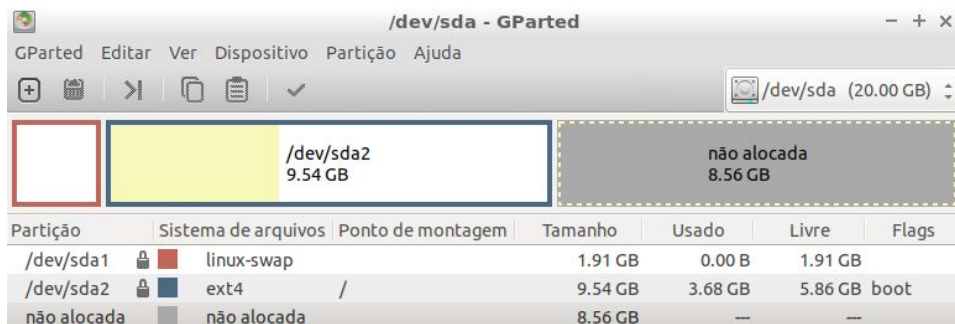


ESTUDO DIRIGIDO: SISTEMAS DE ARQUIVOS

PAULO ROBERTO DE SCHULLER BANJAR

1. Crie uma Máquina Virtual com um HD de 20GB.
Porém, durante a instalação, quando o instalador perguntar como *particionar o disco*, escolha **customizar**.
Crie uma instalação com os seguintes parâmetros:
 1. 10GB para a pasta raiz /
 2. 2GB para área de swap**Criem um Snapshot após a instalação do Sistema Operacional!!! Potencialmente vocês podem perder tudo nesse estudo!**
Após a instalação, inicie o sistema operacional.

Máquina instalada já com essa configuração pré-existente!



2. Abra um terminal. Execute o comando **fdisk /dev/sda1** e descubra o que as seguintes opções fazem:

O comando **fdisk** lista todas as partições de todos os discos conectados. No caso de ter vários discos, eles serão mostrados na ordem de dispositivo.

- a) p
- b) m
- c) n
- d) v
- e) w

```
Genérico
d  exclui uma partição
F  lista partições não particionadas livres
l  lista os tipos de partições conhecidas
n  adiciona uma nova partição
p  mostra a tabela de partição
t  altera o tipo da partição
v  verifica a tabela de partição
i  mostra informação sobre uma partição

Miscelânea
m  mostra este menu
```

3. Com o fdisk, descubra qual é o tamanho do bloco do disco e quantos blocos há na partição /dev/sda2

sudo fdisk -l : lista todos os tipos de partições conhecidas.

```
usuario@usuario-VirtualBox:~$ sudo fdisk -l
Disco /dev/sda: 20 GiB, 21474836480 bytes, 41943040 setores
Unidades: setor de 1 * 512 = 512 bytes
Tamanho de setor (lógico/físico): 512 bytes / 512 bytes
Tamanho E/S (mínimo/ótimo): 512 bytes / 512 bytes
Tipo de rótulo do disco: dos
Identificador do disco: 0xc10c5ee6
```

Dispositivo	Inicializar	Início	Fim	Setores	Tamanho	Id	Tipo
/dev/sda1		2048	3999743	3997696	1,9G	82	Linux swap / Solaris
/dev/sda2	*	3999744	24000511	20000768	9,6G	83	Linux
/dev/sda3		24000512	28194815	4194304	2G	83	Linux
/dev/sda4		28194816	32389119	4194304	2G	5	Estendida
/dev/sda5		28196864	32389119	4192256	2G	83	Linux

Multiplique a quantidade de blocos pelo tamanho do bloco. O que o valor resultante significa? É o valor esperado? Justifique.



O valor resultante significa o tamanho total do bloco, e ele é o esperado, tendo em vista que multiplicando a quantidade de setores (20000768) por 512 (Bytes), que é o tamanho de cada setor, obtemos o tamanho total do bloco em bytes (10240393216), que corresponde a aproximadamente 9,6G, que é o tamanho total.

10240393216 Bytes = 9.5371 Gigabytes

4. Com o fdisk, crie 3 partições de 2GB. Para isso, utilize a opção *n*

```
usuario@usuario-VirtualBox:~$ sudo fdisk /dev/sda -l
Disco /dev/sda: 20 GiB, 21474836480 bytes, 41943040 setores
Unidades: setor de 1 * 512 = 512 bytes
Tamanho de setor (lógico/físico): 512 bytes / 512 bytes
Tamanho E/S (mínimo/ótimo): 512 bytes / 512 bytes
Tipo de rótulo do disco: dos
Identificador do disco: 0xc10c5ee6
```

Dispositivo	Inicializar	Início	Fim	Setores	Tamanho	Id	Tipo
/dev/sda1		2048	3999743	3997696	1,9G	82	Linux swap / Solar
/dev/sda2	*	3999744	24000511	20000768	9,6G	83	Linux
/dev/sda3		24000512	27906047	3905536	1,9G	83	Linux
/dev/sda4		27906048	35719167	7813120	3,7G	5	Estendida
/dev/sda5		27908096	31813631	3905536	1,9G	83	Linux
/dev/sda6		31815680	35719167	3903488	1,9G	83	Linux

5. A ferramenta **mkfs** é a responsável por criar sistemas de arquivos em partições GNU/Linux. Cada sistema de arquivos possui uma variação desse programa do tipo **mkfs.[tipo]**, em que [tipo] é o tipo de sistema de arquivos.

Ou seja, o comando **mkfs.ext4** é o responsável por criar sistemas de arquivos do tipo ext4. Descubra o que os seguintes parâmetros fazem:

O mkfs torna fácil formatar drives de vários tipos, tais como pendrives, cartões de memória, HDs, SSDs etc. Além disso é possível criar um sistema de arquivos em um arquivo — semelhante a um disco rígido virtual. Já o “ext” significa “Extended file system” ou “Sistema de arquivos estendido”, foi o primeiro sistema de arquivos criados unicamente para o linux em 1992. A atual versão dos tipos Ext. é a Ext4, que possui várias funções vantajosas quando comparada com as suas antecessoras, como redução na fragmentação do sistema.

Vale ressaltar ainda que se deve ser muito cuidadoso, pois não é possível refazer a formatação.

6. Usando o mkfs, crie os seguintes tipos de sistemas de arquivos:

- a) um sistema de arquivos do tipo *ext4* com tamanho de bloco de 4K,

```

usuario@usuario-VirtualBox:~$ sudo mkfs.ext4 -b 4096 /dev/sda3
mke2fs 1.44.1 (24-Mar-2018)
/dev/sda3 contains a ext4 file system
    created on Wed Sep  4 20:29:54 2019
Proceed anyway? (y,N) y
Creating filesystem with 488192 4k blocks and 122160 inodes
Filesystem UUID: 84a34994-b34d-4de3-a34a-ca32a37bba17
Cópias de segurança de superblocos gravadas em blocos:
    32768, 98304, 163840, 229376, 294912

Allocating group tables: pronto
Gravando tabelas inode: pronto
Creating journal (8192 blocks): concluído
Escrevendo superblocos e informações de contabilidade de sistema de arquivos: 0
concluído

```

- b) um sistema de arquivos do tipo *ext4* com tamanho de bloco de 1K,

```

usuario@usuario-VirtualBox:~$ sudo mkfs.ext4 -b 1024 /dev/sda5
[sudo] senha para usuario:
mke2fs 1.44.1 (24-Mar-2018)
Creating filesystem with 1952768 1k blocks and 122368 inodes
Filesystem UUID: 4c3be477-3bb3-49f7-bd1e-7ef521d3579f
Cópias de segurança de superblocos gravadas em blocos:
    8193, 24577, 40961, 57345, 73729, 204801, 221185, 401409, 663553,
    1024001

Allocating group tables: pronto
Gravando tabelas inode: pronto
Creating journal (16384 blocks): concluído
Escrevendo superblocos e informações de contabilidade de sistema de arquivos:
concluído

```

- c) um sistema de arquivos do tipo *ext2*

```

usuario@usuario-VirtualBox:~$ sudo mkfs.ext2 /dev/sda6
mke2fs 1.44.1 (24-Mar-2018)
Creating filesystem with 487936 4k blocks and 122160 inodes
Filesystem UUID: 6652b342-dcdf-4654-a53f-ee036af1a7c8
Cópias de segurança de superblocos gravadas em blocos:
    32768, 98304, 163840, 229376, 294912

Allocating group tables: pronto
Gravando tabelas inode: pronto
Escrevendo superblocos e informações de contabilidade de sistema de arquivos: 0
concluído

```

7. Crie 3 pastas:

- a) /media/particaoA
- b) /media/particaoB
- c) /media/particaoC

O comando `mkdir` é responsável por criar pastas e diretórios. Para indicar o caminho se usa o seguinte caminho: /media/particaoA(a, b ou c), que está indicando o local onde a pasta irá ser criada.


```

usuario@usuario-VirtualBox:~$ sudo mkdir /media/particaoA
[sudo] senha para usuario:
usuario@usuario-VirtualBox:~$ sudo mkdir /media/particaoB
usuario@usuario-VirtualBox:~$ sudo mkdir /media/particaoC
usuario@usuario-VirtualBox:~$ █

```

8. Monte os sistemas de arquivos do item anterior usando o comando *mount*. Tente escrever algum arquivo nelas com o comando *touch*. Você consegue escrever? Se não, o que você deve fazer para poder conseguir?

O comando “mount” é utilizado para que o sistema de arquivos sejam “mostrados” no sistema de arquivos.

Já o “Touch” é o comando utilizado principalmente para criar arquivos vazios, além de alterar o registro de data e hora (timestamp) de arquivos ou pastas.

E o “chmod” é um utilitário via linha de comando usado para mudar as permissões de um arquivo ou diretório. Os parâmetros passados para o programa podem ser números ou letras. No comando “sudo chmod =+wx” está permitindo que possam ser realizadas a escrita, execução e leitura do arquivo existente no diretório media/particaoA.

```

usuario@usuario-VirtualBox:~$ sudo mount -t ext4 /dev/sda3 /media/particaoA
[sudo] senha para usuario:
usuario@usuario-VirtualBox:~$ sudo mount -t ext4 /dev/sda5 /media/particaoB
usuario@usuario-VirtualBox:~$ sudo mount -t ext2 /dev/sda6 /media/particaoC
usuario@usuario-VirtualBox:~$ █

```

Para que seja possível escrever, é necessário que a permissão seja alterada. Como é indicado no print abaixo.

```

usuario@usuario-VirtualBox:~$ cd /media/particaoA
usuario@usuario-VirtualBox:/media/particaoA$ sudo touch teste.txt
[sudo] senha para usuario:
usuario@usuario-VirtualBox:/media/particaoA$ sudo chmod =+wx /media/particaoA
usuario@usuario-VirtualBox:/media/particaoA$ cd /media/particaoB
usuario@usuario-VirtualBox:/media/particaoB$ sudo touch testeB.txt
usuario@usuario-VirtualBox:/media/particaoB$ sudo chmod =+wx /media/particaoB
usuario@usuario-VirtualBox:/media/particaoB$ cd /media/particaoC
usuario@usuario-VirtualBox:/media/particaoC$ sudo touch testeC.txt
usuario@usuario-VirtualBox:/media/particaoC$ sudo chmod =+wx /media/particaoC
usuario@usuario-VirtualBox:/media/particaoC$ █

```

9. Desmonte todos os sistemas de arquivos montados no item anterior.

```

usuario@usuario-VirtualBox:~$ sudo umount /media/particaoA
[sudo] senha para usuario:
usuario@usuario-VirtualBox:~$ sudo umount /media/particaoB
usuario@usuario-VirtualBox:~$ sudo umount /media/particaoC
usuario@usuario-VirtualBox:~$ █

```

10. Edite o arquivo `/etc/fstab` para que os sistemas de arquivos criados sejam automaticamente montados ao iniciar o computador.

```
GNU nano 2.9.3 /etc/fstab Modificado

# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point> <type> <options> <dump> <pass>
# / was on /dev/sda2 during installation
UUID=90ce04a1-ca0b-42a4-ac65-8eac9497a6a2 / ext4 errors=remoun$
# swap was on /dev/sda1 during installation
UUID=be18e939-3b3f-4b24-a072-d11e95b3833b none swap sw $
/dev/sda3 /media/particaoA ext4 defaults 0 0
/dev/sda5 /media/particaoB ext4 defaults 0 0
/dev/sda6 /media/particaoC ext2 defaults 0 0

```

GNU nano 2.9.3 interface showing the `/etc/fstab` file. The file contains static file system information. The bottom of the screen shows the nano editor's command palette with options like 'Obter Ajuda', 'Gravar', 'Onde está?', 'Recortar', 'Justificar', 'Pos atual', 'Sair', 'Ler o arquivo', 'Substituir', 'Colar', 'Verificar ortografia', and 'Ir para linha'.

Após reiniciar o computador os sistemas de arquivos foram montados automaticamente.

