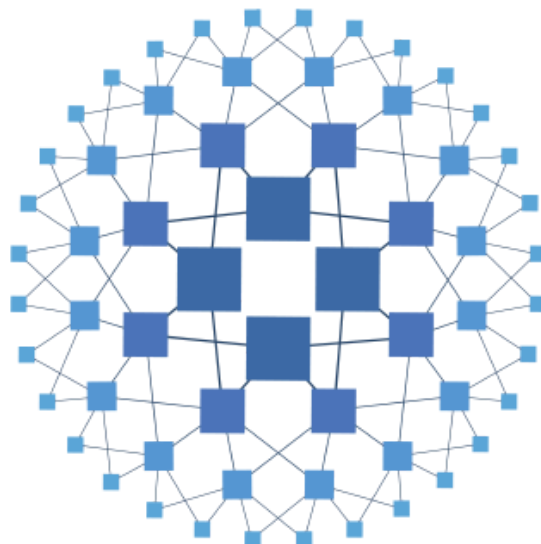




HAProxy et DNS

| | |
|--|----------|
| 1 - Création d'une Machine Virtuelle (VM) avec VirtualBox | 2 |
| Prérequis | 2 |
| Création de la VM | 2 |
| Configuration de la Carte Réseau | 2 |
| 2 - Installation de Debian 13 | 3 |
| Prérequis | 3 |
| Processus d'Installation | 3 |
| Déploiement d'un Serveur DNS (Direct et Inversé) sur Debian | 4 |
| Prérequis DNS | 4 |
| Configuration du DNS Direct | 4 |
| Configuration du DNS Inversé | 5 |
| Configuration HAProxy | 6 |
| Configuration sur la machine « haproxy » | 6 |
| Configuration sur la machine « web1 » | 6 |
| Configuration sur la machine « web2 » | 6 |
| Modification des Sites Apache2 | 7 |
| Configuration d'HAProxy | 7 |
| Test Final | 7 |



HAPROXY





1 - Création d'une Machine Virtuelle (VM) avec VirtualBox

Prérequis

Avant de commencer la création de la VM, assurez-vous d'avoir :

- Installé VirtualBox (téléchargement officiel : [virtualbox.org/wiki/Downloads](https://www.virtualbox.org/wiki/Downloads))
- Téléchargé le fichier **ISO** du système d'exploitation que vous souhaitez installer

Création de la VM

1. Lancez VirtualBox.
2. Cliquez sur le bouton « **Nouvelle** ».

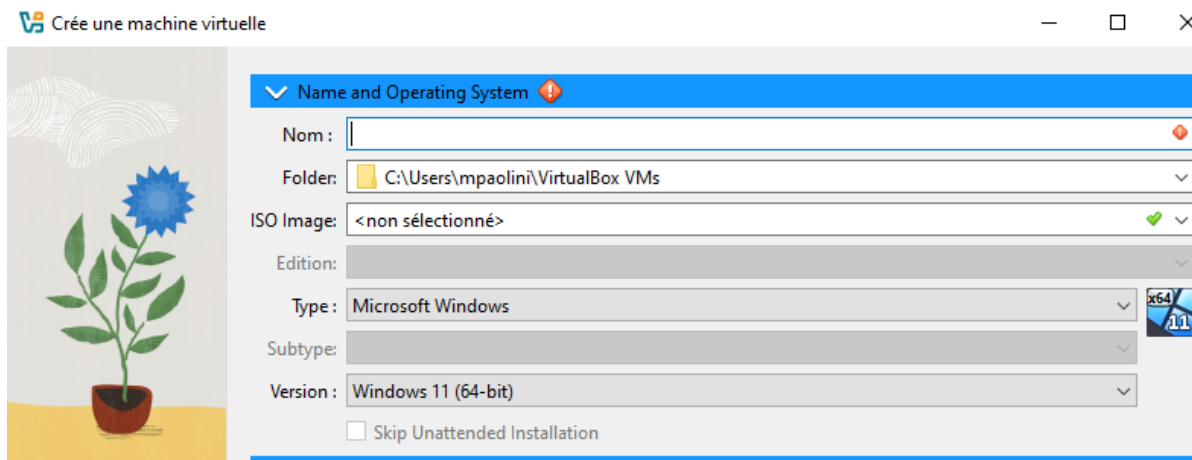


3. Dans la fenêtre de création :
 - **Nommez** la machine virtuelle.
 - Choisissez **l'emplacement de stockage**.
 - Sélectionnez **l'ISO** du système d'exploitation.



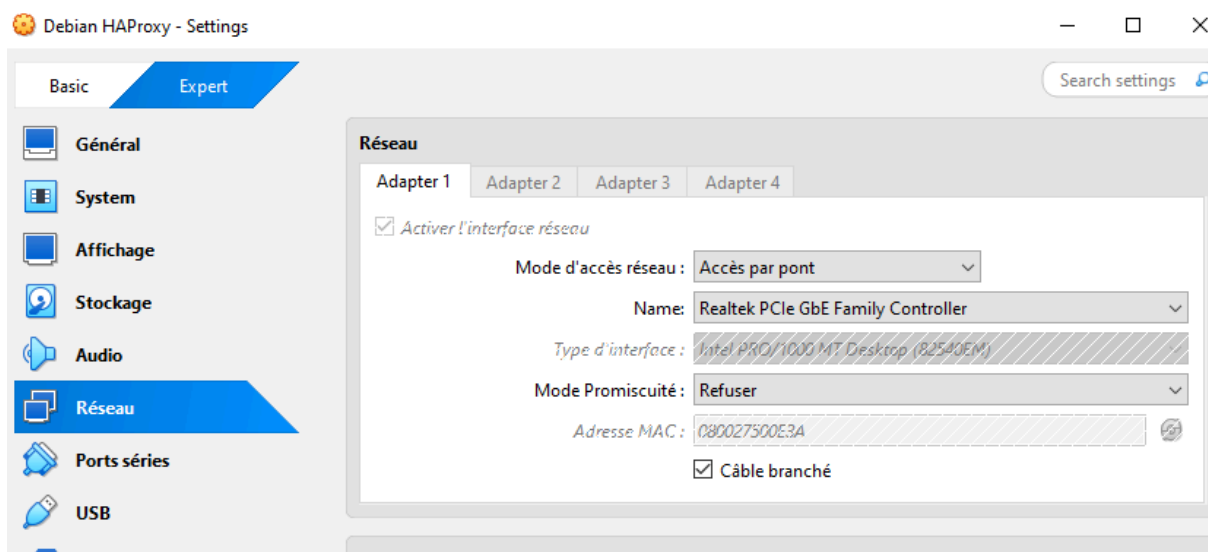


- Cochez « **Skip Unattended Installation** » pour procéder à une installation manuelle classique.



Configuration de la Carte Réseau

1. Sélectionnez votre VM.
2. Cliquez sur « **Configuration** ».
3. Allez dans la section « **Réseau** ».
4. Cochez « **Activer l'interface réseau** ».
5. **Choix du mode d'accès :**
 - **NAT** (par défaut) : À utiliser si vous êtes connecté en **Wi-Fi**.
 - **Accès par pont** : À utiliser si vous êtes connecté en **Ethernet**. Dans ce cas, assurez-vous de sélectionner la **bonne carte réseau**.





Votre machine virtuelle est maintenant prête à être utilisée.

2 - Installation de Debian 13

Prérequis

Pour installer Debian 13, vous devez disposer :

- D'une machine physique ou virtuelle.
- De l'ISO d'installation de Debian 13 (lien de téléchargement : <https://www.debian.org/download.fr.html>).

Processus d'Installation

L'installation de Debian 13 est guidée et consiste à suivre les étapes affichées à l'écran.

Les étapes principales sont :

1. Démarrer la machine depuis l'ISO Debian 13.
2. Choisir « **Install** » ou « **Graphical Install** ».
3. Sélectionner :
 - La **langue**.
 - Le **pays**.
 - La **configuration du clavier**.
4. Configurer le **réseau** (nom de machine, nom de domaine éventuel).
5. Créer un **mot de passe administrateur** (utilisateur **root**).
6. Créer un **utilisateur standard**.
7. Choisir le **partitionnement** (assisté recommandé pour les débutants, ou manuel).
8. Choisir et installer les **logiciels** (environnement de bureau, utilitaires, etc.).
9. Installer le **chargeur d'amorçage GRUB**.
10. Redémarrer la machine.





Votre Debian 13 est maintenant installée et opérationnelle.

Déploiement d'un Serveur DNS (Direct et Inversé) sur Debian

Prérequis DNS

- Avoir installé Debian 12 (note : l'installation précédente concernait Debian 13, mais la procédure DNS utilise Debian 12).
- Configurer la machine avec une **IP fixe**.
- Installer, configurer et démarrer **bind9** :
 - `apt update`
 - `apt install bind9`
 - Vérifier le statut : `systemctl status bind9`
 - Ajouter le nom de votre VM (ex: **DebianDNS1**) dans `/etc/hosts` : `nano /etc/hosts`
 - Dans `/etc/resolv.conf`, ajouter l'adresse du DNS. Le serveur doit s'utiliser lui-même : `nano /etc/resolv.conf`.
- Disposer d'une machine cliente pour effectuer les tests.

Configuration du DNS Direct

1. Modifiez le fichier de configuration des zones (l'exemple utilise **paolini.local**) :
`nano /etc/bind/named.conf.local`

```
GNU nano 8.4
//
// Do any local configuration here
//

zone "paolini.local" IN {
type master;
file "db.paolini.local;
};
```





2. Créez le fichier de zone directe et ajoutez les enregistrements nécessaires :

[nano /var/cache/bind/db.paolini.local](#)

```
GNU nano 2.9.4 /var/cache/bind/db.paolini.local
$TTL 86400
paolini.local. IN SOA debiandns.paolini.local. root.paolini.local {
    1 ; Serial
    604800 ; Expire
    86400 ; Retry
    2419200 ; Expire
    86400 ); Negative Cache TTL
}
paolini.local. IN NS debiandns.paolini.local.
debiandns.paolini.local. IN A 172.17.219.58
web1 IN A 10.10.10.1
web2 IN A 10.10.10.2
haproxy IN A 10.10.10.3
maint IN A 172.17.219.76
```

3. Vérifiez la syntaxe : [named-checkconf -z](#).
4. Redémarrez le service : [systemctl restart bind9](#).
5. Vérifiez le statut : [systemctl status bind9](#).
6. **Test Client (Résolution Directe) :**
 - Paramétrez le DNS sur la carte réseau du client avec l'adresse IP de la VM (trouvée avec [ip a](#)). [Image showing a client network configuration window with 'Utiliser l'adresse de serveur DNS suivante' selected, and 'Serveur DNS préféré' set to 172.17.219.163.]
 - Ouvrez l'invite de commande sur le client et testez : [nslookup poste01.paolini.local](#). [Image showing the result of [nslookup poste01.paolini.local](#) on the client, successfully resolving the name to the address 192.168.1.188.]

Configuration du DNS Inversé

1. Modifiez à nouveau le fichier [named.conf.local](#) pour ajouter la zone inverse :
[nano /etc/bind/named.conf.local](#)
2. Créez le fichier de zone inverse et ajoutez les enregistrements PTR : [nano /var/cache/bind/db.172.17.219](#)





3. Vérifiez la syntaxe : [named-checkconf -z](#).
 4. Redémarrez le service : [systemctl restart bind9](#).
 5. Vérifiez le statut : [systemctl status bind9](#).
 6. **Test Client (Résolution Inverse) :**
 - Vérifiez que le DNS du client est toujours paramétré sur l'IP de la VM.
[Image showing a client network configuration window with 'Serveur DNS préféré' set to 172.17.219.163.]
 - Ouvrez l'invite de commande sur le client et testez avec une adresse IP (l'exemple utilise 8.8.8.8 pour vérifier que le DNS fonctionne correctement) : [nslookup 8.8.8.8](#). [Image showing the result of [nslookup 8.8.8.8](#) on the client, successfully resolving the address 8.8.8.8 to the name dns.google.]
 - La résolution de l'adresse 8.8.8.8 vers [dns.google](#) confirme que la procédure est fonctionnelle.
-

Configuration HAProxy

Prérequis Réseau (3 VMs : haproxy, web1, web2)

Configuration sur la machine « haproxy »

1. Ajoutez une **deuxième carte réseau**. 2. Les interfaces réseau sont nommées [enp0s3](#) et [enp0s8](#).
2. Configurez-les dans VirtualBox :
 5. [enp0s3](#) : **Accès par pont** (réseau public/WAN)
 6. [enp0s8](#) : **Réseau interne** (réseau privé/LAN)
3. Configurez les IP fixes dans [/etc/network/interfaces](#) :
 5. **enp0s3 (Public)** : [address 172.17.219.76](#), [netmask 255.255.0.0](#),
[gateway 172.17.0.1](#)
 6. **enp0s8 (Interne)** : [address 10.0.0.3](#), [netmask 255.255.255.0](#)





7. Activez les interfaces et vérifiez : `ifup enp0s3` et `enp0s8`, puis `ip a`.

Configuration sur la machine « web1 »

1. La carte réseau doit être configurée en **Réseau interne**.
2. Installez `apache2` : `apt install apache2 -y`.
3. Vérifiez le statut : `systemctl status apache2`. [Image showing the status of the apache2 service as 'active (running)'.]
4. Configurez l'IP fixe dans `etc/network/interfaces` :
 - `enp0s3` : `address 10.10.10.1, netmask 255.255.255.0`

Configuration sur la machine « web2 »

1. Installez `apache2` : `apt install apache2 -y`.
2. Configurez l'IP fixe (la machine web2 doit aussi être en Réseau interne) :
 - `enp0s3` : `address 10.10.10.2, netmask 255.255.255.0`

La configuration réseau est terminée.

Modification des Sites Apache2

Sur les machines « **web1** » et « **web2** », modifiez les pages `index.html` par défaut pour faciliter les tests et identifier le serveur utilisé.

1. **Sur `web1`** :
 - `nano /var/www/html/index.html`
 - Utilisez ce contenu HTML (Titre : `Web 1`, Contenu : `Bienvenue sur web 1`).
2. **Sur `web2`** :
 - `nano /var/www/html/index.html`
 - Utilisez ce contenu HTML (Titre : `Web 2`, Contenu : `<h1>Bienvenue sur web 2</h1>`

Configuration d'HAProxy

1. Installez HAProxy sur la machine `haproxy` : `apt install haproxy`.





2. Éditez le fichier de configuration : `nano /etc/haproxy/haproxy.cfg`.
3. Ajoutez les sections **frontend** et **backend** pour le load balancing (remplacez l'IP publique si nécessaire) :
 4. **frontend proxypublic** : Écoute sur l'IP publique (**172.17.219.76**) sur le port 80.
 5. **backend fermeweb** : Utilise l'algorithme d'équilibrage **roundrobin**.
 6. Les serveurs sont **web1** (**10.10.10.1:80**) et **web2** (**10.10.10.2:80**).
 7. Active les statistiques sur **/statsHaproxy** avec l'authentification **admin:admin**.
 8. Redémarrez le service : `systemctl restart haproxy`.

Test Final

1. Ouvrez un navigateur et accédez à l'adresse IP publique de la machine HAProxy (par exemple, <http://172.17.219.76/>).
2. En rafraîchissant la page, vous devriez voir le contenu des pages **web 1** et **web 2** alterner, confirmant que l'équilibrage de charge fonctionne.
3. Vous pouvez consulter les statistiques de HAProxy en accédant à : <http://172.17.219.76/statsHaproxy>.

