

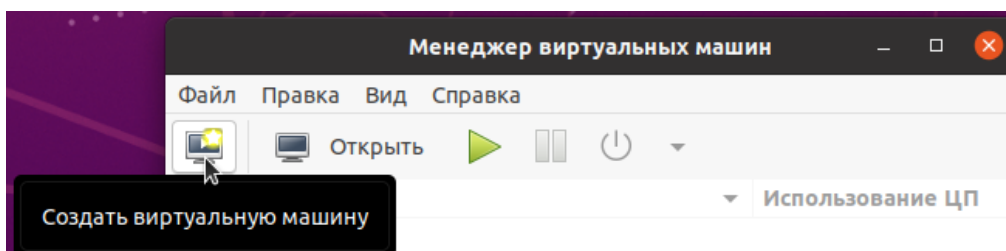
Курс «Системы виртуализации».

Методические указания по выполнению домашнего задания

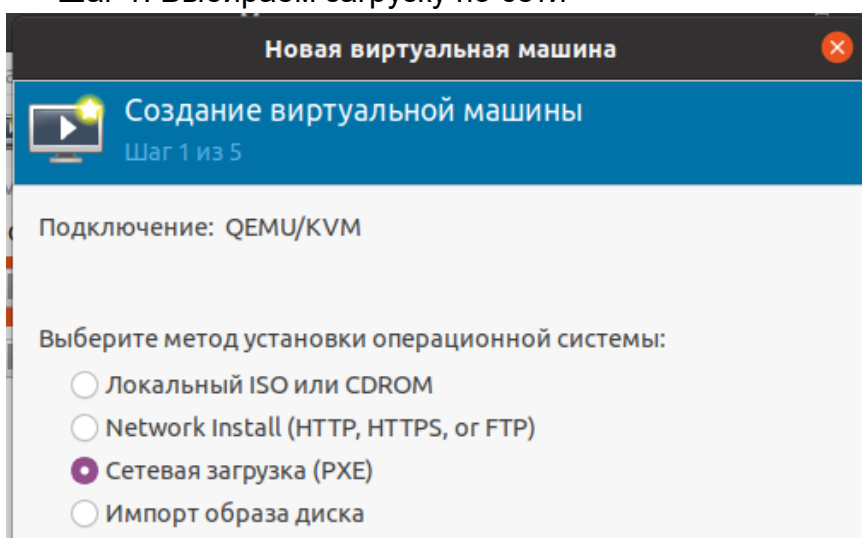
«Установка sheepdog»

1. Создание и настройка виртуальной машины (VM).

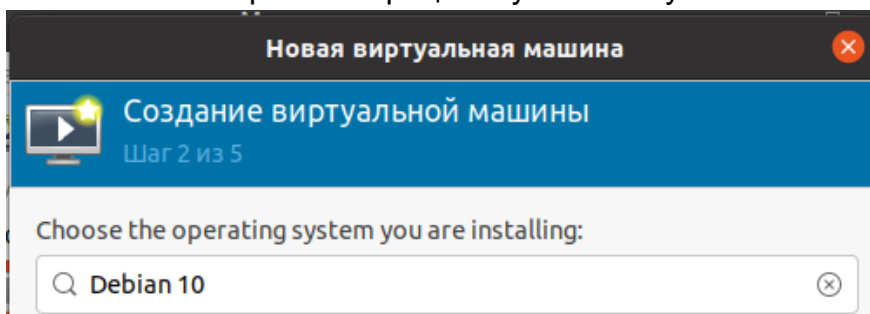
Создаем новую VM:



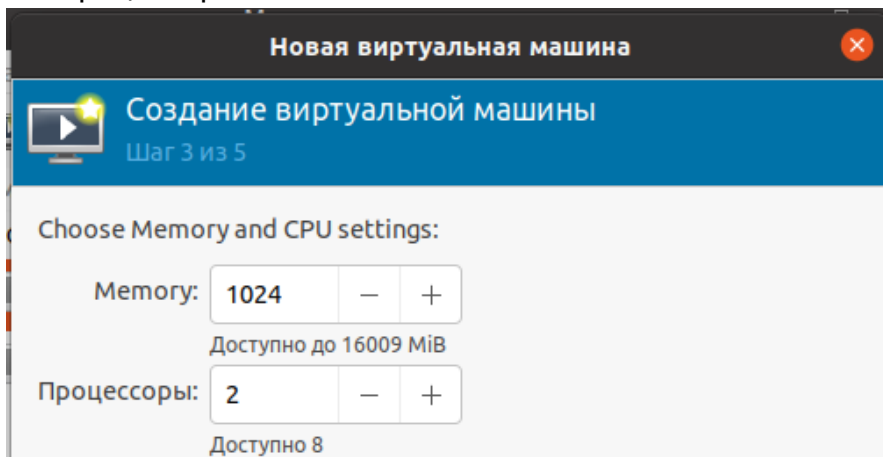
Шаг 1. Выбираем загрузку по сети



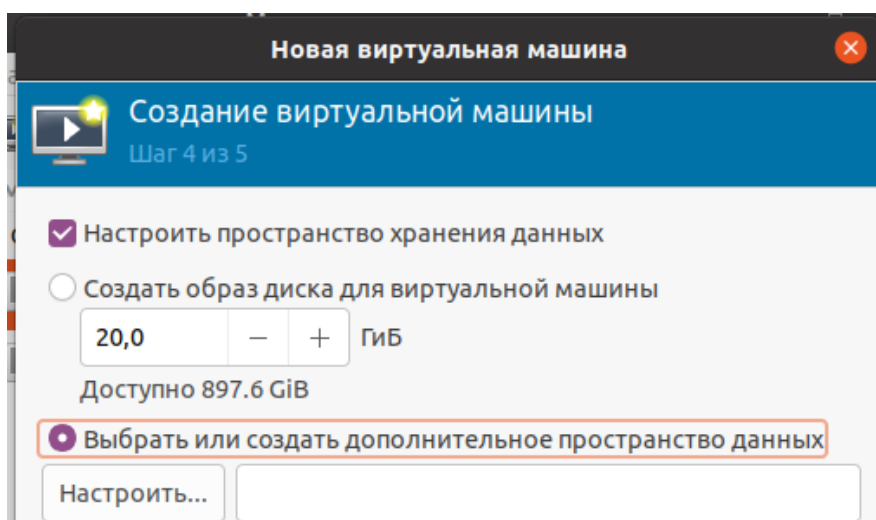
Шаг 2. Выбираем операционную систему Debian



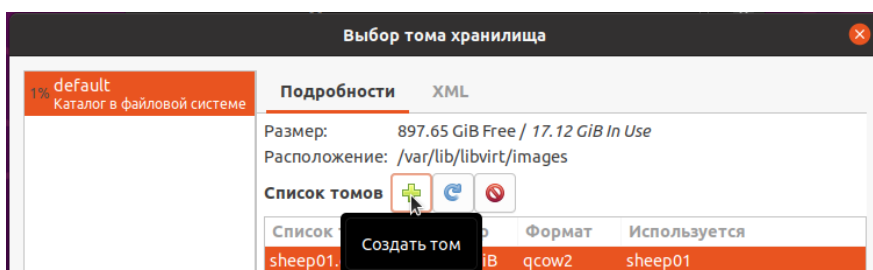
Шаг 3. Устанавливаем размер оперативной памяти и количество процессоров



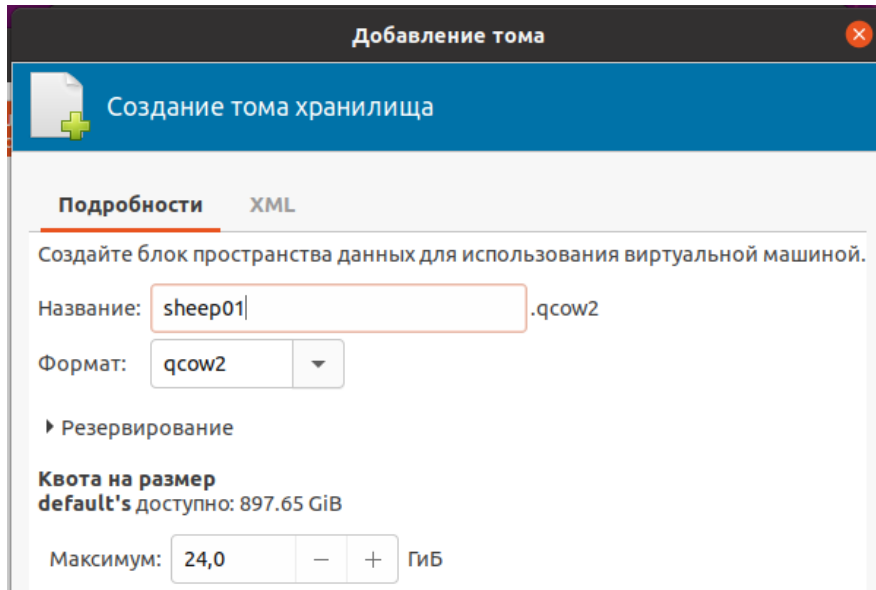
Шаг 4. Выбираем дополнительное пространство данных



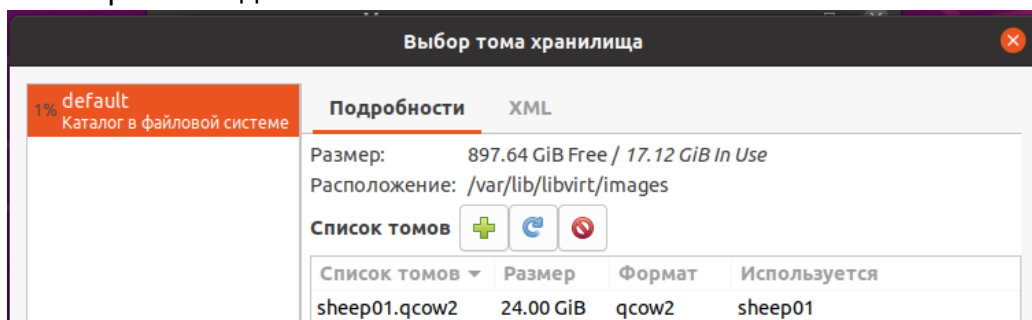
Создаем новый том



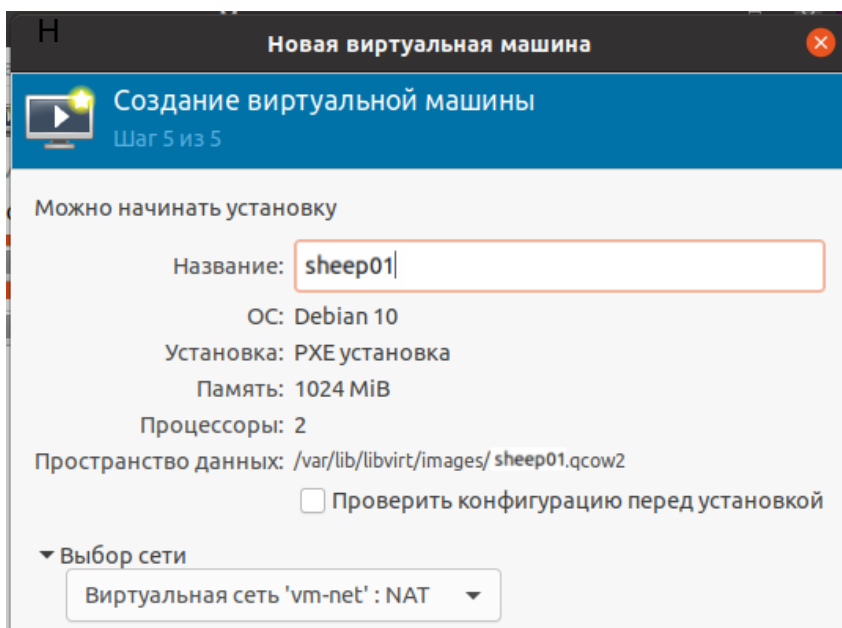
Указываем название: sheep01 и размер диска 24 ГБ.



Выбираем созданный выше том.



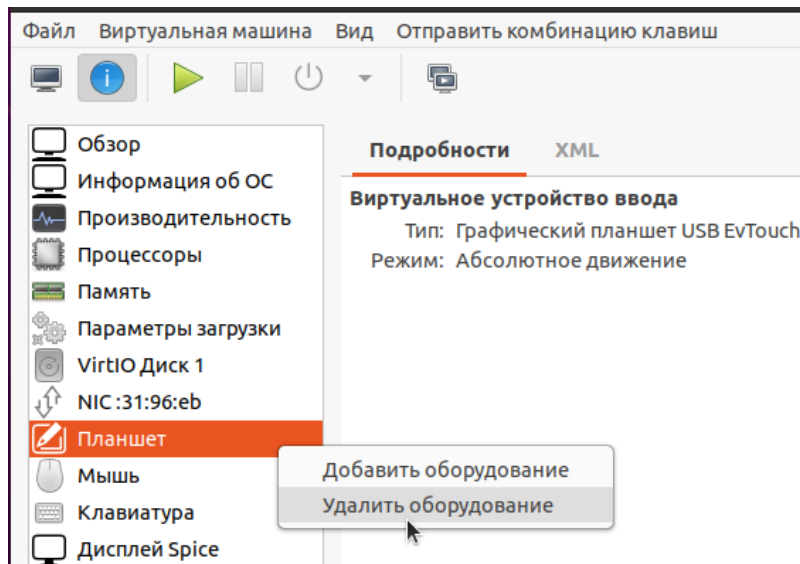
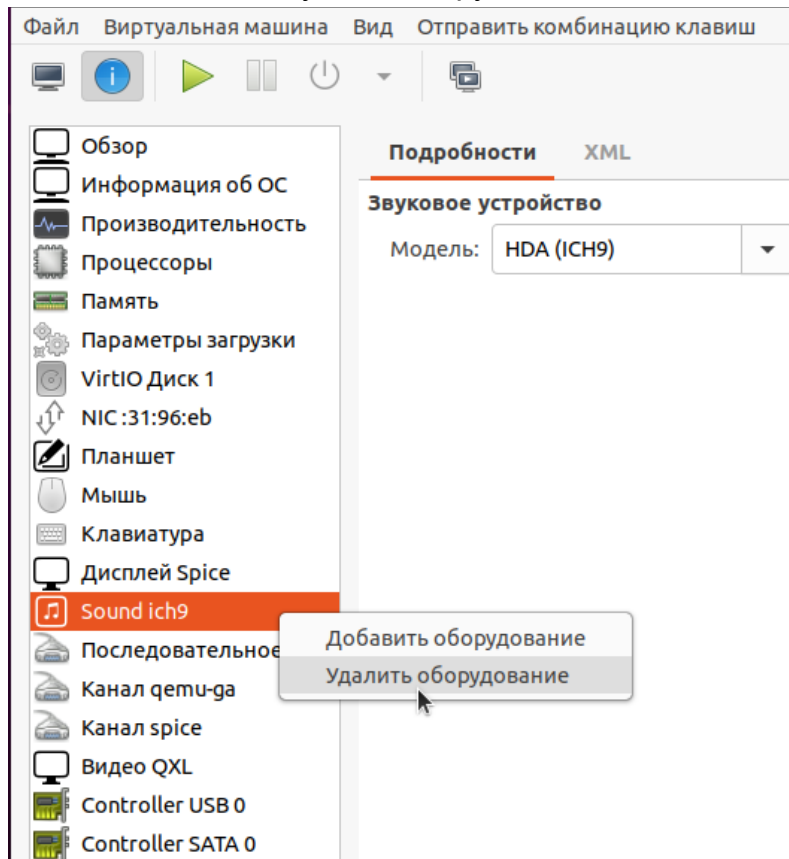
Шаг 5. Указываем название VM.



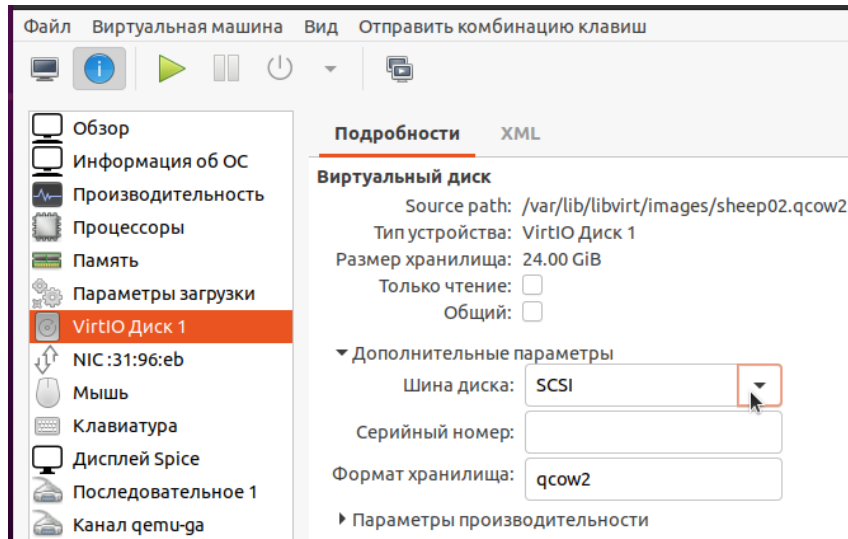
Настройка VM закончена. Далее идет автоматический запуск VM.
Выключаем ее принудительно.

Заходим в настройки VM.

Удаляем неиспользуемое оборудование.



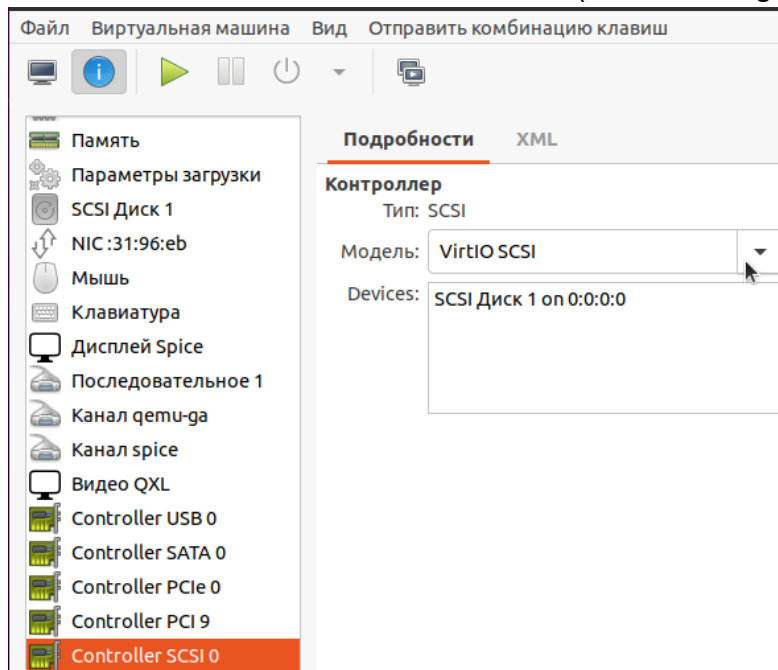
Для оборудования VirtIO Диск 1 устанавливаем шину SCSI.



Далее кнопка Применить.

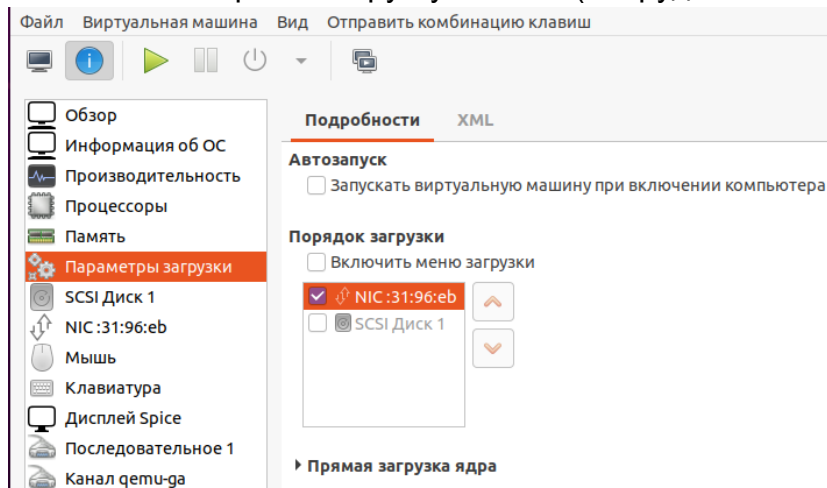
Появится новое оборудование Controller SCSI 0.

Здесь выставить Модель: VirtIO SCSI (вместо Isilogic).



Заходим в Параметры загрузки.

Выставляем первым загрузку по сети (оборудование NIC).



Запускаем VM.

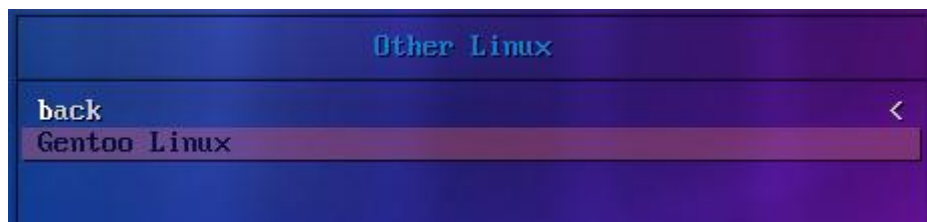
2. Установка операционной системы.

После запуска VM, по сети загрузится образ oVirt-toolsSetup.iso

Выбираем Other Linux



Затем Gentoo Linux



После загрузки ОС переходим к созданию разделов на диске.

Набираем в консоле: # `gdisk /dev/sda`

```
livecd ~ # gdisk /dev/sda
GPT fdisk (gdisk) version 1.0.3

Partition table scan:
  MBR: not present
  BSD: not present
  APM: not present
  GPT: not present

Creating new GPT entries.
```

Далее команда `o` – удаляем все разделы (если таковые были).

Команда `n` – создаем новый раздел:

- первый раздел (Enter, Enter, +1M, ef02)

```
Command (? for help): n
Partition number (1-128, default 1):
First sector (34-50331614, default = 2048) or {+-}size{KMGTP}:
Last sector (2048-50331614, default = 50331614) or {+-}size{KMGTP}: +1M
Current type is 'Linux filesystem'
Hex code or GUID (L to show codes, Enter = 8300): ef02
Changed type of partition to 'BIOS boot partition'
```

- второй (Enter, Enter, +512M, 8200)

```
Command (? for help): n
Partition number (2-128, default 2):
First sector (34-50331614, default = 4096) or {+-}size{KMGTP}:
Last sector (4096-50331614, default = 50331614) or {+-}size{KMGTP}: +512M
Current type is 'Linux filesystem'
Hex code or GUID (L to show codes, Enter = 8300): 8200
Changed type of partition to 'Linux swap'
```

- третий (Enter, Enter, +8G, Enter)

```
Command (? for help): n
Partition number (3-128, default 3):
First sector (34-50331614, default = 1052672) or {+-}size{KMGTP}:
Last sector (1052672-50331614, default = 50331614) or {+-}size{KMGTP}: +8G
Current type is 'Linux filesystem'
Hex code or GUID (L to show codes, Enter = 8300):
Changed type of partition to 'Linux filesystem'
```

- четвертый (Enter, Enter, Enter, Enter)

```
Command (? for help): n
Partition number (4-128, default 4):
First sector (34-50331614, default = 17829888) or {+-}size{KMGTP}:
Last sector (17829888-50331614, default = 50331614) or {+-}size{KMGTP}:
Current type is 'Linux filesystem'
Hex code or GUID (L to show codes, Enter = 8300):
Changed type of partition to 'Linux filesystem'
```

Далее записываем произведенные действия командой w.

```
Command (? for help): w
```

```
Final checks complete. About to write GPT data. THIS WILL OVERWRITE EXISTING PARTITIONS!!
```

```
Do you want to proceed? (Y/N): Y
```

```
OK: writing new GUID partition table (GPT) to /dev/sda.
```

```
The operation has completed successfully.
```

```
livecd ~ # reboot_
```

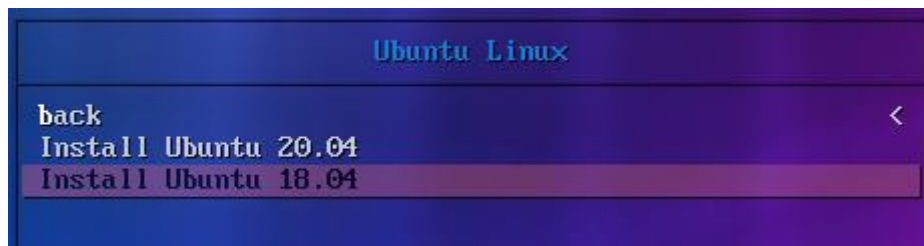
Подтверждаем – Y.

Перезагружаемся: `livecd ~ # reboot`

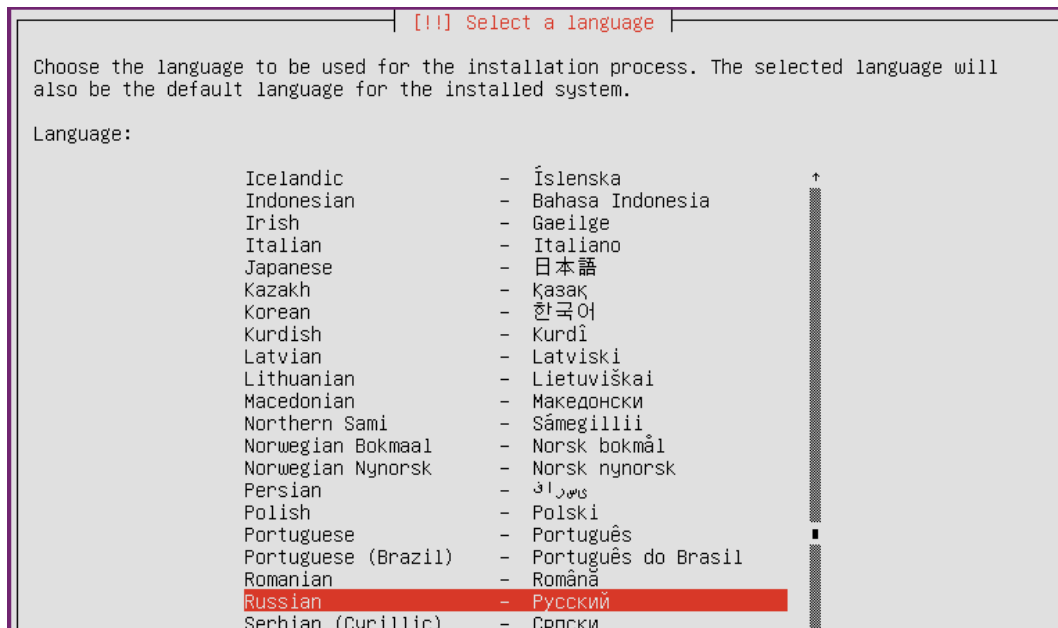
В меню загрузки выбираем Ubuntu Linux.



Затем Install Ubuntu 18.04



Устанавливаем Ubuntu 18.04



[!!] Выберите местонахождение

Выбранное местоположение будет учтено при настройке часового пояса и создании списка при выборе системной локали. Обычно, здесь указывается страна, в которой вы живёте.

Данный сокращённый список основан на выбранном вами языке. Выберите "другая", если вашего местоположения нет в списке.

Страна, область или регион:

Российская Федерация
Украина
другая

<Вернуться>

[!!] Настройка клавиатуры

You can try to have your keyboard layout detected by pressing a series of keys. If you do not want to do this, you will be able to select your keyboard layout from a list.

Detect keyboard layout?

<Вернуться>

<Да> <Нет>

[!!] Настройка клавиатуры

В каждой стране используется своя раскладка клавиатуры, а в некоторых странах используется несколько общих раскладок. Выберите страну, для которой произведена данная клавиатура компьютера.

Страна, для которой предназначена клавиатура:

Kazakh
Khmer (Cambodia)
Korean
Kyrgyz
Lao
Latvian
Lithuanian
Macedonian
Malay (Jawi, Arabic Keyboard)
Maltese
Maori
Moldavian
Mongolian
Montenegrin
Nepali
Norwegian
Persian
Polish
Portuguese
Portuguese (Brazil)
Romanian
Russian



[!!] Настройка клавиатуры

Выберите подходящую раскладку клавиатуры.

Раскладка клавиатуры:

Russian
Russian - Bashkirian
Russian - Chuvash
Russian - Chuvash (Latin)

[!] Настройка клавиатуры

Вам нужно указать способ переключения клавиатуры между национальной раскладкой и стандартной латинской раскладкой.

Наиболее эргономичным способом считаются правая клавиша Alt или Caps Lock (в последнем случае для переключения между заглавными и строчными буквами используется комбинация Shift+Caps Lock). Ещё одна популярная комбинация: Alt+Shift; заметим, что в этом случае комбинация Alt+Shift потеряет своё привычное действие в Emacs и других, использующих её, программах.

Не на всех клавиатурах есть перечисленные клавиши.

Способ переключения между национальной и латинской раскладкой:

Caps Lock	↑
правый Alt (AltGr)	
правый Control	
правый Shift	
правая клавиша с логотипом	
клавиша с меню	
Alt+Shift	
Control+Shift	
Control+Alt	
Alt+Caps Lock	

Вводим название компьютера: sheep01

[!] Настройка сети

Введите имя этого компьютера.

Имя компьютера -- это одно слово, которое идентифицирует вашу систему в сети. Если вы не знаете каким должно быть имя вашей системы, то посоветуйтесь с администратором вашей сети. Если вы устанавливаете вашу собственную домашнюю сеть, можете выбрать любое имя.

Имя компьютера:

sheep01

<Вернуться>

<Продолжить>

[!] Выбор зеркала архива Ubuntu

Выберите зеркало архива Ubuntu, расположенное в ближайшей к вам сети. Имейте в виду, что зеркало в ближайшей стране (или даже в вашей собственной) не всегда будет наилучшим выбором.

Страна, в которой расположено зеркало архива Ubuntu:

Остров Норфолк	↑
Остров Рождества	
Остров Святой Елены, Остров Вознесения и Тристан-да-Кунья	
Остров Херд и острова МакДональд	
Острова Кука	
Острова Туркс и Каикос	
Острова северной Марианы	
Пакистан	
Палау	
Палестина	
Панама	
Папуа – Новая Гвинея	
Парагвай	
Перу	
Питкэрн	
Польша	
Португалия	
Пуэрто-Рико	
Республика Корея	
Республика Македония	
Реюньон	
Российская Федерация	
	↓

[!] Выбор зеркала архива Ubuntu

Выберите зеркало архива Ubuntu. Если вы не знаете, с каким зеркалом у вас наилучшая связь, выберите находящееся в вашей стране или регионе.

Обычно <код вашей страны>.archive.ubuntu.com является хорошим выбором.

Зеркало архива Ubuntu:

ru.archive.ubuntu.com

<Вернуться>

Оставляем пустым:

[!] Выбор зеркала архива Ubuntu

Если вам необходимо использовать HTTP-прокси для доступа к внешнему миру, укажите в этом поле информацию о прокси. Если нет -- оставьте поле пустым.

Информацию о прокси следует вводить в стандартном виде
http://[пользователь][:пароль]@узел[:порт]/

Информация о HTTP-прокси (если прокси нет -- не заполняйте):

<Вернуться>

<Продолжить>

Вводим полное имя пользователя:

[!] Настройка учётных записей пользователей и паролей

Будет создана учётная запись пользователя, которая будет использоваться вместо учётной записи суперпользователя (root) для выполнения всех действий, не связанных с администрированием.

Введите реальное имя этого пользователя. Эта информация будет использована в письмах в поле "От кого", посылаемых этим пользователем, а также всеми программами, которые показывают или используют реальное имя пользователя в своей работе. Ваше имя и фамилия вполне подходят.

Введите полное имя нового пользователя:

Tech

<Вернуться>

<Продолжить>

И имя учётной записи:

[!] Настройка учётных записей пользователей и паролей

Выберите имя пользователя (учётную запись), под которым вы будете известны в системе. В качестве учётной записи может быть использовано ваше реальное имя. Учётная запись должна начинаться со строчной латинской буквы, за которой может следовать любое количество строчных латинских букв или цифр.

Имя вашей учётной записи:

tech

<Вернуться>

<Продолжить>

Вводим пароль для учетной записи:

[!!] Настройка учётных записей пользователей и паролей

Хороший пароль представляет из себя смесь букв, цифр и знаков препинания, и должен периодически меняться.

Введите пароль для нового пользователя:

Show Password in Clear

<Вернуться> <Продолжить>

И подтверждаем его.

[!!] Настройка учётных записей пользователей и паролей

Проверка правильности ввода осуществляется путём повторного ввода пароля и сравнения результатов.

Введите пароль ещё раз:

Показывать вводимый пароль

<Вернуться> <Продолжить>

[!] Настройка времени

Based on your present physical location, your time zone is Europe/Moscow.

If this is not correct, you may select from a full list of time zones instead.

Is this time zone correct?

<Вернуться> Да <Нет>

Разметку дисков делаем вручную:

[!!] Разметка дисков

Программа установки может провести вас через процесс разметки диска (предлагая разные стандартные схемы) на разделы, либо это можно сделать вручную. Если выбрать использование инструмента управления разметкой, у вас всё равно будет возможность позже посмотреть и подправить результат.

Если выбрать использование инструмента управления разметкой всего диска, то далее вас попросят указать нужный диск.

Метод разметки:

- Авто - использовать весь диск
- Авто - использовать наибольшее свободное место
- Авто - использовать весь диск и настроить LVM
- Авто - использовать весь диск с шифрованным LVM
- Вручную

<Вернуться>

Первый раздел не трогаем.
Настраиваем второй раздел:

[!!] Разметка дисков

Перед вами список настроенных разделов и их точек монтирования. Выберите раздел, чтобы изменить его настройки (тип файловой системы, точку монтирования и так далее), свободное место, чтобы создать новый раздел, или устройство, чтобы создать на нём новую таблицу разделов.

Автоматическая разметка
Настройка программного RAID
Настройка менеджера логических томов (LVM)
Настроить шифрование для томов
Настроить тома iSCSI

SCSI1 (0,0,0) (sda) - 25.8 GB QEMU QEMU HARDDISK

	1.0 MB		СВОБОДНОЕ МЕСТО	
#1	1.0 MB	K	biosgrub	BIOS boot pa
#2	536.9 MB			Linux swap
#3	8.6 GB			Linux filesy
#4	16.6 GB			Linux filesy

Отменить изменения разделов
Закончить разметку и записать изменения на диск

<Вернуться>

[!!] Разметка дисков

Вы изменяете раздел #2 на устройстве SCSI1 (0,0,0) (sda). На этом разделе не найдено файловых систем.

Настройки раздела:

Имя: Linux swap
Использовать как: не использовать

Метка 'загрузочный': выкл

Стирание данных на этом разделе
Удалить раздел
Настройка раздела закончена

<Вернуться>

[!!] Разметка дисков

Принцип применения этого раздела:

Журналируемая файловая система Ext4
Журналируемая файловая система Ext3
Файловая система Ext2
Журналируемая файловая система btrfs
Журналируемая файловая система JFS
Журналируемая файловая система XFS
Файловая система FAT16
Файловая система FAT32
раздел подкачки
Резервная загрузочная область BIOS
физический том для шифрования
физический том для RAID
физический том для LVM
не использовать раздел

<Вернуться>

[!!] Разметка дисков

Вы изменяете раздел #2 на устройстве SCSI1 (0,0,0) (sda). На этом разделе не найдено файловых систем.

Настройки раздела:

Имя: Linux swap
Использовать как: раздел подкачки

Метка 'загрузочный': выкл

Стирание данных на этом разделе

Удалить раздел

Настройка раздела закончена

<Вернуться>

Настраиваем третий раздел:

[!!] Разметка дисков

Перед вами список настроенных разделов и их точек монтирования. Выберите раздел, чтобы изменить его настройки (тип файловой системы, точку монтирования и так далее), свободное место, чтобы создать новый раздел, или устройство, чтобы создать на нём новую таблицу разделов.

Автоматическая разметка
Настройка программного RAID
Настройка менеджера логических томов (LVM)
Настроить шифрование для томов
Настроить тома iSCSI

SCSI1 (0,0,0) (sda) - 25.8 GB QEMU QEMU HARDDISK

	1.0 MB		СВОБОДНОЕ МЕСТО	
#1	1.0 MB	K	biosgrub	BIOS boot pa
#2	536.9 MB	f	подк	Linux swap подк
#3	8.6 GB			Linux filesy
#4	16.6 GB			Linux filesy

Отменить изменения разделов

Закончить разметку и записать изменения на диск

<Вернуться>

[!!] Разметка дисков

Вы изменяете раздел #3 на устройстве SCSI1 (0,0,0) (sda). На этом разделе не найдено файловых систем.

Настройки раздела:

Имя: Linux filesystem
Использовать как: не использовать

Метка 'загрузочный': выкл

Стирание данных на этом разделе

Удалить раздел

Настройка раздела закончена

<Вернуться>

[!!] Разметка дисков

Принцип применения этого раздела:

Журналируемая файловая система Ext4
Журналируемая файловая система Ext3
Файловая система Ext2
Журналируемая файловая система btrfs
Журналируемая файловая система JFS
Журналируемая файловая система XFS
Файловая система FAT16
Файловая система FAT32
раздел подкачки
Резервная загрузочная область BIOS
физический том для шифрования
физический том для RAID
физический том для LVM
не использовать раздел

<Вернуться>

[!!] Разметка дисков

Вы изменяете раздел #3 на устройстве SCSI1 (0,0,0) (sda). На этом разделе не найдено файловых систем.

Настройки раздела:

Имя: Linux filesystem
Использовать как: Журналируемая файловая система Ext4

Точка монтирования: отсутствует
Параметры монтирования: defaults
Метка: отсутствует
Зарезервированные блоки: 5%
Обычное использование: стандарт
Метка 'загрузочный': выкл

Стирание данных на этом разделе
Удалить раздел
Настройка раздела закончена

<Вернуться>

[!!] Разметка дисков

Точка монтирования этого раздела:

/ -- корневая файловая система (root file system)
/boot -- статические файлы системного загрузчика
/home -- домашние каталоги пользователей
/tmp -- временные файлы
/usr -- статичные данные
/var -- изменяемые данные
/srv -- данные служб, предоставляемых системой
/opt -- дополнительные пакеты программного обеспечения
/usr/local -- локальные каталоги
Ввести вручную
Не монтировать этот раздел

<Вернуться>

Вы изменяете раздел #3 на устройстве SCSI1 (0,0,0) (sda). На этом разделе не найдено файловых систем.

Настройки раздела:

```

Имя: Linux filesystem
Использовать как: Журналируемая файловая система Ext4

Точка монтирования: /
Параметры монтирования: defaults
Метка: отсутствует
Зарезервированные блоки: 5%
Обычное использование: стандарт
Метка 'загрузочный': выкл

Стирание данных на этом разделе
Удалить раздел
Настройка раздела закончена

```

<Вернуться>

В параметрах монтирования отмечаем noatime и nodirtime

Параметры монтирования позволяют подстроить поведение файловой системы.

Параметры монтирования:

```

[ ] discard -- обрезать освободившиеся блоки на нижележащем блочном устройстве
[*] noatime -- не обновлять время доступа к inode
[*] nodirtime -- не обновлять время доступа к inode каталога
[ ] relatime -- обновлять время доступа к inode при изменениях
[ ] nodev -- не поддерживать символьные и блочные устройства
[ ] nosuid -- игнорировать биты SUID и SGID
[ ] noexec -- запретить выполнение любых программ
[ ] ro -- монтировать файловую систему в режиме 'только для чтения'
[ ] sync -- включить синхронный ввод-вывод в файловой системе
[ ] usrquota -- разрешить учёт дисковых квот пользователей
[ ] grpquota -- разрешить учёт дисковых квот групп
[ ] user_xattr -- включить расширенные пользовательские атрибуты

```

<Вернуться>

<Продолжить>

Вы изменяете раздел #3 на устройстве SCSI1 (0,0,0) (sda). На этом разделе не найдено файловых систем.

Настройки раздела:

```

Имя: Linux filesystem
Использовать как: Журналируемая файловая система Ext4

Точка монтирования: /
Параметры монтирования: noatime,nodirtime
Метка: отсутствует
Зарезервированные блоки: 5%
Обычное использование: стандарт
Метка 'загрузочный': выкл

Стирание данных на этом разделе
Удалить раздел
Настройка раздела закончена

```

<Вернуться>

[!] Разметка дисков

Перед вами список настроенных разделов и их точек монтирования. Выберите раздел, чтобы изменить его настройки (тип файловой системы, точку монтирования и так далее), свободное место, чтобы создать новый раздел, или устройство, чтобы создать на нём новую таблицу разделов.

Автоматическая разметка
Настройка программного RAID
Настройка менеджера логических томов (LVM)
Настроить шифрование для томов
Настроить тома iSCSI

SCSI1 (0,0,0) (sda) - 25.8 GB QEMU QEMU HARDDISK

	1.0 MB		СВОБОДНОЕ МЕСТО		
#1	1.0 MB	K	biosgrub	BIOS boot pa	
#2	536.9 MB	f	подк	Linux swap	подк
#3	8.6 GB	f	ext4	Linux filesy	/
#4	16.6 GB			Linux filesy	

Отменить изменения разделов

Закончить разметку и записать изменения на диск

<Вернуться>

[!] Разметка дисков

Если вы продолжите, то изменения, перечисленные ниже, будут записаны на диски. Или же вы можете сделать все изменения вручную.

На этих устройствах изменены таблицы разделов:

SCSI1 (0,0,0) (sda)

Следующие разделы будут отформатированы:

раздел #2 на устройстве SCSI1 (0,0,0) (sda) как подк

раздел #3 на устройстве SCSI1 (0,0,0) (sda) как ext4

Записать изменения на диск?

<Да>

<Нет>

Подтверждаем изменения. Разметка дисков закончена.

[!] Настройка RAM

Регулярное обновление - важная часть поддержания вашей системы безопасной.

По умолчанию, обновления применяются вручную с использованием программ управления пакетами. Как вариант, вы можете указать системе автоматически скачивать и устанавливать обновления безопасности, или вы можете выбрать управление системой через интернет как часть группы систем с использованием Canonical сервиса Landscape.

Каким образом вы хотите управлять обновлением системы?

Без автоматического обновления

Устанавливать обновления безопасности автоматически

Управление системой с помощью Landscape

<Вернуться>

Отмечаем для установки OpenSSH server и Basic Ubuntu server:

[!] **Выбор программного обеспечения**

В данный момент, установлена только основа системы. Исходя из ваших потребностей, вы можете выбрать один и более из уже готовых наборов программного обеспечения.

Выберите устанавливаемое программное обеспечение:

```
[ ] Lubuntu minimal installation (Qt part) ↑
[ ] Lubuntu Qt Desktop (Qt part)
[ ] Mail server
[ ] PostgreSQL database
[ ] Print server
[ ] Samba file server
[ ] Ubuntu Budgie desktop
[ ] Ubuntu desktop
[ ] Ubuntu MATE minimal
[ ] Ubuntu MATE desktop
[ ] Audio recording and editing suite
[ ] Ubuntu Studio desktop
[ ] Ubuntu Studio minimal DE installation
[ ] Large selection of font packages
[ ] 2D/3D creation and editing suite
[ ] Photograph touchup and editing suite
[ ] Publishing applications
[ ] Video creation and editing suite
[ ] Vanilla GNOME desktop
[ ] Xubuntu minimal installation
[ ] Xubuntu desktop
[*] OpenSSH server
[*] Basic Ubuntu server ↓
```

[!] **Установка системного загрузчика GRUB на жёсткий диск**

Похоже, что данная система будет единственной на этом компьютере. Если это действительно так, то можно спокойно устанавливать системный загрузчик GRUB в основную загрузочную запись первого жёсткого диска.

Внимание! Если программе установки не удалось обнаружить другую операционную систему, имеющуюся на компьютере, то изменение основной загрузочной записи приведёт к тому, что эту операционную систему некоторое время нельзя будет загрузить. Позднее можно будет настроить GRUB для её загрузки.

Установить системный загрузчик GRUB в главную загрузочную запись?

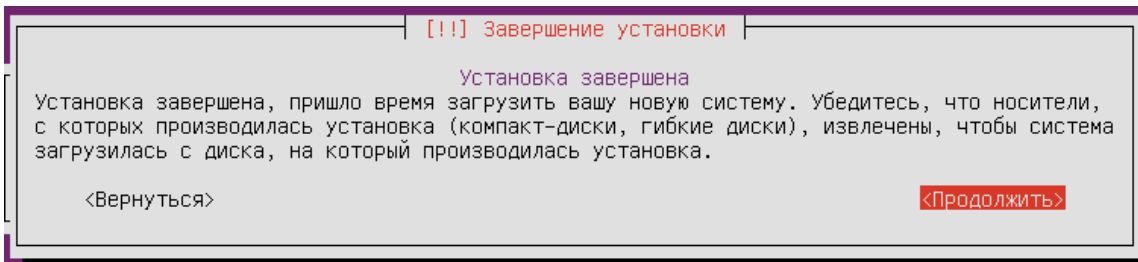
<Вернуться> Да <Нет>

[!] **Завершение установки**

Системные часы обычно показывают универсальное координированное время (UTC). Для преобразования времени системных часов в местное время операционная система использует настройку часового пояса. Если на машине не используются другие операционные системы, которым нужно, чтобы системные часы отражали местное время, рекомендуется настройка системных часов в UTC.

Системные часы показывают UTC?

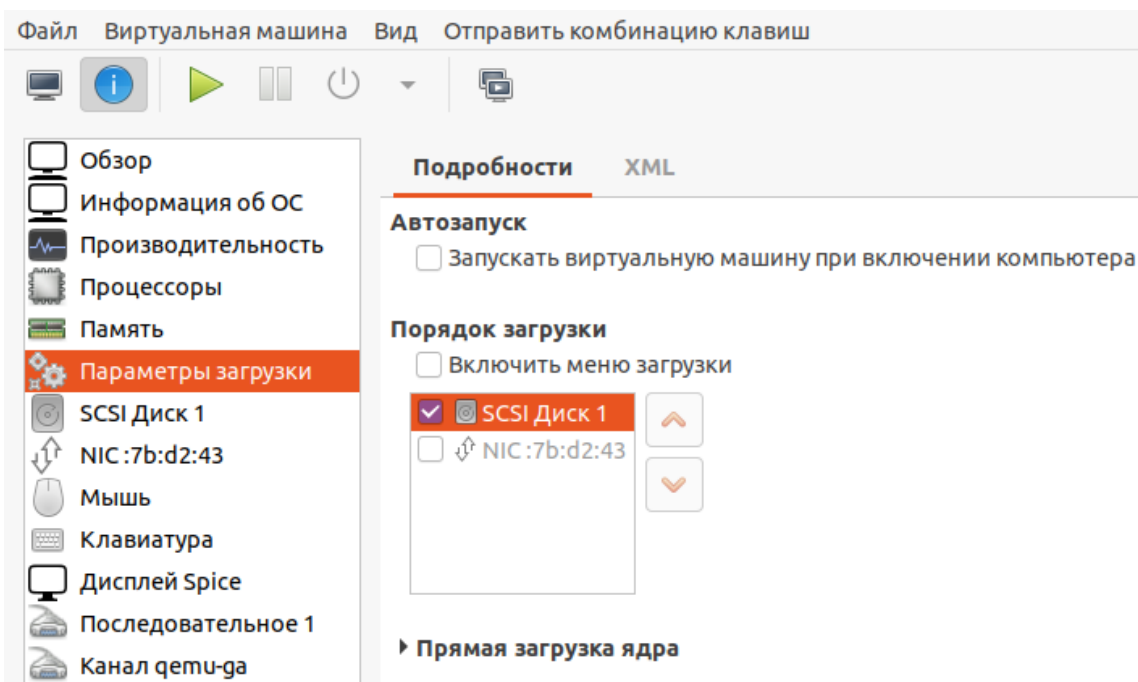
<Вернуться> Да <Нет>



Установка ОС завершена. После нажатия кнопки Продолжить, система перезагрузится.

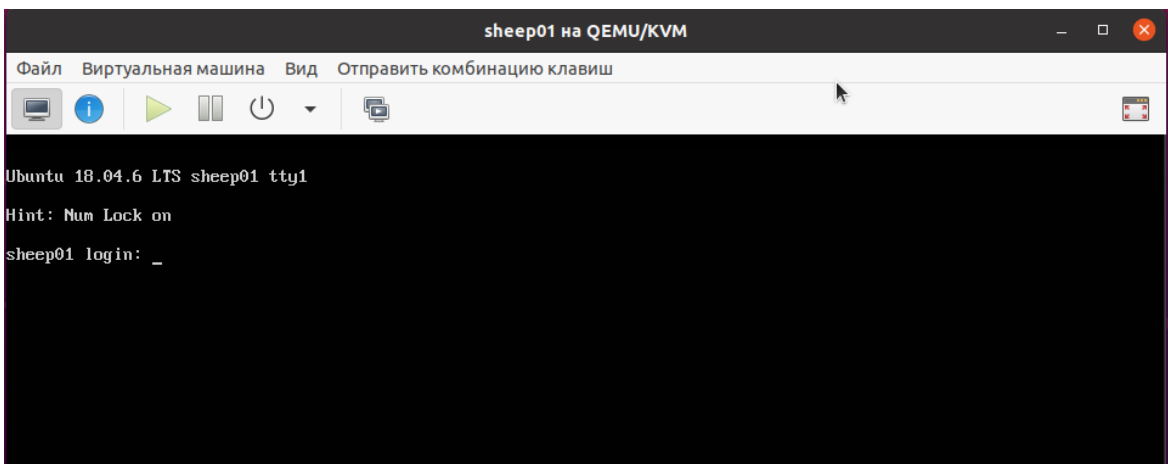
Принудительно выключаем VM. Заходим в настройки VM.

В Параметрах загрузки устанавливаем первым SCSI Диск 1.



Сохраняем изменения и запускаем VM.

После загрузки должна появиться консоль терминала:



Далее выключаем VM и создаем таким же образом еще три VM с именами sheep02, sheep03 и sheep04.

3. Установка и настройка сервисов.

Настроим сеть с помощью `netplan` на каждой VM:

```
root@sheep01:~# vim /etc/netplan/01-netcfg.yaml
```

```
# This file describes the network interfaces available on your system
# For more information, see netplan(5).
network:
  version: 2
  renderer: networkd
  ethernets:
    enp1s0:
      addresses: [10.100.10.11/26]
      gateway4: 10.100.10.1
      nameservers:
        addresses: [8.8.8.8, 8.8.4.4]
      dhcp4: no
```

```
root@sheep01:~# netplan generate
```

```
root@sheep01:~# netplan apply
```

Настроим беспарольный доступ на VM

На host-машине в терминале открываем четыре окна (Ctrl+Shift+T, по числу VM) и вводим следующие команды для каждой VM.

Для VM sheep01:

```
admin@host:~$ sudo su -
```

```
root@host:~# ssh -l tech 10.100.10.11
```

```
tech@sheep01:~$ sudo su -
```

Создаем пару ключей на каждой VM:

```
root@sheep01:~# ssh-keygen (Enter, Enter, Enter)
```

```
root@sheep01:~# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Created directory '/root/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:0U9nmNzmJbLSn4wrceRMeDIP/HVgeT6KaK7+DhAEf4Y root@sheep01
The key's randomart image is:
+---[RSA 2048]-----+
|
|  ..          +
| o .          + .
| E o... . +
| + o*=+. . +
| . . S&Oo.o .
| . o+*BB..
| .ooo+ +
| .o. .
| .o+o..
+----[SHA256]-----+
root@sheep01:~#
```

На каждой VM открываем файл id_rsa.pub

```
root@sheep01:~# cat /root/.ssh/id_rsa.pub
```

```
root@sheep01:~# cat /root/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDDiHi/ZkV94bplWxd0fW8Ag92DPPKt+c7unek1Hn70H6MVLr6S0Q6Znu1pYka
lwZwfbUOXVj/01Tm6+TPPrLdSHVKpxM/x8EQFYI3KJeLImDB09hPuKDVqbdExGbf1Zynyg3q1FQqmCHNMMnbi3k4u2xzI0GL8WP
jyK2log1y3RJC3V/E/Sg131DKkUeKrQll+jEpM240lvx8d240RvYh40Bxhxbxa3LMHTZncj3IRgTY2Aom0UR8b3sJyPuM4tbME
nJwF90p7RgxBNrDnSAppMhxn7zUwccqUbTKfknR00izLmLJoGqH/aQQ4ESLPa9xifKRd3IN4gP6Q5INH5Wh root@sheep01
```

С VM sheep01 подключаемся к host-машине и генерируем ключ:

```
root@sheep01:~# ssh 10.100.10.1
```

```
root@sheep01:~# ssh-keygen
```

На VM sheep01 создаем файл authorized_keys и добавляем в него ключи из остальных VM (sheep02, sheep03, sheep04), включая host-машину.

```
root@sheep01:~# cat > /root/.ssh/authorized_keys
```

```
root@sheep01:~# cat > ~/.ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDdiHi/ZkV94bplWxd0fW8Ag92DPPKt+c7unek1Hn70H6MVLr6S0Q6Znu1pYka
329UEMQY+DGqz5uWnHABD2Dks+N+poy+7ZYxE6zefBCgzKtdg6ob/kdnz+Mv9Za1+eHTSxTFsRCxLUyWZFLRjygAspdXj3CL6A
eagmSTVz1KprEGJUaxKi66VJ157cUI0ziarK6aPBAUsL9pGIVIn/4IspgTnIVskbGu7nrb40ZH4mA5iqjIeVlauS7s0FdL/Zd
k2gClc/p/Y5HhHAeegrtxnYe8dugRxBaTx+XUnkq21Dd8I2RwU/EuSu9PxlUmP57oKz/zGKPjzRDE/aUGerMj1VdK/ju4yN1qlm
5gX8t8wJS4hCGHxLEGQ5vztSV64PQ0H4z4E4CLIJ5HhPAKcc5UKhA/PuFwSxpCFPFWSuZHgvL4cDgrt8pIbpZ7mVzv7NWhXJjPn
rnaa6W0j0oTXpA0SEamXZaW0hC4pfavLcnCvm9isNwJBKCR9WLreucPk= anton@anton-X58A-UD3R
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDdiHi/ZkV94bplWxd0fW8Ag92DPPKt+c7unek1Hn70H6MVLr6S0Q6Znu1pYka
lwZwfbUOXVj/01Tm6+TPPrLdSHVKpxM/x8EQFYI3KJeLImDB09hPuKDVqbdExGbf1Zynyg3q1FQqmCHNMMnbi3k4u2xzI0GL8WP
jyK2log1y3RJC3V/E/Sg131DKkUeKrQll+jEpM240lvx8d240RvYh40Bxhxbxa3LMHTZncj3IRgTY2Aom0UR8b3sJyPuM4tbME
nJwF90p7RgxBNrDnSAppMhxn7zUwccqUbTKfknR00izLmLJoGqH/aQQ4ESLPa9xifKRd3IN4gP6Q5INH5Wh root@sheep01
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDxo321mD5MxQkmp2RGwENCsw0Q0cz/EhreZkCqkXEW8gDmDFpuC+bwsMcvL
g27mL+A610XnHud6mNXnCxysWP0+pujBlAFcPsMjAQ50235Ztqwt1dquq2DsBuZZbYaZHGHY7f5t9jgIwWax33/FVNN2gb/VaMz
Au4hvwe5VopIJHaOhrMn2uFgl+PEF7A85pK1IuOKSg7xVatnQicC7NxuLafLhM6GiD9+Y/Y6U6z6nCvrBU8Dqc+TgVyo2wccz11
Mj5dE/b+/LNgcS203r5P75TS47JNLyXoKtQkp/ZMYay0Di9x8DyQ0FRCUW3cprfE4upApDHCxxvBXMpA0P root@sheep02
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQCE/E+lCElp+L9KaaPBYaJxgHtSwpBVdBhQDzEhD6m8Ifm0q/y9IlecX0dFm3rG
6TnyLpkQUj34cG1DxFlczAe1lGEnzBT1haiMWAhid2/yHi60IeUohE2Jzc8j3+ehxFT/a0dXerB3mM6k0Gnlvzbda6SrmjV+YV
Q/hXb/uPX0WTeZdGqJ9vsEmqkDX4Qwi1dDvu8z1rRfIdQP10JDmVseoTkmWQ44m9kCmg7W0octrzPYlgm772JBr5uSkCmncyV/A
1GUB+4K3T+j75rPBTD6ETX3SEqB3YekpwWxLNd70oSf+3wyy5JMEABxLWkjrCc5ZSbW6Ix/tJenujKEB root@sheep03
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDzgi/07ezKFzTor8mpnYavYhLT6ukONT1SxpPwqTzCABTA91zLfhSCLxTGLE
ML2eaNIvfxgqk1UvpUzp2z9wZdmsnWeZ9BcL20z3GsgNmfBezKPUw+2K9yCPodhFbtjxSojNz8Cm+XXA5AGloCTZhpaxsrBi1DL
aqYtTKRuZ6xBGtTAQtdbi0mogymkBAgBd/wwtDXrp65B5ihVDrM2EoZe656Lzh7JRM0b4lvB1spDj1IyysFTgPGJy1/JvSLFWi
Z8TnG1Lvd34SZeIeaK1JYazi6aoRkieSv7UFd9JF75hNhAVsHR4SP0HP8XTu3s26Ad1N3lbn/8lJXQo67F5 root@sheep04
root@sheep01:~# vim /etc/ssh/ssh_config
```

Файл создан, копируем его на VM sheep02, sheep03, sheep04:

```
root@sheep02:~# scp 10.100.10.11:~/.ssh/authorized_keys ~/.ssh/
```

```
root@sheep03:~# scp 10.100.10.11:~/.ssh/authorized_keys ~/.ssh/
```

```
root@sheep04:~# scp 10.100.10.11:~/.ssh/authorized_keys ~/.ssh/
```

На host-машине открываем файл authorized_keys и вставляем ключи через буфер обмена.

```
root@host:~# cat > /root/.ssh/authorized_keys
```

На VM sheep01 отредактируем файл ssh_config, необходимо раскомментировать параметр **StrictHostKeyChecking** и изменить его значение на **no**

```
root@sheep01:~# vim /etc/ssh/ssh_config
```

```
Host *
# ForwardAgent no
# ForwardX11 no
# ForwardX11Trusted yes
# PasswordAuthentication yes
# HostbasedAuthentication no
# GSSAPIAuthentication no
# GSSAPIDelegateCredentials no
# GSSAPIKeyExchange no
# GSSAPITrustDNS no
# BatchMode no
# CheckHostIP yes
# AddressFamily any
# ConnectTimeout 0
# StrictHostKeyChecking no
# IdentityFile ~/.ssh/id_rsa
# IdentityFile ~/.ssh/id_dsa
# IdentityFile ~/.ssh/id_ecdsa
# IdentityFile ~/.ssh/id_ed25519
```

Сохраняем файл и копируем его на остальные VM:

```
root@sheep01:~# for i in 2 3 4; do scp /etc/ssh/ssh_config 10.100.10.1${i}:/etc/ssh/; done
Warning: Permanently added '10.100.10.12' (ECDSA) to the list of known hosts.
ssh_config                               100% 1579      1.2MB/s   00:00
Warning: Permanently added '10.100.10.13' (ECDSA) to the list of known hosts.
ssh_config                               100% 1579      1.1MB/s   00:00
Warning: Permanently added '10.100.10.14' (ECDSA) to the list of known hosts.
ssh_config                               100% 1579      1.1MB/s   00:00
```

На VM sheep01 отредактируем файл hosts:

```
root@sheep01:~# vim /etc/hosts
```

```
127.0.0.1    localhost
127.0.1.1    ubuntu

10.100.10.11  sheep01
10.100.10.12  sheep02
10.100.10.13  sheep03
10.100.10.14  sheep04

# The following lines are desirable for IPv6 capable hosts
::1         localhost ip6-localhost ip6-loopback
ff02::1     ip6-allnodes
ff02::2     ip6-allrouters
```

Сохраняем файл и копируем его на остальные VM:

```
root@sheep01:~# for i in 2 3 4; do scp /etc/hosts 10.100.10.1${i}:/etc/; done
hosts                                     100% 271      212.8KB/s   00:00
hosts                                     100% 271      204.2KB/s   00:00
hosts                                     100% 271      175.5KB/s   00:00
root@sheep01:~#
```

3.2 Установка сервисов

На каждой VM устанавливаем следующие пакеты:

```
root@sheep04:~# apt install pacemaker libvirt-bin libvirt-daemon-driver-storage-sheepdog sheepdog c  
rmsh
```

После установки пакетов, останавливаем следующие сервисы на всех VM:

```
root@sheep01:~# systemctl stop pacemaker  
root@sheep01:~# systemctl stop corosync  
root@sheep01:~# systemctl stop sheepdog  
root@sheep01:~#
```

Отредактируем файл конфигурации corosync.conf:

```
root@sheep01:~# vim /etc/corosync/corosync.conf
```

И приведем его к следующему виду:

```
totem {  
    version: 2  
    cluster_name: debian  
    token: 3000  
    token_retransmits_before_loss_const: 10  
    clear_node_high_bit: yes  
    crypto_cipher: none  
    crypto_hash: none  
  
    interface {  
        ringnumber: 0  
        bindnetaddr: 10.100.10.0  
        mcastaddr: 239.210.25.1  
        mcastport: 5405  
        ttl: 1  
    }  
}  
  
logging {  
    fileline: off  
    to_stderr: no  
    to_logfile: yes  
    logfile: /var/log/corosync/corosync.log  
    to_syslog: no  
    syslog_facility: daemon  
    debug: off  
    timestamp: on  
    logger_subsys {  
        subsys: QUORUM  
        debug: off  
    }  
}  
  
quorum {  
    provider: corosync_votequorum  
    expected_votes: 2  
}  
"/etc/corosync/corosync.conf" 37L, 557C
```

Сохраняем файл и копируем его на остальные VM:

```
root@sheep01:~# for i in 2 3 4; do scp /etc/corosync/corosync.conf 10.100.10.1${i}:/etc/corosync/  
done
```

Установим для удобства на все VM Midnight Commander:

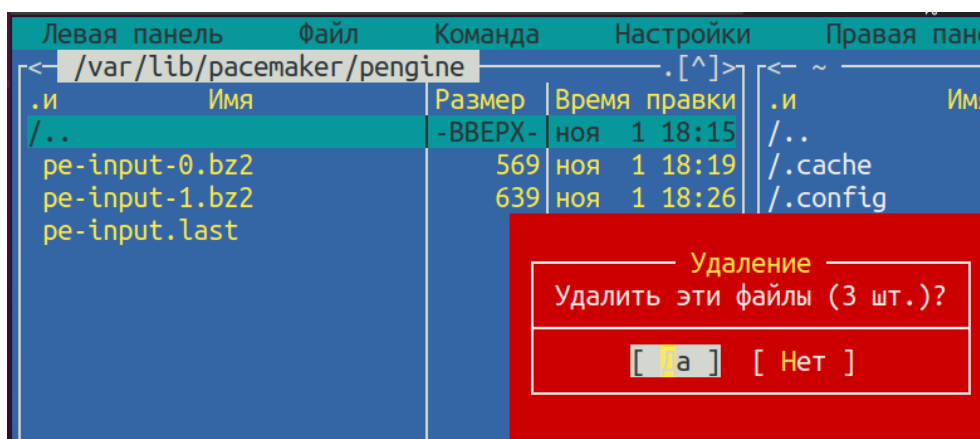
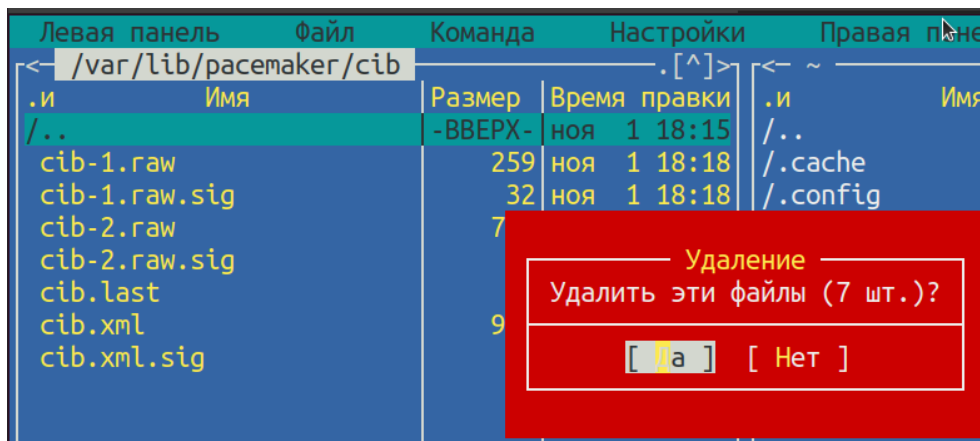
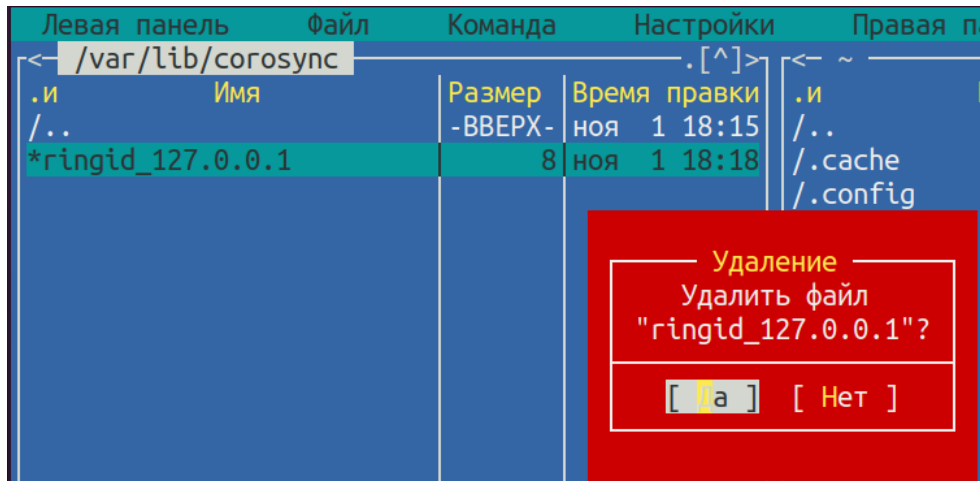
```
root@sheep01:~# apt install mc
```

И запустим его. Удалим все файлы из следующих директорий (на всех VM):

```
/var/lib/corosync
```

```
/var/lib/pacemaker/cib
```

```
/var/lib/pacemaker/engine
```



Запускаем сервисы corosync и pacemaker на всех VM:

```
root@sheep01:~# systemctl start corosync
root@sheep01:~# systemctl start pacemaker
```

На любой VM настроим менеджер ресурсов crm:

```
root@sheep01:~# crm configure edit
```

И приведем к следующему виду:

```
node 174328331: sheep01
node 174328332: sheep02
node 174328333: sheep03
node 174328334: sheep04
primitive s_sheep01 stonith:external/libvirt \
  params hostlist=sheep01 hypervisor_uri="qemu+ssh://root@10.100.10.1/system"
primitive s_sheep02 stonith:external/libvirt \
  params hostlist=sheep02 hypervisor_uri="qemu+ssh://root@10.100.10.1/system"
primitive s_sheep03 stonith:external/libvirt \
  params hostlist=sheep03 hypervisor_uri="qemu+ssh://root@10.100.10.1/system"
primitive s_sheep04 stonith:external/libvirt \
  params hostlist=sheep04 hypervisor_uri="qemu+ssh://root@10.100.10.1/system"
location ls_sheep01 s_sheep01 -inf: sheep01
location ls_sheep02 s_sheep02 -inf: sheep02
location ls_sheep03 s_sheep03 -inf: sheep03
location ls_sheep04 s_sheep04 -inf: sheep04
property cib-bootstrap-options: \
  have-watchdog=false \
  dc-version=1.1.18-2b07d5c5a9 \
  cluster-infrastructure=corosync \
  cluster-name=debian
```

Отообразим конфигурацию crm:

```
root@sheep01:~# crm configure show
```

```
node 174328331: sheep01
node 174328332: sheep02
node 174328333: sheep03
node 174328334: sheep04
primitive s_sheep01 stonith:external/libvirt \
  params hostlist=sheep01 hypervisor_uri="qemu+ssh://root@10.100.10.1/system"
primitive s_sheep02 stonith:external/libvirt \
  params hostlist=sheep02 hypervisor_uri="qemu+ssh://root@10.100.10.1/system"
primitive s_sheep03 stonith:external/libvirt \
  params hostlist=sheep03 hypervisor_uri="qemu+ssh://root@10.100.10.1/system"
primitive s_sheep04 stonith:external/libvirt \
  params hostlist=sheep04 hypervisor_uri="qemu+ssh://root@10.100.10.1/system"
location ls_sheep01 s_sheep01 -inf: sheep01
location ls_sheep02 s_sheep02 -inf: sheep02
location ls_sheep03 s_sheep03 -inf: sheep03
location ls_sheep04 s_sheep04 -inf: sheep04
property cib-bootstrap-options: \
  have-watchdog=false \
  dc-version=1.1.18-2b07d5c5a9 \
  cluster-infrastructure=corosync \
  cluster-name=debian
```

На всех VM создадим директорию /srv/sheep:

```
root@sheep01:~# mkdir /srv/sheep
```

И файловую систему `xf`s на `/dev/sda4`:

```
root@sheep01:~# mkfs.xfs -i size=1024 /dev/sda4
```

```
root@sheep01:~# mkdir /srv/sheep
root@sheep01:~# mkfs.xfs -i size=1024 /dev/sda4
meta-data=/dev/sda4          isize=1024   agcount=4, agsize=1015679 blks
=                               sectsz=512   attr=2, projid32bit=1
=                               crc=1       finobt=1, sparse=0, rmapbt=0, reflink=0
data      =                   bsize=4096  blocks=4062715, imapct=25
=                               sunit=0     swidth=0 blks
naming    =version 2         bsize=4096  ascii-ci=0 ftype=1
log        =internal log    bsize=4096  blocks=2560, version=2
=                               sectsz=512   sunit=0 blks, lazy-count=1
realtime  =none             extsz=4096  blocks=0, rtextents=0
root@sheep01:~#
```

Далее добавим UUID вновь созданного диска в файл `/etc/fstab`:

```
root@sheep01:~# blkid | grep xfs
```

```
root@sheep01:~# blkid | grep xfs
/dev/sda4: UUID="adb005e6-a8bb-41ee-9c3d-00d0272ea2b4" TYPE="xfs" PARTLABEL="Linux filesystem" PART
UUID="e8afab65-4019-4005-ac4d-c5f77cc38166"
root@sheep01:~#
```

```
root@sheep01:~# vim /etc/fstab
```

```
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point> <type> <options>          <dump> <pass>
# / was on /dev/sda3 during installation
UUID=9fadf4ac-6eb1-4ce7-a8ef-9fe78fc6e9f1 /          ext4      noatime,nodiratime,errors=remount-ro 0    1
# swap was on /dev/sda2 during installation
UUID=526f986a-a143-4242-963b-7ffb7105c07c none      swap      sw        0        0
UUID=adb005e6-a8bb-41ee-9c3d-00d0272ea2b4 /srv/sheep xfs       noatime,nodiratime,noexec,nodev,nobarrier,logbufs=8
0 1
```

Указав при этом следующие параметры:

```
/srv/sheep    xfs
noatime,nodiratime,noexec,nodev,nobarrier,logbufs=8    0    1
```

Далее монтируем все устройства:

```
root@sheep01:~# mount -a
```

Отредактируем файл /etc/default/sheepdog:

```
root@sheep01:~# vim /etc/default/sheepdog
```

```
# start sheepdog at boot [yes|no]
START="yes"

# Arguments to run the daemon with
# Options:
# -p, --port           specify the TCP port on which to listen
# -l, --loglevel       specify the level of logging detail
# -d, --debug          include debug messages in the log
# -D, --directio       use direct IO when accessing the object store
# -z, --zone           specify the zone id
# -c, --cluster        specify the cluster driver
DAEMON_ARGS="-p 7000"

# SHEEPDOG_PATH
#   Proper LSB systems will store sheepdog files in /var/lib/sheepdog. The init script uses th
#   is directory by default.
#   The directory must be on a filesystem with xattr support. In the case of ext3, user_xattr
#   should be added to the
#   mount options.
#
#   mount -o remount,user_xattr /var/lib/sheepdog
SHEEPDOG_PATH="/srv/sheep"
~
```

И копируем на остальные ВМ:

```
root@sheep01:~# for i in 2 3 4; do scp /etc/default/sheepdog 10.100.10.1${i}:/etc/default/; done
sheepdog          100% 850    1.0MB/s   00:00
sheepdog          100% 850    623.4KB/s 00:00
sheepdog          100% 850    700.3KB/s 00:00
```

Запускаем сервис sheepdog на всех ВМ:

```
root@sheep01:~# systemctl start sheepdog
```

Выведем список нод:

```
root@sheep01:~# dog node list
```

```
root@sheep01:~# dog node list
Id  Host:Port      V-Nodes  Zone
0   10.100.10.11:7000  128     185230346
1   10.100.10.12:7000  128     202007562
2   10.100.10.13:7000  128     218784778
3   10.100.10.14:7000  128     235561994
```

Создаем кластер:

```
root@sheep01:~# dog cluster format 3
```

```
root@sheep01:~# dog cluster format 3
using backend plain store
```

Создаем образ vdi:

```
root@sheep01:~# dog vdi create VyOS 2G
```

```
root@sheep01:~# dog vdi list
```

```
root@sheep01:~# dog vdi list
Name      Id  Size  Used  Shared  Creation time  VDI id  Copies  Tag
VyOS     0  2.0 GB  0.0 MB  0.0 MB  2021-11-01 19:22  69c87a  3
```

```
root@sheep01:~# qemu-img info sheepdog:VyOS
```

```
root@sheep01:~# qemu-img info sheepdog:VyOS
image: json:{"driver": "raw", "file": {"server.host": "localhost", "server.port": "7000", "tag": "
", "driver": "sheepdog", "server.type": "inet", "vdi": "VyOS"}}
file format: raw
virtual size: 2.0G (2147483648 bytes)
disk size: 0
```

```
root@sheep01:~# cd /srv/kvm-iso
```

Скачиваем iso-образ VyOS:

```
root@sheep01:/srv/kvm-iso# wget
https://vcp.meganet.ru/dists/vyos/1.3.1-c2/live-image-
amd64.hybrid.iso
```

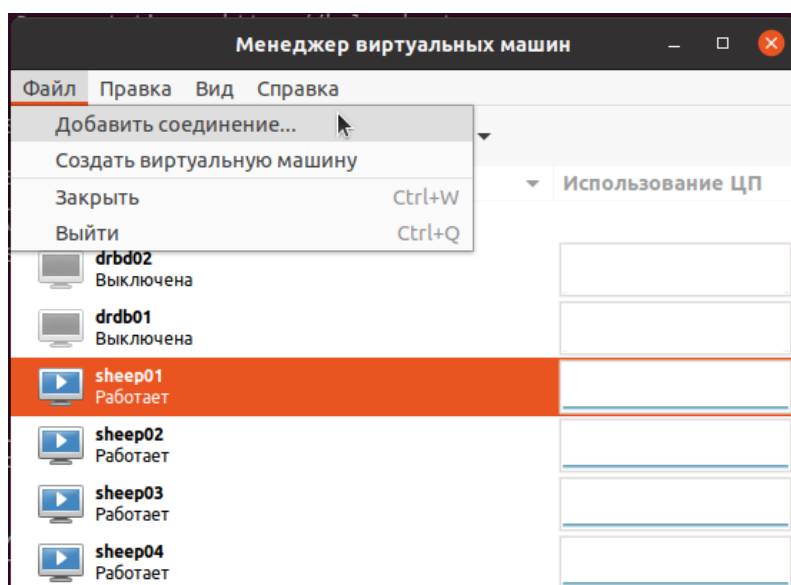
```
root@sheep01:/srv/kvm-iso# wget https://vcp.meganet.ru/dists/vyos/1.3.1-c2/live-image-amd64.hybrid
.iso
--2021-11-01 19:28:48-- https://vcp.meganet.ru/dists/vyos/1.3.1-c2/live-image-amd64.hybrid.iso
Распознаётся vcp.meganet.ru (vcp.meganet.ru)... 37.233.82.14
Подключение к vcp.meganet.ru (vcp.meganet.ru)|37.233.82.14|:443... соединение установлено.
HTTP-запрос отправлен. Ожидание ответа... 200 OK
Длина: 359661568 (343M) [application/octet-stream]
Сохранение в: «live-image-amd64.hybrid.iso»

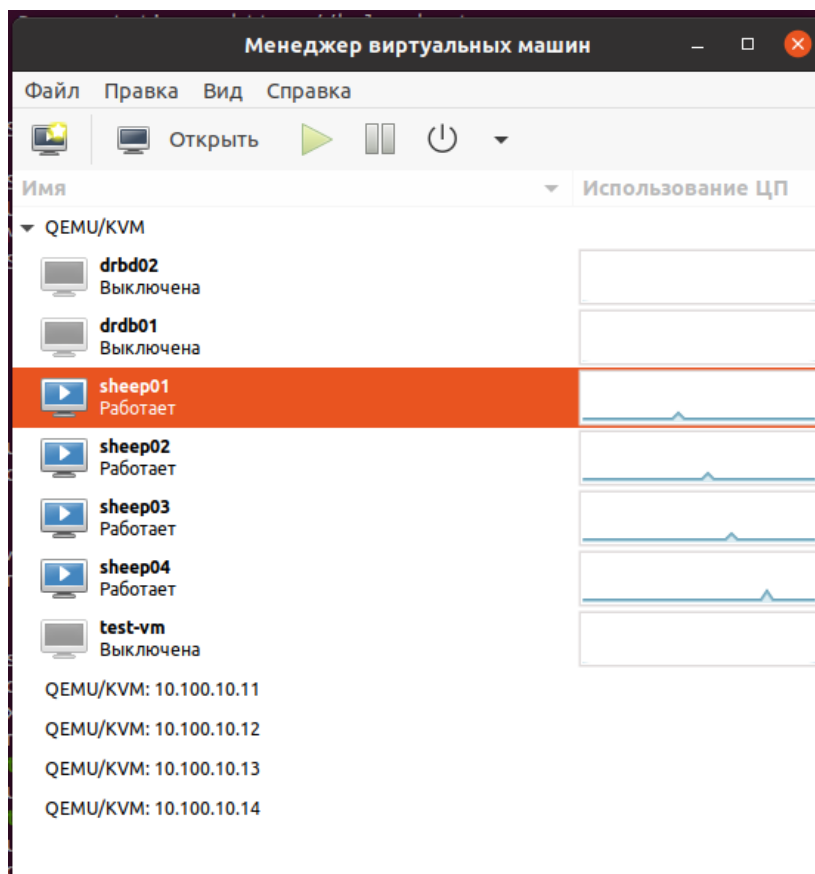
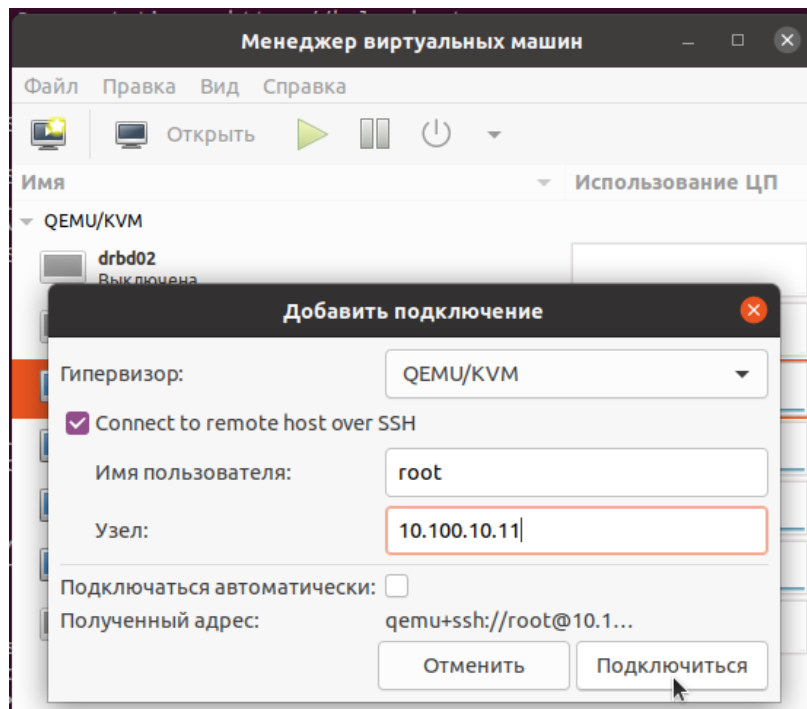
live-image-amd64.hybrid. 40%[=====] ] 139,32M 4,96MB/s ост 33s
```

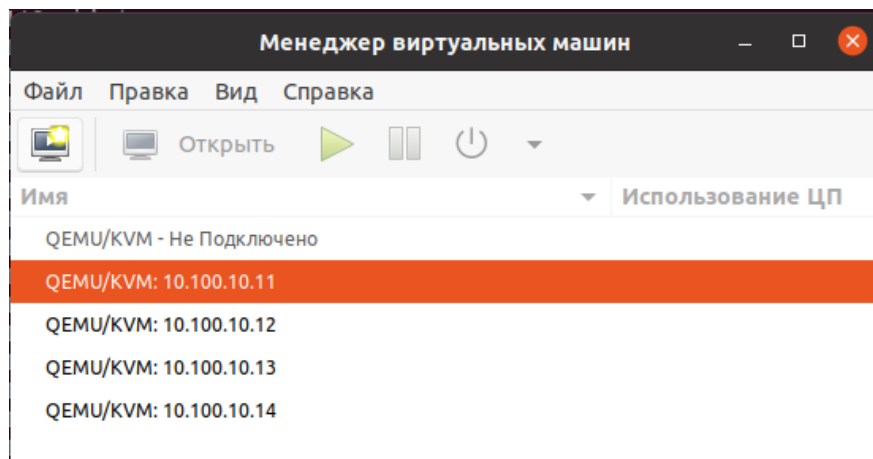
```
root@sheep01:/srv/kvm-iso# cd ..
root@sheep01:/srv# mkdir vm
root@sheep01:/srv# cp /etc/libvirt/qemu/VyOS.xml /srv/vm
копируем на остальные VM:
```

```
root@sheep01:/srv# for i in 2 3 4; do scp -r /srv/vm 10.100.10.1${i}:/srv; done
```

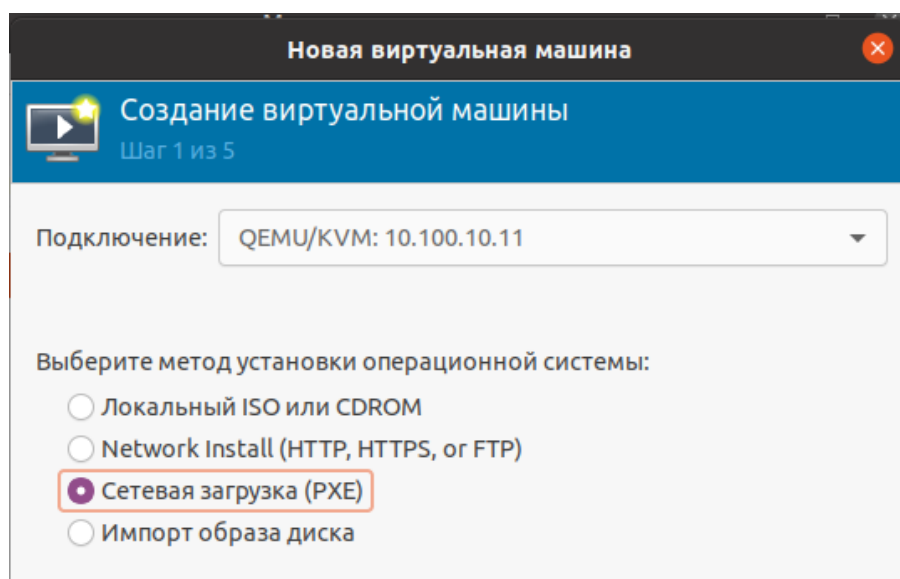
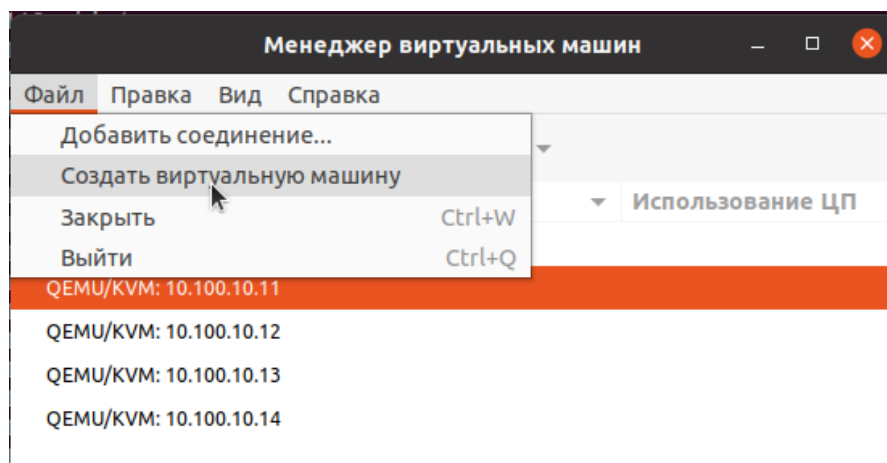
Добавляем новое соединение в Менеджере виртуальных машин,
(соединения создаем по числу VM):

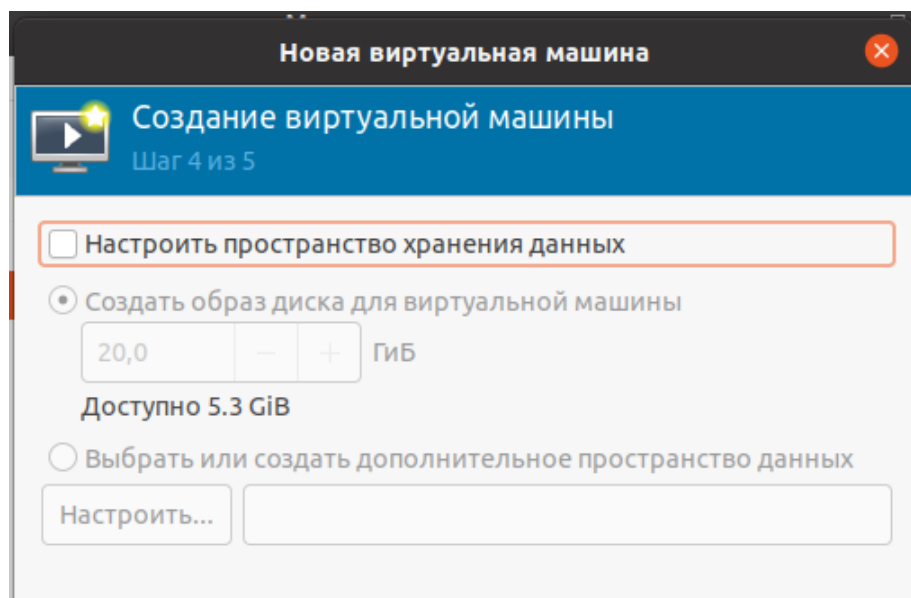
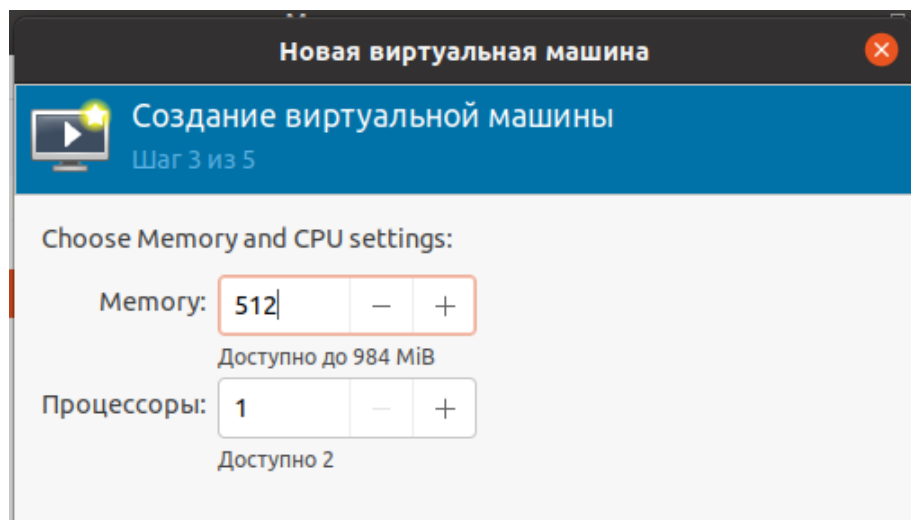
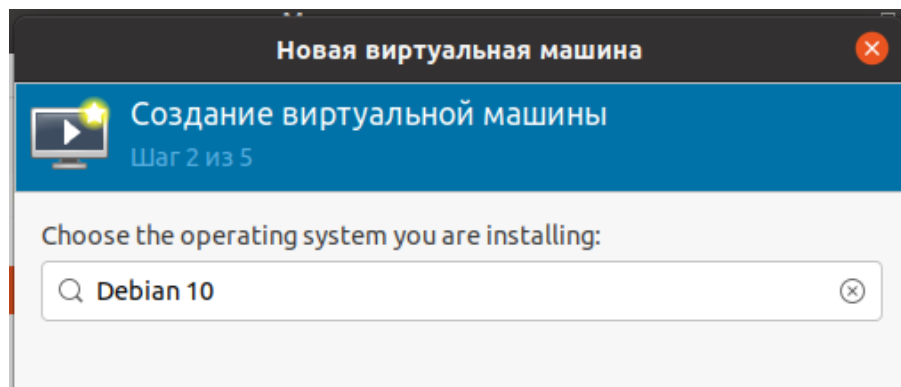


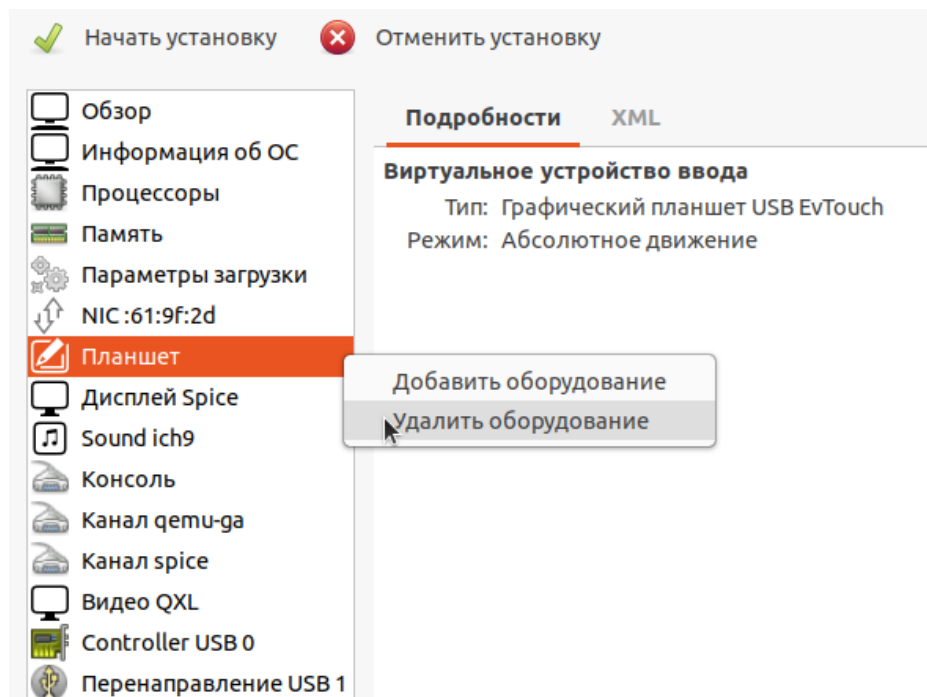
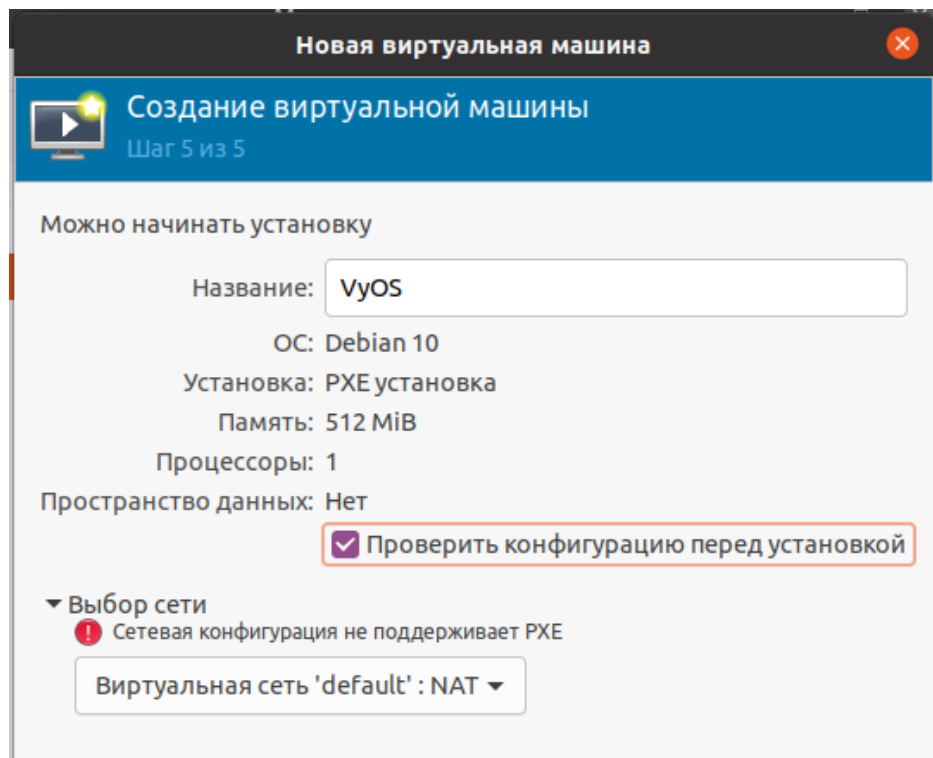


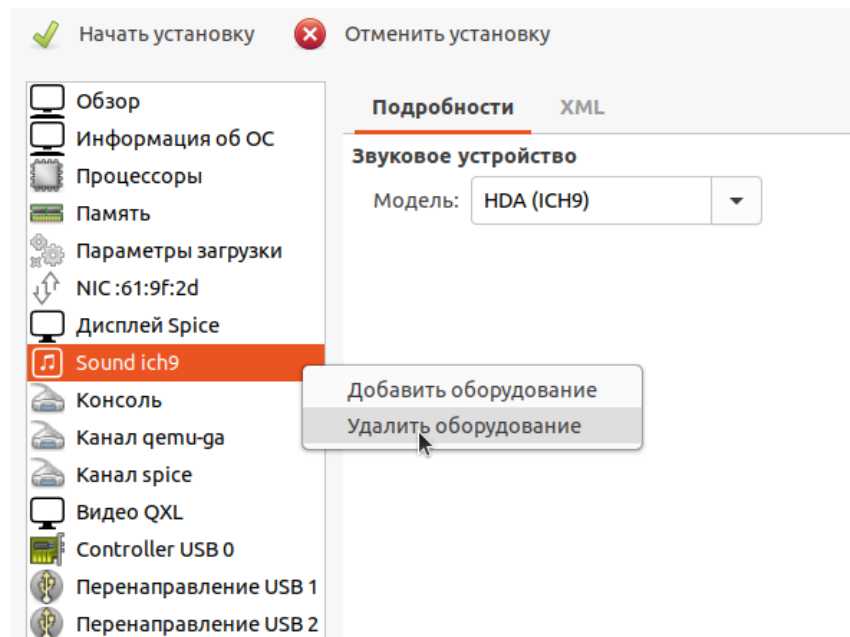


Создаем новую VM:









Удаляем неиспользуемое оборудование. Запускаем VM и останавливаем принудительно.

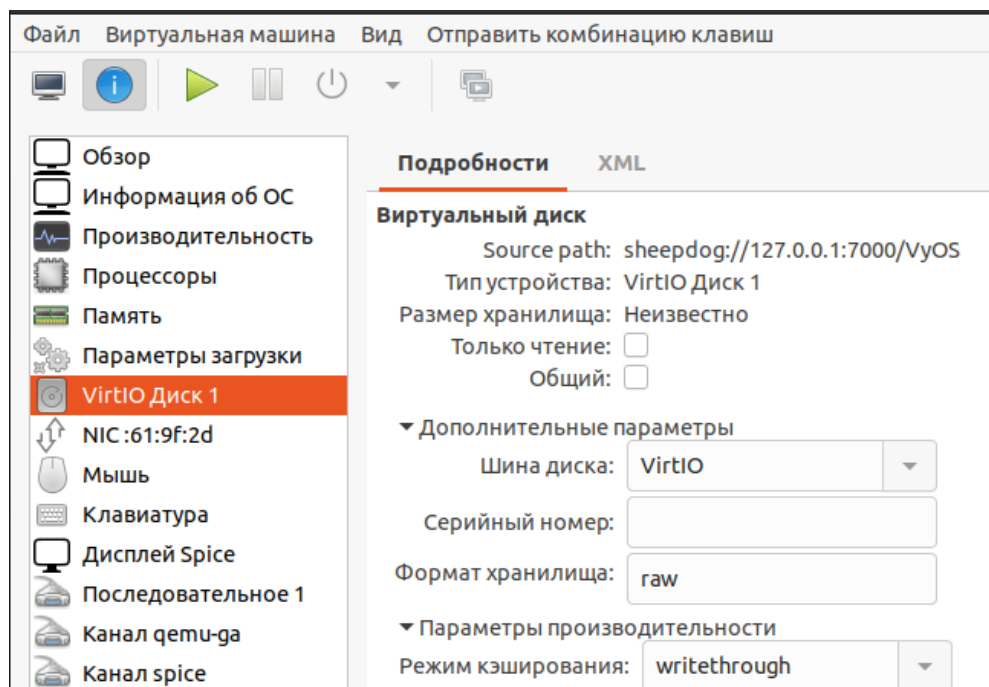
Отредактируем файл VM VyOS:
`root@sheep01:~# virsh edit VyOS`

Вставляем следующий код в блок `<devices>` после `<emulator>`:

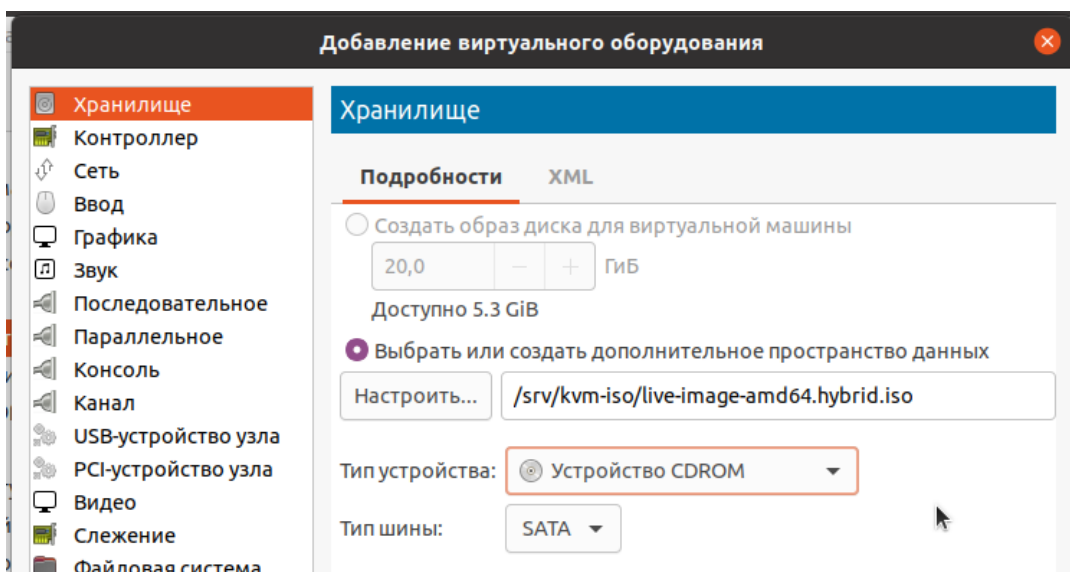
```
<disk type='network' device='disk'>
  <driver name='qemu' type='raw' cache='writethrough'
io='threads' ioeventfd='on' event_idx='off' />
  <source protocol='sheepdog' name='VyOS'>
    <host name='127.0.0.1' port='7000' />
  </source>
  <target dev='vda' bus='virtio' />
  <alias name='virtio-disk0' />
</disk>
```

```
<on_poweroff>destroy</on_poweroff>
<on_reboot>restart</on_reboot>
<on_crash>destroy</on_crash>
<pm>
  <suspend-to-mem enabled='no' />
  <suspend-to-disk enabled='no' />
</pm>
<devices>
  <emulator>/usr/bin/kvm-spice</emulator>
  <disk type='network' device='disk'>
    <driver name='qemu' type='raw' cache='writethrough' io='threads' ioeventfd='on' event_idx='off' />
    <source protocol='sheepdog' name='VyOS'>
      <host name='127.0.0.1' port='7000' />
    </source>
    <target dev='vda' bus='virtio' />
    <address type='pci' domain='0x0000' bus='0x06' slot='0x00' function='0x0' />
  </disk>
  <controller type='usb' index='0' model='qemu-xhci' ports='15'>
    <address type='pci' domain='0x0000' bus='0x02' slot='0x00' function='0x0' />
  </controller>
  <controller type='sata' index='0'>
    <address type='pci' domain='0x0000' bus='0x00' slot='0x1f' function='0x2' />
  </controller>
  <controller type='pci' index='0' model='pcie-root' />
  <controller type='pci' index='1' model='pcie-root-port'>
    <model name='pcie-root-port' />
    <target chassis='1' port='0x10' />
  </controller>
</devices>
```

Записали изменения. Смотрим свойства VM, появилось новое оборудование – VirtIO Диск 1.



Добавляем новое оборудование - CDROM.



Выставляем загрузку по CDROM в Параметрах загрузки.

