Creating Apache webserver VM in Proxmox VE

In this scenario, we will be creating a webserver using ubuntu server LTS 24 ISO hosted on our proxmox server

On top right corner of your proxmox interface click on Create VM



Give your VM a name and for VM ID is has to be unique. Each resource need to have a unique ID. You can decide to leave 1xx for VMs, 2xx for containers etc for easier reference.

A screenshot of a computer

Description automatically generated

Now under OS you can see there are no ISOs for selection. That’s alright because we have not downloaded/uploaded any ISO yet

A screenshot of a computer

Description automatically generated

If you already have the iSO file can upload here:

A screenshot of a computer

Description automatically generated

Click on upload and let it load. Do not close the window until it is completed.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Once completed a window will be shown stating task ok. Can close it.

A screenshot of a computer

Description automatically generated

Now that we see our ISO image in storage

A screenshot of a computer

Description automatically generated

We can proceed to create the VM again

A screenshot of a computer

Description automatically generated

Under System tick this and leave the rest as default then click on Next. For reference on VirtIO SCSI single vs VIRTIO SCSI (<https://forum.proxmox.com/threads/default-setting-of-scsi-controller-is-now-virtio-scsi-single.121491/>)

A screenshot of a computer

Description automatically generated

In earlier setup we removed the unused lvm storage and allocated all in 1 disk and hence it does not show anything here. We can create separately for VMs

A screenshot of a computer

Description automatically generated

Go back to your node click on shell then run lvs and vgs

A screenshot of a computer

Description automatically generated

From here we see that total size of disk is around 476GB

To check exactly how much root is using you can run df -h and see under mounted on / or run df -h / directly. This reflects we are only using 5.5Gb of space for file systems (e.g. like C drive in windows).

A screenshot of a computer program

Description automatically generated

Go to Datacenter > Storage > Add > Directory

A screenshot of a computer

Description automatically generated

Then directory can key in /var/lib/vz and also assign an ID (folder name) and for content select Disk image, container template, container and snippets. Click on Add.

A screenshot of a computer

Description automatically generated

(JUST FOR KNOWLEDGE): if you tick on advanced you can see the options below

A screenshot of a computer

Description automatically generated

They basically means: say you create a VM with a 100GB virtual disk:

1. Default/Thin Provisioning:

* Only allocates space as the VM actually uses it
* If VM only uses 20GB, only 20GB of actual disk space is used
* Space is allocated "on-demand"

1. Full Pre-allocation:

* Reserves the entire 100GB immediately
* All space is allocated even if VM isn't using it

Now back to our VM creation

For the Disks part, tick discard to allow VM to inform host when blocks are not in use anymore. If you are using SSD, can tick it to free up unused space.

32 Gb max storage is enough for web server and since we use default allocation those unused disk space for our host PC won’t be taken up.

Click on next

A screenshot of a computer

Description automatically generated

For cores can crank it up a little to 2 and adjust more if needed once it starts running in prod. Change type to host so that it passes through your actual CPU features.

A screenshot of a computer

Description automatically generated

Leave it as 2Gb RAM and click on next

A computer screen shot of a computer

Description automatically generated

For network can leave it as default and click on next

A screenshot of a computer

Description automatically generated

At confirmation screen click on Finish

A screenshot of a computer

Description automatically generated

Before we begin, you can click on your VM and select options > change start at boot to Yes first

A screenshot of a computer

Description automatically generated

Tick this and click ok

A screenshot of a computer

Description automatically generated

Now click on your VM and select console > start now

A screenshot of a computer

Description automatically generated

Now we can begin installation of ubuntu server for our VM. Choose language and press Enter.

A screenshot of a computer

Description automatically generated

If they prompted to download new update for your ubuntu server installer can skip first by choosing continue without updating

A screenshot of a computer

Description automatically generated

Choose your keyboard layout and select done and press enter

A screenshot of a computer

Description automatically generated

Leave as default and press enter

A screenshot of a computer

Description automatically generated

Choose your ethernet adapter but since we only have 1 can leave as is and press enter

A screenshot of a computer

Description automatically generated

If no proxy server can leave as is and press enter

A screenshot of a computer

Description automatically generated

Can just press enter once they shown this mirror location passed tests

A screenshot of a computer

Description automatically generated

Leave as default and select done

A screenshot of a computer

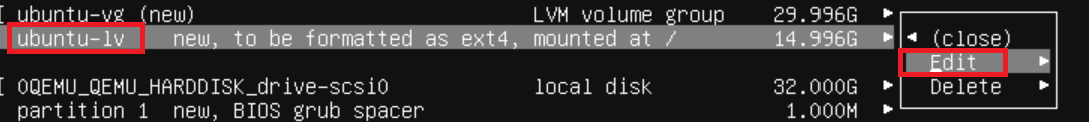
Description automatically generated

As you can see in this default config the free space is 15G. Root is currently using 15GB as well and more than enough to setup web server but to maximise it we can set ubuntu-lv to 29.996G. /root drive is like C drive in windows context.

A screenshot of a computer

Description automatically generated

Select ubuntu-lv and select edit



Set to 29.996G and select Save and press enter

A screenshot of a computer

Description automatically generated

Now select done and enter now that we have set the root space to be same as the volume group so that it can use as much space as possible.

A screenshot of a computer

Description automatically generated

Select continue

A screenshot of a computer error

Description automatically generated

Fill in the blanks accordingly as per your preference and press done.

A screenshot of a computer

Description automatically generated

Keep it as skip for now and continue

A screenshot of a computer

Description automatically generated

Press spacebar to keep openssh server as selected and select Done

A screenshot of a computer

Description automatically generated

Do not install anything. Just press tab to skip straight to the “Done” selection

A computer screen shot of a black and orange screen

Description automatically generated

Now let it load for a few minutes to finish installation

A screenshot of a computer program

Description automatically generated

Once done it will show the reboot now option at the bottom of the screen. Select Reboot Now and press Enter.



If you see these errors can just press enter. They are just trying to unmount the cdrom drive

A screenshot of a computer

Description automatically generated

Now we are setup!

A black screen with white text

Description automatically generated

Login and run packages updates first

A screen shot of a computer

Description automatically generated

To make it easier to run commands you can open your own command prompt/terminal and then ssh into your webserver-vm via its’ ip.

Install qemu-guest-agent

A black screen with white text

Description automatically generated

Install the following as well if you want: tldr, plocate,tree, net-tools, network-manager

Now check the status of the qemu-guest-agent and run sudo systemctl start qemu-guest-agent.service if it is not running. If you try to run sudo systemctl enable qemu-guest-agent.service and they are not letting you to, you can go back to options and check that it is set to enabled

A computer screen shot of a computer

Description automatically generated

A screenshot of a computer

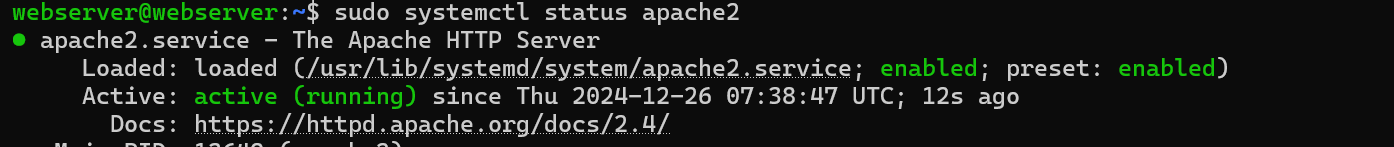
Description automatically generated

Now we can install the apache2 package

A black screen with white text

Description automatically generated

Check the status make sure it is running and enabled



Next, configure your firewall to allow apache and ssh. First as its disabled, we allow ssh first. Then apache before enabling ufw so that you wont get disconnected from ssh.

A screen shot of a computer

Description automatically generated

Now try running sudo ufw status again (if you want to further enforce security can change to **sudo ufw allow from your-ip-range to any port 22**

A screenshot of a computer

Description automatically generated

Now try typing your server ip into a browser and you should see this

A screenshot of a computer

Description automatically generated

Basic directory structure:

* Web files go in: /var/www/html/
* Configuration files in: /etc/apache2/
* Main config file: /etc/apache2/apache2.conf