

Lista 1 - Treinamento Rede GNU/Linux

Juliano Garcia de Oliveira N° USP: 9277086

26 de Janeiro, 2017

Resolução dos exercícios

1.
 - a) `cd ../../../../var/lib/mysql/<nomeArquivo>`.
 - b) Para acessar pelo caminho absoluto: `cd /usr/share/games/supertux/sounds/<nomeArquivo>`.
Relativamente: `cd ../supertux/sounds/<nomeArquivo>`
 - c) Para acessar o diretório raiz: `cd /`. Uma vez no diretório raiz, não há diferença entre acessar relativamente ou absolutamente, a definição de “acessar absolutamente” é acessar relativamente ao diretório raiz.
2.
 - a) A permissão para ambos os arquivos ficaria: `-rwx---`
 - b) A pasta ficaria com permissão `drwxrwxrwx` e os arquivos com permissão `-rwxrwxrwx`
 - c) A pasta `www` ficaria com a permissão `drw-rw---`
3. Usando as funcionalidades da tecla SysRq, que permite executar algumas operações na máquina, com certas combinações de teclas. Pra reiniciar é usado o ALT + SysRq + R , E , S , U , B . Cada uma dessas últimas letras se refere à uma operação diferente, pra garantir a integridade do sistema (interromper processos, dar `umount` nas partições, etc).
4. O comando `touch` serve pra modificar a *timestamp* de um arquivo, e tem várias opções de como fazer isso (passando a data, só o último acesso, etc). Como o comando cria o arquivo quando ele não existe (a menos que se use uma *flag* específica), é bastante usado pra criar arquivos vazios.
5. Várias pastas e arquivos críticos vão ser excluídos, outros não por não ser operação não permitida (mesmo com `sudo`) . O sistema fica praticamente inutilizável depois disso. Usando a *flag* `-i`, o `rm` irá pedir confirmação de exclusão para cada arquivo.
6. Usar o comando *find*: `find / -type f -name "*MAC122*.pdf"`.
7.
 - a) Considerando que o usuário esteja na pasta `home/rickrolled`:
 - `chmod 700 documentos/EPs Fotos/Vihs2 www/fotodamamae.jpg`
 - `chmod 755 Fotos/SouOMaximo desktop/gregoryn00b.jpg Músicas/Never Gonna Give You Up.ogg`
 - `chmod g+w www`
 - b) `chown rflowers Fotos/Vihs2/lovelove.png Fotos/Vihs2/aikilindu.jpg`
 - c) O primeiro apenas adiciona a permissão de execução ao dono do arquivo, enquanto `777` garante total acesso para todos, seja ou não do grupo em questão.

8. Se tivermos acesso à conta *root*, é possível usando o comando *su*. Por exemplo, se eu quiser rodar um *script shell* no usuário *zoao*, eu faço: `'sudo su -c pasta/bot.sh zoao`, e não preciso logar diretamente no usuário *zoao*.
9. Usando uma ferramenta chamada *crontab*, que permite agendar tarefas para serem executadas em um determinado padrão de horário e/ou data. No caso da questão, teríamos que criar um arquivo de *crontab* e então colocar a seguinte linha no arquivo : `14 3 15 1-12 * <comando>`
10.
 - a) `apt-cache search windows`
 - b) `apt list --installed`
 - c) Instalando o pacote *gcc-doc* e *gcc-doc-base* junto com suas dependências: `apt-get install gcc-doc gcc-doc-base`
 - d) Porque o *apt* tem um *easter egg* quando se digita `apt-get moo`, e o *aptitude* não :(
11. O diretório */tmp* armazena arquivos temporários que são usados por programas, muitos desses arquivos são compartilhados através desse diretório. Em várias distribuições de *linux* esse diretório faz parte do *tmpfs*, o que significa que é memória volátil. No Debian o */tmp* não faz parte do *tmpfs*, mas o */dev/shm* sim.
12. Usei a distribuição *puppet linux*. Segue o script:

```
#!/bin/bash
DOWNLINK='http://distro.ibiblio.org/puppylinux/puppy-4.3.1/pup-431.iso'
DOWNPATH='$HOME/Downloads/'
FILENAME=$(basename "$DOWNLINK")
echo "==>Linux Image downloader and installer!"
echo "==> USE AT YOUR OWN RISK! <=="
echo ""
sleep 3
echo "Starting ISO download..."
wget $DOWNLINK -P $HOME/Downloads
cd $HOME/Downloads/
echo ""
echo "Now, type the device name of the pendrive, ex: sdb"
sleep 1
lsblk
echo "Type the name of your pendrive and press [ENTER]: "
read dname
echo "The device selected is $dname. Proceed with script (y / n)? "
read ans
if [ $ans = "n" ]
then
exit
fi
echo ""
echo "Unmounting device /dev/$dname..."
# Not really recommended, but works
sudo umount /dev/$dname
echo ""
echo "Writing iso to device /dev/$dname..."
sudo dd if=$(pwd)/$FILENAME of=/dev/$dname bs=4M && sync
echo "Finished writing!"
```

13. Para cancelar um *job* de impressão, é necessário saber o *ID* do *job*. Pra isso, se usa o comando *lpstat*, por exemplo: *lpstat -d Euclides*, e então simplesmente usar o comando *cancel* para cancelar o *job* especificado, *cancel <ID>*. Para imprimir um arquivo pela linha de comando, na Euclides por exemplo: *lp /caminho/arquivo.pdf -d Euclides*
14. (...)
15. Usando o comando *last* e filtrando algumas coisas (testado na rede linux do IME):

```
#!/bin/bash
echo "==> Check if a user is logged on in the LAN "
echo "Type the username to verify if it's logged on, then press [ENTER]:"
read user
ret=$(last -w | grep still | awk '{print $1 }' | grep $user)
for x in $ret
do
if [ $user = $x ]
then
    echo "YES, the user is logged in!"
    exit
fi
done
echo "NO, the user isn't logged in!"
```

16. a) *find . -type f | grep --extended-regexp "(.\.c|.\.h)" | xargs cat | wc -l*
b) *find . -type f | grep --extended-regexp "(.\.sh|.\.h)" | xargs cat | grep -o '\int\b' | wc -l*