

OTUS

Онлайн-образование

**Не забыл включить запись**

# Меня хорошо видно && слышно?

Ставьте плюсы, если все хорошо  
Напишите в чат, если есть проблемы

# Правила вебинара

- Активно участвуем
- Задаем вопросы в чат или голосом
- Off-topic обсуждаем в Slack #канал группы или #general
- Вопросы вижу в чате, могу ответить не сразу

LVM

# Маршрут вебинара

- LVM теория
- LVM практика

# Маршрут вебинара

- LVM теория
- LVM практика

# Маршрут вебинара

- LVM теория
- LVM практика



# Цели занятия

# Цели занятия

**После занятия вы сможете:**

# Цели занятия

## После занятия вы сможете:

1. Понять что такое LVM, познакомиться с терминологией

# Цели занятия

## После занятия вы сможете:

1. Понять что такое LVM, познакомиться с терминологией
2. Познакомиться с практикой применения LVM

# Смысл

# Смысл

**Зачем вам это уметь:**

# Смысл

## Зачем вам это уметь:

1. Чтобы понимать как работает LVM

# Смысл

## Зачем вам это уметь:

1. Чтобы понимать как работает LVM
2. Чтобы использовать LVM в необходимых случаях



LVM

# LVM

**LVM (Logical Volume Manager)** - специальная подсистема ядра, которая добавляет дополнительный уровень абстракции от "железа", позволяя гибко управлять дисковым пространством

# LVM

LVM решает разнообразные задачи по управлению дисковой подсистемой:

LVM решает разнообразные задачи по управлению дисковой подсистемой:

- Группировка физических томов

LVM решает разнообразные задачи по управлению дисковой подсистемой:

- Группировка физических томов
- Создание и изменение логических томов (на лету)

LVM решает разнообразные задачи по управлению дисковой подсистемой:

- Группировка физических томов
- Создание и изменение логических томов (на лету)
- Snapshots

LVM решает разнообразные задачи по управлению дисковой подсистемой:

- Группировка физических томов
- Создание и изменение логических томов (на лету)
- Snapshots
- Thin provisioning

LVM решает разнообразные задачи по управлению дисковой подсистемой:

- Группировка физических томов
- Создание и изменение логических томов (на лету)
- Snapshots
- Thin provisioning
- Cache volumes



LVM решает разнообразные задачи по управлению дисковой подсистемой:

- Группировка физических томов
- Создание и изменение логических томов (на лету)
- Snapshots
- Thin provisioning
- Cache volumes
- LVM mirrors

LVM решает разнообразные задачи по управлению дисковой подсистемой:

- Группировка физических томов
- Создание и изменение логических томов (на лету)
- Snapshots
- Thin provisioning
- Cache volumes
- LVM mirrors
- LVM stripes

# Device-mapper

**Device mapper (dm)** - модуль ядра, позволяющий работать с виртуальными блочными устройствами (логическими дисками). По сути посылает информацию с виртуального устройства на реальное.

# Device-mapper

Некоторые из его возможностей:

# Device-mapper

Некоторые из его возможностей:

- Кэширование

# Device-mapper

Некоторые из его возможностей:

- Кэширование
- Шифрование

# Device-mapper

Некоторые из его возможностей:

- Кэширование
- Шифрование
- Зеркалирование (mirror drives)

# Device-mapper

Некоторые из его возможностей:

- Кэширование
- Шифрование
- Зеркалирование (mirror drives)
- Multipath



# Device-mapper

Некоторые из его возможностей:

- Кэширование
- Шифрование
- Зеркалирование (mirror drives)
- Multipath
- RAID (mdadm)

# Device-mapper

Некоторые из его возможностей:

- Кэширование
- Шифрование
- Зеркалирование (mirror drives)
- Multipath
- RAID (mdadm)
- Thin-provision

# Device-mapper

Некоторые из его возможностей:

- Кэширование
- Шифрование
- Зеркалирование (mirror drives)
- Multipath
- RAID (mdadm)
- Thin-provision
- Stripe

# Device-mapper

Некоторые из его возможностей:

- Кэширование
- Шифрование
- Зеркалирование (mirror drives)
- Multipath
- RAID (mdadm)
- Thin-provision
- Stripe
- Snapshot

# Практика применения LVM

Популярные use case для LVM:

# Практика применения LVM

Популярные use case для LVM:

- Изменение размеров томов (в любую сторону)

# Практика применения LVM

Популярные use case для LVM:

- Изменение размеров томов (в любую сторону)
- Управление большим количеством дисков (hot swap)

# Практика применения LVM

## Популярные use case для LVM:

- Изменение размеров томов (в любую сторону)
- Управление большим количеством дисков (hot swap)
- Использование на десктопах



# Практика применения LVM

## Популярные use case для LVM:

- Изменение размеров томов (в любую сторону)
- Управление большим количеством дисков (hot swap)
- Использование на десктопах
- Отказоустойчивость

# Практика применения LVM

## Популярные use case для LVM:

- Изменение размеров томов (в любую сторону)
- Управление большим количеством дисков (hot swap)
- Использование на десктопах
- Отказоустойчивость
- High availability кластер

# Структура LVM

Основные элементы структуры LVM:

# Структура LVM

## Основные элементы структуры LVM:

- PV (physical volume) - физический том, по сути любое блочное устройство

# Структура LVM

## Основные элементы структуры LVM:

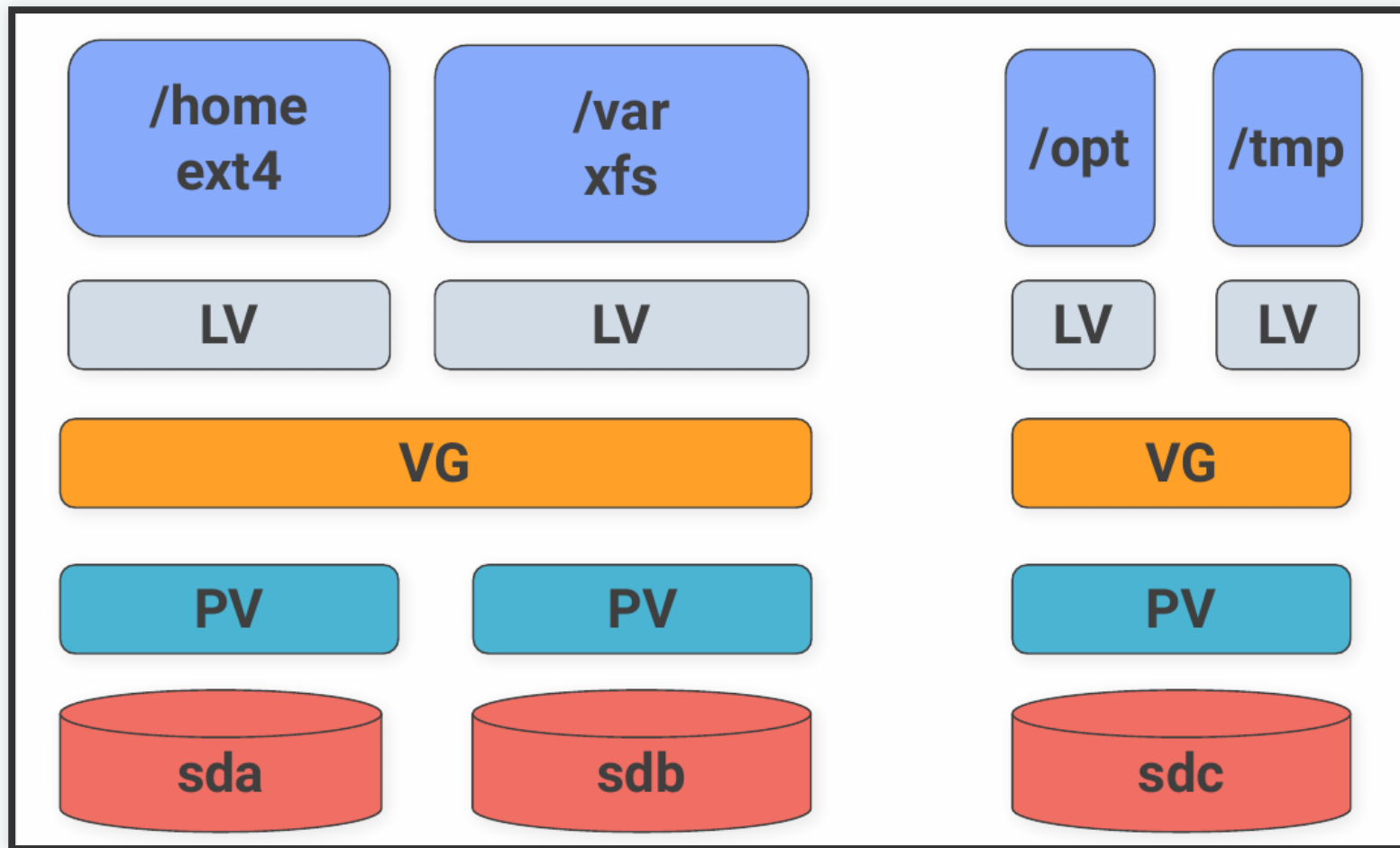
- PV (physical volume) - физический том, по сути любое блочное устройство
- VG (volume group) - группа физических томов

# Структура LVM

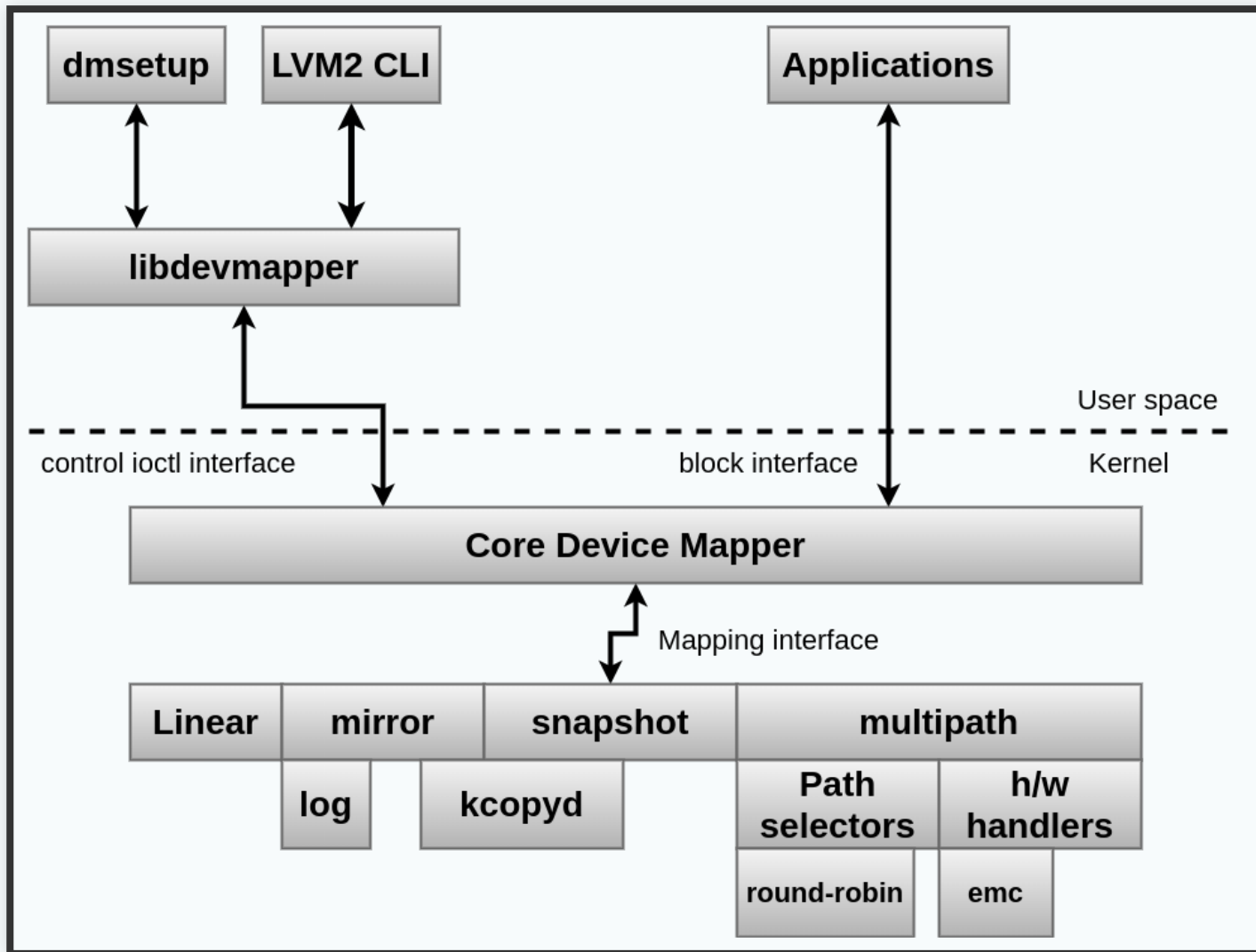
## Основные элементы структуры LVM:

- PV (physical volume) - физический том, по сути любое блочное устройство
- VG (volume group) - группа физических томов
- LV (logical volume) - часть VG, доступная в виде блочного устройства

# Структура LVM

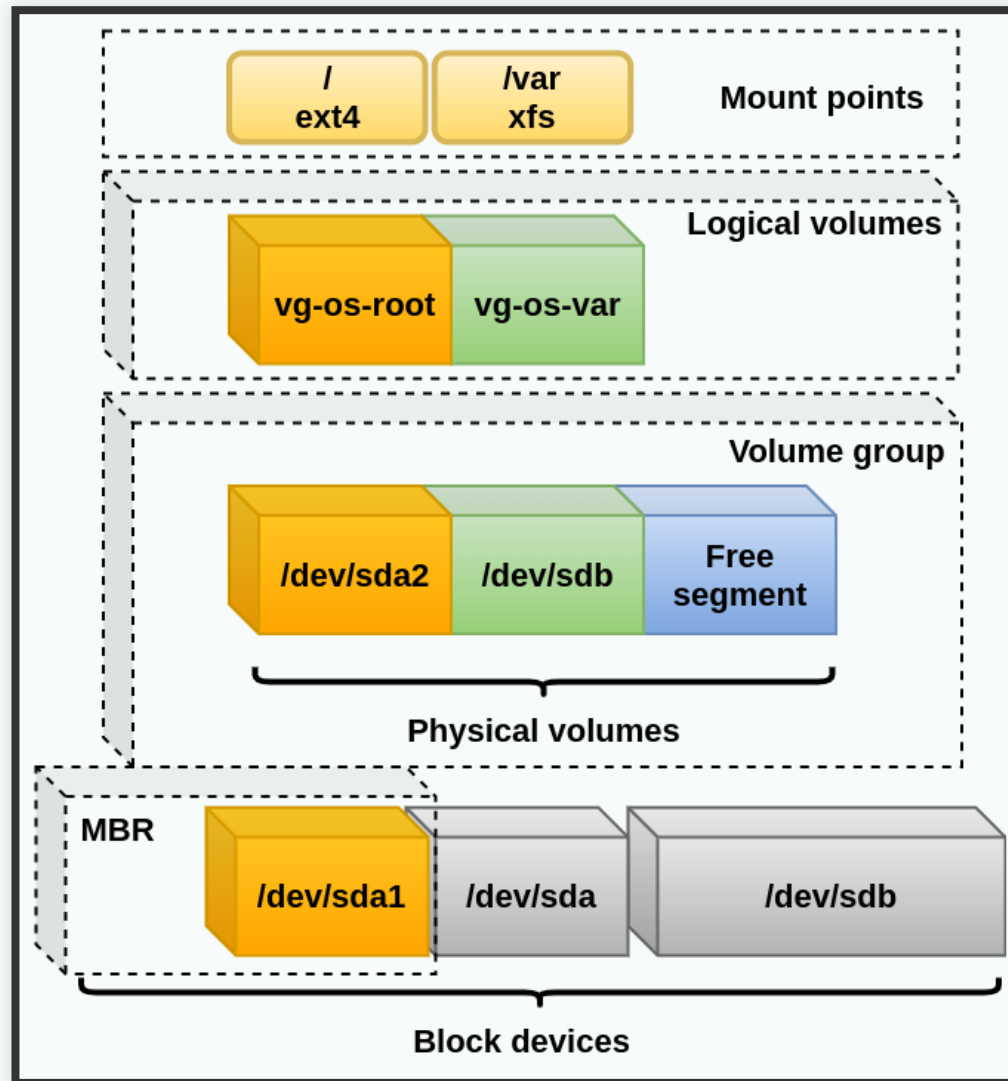


# Архитектура LVM





# Пример использования LVM на VM



Ваши вопросы?

# Управление и конфигурирование LVM

# Управление и конфигурирование LVM

- Для управления LVM необходим пакет lvm2

# Управление и конфигурирование LVM

- Для управления LVM необходим пакет lvm2
- /usr/sbin/lvm - основная утилита, все остальные - лишь симлинки на нее

# Управление и конфигурирование LVM

- Для управления LVM необходим пакет `lvm2`
- `/usr/sbin/lvm` - основная утилита, все остальные - лишь симлинки на нее

Создаем `physical volume` на блочном устройстве:

# Управление и конфигурирование LVM

- Для управления LVM необходим пакет lvm2
- /usr/sbin/lvm - основная утилита, все остальные - лишь симлинки на нее

Создаем physical volume на блочном устройстве:

```
[root@otus ~]# pvcreate /dev/sdb
```

# Управление и конфигурирование LVM

- Для управления LVM необходим пакет lvm2
- /usr/sbin/lvm - основная утилита, все остальные - лишь симлинки на нее

Создаем physical volume на блочном устройстве:

```
[root@otus ~]# pvcreate /dev/sdb
```

Создаем volume group на блочном устройстве:



# Управление и конфигурирование LVM

- Для управления LVM необходим пакет lvm2
- /usr/sbin/lvm - основная утилита, все остальные - лишь симлинки на нее

Создаем physical volume на блочном устройстве:

```
[root@otus ~]# pvcreate /dev/sdb
```

Создаем volume group на блочном устройстве:

```
[root@otus ~]# vgcreate vg0 /dev/sdb
```

# Управление и конфигурирование LVM

# Управление и конфигурирование LVM

Создаем дополнительный physical volume:

# Управление и конфигурирование LVM

Создаем дополнительный physical volume:

```
[root@otus ~]# pvcreate /dev/sdc
```

# Управление и конфигурирование LVM

Создаем дополнительный physical volume:

```
[root@otus ~]# pvcreate /dev/sdc
```

Расширяем volume group на новый physical volume:

# Управление и конфигурирование LVM

Создаем дополнительный physical volume:

```
[root@otus ~]# pvcreate /dev/sdc
```

Расширяем volume group на новый physical volume:

```
[root@otus ~]# vgextend vg0 /dev/sdc
```

# Управление и конфигурирование LVM

Создаем дополнительный physical volume:

```
[root@otus ~]# pvcreate /dev/sdc
```

Расширяем volume group на новый physical volume:

```
[root@otus ~]# vgextend vg0 /dev/sdc
```

Вариант создания logical volume размером 10 экстентов по 4М:

# Управление и конфигурирование LVM

Создаем дополнительный physical volume:

```
[root@otus ~]# pvcreate /dev/sdc
```

Расширяем volume group на новый physical volume:

```
[root@otus ~]# vgextend vg0 /dev/sdc
```

Вариант создания logical volume размером 10 экстентов по 4М:

```
[root@otus ~]# lvcreate -l 10 -n home vg0
```



# Управление и конфигурирование LVM

# Управление и конфигурирование LVM

Вариант создания logical volume размером 1 Гб:

# Управление и конфигурирование LVM

Вариант создания logical volume размером 1 Гб:

```
[root@otus ~]# lvcreate -L 1G -n root vg0
```

# Управление и конфигурирование LVM

Вариант создания logical volume размером 1 Гб:

```
[root@otus ~]# lvcreate -L 1G -n root vg0
```

Создаем logical volume на 100% свободного пространства:

# Управление и конфигурирование LVM

Вариант создания logical volume размером 1 Гб:

```
[root@otus ~]# lvcreate -L 1G -n root vg0
```

Создаем logical volume на 100% свободного пространства:

```
[root@otus ~]# lvcreate -l 100%FREE -n var vg0
```

# Управление и конфигурирование LVM

# Управление и конфигурирование LVM

Сканирование блочных устройств на предмет  
наличия LVM:

# Управление и конфигурирование LVM

Сканирование блочных устройств на предмет наличия LVM:

```
[root@otus ~]# vgscan
```



# Управление и конфигурирование LVM

Сканирование блочных устройств на предмет наличия LVM:

```
[root@otus ~]# vgscan
```

Сделать volume group неактивной:

# Управление и конфигурирование LVM

Сканирование блочных устройств на предмет наличия LVM:

```
[root@otus ~]# vgscan
```

Сделать volume group неактивной:

```
[root@otus ~]# vgchange -an
```

# Управление и конфигурирование LVM

Сканирование блочных устройств на предмет наличия LVM:

```
[root@otus ~]# vgscan
```

Сделать volume group неактивной:

```
[root@otus ~]# vgchange -an
```

Альтернативные команды для просмотра состояний элементов LVM:

# Управление и конфигурирование LVM

Сканирование блочных устройств на предмет наличия LVM:

```
[root@otus ~]# vgscan
```

Сделать volume group неактивной:

```
[root@otus ~]# vgchange -an
```

Альтернативные команды для просмотра состояний элементов LVM:

```
[root@otus ~]# pvs | vgs | lvs
```

# Управление и конфигурирование LVM

# Управление и конфигурирование LVM

Удаление элементов LVM:

# Управление и конфигурирование LVM

Удаление элементов LVM:

```
[root@otus ~]# pvremove | vgremove | lvremove
```

# Конфигурация LVM

Основные элементы конфигурации LVM:



# Конфигурация LVM

## Основные элементы конфигурации LVM:

- `/etc/lvm/` - хранилище кэша конфигурации и резервных копий

# Конфигурация LVM

## Основные элементы конфигурации LVM:

- `/etc/lvm/` - хранилище кэша конфигурации и резервных копий
- `/etc/lvm/lvm.conf` - основная конфигурация для утилиты `lvm`

# Конфигурация LVM

## Основные элементы конфигурации LVM:

- `/etc/lvm/` - хранилище кэша конфигурации и резервных копий
- `/etc/lvm/lvm.conf` - основная конфигурация для утилиты `lvm`
- `archive/` - информация за все время

# Конфигурация LVM

## Основные элементы конфигурации LVM:

- `/etc/lvm/` - хранилище кэша конфигурации и резервных копий
- `/etc/lvm/lvm.conf` - основная конфигурация для утилиты `lvm`
- `archive/` - информация за все время
- `backup/` - бэкап текущей конфигурации

# Конфигурация LVM

## Основные элементы конфигурации LVM:

- `/etc/lvm/` - хранилище кэша конфигурации и резервных копий
- `/etc/lvm/lvm.conf` - основная конфигурация для утилиты `lvm`
- `archive/` - информация за все время
- `backup/` - бэкап текущей конфигурации
- `cache/` - кэш

# Конфигурация LVM

## Основные элементы конфигурации LVM:

- `/etc/lvm/` - хранилище кэша конфигурации и резервных копий
- `/etc/lvm/lvm.conf` - основная конфигурация для утилиты `lvm`
- `archive/` - информация за все время
- `backup/` - бэкап текущей конфигурации
- `cache/` - кэш
- `profile/` - готовые (или самописные профили)

# Конфигурация LVM

# Конфигурация LVM

Тестирование (dry-run) восстановления конфигурации LVM из бэкапа:



# Конфигурация LVM

Тестирование (dry-run) восстановления конфигурации LVM из бэкапа:

```
[root@otuslinux ~]# vgcfgrestore VolGroup00 --test -f \
/etc/lvm/archive/VolGroup00_000000-2099537038.vg
```

# Конфигурация LVM

Тестирование (dry-run) восстановления конфигурации LVM из бэкапа:

```
[root@otuslinux ~]# vgcfgrestore VolGroup00 --test -f \
/etc/lvm/archive/VolGroup00_000000-2099537038.vg
```

Восстановление конфигурации LVM из бэкапа:

# Конфигурация LVM

Тестирование (dry-run) восстановления конфигурации LVM из бэкапа:

```
[root@otuslinux ~]# vgcfgrestore VolGroup00 --test -f \
/etc/lvm/archive/VolGroup00_000000-2099537038.vg
```

Восстановление конфигурации LVM из бэкапа:

```
[root@otuslinux ~]# vgcfgrestore VolGroup00 -f \
/etc/lvm/archive/VolGroup00_000000-2099537038.vg
```

# Конфигурация LVM

Тестирование (dry-run) восстановления конфигурации LVM из бэкапа:

```
[root@otuslinux ~]# vgcfgrestore VolGroup00 --test -f \
/etc/lvm/archive/VolGroup00_00000-2099537038.vg
```

Восстановление конфигурации LVM из бэкапа:

```
[root@otuslinux ~]# vgcfgrestore VolGroup00 -f \
/etc/lvm/archive/VolGroup00_00000-2099537038.vg
```

Далее запускаем `lvscan` для валидации:

# Конфигурация LVM

Тестирование (dry-run) восстановления конфигурации LVM из бэкапа:

```
[root@otuslinux ~]# vgcfgrestore VolGroup00 --test -f \
/etc/lvm/archive/VolGroup00_000000-2099537038.vg
```

Восстановление конфигурации LVM из бэкапа:

```
[root@otuslinux ~]# vgcfgrestore VolGroup00 -f \
/etc/lvm/archive/VolGroup00_000000-2099537038.vg
```

Далее запускаем `lvscan` для валидации:

```
lvscan
```

# LVM snapshots

# LVM snapshots

Принцип действия:

# LVM snapshots

Принцип действия:

- создание нового LV



# LVM snapshots

Принцип действия:

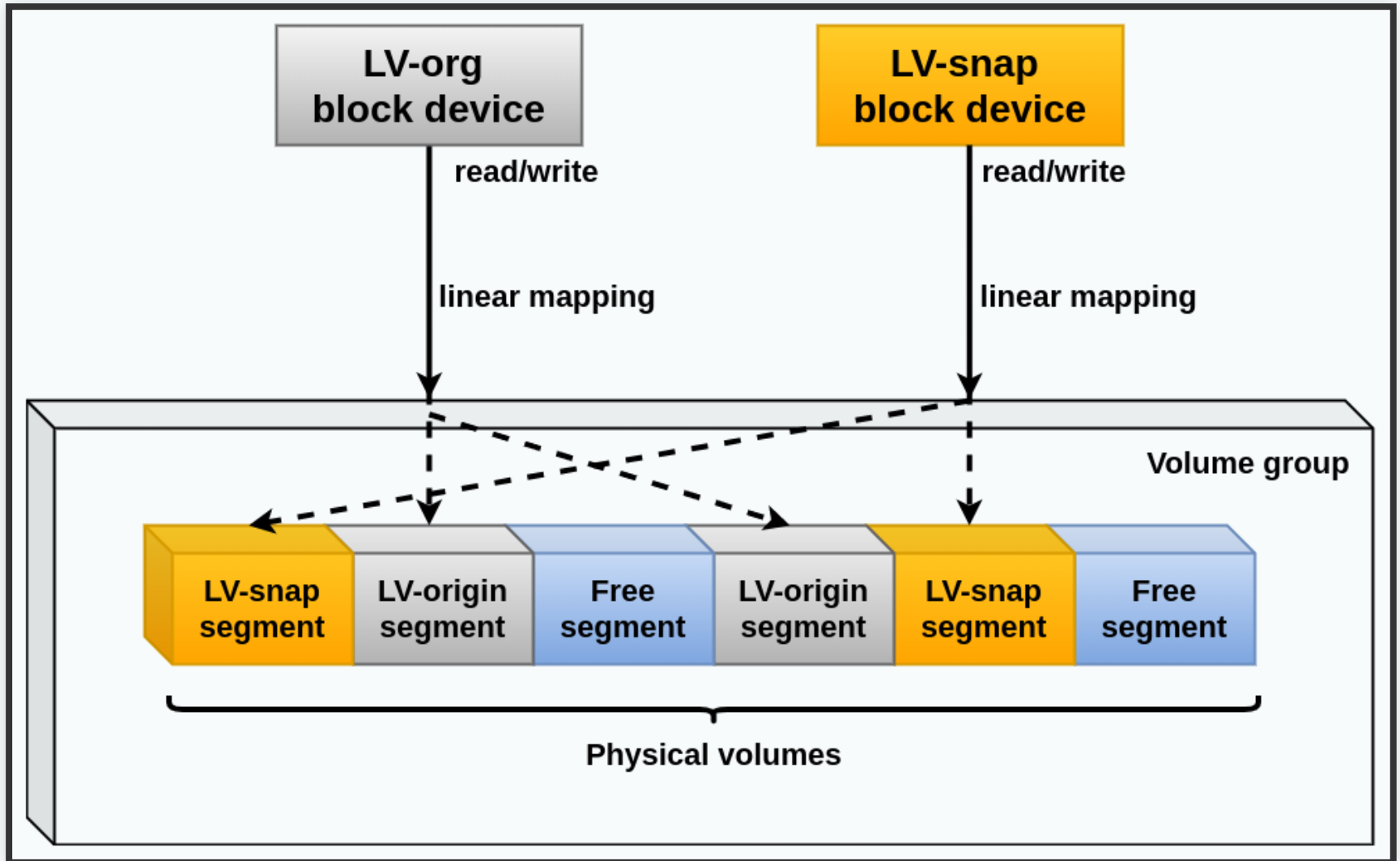
- создание нового LV
- новый LV зависит от оригинального LV

# LVM snapshots

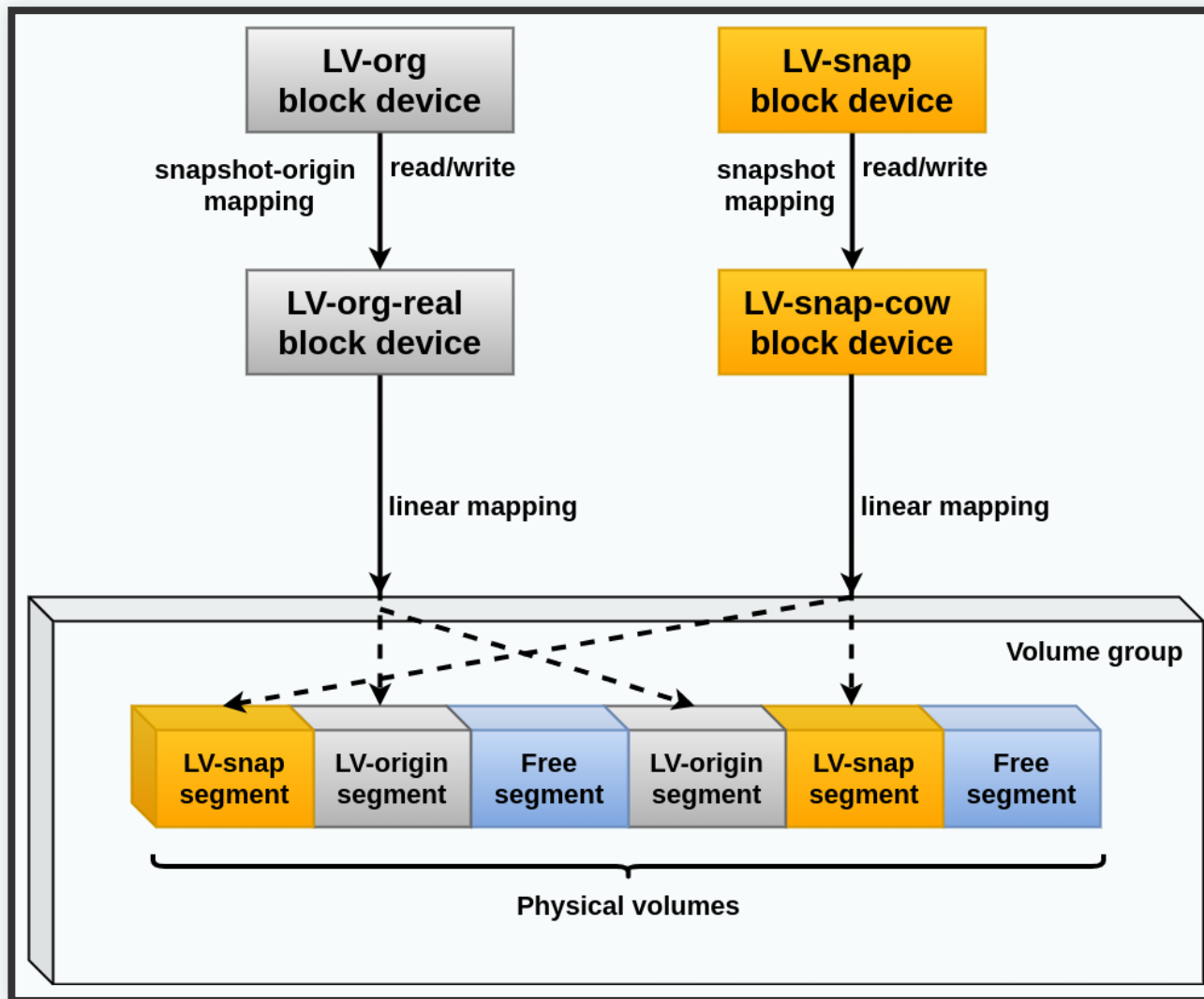
Принцип действия:

- создание нового LV
- новый LV зависит от оригинального LV
- оригинальные блоки данных копируются в новый LV перед тем, как в оригинальный LV будут записаны новые блоки данных (Copy-on-Write)

# LVM snapshots



# LVM snapshots



# LVM snapshots

# LVM snapshots

Особенности:

# LVM snapshots

## Особенности:

- необходимо наличие свободного места в VG под снапшот

# LVM snapshots

## Особенности:

- необходимо наличие свободного места в VG под снапшот
- в результате при использовании снапшотов мы получаем двойную запись и как следствие - замедление дисковых операций



# LVM snapshots

## Особенности:

- необходимо наличие свободного места в VG под снапшот
- в результате при использовании снапшотов мы получаем двойную запись и как следствие - замедление дисковых операций
- удаление снапшота - быстрый процесс

# LVM snapshots

## Особенности:

- необходимо наличие свободного места в VG под снапшот
- в результате при использовании снапшотов мы получаем двойную запись и как следствие - замедление дисковых операций
- удаление снапшота - быстрый процесс
- откат на снапшот - медленный процесс

# LVM snapshots

## Особенности:

- необходимо наличие свободного места в VG под снапшот
- в результате при использовании снапшотов мы получаем двойную запись и как следствие - замедление дисковых операций
- удаление снапшота - быстрый процесс
- откат на снапшот - медленный процесс
- снапшот можно монтировать, в том числе в RW режиме

# LVM snapshots

# LVM snapshots

Создание снапшота:

# LVM snapshots

Создание снапшота:

```
[root@otus ~]# lvcreate -L 500M -s -n test-snap /dev/otus/test
```

# LVM snapshots

Создание снимка:

```
[root@otus ~]# lvcreate -L 500M -s -n test-snap /dev/otus/test
```

Слить (смержить) снимок с оригинальным LV:

# LVM snapshots

Создание снимка:

```
[root@otus ~]# lvcreate -L 500M -s -n test-snap /dev/otus/test
```

Слить (смержить) снимок с оригинальным LV:

```
[root@otus ~]# lvconvert --merge /dev/otus/test-snap
```



# LVM snapshots

Создание снимка:

```
[root@otus ~]# lvcreate -L 500M -s -n test-snap /dev/otus/test
```

Слить (смержить) снимок с оригинальным LV:

```
[root@otus ~]# lvconvert --merge /dev/otus/test-snap
```

Удаление снимка:

# LVM snapshots

Создание снимка:

```
[root@otus ~]# lvcreate -L 500M -s -n test-snap /dev/otus/test
```

Слить (смержить) снимок с оригинальным LV:

```
[root@otus ~]# lvconvert --merge /dev/otus/test-snap
```

Удаление снимка:

```
[root@otus ~]# lvremove /dev/otus/test-snap
```

# LVM snapshots

# LVM snapshots

Примерный сценарий бэкапа базы данных с помощью LVM snapshot'a:

# LVM snapshots

Примерный сценарий бэкапа базы данных с помощью LVM snapshot`a:

- Сбросить данные в таблицах базы данных на диск с блокировкой таблиц на запись:

# LVM snapshots

Примерный сценарий бэкапа базы данных с помощью LVM snapshot`a:

- Сбросить данные в таблицах базы данных на диск с блокировкой таблиц на запись:

```
mysql> flush tables read lock;
```

# LVM snapshots

Примерный сценарий бэкапа базы данных с помощью LVM snapshot`a:

- Сбросить данные в таблицах базы данных на диск с блокировкой таблиц на запись:

```
mysql> flush tables read lock;
```

- Создать LVM снапшот раздела, на котором расположены файлы базы:

# LVM snapshots

Примерный сценарий бэкапа базы данных с помощью LVM snapshot`a:

- Сбросить данные в таблицах базы данных на диск с блокировкой таблиц на запись:

```
mysql> flush tables read lock;
```

- Создать LVM снапшот раздела, на котором расположены файлы базы:

```
lvcreate -l100%FREE -s -n mysql-backup /dev/vg0/var
```



# LVM snapshots

# LVM snapshots

- Разблокировать таблицы в базе данных:

# LVM snapshots

- Разблокировать таблицы в базе данных:

```
mysql> unlock tables;
```

# LVM snapshots

- Разблокировать таблицы в базе данных:

```
mysql> unlock tables;
```

- Обнаружить новый снапшот:

# LVM snapshots

- Разблокировать таблицы в базе данных:

```
mysql> unlock tables;
```

- Обнаружить новый снапшот:

```
lvscan
```

# LVM snapshots

- Разблокировать таблицы в базе данных:

```
mysql> unlock tables;
```

- Обнаружить новый снапшот:

```
lvscan
```

- Примонтировать LVM снапшота как обычную файловую систему:

# LVM snapshots

- Разблокировать таблицы в базе данных:

```
mysql> unlock tables;
```

- Обнаружить новый снапшот:

```
lvscan
```

- Примонтировать LVM снапшота как обычную файловую систему:

```
mkdir -p /mnt/snapshot  
mount /dev/vg0/mysql-backup /mnt/snapshot
```

# LVM Thin Provision



# LVM Thin Provision

Overbooking для LVM:

# LVM Thin Provision

## Overbooking для LVM:

- возможность выделить места больше, чем есть

# LVM Thin Provision

## Overbooking для LVM:

- возможность выделить места больше, чем есть
- используется в виртуализации и контейнеризации

# LVM Thin Provision

## Overbooking для LVM:

- возможность выделить места больше, чем есть
- используется в виртуализации и контейнеризации

## Принцип:

# LVM Thin Provision

## Overbooking для LVM:

- возможность выделить места больше, чем есть
- используется в виртуализации и контейнеризации

## Принцип:

- создается основной LV (thin pool)

# LVM Thin Provision

## Overbooking для LVM:

- возможность выделить места больше, чем есть
- используется в виртуализации и контейнеризации

## Принцип:

- создается основной LV (thin pool)
- создаются зависимые LV с указанием виртуального размера

# LVM Thin Provision

# LVM Thin Provision

Основные команды:



# LVM Thin Provision

Основные команды:

Создаем основной LV (thin pool):

# LVM Thin Provision

Основные команды:

Создаем основной LV (thin pool):

```
[root@otus ~]# lvcreate -L 100G -T vg0/lv-thinpool
```

# LVM Thin Provision

Основные команды:

Создаем основной LV (thin pool):

```
[root@otus ~]# lvcreate -L 100G -T vg0/lv-thinpool
```

Создаем зависимые LV:

# LVM Thin Provision

Основные команды:

Создаем основной LV (thin pool):

```
[root@otus ~]# lvcreate -L 100G -T vg0/lv-thinpool
```

Создаем зависимые LV:

```
[root@otus ~]# lvcreate -V 100G -T vg0/lv-thinpool -n lv1  
[root@otus ~]# lvcreate -V 100G -T vg0/lv-thinpool -n lv2  
[root@otus ~]# lvcreate -V 100G -T vg0/lv-thinpool -n lv3
```

# LVM cache

# LVM cache

Принцип действия кэша LVM - вынос часто используемых данных на SSD

# LVM cache

Принцип действия кэша LVM - вынос часто используемых данных на SSD

- в основном используется на десктопах

# LVM cache

Принцип действия кэша LVM - вынос часто используемых данных на SSD

- в основном используется на десктопах
- суть в том, что добавляется кэш LV



# LVM cache

Принцип действия кэша LVM - вынос часто используемых данных на SSD

- в основном используется на десктопах
- суть в том, что добавляется кэш LV
- в случае использования кэша нельзя использовать снапшоты

# LVM cache

Принцип действия кэша LVM - вынос часто используемых данных на SSD

- в основном используется на десктопах
- суть в том, что добавляется кэш LV
- в случае использования кэша нельзя использовать снапшоты

Документация по кэшу LVM:

# LVM cache

Принцип действия кэша LVM - вынос часто используемых данных на SSD

- в основном используется на десктопах
- суть в том, что добавляется кэш LV
- в случае использования кэша нельзя использовать снапшоты

Документация по кэшу LVM:

<http://man7.org/linux/man-pages/man7/lvmcache.7.html>

# LVM перенос данных

# LVM перенос данных

Перенос данных с LVM представляет собой процедуру переноса использующихся физических ЭКСТЕНТОВ

# LVM перенос данных

Перенос данных с LVM представляет собой процедуру переноса использующихся физических экстентов

Особенности:

# LVM перенос данных

Перенос данных с LVM представляет собой процедуру переноса использующихся физических экстенентов

Особенности:

- для перемещения данных в активной системе используется команда `pvmove`

# LVM перенос данных

Перенос данных с LVM представляет собой процедуру переноса использующихся физических экстентов

Особенности:

- для перемещения данных в активной системе используется команда `pvmove`
- `pvmove` разбивает данные на секции и создает временное зеркало для переноса каждой секции



# LVM перенос данных

# LVM перенос данных

## Пример

# LVM перенос данных

## Пример

Перенос всех данных с физического тома /dev/sdc на другие физические тома в группе:

# LVM перенос данных

## Пример

Перенос всех данных с физического тома /dev/sdc на другие физические тома в группе:

```
pvmove -b /dev/sdc
```

Ваши вопросы?

# Рефлексия

- Назовите пожалуйста 3 момента, которые вам запомнились в процессе занятия
- Что вы будете применять в работе из сегодняшнего вебинара?

Заполните, пожалуйста,  
опрос о занятии по  
ссылке в чате

**Приходите на следующие вебинары**

**Спасибо за внимание!**



