

**Título:** Atividade nº 01



**Data:** 23/10/2023

**Grupo de no Máximo de 5 até 8 alunos**

**Nomes:**

Lucas Ferreira Andrade – 4231921505

Pedro Castro Melo - 4231925168

Bruno Batista Satyro de Castro – 4231922400

Bernardo Marconi Santos De Almeida – 4231920332

Maria Clara Marques Lino – 4231924407

Renzzo Silva Rocha – 4231925318

Erick Oliveira – 4231924394

Filipy da Silva Furtado – 42321649

Victor Henrique Dias – 4231920004

Rogério Lopes da Silva Filho - 4231925527

**Tema:** Atividade de Interação

**Objetivo:** Acompanhamento dos projetos

**Roteiro:** Formar grupos e desenvolver as atividades propostas.

## 8. PROPOSTA DE ATIVIDADES

8.1. Comando “pwd” para certificar em qual diretório estamos localizados.

```
kali@kali: ~  
File Actions Edit View Help  
(kali@kali)-[~]  
$ pwd  
/home/kali
```

- Criação da estrutura do diretório “sor” com as pastas bimestre1 e bimestre 2.

```
kali@kali: ~/sor  
File Actions Edit View Help  
(kali@kali)-[~]  
$ mkdir -p /home/kali/sor/bimestre1  
(kali@kali)-[~]  
$ cd sor  
(kali@kali)-[~/sor]  
$ mkdir bimestre2
```

- Criação da estrutura do diretório “poo” com as pastas bimestre1 e bimestre 2.

```
(kali㉿kali)-[~]  
$ mkdir -p /home/kali/poo/bimestre1  
  
(kali㉿kali)-[~]  
$ cd poo  
  
(kali㉿kali)-[~/poo]  
$ mkdir bimestre2
```

a) Criação do arquivo “conteudo1bim”.

```
kali@kali: ~/sor/bimestre1  
File Actions Edit View Help  
  
(kali㉿kali)-[~]  
$ cd sor/bimestre1  
  
(kali㉿kali)-[~/sor/bimestre1]  
$ gedit  
  
(gedit:10636): Gtk-WARNING **: 11:06:03.665: Calling  
org.xfce.Session.Manager.Inhibit failed: GDBus.Error:  
org.freedesktop.DBus.Error.UnknownMethod: No such met  
hod “Inhibit”  
  
(kali㉿kali)-[~/sor/bimestre1]  
$
```

b) Criação do arquivo “hello.c”.

```
kali@kali: ~/sor/bimestre2  
File Actions Edit View Help  
  
(kali㉿kali)-[~/sor/bimestre2]  
$ gedit  
  
(gedit:39730): Gtk-WARNING **: 12:05:45.264: Calling  
org.xfce.Session.Manager.Inhibit failed: GDBus.Error:  
org.freedesktop.DBus.Error.UnknownMethod: No such met  
hod “Inhibit”  
  
(kali㉿kali)-[~/sor/bimestre2]  
$
```

c) O arquivo foi compilado com sucesso pelo comando dado.

```
kali@kali: ~/sor/bimestre2
File Actions Edit View Help
(kali@kali)-[~/sor/bimestre2]
$ ls
hello.c

(kali@kali)-[~/sor/bimestre2]
$ gcc hello.c -o executavel
```

d) O arquivo foi executado e exibiu a mensagem salva no arquivo “hello.c”.

```
kali@kali: ~/sor/bimestre2
File Actions Edit View Help
(kali@kali)-[~/sor/bimestre2]
$ ./executavel
Estou aprendendo Linux e a linguagem C
```

e) Realizada a cópia dos arquivos para as respectivas pastas, conforme solicitado no exercício.

```
kali@kali: ~
File Actions Edit View Help
(kali@kali)-[~]
$ cp -r /home/kali/sor/bimestre1/conteudo1bim /home/kali/poo/bimestre1

(kali@kali)-[~]
$ cp -r /home/kali/sor/bimestre2/hello.c /home/kali/poo/bimestre2
```

f) Comando “ls -R” para listar todos os arquivos no diretório, usado para confirmar se as pastas foram criadas conforme solicitado no exercício.

```
kali@kali: ~  
File Actions Edit View Help  
(kali@kali)-[~]  
$ pwd  
/home/kali  
  
(kali@kali)-[~]  
$ ls -R sor/  
sor/:  
bimestre1 bimestre2  
  
sor/bimestre1:  
conteudo1bim  
  
sor/bimestre2:  
executavel hello.c
```

\$ ls -R sor/ poo/

```
kali@kali: ~  
File Actions Edit View Help  
(kali@kali)-[~]  
$ ls -R sor/ poo/  
poo/:  
bimestre1 bimestre2  
  
poo/bimestre1:  
conteudo1bim  
  
poo/bimestre2:  
hello.c  
  
sor/:  
bimestre1 bimestre2  
  
sor/bimestre1:  
conteudo1bim  
  
sor/bimestre2:  
executavel hello.c
```

g) O comando utilizado para gravar a estrutura em árvores foi o “tree”. Abaixo o comando sendo realizado e o arquivo contendo a estrutura.

```
kali@kali: ~  
File Actions Edit View Help  
(kali@kali)-[~]  
$ pwd  
/home/kali  
  
(kali@kali)-[~]  
$ tree /home/kali/sor > minhaestrutura  
  
(kali@kali)-[~]  
$ tree /home/kali/poo >> minhaestrutura
```

```
Open  minhaestrutura  Save  
1 /home/kali/sor  
2 | bimestre1  
3 | | conteudo1bim  
4 | bimestre2  
5 | | executavel  
6 | | hello.c  
7  
8 3 directories, 3 files  
9 /home/kali/poo  
10 | bimestre1  
11 | | conteudo1bim  
12 | bimestre2  
13 | | hello.c  
14  
15 3 directories, 2 files
```

## 8.2.

a) O comando “`grep -r “printf” sor/`” filtrou todas as linhas que contêm “printf” em todos os arquivos no diretório sor e seus subdiretórios e após isso, exibe todas essas linhas no terminal.

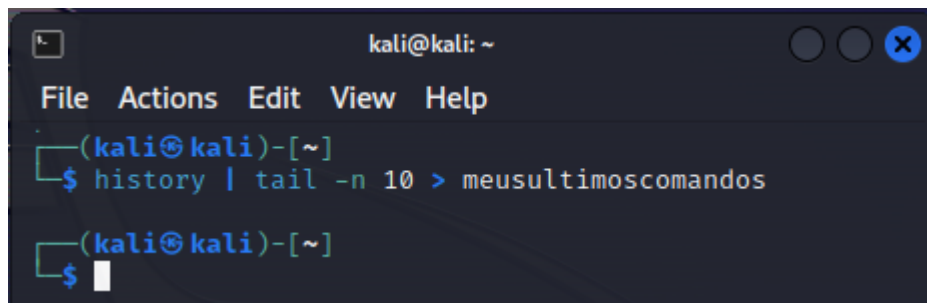
```
kali@kali: ~  
File Actions Edit View Help  
(kali@kali)-[~]  
$ pwd  
/home/kali  
  
(kali@kali)-[~]  
$ grep -r "printf" sor/  
sor/bimestre2/hello.c:printf("Estou aprendendo Linux e  
a linguagem C\n");
```

b) **Comando grep:** É usado para procurar um padrão específico em um ou mais arquivos.

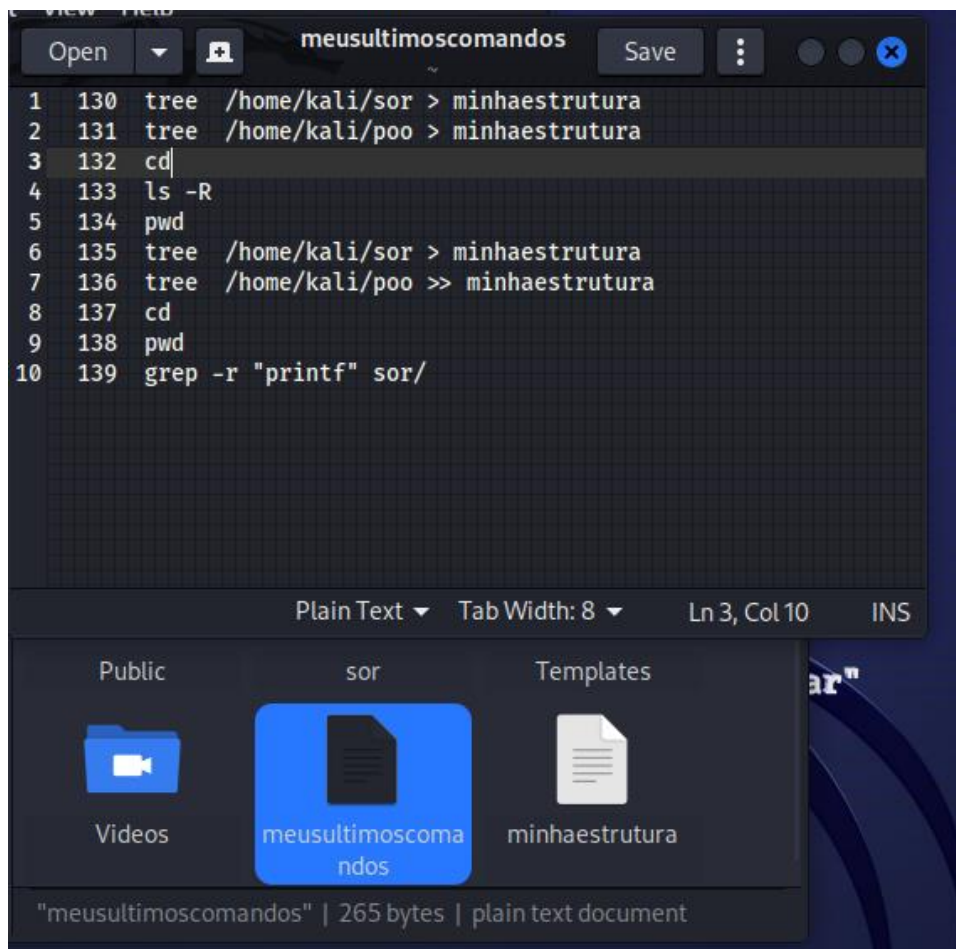
**A opção -r:** faz com que o grep procure recursivamente em todos os arquivos no diretório especificado e em seus subdiretórios.

Ao executar “`grep -r “printf” sor/`”, estamos procurando todas as ocorrências da string “printf” em todos os arquivos no diretório “sor” e em seus subdiretórios.

### 8.3.



```
kali@kali: ~  
File Actions Edit View Help  
(kali@kali)-[~]  
$ history | tail -n 10 > meusultimoscomandos  
(kali@kali)-[~]  
$
```



```
meusultimoscomandos  
1 130 tree /home/kali/sor > minhaestrutura  
2 131 tree /home/kali/poo > minhaestrutura  
3 132 cd|  
4 133 ls -R  
5 134 pwd  
6 135 tree /home/kali/sor > minhaestrutura  
7 136 tree /home/kali/poo >> minhaestrutura  
8 137 cd  
9 138 pwd  
10 139 grep -r "printf" sor/
```

Plain Text ▾ Tab Width: 8 ▾ Ln 3, Col 10 INS

Public sor Templates

Videos meusultimoscomandos minhaestrutura

"meusultimoscomandos" | 265 bytes | plain text document