Initial Server Setup with Ubuntu 12.04

Authored by: ***ASPHostServer Administrator*** *[asphostserver@gmail.com]*
Saved From: *http://faq.asphosthelpdesk.com/article.php?id=160*

The Basics

When you first begin to access your fresh new server, there are a few early steps you should take to make it more secure. Some of the first tasks required on a virtual private server can include setting up a new user, providing them with the proper privileges, and configuring SSH.

Step One—Root Login

Once you know your IP address and root password, login as the main user, root.

It is not encouraged to use root on a server on a regular basis, and this tutorial will help you set up an alternative user to login with permanently.

ssh root@123.45.67.890

The terminal will show:

The authenticity of host '69.55.55.20 (69.55.55.20)' can't be established.

ECDSA key fingerprint is 79:95:46:1a:ab:37:11:8e:86:54:36:38:bb:3c:fa:c0.

Are you sure you want to continue connecting (yes/no)?

Go ahead and type yes, and then enter your root password.

Step Two—Change Your Password

Currently your root password is the default one that was sent to you when you registered your server instance. The first thing to do is change it to one of your choice.

passwd

Step Three— Create a New User

After you have logged in and changed your password, you will not need to login again as root. In this step we will make a new user and give them all of the root capabilities.

You can choose any name for your user. Here I"ve suggested Demo

adduser demo

After you set the password, you do not need to enter any further information about the new user. You can leave all the lines blank if you wish

Step Four— Root Privileges

As of yet, only root has all of the administrative capabilities. We are going to give the new user the root privileges.

When you perform any root tasks with the new user, you will need to use the phrase "sudoâ€• before the command. This is a helpful command for 2 reasons: 1) it prevents the user making any system-destroying mistakes 2) it stores all the commands run with sudo to the file "/var/log/secure' which can be reviewed later if needed.

Let"s go ahead and edit the sudo configuration. This can be done through the default editor, which in Ubuntu is called "nano"

visudo

Find the section called user privilege specification.

It will look like this:

User privilege specification

root ALL=(ALL:ALL) ALL

Under there, add the following line, granting all the permissions to your new user:

demo ALL=(ALL:ALL) ALL

Type "cntrl x" to exit the file.

Press Y to save; press enter, and the file will save in the proper place.

Step Five— Configure SSH (OPTIONAL)

Now it"s time to make the server more secure. These steps are optional. Please keep in mind that changing the port and restricting root login may make logging in more difficult in the future. If you misplace this information, it could be nearly impossible.

Open the configuration file

nano /etc/ssh/sshd_config

Find the following sections and change the information where applicable:

Port 25000

Protocol 2

PermitRootLogin no

We"ll take these one by one.

Port: Although port 22 is the default, you can change this to any number between 1025 and 65536. In this example, I am using port 25000. Make sure you make a note of the new port number. You will need it to log in in the future. This change will make it more difficult for unauthorized people to log in.

PermitRootLogin: change this from yes to no to stop future root login. You will now only be logging on as the new user.

Add these lines to the bottom of the document, replacing *demo* in the AllowUsers line with your username. (AllowUsers will limit login to onlythe users on that line. To avoid this, skip this line):

UseDNS no

AllowUsers demo

Save and Exit

Step Six— Reload and Done!

Reload SSH, and it will implement the new ports and settings.

reload ssh

To test the new settings (don"t logout of root yet), open a new terminal window and login as your new user.

Don"t forget to include the new port number.

ssh -p 25000 demo@123.45.67.890

Your prompt should now say:

[demo@yourname ~]$