

# zfs in proxmox

Кэш надо в proxmox'е отключать, он только мешает - zfs сама кэшем рулит.

Указываем что это ssd, discard

## Оптимальные настройки при создании пула

bash

```
# Создание с оптимизированными параметрами
zpool create -o ashift=13 \
-O compression=lz4 \
-O atime=off \
-O xattr=sa \
-O recordsize=1M \
-O redundant_metadata=most \
ваш_пул mirror /dev/nvme0n1 /dev/nvme1n1
```

## В настройках VM в Proxmox:

- Используйте `virtio-scsi` вместо `virtio-blk`
- Включите `IO thread` и `Discard`
- Установите `cache=none` (ZFS сам управляет кэшем)

## Проблема с ashift=12

- **ashift=12** = 4К сектора ( $2^{12} = 4096$  байт)
- **ashift=13** = 8К сектора ( $2^{13} = 8192$  байт)
- **NVMe диски** обычно работают с 8К-16К физическими секторами

Текущая настройка приводит к:

- ⚠ Write amplification (усиление записи)
- ⚠ Невыровненные операции IO

- ⚠ Потеря до 50% производительности

посмотреть

zdb -C | grep ashift

# VirtIO SCSI (Multi-Queue)

**Архитектура:** Множественные очереди (multi-queue)

bash

```
# В Proxmox отображается как:  
Controller Type: VirtIO SCSI
```

**Характеристики:**

- **Несколько IO очередей** (по умолчанию 4)
- **Параллельная обработка** запросов на разных CPU ядрах
- **Масштабируемость** - количество очередей = количеству vCPU
- **Лучшая производительность** на многопроцессорных VM
- **Поддержка IO Thread** для каждого диска

**Когда использовать:**

- VM с 2+ vCPU
- Высокая IO нагрузка
- Несколько виртуальных дисков
- Современные гостевые ОС (Linux 3.0+, Windows 8+)

---

# VirtIO SCSI Single

**Архитектура:** Одна очередь (single-queue)

bash

```
# В Proxmox отображается как:  
Controller Type: VirtIO SCSI Single
```

**Характеристики:**

- **Одна IO очередь**
- **Проще архитектура** - меньше накладных расходов
- **Лучше для 1-2 vCPU VM**
- **Совместимость** со старыми гостями

- **Меньше потребление** ресурсов гипервизора

#### Когда использовать:

- VM с 1-2 vCPU
- Низкая/средняя IO нагрузка
- Старые гостевые ОС
- Когда нужна максимальная совместимость

## Тестовые сценарии:

bash

```
# Тест для многопоточной нагрузки (лучше для Multi-Queue)
fio --name=test --ioengine=libaio --rw=randrw --bs=4k \
    --numjobs=8 --size=1G --runtime=60 --time_based --group_reporting

# Тест для однопоточной нагрузки (разница минимальна)
fio --name=test --ioengine=libaio --rw=randrw --bs=4k \
    --numjobs=1 --size=1G --runtime=60 --time_based --group_reporting
```

## Результаты:

- **Multi-queue:** +30-50% производительности при 4+ vCPU
- **Single-queue:** Меньше задержка при низкой нагрузке

## Итоговые рекомендации

Сценарий	Рекомендация
NVMe ZFS + 4+ vCPU	<b>VirtIO SCSI Multi-Queue</b>
SSD + 2-4 vCPU	<b>VirtIO SCSI Multi-Queue</b>
1-2 vCPU, легкая нагрузка	<b>VirtIO SCSI Single</b>
Старые гостевые ОС	<b>VirtIO SCSI Single</b>
Максимальная производительность	<b>VirtIO SCSI Multi-Queue + IO Thread</b>

Revision #3

Created 5 October 2025 10:02:09 by Admin

Updated 5 October 2025 10:42:52 by Admin