

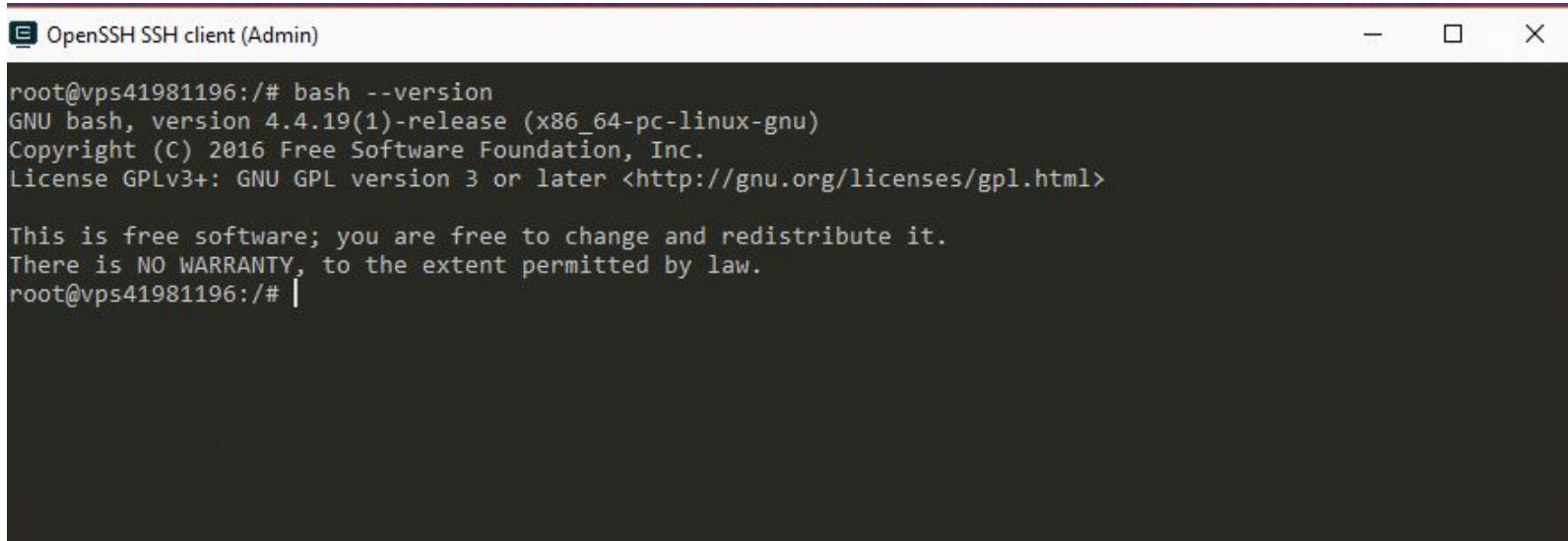


AED II

Linux, Shell, GCC, Make

Pedro do Couto Soares
pscoutosoares@gmail.com

LINUX Shell

A screenshot of an OpenSSH SSH client window titled "OpenSSH SSH client (Admin)". The window has standard Linux window controls (minimize, maximize, close) in the top right. The terminal content shows a user at the root of a server with IP vps41981196 running the command "bash --version". The output displays the GNU bash version 4.4.19(1)-release for x86_64-pc-linux-gnu, along with copyright and license information. It also includes the standard GNU disclaimer: "This is free software; you are free to change and redistribute it. There is NO WARRANTY, to the extent permitted by law." The prompt "root@vps41981196:/" is shown at the bottom with a cursor.

```
OpenSSH SSH client (Admin)

root@vps41981196:/# bash --version
GNU bash, version 4.4.19(1)-release (x86_64-pc-linux-gnu)
Copyright (C) 2016 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>

This is free software; you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
root@vps41981196:/# |
```

Bash - Linux shell



Características

- **Comandos** : Ações a serem executadas pelo shell
 - Ex: Mover uma pasta, apagar um arquivo
- **Opções/Flags**: Definem como o comando ou programa será executado
 - Ex: "RM(Comando) -rf", "LS(Comando) -la"
- **Argumento**: Informação extra passada para a execução de um comando
 - Ex: Diretório de destino, diretório que um arquivo se encontra



Conceito

..
- Indica o diretório anterior

..
- Indica o diretório atual

..
- Indica o diretório home do usuário

..
- Comentário



Segurança / Modo de acesso

- Modo usuário
- Modo root / super usuário(SUDO)

Usuário@Dispositivo:/ \$ - Indica conexão sem privilégios de root

Usuário@Dispositivo:/ # - Indica conexão com privilégios de root



Comandos - Gerenciamento de arquivos

cd - Mudar diretório

- Para retornar ao diretório inicial não utilize argumento neste comando.

ls - Listar arquivos e diretórios

- **ls -l** Listar todos os arquivos e exibir o resultado na forma de uma lista
- **ls -la** Listar todos os arquivos, incluindo arquivos ocultos e exibir na forma de lista

cp - Copiar arquivos

mv - Mover arquivos e diretórios.



Comandos - Gerenciamento de arquivos

rm - Remover arquivos

- **rm -rf** Permite a exclusão de diretório

mkdir - Criar diretório

touch - Criar arquivos

pwd - Exibe o caminho do diretório atual

diff - Compara dois arquivos linha a linha

- **diff -w** Compara os arquivos linha a linha desconsiderando espaços em branco



Redirecionamento de entrada e saída

Operador ">" : Redirecionamento de saída

Exemplo: `ls -lh > file` => Executa o comando *ls* e salva o resultado em um arquivo chamado *file*

Operador "<" : Redirecionamento de entrada

Exemplo: `mail < file` => Executa o comando *mail* recebendo como entrada o arquivo *file*

```
root@vps41981196:/home# ls      Lista todos os arquivos na pasta /home
root@vps41981196:/home# mkdir pasta1 Cria pasta chamada de "pasta1"
root@vps41981196:/home# ls      Lista novamente todos os arquivos na pasta /home
pasta1 Resultado
root@vps41981196:/home# cd pasta1 Entra na "pasta1"
root@vps41981196:/home/pasta1# touch arquivo1.txt Cria arquivo
root@vps41981196:/home/pasta1# ls
arquivo1.txt
root@vps41981196:/home/pasta1# cd .. retorna para a pasta anterior
root@vps41981196:/home# ls
pasta1
root@vps41981196:/home# rm -rf pasta1/ Remove a pasta "pasta1"
root@vps41981196:/home# ls
root@vps41981196:/home# | Não retorna nada. Pasta "home" não possui mais arquivos
```



Comandos - Modo de acesso

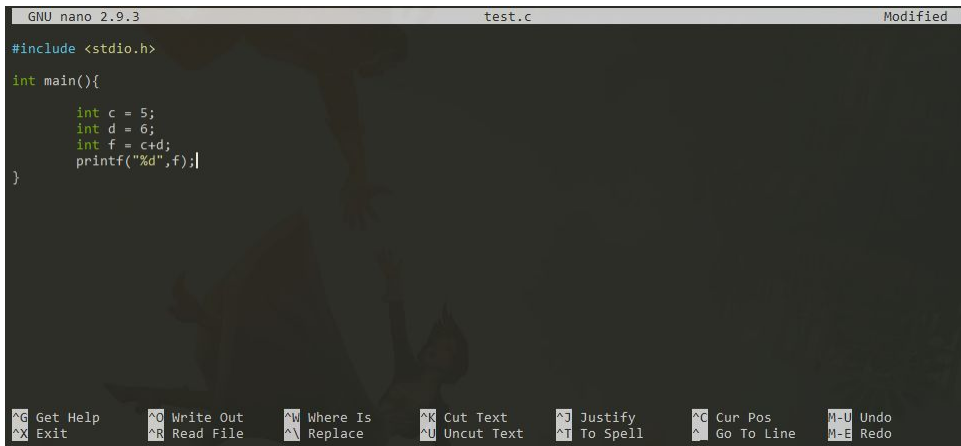
sudo - Utilizar sudo e em seguir algum comando, possibilita a execução do mesmo em um modo de super usuário, permitindo realizar quaisquer ações. (Cuidado!)

sudo su - Possibilita o Shell entrar em modo de super usuário e todos os comandos executados em seguir respeitarão os privilégios de acesso. (Cuidado!)

GCC - GNU Compiler Collection

GCC

- Executar o seguinte comando: **nano test.c**
 - Onde nano é um editor de texto e test.c é um arquivo qualquer escrito em C
 - (Você pode utilizar um editor de texto qualquer, seja gedit, sublime, etc).

A screenshot of the GNU nano 2.9.3 text editor. The window title is "GNU nano 2.9.3" and the file name is "test.c". The code displayed is a simple C program that includes <stdio.h>, defines a main function, and declares variables c, d, and f. It calculates f as the sum of c and d, and prints the value of f. The code is as follows:

```
#include <stdio.h>

int main(){

    int c = 5;
    int d = 6;
    int f = c+d;
    printf("%d",f);
}
```

The bottom status bar shows various keyboard shortcuts for editor functions: ^G Get Help, ^O Write Out, ^W Where Is, ^K Cut Text, ^J Justify, ^C Cur Pos, M-U Undo, ^X Exit, ^R Read File, ^\ Replace, ^U Uncut Text, ^T To Spell, ^_ Go To Line, and M-E Redo.

- Para salvar no nano, pressione *ctrl+ S*.
- Pressione *X* para sair do editor.

```
root@vps41981196:/home# ls
test.c
root@vps41981196:/home# gcc -o executavel test.c
root@vps41981196:/home# ls
executavel test.c
root@vps41981196:/home# ./executavel
11root@vps41981196:/home# |
```

Diagram annotations:

- Compilador (points to `gcc`)
- Flag utilizada para escolher o nome do arquivo resultante (points to `-o`)
- Arquivo a ser compilado (points to `test.c`)
- Nome do arquivo resultante (points to `executavel`)
- Arquivo resultante (points to `executavel` in the second `ls` output)
- Execute o arquivo compilado utilizando o ./ seguido do nome do arquivo resultante (points to `./executavel`)

Output do arquivo test.c

- Podemos utilizar a flag `-g` no processo de compilar para incluir informações de debug
- Ex: `gcc -o executavel test.c -g`



Makefile



Makefile

- Facilita o processo de compilar o código
- Compila apenas o necessário
- Possui estruturas bem definidas
- Apenas necessita salvar um arquivo com o nome *makefile* ou *Makefile* (sem extensão)



Makefile - Estrutura

Target: requisito-1 requisito-2 ...

[**tab**] comando

- O *Target* e o *requisito* são separados por um sinal de dois pontos
- Logo em baixo fica o *comando*, que é necessário iniciar com um tab (espaçamento não funciona)



Exemplo

```
all: executavel
```

Caso make seja executado sem argumento este será executado.

```
executavel: test.c test.h
```

```
    gcc -o executavel test.c
```

```
clean:
```

Caso make seja executado Com o argumento Clean, ele removerá

```
    rm executavel
```

o executavel.

GDB - Debugger

GDB

- Comando básico:
 - `gdb [nome do arquivo compilado]`

```
root@vps41981196:/home# gdb executavel
GNU gdb (Ubuntu 8.1-0ubuntu3) 8.1.0.20180409-git
Copyright (C) 2018 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law. Type "show copying"
and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from executavel...(no debugging symbols found)...done.
(gdb) run
Starting program: /home/executavel
12[Inferior 1 (process 8497) exited normally]
(gdb) |
```

Assim que executar o comando do gdb é necessário executar o comando **run** dentro da execução do gdb, para inicializar o processo de debug.

Caso existam erros neste processo o gdb alerta e dependendo informa a linha de erro.

Para finalizar o gdb, utilize o comando quit.