DEBIAN SERVER INSTALLATION HANDBOOK

# Project team

* BARNOUIN Clément

# Contents

[**Équipe projet 0**](#_8ee9578zwzr)

[**Contents 0**](#_pyzs242d3iy)

[**Introduction 1**](#_brnkrfmz7kzt)

[**Prerequisites 1**](#_tptrxyl5stxc)

[**Les commandes importantes 2**](#_sekwyj25f06y)

[Commandes de base 2](#_cgecsolygk75)

[Commandes de gestion du système 2](#_t60c0qkjw6p2)

[Commandes de gestion des fichiers 2](#_l0qypx2x4ztc)

[Commandes de gestion des paquets 3](#_vty4b6di0uxw)

[Commandes de gestion des services 3](#_o4a3arbp639c)

[Commandes réseau 3](#_80ug4s2vb1wa)

[Commandes PostgreSQL 4](#_s3snzscq0zxi)

[Commandes Apache 4](#_4e90wdgj7xtk)

[Commandes PHP 4](#_tln9yhal29l5)

[Commandes de gestion de clés SSH 4](#_yb68m6kflncg)

[Raccourcis 4](#_7e5izu1v9obz)

[Notations 4](#_2a9j805qv3lx)

[**Installation 5**](#_mn5ahvnj9zdi)

[Step 1: Setting up the Installation 5](#_gq1jl1ctrs2d)

[1. Download the Debian 12 ISO image 5](#_ut02m3v9pdx7)

[2. Vérifier l’intégrité de l’image 5](#_rjaz81h65mpb)

[3.1. [Condition Universitaire] Create a new virtual machine using Qemu/KVM 5](#_sbji0ll62cjl)

[3.2. [Condition Réelle] Create a new virtual machine using Qemu/KVM 5](#_ev373mqwtwop)

[Step 2: Install Debian 12 5](#_pyca8hyrlnzk)

[1. Boot from the ISO and start the installation 5](#_dralk0eolz9m)

[2. Post-installation commands 6](#_ssubv3c6n98j)

[Step 3: Move Disk Image 6](#_el0d2to42bpn)

[1.1. [Condition Universitaire] Move the disk image to the server: 6](#_4ko4vjd0oq2n)

[1.2. [Condition Réelle] Move the disk image to the server: 6](#_d7lrjvwm8hny)

[Step 4: Verify the Server Installation 7](#_p5qw2dr2az85)

[1.1. [Condition Universitaire] Launch the virtual machine 7](#_j7vjc4zd6orl)

[1.2. [Condition Réelle] Launch the virtual machine 7](#_5zo01yi9a2az)

[2. Vérifier le montage des partitions 7](#_r7ayhs67j616)

[3. Check network configuration and ensure connectivity 7](#_kzb0dfxkxfkg)

[Step 5: Install Apache 8](#_vlnq2wqq3rzw)

[1. Install Apache 8](#_azxc9lc7azs3)

[2. Verify Apache installation 8](#_idr6u4fgl8wk)

[3. Access Apache from the host 8](#_fzkbl4riqt3w)

[Step 6: Install PostgreSQL 9](#_pd2x7699iqqg)

[1. Install PostgreSQL 9](#_12zgroig3mt3)

[2. Verify PostgreSQL installation 9](#_4voq79kp3j8o)

[3. Test PostgreSQL 9](#_lxuxqa4uc5rx)

[4. Configure PostgreSQL for accessibility from the host station 9](#_ey5uhmld0pf8)

[5. Interroger la base de données 10](#_uxqxb6fr7n1n)

[6. Vérification du hachage des mots de passe 10](#_331fl0z83tjc)

[Step 7: Install PHP 11](#_j9argzyunuok)

[1. Install PHP 11](#_33g1o395fc1)

[2. Test PHP installation 11](#_u721phfx5ps4)

[3. Access the PHP info page from the host 11](#_kg2730ovfp39)

[4. Access to another page 11](#_jnuhktd51suf)

[Step 8: Install PhpPgAdmin 12](#_w44e70xxv9jc)

[1. Install PhpPgAdmin 12](#_yu46a4c4l1fb)

[2. Configure PhpPgAdmin 12](#_glkvrmw7fnky)

[3. Access PhpPgAdmin from the host 12](#_mpchmd2pom8i)

[Step 9: Security Analysis 13](#_drbhee13eph1)

[1. Espace de stockage final 13](#_9hzdfou1x7c4)

[2. Identify and fix potential security vulnerabilities 13](#_r50wxm3dem8j)

[**Optional Enhancements 13**](#_btdfi6wrbcbs)

[1. SSH Access with Key Authentication 13](#_pj1mr5juydf0)

[Méthode 1 - Accès au compte root par SSH 13](#_b7vjt7nrxzwl)

[Méthode 2 - Accès au serveur SSH 14](#_5nkiveeidf8t)

[2. Enable HTTPS on Apache: 14](#_27iy3rnvxqlf)

[**Conclusion 14**](#_53kele57wgo4)

# Introduction

This guide will provide step-by-step instructions for installing a Debian 12 server with Apache, PostgreSQL, and PHP. This installation will be performed on a Qemu/KVM virtual machine and should be accessible and manageable from the host machine.

# Prerequisites

1. Qemu/KVM installed on the host machine
2. Debian 12 ISO image (netinst)
3. Basic knowledge of Linux commands and system administration

# Commands

## Basic commands

cd [directory] : Changes the current directory.

**Example :** $ cd /var/www/html

ls [options] [directory] : Lists the contents of a directory.

**Example :** $ ls -l

cp [source] [destination] : Copying files or directories.

**Example :** $ cp file.txt /home/user/

mv [source] [destination] : Moves or renames files or directories.

**Example :** $ mv fichier.txt /home/user/new\_name.txt

rm [options] [file/directory] : Delete files or directories.

**Example :** $ rm file.txt

## System management commands

su - root: Login as root from user.

logout: Disconnect.

shutdown -h now: Shuts down the system immediately.

reboot: Restarts the system.

poweroff: Turns off the system.

## File management commands

nano [file]: Command line text editor.

**Example :** $ nano /etc/apache2/apache2.conf

cat [file]: Displays the contents of a file.

**Example :** $ cat /etc/passwd

find [directory] -name [filename]: Finds files in a directory.

**Example :** $ find / -name "file.txt"

grep [pattern] [file]: Searches for a pattern in a file.

**Example :** $ grep "root" /etc/passwd

## Package management commands

apt-get update: Updates the list of available packages.

**Example :** # apt-get update apache2

apt-get upgrade: Updates installed packages.

**Example :** # apt-get upgrade apache2

apt-get install [package]: Installs a package.

**Example :** # apt-get install apache2

apt-get remove [package]: Removes a package.

**Example :** # apt-get remove apache2

apt-get clean: Cleans installation packages.

## Service management commands

systemctl status [service]: Displays the status of a service.

**Example :** # systemctl status apache2

systemctl start [service]: Starts a service.

**Example :** # systemctl start apache2

systemctl stop [service]: Stops a service.

**Example :** # systemctl stop apache2

systemctl restart [service]: Restarts a service.

**Example :** # systemctl restart apache2

systemctl enable [service]: Enables a service to start at system startup.

**Example :** # systemctl enable apache2

systemctl disable [service]: Disables a service so that it does not start at system startup.

**Example :** # systemctl disable apache2

## Network commands

ip addr: Displays IP address and network configuration information.

**Example :** $ ip addr

ssh [user]@[ip\_address] -p [port]: Connects to a remote machine via SSH.

**Example :** $ ssh user@192.168.1.10 -p 22

scp [source] [user]@[ip\_address]:[destination]: Copies files to a remote machine.

**Example :** $ scp fichier.txt user@192.168.1.10:/home/user/

## PostgreSQL commands

su - postgres: Changes the current user to PostgreSQL user.

psql -l: List PostgreSQL databases.

psql -d [database\_name]: Connects to a PostgreSQL database.

psql -h: Connects to a PostgreSQL database on a remote host by specifying the user.

**Example :** $ psql -h localhost -U postgres mydatabase

## Apache Commands

apachectl configtest: Tests Apache configuration for errors.

systemctl reload apache2: Reloads the Apache configuration without restarting the service.

**Example :** # systemctl reload apache2

## PHP Commands

php -v: Shows the installed PHP version.

**Example :** $ php -v

## SSH key management commands

ssh-keygen: Generates a new SSH key pair.

**Example :** $ ssh-keygen

ssh-add [private\_key\_path]: Adds a private key to the SSH agent.

**Example :** $ ssh-add ~/.ssh/id\_rsa

ssh user@ip\_address: SSH connection.

## Shortcuts

CTRL + C: Stop a process in a terminal.

CTRL + L: Clean terminal.

CTRL + W: Search in nano software.

CTRL + O: Save in nano software.

CTRL + X: Exit nano software.

## Notations

Commands marked with a “$” at the beginning can be executed in a user's command terminal. Commands with “#” are only executed in a root terminal. The commands highlighted in blue are to be performed on the host machine and in gray, on the virtual machine.

# Installation

## Step 1: Setting up the Installation

### 1. Download the Debian 12 ISO image

The ISO image can be downloaded from Debian's official site : <https://cdimage.debian.org/cdimage/release/current/amd64/iso-cd/>.

Install Debian version 12.x "bookworm” pour processeurs x86 64 bits avec l'image ISO de type "netinst".

### 2. Check image integrity

Visually compare the two prints by performing the command:

$ sha512 FILE\_NAME

### 3.1. [University Condition] Create a new virtual machine using Qemu/KVM

$ S2.03-lance-installation

### 3.2. [Real Condition] Create a new virtual machine using Qemu/KVM

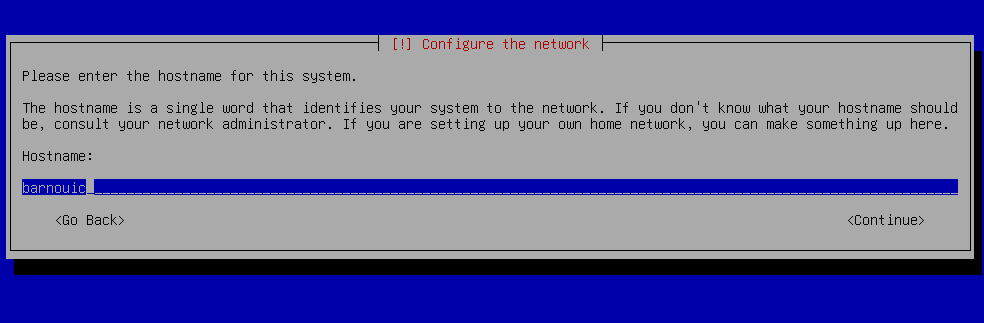
Only if a Qemu/KVM machine is not yet initialized.

# qemu-system-x86\_64 -hda debian12.img -cdrom debian-12-netinst.iso -boot d -m 1024

## Step 2: Install Debian 12

### 1. Boot from the ISO and start the installation

* Language : English
* Location : Other/Europe/France
* Locales : United States, en\_US.UTF-8
* Keyboard : French
* Hostname : `server-YOUR\_UGA\_LOGIN`
* Root Password : `root`
* User Account : Full Name: `Your Name`, Username: `YOUR\_UGA\_LOGIN`, Password: `etu`
* Partition Disks : Guided - use entire disk, All files in one partition
* Software Selection : Uncheck 'Debian desktop', check 'ssh server'
* Install GRUB : Yes
* Device for boot loader : /dev/sda

*Figure 1. Entering the user login name.*

#### 

#### Figure 2. Software selection during installation.

### 2. Post-installation commands

After the first reboot, log in as root and run :

# poweroff

## Step 3: Move Disk Image

### 1.1. [University Condition] Move the disk image to the server

$ S2.03-déplace-image-disque-sur-erebus4

### 1.2. [Real Condition] Move the disk image to the server

# scp debian12.img user@server:/path/to/destination

## 

## Step 4: Verify the Server Installation

### 1.1. [University Condition] Launch the virtual machine

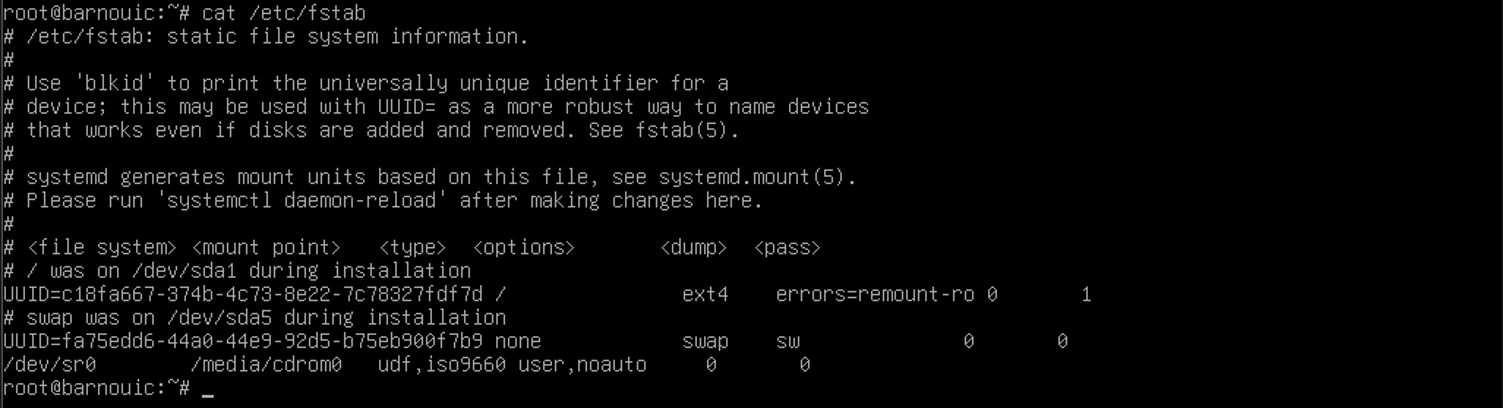
$ S2.03-lance-machine-virtuelle

### 1.2. [Real Condition] Launch the virtual machine

# qemu-system-x86\_64 -hda debian12.img -m 1024

### 2. Check partition mounting

# cat /etc/fstab

*Figure 3. Result of the cat /etc/fstab/ command to verify the installation of the installation.*

### 3. Check network configuration and ensure connectivity

Take note of the IPv4, IPv6 and MAC addresses.

$ ip addr

Check the absence of the Xorg server.

$ dpkg -l | grep xorg

**Port forwarding table:**

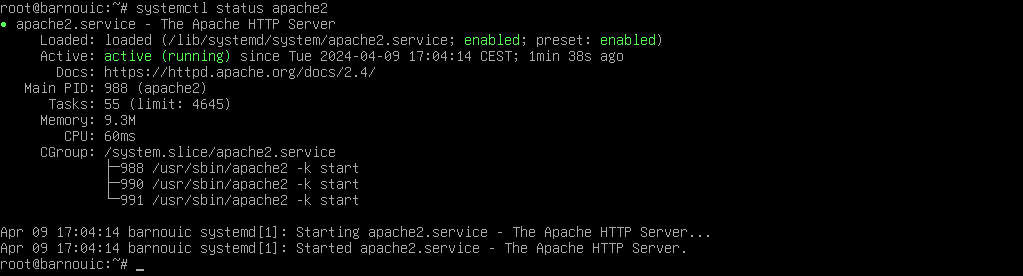
| **Network service** | **VM port** | **Port on Linux station** | **Example of use from the Linux station** |
| --- | --- | --- | --- |
| SSH | 22 | 2222 | $ ssh toto@localhost -p 2222 |
| HTTP | 80 | 8080 | URL: http://localhost:8080/ |
| HTTPS | 443 | 4443 | URL: https://localhost:4443/ |
| PostgreSQL | 5432 | 5432 | $ psql -h localhost -U postgres postgres |

## Step 5: Install Apache

### 1. Install Apache

# apt-get install apache2

### 2. Verify Apache installation

# systemctl status apache2*Figure 4. Verifying that Apache2 is working.*

If Apache is not started :

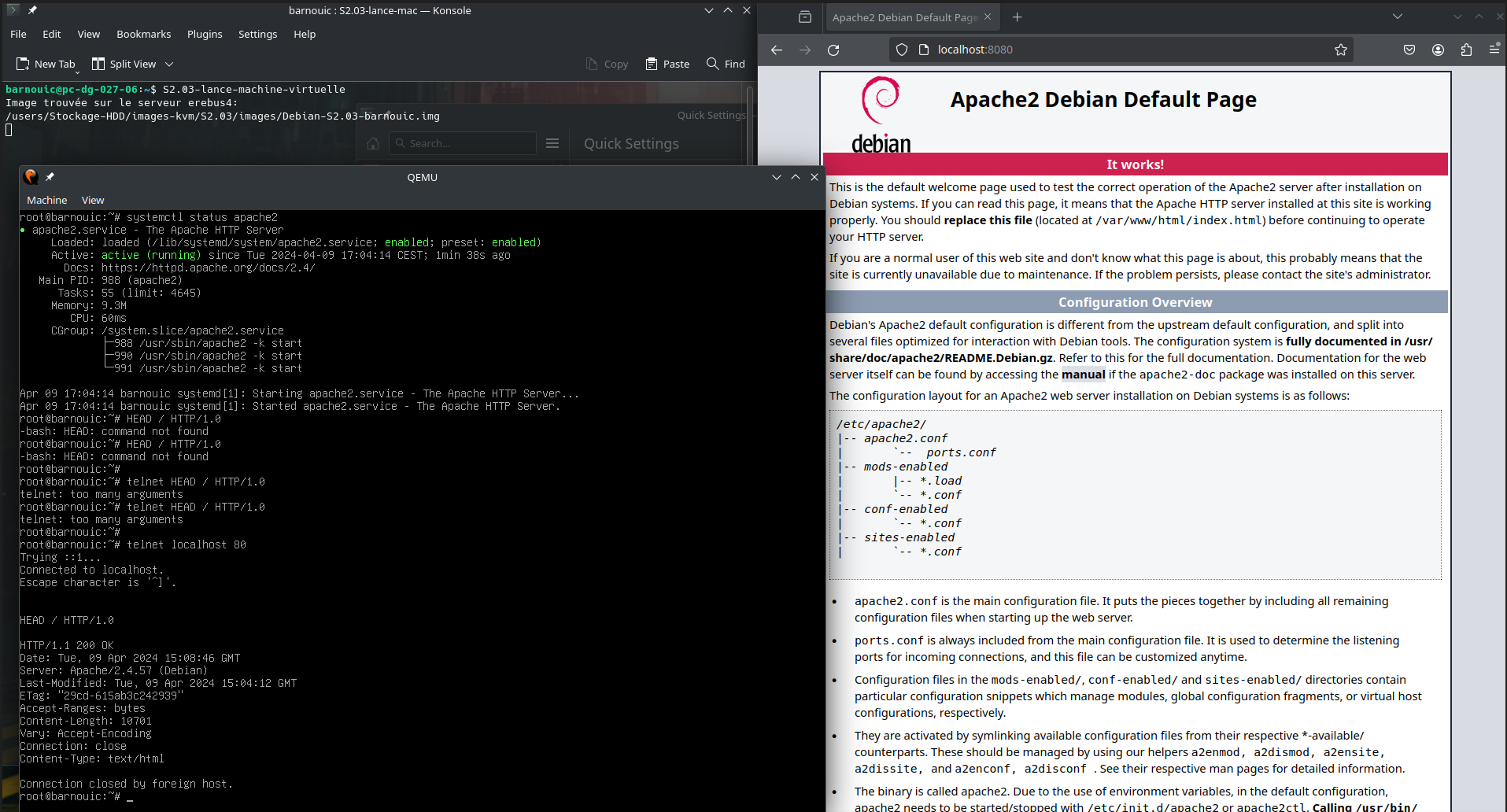
# systemctl start apache2

### 3. Access Apache from the host

$ telnet localhost 80

Write “HEAD / HTTP/1.0 then two newlines. The server should respond “HTTP/1.1 200 OK”

Open a web browser and navigate to `<http://localhost:8080>` sur la machine hôte.

*Figure 5. Web page on the host machine, hosted on the virtual machine using Apache2.*

## Step 6: Install PostgreSQL

### 1. Install PostgreSQL

# apt-get install postgresql

### 2. Verify PostgreSQL installation

# systemctl status postgresql



*Figure 6. Verifying the operation of PostgreSQL.*

### 3. Test PostgreSQL

Connect to PostgreSQL with login postgres and list current databases :

# su - postgres

$ psql -l

Create a new user with your UGA login as the name, then create a new database whose owner is your user with a new table in the database containing several lines.

$ createuser barnouic

CREATE USER barnouic WITH CREATEROLE PASSWORD 'etu';

ALTER USER barnouic WITH CREATEROLE PASSWORD 'etu';

DROP USER barnouic;

GRANT INSERT, UPDATE, SELECT ON TABLE identifiant\_etu TO barnouic;

$ createdb universite -O barnouic

CREATE TABLE identifiant\_etu (id SERIAL PRIMARY KEY, nom VARCHAR(50), age INT);

INSERT INTO identifiant\_etu (nom, age) VALUES ('Clement', 19), ('Loucas', 18);

### 4. Configure PostgreSQL for accessibility from the host station

- Edit `/etc/postgresql/15/main/postgresql.conf` :

# nano /etc/postgresql/15/main/postgresql.conf

**Edit to :** listen\_addresses = '\*'

- Edit `/etc/postgresql/15/main/pg\_hba.conf` :

# nano /etc/postgresql/15/main/pg\_hba.conf

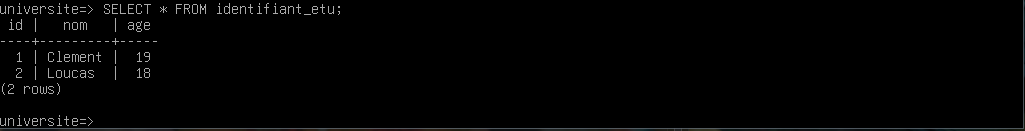
**Edit to :** host all all 0.0.0.0/0 scram-sha-256

# sytemctl restart postgresql

### 5. Query the database

$ psql -d universite

SELECT \* FROM identifiant\_etu;

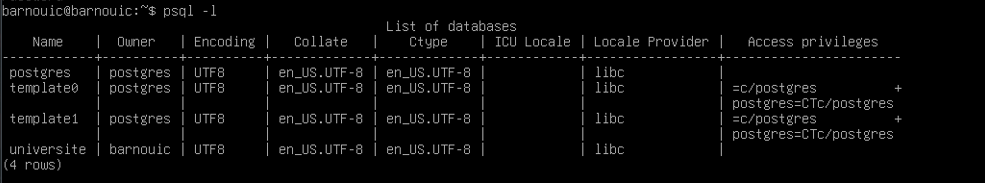
*Figure 7. Querying the university database from the virtual machine.*

$ psql -h posgres-info universite -U barnouic

SELECT \* FROM identifiant\_etu;



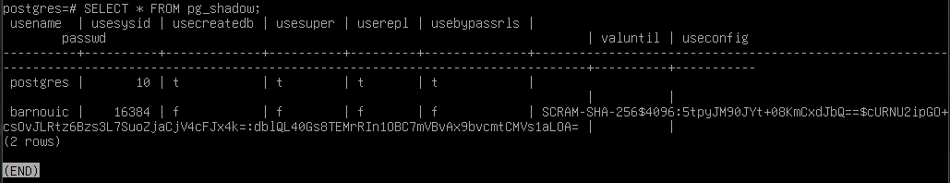
*Figure 8. Querying the university database from the host machine.*

$ psql -l

*Figure 9. List of PostgreSQL database owners from the virtual machine.*

### 6. Checking password hashes

# su - postgres

SELECT \* FROM pg\_shadow;

*Figure 10. Querying the pg\_shadow table from the virtual machine.*

## Step 7: Install PHP

### 1. Install PHP

# apt-get install php

### 2. Test PHP installation

- Create a `info.php` file in `/var/www/html/` :

# nano /var/www/html/info.php

**Write down :**

<?php

phpinfo();

phpinfo(INFO\_MODULES);

?>

### 3. Access the PHP info page from the host

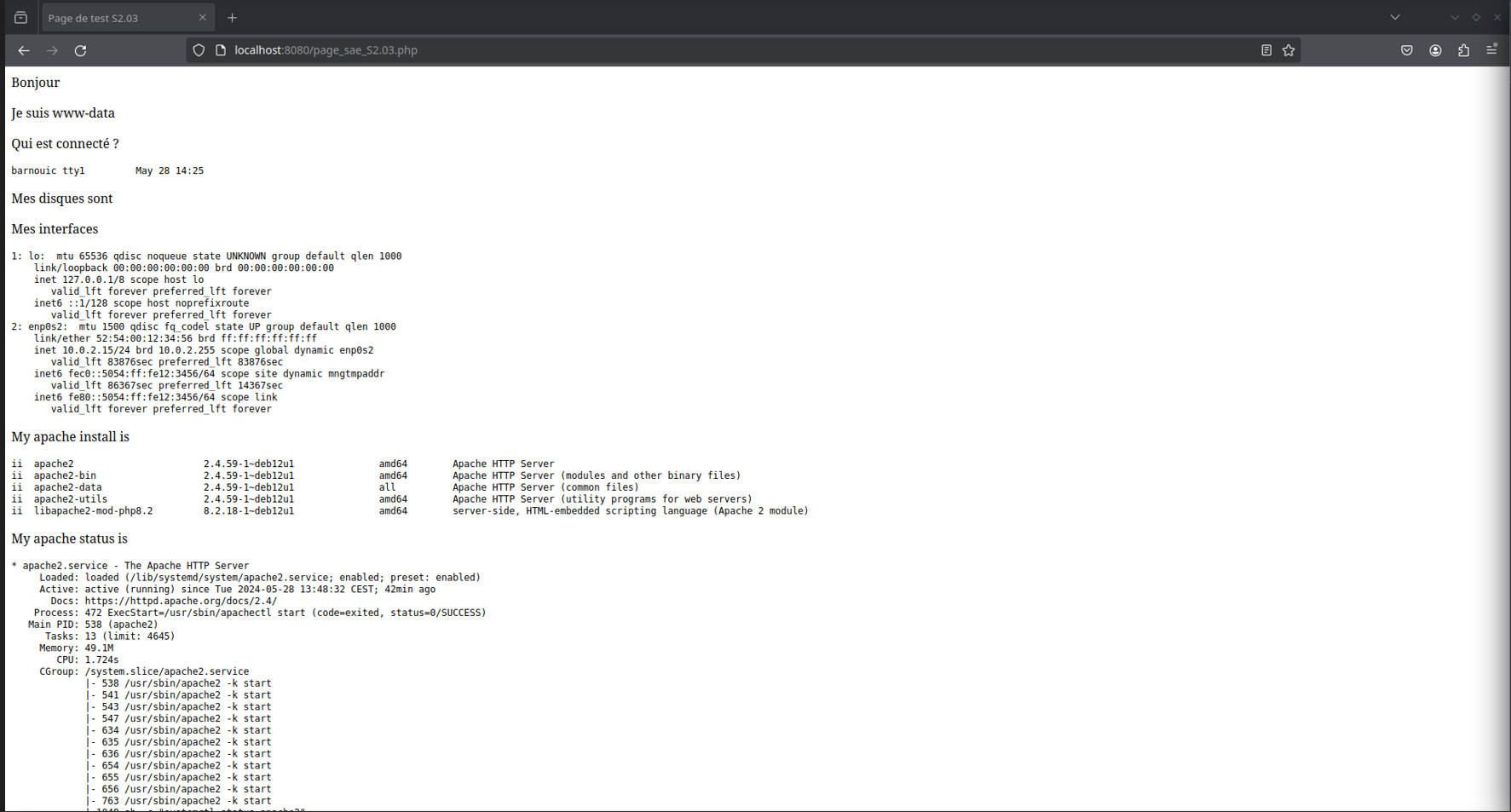
Open a web browser and navigate to `<http://localhost:8080/info.php>`.

### 4. Access to another page

# /sbin/blkid

# scp barnouic@pc-dg-027-10:/users/info/www/intranet/enseignements/S2.03/page\_sae\_S2.03.php /var/www/html

Open a web browser and navigate to `<http://localhost:8080/page_sae_S2.03.php>`.



*Figure 10. Navigation on the page\_sae\_S2.03.php from the host machine using apache2.*

## Step 8: Install PhpPgAdmin

### 1. Install PhpPgAdmin

# apt-get install phppgadmin

### 2. Configure PhpPgAdmin

- Edit `Connection.php` to ensure compatibility :

# find / -name "Connection.php"

#nano /usr/shar/phppgadmin/classes/database/Connection.php

**Edit :**

case '14': return 'Postgres';break;

**To :**

case '15': return 'Postgres';break;

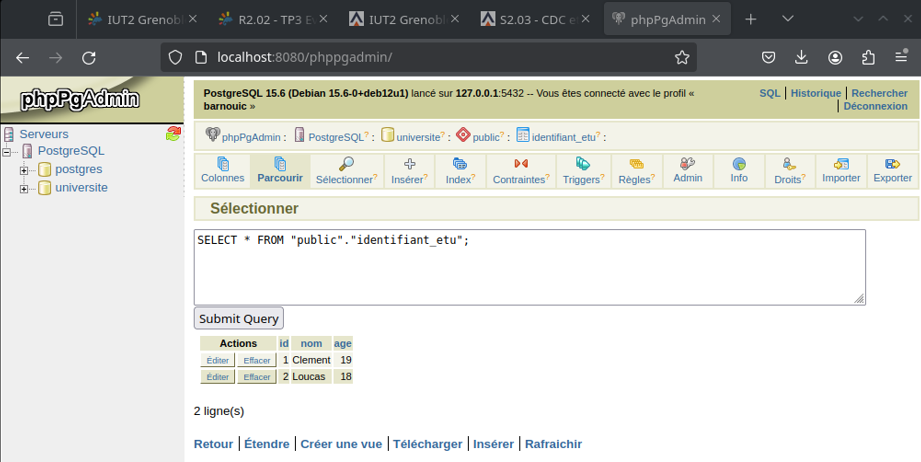
- Edit `phppgadmin.conf` to allow connection :

# nano /etc/apache2/conf-available/phppgadmin.conf

**Write down :**

Require all granted

### 3. Access PhpPgAdmin from the host

Open a web browser and navigate to `[http://localhost:8080/phppgadmin`.](http://localhost:8080/phppgadmin%60.*)

*Figure 11. Querying the university database from PhpPgAdmin on the host machine.*

## Step 9: Security Analysis

### 1. Final storage space

*Figure 12. Storage space after installations.*

### 2. Identify and fix potential security vulnerabilities

- Regularly update system packages :

# apt-get update && apt-get upgrade

- Install Logchecker and apply specific rules if you want or leave as default

# apt-get install logcheck

# nano /etc/logcheck/logcheck.conf

- Configure firewalls and access controls appropriately.

# Optional Enhancements

## 1. SSH Access with Key Authentication

# systemctl status ssh

## 

*Figure 13. Verifying SSH operation.*

### Method 1 – Accessing the root account via SSH

$ ssh-keygen

$ ssh-copy-id root@10.0.2.15

# nano /etc/ssh/sshd\_config

**Decomment / Edit to :**

PermitRootLogin prohibit-password

PubkeyAuthentication yes

# systemctl restart sshd

### Method 2 – Accessing the SSH Server

- Generate SSH keys and configure the server to accept key-based authentication.

$ ssh-keygen

ssh-copy-id barnouic@10.0.2.15

# nano /etc/ssh/sshd\_config

**Decomment :**

PubkeyAuthentication yes

AuthorizedKeysFile .ssh/authorized\_keys

# systemctl restart sshd

## 2. Enable HTTPS on Apache:

- Generate a self-signed certificate using OpenSSL and configure Apache to use it.

# openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/ssl/private/apache-selfsigned.key -out /etc/ssl/certs/apache-selfsigned.crt

- Répondez aux questions interactives pour créer votre certificat.

# nano /etc/apache2/sites-available/default-ssl.conf

Make sure the `SSLCertificateFile` and `SSLCertificateKeyFile` directives point to the correct paths of the certificate and private key files.

# a2enmod ssl

# systemctl restart apache2

- Make sure port 443 (HTTPS) is open in your firewall.

# Conclusion

This guide provides the essential steps to install and configure a Debian 12 server with Apache, PostgreSQL, and PHP. By following these steps, you should have a functional server environment accessible from your host machine.