

Manuel d'installation DEBIAN

I. Installation DEBIAN 12

II. Installation d'un serveur postgres avec PhpPgAdmin

Légende pour les commandes :

Préfixe :

% est une commande avec l'utilisateur simple
est une commande avec le super utilisateur
£ est une commande avec l'utilisateur postgres

Couleur :

commande sur la machine principal
commande sur la machine virtuelle

I. Installation DEBIAN 12

1.1 : télécharger le fichier ISO de debian et les fichiers permettant de vérifier son intégrité avec ce lien :

<https://cdimage.debian.org/cdimage/release/current/amd64/iso-cd/>

vérifier l'intégrité du fichier avec cette commande :

% sha512sum NOM_FICHER

Comparer visuellement la chaine obtenue avec celle qui se trouve sur le site de debian

1.2 : installation de la machine :

exécuter cette commande pour installer la machine :

% S2.03-lance-installation

qemu-system-x86_64 [paramètres...]

La liste de paramètre utilisé pour lancer la machine virtuelle :

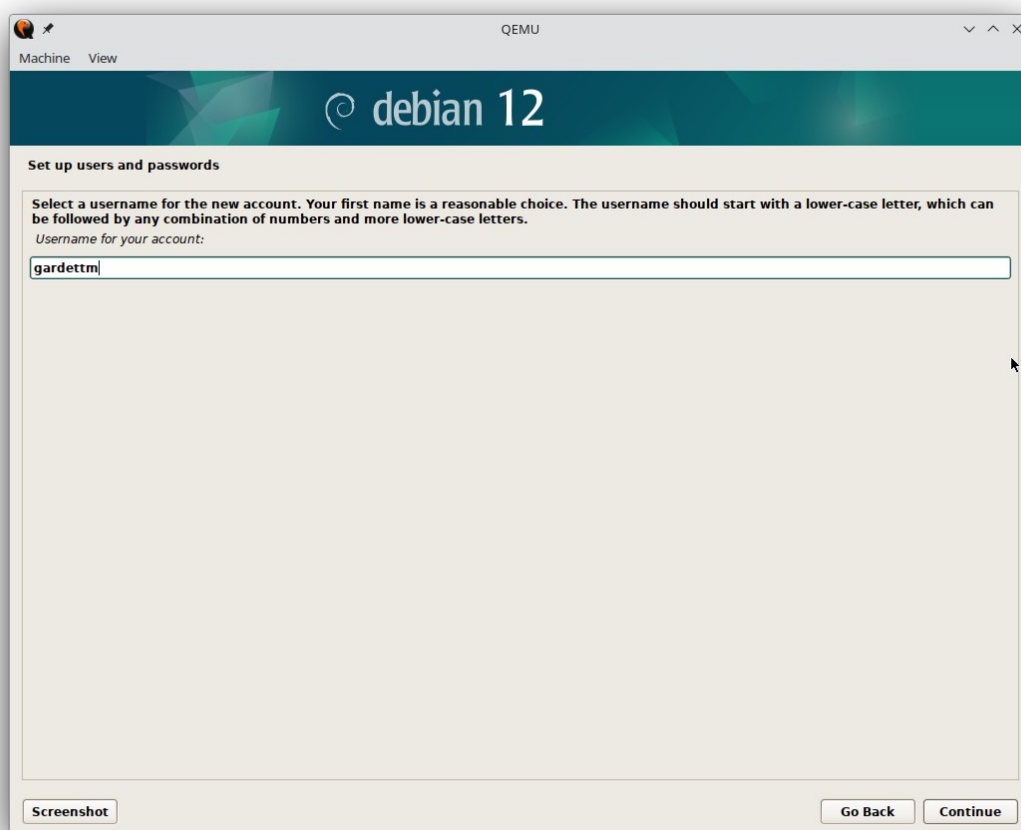
- machine q35 : utilise le chipset Q35, compatible PCIe (plus moderne que pc).
 - cpu host : la VM utilise les mêmes instructions CPU que l'hôte (meilleures performances).
 - m 4G : alloue 4 Go de RAM à la machine virtuelle.
 - enable-kvm : active l'accélération matérielle via KVM (Kernel-based Virtual Machine).
 - device VGA,xres=1024,yres=768 : définit une carte graphique VGA avec une résolution de 1024x768.
 - display gtk,zoom-to-fit=off : utilise une interface graphique GTK, sans zoom automatique.
 - drive format=raw,file=\$image,discard=unmap : monte une image disque brute (raw) avec gestion des blocs inutilisés (discard).
 - device e1000,netdev=net0 : ajoute une carte réseau Intel e1000 connectée à net0.
 - netdev user,id=net0,... : utilise le mode réseau "utilisateur" avec redirection de ports :
 - hostfwd=tcp::2222-:22 : SSH (accès via localhost:2222)
 - hostfwd=tcp::4443-:443 : HTTPS (accès via localhost:4443)
 - hostfwd=tcp::8080-:80 : HTTP (accès via localhost:8080)
 - hostfwd=tcp::5432-:5432 : PostgreSQL (accès via localhost:5432)

On passe ensuite à la configuration du system debian :

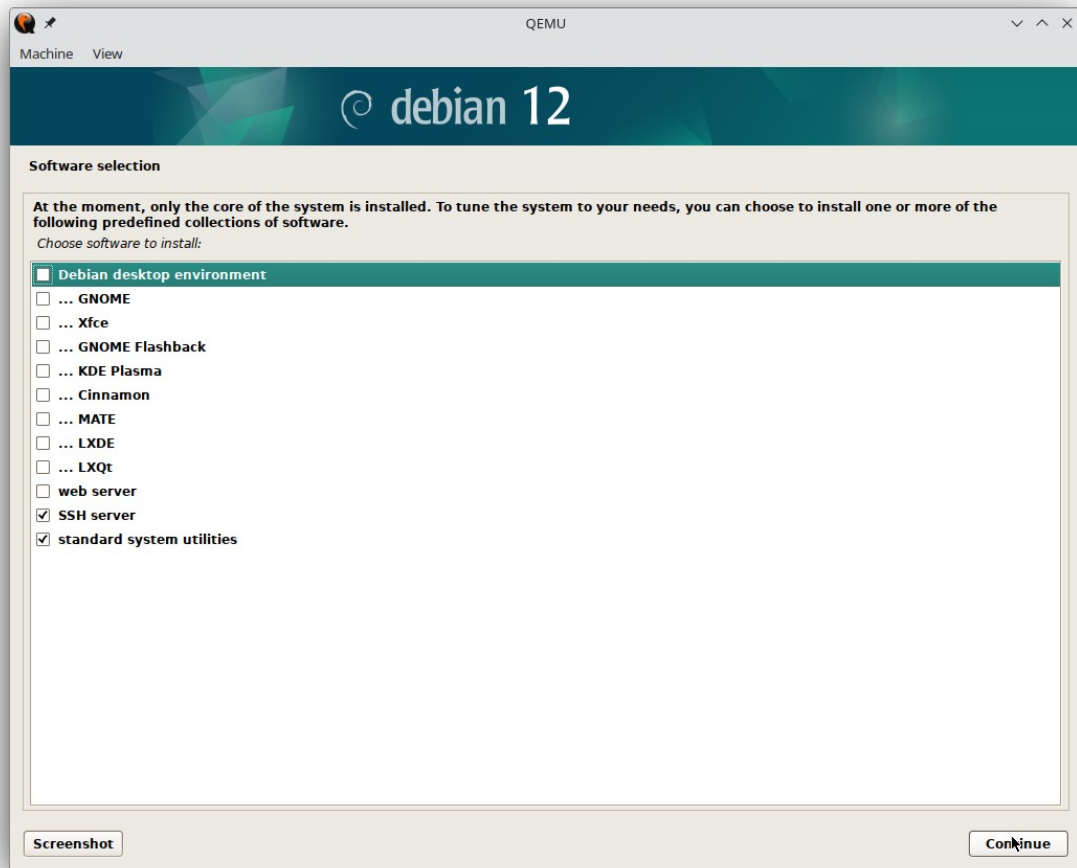
Passez une à une les étapes de l'installation. Quand rien n'est précisé, faire le choix proposé par défaut sinon les choix seront précisés ci dessous pour chaque étapes :

- Language : English
- **Location : other/Europe/France**

- Locales : United States, en_US.UTF-8
- **Keyboard : French**
- **Hostname** : utilisez server-"VOTRE_LOGIN_UGA"
- **Root Password** : un mot de passe simple est conseillé, par exemple "root". Dans ce contexte, cela ne pose pas de problème de sécurité. Cocher la case "Show Password" pour être sûr que le mot de passe saisi est bien celui que vous voulez.
- User Account - **Full Name** : votre nom complet, par exemple "Jean Toto"
- **User Name** : saisir votre nom de login UGA



- **User Password** : saisir un mot de passe simple, par exemple "etu". Cocher la case "Show Password" pour être sûr que le mot de passe saisi est bien celui que vous voulez.
- Partition disks : Guided - use entire disk
- Partition disks : All files in one partition
- **Partition disks : Yes**
- **Software Selection** : vérifier que "Debian desktop" n'est pas coché et que "ssh server" est coché



- Install GRUB : Yes
- **Device for boot loader : /dev/sda**

Lorsque l'installation est terminée et que la machine a redémarrer,
se connecter avec le compte root avec la commande :

```
% su -
```

entrer ensuite le mot de passe du compte root

puis exécuter la commande suivante pour éteindre proprement la machine :

```
# systemctl poweroff
```

Le fichier a été créé sur le disque local de la station Linux. Vous devez donc déplacer cette image sur le serveur erebus4 pour pouvoir utiliser votre machine virtuelle plus facilement par la suite ou vous pouvez aussi copier cette image sur une clé USB.

Pour déplacer l'image sur le serveur erebus4 exécuter cette commande :

```
% S2.03-déplace-image-disque-sur-erebus4
```

Vérifier que la machine est bien fonctionnelle en la relancant avec la commande :

```
% S2.03-lance-machine-virtuelle
```

Noter les adresses ethernet et ip de votre machine virtuelle avec la commande suivante :

```
% ip addr
```

essayer ensuite d'accéder a un serveur extérieur en utilisant la commande ping et un serveur choisie disponible :

```
% ping google.com
```

Vérifiez l'absence du serveur Xorg sur votre machine virtuelle. Taper la commande :

```
% dpkg -l | grep xorg
```

la machine ne possède pas de serveur X puisque notre machine ne possède pas d'interface graphique, c'est uniquement des lignes de commande.

Service réseau	Port de la VM	Port sur la station Linux	Exemple d'utilisation depuis la station Linux
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

Accéder au compte utilisateur simple de la machine virtuelle par ssh :

```
% ssh votreusername@localhost -p 2222
```

puis taper votre mot de passe,

Vous êtes maintenant connecter a votre machine virtuel depuis votre machine principal (les commandes seront quand meme en **rouge** mais vous pourrez les taper ici pour plus de simplicité).

Passer ensuite en compte root avec la même commande que précédament.

Pour tester le bon fonctionnement de ce que vous avez faits essayer d'installer l'éditeur de texte « micro » avec un paquet debian :

```
# apt install micro
```

II. Installation d'un serveur postgres avec PhpPgAdmin

2.1 : installation de Apache :

Pour installer Apache sur la machine on exécute la commande :

```
# apt install apache2
```

vérifier avec cette commande que « active » est en running pour savoir si apache fonctionne :

```
# systemctl status apache2
```

démarré ensuite le serveur apache :

```
# systemctl start apache2
```

pour savoir si le serveur réponds sachant que le serveur n'a pas d'interface graphique, nous devons envoyer une requête manuellement :

```
# telnet localhost 80
```

puis taper ensuite la chaine de caractère suivante :

```
HEAD / HTTP/1.0
```

puis taper 2 fois sur entréer et normalement le serveur devrait renvoyer une réponse positive (200 OK)

Pour accéder a la page web depuis la machine principale, on ouvre un navigateur puis on accède à l'URL : <http://localhost:8080> puis vous devriez on tomber sur la page par défaut d'un serveur Apache

2.2 : installation de PostgreSQL :

installer postgres avec la même commande que celle utilisé de puis le début :

```
# apt install postgresql
```

connecter vous ensuite avec l'utilisateur postgres qui a été créé :

```
# su - postgres
```

puis afficher toutes les bases de données qui sont créées par défaut pour vérifier votre installation :

```
£ psql -l
```

connecter vous ensuite à postgresql toujours avec l'utilisateur postgres :

```
£ psql
```

Créer un utilisateur avec comme nom le nom d'utilisateur, votre nom de login UGA et un mot de passe à retenir :

```
postgres ==> CREATE USER LOGIN_UGA WITH PASSWORD 'mot_de_passe';
```

Il faut ensuite créer une nouvelle base de données avec comme propriétaire le nouveau utilisateur créé :

```
postgres ==> CREATE DATABASE test OWNER LOGIN_UGA;
```

taper les commandes suivantes pour quitter postgresql et vous connecter à votre utilisateur puis vous connecter à la nouvelle base :

```
postgres ==> exit ;
```

```
£ su - LOGIN_UGA
```

```
% psql test
```

Vous devez ensuite créer une nouvelle table simple avec quelques données (un exemple ci-dessous) :

```
test ==> CREATE TABLE hello (  
    collone integer,  
    collone2 integer  
);
```

```
test ==> INSERT INTO hello VALUES (5, 6);  
test ==> INSERT INTO hello VALUES (100, 58);
```

Nous devons ensuite permettre a notre station linux principal de nous connecter à cette base de données et pour cela nous devons modifier 2 fichier de configurations.

Pour commencer connecter vous en root puis modifier le fichier postgresql.conf et pg_hba.conf avec un editeur de texte pour ligne de commande (nano par exemple) :

```
% su
```

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

Modifier le

```
listen_adress = 'localhost' en listen_adress = '*'
```

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

Rajouter une nouvelle ligne juste après le « ipv4 local connection » :

```
host all all 10.0.2.0/24 scram-sha-256
```

Nous allons ensuite redémarrer le service postgres pour prendre en compte les changements :

```
# systemctl restart postgresql
```

Reconnecter vous avec l'utilisateur postgres à la base system :

```
# su - postgres
```

```
£ psql
```

Puis vérifier que les mots de passe ne sont pas visible avec :

```
postgres ==> select * from pg_shadow;
```

2.3 : installation de PHP :

commencons par installer php :

```
# apt install php-common libapache2-mod-php php-cli
```

créer un fichier info.php dans le répertoire /var/www/html/ :

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


dans ce fichier copier coller le texte ci-dessous :

```
<?php
phpinfo();
phpinfo(INFO_MODULES);
?>
```

Sauvegarder bien le fichier avant de quitter.

accéder à la nouvelle page à l'adresse <http://localhost:8080/info.php> sur la machine principal pour voir se qui y s'affiche et vérifier que l'installation est réussi.

2.4 : installation de PhpPgAdmin :

Pour installer PhpPgAdmin nous allons devoir rajouter une nouvelle sources pour les téléchargement puisque PhpPgAdmin n'est pas disponible par défaut

Pour rajouter la souce nous allons modifier un fichier sources.listapt dans /etc/apt/ :

```
# nano /etc/apt/sources.listapt
```

rajouter dans ce fichier une nouvelle ligne avant les derniers commentaires:

```
deb https://deb.debian.org/debian/ bookworm-backports main contib non-free
non-free-firmware
```

Ensuite il faut mettre a jour les paquets disponible puis installer PhpPgAdmin :

```
# apt update
```

```
# apt install phppgadmin
```

Modifier ensuite 2 fichier :

- Connection.php dans /usr/share/phppgadmin/classes/database/ :

```
# nano /usr/share/phppgadmin/classes/database/Connection.php
```

puis changer le 14 par 15 dans le fichier pour avoir la bonne version de postgres

- phppgadmin.conf dans /etc/apache2/conf-available/ :

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

dans ce fichier modifier

```
# Only allow connections from localhost:  
Require local
```

par

```
# Only allow connections from localhost:  
Require all granted
```

redémarrer en ensuite le serveur apache :

```
# systemctl restart apache2
```

Nous allons maintenant essayer de faire une requete sql avec PhpPgAdmin depuis la machine principal pour vérifier que tout est bien installée.

Connecter vous à l'adresse suivante :

<http://localhost:8080/phppgadmin/>

accéder à votre serveur postgres, connecter vous avec votre utilisateur ainsi que son mot de passe.

Ensuite vous pouvez essayer de faire des selections dans votre base test dans l'onglet SQL, par exemple :

```
SELECT * FROM hello ;
```

2.5 : Afficher les informations de votre installation :

Depuis la machine virtuelle :

exécuter la commande :

```
# /sbin/blkid
```

puis copier la page php sur le serveur de l'iut :

```
scp NOM_DE_LOGIN@transit.iut2.univ-grenoble-alpes.fr:/users/info/www/intranet/  
enseignements/S2.03/page_sae_S2.03.php /var/www/html/
```

Vous pouvez ensuite vous connecter depuis la machine principal a l'adresse :

http://localhost:8080/page_sae_S2.03.php