



Linux do Zero Pt1

🔗 Link	<u>Dio</u>
⚙️ Status	Em andamento
📁 Tipo	Curso

Desabilitar o hyper-v para o bom funcionamento do Virtual Box

→ Abra o cmd como administrador

```
bcdedit
```

```
# Tem que aparecer hypervisorlaunchtype off  
bcdedit /set/ hypervisorlaunchtype off
```

→ Reinicie o computador

→ Se hypervisorlaunchtype off NÃO tiver aparecendo

Painel de controle > Ativar ou desativar recursos do Windows

→ Se Plataforma do Hipervisor do Windows estiver desmarcada, MARQUE e reinicie o pc.



→ Repetir o 1º processo

Tipos de virtualização

- Local
- Nuvem
 - Ex: AWS, Azure, GPC

Utilizando o WSL

Subsistema do Windows para Linux

- Windows tem que ser PRO

Acesso Remoto a Máquina Linux

- Win → Linux
 - Software PuTTY
 - Instalar o SSH na máquina Linux
 - `apt-get install openssh-server`
- Linux → Linux
 - No terminal do Linux
 - `ssh nomeusuario@ip`

COMANDOS LINUX

reboot → Reiniciar o computador

shutdown 0 → desliga em 0 segundos

ip a → busca IP

date → retorna data e hora

clear (ctrl+L) → limpa tela

Navegando pelo sistema

pwd → mostra qual pasta estou

cd nomedapasta → vai para a pasta indicada

cd /nomedapasta → vai para a pasta raiz

cd .. → volta um diretório

cd ../nomepasta → volta pasta e abre outro

cd / → volta raiz

cd / + tab 2x → lista as pastas

cd /nome + tab → completa diretório

ls → lista os arquivos do diretório

Filtrando a exibição de arquivos

ls | more → lista de pasta

ctrl da esquerda + c → Volta do ls | more

ls nomepasta → verifica se há a pasta desejada

ls p 2x tab → Lista tudo que tem p

ls p* 2x tab → tudo criado iniciado com p

ls m?g* → retorna que tem primeira letra m e a terceira letra é g

touch arquivo.diretorio → cria arquivo em branco → deixa criar na pasta do usuário

ls arquivo[1-3].txt ou ls arquivo[1-3]* → lista do arquivo1 até arquivo3

ls arquivo[2,5]* → lista arquivo2 e arquivo5

ls arquivo[^2,^5]* → NÃO lista arquivo2 e arquivo5, somente o resto

Localizando arquivos

ls /nomepasta → lista arquivos

ls /sys → lista arquivos da pasta sys

ls /sys/kernel/p* → lista todos os arquivos que comecem com a letra p

Não sei onde está o arquivo, como buscas?

find -name → busca o arquivo pelo nome (dentro do diretório que estou)

Ex: find -name arq* → todos os arquivos que contem arq

```
matheusaganete@servidor1:/home$ find -name arq*  
matheusaganete@servidor1:/home$
```

find -name passwd* → retorna todos os arquivos que possuem passwd, pode ser que dê acesso negado, precisa buscar no diretório indicado usando **sudo**

```
find: './var/cache/apt/archives/partial': Permission denied  
find: './var/cache/ldconfig': Permission denied
```

```
matheusaganete@servidor1:/ $ sudo ls ./var/cache/ldconfig  
aux-cache  
matheusaganete@servidor1:/ $
```

Criando diretórios

mkdir nomePasta → criar pasta

mkdir nomePasta1 nomePasta2 → criará diretório dois diretórios simultaneamente

mkdir Meus Documento → criará dois diretórios Meus e Documentos

mkdir /home/matheusaganete/Planilhas

mkdir 'Meus Documentos' → criará um diretório 'Meus Documentos'

Excluindo arquivos e diretórios

rmdir nomePasta → Exclui diretório (se tiver conteúdo, não deixa excluir)

`rmdir nomePasta1 nomePasta2` → exclui diretório dois diretórios simultaneamente

`rm nomeArquivo` → exclui arquivo

`rm arquivo*` → excluí tudo que começa com arquivo

`rm *.txt` → exclui todos os arquivos txt

`rm -rf nomeDiretorio` → exclui o diretório e todos os arquivos

Obtendo ajuda

`nomeComando --help`

Exemplo ls: `ls -l` `ls -a`

`l` → use a long listing format

`a` → exibe arquivos oculto

`h` → formato humano

Exemplo rm: `rm rfvi`

`r` → remove directories and their contents recursively

`f` → force

`v` → explain what is being done

`i` → prompt before every removal

`man nomeComando` → mostra ajuda no formato de ajuda

Executando tarefas administrativas como root

`cat /etc/group` → vê grupos de usuários

Logando como usuário root

`sudo passwd root` → Atribuir senha ao usuário

`su` → mudar para usuário root(super usuário)

`su nomeUsuario` → volta para usuário comum

Liberando acesso remoto do usuário root

sshd é o serviço de acesso remoto das distribuições linux

cat /etc/ssh/sshd_config → exibe as configurações do sshd

nano /etc/ssh/sshd_config → edita as configurações do sshd

nano é um editor de notas

após mudar, precisa dar o reebot ou reiniciar o serviço

systemctl status sshd → verifica status do serviço sshd

systemctl restart sshd → reinicia o serviço sshd

Trabalhando com arquivos de texto

Editores de textos comuns do linux:

- vi
- nano

Histórico de comando

history → armazena os últimos 1000 comandos (default)

history 30 → exibi os últimos 30 comandos

!nºcomando → executa aquele comando

!! → executa o último comando

!?nomeComando?

history | grep nomeComando → busca comando ou termo usado recentemente

export HISTTIMEFORMAT="%c " → mostra data e hora do history

history -c → limpar histórico (temporario)

set +o history → limpar histórico (definivo)

set -o history → volta histórico (definivo)

Para mudar a quantidade de comando que quero armazenado

- É feito por usuário
 - nano /home/usuario/.bahrc
 - HISTSIZE =qtd desejada

`history -w` → gera arquivo com histórico (`bash_history`)

Para mudar o tamanho, em bytes, de da pasta `bash_history`

- É feito por usuário
 - `nano /home/usuario/.bahrc`
 - `HISTFILESIZE =qtd desejada`

Gerenciando Usuários no Linux

Criando e excluindo usuários

`useradd nomeUsuario` → criar usuário

`username nomeUsuario -m` → cria usuário com pasta home

`user cecilia -m -c "Cecília" -s /bin/bash`

`passwd nomeUsuario` → define senha

`chsh -s /bin/bash nomeUsuario` → define o shell que usará

`userdel nomeUsuario` → exclui usuário

`userdel -fr nomeUsuario` → exclui usuário e e-mail e diretórios

Editando informações do usuário

`useradd guest -c "Convidado" -m -e 17/11/2024` → limita o acesso até a data 17/11 (comando `-e`)

`usermod nomeUsuario` oque trocar (`usermod guest -s /bin/bash`) → troca o parâmetro `-s`

`passwd guest -e` → altera senha

cat /etc/passwd → verifica os usuários criados

Shell Script - Criando usuários em lote

useradd convidado1 -c "convidado especial" -s /bin/bash -m -p \$(openssl passwd -crypt senha123) → cria usuário com senha

Script

cd /

mkdir /scripts

nano criar_user.sh → abre nano para editar script

```
#!/bin/bash

echo "Criando usuários do sistema...."

useradd guest10 -c "Usuário convidado" -s /bin/bash -m -p $(openssl passwd -crypt Senha123)
passwd guest10 -e

useradd guest11 -c "Usuário convidado" -s /bin/bash -m -p $(openssl passwd -crypt Senha123)
passwd guest11 -e

useradd guest12 -c "Usuário convidado" -s /bin/bash -m -p $(openssl passwd -crypt Senha123)
passwd guest12 -e

useradd guest13 -c "Usuário convidado" -s /bin/bash -m -p $(openssl passwd -crypt Senha123)
passwd guest13 -e

echo "Finalizado!!"
```

chmod +x criar_user.sh

./criar_user.sh → executando **script**

Adicionando usuários a grupos

cat /etc/group → visualiza os grupos

usermod -G nomeGrupo1, nomeGrupo2 nomeUsuario → move de grupo

Criando novos grupos

groupadd nomeGrupo → adiciona grupo

groupdel nomeGrupo → exclui grupo

useradd rodrigo -c "Rodrigo Silva" -m -s /bin/bash -p \$(openssl -crypt Senha123) -G nomeGrupo → cria usuário e adiciona no grupo

gpasswd -d omeUsuario nomeGrupo → remove usuário do grupo específico

Conhecendo o sistema de permissões

```
-rw-r--r-- 1 root      root          0 jun 21 18:02 arquivo-root.txt
drwxr-xr-x 3 convidado1 convidado1 4096 jun 21 14:21 convidado1
```

```
drwxrwxr-x 2 matheusaganete matheusaganete 4096 nov 14 01:45 Planilhas
```

- → arquivo

d → diretório

R	W	X	R	W	X	R	W	X
DONO			GRUPO			OUTROS		

R - Read (leitura)

W - Write (gravação)

X - Execute (Execução)

Logado no usuário root → libera permissões ao usuário

chown nomeUsuario: nomeGrupo /nomeDiretorio/ → muda o dono do diretório e atribui grupo específico

chown nomeUsuario: nomeGrupo nomeArquivo → muda o dono do arquivo e atribui grupo específico e edita permissões

Alterando as permissões de um diretório - arquivo

Leitura (R)	4
Gravação (W)	2
Execução (X)	1
Nenhuma	0

chmod (soma dos números)[dono] (soma dos números)[grupo] (soma dos números)[outod] /nomeDiretorio

chmod 750 /adm

- permissão total ao dono
- leitura e execução ao grupo
- nenhuma para os demais usuários

Gerenciando Usuários no Linux

Gerenciamento de pacotes (UBUNTU-DEBIAN)

apt-get

apt

apt list — installed → verificar o que está instalado na máquina

apt list —upgradable → verifica o que precisa ser atualizado

apt search nomePrograma → verificar se há o programa

apt install nomePrograma → instalar programa

apt remove nomePrograma → desinstalar programa

apt remove nomePrograma -y → desinstalar programa sem perguntar

apt edit-sources → lista os repositórios oficiais do ubuntu

Atualização do sistema operacional

Crie um snapshots para não fazer os testes em ambiente de produção

apt update

apt upgrade -y

Realizando a instalação de arquivos DEB

No Ubuntu a extensão de execução é DEB → equivalente ao exe do Windows

apt install nomeArquivo.deb

Gerenciamento de Discos no Linux

Discos, sistemas de arquivos e partições

- Discos: Onde armazenamos os arquivos fisicamente. → HD, SSD
- Sistemas de Arquivos: Uma forma como o sistema operacional usa para controlar como os dados são armazenados e recuperados.
- Partições: Particionamento é a divisão de um disco em partes. Cada parte ou partição é independente uma da outra.

SISTEMA OPERACIONAL	SISTEMA DE ARQUIVOS SUPORTADOS
Linux	EXT3, EXT4, XFS, JFS
MacOS	HFS
Windows	FAT, HPFS, NTFS
FreeBSD, OpenBSD	UFS
Sun Solaris	UFS
IBM AIX	JFS

Adicionando um novo disco

lsblk ou fdisk -l → mostra o disco

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINTS
sda	8:0	0	20G	0	disk	
sda1	8:1	0	1M	0	part	
sda2	8:2	0	1,8G	0	part	/boot
sda3	8:3	0	18,2G	0	part	
ubuntu--vg-ubuntu--lv	252:0	0	10G	0	lvm	/
sr0	11:0	1	1024M	0	rom	

sda → possui um disco com 3 partições

Após criação do disco, não é possível utiliza-lo imediatamente. Pois, é necessário formatar e particionar.

Particionando e formatando discos via terminal

Particionando

fdisk /dev/nomeDisco

n: criar partição

p: primaria

1: n° partição

w → salvar alterações

Formatando

mkfs.nomeSistemadeArquivo /dev/nomeDisco: mkfs.ext4 /dev/sdb

Montando e desmontando discos

cd /mnt → Normalmente, é realizado dentro da pasta mnt

mkdir disco2 → Cria diretório dentro desta pasta

mount /dev/sdb /mnt/disco2/ → monta disco dentro desse disco

umount /dev/sdb → desmonta o disco

Após realizar esse processo, se reiniciar a máquina, o disco não é montado automaticamente

Montando discos automaticamente

nano /etc/fstab → arquivo que verifica discos/partições montados

/dev/sdb /disco2 ext4 defaults 0 0

Copiando Arquivos e Manipulando Processos

Copiando arquivos

cp origem/arquivo/destino → cp copia mesmo. Com o mesmo nome, ele sobrepõe

cp /home/matheusaganete/texto.txt /mnt/disco2 → copia um arquivo

cp /home/matheusaganete/*.txt /disco2 → copia todos os arquivos que contenham .txt

cp /home/matheusaganete/* /disco2 → copia tudo

Se já tivesse em /home/matheusaganete/

cp ./a* /disco2 → copia todos os arquivos que começam com a letra a

cp origem/arquivo/destino -i → -i pergunta se gostaria de sobrepor se tiver o mesmo nome

cp origem/arquivo/destino -r → -r copia arquivos e diretórios

cp origem/arquivo/destino -r -v → -v visualiza o que ele está fazendo

Renomeando Movendo arquivos

mv origem/arquivo/destino → mv move arquivos. Com o mesmo nome, ele sobrepõe

mv nomeAntigo nomeNovo → renomeia arquivos

mv texto.txt resto.txt → muda o nome de texto para resto

Iniciando, visualizando e encerrando um processo

ps -aux

Componentes do comando:

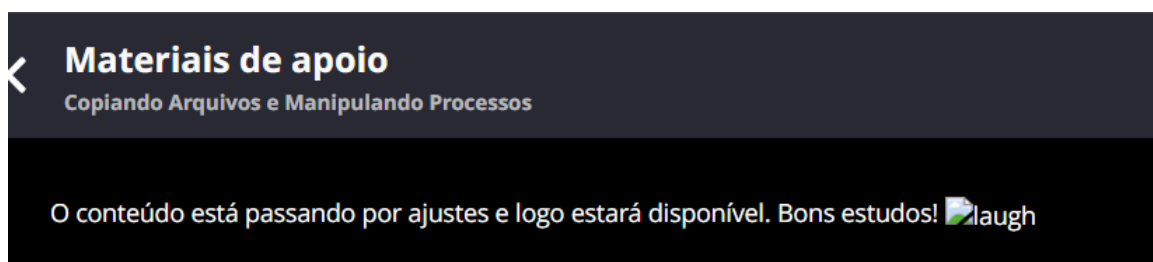
1. **ps** : Exibe informações sobre processos em execução no sistema.
2. **aux** : Combinação de opções que controla a saída:
 - **a** : Mostra processos de todos os usuários, não apenas os do usuário atual.
 - **u** : Exibe as informações no formato de usuário (mostrando detalhes como usuário, PID, %CPU, %MEM, etc.).
 - **x** : Inclui processos que não estão associados a um terminal (como serviços e processos em segundo plano).

kill nomeProcesso(id) → encerra o processo

killall nomePrograma/processo → encerra todos os processos

w → lista quem está logado

who



[DIO | Codifique o seu futuro global agora](#)

PROJETO → Infraestrutura como Código: Script de Criação de Estrutura de Usuários, Diretórios e Permissões

DESCRIÇÃO

Neste projeto iremos criar um script onde toda a infraestrutura de usuários, grupos de usuários, diretórios e permissões serão criadas automaticamente. Será realizado o upload do arquivo de script no GitHub para futuras reutilizações do script. Sendo assim, toda nova máquina virtual que for iniciada já estará pronta para uso quando o script for executado.

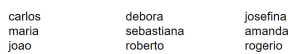
DIRETÓRIOS



GRUPOS



USUÁRIOS



Definições

- Excluir diretórios, arquivos, grupos e usuários criados anteriormente;
- Todo provisionamento deve ser feito em um arquivo do tipo Bash Script;
- O dono de todos os diretórios criados será o usuário root;
- Todos os usuários terão permissão total dentro do diretório **publico**;
- Os usuários de cada grupo terão permissão total dentro de seu respectivo diretório;
- Os usuários não poderão ter permissão de leitura, escrita e execução em diretórios de departamentos que eles não pertencem;
- Subir arquivo de script criado para a sua conta no GitHub.

Execução do Projeto

```
cd /script/
```

```
nano iac.sh
```

```
**Na pasta cd /script/  
nano iac.sh
```

```
#!/bin/bash
```

```
echo "Iniciando processo"
```

```
echo "Criando Diretórios"
```

```
mkdir publico adm ven sec
```

```
echo "Criando Grupos"
```

```
groupadd GRP_ADM GRP_VEN GRP_SEC
```

```
echo "Criando usuários e Adicionando no Grupo"
```

```
useradd carlos -c "Carlos" -s /bin/bash -m -p $(openssl passwd  
passwd carlos -e
```

```
useradd maria -c "Maria" -s /bin/bash -m -p $(openssl passwd  
passwd carlos -e
```

```
useradd joao -c "Joao" -s /bin/bash -m -p $(openssl passwd -c  
passwd carlos -e
```

```
useradd debora -c "Debora" -s /bin/bash -m -p $(openssl passwd  
passwd carlos -e
```

```
useradd sebastiana -c "Sebastiana" -s /bin/bash -m -p $(opens  
passwd carlos -e
```

```
useradd roberto -c "Roberto" -s /bin/bash -m -p $(openssl pas  
passwd carlos -e
```

```
useradd josefina -c "Josefina" -s /bin/bash -m -p $(openssl p  
passwd carlos -e
```

```
useradd amanda -c "Amanda" -s /bin/bash -m -p $(openssl passwd  
passwd carlos -e
```

```
useradd rogerio -c "Rogerio" -s /bin/bash -m -p $(openssl pas  
passwd carlos -e
```

```
echo "Permissões aos diretórios"
```

```
chown root:GRP_ADM /adm
```

```
chown root:GRP_VEN /ven
```

```
chown root:GRP_SEC /sec
```



```
chmod 770 /adm  
chmod 770 /ven  
chmod 770 /sec  
chmod 777 /pulico  
  
echo "Finalizado"
```

```
chmod +x iac.sh
```

```
./iac.sh
```