Initial Server Setup with Arch Linux

Authored by: ***ASPHostServer Administrator*** *[asphostserver@gmail.com]*
Saved From: http://faq.asphosthelpdesk.com/article.php?id=214

The Basics

When you first begin to access your fresh new virtual server, there are a few early steps you should take to make it more secure. Some of the first tasks 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. You can log in through Terminal on a Mac computer and putTy on the PC

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

```
ssh root@123.45.67.890
```

The terminal will show:

```
The authenticity of host '123.45.67.890 (123.45.67.890)' 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 control panel. 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 on the server.

You can choose any name for your user. Throughout this tutorial, I will use the name exampleuser

```
useradd -m exampleuser
```

The "-m" makes a home directory for your user.

To change your new user's password, use

```
passwd exampleuser
```

Step Four— Root Privileges

As of yet, only root has all of the administrative capabilities on the virtual server. 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. Vi, the text editor used for this file, does not recognize arrow keys. Move down using "jâ€•, up with "kâ€•, left with "hâ€•, and right with "lâ€•. Additionally, you can begin editing the text by pressing "aâ€•, and delete text by pressing Escape and then "xâ€•

```
visudo
```

Find the section called user privilege specification.

It will look like this:

```
# User privilege specification
root ALL=(ALL) ALL
```

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

```
exampleuser ALL=(ALL) ALL
```

Press escape and then Shift ZZ to save and exit the file.

Step Five— Configure SSH

Now it"s time to make the server more secure.

Open the configuration file

```
nano /etc/ssh/sshd_config
```

Find the following sections and change the information where applicable. Be sure to uncomment the lines as well, otherwise the changes will not take effect. (You can find words and phrases in the file by pressing Control-W)

```
Port 25000
PermitRootLogin no
```

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.

Step Six— Reload and Done!

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

```
systemctl restart sshd
```

To test the new settings (Do Not Log Out of Root!), open a new terminal window and login as your new user.

Don"t forget to include the new port number.

```
ssh -p 25000 exampleuser@123.45.67.890
```

Your prompt should now say:

```
[exampleuser@yourhostname ~]$
```

Once you know you can login into your virtual private server with your new user, you can exit out of root.