



RUDN  
university

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук  
Кафедра прикладной информатики и теории вероятностей

# Презентация №16

RAID на базе mdadm в Linux

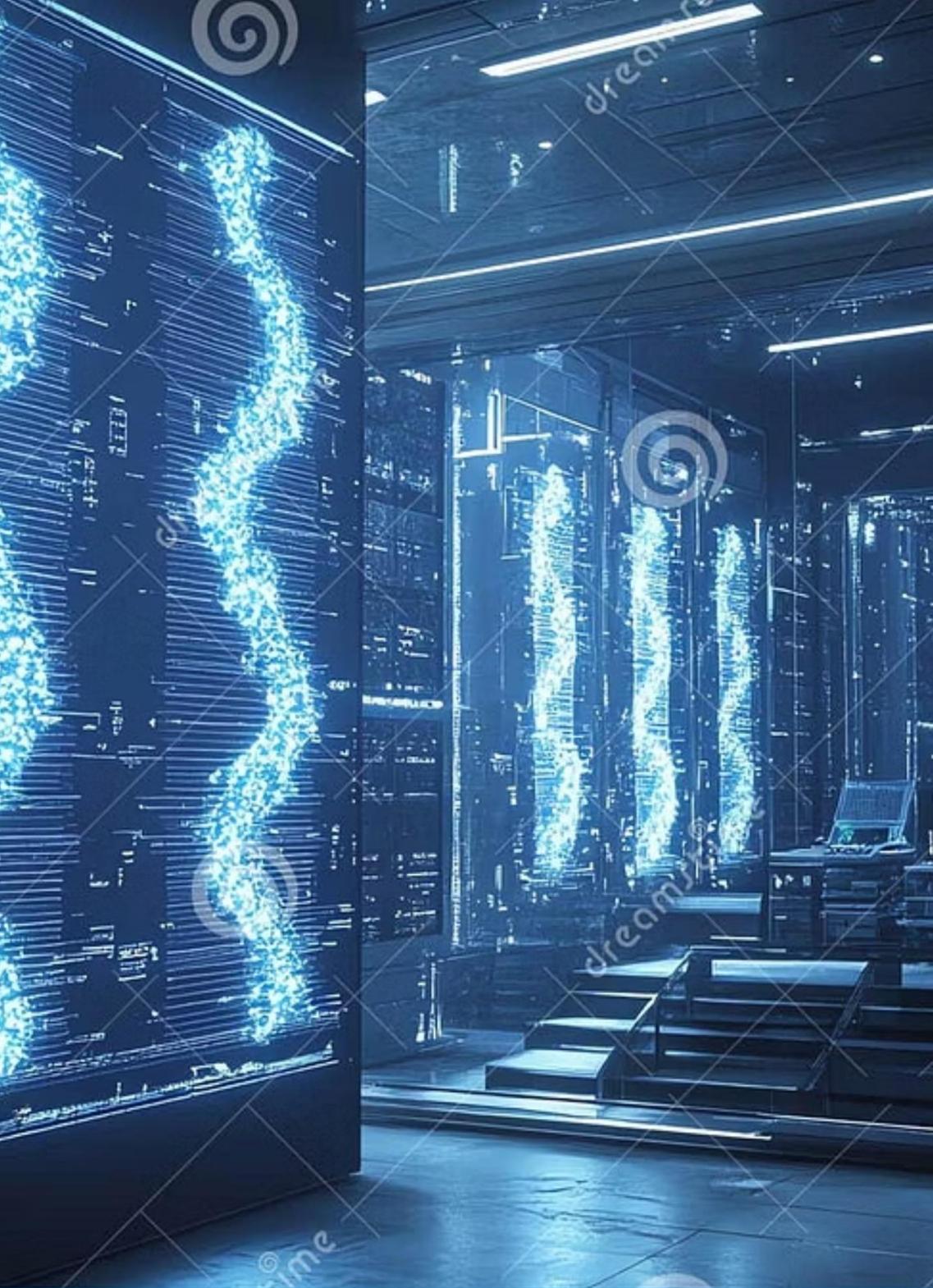
Студент: Эйвази Мани

Группа: НПИбд-03-24

Студенческий билет №: 1032245107

# RAID на базе mdadm в Linux

Программный RAID для отказоустойчивости и производительности



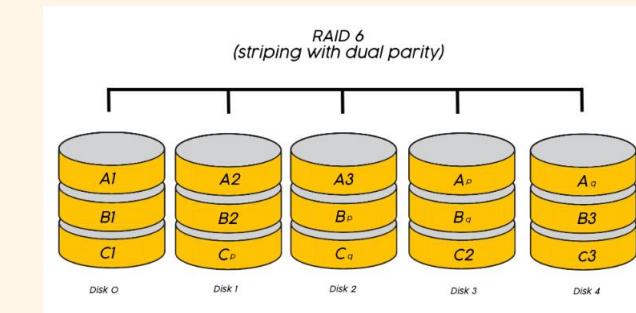
## Глава 1

# Концепция RAID: Аппаратный vs Программный



## Аппаратный RAID

Реализован на отдельном контроллере, который разгружает центральный процессор.  
Обеспечивает высокую производительность и независимость от ОС, но требует специализированного оборудования.



## Программный RAID (mdadm)

Реализуется на уровне ядра операционной системы Linux с помощью модуля **md (Multiple Devices)** и утилиты **mdadm**. Более гибкий, экономичный и не зависит от конкретного оборудования.

## Глава 2

# mdadm: Центральный инструмент управления



### Создание

*Инициализация новых RAID-массивов различных уровней с указанием устройств и параметров.*



### Сборка

*Автоматическое или ручное подключение массивов при загрузке системы.*



### Мониторинг

*Наблюдение за состоянием массива, оповещение о неисправностях и деградации.*



### Управление

*Добавление/удаление дисков, управление горячим резервом, замена отказавших устройств.*

## Глава 3

# Базовые уровни RAID и их применение



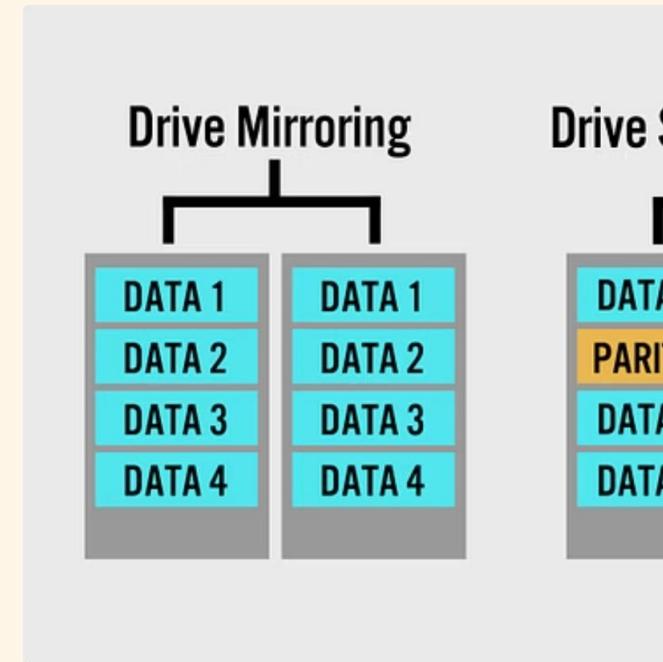
### RAID 0 (Striping)

*Высокая производительность, но нет отказоустойчивости. Данные распределяются по всем дискам.*



### RAID 1 (Mirroring)

*Высокая отказоустойчивость за счет полного дублирования данных. Производительность записи снижается.*



### RAID 5 / RAID 6 (Четность)

*Баланс между производительностью, емкостью и отказоустойчивостью. Используют данные четности для восстановления.*



### RAID 10 (Striped Mirrors)

*Сочетает производительность RAID 0 с отказоустойчивостью RAID 1. Требует минимум 4 диска.*

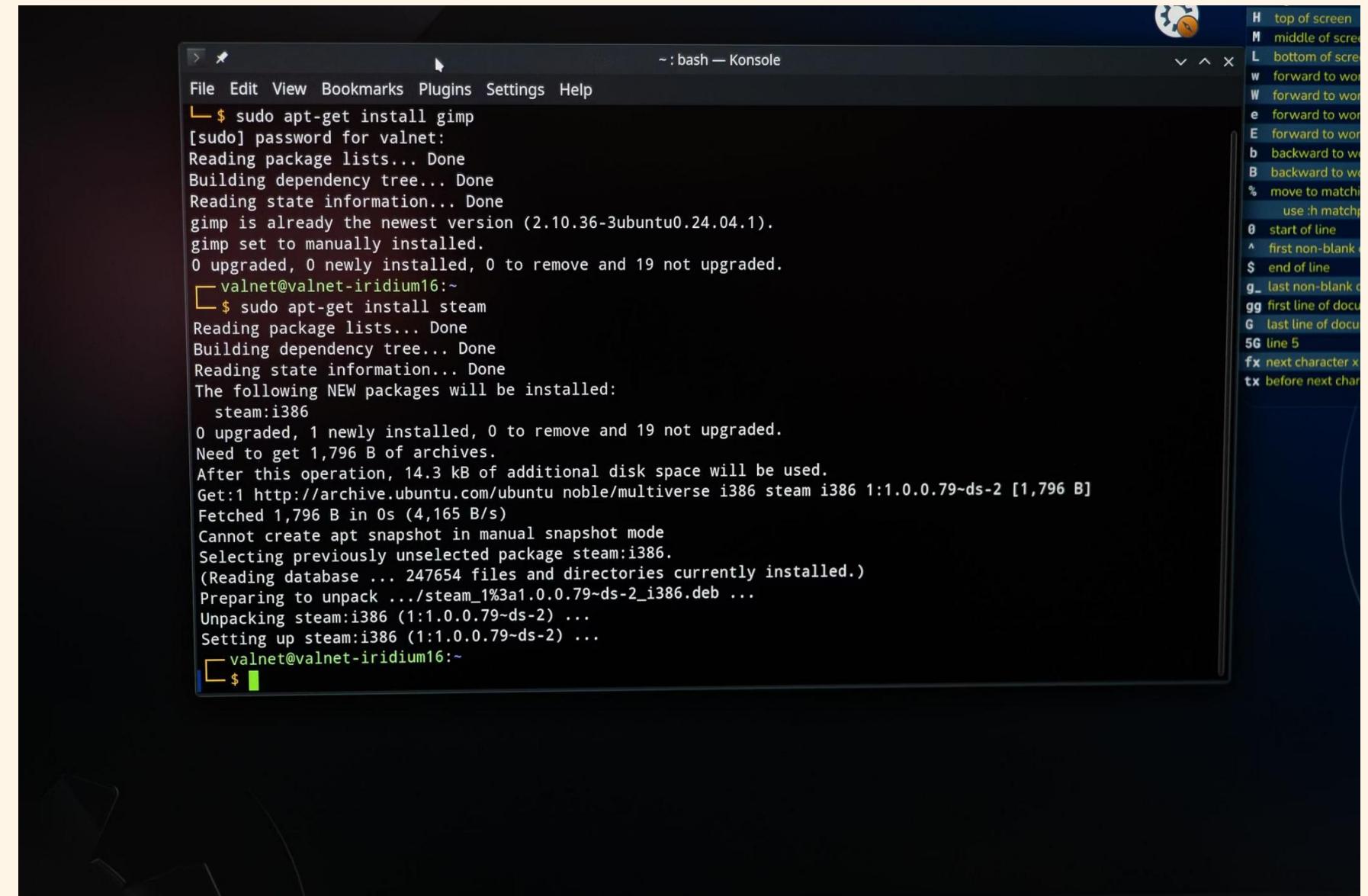
## Глава 4

# Создание RAID-массива с mdadm

Для создания RAID-массива необходимо подготовить разделы дисков с типом Linux raid autodetect (fd). Затем используется команда `mdadm --create` с указанием уровня RAID, количества активных устройств и путей к ним.

```
sudo mdadm --create /dev/md0 --  
level=1 --raid-devices=2 /dev/sdb1  
/dev/sdc1
```

Это создаст массив RAID 1 из двух разделов. Можно также добавить `--spare-devices` для горячего резерва.



The screenshot shows a terminal window titled "bash — Konsole". The terminal displays the following command and its output:

```
$ sudo apt-get install gimp  
[sudo] password for valnet:  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
gimp is already the newest version (2.10.36-3ubuntu0.24.04.1).  
gimp set to manually installed.  
0 upgraded, 0 newly installed, 0 to remove and 19 not upgraded.  
[valnet@valnet-iridium16:~]$ sudo apt-get install steam  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
The following NEW packages will be installed:  
steam:i386  
0 upgraded, 1 newly installed, 0 to remove and 19 not upgraded.  
Need to get 1,796 B of archives.  
After this operation, 14.3 kB of additional disk space will be used.  
Get:1 http://archive.ubuntu.com/ubuntu noble/multiverse i386 steam i386 1:1.0.0.79~ds-2 [1,796 B]  
Fetched 1,796 B in 0s (4,165 B/s)  
Cannot create apt snapshot in manual snapshot mode  
Selecting previously unselected package steam:i386.  
(Reading database ... 247654 files and directories currently installed.)  
Preparing to unpack .../steam_1%3a1.0.0.79~ds-2_i386.deb ...  
Unpacking steam:i386 (1:1.0.0.79~ds-2) ...  
Setting up steam:i386 (1:1.0.0.79~ds-2) ...  
[valnet@valnet-iridium16:~]$
```

A vertical toolbar on the right side of the terminal window provides keyboard shortcuts for navigating through the terminal history.

## Глава 5

# Файл конфигурации mdadm.conf

Файл `/etc/mdadm/mdadm.conf` (или `/etc/mdadm.conf`) содержит информацию о RAID-массивах, что позволяет системе автоматически собирать их при загрузке.

Его можно сгенерировать командой:

```
sudo mdadm --detail --scan >> /etc/mdadm/mdadm.conf
```

Это обеспечит корректную идентификацию и сборку массивов после перезагрузки, предотвращая потерю доступа к данным.

## Важность mdadm.conf

- Автоматическая сборка RAID
- Сохранение конфигурации
- Упрощение восстановления
- Идентификация массивов по UUID

## Глава 6

# Мониторинг и управление RAID-массивом



Состояние RAID-массива можно отслеживать через `/proc/mdstat` и `mdadm --detail`.

```
cat /proc/mdstat
```

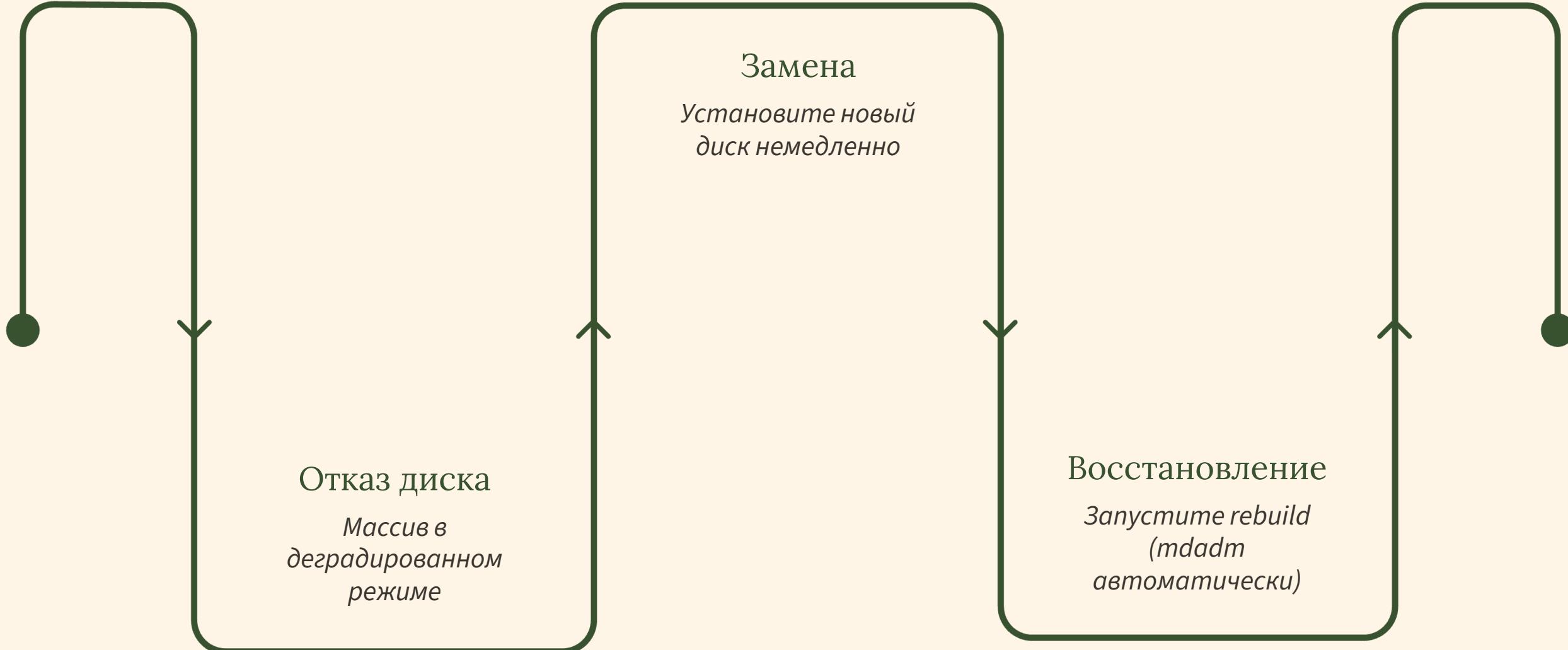
```
sudo mdadm --detail /dev/md0
```

Эти команды предоставляют информацию о статусе дисков, ходе синхронизации и наличии сбоев.

Управление горячим резервом (*spare*), `mark faulty`, `remove` и `add` позволяют оперативно реагировать на проблемы.

## Глава 7

# Отказоустойчивость и восстановление данных



При сбое диска массив переходит в деградированный режим. Важно немедленно заменить вышедший из строя диск новым и запустить процесс восстановления (`rebuild`). `mdadm` автоматически инициирует восстановление, используя данные четности или зеркалирования.

## Глава 8

# Расширение и миграция RAID-массивов

01

## Добавление дисков

*Возможность увеличить емкость массива путем добавления новых дисков, например, к RAID 5 или RAID 6.*

02

## Изменение уровня RAID

*Миграция с одного уровня RAID на другой (например, с RAID 1 на RAID 5) для оптимизации отказоустойчивости или производительности без потери данных.*

03

## Изменение размера компонентов

*Перераспределение пространства на существующих дисках для увеличения доступной емкости массива.*

# Заключение: Выбор и применение mdadm

*mdadm* предоставляет гибкое и экономичное решение для программного RAID в Linux. Выбор уровня RAID зависит от требований к производительности, отказоустойчивости и емкости.

Хотя *LVM* (*Logical Volume Manager*) обеспечивает гибкое управление логическими томами, *mdadm* сосредоточен исключительно на создании и управлении RAID-массивами. В сравнении с аппаратными контроллерами, *mdadm* выигрывает в стоимости и гибкости, но может уступать в производительности при интенсивных нагрузках.



Спасибо за внимание