

199178, Санкт-Петербург, наб. реки Смоленки, д.33
Телефон: +7 (812) 622 16 80
request@raidix.ru
raidix.ru



РУКОВОДСТВО АДМИНИСТРАТОРА RAIDIX ERA 2.0

2019

ОГЛАВЛЕНИЕ

Введение.....	3
Назначение документа	3
Что такое RAIDIX ERA 2.0.....	3
Характеристики RAIDIX ERA 2.0.....	3
1. Управление ERA RAID.....	4
1.1 Описание интерфейса командной строки	4
1.2 Управление лицензией	5
1.3 Создание RAID	7
1.4 Просмотр состояния RAID	8
1.5 Восстановление RAID	11
1.6 Удаление RAID	11
1.7 Выгрузка RAID	12
1.8 Замена или исключение диска	12
1.9 Реконструкция RAID.....	13
1.10 Инициализация RAID	13
1.11 Изменение параметров RAID.....	14
1.12 Действия с конфигурационным файлом	15
1.13 Удаление метаданных с дисков	15
1.14 Операции, связанные с импортом RAID	16
1.15 Сообщения об ошибках	19
2. Рекомендации по настройке ERA RAID.....	22
2.1 Создание RAID	22
2.2 Параметры RAID	22
2.3 Настройка параметров RAID. Рекомендации	23

ВВЕДЕНИЕ

Назначение документа

Данное руководство предназначено для администраторов и пользователей RAID на основе программного обеспечения RAIDIX ERA 2.0.

Документ содержит информацию о настройке и управлении RAID на базе ПО RAIDIX ERA 2.0.

Что такое RAIDIX ERA 2.0

RAIDIX ERA 2.0 – это высокопроизводительный программный RAID, разработанный специально для NVMe-накопителей и новых типов SAN-сетей. Технологии RAIDIX ERA 2.0 эффективно используют потенциал flash-устройств (NVMe, SAS, SATA) для создания быстрого и отказоустойчивого RAID, доступного в виде локального блочного устройства с возможностью дальнейшего экспорта по сети с помощью дополнительного ПО. RAIDIX ERA 2.0 представляет собой модуль для ядра Linux и управляющую утилиту (CLI), которые собраны и сконфигурированы для наиболее популярных дистрибутивов операционной системы (Ubuntu 16.04 LTS; Ubuntu 18.04 LTS; CentOS 7.4, 7.5, 7.6). Программное обеспечение устанавливается на сервера, содержащие слоты для установки flash-накопителей или имеющие подключенные JBOD, и позволяет объединять диски в высокопроизводительный и отказоустойчивый RAID.

Характеристики RAIDIX ERA 2.0

Поддерживаемые уровни RAID	RAID 0, 1, 5, 6, 7.3.
Максимальное количество дисков в RAID	64.
Максимальное количество дисков в системе	Ограничено аппаратной конфигурацией.
Максимальное количество RAID	128.
Максимальный размер RAID	Ограничен размером дисков.
Место под метаданные RAID	100 MiB от начала каждого диска в RAID.

1. УПРАВЛЕНИЕ ERA RAID

Управление программным ERA RAID в Linux выполняется с помощью утилиты `eraraid`.

1.1 Описание интерфейса командной строки

В консоли управления команды вводятся в следующем формате:

```
eraraid {mode} {parameters}
```

Параметр `{mode}` может принимать следующие значения:

<code>create</code>	Создать RAID.
<code>show</code>	Показать состояние RAID.
<code>restore</code>	Восстановить RAID.
<code>destroy</code>	Удалить RAID.
<code>unload</code>	Выгрузить RAID.
<code>replace</code>	Заменить или исключить диск в RAID.
<code>recon</code>	Запустить реконструкцию.
<code>init</code>	Запустить инициализацию.
<code>modify</code>	Изменить параметры.
<code>config</code>	Управление файлом конфигурации.
<code>error-log</code>	Показать ошибки в ERA RAID.
<code>license</code>	Управление лицензией ERA RAID.
<code>drive-clean</code>	Очистить метаданные на дисках.
<code>import-show</code>	Показать все RAID для импорта.
<code>import-all</code>	Импортировать все возможные RAID.
<code>import</code>	Импортировать выбранные RAID с конфигурационных RAID на выбранных дисках.

Для отображения полного списка команд используйте команду:

```
eraraid -h
```

Для отображения версии ERA RAID используйте команду:

```
eraraid -v
```

Особенности синтаксиса консоли управления:

1. Параметры команды вводятся в одну строку.
2. Параметр, заключенный в фигурные скобки или без скобок – обязательный (`{mode}`).
3. Параметр, заключенный в квадратные скобки – уточняющий.
4. Параметры команды разделяются пробелами.
5. Могут быть введены как сокращенные обозначения атрибутов команды, так и полные.

Пример:

```
eraraid create -n (или --name) [введите значение параметра  
- имя массива] -l (или --level) [введите уровень массива] -  
d (или --drives) [введите номера дисков]
```

6. Чтобы получить список всех значений параметров нижних уровней, укажите атрибут `-h` после введенного параметра:

```
eraraid {mode} -h
```

Подробное описание значений параметров представлено ниже.

1.2 Управление лицензией

Чтобы начать работу с системой, добавьте действительный файл лицензии для каждого узла. Для этого потребуется аппаратный ключ конфигурации (`hwkey`), который находится в информации о лицензии. Для просмотра информации о лицензии используйте команду:

```
eraraid license --show
```

Пример вывода команды, когда лицензии нет:

```
hwkey: C6C67AF003871B7C  
license_key: null  
version: 0  
crypto_version: 0  
created: 0-0-0  
expired: 0-0-0  
disks: 0  
levels: 0  
type: nvme  
disks_in_use: 0  
status: expired
```

Пример вывода команды, если лицензия есть:

```
hwkey: C6C67AF003871B7C
license_key: C443B932D9CA66AA550A81862F24E
6D98FB6BB7031CA1071FCED1B38A3593
version: 1
crypto_version: 0
created: 2019-3-27
expired: 2020-12-31
disks: 128
levels: 7
type: nvme
disks_in_use: 0
status: valid
```

Содержание вывода:

hwkey	Аппаратный ключ.
license_key	Ключ лицензии.
version	Версия ПО.
crypto_version	Версия крипто-API для генератора лицензий.
created	Дата создания лицензии.
expired	Дата окончания лицензии.
disks	Максимальное количество дисков для данной лицензии.
levels	Максимальный уровень RAID.
type	Тип дисков.
disks in use	Количество используемых дисков в системе.
status	Статус лицензии.

Содержимое вывода сохраните в текстовый файл при помощи команды:

```
eraraid license --show > license_request.txt
```

Для получения лицензионного ключа, скопируйте полученный аппаратный ключ и отправьте его в службу поддержки RAIDIX по электронной почте.

После получения файла с лицензией, зайдите на сервер и примените полученный лицензионный ключ при помощи команды:

```
eraraid license --update /root/license.txt
```

Для проверки примененной лицензии снова выполните команду:

```
eraraid license --show
```

Полное описание параметров команды:

```
eraraid license [-h] (--show|--update {PATH}|--reset)
```

Параметры:

-h	--help	Справка.
-s	--show	Показать лицензию.
-u	--update	Применить лицензию из файла.
-r	--reset	Сбросить текущую лицензию.

1.3 Создание RAID

Для создания RAID используйте команду:

```
eraraid create [-h] -n NAME -l {0,1,5,6,7} -d {Номера  
дисков} [-ss {16,32,64,128,256}]
```

Обязательные параметры:

-n	--name	Имя RAID.
-l	--level	Уровень RAID: 0, 1, 5, 6, 7.
-d	--drives	Список дисков.

Дополнительные параметры¹:

-ss	--strip-size	Размер стрипа в KiB: 16, 32, 64, 128, 256 , по умолчанию 16 .
-ip	--init-prio	Приоритет инициализации от 0% до 100% , по умолчанию 100 .

¹ См. настройка дополнительных параметров. Параметры по умолчанию.

-rp	--recon-prio	Приоритет реконструкции от 0% до 100%, по умолчанию 100.
-me	--merge-enabled	Включение (1), выключение (0) параметра Merge, по умолчанию 0.
-se	--sched-enabled	Включение (1), выключение (0) параметра Scheduling, по умолчанию 0.
-ml	--memory-limit	Установка лимита на использование оперативной памяти в MiB, по умолчанию неограниченно.
-rl	--request-limit	Включение (1), выключение (0) лимита на запросы на RAID, по умолчанию 0.

Пример: Создание RAID 5 на 4 NVMe-дисках (nvme0n1, nvme1n1, nvme2n1, nvme3n1) с размером страйпа 64, именем era5, с включенным параметром merge:

```
eraraid create -n era5 -s 64 -l 5 -d /dev/nvme0n1
/dev/nvme1n1 /dev/nvme2n1 /dev/nvme3n1 -me 1
```

Для создания RAID уровней 5, 6, 7 требуется минимум 4 диска; для RAID 0 – один диск; для RAID 1 – два диска.

Минимальный лимит на использование оперативной памяти 1024 MiB, максимальный ограничен размером оперативной памяти в системе.

Предупреждение: не рекомендуется создавать ERA RAID поверх ERA RAID устройств.

1.4 Просмотр состояния RAID

Чтобы просмотреть информацию о состоянии RAID, запустите:

```
eraraid show [-h] [-n NAME | -o] [-u {s,k,m,g}] [-f
{table,json,prettyjson}] [-e]
```

Параметры:

-h	--help	Справка.
-n	--name	Имя RAID.
-o	--online	Показать только RAID в статусе Online.
-u	--units	Вывести размер RAID: s – в секторах (1 сектор=512B), k – в килобайтах, m – в мегабайтах, g – в гигабайтах.
-f	--format	Формат вывода информации: <ul style="list-style-type: none"> • table – в виде таблицы;

		<ul style="list-style-type: none"> • json – json; • prettyjson – в удобном для человека виде.
-e	--extended	Показать расширенный вывод.

Пример: Вывод информации о RAID era5:

```
eraraid show -n era5
```

RAID name	static	state	devices	info
era5	size: 19 GiB level: 6 strip_size: 16 active: True config: True	online initialized	0 /dev/sda online 1 /dev/sdb online 2 /dev/sdc online 3 /dev/sdd online 4 /dev/sde online 5 /dev/sdl online	init_progress : 100 recon_progress : 0 memory_usage_mb: -

```
eraraid show -n era5 -e
```

```
[root@raineratest10 ~]# eraraid show -e
```

RAID name	static	state	devices	serials	params	info
era5	size: 19 GiB level: 6 strip_size: 16 active: True config: True	online initialized	0 /dev/sda online 1 /dev/sdb online 2 /dev/sdc online 3 /dev/sdd online 4 /dev/sde online 5 /dev/sdl online	drive-scsi0 drive-scsi13 drive-scsi12 drive-scsi11 drive-scsi10 drive-scsi3	init_prio : 100 recon_prio : 100 memory_limit_mb: 0 merge_enabled : 0 sched_enabled : 0	init_progress : 100 recon_progress : 0 memory_usage_mb: -

Вывод команды:

Параметр	Описание
name	Имя RAID.
static	Статические параметры RAID: <ul style="list-style-type: none"> • Size – размер; • Level – уровень; • Strip size – размер стрипа; • Active: <ul style="list-style-type: none"> ○ True, если блочное устройство RAID присутствует в системе; ○ False, если: <ul style="list-style-type: none"> ▪ RAID не загрузился после перезагрузки. Требуется реконструкция.

Параметр	Описание
	<ul style="list-style-type: none"> ▪ RAID был выгружен. Требуется реконструкция. ▪ Отсутствует или не загружен драйвер <code>raidix_nvme.ko</code>; не найден пакет <code>rpm</code>. Установите или загрузите пакет; • Config: <ul style="list-style-type: none"> ○ True, если конфигурационный файл в системе. ○ False, если файл отсутствует, в данном случае попробуйте восстановить файл с дисков.
state	<p>Состояние RAID:</p> <ul style="list-style-type: none"> • Online – RAID доступен и готов к работе. • Degraded – RAID доступен и готов к работе, но некоторые диски были извлечены или неисправны. • Reconstructing – RAID в процессе реконструкции. • Offline – RAID недоступен. • Need_recon – RAID нуждается в реконструкции. • Need_init – RAID нуждается в инициализации. • Read Only – срок действия лицензии истек. RAID доступен только для чтения.
devices	<p>Список устройств, входящих в RAID, и их состояние:</p> <ul style="list-style-type: none"> • Online – диск активен. • Offline – диск извлечен или недоступен. • Reconstructing – идет процесс реконструкции. • Need_recon – диск нуждается в реконструкции.
info	<p>Динамические показатели RAID:</p> <ul style="list-style-type: none"> • <code>init_progress</code> (только для RAID 5,6,7.3) – прогресс инициализации: от 0% до 100%. • <code>recon_progress</code> (только для RAID 1,5,6,7.3) – прогресс реконструкции: от 0% до 100%. • <code>memory_usage_mb</code> – количество используемой оперативной памяти; значение отсутствует, если при создании не был указан параметр <code>memory_limit_mb</code>.
serials	<p>Серийные номера дисков в RAID (показываются в расширенной версии).</p>
params	<p>Динамические параметры RAID (показываются в расширенной версии):</p> <ul style="list-style-type: none"> • <code>init_prio</code> – приоритет инициализации: от 0% до 100%; • <code>recon_prio</code> – приоритет реконструкции: от 0% до 100%; • <code>memory_limit_mb</code> – ограничение на использование оперативной памяти в мегабайтах; • <code>merge-enabled</code> – включение (1) и выключение (0) параметра <code>merge</code>;

Параметр	Описание
	<ul style="list-style-type: none">• sched-enabled – включение (1) и выключение (0) параметра scheduling.

1.5 Восстановление RAID

После любого сбоя RAID должен восстановиться автоматически. Если этого не произошло, восстановите RAID с помощью команды:

```
eraraid restore [-h] (-n NAME | -a)
```

Параметры:

-h	--help	Справка.
-n	--name	Имя RAID.
-a	--all	Восстановить все RAID в системе.

Пример: Восстановление RAID **era5**:

```
eraraid restore -n era5
```

1.6 Удаление RAID

Для полного удаления RAID используйте команду:

```
eraraid destroy [-h] (-n NAME | -a)
```

Параметры:

-h	--help	Справка.
-n	--name	Имя RAID.
-a	--all	Удалить все RAID в системе.

Пример: Удаление RAID **era5**:

```
eraraid destroy -n era5
```

1.7 Выгрузка RAID

Выгрузка (или деактивация) – это удаление RAID с сохранением конфигурационного файла. В отличие от полного удаления, выгрузка оставляет возможность для последующего восстановления массива. Чтобы выгрузить RAID, используйте команду:

```
eraraid unload [-h] (-n NAME | -a)
```

Параметры:

-h	--help	Вызов справки.
-n	--name	Имя RAID.
-a	--all	Выгрузить все RAID в системе.

Пример: Выгрузка RAID **era5**:

```
eraraid unload -n era5
```

1.8 Замена или исключение диска

Для исключения или замены диска в RAID, используйте команду:

```
eraraid replace [-h] -n NAME -no NUMBER -d DRIVE
```

Параметры:

-h	--help	Вызов справки.
-n	--name	Имя RAID.
-no	--number	Номер диска. Чтобы узнать номер диска, используйте команду <code>eraraid show</code> .
-d	--drive	Новый диск. Для извлечения диска, чтобы пометить его как отсутствующий, задайте значение <code>null</code> .

Пример: Замена в RAID **era5** диска под номером **0** на диск **nvme4n1**:

- Отметьте нулевой диск как отсутствующий:

```
eraraid replace -n era5 -no 0 -d null
```

- Замените диск под номером 0 на диск с именем `nvme4n1`:

```
eraraid replace -n era5 -no 0 -d /dev/nvme4n1
```

1.9 Реконструкция RAID

Для управления процессом реконструкции RAID используйте команду:

```
eraraid recon [-h] -n NAME (--start | --stop)
```

Параметры:

-h	--help	Справка.
-n	--name	Имя RAID.
	--start	Запустить реконструкцию.
	--stop	Остановить реконструкцию.

Пример: Запуск реконструкции RAID `era5`:

```
eraraid recon --start -n era5
```

Чтобы улучшить производительность системы под нагрузкой, [попробуйте изменить приоритет реконструкции](#), изменив соответствующий параметр RAID.

1.10 Инициализация RAID

Инициализация запускается автоматически при создании RAID. Для управления инициализацией используйте команду:

```
eraraid init [-h] -n NAME (--start | --stop)
```

Параметры:

-h	--help	Вызов справки.
-n	--name	Имя RAID.
	--start	Запустить инициализацию.
	--stop	Остановить инициализацию.

Пример: Запуск инициализации RAID `era5`:

```
eraraid init --start -n era5
```

Чтобы улучшить производительность системы под нагрузкой, попробуйте [изменить приоритет инициализации](#), изменив соответствующий параметр RAID. Производительность случайной записи выше на инициализированном RAID.

1.11 Изменение параметров RAID

Чтобы улучшить производительность системы под нагрузкой, попробуйте изменить приоритеты инициализации и реконструкции, изменив соответствующие параметры RAID. При выставлении приоритетов равными 0, инициализация или реконструкция будет идти только при отсутствии нагрузки от клиентов. По умолчанию параметры выставлены в 100%, это означает, что процессы инициализации и реконструкции идут с максимальной возможной скоростью. Для увеличения производительности пользовательской нагрузки уменьшайте приоритеты. Параметр `--merge-enabled` уменьшает количество *read-modify-write*, тем самым увеличивая производительность записи, рекомендуется для маленьких блоков и интенсивного входного потока. Параметр `--sched-enabled` оптимизирует RAID под малопоточную нагрузку.

Для изменения динамических параметров RAID используйте команду:

```
eraraid modify [-h] -n NAME [-ip [0..100]] [-rp [0..100]]
               [-me {0,1}] [-se {0,1}] [-ml MEMORY_LIMIT] [-rl {0,1}]
```

Параметры:

-h	--help	Справка.
-n	--name	Имя RAID.
-ip	--init-prio	Приоритет инициализации: от 0% до 100%.
-rp	--recon-prio	Приоритет реконструкции: от 0% до 100%.
-me	--merge-enabled	Включение (1) и выключение (0) параметра merge.
-se	--sched-enabled	Включение (1) и выключение(0) параметра scheduling.
-ml	--memory-limit	Установить ограничение размера оперативной памяти в MiB.
-rl	--request-limit	Включение (1) и выключение(0) лимита на запросы RAID.

Пример: Изменение приоритета реконструкции RAID `era5` до 50%:

```
eraraid modify -n era5 -rp 50
```

1.12 Действия с конфигурационным файлом

Чтобы сохранить файл текущей конфигурации или восстановить конфигурацию по метаданным на дисках, используйте команду:

```
eraraid config [-h] (-r [RESTORE] | -a | -b | -d | -p)
```

Параметры:

-h	--help	Вызов справки.
-r	--restore	Восстановить файл конфигурации из архива (по умолчанию).
-a	--apply	Применить файл конфигурации для всех восстанавливаемых RAID.
-b	--backup	Сохранить текущий файл конфигурации (сделать бэкап).
-d	--drives	Восстановить файл конфигурации с дискового пути /etc/eraraid.conf.drive .
-p	--print	Вывести файлы конфигурации, хранящиеся на дисках.

Пример: Восстановление конфигурационного файла с дисков:

- Чтение конфигурационных файлов с дисков:

```
eraraid config -d
```

- Восстановление конфигурационного файла:

```
eraraid config -r /etc/eraraid.conf.drive
```

1.13 Удаление метаданных с дисков

Чтобы удалить метаданные с конкретных дисков надо воспользоваться командой:

```
eraraid drive-clean [-h] -d DRIVE [DRIVE ...]
```

Параметры:

-h	--help	Справка.
-d	--drives	Список дисков для удаления метаданных.

Пример: Удаление метаданных с дисков **/dev/nvme5n1** и **/dev/nvme1n1**:

```
eraraid drive-clean -d /dev/nvme1n1 /dev/nvme5n1
```

1.14 Операции, связанные с импортом RAID

RAIDIX ERA 2.0 предоставляет возможность пользователям импортировать RAID-ы, которые есть на дисках, но не содержатся в конфигурационном файле.

Эта функциональность используется в случаях, когда несколько систем ERA RAID, с которыми осуществлялась работа, необходимо объединить в одну.

RAID-ы с одинаковыми именами не могут существовать в одной системе, поэтому при добавлении в систему дисков с данными другой системы, в случае, если имена RAID совпадают, их следует переименовать перед импортом. Если есть конфликт с дисками и системным (находящимся в системе) RAID, RAID импортируется в **degraded** режиме с отсутствующим конфликтным диском. При конфликте с импортируемым диском, конфликтный диск войдет в RAID, который импортируется первым.

Возможные конфликты:

- name: Conflict with system raids – конфликт имени с системным RAID.
- drives: Conflict with system raids – конфликт дисков с системным RAID.
- name: Conflict with import raids – конфликт имени с импортируемым RAID.
- drives: Conflict with import raids – конфликт дисков с импортируемым RAID.
- name: Conflict with system and import raids – конфликт имени с системным и импортируемым RAID.
- drives: Conflict with system and import raids – конфликт дисков с системным и импортируемым RAID.

Команды, связанные с импортом:

```
eraraid import-show [-h] [-f {table,json,prettyjson}] [-d  
DRIVES [DRIVES ...]]
```

Параметры:

-h	--help	Справка.
-f	--format	Формат вывода информации: <ul style="list-style-type: none">• table – в виде таблицы;• json – json;• prettyjson – в удобном для человека виде.
-d	--drives	Показать импортируемые RAID с конкретных дисков.


```
eraraid import-all [-h] [-d DRIVES [DRIVES ...]]
```

Параметры:

-h	--help	Справка.
-d	--drives	Импорт RAID с конкретных дисков.

```
eraraid import [-h] -id UUID [-n NAME] [-d DRIVES [DRIVES ...]]
```

Параметры:

-h	--help	Вызов справки.
-id	--uuid	UUID RAID.
-n	--names	Новое имя RAID.
-d	--drives	Импорт RAID с конкретных дисков.

Пример:

1. Введите команду:

```
eraraid import-show
```

Утилита найдет и отобразит информацию о найденных RAID на дисках, нуждающихся в импорте. На рисунке ниже видно три RAID, найденных для импорта.

```
[root@rainertest10 ~]# eraraid import-show
```

RAIDs	info	devices	serials	import status
4FB99368-2337-4E0F-A1B9-F02D83CCB9AA	name: eral level: 6 strip_size: 16 size: 9 GiB date: 2018-12-18 15:30	/dev/sdb /dev/sde /dev/sdf /dev/sdd	drive-scsi0 drive-scsi11 drive-scsi10 drive-scsi12	name: Conflict with system raids drives: Conflict with system raids eral
9DD8B03A-A527-4715-9112-C3D62382069B	name: eral level: 7 strip_size: 16 size: 14 GiB date: 2018-12-18 15:30	/dev/sdi /dev/sdc null null /dev/sdm /dev/sdl	drive-scsi7 drive-scsi13 null null drive-scsi3 drive-scsi4	name: Conflict with system and import raids drives: OK
D8AC821A-E5C0-4395-B190-2C3344AF984D	name: era2 level: 0 strip_size: 16 size: 9 GiB date: 2018-12-21 14:51	/dev/sdk /dev/sdj	drive-scsi5 drive-scsi6	name: OK drives: OK

Красным выделены конфликты. Видно, что первый RAID конфликтует с системным (тот, который сейчас загружен в систему) по имени и диску. Второй конфликтует с системным и импортируемым по имени. Третий RAID не конфликтует ни с чем. Команда, чтобы посмотреть информацию о системных RAID:

```
eraraid show
```

```
root@raineratest10 ~]# eraraid show
```

-RAIDS-				
name	static	state	devices	info
era1	size: 4 GiB level: 1 strip_size: 16 active: True config: True	online	0 /dev/sdb online 1 /dev/sdn online 2 /dev/sdo online	recon_progress : 0 memory_usage_mb: -

2. Импорт всех RAID, с которыми нет конфликтов:

```
eraraid import-all
```

-RAIDS-				
name	static	state	devices	info
era1	size: 4 GiB level: 1 strip_size: 16 active: True config: True	online	0 /dev/sdb online 1 /dev/sdn online 2 /dev/sdo online	recon_progress : 0 memory_usage_mb: -
era2	size: 9 GiB level: 0 strip_size: 16 active: True config: True	online	0 /dev/sdk online 1 /dev/sdj online	

3. Восстановление RAID с конфликтами по имени и по диску:

```
eraraid import -id 4FB99368-2337-4E0F-A1B9-F02D83CCB9AA -n  
era3
```

Видно, что диск, с которым был конфликт, не восстановился. Чтобы вывести RAID из состояния **degraded**, замените нулевой диск.

```
[root@raineratest10 ~]# eraraid show
-RAIDs-
```

name	static	state	devices	info
era1	size: 4 GiB level: 1 strip_size: 16 active: True config: True	online	0 /dev/sdb online 1 /dev/sdn online 2 /dev/sdo online	recon_progress : 0 memory_usage_mb: -
era2	size: 9 GiB level: 0 strip_size: 16 active: True config: True	online	0 /dev/sdk online 1 /dev/sdj online	
era3	size: 9 GiB level: 6 strip_size: 16 active: True config: True	online degraded initialized	0 null offline 1 /dev/sde online 2 /dev/sdf online 3 /dev/sdd online	init_progress : 100 recon_progress : 0 memory_usage_mb: 0

1.15 Сообщения об ошибках

Если запущенная команда не выполнена или выполнена некорректно, посмотрите журнал ошибок (по умолчанию последние 10 ошибок) с помощью команды:

```
eraraid error-log [-h] [ -n [1..100]]
```

Параметры:

-h	--help	Вызов справки.
-n	--lines	Вывести последние ошибки из в количестве от 1 до 100 .

Пример: Вывод последних 20 сообщений об ошибках:

```
eraraid error-log -n 20
```

Список всех выводимых ошибок:

1. Ошибки инициализации:

- 1.1. "RAID %s cannot be initialized in offline state" – RAID в оффлайн режиме инициализировать невозможно.

- 1.2. "RAID %s is already in init state" – RAID уже инициализируется.
 - 1.3. "RAID %s doesn't need initialization" – RAID не нуждается в инициализации.
 - 1.4. "RAID %s cannot be initialized: reconstruction is in progress" – RAID не может быть проинициализирован, так как идет реконструкция.
 - 1.5. "RAID %s cannot be initialized: need reconstruction" – RAID не может быть проинициализирован, так как нуждается в реконструкции.
 - 1.6. "RAID %s cannot be initialized in degraded state" – RAID в degraded режиме инициализировать невозможно.
2. Ошибки лицензирования:
- 2.1. "Wrong hardware key: %08llX. System key: %08ll" – Неправильный аппаратный ключ для системы.
 - 2.2. "Cannot allocate memory for license_key_txt buffer" – не могу выделить память для лицензии в буфере в текстовом формате.
 - 2.3. "Cannot allocate memory for license_key_bin buffer" – не могу выделить память для лицензии в буфере в бинарном виде.
 - 2.4. "Cannot allocate memory for decrypted_key buffer" – не могу выделить память для ключа дешифрования в буфере.
 - 2.5. "Cannot parse license command" – не могу распарсить команду лицензии.
 - 2.6. "Cannot convert license key" – не могу конвертировать ключ лицензии.
 - 2.7. "Cannot decrypt license key" – не могу расшифровать ключ лицензии.
 - 2.8. "RAID level %d exceeds license level %d" – уровень RAID превышает возможный уровень.
 - 2.9. "Total used disks %d exceeds license disks %d" – количество дисков превосходит количество дисков в лицензии.
 - 2.10. "License expired" – срок действия лицензии истёк.
3. Общие ошибки:
- 3.1. "Cannot parse command" – невозможно распарсить команду.
 - 3.2. "Cannot allocate the devices array" – невозможно разместить массив дисков.
 - 3.3. "Cannot allocate the devices buffer" – невозможно разместить буффер дисков.
 - 3.4. "Cannot create RAID %s without disks" – невозможно создать RAID без дисков.
 - 3.5. "RAID %s with current name already exists" – RAID с данным именем уже существует.
 - 3.6. "Cannot parse UUID for RAID %s" – невозможно распарсить UUID RAID.
 - 3.7. "RAID %s with current UUID %pUB already exists" – RAID с данным UUID уже существует.

- 3.8. "Cannot create RAID %s on current license" – невозможно создать RAID с данной лицензией.
 - 3.9. "Cannot create RAID %s" – невозможно создать RAID.
 - 3.10. "Cannot find RAID %s" – невозможно найти RAID.
 - 3.11. "Too many disks - %d" – слишком много дисков.
 - 3.12. "Cannot parse recon command" – невозможно распарсить команду реконструкции.
 - 3.13. "Unsupported command" – команда не поддерживается.
4. Ошибки RAID:
 - 4.1. "Cannot create offline RAID %s" – невозможно создать офлайн RAID.
 - 4.2. "Wrong device number %d in RAID %s" – ошибочное число дисков в RAID.
 - 4.3. "Cannot open block device %s for replace in RAID %s" – невозможно открыть блочное устройство (диск) для замены в RAID.
 - 4.4. "New device %s is too small for RAID %s: new size %llu < %llu sectors" – новый диск малого размера для данного RAID: новый размер меньше на ...кол-во секторов.
 - 4.5. "Cannot allocate memory for device name in RAID %s" – невозможно выделить память для диска в RAID.
 5. Ошибки реконструкции:
 - 5.1. "RAID %s cannot be reconstructed in offline state" – RAID не реконструируется в офлайн режиме.
 - 5.2. "RAID %s is already in reconstruction state" – RAID уже реконструируется.
 - 5.3. "RAID %s cannot be reconstructed: no devices for" – RAID не может быть реконструирован: нет диска для реконструкции.

2. РЕКОМЕНДАЦИИ ПО НАСТРОЙКЕ ERA RAID

2.1 Создание RAID

Уровень RAID выбирается, исходя из желаемых параметров доступности.

Уровень доступности 99.999% достигается при помощи RAID 6, если количество дисков в массиве меньше 20. При большем количестве дисков используйте RAID 7.3.

Рекомендуемый размер стрипа для RAID из NVMe-дисков – **16** KiB, он выставлен по умолчанию.

2.2 Параметры RAID

Init-prio и Recon-prio.

С помощью режима `{modify}` можно менять приоритеты реконструкции и инициализации. При выставлении приоритетов равными нулю, инициализация или реконструкция будет идти только при отсутствии нагрузки от клиентов. По умолчанию параметры выставлены в **100%**, это означает, что процессы инициализации и реконструкции идут с максимальной возможной скоростью. Для увеличения производительности пользовательской нагрузки уменьшайте приоритеты.

Scheduling.

При поступлении входящего запроса есть два возможных пути его обработки:

- продолжить выполнение на текущем CPU;
- перебросить на другое ядро CPU и выполнить там. При этом на переброску тратится время.

При поступлении большого количества запросов на одно ядро CPU, включение параметра `--sched-enabled` позволяет равномерно распределить нагрузку между процессорами. Однако на многопоточных паттернах этот параметр показывает себя неэффективно, так как тратит время на перекидывание запросов с одного ядра CPU на другой.

Рекомендуется включать параметр `--sched-enabled` при малом количестве потоков.

Merge.

Для улучшения работы с маленькими блоками был внедрен параметр `--merge-enabled`, который для запросов на запись намеренно устанавливает некоторое время ожидания, в течение которого запрос не передается на диски. Последовательные запросы

объединяются, что позволяет существенно сократить количество операций для обработки запросов на синдромных RAID, так как сокращает количество операций *read-modify-write*. Однако, если запись случайная, то такое ожидание не позволит объединить запросы, отрицательно влияя на производительность RAID.

Рекомендуется включать параметр `--merge-enabled` при последовательной записи маленькими блоками и большом количестве потоков.

Request-limit.

Данный параметр ограничивает количество принимаемых запросов.

Например, пишем файлы на файловую систему не синхронизируя запись. Рекомендуется включать эту опцию при работе с файловой системой и записью на нее в режиме `direct` равным `0` для повышения производительности RAID.

2.3 Настройка параметров RAID. Рекомендации

1. При создании синдромного RAID автоматически запускается процесс инициализации. RAID при этом доступен для чтения и записи. Поскольку процесс инициализации по умолчанию имеет приоритет `100`, то рекомендуется дождаться окончания инициализации, или если нагрузка на RAID не *random write*, понизить значение приоритета инициализации. Таким образом, пользовательские I/O будут быстрее обрабатываться, т.к. автоматически уменьшится количество запросов инициализации. При выставлении значения в `0`, запросы на инициализацию не создаются, если есть пользовательские I/O.
2. Процесс реконструкции запускается автоматически. По умолчанию имеет приоритет `100` и реконструкция идет с максимально возможной скоростью. При выставлении значения приоритета реконструкции меньше `100`, пользовательские I/O будут быстрее обрабатываться, т.к. автоматически уменьшится количество запросов инициализации. При выставлении значения в `0`, запросы на инициализацию не создаются, если есть пользовательские I/O.
3. При создании RAID рекомендуемый размер Strip Size = `16 KiB`
4. Выставление профиля в `tuned-adm throughput-performance` выдает лучшую производительность для большинства тестов:

```
tuned-adm profile throughput-performance
```

5. Начиная с ERA 2.0, в ERA ведется учет выделенной памяти и контролируется, чтобы это значение не превысило ограничение. Параметр `--memory-limit` может изменяться в любой момент времени. По умолчанию, верхнее значение

неограниченно. Для повышения производительности отключите учет текущей выделенной памяти и контроль верхнего значения, выставив параметр в **0** или поставив очень большое значение (большее или равное размеру оперативной памяти) с помощью команды `modify`.

NUMA

1. В случае, когда на системе имеется больше одного CPU, объедините в один RAID диски, принадлежащие одной NUMA Node. Чтобы узнать, какая NUMA Node у диска, выполните:

```
cat /sys/block/nvme0n1/device/device/numa_node
```

или через `lspci`:

```
lspci -vvv
```

2. При создании NVMeoF таргета для ERA RAID рекомендуется, чтобы сетевой адаптер принадлежал той же NUMA Node, что и NVMe диски.

Workload

RAIDIX ERA 2.0 работает таким образом, что пользовательские I/O по возможности обрабатываются на том же CPU, на котором были получены от пользователя. Однако для некоторых типов нагрузки иногда имеет смысл передать обработку I/O на другие CPU, чтобы они не простаивали. Начиная с RAIDIX ERA 2.0 есть возможность включить scheduling I/O на все возможные CPU с помощью параметра `--sched-enabled` (**1** – включен, **0** – выключен)

Merge

1. Если запись на ERA RAID происходит последовательными блоками, размер которых меньше `stripe_data_len`, включите `merge` маленьких запросов, чтобы избежать `read-modify-write`. Включается `merge` с помощью параметра `--merge-enabled` (**1** – включен, **0** – выключен).
2. Если заранее известно, что глубины очереди пользовательской нагрузки и размера блока недостаточно, чтобы накопить на целый страйп, то `merge` включать не рекомендуется. Т.е. `merge` стоит включить, если $iodepth * block_size \geq stripe_data_len$, где $stripe_data_len = num_data_strips * strip_size$. Например, для RAID 6 с 6 дисками и $strip_size = 16\text{KiB}$ получаем $stripe_data_len = 4 * 16\text{KiB} = 64\text{KiB}$

NVMeoF

1. Можно создавать ERA RAID поверх устройств, которые были прокинуты через NVMeoF. Чтобы не происходило зависание команд при потере соединения, при подключении этих устройств рекомендуется использовать параметр `--ctrl-loss-tmo` равный **0**:

```
nvme connect -t rdma -n nqn.raidix12_1 -a 10.30.0.12 -s  
4420 --ctrl-loss-tmo=0
```

Актуально для *nvme-cli* версий ≥ 1.4 .

2. При создании NVMeoF таргета для ERA RAID рекомендуется включать *merge*, если предполагается нагрузка в виде последовательной записи. Linux Kernel и Mellanox NVMeoF таргеты разбивают большие запросы на **32KiB** + остаток. Такое поведение приводит к постоянным *read-modify-write*. Для SPDK NVMeoF таргета задавайте параметр `InCapsuleDataSize`, обозначающий, при каком значении разбивать запросы.