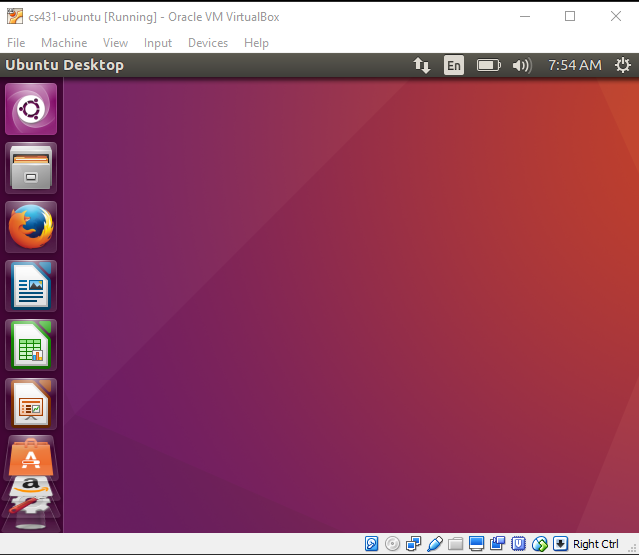
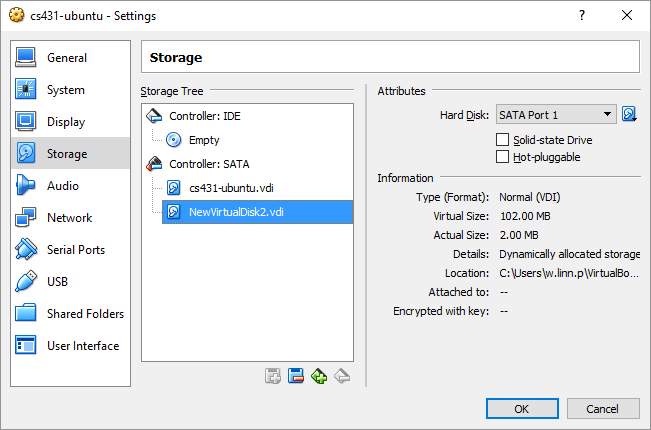
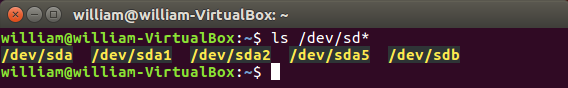
Virtual Box Ubuntu Image



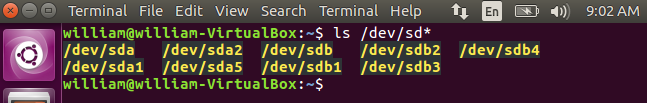
Adding a disk

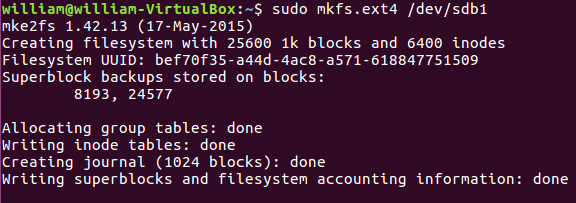


Creating Partitions

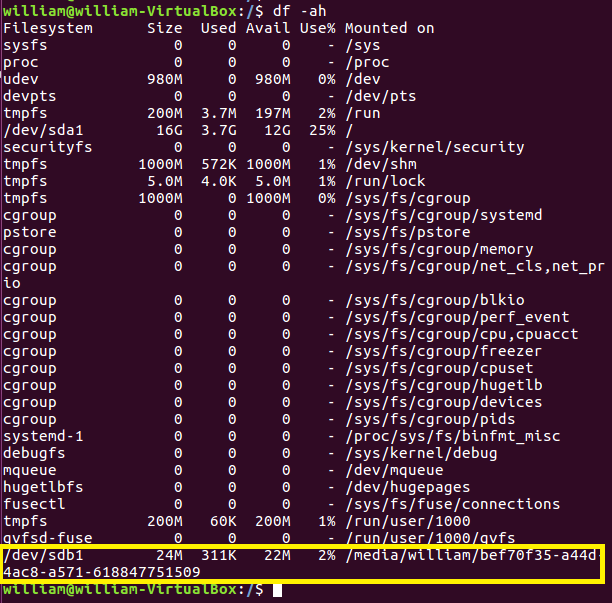


After creating 4 partitions in /dev/sdb using fdisk

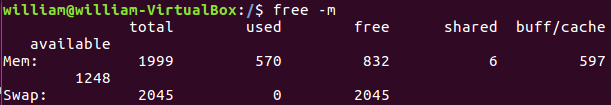


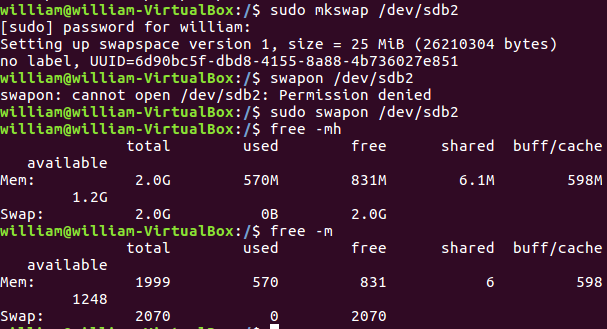
Creating ext4 File system

Automatically mounted.



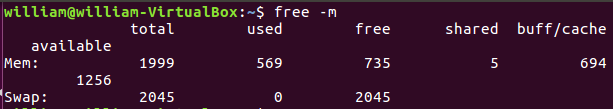
Creating Swap Space



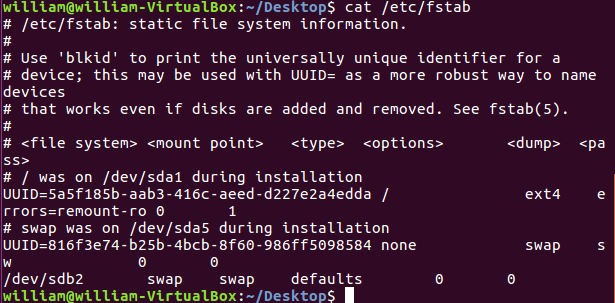
Creating swap space and adding it

Total area increased from 2045 to 2070.

After restarting the machine,



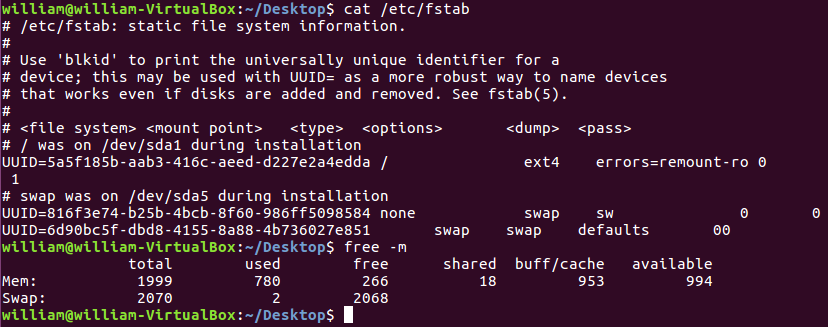
Editing /etc/fstab



FAILED in free –m. Did not see the extra space.

getUUID for /devsdb2



Update fstab using UUID instead of /dev/sb2. free –m works. extra space is added. 

Based on fstab documentation.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| file system | mount point | type | option | dump | pass |
| /dev/sdb2 | swap | swap | defaults | 0 | 0 |

From “man fstab”

The first field (fs\_spec).

This field describes the block special device or remote

filesystem to be mounted.

For ordinary mounts, it will hold (a link to) a block special

device node (as created by mknod(8)) for the device to be

mounted, like `/dev/cdrom' or `/dev/sdb7'. For NFS mounts, this

field is <host>:<dir>, e.g., `knuth.aeb.nl:/'. For filesystems

with no storage, any string can be used, and will show up in

df(1) output, for example. Typical usage is `proc' for procfs;

`mem', `none', or `tmpfs' for tmpfs. Other special filesystems,

like udev and sysfs, are typically not listed in fstab.

LABEL=<label> or UUID=<uuid> may be given instead of a device

name. This is the recommended method, as device names are often

a coincidence of hardware detection order, and can change when

other disks are added or removed. For example, `LABEL=Boot' or

`UUID=3e6be9de-8139-11d1-9106-a43f08d823a6'. (Use a filesystem-

specific tool like e2label(8), xfs\_admin(8), or fatlabel(8) to

set LABELs on filesystems).

It's also possible to use PARTUUID= and PARTLABEL=. These

partitions identifiers are supported for example for GUID

Partition Table (GPT).

See mount(8), blkid(8) or lsblk(8) for more details about device

identifiers.

Note that mount(8) uses UUIDs as strings. The string

representation of the UUID should be based on lower case

characters.

The second field (fs\_file).

This field describes the mount point (target) for the

filesystem. For swap partitions, this field should be specified

as `none'. If the name of the mount point contains spaces these

can be escaped as `\040'.

The third field (fs\_vfstype).

This field describes the type of the filesystem. Linux supports

many filesystem types: ext4, xfs, btrfs, f2fs, vfat, ntfs,

hfsplus, tmpfs, sysfs, proc, iso9660, udf, squashfs, nfs, cifs,

and many more. For more details, see mount(8).

An entry swap denotes a file or partition to be used for

swapping, cf. swapon(8). An entry none is useful for bind or

move mounts.

More than one type may be specified in a comma-separated list.

mount(8) and umount(8) support filesystem subtypes. The subtype

is defined by '.subtype' suffix. For example 'fuse.sshfs'. It's

recommended to use subtype notation rather than add any prefix

to the first fstab field (for example 'sshfs#example.com' is

deprecated).

The fourth field (fs\_mntops).

This field describes the mount options associated with the

filesystem.

It is formatted as a comma-separated list of options. It

contains at least the type of mount (ro or rw), plus any

additional options appropriate to the filesystem type (including

performance-tuning options). For details, see mount(8) or

swapon(8).

Basic filesystem-independent options are:

defaults

use default options: rw, suid, dev, exec, auto, nouser,

and async.

noauto do not mount when "mount -a" is given (e.g., at boot

time)

user allow a user to mount

owner allow device owner to mount

comment

or x-<name> for use by fstab-maintaining programs

nofail do not report errors for this device if it does not

exist.

The fifth field (fs\_freq).

This field is used by dump(8) to determine which filesystems

need to be dumped. Defaults to zero (don't dump) if not

present.

The sixth field (fs\_passno).

This field is used by fsck(8) to determine the order in which

filesystem checks are done at boot time. The root filesystem

should be specified with a fs\_passno of 1. Other filesystems

should have a fs\_passno of 2. Filesystems within a drive will

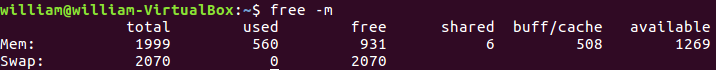
be checked sequentially, but filesystems on different drives

will be checked at the same time to utilize parallelism

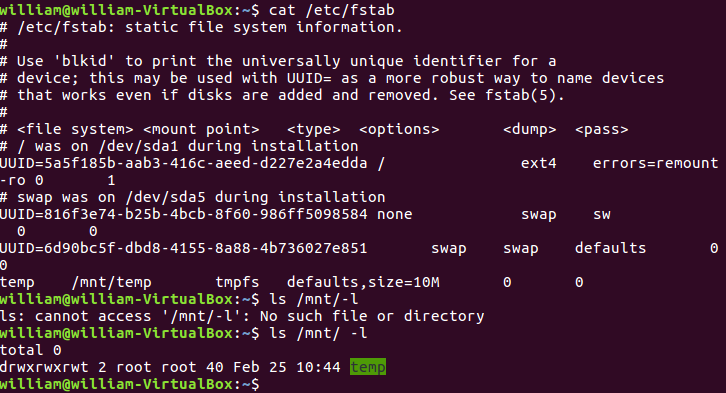
available in the hardware. Defaults to zero (don't fsck) if not

present.

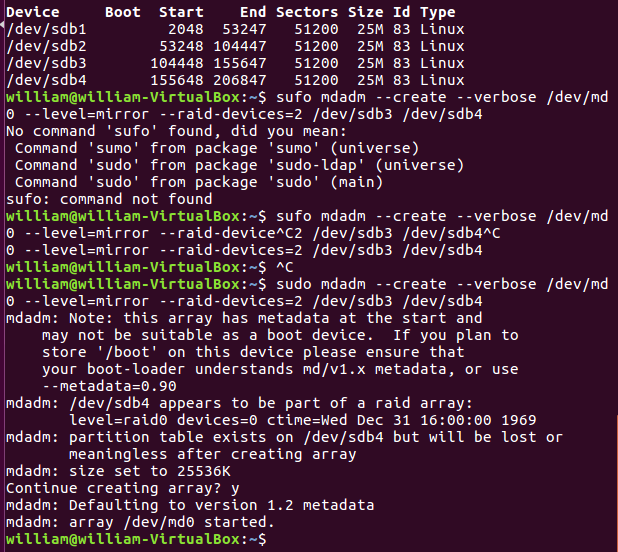
After restarting the VM, free –m still retains swap space from sdb2



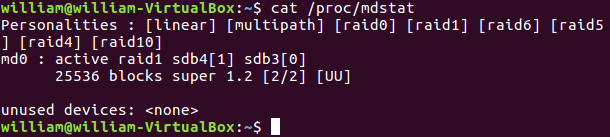
Creating Temporary Storage as instructed.

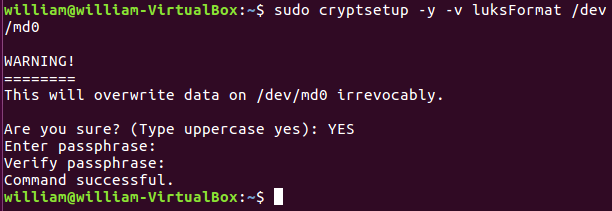


Creating a RAID1 Mirror

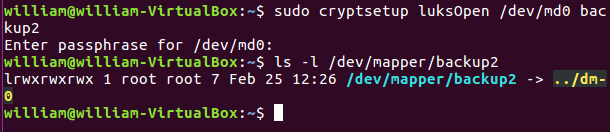
Created /dev/md0 

Printing /proc/mdstat

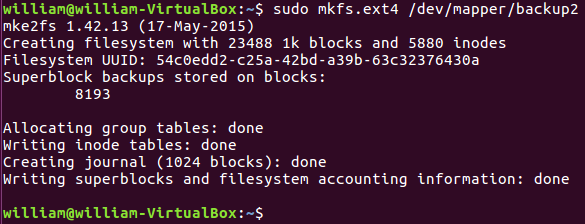


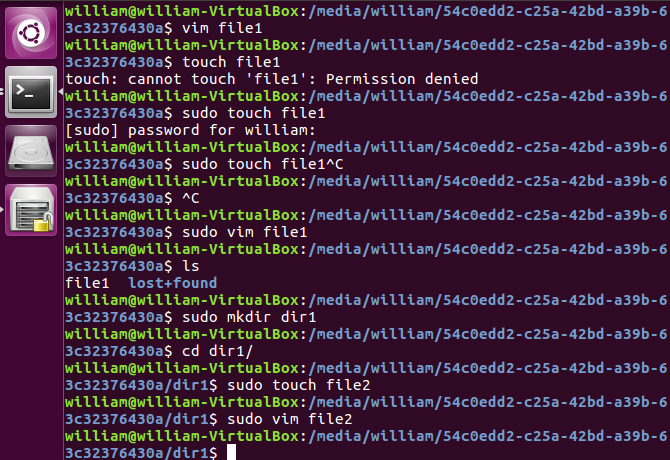
Creating encrypted drive

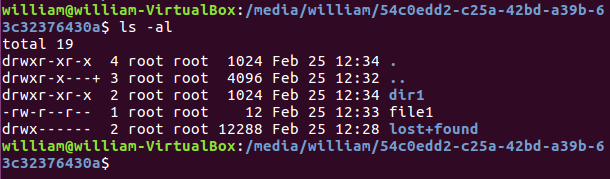
Initializing the volume with name backup2:

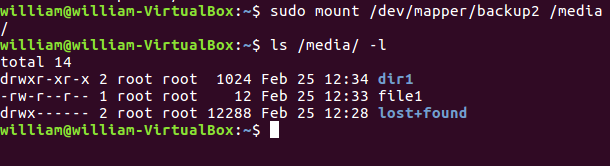


Creating file system on that volume



Mounted automatically. Adding some files to the new file system



Mounting manually after unmounting by UI. 

Still containing file1 and directory created.

Unmounting /media, closing the file system with LukClose, and verifying 