

# KVM (Kernel Virtual Machine)

## Установка KVM на компьютер

Вам потребуется

- Архив `kvm-release.tar.gz`
- Процессор Intel с поддержкой VT (Virtualization Technology), или процессор AMD с поддержкой SVM (Secure Virtual Machine)
- Зависимости qemu:
  - gcc 3.x
  - SDL
  - zlib
  - alsa
  - libuuid (из `e2fsprogs-devel`)
  - код ядра (в Fedora, пакет `kernel-devel`)

Установка:

1. Распакуйте компоненты kvm:
2. Если вы используете модифицированное ядро, (например последние -mm ядра), настройте его как обычно, загрузитесь, выполните:

```
3. tar xzf kvm-release.tar.gz
4. cd kvm-release
5. ./configure --prefix=/usr/local/kvm --with-patched-kernel
6. make
7. sudo make install
8. sudo /sbin/modprobe kvm-intel
9. # or: sudo /sbin/modprobe kvm-amd
```

10.

11. Если у вас немодифицированное ядро:

```
12. tar xzf kvm-release.tar.gz
13. cd kvm-release
14. ./configure --prefix=/usr/local/kvm
15. make
16. sudo make install
17. sudo /sbin/modprobe kvm-intel
18. # or: sudo /sbin/modprobe kvm-amd
```

19.

## 2. Создайте образ диска:

```
• /usr/local/kvm/bin/qemu-img create -f qcow vdisk.img 10G
```

•

## 3. Установите ОС:

```
• sudo /usr/local/kvm/bin/qemu-system-x86_64 \  
• . -hda vdisk.img \  
• -cdrom /path/to/boot-media.iso \  
• -boot d \ -m 384
```

- (используйте `qemu` вместо `qemu-system-x86_64` для i386)  
Если у вас менее 1Гб памяти, вместо `-m 384` (что выделяет 384 Мб памяти виртуальной машине), используйте `-m 192`, или `-m 128` (что равнозначно опущенному `-m`)

## 4. после установки запустите машину:

```
• sudo /usr/local/kvm/bin/qemu-system-x86_64 \  
• . -hda vdisk.img \ -boot c \ -m 384
```

# KVM на Debian Sid

|

Для установки и запуска KVM на Debian, следуйте следующей инструкции:

### 1. выполните как root:

```
• # apt-get update  
• # apt-get install kvm kvm-source qemu module-assistant  
• # m-a a-i kvm
```

•

В зависимости от процессора выполните:

```
• # modprobe kvm-amd
```

или

```
• # modprobe kvm-intel
```

2. Установка kvm создает новую группу kvm в /etc/group. Доавьте пользователей, которым вы хотите разрешить запускать KVM в эту группу:

```
• # adduser username kvm
```

3. Перезайдите в систему для активизации новых прав. Нижеследующие команды могут быть выполнены любым пользователем в группе kvm.

4. Создайте виртуальный диск:

```
• $ qemu-img create vdisk.qcow 10G
```

•

5. Установите ОС

```
• $ kvm \  
• -hda vdisk.qcow \  
• -cdrom /path/to/boot-media.iso \  
• -boot d \  
• -m 384
```

• Если вы устанавливаете Windows, поставьте `-no-acpi`.

6. После установки запустите:

```
• $ kvm \  
• -hda vdisk.img \  
• -m 384
```

•

7. Читайте документацию:

```
• $ man kvm
```

# Бинарные пакеты openSUSE

1. Добавьте

`http://software.opensuse.org/download/home:/kraxel/SUSE_Linux_10.1/` в репозитории `yast`

2. Используйте `yast` для установки `kvm` и `kvm-kmp` RPM

3. Создайте диск:

```
4. qemu-img create -f qcow vdisk.img 10G
```

5. Установите ОС:

```
6. sudo qemu-kvm-x86_64 \  
7. -hda vdisk.img \  
8. -cdrom /path/to/boot-media.iso \  
9. -boot d \  
10. -m 384
```

Если Вы устанавливаете Windows, используйте `-no-acpi`.

11. Запустите ОС после инсталляции:

```
12. sudo qemu-kvm-x86_64 \  
13. -hda vdisk.img \  
14. -boot c \  
15. -m 384
```

16. Для загрузки модуля `kvm` при старте системы исправьте `/etc/sysconfig/kernel` и добавьте `kvm` в строку `MODULES_LOADED_ON_BOOT="..."`. Или используйте `yast2 sysconfig`.

17. Включите в группу `kvm` пользователей, которым разрешен доступ к KVM.

# KVM: миграция VM

KVM поддерживает миграцию VM

Команды миграции предоставляются через qemu-monitor (Alt-Ctrl-2).

После успешной миграции, VM продолжит выполняться на целевом узле.

## Примечание

32-разрядные VM могут быть мигрированы между разнородными узлами на базе Intel и AMD. 64-разрядные VM могут мигрировать только между однотипными узлами.

С устаревшими процессорами, не поддерживающими NX (XD) могут возникать проблемы.

## *Пользовательский интерфейс*

### Управление

```
migrate [-d] <URI>  
migrate_cancel
```

Где '-d' позволяет узнать состояние миграции.

Без '-d' монитор вернет управление только по завершении миграции. URI может быть 'exec:<command>', 'ssh://<ip>' or tcp://<ip>:port

### Состояние

```
info migration
```

### Параметры

```
migrate_set_speed <speed>    set bandwidth control parameters (max transfer  
rate per second)
```

## Примеры

- А – хост-источник, В –целевой хост:

- TCP:

```
1. A: migrate -d tcp://B:4444 (or other PORT)
```

```
2. проверим состояние:
```

```
3. info migration
```

- SSH:

- Не рекомендуется использовать SDL-окно (графический режим) и SSH совместно.

```
A: migrate ssh://B
```

- Заморозка/Разморозка:

```
1. выделим полную полосу пропускания:
```

```
2. A: migrate_set_speed 1g
```

```
3. остановим VM:
```

```
4. A: stop
```

```
5. выполним TCP/SSH миграцию.
```

## ***savevm/loadvm (псевдомиграция)***

- savevm

```
○ stop  
○ migrate "exec:dd of=STATEFILE"
```

○

- loadvm

```
○ <qemu-command-line> -incoming stdio < STATEFILE  
○ cont
```

○