CENTRE ESPECÍFIC
D´EDUCACIÓ A DISTÀNCIA DE
LA COMUNITAT VALENCIANA

UNIT 2.
ACTIVITY

Web Applications Deployment

CFGS DAW

Important: this activity is not mandatory and does not compute for the final grade, but it is necessary for coming activities.

Importante: esta actividad no es obligatoria y no cuenta para la nota final, pero es necesaria para actividades futuras.

Author: Carlos Cacho López

Reviewed by: Lionel Tarazón Alcocer
lionel.tarazon@ceedcv.es

2019/2020

License

Nomenclature

During this unit we are going to use special symbols to distinct some important elements. This symbols are:

Important

Attention

Interesting

INDEX

U02. SERVICES INVOLVED IN WEB DEPLOYMENT ACTIVITY

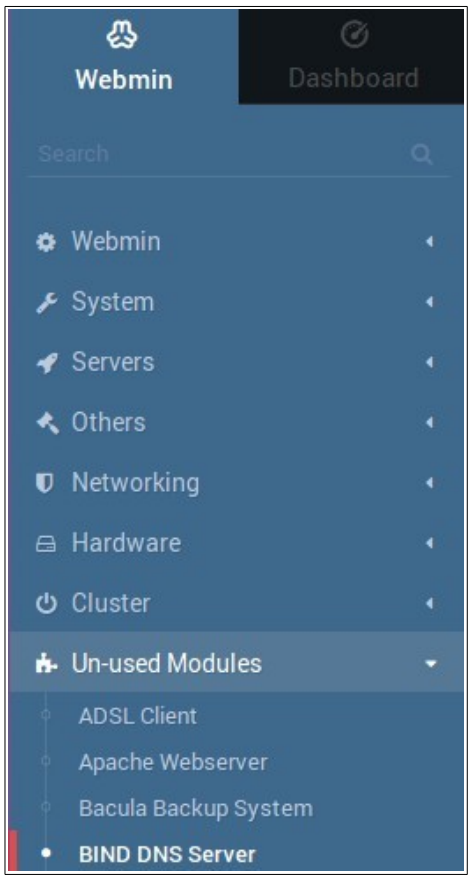
1. INTRODUCTION

In this activity we will install and configure the three services seen in the theory: DNS server, FTP server and SSH server. We will do so in the LinuxServer machine using Webmin => http://192.168.0.2:10000

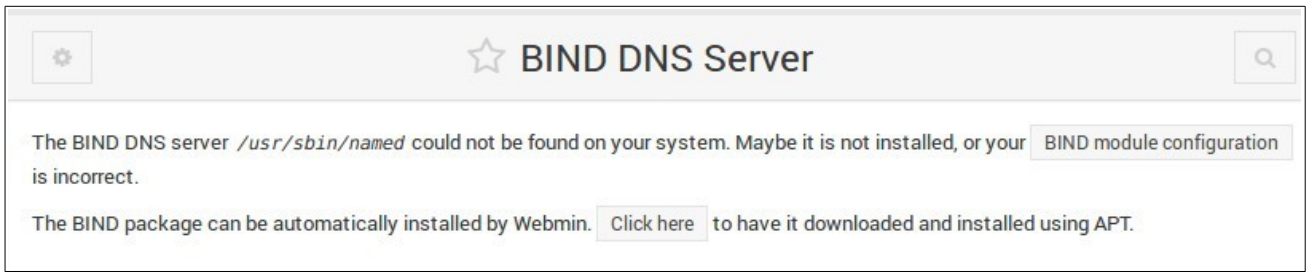
2. DNS SERVER

2.1 Installing

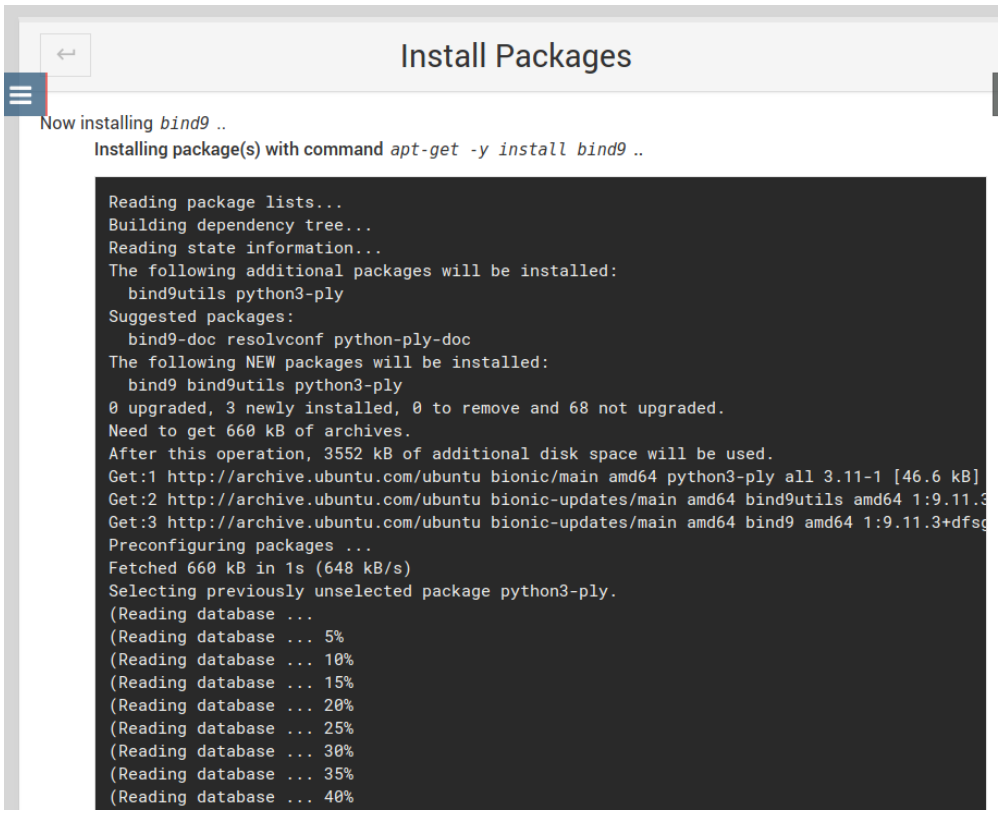
First of all, we have to install the ***Bind*** packet. For that, we have to go to the group “Un-used Modules” and select *Bind DNS Server*.



As we do not have it installed in our system Webmin shows a warning.

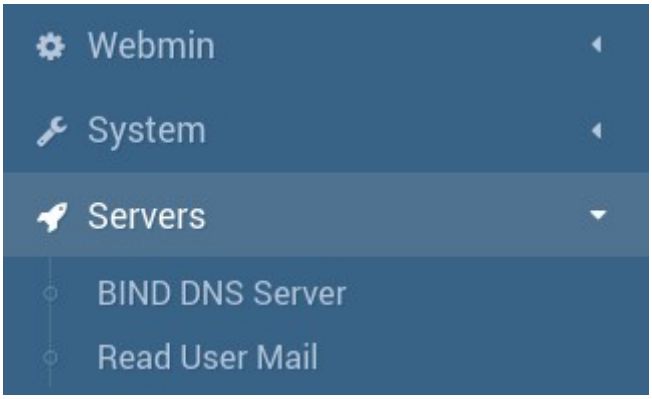


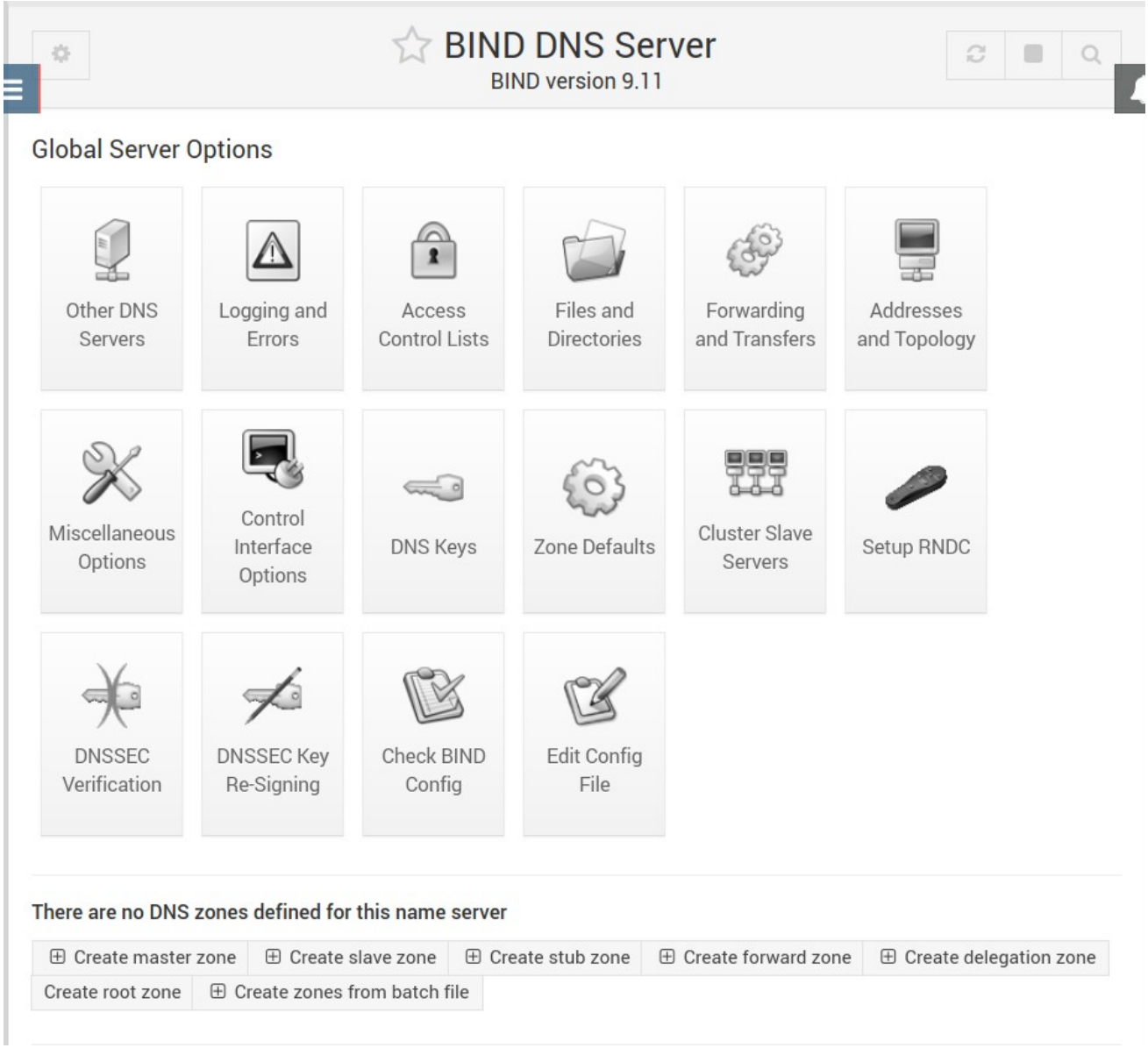
Click below on ***Click here*** to install the packages and follow the steps. Once installed, it will appear something like this and a message “install complete” at the bottom.



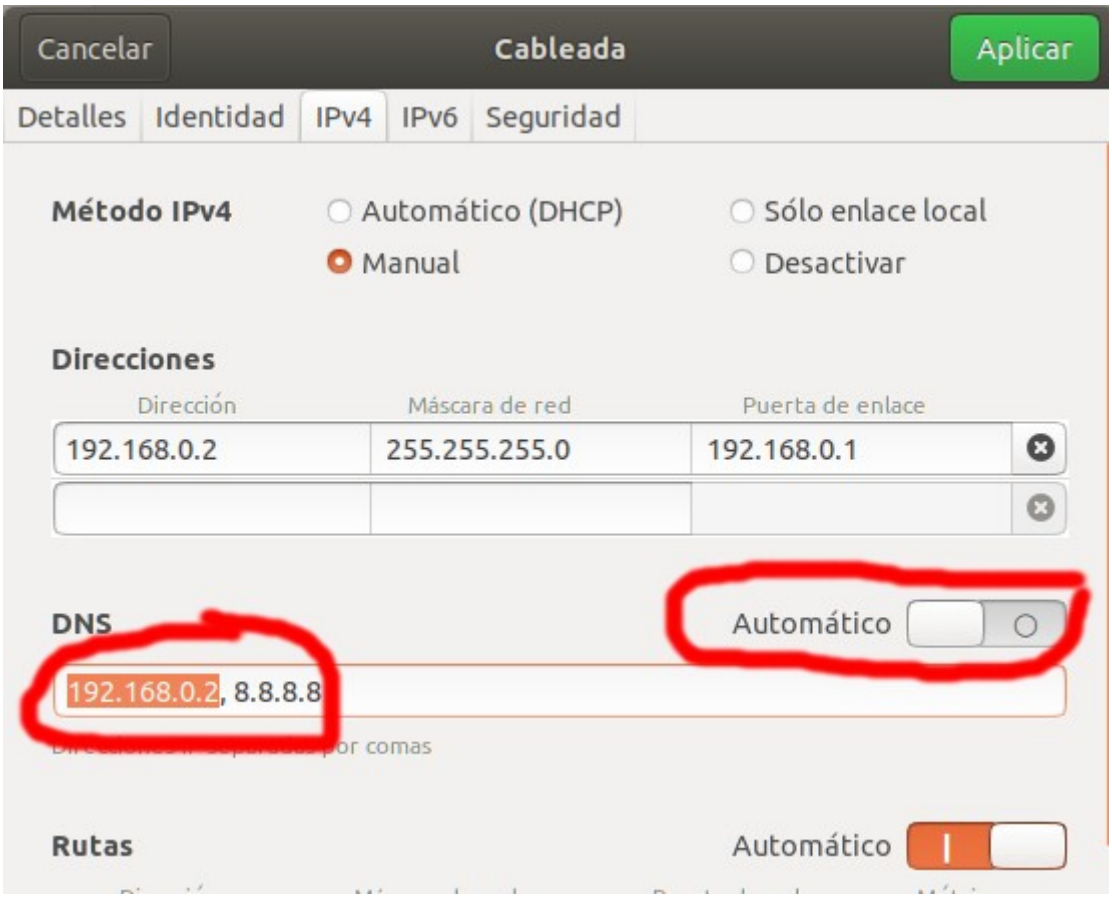
Then click the **Webmin menu** and **Refresh modules**.

From now on the **BIND DNS Server** module will appear in the **Servers** group.





First, we want our LinuxServer machine to use the newly installed DNS server. To do so, go to the **Network Settings** (Configuración → Red) and in the **IPv4 tab,** set the DNS server to the same IP as the LinuxServer (192.168.0.2). Make sure to **disable the Automatic DNS option**. It's also a good idea to add a secondary DNS server (for example 8.8.8.8, it's google's public DNS server).



Then restart the ***network*** to apply the changes (disable it and then enable it).

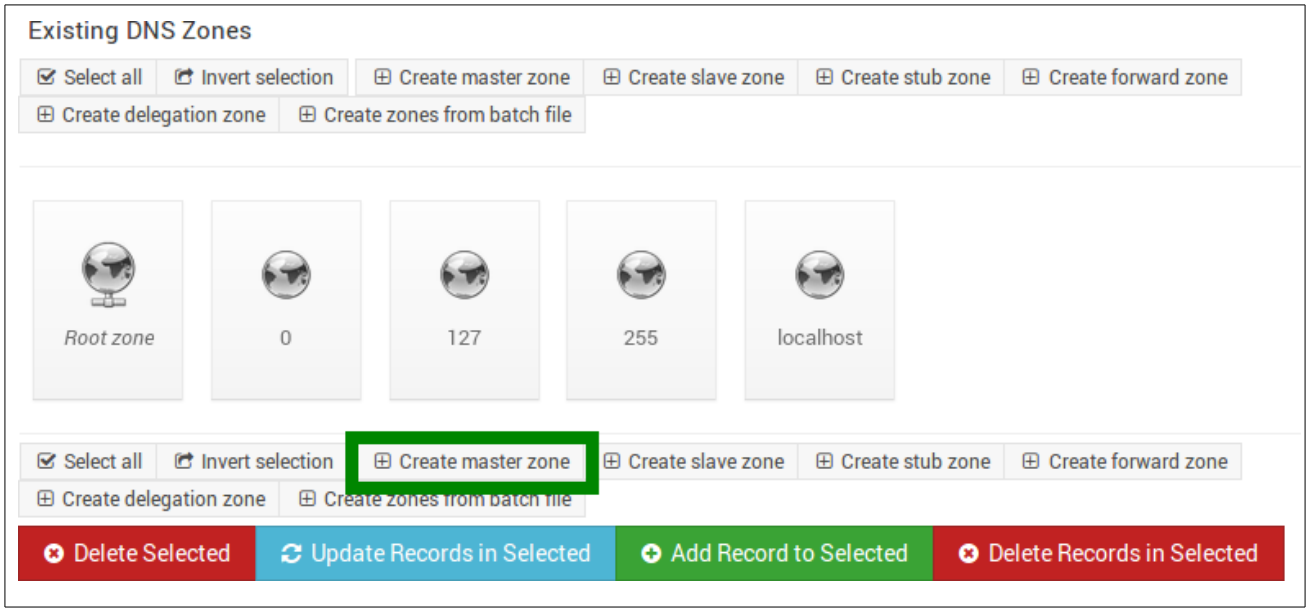
Then go back to Webmin BIND DNS Server and make sure the DNS server is on. If it's off, click on the icon shown below.

Now we are to configure both lookup zones: forward and reverse.

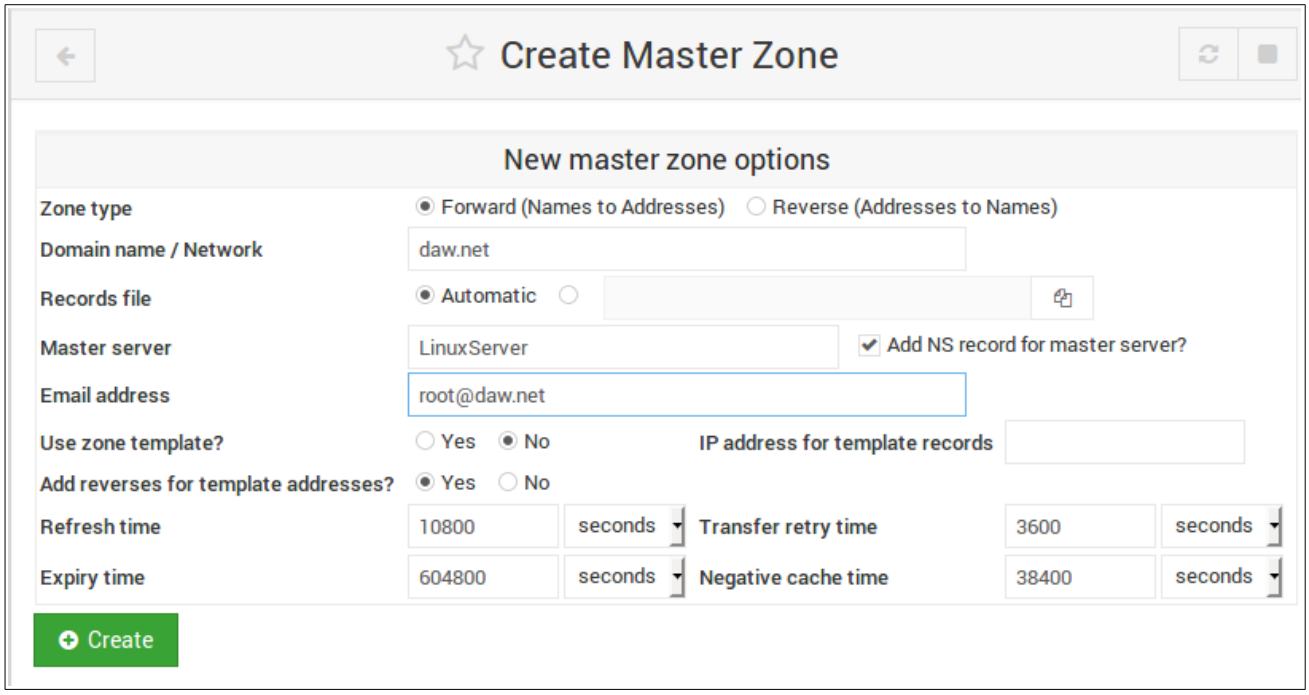
You can get all the information in the official documentation: https://doxfer.webmin.com/Webmin/BIND_DNS_Server

2.2 Configuring Lookup zone forward

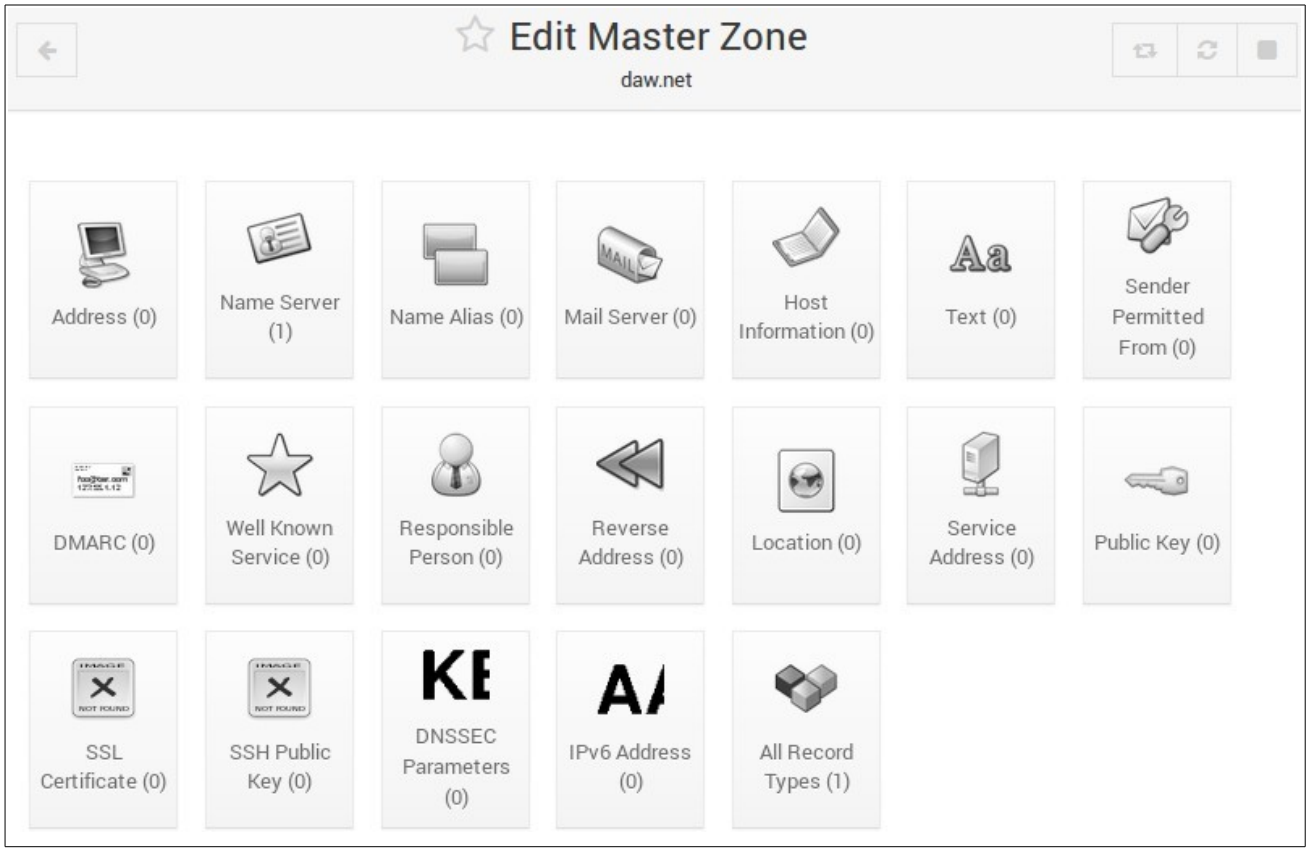
We will start with the forward lookup zone. Its name will be *daw.net* and will be a master zone. For that we have to click on ***Create master zone*** button in the ***Existing DNS Zones*** section.



Now, we have to fill the master zone options, we will write the domain name (*daw.net*), check that the master server is *LinuxServer* (our virtual machine), write the email address (for instance, root@daw.net) and click on the ***Create*** button.

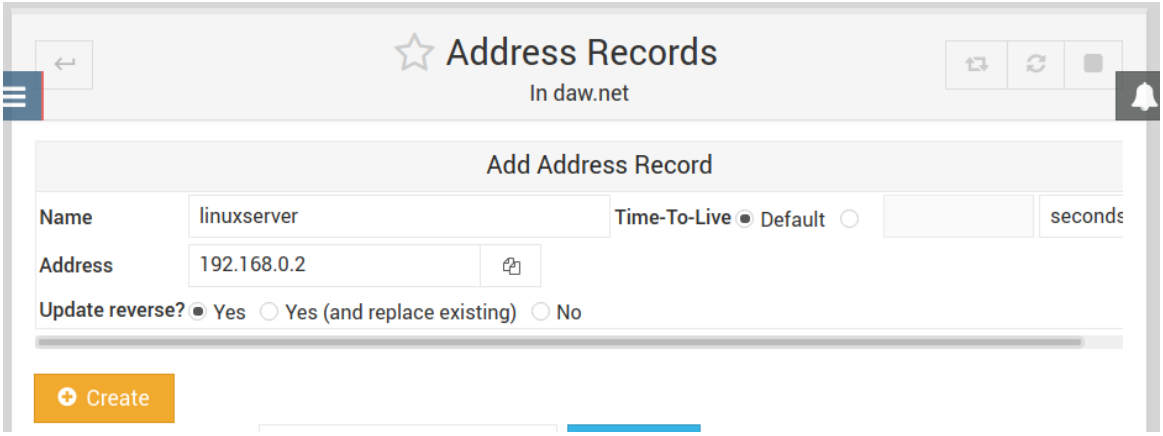


Once the master zone is created, we can edit it to create A (Address) and NS (Name Server) records.



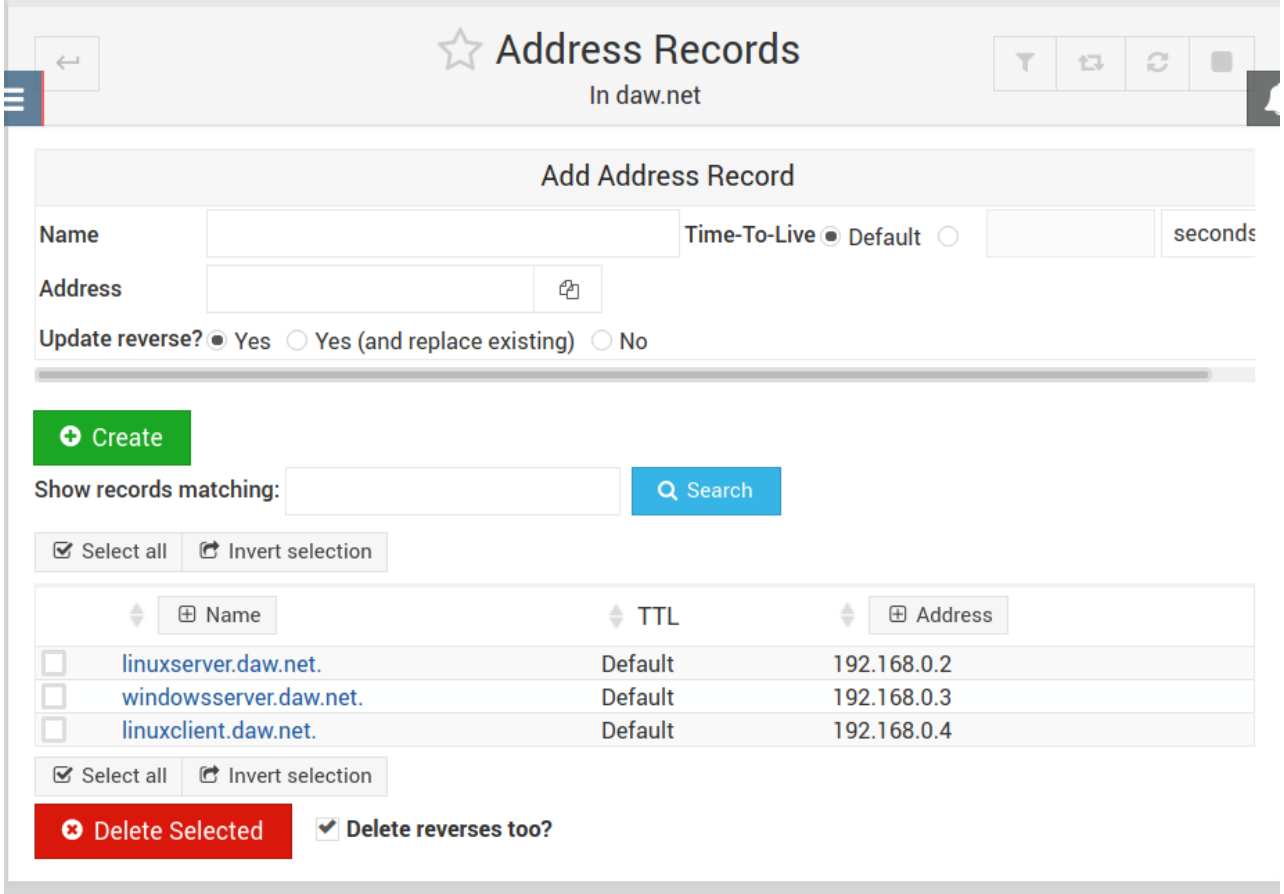
First we are going to create the Address Records of every virtual machine.

To do so we have to click ***Address*** and type the machine name and IP address. Also we can check Yes in ***Update reverse?*** to create the reverse address in the reverse zone. For instance, this will be the form filled for *LinuxServer*:



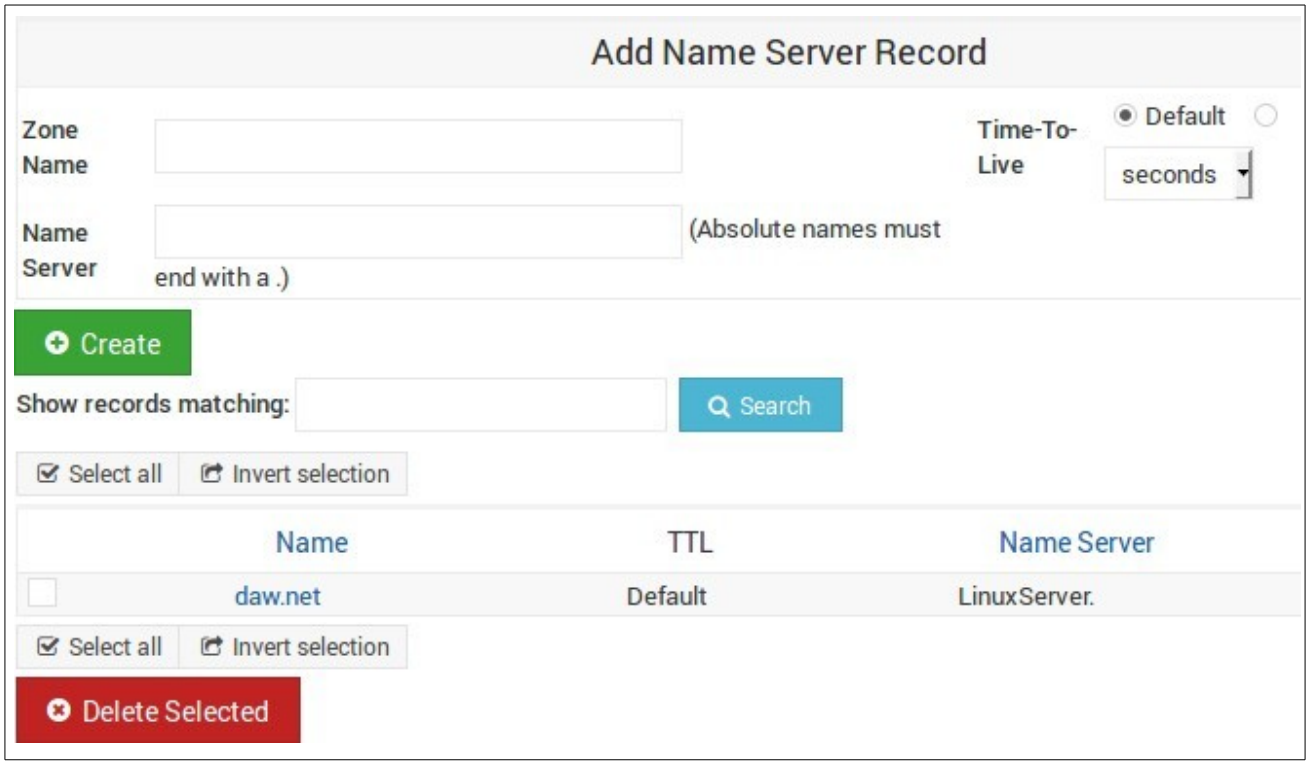
> 📣 Make sure to **use the IP addresses in your network**, they might be different to those shown in this Activity.

Now create Address Records for **LinuxClient** and **WindowsServer**.



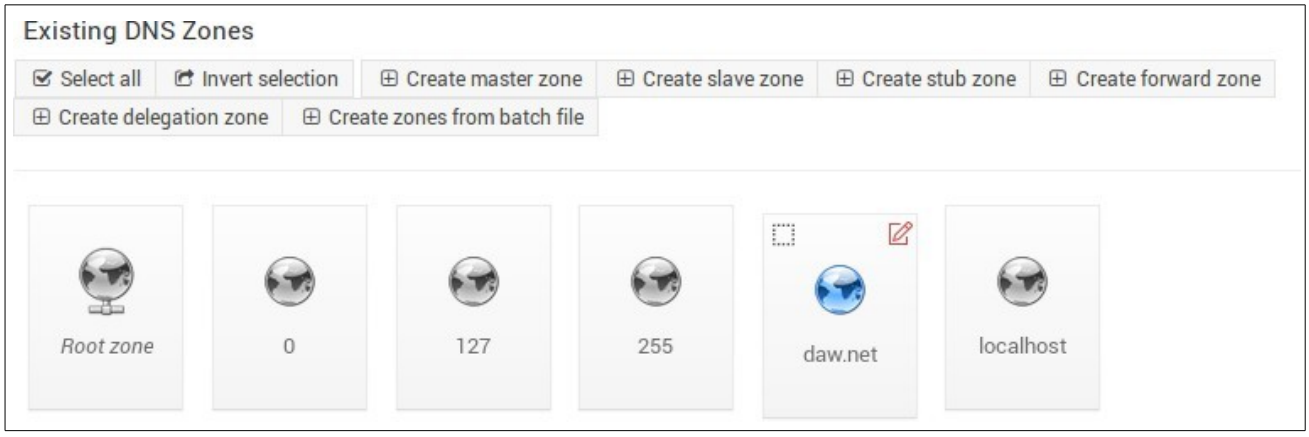
To apply the new configuration we have to click on ***Apply configuration***:

Now if we go to the ***Edit Master Zone*** window and click on ***Name Server*** we can see that there is record called *daw.net* and the name server is *LinuxServer.*.



> 📣 In this redord the final **.** (dot) in the name server is very important because it identifies the **root domain**.

If we need edit the zone again, we have to go to the ***BIND DNS Server*** main window and in the ***Existing DNS zones*** section we can choose the zone and clicking on the pencil we can edit it.



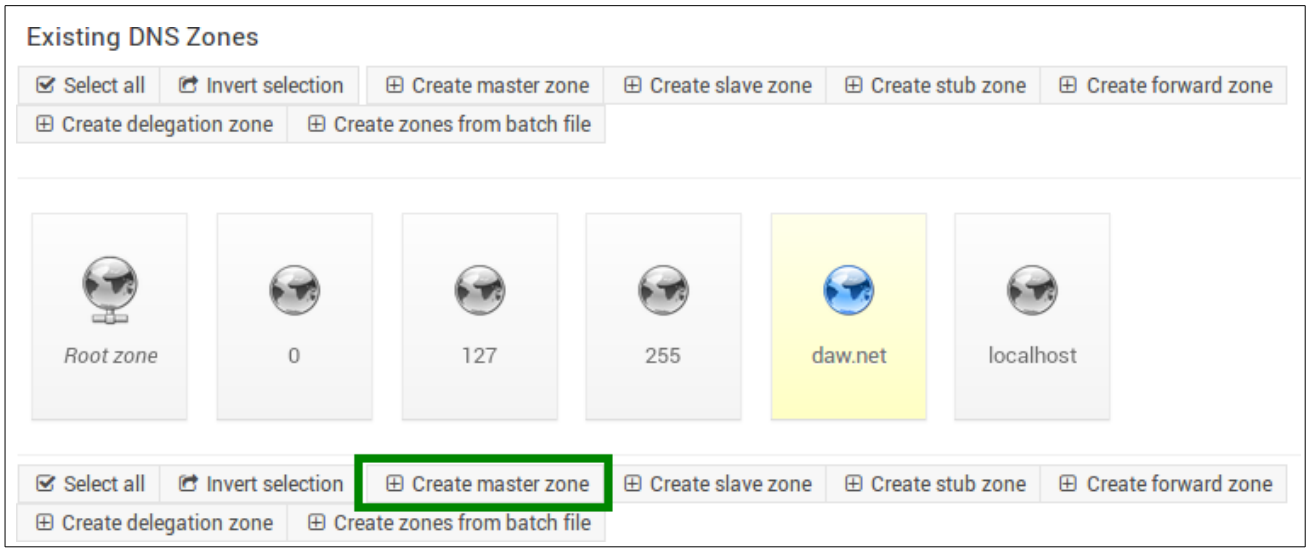
Now we can check the configuration using ***dig*** (remember that your IP addresses might be different).

```
lionel@linuxserver: ~
Archivo  Editar  Ver  Buscar  Terminal  Ayuda

lionel@linuxserver:~$
lionel@linuxserver:~$ dig linuxserver.daw.net

; <<>> DiG 9.11.3-1ubuntu1.9-Ubuntu <<>> linuxserver.daw.net
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 3148
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;linuxserver.daw.net.           IN      A

;; ANSWER SECTION:
linuxserver.daw.net.    38400   IN      A       192.168.0.2

;; Query time: 1 msec
;; SERVER: 127.0.0.53#53(127.0.0.53)
;; WHEN: Wed Oct 02 18:20:16 CEST 2019
;; MSG SIZE  rcvd: 64
```

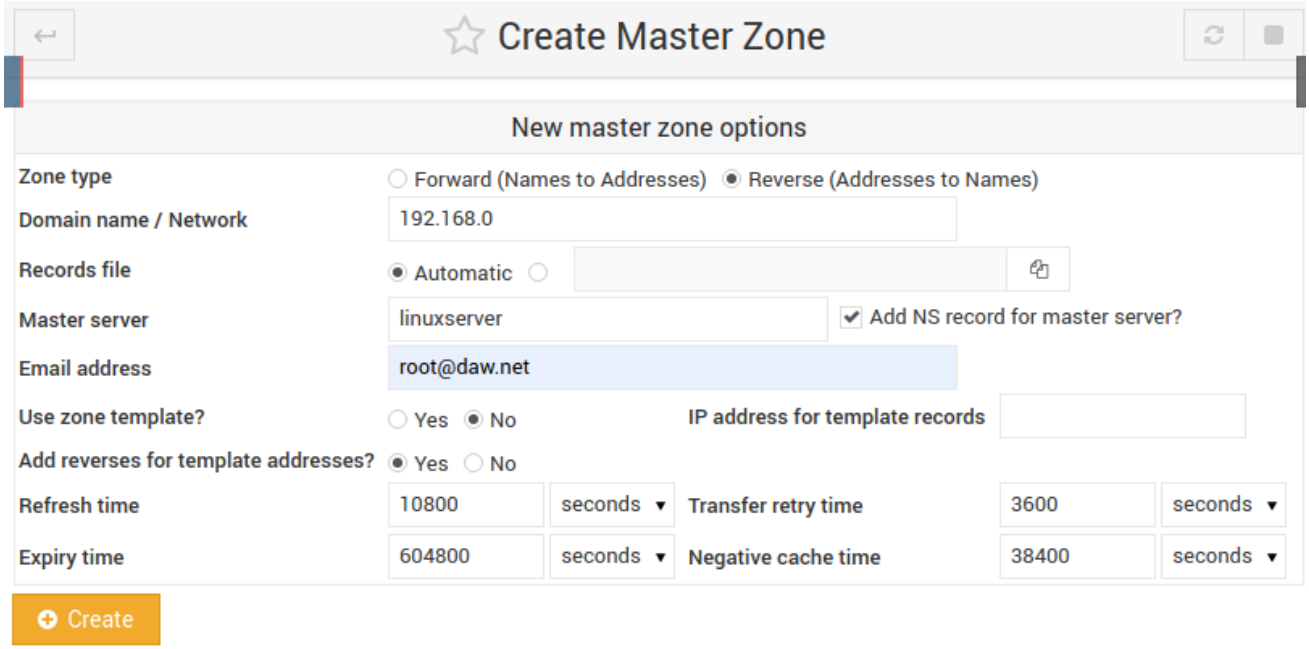
```
lionel@linuxserver: ~
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
lionel@linuxserver:~$ dig linuxclient.daw.net

; <<>> DiG 9.11.3-1ubuntu1.9-Ubuntu <<>> linuxclient.daw.net
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 54986
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;linuxclient.daw.net.           IN      A

;; ANSWER SECTION:
linuxclient.daw.net.    38400   IN      A       192.168.0.4

;; Query time: 0 msec
;; SERVER: 127.0.0.53#53(127.0.0.53)
;; WHEN: Wed Oct 02 18:21:34 CEST 2019
;; MSG SIZE  rcvd: 64
```

2.3 Configuring Lookup zone reverse

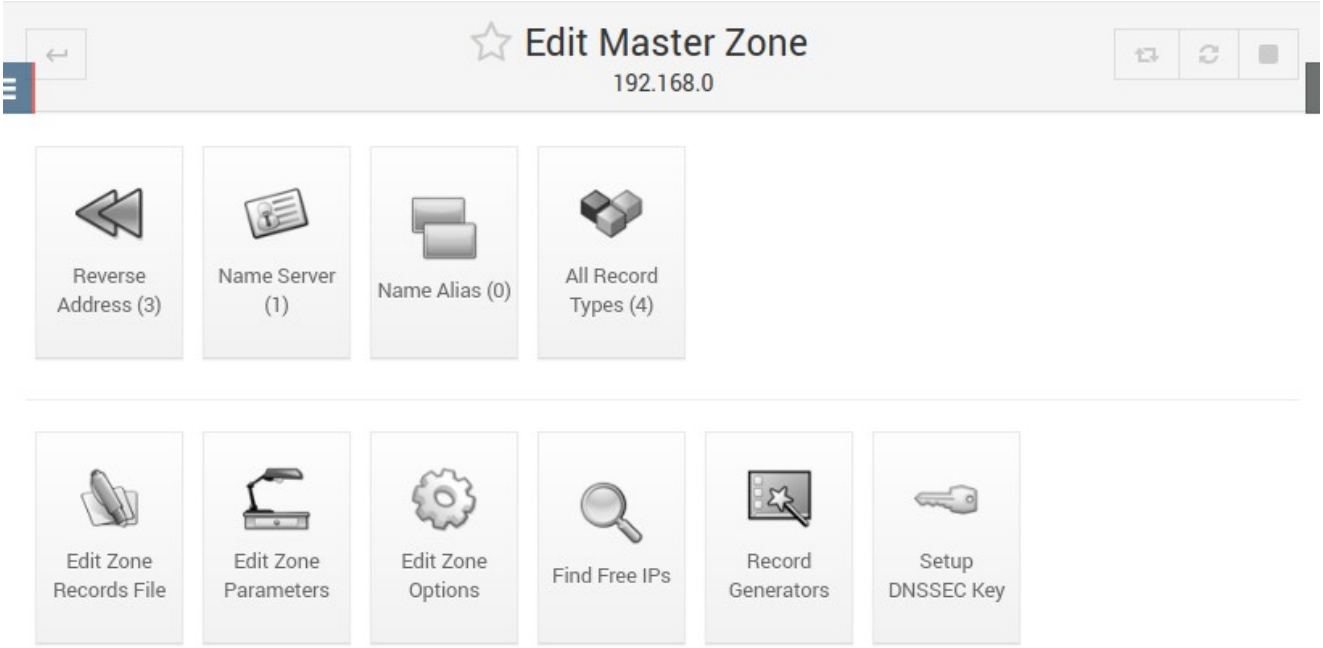
Now we are going to configure the reverse lookup zone. Its name will be *0.168.192.in-addr.arpa* and will be a master zone again.



Now fill the zone options. In this case we have to choose the option ***Reverse***, write the network (192.168.0), check that the master server is *LinuxServer* (our virtual machine), write the email address (root@daw.net) and click on the ***Create*** button.

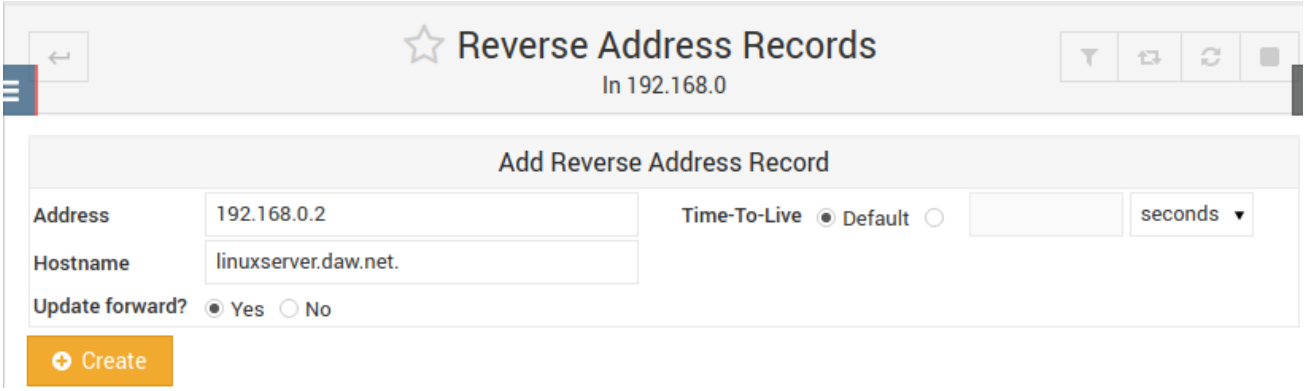


Once the master zone is created, we are going to create new PTR records (***Reverse address*** button).

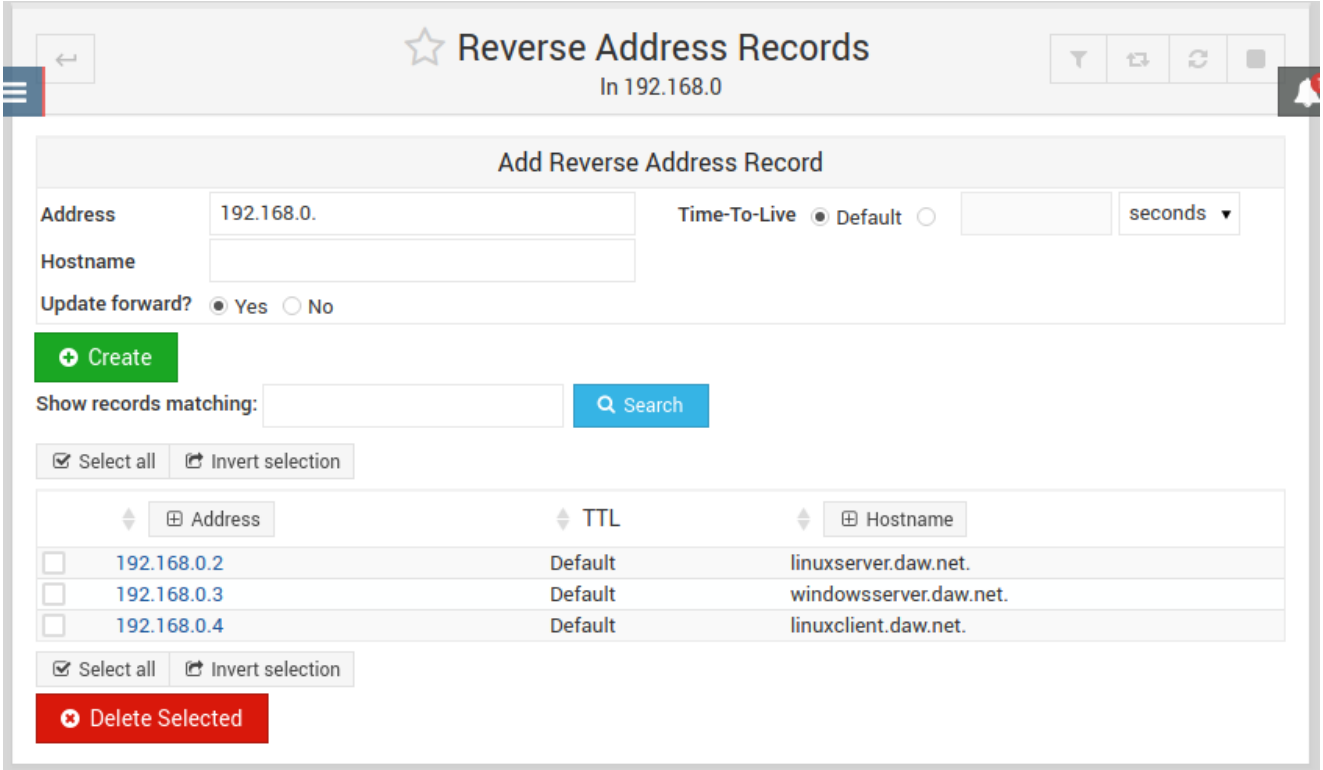


*If this zone had been created before we created the A records (checking Yes in **Update reverse?**) we will have the 3 reverse addresses already created when we created those address records.*

Now we have to create them, we have to click on ***Reverse Address***. We are going to create one for each virtual machine. For instance, this will be the form filled for *LinuxServer*:



Once created all, we will have something like this:



Then, click on ***Apply configuration***.

> 📣 Remember, in this record the final **.** (dot) in the name server is very important because identify the **root domain**.

Now, we can check the configuration using ***nslookup <IP-address>***

```
administrador@LinuxServer:~$ nslookup 192.168.1.3
Server:         192.168.1.2
Address:        192.168.1.2#53

3.1.168.192.in-addr.arpa        name = windowsserver.daw.net.
```

```
administrador@LinuxServer:~$ nslookup 192.168.1.4
Server:         192.168.1.2
Address:        192.168.1.2#53

4.1.168.192.in-addr.arpa        name = linuxclient.daw.net.
```

2.4 Configuring the LinuxClient

Now start the LinuxClient virtual machine and change the network configuration (Settings → Network or Configuración → Red) to set the DNS server the same way you did with LinuxServer. Remember to restart the network.

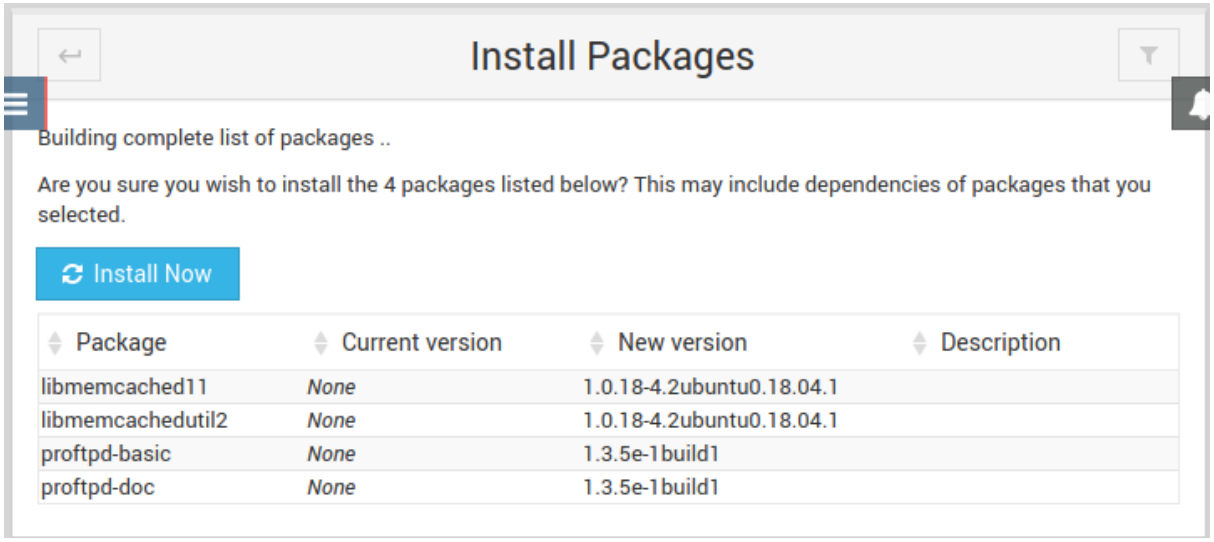
3. FTP

3.1 Installation

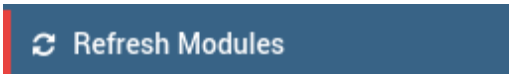
First of all, we need to install the ***ProFTPD*** module. To do so go to the ***Un-used Modules*** group and click on it.

As we do not have installed the package in our system, click on *Click here*

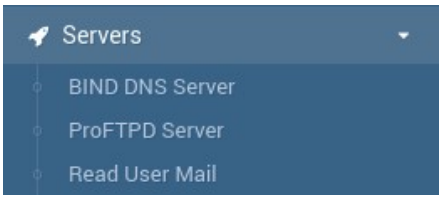
Then confirm the installation clicking *Install Now.*



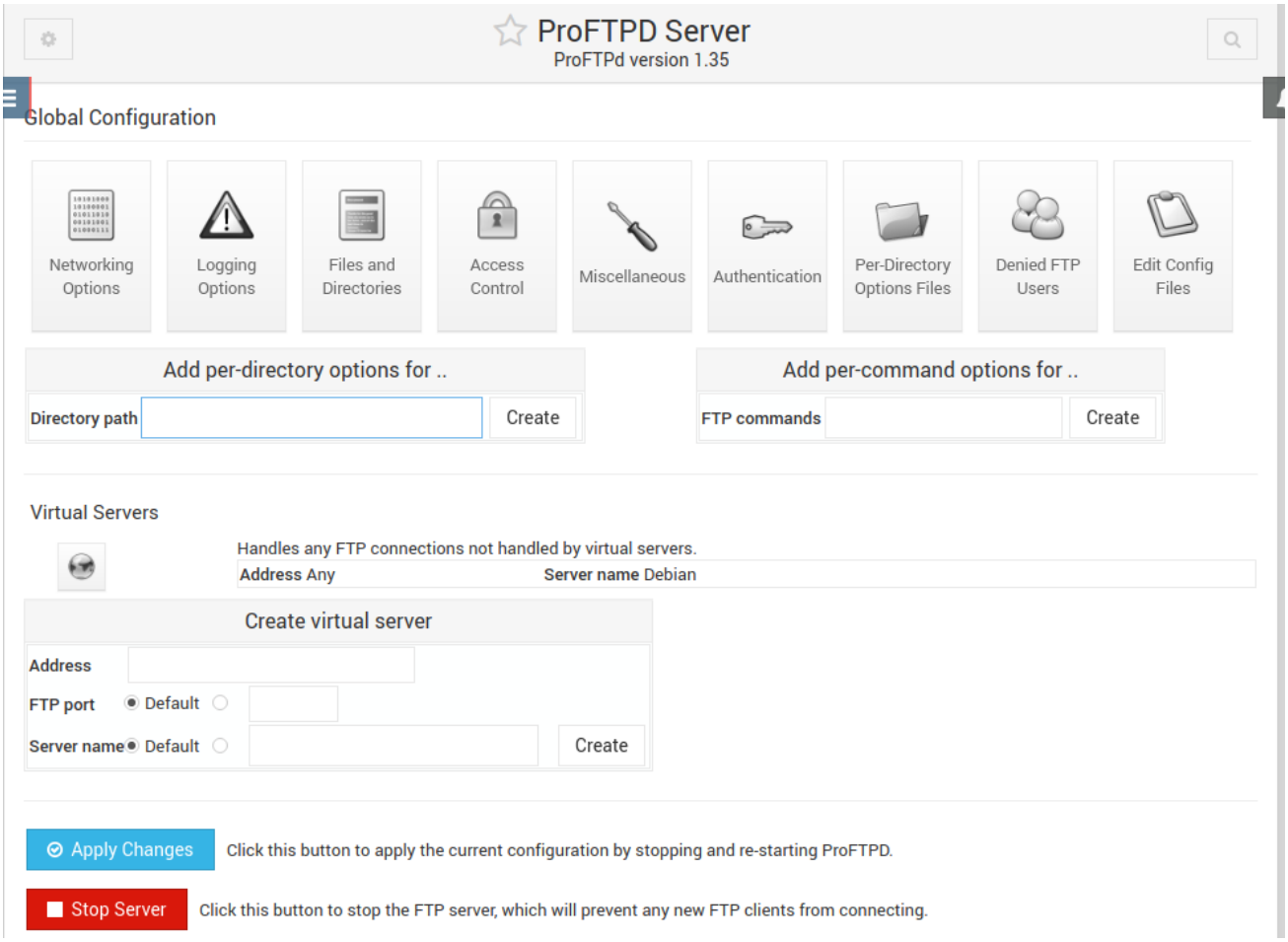
Once installed, click on ***Refresh Modules*** for Webmin to detect the newlly installed module.



Now the ***ProFTPD Server*** should appear in the Servers menu.



Click it now to open its global configuration.



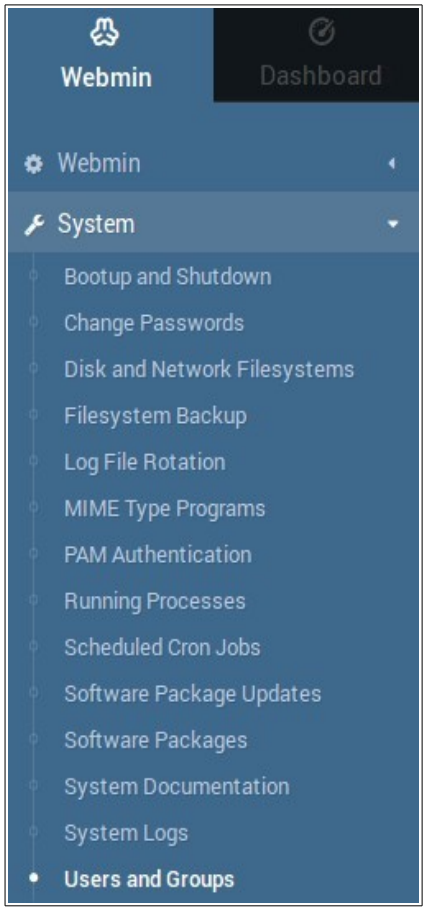
In the next steps we are going to see how a user can access the FTP server to download and upload files, as well as how to allow anonymous connections (without passwords). To do so we will work with a unique virtual server (the default server). We don't need to create several virtual servers because that's useful only if our system has multiple IP addresses.

Anyway, you can get all the information in the official documentation: https://doxfer.webmin.com/Webmin/ProFTPD_Server

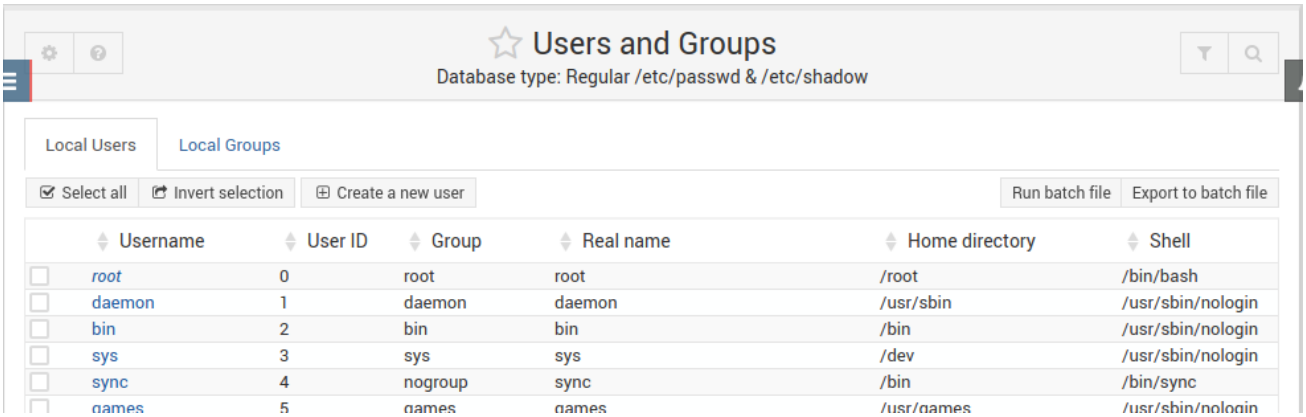
3.2 Test

It's important to understand that FTP servers use the operating system's users to allow connections. Therefore, any user in the system should be able to connect to the FTP Server.

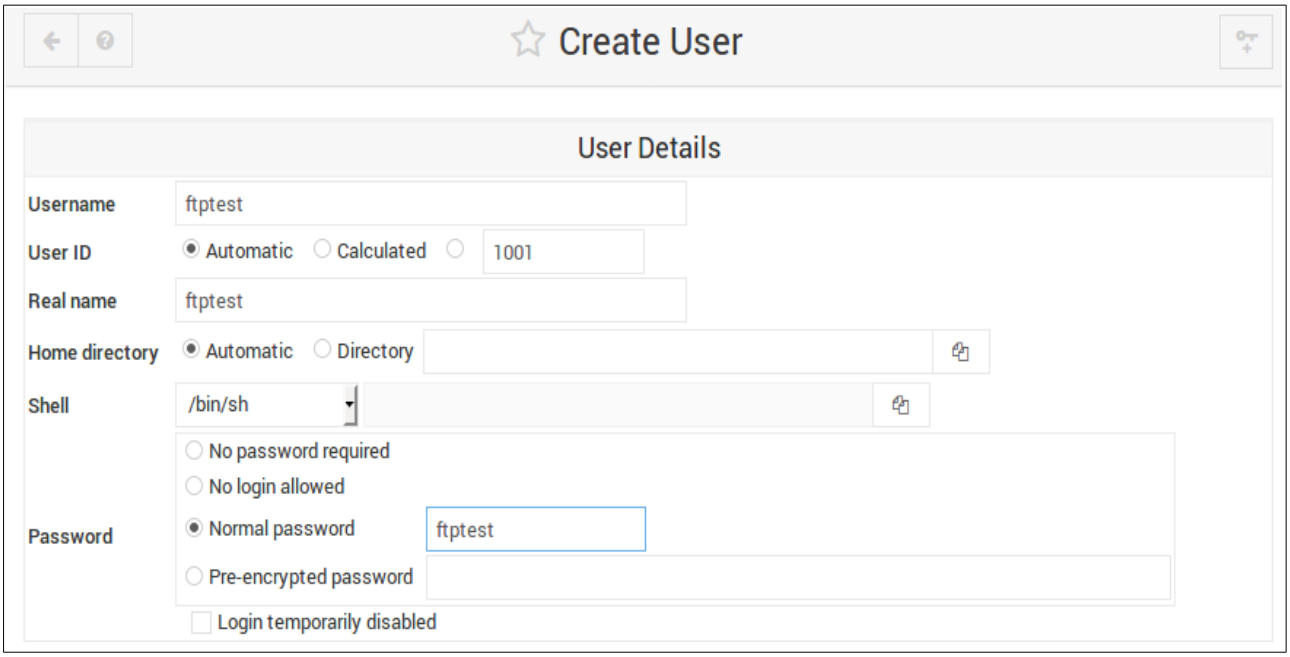
Anyway, for this test we are going to create a specific user. To do so, go to ***Users and Groups*** in the ***System*** group.



There we can see all the users and groups in LinuxServer.



Click on ***Create a new user*** to create a new user called ***ftptest***. Fill in the **Username**, **Real name** and **Normal password** and click ***Create***.



The user should have been created and you should be able to see it in the users list.

	bind	122	bind		/var/cache/bind	/usr/sbin/nologin
	proftpd	123	nogroup		/run/proftpd	/usr/sbin/nologin
	ftp	124	nogroup		/srv/ftp	/usr/sbin/nologin
	ftptest	1001	users	ftptest	/home/ftptest	/bin/sh

Now we are going to test if we can connect from ***LinuxClient*** to the FTP Server in ***LinuxServer:***

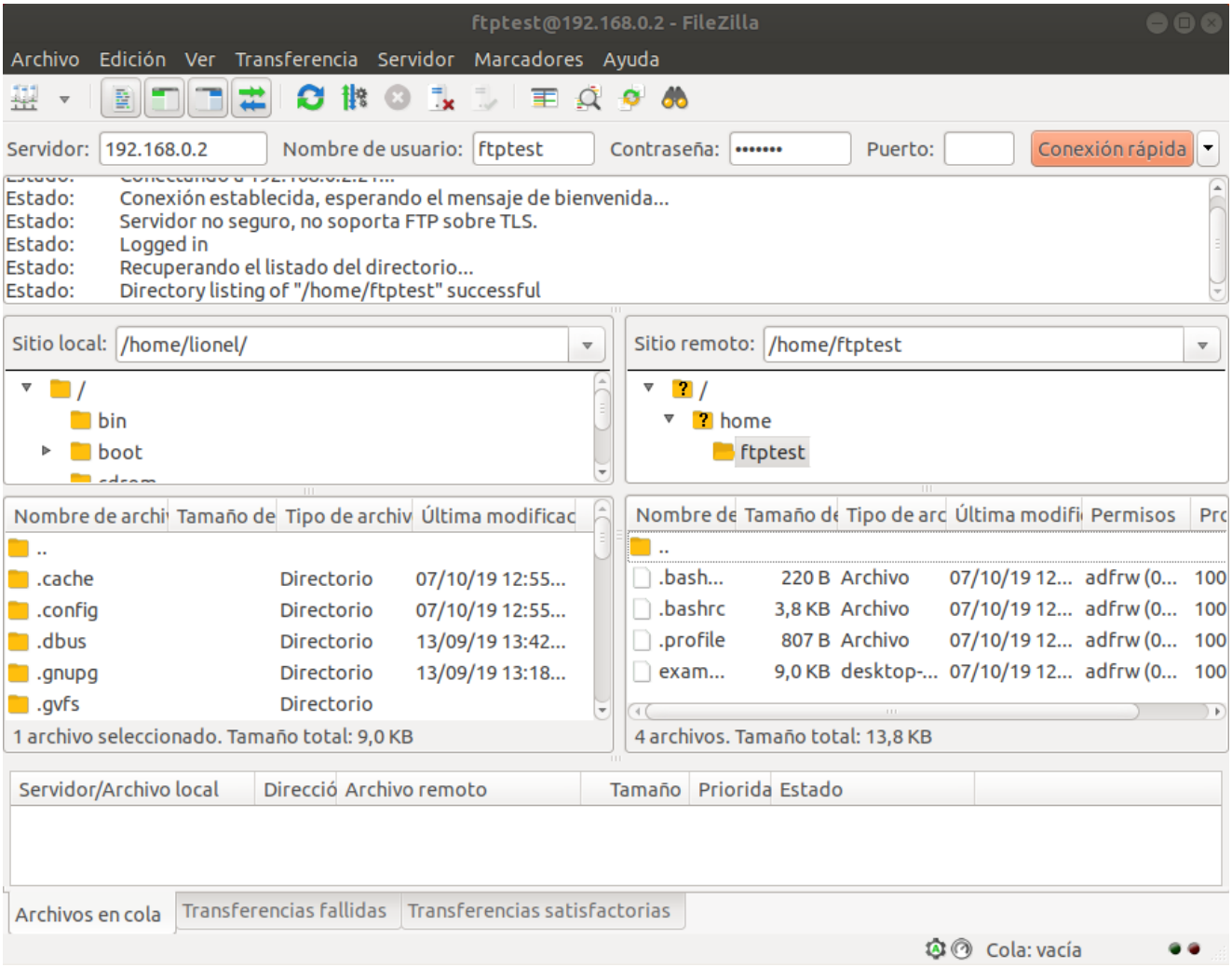
1. **Start the LinuxClient virtual machine** (LinuxServer must be running too).

   If you don't have enough RAM for two virtual machines running at the same time, you can do the test from your physical machine instead of LinuxClient.

2. **Install an FTP Client** software, for example FileZilla https://filezilla-project.org/

   *In Linux you can install it from a terminal with the command **sudo apt install filezilla**.*
   In Windows go to the FileZilla website and download the Windows installer.

3. **Open FileZilla and connect**. To do so, you have to specify the server (192.168.0.2), username (ftptest) and password (ftptest). The port is not necessary, it will use 21 by default. Then click ***Connect***. If everything went fine it should connect propperly.



The left side shows the files and folders in the LinuxClient machine.

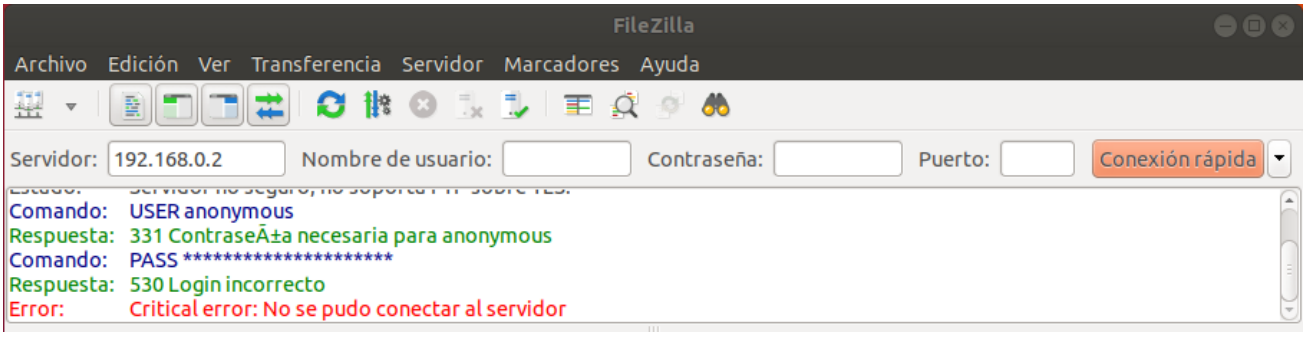
The right side shows the files and folders in the LinuxServer machine.

This FTP connection allows you to transfer files from one machine to the other one very easily (drag and drop), as well as creating, renaming and deleting files and folders.

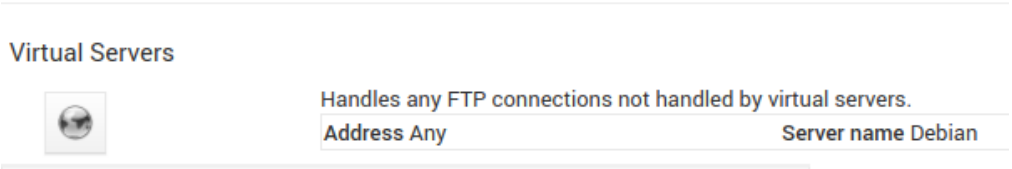
3.3 Allow Anonymous Connections

Sometimes it's usefull to allow anonymous connections. This means, being able to connect to the FTP Server without hacing to use a user and password (anyone can connect). For safety reasonsthis type of connections are limited by default, they can do hardly anything.

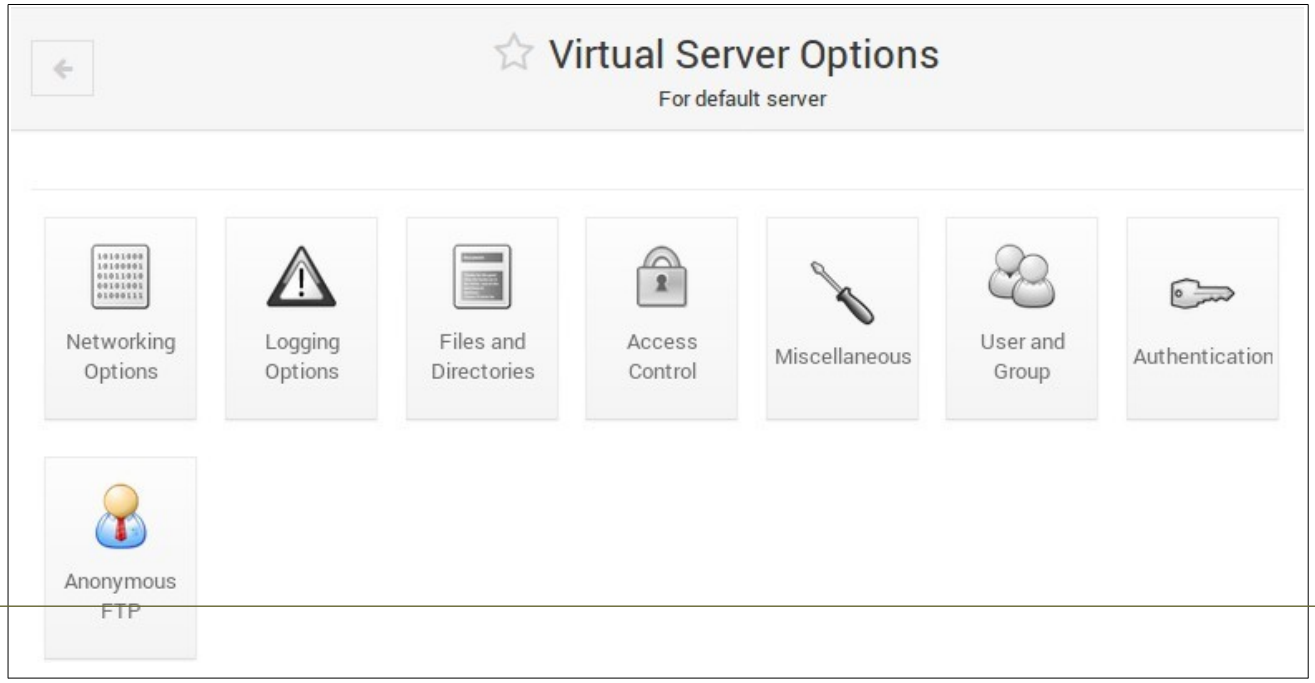
If we try to connect with an anonymous user (empty username and password) the connection will fail. This is normal because anonymous connections are not allowed (yet).

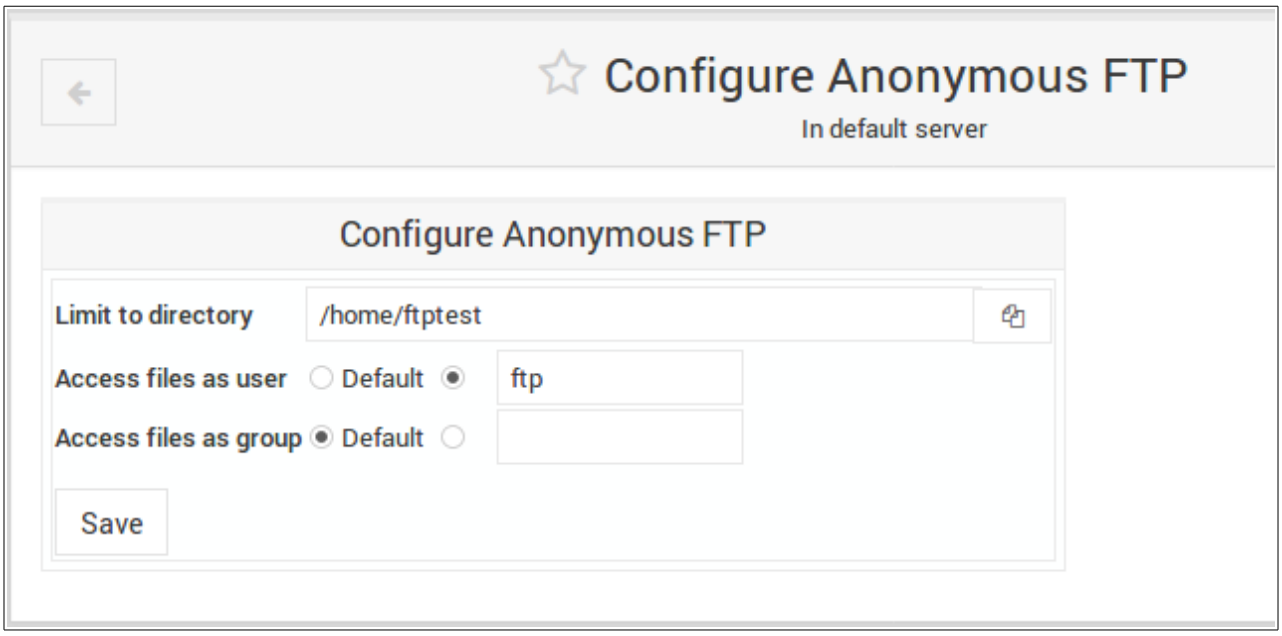


To allow Anonymous Connections we have to go back to **Webmin** => **Servers** => **ProFTPD Server** and click on the *Default Virtual Server* (the world globe).



The virtual server options will appear. Now click on the ***Anonymous FTP*** icon.

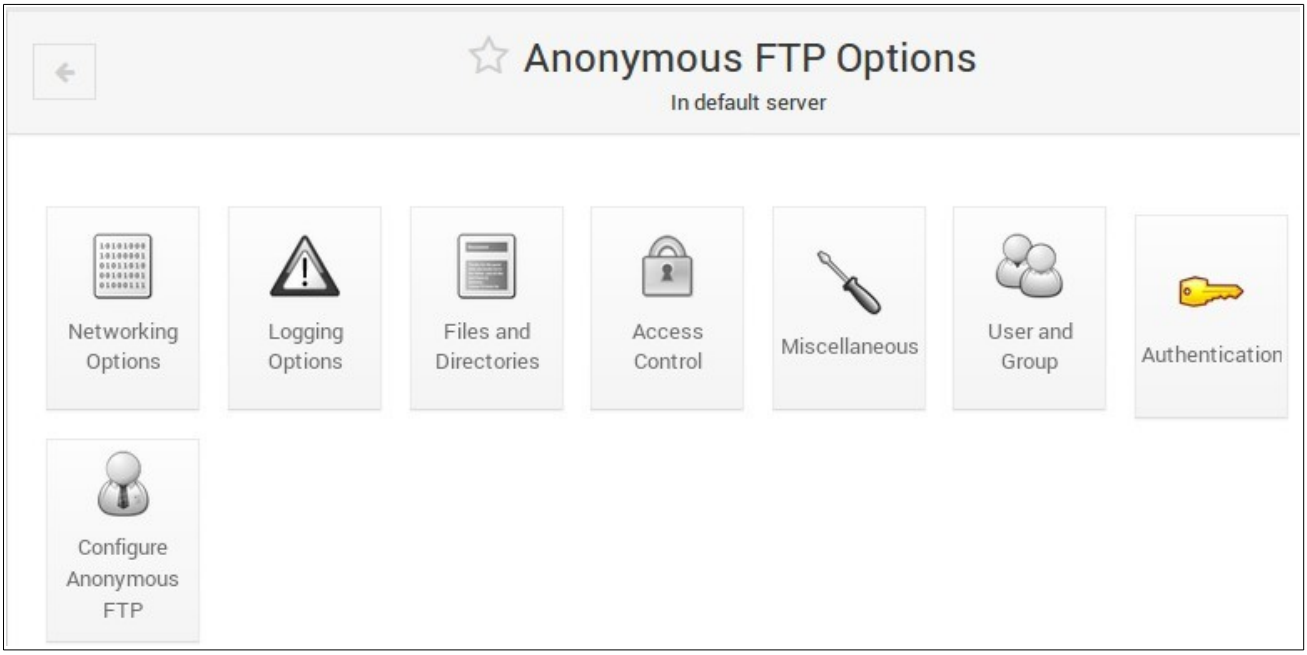


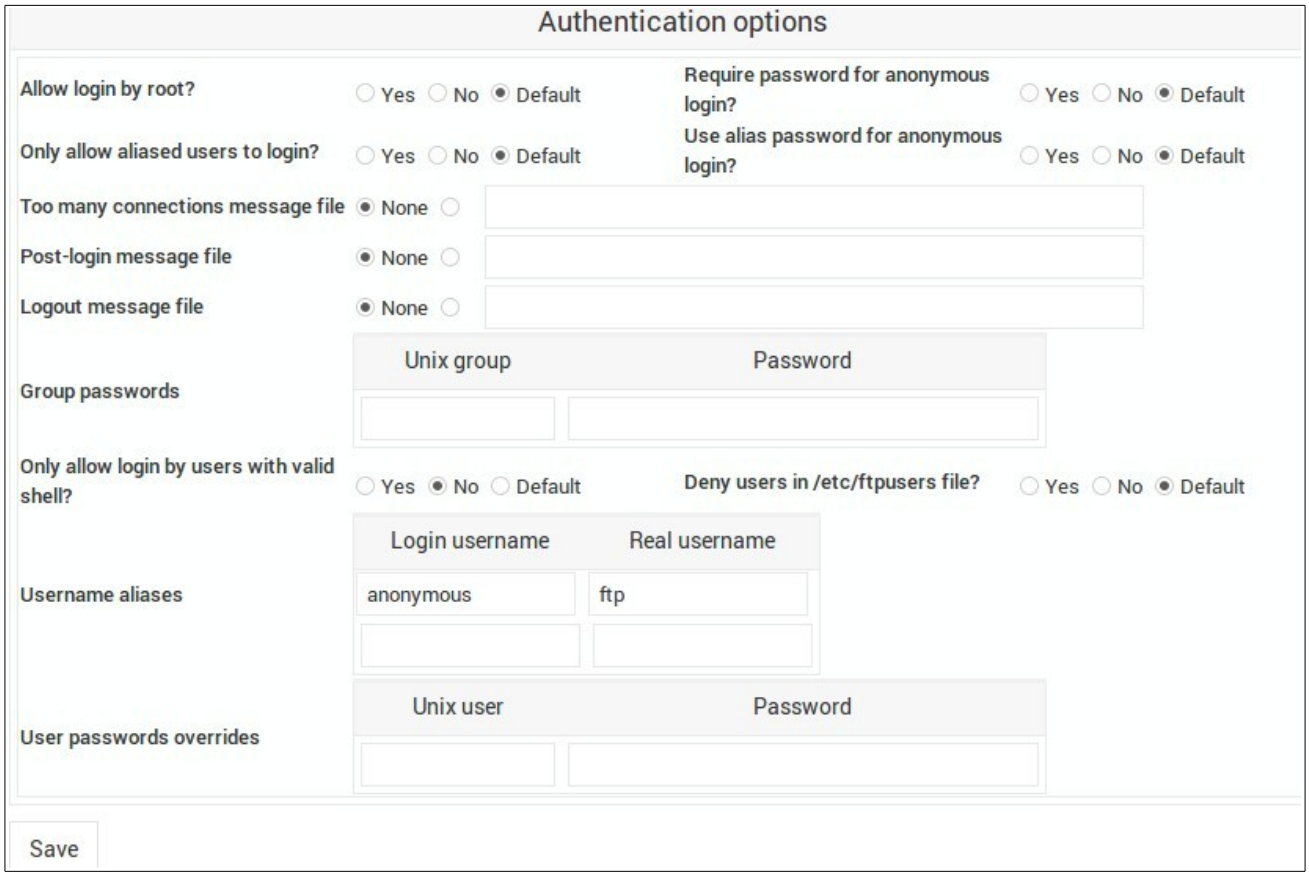


Here, we have to set the directory that anonymous users will be able to access. For instance, we could specify the same as the ***ftptest*** user (its home folder). If you wish you could create a new folder only for anonymous user.

The access to files will be the same as the ***ftp*** user (afterwards we will change the permissions) and the group will be by default. Then click on ***Create***/***Save***.

Now we have to click on the authentication icon:





Here we only have to change the radiobutton of ***Only allow login by users with valid shell?*** to ***No*** to allow the access to the anonymous users to the server. Then we click on ***Save***. From now on the anonymous user can access to the directory specified (/home/testftp/).

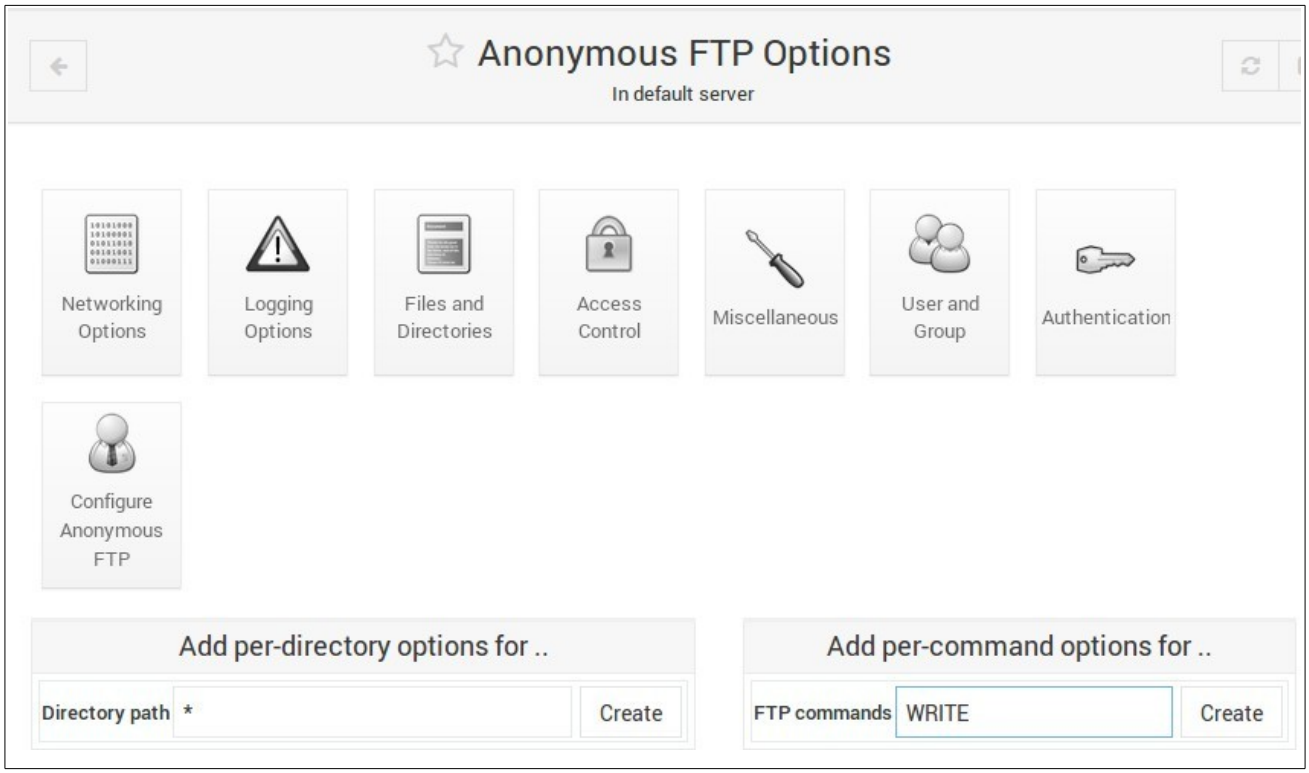
For safety reasons, now we are going to limit the anonymous user actions. We don't want him to be able to write on any folder. To do so:

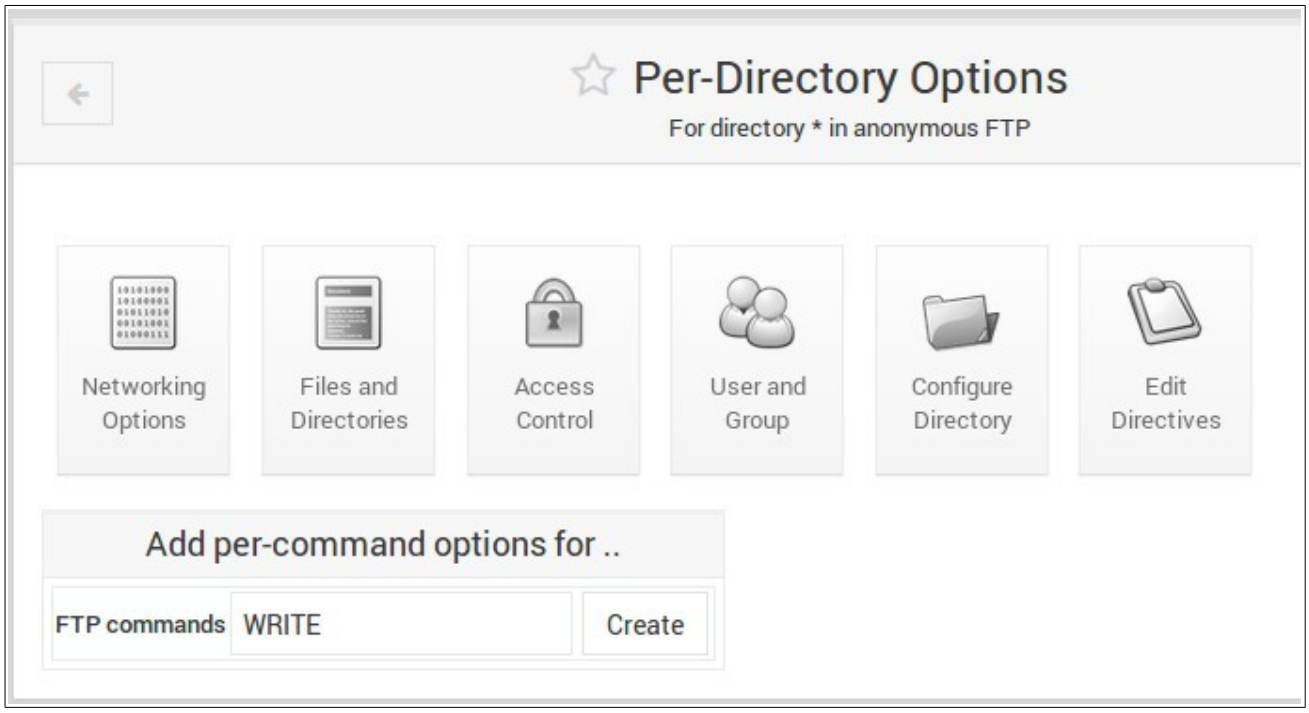
First, in the ***Anonymous FTP Option*** window:

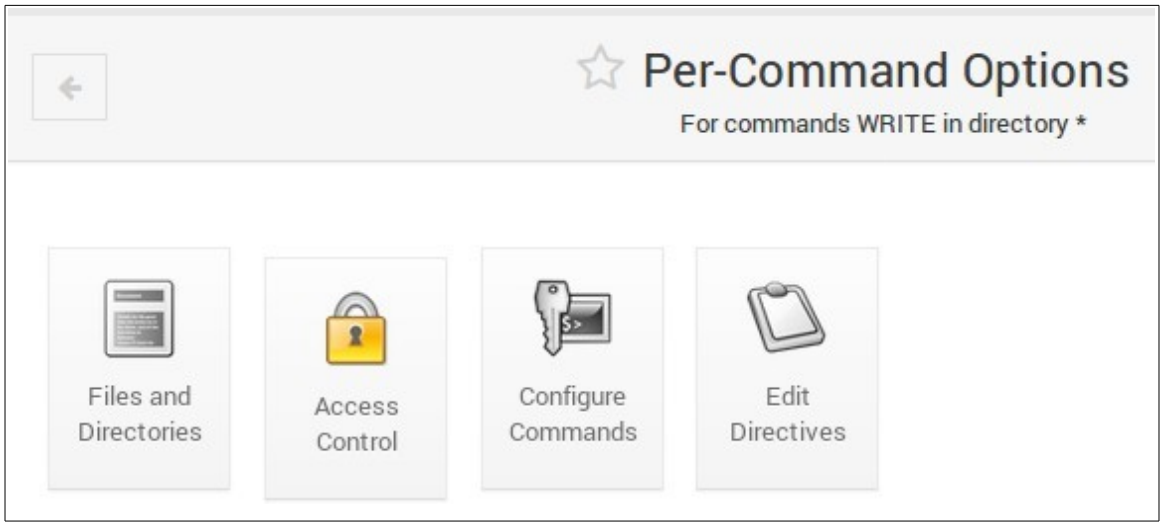
- In the ***Add per-directory options for ..*** section write an asterisk * in Directory path to indicate all directories and click on ***Create***.
- In the ***Add per-command options for ..*** section write ***WRITE*** in FTP commands and click on ***Create***.

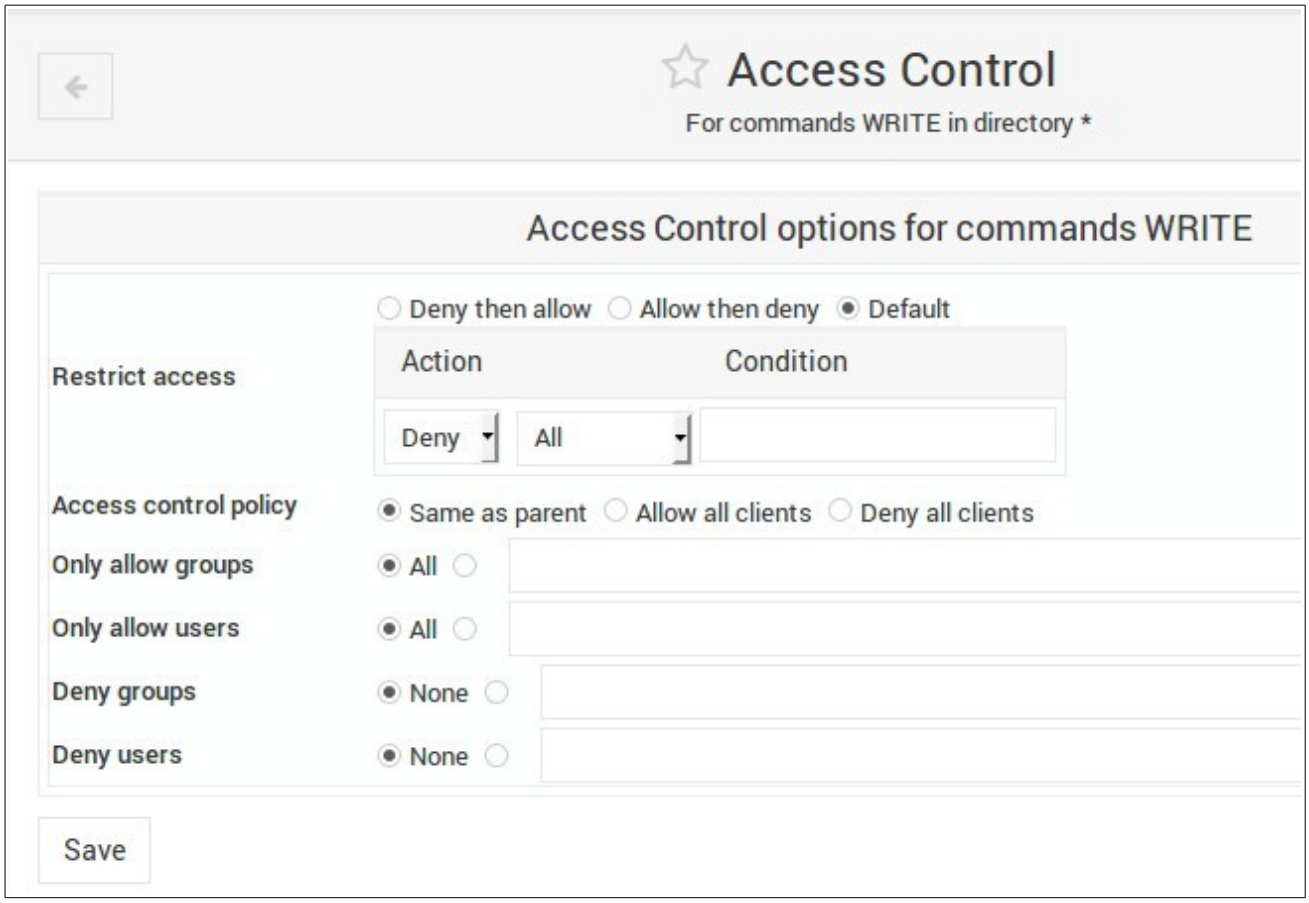
Then, click on ***Access Control*** icon to create an access rule for WRITE in directory * (everywhere). Under ***Action*** select ***Deny*** and ***All*** to indicate that any write action will be denied. Now click ***Save***.

Finally, we have to **apply the changes**:

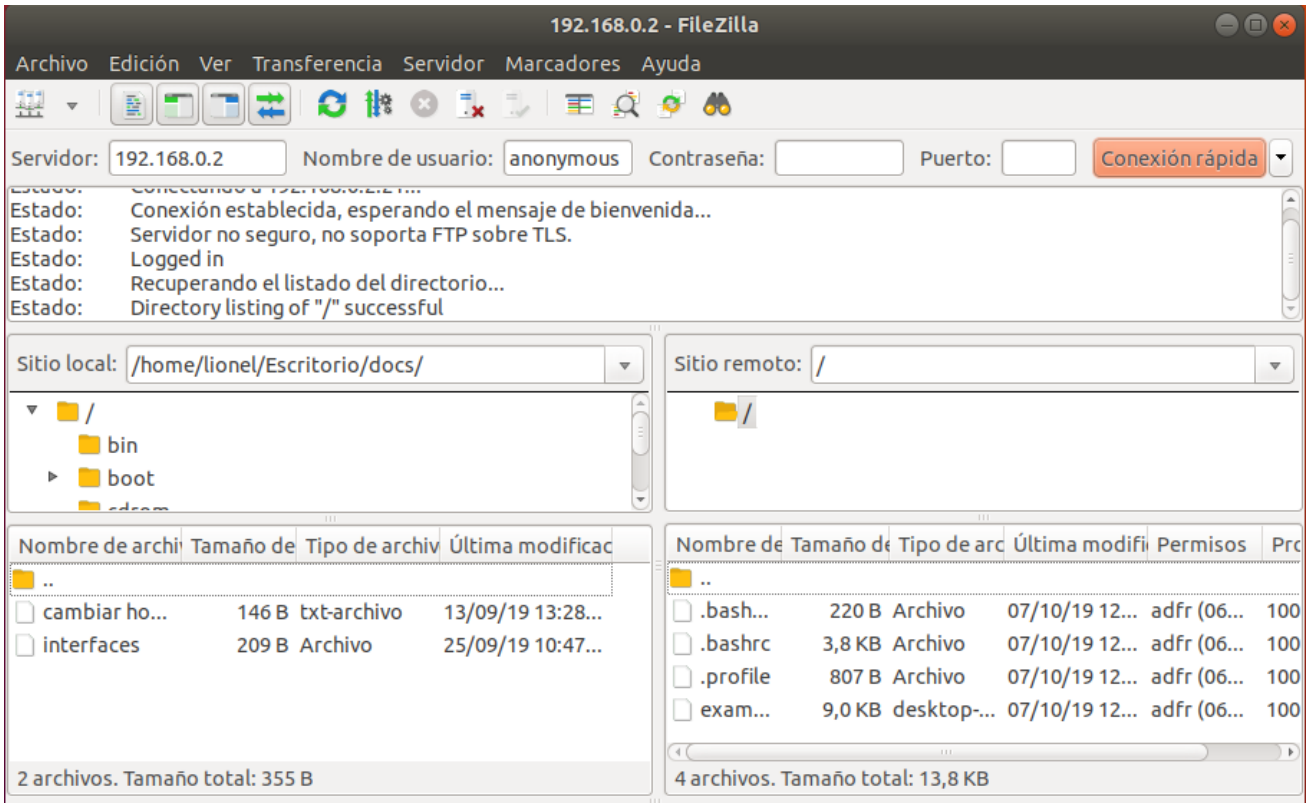

Anonymous FTP Options
In default server
Networking Options
Logging Options
Files and Directories
Access Control
Miscellaneous
User and Group
Authentication
Configure Anonymous FTP
Add per-directory options for ..
Directory path
*
Create
Add per-command options for ..
FTP commands
WRITE
Create


Per-Directory Options
For directory * in anonymous FTP
Networking Options
Files and Directories
Access Control
User and Group
Configure Directory
Edit Directives
Add per-command options for ..
FTP commands
WRITE
Create


Per-Command Options
For commands WRITE in directory *
Files and Directories
Access Control
Configure Commands
Edit Directives


Access Control
For commands WRITE in directory *
Access Control options for commands WRITE
Restrict access
Deny then allow
Allow then deny
Default
Action
Condition
Deny
All
Access control policy
Same as parent
Allow all clients
Deny all clients
Only allow groups
All
Only allow users
All
Deny groups
None
Deny users
None
Save

Now, go back to the ***LinuxClient*** virtual machine and try to connect as an anonymous user.



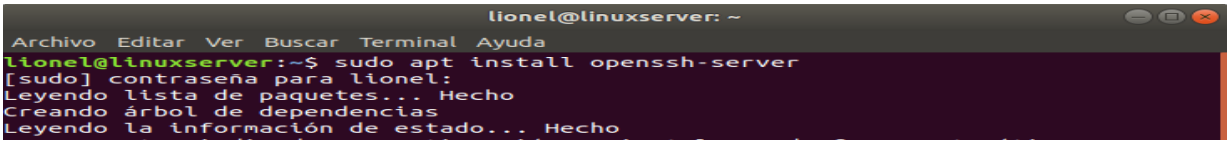
You should be able to connect and navigate all folders, BUT you should not be able to create, delete nor modify any files or folders.

3. SSH

3.1 Installation

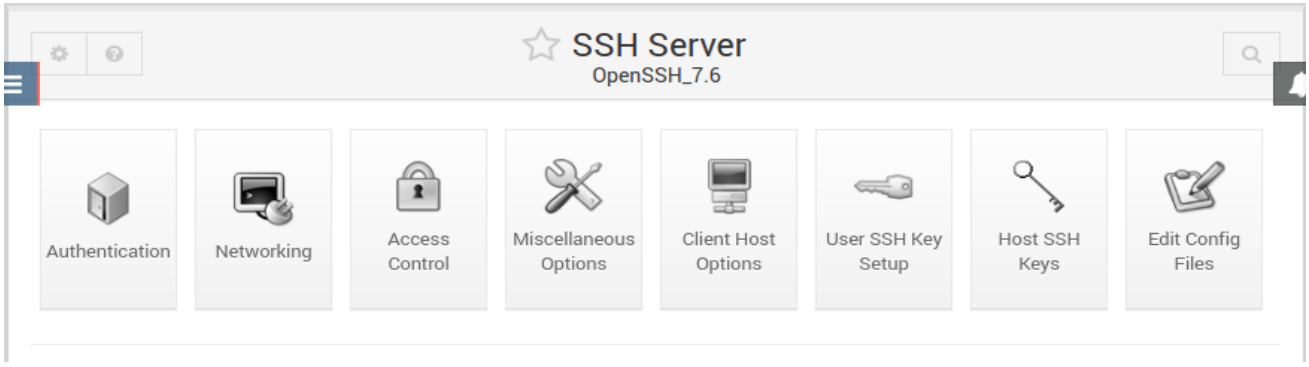
First of all, we have to install the ***SSH*** package. To do so, we have to go to the group ***Un-used Modules*** and select ***SSH Server***. As we do not have the package installed in our system Webmin warns about it, and it will probably not allow you to install it the same way as the DNS and FTP servers.

Open a terminal in the LinuxServer machine and install the ***openssh-server*** package. To do so type: **sudo apt-get install openssh-server**. If the installation fails you might update the Ubuntu repositories using **sudo apt-get update**.



Now click on ***Refresh Modules*** to include the module in the ***Servers*** section.

Now the SSH server is ready to use.



3.2 Test

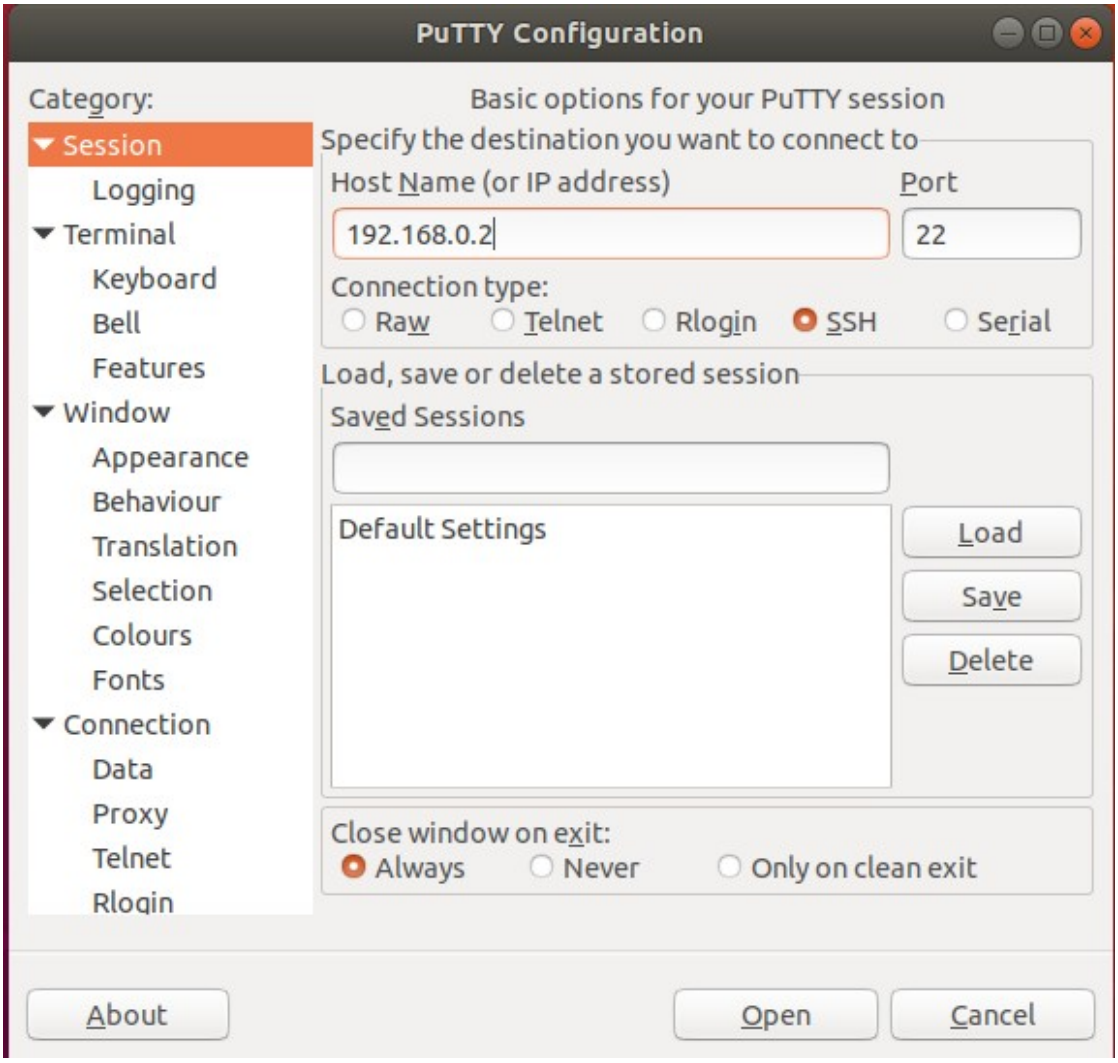
Now we are going to try to connect to the ***LinuxServer*** from the ***LinuxClient*** using the ***ftptest*** user created before (you can also create a new one).

We could use the default SSH-client but lets try installing Putty.

Go to the ***LinuxClient*** machine and type in a terminal: ***sudo apt-get install putty***.

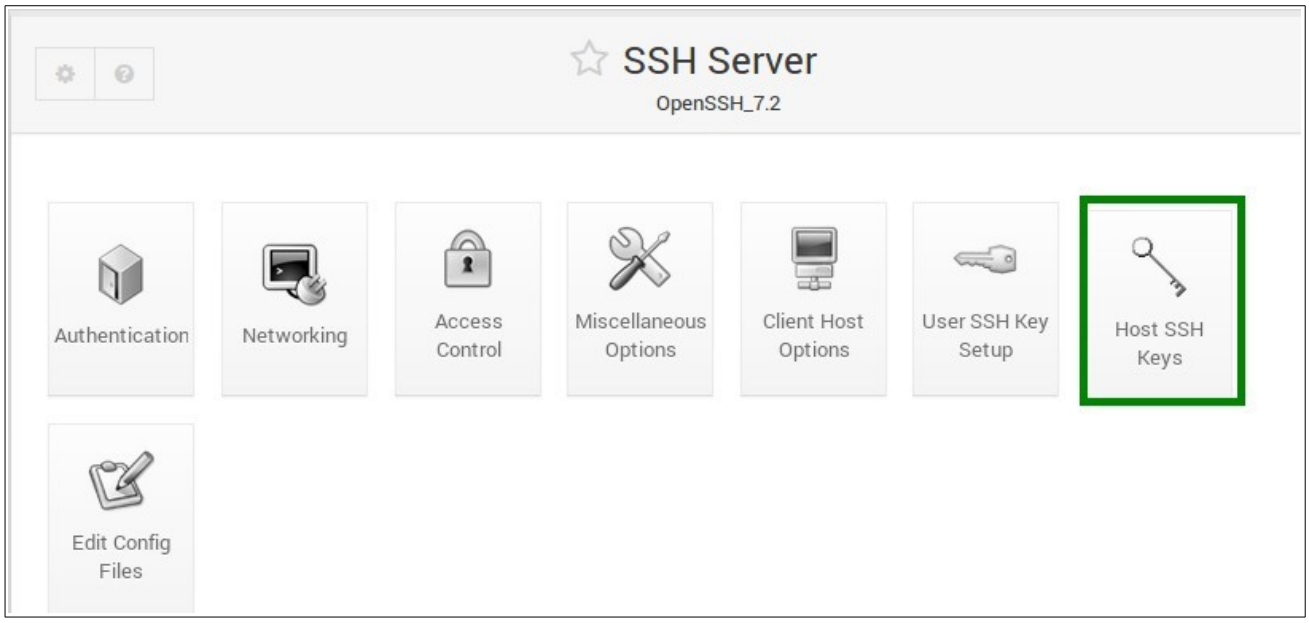
You can also try it from your Windows physical machine by downloading it from the web page http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html

Once installed open it and try connecting to ***LinuxServer*** (192.168.0.2) in port 22 clicking ***Open***:

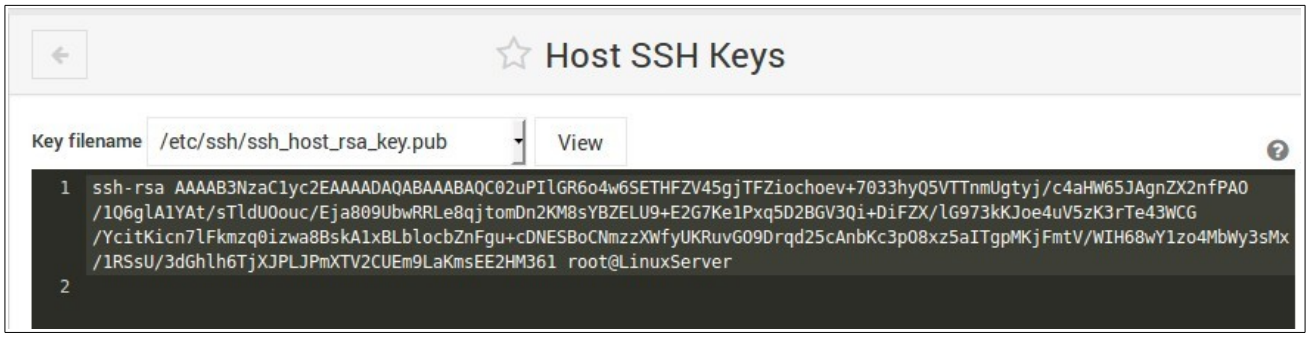


Then, the server will send its key fingerprint:

Now, we are going to check if that fingerprint is the same as our server. We can find the content of the file with the key in ***Host SSH Keys***:



and we choose the file */etc/ssh/ssh_host_rsa_key.pub* (see that in the alert before the key fingerprint is rsa2):



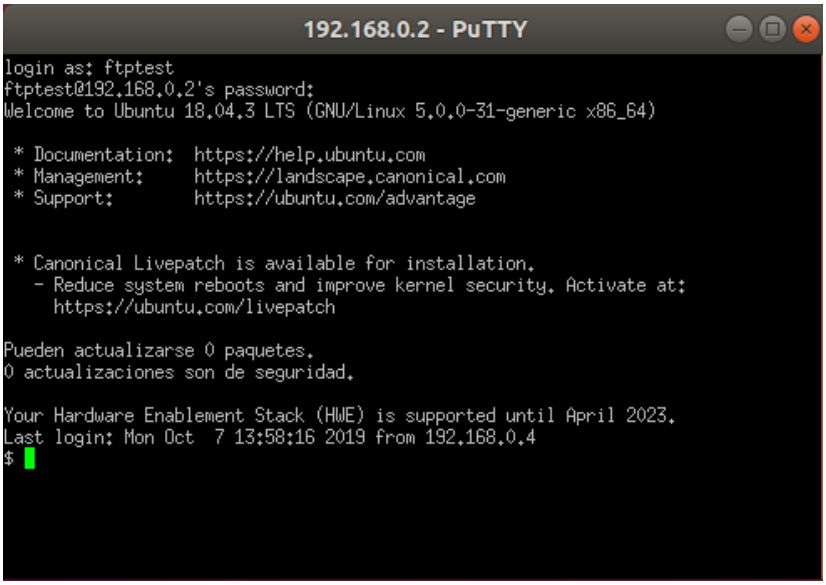
Now, to see the key fingerprint we have to write in the terminal of the *LinuxServer* machine:

ssh-keygen -l -E MD5 -f /etc/ssh/ssh_host_rsa_key.pub

```
administrador@LinuxServer:~$ ssh-keygen -l -E MD5 -f /etc/ssh/ssh_host_rsa_key.pub
2048 MD5:31:7c:63:62:36:35:dc:48:c1:b7:ab:9c:c1:ca:cc:70 root@LinuxServer (RSA)
```

We can check that it is the same, so we click on ***Accept*** in the Putty, so the client stores the fingerprint and it will not show the alert again.

Finally, we access with the ***ftptest*** user and password:



Now you are connected to the LinuxServer machine via a shell (terminal) and you could run linux commands as if you were there :