

Configure Ubuntu Server for Web Server

SSH

1. Installation

The Ubuntu servers have the SSH server installed by default; if it is not installed you can install it using the following command:

```
sudo apt-get install ssh
```

2. Configuration

To access a SSH server, we must have a SSH client; but before installing the SSH client we have to configure the SSH server. The SSH server configuration file is in the /etc/ssh directory, to open and edit it we must type:

```
sudo nano /etc/ssh/ssh_config
```

When the file is open, we have a list of the configurations we can make. The first one is the port number. SSH has a default port number set to 22; for security reasons it is recommended that you change this port number. The reasons why the port is recommended to be changed is that hackers or anyone who may try to brute-force the SSH server would not be able to find the port in which the SSH runs. The port changing is recommended but not necessary, if you wish to change it instead of number 22 write a number that is bigger than 1024 (ports 0 to 1023 are system reserved).

The other configurations that you must do are to change the followings to the values described in the list below:

- PermitRootLogin no
- X11Forwarding no
- UsePAM no
- UseDNS no
- AllowUsers admin

Close the editor by using CTRL+X and save it. Now the SSH server is ready.

3. SSH Client

In the client machine (outside the server), install a SSH client. There are many SSH clients, some of them are:

- For Windows:
 - SSH Secure Shell 3.2.9 : <https://shareware.unc.edu/>
 - Putty : <http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html>

- For Linux:
 - Putty : *sudo apt-get install putty*
- For MacOS:
 - iTerm2 : <http://iterm2.com/downloads.html>

To access the SSH server we have to know the Server's IP address; to find the IP address of the server, write this command to the terminal of the server:

ifconfig

If all the numbers and letters are confusing to you, you may filter the ifconfig using this command:

ifconfig eth0 | grep inet (for ethernet connection)

ifconfig wlan0 | grep inet (for wifi connection)

Use the IP address found in the server and the port number (default is 22 if you did not change it) and connect to the SSH server.

LAMP (Linux Apache MySQL Php)

The LAMP is the collection of open source softwares to configure and run web servers. You may find all these softwares in one package (such as XAMPP for Windows or LAMPP for Linux), but here we will install and configure each software one by one; you may either install them within the server or install them using the SSH.

1. Apache

Apache is an open-source Web Server that is freely available. To install apache first let us check if our Ubuntu Server has the updated packages by running:

sudo apt-get update

After the packages are updated we are able to install the apache web server by running this command:

sudo apt-get install apache2

When the installation is finished we are ready to use apache2, but first we have to install and configure the other packages. To check if the apache is installed, open a browser on the client machine and write the IP address of the server, if a page with Apache informations is displayed it means that Apache has been installed.

2. MySQL

MySQL is a database management system. To install MySQL write the following command:

```
sudo apt-get install mysql-server libapache2-mod-auth-mysql php5-mysql
```

While being installed, MySQL will ask you to set a root password, you may leave it blank or set a password, but it can also be changed later. When the installation is finished, we have to activate MySQL; to do this write the following command:

```
sudo mysql_install_db
```

After this install is finished, run the MySQL set up script from the:

```
sudo /usr/bin/mysql_secure_installation
```

After running this, the prompt will ask you for the MySQL password; after you provide your password it will ask you if you want to change it, you may write either Y or N. After these steps the next steps can be all Y (yes). Now the MySQL is ready for the web server.

To install phpmyadmin, type the following:

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

When asked to select the server, select apache by pressing SPACE (make sure an asterisk is put on the selection), press TAB to select OK and hit ENTER.

If you want to secure phpmyadmin, type:

```
sudo php5enmod mcrypt
```

Restart apache and phpmyadmin is ready:

```
sudo service apache2 restart
```

3. PHP

PHP is an open source, object oriented web scripting language that can be used to build dynamic webpages. To install PHP run the following command:

```
sudo apt-get install php5 libapache2-mod-php5 php5-mcrypt
```

After the PHP is installed, it is recommended that php be added to the directory index, so the web server may recognize index.php as index; to do so edit the file dir.conf:

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

In the *DirectoryIndex* add *index.php*. The file contents should look like this:

```
<IfModule mod_dir.c>
```

```
DirectoryIndex index.php index.html index.cgi index.pl index.php index.xhtml index.htm
```

```
</IfModule>
```

Now the PHP is ready to be used for our web applications. There are many modules that can be installed to the PHP, modules are useful libraries; to see the available modules for the php you can type:

```
apt-cache search php5-
```

This command will display a list of php5 modules that are available to be installed. When you find a module that you want to install, write:

```
sudo apt-get install name_of_the_module
```

Note: php5-mysql module is installed when we installed the MySQL .

4. Testing Web Server

Apache has a default folder for storing the websites, it is in /var/www/html. Websites may also be configured to have different directories outside the default Apache directory, but for now let us create a simple webpage in the www directory. Create and edit a php file inside the /var/www/html (nano command creates a file if it does not exist):

```
sudo nano /var/www/html/info.php
```

When the editor is opened, we can write php scripts here; let us write the phpinfo() function that will display us all the php informations. The file should look like this:

```
<?php
```

```
phpinfo();
```

```
?>
```

Close and Save the editor. Now that we created a simple webpage, we have to test it. Open a browser on the client machine (outside the server) and write the servers ip address followed by “/info.php” (for example: <http://192.168.10.7/info.php>). This will open the info.php file that we created on the server.

You can open directories and write many files in the /var/www/html/ folder to create websites.

FTP (Optional)

FTP Server allows other FTP clients to access the server and transfer files. A commonly used FTP client is FileZilla, and the FTP server that will be installed in the Ubuntu server is vsftpd.

1. Installing FTP

To install FTP run the following command:

```
sudo apt-get install vsftpd
```

2. Configuring FTP

After the FTP is installed, we have to configure it. The configuration file can be edited as described below:

```
sudo nano /etc/vsftpd.conf
```

By default the vsftpd allows anonymous connections, which means that anyone without credentials can access the file system by a ftp client. To change this we have to change the *anonymous_enable=YES* to *anonymous_enable=NO*.

The other configurations that should be changed are: uncommenting *local_enable=YES* (comments begin with # sign) and allowing the user to write to the directories: *write_enabled=YES*.

The last configuration is to make the users work only with their own permitted files and not be able to access the other files. To do so uncomment the *chroot_local_user=YES*. Save and Exit the editor.

Restart the vsftpd service:

```
sudo service vsftpd restart
```

3. FTP Client

To access the FTP Server, the client must have a FTP Client such as FileZilla; providing the IP address of the server, username and password, a client can access the file system of the server.