

Parte 1 e 2

Parte 1

Aula 1 - 26/04/2022

- Unix is a operational system released at 1969.
- multi-task
- multi users

3 Níveis

Kernel

- o responsável por fazer a interação entre o hardware e software.
- começa a funcionar assim que o computador é ligado

Shell

Interface entre o sistema operacional, o usuário e o núcleo do sistema (Kernel)

Função

- ler a linha de comando
- executar o comando
- devolver o resultado

Aplicativos

- Softwares

Tudo o que fazemos no unix é um processo, cada processo tem um identificador unico PID. (process identifier)

Tudo é um arquivo (diretorios são arquivos, arquivos são arquivos e dispositivos são arquivos)

Nível mais alto o Root directory.

Todos os outros estão baixo dele

	/ ROOT (RAIZ)
BOOT	ARQUIVOS DE CONFIGURAÇÃO PARA INICIALIZAÇÃO
BIN	APLICATIVOS BINÁRIOS
DEV	ARQUIVOS DO DISPOSITIVO
ETC	SCRIPTS DE INICIALIZAÇÃO
HOME	DIRETÓRIO HOME PARA DIFERENTES USUÁRIOS
PROC	DIRETÓRIO DINAMICO ESPECIAL, MANTEM INFORMAÇÕES SOBRE O ESTADO DO SISTEMA
ROOT	DIRETÓRIO HOME DO USUÁRIO ROOT
SBIN	BINÁRIOS DE SISTEMAS IMPORTANTES
TMP	ARQUIVOS TEMPORÁRIOS
USR	APLICATIVOS E ARQUIVOS DISPONÍVEIS A TODOS USUÁRIOS
VAR	ARQUIVOS VARIÁVEIS - LOGS E BANCO DE DADOS

Comando	Significado
ls	lista arquivos e diretórios
ls -a	lista todos os arquivos e diretórios
mkdir	cria um diretório
cd <i>diretório</i>	muda para o diretório nomeado
cd	muda para o diretório inicial
cd ~	muda para o diretório inicial
cd ..	muda para o diretório pai
pwd	mostra o caminho do diretório atual
cp <i>arquivo1 arquivo2</i>	copia arquivo1 e o chama de arquivo2
mv <i>arquivo1 arquivo2</i>	move ou renomeia arquivo1 para arquivo2
rm <i>arquivo</i>	remove um arquivo
rmdir <i>diretório</i>	remove um diretório
cat <i>arquivo</i>	exibe um arquivo
less <i>arquivo</i>	exibe um arquivo, uma página por vez
head <i>arquivo</i>	exibe as primeiras linhas de um arquivo
tail <i>arquivo</i>	exibe as últimas linhas de um arquivo
grep ' <i>palavras-chave</i> ' <i>arquivo</i>	procura por palavras-chave no arquivo
wc <i>arquivo</i>	conta o número de linhas / palavras / caracteres no arquivo
*	corresponde a qualquer número de caracteres
?	corresponde a um caracter
man <i>comando</i>	lê a página do manual online do comando informado
whatis <i>comando</i>	breve descrição do comando informado
apropos ' <i>palavra-chave</i> '	corresponde a comandos com palavras-chave em suas páginas de manual

Comandos

less - abrir menos arquivos - Cria uma pagina com os arquivos, para pular de paginas clique em barra de espaço

PAra finalizar clicar a tecla Q

head - mostra os dez primeiros valores

head -5 - traz apenas 5 primeiros valores

tail - dez ultimos valores

tail -5 exhibe os 5 ultimos valores

grep -i (ignora o case sensitive) ' ' - procura por frases, colocar dentro de aspas simples

grep -i v - trz todos os valores que nao estao dentro das aspas

grep -in mostra linha onde está

grep -ic quantidades de vezes que é repetido

wc -w -quantidades de palavras

wc -l quantidade de linhas

wc -c quantidade de caracteres

Utilities	
Shell	
Kernel	
Hardware	

completar o nome com o TAB

touch criar arquivo

renomear MV

. é o diretorio atual

.. é o diretório externo

rm -rf

Recapitulando...

Agora que você passou um bom tempo estudando sobre o Unix e seus comandos, sabemos que você já entende que:

- O Unix é um sistema operacional que inspirou muitos outros sistemas, como MacOS, Linux, Solaris etc.;
- O Unix é composto basicamente pelo Shell, Kernel e os programas;
- A estrutura de diretórios do Unix é organizada no formato de uma árvore invertida, de forma hierárquica, iniciando pelo **root** (raiz);
- Tudo no Unix é um arquivo ou um processo;
- O terminal é seu principal aliado para ter produtividade em ambiente Unix;

Depois de entender como abrir e utilizar o terminal, você aprendeu na prática como utilizar os comandos Bash. Vamos relembrar alguns dos comandos que você aprendeu a usar.

Um bom exercício é, mentalmente, tentar descrever o que cada comando faz.

- `ls`, `mkdir`, `cd`, `pwd`, `cp`, `mv`, `rm`, `rmdir`, `clear`, `cat`, `less`, `head`, `tail`, `grep`, `wc`, `>`, `>>`, `|`, `sort`, `who`, `man`, `whatis`, `apropos`, `*`, `?`, `chmod`, `ps`, `bg`, `kill`, `find`, `history`.

Parte 2

Parte 2

Aula 2 - dia 26/04/2022

Inputs

cat - pra abrir um arquivo

cat + enter - Espera um input do usuário, entrada de dados

Algo que nós inserimos, recebe valores que o usuario coloca, inputs

A gente pode criar esses inputs e atribuir esse valores , para um outro arquivo

> insere valores

>> acrescentar mais valores ao arquivo criado

sort -ordena os valores, sort < altera a ordem de visualização, output

R- Read - leitura do arquivo ou diretório

W- write - editar um arquivo ou modificar o conteúdo de um diretório

X - Execute - permite executar um arquivo ou acessar um diretório pelo comando

CD

- Hífen - no início indica um arquivo comum

D - no inicio indica um diretorio

Copiar

Permissões	Links	Proprietário	Grupo	Tamanho	Data e Hora	Nome
drwxr-xr-x	2	root	root	4096	Out 19 09:10	composer/

- **Permissões** => É possível verificar o tipo do item e nível de permissão para Leitura, Escrita e Execução de item ou diretório;
- **Links** => Número de ligações que o item possui, no caso do diretório, número de subdiretórios que possui;
- **Proprietário** => Quem é a pessoa dona, quem criou. É o diretório padrão da pessoa usuária, o home;
- **Grupo** => Grupo ao qual pertence o item ou diretório. Utilizado para dar permissões à outras pessoas;
- **Tamanho** => Em Bytes;
- **Data e Hora** => Momento em que foi criado ou última modificação;
- **Nome** => Nome do item ou diretório;

Você aprendeu no vídeo como alterar as permissões no modo literal (caracteres), mas também é possível usar o modo octal. Para compreendermos esta outra forma, precisamos entender que eles são administrados por meio de valores como descrito abaixo:

- Leitura **r** - 4
- Escrita **w** - 2
- Execução **x** - 1

Esses valores são permissões com base em bits de **ligados = 1** e **desligados = 0** ,

Para listar todos os arquivos que terminam em .txt

```
find . -name "*.txt"
```

Para localizar todos os diretórios

```
find . -type d
```

Para localizar todos os arquivos

```
find . -type f
```

Localizar tanto arquivos quanto diretórios que comecem

por algum trecho

```
find ./teste -name "exemplo*"
```

Resultado

```
./teste/exemplo.txt
```

```
./teste/exemplo
```

Localizar somente arquivos que comecem por algum trecho

```
find ./teste -type f -name "exemplo*"
```

Resultado

```
./teste/exemplo.txt
```

```
# Localizar somente diretórios que comecem por algum trecho
```

```
find ./teste -type d -name "exemplo*"
```

```
# Resultado
```

```
./teste/exemplo
```

O `history` é um comando que mostra o histórico de comandos que você executou no terminal. A quantidade ou o tamanho desse _"histórico"_ podem ser configurados para um número arbitrário de comandos ou para ver todo o histórico.

```
# Mostra o histórico de comandos
```

```
history
```

```
# Pegar os últimos 10 comandos
```

```
history | tail
```

O `echo` é um comando utilizado em scripts ou no terminal para exibir mensagens na tela ou em um arquivo.

```
echo "Este é um teste"
```

```
# Resultado
```

```
Este é um teste
```

```
# Pode ser usado para colocar textos dentro de arquivos.
```

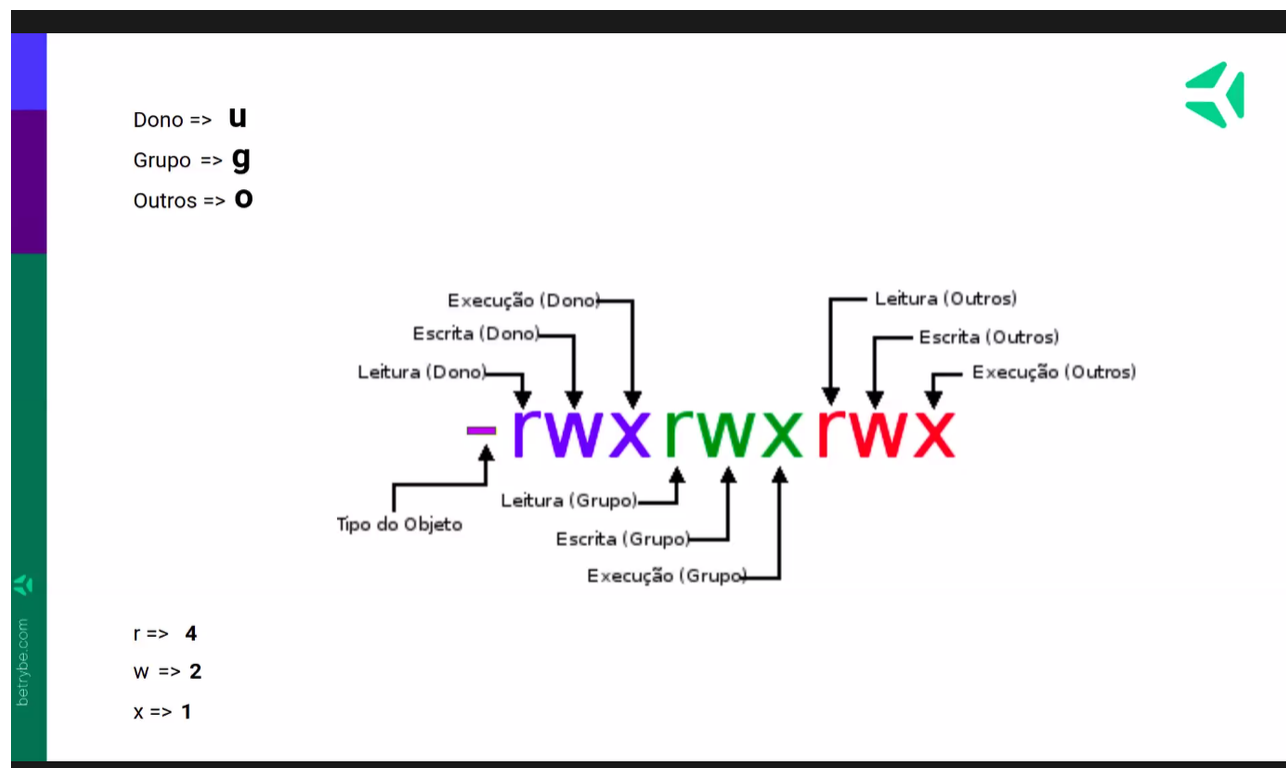
```
echo "Este é mais um teste" > teste.txt
```

```
cat teste.txt
```

```
Este é mais um teste
```

Comando	Significado
comando > arquivo	redireciona a saída padrão para um arquivo
comando >> arquivo	anexa saída padrão a um arquivo
comando < arquivo	redireciona a entrada padrão de um arquivo
comando1 comando2	canaliza a saída do comando1 para a entrada do comando2
cat arquivo1 arquivo2 > arquivo0	concatena arquivo1 e arquivo2 em arquivo0
sort	ordena dados
who	lista os usuários atualmente logados
ls -lag	lista direitos de acesso para todos os arquivos
chmod [options] arquivo	altera os direitos de acesso para o arquivo nomeado
comando &	executa o comando em segundo plano
^C	elimina o trabalho em execução em primeiro plano
^Z	suspende o trabalho em execução em primeiro plano
bg	retorna processos que estão suspensos
jobs	lista processos suspensos e em segundo plano
fg %1	reinicia o primeiro processo suspenso
kill %1	encerra o primeiro processo suspenso
ps	lista processos atuais
kill 26152	encerra o processo de número 26152

Tabela com os comandos do dia



rwX = 111 (7 | Acesso Total)

r-- = 100 (4 | Somente Leitura)
-w- = 010 (2 | Somente Escrita)
--x = 001 (1 | Somente Execução)
rw- = 110 (6 | Somente Leitura e Escrita)
r-x = 101 (5 | Somente Leitura e Execução)
-wx = 011 (3 | Somente Escrita e Execução)
--- = 000 (0 | Todos Acessos Negados)

ps processos

sleep

sleep &