



Пошаговая инструкция по развертыванию Росплатформы

расшифровка к видео инструкции

Table of contents

1. Подготовка конфигурации серверного оборудования по рекомендуемым требованиям ПО Росплатформы.	4
2. Развёртывание ПО Росплатформа.	14
2.1 Краткая инструкция по установке Росплатформа - updates.rospatorma.ru/docs/	15
2.2 Дистрибутив можно загрузить по ссылке (объединяет в себе «Р-Виртуализация» и «Р-Хранилище с функциями S3»)	15
3. Пошаговая инструкция:	17
3.1 загрузка установщика	17
3.2 предустановочные настройки первого сервера с контейнерами управления	37
3.3 дополнительные настройки в процессе установки	40
3.4 проверка доступности контейнеров управления	41
3.5 загрузка установщика последующих серверов (без контейнеров)	43
3.6 подключение к контейнеру управления Р-хранилища для настройки кластера	45
3.7 настройка сетевых ролей сервера	46
3.8 создание кластера	46
3.9 настройка дисковых ролей	47
3.10 регистрация серверов в панели Р-Хранилища	49
3.11 настройка сетевых ролей зарегистрированных серверов	53
3.12	54
3.13 добавление зарегистрированных серверов в кластер	54
3.14	55
3.15 создание кластерного датастора для виртуальных сред	55
3.16 начало работы с панелью Р-Виртуализации	58
3.17 регистрация серверов в панели Р-Виртуализации	59
3.18 установка лицензии для Р-Виртуализации на серверах	61
3.19 настройка хранения виртуальных сред в датасторе	64
3.20 перемещение контейнеров управления в датастор	67
3.21 запуск службы отказоустойчивости на серверах	70
3.22 создание виртуальных сред	72
3.23 создание VLAN и добавление в него виртуальных машин	77
3.24 проверка сетевой доступности между виртуальными машинами одного vlan	87
3.25 проверка живой миграции виртуальных сред	89
3.26 проверка восстановления работы виртуальных сред при аварийном выходе из строя сервера	92
3.27 настройка сети хранилища для обеспечения максимальной производительности	97
4. Проверка корректности настроек кластера и работоспособности серверного оборудования.	102

5. Архитектура

110

1. Подготовка конфигурации серверного оборудования по рекомендуемым требованиям ПО Росплатформы.



Минимальная конфигурация для корректной работы кластера:

В каждом сервере не менее двух физических портов от 10Гбит для сети компонента Р-хранилище (данная сеть является внутренней сетью только для блочного уровня (**SDS**)). Отдельно на каждом сервере не менее двух физических портов от 1Гбит для сети компонента Р-управление (Сеть может быть связана с публичной и использована для виртуализации или доступа к **s3**).

Важно!

Необходимо не менее 3 стоечных сервера архитектуры x86.

Для работоспособности всех сервисов платформы на каждом сервере не менее 64ГБ памяти и от 1-ого процессора с 8-ю или 10-ю физическими ядрами с частотой не менее 2Гигагерца.

Для установки операционной системы Росплатформы физический диск не менее 64ГБ объема. Опционально рекомендуется устанавливать на физический том состоящий из двух физических дисков с **RAID1** за счет аппаратного **RAID** контроллера на борту сервера для защиты операционной системы Росплатформы от сбоя одного из системных дисков (кэш контроллера желательно выключить).

RAID1 для системных дисков настраивается заранее перед началом установки Росплатформы за счет настроек биоса сервера или **firmware**, а для дисков под кластер Р-хранилища отключается.

Диски под систему могут быть как **HDD** так и **SSD** с не менее **DWPD1**. Предпочтительнее устанавливать под систему **SSD** так как в этом случае на системных дисках можно разместить службу метаданных от Р-хранилища, которая не требует большого дискового пространства(3ГБ свободного дискового пространства на 100ТБ данных в кластере).

Для организации блочного хранилища (**SDS-P**-хранилища) с возможностью включения кластера необходимо не менее трех дисков на каждом сервере, желательно не менее гибридного массива, состоящего из одного **SSD** и двух **HDD**, где первый это **SSD** корпоративного уровня с **DWPD** не менее 3, с объемом не менее 100ГБ. Максимальный показатель последовательной скорости записи этого диска должен быть не меньше суммы скорости медленных дисков **HDD**. Два диска **HDD** должны быть так же не менее 100ГБ объемом. Все диски для хранилища должны быть презентованы по **JBOD** через **HBA** или на прямую через **JBOD** от интерфейса материнской платы или **RAID** контроллера, поддерживающего одновременную презентацию дисков для хранилища по **JBOD** и **RAID1** для системного диска.

Если не имеется возможность установить, как минимум гибридный набор дисков, то в этом случае можно использовать только **HDD** также презентованные по **JBOD**, но в этом случае скорость ввода вывода будет ниже, чем у гибридного или у массива, который будет состоять только из **SSD**.

В случае, когда все диски **SSD**, они должны быть не менее **DWPD 1** и в этом случае не требуется назначать роль КЭШ, все диски будут с ролью хранения, а на системном будет совместно использоваться роль метаданных.

Коммутация кластера из 3 или более серверов обеспечивается за счет не менее двух ethernet коммутаторов с поддержкой **MLAG** или **VPC**. На стороне серверов настраивается **LACP** в режиме:

```
xmit-hash-policy=layer3+4 #(1)
```

1. 📡 опция в настройках сетевого интерфейса **Bond**

а также **MTU9000** для интерфейсов от 10Гбит и более. На стороне коммутаторов также необходима поддержка **jumbo frame**.

В случае больше количества дисков чем два для каждой ноды, где сумма скорости дисков превышает пропускную способность сети Р-хранилища желательно использовать 25Гбит или более вместо 10Гбит в агрегации для задействования всех скоростных возможностей дисков, так как синхронная репликация с подтверждением записи выполняется по сети Р-хранилища.

Сеть хранилища должна быть в режиме доступа(**access**) **L2**, то есть не в режиме **Trunk**. Адресация статическая без указания шлюза. Сеть для компонента Р-

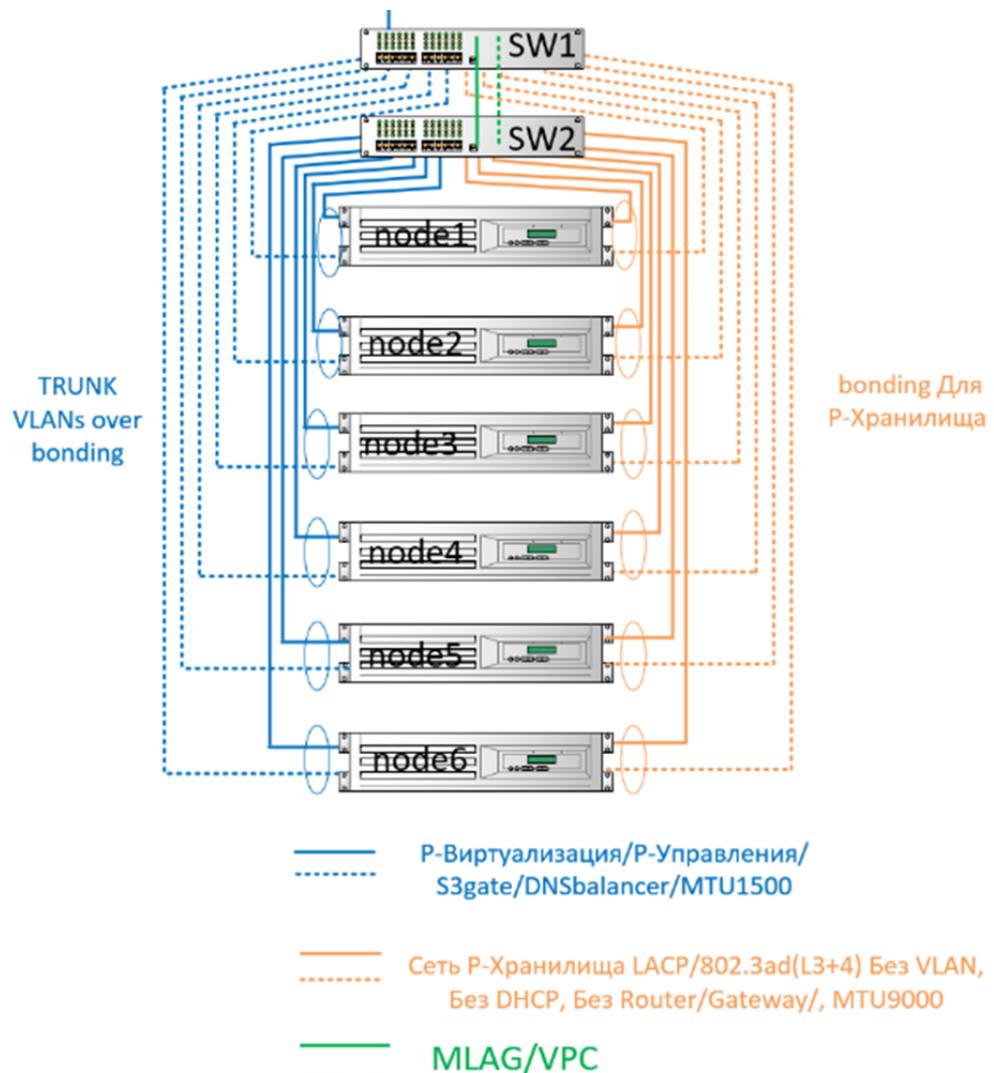
управления может быть в Trunk для разделения трафика управления от сети доступа и от сети виртуализации или **S3**, где каждый трафик в своем теге (**VLAN**).

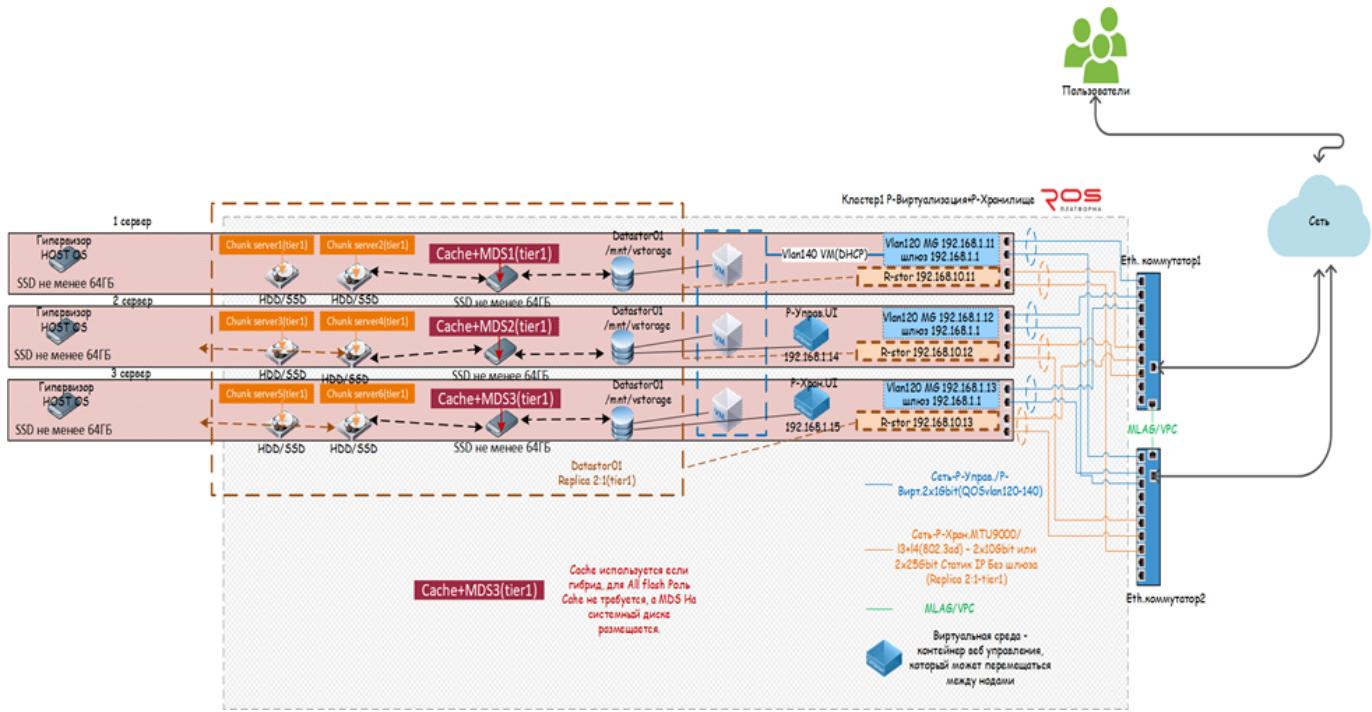
Адресация в сети Р-управления также статическая, для доступа из других сетей или подключения интернета можно прописывать шлюз и **DNS** сервер. Статические адреса могут быть настроены через **DHCP** сервер для исключения появления большего количества шлюзов, прописываемых на сервере. Наличие нескольких шлюзов, прописанных непосредственно в настройках каждого сервера, не поддерживается и требует специальных настроек маршрутов, что усложняет сетевые настройки и сам процесс развертывания такой конфигурации.

 **Важно!**

В случае использования **S3**, для балансировки нагрузки по сети доступа к **S3** между серверами или шлюзами **S3** используется внешние программные или аппаратные балансирующие устройства.

Пример коммутации кластера Росплатформа:





Пример минимальной конфигурации для кластера:

3 сервера:

- 1 CPU 2ГГц 8-10 ядер
- RAM 64 GB+
- SAS/SATA HBA-адаптер с поддержкой режима JBOD
- для загрузки гипервизора/ОС - как минимум 1 диск с объемом не менее 100ГБ или два диска в RAID1 средствами аппаратного RAID-контроллера
- 1 x SSD + 2 x HDD (гибридный массив) JBOD
- 2 x 10 Gbit Ethernet (отдельная сеть для хранилища данных)
- 2 x 1Gbit Ethernet (для сети управления и s3 доступа)
- два коммутатора Ethernet 1/10 Гбит портов в агрегации LACP с поддержкой MLAG или VPC.

Пример конфигураций с большим количеством дисков или массивом другого типа для увеличения производительности:

5 серверов:

- 2 x **CPU** 2ГГц 8-10 ядер
 - **RAM** 128 GB
 - **SAS/SATA HBA**-адаптер с поддержкой режима **JBOD**
 - для загрузки гипервизора/ОС - как минимум 1 диск с объемом не менее 100ГБ или два диска в **RAID1** средствами аппаратного **RAID**-контроллера
 - 2 x **SSD (all flash)**, минимум 2 шт или более (самый быстрый массив, где роль кэш как в гибридне не используется). Все **SSD** диски с **DWPD** не менее 1.
 - 2 x 10 **Gbit Ethernet** или 2 x 25 и более (отдельная сеть для хранилища данных)
 - 2 x 10 **Gbit Ethernet** (для сети управления и виртуализации)
-

Пропорциональность для расчета гибридного массива:

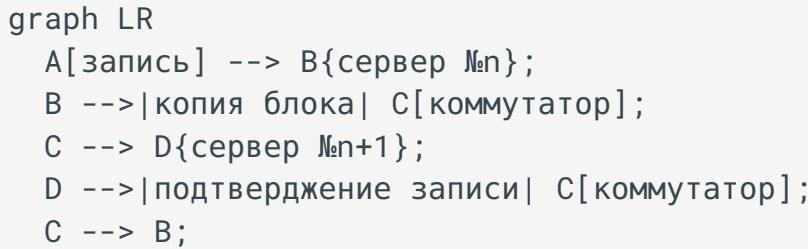
На один **SSD** с ролью кэш подключается 3 или 4 **HDD** с ролью хранения. Если **SSD** больше одного, то на каждый по 3 или 4 **HDD**. Количество **HDD** зависит от максимальной последовательной скорости записи **SSD**, то есть она не должна быть меньше, чем сумма скорости последовательной записи всех **HDD**, которые будут подключены к этому **SSD**. Например: если один **SSD** имеет максимальную скорость записи, заявленную производителем 1000МБ/с, а диски **HDD** с **RPM** 10K в среднем показывают 250МБ/с, то $1000/250=4$ (диска **HDD**).

Для массива **all-flash** роль кэш не используется, и все диски идут с ролью хранения, При этом все диски должны быть с одинаковыми скоростными показателями и объемом.

Пропорциональность расчета пропускной способности сети для внутреннего блочного взаимодействия хранилища следующая:

Сумма скорости всех **SSD** дисков одного сервера должна соответствовать возможности пропускного канала сети хранения блочного уровня. Например, если один диск **SSD** имеет скорость последовательной записи, заявленной

производителем 1000МБ/с то умножив на 8 бит мы получим 8Гбит необходимый пропускной канал для этого диска. Соответственно если на борту одного сервера 10 таких дисков, то пропускной канал сети блочного уровня должен быть не менее 80Гбит. В случае использования только 2x10Гбит, то максимальная скорость записи для одной реплики будет 2250МБ/с (18Гбит) деленная на количество узлов, которые одновременно записывают.



Расчет ресурсов памяти и процессора:

CPU:

- Не менее 8-10 физических ядер на сервер.
- Для операционной системы Росплатформы необходимо не менее 4 ядер.
- Управление 0.5-1 ядро.
- Для службы метаданных блочного уровня одно ядро (одна служба на один сервер).
- Для сервисов блочного хранения на одно ядро 8 дисков.
- Для сервиса клиента блочного уровня (точка монтирования для протокола **fastpath (TCP/IP)**, одна точка на один сервер) 1 ядро.
- Для сервисов **S3** 1.7 ядро.

RAM:

- Не менее 64ГБ памяти на сервер.
- Для операционной системы Росплатформы не менее 4-5ГБ.
- Управление 2-4ГБ.
- Для службы метаданных блочного уровня - 1ГБ на 100ТБ данных.
- Для сервисов блочного хранения - 1ГБ на каждую службу (диск).
- Для сервиса клиента блочного уровня - 1ГБ.
- Для сервисов **S3** - 8ГБ.



Важно!

Ресурсы необходимые для виртуальных сред добавляются в конфигурацию поверх вышеописанных.

Расчет объема полезного хранения и количества узлов:

Р-Хранилище поддерживает ряд режимов для каждого метода избыточности. В таблице ниже представлены издержки данных для разных режимов избыточности.

Первые три строки относятся к репликации, остальные – к избыточному кодированию.

Режим избыточности	Минимальное число необходимых серверов	Допустимое число отказавших серверов	Издержки хранилища, %	Неразмеченное пространство, требуемое для хранения 100ГБ данных
1 реплика без избыточности	1	0	0	100ГБ
2 реплики	2	1	100	200ГБ
3 реплики	3	2	200	300ГБ
дирорвание 1+0 (без избыточности)	1	0	0	100ГБ
Кодирование 1+2	3	2	200	300ГБ
Кодирование 3+2	5	2	67	167ГБ
Кодирование 5+2	7	2	40	140ГБ
Кодирование 7+2	9	2	29	129ГБ
Кодирование 17+3	20	3	18	118ГБ

✓ Примечание:

Режимы избыточного кодирования 1+0 и 1+2 предназначены для небольших кластеров с недостаточным числом серверов для использования других режимов избыточного кодирования, но которые планируется расширить в будущем. Так как после выбора типа избыточности его невозможно поменять (с репликации на избыточное кодирование, или наоборот), данный режим позволяет выбрать избыточное кодирование, даже если кластер не соответствует рекомендациям. После расширения кластера можно выбрать более полезный режим избыточности.

Кроме выше приведенной таблицы необходимо учитывать Кворум на базе алгоритма **Paxos**, где должно оставаться большинство нод в случае выхода из строя. Например, при минимальной конфигурации из трех или четырех серверов может выйти из строя только один сервер, независимо от того, что настроена реплика 2 или реплика 3. При конфигурации из 5 серверов и реплики 3 может из

строя выйти только два сервера. Квorum контролируется на уровне служб метаданных поэтому на каждом сервере по одной такой службе. При большом количестве серверов (больше чем 5) на последующих серверах можно не делать службу метаданных или делать по одной службе на одном из серверов из группы.

Для применения кодирования используются быстрые массивы, начиная от гибридного, где на один **SSD** подключается 3-4 **HDD**, быстрые **all-flash**, где все диски **SSD**, или возможно использование гибридного **all-flash**, где **NVMe** или **intel Optane** в три или четыре раза быстрее обычного **SSD**. В этом случае на один **NVMe** в роли кэша на 3-4 **SSD**.

 **Важно!**

Количество дисков на сервер в среднем считается 20 дисков на узел. Большее число узлов с меньшим количеством дисков предпочтительнее, чем большое количество дисков на один сервер в кластере с минимальным числом серверов. Конфигурация из 5 или 10 серверов, где каждый сервер по 20 дисков будет лучше, чем всего 5 серверов, где каждый сервер по 60 дисков.

2. Развёртывание ПО Росплатформа.

Перед установкой Росплатформы необходимо подготовить и настроить коммутаторы по необходимым параметрам, описанным в подготовка конфигурации серверного оборудования, а также подготовить физические тома на каждом сервере если используется RAID1 для системы.

Видеоинструкцию по развертыванию кластера Р-хранилища, на примере развертывания виртуализации, где все действия подходят так же для развертывания и последующих настроек s3 - rutube.ru или на резервном канале - youtube.com ,где есть тайм теги:

Тайм теги

Ссылки:

1. 0:00 - загрузка установщика
2. 2:29 - предустановочные настройки первого сервера (с контейнерами управления)
3. 17:28 - дополнительные настройки в процессе установки
4. 20:25 - проверка доступности контейнеров управления
5. 21:45 - загрузка установщика последующих серверов (без контейнеров)
6. 24:34 - подключение к контейнеру управления Р-хранилища для настройки кластера
7. 25:23 - настройка сетевых ролей сервера
8. 26:29 - создание кластера
9. 27:20 - настройка дисковых ролей
10. 30:23 - регистрация серверов в панели Р-Хранилища
11. 33:11 - настройка сетевых ролей зарегистрированных серверов
12. 33:58 - добавление зарегистрированных серверов в кластер
13. 36:45 - создание кластерного датастора для виртуальных сред
14. 39:22 - начало работы с панелью Р-Виртуализации
15. 40:20 - регистрация серверов в панели Р-Виртуализации
16. 41:33 - установка лицензии для Р-Виртуализации на серверах
17. 43:14 - настройка хранения виртуальных сред в датасторе
18. 44:59 - перемещение контейнеров управления в датастор
19. 45:59 - запуск службы отказоустойчивости на серверах
20. 48:03 - создание виртуальных сред
21. 52:26 - создание VLAN и добавление в него виртуальных машин
22. 55:18 - проверка сетевой доступности между виртуальными машинами одного vlan
23. 58:03 - проверка живой миграции виртуальных сред
24. 59:25 - проверка восстановления работы виртуальных сред при аварийном выходе из строя сервера
25. 1:02:59 - настройка сети хранилища для обеспечения максимальной производительности

2.1 Краткая инструкция по установке Росплатформа - updates.rosplatforma.ru/docs/

Видеоинструкция по установке и настройке S3 кластера - rutube.ru канал Rosplatforma или на резервном канале - youtube.com канал Rosplatforma

 Тайм теги

Ссылки:

1. [00:00 - введение, техническое описание](#)
2. [00:47 - минимальные требования и установка серверов для s3 кластера](#)
3. [02:40 - подготовка к созданию S3 кластера, добавление необходимых ролей](#)
4. [04:19 - создание и настройка S3 кластера](#)
5. [05:48 - создание пользователя S3 кластера](#)
6. [06:10 - подготовка к установке DNS-сервера для s3 кластера](#)
7. [06:23 - установка DNS-сервера](#)
8. [06:30 - настройка DNS-сервера](#)
9. [06:56 - проверка работы DNS-сервера](#)
10. [08:09 - вход в S3 хранилище с помощью веб-панели Р-Хранилище](#)
11. [09:50 - создание корзины в S3 хранилище](#)
12. [10:03 - загрузка файла в корзину](#)
13. [10:41 - подключение к S3 кластеру с помощью программы s3browser](#)
14. [11:54 - подготовка к подключению к S3 кластеру с помощью утилит s3fs и s3cmd](#)
15. [12:37 - файл с данными для доступа к S3 кластеру](#)
16. [12:47 - настройка s3cmd](#)
17. [14:04 - просмотр списка корзин и их содержимого с помощью s3cmd](#)
18. [14:53 - настройка s3fs](#)
19. [16:06 - включение отказоустойчивости S3 кластера](#)
20. [19:21 - проверка работы отказоустойчивости S3 кластера](#)

2.2 Дистрибутив можно загрузить по ссылке (объединяет в себе «Р-Виртуализация» и «Р-Хранилище с функциями S3»)

http://updates.rosplatforma.ru/r-virtualization/releases/7.0/x86_64/iso/rvirt-iso-7.0.13-31.iso
Документация по установке и администрированию - <http://docs.rosplatforma.ru/>.

 **Важно!**

Документация, помимо вышеописанной инструкции, служит справочником для глубокого понимания продукта. Используя только документацию без видео и краткой инструкции, будет сложно начинающему корректно развернуть продукт, но и использование только видео инструкции будет недостаточно для полноценной эксплуатации кластера. Поэтому на этапе первого развертывания рекомендуется пошаговое видео инструкция, а для дальнейшей эксплуатации и понимания архитектуры можно использовать документацию.

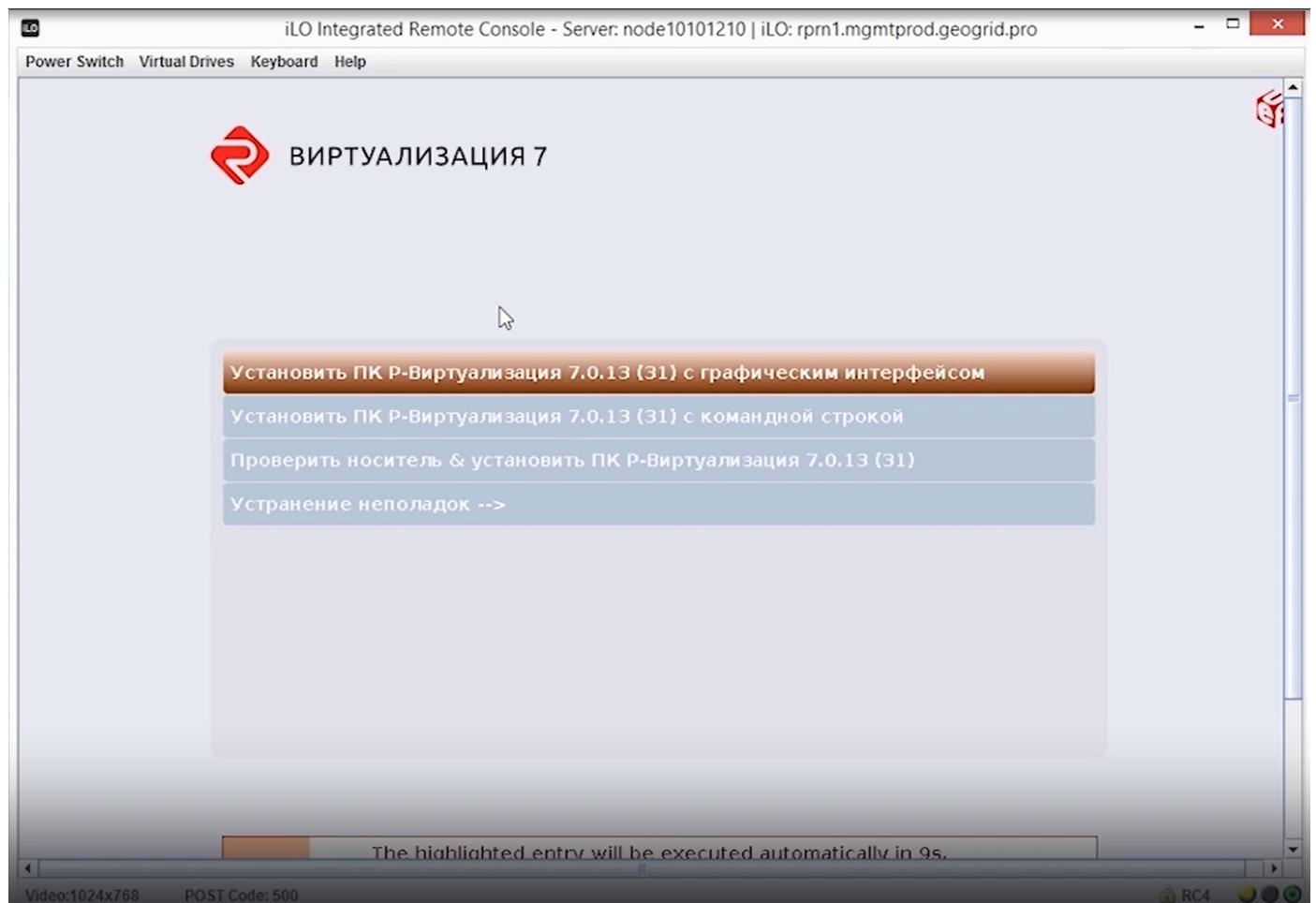
Файлы тестовых лицензий - <https://lic.rosplatforma.ru>

3. Пошаговая инструкция:

Примечание

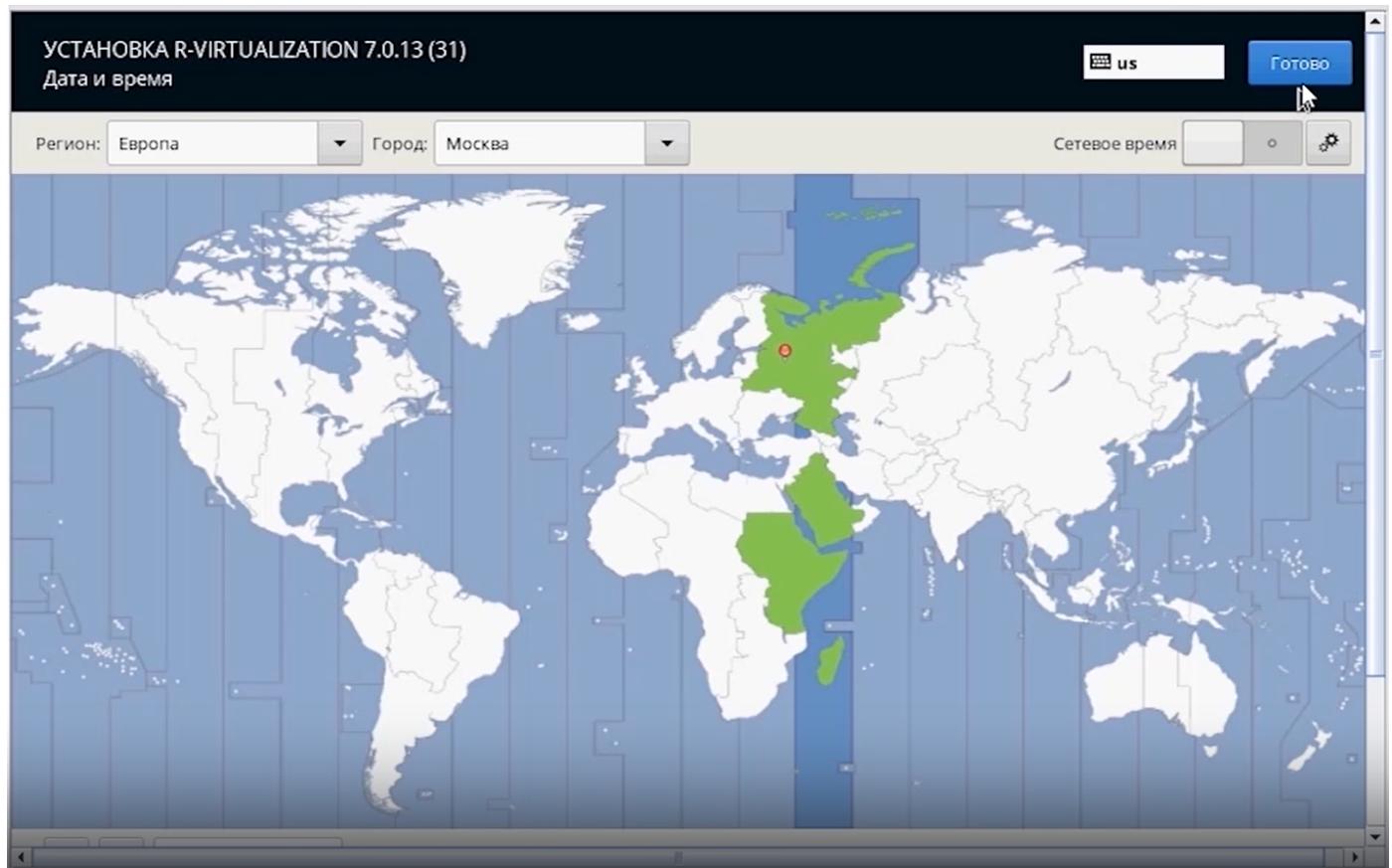
расшифровка видео

3.1 загрузка установщика



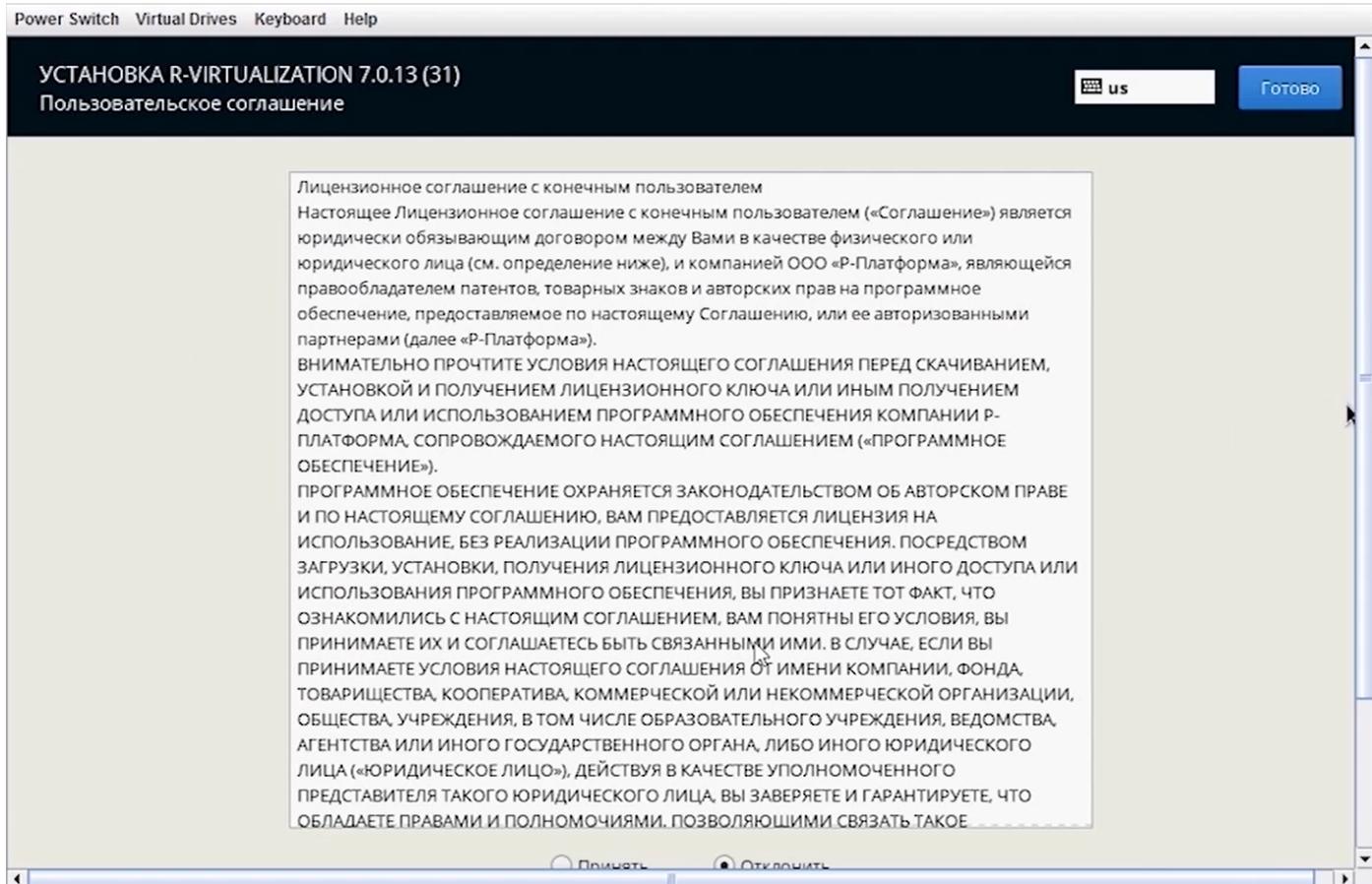
Скриншот 1.

При запуске установщика появится меню как скриншоте 1, где необходимо выбрать первый пункт или если не выбирать, то через несколько секунд система автоматический выберет его.



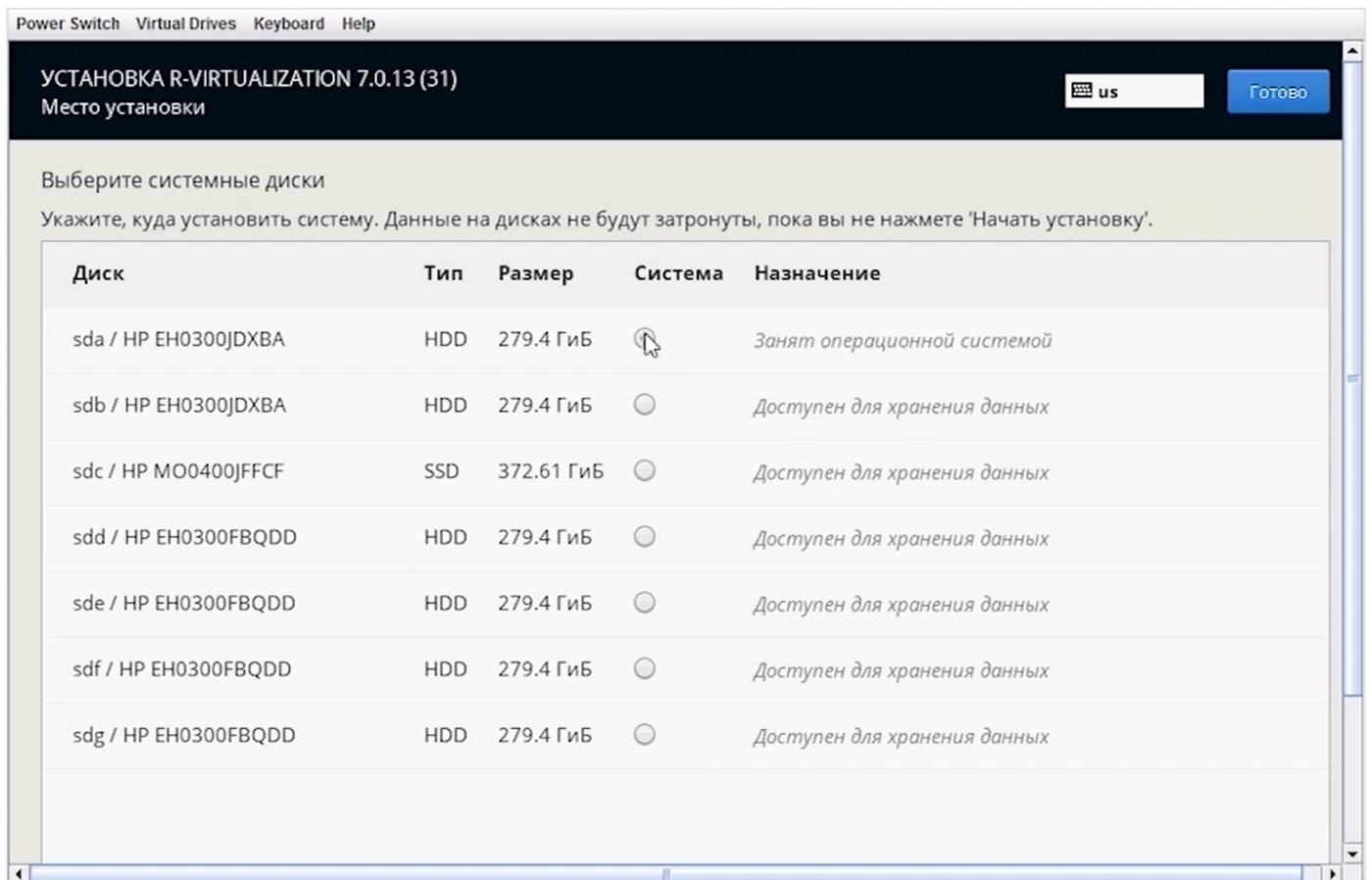
Скриншот 2.

Следующим шагом необходимо выбрать время региона.



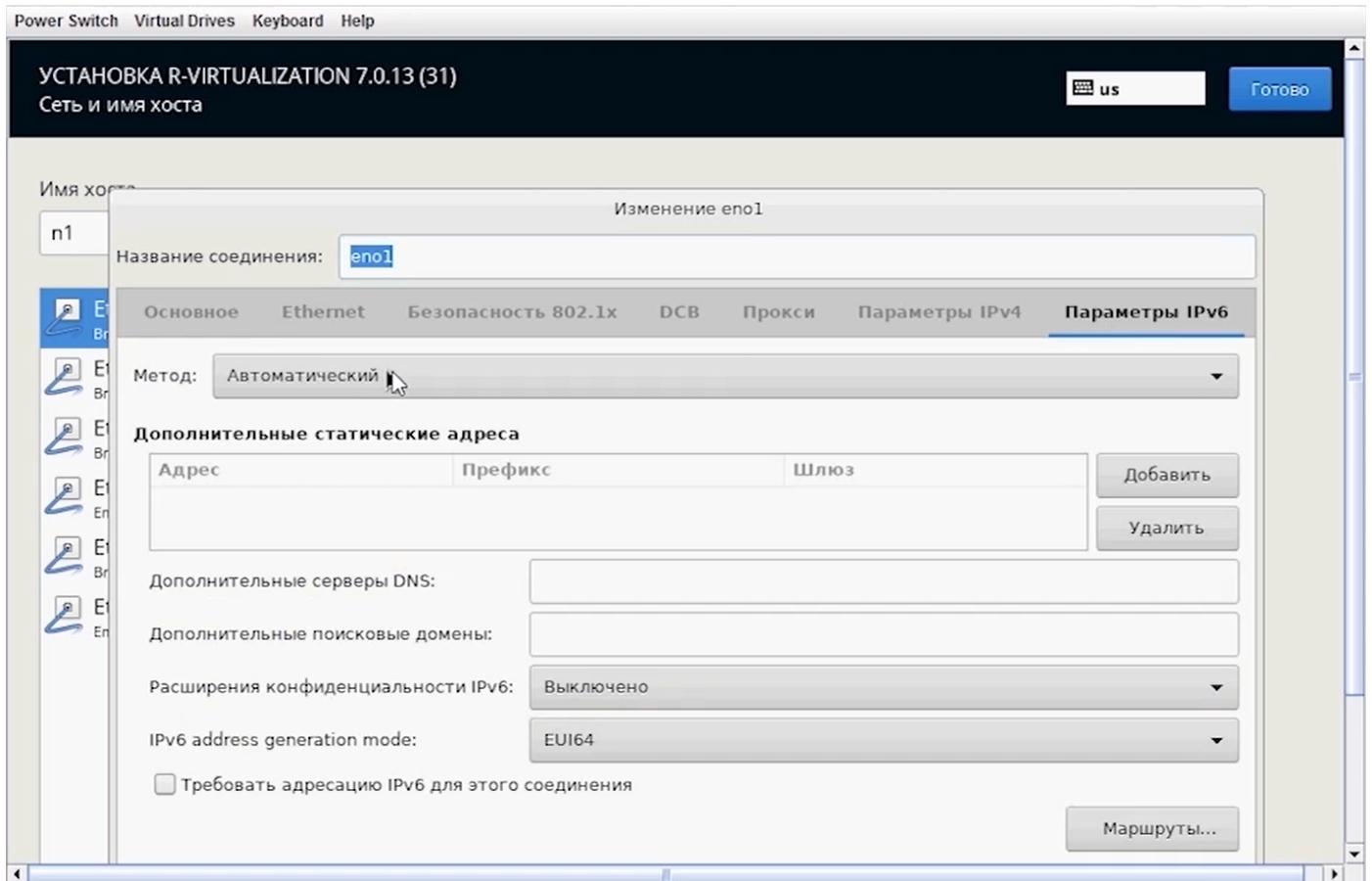
Скриншот 3.

Чтобы продолжить необходимо выбрать принять пользовательское соглашение.



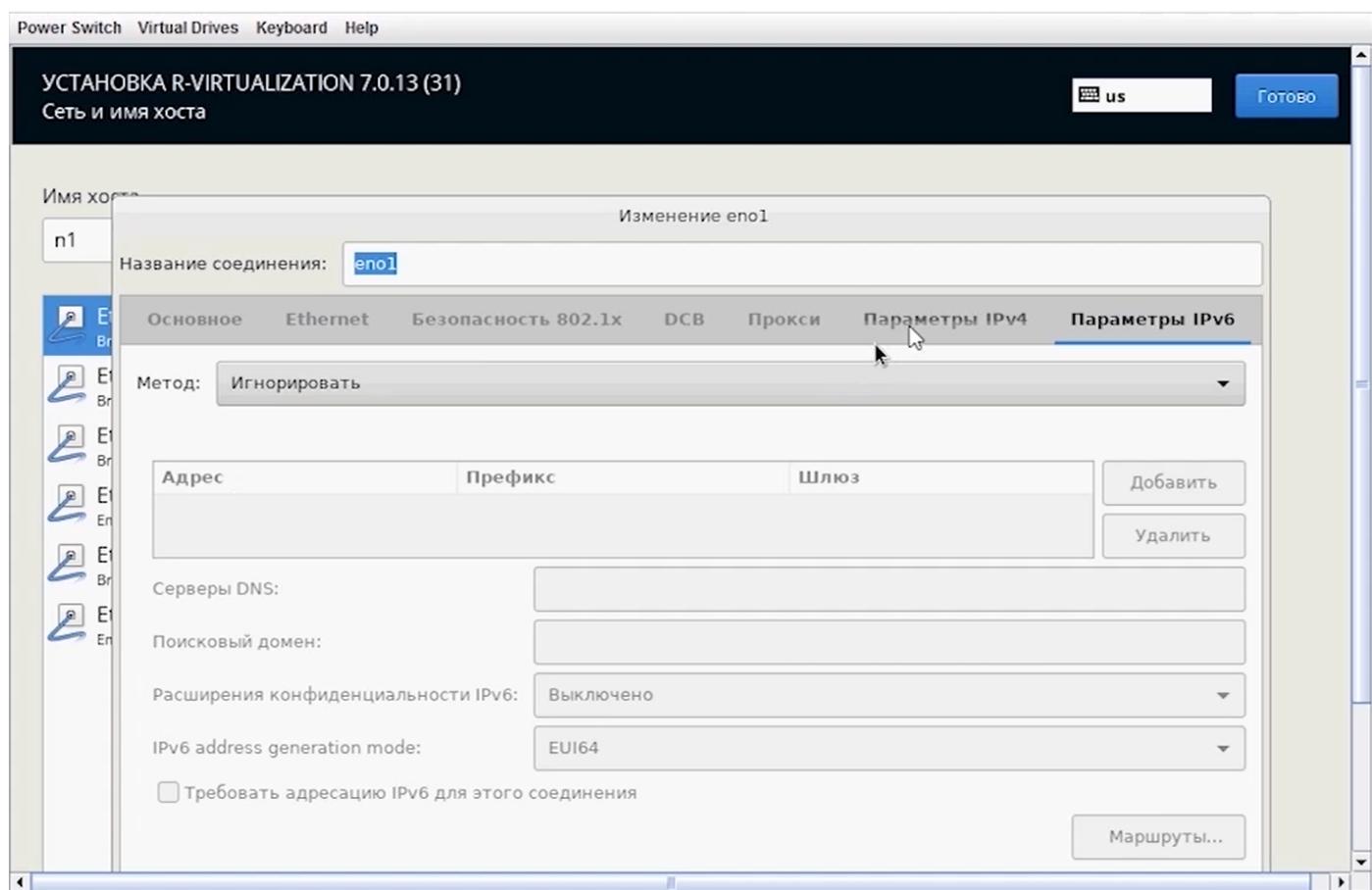
Скриншот 4.

В разделе дисков как на скриншоте 4 выбирается расположение, куда будет непосредственно устанавливаться система. В данном примере на сервере имеется только один **SSD**, и в этом случае его можно использовать под роль КЭШ на шаге настройки Р-хранилища через веб панель, а систему установить на один из **HDD**.

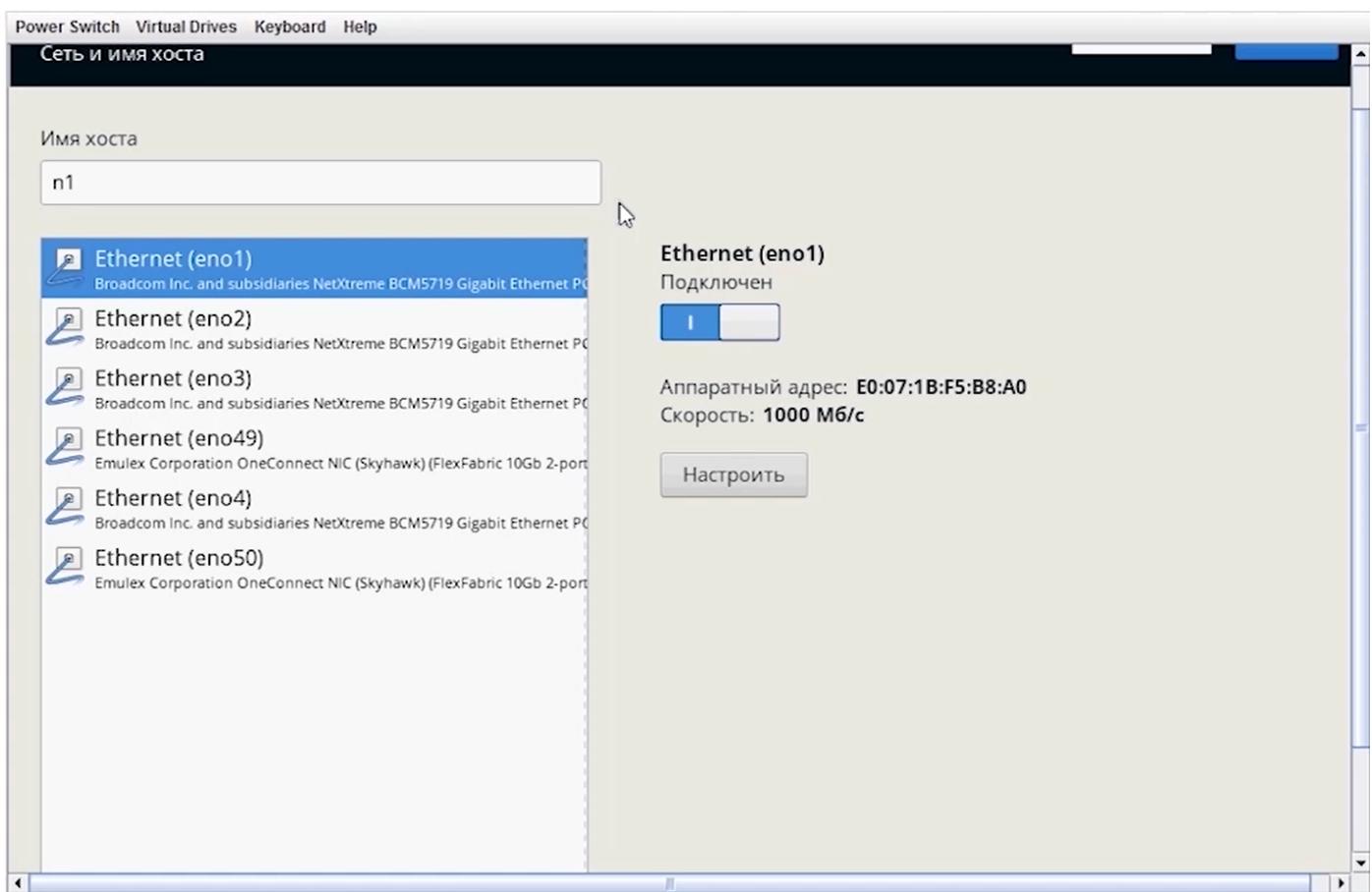


Скриншот 5.

Следующий шаг необходимо выполнить настройку в разделе сети как на скриншоте 5. Это будет первая нода в кластере – первая установка на первый сервер. Укажем имя хоста. Далее проведем первичную настройку и подготовку имеющихся интерфейсов. На всех интерфейсах необходимо выключить **IPv6**, а также **IPv4**.



Скриншот 6.



Скриншот 7.

Важно!

После проведения первичной настройки интерфейсов обратите внимание на то, что все интерфейсы, которые вы планируете использовать в сети управления и в сети хранилища, находятся в статусе "подключен" как на скриншоте 7.

После того как первичная настройка интерфейсов завершена, можно перейти к следующему этапу. Согласно рекомендациям, для обеспечения отказоустойчивости требуется агрегация портов.

Для работы виртуализации требуется создание такого интерфейса, как мост, и при этом с агрегированными портами. Они будут находиться в отдельном **VLAN**, который предназначается для сети управления. Сборка такой связки начинается с создания моста.

✓ Важно!

Самый распространённый сценарий использования сети – это когда сеть панели Р-управления в отдельном **VLAN**. **VLAN** для других сетей можно уже будет сделать после установки и настройки Р-управления.

Выберите тип устройства – выберем **Bridge**. В названии соединения можно указывать не более 16 символов. В нашем примере название соединения будет следующим образом: **br-*mb0.120***. Такое название оптимально с точки зрения удобства администрирования, когда вы собираете связку и в дальнейшем при создании других сетей через панель, она их будет создавать с именами аналогичным образом.

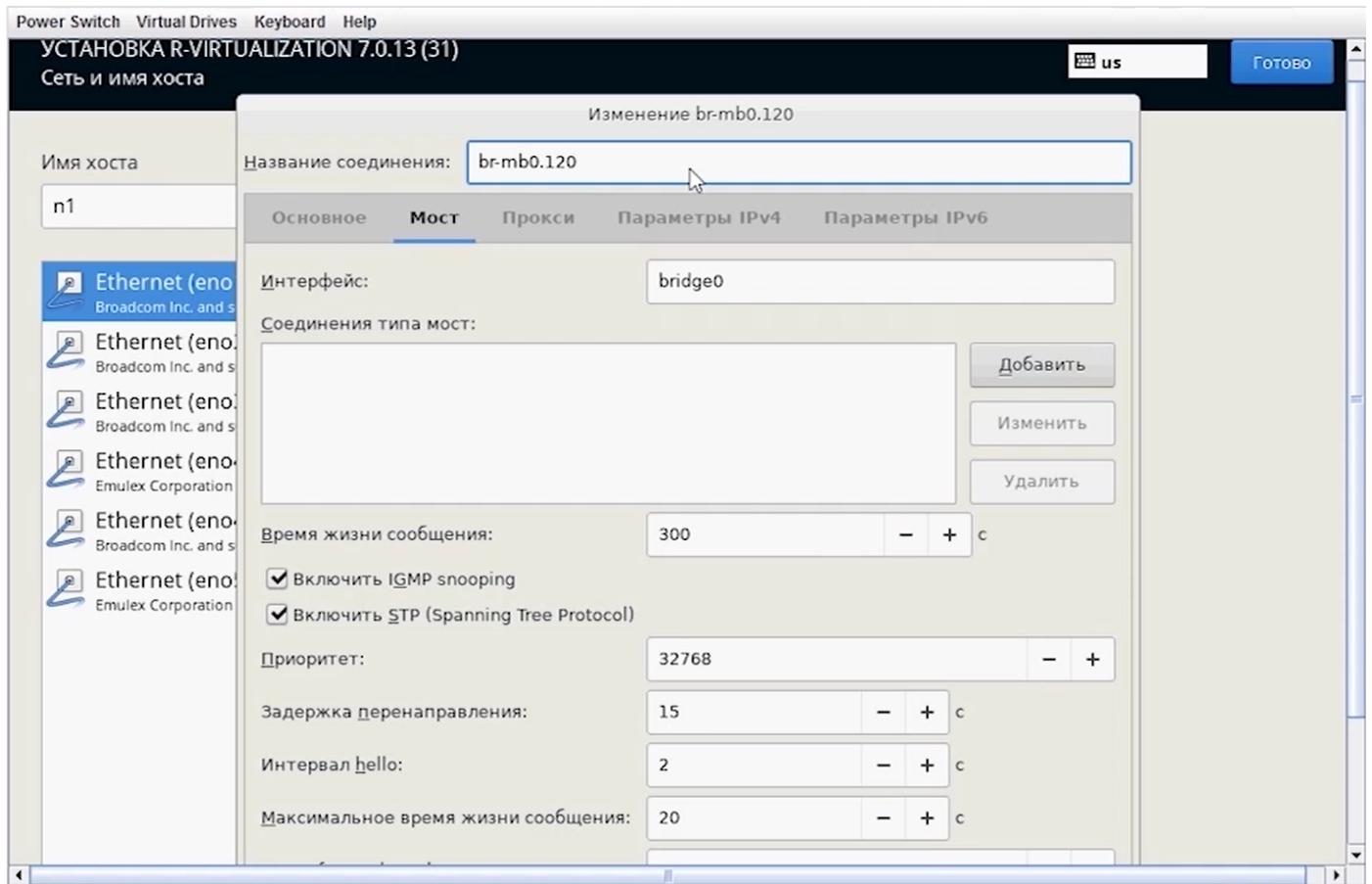
В имени типа создаваемого интерфейса – мост (**Bridge**), дальше идет **mb0**, где **b** означает **Bond** (агрегированные порты), 0 – это его порядковый номер, нумерация начинается с нуля, то есть это первый бонд, первая группа с агрегированных портов, **m** говорит о том, что этот бонд обеспечивает сеть управления (**Management Bond**).

Через точку указывается **VLAN**, в котором будет присутствовать данный агрегированный порт.

✓ Важно!

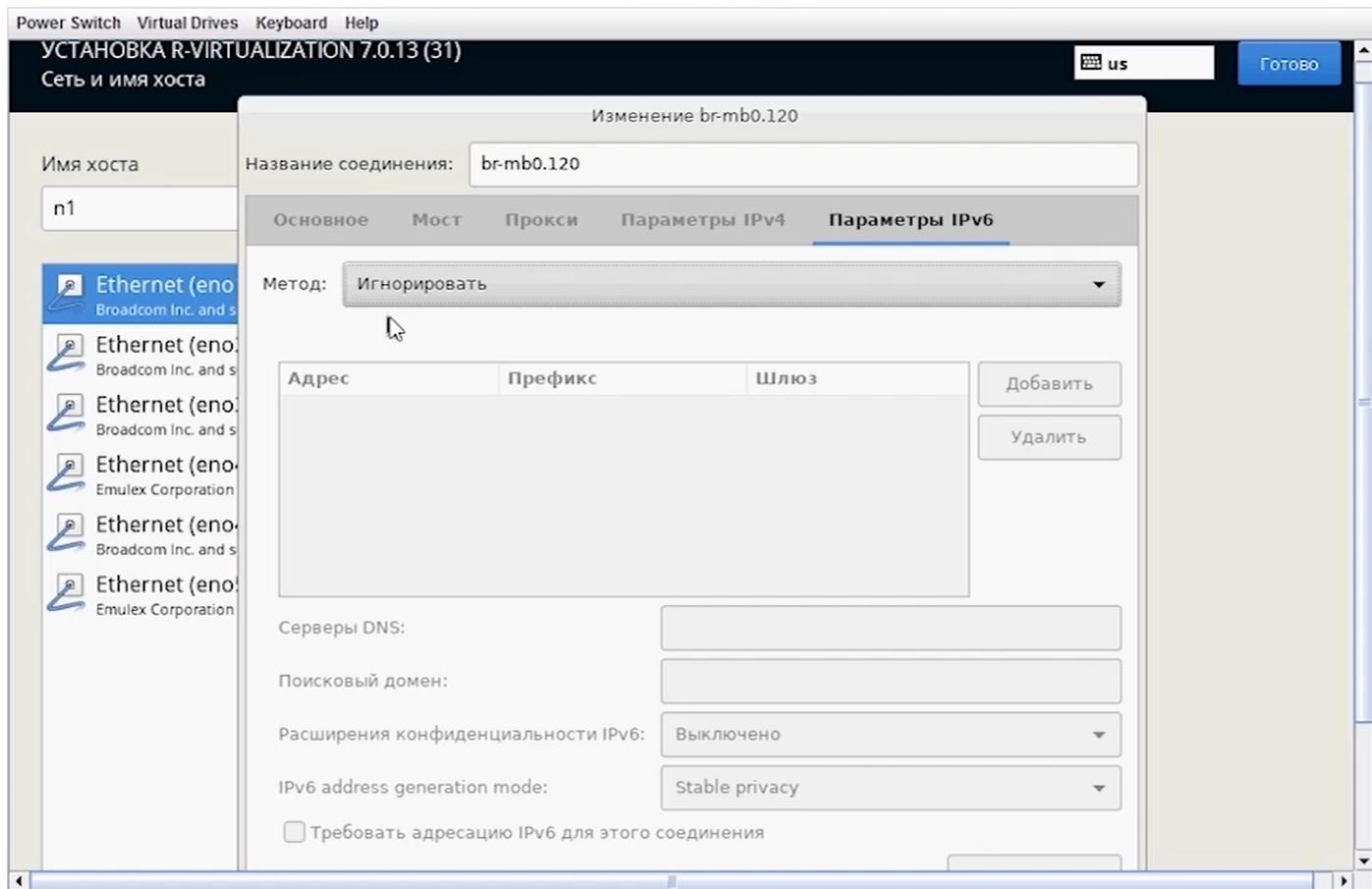
Мы рекомендуем называть таким образом с точки зрения синтаксиса, потому что впоследствии, если вам необходимо будет создать дополнительную виртуальную сеть, то система будет создавать сеть с названием, которое будет иметь аналогичный вид, и в процессе дальнейшей работы и администрирования будет легче ориентироваться среди всех интерфейсов.

Далее в графе "интерфейс" следует продублировать название соединения, укажем **br-*mb0.120*** как на скриншоте 8.

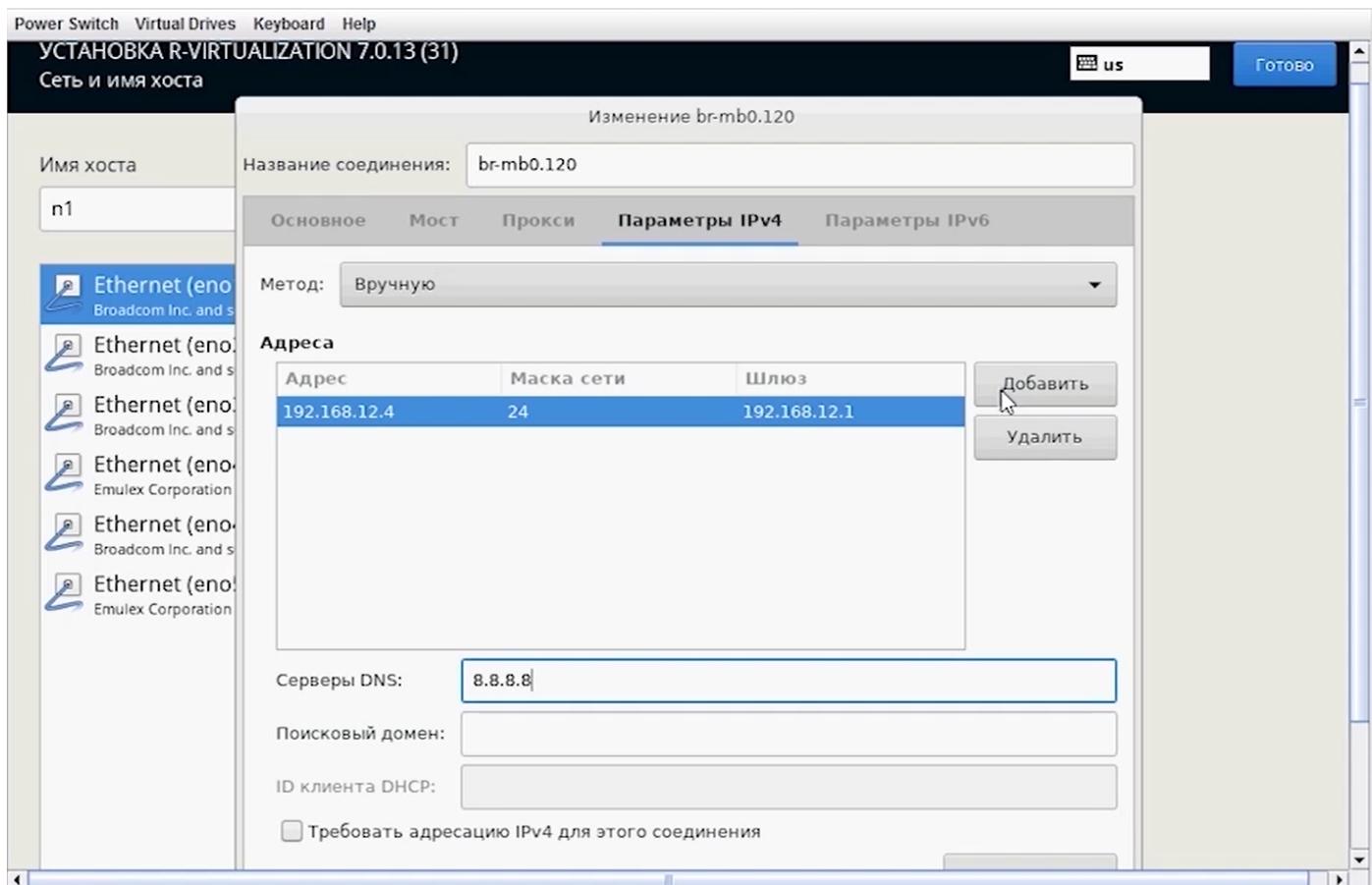


Скриншот 8.

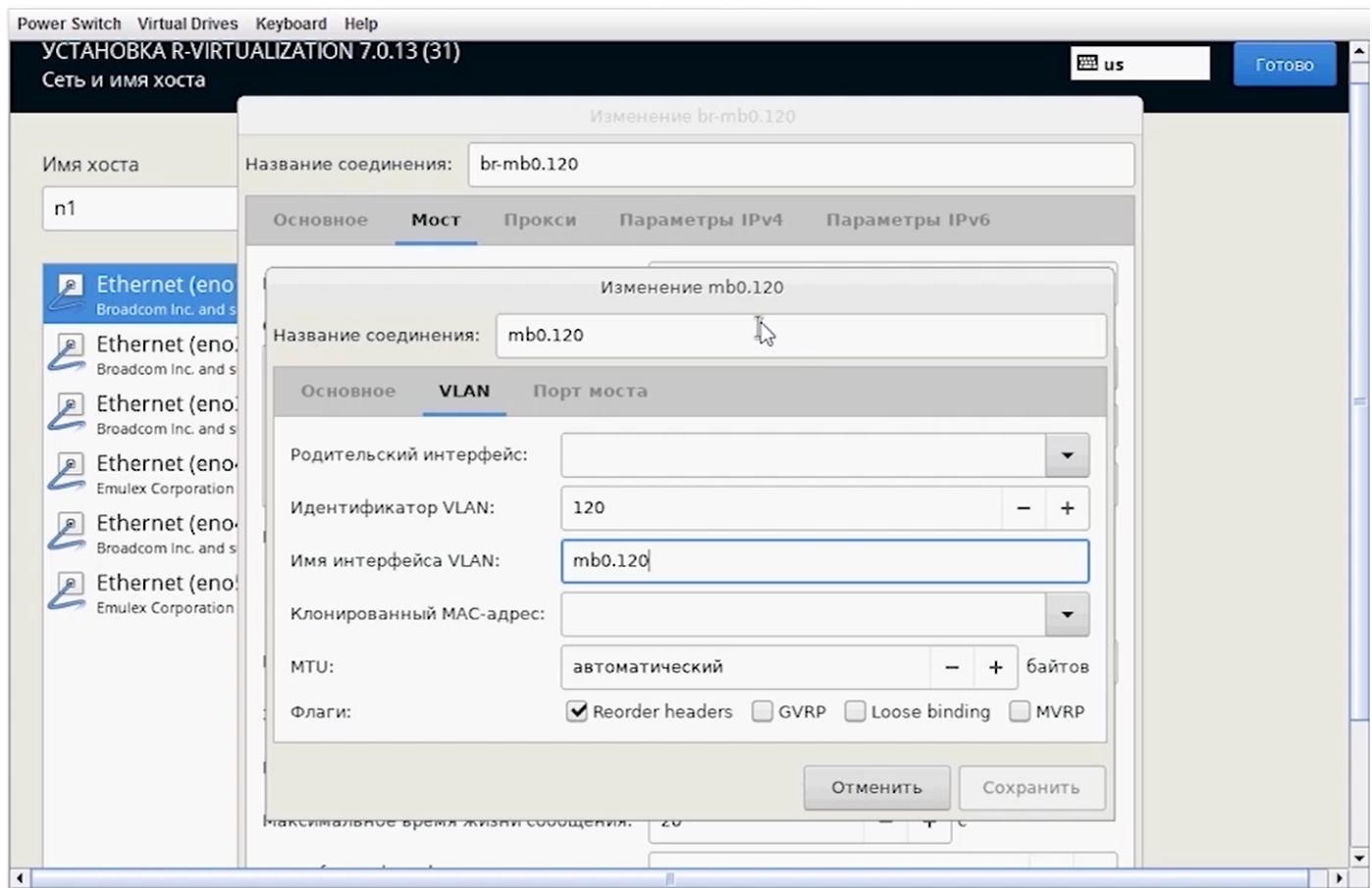
Следующим шагом необходимо задать IP-адрес ноды на скриншоте 10. Параметры **IPv6** мы рекомендуем всегда отключать, поэтому укажем в графе метод "игнорировать". Параметры **IPv4** назначаем вручную как показано на скриншоте 9.



Скриншот 9.

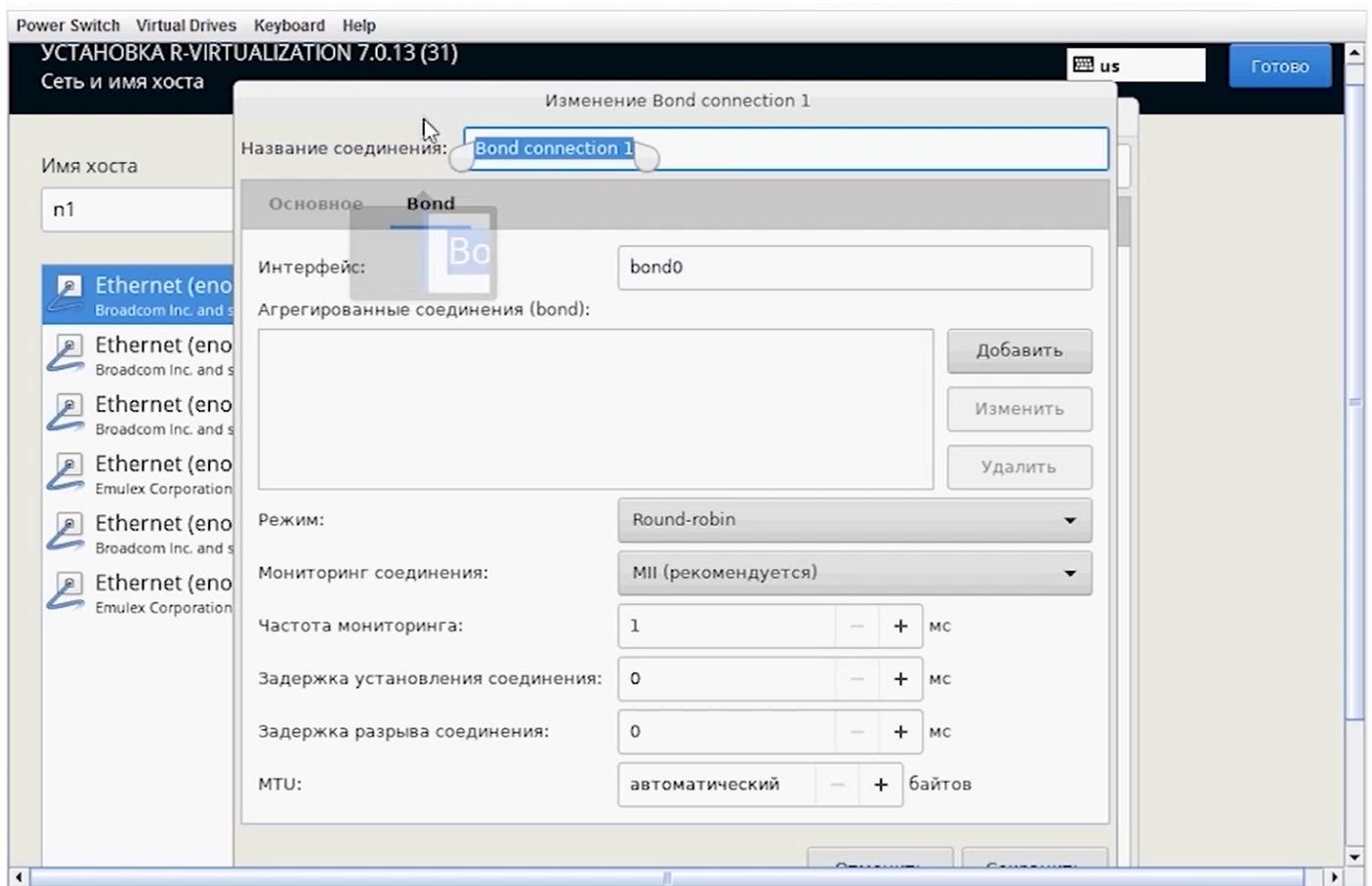


Скриншот 10.



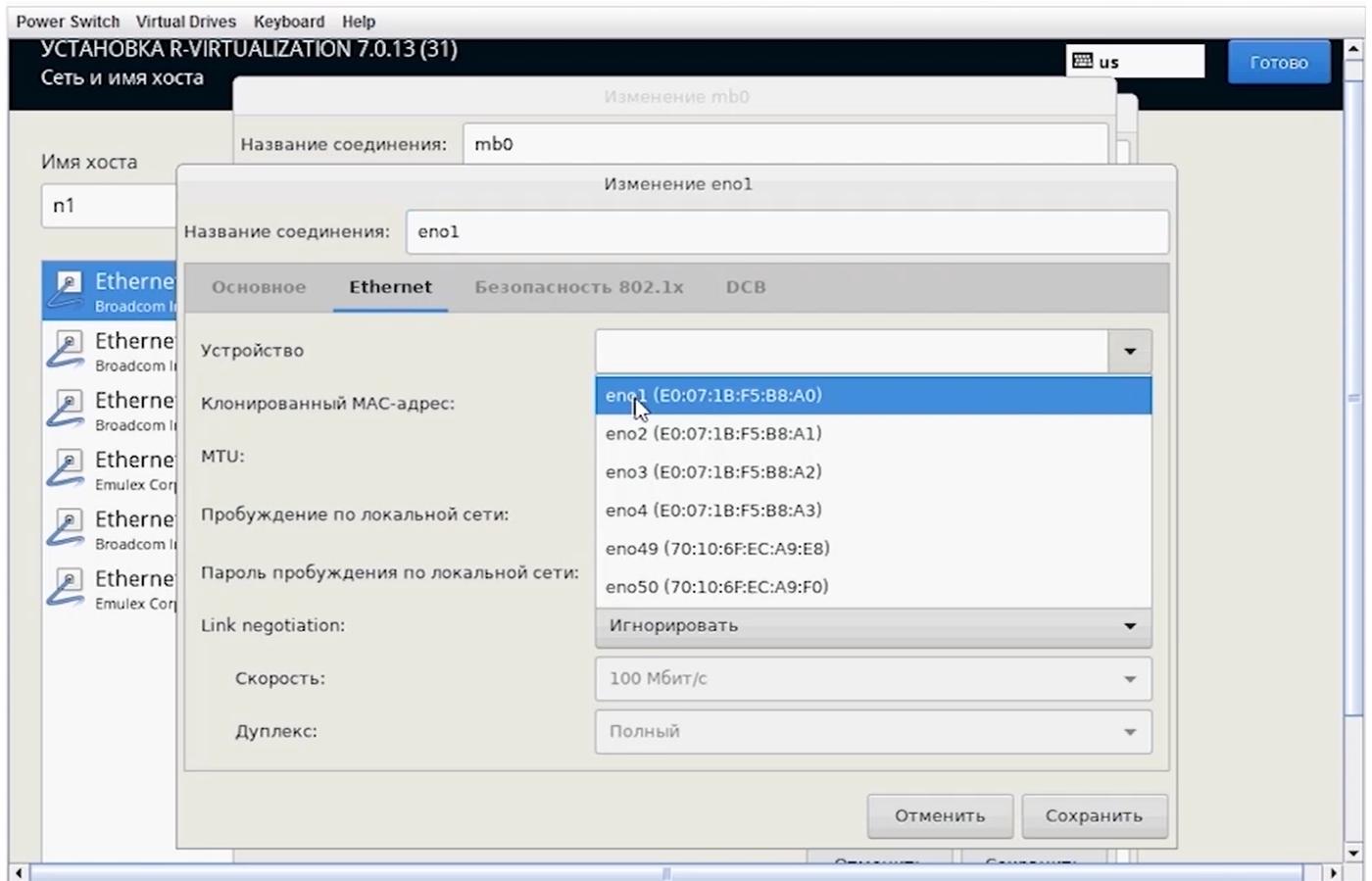
Скриншот 11.

Далее необходимо создать **VLAN** в разделе мост, в котором будет находиться **bond**. Название соединения указывается такое же, каким оно было на этапе создания моста, но только без предшествующего префикса **br**. Подобные названия в дальнейшем будут необходимы при администрировании имеющихся виртуальных сетей. Тег **VLAN** – 120 на скриншоте, у вас будет свой номер в зависимости от настроек вашей сети (физического коммутатора). Имя интерфейса **VLAN** – тут следует продублировать название соединения.

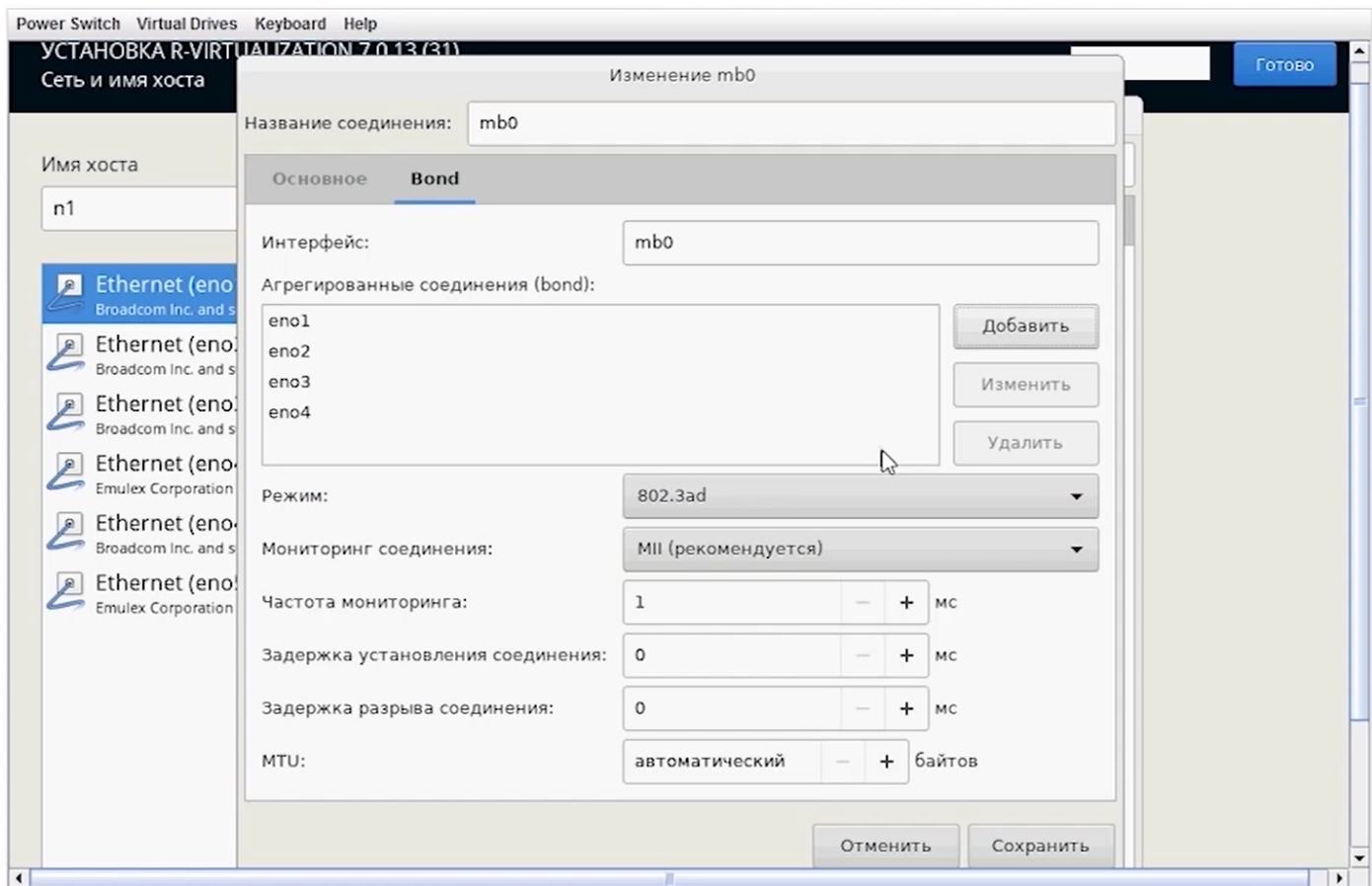


Скриншот 12.

Следующим шагом необходимо добавить **Bond**. В поле “Родительский интерфейс” - в выпадающем меню можно выбрать графу создать соединение. Тип соединения - **Bond**. Название соединения выставляем **mb0**, интерфейс продублируем **mb0**, режим поставим **LACP** он же **802.3ad**, и теперь добавим интерфейсы, которые будут в качестве **slave**.



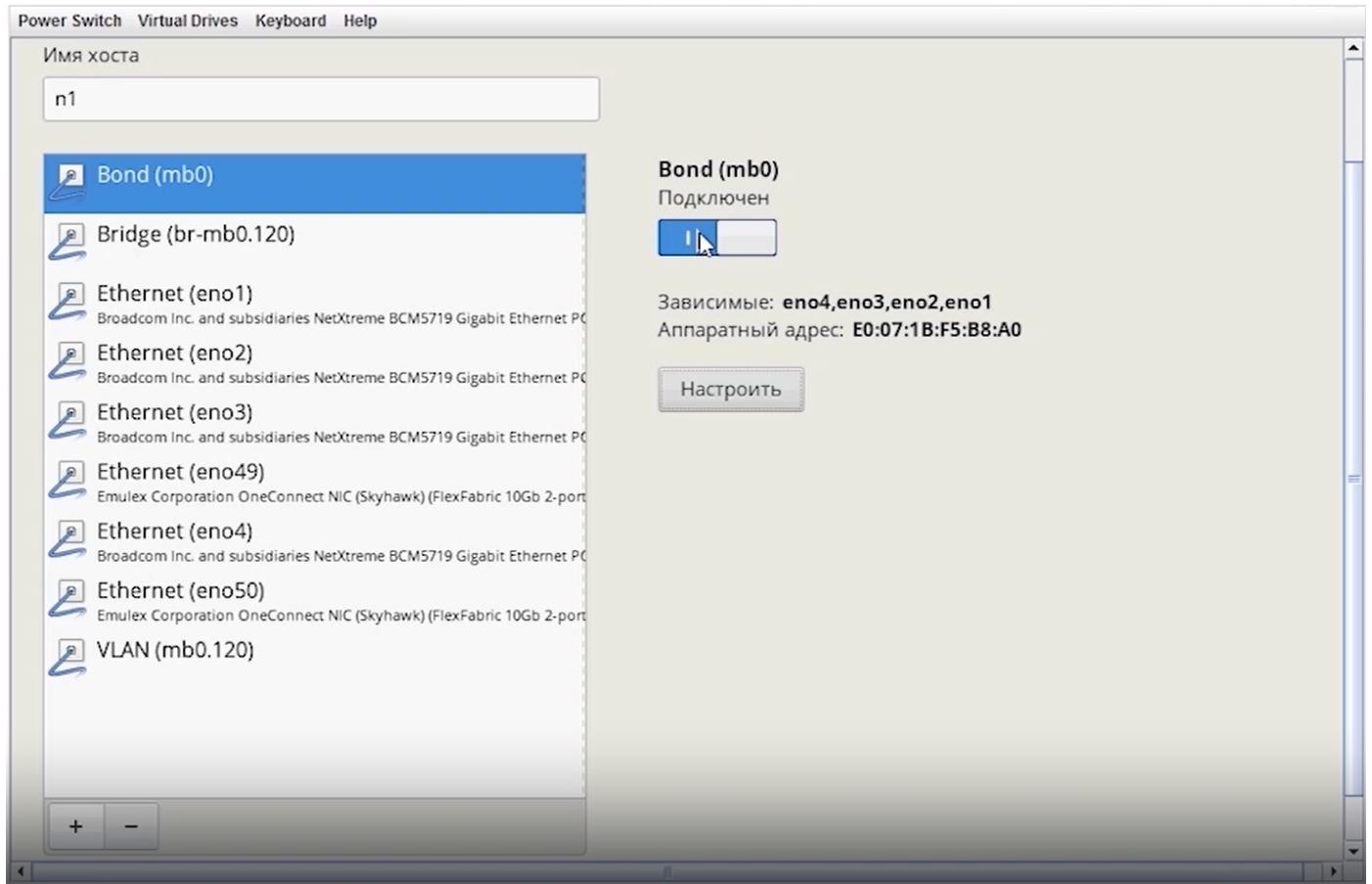
В названии соединения для последующего администрирования прописываются такие же названия интерфейсов, какие назначила система.



Скриншот 14.

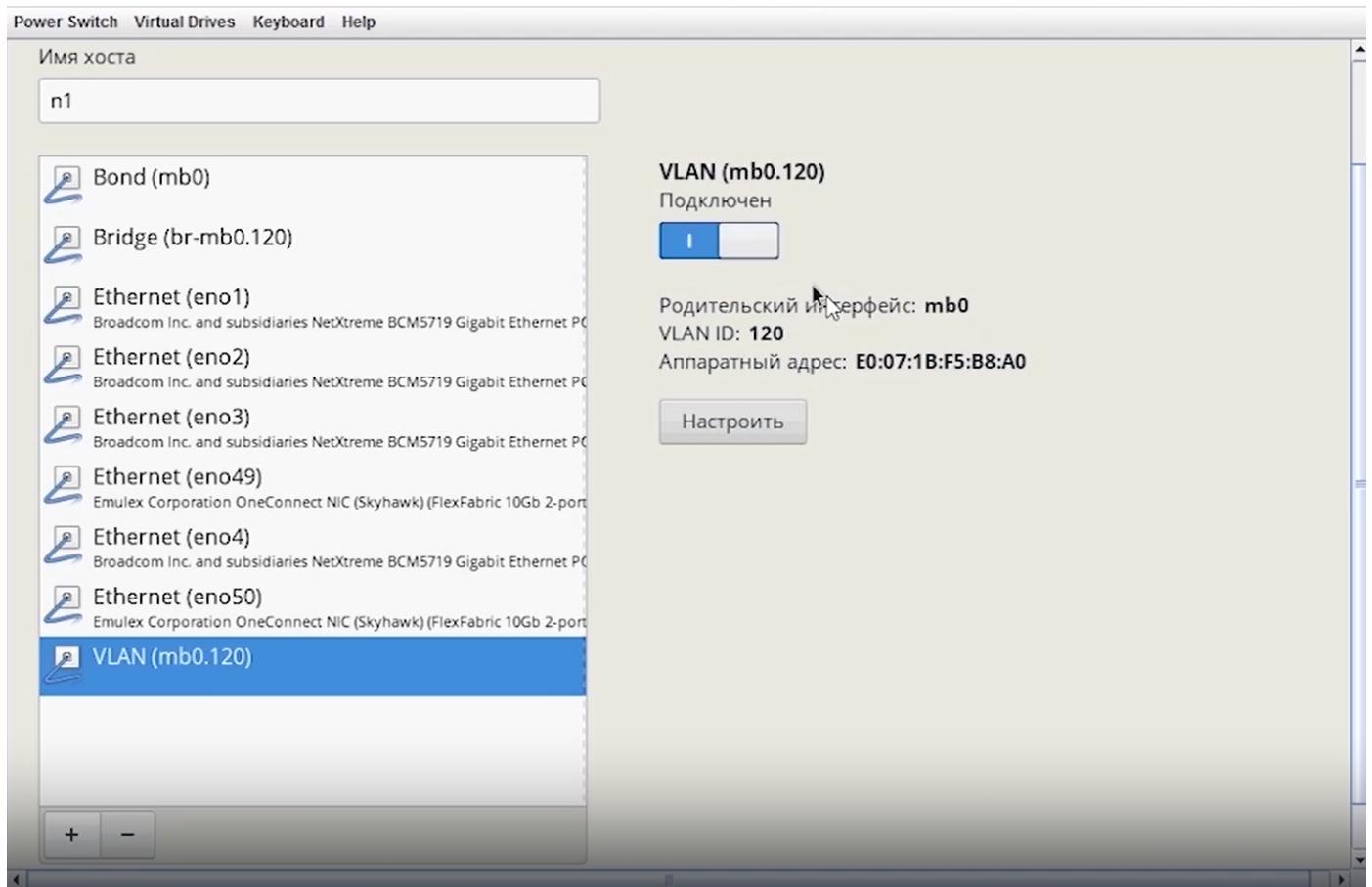
После того как собрана вся связка необходимо по порядку во всех открытых окнах нажать кнопку "сохранить": **Bond**, **VLAN** и мост.

Следующим шагом перейти в настройки **Bond**. В параметре **IPv6** и обязательно не забудьте это сделать, установите опцию "игнорировать". Далее перейдите в параметры **IPv4** и выключите. Это необходимо сделать, для исключения **flapping**-а сетевых интерфейсов.



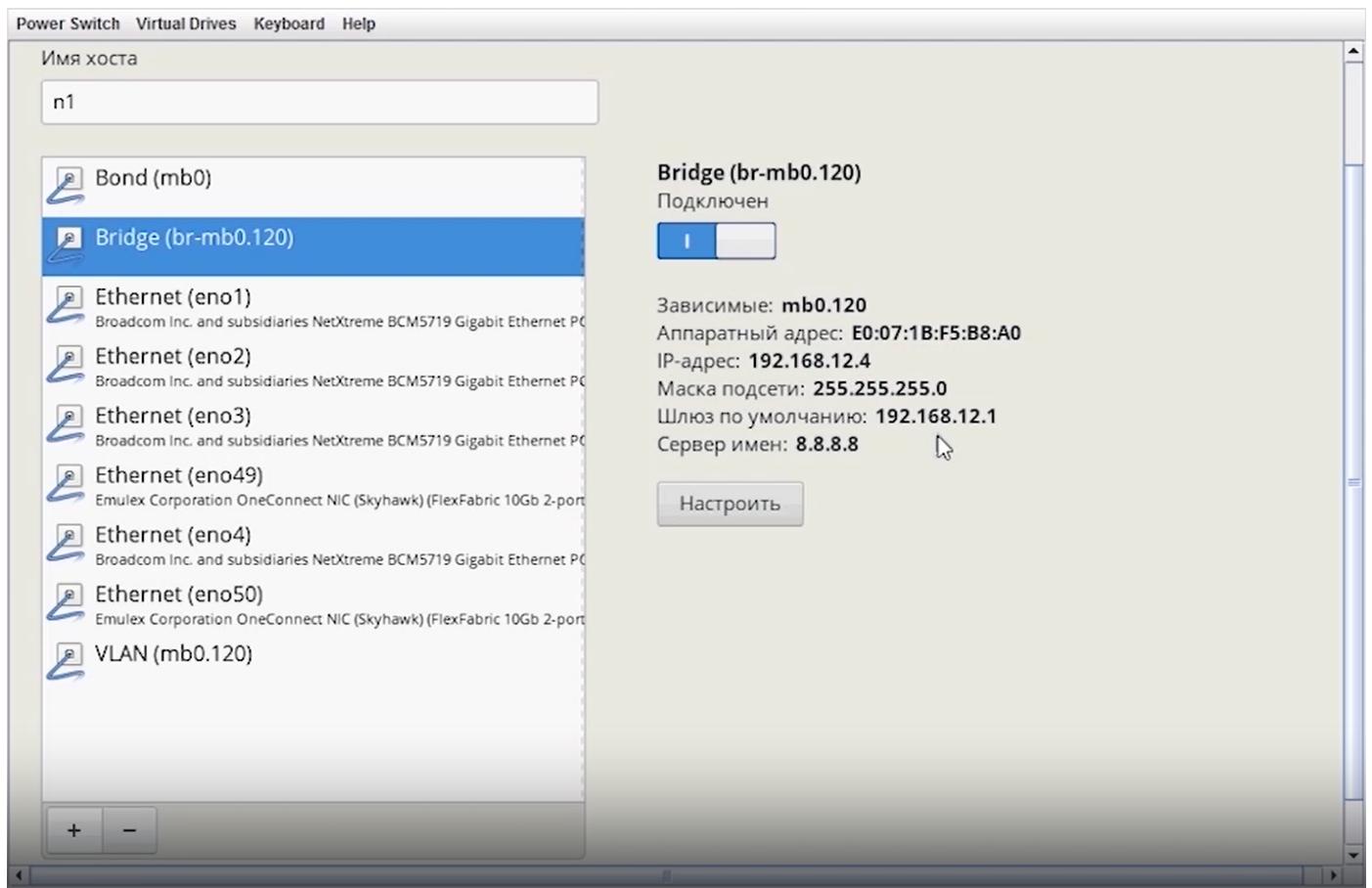
Скриншот 15.

Когда **Bond** включен, он отобразит зависимые интерфейсы как на скриншоте 15. Как минимум должно быть два зависимых интерфейса.



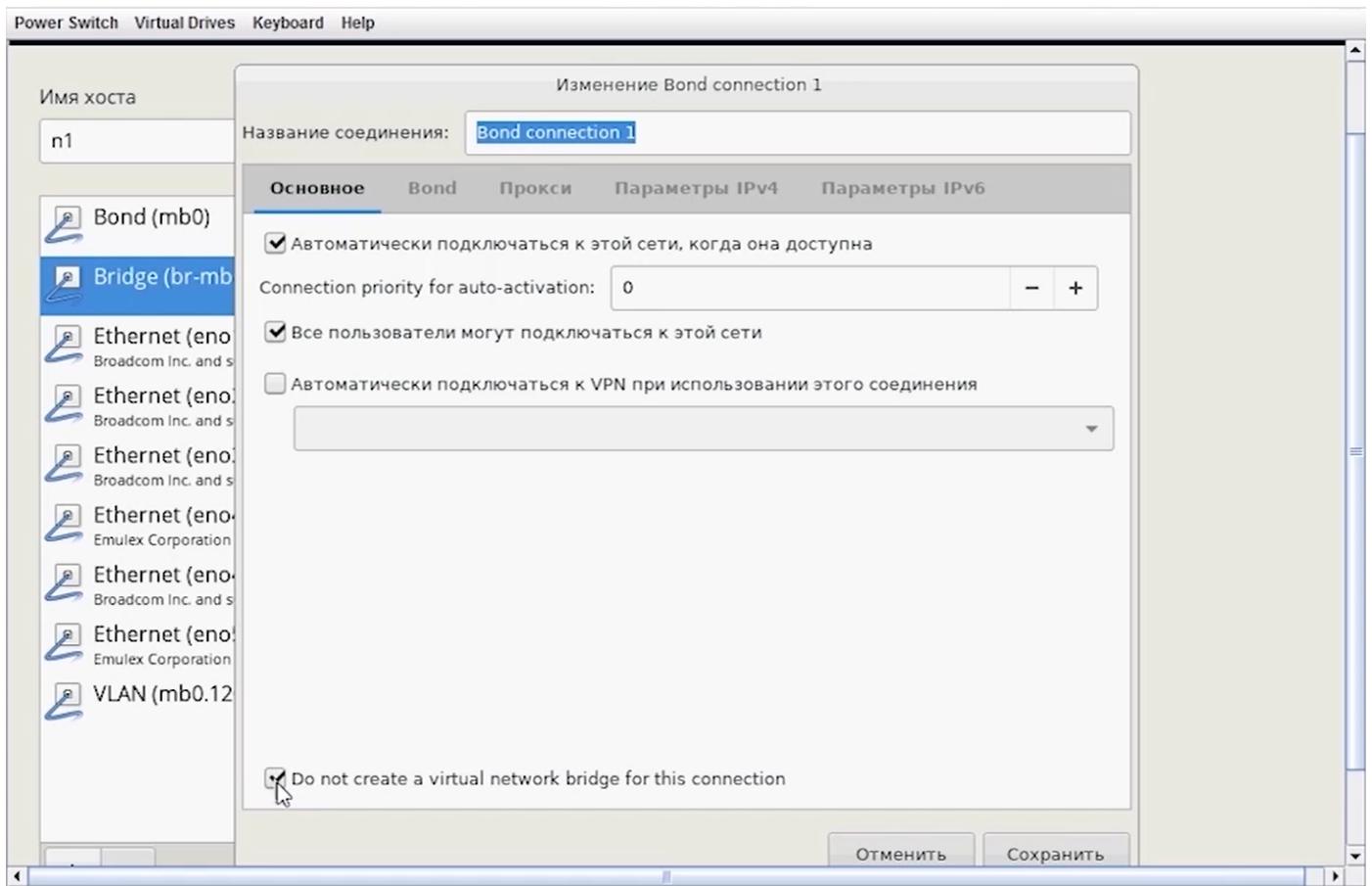
Скриншот 16.

Также должен быть виден в списке **VLAN**, который создавался, и которого родительский интерфейс **Bond**, и мост с сетевыми настройками, которые принадлежат ноде.



Скриншот 17.

Если у вас получилось так как на предыдущих скриншотах и на скриншоте 17, то далее переходим к настройке сети хранилища. Для работы сети хранилища не требуется мост, а также не используется **VLAN**, поэтому переходим сразу к созданию **Bond'a**. Первым делом перейдем во вкладку «основное» и установим галочку, чтобы не создался мост автоматически как на скриншоте 18.



Скриншот 18.

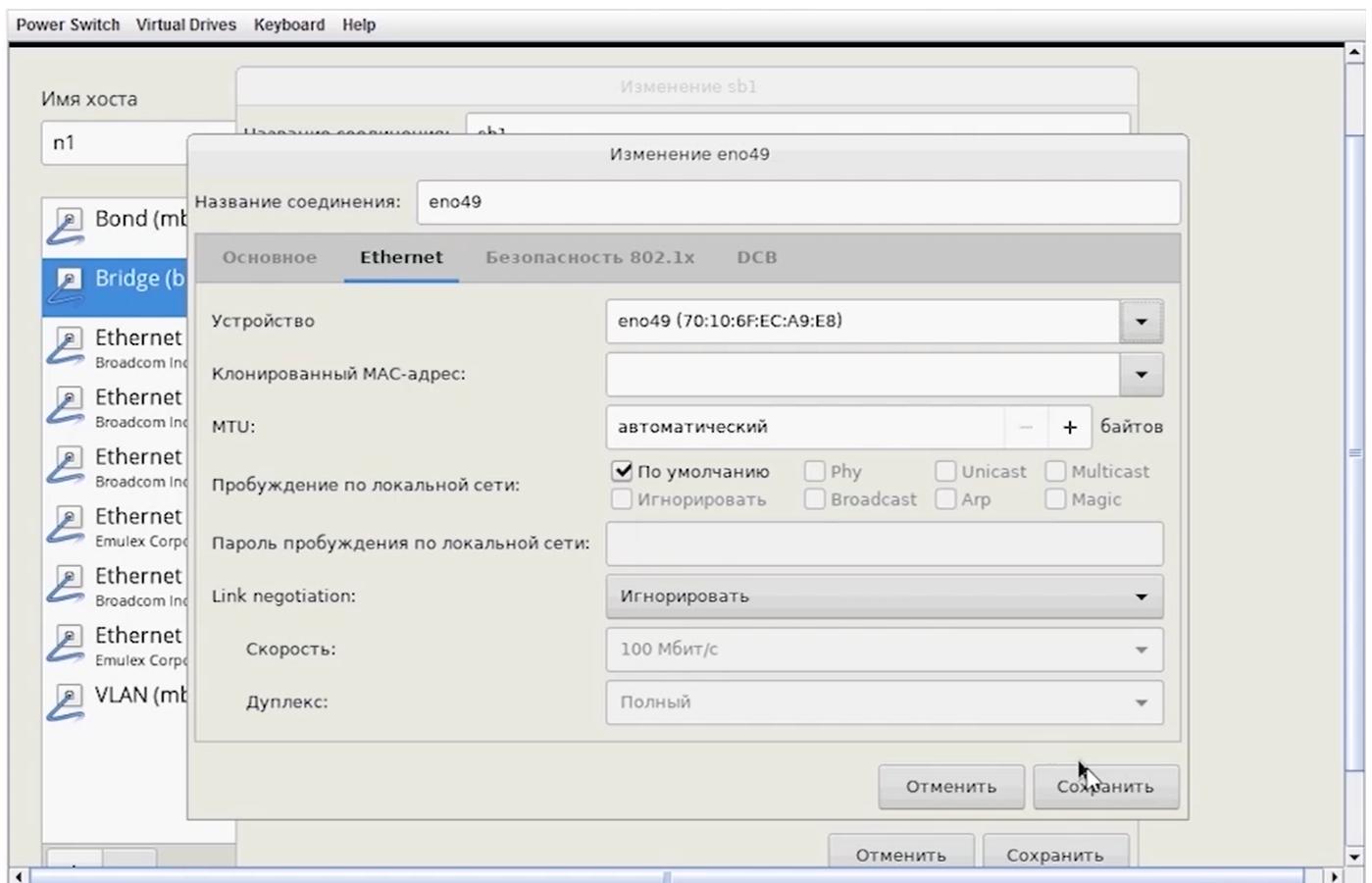
Название соединения прописываем **sb1**, потому что это уже второй Bond по нумерации, поэтому будет цифра один – Storage **Bond 1 (sb1)**. Интерфейс - продублируем название **sb1**. Параметры **IPv6** мы рекомендуем отключать, выставляя значение игнорировать. Поставим в графе параметры **IPv4** "вручную", следует указать тот IP-адрес, который предполагается для сети хранилища на этой ноды.

Прописывается только IP-адрес и маска сети. В настройках сети хранилища не должно быть ни шлюза, ни DNS. Режим выставляем на **LACP (802.3ad)** как на скриншоте 20 и добавим **slave** интерфейсы для агрегации как на скриншоте 19. В текущем случае это будут два интерфейса по 10 Гигабит. Назовём соединение **slave** интерфейсы также, как их называла система.

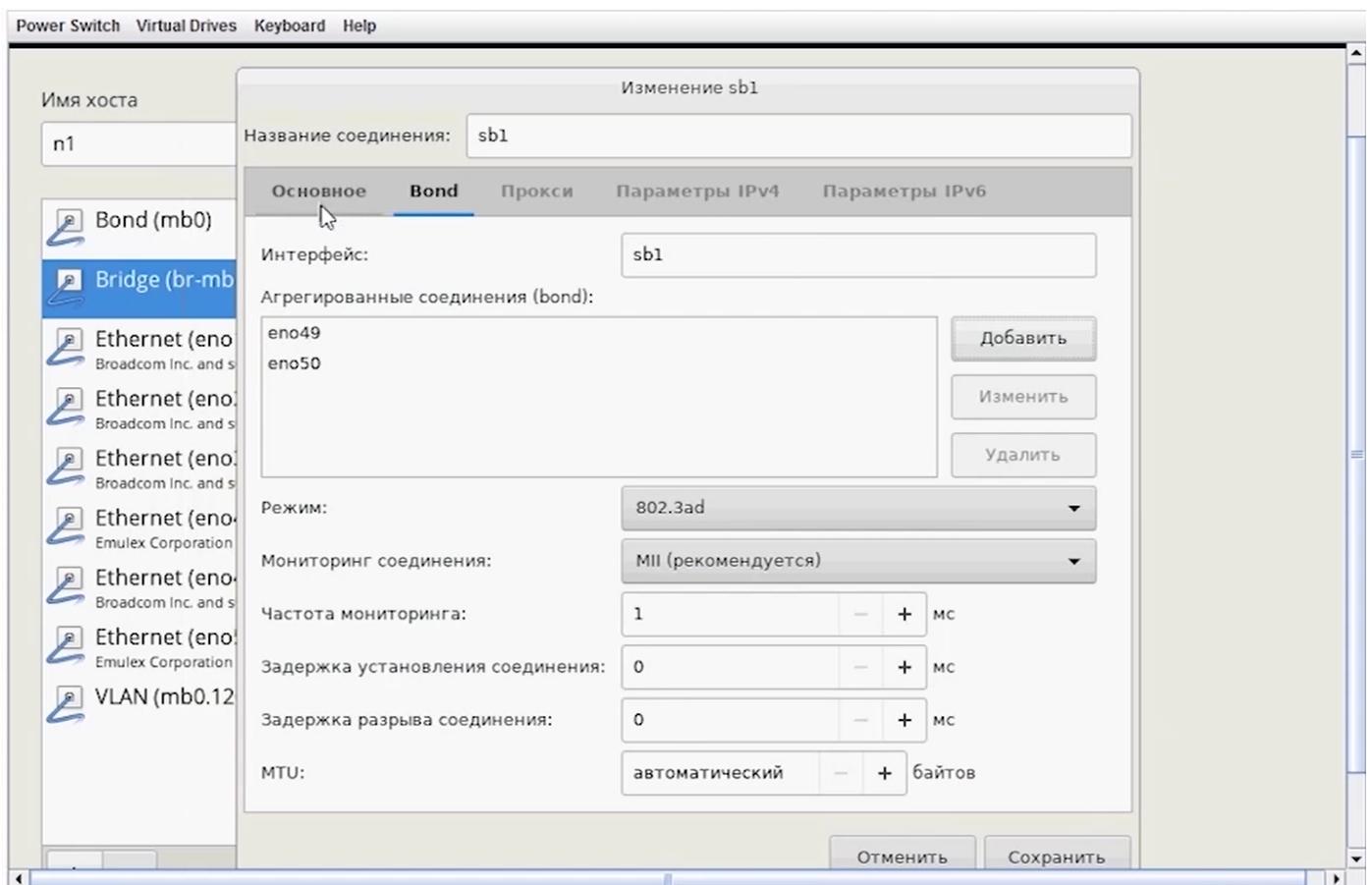
Важно!

После того как настройки сети для хранилища завершены можно проверить с других устройств (если есть такая возможность) из этой же сети с помощью **ping** доступность настроенного адреса ноды. Для сети управления также можно выполнить эту же проверку только со стороны устройств, имеющих адресацию из этой же сети.

В случае отсутствия **ping** необходимо перепроверить правильность выполнения проверки, а далее перепроверить все настройки, выполненные выше. Если все в порядке, **ping** должен быть непрерывным в противном случае необходимо проверять настройки коммутационного оборудования или аппаратные проблемы.



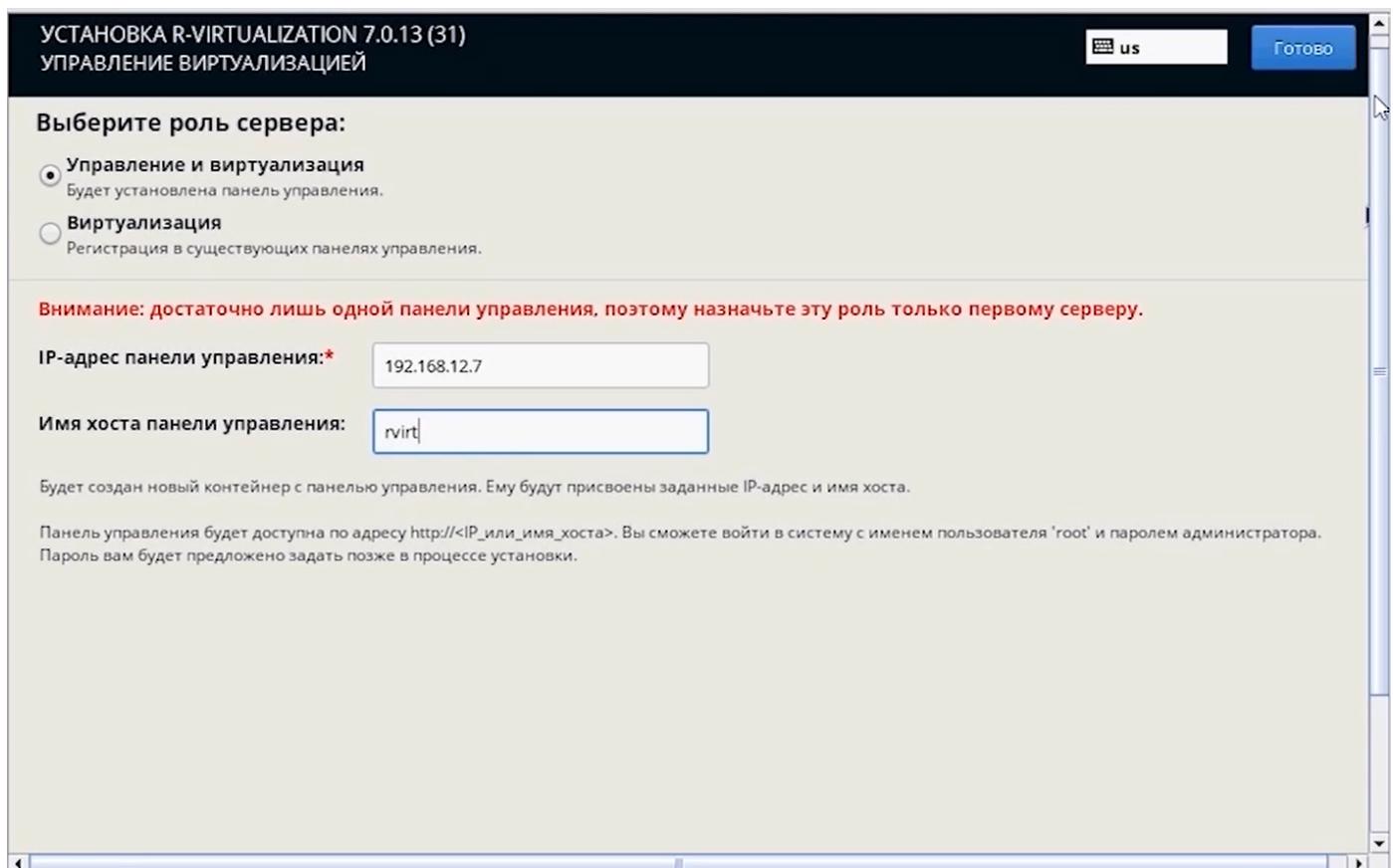
Скриншот 19.



Скриншот 20.

3.2 предустановочные настройки первого сервера с контейнерами управления

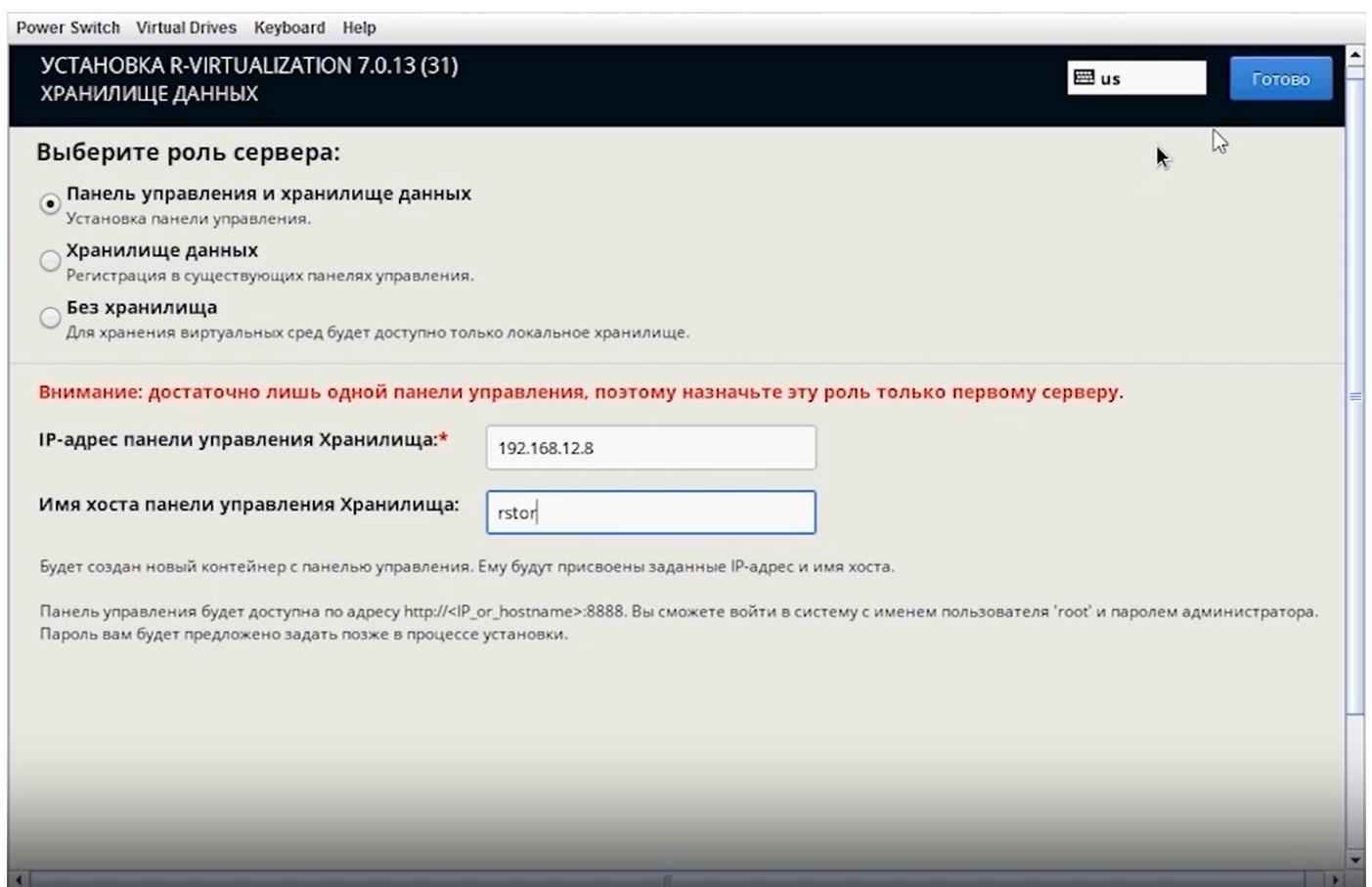
Следующий шаг – это настройка панели управления. Это делается при установке только первой ноды, следует отметить пункт “управления и виртуализации” как на скриншоте 21, чтобы была установлена панель управления, и необходимо назначить IP-адрес панели управления. Должен быть IP-адрес, который заранее выделен и свободен для веб-интерфейса Р-управления, и он обязательно из этой же сети, которая предусмотрена для сети управления. На этапе установки имя хоста панели управления можно прописать любое или ассоциированное с тем, что это будет относится к панели управления виртуализацией, которая может находиться на любых хостах-участниках кластера.



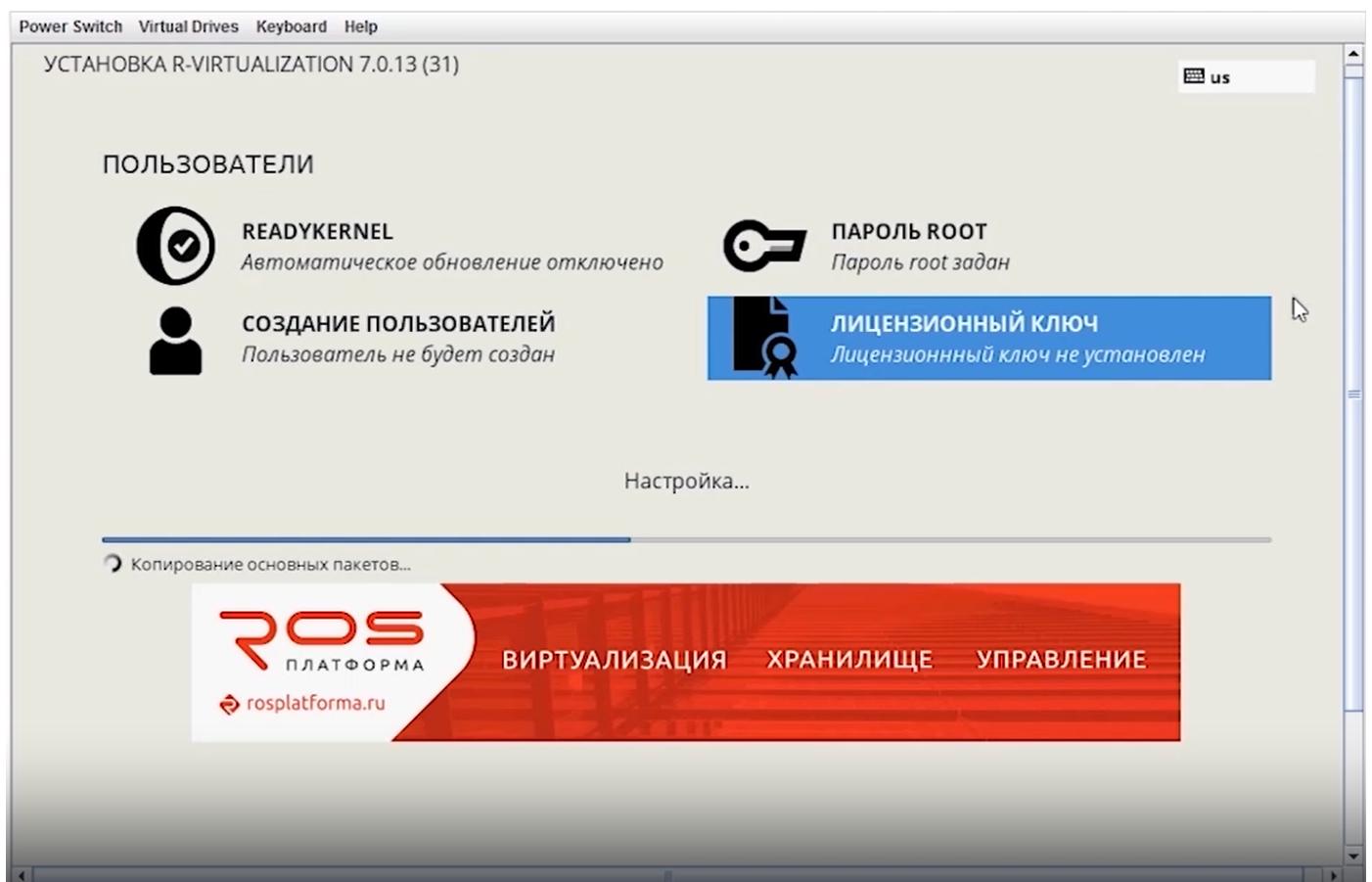
Скриншот 21.

Далее настройка панели **P-Хранилище** также при первой установке на первую ноду. Надо выбрать самый первый пункт панель управления и **P-хранилище** для установки панели управления скриншот 22.

IP-адрес панели управления **P-хранилища** также должен быть в сети управления, и также надо указать заведомо свободный для веб-интерфейса IP-адрес. Имя хоста панели управления хранилищем можно ассоциировать с тем, что это будет панель для визуального управления, так как сам кластер P-хранилища может работать и управляться без панели после завершения всех настроек.



Скриншот 22.

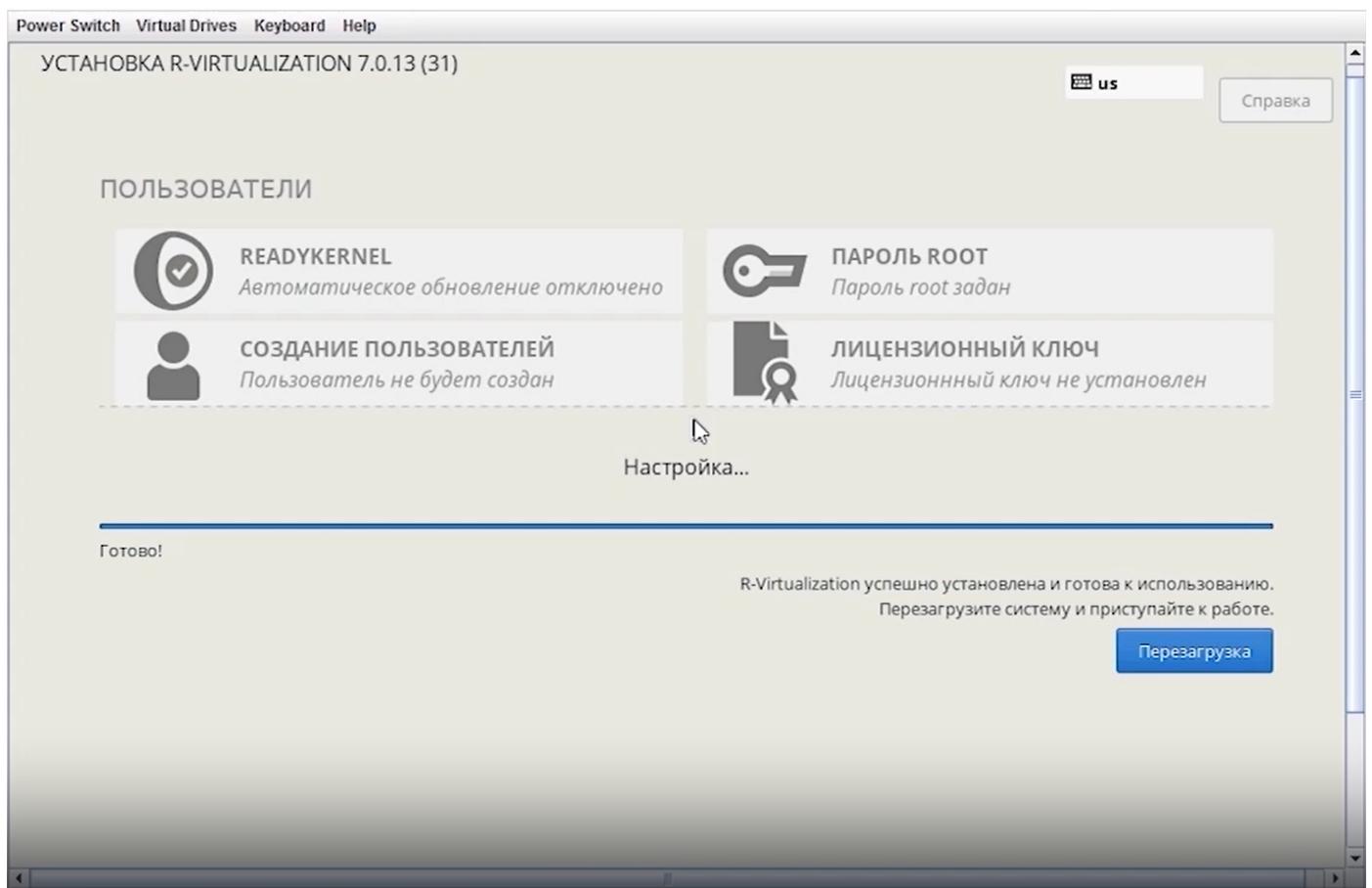


Скриншот 23.

3.3 дополнительные настройки в процессе установки

На этом этапе первичные настройки завершены, можно перейти непосредственно к установке Росплатформы на сервер, где в следующем открывшемся меню можно перейти в **ready kernel** и можно выключить функцию автоматическое обновления ядра, далее зададим пароль для пользователя **root**. Создание дополнительных пользователей на этом этапе не требуется, это можно сделать после.

Перейдем во вкладку "лицензионный ключ", оставим поле пустым, потому что лицензионный ключ будет вводиться позже после установки, нажмем "готово" и дождемся завершения установки.



Скриншот 24.

3.4 проверка доступности контейнеров управления

После перезагрузки ноды, можно на нее зайти, используя логин **root** и пароль по **SSH**, который указывался при установке ноды. Также проверим доступность веб-интерфейса Р-управления и проверим доступность контейнера управления Р-хранилища по тем IP-адресам, которые назначали при установке.

УПРАВЛЕНИЕ 7

Инфраструктура

Обзор Физические серверы Виртуальные среды Ресурсы Журналы Резервные копии Безопасность

Общий статус

- Для 1 физических серверов требуется вмешательство администратора.
- Все виртуальные среды работают в нормальном режиме.

Виртуальные среды		Физические серверы	
Статус	Всего: 2	Статус	Всего: 1
Запущено	2	Онлайн	1
Выключено	0	Оффлайн	0
Технология	Всего: 2	Требуют внимания	1
Контейнеры	2	Продукты виртуализации	Всего: 1
Виртуальные машины	0	R-Виртуализация 7	1

Скриншот 25.

Откроем в браузере веб-панель управления, пропишем IP-адрес, который задавался на этапе установки Росплатформы. Будем использовать учетную запись с паролем, который также был задан на этапе установки.

ХРАНИЛИЩЕ 7

Nodes

UNASSIGNED 1

192.168.12.4

ADD NODE

NODES

AUDIT LOG

SETTINGS

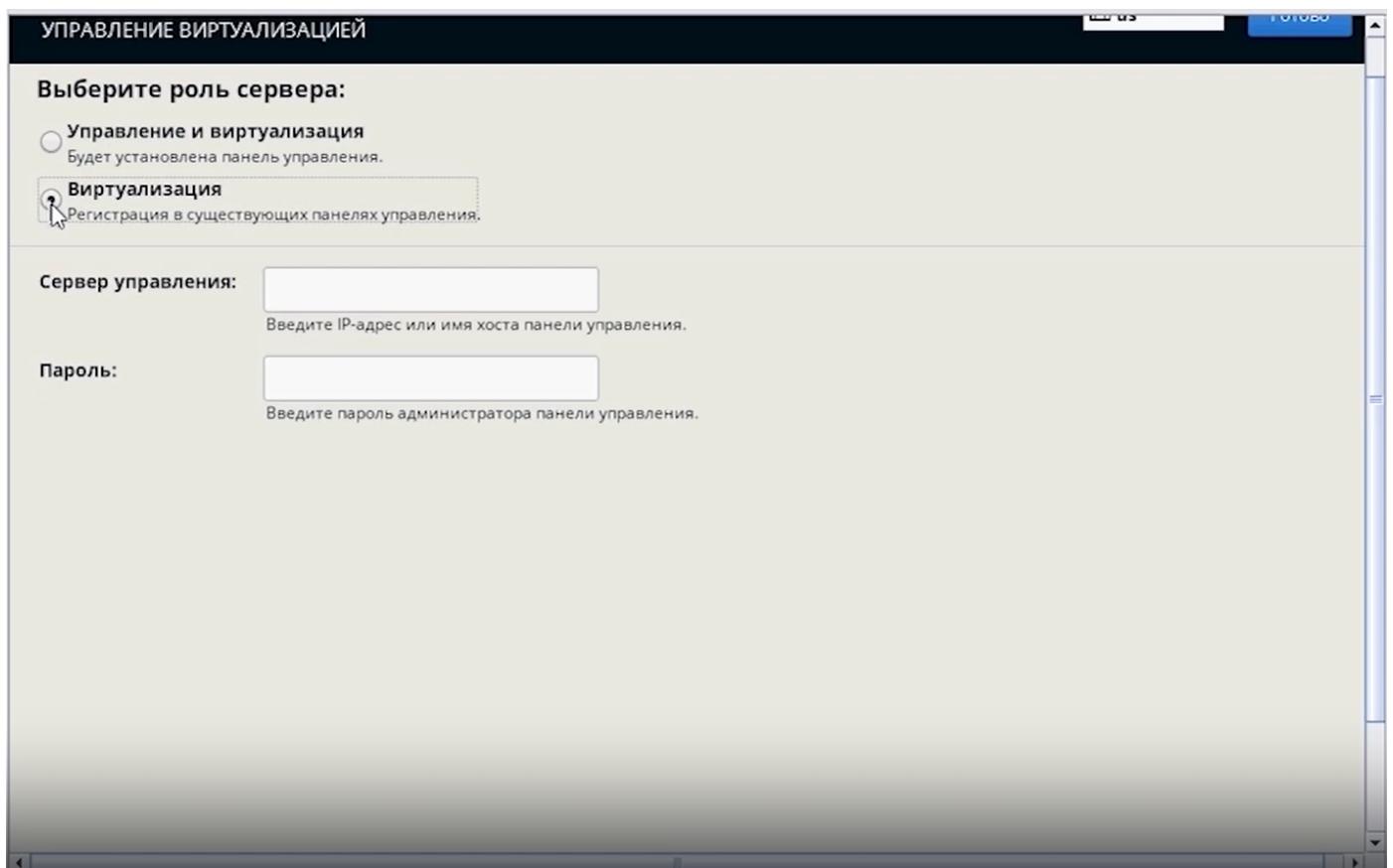
About

Скриншот 26.

Проверим также веб-интерфейс **P-хранилище**. Используя учетную запись root и пароль, заданный при установке, можно войти в интерфейс **P-хранилища**.

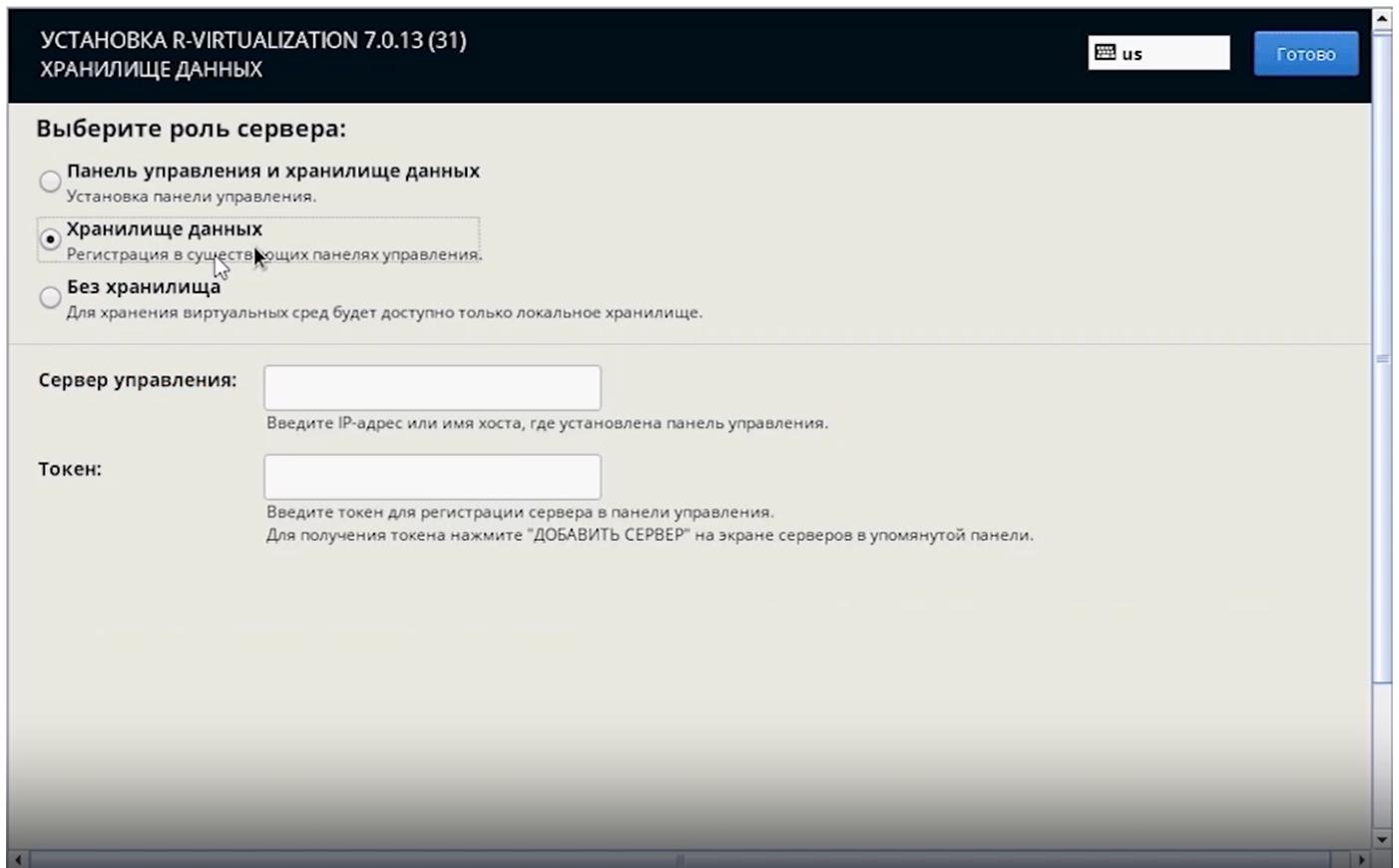
3.5 загрузка установщика последующих серверов (без контейнеров)

После завершения проверки можно переходить к установке Росплатформы на остальные сервера. Установка Росплатформы на оставшиеся сервера выполняется аналогичным способом за исключением одного: во вкладке **P-управление** необходимо выбрать второй из двух пунктов "Виртуализация - регистрация в существующих панелях управления".

*Скриншот 27.*

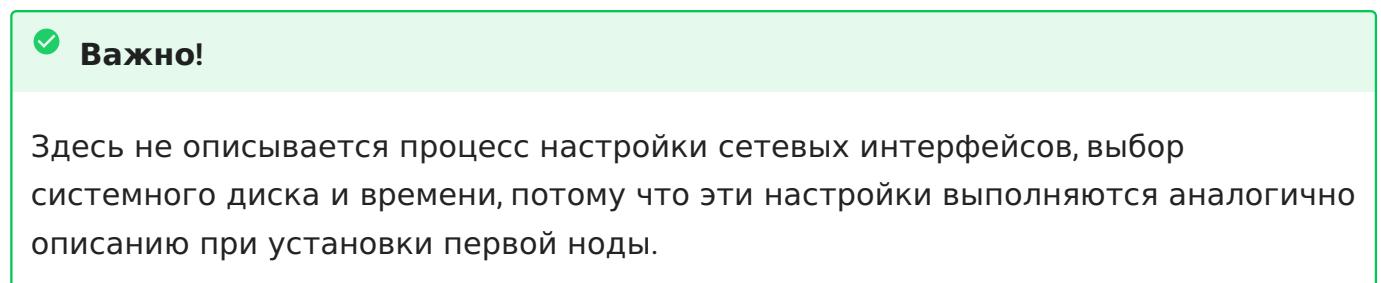
 **Важно!**

Здесь сразу можно выполнить регистрацию, однако рекомендуется выполнять регистрацию вручную после установки для исключения путаницы в сетевой инфраструктуре, поэтому оставим эти графы пустыми.



Скриншот 28.

Теперь перейдем во вкладку **P-хранилище**, выберем второй из трех имеющихся вариантов "Хранилище данных - регистрация существующих панелях управления" и также оставим имеющиеся поля ниже пустыми, нажмем "готово", нажмем "далее", и можно начинать установку.



The screenshot shows the R-Harbor management interface. On the left, there's a sidebar with icons for Nodes, Audit Log, and Settings. The main area is titled "Nodes > 192.168.12.4 > Network". It has tabs for "DISKS" and "NETWORK", with "NETWORK" selected. A table lists network interfaces:

Name	Status	IP addresses	Speed	Roles
sb1	OK	192.168.22.4/24	10 Gb / 10 Gb	Storage, SSH
mb0.120	OK		N/A	SSH
mb0	OK	fe80::e207:1bff:fe5:b...	3 Gb / 3 Gb	SSH
br-mb0.120	OK	192.168.12.4/24	N/A	Management, Web CP...

On the right, there are three buttons: "Details", "Performance", and "Choose role". The URL in the browser bar is https://192.168.12.8:8888/#m=Nodes&nd=da477dcb-8bd9-47d0-a70b-75000b828547&s=network.

Скриншот 29.

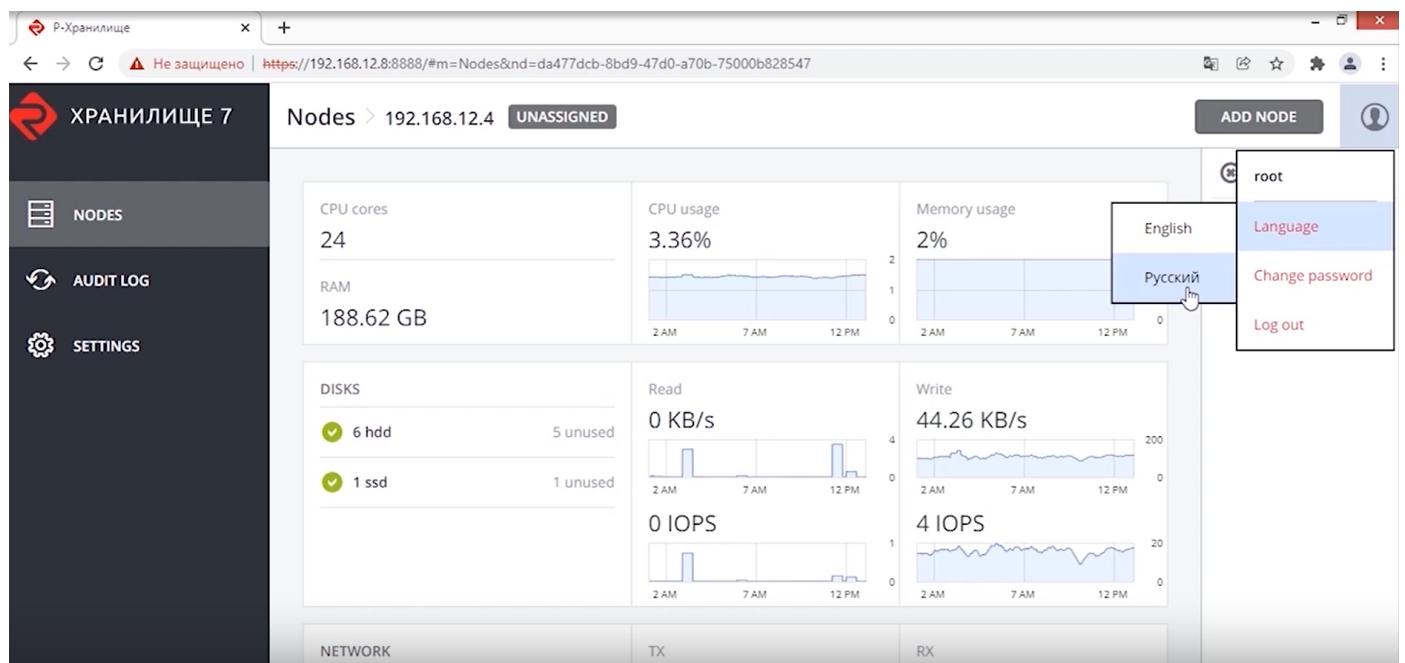
3.6 подключение к контейнеру управления Р-хранилища для настройки кластера

После того как установка Росплатформы выполнена на всех узлах, можно перейти в панель **Р-хранилища**. Для этого введем в адресной строке браузера ее IP-адрес, который указывали на этапе установки. Для входа используется учетная запись root и пароль, который задавался на этапе установки.

3.7 настройка сетевых ролей сервера

Тут пока видим одну единственную ноду, как раз ту, на которой находится контейнер управления и хранилища. Первым делом надо настроить сетевые роли ноды. Зайдем в ноду, далее в настройки её сетевых интерфейсов - **Network**. Выберем сеть Р-управления и назначим роли для сети Р-управления.

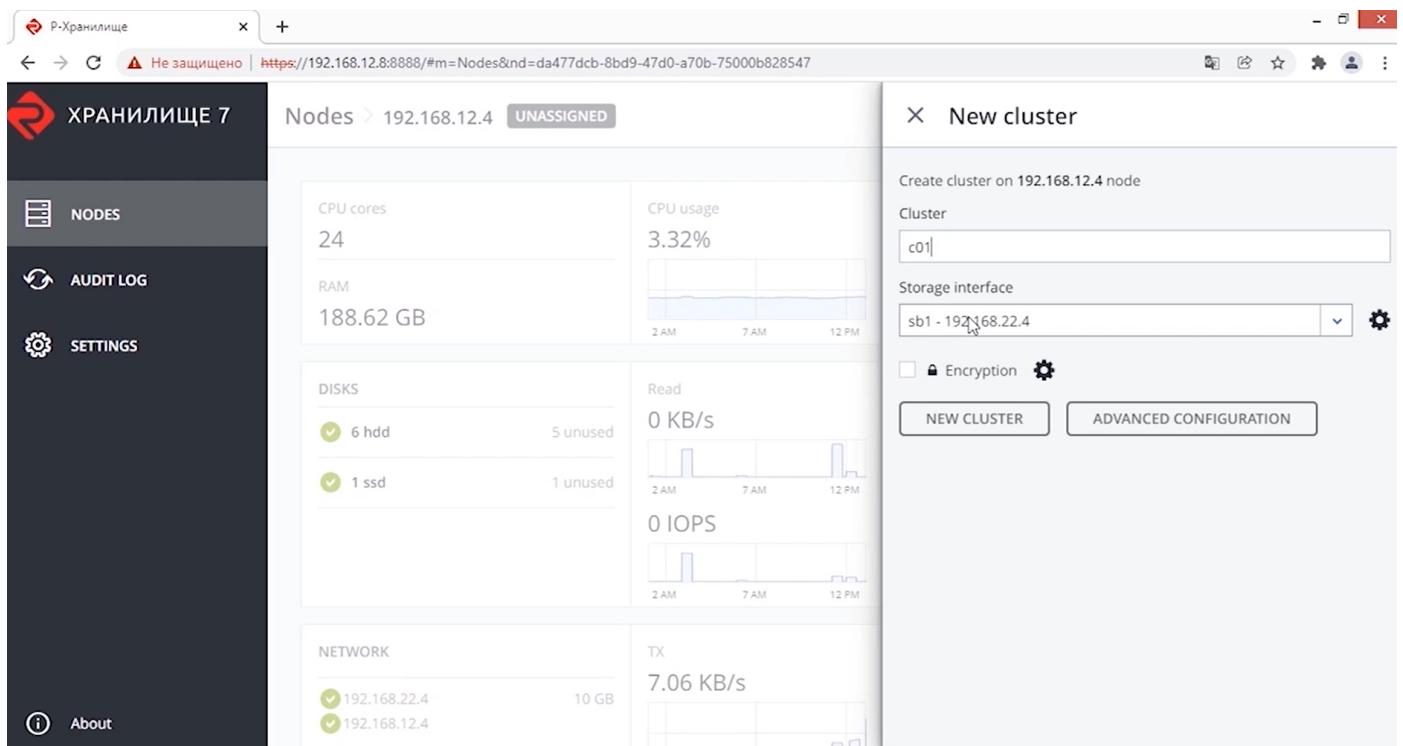
Следует выбрать роли управления Панель управления и **SSH**, лишнее уберем. Применим и перейдем к настройке ролей для сети **Р-хранилища**. Для сети **Р-хранилища** оставим **SSH** и добавим роль непосредственно хранилища "**Storage**", применим заданные параметры.



Скриншот 30.

3.8 создание кластера

Далее после завершения настройки сетевых ролей можно создать кластер. Перейдем для этого еще раз в ноду, и здесь выберем пункт "создать кластер". Язык интерфейса можно сменить на русский.

*Скриншот 31.*

В первой графе можно задать название вашего кластера, в нашем случае **c01**. Сразу автоматически подхватывается сеть **R-хранилища**, если предварительно правильно настроили сетевые роли, и далее перейдем в дополнительные настройки.

3.9 настройка дисковых ролей

В расширенных настройках “**advanced configuration**” можно изменить роли дисков, которые были им присвоены системой по умолчанию.

Дело в том, что можно по-разному организовывать дисковое пространство, объединяя их в разные уровни, которые могут использоваться для хранения разных типов данных, а также для разграничения более производительных дисков от более медленных.

В нашей конфигурации будет использоваться один **SSD**-диск для хранения на нем кэша или, другими словами, согласно нашей архитектуре журналы записи для жестких дисков.

Важно!

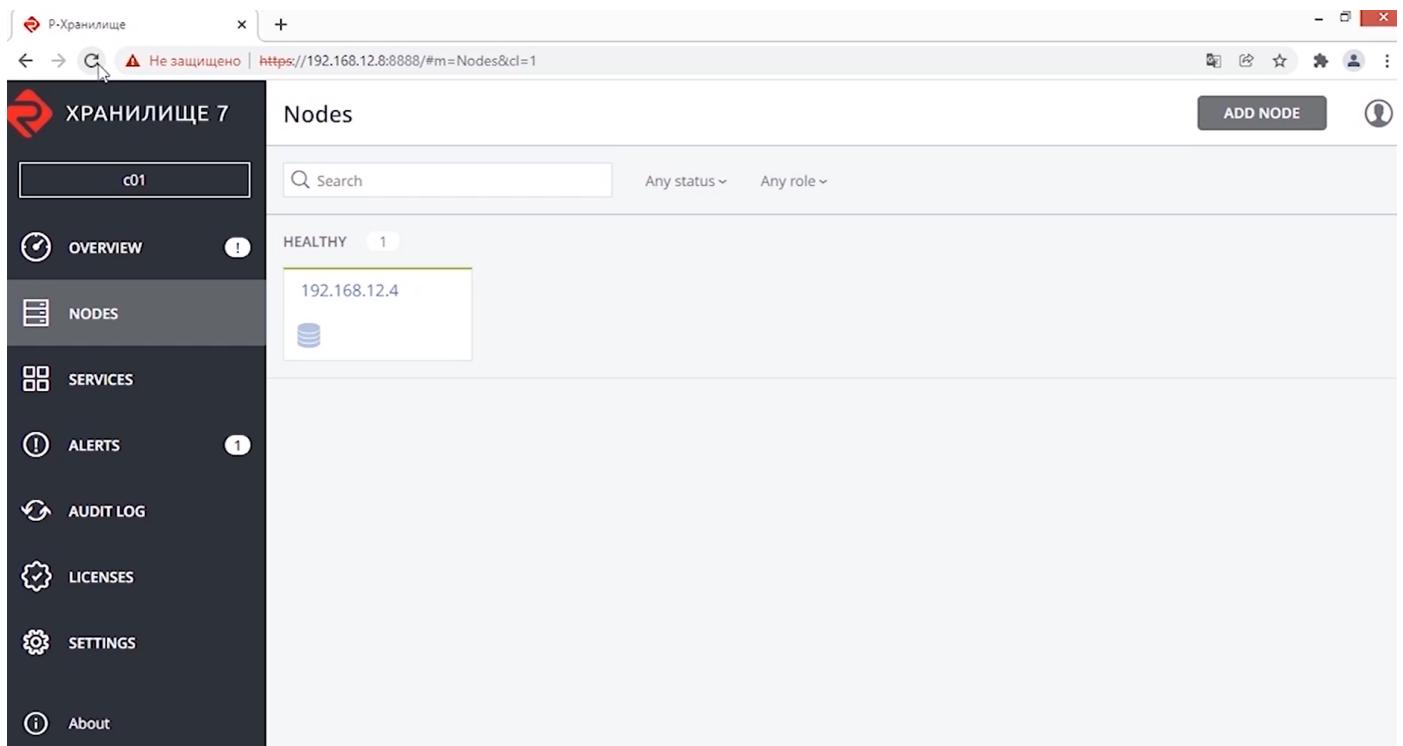
Это рекомендуется для обеспечения более высокой производительности. Как правило, один **SSD**-диск может обеспечить хранение кэша(журналы записи) для 3-4 жестких дисков. Однако это зависит от технических характеристик дисков, и этого **SSD**-диска хватит только на два жестких диска. Чтобы это работало необходимо еще объединить в один уровень. Оставшиеся жесткие диски в другой медленный уровень, так как кэш этих жестких дисков будет храниться на них же, и производительность этого уровня будет несколько ниже.

Disk	Status	Type	Role	Space used	Model	Serial
sda	OK	hdd	System	0.16 of 279.4 GB	EH0300JDXBA	5.00
sdc	OK	ssd	Metadata+Cache	301.82 of 372.61 GB	MO0400JFFCF	5.00
sdb	OK	hdd	Storage Tier 0	0 of 279.4 GB	EH0300JDXBA	5.00
sdf	OK	hdd	Storage Tier 0	0 of 279.4 GB	EH0300FBQDD	5.00
sdg	OK	hdd	Storage Tier 0	0 of 279.4 GB	EH0300FBQDD	5.00
sde	OK	hdd	Storage Tier 0	0 of 279.4 GB	EH0300FBQDD	5.00
sdd	OK	hdd	Storage Tier 0	0 of 279.4 GB	EH0300FBQDD	5.00

Скриншот 32.

Роли изменены в соответствии с вышесказанный рекомендацией, создадим кластер с новыми параметрами.

После создания кластера увидим, что в левой части **web**-интерфейса Р-хранилище появились дополнительные пункты с дополнительными настройками.

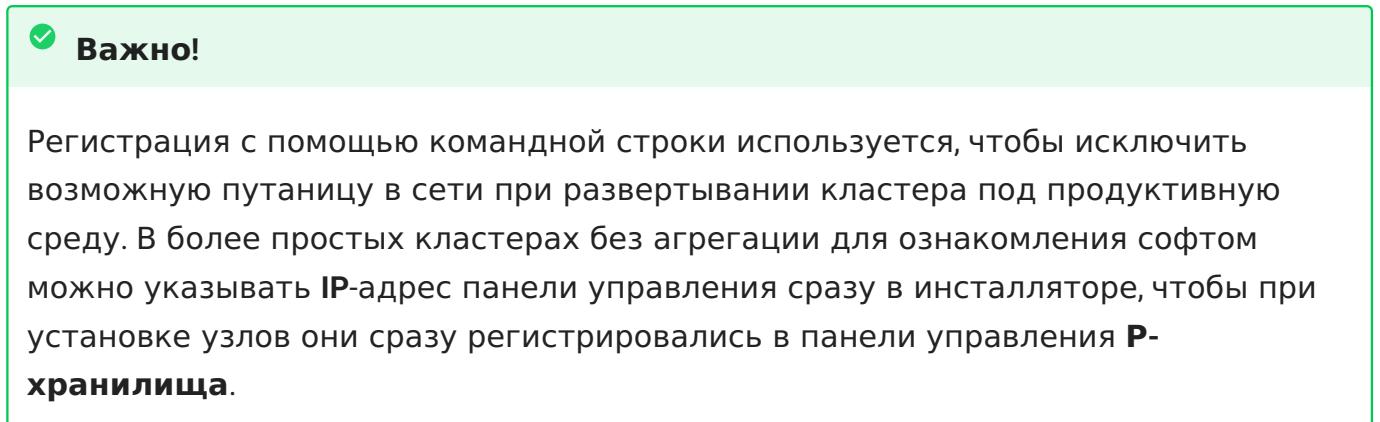


Скриншот 33.

Теперь необходимо проверить, правильно ли применились роли на дисках. Перейдем в ноду, откроем диски и убедимся, что все корректно, все так, как и задавалось на этапе создания кластера.

3.10 регистрация серверов в панели Р-Хранилища

После создания кластера надо зарегистрировать остальные ноды в панели **Р-Хранилища**. Для регистрации остальных нод в панели управления R-хранилища надо зайти на них по **SSH**, используя их **IP-адрес сети Р-управления**, который задавался на этапе установки, и выполнить определенную команду, показанную под скриншотом 34.



The screenshot shows the R-Harbor management interface. On the left sidebar, under the 'NODES' section, there is a list of nodes: 'c01' (OVERVIEW), '192.168.12.8' (HEALTHY). The main content area is titled 'Add node' and has a sub-section 'Add storage nodes to R-Хранилище'. It contains instructions and a token generation interface. A large circular icon of a server is visible on the right.

Add storage nodes to R-Хранилище

On this screen, you can get instructions on how to add storage nodes to your R-Хранилище infrastructure. To add storage nodes to R-Хранилище, do the following:

- ① On this screen, generate a token required to add storage nodes.

5b451b92

Valid till Jan 12, 2022, 11:09 am Generate new token

The token can be used to add multiple storage nodes. Generating a new token invalidates the old one, so you will need to provide the new token when adding nodes.

- ② Boot the node from the ISO image.
- ③ Choose the storage option on the Р-ХРАНИЛИЩЕ screen.
- ④ Specify the IP address or hostname of the management panel in the "Management panel" field.
- ⑤ Specify the token shown on this screen in the "Token" field.
- ⑥ Complete other installation steps if required and finish the installation.

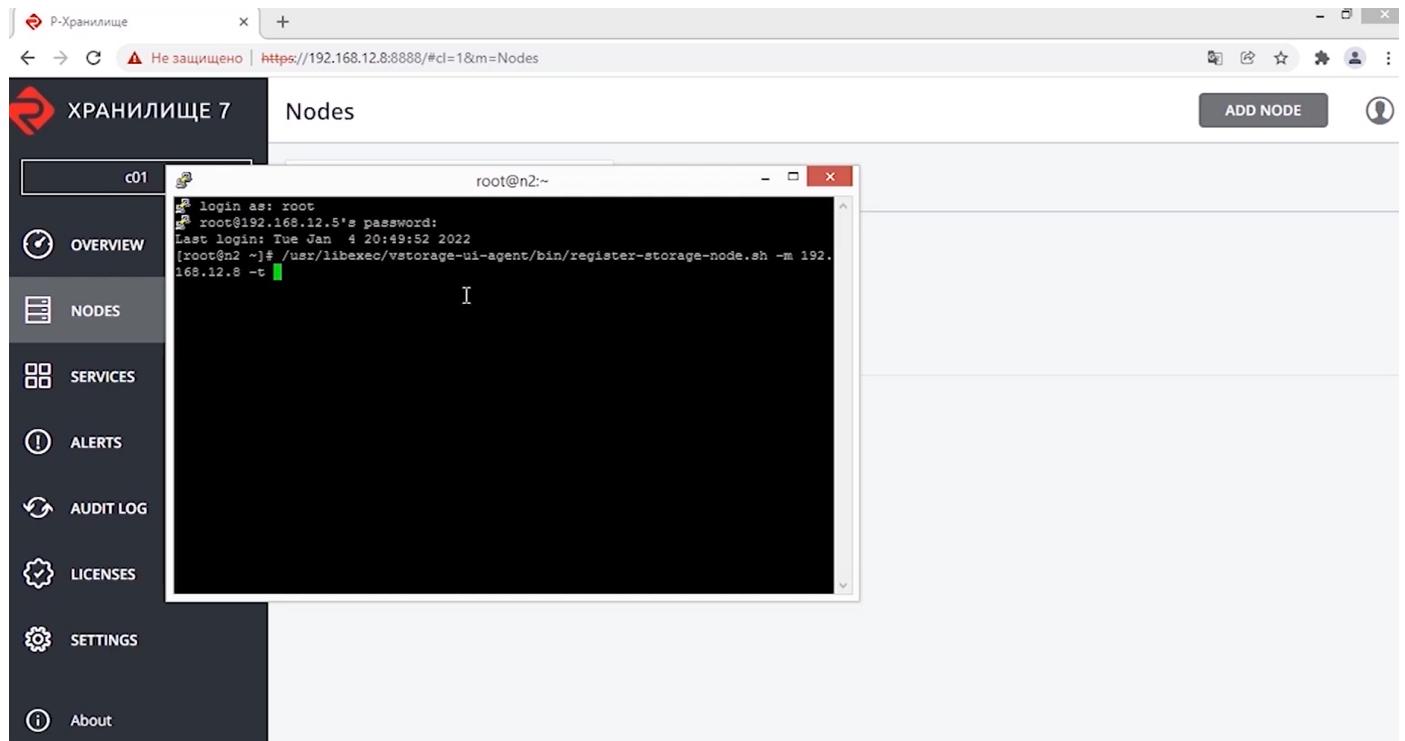
Скриншот 34.

Команда, которую надо выполнить:

```
#/usr/libexec/vstorage-ui-agent/bin/register-storage-node.sh -m
192.168.12.8 -t 5b451b92
```

после ключа **-m** надо указать IP-адрес панели управления Р-хранилище. В нашем случае это **192.168.12.8**, тот самый адрес, который с портом **8888**, и после ключа **-t** надо указать токен.

Токен как на скриншоте 34 можно получить непосредственно в панели управления, нажав кнопку "добавить ноду" или "add node" смотря какая языковая раскладка, его копируем, вставляем в команду и выполняем ее. Пример выполнения на скриншоте 35-36.



Скриншот 35.

После выполнения команды нода появится в панели управления **Р-хранилище**, но сначала на ней отображается символ троеточия, надо дождаться, пока этот символ сменится на вопросительный знак, для ускорения этого процесса можно обновить страницу.

The screenshot shows the R-Hraniлище 7 web interface. On the left, there's a sidebar with various navigation links: Overview, Nodes (selected), Services, Alerts, Audit Log, Licenses, Settings, and About. The main area is titled 'Nodes' and shows two sections: 'UNASSIGNED' and 'HEALTHY'. In the 'UNASSIGNED' section, there is one entry for IP 192.168.12.5. In the 'HEALTHY' section, there is one entry for IP 192.168.12.4. A terminal window is overlaid on the interface, showing the command output for registering a storage node:

```

root@n2:~#
root@n2:~# login as: root
root@n2:~# root@192.168.12.5's password:
Last login: Tue Jan  4 20:49:52 2022
[root@n2 ~]# /usr/libexec/vstorage-ui-agent/bin/register-storage-node.sh -m 192.168.12.8 -t 5b451b92
Configuration saved in /etc/firewalld/zones/pub_br-mb0_120.xml
Remove old configuration /etc/firewalld/zones/--.xml
success
Configuration saved in /etc/firewalld/zones/pub_virbr0.xml
Remove old configuration /etc/firewalld/zones/--.xml
success
Configuration saved in /etc/firewalld/zones/pub_sb1.xml
Remove old configuration /etc/firewalld/zones/--.xml
success
Configuration saved in /etc/firewalld/zones/pub_mb0.xml
Remove old configuration /etc/firewalld/zones/--.xml
success
Configuration saved in /etc/firewalld/zones/pub_mb0_120.xml
Remove old configuration /etc/firewalld/zones/--.xml
success
add network 192.168.12.0/24 in trusted list
Configuration saved in /etc/firewalld/zones/mix_br-mb0_120.xml
Remove old configuration /etc/firewalld/zones/pub_br-mb0_120.xml

```

Скриншот 36.

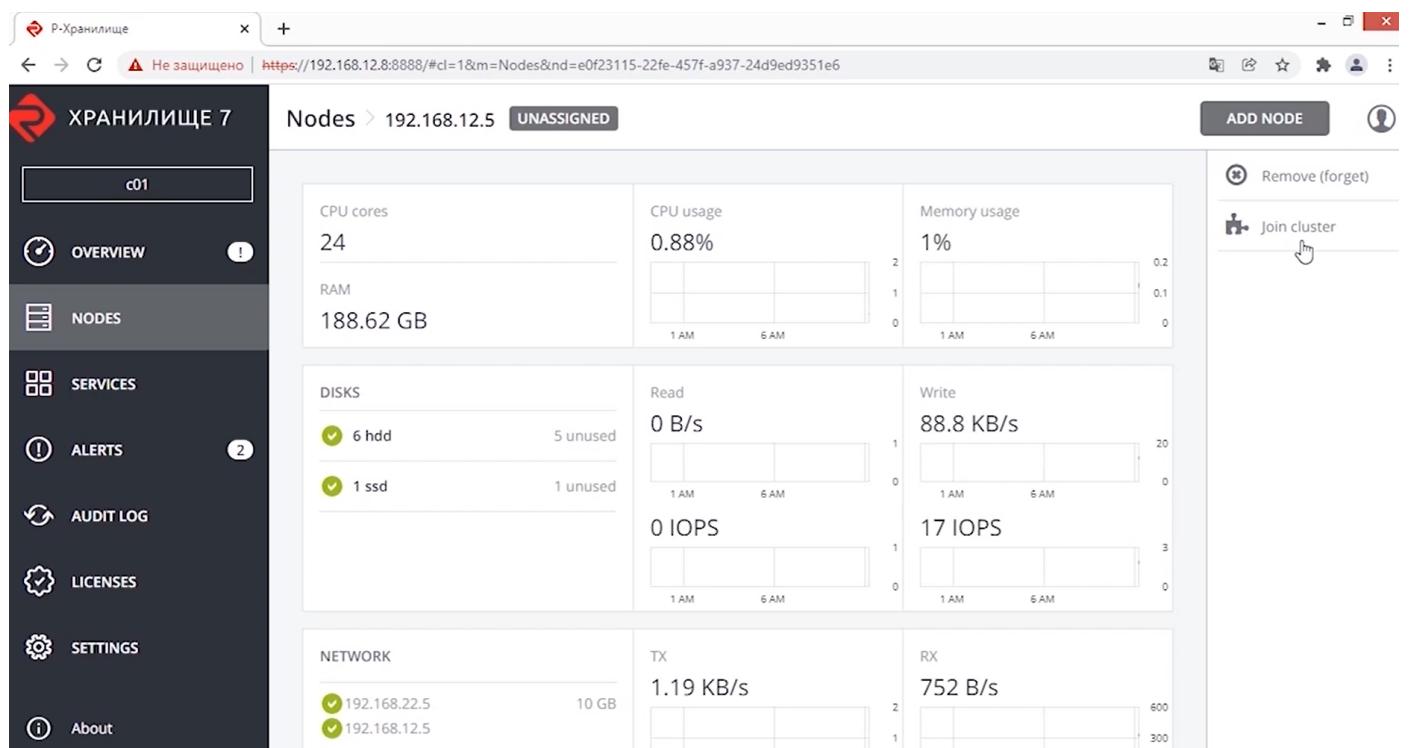
Добавим третью ноду по такому же принципу. Скопируем уже готовую команду, которую вводили на предыдущей ноде.

The screenshot shows the R-Hraniлище 7 web interface. The sidebar and main layout are identical to the previous screenshot. The 'Nodes' section now shows two entries in the 'UNASSIGNED' section: 192.168.12.5 and 192.168.12.6. The 'HEALTHY' section still contains the entry for IP 192.168.12.4.

Скриншот 37.

3.11 настройка сетевых ролей зарегистрированных серверов

Следующий наш шаг – это настройка сетевых ролей нод, которую надо выполнить также, как и на первой ноде.



Скриншот 38.

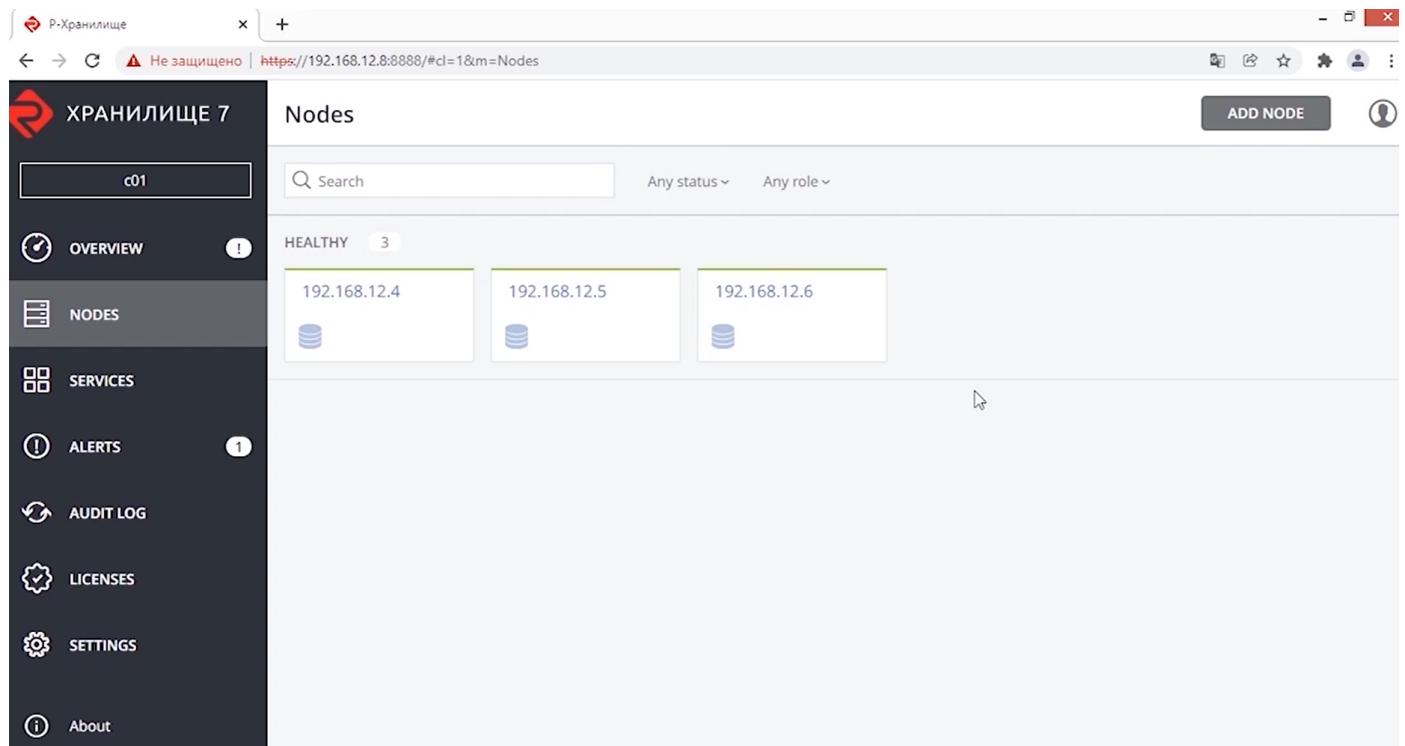
3.12

3.13 добавление зарегистрированных серверов в кластер

После завершения настройки сетевых ролей новых нод добавим их в кластер, настроив дисковые роли аналогично первой ноде. Для этого перейдем в ноду и нажмём «Join Cluster».

Автоматически подхватится нужный интерфейс, который используется для сети хранилища, перейдем в дополнительные настройки, зададим их так же, как и на первой ноде, и дождемся присоединения второй ноды к нашему кластеру. Обновим страницу при необходимости.

После добавления второй ноды в кластер проверим дисковые роли. Теперь можно добавить третью ноду в кластер аналогичным способом.



Скриншот 39.

3.14

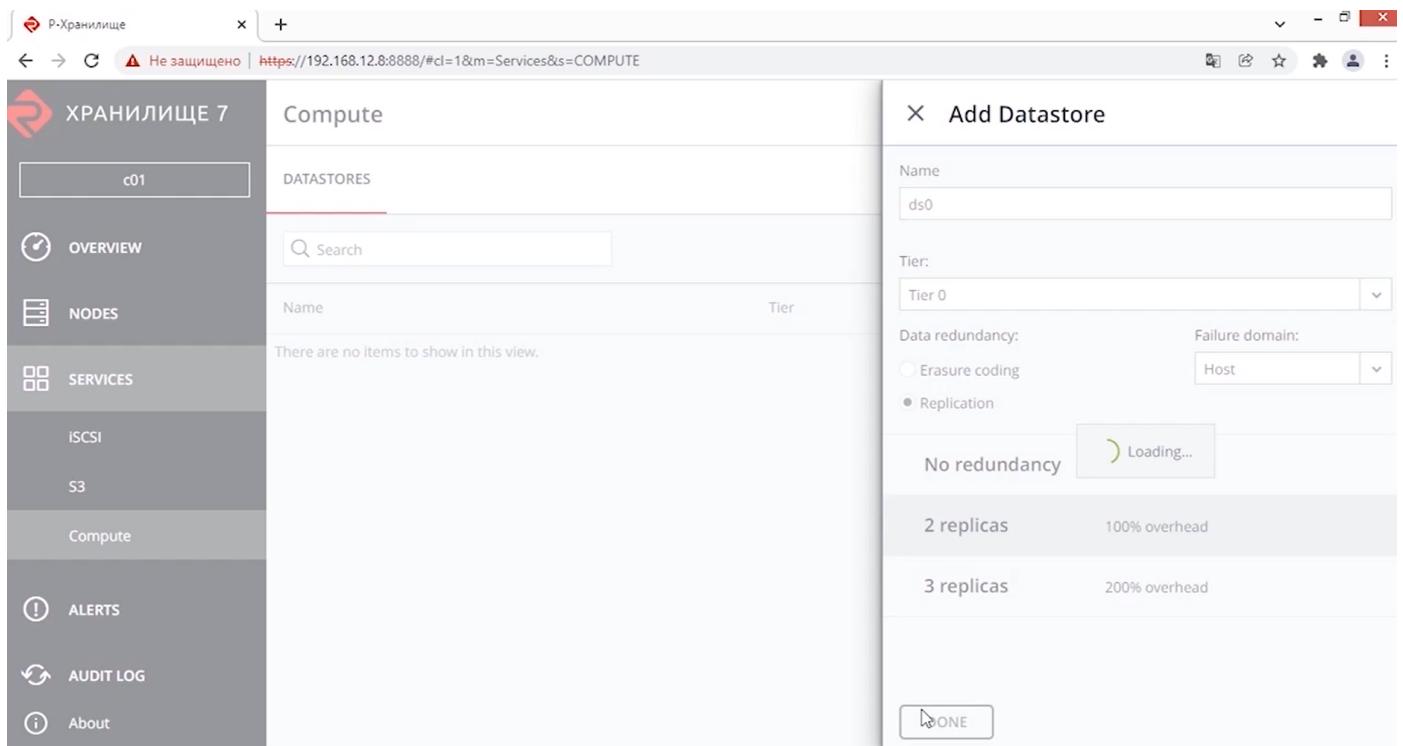
3.15 создание кластерного датастора для виртуальных сред

После того, как добавлены все ноды в кластер как на скриншоте 39, можно приступить к созданию датастора.

The screenshot shows the R-Хранилище 7 web interface. The left sidebar has tabs for Overview, Nodes, Services, iSCSI, S3, Compute, Alerts, Audit Log, and About. The Compute tab is selected. The main area is titled 'Compute' and contains a 'DATASTORES' section. A search bar is at the top of this section. Below it is a table with columns for Name, Tier, and Data redundancy. A message says 'There are no items to show in this view.' To the right of the table is a vertical toolbar with 'ADD DATASTORE' (highlighted with a red box), 'Configure', and 'Delete' buttons. The URL in the browser is https://192.168.12.8:8888/#cl=1&m=Services&s=COMPUTE.

Скриншот 40.

Нажмем на вкладку сервисы, выберем подпункт “Compute”, и в открывшемся окне нажмем на кнопку “add datastore”.



Скриншот 41.

В поле “**Name**” зададим имя нашего датастора, в нашем случае будет **ds0**. В графе “**Tier**” выберем уровень, на котором будет создан этот датастор. Здесь согласно ранее заданным настройкам два уровня: “**Tier 0**” и “**Tier 1**”.

Этот датастор будет создан для уровня “**Tier 0**”. Область отказа остается по умолчанию с параметром хост. При создании датастора надо выбрать один из двух типов избыточности: кодирование или репликация.

Важно!

Кодирование, представляет собой **EC** по типу **RAID 6** и больше подходит для неструктурированных данных таких как: медиа данные, видеопотоки (видеонаблюдение), для резервного копирования или для виртуальных сред с низким вводом выводом не желе чем Репликация.

Репликация используется для высоконагруженных виртуальных сред, виртуальных машин, контейнеров, а вид репликации зависит от количества узлов в кластере.

В пяти узловом кластере можно использовать **реплику 3**, в кластерах из четырех и трех узлов подходит **реплика 2**.

В нашем случае будет выбрана репликация в качестве избыточности, и так как этот кластер состоит из трех узлов, будет выбрана реплика 2. Создадим датастор с заданными параметрами, и создадим сразу еще один аналогичный, но для уровня 1.

The screenshot shows the R-Storage interface for managing storage clusters. The left sidebar has navigation links: Overview, Nodes, Services, iSCSI, S3, Compute, Alerts, Audit Log, and About. The main area is titled 'Compute' and shows 'ХРАНИЛИЩЕ 7'. It lists two datastores: 'ds0' and 'ds1'. A search bar is present above the table. The table columns are Name, Tier, Data redundancy, and an edit icon. The data is as follows:

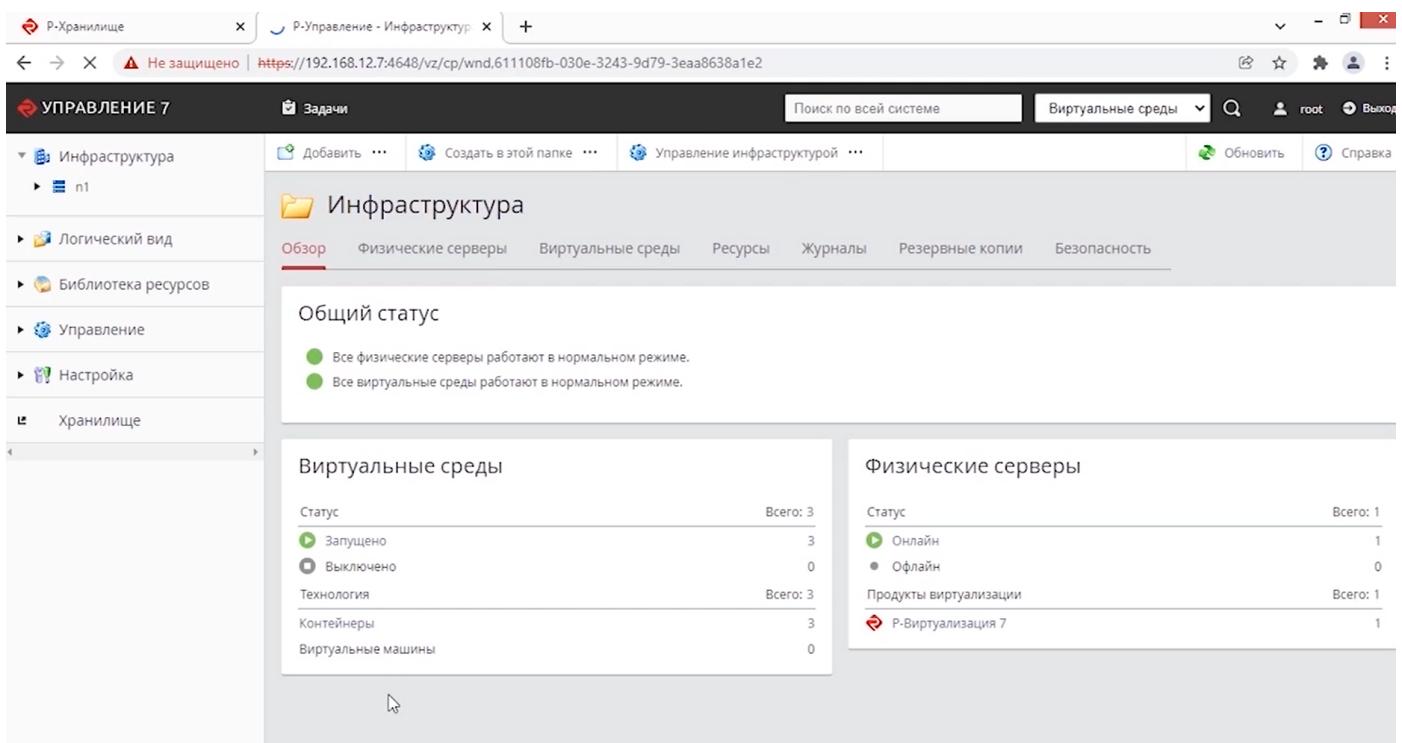
Name	Tier	Data redundancy
ds0	0	2 replicas
ds1	1	2 replicas

Скриншот 42.

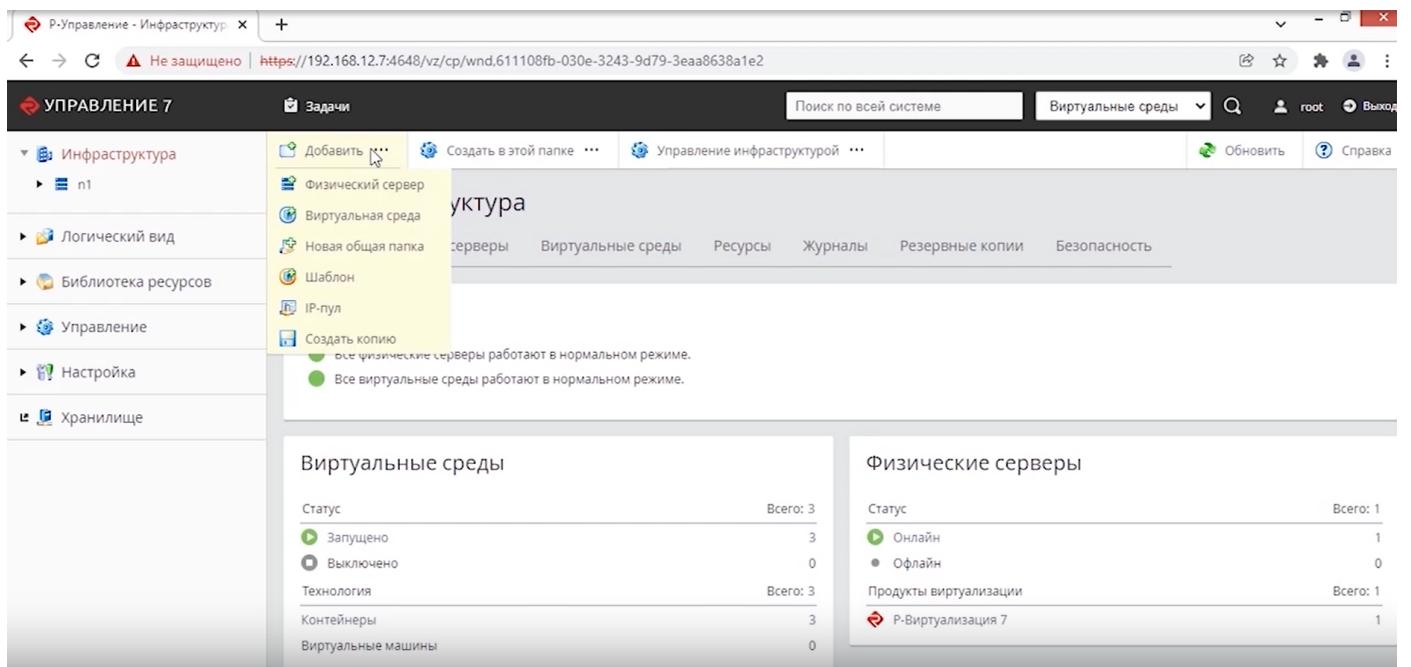
В столбце **Tier** видно на скриншоте 42, что датасторы созданы для разных уровней: для уровня 0 и для уровня 1, и видно какой тип избыточности выбран для каждого датастора.

3.16 начало работы с панелью Р-Виртуализации

На этом этапе настройки в панели управления **Р-хранилища** выполнены, теперь перейдем в панель управления виртуализацией как на следующем скриншоте 43. Для этого в адресной строке браузера введем ее **IP**-адрес, который указывали на этапе установки. Порт можно не указывать, он подхватится автоматически. Для входа используется также пользователь **root** с паролем, который задавался на этапе установки.



Скриншот 43.

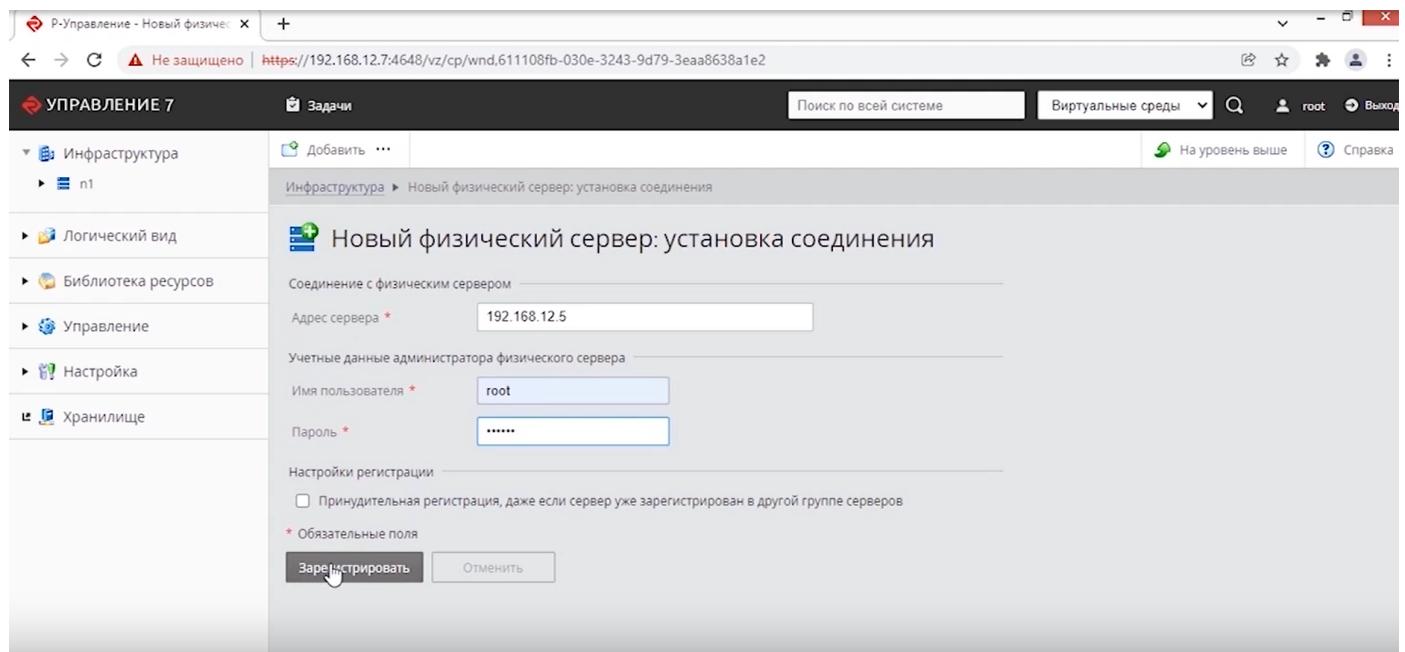


Скриншот 44.

В панели управления **Р-виртуализации** видно, что отображается только одна из трех наших нод, это видно в верхней части **web**-интерфейса во вкладке инфраструктура.

3.17 регистрация серверов в панели Р-Виртуализации

Зарегистрируем оставшиеся две ноды. Для этого в верхней части **web**-интерфейса нажмем кнопку "**добавить**" и выберем пункт "**физический сервер**".



Скриншот 45.

В поле "**адрес сервера**" указывается IP-адрес сети управления добавляемой ноды, который присваивался на этапе установки. В нашей конфигурации это 12.5. И укажем пользователя root и пароль.

Внутри вкладки «**инфраструктура**» появится вторая нода. Повторим еще раз эту процедуру для регистрации третьей ноды.

The screenshot shows the R-Management interface with the following details:

- Left sidebar:** УПРАВЛЕНИЕ 7 (Management 7) with sections: Инфраструктура (Infrastructure), Логический вид (Logical View), Библиотека ресурсов (Resource Library), Управление (Management), Настройка (Configuration), and Хранилище (Storage).
- Top bar:** Не защищено (Not protected) | https://192.168.12.7:4648/vz/cp/wnd.611108fb-030e-3243-9d79-3eaa8638a1e2 | Поиск по всей системе (Search system wide) | Виртуальные среды (Virtual environments) | Выход (Logout).
- Main area:**
 - Section title:** Инфраструктура (Infrastructure).
 - Sub-sections:** Обзор (Overview), Физические серверы (Physical servers) (selected), Виртуальные среды (Virtual environments), Ресурсы (Resources), Журналы (Logs), Резервные копии (Backups), Безопасность (Security).
 - Message:** Информация: Физический сервер успешно зарегистрирован. (Information: Physical server successfully registered.)
 - Filter:** Фильтр (Filter), Имя хоста (Host name), Продукт виртуализации (Virtualization product).
 - Search:** Поиск (Search), Сбросить результаты поиска (Clear search results), Изменить (Change).
 - Table:** Серверы: имеется 3 (Servers: 3 available).

<input type="checkbox"/>	Статус (Status)	Имя хоста (Host name)	IP-адрес по умолчанию (Default IP address)	Продукт виртуализации (Virtualization product)	ЦП (CPU)	Диск (Disk)	Память (Memory)
<input type="checkbox"/>	n1	192.168.12.4	R-Virtualization release 7.0.13 (31)	2.2%	5.5%	2.9%	
<input type="checkbox"/>	n2	192.168.12.5	R-Virtualization release 7.0.13 (31)				
<input type="checkbox"/>	n3	192.168.12.6	R-Virtualization release 7.0.13 (31)				
 - Buttons:** Новый физический сервер (New physical server), Переместить (Move), Отменить регистрацию (Cancel registration).
 - Page controls:** Показывать на странице: 10, 20 (selected), 40, 80, 160.

Скриншот 46.

3.18 установка лицензии для Р-Виртуализации на серверах

После регистрации всех нод кластера в панели управления виртуализации надо установить лицензию на нодах. Для этого перейдем во вкладку "**настройка**" и выберем пункт "**управление лицензиями**".

The screenshot shows the 'Licenses' section of the R-Management interface. On the left sidebar, under 'Инфраструктура', there are three nodes: n1, n2 (with a red warning icon), and n3. The main panel has a search bar and navigation buttons for 'Virtual environments', 'root', and 'Logout'. It displays a table with one row of data:

Номер ключа	Физический сервер	Статус	Дата окончания срока действия
RVZ.00000981.0001	n1	Активно	30 июня 2022, 03:00:00

Скриншот 47.

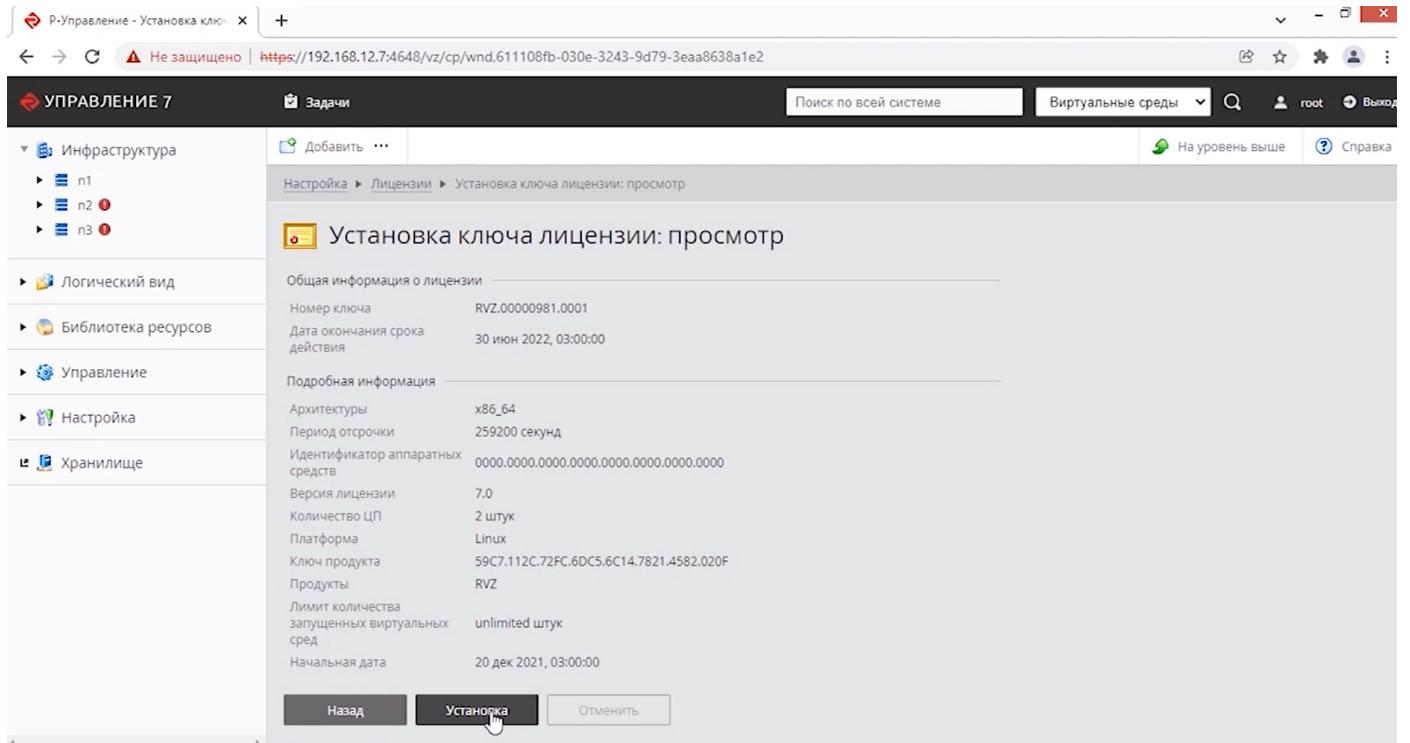
Лицензию для первого узла была установлена ранее, осталось установить для второго и третьего узлов. Для этого нажмем кнопку "**установить файл лицензии**".

The screenshot shows the 'Select physical servers' page in Google Chrome. The URL is https://192.168.12.7:4648/vz/cp/wnd.611108fb-030e-3243-9d79-3eaa8638a1e2. The main panel shows a table of hosts and a file upload dialog titled 'Лицензии: загрузка файла' (Licenses: file upload) with the message 'Положенное ниже поле:' (The field below is intended for).

Статус	Имя хоста	IP-адрес по умолчанию	Продукт виртуализации	ЦП	Диски
n1	192.168.12.4	R-Virtualization release 7.0.13 (31)	2.2%	5.5%	
n3	192.168.12.6	R-Virtualization release 7.0.13 (31)	0.9%	5.3%	
n2	192.168.12.5	R-Virtualization release 7.0.13 (31)	1%	5.3%	

Скриншот 48.

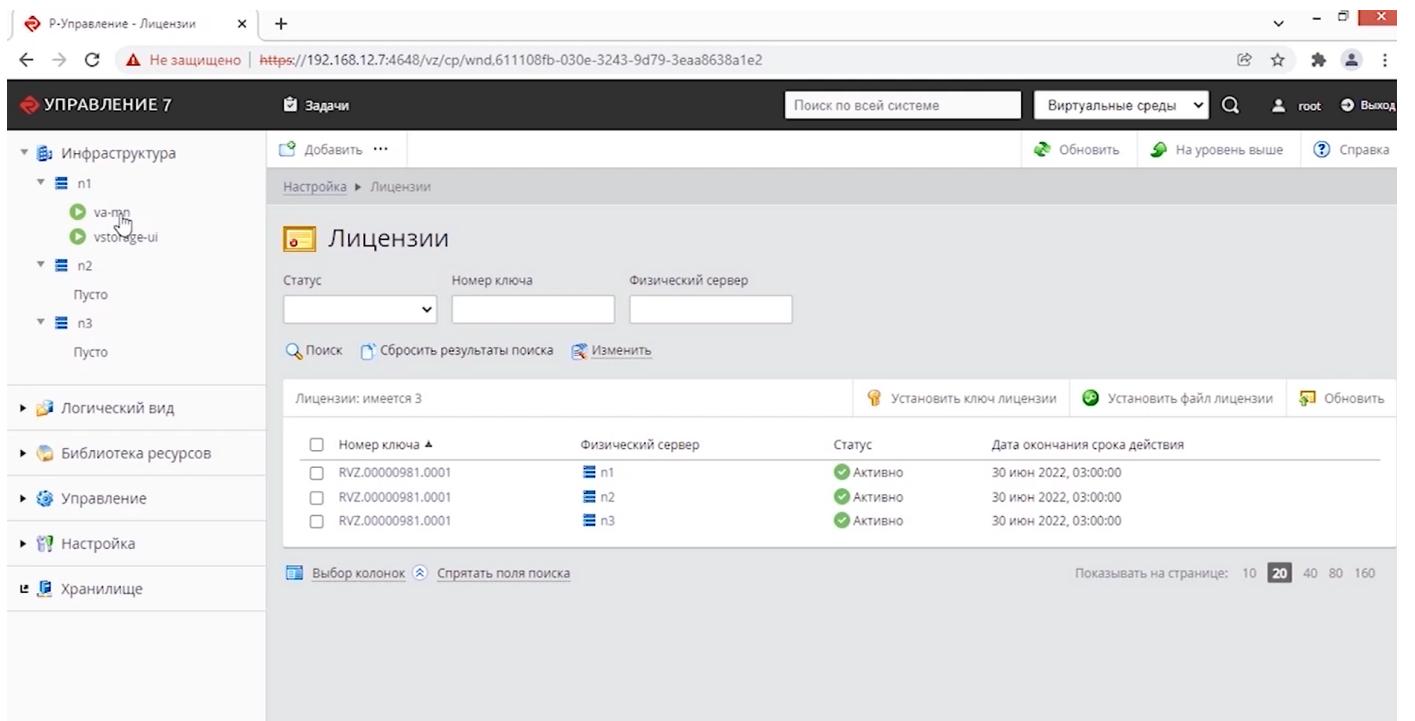
Далее нажмем на ссылку второй ноды n2.

*Скриншот 49.*

Выберем файл лицензии, затем нажмем "далее" и "установка".

Важно!

Обратите внимание, что после установки лицензии на ноду красный восклицательный знак рядом с названием ноды во вкладке инфраструктура пропадает.

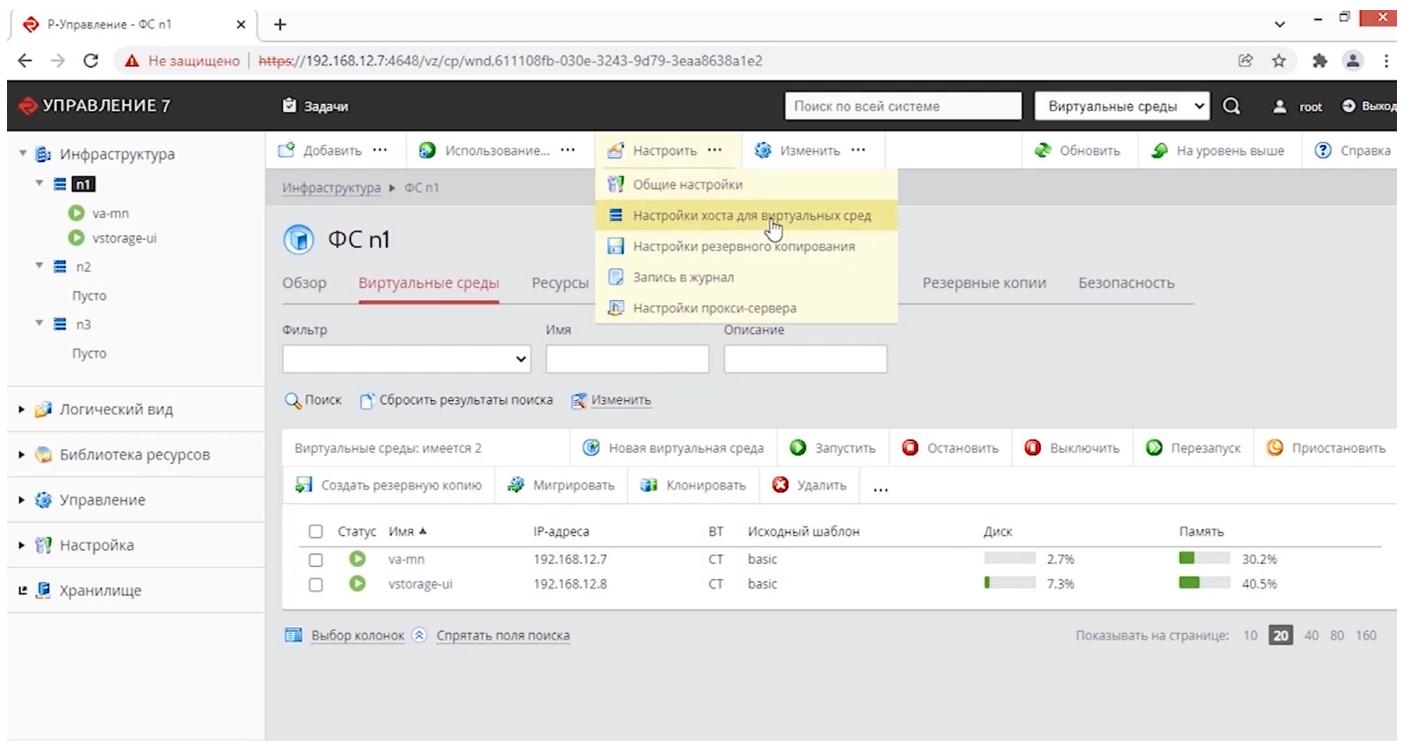


Скриншот 50.

Повторим эту процедуру для третьей ноды. Далее развернем содержимое дерева наших нод. Видим, что третья и вторая ноды пустые, а на первой два контейнера панели управления виртуализацией и хранилищем.

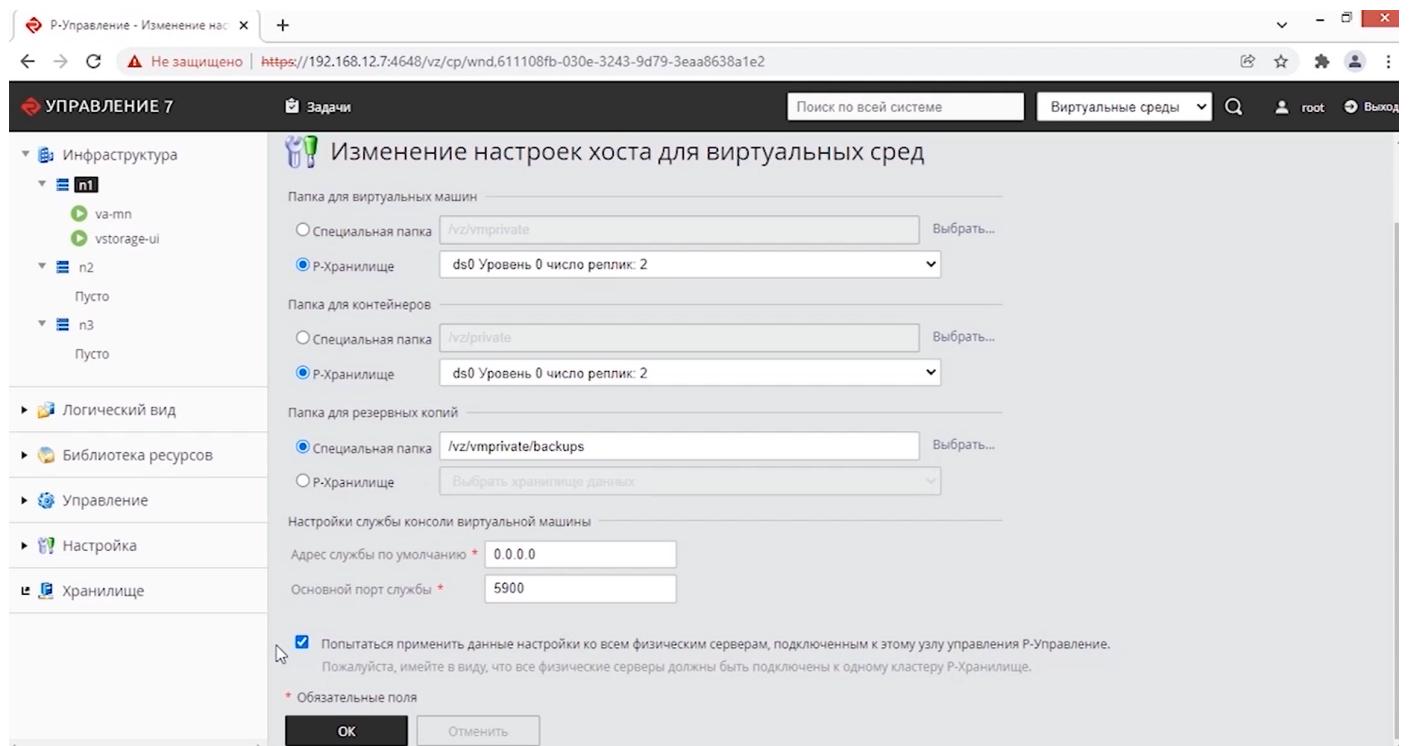
3.19 настройка хранения виртуальных сред в датасторе

Сейчас они хранятся локально на первой ноде и в единственном экземпляре, а для обеспечения избыточности они должны храниться в датасторе, который создавался ранее. Чтобы изменить эту настройку, выберем первую ноду, нажмем на кнопку **"настроить"** и выберем пункт **"настройки Хоста для виртуальных сред"** как на скриншоте 51.



Скриншот 51.

Изменим папки для виртуальных машин и контейнеров, выбрав пункт Р-хранилище, и далее выберем нужный нам датастор. Далее необходимо поставить галочку рядом с пунктом "**попытаться применить данные настройки ко всем физическим серверам, подключенным к этому узлу управления**". Это для того, чтобы заданные настройки распространились на все узлы кластера, и потом нажимаем ОК.



Скриншот 52.

Все заданные настройки у нас применяются ко всем узлам кластера, и теперь все виртуальные среды, то есть виртуальные машины и контейнеры, которые будут создаваться, уже будут создаваться непосредственно в датасторе.

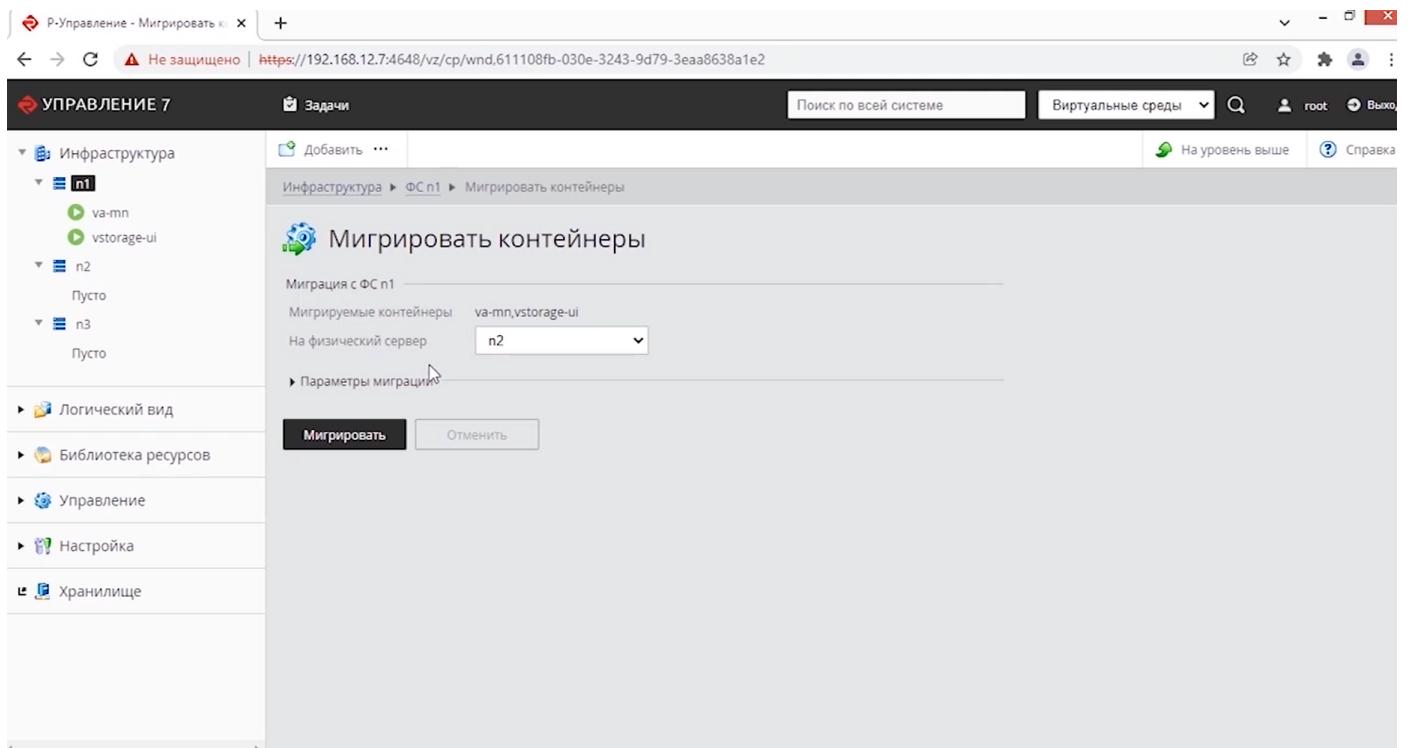
The screenshot shows the 'P-Управление - ФС n1' interface. The left sidebar lists nodes: n1 (selected), n2 (empty), and n3 (empty). The main area is titled 'ФС n1' and shows the 'Виртуальные среды' tab selected. It displays two containers: 'va-mn' and 'vstorage-ui'. The 'Мигрировать' button is highlighted. A table below shows container details:

Статус	Имя	IP-адреса	ВТ	Исходный шаблон	Диск	Память
<input checked="" type="checkbox"/>	va-mn	192.168.12.7	CT	basic	2.7%	30.7%
<input checked="" type="checkbox"/>	vstorage-ui	192.168.12.8	CT	basic	7.3%	40.5%

Скриншот 53.

3.20 перемещение контейнеров управления в датастор

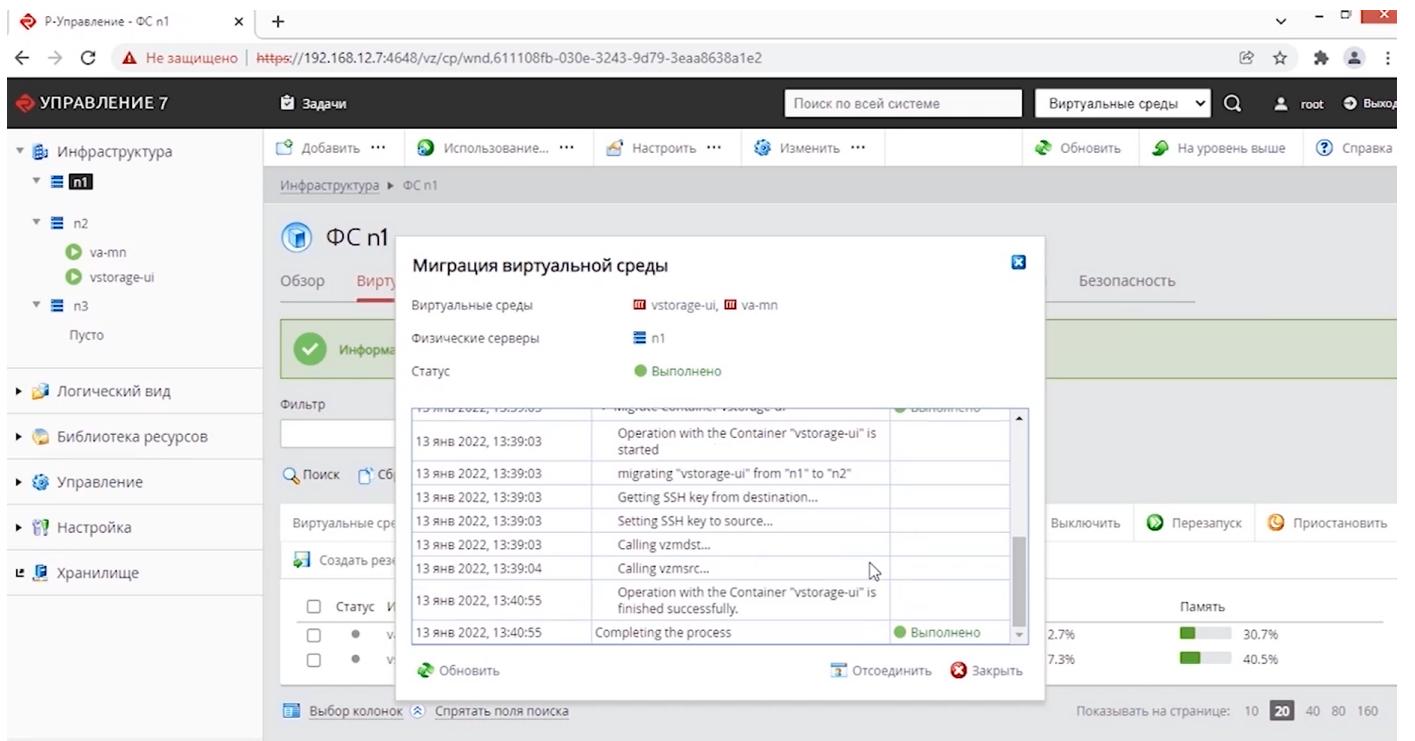
Теперь, чтобы два контейнера управления “**“va-mn”(Р-управление)** и “**“vstorage-ui”(Р-хранилище)** переехали с локального расположения в кластерную точку мониторинга, то есть в указанный датастор, надо их мигрировать на другую ноду.



Скриншот 54.

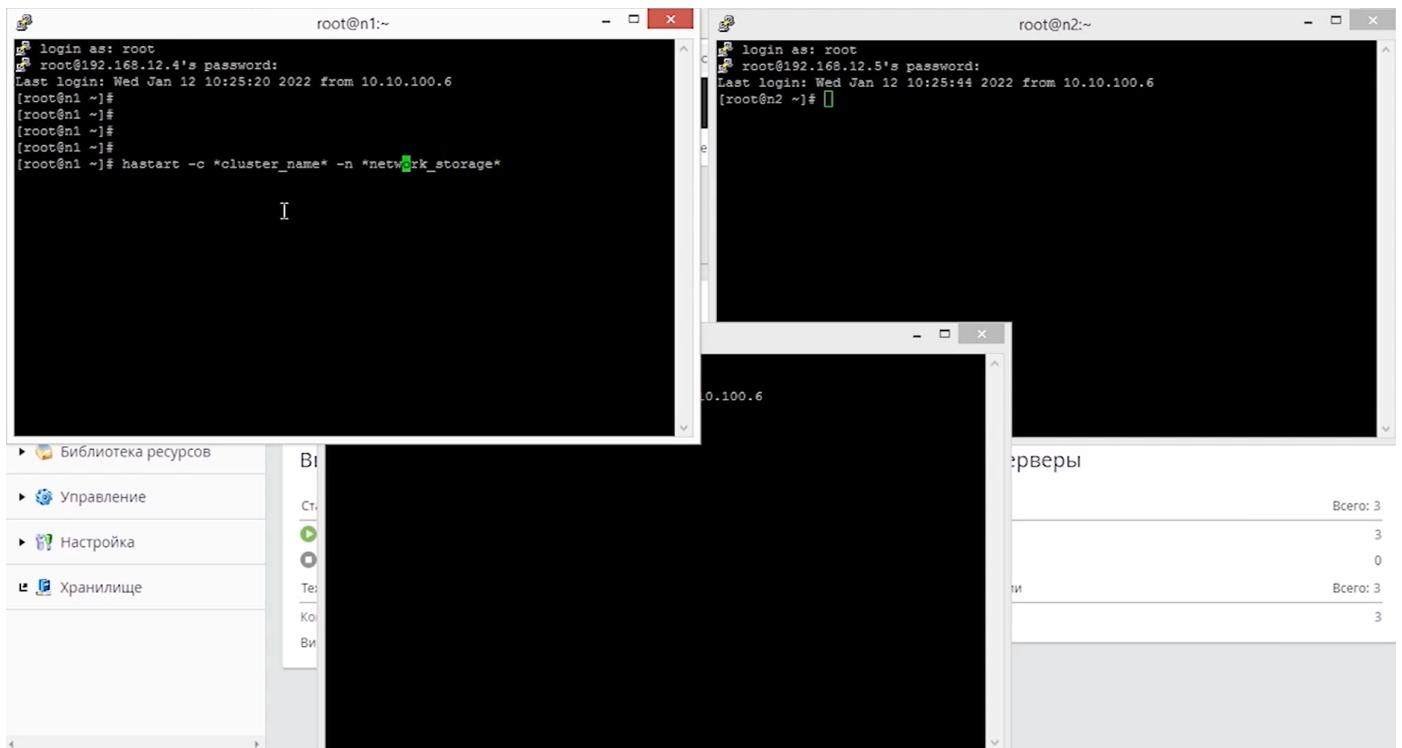
Для этого в верхней части выберем пункт "**виртуальные среды**", выделим оба наших контейнера и нажмем на кнопку "**мигрировать**", например можно на вторую ноду.

Нажмем "**подробности**" для мониторинга выполнения миграции и дождемся завершения выполнения процесса.



Скриншот 55.

Для этого в верхней части выберем пункт "**виртуальные среды**", выделим оба наших контейнера и нажмем на кнопку "**мигрировать**", например можно на вторую ноду.



Скриншот 56.

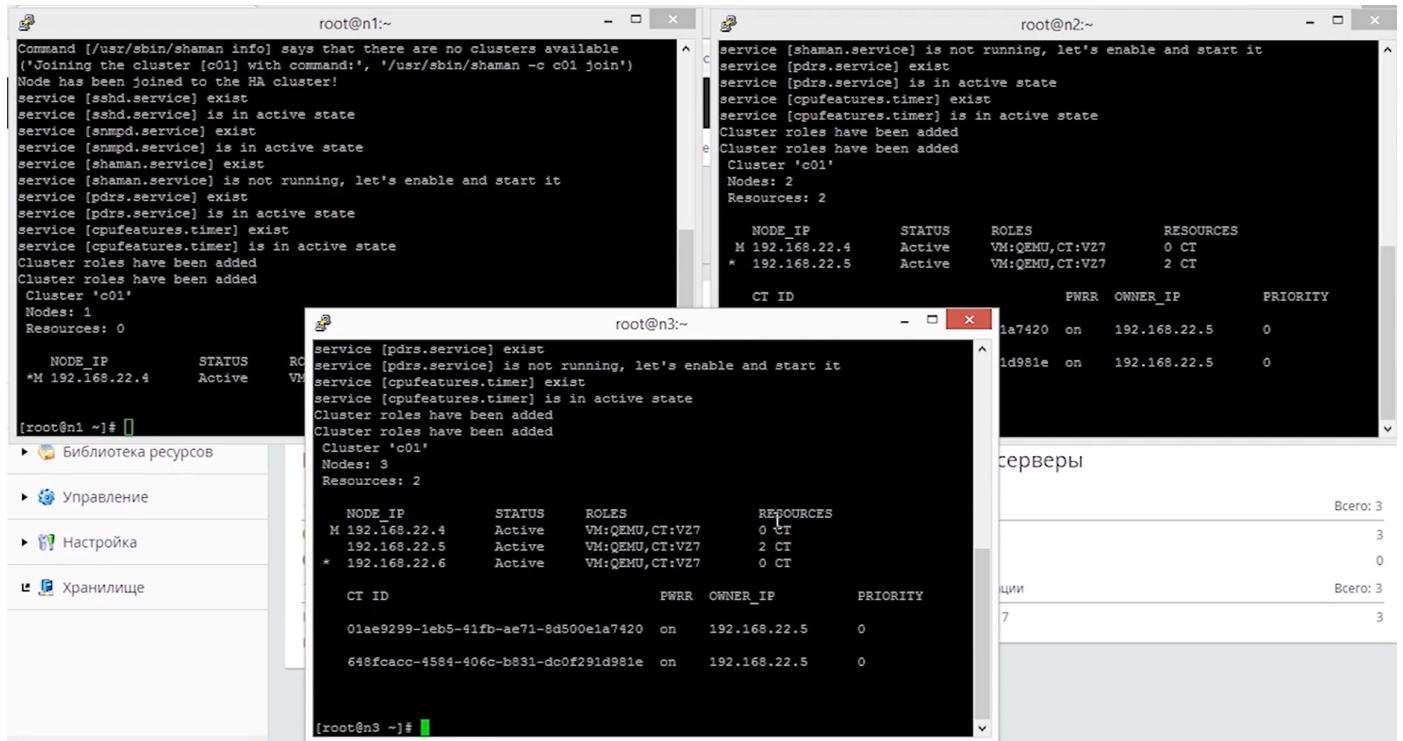
3.21 запуск службы отказоустойчивости на серверах

Далее, чтобы при отказе работы ноды виртуальные среды с нее автоматически возобновляли свою работу на другой исправной ноде, необходимо запустить службу отказоустойчивости **“Shaman”** на всех узлах кластера.

Для этого надо подключиться к нодам по **SSH** и выполнить определенную команду:

```
#hastart -c c01 -n 192.168.22.0/24
```

Значения команды выглядят следующим образом: после ключа **-c** надо указать имя кластера, в нашем случае это **c01**, и после ключа **-n** указать сеть хранилища, в нашем случае это **192.168.22.0/24** - именно сеть хранилища.



Скриншот 57.

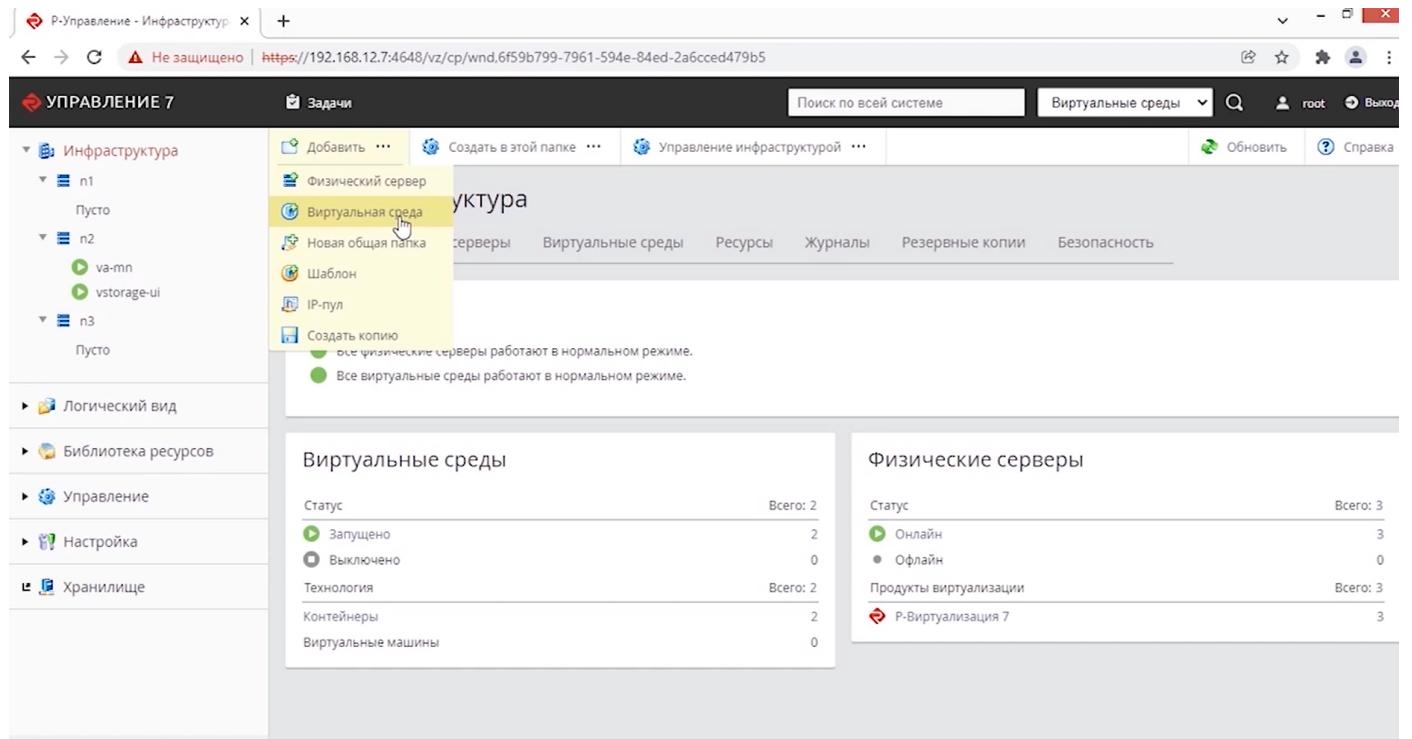
Если команда выполнена, то служба на первой ноде запуститься. Скопируем ее и выполним на оставшихся узлах, после чего в выводе команды видны все три узла, их адреса, статус, и видим, что из ресурсов есть только два контейнера, отображаются их **ID**, ниже видим IP-адреса нод, на которых они находятся. Видим, что здесь это вторая нода. Теперь если какая-либо нода выйдет из строя, работа виртуальных сред автоматически будет возобновлена на других исправных нодах.

Проверка статуса службы НА “**shaman**” выполняется следующей командой на любой ноде:

```
#shaman stat
```

Важно!

Если в процессе инсталляции кроме датасторов для виртуализации настраивалась роль экспорта **iSCSI** или **s3** на сетевых интерфейсах для внешних систем, то служба “**shaman**” автоматический запустится и выполнять команду **hastart** не потребуется.



Скриншот 58.

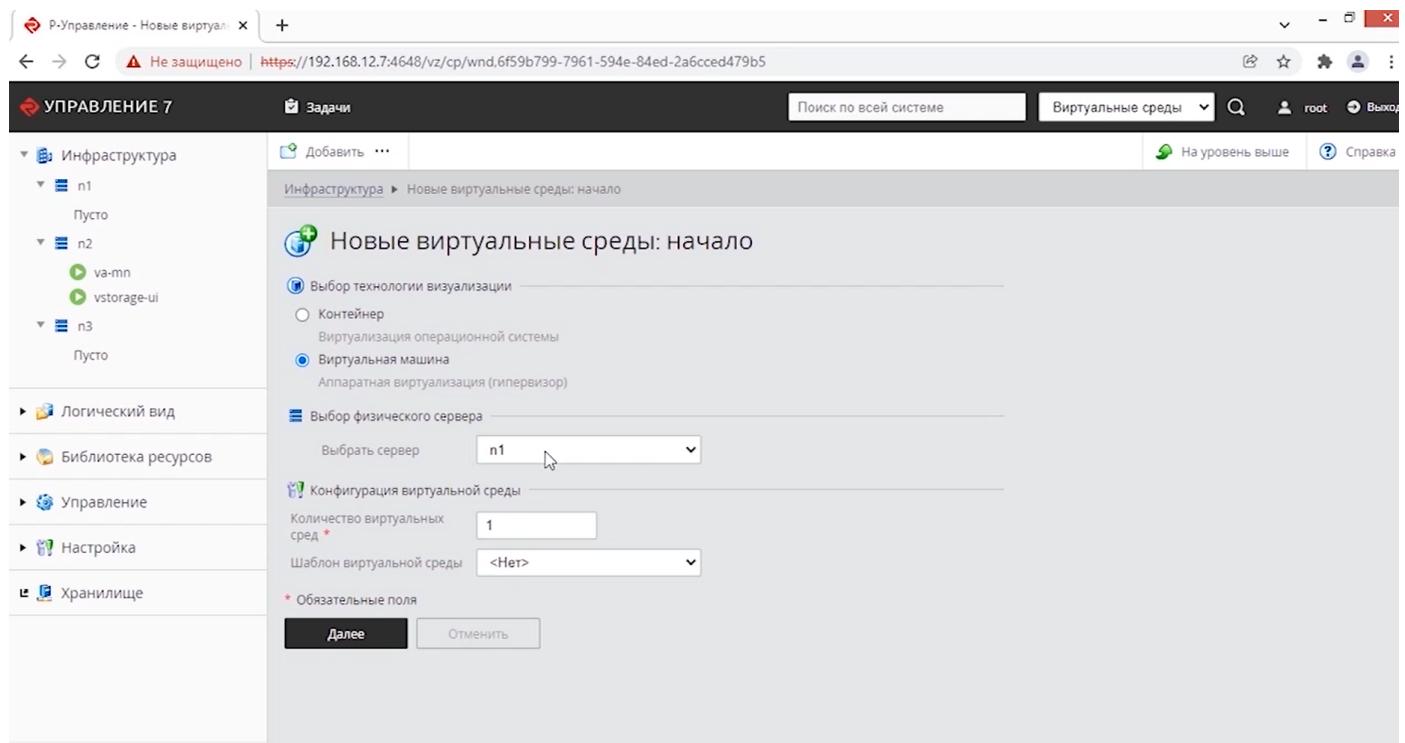
3.22 создание виртуальных сред

Теперь можно перейти к созданию виртуальных сред. Создадим для примера две виртуальные машины с **Windows Server 2019** и **Ubuntu**.

Будет выполняться установка с **iso**-образов, которые были предварительно загружены на первую ноду через программу **SCP** в папку **root**. Для создания виртуальной машины нажмем кнопку "**добавить**", выберем пункт "**виртуальная среда**".

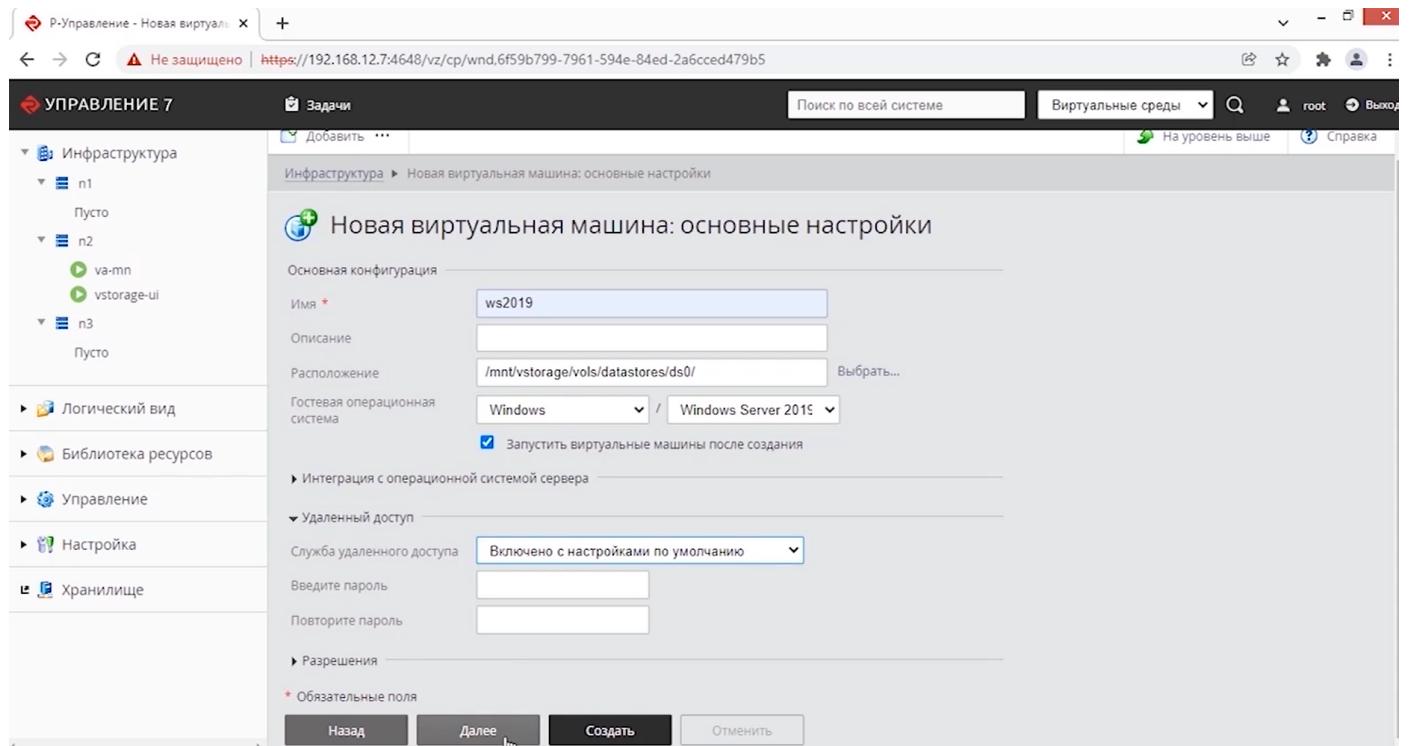
Важно!

имеется несколько вариантов расположения ISO образов для запуска установки гостевых операционных систем: 1. В “Р-управлении” в разделе “**библиотеки ресурсов**” с предварительно настроенной внешней “shared” папкой по протоколу CIFS/SMB. 2. Папка с расположением в датасторе “Р-хранилища” или в папке с расположением самой виртуальной среды (использует полезное пространство “Р-хранилища”). 3. Локальное расположение на ноде в папке пользователя по умолчанию root или в другой папке, но видимость зависит от прав. 4. Сетевое расположения с заранее настроенной смонтированной папкой от внешнего источника по NFS. 5. другие варианты.



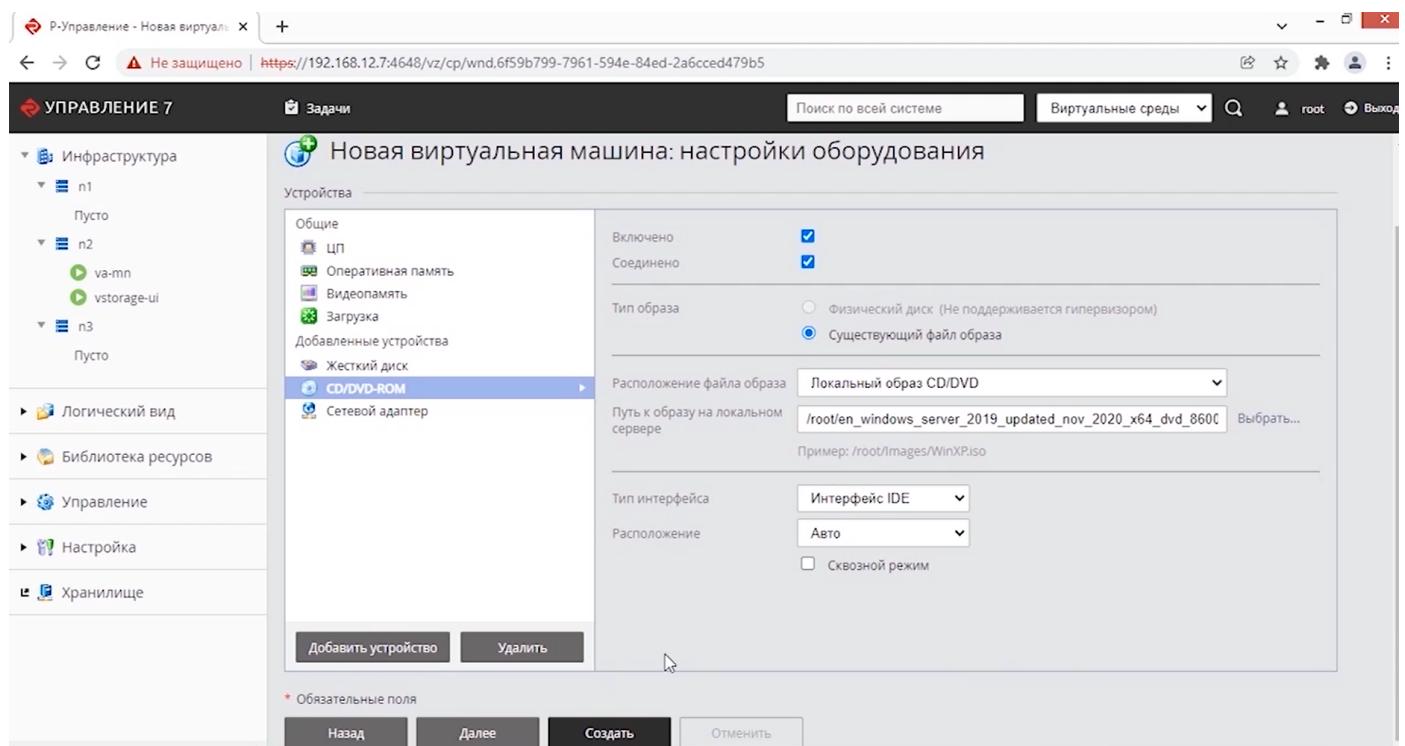
Скриншот 59.

Далее выберем в меню под добавить выберем “**виртуальную машину**”, оставим первую ноду в графе выбора физического сервера, так как именно на нем iso-образы, нажмем “**далее**”.



Скриншот 60.

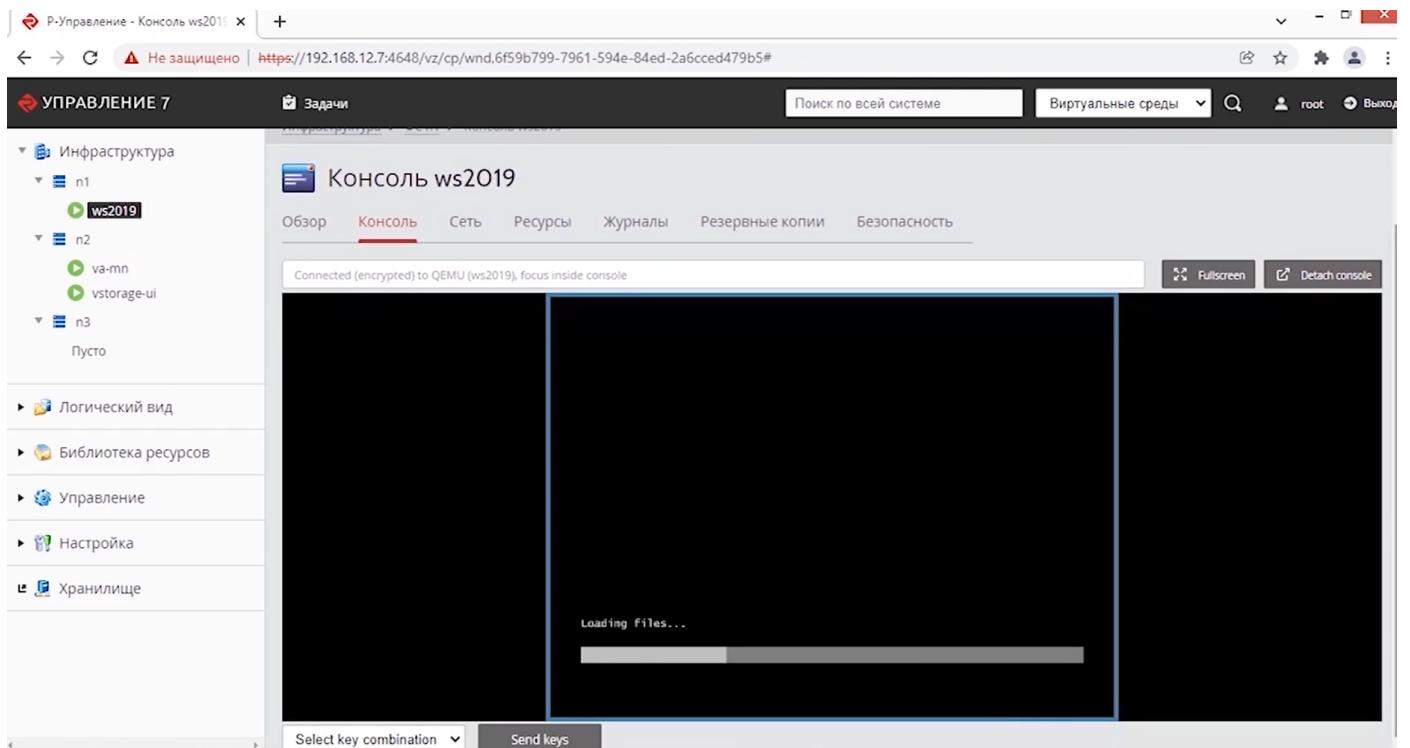
В качестве первой виртуальной машины будет установлена **Windows Server 2019**, поэтому назначим имя **ws2019** и в графе "**гостевая операционная система**" выберем соответствующий пункт. Во вкладке "**удаленный доступ**" выберем параметр **«включено с настройками по умолчанию»**, и остальные настройки остаются без изменений, нажимаем кнопку "**далее**".



Скриншот 61.

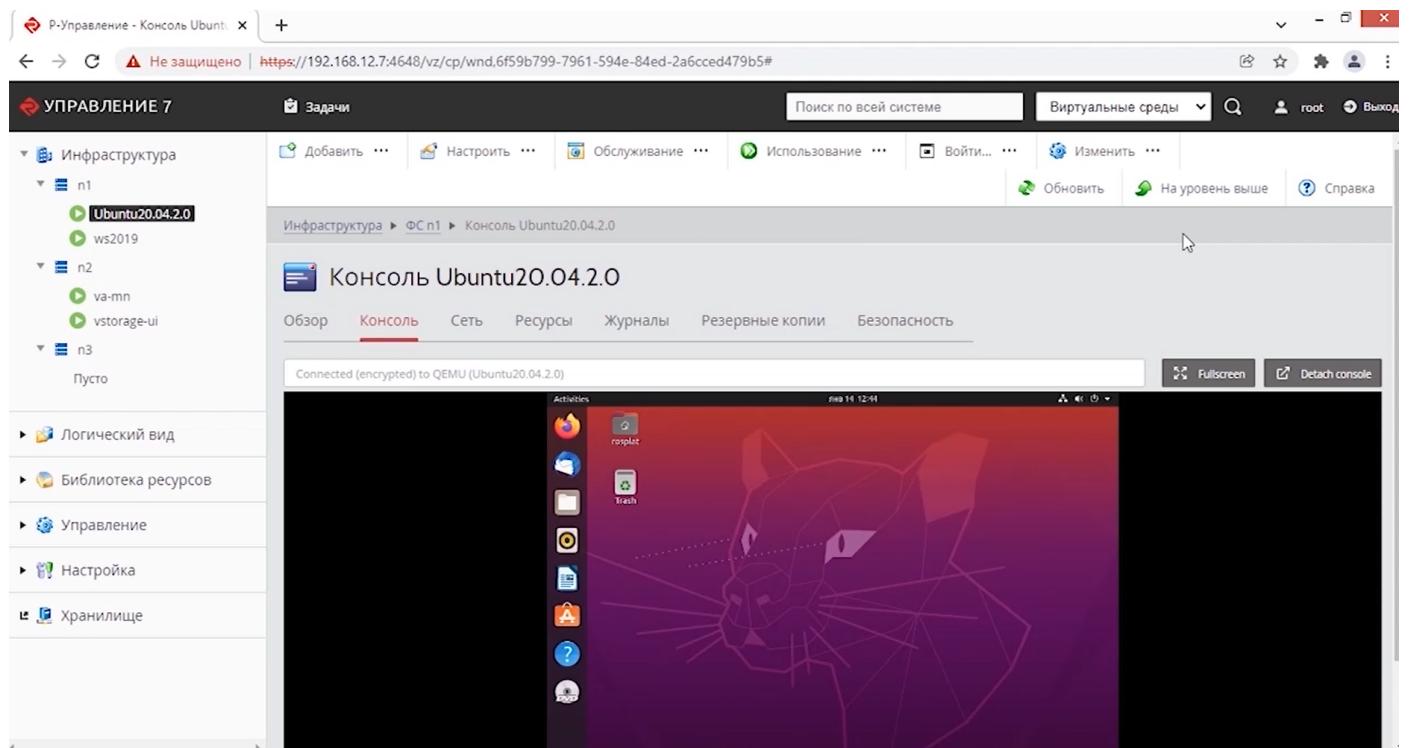
На этом этапе задаем аппаратную конфигурацию виртуальной машины, в настройках **cd/dvd-rom** указывается **ISO** образ. Нажмем кнопку "**выбрать**", откроем папку root и выберем образ **Windows Server 2019**, нажимаем "**выбрать**", затем нажимаем кнопку "**создать**".

Можно нажать "**подробности**" для мониторинга выполнения операции и дождемся завершения создания виртуальной машины. Как создание виртуальной машины выполнено, она у нас появится на первой ноде. Если нажмем на нее и перейдем в "**консоль**", запустится она автоматически сразу после создания, как и было задано в параметрах во время ее создания.



Скриншот 62.

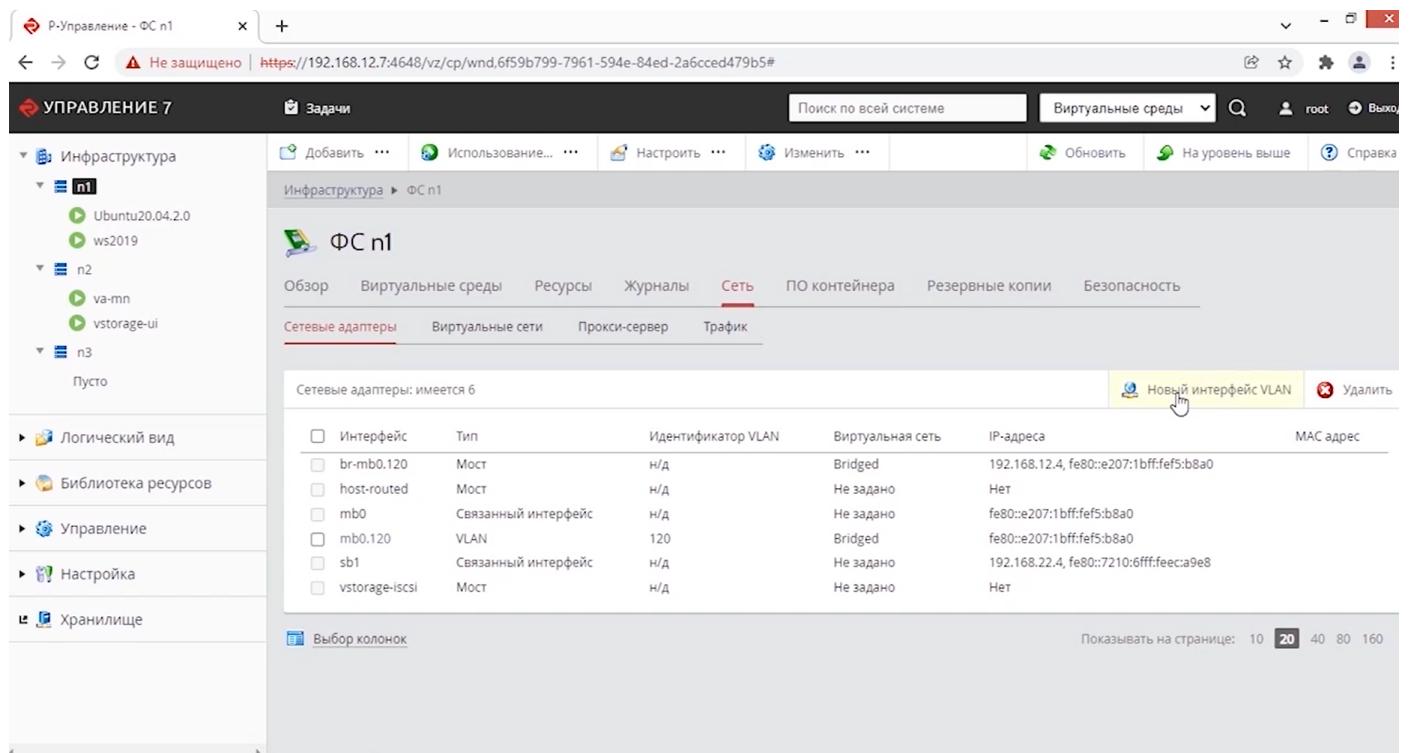
В консоли через браузер(по **VNC**) уже можно пройти процесс установки ОС. Если все прошло удачно, то повторим эту процедуру для создания еще одной виртуальной машины, но уже с **Ubuntu**. Где в графе "**гостевая операционная система**" сменим **Windows** на **Linux** и в уточняющей графе укажем **Ubuntu Linux**, все остальное делаем аналогично.



Скриншот 63.

3.23 создание VLAN и добавление в него виртуальных машин

После того, как созданы виртуальные машины, можно создать виртуальную сеть и добавить виртуальные машины в определенный **VLAN**. Для этого сначала надо создать интерфейс **VLAN**.

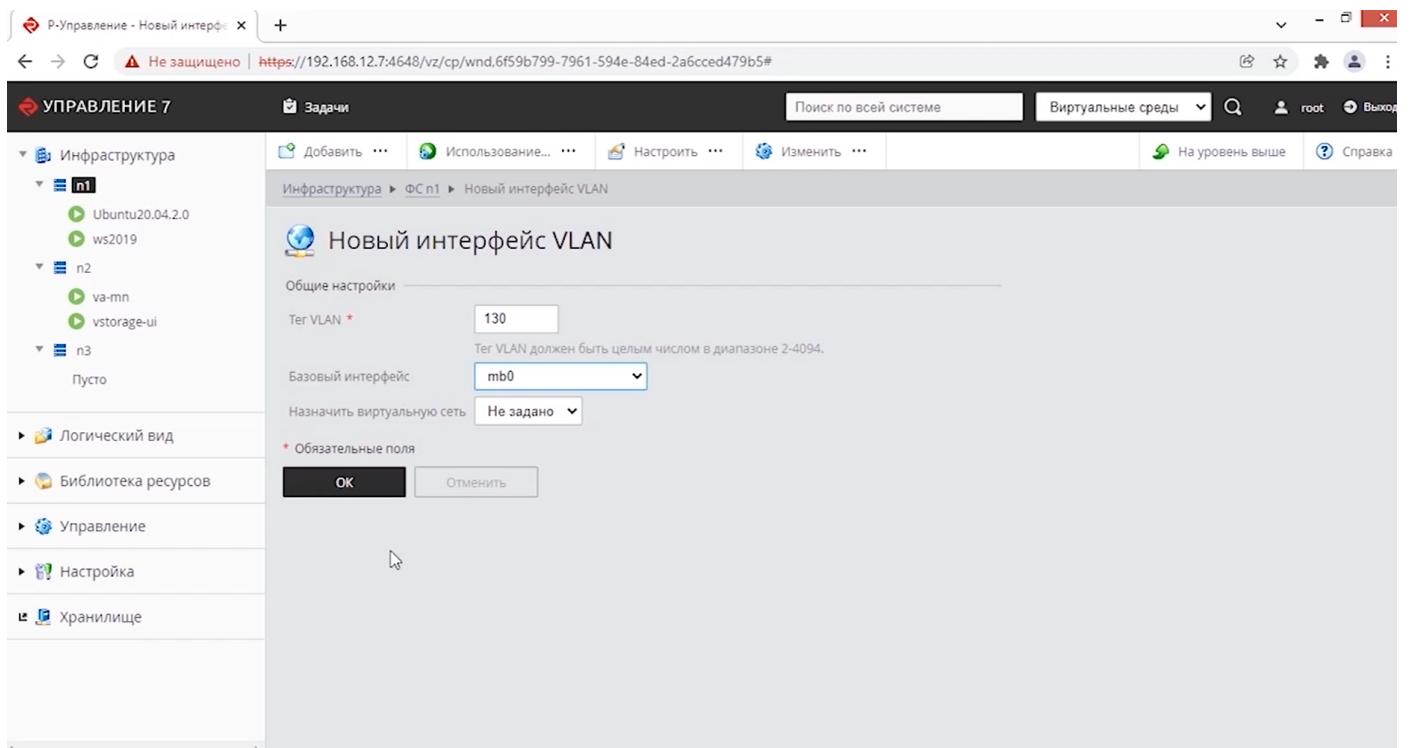


Скриншот 64.

Выберем первую ноду, перейдем во вкладку "**сеть**", и здесь во вкладке "**сетевые адаптеры**" нажмем на кнопку "**новый интерфейс VLAN**".

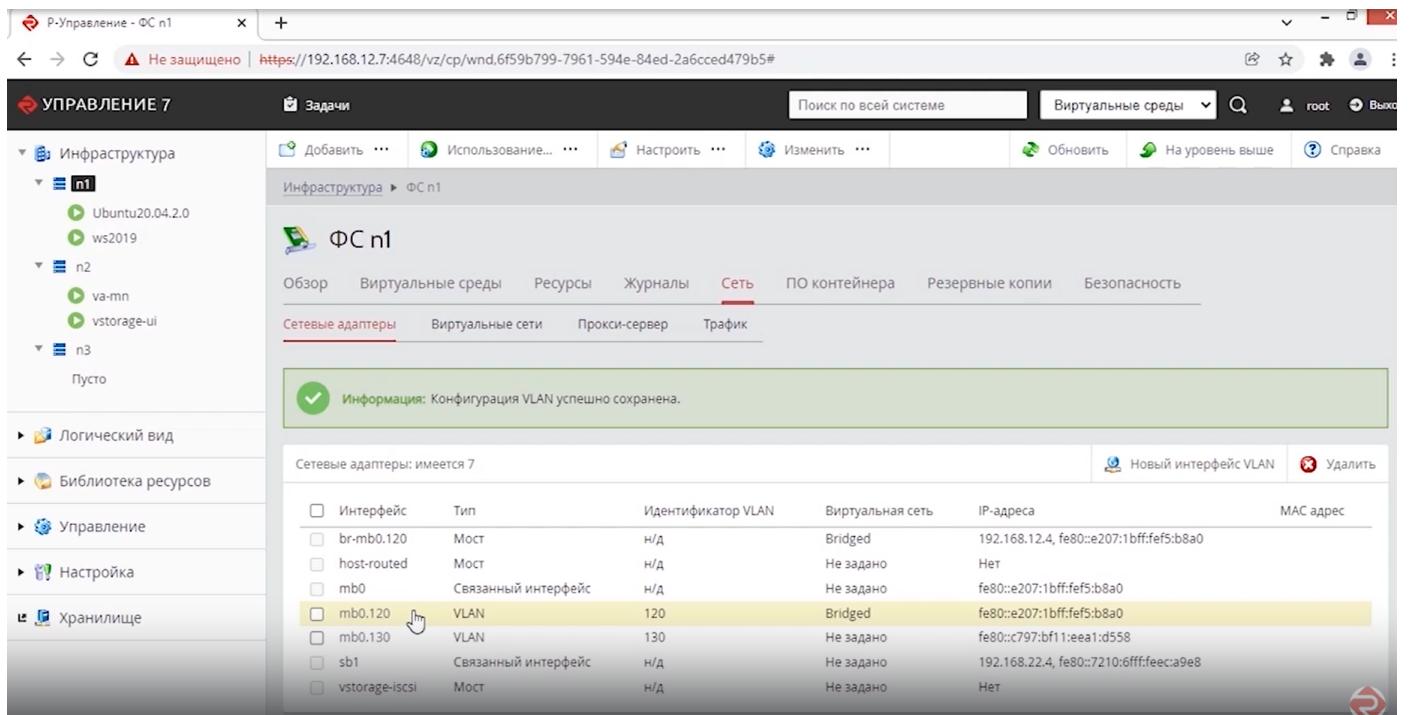
✓ **Важно!**

VLAN теги должны быть предварительно настроенные на физических коммутаторах, которые подключены к серверам, где один из **VLAN** будет использоваться для сети управления и панелей "**P-управления**" и "**P-хранилища**", а остальные уже под ваши задачи.



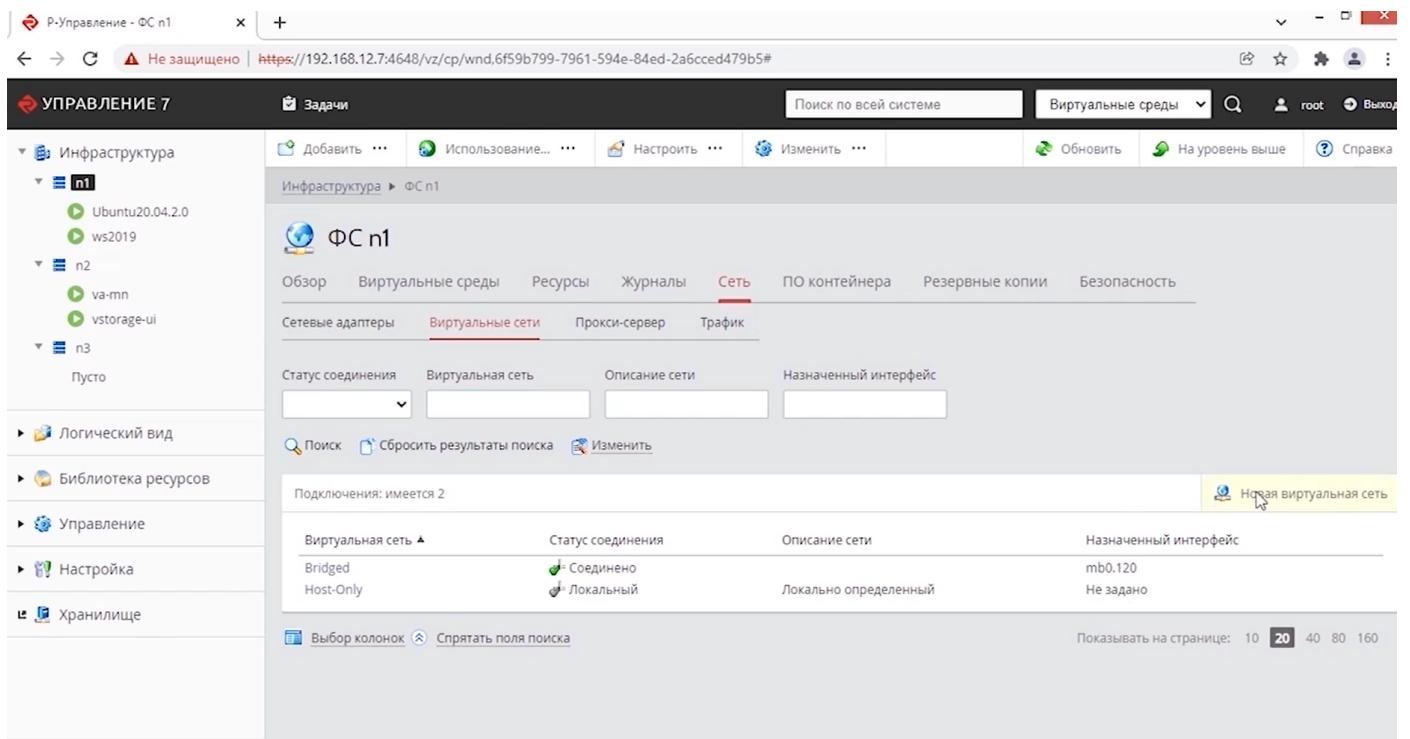
Скриншот 65.

Тег VLAN для примера 130. В качестве базового интерфейса используется Bond сети управления. В нашем случае mb0, нажимаем OK.



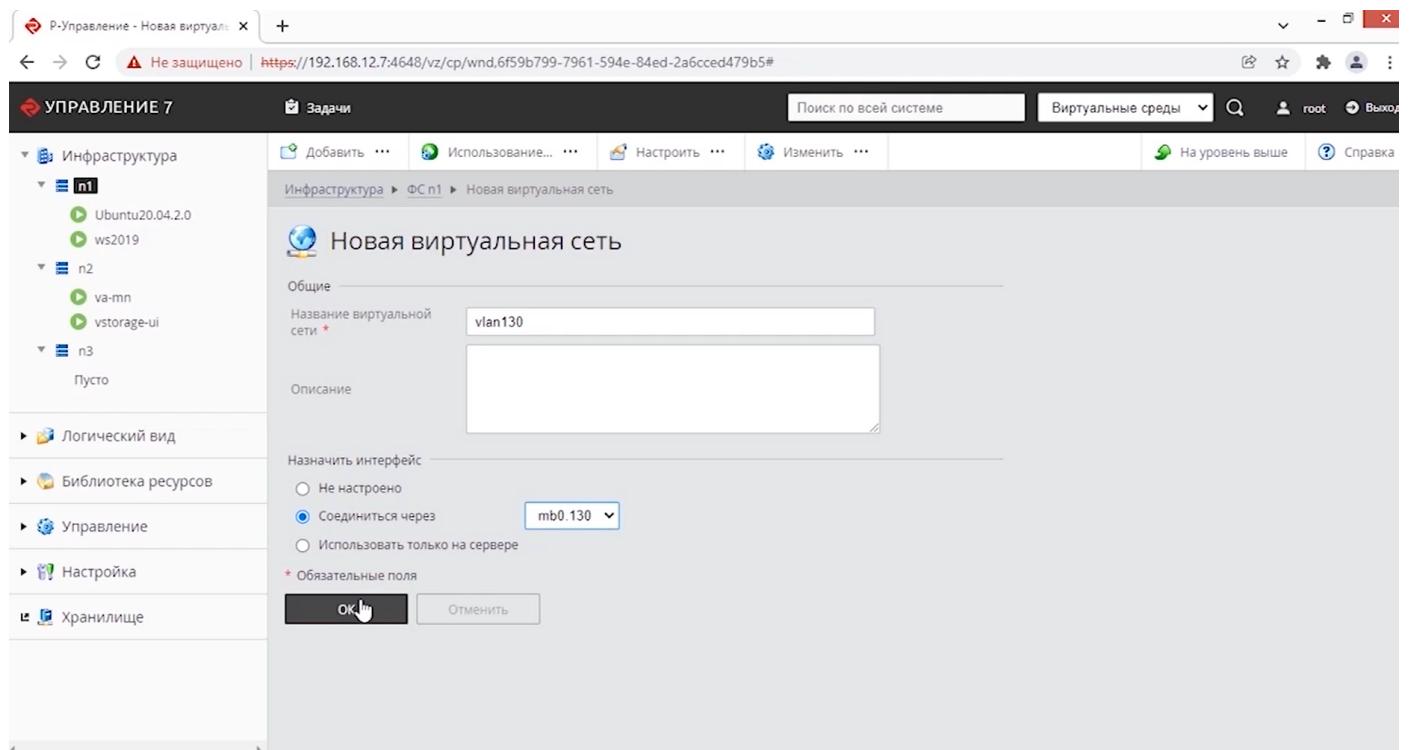
Скриншот 66.

После того как создался новый сетевой адаптер, под называнием **mb0.130**. Он аналогичен тому названию, которое задали на этапе установки для создания интерфейса **VLAN 120**, и дальнейшее администрирование будет происходить с названиями интерфейсов по такому же принципу.

*Скриншот 67.*

Теперь перейдем во вкладку "**виртуальные сети**" и создадим виртуальную сеть для виртуальных машин. Нажмем на кнопку "**новая виртуальная сеть**".

Назовем ее, например, **VLAN130**, в графе "**назначить интерфейс**" выберем "**соединиться через mb0.130**" и нажмем **OK**.



Скриншот 68.

Нажмем "**обновить**", видна новая виртуальная сеть. Эту настройку надо выполнить на всех узлах кластера.

Важно!

на каждом узле необходимо только создать через "**R-управление**" VLAN интерфейс и добавить к виртуальной сети, а виртуальная сеть автоматический создается на всех узлах. Для того чтобы автоматизировать создание и привязку VLAN интерфейсов к виртуальным сетям на всех узлах можно воспользоваться документации с описанием cli команд: https://updates.rosplatforma.ru/docs/7.0.13/r-virt_cli.pdf или написать в поддержку по адресу support@rosplatforma.ru с просьбой выдать готовый скрипт ansible.

УПРАВЛЕНИЕ 7

Инфраструктура

- m1**
 - Ubuntu20.04.2.0
 - ws2019
- n2**
 - va-mn
 - vstorage-ui
- n3**
 - Пусто

ФС n1

Сеть

Виртуальная сеть	Статус соединения	Описание сети	Назначенный интерфейс
Bridged	Соединено	mb0.120	
Host-Only	Локальный	Не задано	
vlan130	Соединено	mb0.130	

Скриншот 69.

Теперь, чтобы добавить виртуальную машину в **VLAN**, нажмем на любую виртуальную машину.

УПРАВЛЕНИЕ 7

Инфраструктура

- n1**
 - Ubuntu20.04.2.0
 - ws2019
- n2**
 - va-mn
 - vstorage-ui
- n3**
 - Пусто

Виртуальная машина

Ubuntu20.04.2.0

Обзор

Статус	Запущено
Физический сервер	n1
Использование ресурсов ЦП	н/д
Работает	н/д
Гостевые инструменты	Не установлено

Общие настройки

Имя	Ubuntu20.04.2.0
UUID	ffbaa78b-b1d6-4f41-9e1f-3a2250063724
Расположение	/mnt/vstorage/vols/datastores/ds0/ffbaa78b-b1d...
IP-адреса	н/д
Гостевая операционная	

Скриншот

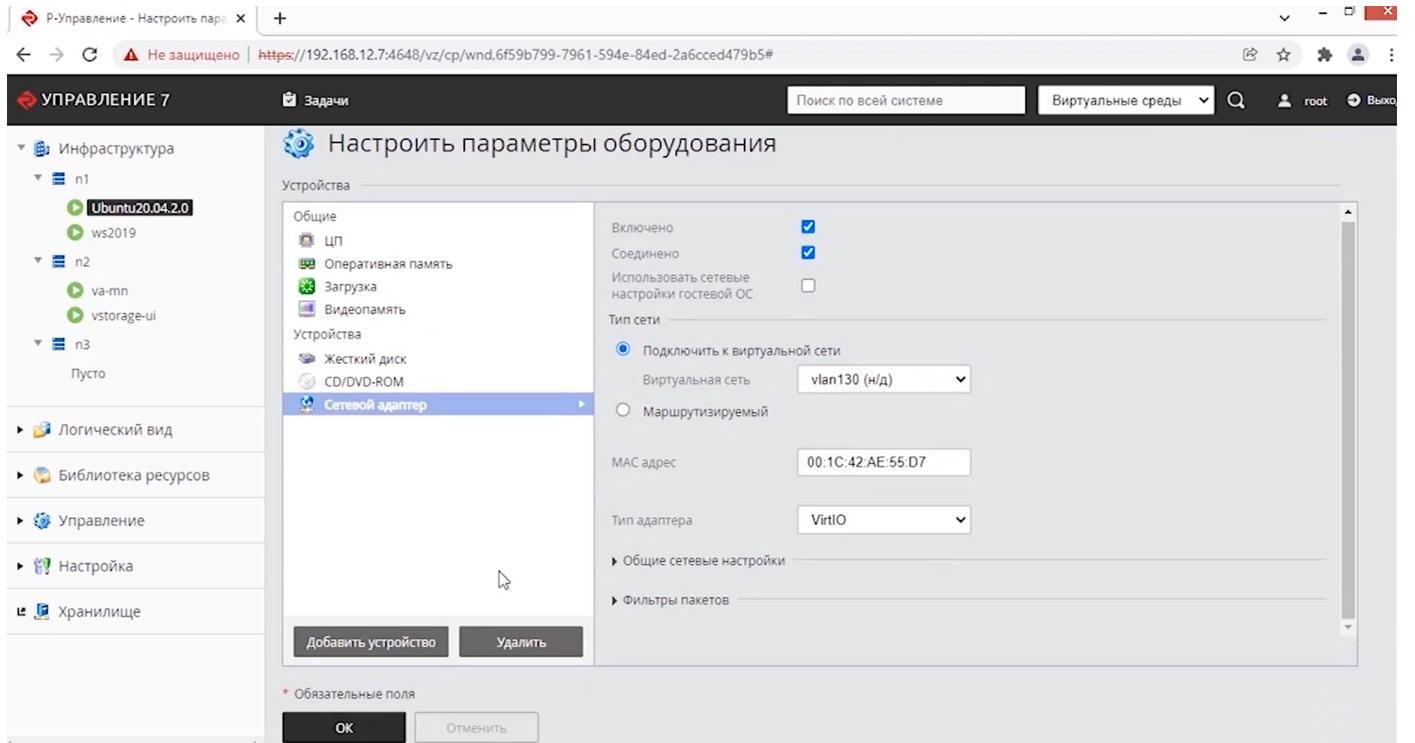
Предупреждения о ресурсах

Виртуальная среда работает в нормальном режиме.

Ресурсы

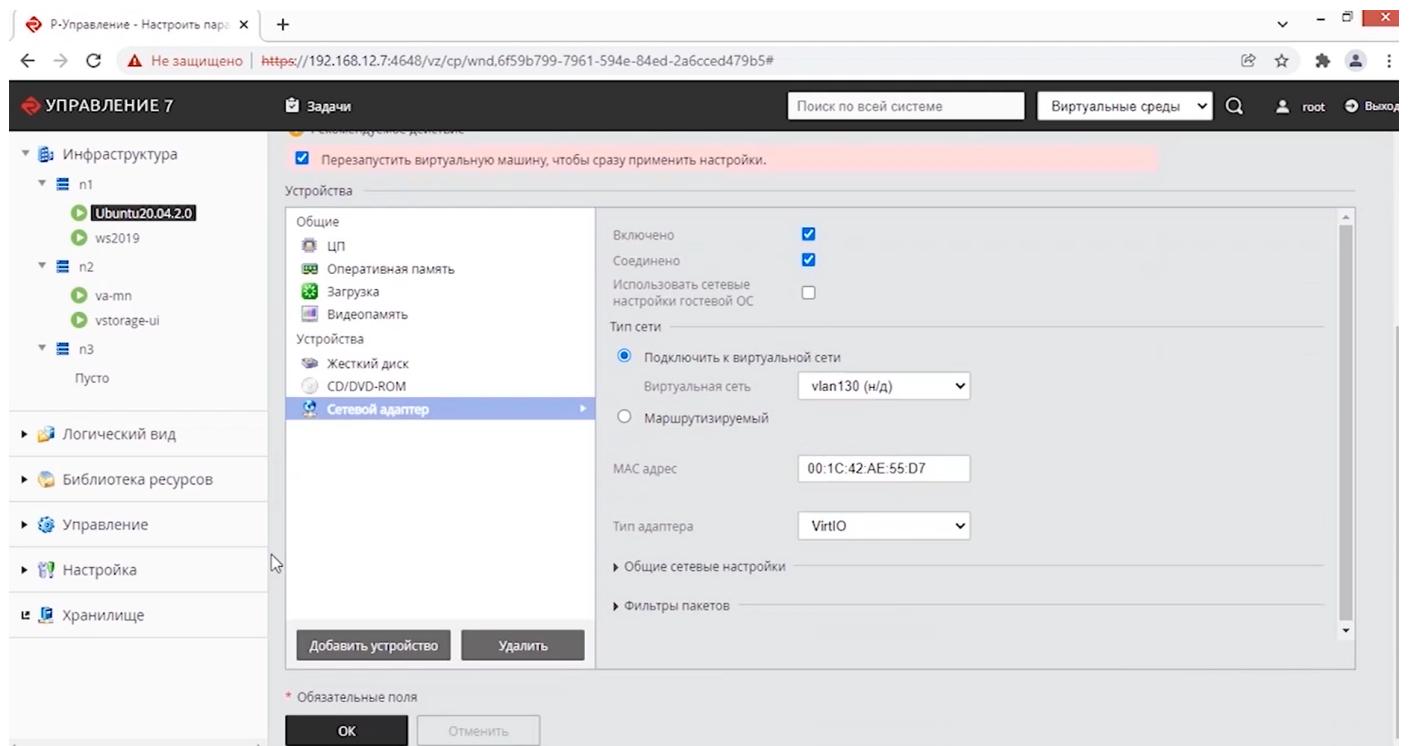
Скриншот 70.

Далее нажмем кнопку "**настроить**" и перейдем в настройки оборудования.

*Скриншот 71.*

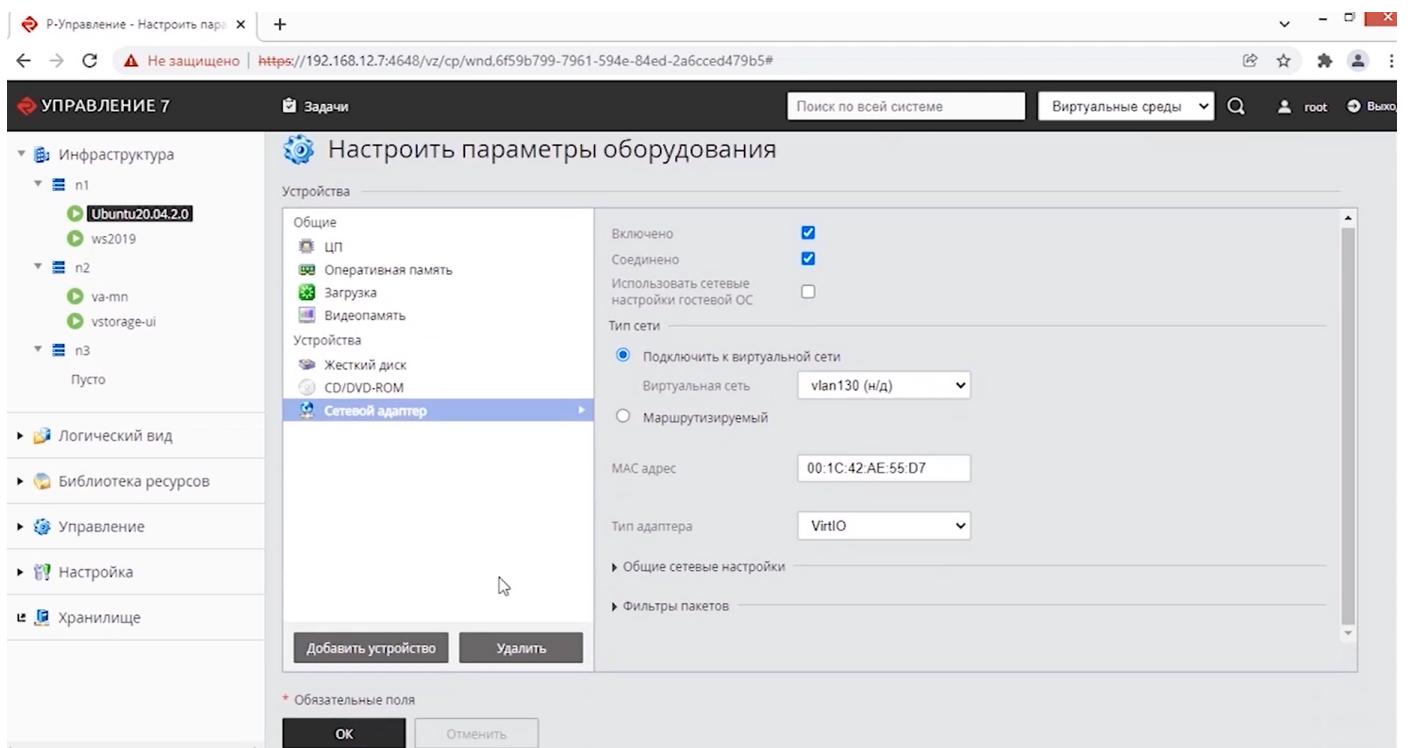
Выберем сетевой адаптер и в качестве виртуальной сети выберем нужную виртуальную сеть. Нажмем **OK**.

Если в настоящий момент виртуальная машина работает, то для применения этой настройки потребуется ее перезагрузка. Поставим соответствующую галочку и еще раз нажмем **OK**.



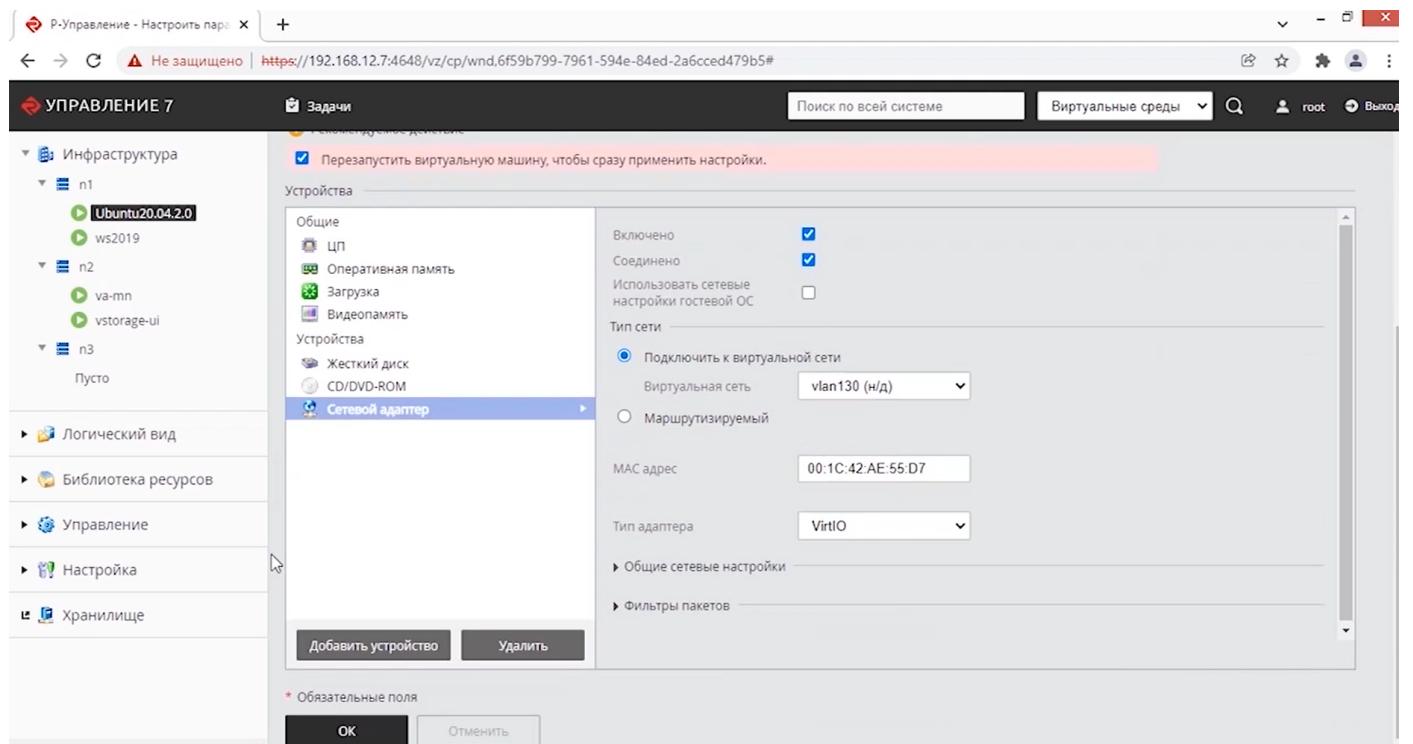
Скриншот 72.

Настройка применена, и теперь виртуальная машина **Ubuntu 20.04.20** находится в **VLAN 130**. После этого одна виртуальная машина находится в **VLAN 120**, вторая в **VLAN 130**. Настроим работу обоих виртуальных машин в **VLAN 130** и проверим сетевую доступность между ними.



Скриншот 73.

В 120 VLAN находится Windows Server 2019. Выберем его, нажмем «настроить» далее - вкладку «настройки оборудования», перейдем в сетевой адаптер и здесь сменим виртуальную сеть **Bridged**, которая в этом случае связана на **vlan 120**, на виртуальную сеть **vlan130**, нажмем **OK**.

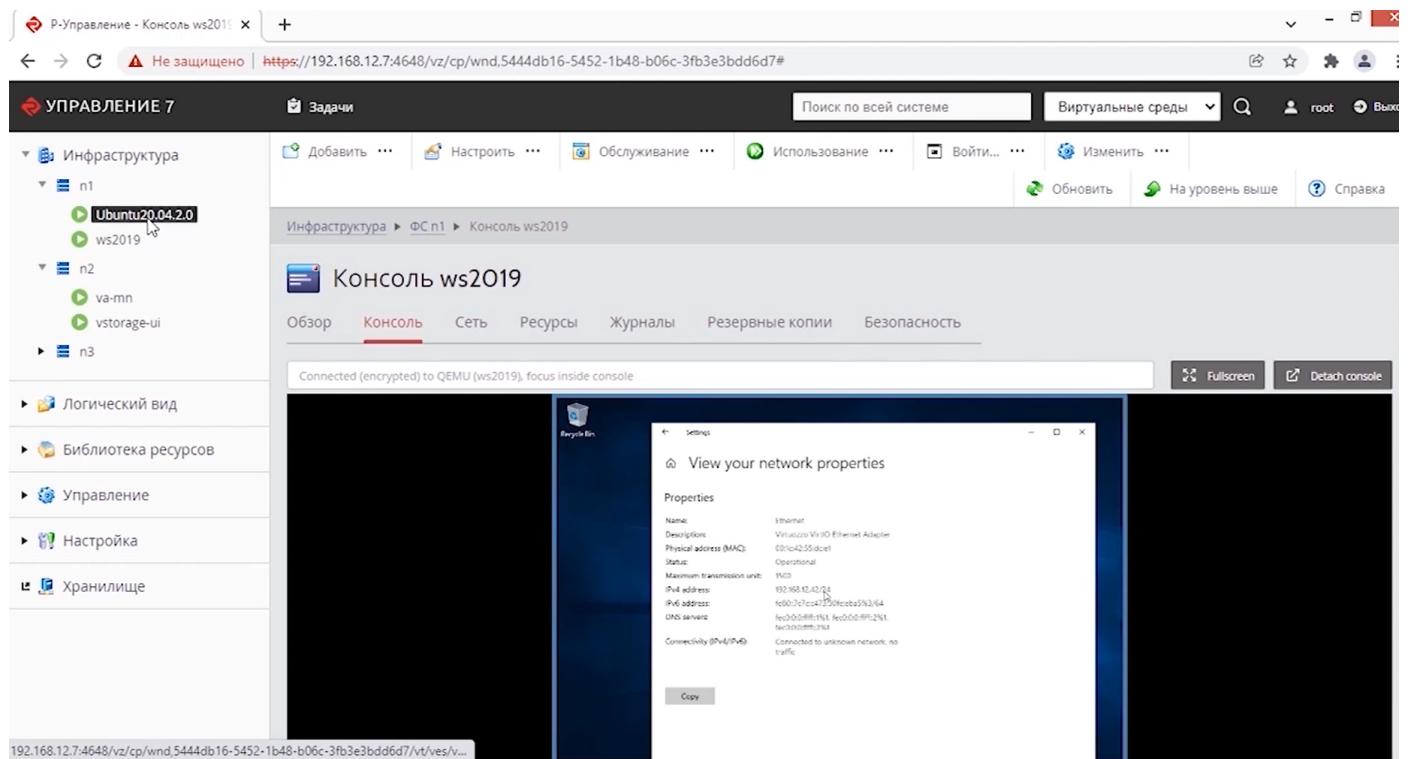


Скриншот 74.

Для применения настроек потребуется перезагрузка ВМ, поставим галочку, ещё раз **OK** и ждем применения настройки.

Важно!

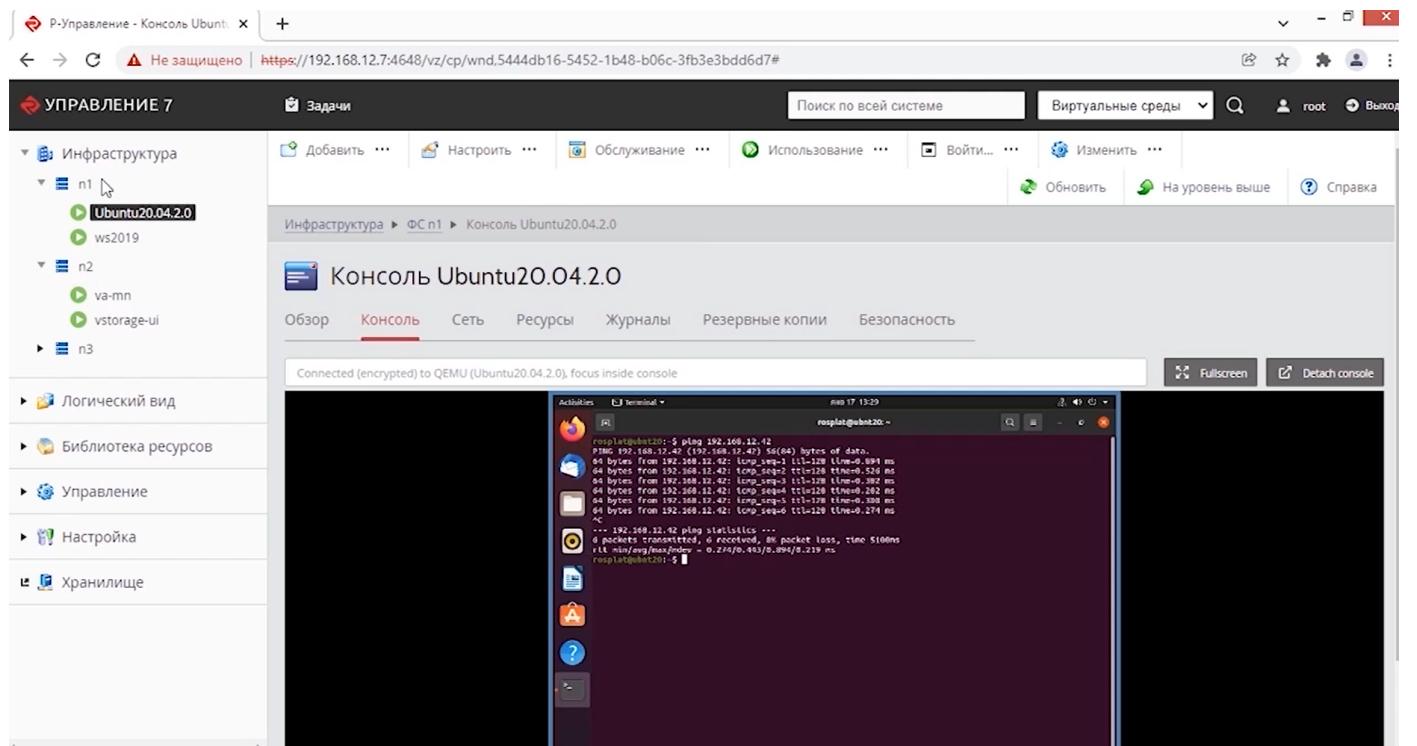
По умолчанию перезагрузка ВМ выполняется для корректного выключения гостевой операционной системы. Выполнить применение сетевых настроек на горячую (без перезагрузки) можно через команду **cli** описанную в следующей документации https://updates.roplatforma.ru/docs/7.0.13/r-virt_cli.pdf



Скриншот 75.

3.24 проверка сетевой доступности между виртуальными машинами одного vlan

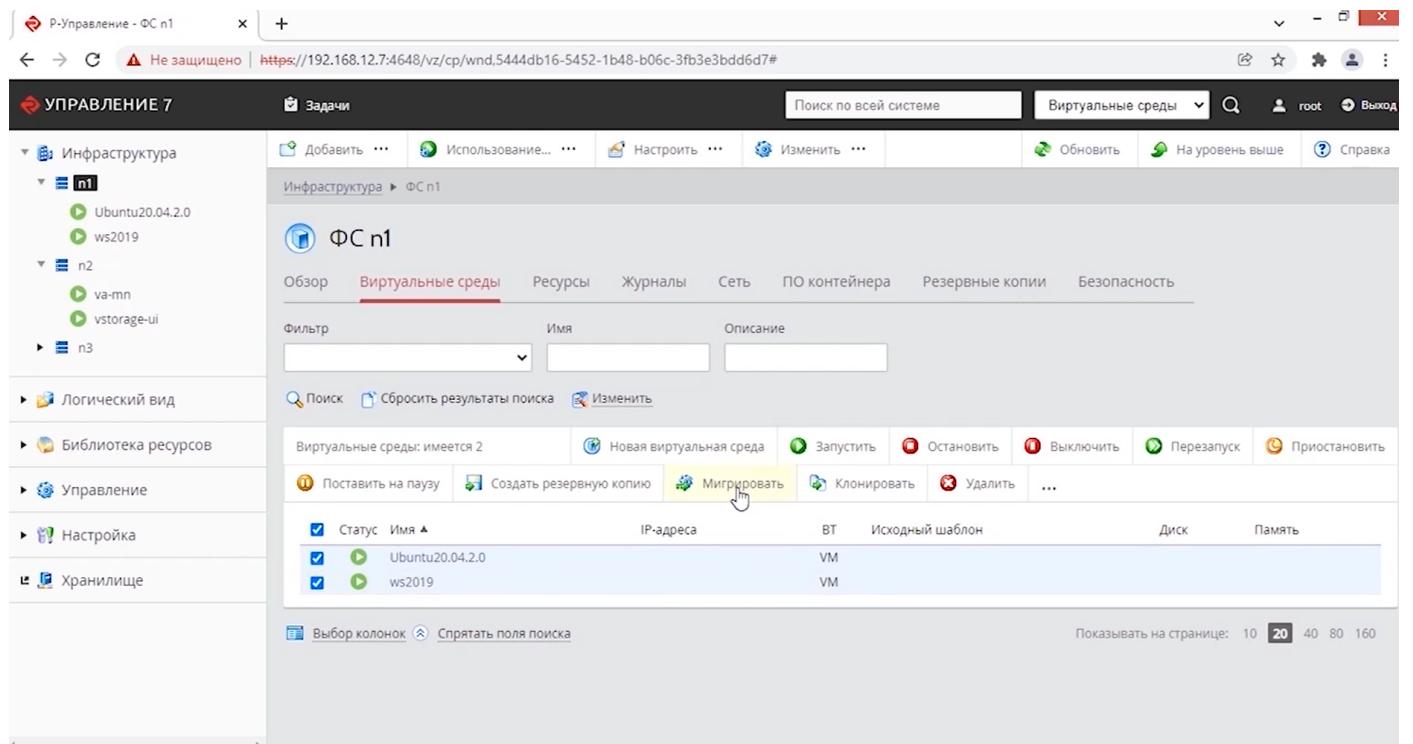
Итак, виртуальная машина перезагрузилась, теперь проверим сетевую доступность между этими виртуальными машинами. Перейдем в консоль, посмотрим IP адрес на Windows сервере. IP адрес - 192.168.12.42



Скриншот 76.

Перейдем в **ubuntu** и попробуем запустить **ping**.

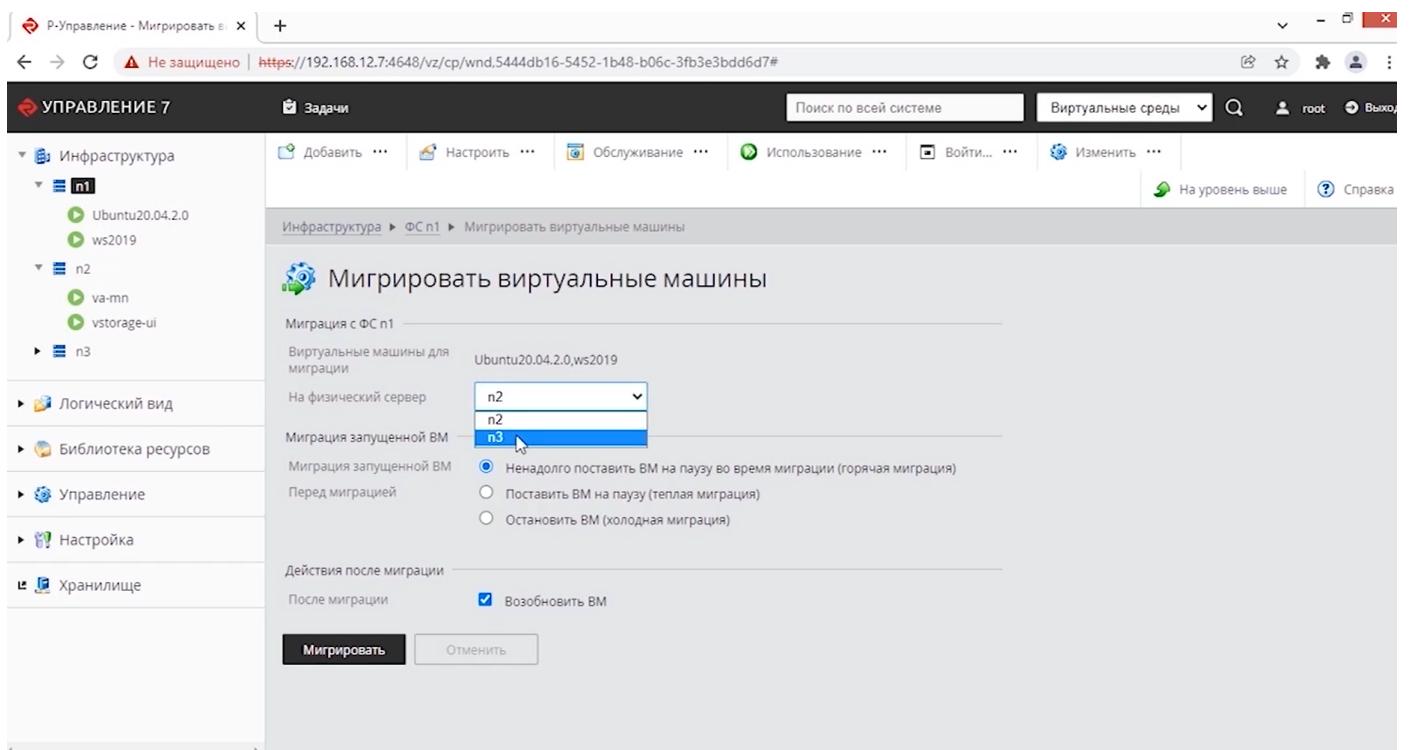
Сетевая доступность есть, работает исправно как показано на скриншоте 76. Можно попробовать проверить работу миграции виртуальных сред с одного узла на другой.



Скриншот 77.

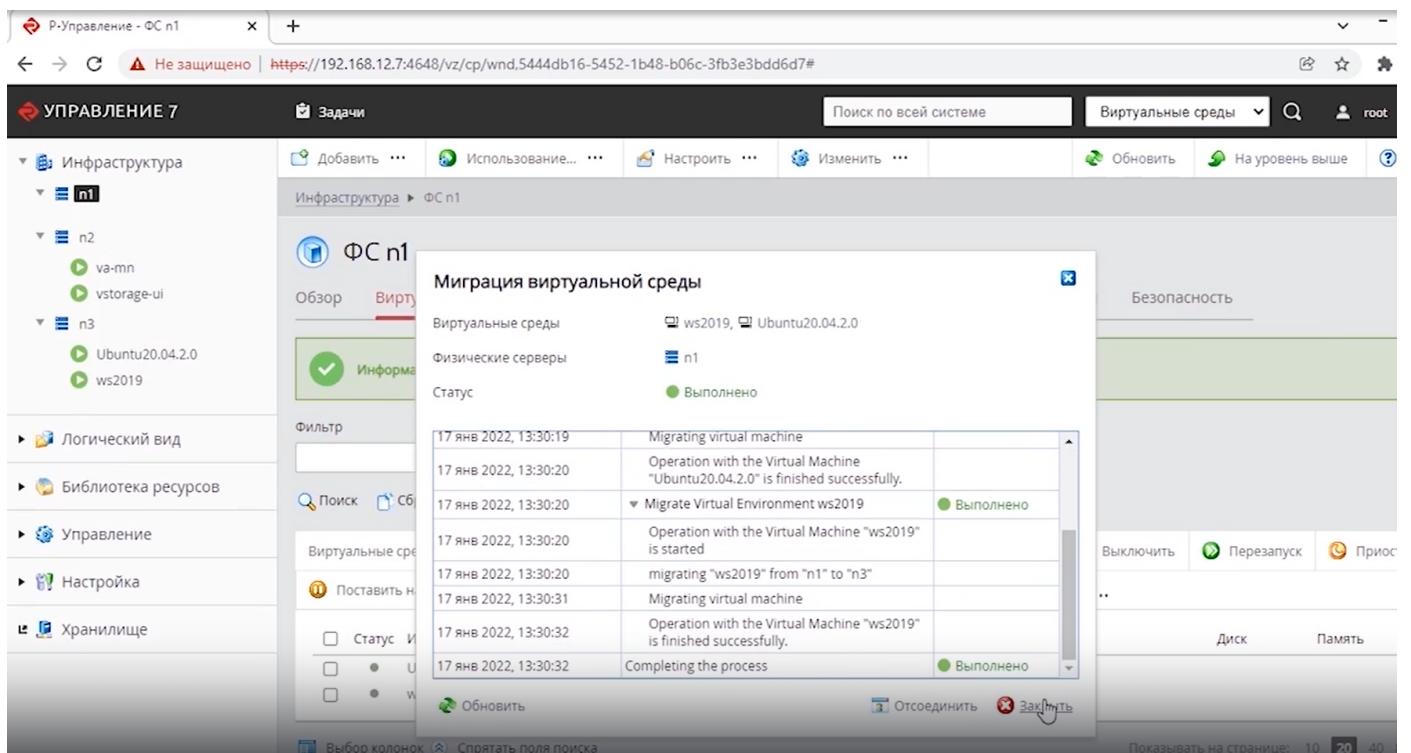
3.25 проверка живой миграции виртуальных сред

Перейдем на первую ноду, выберем виртуальные среды. Будем мигрировать сразу две виртуальные машины, выделим их и нажмём кнопку миграировать.



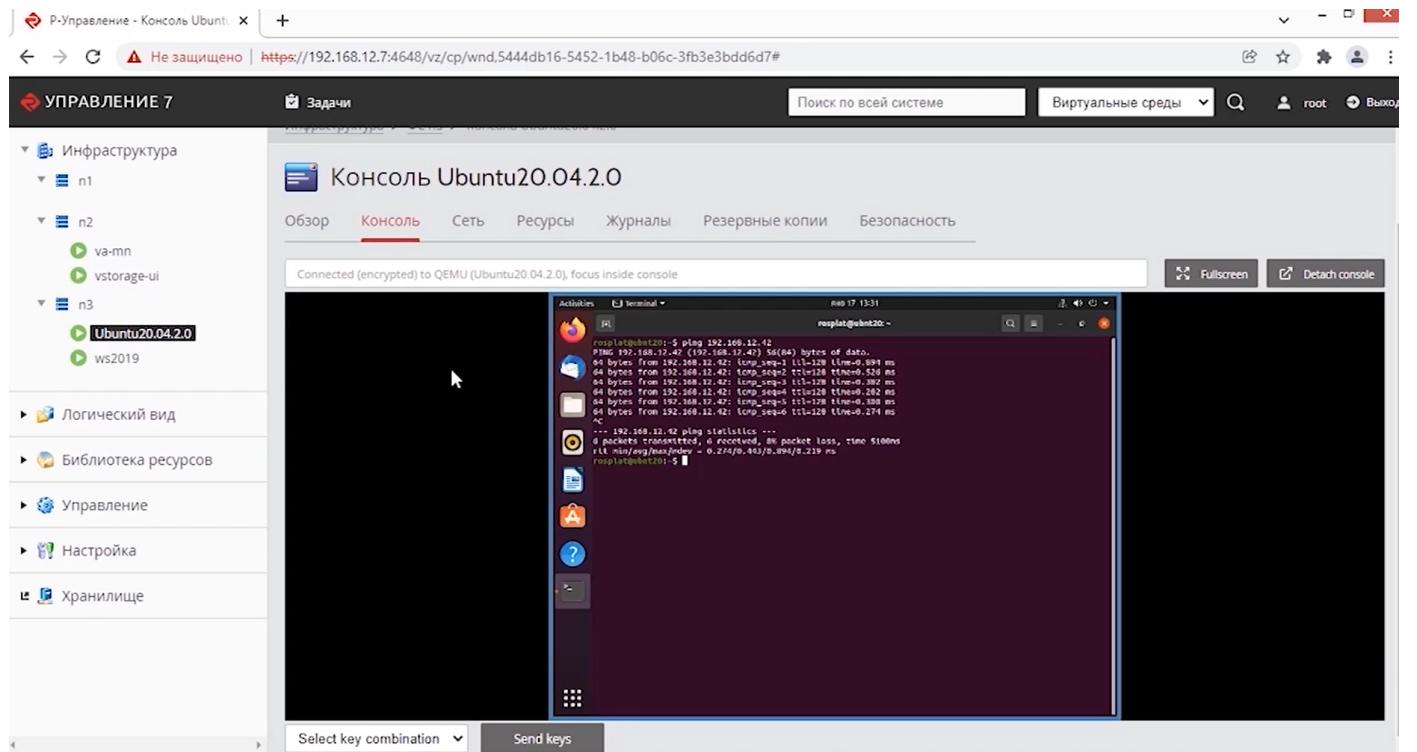
Скриншот 78.

Выберем, например, третью ноду, нажимаем мигрировать и дождемся завершения миграции.

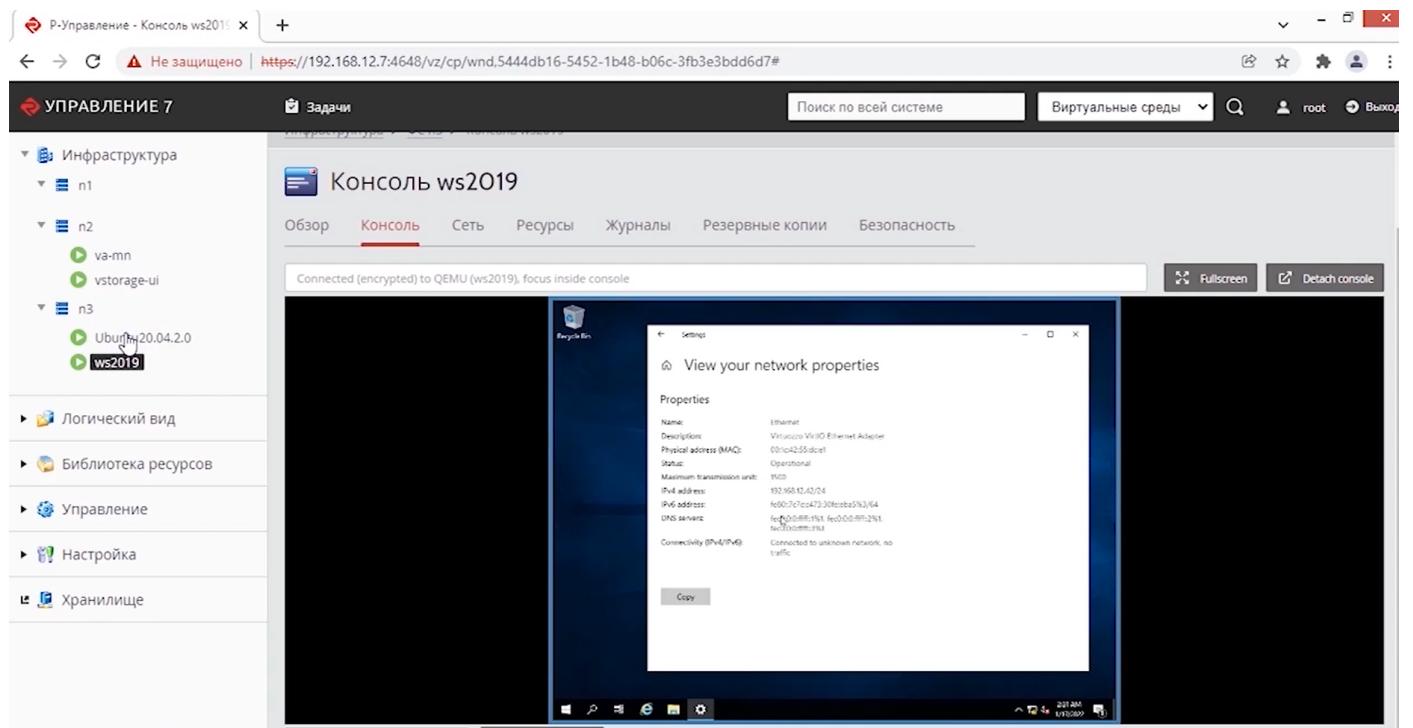


Скриншот 79.

Видно, что первая надо теперь пустая, там ничего не отображается, и те две виртуальные машины, которые мигрировали на третью ноду, теперь отображаются как раз на 3 ноде.

*Скриншот 80.*

Видно в консоли, что все, как было, так и осталось на обоих виртуальных машинах, все сохранено в таком состоянии, в каком было до запуска процесса миграции.



Скриншот 81.

Кроме этого, можем проверить сетевую доступность после миграции.

3.26 проверка восстановления работы виртуальных сред при аварийном выходе из строя сервера

Далее желательно проверить восстановление работы виртуальных сред в случае аварийного выхода из строя ноды. Чтобы сымитировать эту ситуацию, будет выключаться сервер по питанию. Выключаться будет третья нода – **н3**. Для наглядности будут бесконечные пинги до этой ноды.

The screenshot shows a terminal window with the following text:

```
Power Switch Virtual Drives Keyboard Help

Dear R-virtualization user!

vzkernel: 3.10.0-1062.12.1.rv7.131.10.1

Use the following hostname and IP address to connect to this server:

n3
(IP: 192.168.22.6, 192.168.12.6, 10.37.130.2, fdb2:2c26:f4e4::1)

17:48:34 Tue Jan 11 2022

n3 login: _
```

A cursor arrow is visible at the top center of the terminal window.

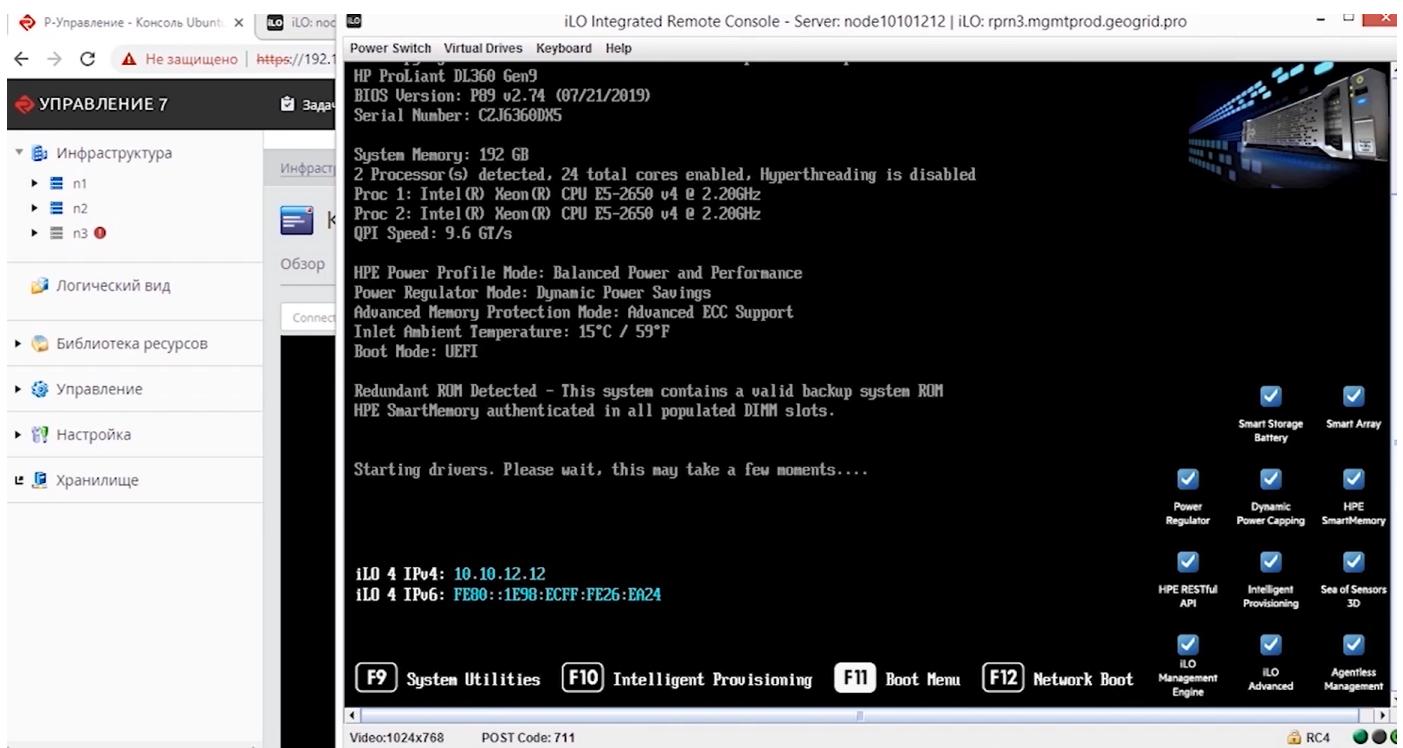
Скриншот 82.

Откроем удаленную консоль через **iLO**(в зависимости от производителя сервера это может быть **IPMI**, **BMC** и т.д.). Для того, чтобы сымитировать подобную ситуацию, будет выполнена нештатная перезагрузка сервера через **iLO** по питанию.

The screenshot shows the iLO 4 interface for a ProLiant DL360 Gen9 server. The left sidebar has a tree view with nodes like System Information, iLO Event Log, Integrated Management Log, Active Health System Log, Diagnostics, Location Discovery Services, Insight Agent, iLO Federation, Remote Console, Virtual Media, Power Management, and Server Power (which is selected). The main panel is titled "Server Power" and contains sections for "Virtual Power Button" and "System Power Restore Settings". Under "Virtual Power Button", it shows "System Power: ON" and four options: "Graceful Power Off" (Momentary Press), "Force Power Off" (Press and Hold), "Force System Reset" (Reset), and "Force Power Cycle" (Cold Boot). Under "System Power Restore Settings", it shows "Auto Power-On" with "Restore Last Power State" selected, and "Power-On Delay" with "Minimum Delay" selected.

Скриншот 83.

Необходимо перейти во вкладку управления питанием и выполнить принудительную перезагрузку.



Скриншот 84.

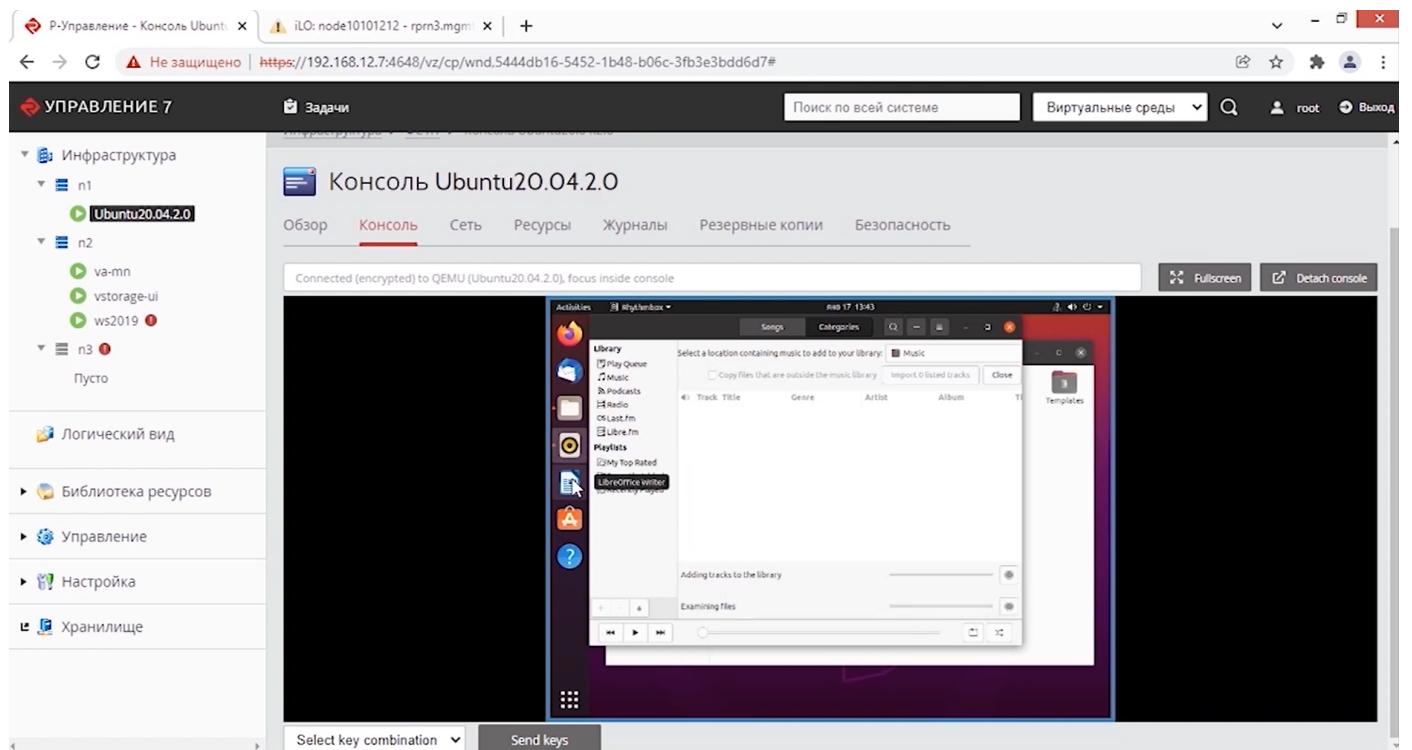
В консоли удаленного доступа видно, что пошла перезагрузка. Во время загрузки можно выбрать опцию старта **Boot Menu**, чтобы нода точно не загрузилась, и точно осталось состояние отсутствия доступа.

Важно!

В случае если сервер по какой-либо причине не имеет функции немедленного выключения питания или перезагрузки, или несмотря на то, что имеются кнопки, но все равно выполняется корректное выключение, то можно выполнить имитацию аварийного выключения с помощью следующей команды:

```
#echo c > /proc/sysrq-trigger
```

после выполнения этой команды сервер уйдёт в аварийную перезагрузку и при необходимости для того, чтобы сервер после перезагрузки не вернулся обратно можно через консоль сервера (**BMC\iLO\IPMI**) выключить его.



Скриншот 85.

В панели управления “Р-виртуализации” видно, что третья нода начала отображаться как недоступная, и выполнилась эвакуация виртуальных машин с третьей ноды. **Windows Server 2019** переехал на вторую ноду, а **ubuntu** на первую ноду, и после эвакуации виртуальные машины работают штатно и полностью доступны.

Можно также проверить статус службы **HA** с любого рабочего сервера следующей командой:

```
#shaman stat
```

Важно!

Подробнее ознакомится со статусами и сообщениями кластера “Р-хранилища” во время выхода из строя сервера можно из следующей документации:

1. Документация **GUI** https://updates.rosplatforma.ru/docs/7.0.13/r-stor_install_gui.pdf страница 39.
2. Документация **CLI** https://updates.rosplatforma.ru/docs/7.0.13/r-stor_install_cli.pdf страница 84.
3. Документация (**shaman**) https://updates.rosplatforma.ru/docs/7.0.13/r-virt_cli.pdf страница 144.

3.27 настройка сети хранилища для обеспечения максимальной производительности

Следующая важная настройка, которую желательно сделать на всех узлах кластера, для обеспечения максимальной производительности и отказоустойчивости - это установить на бонде сети хранилища параметр **xmit_hash_policy** со значением **layer=3+4**, а также задать значение **mtu 9000** как на самом бонде, так и на всех интерфейсах, которые в него входят(эти режимы должны поддерживаться на стороне физических коммутаторов и заранее настроены).

```
root@n3:~ [root@n3 ~]# nmcli c s sb1 | grep bond.options
bond.options:                                downdelay=0,miimon=1,mode=802.3ad,updelay=0
[root@n3 ~]# cat /proc/net/bonding/sb1 | grep Transmit
Transmit Hash Policy: layer2 (0)
[root@n3 ~]#
```

Скриншот 86.

Текущее значение **hash policy** можно посмотреть, как на скриншоте 86 двумя командами:

```
#mncli c s sb1 | grep bond.options
#cat /proc/net/bonding/sb1 | grep Transmit
```

Sb1 – это имя интерфейса для сети “**Р-хранилища**”. По умолчанию значение **hash policy – layer2**.

```
root@n3:~#
[root@n3 ~]# nmcli c s sb1 | grep bond.options
bond.options:                      downdelay=0,miimon=1,mode=802.3ad,updelay=0
[root@n3 ~]# cat /proc/net/bonding/sb1 | grep Transmit
transmit Hash Policy: layer2 (0)
[root@n3 ~]# ip a | grep sb1
6: eno49: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 1500 qdisc mq master sb1 state UP group default qlen 1000
7: eno50: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 1500 qdisc mq master sb1 state UP group default qlen 1000
23: sb1: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    inet 192.168.22.6/24 brd 192.168.22.255 scope global noprefixroute sb1
[root@n3 ~]#
```

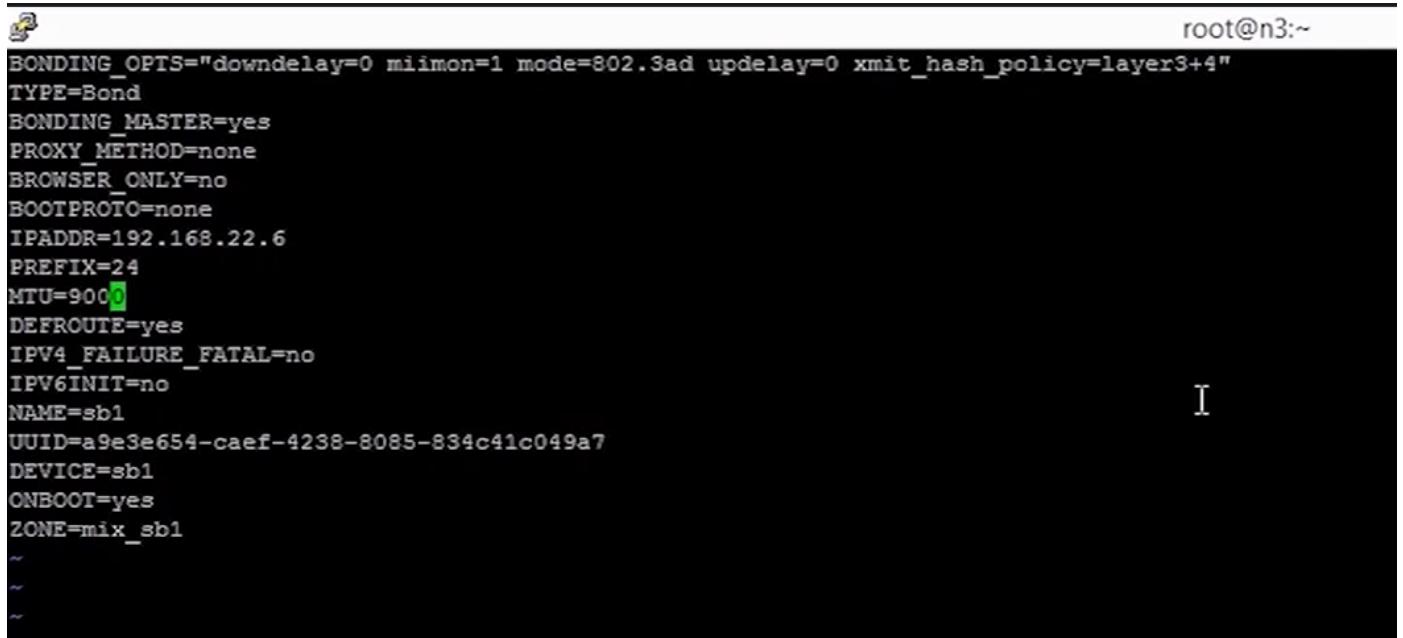
Скриншот 87.

Далее необходимо проверить текущее значение **mtu** на бонде и на входящих в него интерфейсах как на скриншоте 87. Текущее значение **mtu 1500**. Команда для проверки:

```
#ip a | grep sb1
```

Чтобы изменить значение этих параметров, надо открыть конфигурационный файл бонда сети “**Р-хранилища**”. Для этого можно выполнить следующую команду:

```
#vi /etc/sysconfig/network-scripts/ifcfg-sb1
```



```

root@n3:~
BONDING_OPTS="downdelay=0 miimon=1 mode=802.3ad updelay=0 xmit_hash_policy=layer3+4"
TYPE=Bond
BONDING_MASTER=yes
PROXY_METHOD=none
BROWSER_ONLY=no
BOOTPROTO=none
IPADDR=192.168.22.6
PREFIX=24
MTU=9000
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV6INIT=no
NAME=sb1
UUID=a9e3e654-caef-4238-8085-834c41c049a7
DEVICE=sb1
ONBOOT=yes
ZONE=mix_sb1
-
-
-

```

Скриншот 88.

Параметр **hash policy** указывается в графе **bonding_opts**, укажем его в самом конце. Важно не допустить синтаксических ошибок. Параметр **mtu** и его значение указываются в этом же файле отдельный графикой, укажем его сразу после графы с префиксом.

Значение параметра **MTU** применяется к самому бонду и ко всем входящим в него интерфейсам.

Важно!

Значение **MTU** на интерфейсах сети “Р-хранилища” можно также настраивать на этапе установки ОС гипервизора в инсталляторе, где была настройка **sb1** и его **slave** интерфейсов или как в нашем случае после установки ОС.

Далее сохраним файл. Если сразу после редактирования конфигурационного файла проверить текущее значение параметров, то будет видно, что изменения не применились.

Чтобы изменения вступили в силу, надо выполнить перезагрузку бонда сети хранилища. Для этого выключим его и после небольшой паузы в 15-20 секунд снова включим.

Выполнив следующие команды:

```
# ifdown sb1
# ifup sb1
```

```
[root@n3 ~]# ifdown sb1
Device 'sb1' successfully disconnected.
[root@n3 ~]# ifup sb1
Connection successfully activated (master waiting for slaves) (D-Bus active path: /org/freedesktop/NetworkManager/ActiveConnection/26)
[root@n3 ~]#
```

Скриншот 89.

Пример выполнения команд выключения и включения интерфейса для применения настроек.

```
[root@n3 ~]# nmcli c s sb1 | grep bond.options
bond.options:                                downdelay=0,miimon=1,mode=802.3ad,updelay=0,xmit_hash_policy=layer3+4
[root@n3 ~]# cat /proc/net/bonding/sb1 | grep Transmit
Transmit Hash Policy: layer3+4 (1)
[root@n3 ~]# ip a | grep sb1
6: eno49: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 9000 qdisc mq master sb1 state UP group default qlen 1000
7: eno50: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 9000 qdisc mq master sb1 state UP group default qlen 1000
24: sb1: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 9000 qdisc noqueue state UP group default qlen 1000
    inet 192.168.22.6/24 brd 192.168.22.255 scope global noprefixroute sb1
[root@n3 ~]#
```

Скриншот 90.

Проверим значение наших параметров еще раз. Теперь настройки применились и вступили в силу.

Как видно на скриншоте 90 параметры **MTU** заданы не только на интерфейсе **sb1**, но и на его **slave** интерфейсах, в конфигах которых также необходимо прописывать **MTU**.

На этом основная настройка кластера завершена.

 **Важно!**

В случае других сценариев и конфигурации использования Росплатформы таких как: нестандартное сетевое окружение; на каждую ноду большое количество дисков; растянутый кластер, дезагрегированный сценарий, **s3** георепликация и т.д. необходимо выполнять дополнительные настройки такие как особая настройка: **firewall**, настройка локаций и доменов отказа, включение дополнительной отказоустойчивости **cfgd**, сетевая балансировка и т.д. Все это выполняется согласно документации <https://docs.rosplatforma.ru/> или при консультации с support@rosplatforma.ru

4. Проверка корректности настроек кластера и работоспособности серверного оборудования.

Для этого потребуется **fio** установленный на всех нодах, а также профили для него.

Установка fio

1) Для начала необходимо установить **fio** на всех нодах кластера:

```
# yum install fio -y
```

Создание директории

2) Создаем директорию в “ **Р-хранилище**”, которая будет использоваться для тестов (стандартный путь в профилях - /mnt/vstorage/benchmark_dir)

```
# mkdir /mnt/vstorage/benchmark_dir
# vstorage -c <наименование кластера> set-attr -R /mnt/vstorage/
benchmark_dir replicas=2 tier=0
```

С помощью этой команды мы устанавливаем какой уровень хранилища и тип репликации данных нужно протестировать. **Tier** - задает уровень. Для указания

репликации используется два варианта - **replicas** для реплики. **encoding** для кодирования. Есть следующие возможные варианты:

- **replicas=1**
- **replicas=2**
- **replicas=3**
- **encoding=1+0**
- **encoding=1+2**
- **encoding=3+2**
- **encoding=5+2**
- **encoding=7+2**
- **encoding=17+3**

Для проверки кодирования 3+2 на дисках 0 уровня, нужно будет запустить следующую команду:

```
# vstorage -c <наименование_кластера> -R /mnt/vstorage/benchmanr_dir
encoding=3+2 tier=0
```

Загрузка профилей

3) Скачиваем профили **fio**, и сохраняем их на ноде, с которой будет запущен тест **fio**. Ссылка на профили будет в конце письма.

Запуск сервера **fio**

4) Запускаем **fio** сервер на всех нодах кластера

```
# fio --server
```

Будет запущен сервер **fio**, который будет использовать стандартный порт **8765**, поэтому так же необходимо открыть доступ к нему по сети, например с помощью **iptables**.

```
# iptables -I INPUT -p tcp -m tcp -m multiport --dports 8765 -j ACCEPT
```

Редактирование профилей

5) Отредактировать все файлы профилей согласно указанным инструкциям внутри файлов. **size** - должен быть равен двукратному размеру оперативной памяти на ноде. **numjobs** - количество ядер процессора. В случае если включен **Hyper Threading**, то указываем количество потоков.

Создание datasets

6) Создаем **dataset**'ы. Команду запускаем из папки, где находятся профили **fio**.

```
#for N in {<node1>,<node2>,<node3>,...,<nodeN>} ; do fio --client=$N  
prepare-set.fio;done
```

Выполнения команды может занять несколько минут. На этом подготовительный этап завершен. Для запуска теста необходимо запустить следующую команду:

```
# fio --client=<node1> seqread.fio --client=<node2> seqread.fio [...]
```

Количество клиентов равно количеству нод в кластере, вместо значения **nodeN** используем **IP** адрес ноды.

- **seqread.fio** - имя профиля. (1)
- **seqread.fio** - последовательное чтение.
- **randread.fio** - случайное чтение.
- **seqwrite.fio** - последовательная запись
- **randwrite.fio** - случайная запись
- **randrw.fio** - случайные чтение и запись
- **expand.fio** - последовательная запись в постоянно растущий файл, например, файлы резервных копий, базы данных и т.д.

1. 📈 вид теста

[Ссылка для скачивания профилей](#)

 **Важно!**

Если узлы кластера имеют кэши **SSD/NVMe**, убедитесь, что они очищены перед запуском любого профиля. Проверить кэши или журналы записи можно следующим образом: 1. На узле кластера запустите

```
# vstorage -c <наименование_кластера> top
```

2. Находясь на текстовой панели инструментов, нажмите на клавиатуре «**C**», чтобы развернуть вкладку блоков, а затем прокрутите столбцы с помощью клавиши «**i**». пока не увидите столбцы **JRN_FULL** и **s**. 3. Подождите, пока **JRN_FULL** не станет равным **0%**, а каждый идентификатор **CS** будет помечен флагом **c**, например, «**JCc**». Это может занять некоторое время, когда кластер не находится под нагрузкой ввода-вывода.

По завершению теста **fio** отобразит результаты выполнения команды бенчмаркинга. Чем больше узлов участвует в teste, тем больше результатов выдаст **fio**. Тем не менее, резюме будет предоставлено в конце, обозначенном слова «**All clients**».

Пример:

```
All clients: (groupid=0, jobs=80): err= 0: pid=0: Mon Jan 28 19:18:23 2019
read: IOPS=198k, BW=772Mi (809M)(45.3GiB/60078msec)
  slat (usec): min=2, max=1510, avg= 6.77, stdev= 3.29
  clat (nsec): min=1567, max=287148k, avg=3787988.85, stdev=6333559.68
  lat (usec): min=50, max=287152, avg=3794.83, stdev=6333.46
  bw ( KiB/s): min= 2048, max=113536, per=0.08%, avg=9880.44, stdev=4052.50, samples=9600
  iops : min= 512, max=28384, avg=2470.06, stdev=1013.12, samples=9600
write: IOPS=84.8k, BW=331Mi (347M)(19.4GiB/60078msec)
  slat (usec): min=2, max=1055, avg= 7.21, stdev= 3.33
  clat (usec): min=138, max=498280, avg=21347.45, stdev=40779.33
  lat (usec): min=144, max=498296, avg=21354.72, stdev=40779.01
  bw ( KiB/s): min= 968, max=49824, per=0.08%, avg=4240.17, stdev=1738.29, samples=9600
  iops : min= 242, max=12456, avg=1059.99, stdev=434.58, samples=9600
  lat (usec) : 2=0.01%, 20=0.01%, 50=0.01%, 100=0.71%, 250=9.39%
  lat (usec) : 500=7.81%, 750=3.96%, 1000=3.80%
  lat (msec) : 2=12.40%, 4=20.13%, 10=24.61%, 20=9.37%, 50=4.52%
  lat (msec) : 100=1.43%, 250=1.70%, 500=0.17%
  cpu : usr=0.68%, sys=3.44%, ctx=15563500, majf=0, minf=491
IO depths : 1=0.1%, 2=0.1%, 4=0.1%, 8=0.1%, 16=0.1%, 32=100.0%, >=64=0.0%
  submit : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.0%, 64=0.0%, >=64=0.0%
  complete : 0=0.0%, 4=100.0%, 8=0.0%, 16=0.0%, 32=0.1%, 64=0.0%, >=64=0.0%
  issued rwt: total=11866050,5093585,0, short=0,0,0, dropped=0,0,0
```

Это пример выходных данных задания **randrw.fio** (случайное чтение/запись), выполняемого на кластере **all-flash** из пяти узлов.

Показатели

Основными метриками для случайного чтения/записи являются **IOPS** (операции чтения записи в секунду) и пропускная способность (**bw**):

```
read: IOPS=198k, BW=772Mi (809M)(45.3GiB/60078msec)
write: IOPS=84.8k, BW=331Mi (347M)(19.4GiB/60078msec)
```

Обратите внимание на **IOPS** при тестировании случайных рабочих нагрузок и наблюдайте за пропускной способностью при тестировании последовательных операций. При тестировании случайных операций чтения/записи также обратите внимание на параметр **iodepth**, который указывает сколько запросов поставлено в очередь и сколько ожидает.

Кроме того:

- **slat** – это задержка отправки, которая указывает, сколько времени потребовалось для отправки ввода-вывода ядру.
- **clat** – задержка завершения, показывающая, сколько времени прошло между отправкой в ядро и завершение ввода-вывода (исключая задержку отправки).
- **lat** – это сумма **slat** и **clat**.

Важно!

Результаты показаний особенно профиля последовательной записи приблизительно должны совпадать с суммой скорости записи всех дисков с ролью хранения или с ролью кэш минус расходы на реплику (задержка в сети хранилища) для одной ноды. Если результаты сильно разнятся необходимо проверять на корректность настроек кластера, а также корректность работы серверного оборудования. В случае если затрудняетесь проверить самостоятельно, то для такой проверки лучше обратится в поддержку Росплатформы support@rosplatforma.ru.

Параметры профилей

В следующих разделах перечислены профили, которые доступны для упрощения сравнительного анализа. Каждый профиль содержит ряд параметров (на примере **randread.fio**):

Сценарий **fio**. В данном случае случайные чтения:

```
rw=randread
```

Размер набора данных. Должен быть как минимум в два раза больше оперативной памяти:

```
size=8g
```

Количество параллельных задач. Должно быть равно количеству ядер процессора:

```
numjobs=16
```

Размер блока данных:

```
bs=4k
```

Асинхронная библиотека. Требуется параметр «**direct=1**»:

```
ioengine=libaio
```

Глубина очереди для асинхронных запросов:

```
iodepth=32
```

Ограничивает работу сценария **fio** по времени, а не по выполнению сценарий определенное количество раз. Сценарий будет выполнен столько раз, сколько успеет за определенный «**runtime**»:

```
time_based
```

Длительность сценария (по умолчанию в секундах):

```
runtime=60
```

Этот параметр заставляет данные записываться непосредственно на диск без кэширования и позволяет измерять фактическую производительность записи на диск:

```
direct=1
```

Формат имени файла. В данном случае это зависит от количества задач:

```
filename_format=__testfile'${jobnum}'
```

Тип задачи. Если не установлено, вместо потоков будут создаваться процессы:

```
thread
```

Целевой каталог для набора данных:

```
directory=/mnt/vstorage/benchmark_dir
```

Пример профиля

Ниже пример профайла, где необходимо вставить размер памяти в соответствии с конфигурацией оборудования:

```
[seqwrite]
rw=write
size=2x<RAM>/<CPU_cores>
numjobs=<CPU_cores>
bs=1m
ioengine=libaio
iodepth=32
time_based
runtime=60
direct=1
filename_format=__testfile'$jobnum'
thread
directory=/mnt/vstorage/benchmark_dir
```

Укажите требуемый размер и значения **numjobs** в каждом профиле задания узла:

- размер как минимум в два раза больше оперативной памяти узла, разделенного на количество ядер процессора узла,
- numjobs** – количество ядер ЦП узла.

✓ Примечание:

Если **Hyper-Threading** включен, вместо этого используйте число потоков ЦП. Цель – полностью загрузить ЦП, но не под 100%.

5. 5.Архитектура
