

# Работа с виртуализацией: QEMU/KVM. Часть 1

## Задание 0. Построение стенда

Схема виртуального лабораторного стенда



Рисунок 1. Схема стенда

1. Создать виртуальные машины для работы

Название виртуальной машины	Источник	Тип инстанса	Сети для внешнего подключения	Размер диска
labnode-1	Образ-Ubuntu-server20.04	small	external-net	10GB

Так же нужно проверить развернутую инфраструктуру на соответствие схеме на рисунке 1.

## Задание 1. Установка QEMU.

На **labnode-1**:

1. Установить эмулятор аппаратного обеспечения различных платформ:

```
sudo apt update
sudo apt install qemu-kvm qemu-system qemu-utils -y
```

2. Убедиться, что модуль KVM загружен (с помощью команд lsmmod и grep):

```
lsmmod | grep -i kvm
```

```
cloudadmin@labnode-1:~$ lsmod | grep -i kvm
kvm_intel          282624  0
kvm                663552  1 kvm_intel
cloudadmin@labnode-1:~$
```

## Задание 2. Управление образами дисков при помощи qemu-img.

Чтобы запускать виртуальные машины, QEMU требуются образы для хранения определенной файловой системы данной гостевой ОС. Такой образ сам по себе имеет тип некоторого файла, и он представляет всю гостевую файловую систему, расположенную в некотором виртуальном диске. QEMU поддерживает различные образы и предоставляет инструменты для создания и управления ими. Можно построить пустой образ диска с помощью утилиты qemu-img, которая должна быть установлена.

1. Проверить какие типы образов поддерживаются qemu-img:

```
sudo qemu-img -h | grep Supported
```

2. Создать образ qcow2 с названием system.qcow2 и размером 5 ГБ:

```
sudo qemu-img create -f qcow2 system.qcow2 5G
```

3. Проверить что файл был создан:

```
ls -lah system.qcow2
```

4. Посмотреть дополнительную информацию о данном образе:

```
sudo qemu-img info system.qcow2
```

## Задание 3. Изменение размера образа.

Не все типы образов поддерживают изменение размера. Для изменения размера такого образа сначала нужно преобразовать его в образ raw с помощью команды qemu-img convert.

1. Конвертировать образ диска из формата qcow2 в raw:

```
sudo qemu-img convert -f qcow2 -O raw system.qcow2 system.raw
```

2. Добавить дополнительно 5 ГБ к образу:

```
sudo qemu-img resize -f raw system.raw +5G
```

3. Проверить новый текущий размер образа:

```
sudo qemu-img info system.raw
```

4. Конвертировать образ диска обратно из raw в qcow2:

```
sudo qemu-img convert -f raw -O qcow2 system.raw system.qcow2
```

## Задание 4. Загрузка образа OpenWRT.

Для загрузки образов с общедоступных репозиторий требуется утилита **curl**. Загрузить необходимый образ, воспользовавшись **curl**:

```
curl -L https://s3.resds.ru/itt/openwrt.img -o /tmp/openwrt.raw
```

## Задание 5. Создание виртуального окружения с помощью qemu-system.

1. Для того чтобы подключиться к виртуальной машине по протоколу удаленного рабочего стола **VNC**, нужно открыть порт **5900**. В группе безопасности
2. Посмотреть ip адрес вашего сервера

```
ip address
```

3. Запустить систему при помощи qemu-system:

```
sudo qemu-system-x86_64 -hda /tmp/openwrt.raw -m 1024 -vga cirrus -vnc 0.0.0.0:0
```

4. Установите и откройте программу (Remmina). И подключитесь по протоколу **VNC** к гипервизору
5. Набрать команду **uname -a**. Посмотреть на версию ядра ОС. Выключить виртуальную машину, набрав

```
poweroff
```

## Задание 6. Установка Libvirt и Virsh.

```
sudo apt install -y libvirt-daemon-system virtinst
```

## Задание 7. Настройка моста.

Установить пакет **bridge-utils**:

```
sudo apt install -y bridge-utils
```

Вывести на экран имеющиеся интерфейсы:

```
ip -c address
```

Открыть файл **/tmp/labnet.xml**:

```
sudo vi /tmp/labnet.xml
```

И заполнить по примеру

```
<network>
  <name>labnet</name>
  <forward mode='nat'>
    <nat>
      <port start='1024' end='65535'/>
    </nat>
  </forward>
  <bridge name='labnet' stp='on' delay='0'/>
  <ip address='192.168.22.1' netmask='255.255.255.0'>
    <dhcp>
      <range start='192.168.22.2' end='192.168.22.254'/>
    </dhcp>
  </ip>
</network>
```

Добавляем сеть и запускаем ее:

```
sudo virsh net-define /tmp/labnet.xml
sudo virsh net-start labnet
sudo virsh net-autostart labnet
```

## Задание 8. Создание виртуальной машины.

Переместить образ **openwrt** в **/var/lib/libvirt/images/**

```
sudo mv /tmp/openwrt.raw /var/lib/libvirt/images/
```

Следующая команда создаст новую KVM виртуальную машину

```
sudo virt-install --name openwrt \  
--ram 1024 \  
--disk path=/var/lib/libvirt/images/openwrt.raw,cache=none \  
--boot hd \  
--vcpus 1 \  
--network network=labnet \  
--graphics vnc,listen=0.0.0.0 \  
--wait 0
```

Символ \ - обратная косая черта используется для экранирования специальных символов в строковых и символьных литералах. В данном случае нужна, чтобы переместить каретку на новую строку, для наглядности. После ее добавления в команду можно нажать *Enter*, но строка не отправится на выполнение, а ввод команды продолжится. При ошибке в наборе команды, можно не набирать ее заново, а нажать стрелку вверх, исправить ее, и снова нажать  Подробнее о параметрах:

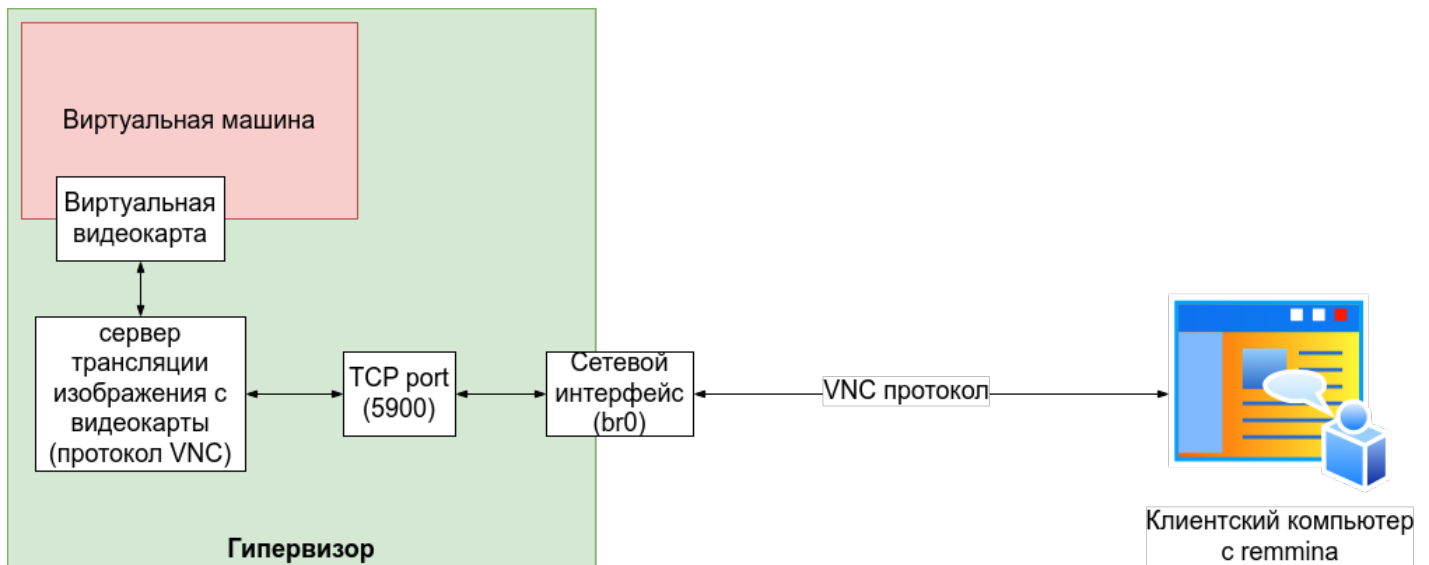
Название параметра	Описание параметра
name	Имя виртуальной машины, которое будет отображаться в virsh
ram	Размер оперативной памяти в МБ
disk	Диск, который будет создан и подключен к виртуальной машине
vcpus	Количество виртуальных процессоров, которые нужно будет настроить для гостя
os-type	Тип операционной системы
os-variant	Название операционной системы
network	Определение сетевого интерфейса, который будет подключен к виртуальной машине
graphics	Определяет графическую конфигурацию дисплея.
cdrom	CD ROM устройство

Далее необходимо подключиться к гипервизору через программу **Remmina**.

Открыть её (название - Remmina). Подключиться по адресу виртуальной машины, выбрав протокол VNC. Вернуться в консоль **labnode-1**. Проверить состояние гостевой системы, используя команду (Если в консоли написано “Domain installation still in progress”, то нажмите ):

```
sudo virsh list --all
```

## Задание 9. Операции с виртуальной машиной.



Рассмотрим работу утилиты **virsh**. Чтобы подключиться к ВМ по протоколу удаленного доступа, используется следующая команда:

```
sudo virsh domdisplay openwrt
```

Результатом исполнения этой команды будет адрес для подключения к графическому интерфейсу ВМ, с указанием номера порта. Получить информацию о конкретной ВМ можно так:

```
sudo virsh dominfo openwrt
```

В результате чего будет выведена информация, об основных параметрах виртуальной машины. Выключить/включить ВМ можно с помощью команды:

```
sudo virsh destroy openwrt
sudo virsh start openwrt
```

Зайти в консоль виртуальной машины можно с помощью команды:

```
sudo virsh console openwrt
```

Посмотрите размер разделов

```
df -h
```

Выйдите из виртуальной машины

Добавление ВМ в автозапуск происходит следующим образом:

```
sudo virsh autostart openwrt
```

Теперь, виртуальная машина будет автоматически запускаться, после перезагрузки сервера. Кроме того, может потребоваться отредактировать XML конфигурацию VM:

```
sudo virsh edit openwrt
```

Необходимо выгрузить конфигурацию VM в XML в файл, используя команду:

```
sudo virsh dumpxml openwrt | tee openwrt.xml
```

Необходимо удалить VM, и убедиться, что её нет в списке виртуальных машин:

```
sudo virsh undefine openwrt  
sudo virsh destroy openwrt  
sudo virsh list --all
```

Увеличиваем размер диска на 1GB

```
sudo qemu-img resize -f raw /var/lib/libvirt/images/openwrt.raw +1G
```

Для создания VM из XML существует следующая команда:

```
sudo virsh define openwrt.xml  
sudo virsh list --all
```

После запуска виртуальной машины необходимо увеличить размер основного раздела с помощью fdisk:

```
fdisk /dev/sda
```

После этого необходимо увеличить размер файловой системы:

```
BOOT="$(sed -n -e "/\s\boot\s.*$/s///p;q)" /etc/mtab)"  
DISK="${BOOT%%[0-9]*}"  
PART="$(( ${BOOT##*[0-9]} +1 ))"  
ROOT="${DISK}${PART}"  
LOOP="$(losetup -f)"  
losetup ${LOOP} ${ROOT}  
fsck.ext4 -y -f ${LOOP}  
resize2fs ${LOOP}  
reboot
```

Проверить размер разделов

df -h

---

Версия #15

Тарабанов Илья Федорович создал 27 ноября 2023 09:59:01

Тарабанов Илья Федорович обновил 16 мая 2024 19:13:52