



УДК 004.451.9

КОНСОЛЬНОЕ И VIRT-MANAGER УПРАВЛЕНИЕ KVM**Балахонова Елизавета Андреевна****Коротун Софья Олеговна**

студенты

Российский государственный университет нефти и газа (НИУ) имени И.М. Губкина, г. Москва

Аннотация. В данной статье рассмотрим функционал KVM, консольное и Virt-Manager управление в виртуальной среде VirtualBox при помощи виртуальной машины с операционной системой Alt.

Ключевые слова: KVM, консольное управление, Virt-Manager, VirtualBox, Alt.

ВВЕДЕНИЕ

В данной статье рассмотрим функционал KVM, консольное и Virt-Manager управление для дистрибутива Альт. Будет проведен сравнительный анализ функционала управления KVM.

***Объект исследования** — процесс управления виртуальными машинами, созданными на базе технологии KVM. Он состоит из функционала KVM, методов управления через консоль и графических интерфейсов.*

***Предмет исследования** — функциональные возможности системы виртуализации KVM. Методы и инструменты консольного управления KVM, возможности графических инструментов управления KVM в операционной системе ALT, установленной в виртуальной машине VirtualBox.*

***Цель исследования** — анализ и описание возможностей, методов и инструментов управления виртуализацией на базе KVM, определение основных свойств и различий между консольным управлением и управлением через Virt-Manager в условиях виртуальной среды VirtualBox с операционной системой ALT.*

Функционал KVM

Для начала нужно разобраться что такое KVM.

KVM — это встроенная в Linux технология виртуализации, которая позволяет создавать виртуальные машины на базе процессоров x86 с аппаратной поддержкой Intel VT или AMD SVM. KVM позволяет использовать уже существующие образы дисков QEMU, VMware и других систем для запуска различных операционных систем на виртуальных машинах. [4]

KVM, будучи не отдельным модулем, а составной частью ядра Linux, напрямую зависит от его развития, это особенно заметно в системах на базе Red Hat. Одними из ключевых преимуществ KVM являются: возможность более эффективно использовать вычислительные

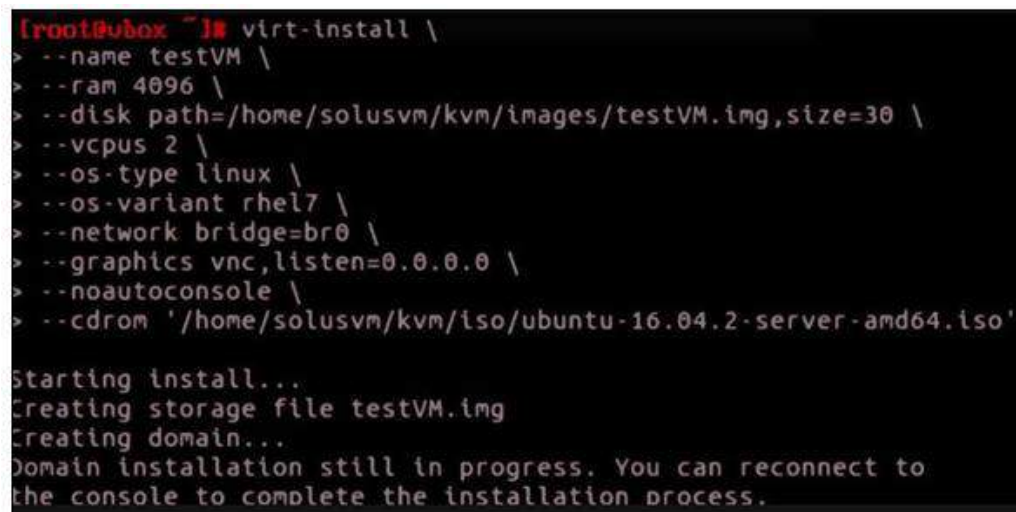
ресурсы с помощью over-committing, гибкое хранение данных и экономия места с помощью thin provisioning, улучшенная производительность на системах с NUMA благодаря автоматической балансировке, а также возможность добавлять виртуальные процессоры без перезагрузки. Управлять KVM можно как через консоль, так и через графический интерфейс, например, с помощью Virt-Manager. [5]

Консольное управление осуществляется с помощью утилиты virsh, которая позволяет создавать, включать, управлять виртуальными машинами, управлять сетевыми подключениями и подключаться к консоли виртуальной машины. [3]

Virt-manager — графический инструмент для управления виртуальными машинами в KVM, предоставляющий удобный простой способ работы с ними. С помощью него также можно создавать и включать машины, редактировать их параметры и следить за ресурсами.

Чтобы понять какое управление использовать для того или иного действия, попробуем создать виртуальную машину и протестировать возможные функции двумя способами.

Для создания виртуальной машины через консоль сначала необходимо скачать образ ISO операционной системы, которую мы хотим установить. Затем пропишем команду «virt-install», которая предоставляет множество опций для настройки (см. рисунок 1). [3]



```
[root@ubox ~]# virt-install \
> --name testVM \
> --ram 4096 \
> --disk path=/home/solusvm/kvm/images/testVM.img,size=30 \
> --vcpus 2 \
> --os-type linux \
> --os-variant rhel7 \
> --network bridge=br0 \
> --graphics vnc,listen=0.0.0.0 \
> --noautoconsole \
> --cdrom '/home/solusvm/kvm/iso/ubuntu-16.04.2-server-amd64.iso'

Starting install...
creating storage file testVM.img
creating domain...
Domain installation still in progress. You can reconnect to
the console to complete the installation process.
```

Рисунок 1 – Создание виртуальной машины в консоли

Разберем каждую строку отдельно, первая строка «name testVM» — это имя виртуальной машины. Далее «ram 4096» выделяет 4096 МБ памяти, следующая строка указывает путь к файлу виртуального диска. «vcpus 2» — означает, что нужно 2 процессора, следом за ней указываются образ и вариант операционной системы. Запись «network bridge=br0» указывает, что используем виртуальный сетевой мост br0. «graphics vnc, listen=0.0.0.0» включает графический доступ через VNC, прослушивая все ip-адреса хост-машины. Следующая команда отключает автоматическое перенаправление консоли в текущий терминал. Последняя строка —

путь к ISO образу. [2]

После завершения установки запускаем виртуальную машину с помощью команды «virsh start testVM». Можно проверить статус работы машины с помощью команд (см. рисунок 2):

```
[root@vbox ~]# virsh net-list
Name                               State    Autostart  Persistent
-----
testVM                             active   yes        yes

[root@vbox ~]# virsh net-info testVM
Name:                               testVM
UUID:                               5e87eb66-07fc-4794-ab2c-ecbc16bea912
Active:                             yes
Persistent:                         yes
Autostart:                          yes
Bridge:                             br0
```

Рисунок 2 – Отслеживание работы виртуальной машины

Теперь попробуем создать виртуальную машину с помощью virt-manager, для этого откроем приложение (см. рисунок 3). [6]

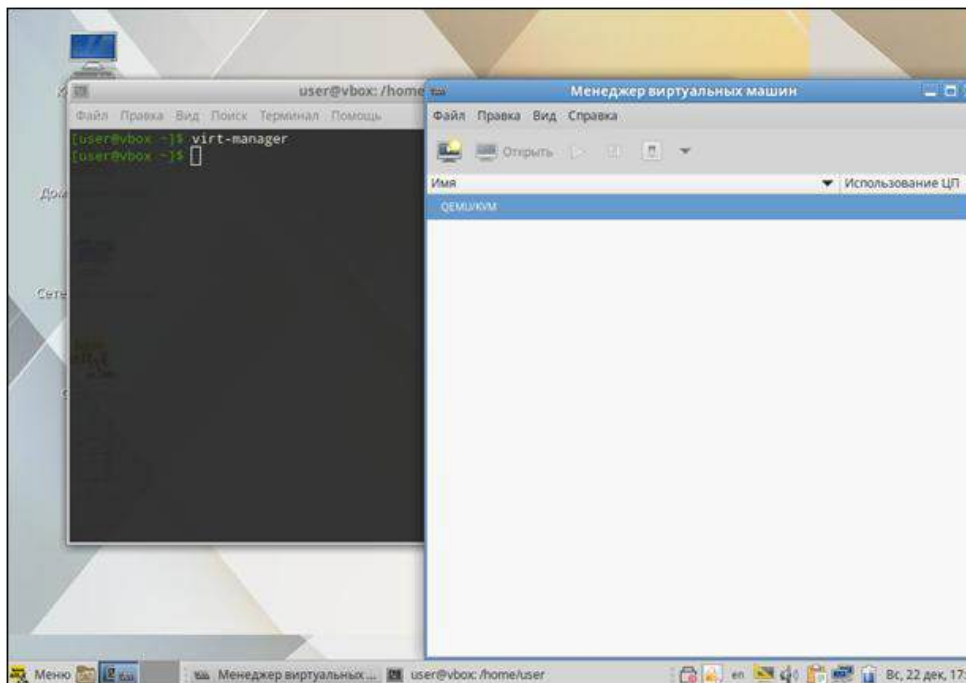


Рисунок 3 – Открытие virt-manager

Далее найдем кнопку «создать» и выбираем необходимые параметры, в том числе и образ, который также как и для консольного управления нужно скачать заранее, и начнем установку машины (см. рисунок 4, 5). [1]

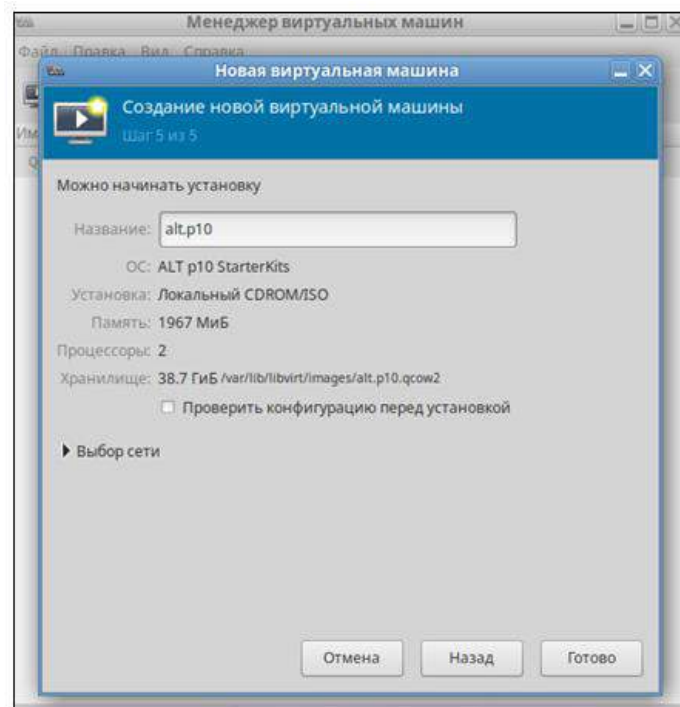


Рисунок 4 – Создание новой машины

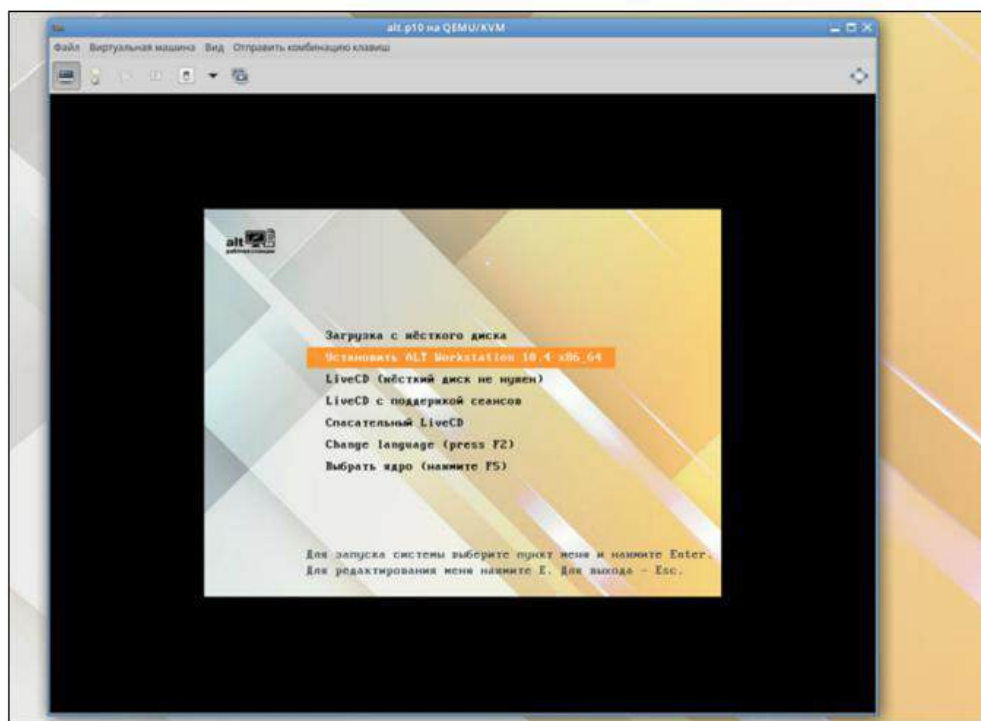


Рисунок 5 – Загрузка виртуальной машины

Помимо создания и включения можно посмотреть базовые настройки приложения, в которой есть возможность редактирования (см. рисунок 6). [7]

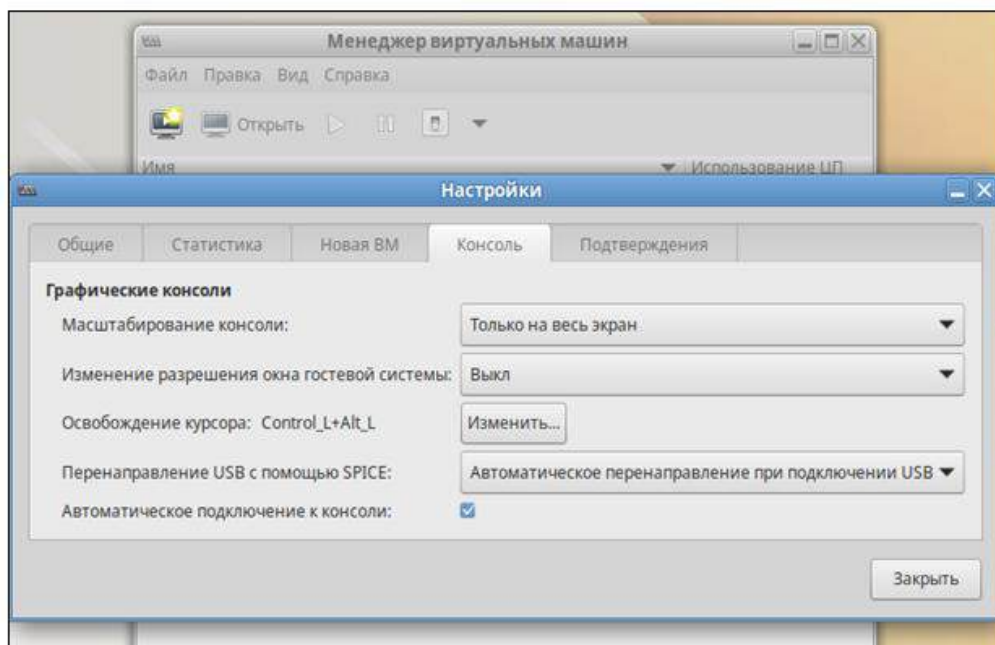


Рисунок 6 – Настройки virt-manager

Выбор управления определяется в зависимости от задачи, которую требуется выполнить. Для полного понимания покажем различия данных управлений в таблице (см. таблица 1).

Сравнительная таблица

Таблица 1 – Сравнительная таблица

Функционал	Консольное управление	Virt-manager
Интерфейс	Командная строка (текстовый)	Графический интерфейс (GUI)
Автоматизация	С помощью скриптов	Ограниченные возможности
Создание виртуальной машины	Необходимо знание параметров virt-install	Более упрощенное создание
Возможности	Полный доступ ко всем функциям KVM через libvirt	Частичный доступ к функциям, ограничен GUI
Удобство использования	Требует знания команд и синтаксиса	Более простой в освоении



Удаленное управление	Идеально подходит	Менее подходит (требует X-сервер или VNC)
Сетевое управление	Сложнее, управление через virsh и nmcli	Визуальное управление, проще в использовании
Мониторинг	Требует использования других утилит	Встроенный мониторинг ресурсов виртуальных машин
Предназначение	Опытные пользователи, серверное окружение, автоматизация	Начинающие пользователи, рабочие станции

ЗАКЛЮЧЕНИЕ

В данной статье мы разобрали функционал KVM, определили что такое консольное и virt-manager управление, их назначение и различия друг с другом. Выбор между консольным управлением и Virt-Manager зависит от конкретных требований и предпочтений пользователя.

Консольное управление обеспечивает большую гибкость и возможность автоматизации, в то время как Virt-Manager упрощает управление и предоставляет визуальное представление. Сочетание этих двух подходов может обеспечить наилучший результат в зависимости от конкретной ситуации.

Список литературы:

1. Уймин, А. Г. Демонстрационный экзамен базового уровня. Сетевое и системное администрирование : Практикум. Учебное пособие для вузов / А. Г. Уймин. – Санкт-Петербург : Издательство "Лань", 2024. – 116 с. – (Высшее образование). – ISBN 978-5-507-48647-2. – EDN BZJRIQ.
2. [Электронный ресурс] // Настройка KVM на Ubuntu Server. : [сайт]. — URL: <https://www.dmosk.ru> (дата обращения: 28.12.2024).
3. [Электронный ресурс] // Операционная система Альт 8 СП (ОС Альт 8 СП). Руководство администратора. Виртуализация. : [сайт]. — URL: <https://www.basealt.ru> (дата обращения: 26.12.2024).
4. [Электронный ресурс] // Сравнение систем виртуализации - VMWARE WORKSTATION, KVM, HYPER-V : [сайт]. — URL: <https://www.elibrary.ru> (дата обращения: 25.12.2024).



5. [Электронный ресурс] // How to Manage KVM Virtual Environment using Command Line Tools in Linux. : [сайт]. — URL: <https://www.geeksforgeeks.org> (дата обращения: 29.12.2024).
6. [Электронный ресурс] // Libvirt (Qemu+KVM+Virt-manager) : [сайт]. — URL: <https://www.altlinux.org> (дата обращения: 27.12.2024).
7. [Электронный ресурс] // Virt-Manager: графическая консоль управления KVM : [сайт]. — URL: <https://winitpro.ru> (дата обращения: 29.12.2024).