Aula 04 - Hackeando a linha de comando

LibrePlanet São Paulo

12 de março de 2015

Terminal

- Um terminal atua como uma interface para entrada e saída de dados no computador.
- Existem diversos emuladores de terminal no GNU/Linux, como o gnome-terminal, o urxvt, o Xfce Terminal, o xterm, etc.
- Para abrir o gnome-terminal, por exemplo, aperte alt-f2 e digite "gnome-terminal"

```
$  echo "Olá mundo!"
```

Shell

- Programa de computador que oferece uma interface de interação do usuário com o sistema, através de uma *linha de comando*.
- Interface poderosa, que oferece muita agilidade e controle sobre a execução de programas
- Permite executar programas, controlando a entrada e saída, encadear a execução de programas, e **scripting.**
- Existem diversos shells para o GNU/Linux: sh, **bash, ksh, zsh, csh, ...**

Por que usar linha de comando?

- Agilidade na execução de tarefas simples.
- Scripting para agilizar execução de tarefas complexas.
- **Pleno controle sobre a execução do programa!**
- Não ficar limitado por aquilo que é oferecido pela interface gráfica.
- Porque é legal!

Prompt

- Indica que o shell está pronto para receber um comando.

```
gabriel@argo$  cd ~/mc102
```

Comandos

Os comandos de terminal têm o seguinte formato

```
$  comando [arg1] [arg2] ...
```

- Parâmetros podem ser opcionais e normalmente são passados após um hífen "-" na forma abreviada ou após dois hífens, na forma completa. Exemplo:

```
$  ls -h
$  ls --help
```

- Lembre-se que isto é uma convenção, e muitos programas podem não seguir!

echo

```
gabriel@argo$  echo “Bixos 2015”
```

ls -a

- *ls -a* : Lista todos os arquivos ocultos.

```
$  ls -a
.config    .vimrc    .bashrc
calcI.pdf    Desktop    Documents
F129      lab03.c     MC102
```

ls -l

- *ls -l* : Lista arquivos e seus atributos.

```
$     ls -l
total 208K
-rw-r--r-- 1 ivan ivan 188K Mar 6 17:09 calcI.pdf
drwxr-xr-x 2 ivan ivan 4,0K Mar 6 17:09 Desktop
drwxr-xr-x 2 ivan ivan 4,0K Mar 6 17:09 Documents
drwxr-xr-x 2 ivan ivan 4,0K Mar 6 17:09 F129
-rw-r--r-- 1 ivan ivan  294 Mar 6 17:09 lab03.c
drwxr-xr-x 2 ivan ivan 4,0K Mar 6 17:09 MC102
```

cd

- Caminha pela árvore de diretórios.

```
gabriel@argo$  cd mc102
gabriel@argo$  cd ~
gabriel@argo$  cd /home/gabriel/
gabriel@argo$  cd -
gabriel@argo$  cd ..
```

- Verifique com pwd!

mkdir

- Criar diretórios

```
gabriel@argo$  mkdir projeto1
```

rm

- Remover arquivos

```
gabriel@argo$  rm file1 file2 file3
gabriel@argo$  rm -r projeto1
```

- Cuidado! Não existe "lixeira", uma vez removido não há um mecanismo direto para desfazer.

Movimentar arquivos

- *cp* : Copia um arquivo de um diretório para outro.

```
$  cp arquivo1 arquivo2
$  cp -r dir1 dir2
```

- *mv* : Move um arquivo de um diretório para outro.

```
$  mv arquivo1 arquivo2
```

- *wget* : Faz download do arquivo para a sua máquina.

```
$  wget http://www.gnu.org/
```

grep

- Imprime linhas de arquivo que correspondam a um padrão

```
$  grep “GNU” free_software.rules
$  grep -i “gNu Is Not” whats_gnu.rules
$  grep -v “windows” good_sw_that.rules
```

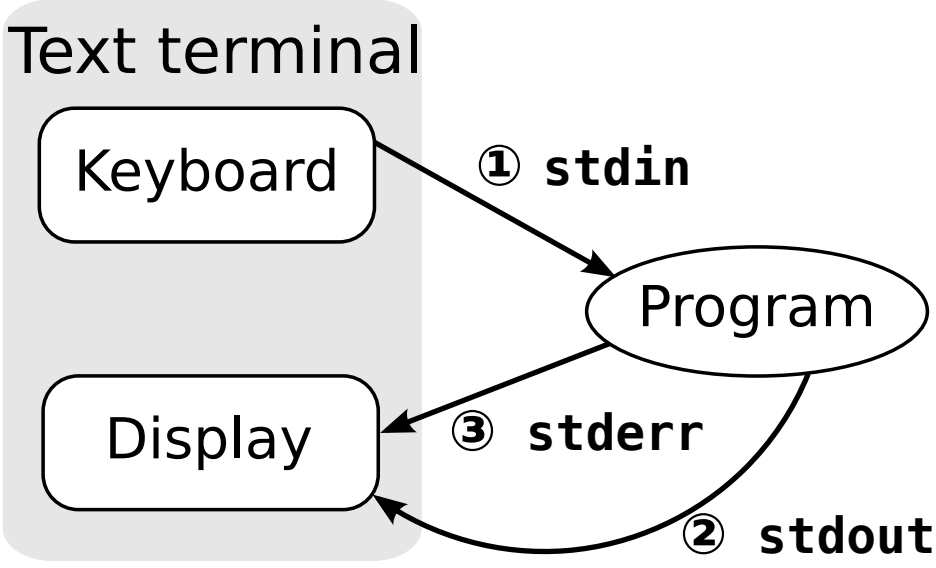
Permissões de arquivos

- Controle de acesso - Leitura, Escrita, Execução.
- Três tipos de pemissão: **U**ser, **G**roup, **O**ther, **A**ll.
- chmod <QUEM>[+|-]<PERM> <ARQUIVO>

```
$  chmod u+r meucodigo.c
$  chmod go-rwx senhas.txt
$  chmod a+x meuprog
```

Entrada e saída padrão

- Arquivos pré-conectados à entrada e saída do seu programa.
- stdin, stdout, stderr;
- fprintf (stderr, "Ops! Não vou passar no Susy! ");



Redirecionamento de I/O

- Redirecionar a entrada a partir de um arquivo texto "input.txt":

```
$  calculator < input.txt
```

- Redirecionar a saída para um arquivo de texto "output.txt":

```
$  calculator > output.txt
```

- Redirecionar stdout para arquivo1 e stderr para arquivo2:

```
$  calculator 1> arq1 2> arq2
```

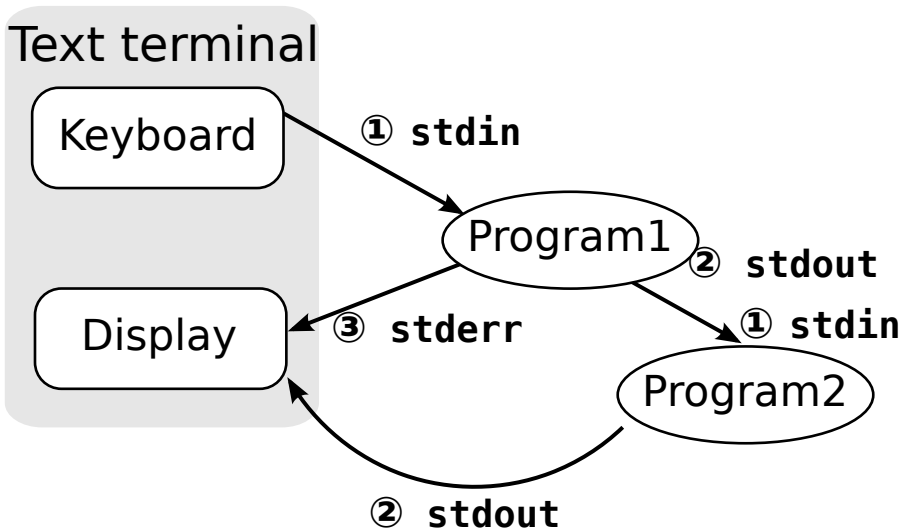
- Append ao final do arquivo:

```
$  calculator >> out.txt
```

Pipe - Encadeando programas

- É possível redirecionar a saída de um programa para outro com pipe "|":

```
$  cat list_of_sw | grep "GNU" | less
$  cat gnu_packages | wc -l
```



Exercício - Comandos básicos

Utilizando o terminal e os comandos vistos execute o seguinte procedimento:

- `http://www.students.ic.unicamp.br/~ra116931/esb.txt`
- Quantas falas LUKE tem no filme?
- Quantas vezes a princesa LEIA fala o nome de Luke?
- *spoiler alert* Descubra o número da linha em que VADER revela seu segredo a LUKE?

Scripting

- Automatizar tarefas repetitivas.
- Não é compilado. O shell intrepreta os comandos dinamicamente.
- suporta toda a sintaxe usual da linha de comando.
- Linguagem de programação completa!
- for, while, if-then-else, variáveis,...

SSH - Secure Shell

- Protocolo seguro para acesso remoto.
- Permite executar programas e acessar seus dados em máquinas remotas.
- O IC oferece uma máquina com acesso SSH para os alunos [user]@ssh.students.ic.unicamp.br
  - Executar e desenvolver seus programas.
  - Imprimir
  - ...

Gabriel Krisman Bertazi - krisman@libreplanetbr.org
Martin Ichilevici de Oliveira - martin@libreplanetbr.org
Sergio Durigan Junior - sergiodj@libreplanetbr.org
libreplanet-br-sp@libreplanet.org || IRC: #lp-br-sp (Freenode.net)