

Linux I

Sumário

Conceitos OpenSource	4
Tipos de licenciamento	4
Conceitos GNU/Linux	5
Distribuições Livres	5
Distribuições Corporativas	5
Modo CLI	6
Modo gráfico	7
GNOME	7
KDE	7
FHS	8
Estrutura de diretórios GNU/Linux	8
bin	8
boot	9
dev	9
etc	9
lib	9
media	9
mnt	9
opt	9
sbin	9
srv	10
tmp	10
usr	10
var	10
proc	10
sys	10
Comandos Básicos	11
pwd	11
Ls	11

Cd	11
Touch	11
Rm	11
Mkdir	12
Cp	12
Mv	12
Echo	12
Cat	13
Less	13
More	13
Laboratório	13
Comandos Intermediários	14
Grep	14
Find	14
Sed	14
Sort	15
Cut	16
Man	17
Rsync	17
Tar	17
Instalação de pacotes	18
Laboratório	19
Bibliografia	20

Conceitos OpenSource

Quando pensamos no termo OpenSource certamente pensamos em um dos sistemas operacionais mais utilizados entre os administradores de sistemas, GNU/Linux, que nos isenta o custo de software com licenças, e que nos faz entender de como a internet se expandiu de forma tão exponencial utilizando esses softwares. Com o uso dessa ideologia os usuários compreendem melhor como surgiu tantas ferramentas. Antes de falar de OpenSource deve tomar um certo cuidado, pois softwares livres não são a mesma coisa que softwares OpenSource, para que seja um software OpenSource deve se seguir 4 pilares fundamentais, sendo eles:

- Liberdade de executar o programa, para qualquer propósito;
- Liberdade de estudar como o programa funciona e adaptá-lo às suas necessidades sendo o acesso ao código-fonte um pré-requisito;
- Liberdade de distribuir cópias ;
- liberdade de melhorar o programa e liberar os seus aperfeiçoamentos, de modo que toda a comunidade se beneficie.

Liberdade de execução para qualquer propósito - Quando falamos em executar um software para qualquer propósito, nos referimos a não ter restrições com licenças ou valores monetários quanto ao uso do software.

Liberdade de estudo e adaptação conforme necessidades - Liberdade de estudo e adaptação é não ter restrições quanto ao estudo de funcionamento/Código e adaptação quanto a necessidade. Exemplo seria modificar o formato de exportação de relatórios de determinado software.

Liberdade de distribuição - Liberdade de distribuição se considera a disponibilidade de cópias modificadas ou não de determinado software uma vez que tenha posse do mesmo.

Liberdade de contribuição ao software e disponibilidade a comunidade - De forma obrigatória, um software OpenSource deve ser disponibilizado a comunidade, para que seja aperfeiçoado e melhorado a cada release do software.

Tipos de licenciamento

Com a criação desse tipo de licenças várias outras foram criadas com o objetivo de defender a liberdade ao conhecimento, abaixo segue alguns exemplos.

- **GFDL** GNU Free Documentation Licence (www.gnu.org/copyleft/fdl.html);
- **OPL** Open Publication Licence (<http://www.opencontent.org/openpub/>);
- **CC** Creative Commons (<http://creativecommons.org/about/licences>);
- **BSD** Berkeley Software Distribution (<http://www.freebsd.org/copyright/licences.html>);
- **SPL** Sun Public Licence (<http://java.sun.com/spl.html>);

Dados relevantes para o mundo OpenSource:

- A Red Hat, empresa que investiu em software livre e gerava lucro vendendo suporte e treinamento para o Red Hat Enterprise Linux, foi comprada pela IBM por \$34 bilhões.
- 67% dos servidores web da internet são software livre.
- 74% dos celulares no mundo utilizam Android, que é um subproduto do projeto AOSP (Android Open Source).
- 100% dos 500 computadores mais rápidos do mundo (Super Computadores) rodam Linux.
- Com a vantagem do código estar acessível, o desenvolvimento se torna muito mais rápido, seguro e com colaboradores ao redor do globo, por exemplo a Microsoft que abriu o código do Visual Studio Code, já possui mais de 800 contribuidores.
- O código se torna mais seguro pois somente é adicionado após uma validação dos mantenedores do projeto.

Conceitos GNU/Linux

O nome GNU/Linux pode ser estranho para algumas pessoas, pois popularmente ele é conhecido somente como Linux, o que é uma falta de consideração com quem vem desenvolvendo o sistema.

GNU é um grupo que foi fundado em 1984 por Richard Stallman, com o objetivo de criar um sistema operacional “Unix-Like”, a partir desse sistema vários outros sistemas foram criados e mantidos pela comunidade.

Em 1991, Linus Torvalds, disponibilizou a primeira versão do sistema “Unix-like” sendo possível a união do kernel do sistema com os softwares GNU dando origem ao que conhecemos como GNU/Linux.

Em meio a todo esse mundo GNU linux em algum momento você deve ter ouvido falar em Debian, Ubuntu, Red Hat entre outras, mas porque tantas distribuições? porque você não se identifica com nenhuma delas e é livre para fazer a sua própria distribuição. Em 1993 Patrick Volkerding, juntou o kernel do Linux com vários outros aplicativos, assim dando origem a distribuição chamada Slackware. foi a primeira versão distribuída em CD, a partir daí várias outras versões foram surgindo desde então.

A partir centenas de distribuição foram surgindo, algumas delas mais famosas do que outras, A maioria das distribuições são mantidas por grandes comunidades de colaboradores, entretanto algumas delas são mantidas por empresas, e são divididas em duas categorias.

Distribuições Livres

São mantidas pela própria comunidade de colaboradores sem fins lucrativos,

tendo em mente a contribuição contínua da distribuição;

Distribuições Corporativas

São distribuições que vendem suporte a essas distribuições, distribuições tais como Red Hat, Suse e Mandriva. Vale ressaltar que o conteúdo vendido por essas empresas não é a distro e sim suporte, treinamentos, garantias, ou seja, o conhecimento do sistema. O fato de não ser mais o software em si é devido a licença GPL Significa que por mais que uma empresa queira fazer um sistema operacional enquanto ele estiver registrado na categoria GPL, obrigatoriamente seu código fonte precisa ser disponibilizado gratuitamente.

Modo CLI

No mundo GNU/Linux utilizamos muito o Shell, que funciona como um interpretador de comandos inseridos por um humano e transformado em linguagem de máquina. O shell é a interface entre o usuário e o kernel do sistema por padrão o GNU/Linux utiliza o bash como shell, existem outros tipos de shell como por exemplo csh, tcsh, ksh e zsh. Quando usamos o shell temos duas categorias de usuários, sendo eles **usuários comuns** e **super usuarios (root)**.

- **Usuário comum** - O usuário comum é qualquer usuário que não tenha permissão de **root**. Geralmente identificados por \$ na linha de comando do console. Todo usuário que tente executar um comando de administrador não é permitido a não ser que ele seja autorizado a isso.
- **Superusuário** - Os superusuários ou root, são users que podem executar comandos que usuários comuns não podem, exemplo, **shutdown**, que desligaria o sistema adicionando alguns parâmetros. Sempre são identificados por um # no final da linha de comando do console.

O modo CLI ou Command Line Interface é o modo do GNU/Linux de texto puro e sem interação com interface ou representações gráficas e pode ser acessado de duas formas.

No modo TTY é o modo de CLI que pode ser acesso pela interface do GNU/Linux, Alguns exemplos de comandos que podem ser executados no terminal. O modo TTY requer autenticação no momento da inicialização, assim como na inicialização do sistema em modo gráfico. para acessar o modo TTY basta usar o seguinte comando e inserir as credenciais de login, geralmente sendo as mesmas do modo gráfico.

Caso necessário segue o link de instalação do [CentOS](#)

CTRL+ALT+F4

Além disso eu seguiria...depois do login, qual seria um próximo passo de interação? Comandos de interação? Linkar com o que vira na sequencia.



Figura 1: Tela de login Linux em modo CLI.

Modo Gráfico

O modo gráfico ou GUI (Graphic Interface User) é o modo convencional do Linux geralmente utilizado por usuários mais comuns, que não tem tanta familiaridade com o texto ou até mesmo ferramentas ou serviços mais complexos. Cada distribuição tem o seu visual diferente por conta da sua interface padrão, porém pode ser modificada instalando outras interfaces. as mais comuns são GNOME, KDE entre outros.

A interface é a critério de do usuário, por padrão distribuições voltadas a servidor como CentOS e Red Hat acompanham a Interface GNOME como interface, porém podem ser instaladas outras interfaces conforme documentação da própria interface.

GNOME

É a interface gráfica mais comum, pois não se limita somente ao visual junto a ela traz outros recursos que compõe a tela.



Figura 2: Interface GNOME.

KDE

Conhecido e muito utilizado assim como GNOME, traz uma série de aplicativos de sistema que facilita o dia a dia.

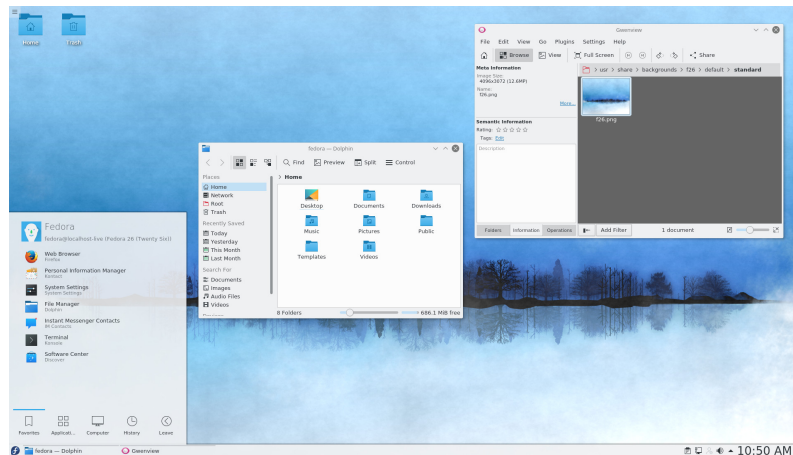


Figura 3: Interface KDE.

FHS

FHS ou Filesystem Hierarchy Standard, é o padrão que define como os arquivos são armazenados em cada diretório. De certa forma isso é importante pois define compatibilidade entre os sistemas existentes no mercado permitindo que qualquer software escrito para GNU/Linux se executado em qualquer distribuição sem maiores problemas. Atualmente sua versão é a 2.3 e é mantida sem fins lucrativos por grandes empresas como IBM, Red Hat e Dell.

Estrutura de diretórios GNU/Linux

Um diretório nada mais é do que um recipiente de arquivos, que podem ser de texto, música, vídeo etc. e cada uma delas devem ser nomeados para serem utilizados quando necessário. Uma coisa importante a se considerar é que o GNU/Linux é *case sensitive*, significa que ele diferencia letras maiúsculas de minúsculas, logo um arquivo chamado **TESTE** é diferente do arquivo **teste**.

A Árvore de diretórios do segue o seguinte modelo:

```
/
bin cdrom etc lib mnt proc root var sys boot dev home media opt
sbin srv tmp usr
```

De todos esses diretórios temos um principal que chamamos de raiz,

representado por "/" assim como a imagem acima, a barra é o diretório é a raiz e os demais diretórios são subdiretórios que vamos entender melhor o que significa cada um deles.

BIN

Esse diretório é essencial para o funcionamento do sistema, dentro dele contém comandos essenciais, esses comandos podem ser usado por usuários comuns e superusuários pois é um diretório público, temos como exemplo comandos como:.

- ls
- cp
- mv
- rm

BOOT

Esse diretório contém arquivos estáticos essenciais para a inicialização do sistema.

DEV

Dentro desse diretório são armazenados os arquivos de dispositivos, e fazem comunicação com todos os periféricos do sistema e facilitando o acesso aos recursos.

ETC

Dentro desse diretório são armazenados todos os arquivos de configuração do sistema.

LIB

Dentro desse diretório fica os arquivos de bibliotecas e módulos do kernel que podem ser usados pelos outros programas.

MEDIA

Dentro desse diretório são armazenados os dispositivos de removíveis.

- **Pendrive**
- **Smartphone**
- **Impressora**
- **CD/DVD**

MNT

Dentro desse diretório são armazenados os arquivos de montagem temporária de sistemas de arquivos de WIndows para Linux, Linux para Linux etc.

OPT

Geralmente armazenam softwares que não fazem parte oficialmente da distribuição.

SBIN

O diretório sbin armazena comandos que somente superusuários podem usar,

dentre ele estão:

- **ifconfig**
- **iptables**
- **apt**

SRV

O diretório `srv` armazena dados que geralmente são fornecidos por aplicações que normalmente são de alcance geral, ou seja, não são dados de usuários.

- `/srv/www/` - servidor web
- `/srv/ftp` - servidor ftp

TMP

Diretório direcionado a arquivos temporários, usado para guardar informação até que uma ação seja completada.

USR

Geralmente guardam programas que não são importantes para o sistema, exemplo navegadores, editores de texto etc.

VAR

são armazenados arquivos de aplicações, normalmente programas que geram log e outras informações são armazenados nesse diretório.

PROC

Nesse diretório são armazenadas dados estáticos do sistema, configurações de sistema entre outros.

SYS

Os diretórios `/root` e `/home` podem estar disponíveis no sistema, mas não precisam obrigatoriamente possuir este nome. Por exemplo, o diretório `/home` poderia se chamar `/casa`, que não causaria nenhum impacto na estrutura do sistema.

Diferente do sistema operacional Windows, onde sua estrutura de diretório é formada por arquivos de sistema, pastas de programas e arquivos de usuários sua estrutura de diretórios é bem maior comparada ao GNU/Linux, por esse motivo é importante conhecer bem a estrutura FHS o funcionamento correto de programas e recursos de recursos de sistema

Comandos Básicos

Quando falamos em comandos básicos nos referimos em comandos de navegação, sendo a base de todo usuário GNU/Linux para manipulação do modo texto, abaixo veremos os comandos mais utilizados no dia a dia de um usuário CLI e como esses comando facilitam as tarefas.

PWD

Em meio a tantos diretórios as vezes podemos nos perder, por isso temos um comando muito útil que é o pwd, ele mostra em qual diretório estamos.

```
pwd
```

```
user@user:~$ pwd  
/home/user
```

LS

Para listar os arquivos de um diretório usamos o comando ls, junto com outros parâmetros pode trazer informações mais detalhadas.

```
ls
```

```
user@user:~$ ls  
Desktop Documents Downloads file.tar.gz file.txt Music Pictures Public Templates Videos VirtualBox VMs
```

CD

Utilizado para entrar em diretórios diversos sendo eles de sistema ou de perfil de usuário.

```
cd
```

```
user@user:~$ cd Pictures/
```

Podemos também passar o caminho absoluto invés de digitar vários cd para acessar determinado arquivo por pasta exemplo.

```
cd /etc/sysconfig/network
```

```
user@user:~$ cd /etc/sysconfig/network/
```

Por vários motivos precisamos voltar diretórios, invés de digitar o caminho absoluto podemos simplesmente inserir `cd ..` e voltar um diretório acima..

```
cd ..
```

```
user@user:/etc/security$ cd ..  
user@user:/etc$ █
```

No exemplo saímos do diretório security e estamos dentro do diretório etc.

TOUCH

Utilizado para criar arquivos dentro dos diretórios, sua utilização é bem simples e fácil.

```
touch <nomedoarquivo>
```

```
user@user:~$ touch arquivo  
user@user:~$ ls  
arquivo Desktop Documents Downloads file.tar.gz file.txt
```

RM

Remover arquivos no GNU/Linux também é bem simples e fácil.

```
rm <nomedoarquivo>
```

```
user@user:~$ rm arquivo  
user@user:~$ ls  
Desktop Documents Downloads file.tar.gz file.txt Music
```

MKDIR

Para criar um diretório, deve informar o nome do diretório antes do comando, segue exemplo abaixo

```
mkdir <nomedodiretório>
```

```
user@user:~$ mkdir files  
user@user:~$ ls  
Desktop Documents Downloads files file.tar.gz file.txt
```

CP

Para copiar arquivos para dentro do diretório que criamos, segue o exemplo,

```
cp arquivo /diretorio
```

```
user@user:~$ cp file.tar.gz file.txt files/  
user@user:~$ ls files/  
file.tar.gz file.txt
```

podemos listar arquivos com o `ls` mesmo não estando no diretório, como fizemos no exemplo acima, passando o caminho absoluto do arquivo.

MV

Mover arquivos para os diretórios é bem semelhante ao processo de cópia, segue o exemplo

```
mv arquivos /diretorios
```

```
user@user:~$ mv file.tar.gz file.txt /tmp/  
user@user:~$ ls /tmp  
file.tar.gz  
file.txt
```

Novamente informamos o caminho absoluto, porém da pasta /tmp.

CAT

Podemos ver conteúdo de arquivos dentro no GNU/Linux usando o comando cat.

```
cat file.txt
```

```
user@user:~$ cat file.txt  
"Hello darkness my old friend"  
"Olá Mundo"
```

Figura 4: Verificando conteúdo do arquivo.

ECHO

Com o comando echo podemos simplesmente exibir mensagens na tela do terminal ou inserir informações dentro de arquivos sem usar editores de texto, segue o exemplo abaixo.

```
echo "Hello darkness my old friend"
```

Quando utilizamos o echo e cat podemos manipular o stdout para outros arquivos ou lugares utilizando os sinal de maior.

```
echo "Hello darkness my old friend" > file.txt
```

Nesse exemplo inserimos a stdout do echo no arquivo file.txt, o símbolo de > faz substituição de todo o conteúdo do arquivo por outro, no caso o stdout do echo.

```
echo "Olá Mundo" >> file.txt
```

Nesse exemplo inserimos a stdout do echo porém sem alterar o conteúdo já existente, o símbolo de >> incrementa o stdout para o final do arquivo sem alterar o conteúdo já existente. Podemos verificar usando o cat no arquivo em questão.

```
user@user:~$ cat file.txt
"Hello darkness my old friend"
"Olá Mundo"
```

LESS

Quando o conteúdo de um arquivo é muito grande a visualização do comando cat pode não ser das melhores. por isso temos o comando less que nos permite navegar entre pelo arquivo usando as setas direcionais do teclado e para sair basta usar a tela q.

```
less file.txt
```

```
user@user:~$ sudo less /var/log/syslog
```

O parâmetro sudo é usado para ter privilégios de superusuário, pois a pasta var é de acesso restrito a usuários comuns.

MORE

Como o próprio nome já diz o comando more nos permite ver o arquivo porém não podemos ver linhas que já navegamos, ou seja, só podemos pressionar enter e as próximas linhas disponíveis, para sair da visualização do comando more também utilizamos a tecla q.

```
more file.txt
```

```
user@user:~$ sudo more /var/log/syslog
```

Laboratório

- verifique em qual diretório você está.
- Crie uma pasta lab e seu nome
- entre na sua pasta e crie um arquivo com o nome.
- insira 3 nomes aleatórios dentro do arquivo.
- veja o conteúdo do arquivo para verificar.
- copie o arquivo para a pasta /tmp
- veja os arquivos da pasta tmp para verificar
- remova o primeiro arquivo que foi criado na pasta /tmp

Comandos Intermediários

GREP

Quando o conteúdo de um arquivo é muita informação junta podemos fazer um filtro utilizando o comando `grep`.

```
cat file.txt | grep darkness
```

```
user@user:~$ cat file.txt | grep darkness
"Hello darkness my old friend"
```

Figura 5: Filtrando stdout do arquivo.

retornou somente uma linha com a palavra `darkness`

FIND

Quando precisamos procurar um arquivo entre muitos o comando `find` pode encurtar e economizar tempo.

```
find . -name file.txt
```

```
user@user:~$ find . -name file.txt
./file.txt
```

Figura 6: Comando Find.

Obs: Na sintaxe do comando `find`, logo após deve se especificar o local onde vai ser feita a procura do arquivo, no exemplo o "." significa o diretório atual, o parâmetro `-name` é utilizado para procurar pelo nome do arquivo.

SED

quando temos um arquivo que precisamos várias alterações iguais no mesmo arquivo, podemos utilizar o comando `sed`, muito utilizado em substituição de arquivos.

```
cat file.txt | sed s/darkness/escuridao/
```

No exemplo acima estamos substituindo a palavra `darkness` por `escuridao`.

```
user@user:~$ cat file.txt | sed s/darkness/escuridao/
"Hello escuridao my old friend"
"Olá Mundo"
```

Figura 7: Comando sed.

SORT

Se quisermos ordenar o conteúdo de um arquivo, podemos usar o comando `sort`, vamos ordenar em arquivo em ordem alfabética.


```
sort file.txt
```

```
user@user:~$ sort file.txt
4
7
C
C
D
D
D
D
F
H
H
```

Figura 8: Comando Sort em ordem crescente.

Podemos classificar em ordem decrescente

```
sort -r file.txt
```

```
user@user:~$ sort -r file.txt
W
T
R
H
H
F
D
D
D
D
C
```

Figura 9: Comando Sort em ordem decrescente.

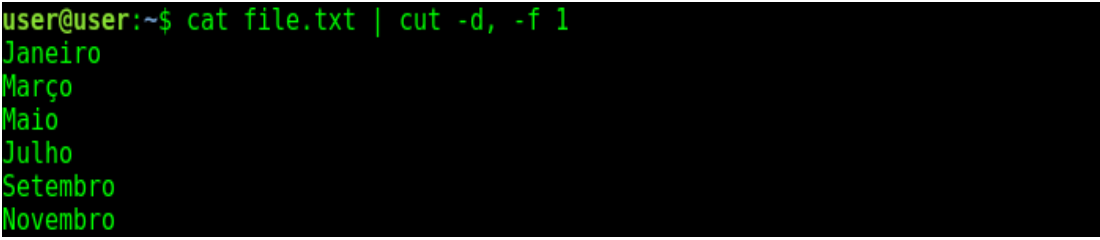
CUT

O comando cut é muito recomendado filtrar dados com pontuação sendo eles, :
;, entre outros abaixo temos um exemplo

```
user@user:~$ cat file.txt
Janeiro, Fevereiro
Março, Abril
Maio, junho
Julho, Agosto
Setembro, Outubro
Novembro, Dezembro
```

Figura 10:

```
cat file.txt | cut -d, -f 1
```

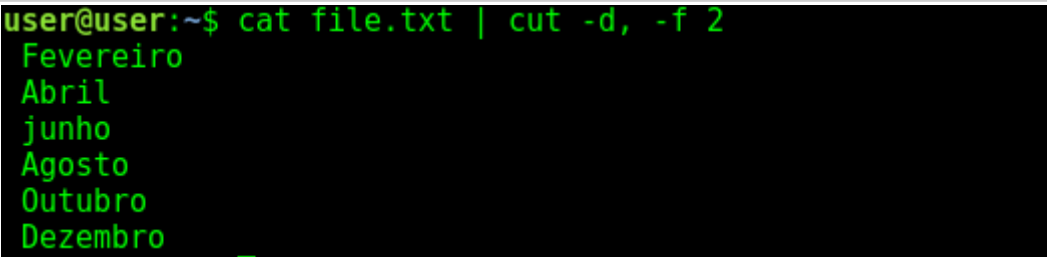


```
user@user:~$ cat file.txt | cut -d, -f 1
Janeiro
Março
Maio
Julho
Setembro
Novembro
```

Figura 11: Comando Cut retornando dados 1º coluna

No exemplo acima, retornamos somente o conteúdo da primeira coluna.

```
cat file.txt | cut -d, -f 2
```



```
user@user:~$ cat file.txt | cut -d, -f 2
Fevereiro
Abril
junho
Agosto
Outubro
Dezembro
```

Figura 12: Comando Cut retornando dados 2º coluna

No exemplo acima, retornamos somente o conteúdo da segunda coluna.

MAN

O comando man é utilizado para consultar a documentação do comando, retornando a todos os parâmetros possíveis. segue exemplo do comando cut.

```
CUT(1) User Commands
NAME
  cut - remove sections from each line of files

SYNOPSIS
  cut OPTION... [FILE]...

DESCRIPTION
  Print selected parts of lines from each FILE to standard output.
  With no FILE, or when FILE is -, read standard input.
  Mandatory arguments to long options are mandatory for short options too.

  -b, --bytes=LIST
        select only these bytes
  -c, --characters=LIST
        select only these characters
  -d, --delimiter=DELIM
        use DELIM instead of TAB for field delimiter
  -f, --fields=LIST
```

Figura 13: Manual do comando Cut

RSYNC

O comando rsync faz a sincronização de dados de um host para o outro ou local, no exemplo vamos sincronizar o arquivo file.txt para um host remoto. Por padrão o rsync não vem instalado com o sistema, abaixo o comando para instalar

```
sudo yum install rsync
```

junto com o comando rsync, devemos informar alguns parâmetros, quando sincronizamos para outro host deve se inserir na sintaxe -avz, como no exemplo abaixo

```
rsync -avz file.txt user@10.162.103.74:/home/user/
```

sendo -a para o arquivo, v para ser o stdout e z para comprimir o arquivo.

TAR

Em linux manipulamos muitos arquivos e em alguns casos precisamos compactar ou descompactar esses arquivos. Para isso usamos o comando tar.

```
tar -czf file.tar.gz file.txt
```

para compactar arquivos são necessários alguns argumentos, o -c é usado para criar o arquivo, z é usado para compactar o arquivo e o f para informar o arquivo que será criado, no nosso caso é o file.tar.gz e logo após informe os arquivos que serão compactados.

```
user@user:~$ tar -czf file.tar.gz file.txt
user@user:~$ ls
Desktop  Documents  Downloads  file.tar.gz  file.txt  Music  Pictures  Public  Templates  Videos
```

Figura 14: Comando tar compactando arquivos

para descompactar o arquivo tar basta trocar o parâmetro c por x e não informe

os arquivos pois estamos descompactando, segue o exemplo.

```
tar -xzf file.tar.gz
```

Instalação de pacotes

A necessidade de instalar pacotes em sistemas GNU/Linux surgiu pois novos programas são desenvolvidos e necessariamente eles não precisam vir com o sistema, essa é uma das vantagens do GNU/Linux por ser leve por conta de não ter embarcado no sistema tantos programas/pacotes como o Windows tem.

Quando falamos em instalação de pacotes temos alguns caminhos por onde podemos fazer as instalações e remoção dos mesmos, uma das formas mais comuns é a instalação por gerenciador de pacotes, O gerenciador de pacotes do Red Hat/CentOS é o yum e podemos atualizar o repositório, instalar programas, remover etc.

```
yum check-update
```

Check update é utilizado para atualizar os repositório so S.O

```
yum install package_name
```

O install juntamente com o nome do pacote é utilizado para instalar determinado pacote.

```
yum remove package_name
```

O remove juntamente com o nome do pacote é utilizado para remover.

```
yum search keyword
```

Quando não sabemos exatamente o nome do pacote podemos utilizar search como parâmetros de busca.

```
yum localinstall caminho absoluto para o nome de arquivo
```

Podemos baixar o pacote de outra fonte e informar seu caminho e fazer a instalação do mesmo.

Exercício prático

Vamos imaginar o seguinte cenário onde precisamos localizar determinado arquivo pelo seu nome e mover para outra pasta sem sair do diretório atual. O arquivo em questão será o syslog onde ficam armazenados todos os eventos que acontecem no sistema.

Primeiro vamos mudar para o usuário root pois o arquivos syslog só está disponível a super usuários, para isso vamos usar o comando su, informe a senha root do sistema.

```
su
```

```
user@user:~$ su
```

```
Password:
```

```
root@user:/home/user#
```

Sabemos que o arquivo syslog está em algum lugar do diretório raiz, por isso iremos utilizar o comando find para localizar o arquivo segue o exemplo abaixo

```
root@user:/home/user# find / -name syslog
find: '/run/user/1000/gvfs': Permission denied
/run/systemd/journal/syslog
/var/log/installer/syslog
/var/log/syslog
```

Na imagem acima mostra que o arquivo se encontra em /var/log, vamos copiar esse arquivo para nosso diretório atual.

```
cp /var/log/syslog .
```

```
root@user:/home/user# ls
```

```
Desktop Documents Downloads files Music Pictures Public syslog Templates Videos VirtualBox VMs
```

No caminho de destino do comando cp colocamos o ponto pois representa nosso diretório atual listamos os arquivos podemos ver o arquivo syslog no diretório, agora iremos mover esse arquivo para a pasta tmp, copiar o arquivo diretamente para a pasta tmp substituindo o ponto por /tmp.

```
mv syslog /tmp
```

```
root@user:/home/user# mv syslog /tmp
```

```
root@user:/home/user# ls /tmp
```

```
pulse-PKdhtXMmr18n systemd-private-749a0ddfe95842659dbb6b8b5a15446a-color.service-FcFTuU Temp-0a7f9428-86a5-4030-8bd9-2a57f9d1e336
ssh-mGWk9wLzEPp7 systemd-private-749a0ddfe95842659dbb6b8b5a15446a-rtkit-daemon.service-Pr7Geg tracker-extract-files.1000
syslog systemd-private-749a0ddfe95842659dbb6b8b5a15446a-systemd-timesyncd.service-QmXGiC
```

Laboratório

- Crie um arquivo com 5 palavras.
- Filtre por uma das palavras e mande o stdout para o arquivo grep dentro da pasta /tmp.
- pesquise pelo nome do arquivo dentro da pasta /tmp.
- altere uma das palavras do arquivo por batata.
- ordene os nomes do arquivo em ordem crescente e mande seu stdout para o arquivo crescente dentro da pasta /tmp
- ordene os nomes do arquivo em ordem decrescente e mande seu stdout para o arquivo decrescente dentro da pasta /tmp
- compacte os arquivos da pasta /tmp com o nome arquivos.tar.gz.
- consulte o manual do comando tar e envie o stdout para a pasta /tmp/tar

Bibliografia

tecnoblog, Compra da IBM pela Red Hat, <https://tecnoblog.net/265135/ibm-compra-red-hat/>, Acessado em: 27 de março de 2019

Wikipédia, comparativo de sistemas: https://en.wikipedia.org/wiki/Usage_share_of_operating_systems#Market_share_by_category Acessado em: 27 de março de 2019.

mxcursos, open source um conceito copyleft disponível em <https://blog.mxcursos.com/open-source-um-conceito-copyleft/> Acessado em: 11 de março de 2019.

pplware, alguns tipos de licenças de software. disponível em <https://pplware.sapo.pt/informacao/alguns-tipos-de-licencas-de-software/> Acessado em: 12 de março de 2019.

Diolinux, os 7 ambientes gráficos mais populares do linux, disponível em <https://www.diolinux.com.br/2016/09/os-7-ambientes-graficos-mais-populares-do-linux.html> Acessado em: 12 de março de 2019.

RedHat, Atualização de Pacotes usando a Linha de Comando com yum, disponível em : https://access.redhat.com/documentation/pt-br/red_hat_network_satellite/5.4/html/reference_guide/pup-yum Acessado em: 13 de março de 2018.