

Программное обеспечение «Базис.Virtual Security». Руководство по установке. Часть 3. Программный модуль «Базис.vCore»

RU.НРФЛ.00002-02 93 01 Ч3

Москва 08/11/2023

Содержание

Аннотация	4
Идентификационные данные	5
Общая информация Базис.vCore	6
Аннотация	6
Термины и определения	6
Требования к оборудованию Базис.vCore	7
Установка Базис.vCore	9
Лемонстрационная инсталляция Базис уСоге	9
Простая инсталляция Базис vCore	12
Ввеление	12
Настройка окружения	
Управляющая часть	
Вычислительная часть	
Инсталляция с поддержкой высокой доступности Базис.vCore	
Введение	
Предварительная настройка	27
Настройка узлов кластера	
настройка Redis и сервиса управления кластером Redis Sentinel	35
Установка vCore	
Интеграция Базис.vCore с Базис.Virtual Security	
Введение	
Управляющая часть	47
Вычислительная часть	
Обновление Базис.vCore	49
Предварительные действия перед обновлением	
Управляющая часть	
Обновление пакетов	
Обновление конфигурации сервисов	
Запуск сервисов	
Вычислительная часть	
Удаление Базис.vCore	53
Управляющая часть	
Вычислительная часть	53
Приложения к руководству по установке Базис.vCore	54
Приложение 1. Список портов сервисов Базис.vCore	
\${action.page.title}	
Приложение 2. Рекомендации по настройке систем хранения для хранилищ	
Базис.vCore	62
Введение	62
NFS	62

Fibre Channel	64

Аннотация

Настоящее руководство содержит описание действий по установке, настройке и обновлению программного модуля «Базис.vCore» (далее Система, Базис.vCore, vCore).

Идентификационные данные

Идентификационные данные ПО	Программное обеспечение «Базис.Virtual Security».Программный модуль «Базис.vCore»
Название документа	Программное обеспечение «Базис.Virtual Security». Руководство по установке. Часть 3. Программный модуль «Базис.vCore»
Обозначение документа	RU.НРФЛ.00002-02 93 01 Ч3
Автор документа	ООО «БАЗИС»

Общая информация Базис.vCore

Аннотация

Настоящая инструкция предназначена для технического администратора ПО и содержит подробное описание работ, необходимых для установки ПО.

Термины и определения

Термин, сокращение	Сокраще ние	Определение
Базис.vCore	vCore	Программный комплекс управления масштабируемой системой виртуализации с высокой надежностью.
Вычислительный узел	ВУ	Структурная единица, представляющая физический вычислительный узел, на котором размещен гипервизор и агент vCore.
Гипервизор		Компонент, предоставляющий функции виртуализации, мониторинга, управления сетью и дисками виртуальных машин.
Дата-центр	дц	Структурная единица, представляющая абстракцию физического дата-центра. Объединяет под собой хранилища, кластеры, сети, шаблоны виртуальных машин. Дата-центр может проецироваться как на физический дата-центр, так и просто на группу нескольких серверов (серверную стойку), которые имеют общий доступ к определенным хранилищам и сетям.
Домен		Структурная единица верхнего уровня, которая объединяет под собой ведение объектов безопасности (пользователи, роли, группы, разрешения), инфраструктурных объектов (дата-центры, хранилища, сети, кластеры и т.д.), и объектов учета ресурсов облака (предложений развертки, проектов, виртуальных машин).
Инфраструктура		Совокупность компьютерного оборудования (серверы, системы хранения данных, коммутационное оборудование и др.) и системного ПО, включая средства виртуализации.
Кластер		Структурная единица, которая объединяет под собой вычислительные узлы одной архитектуры (тип и семейство процессора).
Программное обеспечение	ПО	Совокупность машинного кода, выполняемых вычислительной системой.
Ресурс		Комплекс программно-аппаратных средств облачной платформы.
Управляющий узел/ контроллер		Узел, осуществляющий управление системой виртуализацией. На узле располагаются коммуникационные сервисы, которые связывают компоненты системы виртуализации и исполняет функции сервиса очередей сообщений, базы данных и панели управления, а так же может содержать компоненты балансировки нагрузки.

Требования к оборудованию Базис.vCore

К аппаратному и программному обеспечению, которые используются для функционирования программного модуля «Базис.vCore», предъявляются требования, изложенные в таблице.

Таблица - Минимальные требования к программному и аппаратному обеспечению

Элемент	Параметр		
Гипервизор 1 типа «Базис.vCore»			
Жесткий диск	 100 ГБ и более (SSD серверного уровня); Диск необходим, если требуется полноценная установка гипервизора на вычислительный узел и для локального сохранения конфигурации ВУ 		
Оперативная память	- 32 ГБ (ECC) DDR4 2133 МГц и выше (рекомендовано 64 ГБ и более для промышленной эксплуатации)		
Процессор	 Процессор с тактовой частотой от 2.0 ГГц: 8 ядер ЦП (для тестового и демонстрационного использования); 16 ядер и больше (для промышленного использования); Поддерживаются процессоры двух типов: AMD64 и Intel64. Процессоры должны иметь поддержку аппаратной виртуализации AMD-V или Intel VT. Также необходимо наличие атрибута NX bit¹; Поддерживаемые модели процессоров: AMD: Семейство процессоров EPYC с микроархитектурой Zen и выше; Intel: Семейство процессоров Xeon с микроархитектурой Skylake (6-го поколения) и выше 		
Сетевой адаптер	 Ethernet 1 Гбит/сек. (для тестового использования); Ethernet 10 Гбит/сек. (для промышленного использования) 		
Компонент управления			
Операционная система	 Astra Linux Special Edition сертификат № 2557 (выдан ФСТЭК России 27.01.2012, действителен до 27.01.2026); Альт 8 СП СЕРВЕР сертификат № 3866 (выдан ФСТЭК России 10.08.2018, действителен до 10.08.2028); Альт Сервер 9; РЕД ОС сертификат № 4060 (выдан ФСТЭК России 12.01.2019, действителен до 12.01.2024); РОСА «КОБАЛЬТ» сертификат № 4039 (выдан ФСТЭК России 07.12.2018, действителен до 07.12.2023) 		

¹ https://ru.wikipedia.org/wiki/NX_bit

Элемент	Параметр		
Процессор	 Процессор с тактовой частотой от 2.0 ГГц: 8 ядер ЦП (для тестового и демонстрационного использования); 16 ядер и больше (промышленного использования). Требования по архитектуре процессора: x86_64; Поддерживаемые модели: AMD: с микроархитектурой Zen и выше; Intel: с микроархитектурой Skylake (6-го поколения) и выше 		
Оперативная память	- 16 ГБ (ECC), DDR4 2133 МГц и выше (рекомендовано 24 ГБ и более для промышленной эксплуатации)		
Жесткий диск	- 2 x 100 ГБ в режиме RAID1 (SSD серверного уровня)		
Сетевой адаптер	 Ethernet 1 Гбит/сек. (для тестового использования); Ethernet 10 Гбит/сек. (для промышленного использования) 		
Клиент			
Браузер	 Яндекс браузер; Chromium-gost; Google Chrome; Microsoft Edge; Chromium; Firefox 		
Сетевой адаптер	- Ethernet 100 Мбит/сек		
Монитор	 - Диагональ от 17"; - Разрешение от 1280х1024 (4:3), от 1440х900 (16:9) 		
Периферийное оборудование	Клавиатура;Манипулятор типа мышь		

Установка Базис.vCore

Демонстрационная инсталляция Базис.vCore

Демонстрационная инсталляция платформы осуществляется на одном физическом узле. Дистрибутив предоставляется в ISO-образа, который включает в себя компоненты управляющей и вычислительной части.

🕜ажно

Образ демонстрационной инсталляции предоставляется отдельно по запросу.

- 1. Скачайте и установите ISO-образ.
- 2. Запустите гипервизор:

BASIS	vCore		
Начальное меню	RDO OpenStack Compute		
<mark>Цводка</mark> Настройка агента vCore Сеть Фебенати (источки источки сили сили сили сили сили сили сили си	Конфигуратор vCore 1.1.0 (2022-09-23 11:07) Гипервизор vCore 1.1.0-20220927		
дооавить/далить устроиства muitipatn Аутентификация Параметры оборилования и BIOS	Сетевые интерфейсы		
Виртуальные машины Журналирование Настройка сервисов Язык, клавиатура и часовой пояс Перезагрузка и выключение Комоар	Virtio network device (подключен) MRC-адрес FA:16:3E:1A:D8:AC Интерфейс eth0 IP-адрес		
Завершение сеанса	Нажмите <enter> для просмотра отпечатков ключей SSH этого узла.</enter>		
(Есс/Влево) Назад (Вверх/Вниз) Выбор	(Foter) Пторчатки комчей (F5) Обновить		
Coordination and the print of t			

Начальное меню

3. Перейдите в раздел "Консоль":



Консоль гипервизора

4. Инициализируйте БД:

/usr/bin/postgresql-setup --initdb

5. Откройте файл конфигурации БД - /var/lib/pgsql/data/pg_hba.conf:

nano /var/lib/pgsql/data/pg_hba.conf

6. Укажите параметры:

# TYPE	DATABASE	USER	ADDRESS	METHOD
host	all	all	127.0.0.1/32	md5

7. Запустите и добавьте в автозагрузку сервис БД:

systemctl start postgresql.service
systemctl enable postgresql.service

8. Создайте пользователя vcore и базу данных vcore :

• Переключитесь на пользователя postgres :

sudo -i -u postgres

• Запустите терминальный клиент для работы с БД:

psql

• Создайте пользователя vcore с указанным паролем:

create user vcore with password 'vcore';

• Создайте БД vcore :

create database vcore owner vcore;

• Выйдите из терминального клиента БД:

\q

• Выйдите из род пользователя postgres :

exit

9. Запустите и добавьте в автозагрузку сервис redis:

```
systemctl start redis
systemctl enable redis
```

10. Откройте файл конфигурации Broker - /opt/vcore/broker/config/

application.properties:

nano /opt/vcore/broker/config/application.properties

11. Укажите параметры:

```
quarkus.http.host=BROKER_IP_DNS_ADDR
provider.transport.use-multicast-discovery=false
provider.transport.static-discovery-addresses=tcp://BROKER_IP_DNS_ADDR:61617
provider.transport.bind-address=BROKER_IP_DNS_ADDR
%prod.quarkus.redis.hosts=redis://BROKER_IP_DNS_ADDR/
```

Где: BROKER_IP_DNS_ADDR - IP-адрес или DNS-имя узла, на котором установлен образ.

важно

При запуске образа в сетях с Dual Stack необходимо использовать IPv4-адреса в конфигурации.

12. Запустите Broker при помощи команды:

systemctl start vcore-broker

13. Добавьте сервис Broker в автозагрузку:

systemctl enable vcore-broker

14. Откройте файл конфигурации Control - /opt/vcore/control/config/

application.properties:

nano /opt/vcore/control/config/application.properties

15. Укажите параметры:

```
quarkus.http.host=CONTROL_IP_DNS_ADDR
broker-rest-api/mp-rest/url=http://AGENT_IP_DNS_ADDR:8081/api
quarkus.artemis.url=tcp://CONTROL_IP_DNS_ADDR:61616?
clientFailureCheckPeriod=5000&retryInterval=1000&reconnectAttempts=5
```

Где: CONTROL_IP_DNS_ADDR - IP-адрес или DNS-имя узла, на котором установлен образ. 16. Запустите Control при помощи команды:

systemctl start vcore-control

17. Добавьте сервис Control в автозагрузку:

systemctl enable vcore-control

18. Откройте файл конфигурации Agent - /opt/vcore/agent/config/application.properties:

nano /opt/vcore/agent/config/application.properties

19. Укажите параметры:

```
quarkus.http.host=AGENT_IP_DNS_ADDR
quarkus.artemis.url=tcp://AGENT_IP_DNS_ADDR:61616?
clientFailureCheckPeriod=5000&retryInterval=1000&reconnectAttempts=5&callTimeout=3
000
```

Где: AGENT_IP_DNS_ADDR - IP-адрес или DNS-имя узла, на котором установлен образ.

20. Запустите Agent при помощи команды:

systemctl start vcore-agent

```
21. Добавьте сервис Agent в автозагрузку:
```

systemctl enable vcore-agent

22. Подключитесь к панели управления. В адресной строке браузера введите DNS-имя или IP-адрес узла в формате - https://IP_DNS_ADDR:8082.

Простая инсталляция Базис.vCore

- Введение (см. стр. 12)
- Настройка окружения (см. стр. 12)
- Управляющая часть (см. стр. 12)
- Вычислительная часть (см. стр. 17)

Введение

Установка ПО осуществляется при помощи пакетов RPM.

Процесс установки включает:

- 1. Установка компонентов системы управления.
- 2. Установка и настройка вычислительного узла с агентом гипервизора.

Настройка окружения

Пакеты vCore расположены в репозитории проекта. Для включения этого репозитория нужно сделать следующее.

- 1. Создайте файл /etc/yum.repos.d/vcore-engine.repo.
- 2. В файле /etc/yum.repos.d/vcore-engine.repo укажите следующее:

```
[vcore-engine]
name=vCore Engine
baseurl=https://vcore-public:vcore-public@maven.tionix.ru/artifactory/vcore-rpm/
fedora/linux/$releasever/$basearch
enabled=1
gpgcheck=0
```

3. Запустите команду:

dnf update

Управляющая часть

Установка компонентов системы управления

Система управления в vCore состоит из двух компонентов: Broker и Control.

- Broker является функцию сервиса сообщений между Control и агентской частью. Зависим от внешнего сервиса Redis;
- Control предоставляет инструмент управления платформой в виде веб-приложения и иных функций. Control зависим от состояния сервиса Broker и СУБД PostgreSQL.

Установите Broker и Control :

```
dnf install vcore-broker vcore-control
```

Приложения Control и Broker будут установлены в каталогах /opt/vcore/control и /opt/vcore/ broker соответственно.

Настройка сервисов

По умолчанию сервисы слушают только loopback-интерфейс. При отсутствии прокси-сервера или балансировщика нагрузки можно поменять этот адрес, чтобы получить доступ до нужных сервисов.

Broker

Особенности настройки:

- Путь до конфигурации Broker: /opt/vcore/broker/config/application.properties
- Стандартные порты (основные):
- 8081/tcp порт сервиса обмена сообщениями.
- Зависит от сервиса Redis.

Основные настройки

Укажите IP-адрес или DNS-имя (рекомендуется) узла брокера в следующие параметры:

```
quarkus.http.host=BROKER_IP_DNS_ADDR
provider.transport.bind-address=BROKER_IP_DNS_ADDR
%prod.quarkus.redis.hosts=redis://localhost/
```

Для обнаружения сервисов брокер может использовать два сетевых механизма: *multicast* для динамической конфигурации и *unicast* для статической. По умолчанию рекомендуется использовать multicast.

Настройка multicast

Для обнаружения адресов сервиса обмена сообщениями через multicatst укажите следующие параметры в конфигурацию брокера:

```
provider.transport.use-multicast-discovery=true
provider.transport.multicast-interface-address=BROKER_IP_DNS_ADDR
```

Настройка unicast

Для обнаружения адресов сервиса обмена сообщениями через unicast укажите следующие параметры в конфигурацию брокера:

```
provider.transport.use-multicast-discovery=false
provider.transport.static-discovery-addresses=tcp://BROKER_IP_DNS_ADDR:61617
```

Укажите логин/пароль, который предназначен для входа в сервис обмена сообщениями. Значения используемые по умолчанию:

```
provider.transport.user=user
provider.transport.password=Q0hBTkdFTUUhIQ==
```

Где:

login – Логин.

password - Пароль в формате base64.

Пополнительно

```
При необходимости в параметре provider.transport.acceptor-params можно явно указать необходимый тип протокола: protocols=CORE,AMQP,STOMP.Пример:
```

```
provider.transport.acceptor-params=tcpSendBufferSize=1048576;tcpReceiveBufferSize
=1048576;amqpMinLargeMessageSize=102400;protocols=CORE,AMQP,STOMP;useEpoll=true
;amqpCredits=1000;amqpLowCredits=300;amqpDuplicateDetection=true
```

Доступные типы протоколов:

- CORE протокол ActiveMQ;
- AMQP протокол AMQP;
- STOMP протокол STOMP.

В проекте vCore используется только протокол CORE, а протоколы AMQP и STOMP могут использоваться внешними клиентами при необходимости. Допускается указание нескольких значений.

Control

Особенности настройки:

- Путь до конфигурации Control: /opt/vcore/control/config/application.properties
- Стандартные порты:
- 8082/tcp веб-интерфейс Control;
- Зависит от сервисов Broker и PostgreSQL.

Для Control нужно поменять адрес прослушивания сервиса:

quarkus.http.host=CONTROL_IP_DNS_ADDR

Укажите адрес подключения к СУБД PostgreSQL:

quarkus.datasource.jdbc.url=jdbc:postgresql://localhost:5432/vcore

Укажите адрес Broker API:

broker-rest-api/mp-rest/url=http://BROKER_IP_DNS_ADDR:8081/api

Укажите адрес службы очереди сообщений:

• при варианте multicast нужно указать адрес, прописанный в параметре provider.transport.multicast-group конфигурации брокера:

quarkus.artemis.url=udp://231.7.7.7:9876?
clientFailureCheckPeriod=5000&retryInterval=1000&reconnectAttempts=5

• при варианте unicast нужно указать адрес самого брокера:

```
quarkus.artemis.url=tcp://BROKER_IP_DNS_ADDR:61616?
clientFailureCheckPeriod=5000&retryInterval=1000&reconnectAttempts=5
```

Укажите логин/пароль для взаимодействия сервисов. Значения используемые по умолчанию:

quarkus.artemis.username=user
quarkus.artemis.password=Q0hBTkdFTUUhIQ==

Задайте необходимые значения:

```
quarkus.artemis.username=login
quarkus.artemis.password=password
```

Где:

- login Логин.
- password Пароль в формате base64.

PostgreSQL

1. Инициализируйте БД:

/usr/bin/postgresql-setup --initdb

2. Откройте файл конфигурации БД - /var/lib/pgsql/data/pg_hba.conf:

nano /var/lib/pgsql/data/pg_hba.conf

3. Укажите параметры:

# TYPE	DATABASE	USER	ADDRESS	METHOD
host	all	all	127.0.0.1/32	md5

Сли PostgreSQL будет доступен по сети, то вместо подсети 127.0.0.1/32 нужно прописать подсеть, с которой Control будет подключаться к СУБД или подсеть /32 только с адресом Control явно.

4. Запустите и добавьте в автозагрузку сервис БД:

systemctl start postgresql.service
systemctl enable postgresql.service

- 5. Создайте пользователя vcore и базу данных vcore :
 - Переключитесь на пользователя postgres :

sudo -i -u postgres

• Запустите терминальный клиент для работы с БД:

psql

• Создайте пользователя vcore с указанным паролем:

create user vcore with password 'vcore';

• Создайте БД vcore :

create database vcore owner vcore;

• Выйдите из терминального клиента БД:

\q

• Выйдите из под пользователя postgres:

exit

Redis

важно

Поддерживается Redis версий 7.0 и выше.

Установите сервис:

dnf -y install redis

Запустите и добавьте в автозагрузку сервис:

```
systemctl start redis
systemctl enable redis
```

Запуск сервисов

Фажно При запуске сервисов в сетях с Dual Stack необходимо использовать IPv4-адреса в конфигурации.

Для запуска компонентов системы управления достаточно старта нужных юнитов systemd:

```
systemctl enable vcore-control vcore-broker
systemctl start vcore-broker vcore-control
```

Control и Broker могут достаточно долго инициализироваться, поэтому в systemd включен режим нотификации готовности сервиса к работе.

Примечание

При использовании FirewallD в качестве системы управления Firewall необходимо открыть порты сервисов. Для этого используйте следующие команды:

```
firewall-cmd --zone=public --add-service=vcore-broker --permanent
firewall-cmd --zone=public --add-service=vcore-control --permanent
firewall-cmd --zone=public --add-service=vcore-balancer --permanent
```

Перезапустите Firewall:

firewall-cmd --reload

Для проверки портов выполните:

nc -v IP_ADDR PORT

Где:

```
• IP_ADDR - IP-адрес узла;
```

• PORT - Порт сервиса.

Список портов сервисов по умолчанию доступен по ссылке (см. стр. 54).

Настройка панели управления

После запуска Control по адресу http://CONTROL_IP_DNS_ADDR:8082 будет доступен веб-интерфейс платформы. Укажите первичные данные для пользователя admin:

Для корректной работы системы рекомендуется использовать Google Chrome или Firefox актуальных версий.

BÁSIS
Имя пользователя
admin
Фамилия
Mass .
Отчество
Email
Пароль
Подтверждение
Сохранить

Форма создания пользователя

Заполните поля формы создания пользователя. Подтвердите создание кнопкой "Сохранить". После этого укажите следующие данные для входа в систему:

- домен: master;
- пользователь: admin;
- пароль: указанный для пользователя admin.

Вычислительная часть

Установка гипервизора

(Србраз гипервизора в формате ISO доступен по этой ссылке².

Загрузите образ гипервизора в систему. Образ гипервизора предоставляется отдельно.

Запустите систему из полученного образа:



Меню запуска системы

Выберите первый пункт запуска системы.

Дождитесь завершения загрузки системы и ознакомьтесь с лицензионным соглашением:

² https://vcore-public:vcore-public@maven.tionix.ru:443/artifactory/vcore-hyper-os/



Лицензионное соглашение Базис vCore

Примите лицензионное соглашение при помощи клавиши Enter и перейдите в главное меню:



Главное меню гипервизора

В открывшейся вкладке с отобразившимся меню установки, в первую очередь ознакомьтесь с конфигурацией сети по умолчанию. Для этого выберите опцию меню установки "Сеть" и перейдите в раздел «Сеть», нажав клавишу Enter:



Параметры сети гипервизора

При необходимости установите следующие параметры:

- Настройка сетевых интерфейсов;
- Виртуальные сети (VLAN);
- Bond-интерфейсы;
- Серверы DNS;
- Синхронизация времени (NTP);
- Проверка сети;
- Управление состоянием сети.

🖗ажно

При запуске гипервизора в сети с Dual Stack необходимо использовать IPv4-адреса в конфигурации.

Для установки гипервизора выберите пункт «Установка Гипервизора»:



Меню установки гипервизора

Выберите тип установки:

	BASIS vCore	
Меню установки Установка Гипервизора		
<mark>Установка Гипервизора</mark> Сводка Сеть Журналирование Перезагрузка сервера Выключение сервера Консоль Язык интерфейса	Нажмите <enter> для установки Гипервизора на диск.</enter>	
	Выбор типа установки <mark>Установка на локальный диск</mark> Установка на удаленный диск Установка на RAID <enter> OK <esc> Отмена</esc></enter>	
<esc влево=""> Назад <Вверх/Вниз> Выбор</esc>		

Меню выбора типа установки

Поддерживаемые типы:

- Установка на локальный диск;
- Установка на удаленный диск;
- Установка на RAID. Поддерживаемый тип RAID 1.

Выберите диск для установки:



Меню выбора диска

Подтвердите очистку выбранного диска:

п

Конфигуратор vCore 1.1.2	14:53:18	tvc-hv-fd291728
	BASIS vCore	
Меню установки	Установка Гипервизора	
Установка Гипервизора Сводка	Hажмите <enter> для установки Гипервизора на д</enter>	циск.
сводка Сеть Жирналирование		
Перезагрузка сервера Выключение сервера		
Консоль Язык интерфейса		
	Выбор диска	
Все данные, которые	е ранее были на диске, будут безвозвратно удалены. Продолжить?	
<esc> Нет <f8> Да</f8></esc>		
	CEnter> OK (Esc) Отмена	
<esc влево=""> Назад <Вверх/Вниз> Выбор</esc>		

Подтверждение очистки диска

Дождитесь завершения процесса установки:



Сообщение о завершении процесса установки

Подтвердите перезагрузку системы:

	BH515	vuore	
Меню установки		Установка Гипервизора	
Установка Гипервизора Сводка		Hажмите <enter> для установки Гипере</enter>	зизора на диск.
Сеть Журналирование			
Перезагрузка сервера Выключение сервера Консоль			
Язык интерфейса			
	Вы хотите переза	пустить сервер?	
	<esc> Нет <f8> Да</f8></esc>		
(Lsc/Влево) Назад (Вверх/Вни	з> Выбор		

Подтверждение перезагрузки системы

После завершения процесса перезагрузки необходимо установить пароль к системе:



Указание пароля к системе

Убедитесь, что пароль успешно изменен:

		VLORE
Начальное меню Сводко Хзык и раскладка клавиатуры		QEMU Standard PC (Q35 + ICH9, 2009) Конфигуратор TVC 1.0.0 (2022-06-29 10:38)
		Типервизор Тус 1.0.0-20220701
		Сетевые интерфейсы
		82574L Gigabit Network Connection (подключен) MRC-адрес 52:54:00:90:35:03 Интерфейс eth0 IP-адрес 192.168.122.119
		Нажните ‹Enter› для просмотра отпечатков ключей SSH этого узла.
	Пароль успе	ешно изменен
	<enter> OK</enter>	
<enter> ОК <Вверх/Вниз> Выбор</enter>		<enter> Отпечатки ключей <f5> Обновить</f5></enter>

Сообщение об успешном изменении пароля

Перейдите в меню "Авторизация" и укажите пароль для входа в систему:

	BASIS	vCore	
Начальное меню		Авторизация	
Сводка Язык и раскладка клавиатуры <mark>Авторизация</mark>		Наммите <enter> для авторизации.</enter>	
	Пароль		
	<enter> Далее/ОК <esc> Отмена</esc></enter>		
<enter> ОК <Вверх/Вниз> Выбор</enter>		<enter> Авторизоваться</enter>	

Окно входа в систему

Установка гипервизора завершена, доступны все меню настроек системы.

Настройка агента гипервизора

Реред добавлением гипервизора в vCore убедитесь в том, что в систему добавлен пользователь admin. В противном случае агент не сможет найти данные основного домена master и закончит работу с ошибкой подключения к AMQP.

После перезагрузки системы откроется меню настроек гипервизора:

Основные настройки агента vCore						
Обязательные настро	Обязательные настройки					
ID датацентра ID узла ID домена Адрес агента Порт агента Адрес подключения	81d42d15-cb75-4e6d-8e26-2e083ddf8283 d5a6d503-b655-487e-ab44-baf281c20c47 bd92c60e-d7d5-11ea-87d0-0242ac130003 127.0.0.1 8083 tcp://localhost:61616					
пеорязательные наст	роики					
Имя пользователя Пароль Пароль (еще раз)	USer ********* ********					
Скрытые сопа-интерф Скрытые сети VLAN	скрытые вопа-интерфеисы Скрытые сети VLAN					
Скрытые интерфейсы	Скрытые интерфейсы					
<enter> Далее/ОК <esc> Отмена <f8> ОК</f8></esc></enter>						

Меню настройки гипервизора

Укажите параметры запуска гипервизра:

- Обязательные:
 - ID дата-центра Уникальный идентификатор дата-центра в формате UUID, к которому будет подключен вычислительный узел;
 - ID узла Уникальный идентификатор узла гипервизора в формате UUID;
 - ID домена Уникальный идентификатор домена в формате UUID, к которому будет подключен вычислительный узел;
 - Адрес агента IP-адрес или имя узла гипервизора;
 - Порт агента Номер порта агента. Значение по умолчанию 8083.

- Адрес подключения Адрес службы очереди сообщений:
 - при варианте multicast нужно указать адрес прописанный в параметре provider.transport.multicast-group конфигурации брокера, для подключения по протоколу UDP:

```
udp://231.7.7.7:9876?
clientFailureCheckPeriod=5000&retryInterval=1000&reconnectAttempts=5
```

 при варианте unicast нужно указать адрес самого брокера для подключения по протоколу TCP:

```
tcp://BROKER_IP_DNS_ADDR:61616?
clientFailureCheckPeriod=5000&retryInterval=1000&reconnectAttempts=5
```

 при unicast если брокеров несколько, то необходимо в скобках указать адреса всех брокеров:

```
(tcp://10.18.0.93:61616,tcp://10.18.0.115:61616,tcp://
10.18.0.120:61616)?
clientFailureCheckPeriod=5000&retryInterval=1000&reconnectAttempts=5
```

- Необязательные:
 - Имя пользователя Имя пользователя подключения к шине;
 - Пароль Пароль пользователя для подключения к сервису сообщений. Указывается в формате base64;
 - Подтверждение пароля Подтверждение пароля пользователя для подключения к шине. Указывается в формате base64;
 - · Скрытые bond-интерфейсы Исключаемые bond-интерфейсы;
 - Скрытые сети VLAN Исключаемые VLAN;
 - Скрытые интерфейсы Исключаемые NIC.

Сохраните настройки при помощи клавиши Enter. В открывшемся окне подтвердите перезапуск сервиса:

—————————————————————————————————————					
Обязательные настройки					
ID датацентра ID узла ID домена Адрес агента Порт	81d42d15-cb75-4e6d-8e26-2e083ddf8283 d5a6d503-b655-487e-ab44-baf281c20c47 bd92c60e-d7d5-11ea-87d0-0242ac130003 10.38.7.55				
Адрес Необя Сервис агент	Настройки сохранены. а и сервис sanlock будет перезапущены.				
Имя п <esc> Нет <f8> Да Парол Пароль (еще раз) ******** Скрытые bond-интерфейсы Скрытые сети VLAN Скрытые интерфейсы</f8></esc>					
<enter> Далее/ОК <esc> Отмена <f8> ОК</f8></esc></enter>					

Меню перезапуска гипервизора

▲ обые изменения в конфигурационном файле требуют перезапуска агента. При изменении ID узла требуется изменение параметра в sanlock и перезапуск sanlock-сервиса. В связи с этим перед сохранением и перезапуском сервисов выводится соответствующее сообщение об изменении настроек и запросом на перезапуск сервисов:



Убедитесь, что агент успешно запустился:



Сообщение об успешном запуске гипервизора

Настройка multipath

Данная процедура необходима для корректного подключения iSCSI-устройств к платформе Базис.vCore.

Скопируйте конфигурационный файл multipath:

cp /opt/multipath.conf.bak /etc/multipath.conf

Перезапустить сервис:

systemctl restart multipathd.service

🖗ажно

После установки и настройки на вычислительном узле ПО "Базис.vCore" узел автоматически подключится к платформе со статусом "Обслуживание". Для ввода узла в эксплуатацию с таким состоянием его необходимо активировать, добавив его в один из кластеров. Данная процедура подробна описана в руководстве Администратора vCore.

Инсталляция с поддержкой высокой доступности Базис.vCore

- Введение (см. стр. 27)
- Предварительная настройка (см. стр. 27)
 - Настройка системных параметров (см. стр. 27)
 - Настройка точного времени (см. стр. 27)
- Настройка узлов кластера (см. стр. 28)
 - Настройка доменных имен узлов кластера (см. стр. 28)
 - Установка и настройка Pacemaker (см. стр. 28)
 - Установка пакетов Pacemaker (см. стр. 29)
 - Настройка аутентификации и запуск компонентов кластера (см. стр. 29)
 - Перезапуск узлов в кластере (см. стр. 29)
 - Включение функций фенсинга узлов (см. стр. 30)
 - Создание VIP-ресурсов для сервисов (см. стр. 31)
 - Установка PostgreSQL (см. стр. 31)
 - Установка пакетов (см. стр. 31)
 - Настройка кластера PostgreSQL (см. стр. 32)
 - Настройка master-узлов (см. стр. 32)
 - Настройка slave-узлов (см. стр. 34)
 - Настройка кластера Pacemaker (см. стр. 35)
- Настройка Redis и сервиса управления кластером Redis Sentinel (см. стр. 35)
 - Базовая информация о настройке сервисов (см. стр. 36)
 - Все узлы (см. стр. 36)
 - Мастер-узел (см. стр. 36)
 - Настройка Redis (см. стр. 36)
 - Настроойка Sentinel (см. стр. 37)
 - Узлы репликации (см. стр. 38)
 - Проверка работы репликации (см. стр. 39)
 - Включение сервисов Redis и Sentinel в Pacemaker (см. стр. 39)
- Установка vCore (см. стр. 40)
 - Управляющая часть (см. стр. 40)
 - Настройка окружения (см. стр. 40)
 - Установка балансировщика нагрузки (см. стр. 40)
 - Установка и настройка брокера (см. стр. 42)
 - Установка и настройка контрола (см. стр. 44)
 - Настройка сервисов vCore (см. стр. 45)
 - Настройка кластера Pacemaker (см. стр. 45)

- Вычислительная часть (см. стр. 45)
 - Настройка панели управления (см. стр. 46)

Введение

Установка Базис. vCore в режиме высокой доступности подразумевает наличие нескольких управляющих и вычислительных узлов (минимум по 3). Процесс установки включает в себя:

- Установка компонентов системы управления:
 - Broker выполняет функции брокера сообщений, зависит от сервиса Redis;
 - Control предоставляет инструмент управления платформой в виде веб-приложения и иных функций, зависит от сервиса PostgreSQL.
 - Balancer выполяет функции балансировки нагрузки для сервисов control и broker.
- Установка и настройка гипервизора.

Все шаги по установке и настройке необходимо выполнять на всех узлах кластера, если не указано иное.

Предварительная настройка

Для корректной работы vCore требуются некоторые изменения в настройках OC. Все указанные в этом разделе команды должны быть выполнены во всех узлах кластера.

Настройка системных параметров

Для корректной работы узлов кластера необходимо добавить некоторые изменения в системные параметры ОС.

Для работы VIP-адресов включите возможность привязки сервисов к отсутствующему на данный момент IP-адресу в ОС:

```
echo net.ipv4.ip_nonlocal_bind=1 > /etc/sysctl.d/99-cluster.conf; \
systemctl restart systemd-sysctl; \
sysctl net.ipv4.ip_nonlocal_bind
```

Настройка точного времени

Во всех узлах кластера требуется настроить NTP-клиент с синхронизацией через один и тот же сервер или пул серверов.

Установите сервис chrony:

dnf -y install chrony

Откройте файл /etc/chrony.conf.

Если вы ходите указать свой пул серверов, воспользуйтесь директивой pool:

pool your.ntp.servers.pool.org iburst

Для указания конкретного сервера следует указать директиву server:

server your.ntp.server.org iburst

Сохраните изменения.

Перезапустите сервис chrony и добавьте сервис в автозагрузку:

```
systemctl restart chrony; \
systemctl enable chrony
```

В качестве дополнения проверьте часовой пояс, указанный в ОС:

timedatectl | grep zone

Если указанный пояс не совпадает с вашим, то нужно указать корректный. Например, для Москвы (GMT+3):

timedatectl set-timezone Europe/Moscow

Убедитесь, что пояс успешно изменился с помощью команды из предыдущего шага.

Настройка узлов кластера

В кластерном варианте установки узлы управления настраиваются в определенной последовательности. Дальнейшие подразделы объяснят подробности выполнения каждого шага.

Настройка доменных имен узлов кластера

Для кластера сервисов vCore необходимо использовать имена узлов вместо IP-адресов. Рекомендуется использовать отдельный DNS-сервер, но если это технически невозможно, то допустимо внести соответствующие записи в файл /etc/hosts.

Вначале для примера примем следующие данные узлов, которые будут использованы далее в этой статье:

- первый узел кластера:
 - имя узла: vcore-control-1
 - ІР-адрес: 10.24.24.20
- второй узел кластера:
 - имя узла: vcore-control-2
 - ІР-адрес: 10.24.24.31
- третий узел кластера:
 - имя узла: vcore-control-3
 - ІР-адрес: 10.24.24.202

Также нужно создать имена для двух VIP-адресов: для PostgreSQL и сервисов vCore. Для примера примем следующее:

- VIP-адрес для PostgreSQL:
 - имя для адреса: pgsql-vip
 - ІР-адрес: 10.24.24.70
- VIP-адрес для сервисов vCore:
 - имя для адреса: vcore-vip
 - ІР-адрес: 10.24.24.80

При таких данных файл /etc/hosts должен выглядеть так:

```
10.24.24.20vcore-control-110.24.24.31vcore-control-210.24.24.202vcore-control-310.24.24.70pgsql-vip10.24.24.80vcore-vip
```

Аля корректного распознавания указанных в /etc/hosts имён узлов убедитесь, что они вначале ищутся в файлах hosts и только потом - DNS. Для этого откройте файл /etc/ nsswitch.conf и проверьте поиск имён для hosts:

hosts: files dns myhostname

files должно находиться на первом месте.

Измените файл /etc/hosts во всех узлах кластера.

Проверьте доступность узлов кластера по именам, например, vcore-control-1:

ping vcore-control-1

Установка и настройка Pacemaker

Для установки Pacemaker необходимо выполнить ряд операций.

Установка пакетов Pacemaker

Установите пакеты Pacemaker:

```
dnf -y install pacemaker corosync pcs
```

Добавьте разрешение на использование портов Pacemaker в сервисе Firewalld:

```
firewall-cmd --permanent --add-service=high-availability ; \
firewall-cmd --reload
```

Запустите сервис pcsd и добавьте его в автозапуск:

```
systemctl start pcsd.service
systemctl enable pcsd.service
```

Настройка аутентификации и запуск компонентов кластера

Для объединения узлов в кластер Pacemaker вначале необходимо настроить аутентификацию кластера.

Укажите пароль для пользователя hacluster. Пароль для пользователя hacluster для всех узлов должен совпадать:

passwd hacluster

Далее команды pcs нужно выполнять только на одном узле, изменения будут засинхронизированы автоматически.

Произведите аутентификацию узлов кластера. При выполнении команды нужно указать пользователя и пароль, который был настроен в прошлом шаге:

pcs host auth vcore-control-1 vcore-control-2 vcore-control-3

Создайте кластер:

```
pcs cluster setup cluster vcore-control-1 vcore-control-2 vcore-control-3
```

Сообщения ' No addresses specified for host..." не являются ошибками, их можно игнорировать.

Запустите кластер:

pcs cluster start --all

Проверьте доступность кластера:

pcs status

Перезапуск узлов в кластере

Для сервисов Pacemaker рекомендуется не включать автозапуск сервисов Pacemaker. Поэтому при штатной перезагрузке узла кластера требуется выполнить ряд несложных команд. Также отметим, узлы нужно перезапускать только по одному.

Остановите узел кластера, который вы хотите перезапустить:

```
pcs cluster stop vcore-control-1
```

Дождитесь переезда всех ресурсов с этого узла на другие через команду

```
pcs status
```

После этого штатно перезапустите узел. Когда узел будет доступен по сети, снова запустите сервисы Pacemaker в нём: pcs cluster start vcore-control-1

Включение функций фенсинга узлов

Включение STONITH и определение политики при отсутствии кворума

Для вывода из эксплуатации нестабильных или вышедших из строя узлов в Pacemaker используется технология STONITH. Суть технологии в том, чтобы по независимому каналу, в частности, по протоколу IPMI³ для BMC⁴, в случае с проблем с узлом можно было перезапустить или выключить узел. Для корректной работы STONITH требуется как минимум три узла и активация политики отсутствия кворума. Сам процесс вывода с последующей миграцией ресурсов называется фенсингом (fencing).

Для включения кворума и STONITH укажите параметры сервиса:

pcs property set stonith-enabled=true

Кворум в Расетакет проверяет, сколько узлов функционирует в кластере. При наличии строго больше 50% работающих узлов Расетакег считает, что кластер находится в работоспособном состоянии (то есть имеется кворум). Однако если это правило не выполняется, Расетакег применяет одну из политик отсутствия кворума: остановка всех узлов (stop), freeze (запрет на перемещение ресурсов без остановки ресурсов), suicide (выключение всех узлов) и ignore (продолжение работы всех ресурсов). Для сервисов vCore предлагается использовать политикy stop:

pcs property set no-quorum-policy=stop

В таком варианте безопасно временно терять один узел из трёх, при потере двух узлов Pacemaker выключит все ресурсы.

Дополнительно рекомендуем, чтобы STONITH по умолчанию выключал вышедшие из строя узлы кластера (по умолчанию STONITH их пытается перезапустить):

pcs property set stonith-action=poweroff

Добавление данных IPMI для узлов в кластере

В качестве стандартного метода фенсинга рекомендуется использовать протокол IPMI.

На всех узлах, где запущены компоненты Pacemaker, установите пакет фенсинга для IPMI:

dnf -y install fence-agents-ipmilan

Для включения ресурсов для фенсинга через IPMI необходимо использовать команду pcs stonith. Для каждого узла нужен свой pecypc stonith. Предлагаемые настройки для pecypca:

pcs stonith create vcore_ipmi_vcore-control-1 fence_ipmilan pcmk_host_list="vcorecontrol-1" ip=10.23.10.3 lanplus=true username=ADMIN password=ADMIN auth=MD5

Нужно обратить внимание на следующее:

- Так как для каждого узла требуется создать pecypc stonith, после create в имени pecypca укажите, к какому узлу относится pecypc и его укажите в параметре pcmk_host_list;
- В ір укажите адрес IPMI-сервера. Если используется нестандартный порт, то вместо ір укажите параметр іррогt и укажите порт через двоеточние после адреса (напр. 10.23.10.3:6623);
- По умолчанию предлагается использование IPMI 2.0 через параметр lanplus;
- Остальные параметры определяют данные аутентификации со стандартным методом хэширования пароля MD5.

По умолчанию возможна ситуация, когда pecypc stonith может запуститься на том же узле, ВМС которого контролируется этим ресурсом. Для запрета такой ситуации требуется добавить соответствующее ограничение для каждого из pecypcoв stonith:

pcs constraint location vcore_ipmi_vcore-control-1 avoids vcore-control-1=INFINITY

³ https://ru.wikipedia.org/wiki/Intelligent_Platform_Management_Interface

⁴ https://www.servethehome.com/explaining-the-baseboard-management-controller-or-bmc-in-servers/

Создание VIP-ресурсов для сервисов

В схеме предполагается использование двух VIP-адресов: отдельно для PostgreSQL и сервиса балансировки vCore. Их необходимо создать до установки этих сервисов.

PostgreSQL

Для единой точки входа в СУБД необходимо настроить pecypc Pacemaker с VIP-адресом. Выделенный узел для VIP определит начальный master-узел PostgreSQL.

Добавьте VIP-ресурс в Pacemaker для PostgreSQL:

pcs resource create pgsql-vip ocf:heartbeat:IPaddr2 ip=10.24.24.70 cidr_netmask=32 op monitor interval=10s

Проверьте статус кластера:

pcs status

Вы должны увидеть, что ресурс успешно создался и запустился на одном из узлов кластера Pacemaker:

```
...
Node List:
 * Online: [ vcore-control-1 vcore-control-2 vcore-control-3 ]
Full List of Resources:
 * pgsql-vip (ocf::heartbeat:IPaddr2): Started vcore-control-1 <- Запущенный
pecypc VIP для PostgreSQL</pre>
```

vCore

Отдельный VIP нужно создать для сервиса балансировки vCore.

Создайте VIP-ресурс для vCore:

```
pcs resource create vcore-vip ocf:heartbeat:IPaddr2 ip=10.24.24.80 cidr_netmask=32 op
monitor interval=10s
```

Получите статус ресурсов и определите, на каком узле запустился VIP-адрес:

pcs status

Пример ответа:

```
...
Node List:
 * Online: [ vcore-control-1 vcore-control-2 vcore-control-3 ]
Full List of Resources:
 * pgsql-vip (ocf::heartbeat:IPaddr2): Started vcore-control-1
 * vcore-vip (ocf::heartbeat:IPaddr2): Started vcore-control-1
```

На время разворачивания платформы желательно, чтобы эти сервисы находились на одном узле. Если ресурсы запустились на разных, то воспользуйтесь командой move:

pcs resource move vcore-vip vcore-control-1

Установка PostgreSQL

Установка пакетов

Добавьте репозиторий в /etc/yum.repos.d/resource-agents.repo:

```
[PGSQL-resource]
```

```
name=resource-agents
baseurl=https://ftp.postgresql.org/pub/repos/yum/common/fedora/fedora-37-x86_64/
enabled = 1
gpgcheck = 0
```

Установите пакеты:

dnf -y install resource-agents-paf postgresql-server postgresql-contrib

Настройка кластера PostgreSQL

Для настройки кластера PostgreSQL необходимо выполнить ряд операций.

Добавьте порты сервисов в firewall:

```
firewall-cmd --permanent --add-service=postgresql; \
firewall-cmd --reload
```

Настройка master-узлов

🕜ажно

Следующие операции выполняются на master-узле кластера. Master-узел определяется по тому, в каком узле поднялся VIP-ресурс pgsql-vip в Pacemaker. Далее считается, что master-узлом является vcore-control-1.

Зайдите под пользователем postgres:

su - postgres

Укажите имя узла и имя VIP-адреса PostgreSQL, в примере pgsql-vip, в ~postgres/standby.conf:

```
cat <<EOF > ~postgres/standby.conf
primary_conninfo = 'host=pgsql-vip application_name=vcore-control-1'
EOF
```

Укажите имя узла и имя VIP-адреса PostgreSQL, в примере pgsql-vip, в ~postgres/listen.conf:

```
cat <<EOF > ~postgres/listen.conf
listen_addresses = 'pgsql-vip, vcore-control-1'
EOF
```

Инициализируйте БД:

/usr/bin/postgresql-setup --initdb

Добавьте параметры в postgresql.conf:

```
cat <<EOF>> ~postgres/data/postgresql.conf
hba_file = '/var/lib/pgsql/pg_hba.conf'
wal_keep_size = 100
include = '../standby.conf'
include = '../listen.conf'
EOF
```

Удалите конфигурацию pg_hba.conf:

```
rm ~postgres/data/pg_hba.conf
```

Сформируйте конфигурацию pg_hba.conf, вписав точное имя хоста и имя vip aдреса pgsql, в примере pgsql-vip:

```
cat <<EOF > ~postgres/pg_hba.conf
host all
                all
                                         trust
                all
                         127.0.0.0/8
                                         trust
host all
                all
                        ::1/128
                                         trust
# forbid self-replication from VIP
host replication postgres pgsql-vip
                                        reiect
# forbid self-replication its own IP
local replication all
                                         reject
host replication all
                        vcore-control-1 reject
host replication all127.0.0.0/8rejecthost replication all::1/128reject
# allow any standby connection
host replication postgres 10.24.24.0/24 trust
host vcore
               vcore 10.24.24.0/24 md5
EOF
```

Здесь нужно:

- В строке "host replication postgres pgsql-vip reject" указать VIP-адрес или имя домена VIP-адреса;
- В строке "host replication all vcore-control-1 reject" указать фактический адрес узла или имя домена для этого адреса;
- В строке "host vcore vcore 10.24.24.0/24 md5" нужно указать подсеть, через которую клиенты (в частности, сервисы vCore) могут подключиться к СУБД. Возможно указание адресов сервисов по строке явно через указание подсети /32.

Осуществите выход из под пользователя postgres:

```
exit
```

Запустите PostgreSQL:

```
systemctl start postgresql.service
```

Проверьте статус PostgreSQL:

```
systemctl list-units |grep postgre
postgresql.service
PostgreSQL database server
```

loaded active running

Зайдите под пользователем postgres:

su - postgres

Запустите терминальный клиент для работы с БД:

psql

Внутри сессии psql создайте пользователя vcore с указанным паролем:

create user vcore with password 'УКАЖИТЕПАРОЛЬ';

Создайте БД vcore:

create database vcore owner vcore;

Выйдите из терминального клиента БД:

\q

Осуществите выход из под пользователя postgres:

exit

Настройка slave-узлов

важно

Следующие операции выполняются на slave-узлах кластера.

Зайдите под пользователем postgres:

su - postgres

Укажите имя узла и имя VIP-адреса PostgreSQL, в примере **pgsql-vip**, в ~postgres/standby.conf:

```
cat <<EOF > ~postgres/standby.conf
primary_conninfo = 'host=pgsql-vip application_name=vcore-control-3'
EOF
```

Укажите имя узла и имя VIP-адреса PostgreSQL, в примере pgsql-vip, в ~postgres/listen.conf:

```
listen_addresses = 'pgsql-vip, vcore-control-3'
```

Сформируйте конфигурацию pg_hba.conf, вписав точное имя хоста и имя VIP-адреса pgsql:

```
cat <<EOF > ~postgres/pg_hba.conf
host all
host all
host all
# forb
                                                 trust
                               0.0.0.0/0
                                                 trust
                   all
                              ::/0
                                                 trust
# forbid self-replication from VIP
host replication postgres pgsql-vip reject
# forbid self-replication its own IP
reject
vcore-control-3 reject
host replication all 127.0.0.0/8 reject
host replication all ::1/128
# allow any star
# allow any standby connection
host replication postgres 0.0.0.0/0
                                                 trust
host vcore vcore 10.24.24.0/24 md5
EOF
```

Здесь нужно:

- В строке "host replication postgres pgsql-vip reject" указать VIP-адрес или имя домена VIP-адреса;
- В строке "host replication all vcore-control-1 reject" указать фактический адрес узла или имя домена для этого адреса;
- В строке host vcore vcore 10.24.24.0/24 md5 нужно указать подсеть, через которую клиенты (в частности, сервисы vCore) могут подключиться к СУБД. Возможно указание адресов сервисов по строке явно через указание подсети /32.

Удалите ненужные файлы из каталога:

rm -rf /var/lib/pgsql/data/*

Создайте репликацию из главного (master) узла в текущий (slave) узел:

pg_basebackup -h pgsql-vip -D ~postgres/data/ -X stream -P

Пометьте узел как slave:pg_basebackup -h pgsql-vip -D ~postgres/data/ -X stream -P

touch ~postgres/data/standby.signal

Выйдите из пользователя postgres:

exit

Запустите узел slave:

```
systemctl start postgresql
```

Проверьте статус узла:

systemctl status postgresql

Повторите операции на оставшихся slave-узлах.

Настройка кластера Pacemaker

Теперь когда PostgreSQL находится в master-slave конфигурации, необходимо настроить автоматическое переключение с помощью Pacemaker.

Отключите PostgreSQL. Сначала на slave-узлах, затем на master:

systemctl disable --now postgresql

Coздайте pecypc postgresql:

```
pcs resource create pgsqld ocf:heartbeat:pgsqlms bindir=/usr/bin pgdata=/var/lib/pgsql/
data op start timeout=60s op stop timeout=60s op promote timeout=30s op demote
timeout=120s op monitor interval=15s timeout=10s role="Master" op monitor interval=16s
timeout=10s role="Slave" op notify timeout=60s promotable notify=true
```

Количество master-узлов СУБД не должно превышать 1, добавьте эту настройку в метаданные ресурса Pacemaker:

```
pcs resource meta pgsqld master-max=1
```

Настройте зависимости ресурсов:

```
pcs constraint colocation add pgsql-vip with Promoted pgsqld-clone INFINITY
pcs constraint order promote pgsqld-clone then start pgsql-vip symmetrical=false
kind=Mandatory
pcs constraint order demote pgsqld-clone then stop pgsql-vip symmetrical=false
kind=Mandatory
```

Проверьте статус кластера pgsql в pacemaker. Master нода должна быть только на одном хосте:

```
pcs status
...
* pgsql-vip (ocf::heartbeat:IPaddr2): Started control-03
 * vcore-vip (ocf::heartbeat:IPaddr2): Started control-02
 * Clone Set: pgsqld-clone [pgsqld] (promotable):
 * Masters: [ control-03 ]
 * Slaves: [ control-01 control-02 ]
Failed Resource Actions:
 * pgsqld_promote_0 on control-01 'error' (1): call=45, status='complete', exitreason='
control-03 is the best candidate to promote, aborting current promotion', last-rc-
change='Mon Sep 4 12:48:24 2023', queued=0ms, exec=487ms
```

Настройка Redis и сервиса управления кластером Redis Sentinel

Для задач брокера и для хранения некоторых данных платформы (например, блокировок ВМ) необходимо установить и настроить сервис кластер Redis в режиме Master-Replica. Также для управления ролями инстансов Redis необходимо настроить сервис управления кластером Redis под названием Sentinel. Настройка будет произведена вначале в мастер-узле, потом в узлах репликации. В качестве мастер-узла выберите узел, где мастером был запущен инстанс PostgreSQL.

Базовая информация о настройке сервисов

Перед описанием фактической настройки сервисов Redis и Sentinel необходимо обратить внимание на пару моментов.

Redis (но не Sentinel) будет настроен в режиме protected-mode. В этом режиме вход ограничивается только с указанием данных пользователей, которые находятся в ACL внутри базы Redis (даже для localhost). Также ограничения будут и на прямой запуск команд через стандартную командную строку Linux, для выполнения команд необходимо вначале заходить в окружение сервисов. В качестве пользователя по умолчанию для входа будет использован пользователь default.

Сервисы Sentinel будут настроены в режиме выключенного protected-mode, иначе основные функции управления репликацией Redis работать не будут в силу отсутствия у Sentinel объектов ACL. Sentinel использует данные Redis ACL для входа в Redis. У самого Sentinel есть свои ACL, однако при выключенном protected-mode любой клиент сможет без аутентификации получить функции сервиса Sentinel. Поэтому важно, чтобы порт 26379 должен быть открыт только для адресов брокера vCore.

Все узлы

Для корректной работы репликации между инстансами Redis в ОС всех узлов кластера нужно включить оверкоммитинг памят⁵и, то есть возможность выделения участков памяти больше, чем имеется в ОЗУ и свободно для выделения процессу:

```
echo vm.overcommit_memory = 1 > /etc/sysctl.d/99-redis.conf
systemctl restart systemd-sysctl
```

Настройка зоны фаервола в Firewalld

Доступ до Redis и Sentinel по соображениям безопасности следует ограничить адресами сети управления (особенно это касается Sentinel, так как из-за особенностей работы этого сервиса у него нет полноценной парольной защиты).

Добавьте порты 6379/tcp и 26379/tcp:

```
firewall-cmd --zone=vcore_redis --add-port=6379/tcp --add-port=26379/tcp --permanent
```

Перезагрузите конфигурацию firewalld:

firewall-cmd --reload

Сли в вашей инфраструктуре используется другой фаервол, то примените аналогичные изменения в соответствии с инструкциями к используемому продукту.

Мастер-узел

Настройка Redis

важно

Поддерживается Redis версий 7.0 и выше.

Установите пакет redis (Sentinel включен в базовую поставку пакета):

dnf -y install redis

Очистите файл /etc/redis/redis.conf:

```
> /etc/redis/redis.conf
```

Откройте файл /etc/redis/redis.conf и укажите следующие параметры:

⁵ https://www.kernel.org/doc/Documentation/vm/overcommit-accounting

File: /etc/redis/redis.conf

```
bind 10.24.24.20 127.0.0.1
port 6379
daemonize no
supervised systemd
protected-mode yes
pidfile "/var/run/redis/redis.pid"
logfile "/var/log/redis/redis.log"
dir "/var/lib/redis"
masteruser "default"
masterauth "УКАЖИТЕПАРОЛЬ"
user default on >УКАЖИТЕПАРОЛЬ ~* &* +@all
```

Нужно отметить следующие пункты:

- 127.0.0.1 в bind требуется для входа в Redis через redis-cli с локальной машины без указания адреса сервиса.
- protected-mode включает некоторые ограничения на вход в Redis: в частности полностью запрещает беспарольный вход в Redis (но не в Sentinel);
- в masteruser и masterauth указываются параметры входа в master-инстанс Redis;
- user фактически меняет меняет параметры пользователя default так, чтобы его можно был использовать для входа в Redis. Странные символы после пароля – это симвилическое описание разрешений для пользователя (подр. в документации Redis⁶)

Запустите сервис Redis:

systemctl start redis

```
Зайдите в окружение Redis:
```

redis-cli --user **default** --pass УКАЖИТЕПАРОЛЬ

Внутри окружения Redis проверьте список acl. Вы должны увидеть, что default содержит хэш пароля и параметры, которые были указаны в конфигурации:

```
127.0.0.1:6379> acl list
1) "user default on sanitize-payload
#ef797c8118f02dfb649607dd5d3f8c7623048c9c063d532cc95c5ed7a898a64f ~* &* +@all"
```

Выйдите из окружения Redis:

exit

Настроойка Sentinel

Настройте конфигурационный файл /etc/redis/sentinel.conf: (Обнулить и добавить?)

```
bind 10.24.24.20 127.0.0.1
port 26379
daemonize no
dir "/var/lib/redis"
pidfile "/var/run/redis-sentinel.pid"
logfile "/var/log/redis/sentinel.log"
# Redis Replication Settings
sentinel monitor vcore 10.24.24.20 6379 2
sentinel auth-user vcore default
sentinel auth-pass vcore YKAXKITEΠAPOЛЬ
```

⁶ https://redis.io/docs/management/security/acl/

sentinel down-after-milliseconds vcore 5000

Здесь нужно:

- в sentinel monitor (а так же в auth-параметрах)нужно указать имя мастера (в примере указано как vcore) и указать данные подключения. 2 в конце – это минимальное количество рабочих узлов, при котором возможна мастер-узла (иными словами, это значение кворума);
- В auth-параметрах нужно указать данные аутентификации подключения к инстансу Redis. Они должны совпадать с теми, что указаны в конфигурации Redis;
- Параметр down-after-milliseconds определяет, через сколько миллисекунд Redis будет выбирать нового мастера при потере текущего.

Запустите сервисы Redis и Sentinel:

systemctl restart redis-sentinel redis

Узлы репликации

Настройка почти идентична тому, как настраивается мастер-узел.

Установите пакет redis (Sentinel включен в базовую поставку пакета):

dnf install redis

Настройте конфигурационный файл /etc/redis.conf с данными для включения репликации:

```
bind 10.24.24.31 127.0.0.1
port 6379
daemonize no
supervised systemd
protected-mode yes
pidfile "/var/run/redis/redis.pid"
logfile "/var/log/redis/redis.log"
dir "/var/lib/redis"
replicaof 10.24.24.20 6379
masteruser "default"
masterauth "УКАЖИТЕПАРОЛЬ"
user default on >УКАЖИТЕПАРОЛЬ ~* &* +@all
```

Здесь отдельно отметим:

• в replicaof нужно указать адрес подключения к мастер-узлу Redis. Остальные данные так же нужно указать для мастер-узла Redis.

Настройте конфигурационный файл /etc/redis/sentinel.conf:

```
bind 10.24.24.31 127.0.0.1
port 26379
daemonize no
dir "/var/lib/redis"
pidfile "/var/run/redis-sentinel.pid"
logfile "/var/log/redis/sentinel.log"
# Redis Replication Settings
sentinel monitor vcore 10.24.24.20 6379 2
sentinel auth-user vcore default
sentinel auth-pass vcore УКАЖИТЕПАРОЛЬ
sentinel down-after-milliseconds vcore 5000
```

Здесь отдельно отметим:

• в monitor нужно указать параметры мастер-узла Redis. Остальные данные так же нужно указать для мастер-узла Redis.

Запустите сервисы redis и sentinel:

```
systemctl restart redis
systemctl restart redis-sentinel
```

Выполните эти команды для всех узлов репликации Redis.

Проверка работы репликации

Проверьте, что репликация успешно запустилась:

• на мастер-узле запустите команду:

на узле репликации:

redis-cli info replication Примерный ответ: role:slave master_host:10.24.24.20 master_port:6379 master_link_status:up master_last_io_seconds_ago:2 master_sync_in_progress:0 slave_repl_offset:140 slave_priority:100 slave_read_only:1 replica_announced:1 connected_slaves:0 master_failover_state:no-failover master_replid:e1a86d60fe42b41774f186528661ea6b8fc1d97a master_repl_offset:140 second_repl_offset:-1 repl_backlog_active:1 repl_backlog_size:1048576 repl_backlog_first_byte_offset:1 repl_backlog_histlen:140

(C) татус master_failover_state:no-failover означает отсутствие каких-либо операций по миграции мастера и является штатным режимом работы.

Включение сервисов Redis и Sentinel в Pacemaker

После успешного запуска репликации Redis и системы управления кластером Redis Sentinel эти сервисы нужно добавить в качестве ресурсов в Pacemaker.

Вначале выключите сервисы Redis и Sentinel в systemd во всех узлах, начиная с узлов репликации:

systemctl disable --now redis redis-sentinel

В Pacemaker добавьте pecypc systemd с указанием названия сервисов redis и redis-sentinel в режиме clone:

```
pcs resource create vcore_redis systemd:redis op monitor interval="10s" timeout="15s"
    clone
pcs resource create vcore_sentinel systemd:redis-sentinel op monitor interval="10s"
    timeout="15s" clone
```

В режиме clone сервисы redis и sentinel одновременно будут запущены на всех узлах кластера Pacmaker.

В colocation добавьте очередность запуска сервисов: вначале должен запуститься сервис redis, потом - sentinel:

pcs constraint order start vcore_redis-clone then start vcore_sentinel-clone

Установка vCore

Управляющая часть

Настройка окружения

Пакеты vCore расположены в соответствующем репозитории Basis. Для включения этого репозитория нужно сделать следующее.

Fedora

- 1. Создайте файл /etc/yum.repos.d/vcore.repo.
- 2. В файле /etc/yum.repos.d/vcore.repo укажите следующее:

```
[vcore-engine]
name= vCore Engine
baseurl=https://vcore-public:vcore-public@maven.tionix.ru/artifactory/vcore-rpm/
fedora/linux/$releasever/$basearch
enabled=1
gpgcheck=0
```

3. Для обновления системы запустите команду:

dnf update

Almalinux

- 1. Создайте файл /etc/yum.repos.d/vcore.repo.
- 2. В файле /etc/yum.repos.d/vcore.repo укажите следующее:

```
[vcore-engine]
name=vCore Engine
baseurl=https://vcore-public:vcore-public@maven.tionix.ru/artifactory/vcore-rpm/
almalinux/8/vcore/x86_64/
enabled=1
gpgcheck=0
```

3. Для обновления системы запустите команду:

dnf update

Установка балансировщика нагрузки

Установка балансера выполняется на всех нодах кластера и состоит из следующих этапов:

- 1. Установка пакета балансировщика и первоначальная настройка;
- 2. Создание pecypca systemd в Pacemaker в режиме Active/Standby, который будет запускать балансировщик на одном узле из имеющихся;
- 3. Добавление правила, по которому балансировщик будет всегда запускаться вместе с виртуальным адресом.

Установка сервиса балансировщика

Для установки и настройки балансировщика требуется сделать следующее:

1. Установите пакет балансировщика:

dnf -y install basis-vbalancer

2. В файле /etc/sysconfig/basis-vbalancer укажите адрес прослушивания для сервиса. Для первого параметра необходимо указать выбранный VIP-адрес, а для второго – реальный IP-адрес интерфейса:

BALANCER_BIND_ADDR="10.24.24.80" BALANCER_PRIVATE_BIND_ADDR="10.24.24.20"

3. Убедитесь, что сервис в systemd не содержит флага автозапуска:

systemctl disable basis-vbalancer

Настройка ресурса балансировщика

1. Для добавления ресурса балансировщика в одном из узлов необходимо выполнить следующую команду:

```
pcs resource create basis-vbalancer systemd:basis-vbalancer op monitor
interval="10s" timeout="15s"
```

2. Проверьте статус кластера:

pcs status

Вывод должен измениться так:

```
Cluster name: cluster
Cluster Summary:
  * Stack: corosync (Pacemaker is running)
  * Current DC: vcore-control-1 (version 2.1.6-4.fc37-6fdc9deea29) - partition
with quorum
  * Last updated: Thu Aug 10 11:51:49 2023 on vcore-control-1
  * Last change: Thu Aug 10 11:51:42 2023 by root via cibadmin on vcore-control-1
  * 3 nodes configured
  * 6 resource instances configured
Node List:
  * Online: [ vcore-control-1 vcore-control-2 vcore-control-3 ]
Full List of Resources:
  * pgsql-vip (ocf::heartbeat:IPaddr2):
                                             Started vcore-control-2
               (ocf::heartbeat:IPaddr2):
                                             Started vcore-control-1
  * vcore-vip
  * Clone Set: pgsqld-clone [pgsqld] (promotable):
    * Masters: [ vcore-control-2 ]
    * Slaves: [ vcore-control-1 vcore-control-3 ]
  * basis-vbalancer (systemd:basis-vbalancer): Started vcore-control-3
Daemon Status:
  corosync: active/enabled
  pacemaker: active/enabled
  pcsd: active/enabled
```

3. В выводе можно заметить, что балансировщик запустился на узле, отличным от того, где настроен VIP-адрес. Для исправления этой ситуации необходимо в одном из узлов запустить команду, которая добавит ограничение для ресурсов (colocation):

pcs constraint colocation add basis-vbalancer with vcore-vip

 Также необходимо включить правило очередности запуска ресурсов: вначале должен запускаться vip-адрес, затем ресурс балансировщика. Для этого на одном из узлов выполните следующую команду: pcs constraint order start vcore-vip then start basis-vbalancer

5. Ещё раз проверьте статус сервисов:

pcs status

Вывод должен измениться так:

```
Cluster name: cluster
Cluster Summary:
  * Stack: corosync (Pacemaker is running)
  * Current DC: vcore-control-1 (version 2.1.6-4.fc37-6fdc9deea29) - partition
with quorum
  * Last updated: Thu Aug 10 11:52:52 2023 on vcore-control-1
  * Last change: Thu Aug 10 11:52:50 2023 by root via cibadmin on vcore-control-1
  * 3 nodes configured
  * 6 resource instances configured
Node List:
  * Online: [ vcore-control-1 vcore-control-2 vcore-control-3 ]
Full List of Resources:
  * pgsql-vip
               (ocf::heartbeat:IPaddr2):
                                             Started vcore-control-2
                (ocf::heartbeat:IPaddr2):
                                           Started vcore-control-1
  * vcore-vip
  * Clone Set: pgsqld-clone [pgsqld] (promotable):
    * Masters: [ vcore-control-2 ]
    * Slaves: [ vcore-control-1 vcore-control-3 ]
  * basis-vbalancer (systemd:basis-vbalancer): Starting vcore-control-1
Failed Resource Actions:
  * basis_vbalancer_monitor_10000 on vcore-control-3 'not running' (7): call=86,
status='complete', exitreason='failed', last-rc-change='Thu Aug 10 11:52:38 2023',
queued=0ms, exec=0ms
Daemon Status:
  corosync: active/enabled
  pacemaker: active/enabled
  pcsd: active/enabled
  Racemaker может запустить балансировщик на узле без VIP до настройки привязки его
   ресурса к VIP. В этом случае в статусе будет выводиться лог неудачного запуска:
     * basis_vbalancer_monitor_10000 on vcore-control-3 'not running' (7): call=8
     6, status='complete', exitreason='failed', last-rc-change='Thu Aug 10
     11:52:38 2023', queued=0ms, exec=0ms
```

После указания привязки балансировщик успешно запустится. Для очистки от данных прошлых неудачных запусков запустите команду:

pcs resource cleanup basis-vbalancer

Установка и настройка брокера

Установка брокера выполняется на всех нодах кластера и состоит из следующих этапов::

1. Установите пакет брокера:

dnf -y install vcore-broker

2. Брокер будет установлен в каталог /opt/vcore/broker/. Откройте файл конфигурации брокера по пути /opt/vcore/broker/config/application.properties и включите

следующие изменения (в настройках с IP-адресом указывайте *реальный в интерфейсе* управления узла):

а. настройте адрес прослушивания для брокера

quarkus.http.host=10.24.24.20

b. В качестве адреса сервиса Redis, укажите интерфейсы хостов кластера:

```
# Cache
quarkus.redis.hosts=redis://10.24.24.20:26379,redis://
10.24.24.31:26379,redis://10.24.24.202:26379
quarkus.redis.password='УКАЖИТЕПАРОЛЬ'
quarkus.redis.client-type=sentinel
quarkus.redis.master-name=mymaster
```

B параметрe quarkus.redis.password='УКАЖИТЕПАРОЛЬ' указывается пароль от Redis (параметр masterauth "УКАЖИТЕПАРОЛЬ").

с. Включите поддержку балансировщика нагрузки (предоставляет функцию автоматической регистрации сервиса в балансировщике):

```
quarkus.mod-cluster-client.enabled=true
quarkus.mod-cluster-client.services.api.host=10.24.24.20
quarkus.mod-cluster-client.services.api.port=8081
quarkus.mod-cluster-client.advertise-multicast-port=23365
quarkus.mod-cluster-client.services.api.contexts=/api/storage, /api/transport
```

d. Так же включите обнаружение брокера через multicast для агентов вычислительных узлов и control, а также адрес прослушивания для сервиса сообщений:

```
provider.transport.use-multicast-discovery=true
provider.transport.multicast-interface-address=10.24.24.20
```

 е. При необходимости усиления безопасности доступа к брокеру измените параметры входа для экземпляров кластера и control/агентов вычисления (пароль должен быть закодирован в base64):

```
provider.transport.cluster.user=user
provider.transport.cluster.password=Q0hBTkdFTUUhIQ==
provider.transport.user=user
provider.transport.password=Q0hBTkdFTUUhIQ==
```

🖗ажно

При запуске сервиса в сетях с Dual Stack необходимо использовать IPv4-адреса в конфигурации.

Пополнительно

При необходимости в параметре provider.transport.acceptor-params

можно явно указать необходимый тип протокола: protocols=CORE,AMQP,STOMP. Пример:

```
provider.transport.acceptor-params=tcpSendBufferSize=1048576;tcpReceiv
eBufferSize=1048576;amqpMinLargeMessageSize=102400
;protocols=CORE,AMQP,STOMP;useEpoll=true;amqpCredits=1000
;amqpLowCredits=300;amqpDuplicateDetection=true
```

Доступные типы протоколов:

- CORE протокол ActiveMQ;
- AMQP протокол AMQP;
- STOMP протокол STOMP.

В проекте vCore используется только протокол CORE, а протоколы AMQP и STOMP могут использоваться внешними клиентами при необходимости. Допускается указание нескольких значений.

 После настройки брокера на всех узлах необходимо строго последовательно (с ожиданием возвращения ввода в терминале) запустить каждый сервис брокера:

systemctl start vcore-broker

Также брокер не нужно добавлять как ресурс в Pacemaker.

Установка и настройка контрола

Для установки и настройки control необходимо выполнить следующее:

1. Установите пакет control:

```
dnf -y install vcore-control
```

- 2. control будет установлен в каталог /opt/vcore/control/. Откройте файл конфигурации
 - control по пути /opt/vcore/control/config/application.properties и включите следующие изменения:
 - а. настройте адрес прослушивания для брокера (нужно указать *реальный IP-адрес* в интерфейсе управления узла)

quarkus.http.host=10.24.24.20

b. Укажите адрес базы данных, используйте pgsql-vip:

Storage Connections
quarkus.datasource.jdbc.url=jdbc:postgresql://10.24.24.70:5432/vcore

с. Укажите адрес брокера, доступный через балансировщик нагрузки:

broker-rest-api/mp-rest/url=http://10.24.24.80:8081/api

d. Для multicast. Укажите адрес сервиса сообщений. Обязательно нужно указать multicastгруппу 231.7.7 и протокол UDP:

quarkus.artemis.url=udp://231.7.7.7:9876? clientFailureCheckPeriod=5000&retryInterval=1000&reconnectAttempts=5

е. Для unicast. Укажите адреса всех брокеров сообщений внутри скобок:

```
quarkus.artemis.url=(tcp://10.24.24.20:61616,tcp://10.24.24.202:61616,tcp://
10.24.24.31:61616)?
clientFailureCheckPeriod=5000&retryInterval=1000&reconnectAttempts=5
```

f. Укажите уникальный для каждого узла идентификатор в формате UUID4:

control.node-id=da1c181a-2921-4be0-a21e-ca15081a20d8

g. Включите поддержку регистрации в балансировщике нагрузки:

quarkus.mod-cluster-client.enabled=true
quarkus.mod-cluster-client.advertise-multicast-port=23364

h. Если в настройках брокера вы изменили логин и пароль для transport.user, то укажите их:

```
quarkus.artemis.username=user
quarkus.artemis.password=Q0hBTkdFTUUhIQ==
```

⊕ажно

При запуске сервиса в сетях с Dual Stack необходимо использовать IPv4-адреса в конфигурации.

3. После настройки control на всех узлах запустите сервисы:

systemctl start vcore-control

4. Попробуйте подключиться к веб-интерфейсу, используя VIP-адрес и порт 8080:

http://192.168.122.34:8080

Примечание

При использовании FirewallD в качестве системы управления Firewall необходимо открыть порты сервисов. Для этого используйте следующие команды:

```
firewall-cmd --add-service=vcore-broker --permanent
firewall-cmd --add-service=vcore-control --permanent
firewall-cmd --add-service=basis-vbalancer --permanent
```

Перезапустите Firewall:

```
firewall-cmd --reload
```

Для проверки портов выполните:

```
nc -v IP_ADDR PORT
```

Где:

- IP_ADDR IP-адрес узла;
- РОЯТ Порт сервиса.

Список портов сервисов по умолчанию доступен по ссылке (см. стр. 54).

Настройка сервисов vCore

Проверьте статус сервисов vCore:

```
systemctl list-units | grep vcore
basis-vbalancer.service
loaded active running Basis vBalancer Service
vcore-broker.service
loaded active running vCore Broker Service
vcore-control.service
loaded active running vCore Control Service
```

Остановите сервисы vCore:

systemctl stop vcore-broker vcore-control basis-vbalancer

Настройка кластера Pacemaker

Отключите зависимости systemd. Скомментируйте строку #Wants=:

```
systemctl edit --full vcore-broker
systemctl edit --full vcore-control
```

Отключите контроль systemd над vCore:

systemctl disable -- now basis-vbalancer vcore-broker vcore-control

Создайте ресурсы для vCore:

```
pcs resource create vcore-broker systemd:vcore-broker clone
pcs resource create vcore-control systemd:vcore-control clone
pcs constraint order start basis-vbalancer then start vcore-broker-clone
pcs constraint order start pgsqld-vip then start vcore-control-clone
pcs constraint order start vcore-broker-clone then start vcore-control-clone
```

Проверьте статус кластера:

pcs status

Вычислительная часть

Процесс установки и настройки вычислительной части не отличается от простой инсталляции. Данная процедура подробно описана в разделе – "Установка гипервизора" (см. стр. 17). Все узлы кластера должны иметь одинаковую версию гипервизоров.

важно

После установки и настройки на вычислительном узле ПО Базис узел автоматически подключится к платформе со статусом "Обслуживание". Для ввода узла в эксплуатацию с таким состоянием его необходимо активировать, добавив его в один из кластеров. Данная процедура подробна описана в руководстве Администратора vCore.

Настройка панели управления

Проверьте доступность веб-интерфейса vCore по адресу http://VIP_DNS_ADDR:8080 (в нашем примере - это адрес 10.24.24.80) будет доступен веб-интерфейс платформы. Укажите первичные данные для пользователя admin:

(у) нформация

Для корректной работы системы рекомендуется использовать Google Chrome или Firefox актуальных версий.

BÁSIS
Имя пользователя
admin
Фамилия
Имя
Отчество
Email
Пароль
Подтверждение
Сохранить

Форма создания пользователя

Заполните поля формы создания пользователя. Подтвердите создание кнопкой "Сохранить". После этого укажите следующие данные для входа в систему:

- домен: master;
- пользователь: admin;
- пароль: указанный для пользователя admin.

Интеграция Базис.vCore с Базис.Virtual Security

- Введение (см. стр. 46)
- Управляющая часть (см. стр. 47)
- Вычислительная часть (см. стр. 48)

Введение

Данная статья описывает процесс настройки платформы Базис.vCore для корректной работы с Базис.Virtual Security в рамках единого кластера управления.

Для установки vCore используйте один из возможных вариантов:

- Демонстрационная инсталляция⁷;
- Простая инсталляция⁸;
- Инсталляция с поддержкой высокой доступности⁹.

⁷ https://conf.tionix.ru/pages/viewpage.action?pageId=348849420&src=contextnavpagetreemode

⁸ https://conf.tionix.ru/pages/viewpage.action?pageId=348849423&src=contextnavpagetreemode

⁹ https://conf.tionix.ru/pages/viewpage.action?pageId=348849440&src=contextnavpagetreemode

Подробный процесс установки Базис. Virtual Security описан в "Руководстве по установке" данного программного модуля.

Управляющая часть

Настройте конфигурационный файл сервиса Control - /opt/vcore/control/config/

application.properties:

quarkus.http.port=8086 broker-rest-api/mp-rest/url=http://BROKER_IP_DNS_ADDR:8085/api

Настройте конфигурационный файл сервиса Broker - /opt/vcore/broker/config/

application.properties:

quarkus.http.port=8085
quarkus.mod-cluster-client.services.api.port=8085

Настройте firewall:

firewall-cmd --permanent --add-port=8085/tcp
firewall-cmd --permanent --add-port=8086/tcp
firewall-cmd --reload

Перезапустите сервисы Control и Broker:

systemctl restart vcore-broker vcore-control

После перезапуска сервисов веб-интерфейс платформы vCore будет доступен по адресу - http:// CONTROL_IP_DNS_ADDR:8086.

Добавьте репозиторий проекта в - /etc/yum.repos.d/bvs.repo:

```
[bvs]
name=BVS $releasever
baseurl=https://tvs-public:tvs-public@maven.tionix.ru/artifactory/tvs-rpm/dev/fedora35/
$basearch
enabled=1
gpgcheck=0
[bvs-noarch]
name=BVS $releasever
baseurl=https://tvs-public:tvs-public@maven.tionix.ru/artifactory/tvs-rpm/dev/fedora35/
noarch
enabled=1
gpgcheck=0
[vcore-contrib]
name=vCore Contrib Packages
baseurl=https://vcore-public:vcore-public@maven.tionix.ru/artifactory/vcore-contrib/
fedora/linux/$releasever/$basearch
enabled=1
gpgcheck=0
```

Обновите данные репозиториев:

dnf update

Установите пакет postgresql:

dnf -y install postgresql12-server

Установите пакет tionix-tvs-system:

dnf install tionix-tvs-system

Запустите скрипт настройки:

```
tvs_configure.sh
```

Укажите необходимые параметры. По окончании процесса настройки все сервисы будут запущены автоматически.

Вычислительная часть

В данном разделе описана процедура установки агентского программного обеспечения Базис. Virtual Security на вычислительном узле (гипервизоре) vCore.

Добавьте репозиторий проекта в - /etc/yum.repos.d/bvs.repo:

```
[bvs]
name=BVS $releasever
baseurl=https://tvs-public:tvs-public@maven.tionix.ru/artifactory/tvs-rpm/dev/
fedora35/$basearch
enabled=1
gpgcheck=0
[bvs-noarch]
name=BVS $releasever
baseurl=https://tvs-public:tvs-public@maven.tionix.ru/artifactory/tvs-rpm/dev/fedora35/
noarch
enabled=1
gpgcheck=0
```

Установите необходимые пакеты:

```
dnf install -y tionix-tvs-agent
```

Запустите скрипт настройки

```
tvs_configure.sh
```

По окончании процесса настройки все сервисы будут запущены автоматически.

Перезапустите службу агента вычислительного узла:

systemctl restart vcore-agent

Обновление Базис.vCore

🗗ажно

Обновление Базис. vCore v.1.5 с более ранних версий не поддерживается.

Данная инструкция описывает процесс обновления Базис. vCore v.1.5 в рамках минорных версий продукта:

- Предварительные действия перед обновлением (см. стр. 49)
- Управляющая часть (см. стр. 49)
 - Обновление пакетов (см. стр. 49)
 - Обновление конфигурации сервисов (см. стр. 49)
 - Запуск сервисов (см. стр. 50)
- Вычислительная часть (см. стр. 50)

Предварительные действия перед обновлением

На период обновления платформы необходимо остановить все работы в облаке:

- 1. Остановите все запущенные ВМ.
- 2. Переведите гипервизоры в режим обслуживания.
- 3. Выключите все сервисы агентов, подключенные к системе управления vCore.
- 4. Создайте резервную копию БД.

Управляющая часть

Обновление пакетов

Остановите службы системы управления:

systemctl stop vcore-control
systemctl stop vcore-broker

Проверьте наличие обновлений для пакетов vCore:

```
dnf check-update --refresh | grep 'vcore-'
```

Обновите пакеты Базис vCore:

dnf -y update vcore-control vcore-broker

Для обновления всей системы используйте обычную команду update:

dnf -y update

🕜ажно

При перезапуске узла при обновлении системных пакетов (например, ядра) необходимо предварительно выключить автозагрузку служб Базис vCore:

systemctl disable vcore-control vcore-broker

После чего произведите перезапуск.

Обновление конфигурации сервисов

После обновления пакетов имеющийся файл конфигурации сервисов будут переименован как applications.properties.rpmsave.

```
ls /opt/vcore/broker/config/
```

application.properties application.properties.rpmsave

```
ls /opt/vcore/control/config/
application.properties application.properties.rpmsave
```

В новом файле application.properties необходимо перенести все указанные в старой конфигурации параметры сервисов. После чего сохраните изменения.

Запуск сервисов

Запустите службы:

```
systemctl start vcore-broker
systemctl start vcore-control
```

Включите автозагрузку служб:

systemctl enable vcore-control vcore-broker

Вычислительная часть

Остановите агент vCore:



Окно изменения состояния гипервизора

Скопируйте параметры основных настроек:



Окно настроек гипервизора

Выключите узел:

	BASIS	vCore
Перезагрузка и выключение		Выключение сервера
Перезагрузка сервера Выключение сервера		Нажмите <enter> для выключения сервера.</enter>
	Вы хотите выкл	очить сервер?
	KESCS HET KP85	44
<esc влево=""> Назад <Вверх/Вниз> Выбор</esc>		«Enter» Выключение

Окно выключения гипервизора

Загрузите новую версию образа гипервизора в систему. Запустите и установите систему из полученного образа.

Настройте параметры гипервизора

—————————————————————————————————————						
ID датацентра ID узла ID домена Адрес агента Порт агента Адрес подключения	81d42d15-cb75-4e6d-8e26-2e083ddf8283 d5a6d503-b655-487e-ab44-baf281c20c47 bd92c60e-d7d5-11ea-87d0-0242ac130003 127.0.0.1 8083 tcp://localhost:61616					
Необязательные настр	зойки					
Имя пользователя	user					
Пароль	*****					
Пароль (еще раз)	*****					
Скрытые bond-интерфе	ейсы					
Скрытые сети VLAN	Скрытые сети VLAN					
Скрытые интерфейсы						
<enter> Далее/ОК <esc> Отмена <f8> ОК</f8></esc></enter>						

Окно настроек гипервизора

Сохраните настройки при помощи клавиши Enter. В открывшемся окне подтвердите перезапуск сервиса:

Основные настройки агента vCore —					
Обязательные настройки					
ID датацентра ID узла ID домена Адрес агента Поот	81d42d15-cb75-4e6d-8e26-2e083ddf8283 d5a6d503-b655-487e-ab44-baf281c20c47 bd92c60e-d7d5-11ea-87d0-0242ac130003 10.38.7.55				
норт Адрес Необя Сервис аген	Настройки сохранены. га и сервис sanlock Будет перезапущены.				
Имя п <esc> Нет <f8> Да Парол Пароль (еще раз) ******** Скрытые bond-интерфейсы Скрытые сети VLAN Скрытые интерфейсы</f8></esc>					
<enter> Далее/ОК <esc> Отмена <f8> ОК</f8></esc></enter>					

Меню перезапуска гипервизора

№ юбые изменения в конфигурационном файле требуют перезапуска агента. При изменении ID узла требуется изменение параметра в sanlock и перезапуск sanlock-сервиса. В связи с этим перед сохранением и перезапуском сервисов выводится соответствующее сообщение об изменении настроек и запросом на перезапуск сервисов:



Убедитесь, что агент успешно запустился:



Сообщение об успешном запуске гипервизора

Удаление Базис.vCore

Для удаления платформы Базис.vCore необходимо удалить пакеты сервисов на всех управляющих и вычислительных узлах. Перед удалением платформы рекомендуется сохранить конфигурацию сервисов, так как она будет удалена вместе с пакетами сервисов. Расположение конфигурационных файлов можете получить в разделе - Установка Базис.vCore¹⁰.

Управляющая часть

Остановите службы системы управления:

```
systemctl stop vcore-control
systemctl stop vcore-broker
```

Удалите пакеты системы управления:

dnf remove vcore-*

Вычислительная часть

Остановите службу агента:

systemctl stop vcore-agent

Удалите агент:

dnf remove vcore-*

¹⁰ https://conf.tionix.ru/pages/viewpage.action?pageId=290095209&src=contextnavpagetreemode

Приложения к руководству по установке Базис.vCore

Приложение 1. Список портов сервисов Базис.vCore

\${action.page.title}

Компонент, инициализ ирующий подключен ие (исх.трафи к)	Компон ент, обслуж ивающи й подклю чение (вх.тра фик)	Проток ол/порт	Назначени е порта (содержан ие трафика)	Файл настройки компонента	Параметр настройки компонента
vcore- control	vcore- broker	TCP/ 8081	Порт АРІ- запросов.	<pre>/opt/vcore/broker/ config/ application.properties</pre>	quarkus.http.port=<значение>
vcore- balancer	vcore- broker	TCP/ 8081	Порт, анонсируе мый балансиров щику.	<pre>/opt/vcore/broker/ config/ application.properties</pre>	quarkus.mod-cluster- client.services.api.port=<значение>
vcore-agent	vcore- broker	TCP/ 61616	Порт шины очереди сообщений ActiveMQ.	<pre>/opt/vcore/broker/ config/ application.properties</pre>	provider.transport.bind-port=<значение>

Компонент, инициализ ирующий подключен ие (исх.трафи к)	Компон ент, обслуж ивающи й подклю чение (вх.тра фик)	Проток ол/порт	Назначени е порта (содержан ие трафика)	Файл настройки компонента	Параметр настройки компонента
vcore- broker	vcore- broker	TCP/ 61617	Порт для статическог о определен ия сервисов брокера в сети.	/opt/vcore/broker/ config/ application.properties	provider.transport.static-discovery- addresses=tcp:// <ipv4_address>:<значение></ipv4_address>
vcore- balancer	vcore- broker	UDP/ 23365	muticast порт для взаимодейс твия с балансиро вщиком.	/opt/vcore/broker/ config/ application.properties	quarkus.mod-cluster-client.advertise-multicast- port=<значение>
vcore- agent, vcore- control	vcore- broker	UDP/ 9876	multicast порт для динамическ ого определен ия сервисов брокера в сети.	/opt/vcore/broker/ config/ application.properties	provider.transport.multicast-port=<значение>

Компонент, инициализ ирующий подключен ие (исх.трафи к)	Компон ент, обслуж ивающи й подклю чение (вх.тра фик)	Проток ол/порт	Назначени е порта (содержан ие трафика)	Файл настройки компонента	Параметр настройки компонента
Браузер	vcore- control	TCP/ 8082	Стандартн ый порт веб- интерфейс а.	<pre>/opt/vcore/control/ config/ application.properties</pre>	quarkus.http.port=<значение>
vcore- balancer	vcore- control	UDP/ 23364	multicast порт для взаимодейс твия с балансиро вщиком.	<pre>/opt/vcore/control/ config/ application.properties</pre>	quarkus.mod-cluster-client.advertise-multicast- port=<значение>
Браузер	vcore- balancer	TCP/ 8080	Стандартн ый порт веб- интерфейс а, при кластерной инсталляци и.	/opt/vcore/balancer/ config/standalone.xml	socket-binding name="http" port="\$ {jboss.http.port:<значение>}"
vcore- control	vcore- balancer	TCP/ 8081	Порт АРІ- запросов.	/opt/vcore/balancer/ config/standalone.xml	socket-binding name="http-broker" port="<значение>"

Компонент, инициализ ирующий подключен ие (исх.трафи к)	Компон ент, обслуж ивающи й подклю чение (вх.тра фик)	Проток ол/порт	Назначени е порта (содержан ие трафика)	Файл настройки компонента	Параметр настройки компонента
	vcore- balancer	TCP/ 8090	Порт управления	/opt/vcore/balancer/ config/standalone.xml	socket-binding name="mcmp-management" interface="private" port="\$ {jboss.mcmp.port:<значение>}"
vcore- broker	vcore- balancer	UDP/ 23365	multicast, для взаимодейс твия с брокером.	/opt/vcore/balancer/ config/standalone.xml	socket-binding name="modcluster-broker" interface="private" multicast- address="224.1.0.105" multicast-port="<значение>"
vcore- control	vcore- balancer	UDP/ 23364	multicast, для взаимодейс твия с контроллер ом.	/opt/vcore/balancer/ config/standalone.xml	socket-binding name="modcluster" interface="private" multicast-address="\$ {jboss.modcluster.multicast.address:224.1.0.105}" multicast-port="<значение>"
vcore- control	vcore- agent	TCP/ 8083	НТТР порт.	<pre>/opt/vcore/agent/ config/ application.properties</pre>	quarkus.http.port=<значение>

Компонент, инициализ ирующий подключен ие (исх.трафи к)	Компон ент, обслуж ивающи й подклю чение (вх.тра фик)	Проток ол/порт	Назначени е порта (содержан ие трафика)	Файл настройки компонента	Параметр настройки компонента
vcore- balancer	vcore- agent	UDP/ 23364	multicast, для взаимодейс твия с балансиров щиком.	<pre>/opt/vcore/agent/ config/ application.properties</pre>	quarkus.mod-cluster-client.advertise-multicast- port=<значение>
vcore- control	postgres ql	TCP/ 5432	Порт доступа к СУБД.	/var/lib/pgsql/data/ postgresql.conf	port =<значение>
vcore- broker, redis- sentinel	redis	TCP/ 6379	Порт доступа.	/etc/redis/redis.conf	port <значение>
vcore- broker	redis- sentinel	TCP/ 26379	Порт доступа.	/etc/redis/ sentinel.conf	port <значение>
libvirt	libvirt	TCP/ 16509	Порт для обеспечени я живой миграции.		

Компонент, инициализ ирующий подключен ие (исх.трафи к)	Компон ент, обслуж ивающи й подклю чение (вх.тра фик)	Проток ол/порт	Назначени е порта (содержан ие трафика)	Файл настройки компонента	Параметр настройки компонента
Клиент ssh	ssh	TCP/22	Управлени е платформо й.	<pre>/etc/ssh/sshd_config</pre>	Port <значение>
corosync, pcsd, pacemaker	corosync , pcsd, pacemak er	TCP/ 2224	Кластер высокой доступност и.		
corosync, pcsd, pacemaker	corosync , pcsd, pacemak er	TCP/ 3121	Кластер высокой доступност и.		
corosync, pcsd, pacemaker	corosync , pcsd, pacemak er	UDP/ 5404	Кластер высокой доступност и.		
corosync, pcsd, pacemaker	corosync , pcsd, pacemak er	TCP/ 5403	Кластер высокой доступност и.		

Компонент, инициализ ирующий подключен ие (исх.трафи к)	Компон ент, обслуж ивающи й подклю чение (вх.тра фик)	Проток ол/порт	Назначени е порта (содержан ие трафика)	Файл настройки компонента	Параметр настройки компонента
corosync, pcsd, pacemaker	corosync , pcsd, pacemak er	UDP/ 5405-54 12	Кластер высокой доступност и.		
corosync, pcsd, pacemaker	corosync , pcsd, pacemak er	TCP/ 9929	Кластер высокой доступност и.		
corosync, pcsd, pacemaker	corosync , pcsd, pacemak er	UDP/ 9929	Кластер высокой доступност и.		
corosync, pcsd, pacemaker	corosync , pcsd, pacemak er	TCP/ 21064	Кластер высокой доступност и.		

Приложение 2. Рекомендации по настройке систем хранения для хранилищ Базис.vCore

- Введение (см. стр. 62)
- NFS (см. стр. 62)
- iSCSI (см. стр. 63)
- Fibre Channel (см. стр. 64)

Введение

Данная статья содержит информацию с примерами настройки различных типов систем хранения для их добавления в качестве хранилища для vCore.

NFS

В качестве примера настройки приведём конфигурацию NFS-сервера в ОС Linux.

Важно

Не запускайте NFS-сервер на управляющих узлах vCore (исключение - all-in-one-инсталляции, такая конфигурация пока считается экспериментальной).

NFS-сервер должен поддерживать параметры идентификаторов пользователей и группы (anonuid, anongid), так как vCore не поддерживает запись на каталог NFS от имени root.

Установите необходимые пакеты:

```
dnf -y install nfs-utils
```

В Firewalld (если он используется) необходимо открыть порты, необходимые для функционирования NFS-сервера:

```
firewall-cmd --permanent --zone=public --add-service=nfs
```

Создайте директорию для ISO-хранилища:

mkdir -p /srv/nfs/iso

Создайте директорию для хранилища данных:

mkdir -p /srv/nfs/volumes

Дайте права доступа к хранилищам:

chown vcore:vcore /srv/nfs/iso /srv/nfs/volumes

Для работы с vCore требуется указание правильных настроек экспорта каталогов. Настройте параметры экспорта в файле /etc/exports :

/srv/nfs/iso IP_ADDR/24(rw,sync,anonuid=69,anongid=69,all_squash)
/srv/nfs/volumes IP_ADDR/24(rw,sync,anonuid=69,anongid=69,all_squash)

Где:

- IP_ADDR/24 подсеть сети хранения vCore;
- **rw,sync,anonuid=69,anongid=69,all_squash** параметры экспорта NFS-каталога. Обязательно укажите именно эти параметры.

Запустите NFS-сервисы:

```
systemctl restart nfs-utils systemctl restart nfs-server
```

Добавьте сервисы в автозагрузку:

systemctl enable nfs-utils nfs-server

Проверьте, что каталоги NFS добавлены:

exportfs

Для добавления хранилища в vCore воспользуйтесь инструкцией - Настройка хранилища с сетевой файловой системой (NFS)¹¹.

iSCSI

Установите пакет targetcli:

dnf -y install targetcli

Добавьте сервис target в автозагрузку:

systemctl enable target

Запустите сервис target:

systemctl start target

Посмотрите расположение дискового устройства, который необходимо подключить к платформе:

fdisk -l

Пример ответа:

```
Disk /dev/vdb: 500 GiB, 536870912000 bytes,
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

Откройте targetcli:

targetcli

Перейдите в директорию - /backstores/block :

cd /backstores/block

Создайте привязку луна lun1 к физическому устройству - /dev/vdb:

/backstores/block> create lun1 /dev/vdb

Перейдите в директорию /iscsi:

cd /iscsi

Создайте iSCSI-таргет:

create

Пример ответа:

Created target iqn.2003-01.org.linux-iscsi.vcore-hv-8add8fd1.x8664:sn.95b3c20c7158. Created TPG 1. Global pref auto_add_default_portal=**true**

¹¹ https://conf.tionix.ru/pages/viewpage.action?pageId=375882161#id-УправлениехранилищамиБазис.vCorev.1.3-Настройкахранилищассетевойфайловойсистемой(NFS)

Программное обеспечение «Базис.Virtual	Security». Руководство по установке.	Часть 3. Программный модуль
	«Базис.vCore»	RU.НРФЛ.00002-02 93 01 ЧЗ

Created **default** portal listening on all IPs (0.0.0.0), port 3260.

Перейдите в директорию - /iscsi/iqn.2003-01.org.linux-iscsi.vcorehv-8add8fd1.x8664:sn.95b3c20c7158/tpg1/luns:

cd /iscsi/i	qn.2003	3-01.org.linux-	iscsi	.vcore-	hv-8add8	3fd1.x8664:sn.95b3c20c7158/tpg1/luns
Создайте	Г	ривязку	тар	огета		iqn.2003-01.org.linux-iscsi.vcore-
hv-8add8fd1	x8664	sn.95b3c20c71	.58 к	lun1:		
create /bac	kstore	s/block/lun1				
Перейдите	в	директорию		_	/iscsi,	/iqn.2003-01.org.linux-iscsi.vcore-

hv-8add8fd1.x8664:sn.95b3c20c7158/tpg1/acls:

cd /iscsi/iqn.2003-01.org.linux-iscsi.vcore-hv-8add8fd1.x8664:sn.95b3c20c7158/tpg1/acls

Создайте привязку iSCSI-инициатора:

create iqn.1994-05.com.redhat:213138e11956

Где: iqn.1994-05.com.redhat:213138e11956 - iSCSI-инициатор узла, через который будет происходить управление хранилищем.

(SCSI-инициатор узла можно посмотреть на панели управления платформой vCore, во вкладке
"Инфраструктура" - "Узлы" - "Детали узла":

	Запущено виртуальных машин	0					
🛢 Домены	Статус 🚯	Активен					
🛍 Инфраструктура	Аппаратное обеспечение						
📱 Дата-центры	Модель ЦПУ 📵	Intel Core Processor (Haswell, no TSX, IBRS)					
& Кластеры	Количество ЦПУ О	8					
🔳 Узлы	KUNINGECTBU LITT G	0					
🛋 Хранилища	Количество ядер ЦПУ 🕕	8 x 2.00 ГГц					
😤 Сети							
Пулы MAC-адресов							
С Брокеры	Загрузка ядер, % 🛈						
 Данные Безопасность 	Уязвимости ЦПУ 🕕	SPC-LINE-VZ Multiplicity immediants, parts: conditional; ens.; yw, sine' raidboeg, iesh immig, ywschereienes (no directediu); ILES, MULTI, WARA: Mitigations' MAX disableeb), XMML2, NALL_DAN, No mitigations' MAS (Yukanetake) (SSC / SSC					
	Объем памяти	98% Boero 15.6 FG, Caologeo 239.8 M6					
	Производитель	RDO , 20.4.1-1.el8 , OpenStack Compute , Virtual Machine					
	Серия/идентификатор продукта	6835628b-2188-42ec-9411-8fb0a9b07e5e , 6835628b-2188-42ec-941f-8fb0a9b07e5e					
	Программное обеспечение						
	Программное обеспечение ОС	Basis vCore 1.3.0-20230411					
	Программное обеспечение ОС Режим selinux	Basis vCore 1.30-20230411 enabled (enforcing)					
	Программное обеспечение ОС Режим selinux Ядро	Basis vCore 1.30-20230411 enabled (enforcing) Linux 6.0.12-100.fc35.x86.64 #1 SMP PREEMPT_DYNAMIC Thu Dec 8 16-53-55 UTC 2022 x86.64					
	Программное обеспечение ОС Режим selinux Ядро Версия агента	Basis vCore 1.3.0-20230411 enabled (enforcing) Linux 6.0.12-100.fc35.x88.64 #1 SMP PREEMPT_DYNAMIC Thu Dec 8 16:53:55 UTC 2022 x88.64 1.3.0-SNAPSHOT					
	Программное обеспечение ОС Режим selinux Ядро Версии агента Версии библистек ()	Basis vCore 1.3.0-20230411 enabled (enforcing) Linux 6.0.12-10.3.c35.x88.64 #1 SMP PREEMPT_DYNAMIC Thu Dec 8 16:53:55 UTC 2022 x88.64 1.3.0-SNAPSHOT libuirt: 7.8.0, QEMU: 6.10, NmState: 2.1.4					

Для сохранения изменений в targetcli выполните:

exit

Для добавления хранилища в vCore воспользуйтесь инструкцией - Настройка iSCSI-хранилища¹².

Fibre Channel

Особых настроек для работы Fibre Channel с vCore не требуется. Для корректной работы Fibre Channel необходимо наличие HBA-контроллеров на всех вычислительных узлах vCore и настроенный конфигурационный файл для multipath.

¹² https://conf.tionix.ru/pages/viewpage.action?pageId=375882161#id-УправлениехранилищамиБазис.vCorev.1.3-HaстройкаiSCSIхранилища

Процесс настройки системы хранения зависит от конкретного оборудования. За подробной инструкцией обратитесь к производителю, оборудование которого используется для хранилищ vCore.