GRAID5 - програмный RAID 5

Опубликовано muff в Сб, 2018-01-27 04:29



В одном из проектов необходимо было реализовать небольшое хранилище с минимальным резервированием (для хранения бекапов). Решение - построение програмного **RAID5** на **FreeBSD**. Рассматривать и сравнивать типы **RAID** не будем, рассмотрим только реализацию даной задачи.

Итак, диски присутствующие в системе:

```
# sysctl kern.disks
kern.disks: ada0 aacd3 aacd2 aacd1 aacd0
```

- **ada0** - диск, на который установлена система (**FreeBSD 11.1 amd64**)
- **aacd3 aacd2 aacd1 aacd0** - диски, подключенные через **SAS**-контроллер, именно из них и будем собирать **RAID5**.

FreeBSD поддерживает организацию **RAID5** двумя менеджерами дискового пространства: **VINUM** и **GEOM**. Остановимся на реализации с помощью **GEOM**.

Чтобы задействовать функционал **GEOM RAID5**, необходимо выполнить установку порта **graid5**. Стоит отметить, что для сборки **graid5** требуется наличие исходных текстов, поэтому предварительно оновим **src**:

```
# svnlite checkout svn://svn.freebsd.org/base/releng/11.1/ /usr/src
```

После этого выполним установку **graid5**:

```
# cd /usr/ports/sysutils/graid5 && make install clean && rehash
```

Создадим директорию, куда в дальнейшем будем монтировать наш **RAID5**:

```
# mkdir /raid5
```

Создаем сам массив:

```
# graid5 label -v -s 256k raid5 /dev/aacd0 /dev/aacd1 /dev/aacd2 /dev/aacd3
Metadata value stored on /dev/aacd0.
Metadata value stored on /dev/aacd1.
Metadata value stored on /dev/aacd2.
Metadata value stored on /dev/aacd3.
Wasting 1046528 bytes (>=0GiB).
Done.
```

- **256k** - задаем значение **Stripe Size** (размер блока данных, записываемый на каждый диск **RAID**-массива в каждый **stripe**, тоесть мы будем записывать/читать данные блоками по **256k**)
- **raid5** - указываем название массива
- **/dev/aacd0 ... /dev/aacd3** - перечисляем устройства, которые добавляем в массив

Создаем файловую систему на созданном устройстве:

```
# newfs /dev/raid5/raid5
```

Примонтируем созданный раздел к ранее созданной нами директории:

```
# mount -t ufs /dev/raid5/raid5 /raid5
```

Также добавим запись в **fstab**, чтобы раздел монтировался при загрузке системы:

```
# echo '/dev/raid5/raid5       /raid5  ufs     rw      2       2' >> /etc/fstab
```

И добавим загрузку модуля **geom_raid5.ko** при старте системы:

```
# echo 'geom_raid5_load="YES"' >> /boot/loader.conf
```

На этом все...

Проверяем статус массива:

```
# graid5 status


      Name           Status  Componentsraid5/raid5  REBUILDING HOT
 aacd0 (115764101120 / 2% (p:192))                  aacd1 (115764101120 / 2% (p:192))
        aacd2 (115764101120 / 2% (p:192))                  aacd3 (115764101120 / 2% (p:192)
)
```

Идет пересборка... По завершению пересборки статус массива изменится на **COMPLETE**:

```
# graid5 status


      Name         Status  Componentsraid5/raid5  COMPLETE CALM
 aacd0               aacd1               aacd2               aacd3
```

Перезагружаем систему, чтобы убедиться, что после старта массив запустится в штатном режиме:

```
# reboot
```

После перезагрузки массив работает в штатноме режиме, все в порядке. На этом настройка завершена.

Данная комманда может пригодиться в дальнейшем... Чтобы массив начал синхронизироваться после замены диска, необходимо выполнить комманду:

```
# graid5 insert raid5 /dev/aacd3
```

- **raid5** - название нашего массива
- **/dev/aacd3** - диск, который мы заменили

Ну и фактический результат проверки скорости записи/чтения на массив, выполненый с помощью утилиты **dd**.

Проверяем скорость записи:

```
# dd if=/dev/zero of=/raid5/test_file.zero bs=100M count=100
100+0 records in
100+0 records out
10485760000 bytes transferred in 101.761061 secs (103042951 bytes/sec)
```

Проверяем скорость чтения:

```
# dd if=/raid5/test_file.zero of=/dev/null
20480000+0 records in
20480000+0 records out
10485760000 bytes transferred in 23.882999 secs (439047037 bytes/sec)
```

Источник (получено *2026-11-03 15:03*): http://muff.kiev.ua/content/graid5-programnyi-raid-5