Installing Webmin on Ubuntu

Contents

[1. Prerequistes 2](#_Toc39157228)

[1.1 Account Creation 2](#_Toc39157229)

[1.2 Sudo access……………………………………………………………………………………………………………………………………4](#_Toc39157230)

[2. Setup Webmin and install through CLI 5](#_Toc39157231)

[2.1 Package lists and dependancies 5](#_Toc39157232)

[2.2 Enable repository 7](#_Toc39157233)

[3. Adjusting Firewall/Install UFW 8](#_Toc39157234)

[3.1 Install UFW and check status 8](#_Toc39157235)

[3.2 Allowing Connections 9](#_Toc39157236)

[4. Accessing Webmin Web Interface 10](#_Toc39157237)

[4.1 Access Interface 10](#_Toc39157238)

[4.2 Upgrading Webmin 13](#_Toc39157239)

[5. Installing BIND DNS with Webmin 19](#_Toc39157241)

[5.1 Access through web interface 19](#_Toc39157242)

[6. Extended Content: Installing Ubuntu Guide 23](#_Toc39157244)

[6.1 Install for Windows 23](#_Toc39157245)

7. Resources ………………………………………………………………………………………………………………………………………….24

# Installing Webmin on Ubuntu Guide

Prerequisites to this guide: User must be logged in as root or user with sudo privileges. For follow on instructions on how to log in a user with sudo privileges refer to subsections below. To continue to the installation of Webmin, proceed to **Section I** on this document.

* Note: Logging in as a sudo user will allow user to execute administrative commands without having to log in as a root user.

## Prerequisite :

Preform the following steps to log in with sudo privileges.

Log in to the system as root user:

$ ssh root@server\_ip\_address

## 1.1 New user account creation:

Adduser username ( replace username with custom name)

-Note: You will need to set and confirm a new secure password after this command.

Adding user `username' ...

Adding new group `username' (1001) ...

Adding new user `username' (1001) with group `username' ...

Creating home directory `/home/username' ...

Copying files from `/etc/skel' ...

New password:

Retype new password:

passwd: password updated successfully

* Optional: Changing user information for the new username. To skip, press enter to continue or fill out the appropriate information.

## 1.2 Adding new user to sudo group:

* Note: All members of the sudo group are granted with sudo access by default on Ubuntu systems. If you want to add the new user created above to the sudo group, use the “user mod command” below.

$ usermod -aG sudo username

Testing sudo access:

su – username

sudo whoami

-Note: if the command output shows: root. You have successfully created the sudo user. To use sudo use command: ( you will need to enter the password created in the above steps)

sudo ls -l /root

How to install Webmin on an Ubuntu 18.04 server

\*These instructions apply for Ubuntu 16.04 as well.

1. Enable Webmin repository and installing the Webmin package through CLI.

2.1 Updating packages list and install dependencies

$ sudo apt update

$ sudo apt install software-properties-common apt-transport-https wget

Import Webmin GPG key using *wget command*

$ wget -q http://www.webmin.com/jcameron-key.asc -O- | sudo apt-key add –

$ sudo apt-get update

## Enable Webmin repository

$ sudo add-apt-repository "deb [arch=amd64] http://download.webmin.com/download/repository sarge contrib"

Install Webmin Latest Version

$ sudo apt-get install webmin

Finishing install

Webmin install complete. You can now login to https://your\_server\_ip\_or\_hostname:10000/

as root with your root password, or as any user who can use sudo

to run commands as root.

Note: The output of a successful install will display the following message.

3. Adjusting Firewall/Install UFW

-Note: Webmin listens for connections on port 10000 for all network interfaces. If your server runs UFW, the Webmin port must be open. UFW is installed by default in Ubuntu 18.04, if it is **NOT**. Please follow the instructions below. If it is, skip to **Section III**. Keep in mind that best practice is to run administrative commands as sudo user.

## 3.1 Install UFW

$ sudo apt install ufw

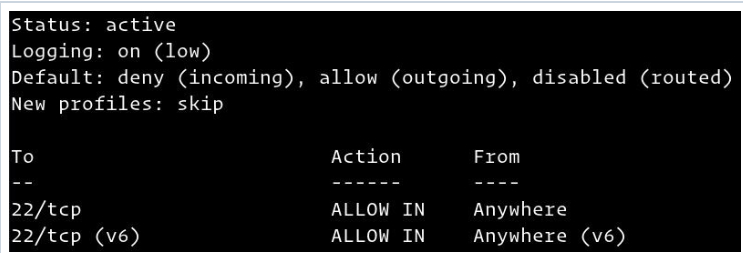
Check Status

$ sudo ufw status verbose

-Note: If you have never activated UFW the output will display:

Status: inactive

If UFW is already activated the output will display:



* Note: continue to **Section III** if UFW is in an active state.

Application Profiles

$sudo ufw app list

* Note: This command will display a similar output to the following:

Available applications:

Dovecot IMAP

Dovecot POP3

Dovecot Secure IMAP

Dovecot Secure POP3

Nginx Full

Nginx HTTP

Nginx HTTPS

OpenSSH

Postfix

Postfix SMTPS

Postfix Submission

To view more profile specifications and rules use the following command:

$ sudo ufw app info 'Nginx Full'

The output will resemble the following display:

Profile: Nginx Full

Title: Web Server (Nginx, HTTP + HTTPS)

Description: Small, but very powerful and efficient web server

Ports:

80,443/tcp

3.2 Allowing SSH Connections

-Note: You must enable a rule allowing incoming SSH connections before enabling the UFW firewall. **Do not enable UFW firewall before allowing incoming SSH connections if you are connecting to the server remotel**y. This will block you from connecting to the Ubuntu server.

Configuring UFW firewall for SSH connections:

$sudo ufw allow ssh

Output display:

Rules updated

Rules updated (v6)

-Note: If you need to change the port from port 22, you will need to do the following to open the port. For example, if your sshd listens on port 4422, use the following command with your respective port.

$ sudo ufw allow 4422/tcp

Enabling UFW

$ sudo ufw enable

The output will be as followed. Type *Y* and *enter* to continue.

Command may disrupt existing ssh connections. Proceed with operation (y|n)? y

Firewall is active and enabled on system startup

Allowing connection on other ports

Note: Depending on the applications running on your server and your user needs, you will have to open access to other parts. See the subsection below based on your needs.

Opening port 80/8080/443- HTTP

$sudo ufw allow http

or you can use the port number 80/8080/443:

* Note: you can allow access to port ranged by replacing “80” with a range for example 7100:7200/tcp

$sudo ufw allow 80/tcp

or you can use the application profile

$sudo ufw allow 'Nginx HTTP'

Allowing Specific IP Address on Specific port/ Allowing subnets

-Note: replace the example IP address and example port with your local machine configurations.

$ sudo ufw allow from 64.63.62.61 to any port 22

To allow access to a subnet of IP addresses, use the same command as above but include the netmask after the IP address.

Allowing connections on a specific network interface

This command will allow you to replace the port and network interface needed.

For example:

$ sudo ufw allow in on eth2 to any port 3306

Denying Connections

If you need to deny connections for any reason use the following command examples with your desired port/configurations.

$ sudo ufw deny from 23.24.25.0/24

or

$ sudo ufw deny from 23.24.25.0/24 to any port 80sudo ufw deny from 23.24.25.0/24 to any port 443

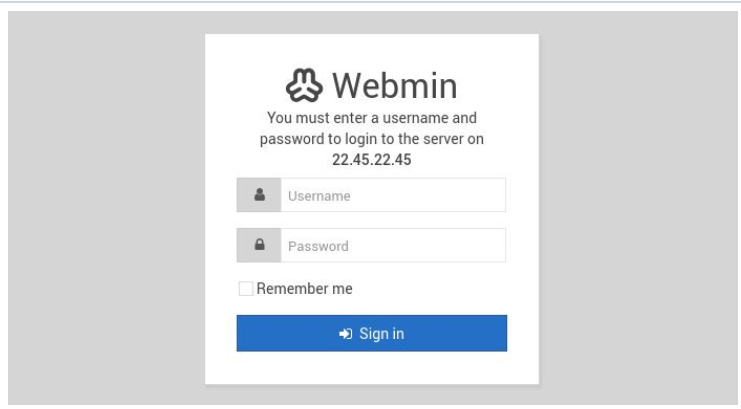
-Note: In conclusion, be sure to allow incoming connections while limiting unnecessary connections.

## 4. Accessing Webmin Web interface

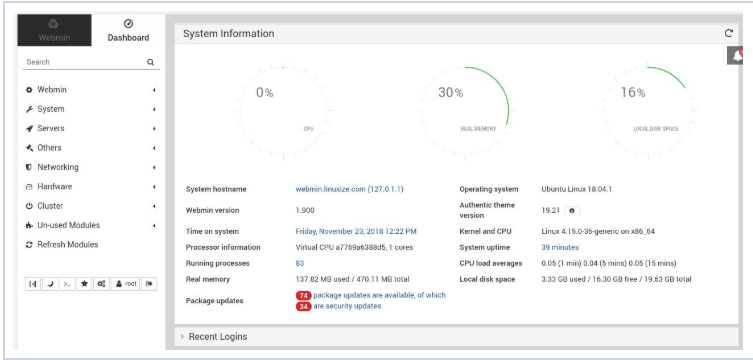
## 4.1 Access Interface

Open up your preferred browser with your servers hostname or public IP address followed by Webmin port 10000: Please note you must include the Webmin port.

https://your\_server\_ip\_or\_hostname:10000/



**-**Note: You will be prompted to log in with your root or sudo user credentials.



You can now start configuring and managing Ubuntu 18.04 server as your requirements.

## 4.2 Upgrading Webmin Installation

To upgrade your Webmin installation when new releases are published, you can use the apt package manager normal upgrade procedure:

$ sudo apt update

$ sudo apt upgrade

You have successfully installed Webmin on your Ubuntu 18.04 machine. You can now install [FTP server](https://linuxize.com/post/how-to-setup-ftp-server-with-vsftpd-on-ubuntu-18-04/) or [LAMP](https://linuxize.com/series/how-to-install-lamp-stack-on-ubuntu-18-04/)/[LEMP](https://linuxize.com/series/how-to-install-lemp-stack-on-ubuntu-18-04/) stack and start managing the services through the Webmin web interface.

To learn more about Webmin go to their documentation page at:

<http://www.webmin.com/docs.html>

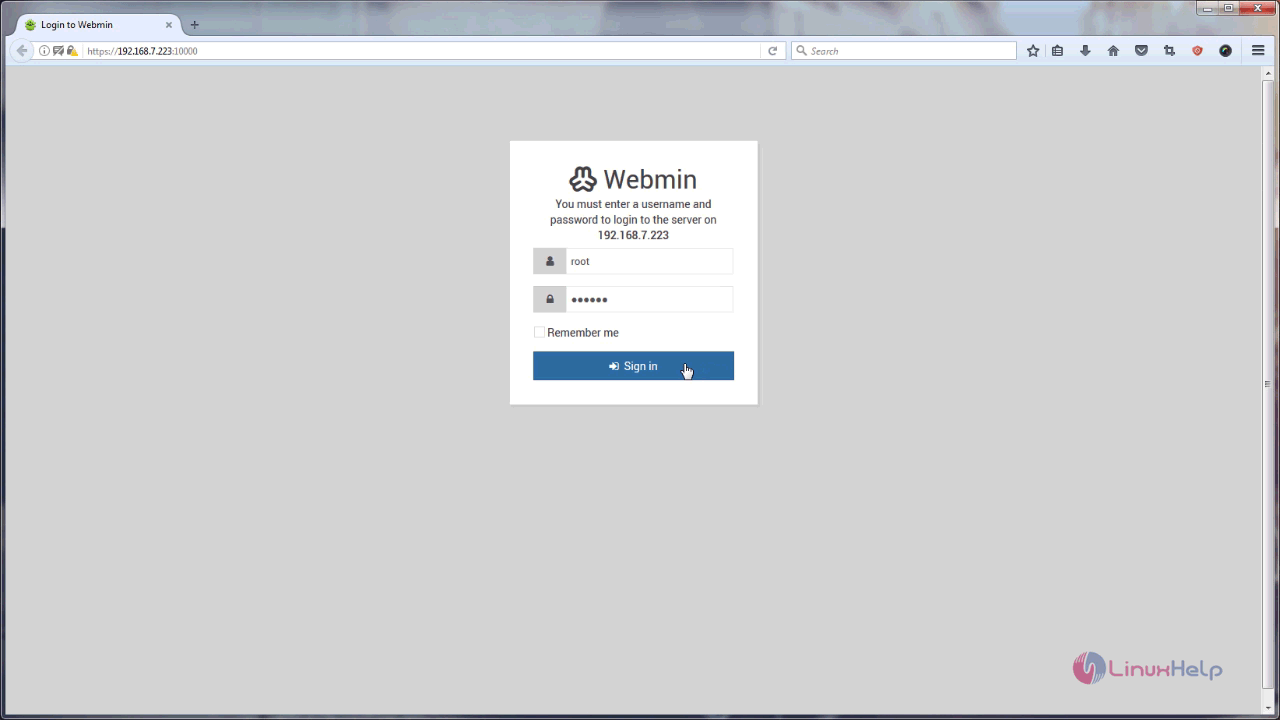
How to Install and configure BIND DNS on Webmin

## 5. Install BIND DNS with Webmin

Webmin is a Control panel application similar to CPanel which is available for open source, and it can be installed on Unix-Linux like operating systems. Webmin has many features for managing a linux server via Web browser, we can also configure a Bind DNS server through webmin for running a DNS server.

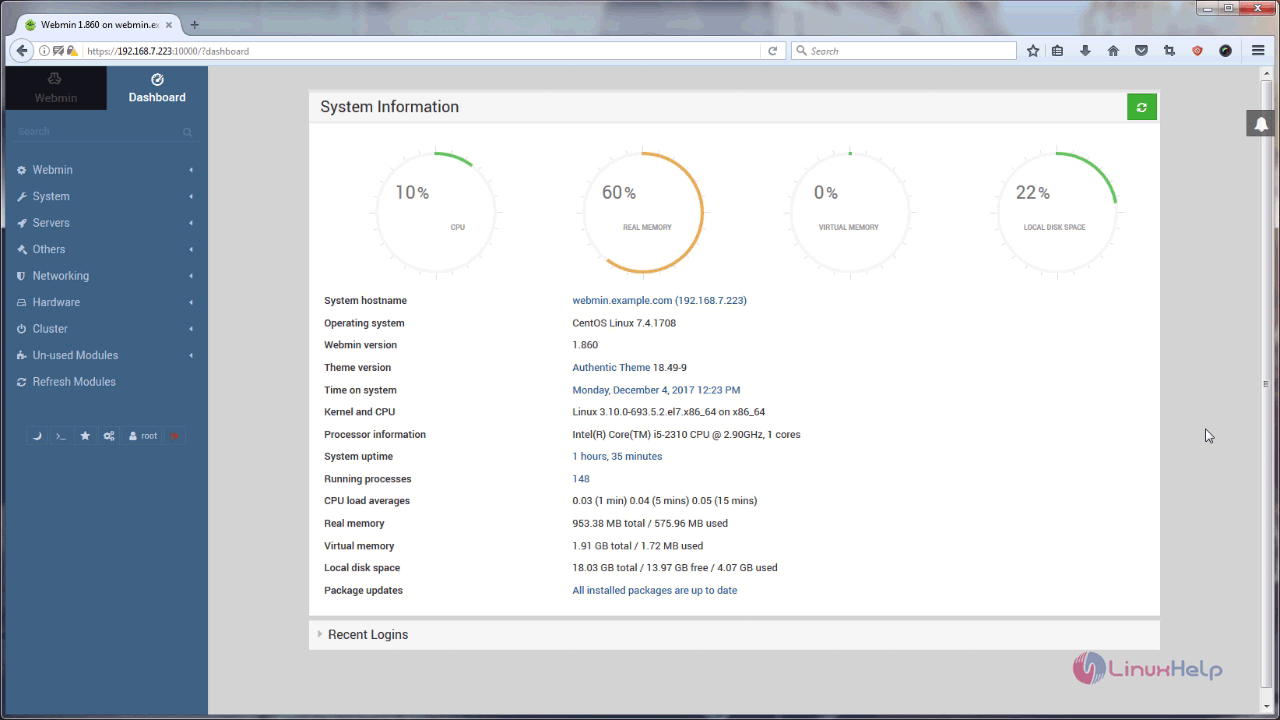
Installing Webmin

You can install Webmin on your server after that you can follow the below steps to configure Bind DNS. Once Webmin is installed you can access it through web browser using the URL for example https://192.168.7.223:10000. You can login with your server credentials.

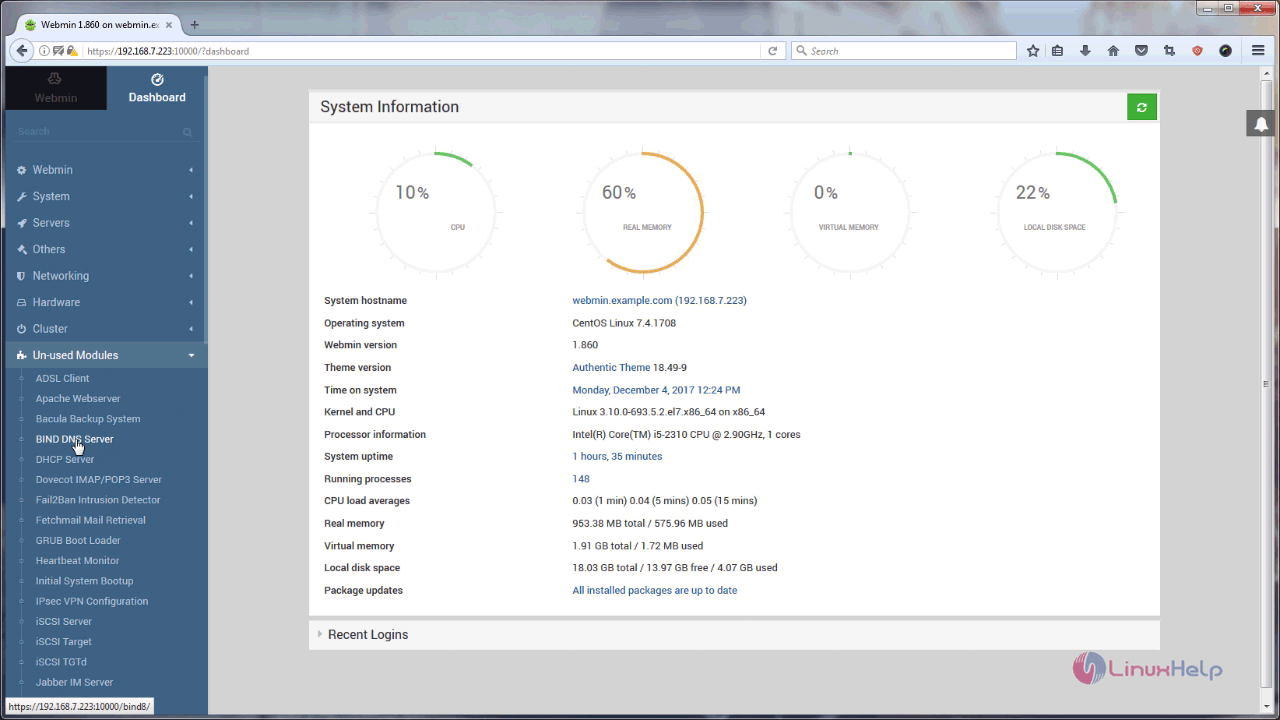


You will be taken to your Webmin Dashboard and it looks lile the one displayed in the following image.

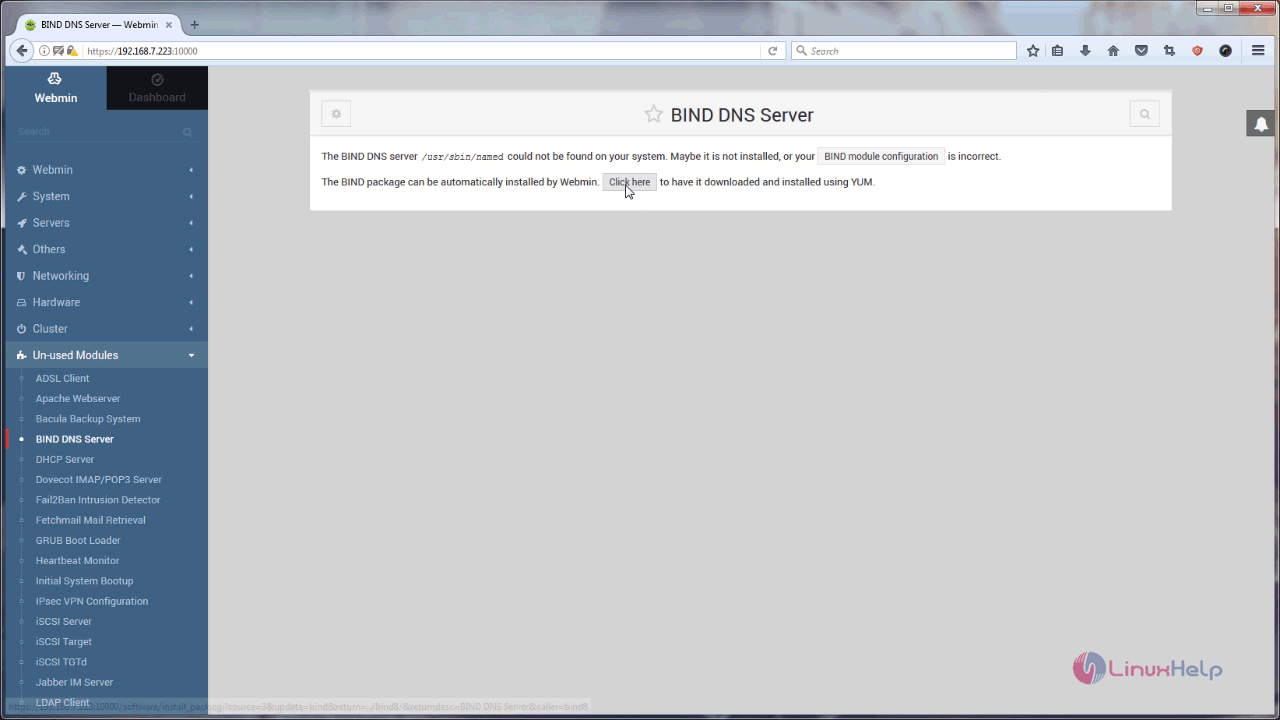
## 5.1 Accessing through web interface



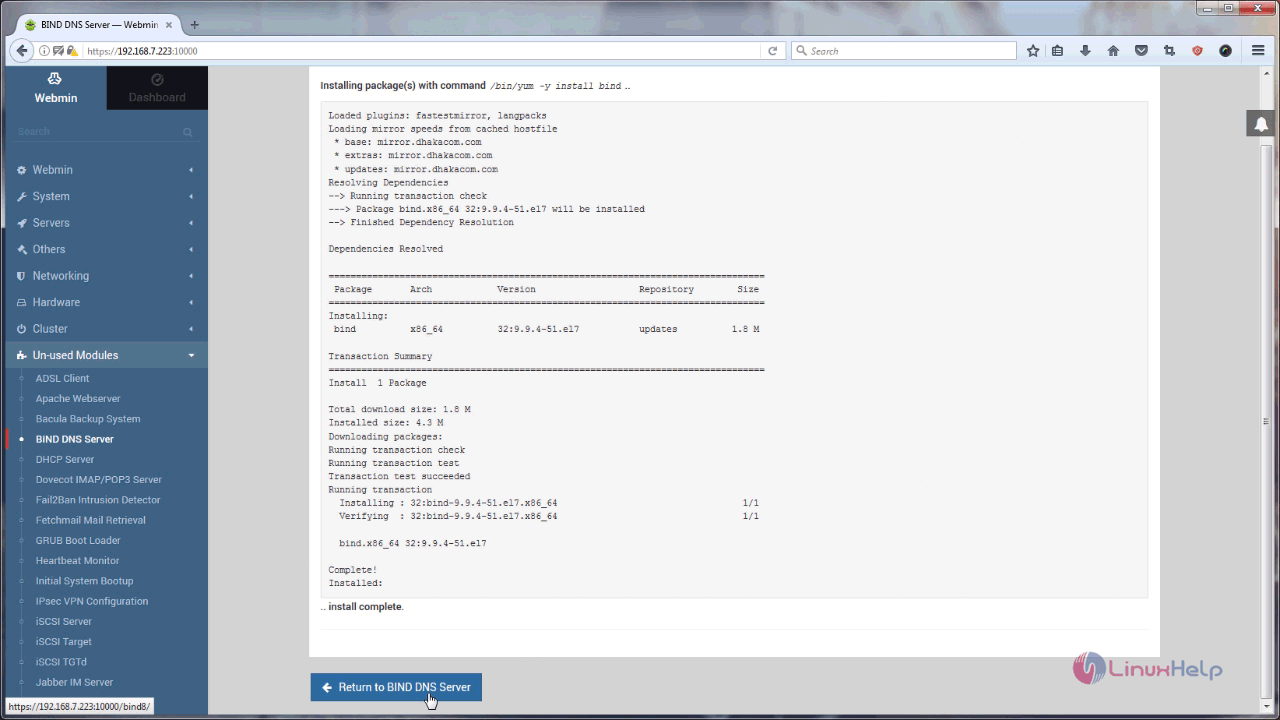
In that dashboard, you need to go to Un-used Modules in the left panel and choose BIND DNS Server.



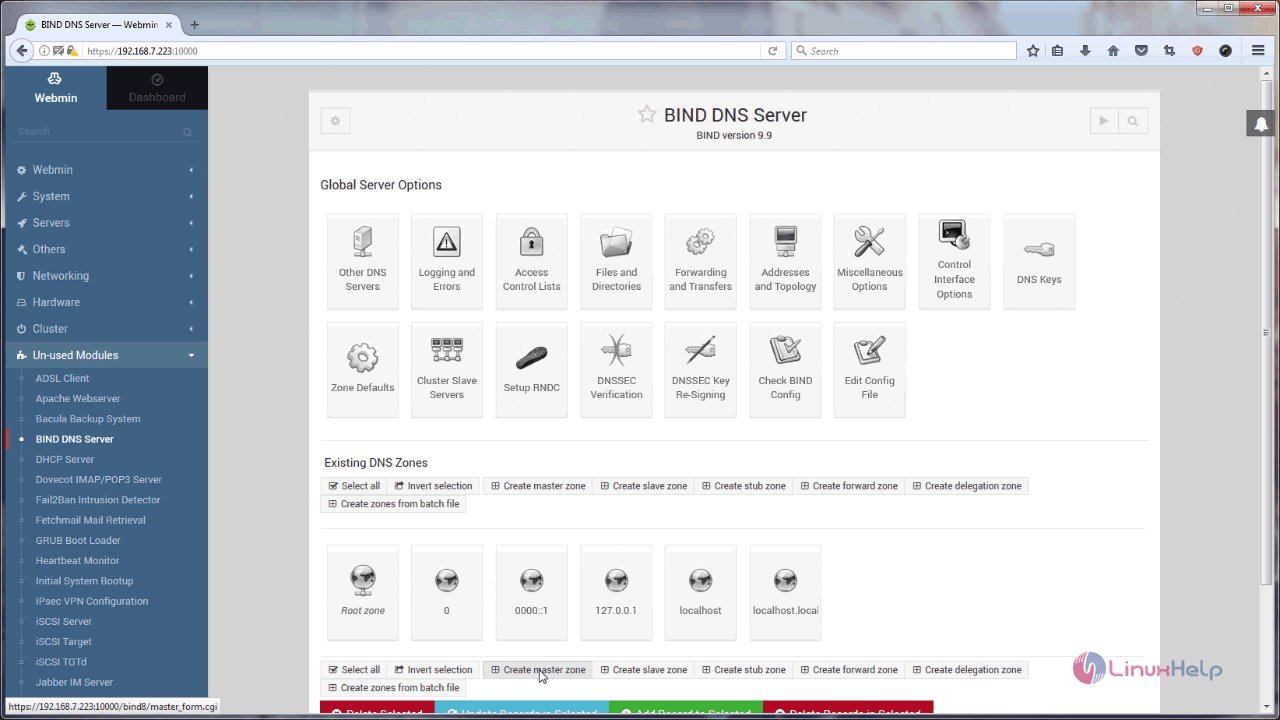
You need to install Bind DNS Server. The link to install it is given in that page. Click on it.



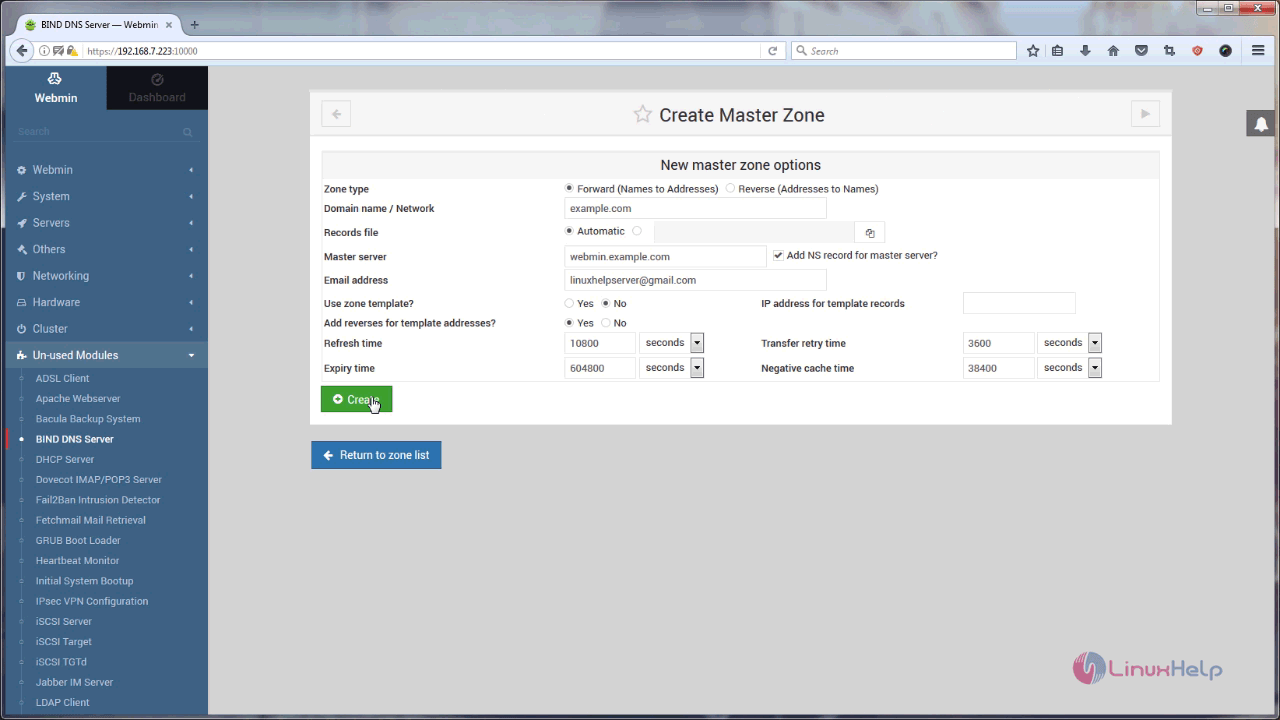
The installation procedure has started. Once it is done, click on Return to BIND DNS server.



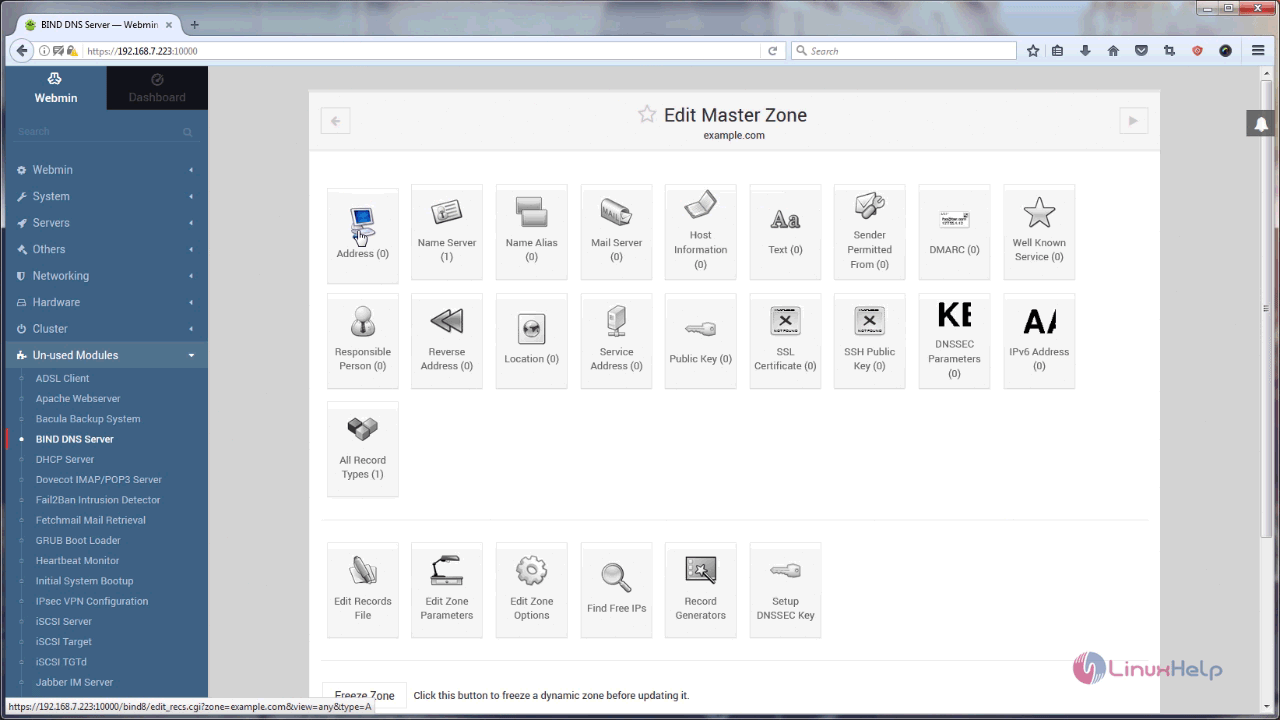
Now Bind is installed, now let’ s configure master zone for the domain webmin.example.com.



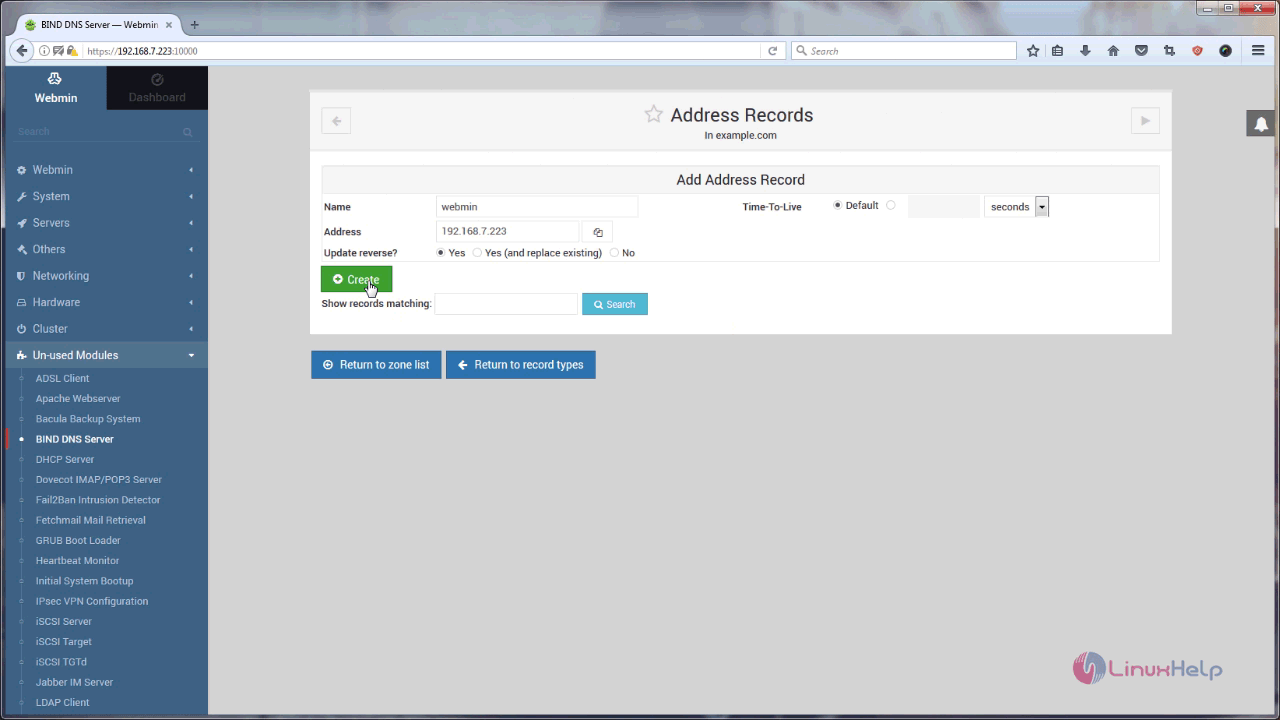
For that, you need to edit your zone details.



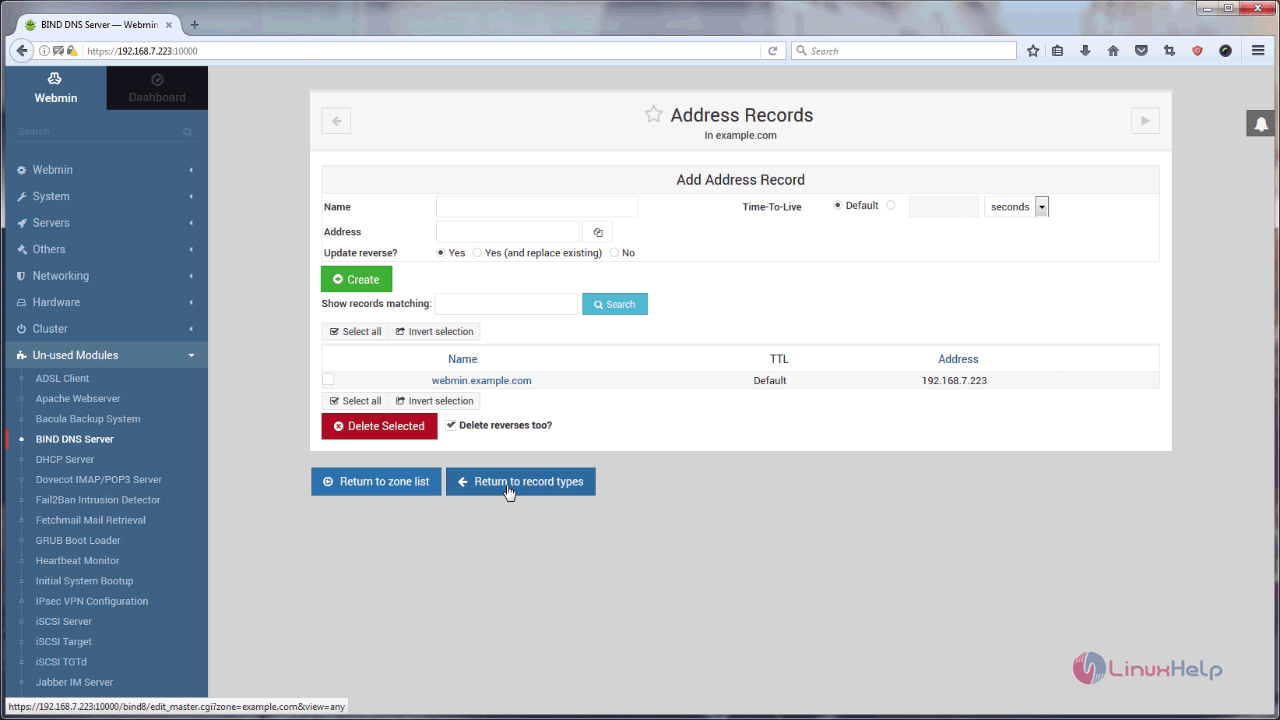
After that, you should configure A (Address) record.



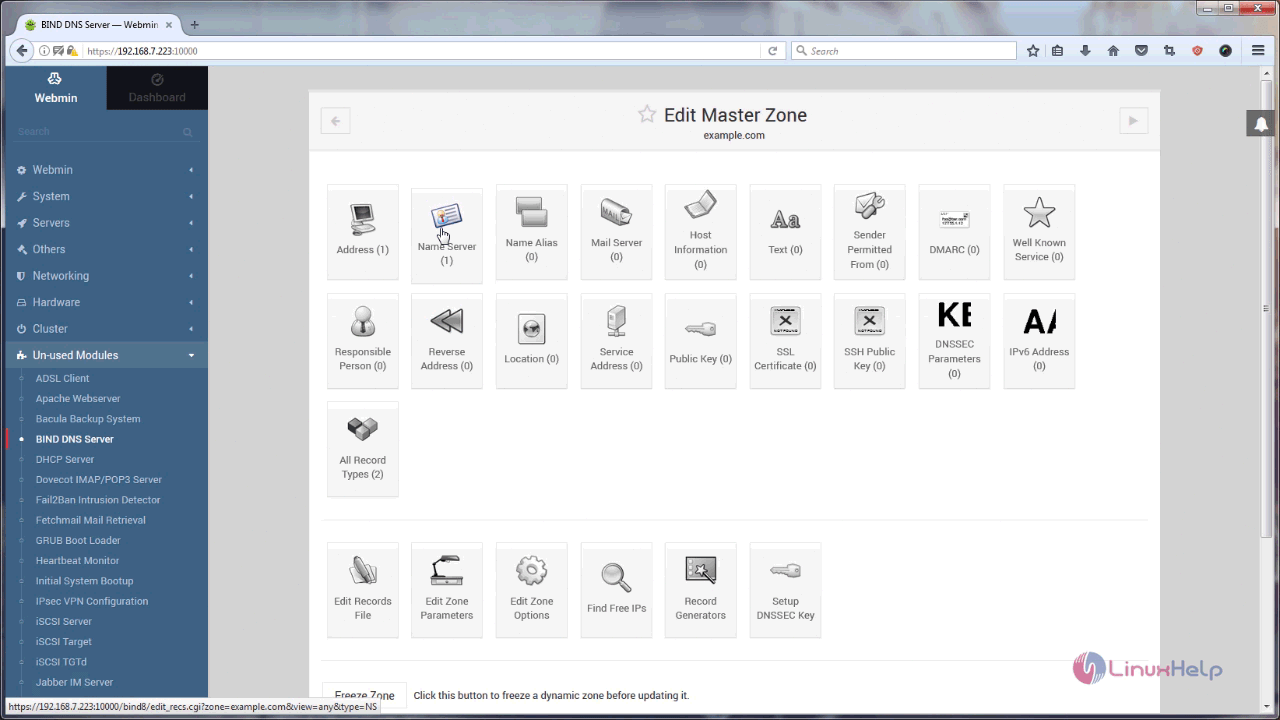
Then create an Address Record. Type your hostname and ip address for your domain in that.



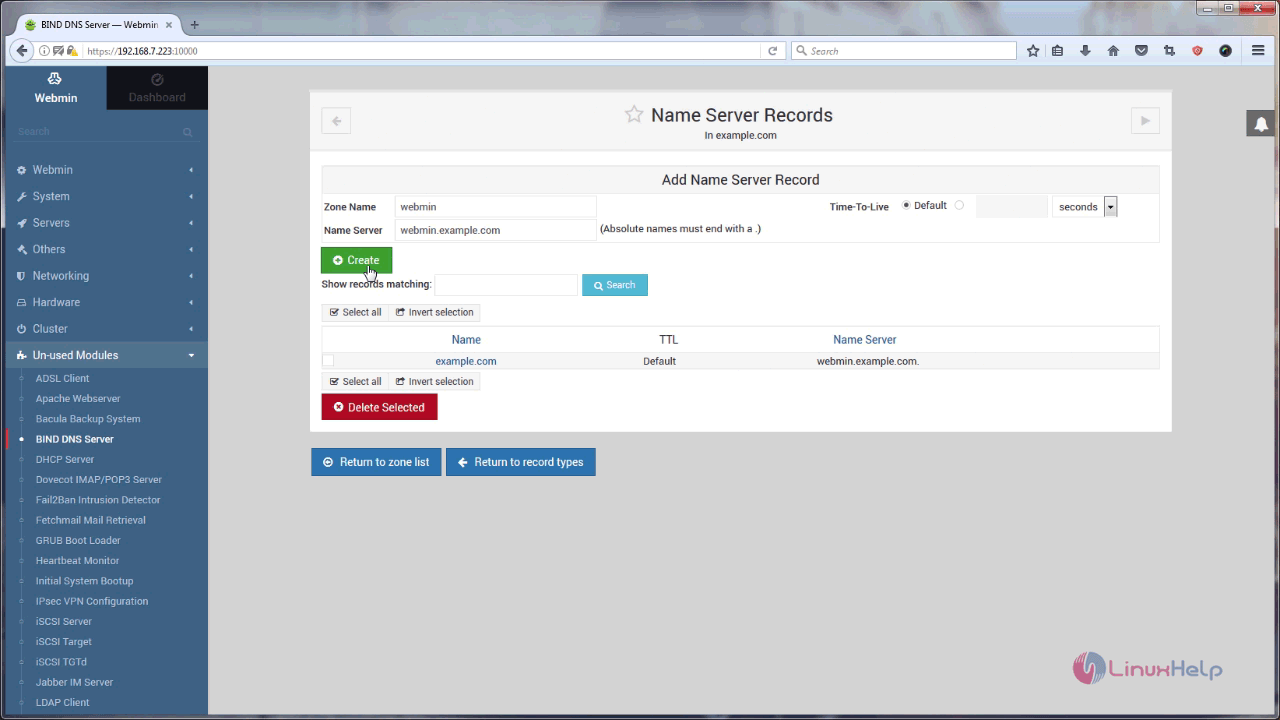
Once it is done, click on the Return to record types.

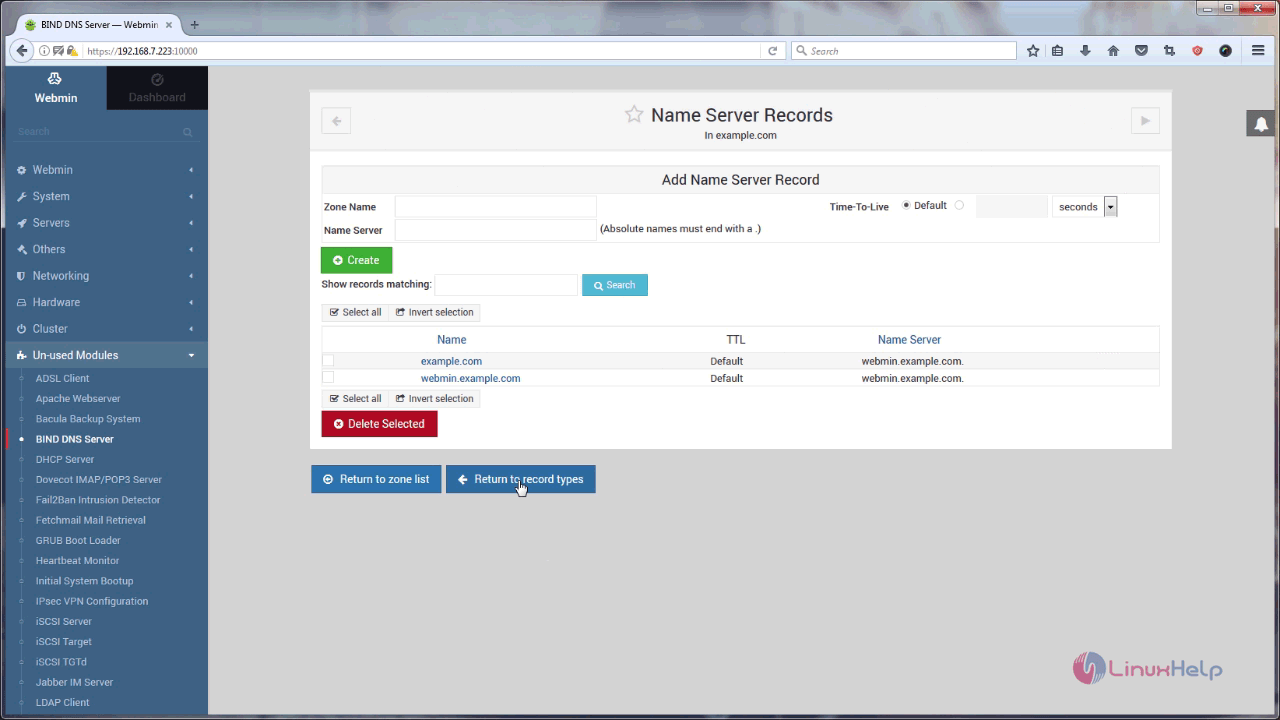


After that, you need to create Name Server Entry. So choose that icon from your Master Zone.

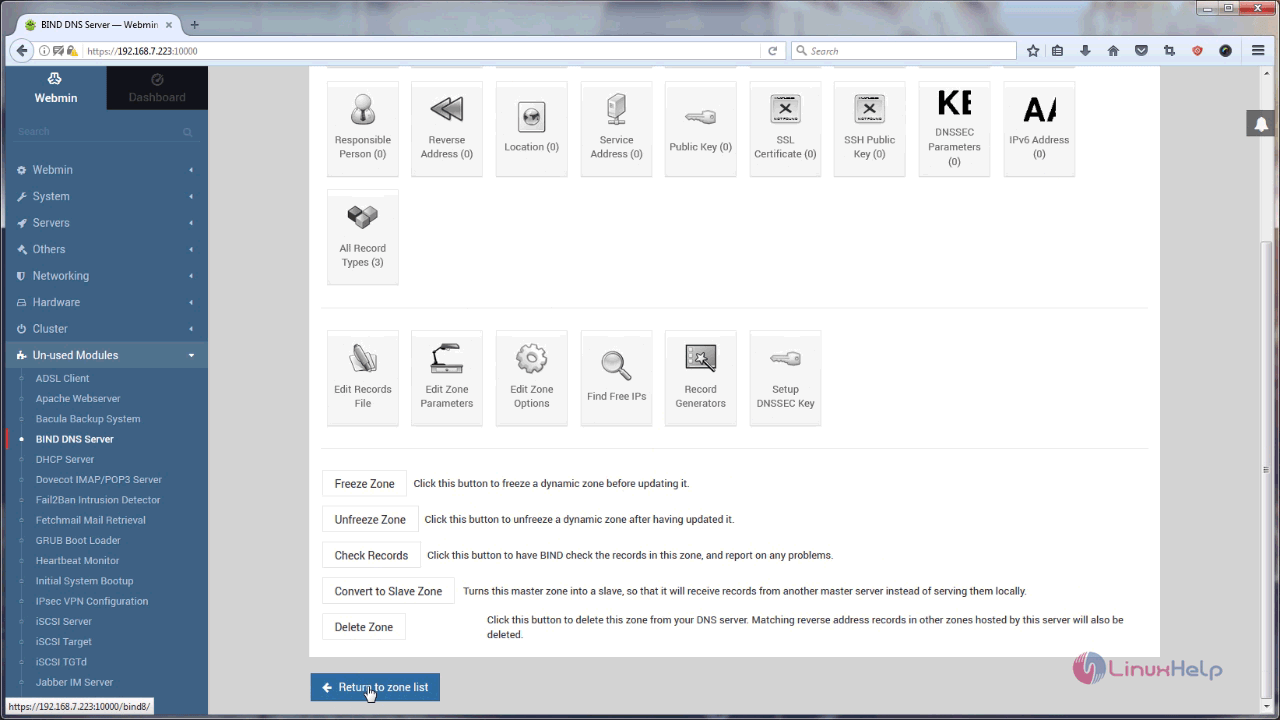


You need to fill the details respective with your domain. Give them all and click on the Create button.

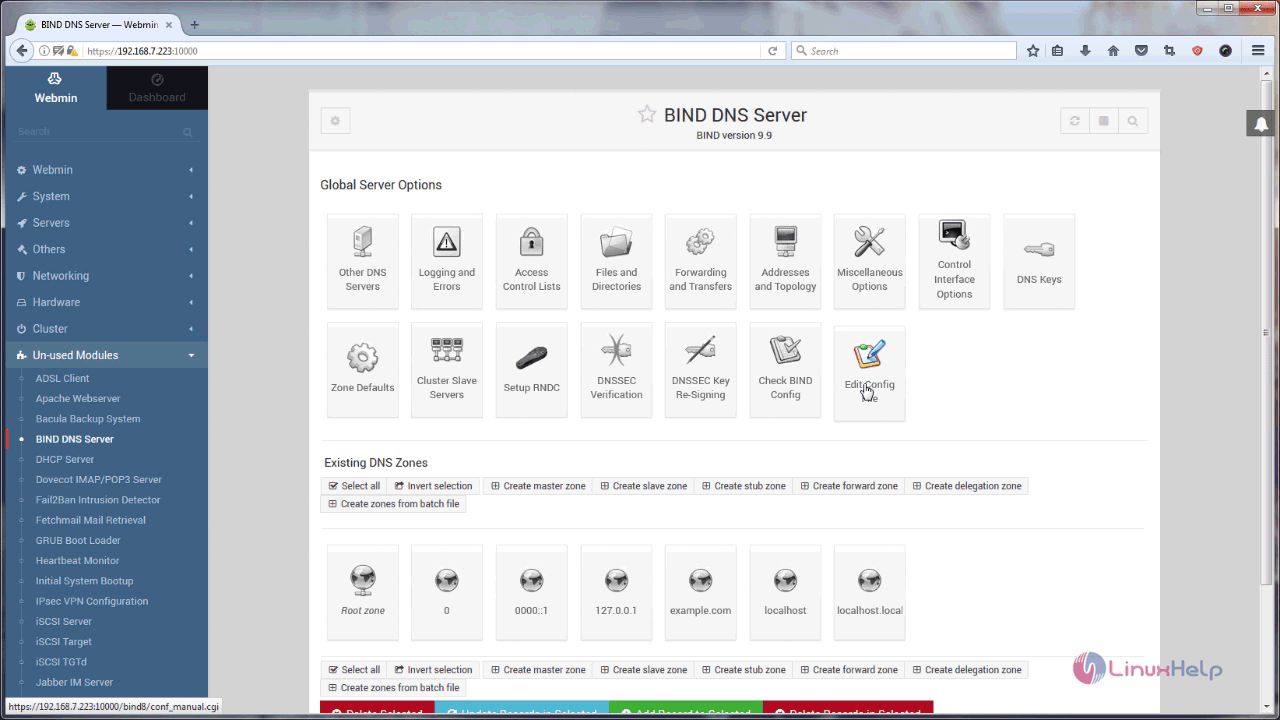




Go back to your Master Zone and to your Zone list.



Now we need to edit Bind DNS Configuration, for that you can use terminal or just choose Edit config with this.

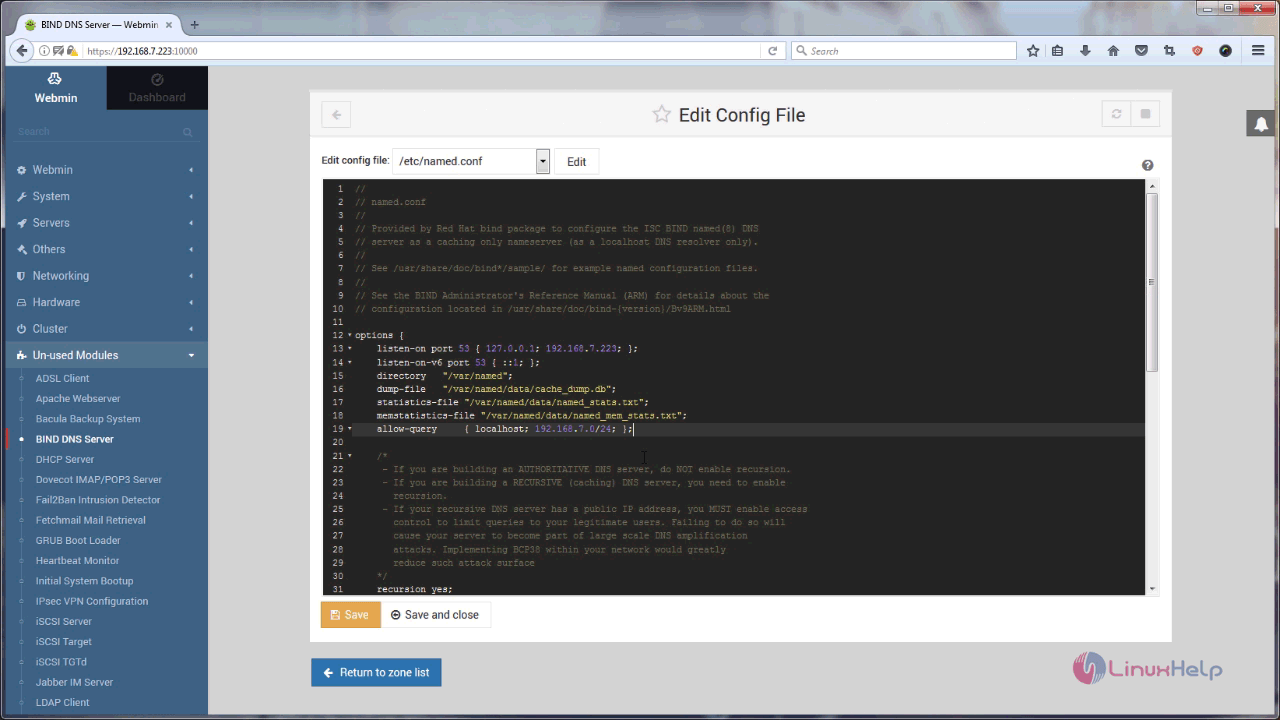


In Bind configuration you need to type the ip address of the DNS server on listen-on-port tag and also you need to type allow-query entry for any particular ip address or particular network.

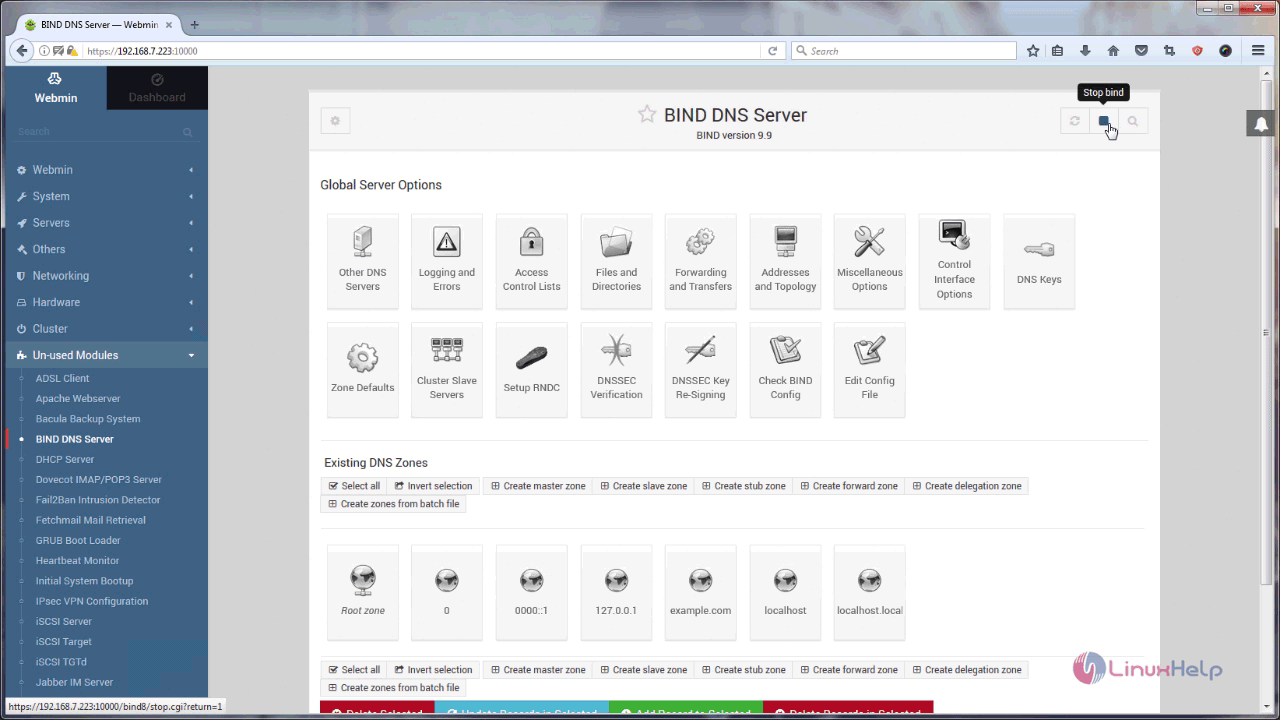
For example

listen-on-port 53 { 127.0.0.1 192.168.7.223 }  
allow-query { localhost 192.168.7.0/24 }

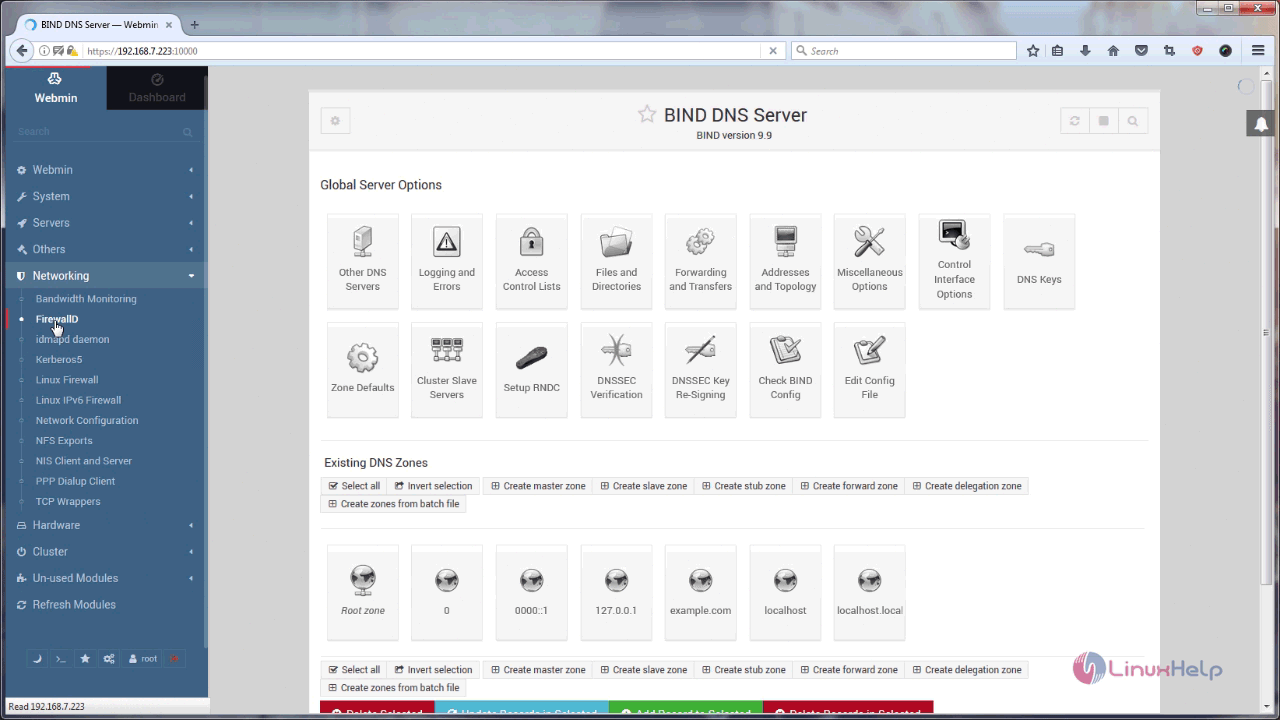
Now type the above entry in your configuration and save the file.



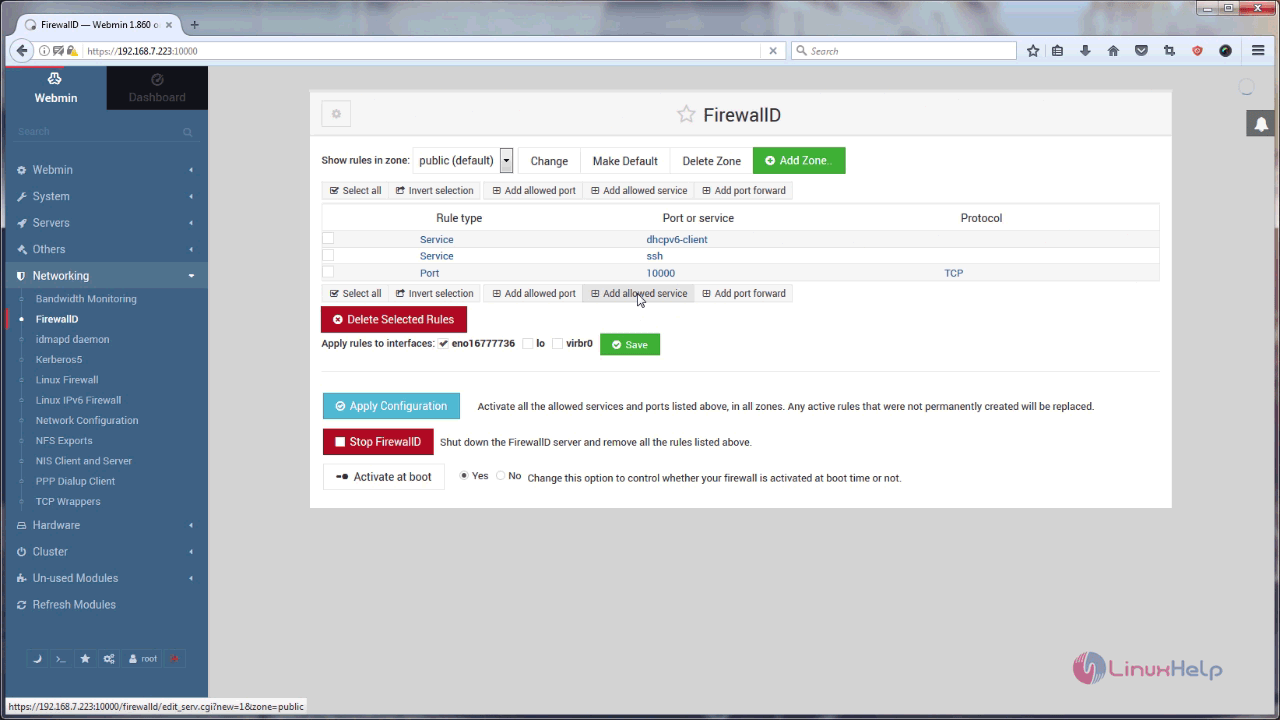
Now restart the Bind Service from command line or from browser by using the control buttons on top right side corner as shown in following image.



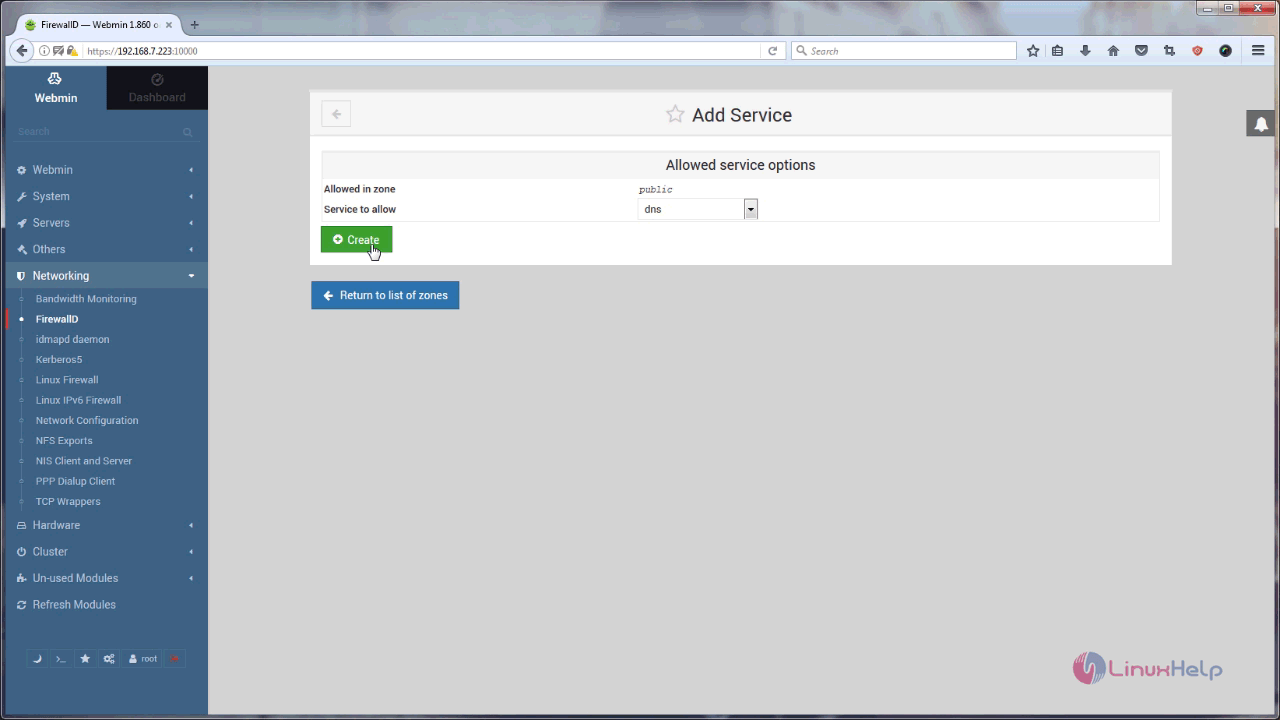
Now, make firewall settings. So, go to Networking in the left panel and choose FirewallD.



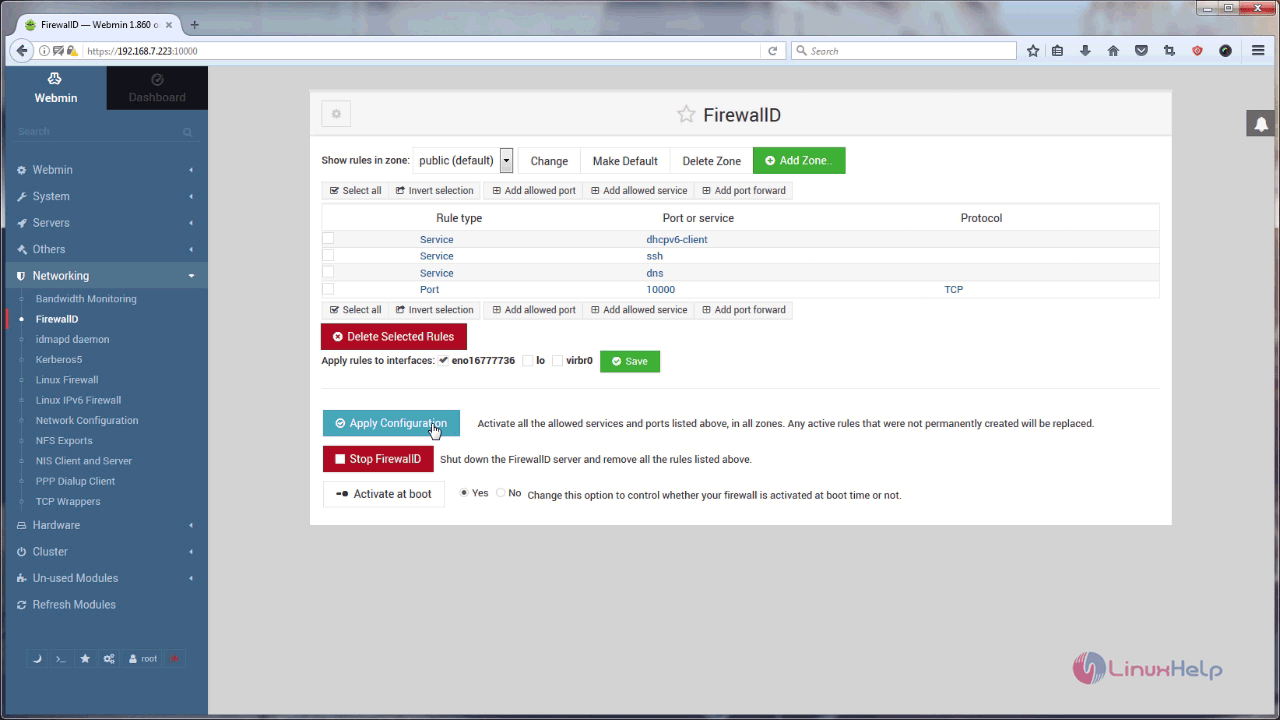
You should now add the allowed service.



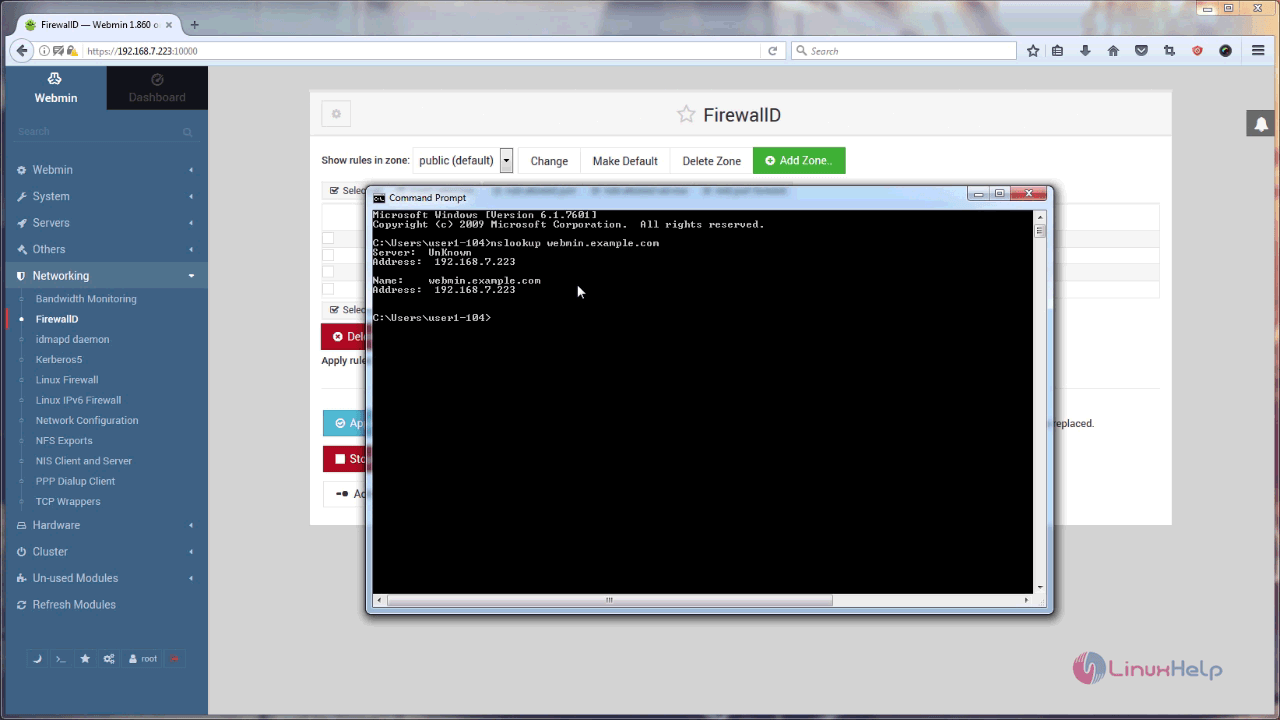
Choose DNS service in drop down menu and click Create button to create an allowed service.



Also, you need to apply changes. So, click on the Apply Configuration button.



Now BIND DNS is configured and running you, can check it by adding the DNS IP address to your network interface, and then you can check if internet service is working or not. You can query for the respective domain that you have created in bind dns webmin.example.com via nslookup command.



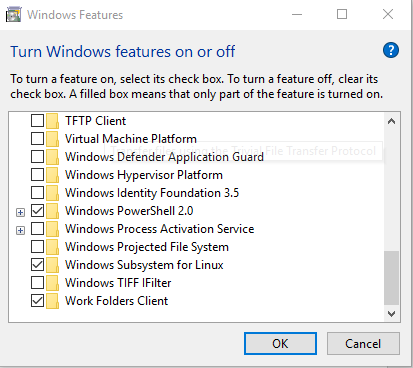
Now you can see the nslookup command give the answer (IP address) for the query (webmin.example.com) domain. With this, the configuration on BIND DNS comes to an end.

## 6. Extended Content: Installing Ubuntu For Windows 10 And OS X

-Note: For ansible installation, there is not a ansible.exe file available or an Installer. We need to install a Linux Subsystem to get ansible running.

In order to run ansible in Windows, we need to install a Linux distribution. Windows 10 has support for this, without having to install VMWare, docker, or dual booting a device. You may have to enable the Linux subsystem in Windows first.

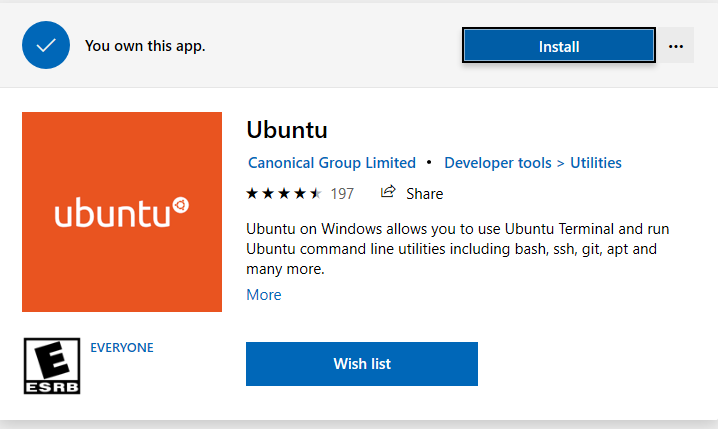
Search for “Turn Windows features on or off”

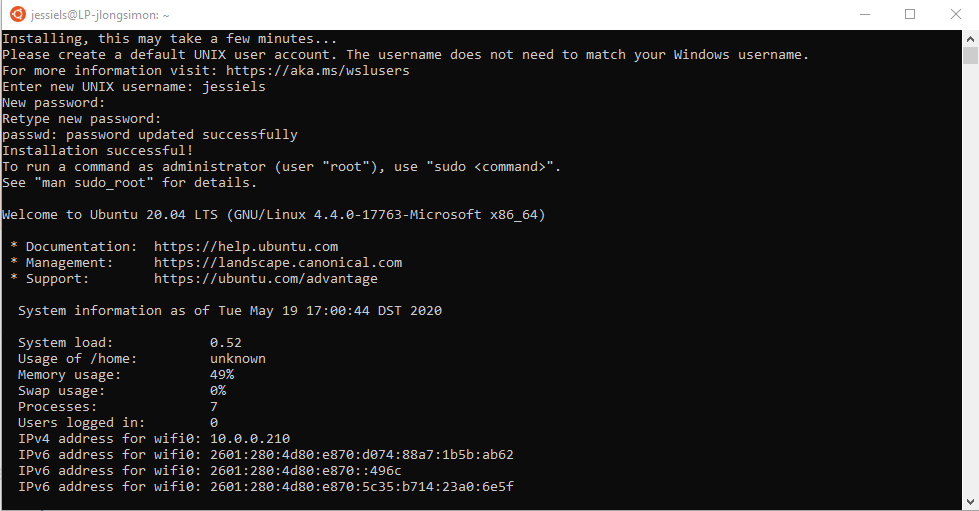
* Select “Windows Subsystem for Linux”

6.1 Install for Windows

Go to the Microsoft Store to download Ubuntu App

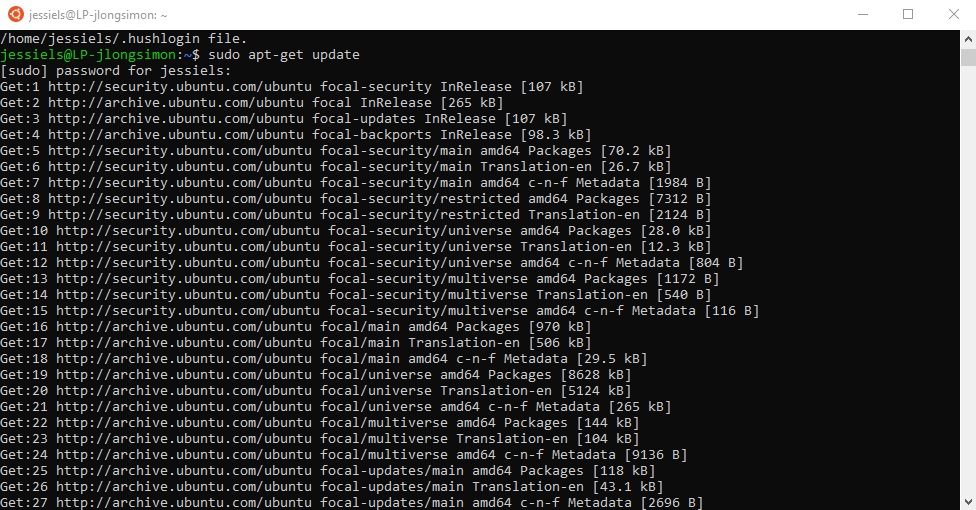
<https://www.microsoft.com/en-us/p/ubuntu/9nblggh4msv6?activetab=pivot:overviewtab>



Launch application once installed and create a username and password as shown belo

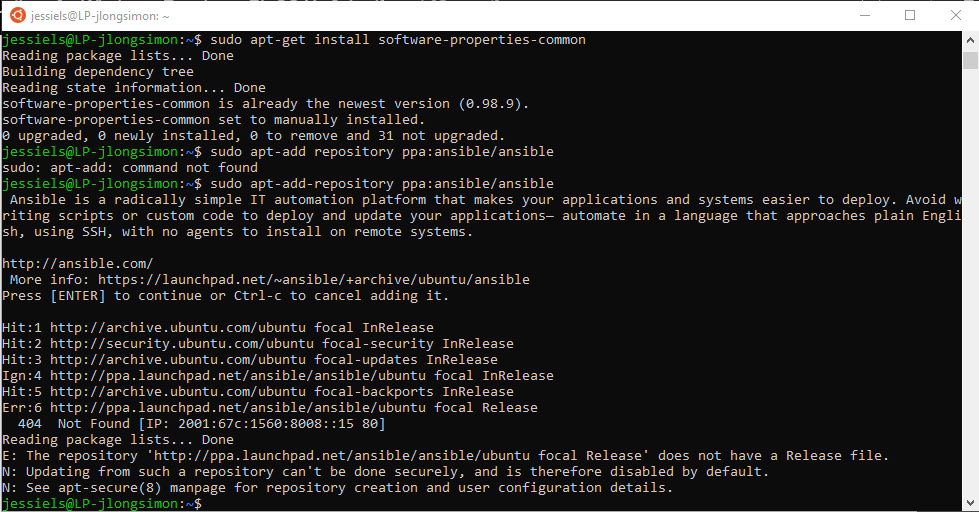
In your terminal, which if you open a command prompt and type in BASH you can get into at any time, type in the command “sudo apt-get update”.

“sudo” - Run the command with Privileges

“apt-get update” – Apt is a package software manager for Ubuntu. Update signifies that we are going to Update the installable software list from apt sources

* To install a new repository to where the latest version of ansible is, we are going to add a source.
* Type in:

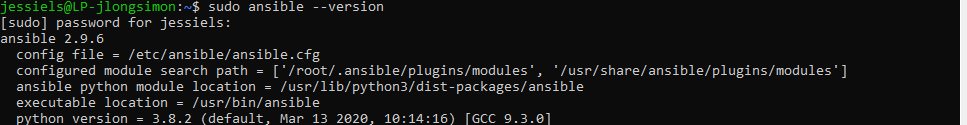
“**sudo apt-add-repository ppa:ansible/ansible**”



* Check to see if Ansible is at least version 2.7.8 or greater now, we can use the command “**sudo apt show ansible**” to see the entire package
* Install it with “**sudo apt-get install ansible**”
* Press Y to continue to install it.

Type in the following to verify that ansible is installed

**sudo ansible --version**



* Download the ICX Ansible software from the Ruckus support site. The current file name is “icx\_ansible\_modules\_v1.0.tar.gz”
* You can access the Windows files from the Ubuntu Terminal window. If you have saved the file in your Windows download folder, you can visit that folder in the following location:

**cd /mnt/c/Users/<username>/Downloads**

* You can move the file, leave it where it is etc… but I am going to create a new directory and copy the file into it. The new directory will be in my downloads folder. I will continue to use Linux commands to create the directory and move the files:

**mkdir Ansible-ICX** #create the Ansible-ICX directory

**mv icx\_ansible\_modules\_v1.0.tar.gz Ansible-ICX** #move the downloaded file

**cd Ansible-ICX** #change directory to Ansible-ICX folder

Run the easy\_install file by using the following command:

**sudo ./easy\_install**

Enter in your password

* The following files are installed in the /usr/lib/python2.7/dist-packages/ansible folder
* These modules are referenced as part of the playbooks you will use to interact with the ICX switches
* In the ansible icx folder, there is a file called icx\_examples.tar.gz. This file contains all the example playbooks that we will use in the next section. Expand this file using the command
* tar xvf icx\_examples.tar.gz

7. Resources

<http://www.webmin.com/>

<https://ubuntu.com/tutorials/tutorial-install-ubuntu-desktop#1-overview>

<https://www.digitalocean.com/community/tutorials/how-to-configure-bind-as-a-private-network-dns-server-on-ubuntu-14-04>

<https://docs.ansible.com/ansible/latest/installation_guide/intro_installation.html>