How To Install and Secure phpMyAdmin on Debian 7

*Authored by: **ASPHostServer Administrator** [asphostserver@gmail.com]*
Saved From: http://faq.asphosthelpdesk.com/article.php?id=198

phpMyAdmin

phpMyAdmin is a free web software to work with MySQL on the web. It provides a convenient visual front end to the MySQL capabilities.

Setup

Before working with phpMyAdmin, you need to have LAMP installed on your server. If you don't have the Linux, Apache, MySQL, PHP stack on your server, you can find the tutorial for setting it up here.

Once you have the user and required software, you can start installing phpMyAdmin on your server!

Install phpMyAdmin

The easiest way to install phpmyadmin is through apt-get:

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

During the installation, phpMyAdmin will walk you through a basic configuration. Once the process starts up, follow these steps:

- Select Apache2 for the server

- Choose YES when asked about whether to Configure the database for phpmyadmin with dbconfig-common

- Enter your MySQL password when prompted

- Enter the password that you want to use to log into phpmyadmin

After the installation has completed, add phpmyadmin to the apache configuration.

```
sudo nano /etc/apache2/apache2.conf
```

Add the phpmyadmin config to the file.

```
Include /etc/phpmyadmin/apache.conf
```

Restart apache:

```
sudo service apache2 restart
```

You can then access phpmyadmin by going to youripaddress/phpmyadmin. The screen should look like this.

Security

Unfortunately, older versions of phpMyAdmin have had serious security vulnerabilities, including allowing remote users to eventually exploit root on the underlying virtual private server. One can prevent a majority of these attacks through a simple process: locking down the entire directory with Apache's native user/password restrictions which will prevent these remote users from even attempting to exploit older versions of phpMyAdmin.

Set Up the .htaccess File

To set this up, start off by allowing the .htaccess file to work within the phpmyadmin directory. You can accomplish this in the phpmyadmin configuration file:

```
sudo nano /etc/phpmyadmin/apache.conf
```

Under the directory section, add the line "AllowOverride All" under "Directory Index", making the section look like this:

```
<Directory /usr/share/phpmyadmin>
        Options FollowSymLinks
        DirectoryIndex index.php
        AllowOverride All
        [...]
```

Configure the .htaccess file

With the .htaccess file allowed, we can proceed to set up a native user whose login would be required to even access the phpmyadmin login page.

Start by creating the .htaccess page in the phpmyadmin directory:

```
sudo nano /usr/share/phpmyadmin/.htaccess
```

Follow up by setting up the user authorization within .htaccess file. Copy and paste the following text in:

```
AuthType Basic
AuthName "Restricted Files"
AuthUserFile /path/to/passwords/.htpasswd
Require valid-user
```

Below you"ll see a quick explanation of each line:

AuthType: This refers to the type of authentication that will be used to the check the passwords. The passwords are checked via HTTP and the keyword Basic should not be changed.
AuthName: This is text that will be displayed at the password prompt. You can put anything here.
AuthUserFile: This line designates the server path to the password file (which we will create in the next step.)
Require valid-user: This line tells the .htaccess file that only users defined in the password file can access the phpMyAdmin login screen.

Create the htpasswd file

Now we will go ahead and create the valid user information. Start by creating a htpasswd file. Use the htpasswd command, and place the file in a directory of your choice as long as it is not accessible from a browser. Although you can name the password file whatever you prefer, the convention is to name it .htpasswd.

```
sudo htpasswd -c  /path/to/passwords/.htpasswd username
```

A prompt will ask you to provide and confirm your password. Once the username and passwords pair are saved you can see that the password is encrypted in the file.

Finish up by restarting apache:

```
sudo service apache2 restart
```

Accessing phpMyAdmin

phpMyAdmin will now be much more secure since only authorized users will be able to reach the login page. Accessing youripaddress/phpmyadmin should display a screen like this.

Fill it in with the username and password that you generated. After you login you can access phpmyadmin with the MySQL username and password.