

Почему RAID-6?

В качестве вступления

Отсюда: <http://habrahabr.ru/post/214707/>

Диски имеют параметр, именуемый Unrecoverable Read Error Rate, который на современных бюджетных моделях SATA составляет 1 сектор на каждые сто триллионов битов. Т.е. примерно из каждых записанных 12Тб диск один раз скажет «прости, хозяин, но выдать обратно нужный сектор совершенно никак невозможно; ошибка чтения». Это методическая ошибка, заложенная производителем и потому **теоретически гарантирующая невозможность реконструкции массива RAID5 емкостью более 12Тб на дешевых дисках** (справедливости ради отметим, что URE на дисках SAS, как минимум, на порядок меньше, а критический объем, соответственно, больше). Эпитафию RAID5 написал Robin Harris в своей статье «Why RAID 5 stops working in 2009»

<http://www.zdnet.com/blog/storage/why-raid-5-stops-working-in-2009/162>

Почему RAID-6?

Источник: <http://habrahabr.ru/blogs/linux/111036/>

Как известно, RAID-5 выдерживает смерть одного веника, и после этой самой смерти – до момента когда закончится восстановление рейда с новым винчестером ваши данные под угрозой – восстановление обычно занимало до 70 часов для больших массивов и еще один веник может легко умереть в это время. RAID-6 выдерживает смерть 2-х любых веников. Из минусов – общепризнанное мнение что тормозит, особенно запись, даже по сравнению с RAID-5. Что-ж, проверим.

Почему софтрейд?

Железный рейд нужен только в одном случае – если у него есть батарейка и набортный кеш. Тогда контроллер сразу отвечает ОС что запись на диск завершена на физическом уровне и всякие ACID базы работают очень быстро и безопасно. В остальных случаях никаких бонусов по сравнению с софт-рейдом нет, одни минусы:

1. Сгорело железо? Новый сервер? Будьте добры купить тот же контроллер, ну или молитесь о совместимости. Софтрейд из тех-же дисков собирается где угодно.
2. Цена. Собственно, из-за этого нормальных рейдов с батарейкой я в руках так ниразу и не держал

Ну а те «рейд-контроллеры» которые стоят на обычных материнских платах – вообще никогда не стоит использовать. Они просто дают грузить ОС с рейда за счет набортного биоса (который выполняется центральным процессором, своего процессора нет), на этом их польза заканчивается, и остаются только минусы.

О паре мифов софтрейда

1. **Он жрет много драгоценного процессора.** Если мы одним глазком глянем в исходники драйвера RAID в ядре Linux, то увидим, что там давно все оптимизированно под SSE2. А с SSE2 процессор может считать XOR от 16 байт за 1 такт на 1 ядре современного процессора и все упирается в скорость обмена с памятью. Можете прикинуть сколько %

загрузки одного ядра сгенерирует поток в 1Гб/сек 😊 А ядер то много 😊 На практике, с моим Opteron 165 (1.8Ghz 2 ядра) скорость никогда не упиралась в CPU.

2. **Он разваливается и потом хрен соберешь.** Если что-то и отваливается – то из-за железа (например обычные винты любят иногда делать всякие фоновые задачи). Добавление вывалившегося веника – простая операция, которая кроме того может проводится автоматически. Впрочем, в среднем это надо делать раз в год.

```
# mdadm /dev/md0 -a /dev/sde1
```

3. **У софтрейда хреновый мониторинг.** С мониторингом все отлично и настраиваемо. Достаточно например просто мыло указать в конфиге mdadm и он пришлет вам письмо если что-то случиться с вашим массивом. Очень удобно. Вот например что приходит если один веник отвалился:

```
This is an automatically generated mail message from mdadm running
on XXXXX
A DegradedArray event had been detected on md device /dev/md0.
Faithfully yours, etc.
P.S. The /proc/mdstat file currently contains the following:
Personalities: [raid6] [raid5] [raid4]
md0: active raid6 sda1[1] sdc1[4] sdd1[3] sde1[2]
2929683456 blocks super 1.2 level 6, 1024k chunk, algorithm 2 [5/4]
[_UUUU]
unused devices: none
```

Рекоменую протестировать перед использованием:

```
# mdadm --monitor -l -m myname@myisp.com /dev/md0 -t
```

4. **У софтрейда очень низкая скорость перестройки массива.** В дефолтной конфигурации – да. А если вы дочитаете до конца статьи – узнаете как сделать так, чтобы все перестраивалось со скоростью самого медленного веника.

О роли bitmap

Linux-овый софтрейд поддерживает замечательную фичу: bitmap. Там отмечаются измененные блоки на диске, и если у вас почему-то отвалился один диск из массива, а потом вы его обратно добавили – полная перестройка массива не нужна. Чертовски полезно. Хранить можно на самом рейде – internal, а можно в отдельном файле – но тут есть ограничения (на тип ФС например). Я сделал internal bitmap. И зря. Internal bitmap тормозит безбожно т.к. постоянно дергается головка веников при записи.

Посмотрим на скорость:

Скорость можно тестировать например так:

```
# time sh -c «dd if=/dev/zero of=ddfile bs=1M count=5000»  
# time sh -c «dd if=ddfile of=/dev/null bs=1M count=5000»
```

Результаты для моего RAID-6 из 5xWD 1T6 получились следующие: чтение 268МБ/сек, запись 37МБ/сек. Все разводят руками и говорят: ну а чего же вы хотели? RAID-6 тормозит при записи, ведь ему надо прочитать то что было записано раньше, чтобы посчитать обновленные контрольные суммы для всех дисков. А еще и этот битмар... Скорость перестройки массива – около 25МБ/сек – полная перестройка массива до 15 часов. Вот он, ваш ночной кошмар.

Решаются проблемы просто:

1. У драйвера рейда в Linux есть такой полезный параметр: **stripe_cache_size**, значение по умолчанию которого равно 256. Слишком низкое значение – резко снижает скорость записи (как оказалось). Оптимальное значение для многих – 8192. Это — кол-во блоков памяти на 1 диск. 1 блок это обычно 4kb (зависит от платформы), для 5-и дискового массива кеш займет $8192 * 4kb * 5 = 160MB$.

```
# echo 8192 > /sys/block/md0/md/stripe_cache_size
```

Действовать начинает моментально. Теперь в большинстве случаев драйверу не приходится читать диск перед записью (особенно при линейной записи), и производительность резко вырастает. После перезагрузки пропадает, чтобы не пропало — добавляем в какой-нибудь /etc/rc.local например. Скорость перестройки массива теперь – 66МБ/сек (это сразу по всем дискам, около 5 часов на весь массив), скорость чтения осталась той-же, а вот скорость записи – выросла до 130МБ/сек (с 37).

2. Переносим битмар на отдельный диск (в моём случае — системный). Если системный веник сдохнет — ничего страшного, массив восстановится и без битмар-а. Головка больше не дергается при записи лишний раз, и скорость записи вырастает до 165МБ/сек.

```
# mdadm -G /dev/md0 -b /var/md0_intent
```

Итак, за 10 секунд мы подняли скорость записи с удручающих 37 МБ/сек до вполне приличных 165 МБ/сек (более чем в 4 раза!!). Теперь через Samba по сети файлы и пишутся и читаются 95-100 МБ/сек, и планировавшийся из-за низкой скорости рейда апгрейд сервера придется отложить на неопределенное время – производительности дохленького Opteron 165 теперь с лихвой хватает для всех поставленных задач.

[linux](#), [ubuntu](#), [raid](#), [raid-6](#)

From:

<https://wiki.rtzra.ru/> - RTzRa's hive

Permanent link:

<https://wiki.rtzra.ru/software/mdadm/why-raid6>

Last update: **2024/08/04 14:54**



