

Linux

`pwd`

`cd`

`./ > for local`

`dpkg install ./ > Instala arquivo .deb. O ./ pode ser usado para instalar direto no diretório sem precisar digitar tudo.`

PERMISSÕES

Basta usar `chmod`

Owner – Groups – Public

`sudo chown user:group`

`sudo chwon root:felipe file.txt`

`chmod 6(ler e escrever) / 4(leitura) – Pode usar os numeros para corresponder,`
por exemplo: `sudo chmod 646 file.txt`

`nano file.txt`

EXCLUIR E CRIAR DIRETÓRIO

Para deletar um arquivo único, basta digitar `> rm file.txt`

Para criar um diretório use `> mkdir nome`

CRIAR UMA ARQUIVO

```
sudo nano ./nome/file.txt
```

PERMISSÃO DE TODO O DIRETÓRIO

```
sudo chown -R felipe:felipe ./nomediretorio (argumento -R faz ser  
recursivo,  
trocando toda permissão do diretório.)
```

```
chmod 755 ou 777 (ruim) ou 666 (nunca fazer) 644 (ver esse)
```

CRIAR UM ARQUIVO e EXCLUIR UM ARQUIVO COM O rm E USANDO O /*

```
touch file1txt file2txt file3.cpp file4.cpp
```

* Use o touch para criar o arquivo do próprio terminal, sem precisar escrever

o nano e ter que abrir o arquivo e salvar.

Se utilizar rm ./*.cpp irá remover todos os arquivos que tiverem a extensão

.cpp daquele diretório.

rm diretorio/* apaga tudo o que tiver no diretório.

Para remover um diretório basta usar o rm -rf diretorio

COPIAR UM ARQUIVO

```
cp nomearquivo ./diretorio/nomedoarquivo2txt  
rm filename.txt
```

* pode usar `rm filename.*` ou `rm *.txt` ou `rm /*`

Para remover o diretório e tudo que está nele digite `rm -rf diretorio`
`mv filename filename2`

CREATING NEW DIRECTORIES AND MOVING FILES

`mkdir /usr/bin/diretorio` ---- `mkdir ../../novodiretorio` (sobe dois levels) ou pode somente usar o `./diretorio` para criar um diretório no mesmo local.

`mv arquivo.txt novodiretorio/arquivo.txt`

RENOMEAR

Para renomear, basta usar o comando `mv` dessa forma: `mv diretorio meusegundodiretorio`

COPIAR e RENOMEAR COM O `cp`

`cp novodiretorio/arquivo.txt segundodiretorio/arquivo`

`cp ./novodiretorio/nome1.out ./novodiretorio/nome2.txt`

Find commands and it's uses

COMANDO FIND e como encontrar arquivos.

Obs: O ponto (.) antes do `-type` é para indicar o local que está.

Obs: o argumento (f) para encontrar files e (d) para encontrar diretórios.

`find . -type f -name "*.php" > qualquer arquivo com a extensão php`

`find . -type f -iname "*.php" >` nesse caso ignora letras maiúsculas e minúsculas.

`find . -type f -iname "file*" >` retorna todos os arquivos com o nome file, independente do formato.

`find . -type f -perm 0664 >` encontra arquivos com a permissão 664

`find . -size +100 ou 1M ou +100k >` esse comando é para encontrar arquivos com esses pesos.

OPERADOR NOT

`find . -type f -iname "*.php"`

para encontrar o que não é php:

`find . -type f -not -iname "*.php" >` mostra tudo o que não é php.

`find . -maxdepth 1 -type f -iname "*.conf" >` Use o (-maxdepth) se não quiser que seja recursivo.

`find . -type f -iname "*.conf" -size +10k`

GREP command

GREP COMMAND

`grep "function" file.txt >` (grep) ("o que estamos procurando") (em qual arquivo você está procurando) (é possível adicionar mais de um arquivo.)

`grep -i ./ * >` grep (-i) que ignora letras maiúsculas e minúsculas (./ *) em todos os arquivos.

`grep -n -i "function" ./ * > grep (-n) informa a linha que a string foi encontrada.`

FIND with GREP

FIND with GREP

```
find . -type f -iname "*.php" - exec grep -i -n "function" {} +
```

```
find . -type f -size -10k -iname "*.php" - exec grep -i -n  
"function" {} +
```

Redirecting the output of a command

REDIRECTING THE OUTPUT OF A COMMAND

```
ls > file.txt
```

```
find . -type f -size -10k -iname "*.php" -exec grep -i -n  
"sandwich" {} + > file.txt
```

ATENÇÃO AO GREP:

```
find . -type f -size -10k -iname "*.php" - exec grep -i -n  
"function" {} + | tee of.txt
```

- O comando (`| tee of.txt`) mostra na tela o resultado, mas também cria um arquivo `of.txt` com o que for exibido na tela.

The TOP command and its uses

TOP COMMAND AND ITS USES

`top` > mostra os processos em execução. Praticamente um gerenciador de tarefas do windows.

- PID é o ID do processo
- USER quem está rodando o processo
- TIME+ quanto tempo a aplicação está aberta
- COMMAND são os comandos associados ao processo.

`ps box` > tira um “print” dos processos, não é em tempo real.

`ps aux | grep chrome-browser` > Isso vai procurar por qualquer processo que tenha chrome-browser nele.

`pgrep chrome-browser` > vai mostrar os processos que estão rodando.

- Se tiver mais de um processo igual rodando, esse comando mostra os IDs diferentes.

COMMANDO KILL

`kill -p 6300` > (6300) é o ID do processo.

`killall chrome-browser` > Fecha todos os processos abertos que sejam do chrome-browser.

Services Explained

SERVICES EXPLAINED

`apt-cache policy elasticsearch`

```
sudo service elasticsearch start
```

Service é como um processo que fica rodando no background que pode ser iniciado ou pode ser interrompido.

```
find /etc -type f -iname "elasticsearch*" > Procura o arquivo de configuração do elasticsearch
```

Encontrando o arquivo, basta abri-lo com o nano:

```
sudo nano /etc/elasticsearch/elasticsearch.yml
```

É necessário reiniciar o server

```
sudo service elasticsearch reload
```

```
sudo service elasticsearch restart
```

```
sudo service elasticsearch stop
```

```
sudo systemctl start elasticsearch
```

```
sudo systemctl stop elasticseatch > a diferença é que o comando start ou stop vem antes do serviço.
```

CRONTABS to schedule tasks

CRONTABS to schedule tasks

```
contrab -e > vai abrir o arquivo contrab e que é salvo no diretório etc
```

```
# m h dom mon dow  command  (minutes) (hours)      (day of month)
(month number) (day of week)  (command)
```

```
15 14 * * * ls > /home/nick/lt/cronres.txt
```

0 exemplo para criar um backup

```
0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
```

para criar um contrab que execute um comando com sudo

```
sudo contrab -e
```

```
0 7 * * 1 apt-get upgrade -y
```

Adding/removing users through GUI

ADDING/REMOVING USERS THROUGH CLI

```
sudo adduser felipe2
```

É possível colocar outros dados para a conta, se não quiser é só pular com o enter.

para trocar o usuário basta digitar:

```
su user2
```

ADICIONAR USUARIO AO ARQUIVO SUDO

```
sudo adduser felipe2 sudo (adiciona felipe2 ao grupo sudo) (é possível mudar o sudo pelo grupo que quiser fazer parte)
```


Adding and removing user through CLI

REMOVING USER THROUGH CLI

```
sudo deluser user2
```

Changing users password

CHANGING USERS PASSWORD

```
sudo passwd user2
```

- enter new password
- retype password
 - ◇ password updated

Adding new groups

ADDING NEW GROUPS

```
sudo groupadd nomedogrupo2
```

```
chown user:nomedogrupo2 arquivo
```

```
sudo adduser user2 tt
```

network commands

NETWORK COMMANDS

ping www.host.com

ifconfig (linux)

ipconfig (windows)

tcpdump > analisar pacotes

- tcpdump -c 10 > captura somente 10 pacotes
- tcpdump -c 5 -A > mostra o pacote que esta recebendo

tcpdump -c 5 -i wlo1 > escuta em uma única interface

tcpdump -XX -i -wlo1 > mostra os pacotes em hexadecimal

tcpdump -i wlo1 port 22 > escuta a porta determinada

netstat -nr (kernel ip routing table)

netstat -i (mostra o uso de cada device)

netstat -ta (procura por portas abertas)

netstat -tan (mostra os ips ao inves dos hosts)

Linux host file

LINUX HOST FILE

esta em /etc/hosts

DNS procura por qual IP e o host

Se o computador não encontrar no arquivo, ele vai para o roteador e depois para o ISP

IP Address	domain/host name	atalho
------------	------------------	--------

HOSTNAME

```
sudo hostnamectl set-hostname User
```

```
sudo nano /etc/hosts
```

TRACEROUTE command

TRACEROUTE

```
traceroute www.google.com
```

- Mostra os pulos até chegar no servidor do google

Network mapping explained

NETWORK MAPPING

`nmap >` mostra quais dispositivos estão na rede e quais serviços a máquina está oferecendo.

`sudo apt-get install nmap`

`nmap -v ip >` mostra mais informações do que normalmente conseguimos ver

`nmap 192.168.0.100,1 >` escaneia múltiplos ips

`nmap 192.168.0.1-100 >` escaneia todos os IPs nesse range

é possível utilizar o `*` no final para escanear todos os ips possíveis

é possível colocar em um arquivo de texto os ips e hosts que deseja escanear, para isso use o comando:

`nmap -iL ~/network.txt`

coloque o arquivo dessa forma:

192.168.0.1

192.168.0.100

www.host.com

`nmap -A 192.168.0.1 >` habilita o scan para SO

Secure Shell

SECURE SHELL

`ssh username@remotehost.com`

password e conecta

SFTP

SFTP

Usa porta 21

Transferência de arquivos

```
sftp user@user.net
```

```
put > envia
```

```
get > faz o download
```

```
exit > sai do sftp
```

Criando um servidor SSH

CRIANDO UM SERVIDOR SSH

primeiro é necessário instalar

```
sudo apt-get install openssh-server
```

```
sudo nano /etc/ssh/sshd_config
```

- Alterar a porta de 22 para 2212 (mais segurança)
- Login root como no
- AllowUsers felipeuser (permite os usuarios a logar no servidor)

```
sudo systemctl restart ssh
```

É necessário fazer portforwarding no roteador.

MAN command explained

MAN COMMAND EXPLAINED

`man > manual`

- `man ssh`
 - ◊ Ira mostrar uma página manual sobre ssh