

Comandos Linux

SHELL

- O Shell é um módulo que atua como interface usuário - sistema operacional, possuindo diversos comandos internos que permitem ao usuário solicitar serviços do sistema operacional;
- O Shell também implementa uma linguagem simples de programação que permite o desenvolvimento de pequenos programas (os famosos Shell script);
- Desenvolvido por S.R. Bourne em 1975, o Bourne Shell foi um dos primeiros Shells desenvolvidos. E continua sendo até hoje um dos mais rápidos e mais leves;
- O Bash (Bourne Again Shell) surgiu como uma reimplementação do Bourne Shell realizada pelo projeto GNU e se popularizou com a expansão de sistemas GNU/Linux.

Interpretador de comandos Shell

```
madoka:/home/hugo$  
[hugo@madoka hugo]$
```

- A primeira coisa a saber é que você estará utilizando um INTERPRETADOR DE COMANDOS e estará em um prompt parecido com alguns destes acima;
- Você estará dentro de um interpretador de comandos, que no caso é o bash. Você digita algum comando e o interpretador lê o que você digitou e executa a tarefa correspondente. Sem este o usuário não poderia interagir com o sistema;
- Os interpretadores mais comuns são: sh, bash, ash, csh, tcsh e ksh. Na maioria dos casos, você estará usando o bash;
- Quando quiser sair da shell, utilize o comando exit.



Nota

Nos prompts do Linux, quando se acaba com o símbolo \$, isto quer dizer que você está logado com um usuário normal. Se você estiver logado com o root, ao invés do símbolo \$, no final do prompt existirá o símbolo #

Comando SUDO

```
sudo [-u usuário] comando
```

- O comando sudo permite a usuário obter privilégios de outro usuário, em geral, o super usuário, para executar tarefas específicas dentro do sistema de maneira segura e controlável pelo administrador;
- O nome sudo é uma forma abreviada de **s**ubstitute **u**ser **d**o (fazer substituindo usuário) ou **s**uper **u**ser **d**o (fazer como super usuário);
- Um super usuário precisa definir no arquivo /etc/sudoers quais usuários podem executar sudo, em quais computadores podem fazê-lo e quais comandos podem ser executados através dele;
- A opção **-u usuário** serve para especificar qual usuário deve ser utilizado para executar o comando, se omitida sudo considera o usuário root;
- Ao ser executado, sudo confirma a identidade do usuário pedindo sua própria senha. Se o usuário tem permissão garantida na configuração, o comando é executado como super usuário e retorna. A linha de comando continua sendo a do usuário que executou sudo e não a do super usuário;
- Algumas distribuições desativam o super usuário e exigem que todas as tarefas administrativas sejam realizadas através do sudo. Ubuntu e Kurumin são exemplo destas distribuições;
- Exemplo:
sudo apt-get install vim

Pacotes DEB

- Para manipular os pacotes .deb, o Debian e seus derivados utilizam um programa chamado dpkg. Vamos a alguns procedimentos básicos deste programa.
 - Para instalar um pacote, utilize a opção -i:
`# sudo dpkg -i pacote.deb`
 - Para remover um pacote do sistema, use o -r:
`# sudo dpkg -r pacote`
 - Para forçar a remoção de um pacote, use -force-depends
`# sudo dpkg -r -force-depends pacote`
 - Para ver uma lista com todos os pacotes instalados no sistema:
`# sudo dpkg -l`
 - Para verificar se um pacote está instalado:
`# sudo dpkg -l vim`
 - Para obter informações detalhadas sobre um determinado pacote:
`# sudo dpkg -p grub`



Cuidado

Forçar a remoção de um pacote sem resolver as dependências pode causar os programas dependentes do pacote a não funcionarem corretamente. Use apenas se você souber o que estiver fazendo.

Gerenciado APT

- O APT (advanced Package Manager) é um poderoso gerenciador de pacotes que resolve as complicações de dependências de pacotes e instalação via internet. Com ele é possível com apenas um comando, baixar a versão mais atual de um programa e instalar no sistema;
- Isto foi feito inicialmente para as distribuições Debian e seus pacotes .deb, mas com o tempo o apt foi portado para o tipo de pacote .rpm, podendo ser assim usado em distribuições baseadas em RPM;
- Além dos pacotes APT e .DEB existem também os .TGZ no slackware, o .RPM no Red Hat e o YUM no Fedora;
- Os arquivos do apt estão todos localizados no diretório /etc/apt. Dentro desse diretório há um importante arquivo chamado sources.list, que contém uma lista de todos os repositórios que você irá usar. Exemplo de repositório:

```
deb http://ftp.debian.org/debian/ sarge main non-free contrib
```

Exemplo 3.1. Repositório Debian Base (Sarge)

Comandos APT

- Para atualizar a lista de pacotes, utilize:
`# sudo apt-get update`
- Para procurar por um pacote, utilizando uma palavra relacionada:
`# sudo apt-cache search DVD`
- Para instalar um pacote e todas as suas dependências:
`# sudo apt-get install pacote`
- Para remover um pacote e todas as suas dependências:
`# sudo apt-get remove pacote`
- Para atualizar todo o sistema, instalando as atualizações de todos os pacotes:
`# sudo apt-get upgrade`
- Para atualizar a versão de uma distribuição totalmente, use:



Nota

O comando `install` também atualiza um pacote caso exista uma versão superior a já instalada no sistema.

Usuários e grupos

- A noção de usuários está muito presente no linux:
 - Cada usuário registrado possui um nome de login, uma senha e um identificador numérico associado (UID);
 - Os usuários são organizados em grupos, onde sempre pertence a um grupo primário ou a outros secundários;
 - Cada grupo é identificado por um nome e um identificador de grupo (GID);
- A informação de usuários e grupo é usada pelo sistema para gerenciar o acesso aos recursos (arquivos e diretórios) e para gerência dos processos de usuários distintos;
- O usuário com UID = 0 é chamado de root (super usuário) e possui poderes especiais no sistema:
 - Acesso a todos os arquivos e diretórios;
 - Lançamento e cancelamento de qualquer processo;
 - Montagem de diretórios de rede e de dispositivos externos;

O usuário root

- O linux “vê” o usuário root como algo especial, já que ele pode alterar a configuração do sistema, configurar interfaces de rede, manipular usuários e grupos, alterar prioridades de processamento, entre outras;
- Por ser tão poderoso, o usuário root é perigoso. Por isso, ele só deve ser usado em situações que não podem ser trabalhadas por usuários que não possuem privilégios de super usuário;
- O usuário root é tão importante que até sua senha deve ser bem elaborada. É recomendável que ela tenha ao menos 8 caracteres e que misture letras e números;
- Quando estiver utilizando o sistema e necessitar realizar alguma função com privilégios de root, execute o seguinte comando:

su

- O sistema pedirá a senha do root para continuar. Para fazer que com o usuário atual adquira toda a configuração do usuário root, digite:

su -

- Para voltar a ser usuário normal digite:

exit

LOGIN E LOGOUT

- **login** - inicializa a sessão com o nome do usuário
- **logout** - finaliza sessão
- **exit** - finaliza sessão
- **shutdown** - Reinicia o computador
 - shutdown -r now - reinicializa
 - Shutdown -h now - encerra
 - Ctrl + Alt + Del - reinicializa
- **su** - alterna do usuário atual para o root
- **halt** - desliga o computador
- **poweroff** - desliga o computador
- **reboot** - reinicia o computador

Identificação de usuários

- **id** - mostra o id do usuário (userId, groupId, full name, home directory, login shell)
- **logname** - mostra o login do usuário
- **who** - identifica os usuários ativos
- **whoami** - mostra o seu login
- **passwd** - altera a senha

Outros comandos

- Cat - mostra o conteúdo de um arquivo
- Mkdir - cria um diretório novo
- Passwd - altera a senha do usuário
- Rmdir - apaga um diretório específico, desde que vazio
- Useradd - cria uma nova conta de usuário
- Userdel - apaga a conta do usuário especificado
- Uptime - mostra a quantas horas seu computador está ligado
- W - mostra os usuário logados no sistema atualmente