Таблица 6

Параметр	Значение
NAME	Имя хранилища
TYPE	IMAGE_DS
DS_MAD	fs
TM_MAD	fs_lvm
DISK_TYPE	BLOCK
SAFE_DIRS	Необязательный параметр — перечень каталогов, разделенных символом пробела, в которые разрешается размещать образы. По умолчанию размещать образы запрещено во все подкаталоги корневого каталога «/».

Окончание таблицы 6

Параметр	Значение
BRIDGE_LIST	Список узлов, разделенных пробелами, через которые осуществляется доступ к системе хранения данных (SAN). Не используется, если фронтальная машина имеет прямой доступ к системе хранения данных.

Пример

Создание хранилища LVM с использованием конфигурационного файла:

1) создать файл imageds.txt со следующим содержанием:

```
NAME = lvm_image
TYPE = IMAGE_DS
DS_MAD = fs
TM_MAD = fs_lvm
DISK_TYPE = "BLOCK"
SAFE_DIRS="/var/tmp /tmp"
BRIDGE_LIST = "NODE1 NODE2"
```

2) выполнить команду:

```
onedatastore create imageds.txt
```

Пример вывода после выполнения команды:

```
ID: 101
```

3.4.1.3. Настройка фронтальной машины

Дополнительная настройка фронтальной машины не требуется.

3.4.1.4. Настройка узла виртуализации

Узлы должны отвечать следующим требованиям:

- на узлах должен быть установлен пакет lvm2;
- все узлы должны иметь доступ к одним и тем же LUN;
- на одном узле должна быть создана группа томов совместно используемых LUN для каждого хранилища с именем вида:

vg-one-<идентификатор_системного_хранилища>;

Пример

В системе две виртуальные машины (9 и 10) используют образ с идентификатором «0», размещенный в хранилище LVM. На узлах выполнена настройка совместно используемого LUN и создана группа томов с именем vg-one-0. Хранилище будет иметь следующую схему (пример вывода после выполнения команды lvs):

```
LV          VG        Attr        LSize  Pool  Origin  Data%  Meta%  Move
lv-one-10-0 vg-one-0  -wi-------  2.20g
lv-one-9-0  vg-one-0  -wi-------  2.20g
```

3.4.2. Драйвер хранилища LVM_LVM

3.4.2.1. Общие сведения

Драйвер позволяет организовать хранилище образов и системное хранилище в LVM. При использовании драйвера хранилища LVM_LVM необходимо наличие на всех узлах кластера общих блочных устройств хранения данных. При этом, в отличие от использования драйвера FS_LVM, нет необходимости создавать общую сетевую файловую систему для образов.

Особенности функционирования драйвера хранилища LVM_LVM:

- предварительно необходимо создать отдельные группы LVM-томов для хранилища образов и системного хранилища;
- при загрузке образа диска ВМ в хранилище образов автоматически создается LVM-том, в который записывается загружаемый образ в формате `raw`;
- при развертывании ВМ в системном хранилище автоматически создается копия LVM-тома из хранилища образов.

ВНИМАНИЕ! В драйвере хранилища LVM_LVM не поддерживается создание снапшотов диска и создание «тонких» дисков.

3.4.2.2. Общие настройки ПК СВ

Настройка ПК СВ выполняется в два этапа:

- регистрация системного хранилища;
- регистрация хранилища образов.

Системные хранилища должны создаваться со значениями, приведенными в таблице 7

Таблица 7

Параметр	Значение
NAME	Имя хранилища
TYPE	SYSTEM_DS
TM_MAD	lvm_lvm
DRIVER	raw

Пример

Регистрация системного хранилища с использованием конфигурационного файла.

```
cat > system-ds.conf <<EOT
NAME="lvm-lvm-system"
TYPE="SYSTEM_DS"
TM_MAD="lvm_lvm"
DRIVER="raw"
```

```
EOT
onedatastore create system-ds.conf
```

Пример вывода после выполнения команды:

```
ID: 100
```

Хранилища образов должны создаваться со значениями, приведенными в таблице 8

Таблица 8

Параметр	Значение
NAME	Имя хранилища
TYPE	IMAGE_DS
DS_MAD	lvm
TM_MAD	lvm_lvm
DISK_TYPE	BLOCK
DRIVER	raw

Пример

Создание хранилища с использованием конфигурационного файла.

```
cat > images-ds.conf <<EOT
NAME="lvm-images"
TYPE="IMAGE_DS"
DISK_TYPE="BLOCK"
DS_MAD="lvm"
TM_MAD="lvm_lvm"
DRIVER="raw"
EOT
onedatastore create images-ds.conf
```

Пример вывода после выполнения команды:

```
ID: 101
```

3.4.2.3. Настройка фронтальной машины

На фронтальной машине необходимо создать отдельные группы LVM-томов для хранилища образов и системного хранилища.

Пример

Создание групп LVM-томов для хранилища образов и системного хранилища на блочных устройствах (физических дисках) /dev/sdc и /dev/sdb:

```
pvcreate /dev/sdc
pvcreate /dev/sdb
vgcreate vg-one-<идентификатор_хранилища_образов> /dev/sdc
```

```
vgcreate vg-one-<идентификатор_системного_хранилища> /dev/sdb
```

3.4.2.4. Настройка узла виртуализации

Дополнительных действий по настройке не требуется.

3.4.3. Драйвер хранилища LVM_THIN

3.4.3.1. Общие сведения

Драйвер позволяет организовать хранилище образов и системное хранилище в LVM. При использовании драйвера хранилища LVM_THIN, в отличие от драйвера LVM_LVM, системное хранилище организуется индивидуально — для этого необходимо указать отдельное блочное устройство, с которым будет взаимодействовать узел. Это может быть локальное блочное устройство узла виртуализации или выделенное (для каждого узла) блочное устройство системы хранения данных.

Особенности функционирования драйвера хранилища LVM_THIN:

- предварительно необходимо создать отдельные группы LVM-томов для хранилища образов и системного хранилища;
- при загрузке образа диска ВМ в хранилище образов автоматически создается LVM-том, в который пишется загружаемый образ в формате `raw`;
- при развертывании ВМ в системном хранилище из образа диска автоматически создается тонкий LVM-том в формате `qcow2`.

ВНИМАНИЕ! В драйвере хранилища LVM_THIN не поддерживается миграция ВМ.

3.4.3.2. Общие настройки ПК СВ

Настройка ПК СВ выполняется в два этапа:

- регистрация системного хранилища;
- регистрация хранилища образов.

Примеры:

1. Регистрация хранилища образов с использованием конфигурационного файла:

```
cat > images-ds.conf <<EOT
NAME="lvm-thin-images"
TYPE="IMAGE_DS"
DISK_TYPE="BLOCK"
DS_MAD="lvm_thin"
TM_MAD="lvm_thin"
DRIVER="qcow2"
EOT
onedatastore create images-ds.conf
```

2. Регистрация системного хранилища с использованием конфигурационного файла:

```
cat > system-ds.conf <<EOT
```

```
NAME="lvm-thin-system"
TYPE="SYSTEM_DS"
TM_MAD="lvm_thin"
EOT
onedatastore create system-ds.conf
```

3.4.3.3. Настройка фронтальной машины

Дополнительных действий по настройке не требуется.

3.4.3.4. Настройка узла виртуализации

На каждом узле виртуализации необходимо создать отдельную группу LVM-томов для хранилища образов и системного хранилища.

Примеры:

1. Создание группы LVM-томов для хранилища образов на блочном устройстве (физическом диске) `/dev/sdc` каждого узла:

```
pvcreate /dev/sdc
vgcreate vg-one-<идентификатор_хранилища_образов> /dev/sdc
```

2. Создание группы LVM-томов для системного хранилища на локальном блочном устройстве (физическом диске) `/dev/sdb` каждого узла:

```
pvcreate /dev/sdb
vgcreate vg-one-<SYSTEM_DS_ID>-<HOSTNAME> /dev/sdb
lvcreate -T -L <SIZE>G vg-one-<SYSTEM_DS_ID>-<HOSTNAME>/onethinpool
```

где `<SYSTEM_DS_ID>` — идентификатор системного хранилища;

`<HOSTNAME>` — имя узла кластера;

`<SIZE>` — размер тонкого пула в ГБ, должен быть не больше размера локального блочного устройства (в представленном примере — `/dev/sdb`).

Примечания:

1. При создании тонкого LVM-тома дополнительно к исходному LVM-тому создаются два скрытых тома под метаданные (информация о выделенных блоках).
2. При исчерпании свободного места на скрытых томах, выделенных под метаданные, возникает потеря метаданных и файловых систем на тонких томах, что приводит к ошибке записи.
3. Поврежденные метаданные возможно восстановить с помощью команды `lvconvert --repair VG/ThinPoolName`.
4. Драйвер хранилища LVM_THIN поддерживает автоматическое увеличение размера томов, в том числе и скрытых томов, выделенных под метаданные. Для этого необходимо всегда оставлять свободное место в группе LVM-томов. Автоматическое увеличение размера томов настраивается в конфигурационном файле

/etc/lvm.conf, параметр: thin_pool_autoextend.

3.5. Облачные хранилища Ceph

Программно-определяемая технология хранения Ceph обеспечивает возможность использования блочных устройств Ceph для размещения образов и дисков виртуальных машин.

ВНИМАНИЕ! Для работы данного драйвера необходимо, чтобы узлы виртуализации, использующие технологию Ceph, являлись Ceph-клиентами работающего Ceph-кластера.

3.5.1. Общие сведения

Образы и диски виртуальных машин хранятся в одном Ceph-пуле. Каждый образ имеет в пуле имя вида: one-<идентификатор_образа>. Виртуальные машины будут использовать RBD-тома для своих дисков если образы помечены как постоянный, в противном случае создаются новые снапшоты образа с наименованием вида: one-<идентификатор_образа>-<идентификатор_ВМ>-<идентификатор_диска_ВМ>.

3.5.2. Настройка Ceph-кластера

Предварительно должен быть развернут Ceph-кластер. Порядок развертывания Ceph представлен в документе РУСБ.10015-01 95 01-1 «Операционная система специального назначения «Astra Linux Special Edition». Руководство администратора. Часть 1» из комплекта поставки ОС СН.

Дополнительно необходимо выполнить следующие настройки:

1) создать пул для облачных хранилищ, указав наименование пула one:

```
ceph osd pool create one 128
ceph osd lspools
```

Пример вывода после выполнения команды:

```
0 data,1 metadata,2 rbd,6 one,
```

2) определить Ceph-пользователя, который будет иметь доступ к пулу хранилищ. Данный пользователь будет также использоваться libvirt для доступа к образам дисков, например:

```
ceph auth get-or-create client.libvirt \
mon 'profile rbd' osd 'profile rbd pool=one'
```

Кроме того, необходимо получить копию ключа данного пользователя для ее дальнейшей передачи на узлы виртуализации, например:

```
ceph auth get-key client.libvirt | tee client.libvirt.key
ceph auth get client.libvirt -o ceph.client.libvirt.keyring
```

3) несмотря на то, что в ПК СВ поддерживается RBD-формат «1», настоятельно рекомендуется использовать RBD-формат «2», для этого в файле ceph.conf указать:

```
[global]
rbd_default_format = 2
```

4) выбрать группу клиентских узлов кластера для использования в качестве мостов хранилищ `storage bridges`. Эти узлы будут использоваться для импорта образов в Ceph-кластер из ПК СВ. В данных узлах должна быть установлена программа `qemu-img`.

3.5.3. Настройка фронтальной машины

Для фронтальной машины не требуется специальная настройка Ceph. Фронтальная машина будет выполнять доступ к Ceph-кластеру через мосты хранилищ.

3.5.4. Настройка узла виртуализации

Для использования Ceph-кластера необходимо выполнить следующую настройку узлов:

1) должны быть установлены клиентские инструментальные средства Ceph;

2) сервис `MON` должен быть определен в `ceph.conf` для всех узлов, поэтому значения `hostname` и `port` не требуется указывать в Ceph-командах;

3) от имени пользователя (например, `admin`), входящего в группы `astra-admin` и `astra-console`, скопировать набор ключей Ceph-пользователя `ceph.client.libvirt.keyring` на узлы в каталог `/etc/ceph` и ключ пользователя `client.libvirt.key` в каталог `/var/lib/one`:

```
scp ceph.client.libvirt.keyring admin@node:/tmp
scp client.libvirt.key admin@node:/tmp
ssh admin@node "sudo mv /tmp/ceph.client.libvirt.keyring /etc/ceph"
ssh admin@node "sudo mv /tmp/client.libvirt.key /var/lib/one"
ssh admin@node "sudo ln -s /var/lib/one/client.libvirt.key \
   /root/client.libvirt.key"
```

4) сгенерировать секретный ключ для Ceph-пользователя и скопировать его на узлы в каталог `/var/lib/one`. Зафиксировать универсальный уникальный идентификатор (UUID) для дальнейшего использования:

```
UUID='uuidgen'
cat > secret.xml <<EOF
<secret ephemeral='no' private='no'>
   <uuid>$UUID</uuid>
   <usage type='ceph'>
            <name>client.libvirt secret</name>
   </usage>
</secret> EOF
```

```
scp secret.xml admin@node:/tmp
ssh admin@node "sudo mv /tmp/secret.xml /var/lib/one"
ssh admin@node "sudo ln -s /var/lib/one/secret.xml /root/secret.xml"
```

5) установить секретный ключ libvirt и удалить файлы ключа на узлах:

```
ssh admin@node "sudo virsh -c qemu:///system secret-define secret.xml"
ssh admin@node "sudo virsh -c qemu:///system secret-set-value --secret \
     $UUID --base64 $(cat /root/client.libvirt.key)"
ssh admin@node "sudo rm /root/client.libvirt.key"
ssh admin@node "sudo rm /var/lib/one/client.libvirt.key"
```

6) убедиться в том, что Ceph-клиент имеет корректные настройки на узле:

```
ssh admin@node "sudo rbd ls -p one --id libvirt"
```

7) убедиться в том, что на узлах выделено достаточно места для хранения вспомогательных файлов виртуальных машин, таких как context-диски, файлы развертывания и файлы контрольной точки.

3.5.5. Общие настройки ПК СВ

Для использования Ceph-кластера в качестве облачного хранилища необходимо зарегистрировать системное хранилище и хранилище образов. Оба хранилища совместно используют одни и те же параметры конфигурации, приведенные в таблице 9, и Ceph-пул.

Примечание. Можно добавить дополнительные хранилища образов и системные хранилища, указав другие пулы с отличными от Ceph политиками распределения ресурсов/репликации.

Таблица 9

Параметр	Описание	Обязательный
NAME	Имя хранилища	ДА
POOL_NAME	Имя Ceph-пула	ДА
CEPH_USER	Имя Ceph-пользователя, используемое командами libvirt и rbd	ДА
CEPH_KEY	Полный путь файла секретного ключа для пользователя, если не используется стандартный файл (/var/lib/one/client.libvirt.key)	НЕТ
CEPH_CONF	Нестандартный конфигурационный файл Ceph, если необходим	НЕТ
RBD_FORMAT	По умолчанию будет использоваться RBD-формат «2»	НЕТ
BRIDGE_LIST	Разделенный пробелами список узлов виртуализации, используемых в качестве мостов хранилищ	ДА
CEPH_HOST	Разделенный пробелами список узлов виртуализации, с инициированным сервисом монитора (MON), например, NODE1:port1 NODE2:port2 host4:port4	ДА

Окончание таблицы 9

Параметр	Описание	Обязательный
CEPH_SECRET	Универсальный уникальный идентификатор (UUID) секретного ключа libvirt	ДА

3.5.5.1. Регистрация системного хранилища

При регистрации системного хранилища дополнительно к параметрам, приведенным в таблице 9, устанавливаются параметры, указанные в таблице 10.

Таблица 10

Параметр	Описание	Обязательный
TYPE	SYSTEM_DS	ДА
TM_MAD	ceph	ДА

Пример

Создание хранилища с использованием конфигурационного файла:

1) создать файл systemds.txt со следующим содержанием:

```
NAME = ceph_system
TYPE = SYSTEM_DS
TM_MAD = ceph

POOL_NAME = one
CEPH_USER = libvirt
BRIDGE_LIST = NODE1 NODE1
CEPH_HOST = NODE1:port1 NODE2:port2
CEPH_USER = libvirt
CEPH_SECRET = "6f88b54b-5dae-41fe-a43e-b2763f601cfc"
```

2) выполнить команду:

```
onedatastore create systemds.txt
```

Пример вывода после выполнения команды:

```
ID: 100
```

Примечание. Также Ceph может работать с системным хранилищем, построенном на базе файловой технологии хранения с использованием метода совместной передачи (shared) — см. 3.3. В этом случае энергозависимые диски и диски подкачки создаются в виде обычных файлов в системном хранилище. Кроме Ceph-кластера необходимо выполнить установку и настройку распределенной файловой системы, например, NFS.

Примечание. Особенности настройки NFS представлены в 3.3.3.4.

3.5.5.2. Регистрация хранилища образов

При регистрации хранилища образов дополнительно к параметрам, приведенным в таблице 9, устанавливаются параметры, указанные в таблице 11.

Таблица 11

Параметр	Описание	Обязательный
DS_MAD	ceph	ДА
TM_MAD	ceph	ДА
DISK_TYPE	RBD	ДА
STAGING_DIR	Путь по умолчанию для операций с образом в мостах	НЕТ

Пример

Создание хранилища с использованием конфигурационного файла:

1) создать файл imageds.txt со следующим содержанием:

```
NAME = "cephds"
DS_MAD = ceph
TM_MAD = ceph

DISK_TYPE = RBD

POOL_NAME = one
CEPH_HOST = NODE1:port1 NODE2:port2
CEPH_USER = libvirt
CEPH_SECRET = "6f88b54b-5dae-41fe-a43e-b2763f601cfc"

BRIDGE_LIST = NODE1 NODE1
```

2) выполнить команду:

```
onedatastore create imageds.txt
```

Пример вывода после выполнения команды:

```
ID: 101
```

3.5.6. Дополнительные параметры

В файле /var/lib/one/remotes/datastore/ceph/ceph.conf могут быть установлены значения по умолчанию для следующих параметров Ceph-драйвера:

- POOL_NAME — имя Ceph-пула;
- STAGING_DIR — путь для операций с образом;
- RBD_FORMAT — формат для RBD томов.

3.6. Хранилище образов Raw Device Mapping

Хранилище Raw Device Mapping (RDM) является хранилищем образов, обеспечивающим динамический доступ к блочным устройствам узла.

3.6.1. Общие сведения

Хранилище RDM предназначено для регистрации уже существующих блочных устройств узла. Устройства должны быть установлены и доступны, а виртуальные машины, использующие эти устройства, должны быть настроены для работы в подготовленных для них узлах. Дополнительные файлы виртуальных машин, такие как файлы развертывания или энергозависимые диски, создаются как обычные файлы.

3.6.2. Настройка фронтальной машины

Дополнительная настройка не требуется.

3.6.3. Настройка узла виртуализации

Дополнительная настройка не требуется.

3.6.4. Общие настройки ПК СВ

После установки хранилища настройка ПК СВ выполняется в два этапа:

- регистрация системного хранилища;
- регистрация хранилища образов.

3.6.4.1. Регистрация системного хранилища

Хранилище RDM может работать с системным хранилищем, построенным на базе файловой технологии хранения (NAS/NFS или локальная файловая система). При этом могут применяться следующие методы передачи данных:

- метод совместно используемой передачи (shared);
- метод передачи ssh.

Файловая технология хранения используется только для энергозависимых дисков и context-устройств.

3.6.4.2. Регистрация хранилища образов

Хранилища образов должны создаваться со значениями, приведенными в таблице 12.

Таблица 12

Параметр	Значение
NAME	Имя хранилища
`TYPE`	`IMAGE_DS`
`DS_MAD`	`dev`
`TM_MAD`	`dev`

Окончание таблицы 12

Параметр	Значение
DISK_TYPE	BLOCK

Пример

Создание хранилища с использованием конфигурационного файла:

1) создать файл `imageds.txt` со следующим содержанием:

```
NAME = rdm_datastore
TYPE = "IMAGE_DS"
DS_MAD = "dev"
TM_MAD = "dev"
DISK_TYPE = "BLOCK"
```

2) выполнить команду:

```
onedatastore create imageds.txt
```

Пример вывода после выполнения команды:

```
ID: 101
```

3.6.5. Использование хранилища

В хранилище можно добавлять новые образы с указанием пути. При использовании инструмента командной строки нельзя применять сокращенные параметры, т.к. вначале проверяется, существует ли файл и устройство на фронтальной машине.

Пример

Регистрация в хранилище `101` образа, которому соответствует диск `/dev/sdb`:

1) создать `image.tmpl` со следующим содержанием:

```
NAME=scsi_device
PATH=/dev/sdb
PERSISTENT=YES
```

2) выполнить команду:

```
oneimage create image.tmpl -d 101
```

Примечание. Данное хранилище является контейнером для существующих устройств, и образы используют его память. Все зарегистрированные устройства имеют размер 0, а хранилище устройств в целом занимает не более 1 МБ доступного пространства.

3.7. Хранилище образов iSCSI-Libvirt

Хранилище iSCSI-Libvirt является хранилищем образов, предназначенное для регистрации уже существующих томов iSCSI, доступных узлам гипервизора.

3.7.1. Настройка фронтальной машины

Дополнительная настройка не требуется.

3.7.2. Настройка узла виртуализации

Дополнительная настройка не требуется.

3.7.3. Аутентификация iSCSI CHAP

Для использования аутентификации по протоколу CHAP необходимо создать секретный ключ `libvirt` на всех узлах.

При использовании аутентификации CHAP необходимо учесть следующее:

- поле `incominguser` в файле аутентификации iSCSI должно соответствовать параметру хранилища `ISCSI_USER`;
- поле `<target>` в `XML`-файле секретного ключа должно содержать параметр `ISCSI_USAGE`;
- действия должны быть выполнены на всех узлах.

Для настройки аутентификации CHAP необходимо:

1) создать файл аутентификации iSCSI-источника, например, следующего содержания:

```
<target iqn.2013-07.com.example:iscsi-pool>
      backing-store /home/tgtd/iscsi-pool/disk1
      backing-store /home/tgtd/iscsi-pool/disk2
      incominguser myname mysecret
</target>
```

2) создать файл `iscsi-secret.xml` следующего содержания:

```
<secret ephemeral='no' private='yes'>
      <description>Passphrase for the iSCSI example.com server</description>
      <usage type='iscsi'>
            <target>libvirtiscsi</target>
      </usage>
</secret>
```

3) зарегистрировать созданный `XML`-файл в libvirt:

а) выполнить команду:

```
sudo virsh secret-define iscsi-secret.xml
```

Пример вывода после выполнения команды:

```
Secret c4dbe20b-b1a3-4ac1-b6e6-2ac97852ebb6 created
```

б) выполнить команду:

```
sudo virsh secret-list
```

Пример вывода после выполнения команды:

```
 UUID Usage
--------------------------------------------------------------
c4dbe20b-b1a3-4ac1-b6e6-2ac97852ebb6 iscsi libvirtiscsi
```

в) выполнить команды:

```
MYSECRET='printf %s "mysecret" | base64'
```

```
sudo virsh secret-set-value c4dbe20b-b1a3-4ac1-b6e6-2ac97852ebb6 \
$MYSECRET
```

Пример вывода после выполнения команды:

```
Secret value set
```

3.7.4. Общие настройки ПК СВ

После установки хранилища настройка ПК СВ выполняется в два этапа:

- регистрация системного хранилища;
- регистрация хранилища образов.

3.7.4.1. Регистрация системного хранилища

Хранилище iSCSI-Libvirt может работать с системным хранилищем, построенным на базе файловой технологии хранения (NAS/NFS или локальная файловая система). При этом могут применяться следующие методы передачи данных:

- метод совместно используемой передачи (shared);
- метод передачи ssh.

Файловая технология хранения используется только для энергозависимых дисков и context-устройств.

3.7.4.2. Регистрация хранилища образов

Хранилища образов должны создаваться со значениями, приведенными в таблице 13

Таблица 13

Параметр	Значение
NAME	Имя хранилища
TYPE	IMAGE_DS
DS_MAD	iscsi
TM_MAD	iscsi
DISK_TYPE	iscsi
ISCSI_HOST	Узел iSCSI. Например, host или host:port

При необходимости использовать аутентификацию CHAP добавить к хранилищу параметры, приведенные в таблице 14.

Таблица 14

Параметр	Значение
ISCSI_USAGE	Использование секретного ключа со строкой аутентификации CHAP
ISCSI_USER	Аутентификация iSCSI CHAP пользователя

Пример

Создание хранилища с использованием конфигурационного файла:

1) создать файл imageds.txt со следующим содержанием:

```
NAME = iscsi
DISK_TYPE = "ISCSI"
DS_MAD = "iscsi"
TM_MAD = "iscsi"
ISCSI_HOST = "the_iscsi_host"
ISCSI_USER = "the_iscsi_user"
ISCSI_USAGE = "the_iscsi_usage"
```

2) выполнить команду:

```
onedatastore create imageds.txt
```

Пример вывода после выполнения команды:

```
ID: 101
```

ВНИМАНИЕ! Образы, создаваемые в данном хранилище, должны быть помечены как постоянный. В противном случае, появляется возможность использования данного устройства более чем одной ВМ, что может привести к возникновению проблем и повреждению данных.

3.7.5. Использование хранилища

Можно добавлять новые образы с указанием полного пути. При использовании CLI не следует применять сокращенные параметры, т.к. CLI проверят, существует ли файл и устройство на фронтальной машине.

Пример

Регистрация в хранилище 101 образа, в который добавляется iscsi-target с идентификатором iqn.1992-01.com.example:storage:diskarrays-sn-a8675309:

1) создать image.tmpl со следующим содержанием:

```
NAME = iscsi_device
PATH = iqn.1992-01.com.example:storage:diskarrays-sn-a8675309
PERSISTENT = YES
```

2) выполнить команду:

```
oneimage create image.tmpl -d 101
```

ВНИМАНИЕ! Данное хранилище является контейнером для существующих устройств, и образы используют его память. Все зарегистрированные устройства имеют размер 0, а хранилище устройств в целом занимает не более 1 МБ доступного пространства.

Примечание. В шаблоне образа можно переопределить значения параметров ISCSI_HOST, ISCSI_USER, ISCSI_USAGE и ISCSI_IQN. Изменения будут применены для новых виртуальных машин.

Пример

Шаблон образа, в который добавляется LUN с идентификатором 0, принадлежащего

iscsi-target с идентификатором `iqn.2014.01.192.168.50.61:test:7cd2cc1e`:

```
NAME=iscsi_device_with_lun
PATH=iqn.2014.01.192.168.50.61:test:7cd2cc1e/0
ISCSI_HOST=192.168.50.61
PERSISTENT=YES
```

3.8. Хранилище файлов и ядер

Хранилище файлов и ядер используется для хранения обычных файлов. Такими файлами могут быть ядра виртуальных машин (kernels), временные диски (ramdisks) или контекстные файлы. Например, в хранилище файлов и ядер можно поместить определенный инит-скрипт и указать его в контекстуализации для ВМ. Тогда при загрузке ОС этой ВМ будет выполняться указанный инит-скрипт. Для передачи файлов из хранилища файлов в runtime-директорию ВМ используется драйвер ssh (устанавливается по умолчанию).

ВНИМАНИЕ! Если в ПК СВ для обеспечения отказоустойчивости сервиса фронтальной машины применяется технология Raft, хранилище файлов и ядер должно быть построено на базе файловой технологии хранения. При этом должна использоваться общая (распределенная) файловая система.

3.8.1. Требования

Специальные требования отсутствуют. В ходе работы используются стандартные утилиты, например, `cp`, `ln`, `mv`, `tar`, `mkfs`, которые установлены в системе по умолчанию.

3.8.2. Настройка фронтальной машины

Большинство критериев настройки, используемых для хранилищ образов дисков, применяются к файловому хранилищу.

Особые атрибуты для драйвера данного хранилища перечислены в таблице 15.

Таблица 15

Параметр	Значение
TYPE	FILE_DS
DS_MAD	fs
TM_MAD	ssh

Пример

Создание хранилища файлов и ядер с использованием конфигурационного файла:

1) создать файл `kernels_ds.txt` со следующим содержанием:

```
NAME = kernels
DS_MAD = fs
TM_MAD = ssh
```

```
TYPE = FILE_DS
SAFE_DIRS = /var/tmp/files
```

2) выполнить команду:

```
onedatastore create kernels_ds.txt
```

Пример вывода после выполнения команды:

```
ID: 100
```

Значения параметров `DS` и `TM MAD` можно впоследствии изменить командой `onedatastore update`. Подробные значения параметров хранилища можно просмотреть с помощью команды `onedatastore show`.

3.8.3. Настройка узла виртуализации

Рекомендуемый драйвер ssh для хранилища файлов не требует особой настройки. Достаточно убедиться в том, что на дисковом ресурсе, соответствующем этому хранилищу, достаточно места для размещения файлов ВМ на фронтальной машине и на узлах.

4. НАСТРОЙКА ОБЛАЧНОЙ СЕТИ

4.1. Общие сведения

При запуске новой ВМ сетевые интерфейсы этой ВМ, определяемые параметром NIC в настройках ВМ, подключаются к физическим устройствам узла виртуализации в соответствии с настройками виртуальной сети. Это позволяет ВМ иметь доступ к публичным и частным сетям.

ПК СВ поддерживает четыре сетевых режима:

1) режим Сетевой мост (Bridged) — ВМ напрямую соединяется с существующим мостом в узле виртуализации. Данный режим может быть настроен на использование групп безопасности и изоляции сети на уровне L2;

2) режим VLAN — виртуальные сети внедряются с применением технологии назначения портов на виртуальные локальные сети VLAN стандарта IEEE802.1Q (VLAN-тегирование стандарта IEEE802.1Q);

3) режим VXLAN — виртуальные сети задействуют сети VLAN, используя протокол VXLAN, основанный на UDP-инкапсуляции и групповой адресации IP;

4) режим Open vSwitch — аналогичен режиму VLAN, но использует программный коммутатор Open vSwitch (OVS) вместо сетевого моста. Группы безопасности данным режимом не поддерживаются.

ВНИМАНИЕ! Режим Open vSwitch не поддерживает классификационные метки и может использоваться только на минимальном уровне конфиденциальности.

Сетевой стек облачной сети может объединяться с внешним диспетчером IP-адресов (IPAM). Для этого необходимо добавить связующий элемент.

4.2. Режим Сетевой мост

В данном сетевом режиме трафик ВМ напрямую передается через существующий сетевой мост в узлах виртуализации. При этом устанавливается один из режимов фильтрации трафика, применяемой в облачной сети:

- режим «сетевой мост без фильтрации» (Bridged);
- режим «сетевой мост с группами безопасности» (Bridged with Security Groups, далее по тексту Security Group) — устанавливаются правила iptables для внедрения правил групп безопасности;
- режим «сетевой мост с правилами ebtables» (Bridged with ebtables isolation, далее по тексту ebtables VLAN) — тоже что и для режима Security Group, но с дополнительными правилами ebtables для изоляции (L2) всех виртуальных сетей.

4.2.1. Особенности и ограничения

При фильтрации трафика необходимо учитывать следующее:

- в режимах Bridged и Security Group можно добавлять тегированные сетевые интерфейсы для обеспечения сетевой изоляции. Данный режим является рекомендуемой стратегией развертывания в работающих системах (не тестовых);
- режим ebtables VLAN предназначен для небольших сред без соответствующей аппаратной поддержки для внедрения сетей VLANS. Данный режим ограничен сетями с длиной префикса 24 бита (/24) и IP-адреса не могут перекрываться в виртуальных сетях. Рекомендуется только для целей тестирования.

4.2.2. Настройка узла виртуализации

Для настройки данного сетевого режима необходимо выполнение следующих требований:

- на узлы виртуализации необходимо установить пакет `bridge-utils`;
- если планируется использовать режим фильтрации ebtables VLAN, на узлы необходимо установить пакет `ebtables`, который по умолчанию обеспечивает изоляцию сети.

На узле виртуализации необходимо создать сетевой мост для каждой сети, в которой будут работать виртуальные машины. При этом следует использовать одно имя сети на всех узлах.

Пример

Содержание файла `/etc/network/interfaces` с настройками сетевого моста

```
auto eth0
iface eth0 inet manual
auto br0
iface br0 inet static
bridge_ports eth0
address 172.16.1.20
netmask 255.255.255.0
gateway 172.16.1.1
```

4.2.3. Настройка фронтальной машины

Облачная сеть, функционирующая режиме Сетевой мост, не требует специальных настроек

4.2.4. Создание облачной сети

Для создания сети необходимо указать параметры, приведенные в таблице 16.

Таблица 16

Параметр	Значение	Обязательный
NAME	Имя облачной сети	ДА
VN_MAD	bridge — для режима без фильтрации; fw — для режима фильтрации с группами безопасности; ebtables — для режима фильтрации с изоляцией ebtables	ДА
BRIDGE	Имя сетевого моста в узлах виртуализации	ДА

Примеры:

1. Создание облачной сети с использованием конфигурационного файла. Будет создана облачная сеть, работающая в режиме сетевой мост с использованием режима фильтрации Security Group:

а) создать файл new-net.conf со следующим содержанием:

```
NAME = "bridged_net"
VN_MAD = "fw"
BRIDGE = "vbr1"
```

б) выполнить команду:

```
onevnet create new-net.conf
```

Пример вывода после выполнения команды:

```
ID: 1
```

2. Правила ebtables, которые создаются при необходимости отладки настройки:

```
# Игнорировать пакеты, которые не соответствуют MAC-адресу сети
-s ! <mac_address>/ff:ff:ff:ff:ff:0 -o <tap_device> -j DROP
# Предотвратить MAC-спуфинг
-s ! <mac_address> -i <tap_device> -j DROP
```

4.3. Сетевой режим VLAN

В данном сетевом режиме создается мост для каждой облачной сети и подключается VLAN-тегированный сетевой интерфейс к мосту. Механизм совместим со стандартом IEEE 802.1Q.

Идентификационный номер VLAN рассчитывается автоматически и будет одинаковым для всех интерфейсов в конкретной сети. Возможно также принудительно указать значение параметра VLAN_ID в настройках облачной сети.

4.3.1. Настройка узла виртуализации

Для настройки сетевого режима VLAN необходимо выполнение следующих требований:

- модуль 802.1Q должен быть загружен в ядро;

- наличие сетевого коммутатора, способного направлять VLAN-тегированный трафик. Физические порты сетевого коммутатора должны быть каналами связи VLAN.

4.3.2. Настройка фронтальной машины

Значение параметра VLAN_ID рассчитывается в соответствии с настройками, указанными в конфигурационном файле /etc/one/oned.conf (см. 5.2).

Изменением значения данного параметра можно зарезервировать некоторые сети VLAN, и они не будут назначаться облачной сети. Можно также указать первый номер VLAN_ID. При создании новой изолированной облачной сети определяется свободный номер VLAN_ID из пула VLAN. Этот пул является глобальным и совместно используется с сетевым режимом Open vSwitch.

В файле /var/lib/one/remotes/vnm/OpenNebulaNetwork.conf можно откорректировать параметр настройки validate_vlan_id. Установив значение true можно проверить, что другие сети VLAN не подсоединены к мосту.

4.3.3. Создание облачной сети

Для создания сети необходимо задать значения, приведенные в таблице 17.

Таблица 17

Параметр	Значение	Обязательный
NAME	Имя облачной сети	ДА
VN_MAD	802.1Q	ДА
PHYDEV	Имя физического сетевого устройства, которое будет подключено к сетевому мосту	ДА
BRIDGE	Имя сетевого моста, назначается по умолчанию onebr.<net_id> или onebr.<vlan_id>	НЕТ
VLAN_ID	Идентификационный номер сети VLAN. Будет сгенерирован, если не указан, а AUTOMATIC_VLAN_ID устанавливается на YES	ДА (если не AUTOMATIC_VLAN_ID)
AUTOMATIC_VLAN_ID	Обязательный и должен быть установлен на YES, если VLAN_ID определен	ДА (если не VLAN_ID)
MTU	Максимальный передаваемый модуль данных (MTU) для тегированного интерфейса и моста	НЕТ

Пример

Создание облачной сети, работающей в режиме VLAN, с использованием конфигурационного файла:

1) создать файл new-net.conf со следующим содержанием:

```
NAME = "hmnet"
VN_MAD = "802.1Q"
PHYDEV= "eth0"
```

```
VLAN_ID = 50
BRIDGE= "brhm"
```

2) выполнить команду:

```
onevnet create new-net.conf
```

Пример вывода после выполнения команды:

```
ID: 1
```

В данном примере проверяется наличие моста `brhm`. Если он не существует, то будет создан. Сетевое устройство `eth0` будет тегировано (`eth0.50`) и подсоединено к `brhm`.

4.4. Сетевой режим VXLAN

В данном сетевом режиме создается мост для каждой облачной сети и подключает VXLAN-тегированный сетевой интерфейс к мосту.

Идентификационный номер VLAN рассчитывается автоматически и будет одинаковым для всех интерфейсов в конкретной сети. Возможно также принудительно указать значение параметра `VLAN_ID` в шаблоне виртуальной сети.

Кроме того, каждая сеть VLAN назначает групповой адрес для инкапсуляции транслирования L2 и группового трафика. Данный адрес назначается по умолчанию диапазону 239.0.0.0/8 в соответствии с RFC 2365 (административно назначаемая групповая адресация IP). В частности, групповой адрес получается добавлением `VLAN_ID` к основному адресу 239.0.0.0/8.

4.4.1. Особенности и ограничения

В данном сетевом режиме задействован стандартный UDP-порт сервера 8472.

Трафик VXLAN направляется на физическое устройство, которое может быть установлено как VLAN-тегированный интерфейс, но в этом случае необходимо убедиться в том, что тегированный интерфейс будет создан вручную изначально на всех узлах.

4.4.2. Настройка узла виртуализации

Для настройки сетевого режима VXLAN необходимо чтобы при подключении всех узлов к одной подсети, групповой трафик не фильтровался правилами iptables в узлах. Если групповой трафик должен проходить через маршрутизаторы, необходимо настроить в сети многоадресный протокол, например, IGMP.

4.4.3. Настройка фронтальной машины

Значение параметра `VXLAN_ID` рассчитывается в соответствии с настройками, указанными в конфигурационном файле `/etc/one/oned.conf` (см. 5.2).

Параметры настройки, приведенные в таблице 18, можно откорректировать в файле `/var/lib/one/remotes/vnm/OpenNebulaNetwork.conf`.

Таблица 18

Параметр	Описание
vxlan_mc	Основной групповой адрес для каждой сети VLAN. Групповой адрес: vxlan_mc + vlan_id
vxlan_ttl	Время жизни (TTL) должно быть меньше 1 в маршрутизируемых многоадресных сетях (IGMP)
validate_vlan_id	Установить на true для проверки, что другие сети VLAN не подсоединены к мосту

4.4.4. Создание облачной сети

Для создания сети VXLAN необходимо задать значения, приведенные в таблице 19.

Таблица 19

Параметр	Значение	Обязательный
NAME	Имя облачной сети	ДА
VN_MAD	vxlan	ДА
PHYDEV	Имя физического сетевого устройства, которое будет подключено к мосту	ДА
BRIDGE	Имя сетевого моста, назначается по умолчанию onebr.<net_id> или onebr.<vlan_id>	НЕТ
VLAN_ID	Идентификационный номер сети VLAN. Будет сгенерирован, если не указан	НЕТ
MTU	Максимальный передаваемый модуль данных (MTU) для тегированного интерфейса и моста	НЕТ

Пример

Создание облачной сети, работающей в режиме VXLAN, с использованием конфигурационного файла:

1) создать файл new-net.conf со следующим содержанием:

```
NAME = "vxlan_net"
VN_MAD = "vxlan"
PHYDEV = "eth0"
VLAN_ID = 50 # optional
BRIDGE = "vxlan50" # optional
```

2) выполнить команду:

```
onevnet create new-net.conf
```

Пример вывода после выполнения команды:

```
ID: 1
```

В данном примере драйвер проверяет наличие моста vxlan50. Если он не существует, то будет создан. Сетевое устройство eth0 будет тегировано (eth0.50) и подсоединено

к `vxlan50`. Сетевое устройство eth0 может иметь 802.1Q тегированный интерфейс, если предполагается изолировать трафик облачной сети VXLAN.

4.5. Сети Open vSwitch

Сети Open vSwitch создаются на базе программного коммутатора Open vSwitch.

Open vSwitch — программный многоуровневый коммутатор, обеспечивающий изоляцию сети с помощью сетей VLAN путем тегирования портов и фильтрацию базовой сети с помощью OpenFlow.

Архитектура OVS состоит из трех основных компонентов: базы данных, непосредственно программного коммутатора и управляющего контроллера. На каждом из физических узлов вместе с гипервизором располагаются собственные БД и коммутатор. Эти два компонента образуют отдельно стоящий коммутатор, не знающий о других программных коммутаторах на соседних узлах.

ВНИМАНИЕ! Группы безопасности данным сетевым режимом не поддерживаются.

4.5.1. Особенности конфигурирования

Конфигурация всех Open vSwitch коммутаторов, портов, настройки поддерживаемых протоколов хранятся в собственной базе данных OVS (OVSDB). В стандартной конфигурации в OVSDB существуют следующие таблицы:

- Open_vSwitch — Схема
- Bridge
- Port
- Interface
- Flow_Table — конфигурация OpenFlow
- QoS
- Mirror
- Controller — параметры подключения к контроллеру OpenFlow
- Manager — конфигурация OVSDB
- NetFlow
- SSL
- sFlow
- IPFIX
- Flow_Sample_Collector_Set

Изначально почти все таблицы пусты, так как конфигурация отсутствует. Утилита `ovs-vsctl` предоставляет интерфейс для внесения изменений в БД. Для внесения изменений используется команда вида:

```
sudo ovs-vsctl <команда> <таблица> <запись> <ключ=значение>
```

Для создания программного коммутатора с именем ovs-sw0 необходимо выполнить следующую команду:

```
sudo ovs-vsctl add-br ovs-sw0
```

После появляется возможность подключения ВМ. При этом ВМ, подключенные к ovs-sw0, будут работать в изолированной сети. Для того, чтобы предоставить им доступ к внешней сети, необходимо подключить к ovs-sw0 в качестве порта физический интерфейс eth0, выполнив команду:

```
sudo ovs-vsctl add-port ovs-sw0 eth0
```

Для того чтобы разрешить порту eth0 пропускать во внешнюю сеть трафик из определенных VLAN, необходимо выполнить команду:

```
sudo ovs-vsctl set port eth0 trunks=10,20,30,40,50
```

Ниже представлен список вариантов команд с параметрами вызова:

- list <таблица> <запись>
- find <таблица> <условие>
- get <таблица> <запись> <ключ=значение>
- add <таблица> <запись> <ключ=значение>
- remove <таблица> <запись> <ключ=значение>
- clear <таблица> <запись> <ключ>
- create <таблица> <запись> <ключ=значение>
- destroy <таблица> <запись>
- wait-until <таблица> <запись> <ключ=значение>

Для просмотра записей, присутствующих в таблице, описывающей порты, выполнить команду:

```
sudo ovs-vsctl list port
```

Для вывода списка портов, включенных в VLAN, необходимо выполнить команду:

```
sudo ovs-vsctl find port tag=10
```

4.5.2. Агрегирование физических интерфейсов

Для повышения пропускной способности и уровня отказоустойчивости в Open vSwitch коммутатор могут быть включены несколько физических интерфейсов с задействованной на них агрегацией по протоколу LACP (Link Aggregation Control Protocol). Выполняется на канальном уровне путем создания объединенного интерфейса (Bonding).

Для создания объединенного интерфейса на базе физических интерфейсов eth0 и eth1 необходимо выполнить следующую команду:

```
sudo ovs-vsctl add-bond ovs-sw0 bond0 eth0 eth1
```

После следует включить lacp на созданном объединенном интерфейсе:

```
sudo ovs-vsctl set port bond0 lacp=active
```

На этом настройка отказоустойчивости сетевых интерфейсов завершена.

4.5.3. Зеркалирование портов

Open vSwitch позволяет направлять копию потока трафика из одного или нескольких интерфейсов в другой. Так же он может организовать перенаправление трафика из всей VLAN в конкретный порт или наоборот. Зеркалироваться может только входящий, только исходящий или оба типа трафика. Использование такой возможности позволит вести контроль сетевого трафика, передаваемого между ВМ с целью обнаружения (предупреждения) компьютерных атак.

Пример

Зеркалирование трафика из интерфейса `vnet2`, принадлежащего одной ВМ, в специально созданный для прослушивания порт `mirror0` с типом `internal`.

```
sudo ovs-vsctl -- set Bridge ovs-sw0 mirrors=@m -- \
    --id=@mirror0 get Port mirror0 -- --id=@vnet2 get Port vnet2 -- \
    --id=@m create Mirror name=mymirror select-dst-port=@vnet2 \
    select-src-port=@vnet2 output-port=@mirror0
```

где конструкцией `–id=@<имя_переменной>` определяется использование переменной;

командой `set Bridge ovs-sw0 mirrors=@m` создается зеркало, имя и параметры которого получаются из переменной `@m` (см. ниже);

командой `–id=@mirror0 get Port mirror0 – –id=@vnet2 get Port vnet2` определяются значения переменных `@mirror0, @vnet2` — записываются идентификаторы соответствующих портов;

командой `–id=@m create Mirror name=mymirror select-dst-port=@vnet2 select-src-port=@vnet2 output-port=@mirror0` определяется значение переменной `@m` — записываются имя и параметры зеркала;

`select-dst-port` — зеркалирование входящего трафика;

`select-src-port` — зеркалирование исходящего трафика;

`output-port` — место перенаправления трафика.

С помощью консольной утилиты `tcpdump`, запущенной на узле, можно прослушивать весь трафик поступающий, например, на порт `mirror0`. Для этого необходимо выполнить команду:

```
tcpdump -i mirror0
```

Также можно организовать ретрансляцию всех пакетов, например, пришедших на порт `eth0` или `eth1` на порт `eth2`:

```
sudo ovs-vsctl -- set Bridge ovs-sw0 mirrors=@m \
    -- --id=@eth0 get Port eth0 -- --id=@eth1 get Port eth1 \
    -- --id=@eth2 get Port eth2 \
    -- --id=@m create Mirror name=mymirror -- select-dst-port=@eth0,@eth1 \
```

```
select-src-port=@eth0,@eth1 output-port=@eth2
```

где конструкцией `-id=@<имя_переменной>` определяется использование переменной;

командой `set Bridge ovs-sw0 mirrors=@m` создается зеркало, имя и параметры которого получаются из переменной `@m` (см. ниже);

командой `- -id=@eth0 get Port eth0 - -id=@eth1 get Port eth1 - -id=@eth2 get Port eth2` определяются значения переменных `@eth0`, `@eth1` и `@eth2`;

командой `-id=@m create Mirror name=mymirror select-dst-port=@eth0, @eth1 select-src-port=@eth0,@eth1 output-port=@eth2` определяется значение переменной `@m` — записываются имя и параметры зеркала;

`select-dst-port` — зеркалирование входящего трафика;

`select-src-port` — зеркалирование исходящего трафика;

`output-port` — место перенаправления трафика.

Для отмены зеркалирования выполнить команду:

```
sudo ovs-vsctl remove Bridge ovs-sw0 mirrors mymirror
```

4.5.4. Настройка узла виртуализации

4.5.4.1. Требования

Для настройки данного сетевого режима необходимо чтобы на каждом узле был установлен пакет `openvswitch-switch` (данный пакет размещен в базовом репозитории ОС СН).

4.5.4.2. Настройка

Для настройки необходимо создать программный коммутатор для каждой сети, в которой будут работать виртуальные машины. На всех узлах необходимо использовать одно и тоже имя для программного коммутатора. Затем добавить физический сетевой интерфейс к этому программному коммутатору.

Пример

Узел, который направляет трафик виртуальных сетей через сетевой интерфейс `enp0s8`. Пример вывода после выполнения команды: `sudo ovs-vsctl show`

```
c61ba96f-fc11-4db9-9636-408e763f529e Bridge "ovsbr0"
Port "ovsbr0"
Interface "ovsbr0" type: internal
Port "enp0s8"
Interface "enp0s8"
```

4.5.5. Общие настройки ПК СВ

Значение параметра VLAN_ID рассчитывается в соответствии с настройками, указанными в конфигурационном файле /etc/one/oned.conf (см. 5.2).

Изменением данного параметра можно зарезервировать некоторые сети VLAN, и они не будут назначаться виртуальной сети. Можно также указать первый номер VLAN_ID. При создании новой изолированной облачной сети находит свободный номер VLAN_ID из пула VLAN. Этот пул является глобальным, а также совместно используется с сетевым режимом 802.1Q VLAN.

В файле /var/lib/one/remotes/vnm/OpenNebulaNetwork.conf можно откорректировать параметр настройки arp_cache_poisoning, отвечающий за подключение правила предотвращения изменения кэша ARP (ARP Cache Poisoning).

ВНИМАНИЕ! После корректировки значения параметра arp_cache_poisoning необходимо выполнить команду onehost sync для применения изменений на всех узлах виртуализации.

4.5.6. Создание облачной сети

Для создания облачной сети Open vSwitch необходимо задать значения, приведенные в таблице 20.

Таблица 20

Параметр	Значение	Обязательный
NAME	Имя облачной сети	ДА
VN_MAD	ovswitch	ДА
PHYDEV	Имя физического сетевого устройства, которое будет подключено к мосту	ДА
BRIDGE	Имя сетевого моста, назначается по умолчанию onebr.<net_id> или onebr.<vlan_id>	НЕТ
VLAN_ID	Идентификационный номер сети VLAN. Будет сгенерирован, если не указан и для параметра AUTOMATIC_VLAN_ID установлено значение YES	НЕТ
AUTOMATIC_VLAN_ID	Игнорируется, если параметр VLAN_ID определен. Следует установить значение YES, если необходимо в автоматическом режиме генерировать идентификационный номер сети VLAN	НЕТ
MTU	Максимальный передаваемый модуль данных (MTU) для сети Open vSwitch	НЕТ

Пример

Создание облачной сети Open vSwitch с использованием конфигурационного файла:

1) создать файл new-net.conf со следующим содержанием:

```
NAME = "ovswitch_net"
VN_MAD = "ovswitch"
BRIDGE = vbr1
VLAN_ID = 50 # optional
...
```

2) выполнить команду:

```
onevnet create new-net.conf
```

Пример вывода после выполнения команды:

```
ID: 1
```

4.5.7. Многоканальные сети VLAN (VLAN транкинг)

VLAN транкинг поддерживается путем добавления тега `VLAN_TAGGED_ID:` к элементу `NIC` в шаблоне ВМ или шаблоне виртуальной сети. Тег позволяет указать диапазон сетей VLAN, подлежащий тегированию, например, `1,10,30,32`.

4.5.8. Правила OpenFlow

4.5.8.1. MAC-спуфинг

Данные правила предотвращают выход любого трафика с порта, если был изменен MAC-адрес.

Пример

```
in_port=<PORT>,dl_src=<MAC>,priority=40000,actions=normal
in_port=<PORT>,priority=39000,actions=normal
```

4.5.8.2. IP-захват

Данные правила предотвращают выход любого трафика с порта для IPv4, если не настроен IP-адрес для ВМ.

Пример

```
in_port=<PORT>,arp,dl_src=<MAC>priority=45000,actions=drop
in_port=<PORT>,arp,dl_src=<MAC>,nw_src=<IP>,priority=46000,actions=normal
```

4.5.8.3. Черные порты

Применяется одно правило на порт.

Пример

```
tcp,dl_dst=<MAC>,tp_dst=<PORT>,actions=drop
```

4.5.8.4. ICMP-игнорирование

С помощью данной настройки можно, например, заблокировать ping-запросы к ВМ.

Пример

```
icmp,dl_dst=<MAC>,actions=drop
```

5. КОНФИГУРИРОВАНИЕ ПК СВ С ПОМОЩЬЮ СЛУЖБЫ ONED

Служба `oned` управляет облачными сетями, облачными хранилищами, узлами виртуализации и виртуальными машинами. Настройки службы `oned` размещены в конфигурационном файле `/etc/one/oned.conf`.

ВНИМАНИЕ! Изменение значений параметров службы `oned` производится в конфигурационных файлах каталога `/etc/one/one.d/`. Допускается править параметры в имеющихся файлах или добавлять новые параметры в виде отдельных файлов с расширением *.conf. После внесения изменений необходимо перезапустить сервис `oned`. После перезапуска в новый файл конфигурации `/etc/one/oned.conf` будут собраны значения параметров из всех файлов каталога `/etc/one/one.d/`.

5.1. Параметры настройки службы oned

Файл конфигурации службы поддерживает настройку параметров, приведенных в таблице 21.

Таблица 21

Параметр	Описание
MANAGER_TIMER	Время в секундах, необходимое службе для оценки периодических функций
MONITORING_INTERVAL_DATASTORE	Время в секундах между циклами мониторинга облачного хранилища. Параметр не может иметь значение меньше, чем параметр MANAGER_TIMER
MONITORING_INTERVAL_MARKET	Время в секундах между циклами мониторинга магазина приложений. Параметр не может иметь значение меньше, чем параметр MANAGER_TIMER
MONITORING_INTERVAL_DB_UPDATE	Время в секундах между циклами записи в БД информации мониторинга ВМ. Параметр не может иметь значение меньше, чем параметр MANAGER_TIMER. Чтобы запретить запись в БД информации мониторинга ВМ, необходимо установить значение «-1». Чтобы записывать в БД информацию мониторинга ВМ, получаемую при каждом цикле мониторинга, необходимо установить значение «0»
DS_MONITOR_VM_DISK	Количество интервалов времени MONITORING_INTERVAL_DATASTORE, по прошествии которых будет запущена процедура мониторинга дисков ВМ. Применяется только для облачных хранилищ, построенных на базе файловой технологии хранения, или использующих драйвер хранилища FS_LVM. Чтобы отключить процедуру мониторинга дисков ВМ, необходимо установить значение «0»
SCRIPTS_REMOTE_DIR	Удаленный путь для хранения скрипта мониторинга и управления ВМ

Окончание таблицы 21

Параметр	Описание
PORT	Порт, на котором oned будет принимать запросы xml-rpc
LISTEN_ADDRESS	IP-адрес узла для приема запросов xml-rpc (по умолчанию все IP-адреса)
DB	Блок настройки БД, по умолчанию в ПК СВ используется БД PostgreSQL: - BACKEND = "postgresql" (наименование БД), - SERVER — IP-адрес или сетевое имя компьютера, на котором запущена служба PostgreSQL-сервра, - PORT = 0 (порт для подключения к БД), - USER = "oneadmin" (имя пользователя БД), - PASSWD — пароль пользователя БД, - DB_NAME = "opennebula" (наименование БД)
VNC_PORTS	Пул портов VNC для автоматического назначения портов VNC, по возможности, устанавливать порт на START+VMID: - start — первый назначаемый порт; - reserved — список зарезервированных портов, разделенный запятыми. Два номера, разделенные двоеточием, указывают диапазон
VM_SUBMIT_ON_HOLD	Принудительное создание ВМ в состоянии удержания вместо состояния ожидания. Возможные значения YES (ДА) или NO (НЕТ)
LOG	Блок настройки системы регистрации: 1) SYSTEM — тип системы регистрации, возможные значения: - file (по умолчанию) — файловая система регистрации, - syslog — регистрация системных журналов, - std — регистрация в стандартный поток ошибок; 2) DEBUG_LEVEL — устанавливает уровень отладки зарегистрированных сообщений. Возможные значения: - 0 — ошибка, - 1 — предупреждение, - 2 — информация, - 3 — отладка

Пример

Значения параметров службы oned, установленные по умолчанию

```
LOG = [
      SYSTEM = "file",
      DEBUG_LEVEL = 3
      ]
```

```
#MANAGER_TIMER = 15

MONITORING_INTERVAL_DATASTORE = 300
MONITORING_INTERVAL_MARKET = 600
MONITORING_INTERVAL_DB_UPDATE = 0

#DS_MONITOR_VM_DISK = 10

SCRIPTS_REMOTE_DIR=/var/tmp/one
PORT = 2633
LISTEN_ADDRESS = "0.0.0.0"

DB = [ BACKEND = "postgresql",
       SERVER = "localhost",
       PORT = 0,
       USER = "oneadmin",
       PASSWD = "<thepassword>",
       DB_NAME = "opennebula"
       ]

VNC_PORTS = [
START = 5900,
RESERVED = "32768:65536"
# RESERVED = "6800, 6801, 6810:6820, 9869"
]

#VM_SUBMIT_ON_HOLD = "NO"
```

5.2. Параметры настройки облачных сетей

Облачные сети поддерживают настройку параметров, приведенных в таблице 22.

Таблица 22

Параметр	Описание
NETWORK_SIZE	Определяет размер по умолчанию для виртуальных сетей
MAC_PREFIX	MAC-префикс по умолчанию, предназначенный для создания автоматически генерируемых MAC-адресов (может переписываться шаблоном виртуальной сети)

Окончание таблицы 22

Параметр	Описание
VLAN_IDS	Блок настройки пула идентификаторов для автоматического назначения VLAN_ID. Данный пул предназначен для сетей 802.1Q (Open vSwitch и драйверы 802.1Q). Первый идентификатор будет иметь значение [START (см. ниже) + VNET_ID]: - START — начальное значение для определения пула идентификаторов VLAN_ID; - RESERVED — перечень зарезервированных идентификаторов VLAN_ID, разделенных запятыми. Два номера, разделенные двоеточием, указывают диапазон
VXLAN_IDS	Блок настройки автоматического назначения идентификатора сети VXLAN (VNI). Используется для сетей VXLAN. START — первый VNI, который может использоваться. Резервирование идентификаторов не применяется.

Пример

Значения параметров облачных сетей, установленные по умолчанию

```
NETWORK_SIZE = 254

MAC_PREFIX = "02:00"

VLAN_IDS = [
      START = "2",
      RESERVED = "0, 1, 4095"
      ]

VXLAN_IDS = [
      START = "2"
      ]
```

5.3. Параметры настройки облачных хранилищ

В хранилищах и шаблонах ВМ (настройках, касающихся образов) можно настроить значения параметров, приведенных в таблице 23.

Таблица 23

Параметр	Описание
DATASTORE_LOCATION	Путь к хранилищам. Одинаков для всех узлов и фронтальной машины. По умолчанию /var/lib/one/datastores
DATASTORE_CAPACITY_CHECK	Проверяет наличие достаточного пространства до создания нового образа. Значение по умолчанию Yes

Окончание таблицы 23

Параметр	Описание
DEFAULT_IMAGE_TYPE	Значение по умолчанию для поля TYPE, если оно отсутствует в шаблоне. Возможные значения: - OS — файл образа, содержащий операционную систему; - CDROM — файл образа, содержащий CDROM; - DATABLOCK — файл образа, содержащий блок данных, создаваемый как пустой блок
DEFAULT_DEVICE_PREFIX	Значение по умолчанию для поля DEV_PREFIX, если оно отсутствует в шаблоне. Отсутствующее поле DEV_PREFIX заполняется, когда создаются образы, поэтому изменение префикса не повлияет на существующие образы. Возможные значения: - префикс hd — для устройства IDE; - префикс sd — для устройства SCSI; - префикс vd — для устройства Virtio
DEFAULT_CDROM_DEVICE_PREFIX	Аналогично DEFAULT_DEVICE_PREFIX, но для устройств CDROM
DEFAULT_IMAGE_PERSISTENT	При клонировании или сохранении образа (командами oneimage clone и onevm disk-saveas) устанавливает тип образа «постоянный». Если этот параметр не определен, то тип образа наследуется из исходного образа.
DEFAULT_IMAGE_PERSISTENT_NEW	При создании образа (командой oneimage create) устанавливает тип образа «постоянный». По умолчанию для создаваемых образов установлен тип «непостоянный».

Пример

Значения параметров облачных хранилищ, установленные по умолчанию

```
#DATASTORE_LOCATION = /var/lib/one/datastores

DATASTORE_CAPACITY_CHECK = "yes"

DEFAULT_DEVICE_PREFIX = "sd"
DEFAULT_CDROM_DEVICE_PREFIX = "hd"

DEFAULT_IMAGE_TYPE = "OS"
#DEFAULT_IMAGE_PERSISTENT = ""
#DEFAULT_IMAGE_PERSISTENT_NEW = ""
```

5.4. Параметры настройки системы мониторинга

Для указания настроек системы мониторинга в конфигурационном файле используется блок IM_MAD, в котором указываются значения параметров, приведенных в таблице 24.

Таблица 24

Параметр	Описание
NAME	Имя сервиса
EXECUTABLE	Путь исполняемого модуля сервиса, может быть абсолютным или относительным (относительно каталога /usr/lib/one/mads/)
ARGUMENTS	Конфигурационный файл для сервиса, может быть абсолютным или относительным (относительно каталога /etc/one/)
THREADS	количество потоков, т.е., узлов, контролируемых одновременно

Пример

Настройки системы мониторинга, установленные по умолчанию после инициализации сервисов ПК СВ

```
IM_MAD = [
      NAME = "monitord",
      EXECUTABLE = "onemonitord",
      ARGUMENTS = "-c monitord.conf",
      THREADS = 8 ]
```

5.5. Система хуков

Хуки в ПК СВ являются программами, выполняемыми при изменении состояния ВМ или узлов. Хуки могут выполняться как локально, так и удаленно в узле, где работает ВМ. Для настройки системы хуков необходимо установить следующие значения в конфигурационном файле /etc/one/oned.conf:

- executable — путь исполняемого модуля драйвера хука, может быть абсолютным или относительным (относительно каталога /usr/lib/one/mads/);
- arguments — конфигурационный файл для исполняемого модуля драйвера хука, может быть абсолютным или относительным (относительно каталога /etc/one/).

Пример

```
HM_MAD = [
executable = "one_hm"
]
```

5.5.1. Хуки виртуальной машины (VM_HOOK)

Хуки ВМ определяются по следующим параметрами:

- name — имя хука;
- on — условия выполнения хука:

- CREATE — при создании ВМ;
- PROLOG — при нахождении ВМ в состоянии PROLOG;
- RUNNING — после успешной загрузки ВМ;
- UNKNOWN — при нахождении ВМ в неизвестном состоянии;
- SHUTDOWN — после отключения ВМ;
- STOP — после остановки ВМ (включая передачу образов ВМ);
- DONE — после удаления или отключения ВМ;
- CUSTOM — определяемое пользователем конкретное состояние STATE и комбинация состояний LCM_STATE для запуска хука;

- command — путь может быть абсолютным или относительным (относительно каталога /usr/share/one/hooks);
- arguments — аргументы для хука. Можно просмотреть информацию по ВМ с помощью команды $:
  - $ID — идентификатор ВМ;
  - $TEMPLATE — шаблон ВМ в формате xml с кодированием base64;
  - PREV_STATE — предыдущее состояние ВМ;
  - PREV_LCM_STATE предыдущее LCM-состояние ВМ;
- remote — удаленное выполнение. Возможные значения:
  - YES — хук выполняется на узле виртуализации, где установлена ВМ;
  - NO — хук выполняется на фронтальной машине. Является значением по умолчанию.

Пример

```
VM_HOOK = [
name = "advanced_hook",
on = "CUSTOM",
state = "ACTIVE", lcm_state = "BOOT_UNKNOWN", command = "log.rb",
arguments = "$ID $PREV_STATE $PREV_LCM_STATE"
]
```

5.5.2. Хуки узла (HOST_HOOK)

Хуки узла определяются по следующим параметрами:

- name — имя хука;
- on — условия выполнения хука:
  - CREATE — при создании узла (использование команды onehost create);
  - ERROR — при нахождении узла в состоянии сбоя;
  - DISABLE — после отключения узла;

- command — путь может быть абсолютным или относительным (относительно каталога /usr/share/one/hooks);
- arguments — аргументы для хука. Можно использовать следующую информацию об узле:
    - $ID — идентификатор узла;
    - $TEMPLATE — шаблон узла в формате xml с кодированием base64;
- remote — удаленное выполнение. Возможные значения:
    - YES — хук выполняется на узле;
    - NO — хук выполняется на фронтальной машине. Является значением по умолчанию.

5.6. Особенности работы ПК СВ в условиях применения мандатного управления доступом

Для того чтобы обеспечить возможность управления ПК СВ с использованием утилит командной строки (onevm, onehost и т. д.) в условиях применения мандатного управления доступом, необходимо:

1) на фронтальной машине в файле /etc/parsec/privsock.conf добавить следующее строки:

```
#brest
/usr/bin/oned
```

2) перезагрузить фронтальную машину.

6. МОНИТОРИНГ И УЧЕТ

6.1. Мониторинг

В ПК СВ используется распределенная система мониторинга. Сервис системы мониторинга `onemonitord` является составной частью службы `oned` и собирает информацию, касающуюся узлов виртуализации и виртуальных машин, такую как: состояние узла виртуализации, его основные показатели производительности, состояние ВМ и вычислительные ресурсы, потребляемые ВМ. Информация мониторинга формируется в результате выполнения ряда тестовых программ на узлах виртуализации и транслируется по сети на фронтальную машину (по умолчанию используется TCP/UDP-порт 4124).

ВНИМАНИЕ! Межсетевой экран фронтальной машины должен разрешать получение пакетов по прослушиваемому порту.

Схема работы системы мониторинга приведена на рис. 6.

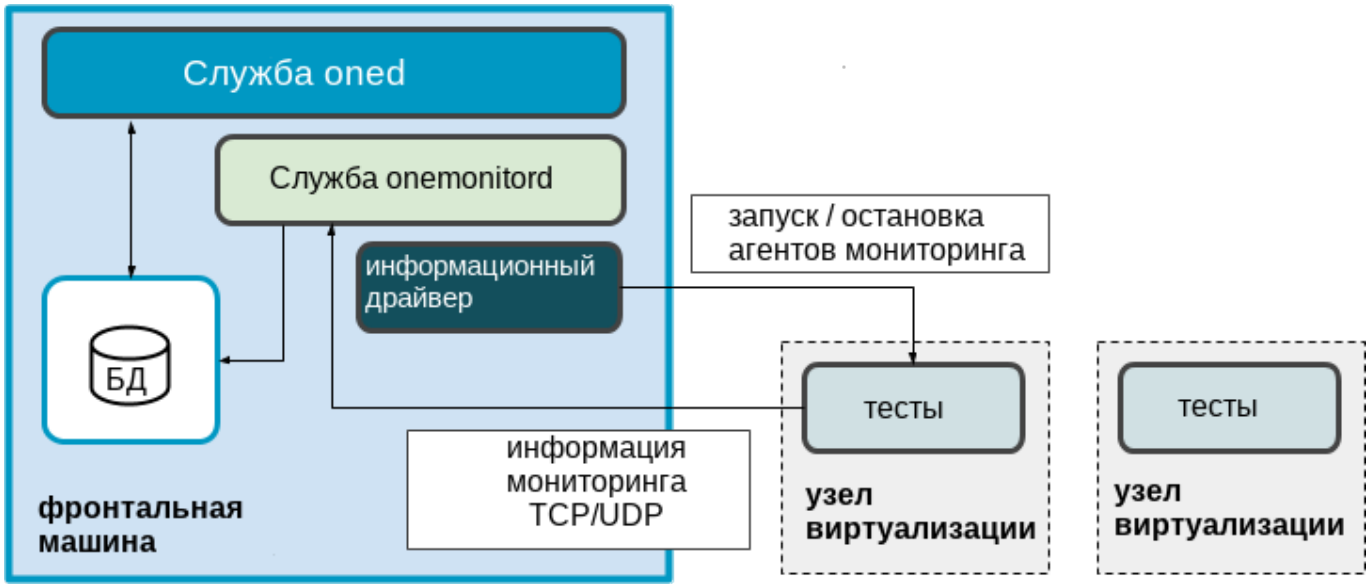

Рис. 6

При первичном запуске система мониторинга подключается к узлу виртуализации с помощью ssh и запускает на нем сервис агента мониторинга, который выполняет тесты и затем отсылает собранные данные сервису `onemonitord` на фронтальной машине. После этого агент мониторинга с заданной периодичностью выполняет тесты и отсылает собранные данные. В связи с этим системе мониторинга не требуется выполнять новые ssh-соединения для получения данных.

Для подключения к узлам виртуализации и управления агентами мониторинга используется специализированный информационный драйвер.

6.1.1. Конфигурация ПК СВ

При инициализации сервиса фронтальной машины автоматически запускается система мониторинга, параметры запуска которой определены в конфигурационном файле `/etc/one/oned.conf` (см. 5.4).

Настройки сервиса `onemonitord` размещены в конфигурационном файле `/etc/one/monitord.conf`, в котором можно задать значения параметров, приведенных в таблице 25.

Таблица 25

Параметр	Описание
`MANAGER_TIMER`	Время в секундах, необходимое сервису для оценки периодических функций
`MONITORING_INTERVAL_HOST`	Время ожидания отклика от агента мониторинга (в секундах). Если за указанное время отклик не получен, то тест выполняется заново
`HOST_MONITORING_EXPIRATION_TIME`	Время в секундах, по истечении которого информация мониторинга узла виртуализации устаревает. Если необходимо отключить мониторинг, необходимо установить значение «0»
`DB`	Блок настроек подключения к БД. Основные настройки подключения к БД указаны в конфигурационном файле `/etc/one/oned.conf`. В файле `/etc/one/monitord.conf` указывается только значение параметра `CONNECTIONS` — количество одновременных подключений к БД
`NETWORK`	Блок настроек сетевого соединения. Указываются значения следующих параметров: - `ADDRESS` — IP-адрес, на котором принимать информацию мониторинга (на заданный TCP/UDP-порт), - `MONITOR_ADDRESS` — IP-адрес фронтальной машины, на который агенты мониторинга отправляют информацию мониторинга, - `PORT` — TCP/UDP-порт, на котором принимать информацию мониторинга, - `THREADS` — количество одновременно наблюдаемых агентов мониторинга, - `PUBKEY` — абсолютный путь для открытого ключа. Не указывается, если шифрование не применяется, - `PRIKEY` — абсолютный путь для закрытого ключа. Не указывается, если шифрование не применяется.

Окончание таблицы 25

Параметр	Описание
PROBES_PERIOD	Блок настроек тестов. Указываются значения следующих параметров: - BEACON_HOST — Время в секундах, по прошествии которого на узел виртуализации отправляется тестовый пакет, для проверки его работоспособности, - SYSTEM_HOST — Время в секундах, по прошествии которого, должна быть получена информация о состоянии и конфигурации узла виртуализации, - MONITOR_HOST — Время в секундах, по прошествии которого, должна быть получена информация мониторинга узла виртуализации (о вычислительных ресурсах и сетевой нагрузке), - STATE_VM — Время в секундах, по прошествии которого, должна быть получена информация о состоянии ВМ, - MONITOR_VM — Время в секундах, по прошествии которого, должна быть получена информация о вычислительных ресурсах, потребляемых ВМ, - SYNC_STATE_VM — Время ожидания информации мониторинга ВМ. Если за указанное время информация не получена, то направляется полный отчет о ВМ.

Пример

Значения параметров системы мониторинга, установленные по умолчанию

```
#MANAGER_TIMER = 15
MONITORING_INTERVAL_HOST = 30 #rbt: automigration
#HOST_MONITORING_EXPIRATION_TIME = 43200
#VM_MONITORING_EXPIRATION_TIME = 43200

DB = [
CONNECTIONS = 15
]

NETWORK = [
ADDRESS = "0.0.0.0",
MONITOR_ADDRESS = "auto",
PORT = 4124,
THREADS = 8,
PUBKEY = "",
PRIKEY = ""
]

PROBES_PERIOD = [
BEACON_HOST = 30,
```

```
SYSTEM_HOST = 600,
MONITOR_HOST = 120,
STATE_VM = 5,
MONITOR_VM = 30,
SYNC_STATE_VM = 180
]
```

Для указания настроек информационного драйвера в конфигурационном файле `/etc/one/monitord.conf` используется блок `IM_MAD`, в котором указываются значения параметров, приведенных в таблице 26.

Таблица 26

Параметр	Описание
`NAME`	Наименование информационного драйвера
`SUNSTONE_NAME`	Тип гипервизора, установленного на узле виртуализации
`EXECUTABLE`	Путь исполняемого модуля драйвера, может быть абсолютным или относительным (относительно каталога `/usr/lib/one/mads/`)
`ARGUMENTS`	Параметры настройки функционирования информационного драйвера - `r` — количество перезапусков агента мониторинга, выполняемых в случае отсутствия конформации мониторинга в заданный период времени; - `t` — количество наблюдаемых агентов мониторинга; - `w` — таймаут (в секундах) до повторного выполнения внешних команд (по ssh);
`THREADS`	количество потоков, т.е., узлов, контролируемых одновременно

Пример

Настройки информационного драйвера для узла виртуализации с гипервизором kvm, установленные по умолчанию

```
IM_MAD = [
      NAME = "kvm",
      SUNSTONE_NAME = "KVM",
      EXECUTABLE = "one_im_ssh",
      ARGUMENTS = "-r 3 -t 15 -w 90 kvm",
      THREADS = 0
      ]
```

6.1.2. Отчет системы мониторинга

Отчеты системы мониторинга сохраняются в файл `/var/log/one/monitor.log`.

Пример

```
Thu Jun 23 12:35:28 2022 [Z0][DrM][I]: Loading drivers.
Thu Jun 23 12:35:28 2022 [Z0][DrM][I]: Loading driver: kvm
Thu Jun 23 12:35:28 2022 [Z0][DrM][I]:  Driver loaded: kvm
Thu Jun 23 12:35:28 2022 [Z0][DrM][I]: Loading driver: qemu
Thu Jun 23 12:35:28 2022 [Z0][DrM][I]:  Driver loaded: qemu
Thu Jun 23 12:35:28 2022 [Z0][DrM][I]: Loading driver: lxd
Thu Jun 23 12:35:28 2022 [Z0][DrM][I]:  Driver loaded: lxd
Thu Jun 23 12:35:28 2022 [Z0][DrM][I]: Loading driver: lxc
Thu Jun 23 12:35:28 2022 [Z0][DrM][I]:  Driver loaded: lxc
Thu Jun 23 12:35:28 2022 [Z0][DrM][I]: Loading driver: firecracker
Thu Jun 23 12:35:28 2022 [Z0][DrM][I]:  Driver loaded: firecracker
Thu Jun 23 12:35:28 2022 [Z0][DrM][I]: Loading driver: vcenter
Thu Jun 23 12:35:28 2022 [Z0][DrM][I]:  Driver loaded: vcenter
Thu Jun 23 12:35:32 2022 [Z0][HMM][I]: Raft status: SOLO
Thu Jun 23 12:35:41 2022 [Z0][HMM][I]: Successfully monitored VM: 10
Thu Jun 23 12:36:17 2022 [Z0][HMM][D]: Monitoring host fn.brest.local(0)
Thu Jun 23 12:36:23 2022 [Z0][HMM][I]: Successfully monitored VM: 10
Thu Jun 23 12:36:23 2022 [Z0][HMM][I]: Successfully monitored host: 0
Thu Jun 23 12:36:23 2022 [Z0][HMM][D]: Start monitor success, host: 0
Thu Jun 23 12:36:24 2022 [Z0][HMM][I]: Successfully monitored host: 0
Thu Jun 23 12:36:24 2022 [Z0][HMM][I]: Successfully monitored VM: 10
Thu Jun 23 12:36:55 2022 [Z0][HMM][I]: Successfully monitored VM: 10
Thu Jun 23 12:37:25 2022 [Z0][HMM][I]: Successfully monitored VM: 10
Thu Jun 23 12:38:02 2022 [Z0][HMM][D]: Monitoring host fn.brest.local(0)
```

Кроме того, в файле `/var/log/one/oned.log` также фиксируются сообщения, относящиеся к системе мониторинга.

Пример

```
Thu Jun 23 12:35:40 2022 [Z0][InM][D]: Monitoring datastore images-ds (100)
Thu Jun 23 12:35:40 2022 [Z0][InM][D]: Monitoring datastore system-ds (101)
Thu Jun 23 12:35:40 2022 [Z0][InM][D]: Monitoring datastore default (1)
Thu Jun 23 12:35:40 2022 [Z0][InM][D]: Monitoring datastore files (2)
...
Thu Jun 23 12:36:23 2022 [Z0][InM][D]: Host fn.brest.local (0)
    successfully monitored.
Thu Jun 23 12:36:26 2022 [Z0][InM][D]: Host fn.brest.local (0)
    successfully monitored.
```

6.1.3. Настройка и расширение

6.1.3.1. Шифрование информации мониторинга

В ПК СВ можно включить шифрование сообщений, которые агенты мониторинга направляют на фронтальную машину. Для этого необходимо выполнить следующие действия:

1) на фронтальной машине войти в ОС СН под учетной записью администратора с высоким уровнем целостности;

2) создать открытый и закрытый ключи, сохранив их, например, в каталог `/etc/one/`, командой:

```
sudo ssh-keygen -f /etc/one/onemonitor
```

Пример вывода после выполнения команды:

```
Generating public/private rsa key pair.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /etc/one/onemonitor.
Your public key has been saved in /etc/one/onemonitor.pub.
The key fingerprint is:
SHA256:LM2VAFRScOxz5UCeddyxuqQc9yOPBUUBWOfrn4Wh9/M root@fn.brest.local
```

3) создать новый файл открытого ключа в формате `PKCS#1` командой:

```
sudo ssh-keygen -f /etc/one/onemonitor.pub -e -m pem | \
    sudo tee /etc/one/onemonitor_pem.pub
```

4) в конфигурационном файле `/etc/one/monitord.conf`, в блоке настроек сетевого соединения указать абсолютные пути открытого и закрытого ключей;

Пример

```
NETWORK = [
...
    PUBKEY = "/etc/one/onemonitor_pem.pub",
    PRIKEY = "/etc/one/onemonitor"
    ]
```

5) перезапустить службу `oned` командой:

```
sudo systemctl restart opennebula
```

6) перезапустить агенты мониторинга на узлах виртуализации командой:

```
sudo -u oneadmin onehost sync -f
```

Пример вывода после выполнения команды:

```
* Adding fn.brest.local to upgrade
[========================================] 1/1 fn.brest.local
All hosts updated successfully.
```

6.1.3.2. Тесты

Тесты представляют собой специальные программы, которые обеспечивают получение контрольных показателей мониторинга. Конфигурационные файлы тестов определяются для каждого гипервизора и находятся по адресу `/var/lib/one/remotes/etc/im/<hypervisor>-probe.d/probe_db.conf`. Следующие параметры доступны для корректировки значений:

- `obsolete` — период времени (в минутах), по истечению которого информация о статусе ВМ считается устаревшей и будет удалена;
- `times_missing` — количество тестов, завершившихся неудачей, после которых для ВМ устанавливается статус «недоступна».

Пример

Настройки теста для узла виртуализации с гипервизором kvm, установленные по умолчанию

```
:obsolete: 720
:times_missing: 5
```

После внесения изменений в конфигурационные файлы тестов необходимо перезапустить агенты мониторинга на узлах виртуализации командой:

```
sudo -u oneadmin onehost sync -f
```

6.1.4. Получение информации о потреблении ресурсов

Для вывода информации о потреблении ресурсов узла виртуализации используется команда:

```
onehost monitoring <идентификатор_узла> <параметр_мониторинга> <вид_отображения>
```

Описание параметров мониторинга приведено в таблице 27.

Таблица 27

Параметр	Описание
`FREE_CPU`	Количество свободных ЦП
`FREE_MEMORY`	Объем свободной памяти
`USED_CPU`	Количество ЦП, выделенных для работы всех ВМ
`USED_MEMORY`	Объем памяти, выделенной для работы всех ВМ
`NETRX`	Объем входящего сетевого трафика
`NETTX`	Объем исходящего сетевого трафика

Если не указывать вид отображения, то информация мониторинга будет выведена в виде графика (в ОС СН должен быть установлен пакет `gnuplot`). Кроме того, в качестве вида отображения информации мониторинга можно указать следующее:

- `table` — табличный вид отображения;

- csv '<символ_разделителя>' — отображение в формате csv.

Дополнительно можно указать следующие параметры отображения:

- n — отображать последние «n» записей;

- unit <единицы_измерения> — отображение в заданных единицах измерения (например, «G» — в гигабайтах).

- start <дата> — отображать записи, начиная с указанной даты;

- end <дата> — отображать записи, до указанной даты.

ВНИМАНИЕ! Вид и параметры отображения указываются в качестве параметров команды, поэтому необходимо использовать префикс «- -».

Примеры:

1. Для отображения в виде графика (в ОС СН должен быть установлен пакет gnuplot) необходимо выполнить команду:

```
onehost monitoring 0 FREE_CPU --n 10 --unit G
```

Пример вывода после выполнения команды:

```
gnuplot 5.2 patchlevel 6
Host 0 FREE_CPU from 07/07/2022 10:00 to 07/07/2022 12:11
400 +-----------------------------------------------------------+
    |        +      **        +        +        +       +        |
395 |-+           ** *                 A        A              +-|
    |            *   *               ***       **               |
390 |-+         *    *              *  *      * *              +-|
    |          *     *            **   *     *  *               |
385 |-+       *      *           *     *    *   *              +-|
    |       **       *         **      *    *   *               |
380 |-+    *         *        *        *   *    *              +-|
375 |-+   *          *      **         *  *     *              +-|
    |    *           *     *           * *      *               |
370 |-+ *            *   **            **       *              **|
    | **             *  *              A        *        ******  |
365 |*+              ***                       *  ******      +-|
    |                A                         A**              |
360 |-+                                                        +-|
    |        +       +        +        +        +       +        |
355 +-----------------------------------------------------------+
12:04    12:05   12:06    12:07    12:08    12:09   12:10    12:11
```

2. Для отображения в виде таблицы необходимо выполнить команду:

```
onehost monitoring 0 FREE_CPU  --table --n 10 --unit G
```

Пример вывода после выполнения команды:

```
Host 0 FREE_CPU from 07/07/2022 10:00 to 07/07/2022 12:06

TIME VALUE
11:59 396
11:59 360
12:01 396
12:01 360
12:02 400
12:02 356
12:04 396
12:04 364
12:06 400
12:06 364
```

3. Для отображения в формате csv необходимо выполнить команду:

```
onehost monitoring 0 FREE_CPU  --csv ';' --n 10 --unit G
```

Пример вывода после выполнения команды:

```
TIME;VALUE
11:59;396
11:59;360
12:01;396
12:01;360
12:02;400
12:02;356
12:04;396
12:04;364
12:06;400
12:06;364
```

6.2. Логирование

6.2.1. Настройка системы регистрации

В ПК СВ обеспечивается ведение журналов для большинства ресурсов. Поддерживается три системы регистрации: файловая система регистрации, регистрация системных журналов и регистрация в стандартный поток ошибок. Для настройки системы регистрации используется блок настроек `LOG` в конфигурационном файле `/etc/one/oned.conf` (см. 5).

При использовании файловой регистрации создаются отдельные файлы журналов для каждого активного компонента, при этом все они хранятся в каталоге `/var/log/one`. В качестве таких активных компонент могут выступать:

- служба `oned`, регистрационная информация которой выгружается в файл `/var/log/one/oned.log`;

- сервис системы мониторинга onemonitord, регистрационная информация которого выгружается в файл /var/log/one/monitor.log;
- виртуальные машины — информация, относящаяся к ВМ, будет выгружаться в файл журнала /var/log/one/<идентификатор_ВМ>.log.

6.2.2. Регистрационный формат

Сообщения для файловой системы регистрации имеют следующую структуру:

```
<дата> [Z<zone_id>][<module>][<log_level>]: <текст_сообщения>
```

где <zone_id> — идентификатор зоны при объединении экземпляров ПК СВ в единый ЦОХД (служебный режим «федерация»), для независимого экземпляра ПК СВ имеет значение «0»;

<module> — краткое наименование компонента ПК СВ (VMM — для ВМ, InM — для информационного драйвера, TM — для драйвера передачи данных и т.д.);

<log_level> — представляет собой отдельный символ, указывающий уровень регистрации: I — для информации, D — для отладки и т.д.

Пример

Сообщения для файловой системы регистрации представленные в файле /var/log/one/oned.log

```
Thu Jul 7 16:29:34 2022 [Z0][TrM][D]: Message received: TRANSFER SUCCESS 26 -/
    1 1
Thu Jul 7 16:29:34 2022 [Z0][VMM][I]: Successfully execute transfer manager /
   driver operation: tm_context.
Thu Jul 7 16:29:35 2022 [Z0][VMM][I]: ExitCode: 0
Thu Jul 7 16:29:35 2022 [Z0][VMM][I]: Successfully execute network driver /
   operation: pre.
Thu Jul 7 16:29:35 2022 [Z0][VMM][I]: Successfully execute virtualization /
   driver operation: /bin/mkdir -p.
Thu Jul 7 16:29:35 2022 [Z0][VMM][I]: Successfully execute virtualization /
   driver operation: /bin/cat - >/var/lib/one/vms/26/vm.xml.
Thu Jul 7 16:29:35 2022 [Z0][VMM][I]: Successfully execute virtualization /
   driver operation: /bin/cat - >/var/lib/one/vms/26/ds.xml.
Thu Jul 7 16:29:35 2022 [Z0][VMM][I]: Successfully execute virtualization /
   driver operation: deploy.
...
Thu Jul 7 16:30:25 2022 [Z0][InM][D]: Host fn.brest.local (0) successfully /
   monitored.
Thu Jul 7 16:30:27 2022 [Z0][InM][D]: Host fn.brest.local (0) successfully /
   monitored.
```

Сообщения для регистрации системных журналов имеют следующую структуру:

```
<дата> <имя_компьютера> process[<pid>]: [Z<zone_id>][module][log_level]:
```

<текст_сообщения>

При этом сообщения о состоянии ВМ для регистрации системных журналов имеют следующую структуру:

```
<дата> <имя_компьютера> process[<pid>]: [<идентификатор_ВМ>][Z<zone_id>]
   [module][log_level]: <текст_сообщения>
```

Пример

Сообщения ПК СВ, представленные в файле /var/log/syslog

```
Jul 7 16:40:49 fn oned[25658]: [VM 26][Z0][VM][I]: New state is ACTIVE
Jul 7 16:40:49 fn oned[25658]: [VM 26][Z0][VM][I]: New LCM state is /
   BOOT_POWEROFF
Jul 7 16:40:49 fn oned[25658]: [VM 26][Z0][VMM][I]: Generating deployment /
   file: /var/lib/one/vms/26/deployment.1
...
Jul 7 16:40:50 fn oned[25658]: [Z0][VMM][I]: Successfully execute transfer /
   manager driver operation: tm_context.
Jul 7 16:40:50 fn oned[25658]: [Z0][VMM][I]: ExitCode: 0
Jul 7 16:40:50 fn oned[25658]: [Z0][VMM][I]: Successfully execute network /
   driver operation: pre.
Jul 7 16:40:50 fn oned[25658]: [Z0][VMM][I]: Successfully execute /
   virtualization driver operation: /bin/mkdir -p.
Jul 7 16:40:50 fn oned[25658]: [Z0][VMM][I]: Successfully execute /
   virtualization driver operation: /bin/cat - >/var/lib/one/vms/26/vm.xml.
Jul 7 16:40:50 fn oned[25658]: [Z0][VMM][I]: Successfully execute /
   virtualization driver operation: /bin/cat - >/var/lib/one/vms/26/ds.xml.
Jul 7 16:40:51 fn oned[25658]: [Z0][VMM][I]: Successfully execute /
   virtualization driver operation: deploy.
Jul 7 16:40:52 fn oned[25658]: [Z0][VMM][I]: ExitCode: 0
Jul 7 16:40:52 fn oned[25658]: [Z0][VMM][I]: Successfully execute network /
   driver operation: post.
Jul 7 16:40:52 fn oned[25658]: [VM 26][Z0][VM][I]: New LCM state is RUNNING
```

Сообщения для регистрации в стандартный поток ошибок имеют следующую структуру:

```
<дата> [Z<zone_id>][<module>][<log_level>]: <текст_сообщения>
<дата> [<идентификатор_ВМ>][Z<zone_id>][<module>][<log_level>]:
   <текст_сообщения>
```

Пример

Сообщения регистрации в стандартный поток ошибок:

```
Thu Jul 7 17:02:46 2022 [Z0][VMM][I]: ExitCode: 0
Thu Jul 7 17:02:46 2022 [Z0][VMM][I]: Successfully execute network driver /
   operation: clean.
Thu Jul 7 17:02:46 2022 [Z0][IPM][D]: Message received: SHUTDOWN SUCCESS 26 -/
```

```
    0 0
Thu Jul 7 17:02:46 2022 [VM 26][Z0][VM][I]: New state is POWEROFF
Thu Jul 7 17:02:46 2022 [VM 26][Z0][VM][I]: New LCM state is LCM_INIT
Thu Jul 7 17:03:06 2022 [Z0][InM][D]: Host fn.brest.local (0) successfully /
   monitored.
```

6.2.3. Вывод информации о виртуальной машине

Для получения информации о ВМ необходимо выполнить команду:

```
onevm show <идентификатор_ВМ>
```

Пример

Вывод информации о ВМ с идентификатором «0»:

```
VIRTUAL MACHINE 0 INFORMATION
ID : 0
NAME : tmp-for-install-os
USER : brest-admin
GROUP : brestadmins
STATE : DONE
LCM_STATE : LCM_INIT
LOCK : None
RESCHED : No
START TIME : 06/21 13:02:01
END TIME : 06/21 14:53:10
DEPLOY ID : 12ba00af-4eda-49a1-bd29-7efc1df27b77
...
USER TEMPLATE
AUTOSTARTVM="0"
HOT_RESIZE=[
CPU_HOT_ADD_ENABLED="NO",
MEMORY_HOT_ADD_ENABLED="NO" ]
HYPERVISOR="kvm"
INPUTS_ORDER=""
MEMORY_UNIT_COST="MB"
SCHED_DS_REQUIREMENTS="ID=\"0\""
SCHED_MESSAGE="Thu Jun 23 17:18:34 2022: Cannot dispatch VM to any Host. /
   Possible reasons: Not enough capacity in Host or System DS, dispatch limit/
    reached, or limit of free leases reached."
SERVICEUSERVM="0"
```

Ошибка, приведенная в примере (поле SCHED_MESSAGE), указывает на то, что было невозможно разместить ВМ на узле виртуализации, возможно недостаточно свободных вычислительных ресурсов.

6.2.4. Вывод информации об узле виртуализации

Для получения информации об узле виртуализации необходимо выполнить команду:

```
onehost show <идентификатор_узла>
```

Пример

Вывод информации об узле виртуализации с идентификатором «0»:

```
HOST 0 INFORMATION
ID : 0
NAME : fn.brest.local
CLUSTER : default
STATE : MONITORED
IM_MAD : kvm
VM_MAD : kvm
LAST MONITORING TIME : 07/07 14:57:49

HOST SHARES
RUNNING VMS : 2
MEMORY
      TOTAL : 5.8G
      TOTAL +/- RESERVED : 5.8G
      USED (REAL) : 3.3G
      USED (ALLOCATED) : 4G
CPU
      TOTAL : 400
      TOTAL +/- RESERVED : 400
      USED (REAL) : 44
      USED (ALLOCATED) : 50

LOCAL SYSTEM DATASTORE #0 CAPACITY
TOTAL: : 61.8G
USED: : 24.2G
FREE: : 34.5G
...

VIRTUAL MACHINES

ID USER  GROUP NAME STAT CPU MEM HOST TIME
25 brest-ad brestadm ALSE runn 0.25 2G fn.brest.local 13d 02h22
24 brest-ad brestadm ALCE runn 0.25 2G fn.brest.local 13d 02h31
```

7. ИНТЕГРАЦИЯ В ЕДИНЫЙ ЦОХД («ФЕДЕРАЦИЯ»)

7.1. Общие сведения

Несколько экземпляров ПК СВ могут быть объединены в единый центр обработки и хранения данных (ЦОХД), который называется «федерация». В этом случае каждый экземпляр ПК СВ называется зоной. Один из экземпляров ПК СВ настраивается как ведущий, остальные — ведомые.

«Федерация» позволяет конечным пользователям использовать ресурсы, распределенные Администраторами единого ЦОХД, независимо от места их нахождения. Интеграция проходит комплексно, то есть пользователю, авторизованному в веб-интерфейсе определенной зоны, не придется выходить из системы и вводить адрес другой зоны. веб-интерфейс ПК СВ позволяет изменять активную зону в любое время, а также автоматически перенаправляет запросы в ПК СВ в целевой зоне.

Служебный режим «федерация» является интеграцией с непосредственными связями. Все экземпляры ПК СВ имеют общую конфигурацию (общие таблицы БД) учетных записей пользователей, групп и полномочий. Доступ возможно ограничить до конкретных зон, а также до конкретных кластеров внутри данной зоны. Только ведущая зона ПК СВ имеет права на внесение записей в общие таблицы, у ведомых зон хранится локальная копия для чтения. Это гарантирует целостность данных без ущерба для скорости действий по считыванию.

Синхронизация выполняется путем настройки конфигурации ПК СВ для репликации только определенных таблиц. Репликация способна работать при соединениях на больших расстояниях и при нестабильных соединениях. В случае сбоя ведущей зоны и ее длительной перезагрузки ведомые зоны могут продолжать работать в нормальном режиме, за исключением нескольких действий, например, создание нового пользователя или обновление паролей.

Новые ведомые зоны можно добавлять к существующей «федерации» в любой момент. Кроме того, администратор может добавить абсолютно новый экземпляр ПК СВ или импортировать существующую развертку в «федерацию», сохранив действующих пользователей, групп, конфигурацию и виртуальные ресурсы.

Что касается обновлений ПК СВ, БД разработана таким образом, чтобы различные версии ПК СВ являлись частью одной и той же «федерации». При необходимости обновить локальные таблицы (ВМ, Образ, объекты VNet) новые версии сохранят совместимость с общими таблицами. На практике это означает, что в случае выхода новой версии ПК СВ можно обновить каждую зону в разное время без каких-либо последствий для «федерации».

Сервис веб-интерфейса ПК СВ ведущей зоны подключен ко всем экземплярам служ-

бы `oned` в «федерации», что позволяет пользователям менять зоны. Возможно использовать один веб-интерфейс ПК СВ для всей «федерации» или в каждой зоне использовать свой веб-интерфейс ПК СВ (рис. 7).

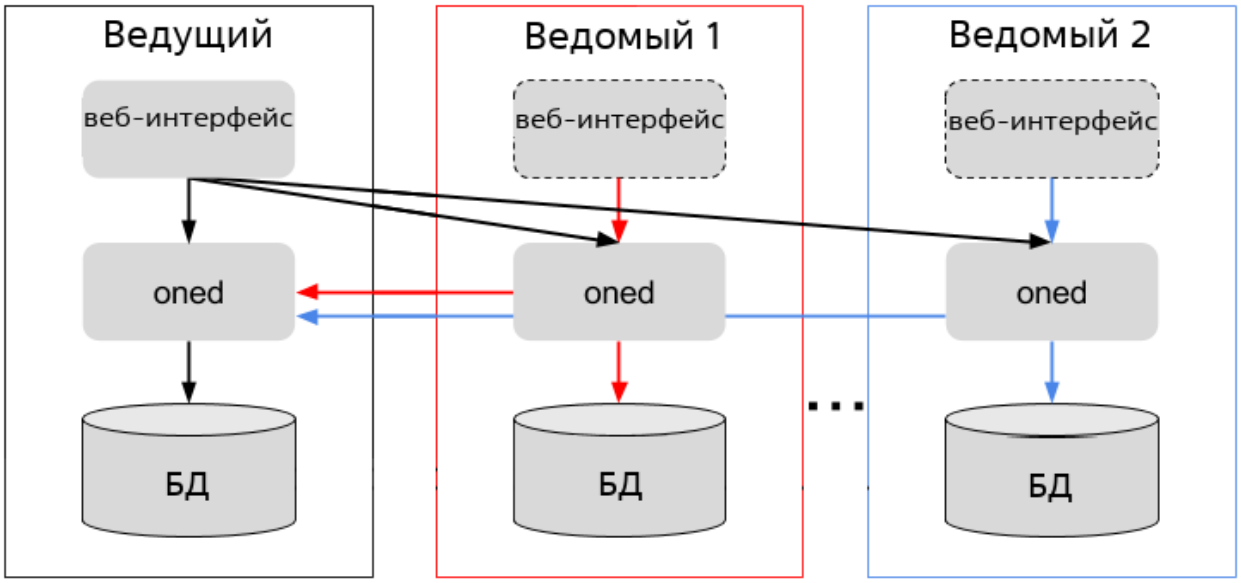

Рис. 7

ВНИМАНИЕ! Несмотря на то, что один сервис веб-интерфейса ПК СВ может подключаться к различным зонам, все остальные сервисы ПК СВ будут работать только с локальными ресурсами зоны.

В «федерации» зарегистрирован уникальный аккаунт `oneadmin`, обладающий правами администратора и являющийся аккаунтом администратора «федерации». В безопасной среде администратор каждой зоны авторизуется с помощью аккаунта в группе `oneadmin`. В других сценариях администратор «федерации» может создать специальную административную группу с полными правами доступа только для одной зоны.

Администраторы могут совместно использовать устройства в зонах благодаря сервису Магазин приложений.

7.2. Настройка «федерации»

Для автоматической настройки «федерации» можно воспользоваться скриптом `brestcloud-federation-configure`, который запускается на фронтальной машине ведущей зоны от имени администратора командой:

```
sudo  brestcloud-federation-configure
```

Далее необходимо следовать указаниями мастера настройки.

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

БД — база данных

ВМ — виртуальная машина

ГИП — графический интерфейс пользователя (GUI)

ЕПП — единое пространство пользователей

ОС — операционная система

ОС СН — операционная система специального назначения «Astra Linux Special Edition»

ПК СВ — программный комплекс «Средства виртуализации «Брест»

ПО — программное обеспечение

СЗИ — средства защиты информации

ФС — файловая система

ЦОХД — центр обработки и хранения данных

ЦП — центральный процессор

ALD — Astra Linux Directory (единое пространство пользователей)

CHAP — Challenge Handshake Authentication Protocol (протокол аутентификации с косвенным согласованием, предусматривающим передачу не самого пароля пользователя, а косвенных сведений о нм)

CephFS — Ceph File System (POSIX-совместивамя файловая система на базе кластера Ceph)

CLI — Command Line Interface (интерфейс командной строки)

CLVM — Clustered Logical Volume Manager (кластерное управление логическими томами)

CPU — Central Processing Unit (центральный процессор)

DRBD — Distributed Replicated Block Device (распределнное реплицируемое блочное устройство)

I/O MMU — Input/Output Memory Management Unit (блок управления памятью для операций ввода-вывода)

IPMI — Intelligent Platform Management Interface (интерфейс, обеспечивающий автономный мониторинг, восстановление и журналирование работы функций, встроенных непосредственно в аппаратное и микропрограммное обеспечения серверных платформ)

iSCSI — Internet Small Computer System Interface (протокол на базе TCP/IP для взаимодействия и управления системам хранения данных, серверов и клиентов)

KVM — Kernel-based Virtual Machine (программное решение, обеспечивающее виртуализацию в среде Linux на платформе, которая поддерживает аппаратную виртуализацию на базе Intel VT (Virtualization Technology) либо AMD SVM (Secure

Virtual Machine))

LIO — Linux-IO (программный комплекс, выполняющий функции цели (target))

LUN — Logical Unit Number (номер объекта внутри цели (target))

LV — Logical Volume (логический том)

LVM — Logical Volume Manager (менеджер логических томов)

MDS — Metadata Server (сервер кластера Ceph, отслеживающий метаданные файловой иерархии для CephFS)

MON — Monitor (демон, отслеживающий состояние кластера Ceph)

NBD — Network Block Device (сетевое блочное устройство)

NFS — Network File System (сетевая файловая система)

OSD — Object Storage Device (основное устройстве хранения объектов Ceph, обычно связанное с одним физическим диском, в котором хранятся фактические данные пользователя)

OVS — Open vSwitch (программный многоуровневый коммутатор для работы в гипервизорах и на компьютерах с виртуальными машинами)

OVSDB — база данных OVS

PCI — Peripheral component interconnect (шина ввода-вывода для подключения периферийных устройств к материнской плате компьютера)

RDP — Remote Desktop Protocol (протокол удалнного рабочего стола)

QEMU — Quick Emulator (средства эмуляции аппаратного обеспечения)

RADOS — Reliable Autonomic Distributed Object Store (хранилище, отвечающее за хранение объектов кластера Ceph независимо от их типа данных)

RBD — Rados block device (блочное хранилище кластера Ceph, которое может отображаться, форматироваться и монтироваться в точности как любой другой диск в сервере)

RDM — Raw Device Mapping (используется для прямого подключения к виртуальной машине существующих блочных устройств в узлах)

SAN — Storage Area Network (сеть хранения данных)

SCSI — Small Computer System Interface (системный интерфейс малых компьютеров)

SPICE — Simple Protocol for Independent Computing Environments (простой протокол для независимой вычислительной среды)

SSH — Secure Shell Protocol (протокол защищенной передачи информации)

TLS — Transport Layer Security (безопасность транспортного уровня)

TM — Transfer Manager (драйвер передачи)

UDP — User Datagram Protocol (протокол пользовательских дейтаграмм)

UUID — Universally Unique Identifier (универсальный уникальный идентификатор)

VCPU — Virtual Central Processing Unit (виртуальный центральный процессор)

VDI — Virtual Desktop Infrastructure (инфраструктура виртуальных рабочих столов)

VLAN — Virtual Local Area Network (виртуальная локальная вычислительная сеть)

VNC — Virtual Network Computing (система удалнного доступа к рабочему столу компьютера)

VXLAN — Virtual Extensible Local Area Network (виртуальная масштабируемая локальная вычислительная сеть)

Лист регистрации изменений

Изм.	Номера листов (страниц)				Всего листов (страниц) в документе	Номер документа	Входящий номер сопро-водительного документа и дата	Подпись	Дата
	изменен-ных	заменен-ных	новых	аннулиро-ванных					
1		Все				РДЦП.03-22 от 03.11.2022			