

ESTUDO DIRIGIDO: SISTEMAS DE ARQUIVOS

BRENDA DA COSTA CONCEIÇÃO ANDRADE

INSTRUÇÕES:

1. Os exercícios abaixo deverão ser feitos dentro do terminal do Linux
2. Os exercícios estão, em sua maioria, dependentes um dos outros. Ou seja, o segundo depende do primeiro, o terceiro do segundo, e assim sucessivamente. Procure fazê-los na ordem.
3. **Deverá ser entregue um relatório, até a data final especificada, contendo a sequência de comandos que vocês digitaram para realizar cada item do trabalho.**
A sequência de comandos **deverá** ser um screenshot da tela de terminal com os comandos que vocês utilizaram.
4. **Os comandos deverão ser explicados, bem como o significado de seus parâmetros.**
5. Entregue este trabalho em formato PDF! Coloque o seu nome no lugar do nome do aluno acima!

ESPECIFICAÇÃO:

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.
2. Abra um terminal. Execute o comando **fdisk /dev/sda1** e descubra o que as seguintes opções fazem:
 - a) p → Esta opção mostra a tabela de partição;
 - b) m → Esta opção mostra o menu do enunciado comando do enunciado;
 - c) n → Esta opção adiciona uma nova partição
 - d) v → Esta opção verifica a tabela de partição
 - e) w → Esta opção grava a tabela no disco e sai

```

usuario@usuario-VirtualBox:~$ sudo fdisk /dev/sda
Bem-vindo ao fdisk (util-linux 2.31.1).
As alterações permanecerão apenas na memória, até que você decida gravá-las.
Tenha cuidado antes de usar o comando de gravação.

Comando (m para ajuda): m

Ajuda:

DOS (MBR)
a alterna a opção de inicialização
b edita o rótulo do disco BSD aninhado
c alterna a opção "compatibilidade"

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
u altera as unidades das entradas mostradas
x funcionalidade adicional (somente para usuários avançados)

Script
I carrega layout de disco de um arquivo script de sfdisk
O despeja layout de disco para um arquivo script de sfdisk

Salvar & sair
w grava a tabela no disco e sai
q sai sem salvar as alterações

Cria um novo rótulo
g cria uma nova tabela de partição GPT vazia
G cria uma nova tabela de partição SGI (IRIX) vazia
o cria uma nova tabela de partição DOS vazia
s cria uma nova tabela de partição Sun vazia

Comando (m para ajuda): █

```

3. Com o fdisk, descubra qual é o tamanho do bloco do disco e quantos blocos há na partição /dev/sda2
Escrevendo o comando **sudo fdisk -l** serão listados todos os tipos de partição conhecidas

```
usuario@usuario-VirtualBox: ~
Arquivo Editar Abas Ajuda

Salvar & sair
w grava a tabela no disco e sai
q sai sem salvar as alterações

Cria um novo rótulo
g cria uma nova tabela de partição GPT vazia
G cria uma nova tabela de partição SGI (IRIX) vazia
o cria uma nova tabela de partição DOS vazia
s cria uma nova tabela de partição Sun vazia

Comando (m para ajuda): p
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

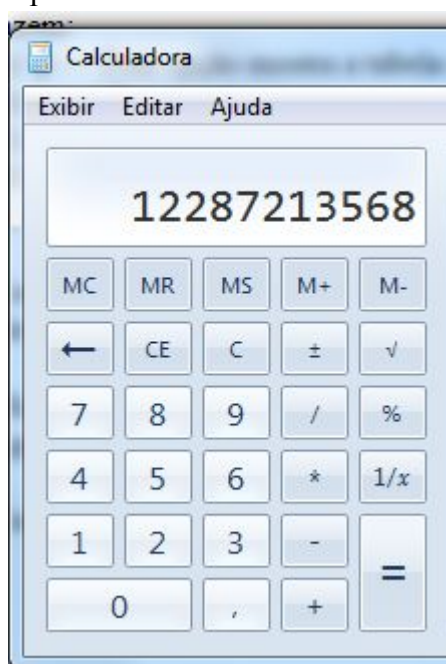
Dispositivo Inicializar Início Fim Setores Tamanho Id Tipo
/dev/sda1 2048 3999743 3997696 1,9G 82 Linux swap / Solari
/dev/sda2 * 3999744 24000511 20000768 9,6G 83 Linux

Comando (m para ajuda):
```

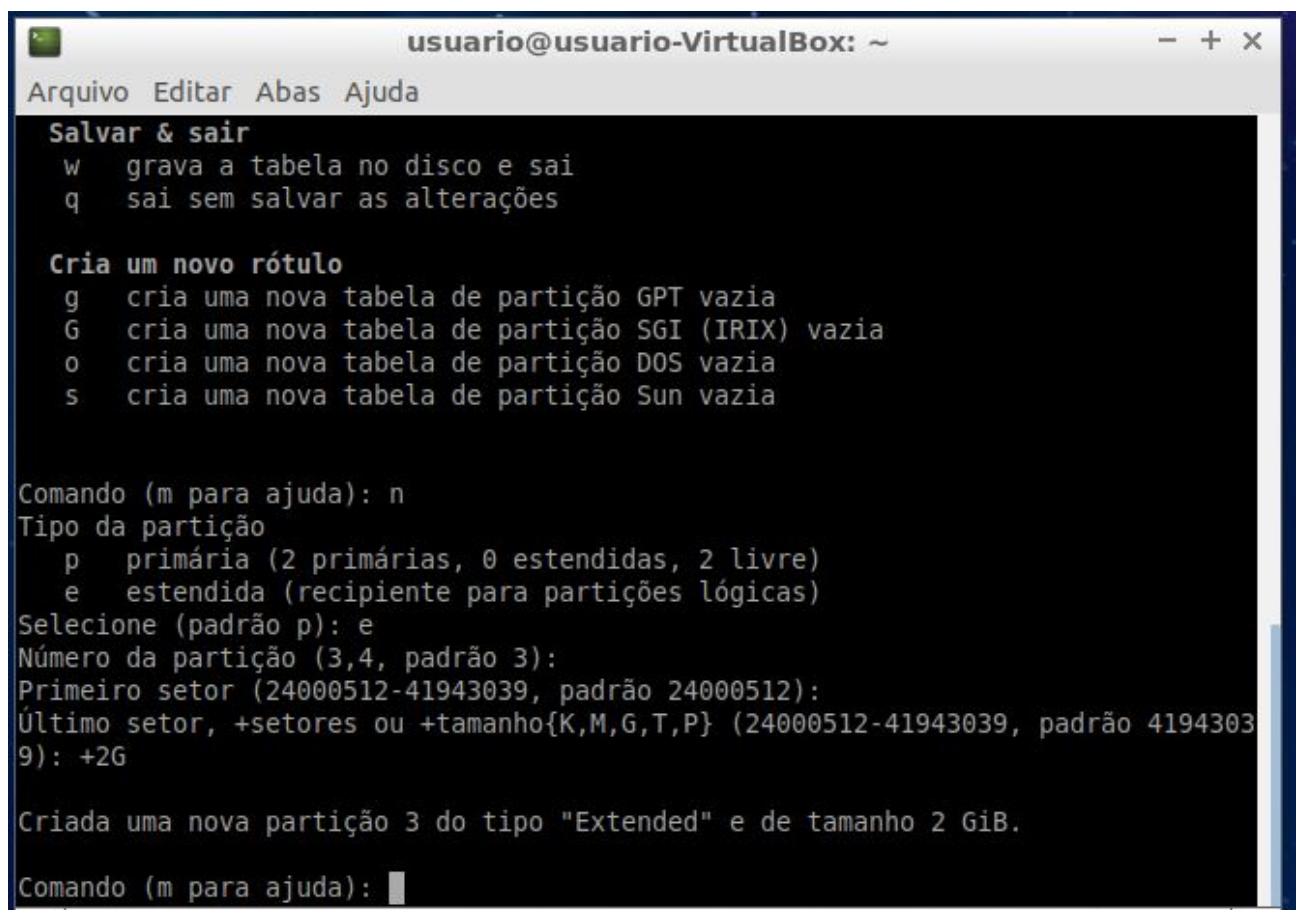
Multiplique a quantidade de blocos pelo tamanho do bloco. O que o valor resultante significa? É o valor esperado? Justifique.

o valor resultante é o especificado no resultado da calculadora.

através da calculadora vemos que o valor esperado não é exatamente a mesma coisa que o tamanho total mas sim a aproximação, fundamentando que o tamanho exatamente o esperado



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



```
usuario@usuario-VirtualBox: ~
Arquivo  Editar  Abas  Ajuda

Salvar & sair
w  grava a tabela no disco e sai
q  sai sem salvar as alterações

Cria um novo rótulo
g  cria uma nova tabela de partição GPT vazia
G  cria uma nova tabela de partição SGI (IRIX) vazia
o  cria uma nova tabela de partição DOS vazia
s  cria uma nova tabela de partição Sun vazia

Comando (m para ajuda): n
Tipo da partição
  p  primária (2 primárias, 0 estendidas, 2 livre)
  e  estendida (recipiente para partições lógicas)
Selecione (padrão p): e
Número da partição (3,4, padrão 3):
Primeiro setor (24000512-41943039, padrão 24000512):
Último setor, +setores ou +tamanho{K,M,G,T,P} (24000512-41943039, padrão 41943039): +2G

Criada uma nova partição 3 do tipo "Extended" e de tamanho 2 GiB.

Comando (m para ajuda):
```

```
usuario@usuario-VirtualBox: ~
Arquivo Editar Abas Ajuda
Tipo da partição
  p  primária (2 primárias, 1 estendidas, 1 livre)
  l  lógica (numerado a partir de 5)
Selecione (padrão p):

Usando resposta padrão p.
Selecionou a partição 4
Primeiro setor (28194816-41943039, padrão 28194816):
Último setor, +setores ou +tamanho{K,M,G,T,P} (28194816-41943039, padrão 41943039): +2G

Criada uma nova partição 4 do tipo "Linux" e de tamanho 2 GiB.

Comando (m para ajuda): n
Todas as partições primárias estão em uso.
Adicionando uma partição lógica 5
Primeiro setor (24002560-28194815, padrão 24002560):
Último setor, +setores ou +tamanho{K,M,G,T,P} (24002560-28194815, padrão 28194815): +2G
Valor fora do intervalo.
Último setor, +setores ou +tamanho{K,M,G,T,P} (24002560-28194815, padrão 28194815):

Criada uma nova partição 5 do tipo "Linux" e de tamanho 2 GiB.
```

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.


```
usuario@usuario-VirtualBox: ~  
Arquivo Editar Abas Ajuda  
[-m reserved-blocks-percentage] [-o creator-os]  
[-g blocks-per-group] [-L volume-label] [-M last-mounted-directory]  
[-O feature[,...]] [-r fs-revision] [-E extended-option[,...]]  
[-t fs-type] [-T usage-type ] [-U UUID] [-e errors_behavior][-z undo_fil  
e]  
[-jnvDFSv] device [blocks-count]  
usuario@usuario-VirtualBox:~$ sudo mke2fs -t ext4 -b 400 /dev/sda3  
mke2fs: tamanho do bloco inválido - 400  
usuario@usuario-VirtualBox:~$ sudo mke2fs -t ext4 -b 4096 /dev/sda3  
mke2fs 1.44.1 (24-Mar-2018)  
Found a dos partition table in /dev/sda3  
Proceed anyway? (y,N) y  
Creating filesystem with 1048576 2k blocks and 131072 inodes  
Filesystem UUID: 3e0117c9-2216-4018-af5e-d2f3655625c9  
Cópias de segurança de superblocos gravadas em blocos:  
16384, 49152, 81920, 114688, 147456, 409600, 442368, 802816  
  
Allocating group tables: pronto  
Gravando tabelas inode: pronto  
Creating journal (16384 blocks): concluído  
Escrevendo superblocos e informações de contabilidade de sistema de arquivos: 0  
concluído  
usuario@usuario-VirtualBox:~$
```

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

- a) **-b** : este comando **-b** especifica o tamanho em bytes do bloco. Os valores que são válidos para os blocos são : 1024, 2048 e 4096. Se o tamanho do bloco for omitido pelo sistema e se for negativo o sistema a adequa pois o tamanho é necessariamente múltiplo de 2.

Utilizando o comando **man mkfs.ext4** é possível acessar o menu que vai nos mostrar as orientações de cada comando presente na função.

```
usuario@usuario-VirtualBox:~$ man mkfs.ext4
```

O print mostra o comando do tamanho do bloco, que especifica-o **-b**

```
usuario@usuario-VirtualBox: ~
Arquivo  Editar  Abas  Ajuda

The defaults of the parameters for the newly created filesystem, if not
overridden by the options listed below, are controlled by the
/etc/mke2fs.conf configuration file.  See the mke2fs.conf(5) manual
page for more details.

OPTIONS
-b block-size
    Specify the size of blocks in bytes.  Valid block-size values
    are 1024, 2048 and 4096 bytes per block.  If omitted, block-size
    is heuristically determined by the filesystem size and the
    expected usage of the filesystem (see the -T option).  If block-
    size is preceded by a negative sign ('-'), then mke2fs will use
    heuristics to determine the appropriate block size, with the
    constraint that the block size will be at least block-size
    bytes.  This is useful for certain hardware devices which
    require that the blocksize be a multiple of 2k.

-c      Check the device for bad blocks before creating the file system.
    If this option is specified twice, then a slower read-write test
    is used instead of a fast read-only test.

-C cluster-size
    Specify the size of cluster in bytes for filesystems using the
Manual page mkfs.ext4(8) line 37 (press h for help or q to quit)
```

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
C pias de seguran a de superblocos gravadas em blocos:
    32768, 98304, 163840, 229376, 294912

Allocating group tables: pronto
Gravando tabelas inode: pronto
Creating journal (8192 blocks): concl ido
Escrevendo superblocos e informa  es de contabilidade de sistema de arquivos:  
concl ido
```

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:

utilizando o comando *mkdir* ele é o local onde serão alocadas e criadas as pastas e diretórios, especificando como / **media/particaoA** (ou ...**B** ou ...**C**) indicando as pastas criadas.

a) /media/particaoA

```

usuario@usuario-VirtualBox:~$ sudo mkdir /media/particaoA
usuario@usuario-VirtualBox:~$

```

b) /media/particaoB

```

usuario@usuario-VirtualBox:~$ sudo mkdir /media/particaoB

```

c) /media/particaoC

```

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?

Usando o comando “**mount**” para montar o sistema de arquivos para que sejam mostrados nos sistemas de arquivos. Também como o comando “**Touch**” é o comando utilizado principalmente para criar arquivos vazios, e também altera data e hora de arquivos das pastas alocadas.


```

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:~$ █

```

pra finalizar o utilizaremos o comando “**chmod**” que é utilitário pela linha de comando para modificar as permissões de um arquivo ou diretórios. Podendo ser passados números e letras como parâmetros. utilizando o comando “**sudo chmod+=wrx**” permite que possa escrever executar e ler do arquivo já existente no diretório **media/particaoA**

```

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
█

^G Obter Ajud^O Gravar      ^W Onde está?^K Recort txt^J Justificar^C Pos atual
^X Sair        ^R Ler o arq  ^\ Substituir^U Colar txt ^T Verfo rtog ^_ Ir p/ linha

```

Para escrever a permissão precisa ser alterada como fiz na permissão abaixo solicitada.

```

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 +=wrx /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 +=wrx /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 +=wrx /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=remounts
# 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

```

Nesta imagem depois de reiniciar o computador, vê-se que os sistemas de arquivos criados foram montados automaticamente como pede a questão.

computador reiniciado na imagem abaixo:

