# **Задание 1**

Какие виды RAID увеличивают производительность дисковой системы?

*Приведите ответ в свободной форме.*

# **Решение 1**

RAID 0, дисковый массив с чередованием. Производительность повышается за счет того, что данные разбиваются на фрагменты, которые записываются поочередно на два и более дисков. Избыточностью RAID 0 не страдает, в случае потери данных на одном диске, данные ни с него, ни с других дисков восстановить не удастся. Одним из вариантов решения данной проблемы может быть RAID 50: при достаточно небольшой потере в скорости обеспечивается приличная безотказность системы. С другой стороны RAID 50 начинает строиться от 6 дисков, соответственно и стоимость существенно возрастает.

# **Задание 2**

Назовите преимущества использования VFS. Используется ли VFS при работе с tmpfs? Почему?

*Приведите развернутый ответ в свободной форме.*

# **Решение 2**

VFS является прослойкой между системными вызовами и реальными файловыми системами. Используя абстрактные методы структуры file\_operations она позволяет ОС взаимодействовать с разными типами файловых систем без необходимости изменения оперируемых данных. VFS при работе с tmpsf не используется, так как tmpfs хранится в оперативной памяти и методами file\_operations не пользуется.

# **Задание 3**

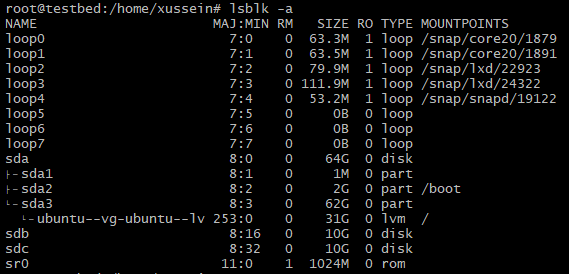
Подключите к виртуальной машине 2 новых диска.

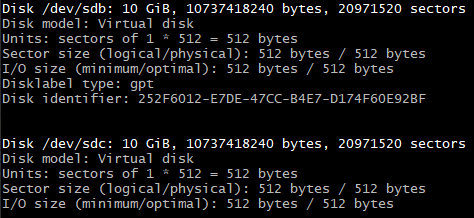
1. На первом диске создайте таблицу разделов MBR, создайте 4 раздела: первый раздел на 50% диска, остальные диски любого размера на ваше усмотрение. Хотя бы один из разделов должен быть логическим.
2. На втором диске создайте таблицу разделов GPT. Создайте 4 раздела: первый раздел на 50% диска, остальные любого размера на ваше усмотрение.

*В качестве ответа приложите скриншоты, на которых будет видно разметку диска (например, командами lsblk -a; fdisk -l)*

# **Решение 3**

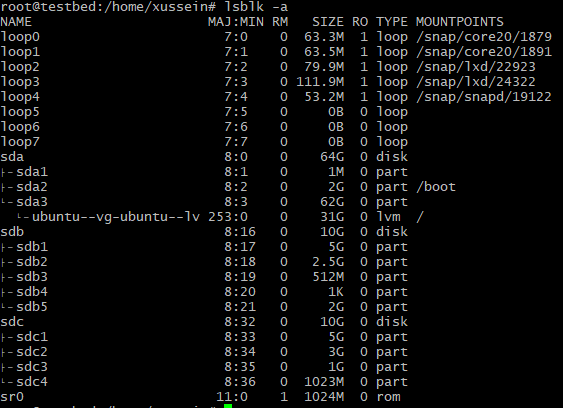
До:

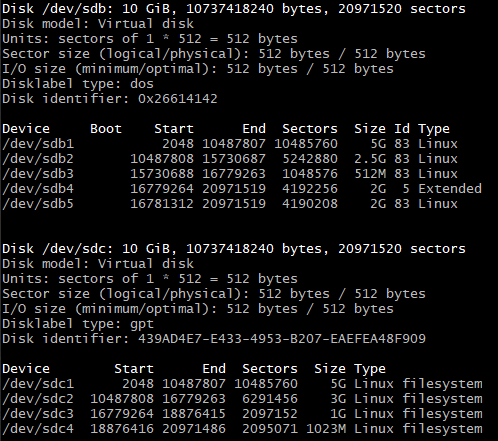




Диск sdb уже был в gpt так как во время лекции экспериментировал, не удалил. Через fdisk принудительно перевел в mbr (команда o).

После:





MRB (sdb) делал через fdisk, GPT (sdc) через gdisk.

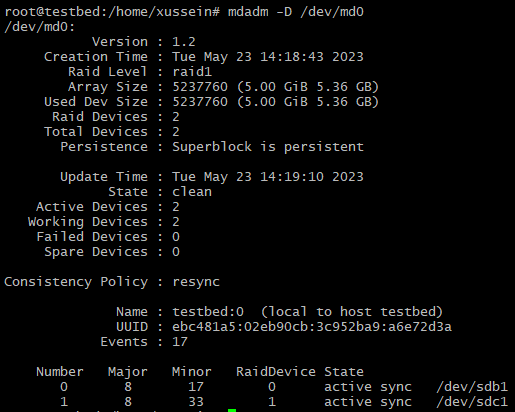
# **Задание 4**

Создайте программный RAID 1 в вашей ОС, используя программу mdadm.

Объем RAID неважен.

*В качестве ответа приложите скриншот вывода команды mdadm -D /dev/md0, где md0 - это название вашего рейд массива (может быть любым).*

# **Решение 4**



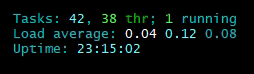
# **Задание 5**

Влияет ли количество операций ввода-вывода на параметр load average?

*Приведите развернутый ответ в свободной форме.*

# **Решение 5**

Система без нагрузки на диск:

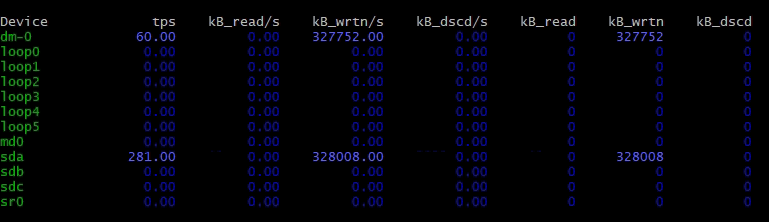


С запуском 4х потоков dd записи на диск:



И возрастает (в моем случае коснулось 3.9 и замерло).

Диск при этом, разумеется, показывает обращения к нему:



Параметр load average является численным отражением общей загруженности системы. “*Суть «средней нагрузки» — предоставить численную оценку занятости системы с точки зрения человека*” - Matthias Urlichs, ответственный за сегодняшнее представление load average. Внезапно, оказывается, что это ни разу не среднее значение за 1, 5 и 15 минут, а очень даже экспоненциальное (exponentially-damped moving sums of a five second average). В целом, значение load average это аналог средней температуры по больнице, за исключением морга. То есть анализ трех значений позволит понять динамику системы и увидеть есть ли в ней проблемы. Но для детального анализа нужно воспользоваться более тонкими метриками.

Лучше всего описывает этот параметр комментарий из кода ядра loadavg.c:

\* This file contains the magic bits required to compute the global loadavg

\* figure. Its a silly number but people think its important. We go through

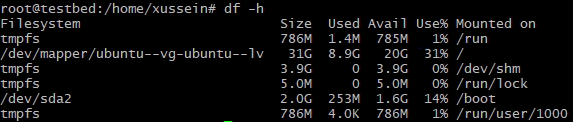
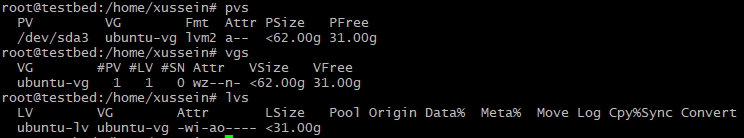
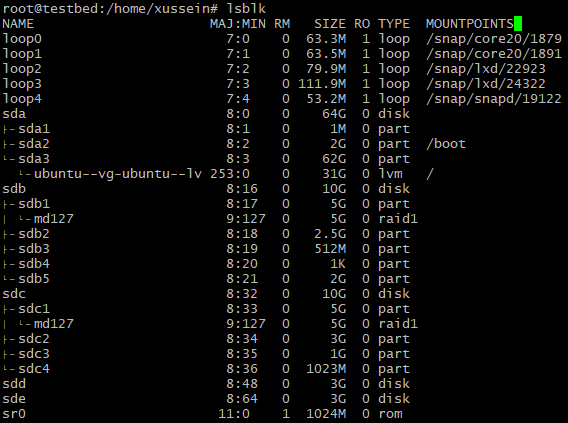
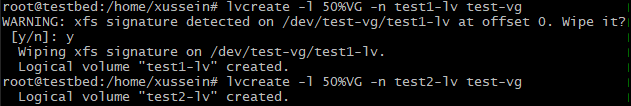
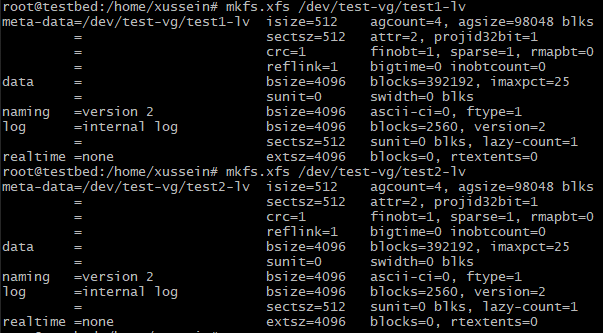
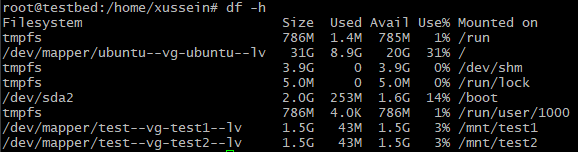
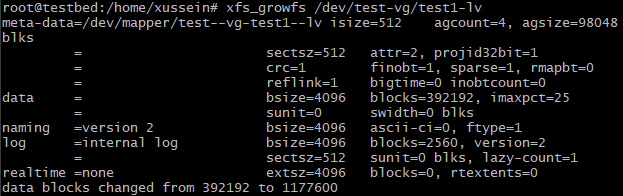
\* great pains to make it work on big machines and tickless kernels.

**Задание 6**

1. Сделайте скриншоты вывода комманд df -h, pvs, lvs, vgs.
2. Подключите к ОС 2 новых диска.
3. Создайте новую VG, добавьте в него 1 диск.
4. Создайте 2 LV, распределите доступное пространство между ними поровну.
5. Создайте на обоих томах файловую систему xfs.
6. Создайте две точки монтирования и смонтируйте каждый из томов.
7. Сделайте скриншот вывода комманд df -h.
8. Добавьте в VG второй оставшийся диск.
9. Расширьте первый LV на объем нового диска.
10. Расширьте файловую систему на размер нового доступного пространства.
11. Сделайте скриншоты вывода комманд df -h, pvs, lvs, vgs.

*В качестве ответа приложите созданные скриншоты и скриншоты выполнения.*

# **Решение 6**

1.   
   
2. Новые диски sdd и sde  
   
3. 
4. 
5. 
6. 
7. 
8. 
9. 
10. 
11. 