



**Curso de Operador de Tecnologia da Informação e  
Comunicações**

# **Desenvolvimento Web**

## **Característica do PHP**

**2024**



Objetivo de  
Aprendizado

Entender como o PHP funciona e quais os requisitos necessários para ter um servidor completo capaz de hospedar páginas em PHP.



A linguagem PHP é uma das mais utilizadas no mundo devido a sua versatilidade e também por causa da facilidade de aprendizado. Existem várias ferramentas que podem ser utilizadas com o PHP relacionadas a frameworks que podem ajudar no desenvolvimento rápido e seguro.

# Introdução a linguagem PHP



A abreviação PHP é um acrônimo recursivo para PHP Hypertext Preprocessor, originalmente Personal Home Page [Achour et al., 2014]. O PHP é uma linguagem de **script** open source projetada para o **desenvolvimento web**, mas também usada como uma linguagem de programação de propósito geral.

Os scripts PHP, diferente do Javascript, são interpretados pelo **módulo PHP** no lado **servidor**, gerando uma página web para ser visualizada no lado cliente.

O PHP pode ser utilizado em diferentes plataformas como o Linux, várias variantes do Unix (incluindo HP-UX, Solaris e OpenBSD), Microsoft Windows, Mac OS X, RISC OS, e provavelmente outros. O PHP também é suportado pela maioria dos servidores web atuais, incluindo Apache, Microsoft Internet Information Server, Personal Web Server, Netscape and iPlanet Servers, O'Reilly Website Pro Server, Caudium, Xitami, OmniHTTPd, e outros.

Talvez a mais forte e mais significativa característica do PHP é seu suporte a uma ampla variedade de banco de dados. Entre eles estão o MySQL, PostgreSQL, IBM DB2, Oracle (OCI7 and OCI8), mSQL e muitos outros.

Resumidamente, temos:

- uma linguagem de script;

- open source;

- projetada para o desenvolvimento web;

- Os scripts PHP são interpretados pelo módulo PHP;

- O PHP pode ser utilizado em diferentes plataformas; e

- suporte a uma ampla variedade de banco de dados.

## INSTALAÇÃO

A pilha "LAMP" é um grupo de softwares open source que é tipicamente instalado em conjunto para permitir a um servidor hospedar websites dinâmicos e aplicações web. Este termo é atualmente um acrônimo que representa o sistema operacional Linux, com o servidor web Apache. A informação do site é armazenada em uma base de dados MySQL, e o conteúdo dinâmico é processado pelo PHP.

Neste exemplo, você vai precisar ter um servidor Ubuntu 18.04 com uma conta de usuário que não seja root, com privilégios sudo configurada e um firewall básico. Isso pode ser configurado utilizando nosso guia de Configuração Inicial de servidor com Ubuntu 18.04

## Passo 1 — Instalação do Apache e Atualização do Firewall

O servidor web Apache está entre os servidores web mais populares do mundo. É bem documentado, e tem sido amplamente utilizado em grande parte da história da web, o que faz dele uma ótima escolha padrão para hospedar um website.

Instale o Apache apt:

```
sudo apt update
sudo apt install apache2
```

Como estamos utilizando um comando sudo, essas operações são executadas com privilégios de root. Ele irá pedir a senha do usuário comum para verificar suas intenções.

Uma vez que você tenha digitado sua senha, o **apt** irá lhe dizer quais pacotes ele planeja instalar e quanto de espaço extra em disco ele irá consumir. Pressione **Y** e aperte **Enter** para continuar, e a instalação prosseguirá.

### Ajustar o Firewall para Permitir Tráfego Web

Agora, assumindo que você seguiu as instruções de configuração inicial do servidor para habilitar o firewall UFW, certifique-se de que seu firewall permite tráfego HTTP e HTTPS. Você pode certificar-se de que o UFW tem um perfil de aplicativo para o Apache assim:

```
sudo ufw app list
```

Output

```
Available applications:
```

```
Apache
Apache Full
Apache Secure
OpenSSH
```

Se você olhar para o perfil **Apache Full**, ele deve mostrar que ele habilita tráfego para as portas 80 e 443:

```
sudo ufw app info "Apache Full"
```

Output

```
Profile: Apache Full
Title: Web Server (HTTP,HTTPS)
Description: Apache v2 is the next generation of the
omnipresent Apache web
server.
```

Ports:

```
80,443/tcp
```

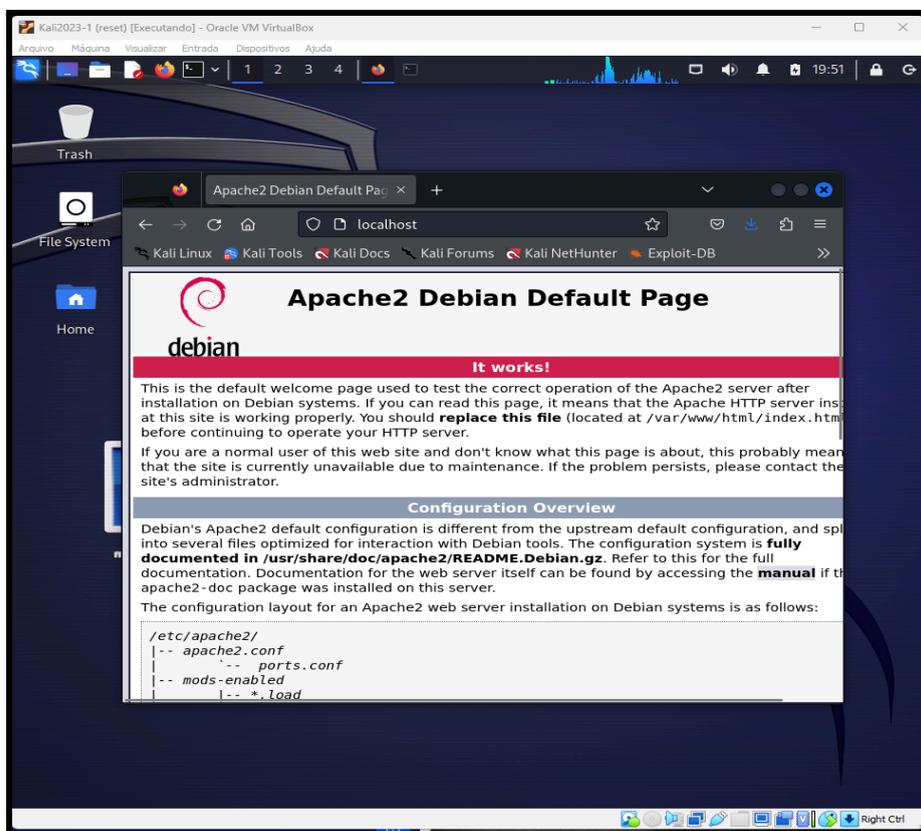
Permita o tráfego entrante HTTP e HTTPS para esse perfil:

```
sudo ufw allow in "Apache Full"
```

Você pode fazer uma verificação imediata para verificar se tudo correu como planejado visitando o endereço IP público do seu servidor no seu navegador web (Veja a nota abaixo do próximo cabeçalho para descobrir qual é o seu endereço IP público se você ainda não tiver essa informação):

```
http://ip_do_seu_servidor
```

Você verá a página web padrão do kali, que está lá para fins de teste e informação. Ela deve ser algo assim:



Se você ver esta página, então seu servidor web agora está corretamente instalado e acessível através do seu firewall.

## Passo 2 — Instalação do MySQL

Agora que temos nosso servidor web pronto e funcionando, é hora de instalar o MySQL. O MySQL é um sistema de gerenciamento de bancos de dados. Basicamente, ele irá organizar e fornecer acesso às bases de dados onde nosso site pode armazenar informação.

O próximo passo é instalar o MariaDB, para a gerência dos bancos de dados hospedados em seu Servidor Web. Para instalação adicionaremos os pacotes necessários conforme abaixo:

```
# apt install -y mariadb-server mariadb-client
```

Por padrão o pacote MariaDB no Debian usa `unix_socket` para autenticar o login do usuário, o que basicamente significa que você pode usar o nome de usuário e a senha do sistema operacional para efetuar login no console do MariaDB. Assim, você pode logar diretamente sem fornecer a senha root do MariaDB.

Uma vez instalado o MariaDB, é recomendado executar o seguinte script de segurança que removerá algumas configurações padrão inseguras e desabilitar o acesso ao seu sistema de banco de dados.

```
#mysql_secure_installation
```

Enter current password for root (enter for none): **ENTER**

OK, successfully used password, moving on...

Switch to `unix_socket` authentication [Y/n] **n**

... skipping.

Change the root password? [Y/n] **Y** ou **ENTER**

New password: **escom**

Re-enter new password: **escom**

Password updated successfully!

Reloading privilege tables..

... Success!

Remove anonymous users? [Y/n] Y ou **ENTER**

... Success!

Disallow root login remotely? [Y/n] Y ou **ENTER**.

... Success!

Remove test database and access to it? [Y/n] Y ou **ENTER**.

- Dropping test database...

... Success!

- Removing privileges on test database...

... Success!

Reload privilege tables now? [Y/n] Y ou **ENTER**.

... Success!

Cleaning up...

**All done! If you've completed all of the above steps, your MariaDB**

**installation should now be secure.**

**Thanks for using MariaDB!**

### **Passo 3 - Instalação do PHP**

O próximo recurso a ser instalado é o PHP. Serão instalados alguns pacotes adicionais, que poderão ser úteis, dependendo do contexto e da necessidade de utilização, pelo servidor web:

```
# apt install libapache2-mod-php php php-mysql php-cli php-pear php-curl  
php-xml php-zip
```

Para que o php tenha efeito sobre o Apache, reinicie o serviço com o seguinte comando:

```
# systemctl restart apache2.service
```

Você pode criar um arquivo php com a função phpinfo() para mostrar todas as informações sobre o seu PHP:

```
# echo '<?php phpinfo(); ?>' > /var/www/html/phpinfo.php
```

Depois acesse através do navegador do host  
http://192.168.10.100/phpinfo.php:

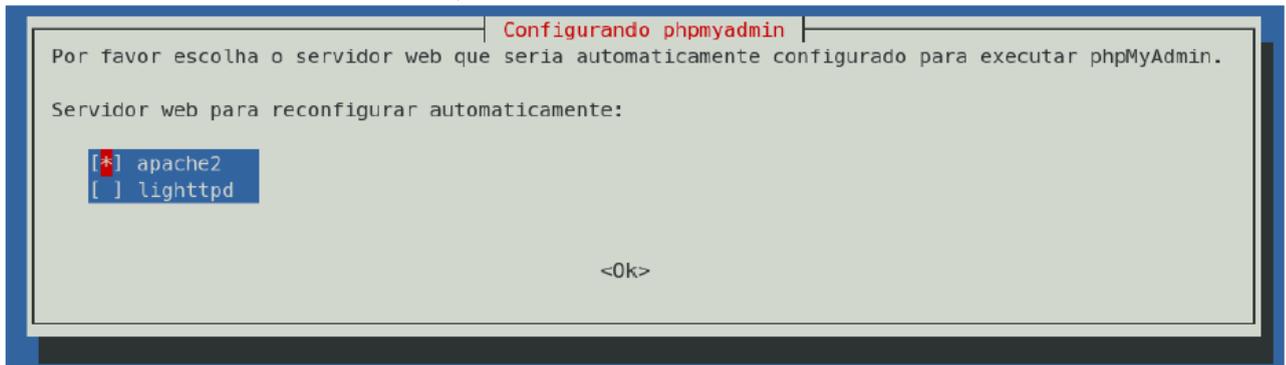
#### **Passo 4 - Instalação do PHPMYADMIN**

De volta aos repositórios do Debian, o phpMyAdmin pode ser instalado através do seguinte comando:

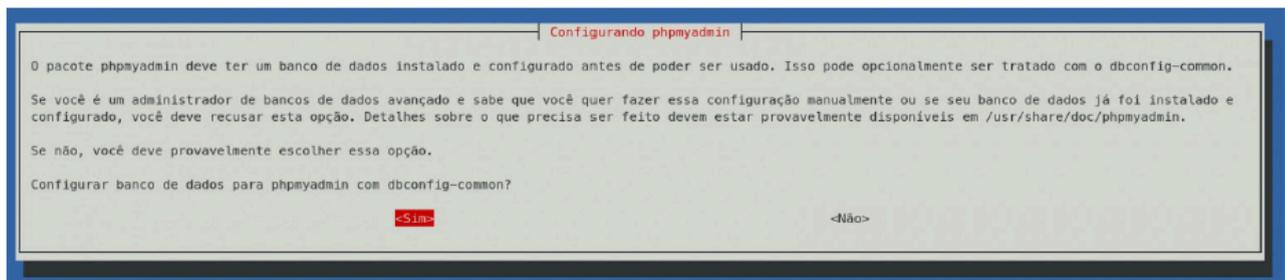
```
# apt install phpmyadmin
```

Utilizando a barra de espaço, selecione apache2, após tecla ENTER:

Utilizando a barra de espaço, selecione **apache2**, após tecla ENTER:



Na janela abaixo, tecla em **Sim**:



Quando solicitado, informe a senha para o banco de dados do phpMyAdmin, e aguarde finalizar.

Para acessar o phpMyAdmin, digite no seu navegador <http://192.168.100.10/phpmyadmin>. Para login no phpmyadmin o usuário é: **phpmyadmin** e senha: definida pelo operador. Ou pode logar pelo usuário root do mariadb. A tela será semelhante a esta:



Caso ocorra impossibilidade de acesso ao phpmyadmin (erro 404) adicione ao final do arquivo `/etc/apache2/apache2.conf` a linha abaixo:

**Include /etc/phpmyadmin/apache.conf**

## Observações da Instalação do PHP

O PHP é o componente da nossa configuração que irá processar código para exibir o conteúdo dinâmico. Ele pode executar script, conectar às nossas bases de dados MySQL para obter informações, e entregar o conteúdo processado para o nosso servidor web exibir.

Uma vez mais, aproveite o sistema apt para instalar o PHP. Adicionalmente, inclua alguns pacotes auxiliares dessa vez para que o código PHP possa rodar sob o servidor Apache e falar com o seu banco de dados MySQL:

```
sudo apt install php libapache2-mod-php php-mysql
```

Isto permitirá instalar o PHP sem problemas. Vamos testar isso em instantes.

Na maioria dos casos, vamos querer modificar a forma com a qual o Apache serve arquivos quando uma pasta é requisitada. Atualmente, se um usuário requisita uma pasta do servidor, o Apache irá olhar primeiramente para um arquivo chamado index.html. Queremos informar ao nosso servidor web para dar preferência aos arquivos PHP, então faremos o Apache olhar para um arquivo index.php primeiro.

Para fazer isto, digite este comando para abrir o arquivo **dir.inf** em um editor de texto com privilégios de root:

```
sudo nano /etc/apache2/mods-enabled/dir.conf
```

Ele terá esta aparência:

Arquivo /etc/apache2/mods-enabled/dir.conf

```
/etc/apache2/mods-enabled/dir.conf

<IfModule mod_dir.c>
  DirectoryIndex index.html index.cgi index.pl index.php index.xhtml index.htm
</IfModule>
```

Mova o arquivo de índice PHP (em destaque acima) para a primeira posição depois da especificação DirectoryIndex , como segue:

Arquivo /etc/apache2/mods-enabled/dir.conf

```
/etc/apache2/mods-enabled/dir.conf

<IfModule mod_dir.c>
  DirectoryIndex index.php index.html index.cgi index.pl index.xhtml index.htm
</IfModule>
```

Quando você tiver concluído, salve e feche o arquivo pressionando CTRL-X. Confirme a gravação digitando Y e em seguida pressione ENTER para confirmar a localização de salvamento do arquivo.

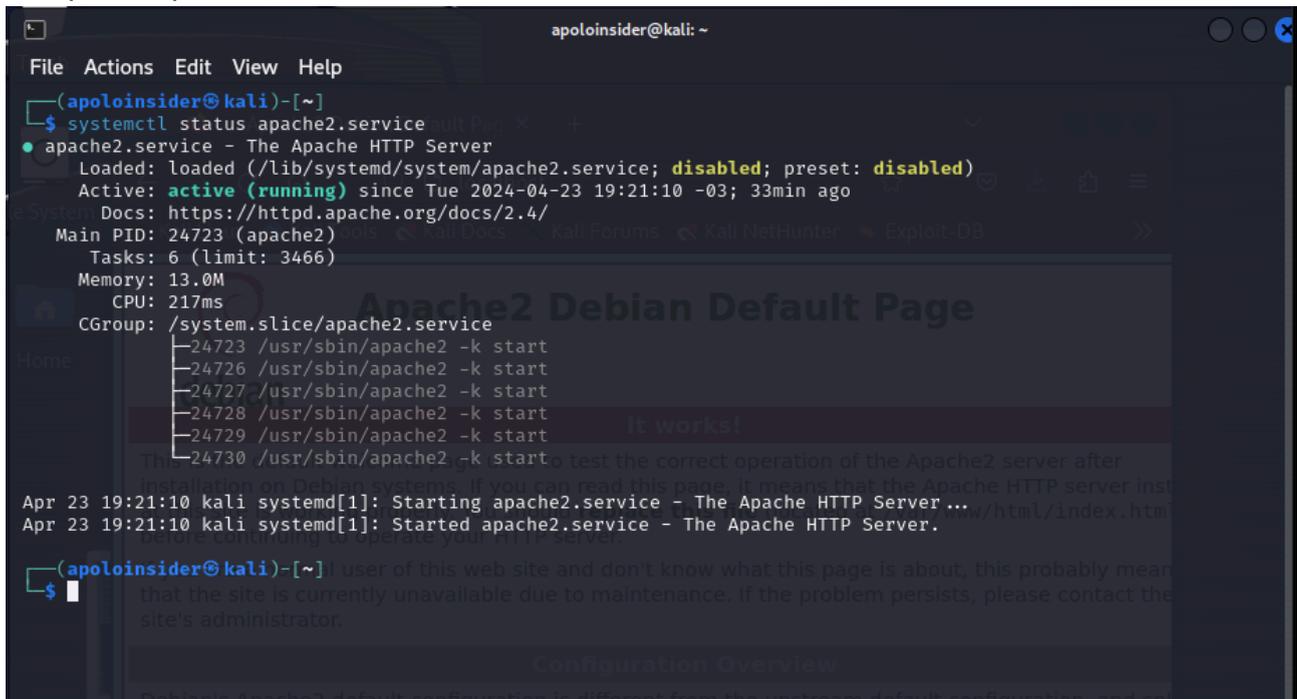
Após isso, reinicie o servidor web Apache de forma que nossas alterações sejam reconhecidas. Você pode fazer isso digitando o seguinte:

```
systemctl restart apache2.service
```

Você pode também verificar o status do serviço apache2 utilizando systemctl:

```
systemctl status apache2.service
```

Sample Output



```
apoloinsider@kali: ~
File Actions Edit View Help
(apoloinsider@kali)-[~]
└─$ systemctl status apache2.service
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; disabled; preset: disabled)
   Active: active (running) since Tue 2024-04-23 19:21:10 -03; 33min ago
     Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 24723 (apache2)
     Tasks: 6 (limit: 3466)
  Memory: 13.0M
     CPU: 217ms
   CGroup: /system.slice/apache2.service
           └─24723 /usr/sbin/apache2 -k start
             └─24726 /usr/sbin/apache2 -k start
               └─24727 /usr/sbin/apache2 -k start
                 └─24728 /usr/sbin/apache2 -k start
                   └─24729 /usr/sbin/apache2 -k start
                     └─24730 /usr/sbin/apache2 -k start

Apr 23 19:21:10 kali systemd[1]: Starting apache2.service - The Apache HTTP Server ...
Apr 23 19:21:10 kali systemd[1]: Started apache2.service - The Apache HTTP Server.

(apoloinsider@kali)-[~]
└─$
```

Para melhorar a funcionalidade do PHP, você tem a opção de instalar alguns módulos adicionais. Para ver as opções disponíveis para os módulos e bibliotecas PHP, direcione os resultados do comando apt search para o comando less, um paginador que lhe permite percorrer a saída de outros comandos:

```
apt search php- | less
```

Use as teclas de seta para rolar para cima e para baixo e Q para sair.

O resultado são todos os componentes opcionais que você pode instalar. Ele lhe dará uma breve descrição de cada um:

Para aprender mais sobre o que cada módulo faz, você pode pesquisar na internet para maiores informações sobre eles. Alternativamente, veja a descrição longa do pacote digitando:

```
apt show nome_do_pacote
```

Haverá uma grande quantidade de saída, com um campo chamado Description que terá uma explicação mais longa da funcionalidade que o módulo oferece.

Por exemplo, para encontrar o que o módulo php-cli faz, você pode digitar isto:

```
apt show php-cli
```

Se, após pesquisar, você decidir que gostaria de instalar um pacote, você pode fazê-lo utilizando o comando apt install assim como fizemos para nossos outros softwares.

Se decidir que o php-cli é algo que você precisa, você poderia digitar:

```
sudo apt install php-cli
```

Se você quiser instalar mais de um módulo, você pode fazer isso listando cada um, separado por um espaço, seguindo o comando apt install, como abaixo:

```
sudo apt install pacote1 pacote2 ...
```

Nesse ponto, sua pilha **LAMP** está instalada e **configurada**. Antes de fazer mais alterações ou implantar um aplicativo, seria útil testar proativamente sua configuração do PHP, para o caso de haver algum problema que deva ser resolvido.

## Testando o Processamento PHP no seu Servidor Web

A fim de testar se seu sistema está corretamente configurado para o PHP, crie um script PHP bem básico denominado info.php. Para que o Apache possa encontrar o arquivo e servi-lo corretamente, ele deve ser salvo em um diretório muito específico, o qual é chamado de "web root".

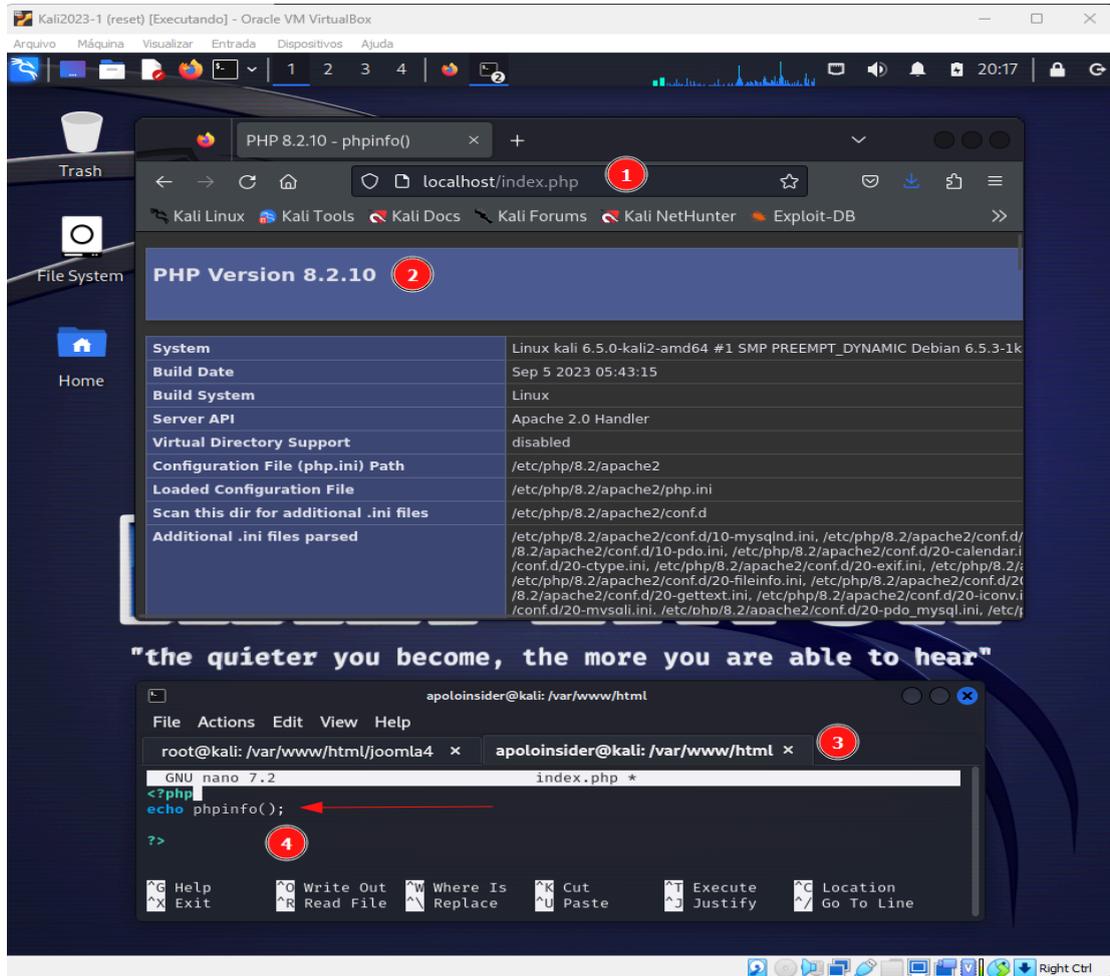
```
info.php
<?php
phpinfo();
?>
```

Quando você tiver concluído, salve e feche o arquivo.

Agora você pode testar se seu servidor web pode exibir corretamente o conteúdo gerado por esse script PHP. Para testar isso, visite esta página em seu navegador web. Você vai precisar novamente do endereço IP público do seu servidor.

O endereço que você vai querer visitar é:

[http://ip\\_do\\_seu\\_servidor/info.php](http://ip_do_seu_servidor/info.php)



## Resumo do processo de instalação - LAMP



### 1) Duas coisas são necessárias:

#### a) Preciso de um pacote LAMP

sistema operacional Linux,  
com o servidor web **A**pache.

A informação do site é armazenada em uma base de dados **M**ySQL,  
O conteúdo dinâmico é processado pelo **P**HP.

b) Um conta de usuário que não seja o root para fazer as instalações

## 2) Passos da instalação dos pacotes

a) Passo 1 — Instalação do Apache e Atualização do Firewall

```
sudo apt update  
sudo apt install apache2
```

b) Passo 2 — Instalação do MySQL

```
sudo apt install mysql-server
```

c) Passo 3— Instalação do PHP e PHPMYADMIN

```
sudo apt install php libapache2-mod-php php-mysql
```

.  
sudo apt install phpmyadmin