

Guia de instalação Ubuntu

**Apache
MariaDB
PHP 7.2**

Atualizando o Ubuntu

```
sudo apt-get update  
sudo apt-get upgrade
```

```
sudo apt-get dist-upgrade
```

Instalando o Servidor Apache

```
sudo apt-get install apache2 apache2-utils
```

Depois de instalado o Apache deve ser iniciado automaticamente. Confira seu status com o comando `systemctl`.

```
systemctl status apache2
```

Se não estiver em execução, use o comando `systemctl start` para iniciá-lo:

```
sudo systemctl start apache2
```

Para definir que o processo se inicie automaticamente você pode usar o comando:

```
sudo systemctl enable apache2
```

Para verificar a versão do Apache:

```
apache2 -v
```

Instalar o Curl

```
sudo apt-get install curl
```

Instalar o MariaDB

```
sudo apt-get install mariadb-server mariadb-client
```

Nota: para instalar o MySQL o comando seria:

```
sudo apt-get install mysql-server mysql-client
```

Para verificar o status do MariaDB você pode usar o comando:

```
systemctl status mysql
```

Se não estiver em execução, você pode iniciar o serviço com o comando:

```
sudo systemctl start mysql
```

Para ativar o mariaDB automaticamente quando o Ubuntu for iniciado:

```
sudo systemctl enable mysql
```

Execute o script de segurança pós-instalação:

```
sudo mysql_secure_installation
```

Quando pedir que você digite a senha do root do MariaDB, pressione enter porque você ainda não definiu a senha do root.

Em seguida, digite **y** para definir a senha de root para o servidor MariaDB.

Em seguida, basta pressionar Enter para responder a todas as perguntas restantes. Isso removerá o usuário anônimo, desativará o login de root remoto e removerá o banco de dados de teste. Esta etapa é um requisito básico para a segurança do banco de dados MariaDB.

Você precisará criar um usuario para o MariaDB para evitar erros na hora de fazer a conexao com o banco de dados.

Para entrar no banco de dados MariaDB utilize o comando:

```
sudo mysql -u root -p
```

digite a senha definida para o root e então você acessará o MariaDB. Em seguida digite os comandos:

```
create user 'admin'@'%' IDENTIFIED BY 'workspace';
```

```
GRANT ALL PRIVILEGES ON *.* TO 'admin'@'%' WITH GRANT OPTION;
```

Nesse exemplo, foi criado um usuario 'admin'.

Instalando o PHP 7.2

```
sudo apt-get install php7.2-cli libapache2-mod-php7.2 php7.2-mysql php7.2-curl  
php-memcached php7.2-dev php7.2-sqlite3 php7.2-mbstring php7.2-gd php7.2-json
```

para verificar se deu tudo certo:

```
php -v
```

para verificar todos os pacotes disponiveis para PHP 7.2:

```
sudo apt-cache search php7.2
```

Dica de segurança

Vamos fazer uma pequena alteração na configuração do PHP para tornar nossa configuração mais segura.

Abra o arquivo `/etc/php/7.2/apache2/php.ini` com o nano e procure pela linha (ctrl + w) **`cgi.fix_pathinfo`**, ela está comentada por `;` e com valor setado para 1, descomente a linha e defina o valor para zero.

```
Sudo nano /etc/php/7.2/apache2/php.ini
```

```
cgi.fix_pathinfo=0
```

Esta é uma configuração previne que o PHP tente executar o arquivo mais PHP próximo se o arquivo solicitado não puder ser encontrado. Isso basicamente permitiria aos usuários elaborar pedidos PHP de uma forma que permitisse executar scripts que não deveriam ser autorizados a executar. Tudo certo, reinicie o Apache com o comando abaixo

```
sudo systemctl restart apache2
```

Instalar o Xdebug

Para instalar o xdebug basta usar o comando:

```
sudo apt-get install php-xdebug
```

Instalar o GIT

para instalar o git basta usar o comando:

```
sudo apt-get install git
```

Instalar o Unzip

para instalar o unzip para descompactar pacotes use o comando:

```
sudo apt-get install unzip
```

Instalar o composer

Use o comando:

```
sudo apt-get install composer
```

Para instalar o `composer` globalmente, utilize o seguinte comando que irá baixar e instalar o Composer como um comando disponível para todo o sistema chamado `composer`, dentro de `/usr/local/bin`:

```
sudo php composer-setup.php --install-dir=/usr/local/bin --filename=composer
```

Instalar o PhpmyAdmin

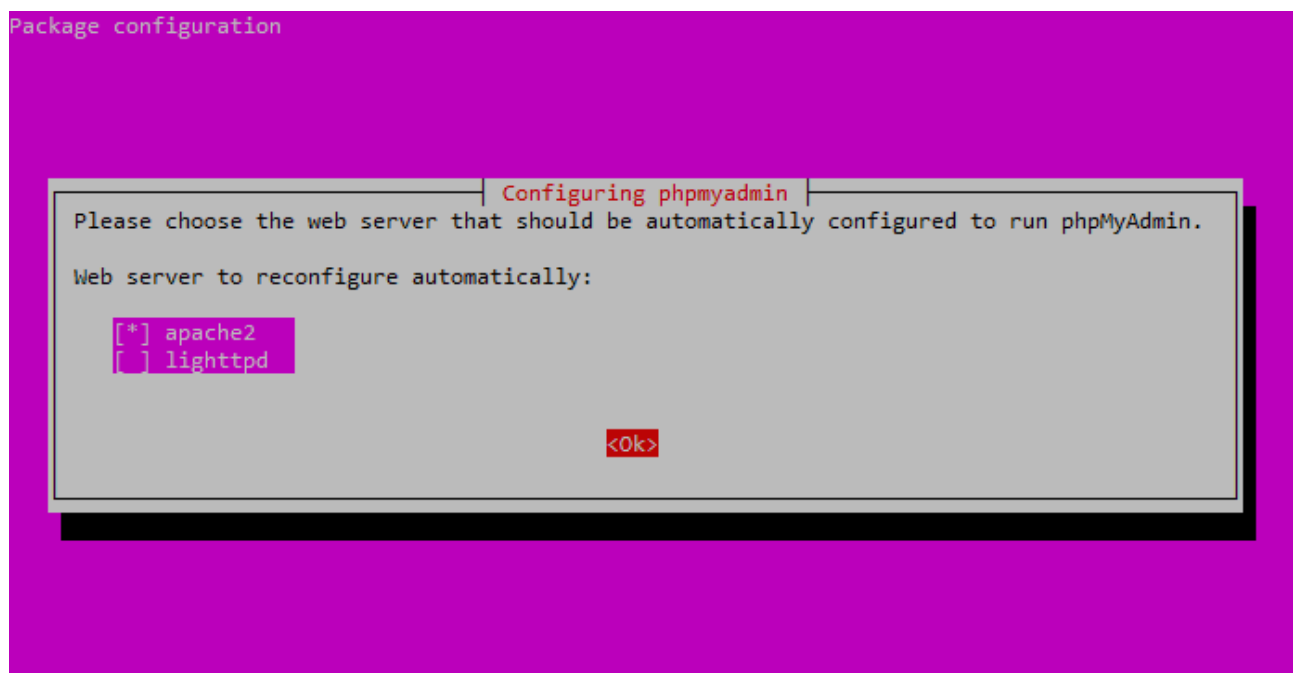
Certifique-se que você possui os pacotes `php-mbstring` e `php-gettext` instalados. Caso não os tenha instale com o comando:

```
sudo apt-get install php-mbstring php-gettext
```

depois disso você deverá instalar o `phpMyAdmin` no sistema com o comando:

```
sudo apt-get install phpmyadmin -y
```

Aparecerão algumas telas de configuração do ambiente do `phpMyAdmin`:



Selecione a opção `apache2`

Package configuration

Configuring phpmyadmin

The phpmyadmin package must have a database installed and configured before it can be used. This can be optionally handled with dbconfig-common.

If you are an advanced database administrator and know that you want to perform this configuration manually, or if your database has already been installed and configured, you should refuse this option. Details on what needs to be done should most likely be provided in /usr/share/doc/phpmyadmin.

Otherwise, you should probably choose this option.

Configure database for phpmyadmin with dbconfig-common?

<Yes>

<No>

Escolha "yes" para a configuração ser feita agora no terminal.

Package configuration

Configuring phpmyadmin

Please provide the password for the administrative account with which this package should create its MySQL database and user.

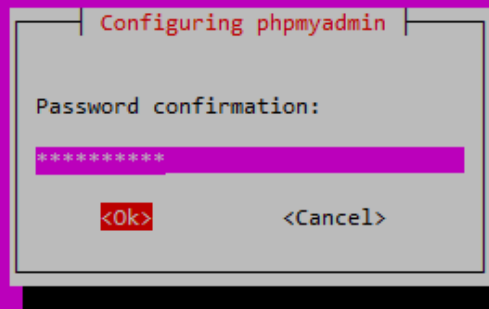
Password of the database's administrative user:

<Ok>

<Cancel>

Informe a senha de administrador do banco de dados.

Package configuration



Confirme a senha.

Depois disso, faça o download do PhpMyAdmin no site <https://www.phpmyadmin.net/>. Extraia o arquivo e renomeie o nome da pasta caso ache necessário, por exemplo. 'Phpmyadmin' em caixa baixa.

O comando `ls` lista os diretórios do sistema. Você pode usar o comando `cd` para mudar para a pasta Downloads. Digite o seguinte comando para mover a pasta do PhpMyAdmin:

```
sudo mv phpmyadmin/ /var/www/html/
```

O comando `mv` é usado para mover arquivos e diretórios.

Reinicie o servidor Apache com o comando:

```
sudo /etc/init.d/apache2 restart
```

ou

```
sudo systemctl restart apache2
```

ou

```
sudo service apache2 restart
```

Você não conseguirá usar o phpMyAdmin com o usuário `root`, por isso, você deve criar um novo usuário como demonstrado na instalação do MariaDB:

```
sudo mysql -u root -p
```

Uma vez logado no MariaDB, se necessário for, você deverá conceder as permissões necessárias para o usuário `root`:

```
GRANT ALL PRIVILEGES ON *.* TO 'phpmyadmin'@'localhost';
```

```
FLUSH PRIVILEGES;
```

ou criar um usuário novo:

```
CREATE USER USERNAME IDENTIFIED by 'PASSWORD';
```

```
GRANT ALL PRIVILEGES ON *.* TO 'username'@'localhost';  
FLUSH PRIVILEGES;
```

Referência

<https://matheuslima.com.br/instalando-o-apache-php-72-mysql-lamp>

<https://sempreupdate.com.br/como-instalar-apache-mariadb-php7-lamp-stack-ubuntu-16-04-lts/>

mysqli_real_connect() (HY0001698) Access denied for user 'root'@'localhost'

<https://www.youtube.com/watch?v=iqx6AqqCgCI>