

Documentation technique

Création de 2 serveurs Linux pouvant se synchroniser automatiquement toutes les 5 minutes et accessibles seulement depuis un serveur reverse proxy.

Nombre tâches	Progressior	Jours	Ecart				chef de groupe : Guilla		
5		1	0						
#	Dépend de	Assignée à	Description	Niveau	Jours	Début	Fin (prévue)	Fin (réelle)	Statut
1		Tous	Répartition des tâches	1	1	22/09/2022 - 14h50	22/09/2022 - 15h00	22/09/2022 - 14h55	Terminée
2		Maël	Recherche sur la mise en place du round robin		1	22/09/2022 - 15h00	22/09/2022 - 17h00	22/09/2022 - 17h20	Terminée
3		Océane	Recherche sur comment synchronisation et automatiser des serveurs		1	22/09/2022 - 15h00	22/09/2022 - 17h00	22/09/2022 - 17h40	Terminée
4		Guillaume	Mise en place et test des recherches sur la mise en place du Round Robin et sur la synchronisation et		2	22/09/2022 - 15h30	22/09/2022 - 17h30	23/09/2022 - 19h00	Terminée
5	2,3 et 4	Eloise	Documentation technique		2	22/09/2022 - 15h35	22/09/2022 - 18h00	23/09/2022 - 19h30	Terminée

Table des matières

1. Définition
2. Prérequis
3. Etape 1 : Paramétrer une machine virtuelle
4. Etape 2 : Installation du serveur
5. Etape 3 : Installer les paquets sur le serveur et prise en main à distance
6. Etape 4 : Cloner le serveur (La machine virtuelle) pour créer le second serveur
7. Etape 5 : Configuration réseau des 3 serveurs
8. Etape 6 : Configuration du lien SSH entre les 2 serveurs et synchronisation automatique des 2 serveurs

1. Définitions

Linux : Linux est un système d'exploitation de type Unix. Il a été conçu pour équiper les ordinateurs personnels d'un système d'exploitation gratuit ou à très faible coût, comparable aux versions Unix classiques, généralement plus coûteuses.

(source : <https://www.lemagit.fr/definition/Linux>)

Serveur Proxy : Un serveur proxy joue le rôle de passerelle entre Internet et l'utilisateur. C'est un serveur intermédiaire qui sépare les utilisateurs, des sites Web sur lesquels ils naviguent. Les serveurs proxy assurent différents niveaux de fonctionnalité, de sécurité et de confidentialité, selon votre type d'utilisation, vos besoins ou la politique de votre entreprise.

(source : <https://www.varonis.com/fr/blog/serveur-proxy>)

Reverse Proxy : Un proxy inverse ou serveur mandataire inverse est un type de serveur, habituellement placé en frontal de serveurs web. Contrairement au serveur proxy qui permet à un utilisateur d'accéder au réseau Internet, le proxy inverse permet à un utilisateur d'Internet d'accéder à des serveurs internes. (source : https://fr.wikipedia.org/wiki/Proxy_inverse)

Load-balancing : Le Load Balancing consiste à répartir efficacement le trafic réseau entrant sur un groupe de serveurs, également connu sous le nom de parc de serveurs ou de pool de serveurs. (source : <https://actualiteinformatique.fr/cloud/definition-load-balancing>)

HaProxy : HAProxy est une solution libre, fiable et très performante de répartition de charge de niveau 4 (TCP) et 7 (HTTP). (source : <https://linuxfr.org/news/haproxy-le-repartiteur-de-charge-fiable-et-performant>)

Apache2 : Apache est un logiciel de serveur web gratuit et open-source.

Roundrobin : Le roundrobin est une répartition de charge (load balancing) équitable entre serveurs d'une ferme informatique (cluster). Chaque serveur traite le même nombre de requêtes. (source : <https://fr.wikipedia.org/wiki/Round-robin>)

2. Prérequis

Pour créer nos serveurs Linux qui peuvent se synchroniser nous avons besoin d'un logiciel de virtualisation, type VirtualBox ou VmWare Workstation.

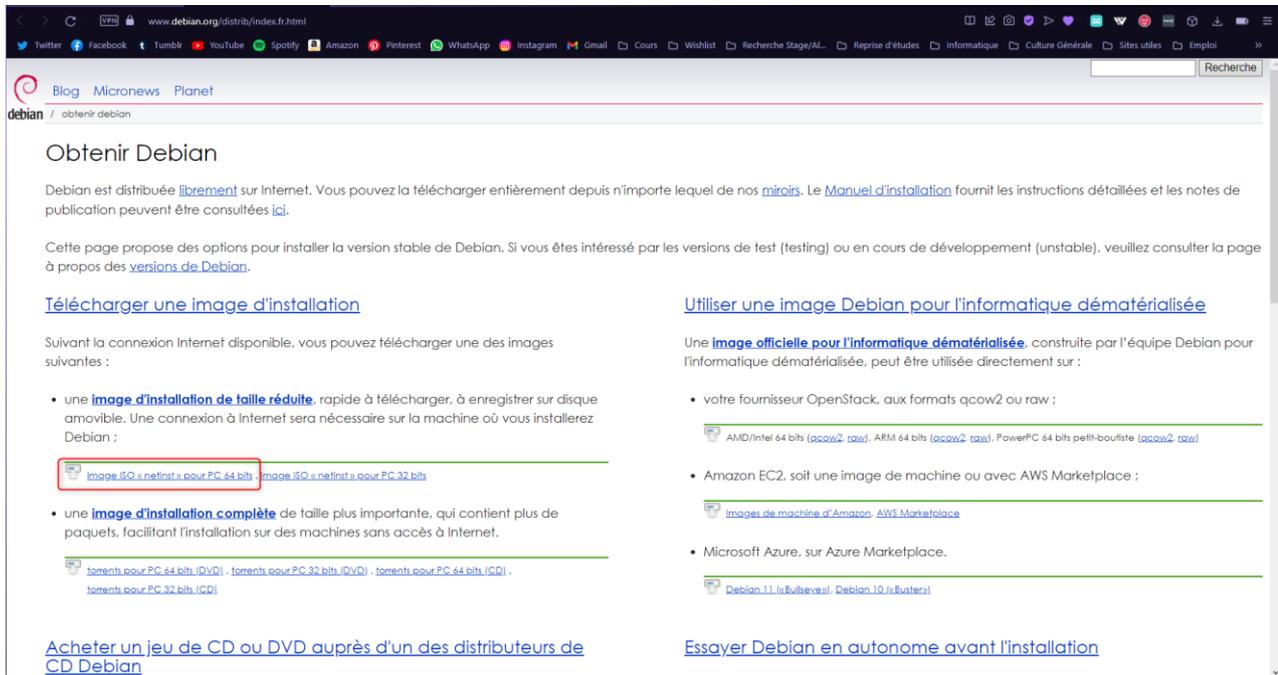
Pour fonctionner, prévoir d'allouer à minima pour chaque machine virtuelle :

- 1 Gb de mémoire
- 1 Processeur
- 20 Gb pour le disque dur

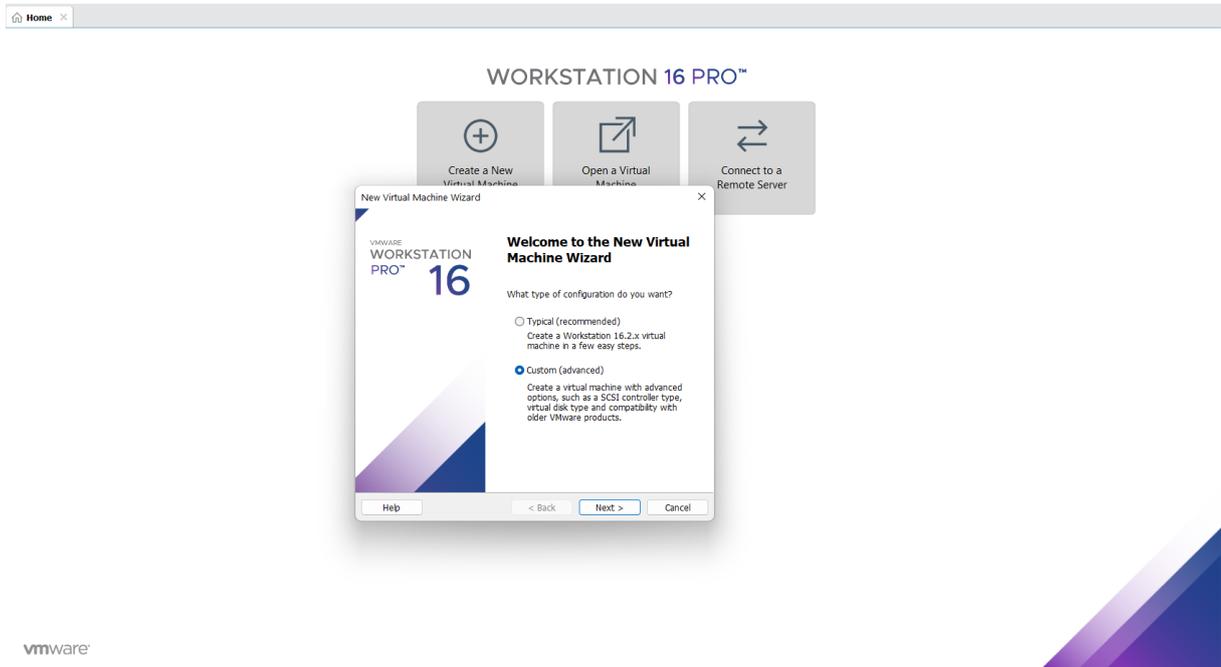
Votre PC doit disposer d'autre moins 8 Go de RAM et d'une carte réseau.

ETAPE 1 – Paramétrer une machine virtuelle

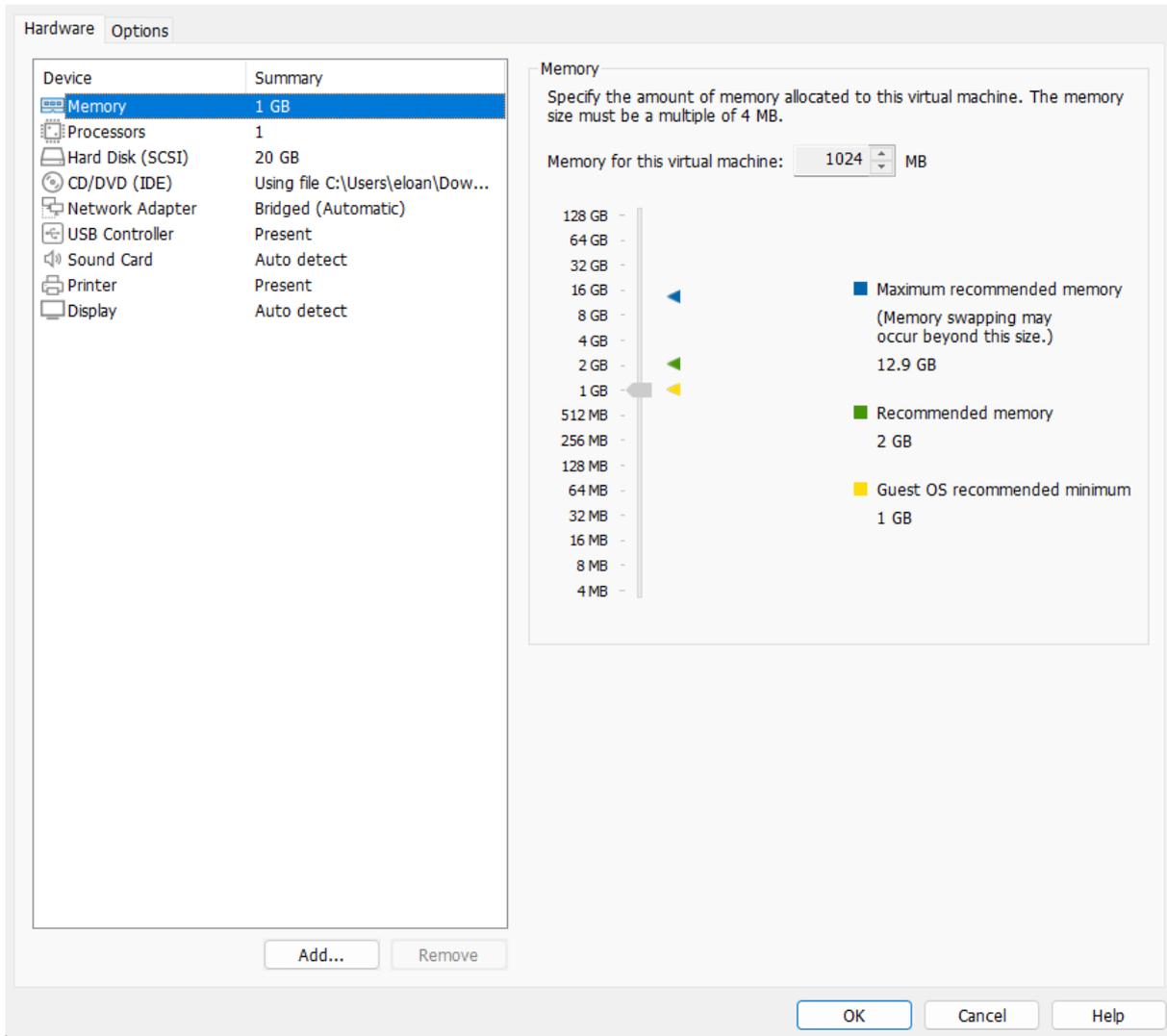
Nous allons tout d’abord télécharger Debian depuis : <https://www.debian.org/distrib/index.fr.html>. Ici nous utiliserons une image d’installation de taille réduite.



Nous allons ensuite créer une machine Virtuelle (ici depuis le logiciel VMware Workstation)



Une fois notre machine virtuelle créée, nous allons procéder aux paramétrages.



Mémoire → Mettre environ 1GB de mémoire

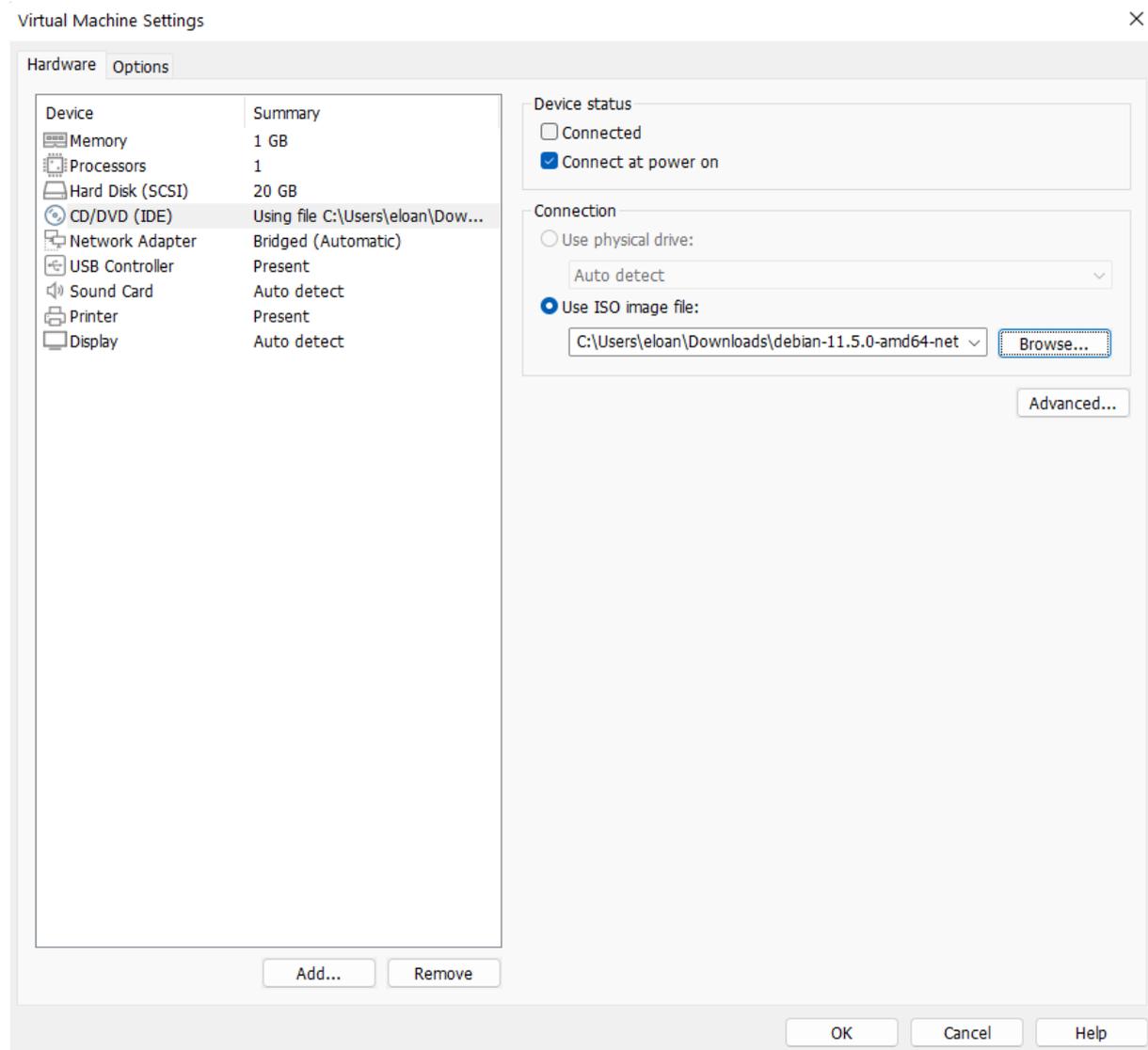
Processeur → Un seul processeur suffit

Disque dur → Mettre 20 GB

Network → Bridged (Automatique)

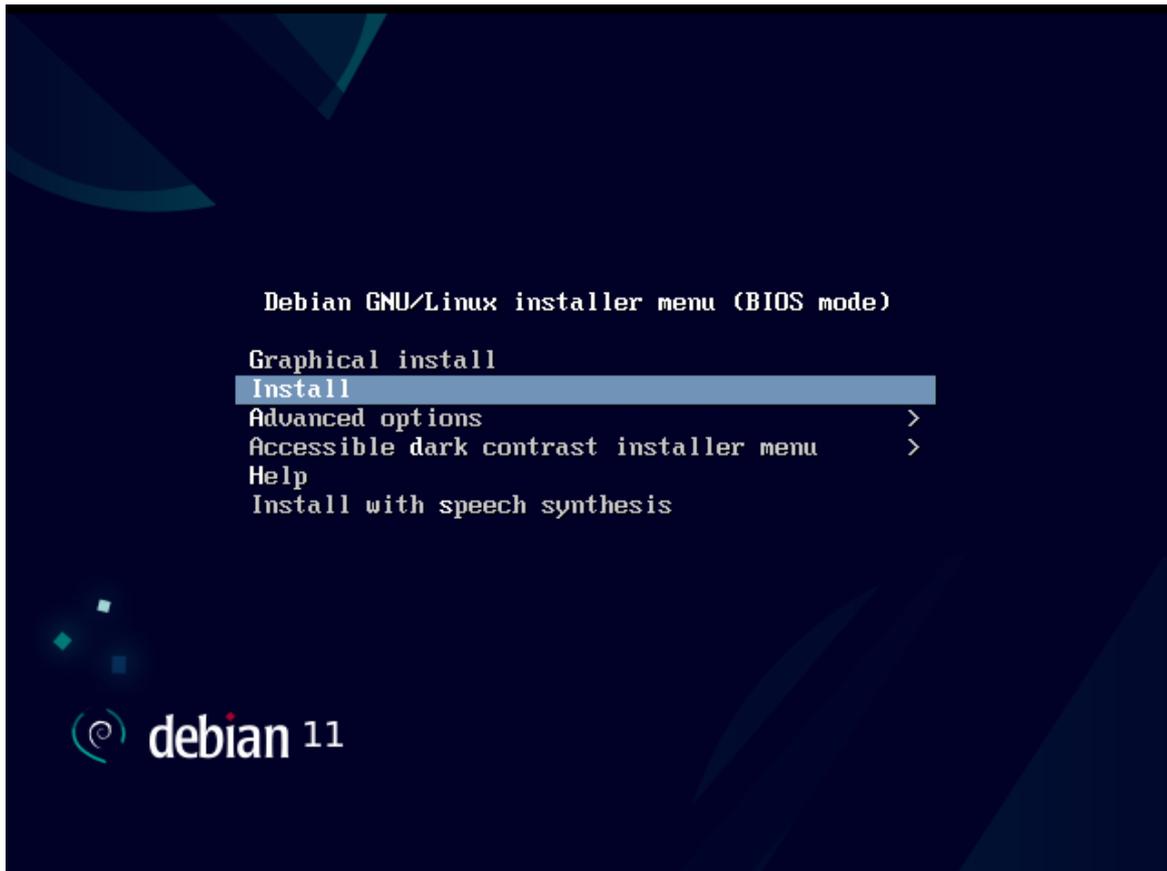
Les autres paramètres peuvent être laissés par défaut.

Pour la partie CD/DVD → Nous allons pouvoir insérer notre image précédemment téléchargée dans « Use ISO image file » comme ci-dessous :

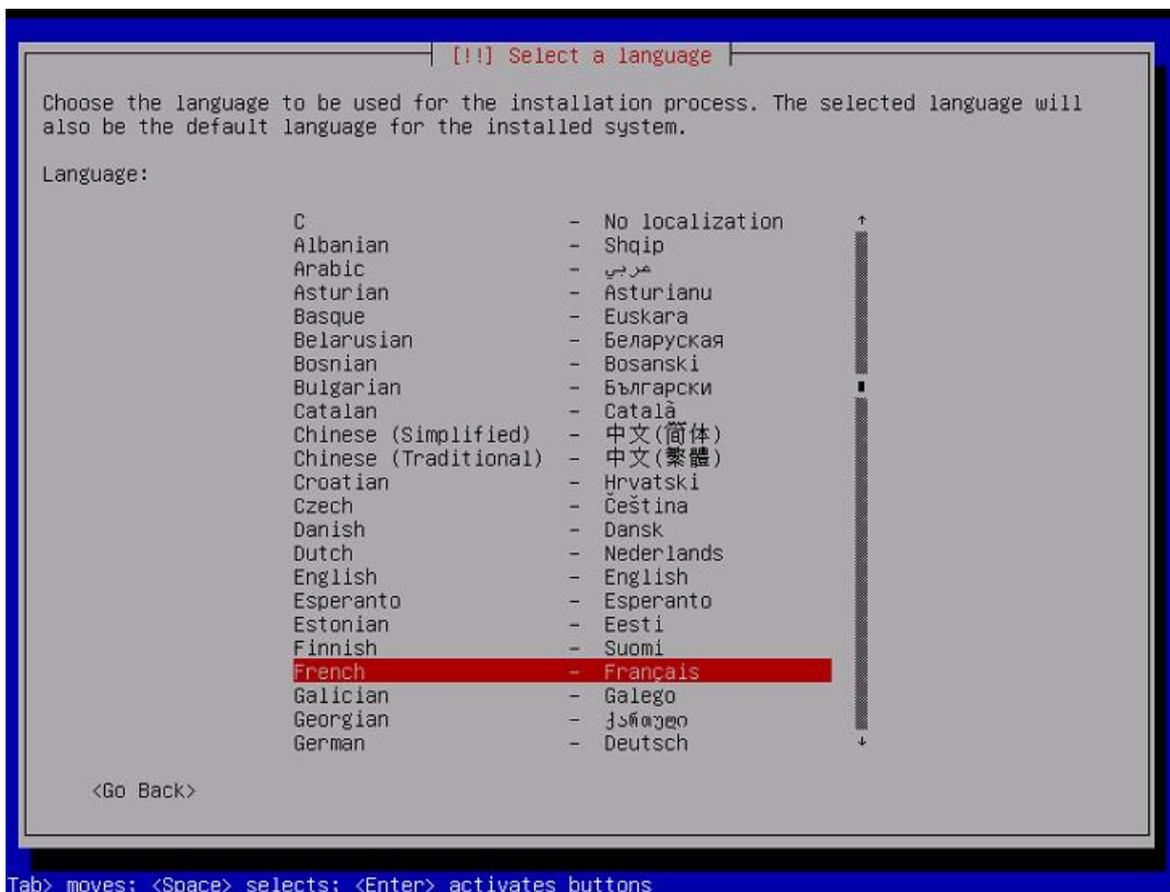


ETAPE 2 : Installation du serveur

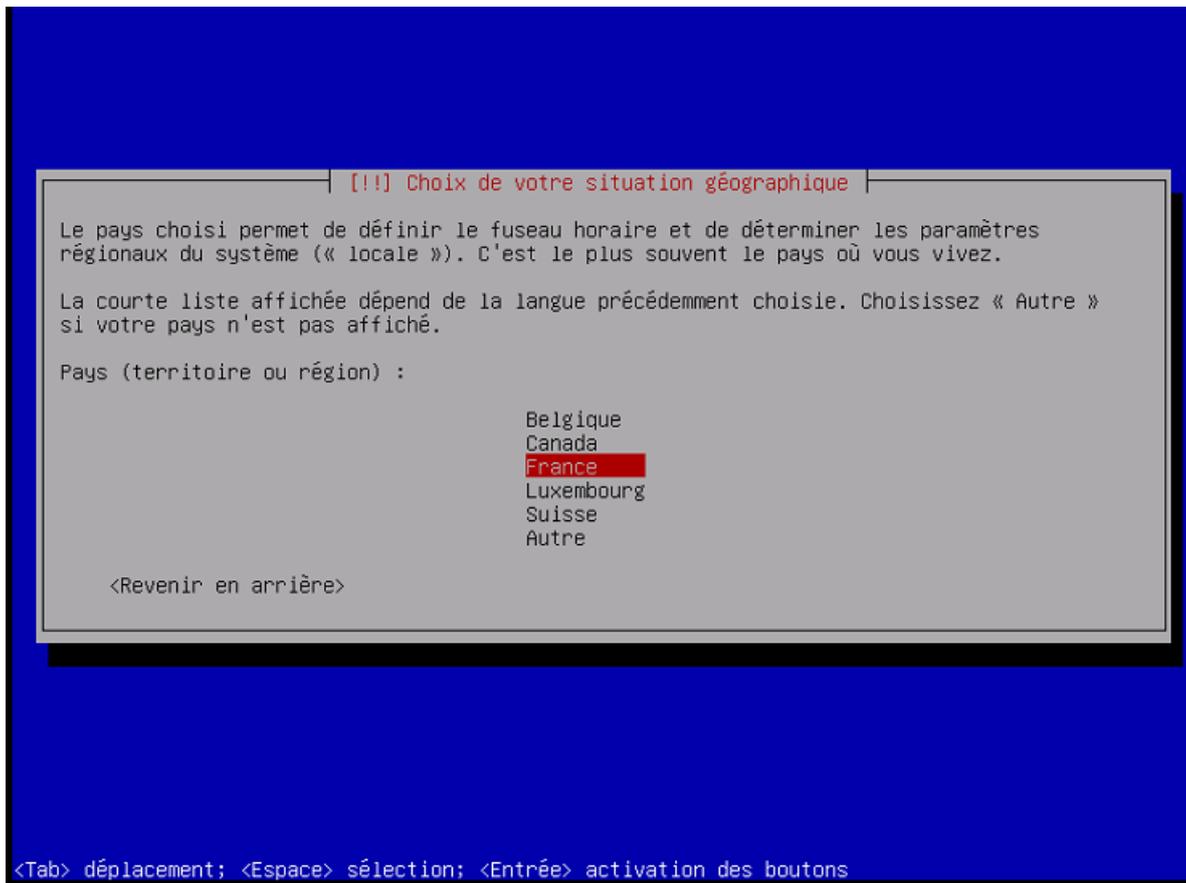
Allumer la machine virtuelle. Depuis l'écran d'accueil, sélectionner « INSTALL »



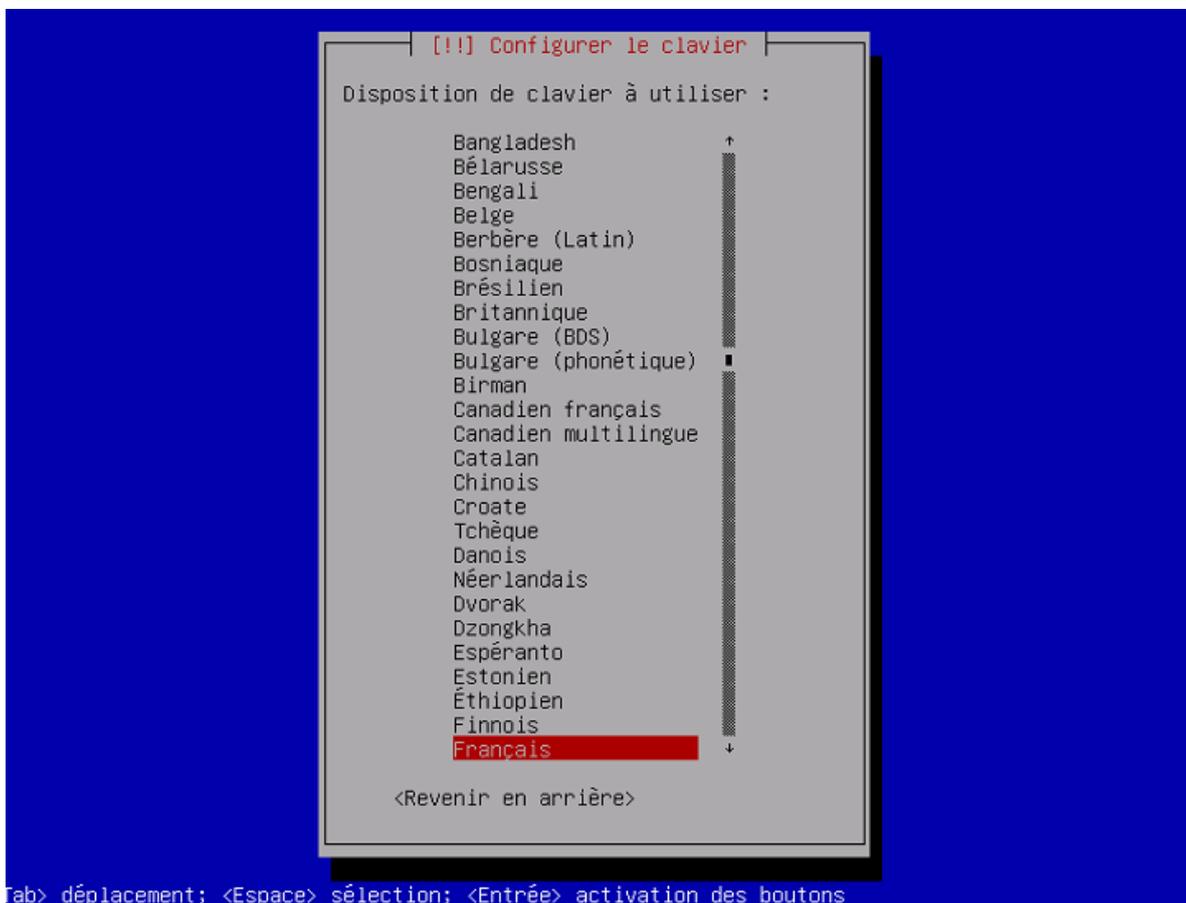
Dans l'écran suivant, choisir le langage, ici « French - Français »



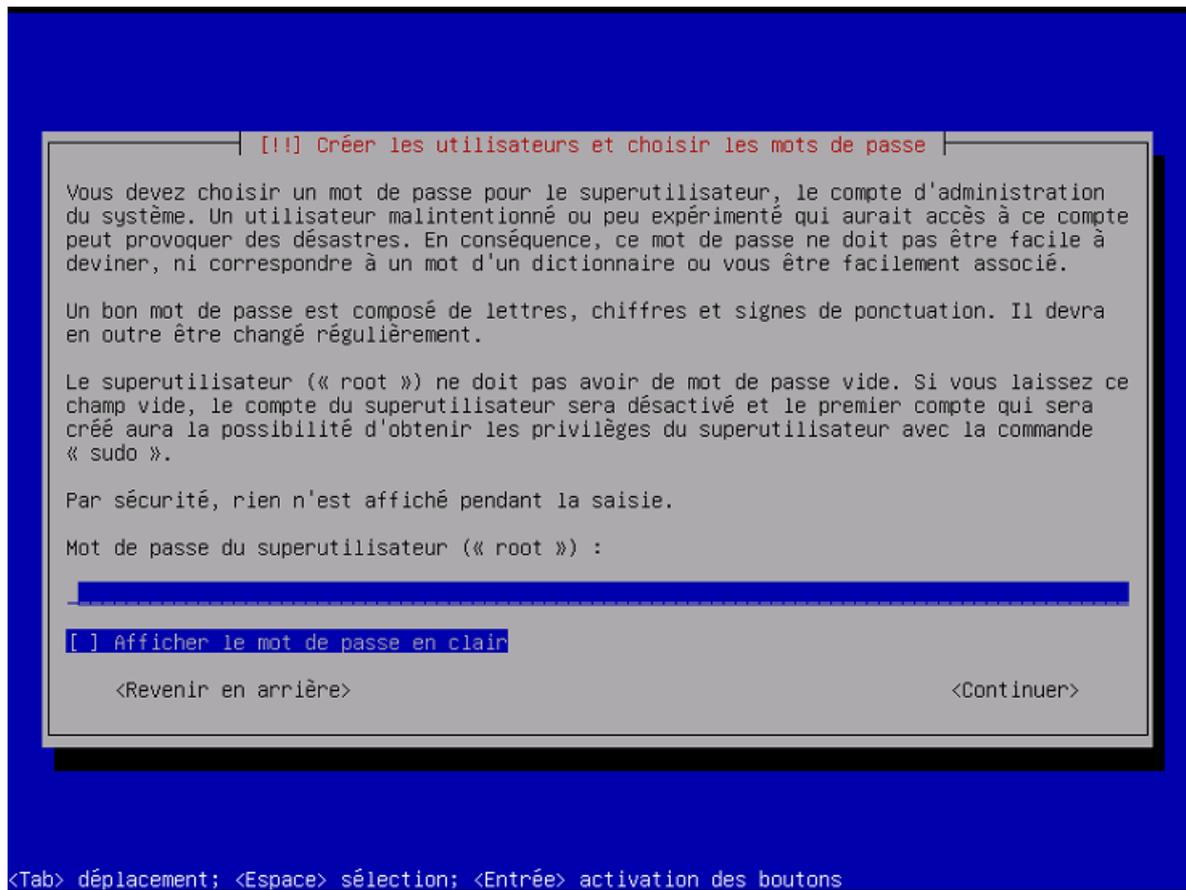
Choisir la situation géographique, ici « France »



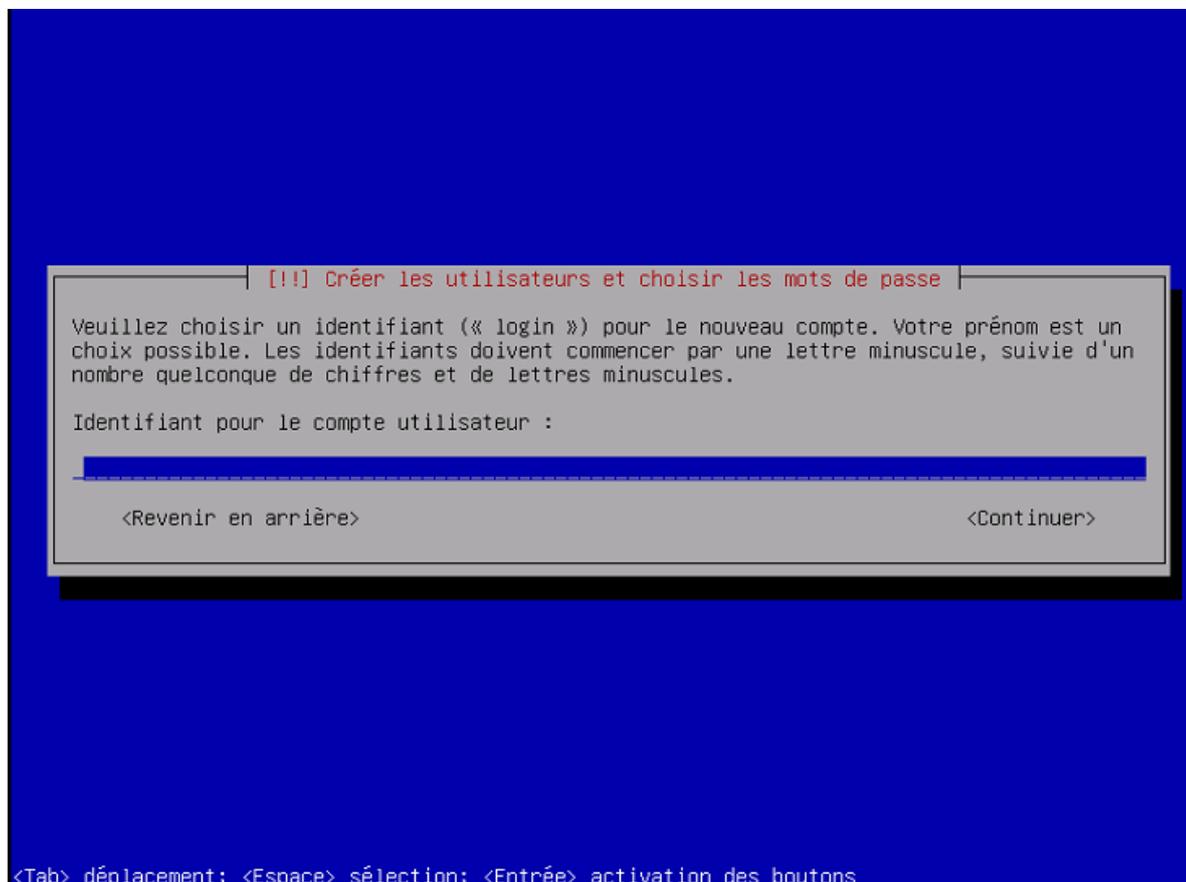
Choisir la configuration clavier – Ici « français »

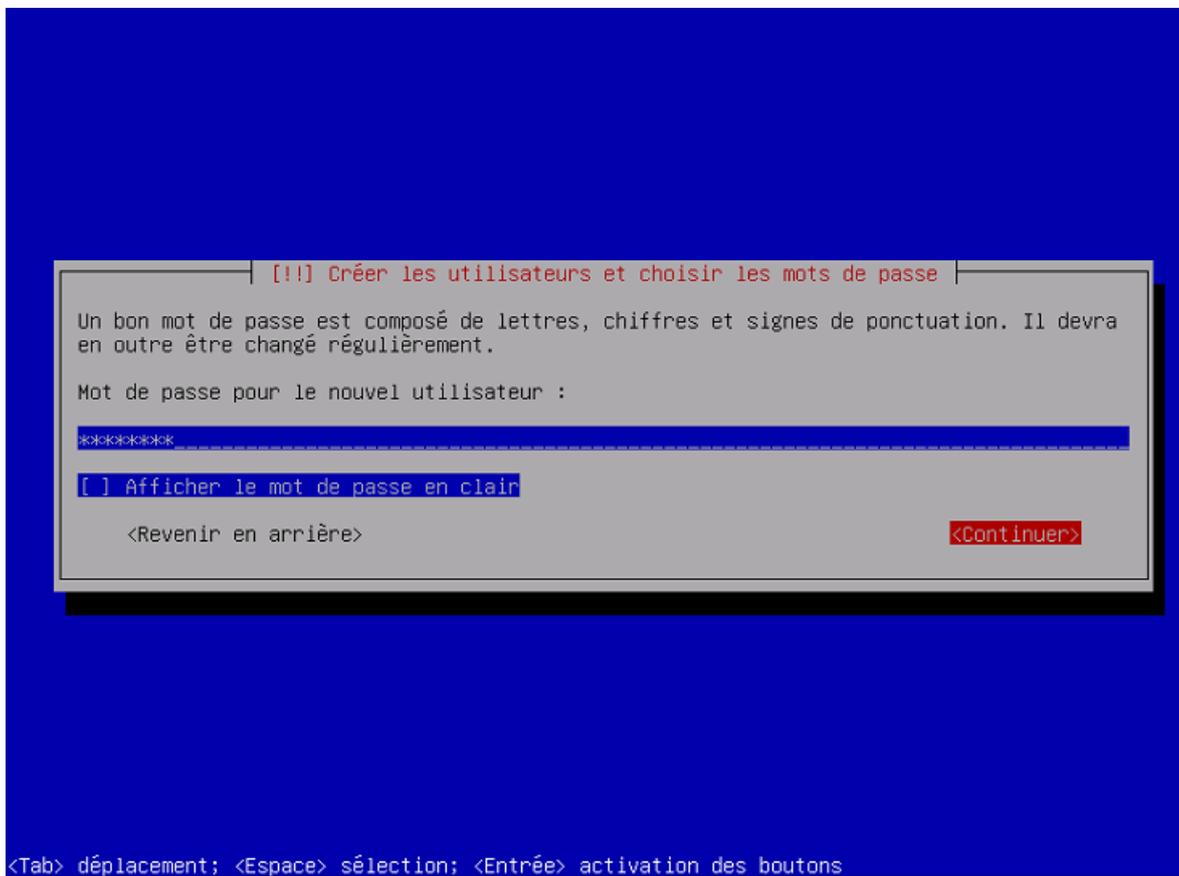


Création des utilisateurs – Choisir un nom de superutilisateur puis le mot de passe associé.

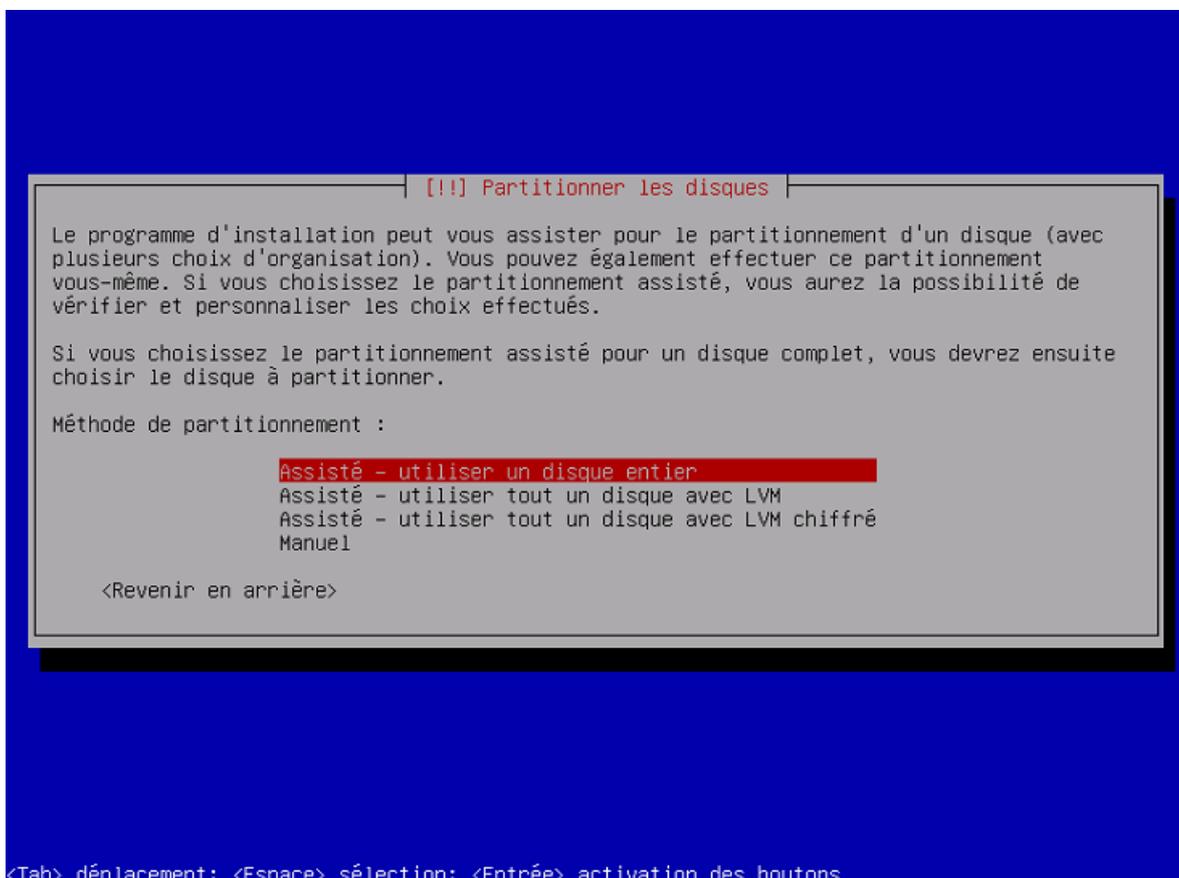


Dans l'écran suivant, choisir le nom d'utilisateur puis le mot de passe associé

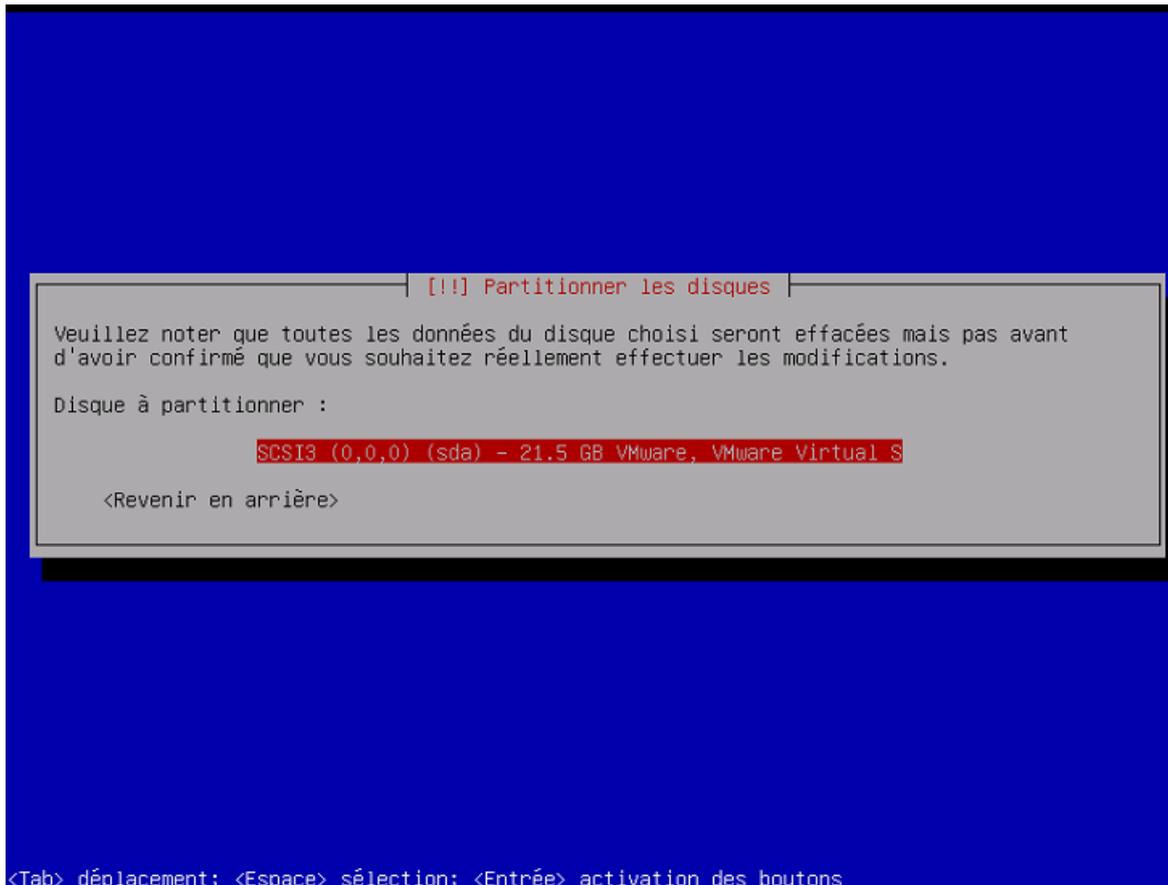




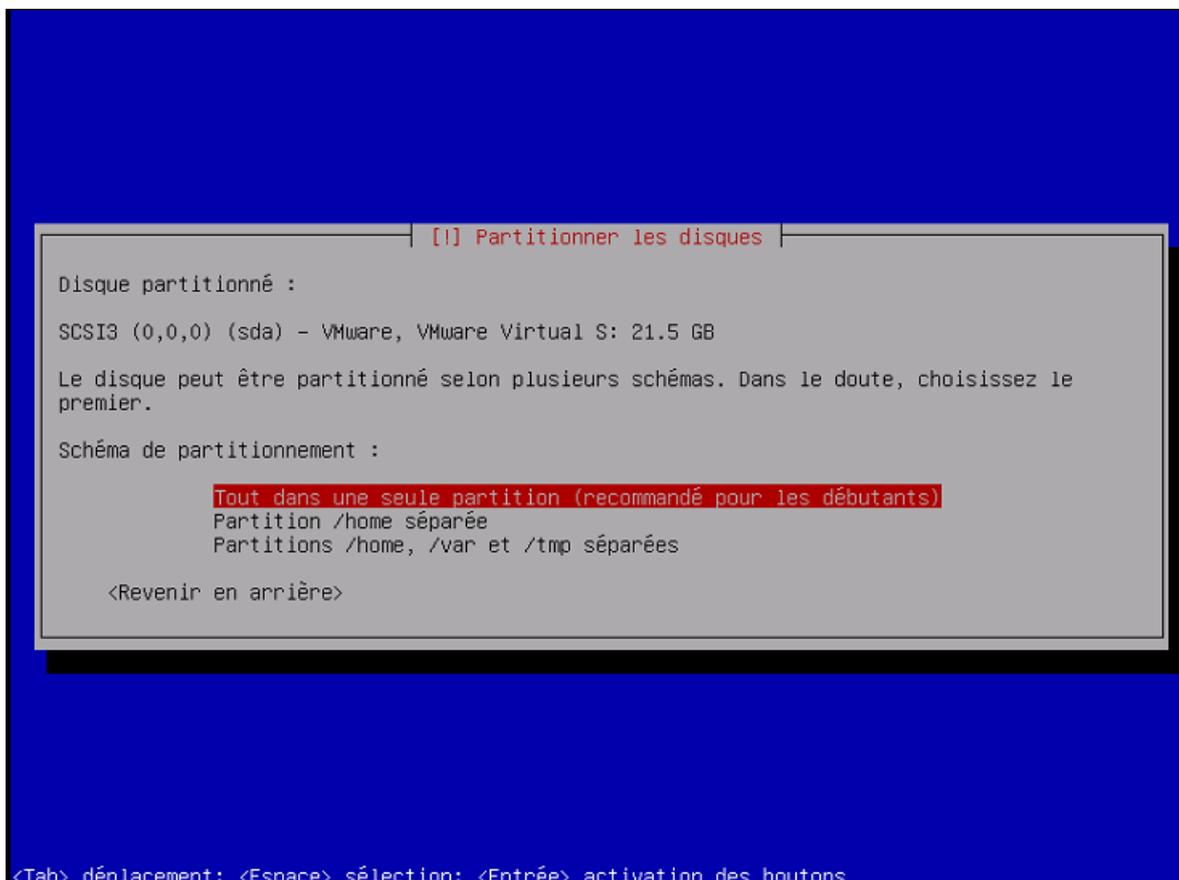
Dans « Partitionner les disques » sélectionner « Assisté – utiliser un disque entier »



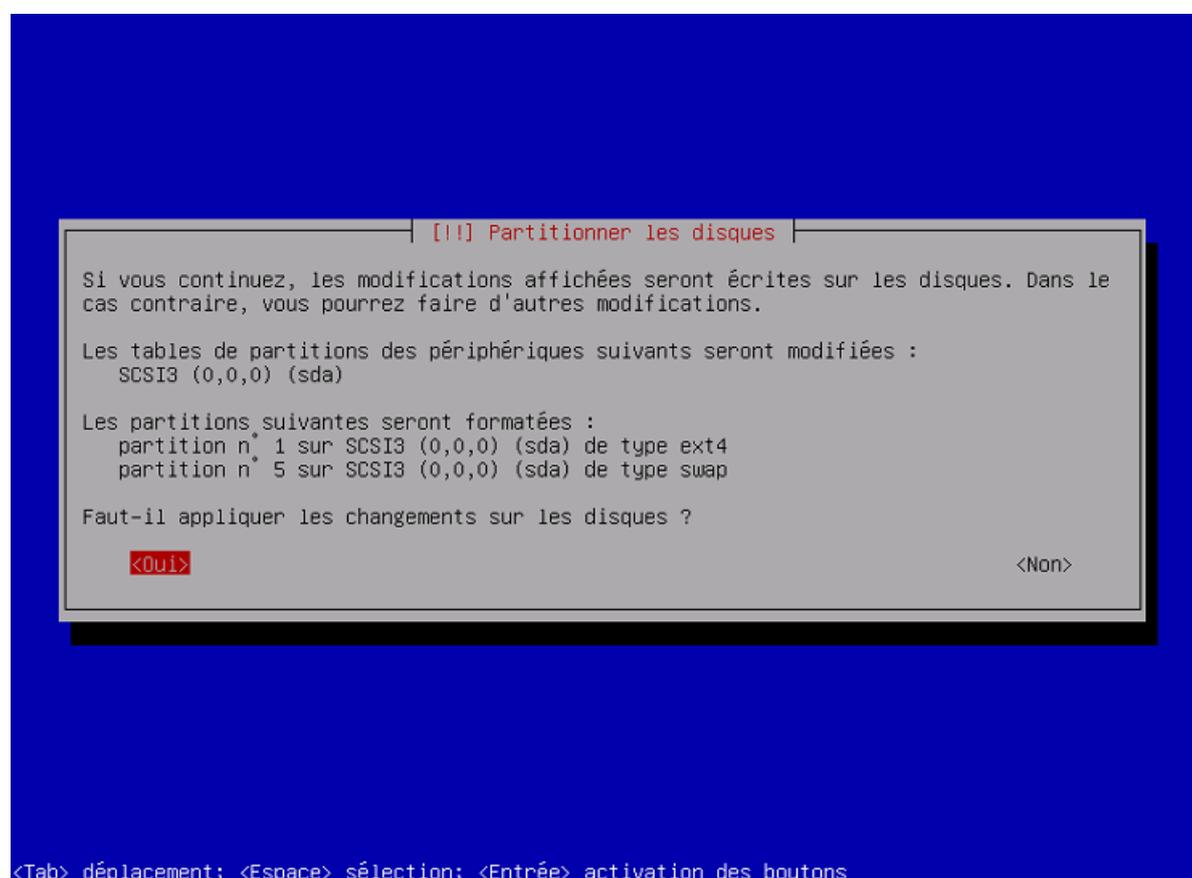
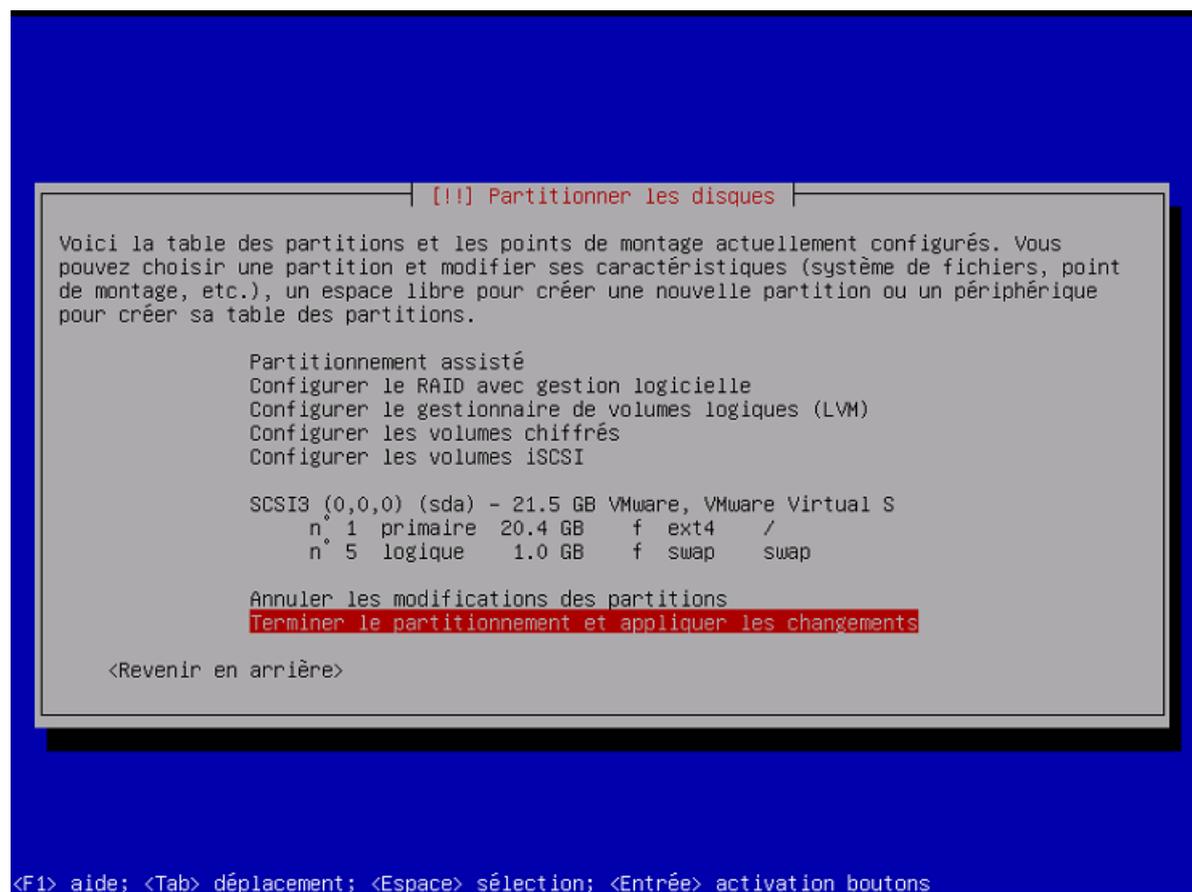
Dans l'écran suivant, sélectionner le seul disque qui apparaît :



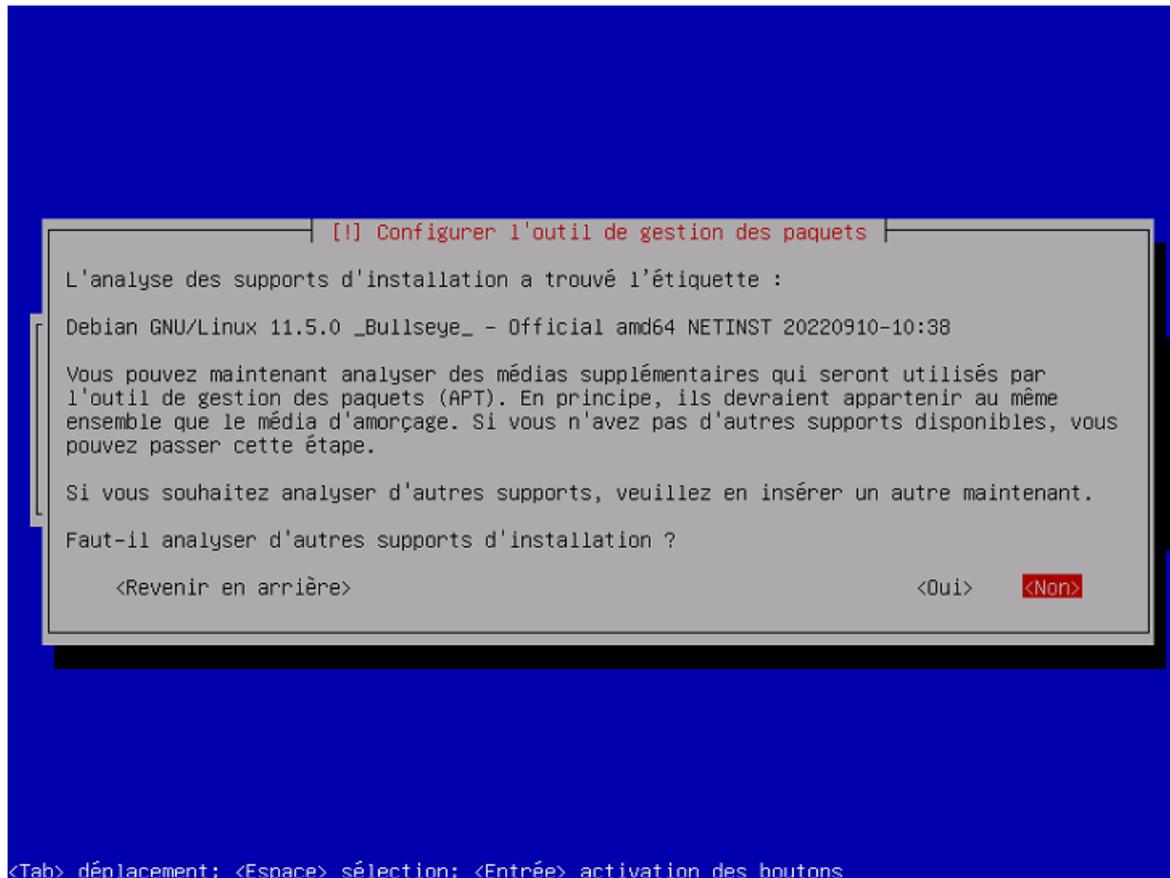
Dans l'écran « Schéma de partitionnement » sélectionner « Tout dans une seule partition » (recommandé)



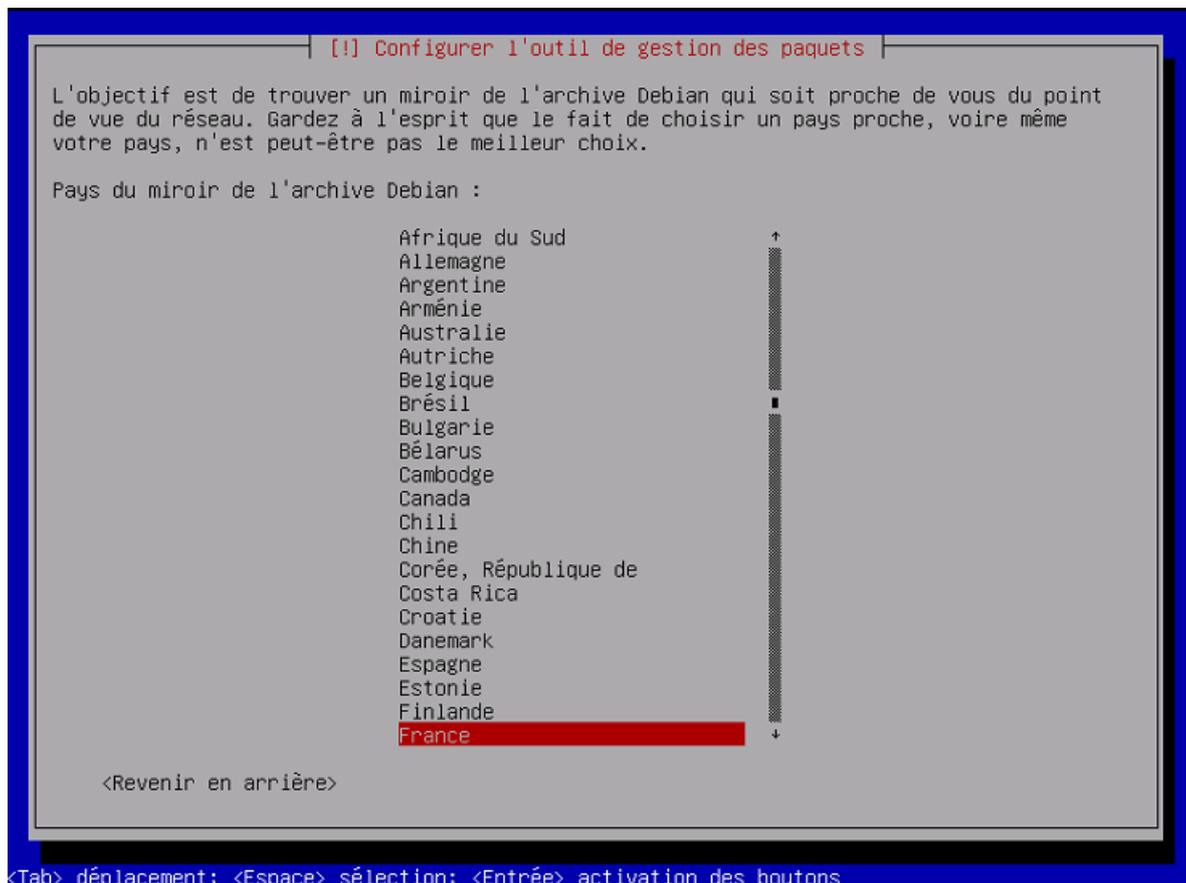
Dans l'écran suivant, cliquer sur « Terminer le partitionnement et appliquer les changements » pour terminer la partie partition des disques puis sélectionner « oui » pour appliquer ces changement.



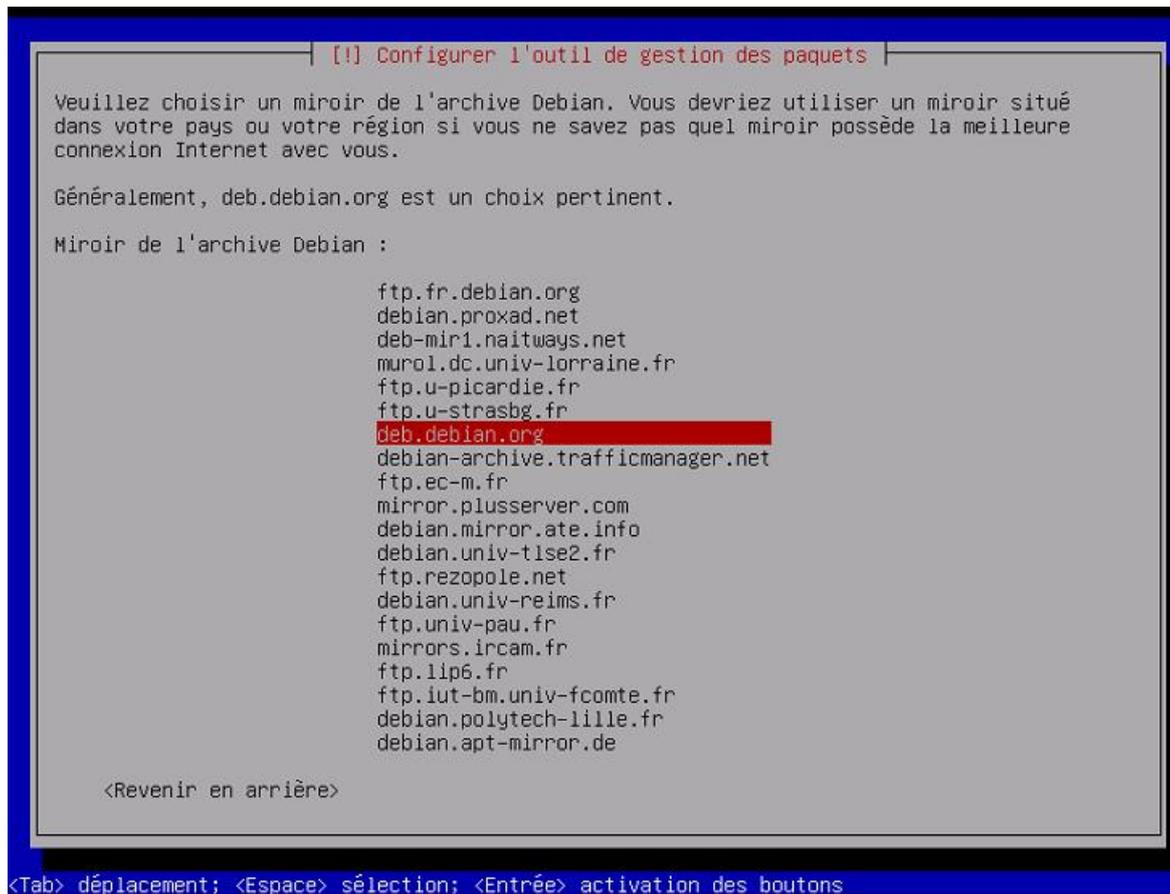
Dans « configurer l'outil de gestion des paquets » répondre « non » à « analyser d'autres support d'installation »



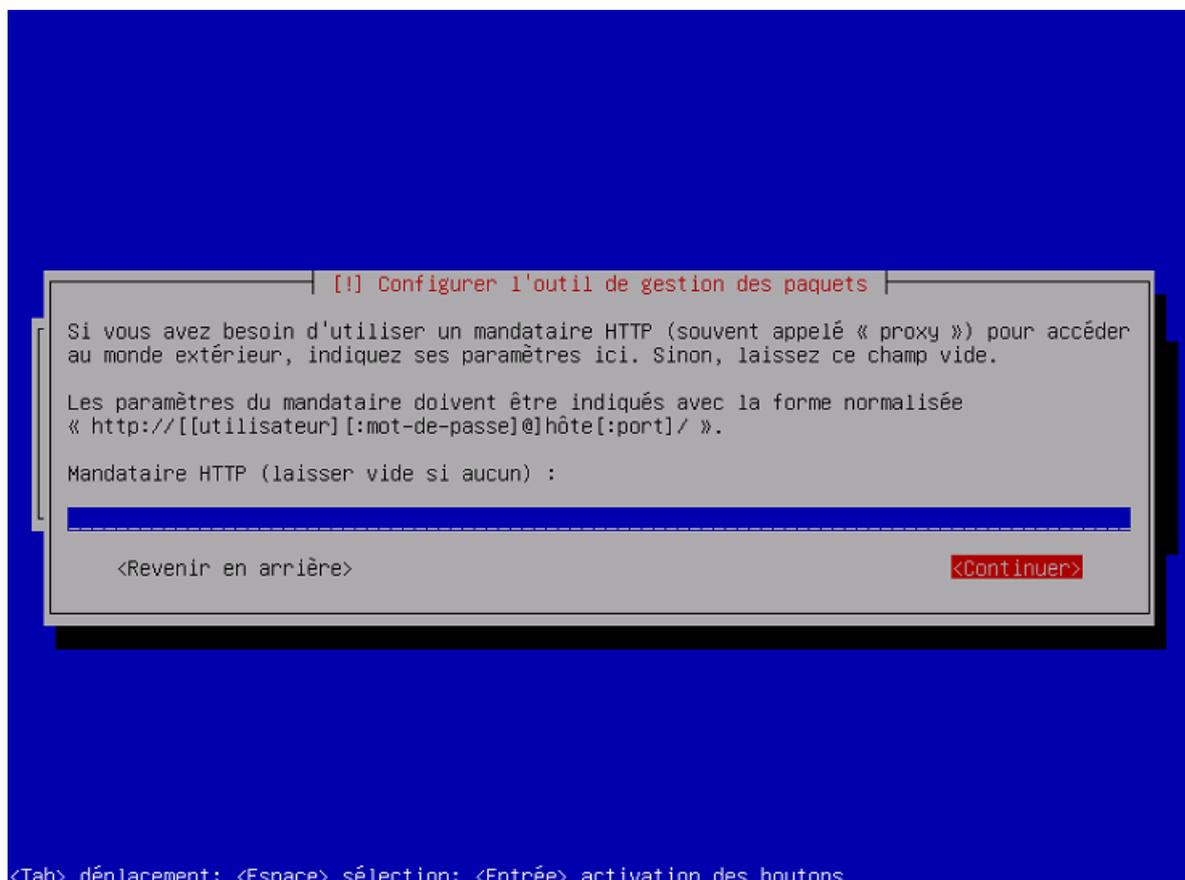
Toujours dans la configuration de gestion des paquets, sélectionner le pays miroir de l'archive Debian, ici « France ». (Note : sélectionner le pays où vous vous trouvez)



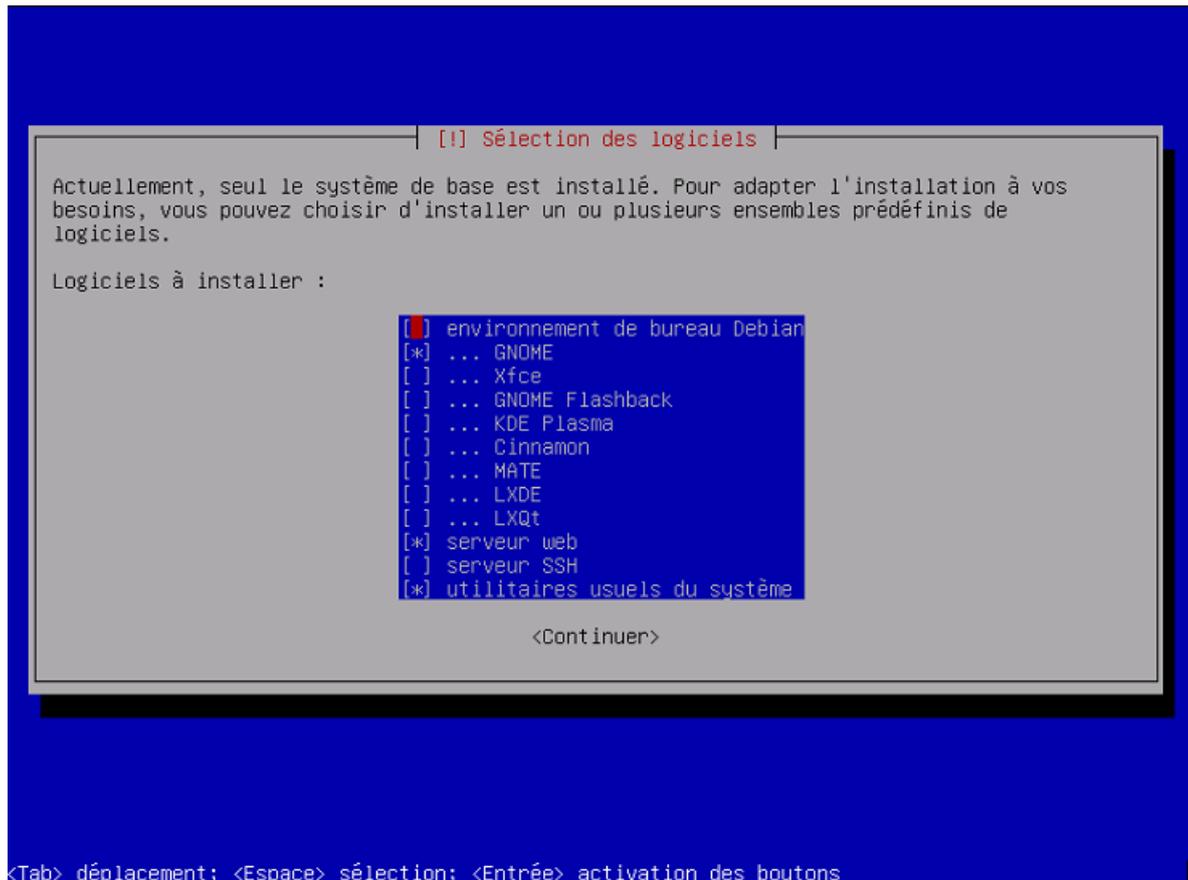
Dans la sélection du miroir de l'archive Debian, sélectionner « deb.debian.org »



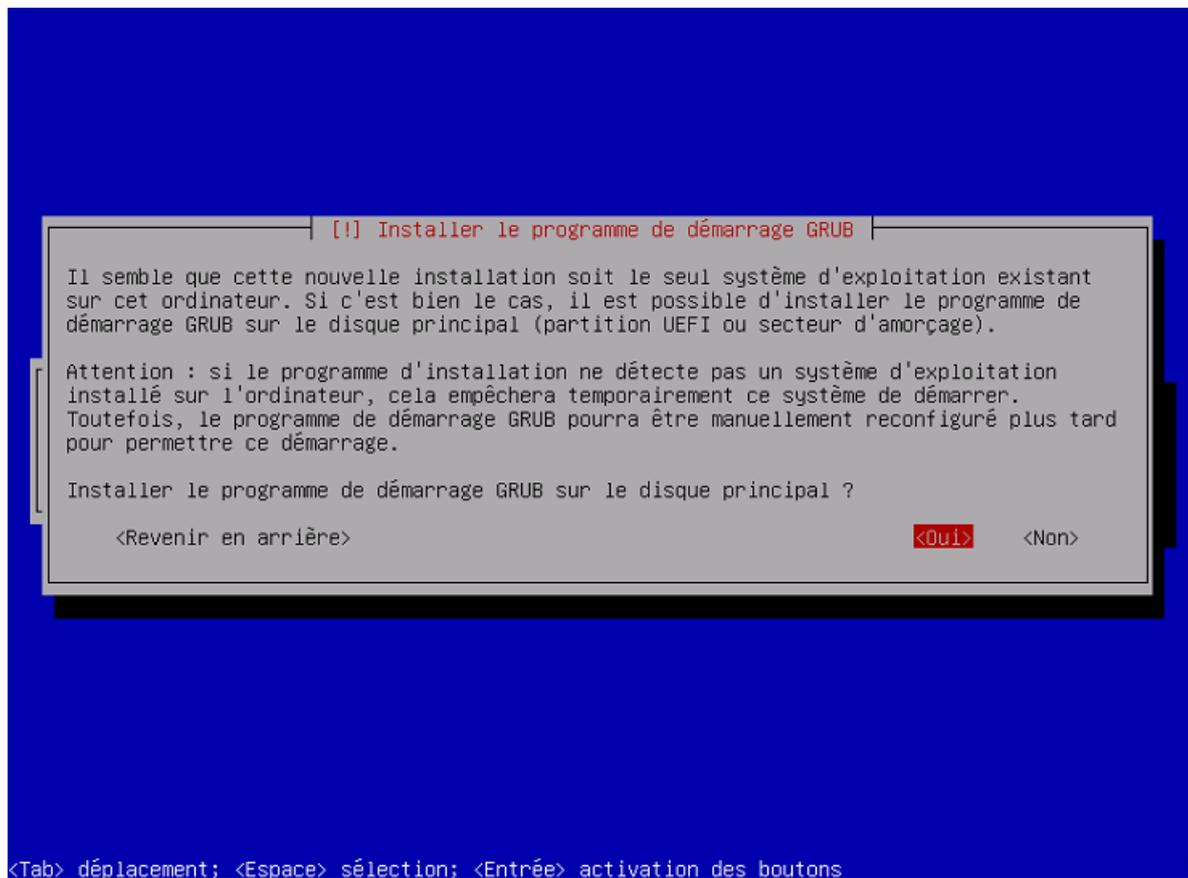
Dans l'écran suivant « Mandataire http », laisser le champ vide.



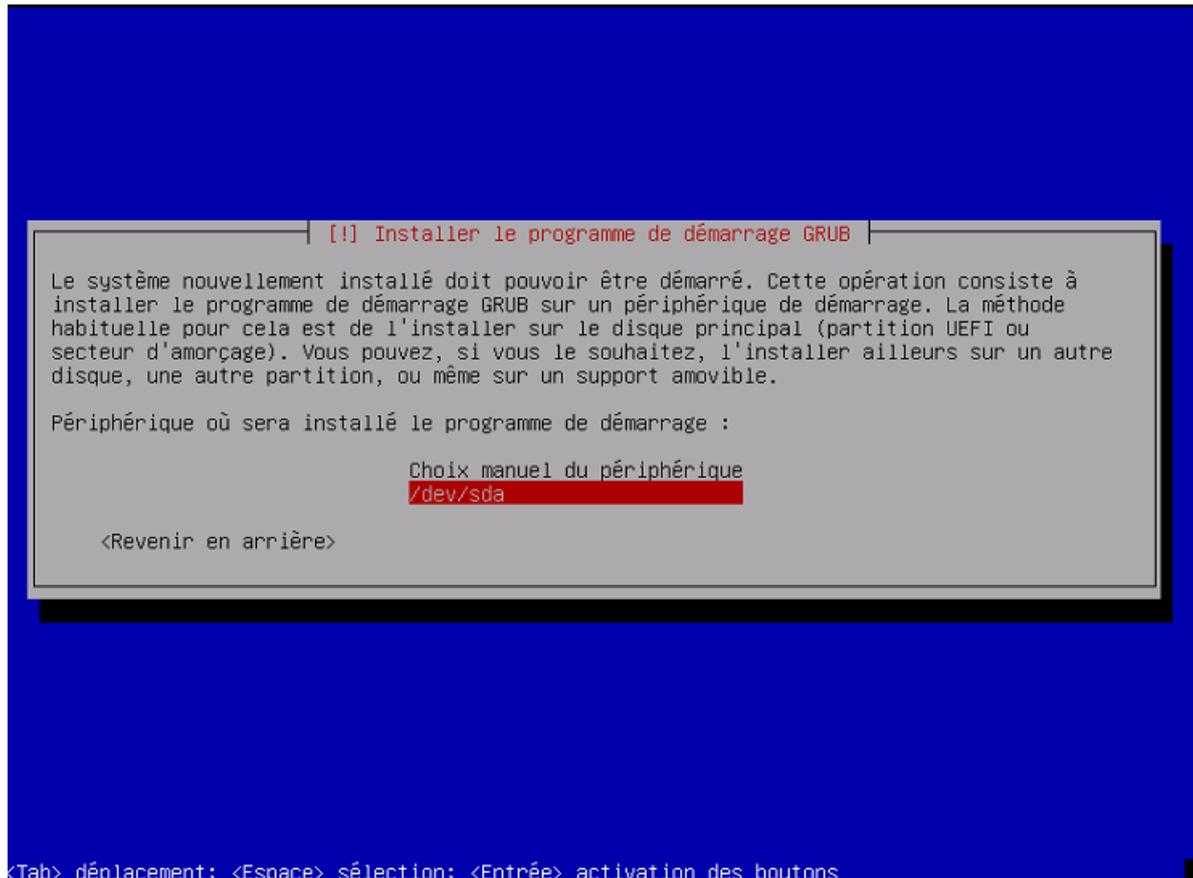
Dans l'écran « Sélection des logiciels » décocher « environnement de bureau Debian » et « GNOME » à l'aide de la touche « espace ». Laisser uniquement coché « utilitaire usuels du système ». Puis « continuer ».



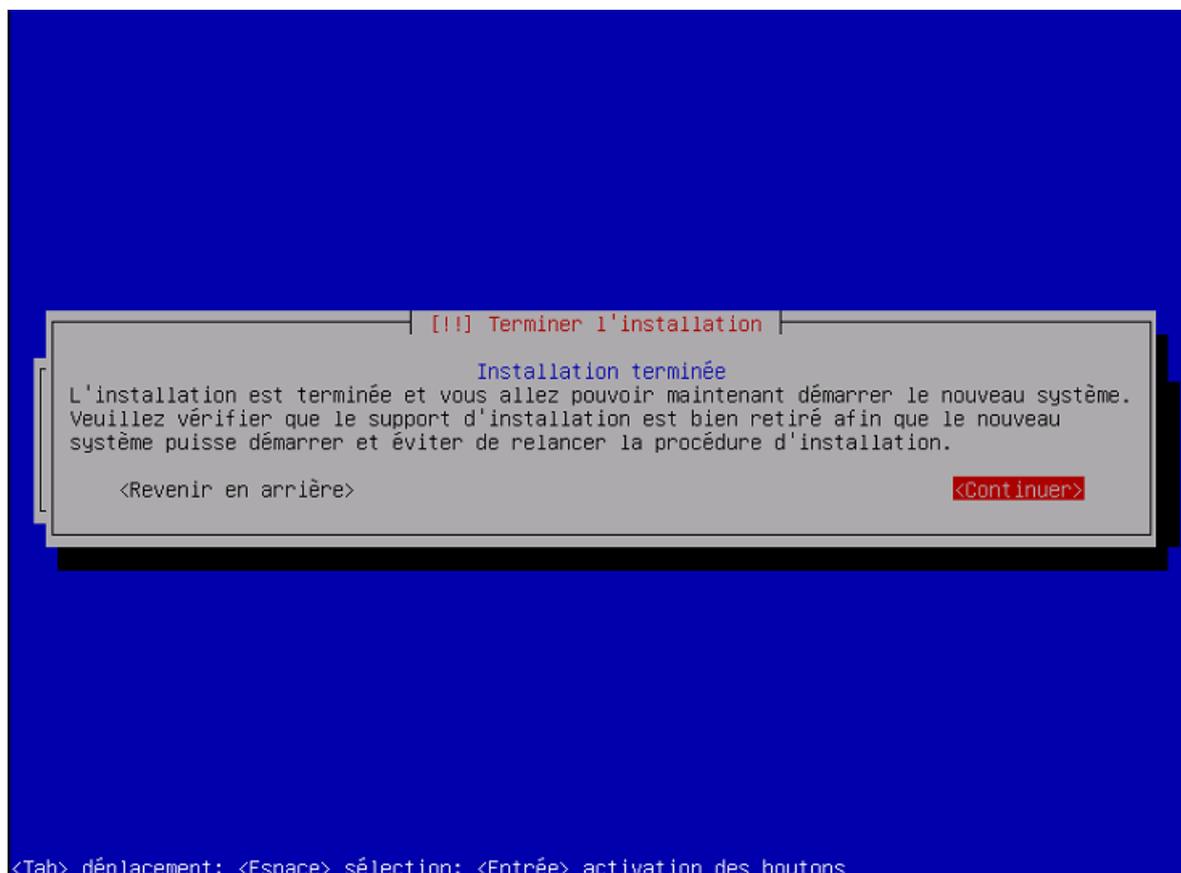
Dans l'écran « Installer le programme de démarrage GRUB » sélectionner « Oui ».



Dans l'écran suivant, sélectionner le seul périphérique proposé, ici « /dev/sda »



Dans le dernier écran, cliquer sur « terminer » pour terminer l'installation.



Etape 3 : Installer les paquets sur le serveur et prise en main à distance

Après le redémarrage de la machine virtuelle, on se retrouve sur l'interface du serveur. Commencer par écrire la commande « ip a » dans l'invite de commande pour connaître son ip.

Installer ensuite les logiciels nécessaires pour le serveur apache en indiquant la commande suivante :

```
root@haver:~# apt install apache2 php libapache2-mod-php mysql-server php-mysql_
```

Il faut ensuite se déplacer dans le fichier de la page web avec la commande suivante :

```
root@haver:~# cd /var/www/html
```

Test du serveur grâce à une page HTML. Modifier la page index.html déjà présente avec les informations souhaitées. (Ici, une simple page avec des liens et une vidéo youtube)

```
GNU nano 5.4 /var/www/html/index.html
<!doctype html>
<html lang="fr">
<head>
  <meta charset="utf-8">
  <title>Ma page</title>
</head>
<body>
<ul style="list-style-type:none;">
  <li><a style="color:#00B7FF" href="https://www.google.fr">Lien vers Google</a></li>
  <li><a style="color:#E60000" href="https://www.youtube.com">Lien vers Youtube</a></li>
  <li><a style="color:#FF8C00" href="https://www.amazon.fr">Lien vers Amazon</a></li>
  <li><a style="color:#A60000" href="https://www.netflix.com">Lien vers Netflix</a></li>
  <iframe width="710" height="480" src="https://www.youtube.com/embed/dYuCKN900I0" frameborder=
</iframe>
</ul>
</body>
</html>
```

Sur votre PC personnel (hors machine virtuelle donc), ouvrir votre navigateur, dans la barre d'adresse, indiquer l'adresse IP du serveur précédemment recherchée. La page créée dans le serveur s'affiche :



[Lien vers Google](#)
[Lien vers Youtube](#)
[Lien vers Amazon](#)
[Lien vers Netflix](#)



De retour sur le serveur, installer ensuite les service openssh pour pouvoir prendre la main du serveur Linux depuis notre PC personnel grâce à la commande suivante :

```
root@haver:/var/www/html# apt install openssh-server_
```

Ensuite, se connecter sur le serveur Debian depuis Windows Powershell du PC personnel. Pour cela, indiquer « ssh » suivi du nom d'utilisateur @adresse ip du serveur, comme ci-dessous :

```
PS C:\Users\duval> ssh haver@192.168.22.59
The authenticity of host '192.168.22.59 (192.168.22.59)' can't be established.
ECDSA key fingerprint is SHA256:bPi6/0o2lNhh3+d0k1sven+AZtmlqRnXm8bxKyK7beI.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.22.59' (ECDSA) to the list of known hosts.
haver@192.168.22.59's password:
Linux haver 5.10.0-18-amd64 #1 SMP Debian 5.10.140-1 (2022-09-02) x86_64

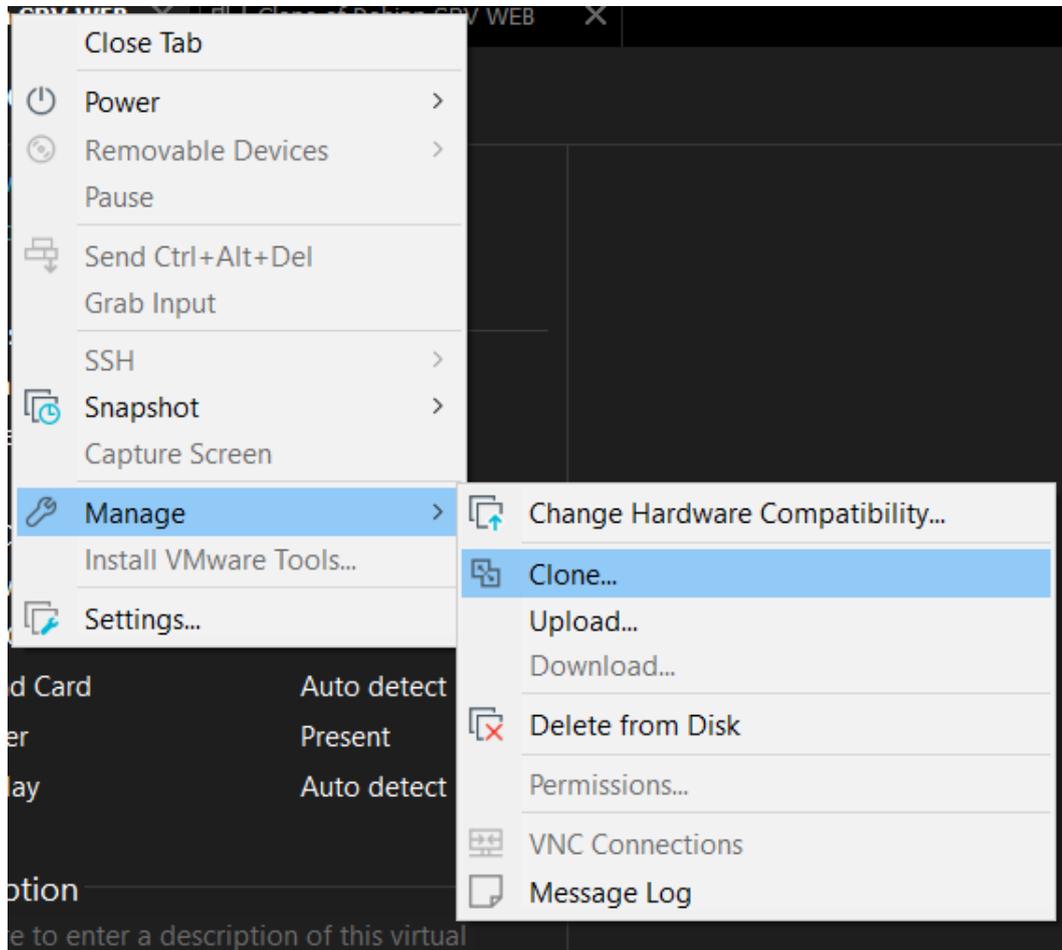
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Sep 22 12:59:41 2022 from 192.168.22.59
```

L'accès au serveur depuis le PC personnel est maintenant fonctionnel.

Etape 4 : Cloner le serveur (La machine virtuelle) pour créer le second serveur

Depuis l'outil de virtualisation de machines (ici VMware Workstation) dupliquer la machine précédemment créé : Manage → Clone



Dans l'écran qui apparaît, sélectionner « The current state in the virtual machine » puis cliquer sur « suivant »

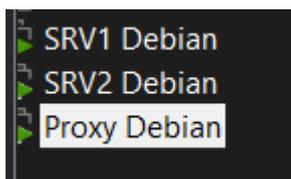
The screenshot shows a dialog box titled "Clone Virtual Machine Wizard" with a close button (X) in the top right corner. The main heading is "Clone Source" with the sub-heading "Which state do you want to create a clone from?". Below this, there are two radio button options. The first option, "The current state in the virtual machine", is selected and includes the text "Creating a linked clone from the current state will create a new snapshot." The second option, "An existing snapshot (powered off only):", is unselected and includes a dropdown menu showing "Snapshot for Clone of Debian SRV WEB" and the text "Created by clone operation." At the bottom of the dialog, there are three buttons: "< Précédent", "Suivant >" (which is highlighted with a blue border), and "Annuler".

Dans l'écran suivant, indiquer un nom pour votre second serveur, puis terminer.

The screenshot shows the same dialog box, now at the "Name of the New Virtual Machine" step. The sub-heading is "What name would you like to use for this virtual machine?". There are two input fields. The first is labeled "Virtual machine name" and contains the text "Clone of Debian SRV WEB". The second is labeled "Location" and contains the path "D:\Virtual Machines\Clone of Debian SRV WEB", with a "Browse..." button to its right. At the bottom, the buttons are "< Précédent", "Terminer" (highlighted with a blue border), and "Annuler".

Réitérer l'opération afin de créer le 3^{ème} serveur (le serveur Reverse Proxy)

Vous avez maintenant vos 3 serveurs.



Attention : Bien penser à nommer correctement chaque machine virtuelle afin de bien distinguer chaque serveur.

Etape 5 : Configuration réseau des 3 serveurs

Depuis les machines virtuelles sur VMware, et aller dans les paramètres de chaque machine. Mettre les 3 serveurs/machines virtuelles sur le réseau « Vmnet 2 ». (Network Adapter)

Virtual Machine Settings



The screenshot shows the 'Virtual Machine Settings' dialog box with the 'Options' tab selected. The 'Hardware' section on the left lists various devices and their configurations. The 'Memory' section on the right shows the current memory allocation and provides a slider to adjust it. The current memory is set to 1024 MB. The slider has three markers: a blue triangle for 'Maximum recommended memory' at 27.9 GB, a green triangle for 'Recommended memory' at 2 GB, and a yellow triangle for 'Guest OS recommended minimum' at 1 GB. The slider scale ranges from 4 MB to 128 GB.

Device	Summary
Memory	1 GB
Processors	1
Hard Disk (SCSI)	18 GB
CD/DVD (IDE)	Using file C:\Users\duval\Do...
Network Adapter	Custom (VMnet2)
USB Controller	Present
Sound Card	Auto detect
Printer	Present
Display	Auto detect

Memory

Specify the amount of memory allocated to this virtual machine. The memory size must be a multiple of 4 MB.

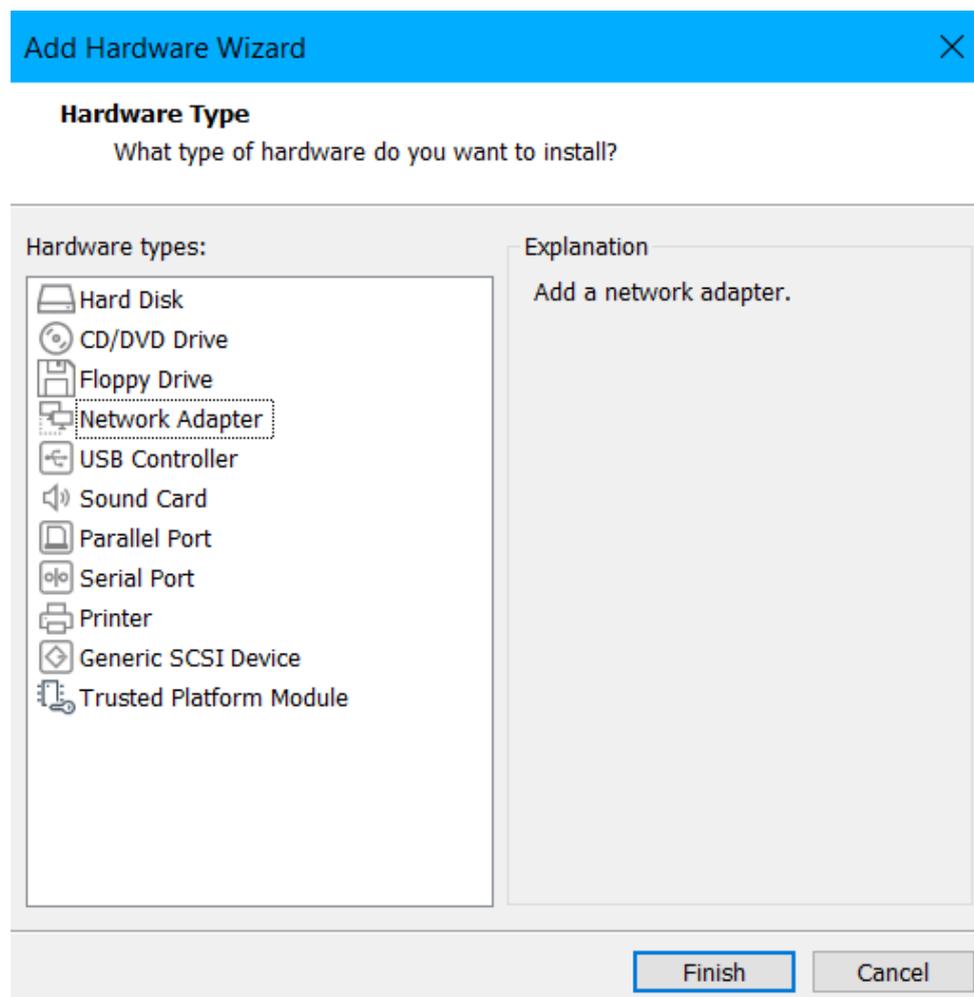
Memory for this virtual machine: 1024 MB

128 GB -
64 GB -
32 GB -
16 GB -
8 GB -
4 GB -
2 GB -
1 GB -
512 MB -
256 MB -
128 MB -
64 MB -
32 MB -
16 MB -
8 MB -
4 MB -

- Maximum recommended memory (Memory swapping may occur beyond this size.) 27.9 GB
- Recommended memory 2 GB
- Guest OS recommended minimum 1 GB

Buttons: Add..., Remove, OK, Cancel, Help

Ajouter ensuite une carte réseau sur le serveur reverse proxy uniquement depuis « Network Adapter »



Hardware Options

Device	Summary
Memory	1 GB
Processors	1
Hard Disk (SCSI)	18 GB
CD/DVD (IDE)	Using file C:\Users\duval\Do...
Network Adapter	Bridged (Automatic)
Network Adapter 2	Custom (VMnet2)
USB Controller	Present
Sound Card	Auto detect
Printer	Present
Display	Auto detect

Memory

Specify the amount of memory allocated to this virtual machine. The memory size must be a multiple of 4 MB.

Memory for this virtual machine: 1024 MB

128 GB -
64 GB -
32 GB -
16 GB -
8 GB -
4 GB -
2 GB -
1 GB -
512 MB -
256 MB -
128 MB -
64 MB -
32 MB -
16 MB -
8 MB -
4 MB -

- Maximum recommended memory (Memory swapping may occur beyond this size.) 27.9 GB
- Recommended memory 2 GB
- Guest OS recommended minimum 1 GB

Add... Remove

OK Cancel Help

Depuis l'interface de chaque serveur, indiquer la commande « ip a » pour vérifier que chaque serveur possède bien sa propre adresse IP.

On remarque que sur le Serveur Reverse Proxy, la seconde carte réseau précédemment ajoutée n'est pas activée. En effet, lors du test de la commande « ip a » on remarque que la carte réseau « ens36 » n'a pas d'ip.

Taper la commande :

```
root@haver:~# nano /etc/network/interfaces_
```

Puis activer la seconde carte réseau, copier les informations de la première carte réseau (ici #the primary network interface) puis coller ces informations en dessous (ici sous #The secondary network interface) en indiquant bien le nom de la seconde carte réseau (ici « ens36 » comme vu précédemment)

```
GNU nano 5.4 interfaces *
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
allow-hotplug ens33
iface ens33 inet dhcp

# The Secondary network interface
allow-hotplug ens36
iface ens36 inet dhcp

# This is an autoconfigured IPv6 interface
iface ens33 inet6 auto

^G Aide          ^O Écrire      ^W Chercher    ^K Couper      ^T Exécuter    ^C Emplacement  M-U Annuler
^X Quitter      ^R Lire fich. ^N Remplacer  ^U Coller      ^J Justifier   ^_ Aller ligne M-E Refaire
```

Etape 6 : Configuration du lien SSH entre les 2 serveurs et synchronisation automatique des 2 serveurs

Modifier ensuite les droits du fichier index.html dans le Serveur 2 Debian en entrant la commande suivante :

```
root@haver:/var/www/html# chmod 666 index.html
```

Ainsi, les serveurs seront capables de communiquer et de modifier le contenu de leurs fichiers respectifs.

Générer ensuite la clé privée et publique pour que les 2 serveurs puissent communiquer en ssh.

Ensuite, appuyer 3 fois sur « entrée » pour skipper tous les mots de passe.

```
root@haver:/var/www/html# ssh-keygen -t rsa
```

Modifier le fichier sshd_config grâce à la commande « nano/etc/ssh/sshd_config »

Relancer le service sshd grâce à la commande :

```
root@haver:~# systemctl restart sshd
```

Ensuite, autoriser le serveur 1 à se connecter avec l'utilisateur root (administrateur) du serveur 2 en indiquant les commandes ci-dessous :

```
Include /etc/ssh/sshd_config.d/*.conf

#Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
PermitRootLogin yes
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes
```

Retourner sur le serveur 1. Aller dans le répertoire /root/.ssh grâce à la commande suivante :

```
root@haver:~# cd /root/.ssh/_
```

Saisir la commande « ssh-copy-id -i id_rsa.pub root@192.168.92.128 » puis entrer le mot de passe du root du serveur 2.

Note : L'adresse ip doit correspondre à l'adresse de votre serveur 2.

Cela permet alors d'envoyer les clés autorisées au serveur 2.

```
root@haver:~/ssh# ssh-copy-id -i id_rsa.pub root@192.168.92.128
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install all the new keys
root@192.168.92.128's password:
Permission denied, please try again.
root@192.168.92.128's password:
Connection closed by 192.168.92.128 port 22
root@haver:~/ssh# ssh-copy-id -i id_rsa.pub root@192.168.92.128
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install all the new keys
root@192.168.92.128's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'root@192.168.92.128'"
and check to make sure that only the key(s) you wanted were added.
```

Vérifier sur le serveur 2 que les clés ont bien été envoyées à l'aide de la commande suivante :

```
root@haver:~/ssh# ls
authorized_keys  known_hosts
```

Dans le serveur 1, afin d'inclure le timer de 5 minutes, entrer la commande suivante qui va permettre d'indiquer la durée voulue :

```
root@haver:~# crontab -e
```

Choisir l'option 1 pour configurer la commande d'automatisation dans le fichier « crontab » qui sera exécuté par cron :

```
no crontab for root - using an empty one

Select an editor. To change later, run 'select-editor'.
 1. /bin/nano      <---- easiest
 2. /usr/bin/vim.tiny

Choose 1-2 [1]: 1
```

Cette option va permettre de configurer le script du timer.

Entrer ensuite la commande suivante dans le fichier crontab :

```
* /5 * * * * scp /var/www/html haver@192.168.92.128:/var/www/html/index.html
```

Note : Les étoiles « * » vont respectivement représenter les durées. Le « / » va permettre d'indiquer que le timer doit se faire toutes les 5 minutes.

Une fois le timer configuré, écrire la commande que vous voulez automatiser à la suite des « * ».

Afin de vérifier que le script du timer est opérationnel, entrer la commande suivante, toujours dans le serveur 1 :

```
root@haver:~# cat /var/log/syslog
```

Dernière étape : Installer et paramétrer Haproxy

Aller sur l'interface du serveur reverse Proxy, pour installer HaProxy grâce à la commande suivante :

```
root@haver:~# cd /etc/haproxy/
```

Modifier le fichier Haproxy avec la commande suivante :

```
root@haver:/etc/haproxy# nano haproxy.cfg
```

Le serveur est maintenant paramétré et fonctionnel.

```

    crt-base /etc/ssl/private

# See: https://ssl-config.mozilla.org/#server=haproxy&server-version=2.0.3&config=intermedi
ssl-default-bind-ciphers ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-EC
ssl-default-bind-ciphersuites TLS_AES_128_GCM_SHA256:TLS_AES_256_GCM_SHA384:TLS_CHACHA20_PO
ssl-default-bind-options ssl-min-ver TLSv1.2 no-tls-tickets

defaults
    log      global
    mode     http
    option   httplog
    option   dontlognull
    timeout connect 5000
    timeout client 50000
    timeout server 50000
    errorfile 400 /etc/haproxy/errors/400.http
    errorfile 403 /etc/haproxy/errors/403.http
    errorfile 408 /etc/haproxy/errors/408.http
    errorfile 500 /etc/haproxy/errors/500.http
    errorfile 502 /etc/haproxy/errors/502.http
    errorfile 503 /etc/haproxy/errors/503.http
    errorfile 504 /etc/haproxy/errors/504.http

frontend gemo
    bind *:80
    default_backend web_servers

backend web_servers
    balance roundrobin
    cookie SERVID prefix
    server server1 192.168.92.129:80 cookie A check
    server server2 192.168.92.128:80 cookie B check

root@haver:/home/haver# systemctl restart haproxy
root@haver:/home/haver#
```