



SÃO
PAULO
TECH
SCHOOL

Sistemas Operacionais

Comandos de gerenciamento Linux

Marcio Santana

`marcio.santana@sptech.school`

Comandos mais usuais

ls: Lista todos os arquivos do diretório (list)

ls -l: Lista o tipo de arquivo e permissões

df: Mostra quantidade de espaço usada no disco rígido (disk file)

top: Mostra os processos consumindo memória

cd: Acessa uma determinada pasta (diretório) como por exemplo cd diretório, cd .., cd /

mkdir: Cria um diretório

rm: Remove um arquivo/diretório (**rm -r** para remover de forma recursiva)

cat: Abre um arquivo

vim: Abre o editor vim para editar/criar arquivos

nano: Abre o editor nano para editar/criar arquivos

Comandos de ajuda e documentação

apropos: Localiza comandos por pesquisa de palavra-chave

info: Abre o explorador de informações

man: Manual muito completo, pesquisa informação acerca de todos os comandos que necessitemos de saber, como por exemplo `man find`

whatis (o que é): Descreve o que um determinado comando é/faz

whereis (onde é): Localizar a página de ajuda (man page), código fonte, ou arquivos binários, de um determinado programa.

Comandos de gestão de arquivos e diretórios

chmod: Mudar a proteção de um arquivo ou diretório

- r = leitura.
- w = gravação.
- x = execução (para arquivos) ou autorização de acesso (para diretórios).
- u = as permissões do dono do arquivo. (user owner)*
- g = as permissões do grupo. (group)*
- o = as permissões dos outros usuários do sistema. (others)*
- a = todos os usuários do sistema. (all)*

Comandos de gestão de arquivos e diretórios

cp: Copia arquivos, como o copy do MS-DOS

diff: Compara o conteúdo de dois arquivos ASCII

grep: Procura um arquivo por um padrão, sendo um filtro muito útil e usado, por exemplo um `cat a.txt | grep ola` irá mostrar-nos apenas as linhas do arquivo a.txt que contenham a palavra "ola" – ex: `grep "texto" arquivo.txt` O comando grep imprime na tela as linhas que correspondem a um padrão em cada arquivo.

Um bom argumento para se utilizar (man grep) é o "-i" que ignora a distinção de letras maiúsculas e minúsculas.

mkdir: Cria uma diretório, vem de make directory"

mv: Move ou renomeia arquivos ou diretórios

Comandos de gestão de arquivos e diretórios

tar: O 'tar' do Linux significa arquivo em fita (tape archive), que é usado para criar o arquivo e extrair os arquivos do arquivo. Podemos usar o comando tar do Linux para criar arquivos compactados ou descompactados e também mantê-los e modificá-los.

zip: O comando zip compacta arquivos. Cada arquivo é compactado em um único arquivo. Se receber um arquivo como argumento, o zip compacta o arquivo e adiciona o .zip como extensão

sudo apt install zip

zip nome_arquivo.zip arquivo1 arquivo2 arquivo3

zip -r nome_arquivo.zip diretório1

sudo apt install unzip

unzip nome_arquivo.zip

Comandos de rede

ip: Manipulação do roteamento para atribuir e configurar parâmetros de rede

traceroute: Identificar a rota tomada pelos pacotes para chegar ao host

tracert: Obtém a unidade de transmissão máxima ao rastrear o caminho para o host de rede

ping: Frequentemente usado para verificar a conectividade entre o host e o servidor

ss: Obtém detalhes sobre soquetes de rede

dig: Fornece todas as informações necessárias sobre o servidor de nomes DNS

host: Imprime o endereço IP de um domínio específico e vísceras

hostname: Usado principalmente para imprimir e alterar o nome do host

curl: Transfere dados pela rede, suportando vários protocolos

Comandos de rede

mtr: Uma combinação de ping e traceroute é usada para diagnosticar a rede

whois: Obtém informações sobre domínios registrados, endereços IP, servidores de nomes

nmap: Usado principalmente para auditar a segurança da rede

speedtest-cli: Utilitário CLI de speedtest.net para verificar as velocidades da Internet

Comando `kill`

Utilizando o terminal Linux, vamos ver como matar um ou vários processos que estão em funcionando na nossa máquina. O comando **kill** necessita de argumentos para podermos encerrar um processo **kill -l**, nos lista esses argumentos

```
urubu100@sptech02608:~$ kill -l
1) SIGHUP      2) SIGINT      3) SIGQUIT     4) SIGILL      5) SIGTRAP
6) SIGABRT     7) SIGBUS     8) SIGFPE      9) SIGKILL     10) SIGUSR1
11) SIGSEGV    12) SIGUSR2    13) SIGPIPE    14) SIGALRM     15) SIGTERM
16) SIGSTKFLT  17) SIGCHLD    18) SIGCONT    19) SIGSTOP     20) SIGTSTP
21) SIGTTIN    22) SIGTTOU    23) SIGURG     24) SIGXCPU     25) SIGXFSZ
26) SIGVTALRM  27) SIGPROF    28) SIGWINCH    29) SIGIO        30) SIGPWR
31) SIGSYS     34) SIGRTMIN    35) SIGRTMIN+1  36) SIGRTMIN+2  37) SIGRTMIN+3
38) SIGRTMIN+4 39) SIGRTMIN+5 40) SIGRTMIN+6 41) SIGRTMIN+7 42) SIGRTMIN+8
43) SIGRTMIN+9 44) SIGRTMIN+10 45) SIGRTMIN+11 46) SIGRTMIN+12 47) SIGRTMIN+13
48) SIGRTMIN+14 49) SIGRTMIN+15 50) SIGRTMAX-14 51) SIGRTMAX-13 52) SIGRTMAX-12
53) SIGRTMAX-11 54) SIGRTMAX-10 55) SIGRTMAX-9  56) SIGRTMAX-8  57) SIGRTMAX-7
58) SIGRTMAX-6 59) SIGRTMAX-5 60) SIGRTMAX-4 61) SIGRTMAX-3 62) SIGRTMAX-2
63) SIGRTMAX-1 64) SIGRTMAX
```

man signal: nos dá o manual de sinais no Linux com a descrição de cada tipo de sinal

kill -nº <processo>: aplica o sinal de kill em algum processo (9 e 15 mais comuns)

Comando **ccrypt**

Utilizando o terminal Linux, vamos ver como proteger um arquivo usando **criptografia**. Iremos criptografar um arquivo utilizando o comando **ccrypt** que utiliza a cifra Rijndael, a mesma cifra usada na criptografia padrão AES. Embora o padrão AES use um tamanho de bloco de 128 bits, o ccrypt usa um tamanho de bloco de 256 bits. Uma vez criptografados os arquivos por ccrypt, uma extensão **.cpt** é adicionada aos arquivos.

ccrypt (nome do arquivo): para criptografar um arquivo

ccrypt -c (nome do arquivo.cpt): para ler um arquivo criptografado

ccrypt -d (nome do arquivo.cpt): descriptografa um arquivo

ccrypt -r (nome do diretório.cpt): criptografa um diretório e todos os arquivos

ccrypt -d -r (nome do diretório.cpt): descriptografa o diretório

Agradeço
a sua atenção!

Marcio Santana

marcio.santana@sptech.school

SÃO
PAULO
TECH
SCHOOL