

Conteúdo: **Gerenciamento do sistema de arquivos.**

CONTEXTUALIZAÇÃO

- **Nesta aula teremos como **objetivo** aprender o seguinte:**
 - Gerenciar partições em sistemas de arquivos;
 - Manter a integridade dos sistemas de arquivos;
 - Controlar a montagem e desmontagem de sistemas de arquivos.


SISTEMAS DE ARQUIVOS

- O que é um Sistema de arquivo (Filesystem) ?
 - Um **Sistema de Arquivos** é um conjunto de rotinas utilizado para gerenciar arquivos nos **discos (HD, CD, etc)**, como por exemplo:
 - » A criação de um arquivo;
 - » A abertura de um arquivo;
 - » A gravação no arquivo;
 - » Etc.
 - O sistema operacional invoca essas rotinas sempre que precisa.

SISTEMAS DE ARQUIVOS

- Um sistema de arquivos permite agregar características a cada arquivo como:
 - Um nome;
 - Permissões de acesso;
 - Atributos especiais;
 - Um índice, que é uma lista de arquivos na partição que informa onde cada arquivo está localizado no disco;
 - Etc.

SISTEMAS DE ARQUIVOS

- O Linux oferece suporte a muitos sistemas de arquivos, entre eles:
 - Ext2;
 - Ext3;
 - Ext4;  Sendo esse o utilizado por padrão no DEBIAN.
 - ReiserFS;
 - XFS;
 - Etc.

DISCO (HD) NO LINUX

- O Linux fornece suporte a muitos tipos de discos, entre eles: IDE, SATA, SAS, SCSI e SSD;

→ São tecnologias diferentes.


Nos casos onde a tecnologia IDE é utilizada.

→ Local onde o disco é ligado.


- A maioria dos computadores tem uma interface **IDE** primária (master) e uma secundária (slave);
- Juntas, essas interfaces permitem o uso de até quatro discos (máster primário, escravo primário, máster secundário, escravo secundário);
- Os discos **SCSI** conseguem lidar com até 15 dispositivos.

→ Outra tecnologia.

DISCO (HD) NO LINUX

- **Dispositivos de disco:** por padrão o Linux define os arquivos de dispositivos IDE da seguinte maneira:
 - /dev/hd**a**  Sempre começa com “hd”
 - Master primária IDE
 - /dev/hd**b**
 - Escravo primário IDE
 - /dev/hd**c**
 - Master secundário IDE
 - /dev/hd**d**
 - Escravo secundário IDE

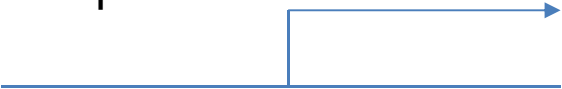
DISCO (HD) NO LINUX

- Por padrão o Linux define os arquivos de dispositivos SCSI da seguinte maneira:
 - /dev/sd**a**  Sempre começa com “sd”
 - Primeiro drive SCSI.
 - /dev/sd**b**
 - Segundo drive SCSI.
 - /dev/sd**c**
 - Terceiro drive SCSI (e assim por diante, até 15).



Veja junto com o professor na máquina o tipo de partições
Que ele possui e suas características.
pode usar o comando **fdisk -l**, e na sequencia **df -lh**
com o usuário root (usar o su – para se tornar root)

PARTIÇÕES EM SISTEMAS DE ARQUIVOS

- É a divisão do disco em partes.  Vimos na instalação, lembram?
- O Linux permite até quatro partições primárias;
- Uma partição primária pode ser substituída por uma **partição estendida**, que no geral pode conter até 64 partições (chamadas de partições lógicas);
 - O Linux porém limita as partições por dispositivo a 15 partições para discos SCSI (3 usadas como partições primárias, 12 partições lógicas), e 64 partições num dispositivo IDE (3 usadas como partições primárias, 1 como estendida e 60 partições lógicas);

PARTIÇÕES EM SISTEMAS DE ARQUIVOS

- **Partições primárias:** esse tipo de partição contém um sistema de arquivos.
- Se todas as quatro partições primárias existirem em um disco IDE, elas são numeradas da seguinte forma (se tivesse mais um disco, estaria ligado em outro drive IDE, aí teríamos um “hdb”):

- /dev/hda1;
- /dev/hda2;
- /dev/hda3;
- /dev/hda4 .

Começou com “hd”, significa que esse disco está ligado em um interface IDE.
1, 2, 3 e 4 são as partições do disco

- Note: o tipo de partição, bem como o tipo do dispositivo (IDE, SATA, etc) afeta o **nome do dispositivo** que o Linux usa para acessar a partição.

PARTIÇÕES EM SISTEMAS DE ARQUIVOS

- **Partições estendidas:** é uma variante de uma partição primária, **mas não é capaz de conter um sistema de arquivos**;
- Uma partição estendida contém partições lógicas (**as lógicas podem ter um sistema de arquivos**);
 - **Apenas uma partição estendida pode existir em um disco físico**;
- **Partições lógicas:**
 - Existem dentro da partição estendida;
 - São numeradas de 5 a 16 em discos SCSI, e de 5 a 64 no IDE, por exemplo.

PARTIÇÕES EM SISTEMAS DE ARQUIVOS

- As partições de um disco com **uma primária, uma estendida e quatro lógicas**, poderiam ser numeradas da seguinte forma:
 - /dev/hda1 (primária);
 - /dev/hda2 (estendida);
 - /dev/hda5 (lógica);
 - /dev/hda6 (lógica);
 - /dev/hda7 (lógica);
 - /dev/hda8 (lógica).
- Obs: se fosse em um HD SCSI, o **hda** seria substituído por **sda**.

PARTIÇÕES EM SISTEMAS DE ARQUIVOS

Exemplo de esquema de particionamento:

Partição	Tipo	Sistema de arquivo montado	Tamanho
/dev/hda1	Primária	/boot	100MB
/dev/hda2	Primária	/	60GB
/dev/hda3	Primária	swap	1GB
/dev/hda4	Estendida	-	-
/dev/hda5	Lógica	/var	2GB
/dev/hda6	Lógica	/tmp	1GB
/dev/hda7	Lógica	/home	100GB

- Uma vez particionado o disco (na instalação), deve-se usar softwares específicos para modificar as partições. Também é recomendado usar o LVM para estender o disco.

CRIANDO PARTIÇÕES

- Existem dois aplicativos básicos para o particionamento de disco: **fdisk** e o **cfdisk**;
- **Esse aplicativos são usados quando colocarmos um disco em nosso sistema Linux depois da instalação e precisamos criar partição nesse disco.**
- Sintaxe:
 - **fdisk** <nome do dispositivo >
 - Obs: /dev/hda é o dispositivo e /dev/hda1 é a partição.


Importante

CRIANDO PARTIÇÕES

- **Exemplo 1:** exibindo a tabela de partições de hda
 - **`fdisk /dev/sdb`**
- **Exemplo 2:** criando uma partição primária em `/dev/hda1`, uma estendida em `/dev/hda2` e uma lógica em `/dev/hda5`
 - **`fdisk /dev/sdb`**
- Mas normalmente, quando um disco é colocado após a instalação, cria-se um única partição.

CRIANDO PARTIÇÕES

- fdisk opções frequentes:
 - m : exibe um menu;
 - l: lista os tipos de partições conhecidas;
 - n: adiciona uma nova partição;
 - p: exibe a tabela de partições;
 - w: salva a tabela de partições no disco e sai;
 - q: sai sem salvar as modificações.



Junto com o professor , utilize o **fdisk** no dispositivo que já está em seu Linux, mas Somente para visualizar, sem alterar nada.

CRIANDO SISTEMAS DE ARQUIVOS EM PARTIÇÕES

- Uma vez particionado o disco pode-se criar sistemas de arquivos nessas partições
 - Que é na verdade formatar a partição com um determinado sistema de arquivos;
- O utilitário para criar partições é o **mkfs**;
- **Sintaxe**
 - **mkfs -t *fstype* fs_opções dispositivo**
 - *fstype*: tipo do sistema de arquivos, se ele for omitido o ext2 é o padrão.

CRIANDO SISTEMAS DE ARQUIVOS EM PARTIÇÕES

- Opções frequentes do mkfs
 - -c , verifica se tem bad blocks
 - -q, modo silencioso
 - -v, modo verbose
 - Obs: usar o mkfs -t ext3 ou ext4, é a mesma coisa que usar o mkfs.ext3 ou mkfs.ext4.

CRIANDO SISTEMAS DE ARQUIVOS EM PARTIÇÕES


- **Exemplo:** cria um sistema de arquivos ext4 na partição /dev/sdb1 existente, verificando se há bad blocks e com saída verbose completo:
- **mkfs.ext4 -cv /dev/sdb1**

CRIANDO SISTEMAS DE ARQUIVOS EM PARTIÇÕES

Praticando

- Junto com o professor, utilizando o VirtualBox, inclua um novo disco no seu Linux, crie uma partição nele com o **fdisk** e crie um sistema de arquivos ext4 nessa partição, junto com a formatação da mesma com o **mkfs**.

MONTAGEM E DESMONTAGEM DE PARTIÇÕES

- As partições e sistemas de arquivos recém criadas podem ser visualizadas, montadas e utilizadas.
 Temos que montar a partição para poder usar ela.
- Montando manualmente com o utilitário **mount**
 - **Sitaxe:** mount “opções” “dispositivo” “diretório”
 - **Exemplo 1:** montar as partições recém criadas em /home/particao1
Ex: **mount** /dev/sdb1 /home/particao1
 - **Exemplo 2:** exibir os sistemas de arquivos atualmente montados
 - **mount**
 - **Obs:** antes de montar as partições, as mesmas devem ser formatadas.

MONTAGEM DE DESMONTAGEM DE SISTEMAS DE ARQUIVOS

- **Desmontando sistemas de arquivos**
 - **Sintaxe:** `umount` “opções ” “dispositivo”
 - Exemplo 1: `umount /home/particao1`
 - Exemplo 2: `umount /home/particao2`

MONTAR AS PARTIÇÕES AUTOMATICAMENTE NO MOMENTO DO BOOT

- Quando incluimos um novo disco depois da instalação do Linux, precisamos fazer com que esse disco seja montado sempre o sistema seja reiniciado;
- Pois a montagem manual é perdida sempre que o sistema é reiniciado;
- Para montar automaticamente deve-se incluir os pontos de montagem no arquivo **/etc/fstab**
 - Todos os pontos de montagens descritos neste arquivo são verificados quando o sistema é inicializado.

MONTAR AS PARTIÇÕES AUTOMATICAMENTE NO MOMENTO DO BOOT

- Exemplo do /etc/fstab:

# <file system>	< dir>	<type>	<options>	<dump>	<pass>
/dev/sda1	/	ext4	defaults	1	1
/dev/hdxx	/usr	ext4	defaults	1	1
/dev/sda5	swap	swap	defaults	0	0

MONTAR AS PARTIÇÕES AUTOMATICAMENTE NO MOMENTO DO BOOT

- Dispositivo;
- Ponto de montagem;
- Tipo de sistema de arquivos;
- Opção de montagem: uma lista de opções, separadas por vírgulas, para serem executadas no momento da montagem (ex: noexec, ro, rw, etc);
- Frequência do dump: informa se o utilitário de backup (conhecido como dump) para o sistema de arquivo está ativo (1) ou não (0).
- Número de passe para o fsck: representa a checagem do dispositivo na busca de erros no momento da inicialização do Sistema Operacional.

EXERCÍCIOS

1. Exiba as tabelas de partições dos seus discos;
2. Adicione um disco de 1GB ao virtualbox;
3. Crie uma partição primária de 500MB, uma estendida e uma lógica com o restante do disco adicionado;
4. Verifique se as partições foram criadas corretamente (**fdisk -l**);
5. Crie um sistema de arquivos ext4 na partição primária e um ext3 na partição lógica;
6. Monte as duas partições criadas em “/home/particao3 e /home/particao4 ”;
7. Insira algum conteúdo nas partições criadas, desmonte as partições montadas, monte novamente e verifique os conteúdos;
8. Monte as partições criadas para serem automaticamente montadas no momento do boot (reinicie máquina e teste).

MANTENDO A INTEGRIDADE DOS SISTEMAS DE ARQUIVOS

- Com o passar do tempo, os sistemas de arquivos/partições podem desenvolver problemas, tais como:
 - Sistema de arquivos com a capacidade lotada;
 - Sistema de arquivos corrompidos (devido a uma queda de energia por exemplo);
 - Ficar sem inodes (assim não pode-se criar novos objetos no sistema);
- Monitorar cuidadosamente e verificar os sistemas de arquivos do Linux regularmente, podem ajudar a prevenir e corrigir esses tipos de problemas.

MANTENDO A INTEGRIDADE DOS SISTEMAS DE ARQUIVOS


- **Inodes:**
- inodes são estruturas responsáveis por conter informações básicas sobre cada arquivos e pastas, como **permissões de acesso, identificação dos donos dos arquivos, data e hora do último acesso e alterações, tamanho** e, o mais importante, os famosos ponteiros para o arquivo em si;
- De modo geral, o Inode é a identidade de um arquivo ou diretório, é uma identificação única para ele;
- Ao ler qualquer arquivo ou diretório, o Kernel trata de ler primeiramente o Inode do arquivo, para depois chegar no arquivo ou diretório.

MANTENDO A INTEGRIDADE DOS SISTEMAS DE ARQUIVOS

- **Inodes:**
- Todo sistema de arquivos contém um número finito de inodes, que no geral é muito grande;
- Mas é possível ficar sem inodes se uma partição tiver muitos arquivos pequenos
- O comando “**df**” fornece informações sobre o status tanto da utilização do disco, como dos inodes.

MANTENDO A INTEGRIDADE DOS SISTEMAS DE ARQUIVOS

- Comando **df**
 - **Sintaxe:** df “opções”
 - Opções frequentes:
 - - h: fornece os resultados em formatos legíveis, incluindo sufixos em MB e GB;
 - - i: exibe informações sobre os inodes restantes.
- **Exemplo 1:** df -h
- **Exemplo 2:** df -i
- **Exemplo 3:** df . (ver somente a partição atual)

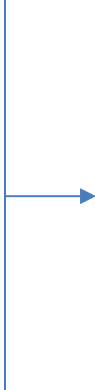


Fazer os
exemplos
junto com
o professor.

MANTENDO A INTEGRIDADE DOS SISTEMAS DE ARQUIVOS

- **Monitorando o uso do disco**
- O comando utilizado para monitorar é o “**du**”
- O “**du**” monitora recursivamente os diretórios e relata informações sobre a quantidade de espaço consumido.
- Sintaxe:
 - **du** “opções” “diretórios”

MANTENDO A INTEGRIDADE DOS SISTEMAS DE ARQUIVOS

- **Opções frequentes:**
 - -a mostra os arquivos, não somente os diretórios;
 - -h: fornece os resultados em formatos legíveis, incluindo sufixos em MB e GB;
 - -c: reproduz o total geral para todos os itens listados;
 - -s: exibe um resumo para cada um dos diretórios especificados, em vez de totais para cada subdiretório encontrado recursivamente;
 - -S: Exclui subdiretórios , limitado aos totais do diretório.
 - **Exemplo 1:** `du - h /home`
 - **Exemplo 2:** `du - ah /home/*`
 - **Exemplo 3:** `du - csh /home`
 - **Exemplo 4:** `du - csh /home/*`
- 
- Fazer os exemplos junto com o professor.

VERIFICANDO E CONSERTANDO SISTEMAS DE ARQUIVOS

- Não importa o quanto sejam estáveis, os computadores eventualmente falham (ex: desligam abruptamente);
- Esse tipo de falhas e outras podem causar inconsistências nos sistemas de arquivos;
- Uma vez com falhas, os sistemas de arquivos precisam ser verificados e corrigidos;
- O utilitário utilizado para isso é o **fsck**

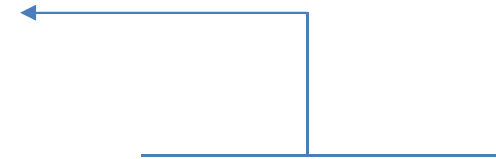
VERIFICANDO E CONSERTANDO SISTEMAS DE ARQUIVOS

- **fsck** (**fsck.ext2**, **fsck.ext3**, **fsck.ext4**, **fsck.msdos**, etc)
- **Sintaxe**
 - **fsck** “opções” “sistemas de arquivos”
- **Principais opções**
 - **-A:** Roda verificações em todos os sistemas de arquivos montados em /etc/fstab (essa opção é usada no momento do boot, antes dos sistemas de arquivos serem montados);
 - **-N:** não executa, mas mostra o que seria feito;
 - **-t:** especifica o tipo do sistemas de arquivo (alternativa para fsck.ext2, fsck.ext3);
 - **-c:** verifica se há bad blocks;
 - **-f:** força a verificação (mesmo que o sistema de arquivos pareça OK);
 - **-P:** repara automaticamente o sistema de arquivos (sem pedir confirmação ao usuário);
 - **-y:** Responde “yes” para todos os prompts interativos;

VERIFICANDO E CONSERTANDO SISTEMAS DE ARQUIVOS

Exemplo 1:

A partição deve estar desmontada
para que possa ser verificada



Verifica o sistema de arquivos ext4 em /dev/hda5, **o qual não se encontra montado.**

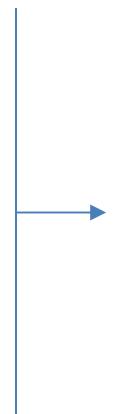
fsck /dev/hda5

A partição não tinha problemas, portanto o fsck não a verificou

Exemplo 2:

Força uma verificação

fsck -f /dev/hda5



Junto com o Professor,
desmonte a partição
adicionada
anteriormente e
verifique ela.

VERIFICANDO E CONSERTANDO SISTEMAS DE ARQUIVOS

- **Importante:**
- Quando o Linux faz o boot, o kernel realiza uma verificação de todos os sistemas de arquivos em `/etc/fstab`, usando a **opção -A do fsck** (a não ser que a entrada de `/etc/fstab` contenha a opção `noauto`);
- Se um erro for encontrado, o sistema passa para o modo “single user” para que o administrador possa usar o **fsck** manualmente;
- Em alguns casos um sistema de arquivos poderá ser irreparável.

EXERCÍCIOS

1. Verifique a utilização do espaço em disco de todos os sistemas de arquivos;
2. Verifique a utilização dos inodes de todos os sistemas de arquivos;
3. Exiba a utilização total, incluindo os arquivos em /etc;
4. Exiba somente a utilização total do /etc.
5. Exiba um resumo de todos os subdiretórios /home, mas classifique os resultados exibidos em ordem do maior para o menor;
6. Utilize o **fsck** (na partição que você adicionou), forçando a verificação e permitindo que ele realize automaticamente todos os reparos necessários.