

SHELL SCRIPT COM PHP

Autor: Eduardo Assis <eduardo at proredes.com.br>

Data: 26/08/2003

COMEÇANDO COM SHELL SCRIPT

Shell script é uma linguagem de programação de scripts do Linux ([//www.vivaolinux.com.br/linux/](http://www.vivaolinux.com.br/linux/)). É bastante útil para nós que somos administradores de redes. O básico do shell script começa com `#!/bin/bash` dentro de um arquivo que pode ter a extensão que desejar, desde que no início do arquivo contenha o caminho para o interpretador.

OS SHELLS MAIS POPULARES

- ▶ SH (Bourne Shell) - shell original do Unix, porém com recursos muito limitados.
- ▶ CSH (C Shell) - shell desenvolvido para permitir um conjunto de comandos e ambientes de script derivado da linguagem C.
- ▶ KSH (korn Shell) - o primeiro shell a introduzir recursos avançados, sendo considerado o mais popular em sistemas Unix.
- ▶ BASH (Bourne Again Shell) - shell padrão em distribuições Linux, uma implementação do SH, com recursos avançados, tais como: edição de linha de comando, histórico de comandos e término de nomes de arquivos e/ou diretórios.

VERIFICANDO O SHELL ATUAL

Digite:

```
# echo $SHELL  
/bin/bash
```

No Meu caso uso *Bash* e mostrarei como usar com ele.

EDITANDO O SHELL

```
$ mcedit shell.sh  
ou  
$ mcedit shell
```

A extensão não importa desde que comece pela linha (`#!/bin/bash`)

```
#!/bin/bash  
  
echo "Digite seu nome"  
read nome  
echo $nome
```

Salve o arquivo e temos que dar permissão para executar:

\$ chmod +x shell.sh (ou shell, dependendo do nome que você deu ao arquivo)

depois:

./shell.sh

Digite seu nome

eduardo

eduardo

No caso digitei *eduardo* e ele me mostrou a saída: *eduardo*

Mais um exemplo:

```
#!/bin/bash

clear
echo "Bem vindo ao meu Script"
echo "Escolha uma das opções abaixo"
echo
echo "[1] Listar as portas do seu computador"
echo "[2] Listar os arquivos do diretório atual"
echo "[3] Sair"
echo -n "Digite a opção: "
read opcao
case $opcao in
  1) nmap -sS localhost
     sleep 3
     ;;
  2) ls -l
     sleep 3
     ;;
  3) exit
     clear
     ;;
*)
```

Esse também já está interagindo com o usuário e dependendo do número que ele digitar o script vai tomar uma decisão.

Acho que pra quem nunca mexeu com shell script já deu pra ter uma noção, basta saber que o shell script roda os comandos do Linux.

Para saber mais procure comprar o Livro:

- "Programação Shell Linux", de Julio Cezar Neves / Editora Brasport

Que poderá ajudar mais ainda ... pois eu tenho ele e me ensinou várias coisas.

INTEGRANDO SHELL COM PHP

Bem como já aprendemos a usar shell script passaremos a usar PHP junto.

Foi publicado na *Revista do Linux* ([//www.vivaolinux.com.br/linux/](http://www.vivaolinux.com.br/linux/)) edição 43 e estou postando para que outras pessoas tenham acesso.

\$ mcedit script.php (lembrando que a extensão não importa)

```
#!/usr/bin/php -q

<?
echo " Ola, World \n" ;
?>
```

\$ chmod +x script.php

depois:

\$./script.php

Aparecerá:

Ola, World

Pronto, temos nosso primeiro shell script usando PHP!

Quem já manja de PHP tenho certeza que fica pensando já no que pode fazer e digo que realmente o que vocês estão pensando... sim, dá para fazer sim, pois o PHP é uma excelente ferramenta.

Vejamos mais um exemplo:

```
#!/usr/bin/php -q
<?
echo " Informe uma palavra de até 100 Caracteres: ";

$var_stdin = fopen('php://stdin', 'r');
$var_palavra = str_replace("\n", "", fgets($var_stdin,100));
echo " A palavra digitada foi $var_palavra e possui " . strlen($var_palavra) . " caracteres \n ";
?>
```

e depois:

\$./script.php

Outro exemplo e o mais esperado: *Como interagir um shell script com o PHP.*

\$ mcedit script.sh

```
#!/bin/bash

echo -n " Digite Seu Nome: "
read nome
echo $nome
echo
#!/usr/bin/php -q << PHP
<?
echo " Ola, World \n" ;
?>
PHP
```

Pronto, temos shell script com PHP!

SHELL, PHP E BANCO DE DADOS

Utilizaremos agora conexão com o banco de dados MySQL.

Vamos entender que você já sabe mexer com MySQL e que tem um banco criado com o nome de *intranet* e uma tabela com as seguintes características:

- nome da tabela: agenda
- campos: nome, tel

Tendo isso configurado passaremos para o script:

```
#!/bin/bash

echo -n "Digite o Nome: "
read nome
echo -n "Digite o numero de seu Telefone: "
read tel
/usr/bin/php -q << PHP
<?
\ $host = "localhost";
\ $user = "usuario";
\ $senha = "senha";
\ $conexao = mysql_connect(\ $host,\ $user,\ $senha); mysql_select_db("intranet", \ $conexao);
\ $insert = "insert into agenda (nome,tel) values ('$nome','$tel)";
mysql_query(\ $insert,\ $conexao);
echo "Dados Inseridos com Sucesso !!! \n";
?>
PHP
```

Nesse exemplo o usuário digitará o nome e depois um telefone e será gravado na tabela.

Script para consulta de dados no banco:

```
#!/bin/bash

echo -n "Digite o nome para ser consultado: "
read nome
/usr/bin/php -q << PHP
<?
\ $host = "localhost";
\ $user = "usuario";
\ $senha = "senha";
\ $conexao = mysql_connect(\ $host,\ $user,\ $senha); mysql_select_db("intranet", \ $conexao);
\ $consultar_login="select * from agenda where nome like '%$nome%'";
\ $resultado=mysql_query(\ $consultar_login,\ $conexao) or die (mysql_error());
while(\ $campo = mysql_fetch_array(\ $resultado)) { \ $nome = \ $campo["nome"]; \ $tel = \ $campo["tel"];
echo "\ $nome | \ $tel \n" ; }
?>
PHP
```

Aqui estamos usando uma consulta com a opção de like no SQL, bastando para isso digitar apenas uma palavra para consulta.

Vamos entender o que acontece com o insert e com o consulta

Abrimos a script com shell script (*#!/bin/bash*) e depois colocamos outro interpretador dentro dele (*/usr/bin/php -q << PHP*). Para finalizarmos usamos o comando PHP e podemos ainda continuar com o shell script normalmente.

Só mais um lembrete: para usarmos variáveis do shell script dentro do PHP usamos a variável (\$) seguido do nome do campo e para usarmos variáveis dentro do PHP, usamos (\) seguido do caractere (\$), ficando assim:

\ \$nome

CONCLUSÃO

Espero ter ajudado alguém e que com esse artigo ele possa desenvolver de acordo com o que ele vai usar.

Qualquer dúvida me contate por mail...

Esses dados foram tirados da *Revista do Linux* ([//www.vivaolinux.com.br/linux/](http://www.vivaolinux.com.br/linux/)) e alterado por tucs.

Abraços

tucs

[↩ Voltar \(verArtigo.php?codigo=347\)](#)