



Treinamento: FreeBSD – Introdução e Prática



Instrutor: Danilo Perillo Chiacchio



Nessa Aula Vamos Aprender:

- ✓ Entender o que são Shells e como podem ser utilizados.





Tópico 4: Administração do Sistema Operacional

Entendendo o que são Shells

- É muito importante entender o que é um ambiente de Shell. Podemos dizer que um ambiente de shell é uma área destinada a prover um ambiente de trabalho que proporcione a interação do usuário com o sistema operacional. Existem vários tipos de shell que podem ser instalados via ports (`/usr/ports/shells`);
- Podemos dizer adicionalmente, que um shell é como um programa utilizado para realizar rotinas no sistema operacional. Essas rotinas podem ser automatizadas utilizando para isso uma linguagem de programação para shells, chamada “shell script”;
- Basicamente, é possível criar um arquivo contendo todas as instruções/rotinas necessárias e executá-lo de uma única só vez. Além disso, é possível programar a execução automática desse shell script dentro do sistema, deixando todo o processo automatizado.





Tópico 4: Administração do Sistema Operacional

Entendendo o que são Shells

- É importante ressaltar que os scripts em shell não são instruções compiladas, onde dessa forma é necessário ficar atento com os dados que neles contêm. É sempre interessante revisar linha por linha de um script shell antes de realizar a sua execução no sistema. É extremamente interessante (em ambientes de produção com alta criticidade), testar o script shell em ambiente de laboratório para verificar seu correto funcionamento;
- Para que um script shell possa ser executado, ele precisa possuir a permissão de “bit execute” ou ser executado utilizando o interpretador de shell;
- A seguir, exemplo de um simples script em shell para facilitar o entendimento do que é um script em shell e o que pode ser feito com ele:





```
#!/bin/sh
clear
echo "#####"
echo "## Script para verificar existencia de usuario no sistema ##"
echo "#####"
echo ""
echo "Digite o nome do usuario:"
read user
echo ""
echo "Por favor aguarde! Sua solicitacao esta em processamento..."
echo ""
sleep 2
#
verifica_user=`cat /etc/passwd | grep -v ^# | awk -F':' '{print }' | grep $user`
resultado=$?
#
if [ $resultado == 0 ] ; then
    echo "Usuario $user existe, OK"
else
    echo "Usuario $user inexistente!"
fi
```





Tópico 4: Administração do Sistema Operacional

Entendendo o que são Shells

- Explicando:
 - A primeira linha “#!/bin/bash” é na verdade uma convenção utilizada para especificar que o arquivo trata-se de um script shell, onde na qual foi devidamente construído para ser executado com o interpretador de comandos /bin/sh;
 - As linhas iniciadas com “#” não são processadas/interpretadas pelo shell. Dessa forma, são muito utilizadas para inserir comentários dentro de um script em shell ou separar blocos de comandos para melhor organizar o conteúdo do script;
 - As linhas com comando “echo” são utilizadas para mostrar determinada informação na saída padrão, ou seja, na tela no momento da execução do script;





Tópico 4: Administração do Sistema Operacional

Entendendo o que são Shells

- O comando “read” é responsável por coletar os dados inseridos pelo usuário e armazená-los dentro de uma variável, que nesse exemplo é chamada de “user”;
- O comando “sleep” é utilizado para suspender a execução de um próximo comando por alguns segundos. Em nosso exemplo, ele está “retardando” a execução do próximo comando por 2 segundos;
- Logo em seguida, estamos definindo uma variável chamada “verifica_user”. Essa variável contém os comandos necessários para fazer a busca do usuário no sistema, utilizando como parâmetro o usuário fornecido no momento da execução do script;





Tópico 4: Administração do Sistema Operacional

Entendendo o que são Shells

- A variável resultado tem por objetivo armazenar o “error level” da execução do último comando. O “error level” pode ser obtido através da variável \$?;
- Já a estrutura condicional com “if” é utilizada para mostrar a mensagem na tela sobre a existência ou não do usuário no sistema. Se o usuário existir no sistema, o “error level” do comando de verificação será “0”. Qualquer valor diferente disso é considerado problema e, nesse caso, significa que o usuário não existe no sistema.





Tópico 4: Administração do Sistema Operacional

Entendendo o que são Shells

- Vamos verificar o funcionamento do script, solicitando que o script valide a existência de um usuário chamado “toor”:

```
#####  
## Script para verificar existencia de usuario no sistema ##  
#####  
  
Digite o nome do usuario:  
toor  
  
Por favor aguarde! Sua solicitacao esta em processamento...  
  
Usuario toor existe, OK  
root@freebsd03:~/scripts # █
```





Tópico 4: Administração do Sistema Operacional

Entendendo o que são Shells

- Observe que o usuário “toor” existe no sistema. Vamos agora pesquisar a existência do usuário chamado “joao”. Observe que o usuário “joao” não existe e o script reporta isso de forma correta:

```
#####  
## Script para verificar existencia de usuario no sistema ##  
#####  
  
Digite o nome do usuario:  
joao  
  
Por favor aguarde! Sua solicitacao esta em processamento...  
  
Usuario joao inexistente!  
root@freebsd03:~/scripts # █
```

