Automatizando com Shell Script: uma breve introdução prática à poderosa linguagem de comandos do mundo *nix

Daniel Bauermann
dbauermann@uol.com.br

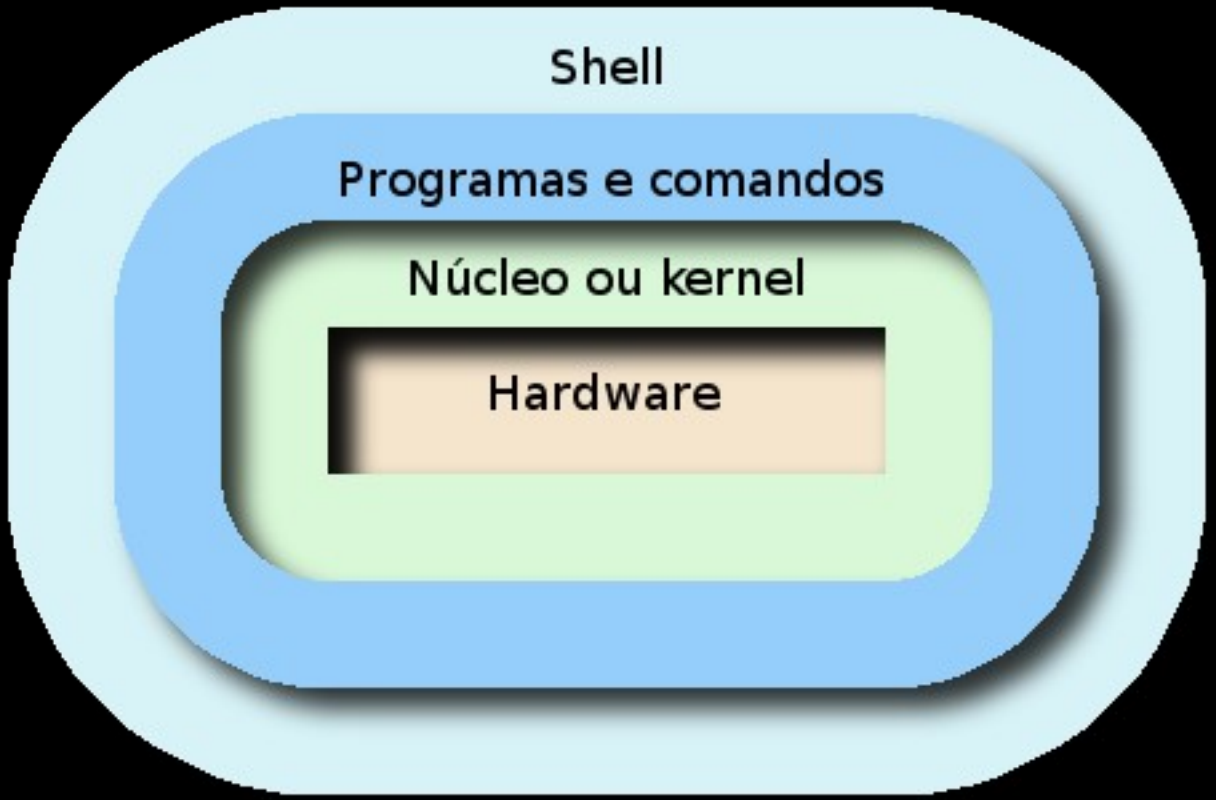
O que vamos ver hoje?

- Introdução sobre Shell Script
- Um curso básico de Shell Script
- Praticando aprendizado

Introdução

O que é Shell?

- Shell = Concha 



Fonte: http://twiki.softwarelivre.org/bin/view/TWikiBar/TWikiBarPapo001

Para que usamos o Shell?

- Executar programas do sistema

```
$ cd documentos
$ cat conteudo.txt
$ ls
$ rm arquivo_antigo.pdf
```

- MSDOS melhorado (**bem** melhorado!)

Existem opções de Shell?

- Bourne Shell (sh)
- Korn Shell (ksh)
- Bourne Again Shell (bash)
- C Shell (csh)
- TENEX C Shell (tcsh)
- Z Shell (zsh)

Qual a diferença do Shell Script?

- + que comandos básicos, possui estruturas de linguagens de programação
  - variáveis, if, for, while, ...
- Diversos usos
  - CGIs
  - instaladores
  - rotinas *backup*
  - rotinas de administração

Curso básico

O mínimo exigido

- Conhecimento básico de comandos Linux
  - (`date`, `echo`, etc)
- 1 editor
- Lógica

Executando comandos

- Alguns comando básicos

```
$ clear
$ cd /home/$USER
$ ls
$ date
```

Primeiro exemplo

- Um *script* reunindo um conjunto de comandos

```
$ vi primeiro.sh
```

```
#!/bin/bash
# primeiro.sh - Script para execucao de alguns
#               comandos.
#
# 06/05/2009  Daniel Bauermann

# Limpar tela e listar arquivos
clear
cd /home/$USER
ls
# Informa data/hora
date
```

Explicando (1)
Escolhendo o Shell

```
#!/bin/bash
```

- Deve ser a primeira linha
- Deve iniciar com #!
- Deve ser informado caminho completo

Explicando (2)
Comentários

```
# primeiro.sh - Script para execucao de alguns
#               comandos.
#
# 06/05/2009  Daniel Bauermann
```

- Não é obrigatório
- Mas ajuda MUITO

Executando

- Ajustar permissão de execução

```
$ chmod u+x primeiro.sh
```

- Rodar *script*

```
$ ./primeiro.sh
```

Condições (1)
Comando `if`

- O famigerado `if`
- Testa um **comando** (não uma condição!)
- Sintaxe:

```
if COMANDO
then
  comandos
else
  comandos
fi
```

Condições (1)
Comando `if`

- O famigerado `if`
- Testa um **comando** (não uma condição!)
- Sintaxe:

```
if test EXPRESSÃO
then
  comandos
else
  comandos
fi
```

Condições (1)
Comando `if`

- O famigerado `if`
- Testa um **comando** (não uma condição!)
- Sintaxe:

```
if [ EXPRESSÃO ]
then
  comandos
else
  comandos
fi
```

Condições (2)

Comando `if`

- Expressões

	Descrição
-lt	menor que (*LessThan*)
-gt	maior que (*GreaterThan*)
-le	menor igual (*LessEqual*)
-ge	maior igual (*GreaterEqual*)
-eq	igual (*Equal*)
-ne	diferente (*NotEqual*)
=	igual (*string*)
!=	diferente (*string)*

Condições (3)

Comando `if`

- Arquivos e união de expressões

	Descrição
-d	diretório
-f	arquivo
-a	e lógico (*And*)
-o	ou lógico (*Or*)

`man test`

Segundo exemplo

- Criando uma condição de avaliação

```
$ vi segundo.sh
```

```
#!/bin/bash
# segundo.sh    Saudacao conforme o horario.
#
# 06/05/2009  Daniel Bauermann

if [ `date +%H` -lt 13 ]
then
   echo   Bom dia
elif [ `date +%H` -lt 18 ]; then
   echo   Boa tarde
else
   echo   Boa noite
fi
```

Variáveis

- Sintaxe atribuição:

  `VARIAVEL=valor`

  (sem espaços entre =)

- Sintaxe uso:

  `$VARIAVEL`

- Sem prototipação

Segundo exemplo com variável

```
$ vi segundo.sh
#!/bin/bash
# segundo.sh    Saudacao conforme o horario.
#
# 06/05/2009  Daniel Bauermann

HORA=`date +%H`
if [ ${HORA} -lt 13 ]
then
   echo   Bom dia
elif [ ${HORA} -lt 18 ]; then
   echo   Boa tarde
else
   echo   Boa noite
fi
```

Recebendo dados

- Comando `read`
- Sintaxe:

```
read VARIÁVEL
```

Terceiro exemplo

- Interagindo e limpando a tela

```
$ vi terceiro.sh
```

```
#!/bin/bash
# terceiro.sh    Le e limpa (ou não) a tela.
#
# 06/05/2009  Daniel Bauermann

# Imprime pergunta e le resposta do usuario
echo   Deseja limpar a tela? [sn]
read RESPOSTA

# Limpa tela se pressionado S
if [   ${RESPOSTA}   =   S   -o   ${RESPOSTA}   =   s
then
   clear
fi
```

Laços (1)
Comando `while`

- Executa enquanto um **comando** seja válido
- Sintaxe:

```
while COMANDO
do
   comandos
done
```

Quarto exemplo

- Um contador

```
$ vi quarto.sh
```

```
#!/bin/bash
# quarto.sh    Um simples contador.
#
# 06/05/2009  Daniel Bauermann

# Imprime valores de 1 a 10
NUMERO=0
while [ ${NUMERO} -lt 10 ]
do
   NUMERO=$((NUMERO+1))
   echo   Contando: ${NUMERO}
done
```

Laços (1)
Comando `for`

- Executa individualmente valores de um conjunto
- Sintaxe:

```
for VARIÁVEL in LISTA
do
   comandos
done
```

Quinto exemplo

- Contador com `for`

```
$ vi quinto.sh
```

```
#!/bin/bash
# quinto.sh    Um simples contador com   for  .
#
# 06/05/2009  Daniel Bauermann

# Imprime valores de uma lista
for NUMERO in um dois três quatro cindo
do
   echo   Contando: ${NUMERO}
done
```

O poder do *pipe* (|)

- Redireciona saída de um comando como para entrada de outro

- Exemplo:

```
$ ls | wc -l
```

Praticando

Exercício 1

Listar todos os usuários cadastrados no meu Linux

Exercício 1

Listar todos os usuários cadastrados no meu Linux

- Listando conteúdo de arquivos: `cat`

Exercício 1

Listar todos os usuários cadastrados no meu Linux

- Listando conteúdo de arquivos: `cat`
- Cortando em pedaços: `cut`

Exercício 1

Listar todos os usuários cadastrados no meu Linux

- Listando conteúdo de arquivos: `cat`
- Cortando em pedaços: `cut`
- Substituindo *strings*: `tr`

 Abuse do `man`

Solução exercício 1

```
$ vi exercicio1.sh
```

```
#!/bin/bash
# exercicio1.sh    Lista usuarios do sistema.
#
# 06/05/2009  Daniel Bauermann

# Le usuarios e formata saida
cat /etc/passwd | cut -d : -f 1,5 | tr : '\t'
```

Exercício 2

Listar todos os usuários cadastrados no meu Linux (resolver com `while` *)*

- Dica: `cat ARQUIVO | while read LINHA`

Solução exercício 2

```
$ vi exercicio2.sh
#!/bin/bash
# exercicio2.sh    Lista usuarios do sistema.
#
# 06/05/2009  Daniel Bauermann

# Le usuarios e formata saida
echo -e   Usuário\tNome
cat /etc/passwd | while read LINHA
do
   USUARIO=`echo ${LINHA} | cut -d : -f 1`
   NOME=`echo ${LINHA} | cut -d : -f 5`
   echo -e   ${USUARIO}\t${NOME}
done
```

Exercício 3

E o meu backup? Não vou fazer?

- Definir arquivos a copiar (façamos de maneira elegante: crie uma variável)
- Definir destino da copia
- Criar!

Solução exercício 3

```
$ vi meu_backup.sh
#!/bin/bash
# meu_backup.sh    Criar copia de meus arquivos.
#
# 06/05/2009  Daniel Bauermann

### Variaveis
ARQUIVO_TGZ=`date +  /tmp/copia_%Y%m%d-%H%M.tgz  `
DIRETORIOS_COPIAR=  /etc /home

### Execucao
tar cvzf ${ARQUIVO_TGZ} ${DIRETORIOS_COPIAR}
## (copia para o local apropriado...) ##
# Remove arquivo temporario
if [ -f ${ARQUIVO_TGZ} ]; then
   rm ${ARQUIVO_TGZ}
fi
```

Sobrou tempo?

Melhorando a aparência

- Criar “janelas” com `dialog`
- Versões para ambiente X:
  - Xdialog
  - Kdialog
  - gdialog
  - Zenity

Veja com seus próprios olhos (1)

```
$ dialog --msgbox 'Meu nome é XYZ' \
    5 40

$ dialog --title 'Preencha' \
   --inputbox 'Informe seu nome' 0 0

$ dialog --title 'Calendário' \
   --calendar '' 0 0
```

Veja com seus próprios olhos (2)

```
$ dialog --title 'Opções' \
  --menu 'Escolha as suas' \
  0 0 0 \
  um 'Um' \
  dois 'Dois' \
  três 'Três'
```

Veja com seus próprios olhos (3)

```
$ dialog --title 'Opções' \
  --checklist 'Escolha as suas' \
  0 0 0 \
  um 'Um' off \
  dois 'Dois' off \
  três 'Três' on
```

Abuse do `man`

Sexto exemplo

- Um exemplo “gráfico”

```
$ vi sexto.sh
```

```
#!/bin/bash
# sexto.sh    Usando o dialog.
#
# 06/05/2009  Daniel Bauermann

# Solicita nome e armazena em variavael
NOME=$( dialog --title   Sexto   --stdout \
           --inputbox   Informe seu nome:   0 0 )

# Emite saudacao
dialog --title   Sexto   --msgbox   Olá ${NOME}   0 0
```

Finalizando

E agora, o que fazer?

- Pense em suas tarefas cotidianas
- Encontre algo para automatizar
- Elabore
- Arregace as mangas
- Tente, experimente
- Diverta-se acima de tudo!!!

Referências

- Aurélio Marinho Jargas
  - http://aurelio.net/shell/
- Papo de Botequim – Júlio Neves
  - http://www.julioneves.com

Automatizando com Shell Script: uma breve introdução prática à poderosa linguagem de comandos do mundo *nix

Perguntas?

Daniel Bauermann
dbauermann@uol.com.br

Apresentações

- 06/05/2009 – [SoftwareLivreVS] Semana da Informática IENH – Novo Hamburgo – RS