

Installation guide of a Debian 11 server equipped with Apache, PostgreSQL, and PHP

I. Debian installation

1. First, launch your virtual machine Qemu/KVM by typing this command in the terminal:

S2.03-lance-installation

This command contains a command to launch Qemu:

```
lance_qemu="qemu-system-x86_64 -M q35 -cpu host -m 4G -enable-kvm -device  
VGA,xres=1024,yres=768 -display gtk,zoom-to-fit=off -drive $drive -device  
e1000,netdev=net0 -netdev user,id=net0,hostfwd=tcp::2222-  
:22,hostfwd=tcp::4443-:443,hostfwd=tcp::8080-:80,hostfwd=tcp::5432-:5432"
```

Explanation of this command:

- M q35: Specifies the virtual machine model (q35 here).
- cpu host: Using the same processor type as the host for the virtual machine.
- m 4G: Allocates 4 Go of RAM to the virtual machine
- enable-kvm: Activates the hardware virtualization support (KVM).
- device VGA,xres=1024,yres=768: Allocates a graphics device VGA with a 1024x768 resolution to the virtual machine.
- display gtk,zoom-to-fit=off: Displays the graphical output of the virtual machine in a GTK window and deactivates the automatic zoom. [OBJ]

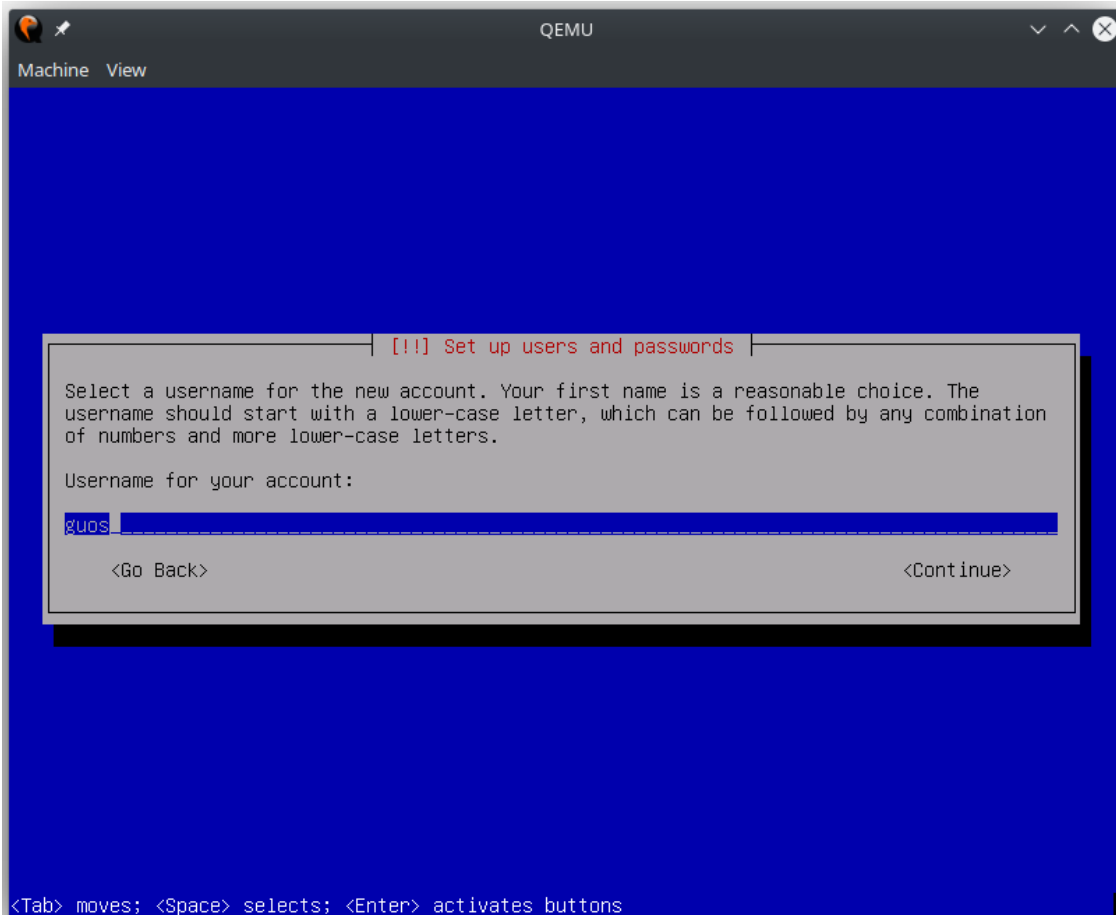
-drive \$drive: Specifies the disk image's path used by the virtual machine.

-device e1000,netdev=net0: Adds an e1000 type network card to the virtual machine and it connects it to a virtual network.

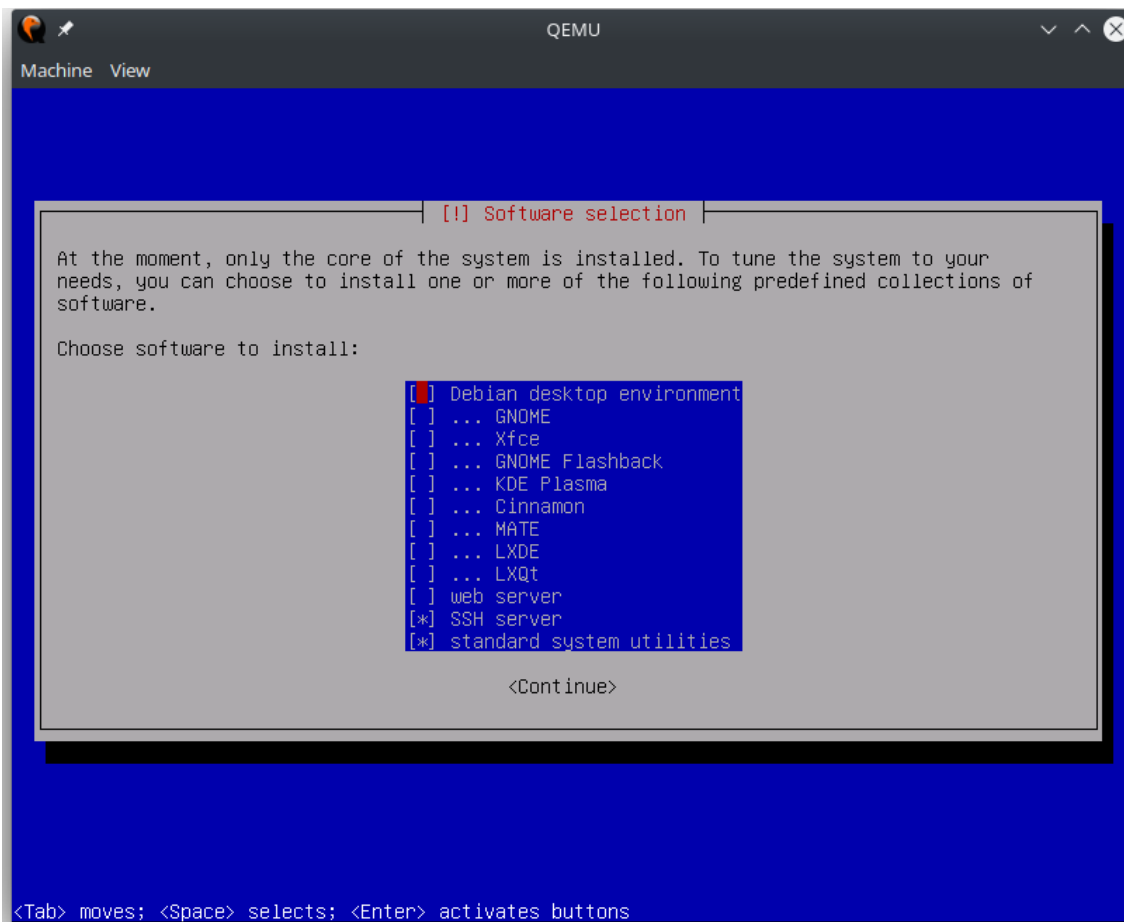
-netdev user, id=net0, hostfwd=tcp::2222-:22, hostfwd=tcp::4443-:443, hostfwd=tcp::8080-:80, hostfwd=tcp::5432-:5432: Configures the virtual network of the virtual machine. The TCP ports of the host machine (2222, 4443, 8080, 5432) are redirected to the corresponding ports on the virtual machine (22, 443, 80, 5432).

2. Now, you will arrive at the installation steps. Choose to install Debian **without graphical interface**. Then, **except for a few steps** which will be detailed below, always make the default choice:

- Hostname: Use server-"your UGA login"
- User name: Your UGA login name (like the screenshot below)



- Partition disks: Guided - use entire disk
- Partition disks: All files in one partition
- Partition disks: Yes
- Software Selection: **"Debian desktop" must not be checked, but "ssh server" must** (like the screenshot below)



- Install Grub: Yes
- Device for boot loader: /dev/sda

Then, log into the **root** count with the command **su -**. Turn off your virtual machine properly by taping **poweroff**. You can restart your virtual machine by taping **S2.03-lance-machine-virtuelle** in the terminal of the host machine.

3. You can type **cat /etc/fstab** in your VM (virtual machine) to list all disks and partitions.

```
Machine View
guos@server-guos:~$ cat /etc/fstab
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# systemd generates mount units based on this file, see systemd.mount(5).
# Please run 'systemctl daemon-reload' after making changes here.
#
# <file system> <mount point> <type> <options> <dump> <pass>
# / was on /dev/sda1 during installation
UUID=76d1c358-2f8d-4dfd-aaef-ded7a3303151 / ext4 errors=remount-ro 0 1
# swap was on /dev/sda5 during installation
UUID=dd3935db-1e02-4fef-9553-68093a219dff none swap sw 0 0
/dev/sr0 /media/cdrom0 udf,iso9660 user,noauto 0 0
guos@server-guos:~$
```

II. Apache, PostgreSQL, and PHP installation

When there is “#” in front of a command, it means it is on root. Otherwise, it will be written “\$”. Those 2 symbols must **not** be typed. If you want to logout (from root to non-root or from postgresSQL to root from example), type **exit**.

1. Apache

Connect in **root** (su -) and write **apt install apache2**. Type **yes** or **y** according to what the terminal request. Apache needs to be started. Type **# service apache2 start**. And this is installed.

```
Machine View
root@server-guos:~# service apache2 start
root@server-guos:~# _
```

Verify Apache is started by writing **systemctl status apache2** in root.

```
Machine View
root@server-guos:~# systemctl status apache2
• apache2.service - The Apache HTTP Server
  Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
  Active: active (running) since Wed 2023-05-10 13:53:25 CEST; 21min ago
    Docs: https://httpd.apache.org/docs/2.4/
  Main PID: 441 (apache2)
    Tasks: 55 (limit: 4661)
  Memory: 12.1M
    CPU: 124ms
  CGroup: /system.slice/apache2.service
          └─441 /usr/sbin/apache2 -k start
            └─444 /usr/sbin/apache2 -k start
              └─445 /usr/sbin/apache2 -k start

May 10 13:53:25 server-guos systemd[1]: Starting The Apache HTTP Server...
May 10 13:53:25 server-guos apache2[423]: AH00557: apache2: apr_socketaddr_info_get() failed for server-guos
May 10 13:53:25 server-guos apache2[423]: AH00558: apache2: Could not reliably determine the server's fully qualified domain
May 10 13:53:25 server-guos systemd[1]: Started The Apache HTTP Server.
lines 1-17/17 (END)
```

If Apache is not started, restart it: **# *systemctl restart apache2***

That's not possible to display html pages because there is no graphical interface. However, we can display one on the host machine.

Connect to Apache: **# *telnet localhost 80***. Then, type **"HEAD / HTTP/1.0"**.

Warning, the quotes are important!

```
Machine View
root@server-guos:~# telnet "HEAD / HTTP/1.0"
telnet: could not resolve HEAD / HTTP/1.0/telnet: Name or service not known
root@server-guos:~# telnet localhost 80
Trying ::1...
Connected to localhost.
Escape character is '^]'.
HEAD / HTTP/1.0

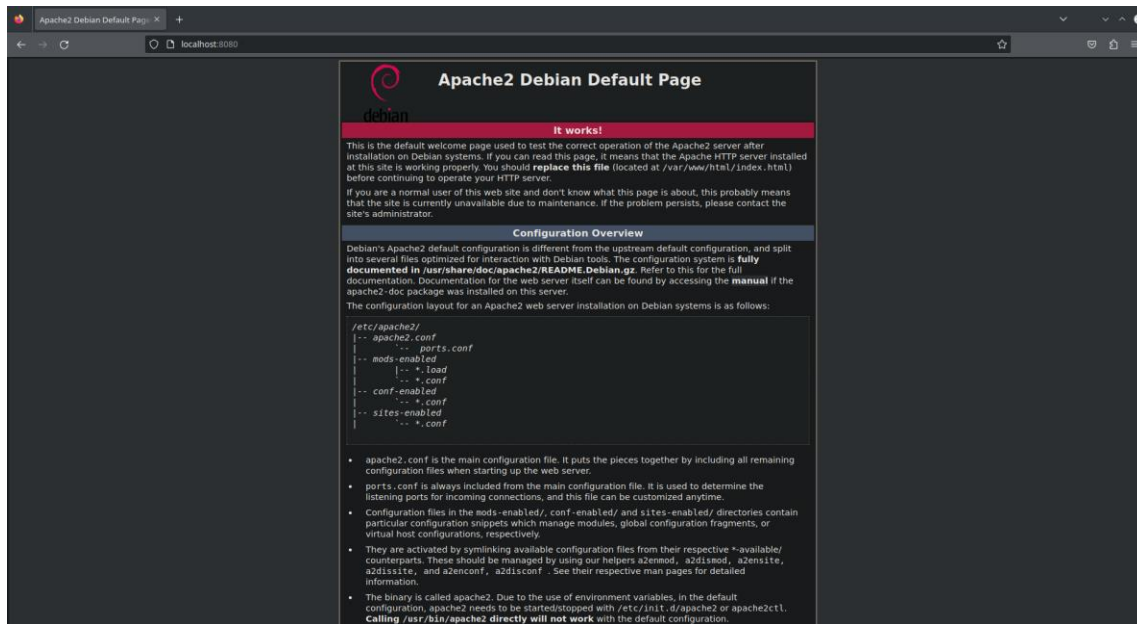
HTTP/1.1 200 OK
Date: Wed, 10 May 2023 12:00:11 GMT
Server: Apache/2.4.56 (Debian)
Last-Modified: Wed, 10 May 2023 11:32:57 GMT
ETag: "29cd-5fb553bce1bfd"
Accept-Ranges: bytes
Content-Length: 10701
Vary: Accept-Encoding
Connection: close
Content-Type: text/html

Connection closed by foreign host.
root@server-guos:~#
```

On the host machine now, open a web navigator and go to this URL:

<http://localhost:8080/>

You should arrive on the Apache2 Debian default page:



2. PostgreSQL

Install it: **# apt install postgresql** and verify it's started (**# systemctl status postgresql**):

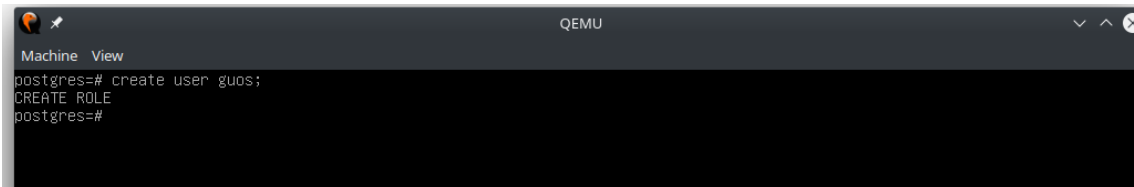
```
Machine View
root@server-guos:~# systemctl status postgresql
● postgresql.service - PostgreSQL RDBMS
   Loaded: loaded (/lib/systemd/system/postgresql.service; enabled; vendor preset: enabled)
   Active: active (exited) since Wed 2023-05-10 14:08:23 CEST; 1min 55s ago
   Main PID: 1920 (code=exited, status=0/SUCCESS)
     Tasks: 0 (limit: 4661)
    Memory: 0B
       CPU: 0
   CGroup: /system.slice/postgresql.service

May 10 14:08:23 server-guos systemd[1]: Starting PostgreSQL RDBMS...
May 10 14:08:23 server-guos systemd[1]: Finished PostgreSQL RDBMS.
root@server-guos:~#
```

Verify it's installed: **# su - postgres**. It will connect you with the login *postgres*.

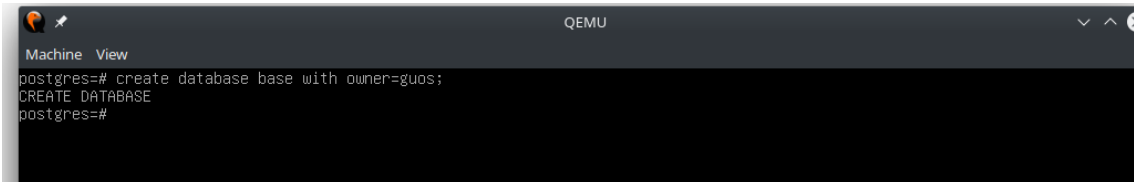
\$ psql -l to shows the default database list.

Create a user as your login name:



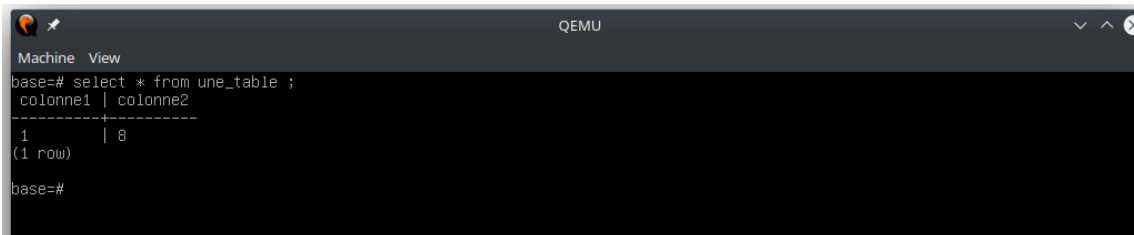
```
Machine View
postgres=# create user guos;
CREATE ROLE
postgres=#
```

And create a database where you are the owner:



```
Machine View
postgres=# create database base with owner=guos;
CREATE DATABASE
postgres=#
```

Create a table and insert or not some data:



```
Machine View
base=# select * from une_table ;
colonne1 | colonne2
-----+-----
1         | 8
(1 row)

base=#
```

We are going to configure PostgreSQL to be accessible by the host machine.