



# DDWS

JOB 1

JOB 2

JOB 3

JOB 4

JOB 5

JOB 6

JOB 7

JOB 8

# JOB 1 / INSTALLATION DE LA VM



Tout d'abord, j'ai opté pour la plateforme de VM de base. J'ai initialement débuté le projet avec VMware, mais j'ai rencontré des problèmes pour mettre en place l'SSH et le bridge. Par conséquent, je me suis tourné vers VirtualBox. Ensuite, j'ai choisi une image ISO complète de Debian pour minimiser les installations supplémentaires. J'ai rencontré quelques problèmes avec VirtualBox, notamment des étapes d'installation omises, mais comme souvent depuis le début de l'année, il s'agit généralement d'une petite case à cocher cachée parmi les vingt étapes, susceptible de bloquer le processus.

The image shows the Debian 12 logo, which features a stylized white '@' symbol (the Debian logo) to the left of the text 'debian 12' in a white, sans-serif font. The background is a dark teal color with a geometric, low-poly pattern.

# JOB 2 / installer un serveur Web Apache2

Tout d'abord, veuillez vous assurer que tous vos paquets sont à jour à l'aide de la commande suivante :

bash

Copy code

```
$ sudo apt update && sudo apt upgrade
```

Ensuite, installez le paquet Apache2 en utilisant la commande suivante :

bash

Copy code

```
$ sudo apt install apache2
```

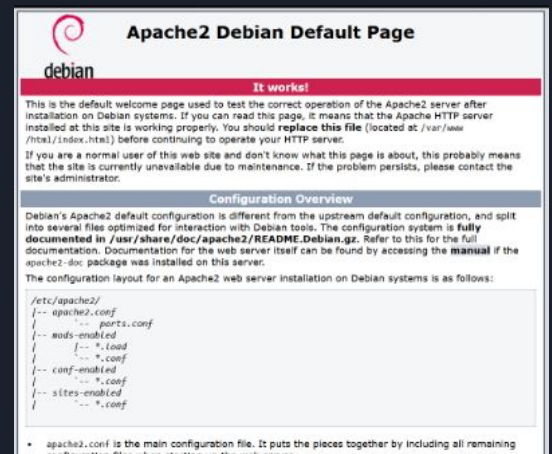
Après pour l'activer et même après redémarrage :

bash

Copy code

```
$ sudo systemctl start apache2  
$ sudo systemctl enable apache2
```

Pour vérifier si cela fonctionne, entrez simplement l'adresse IP dans un navigateur. Si la page s'affiche, félicitations, c'est terminé.



# JOB 3 / Different serveur web

Serveur Web	Avantages	Inconvénients
Apache	Open source et gratuit, grande communauté de support.	Peut être moins performant, configuration complexe.
Nginx	Très performant et évolutif, excellente gestion des connexions concurrentes.	Configuration complexe pour les débutants.
NetWare	Historiquement utilisé pour les réseaux Novell, sécurité renforcée.	En déclin, peu utilisé actuellement, moins de support et mises à jour.
IIS (Microsoft)	Intégration aisée avec Microsoft, bonne prise en charge des technologies MS.	Principalement pour environnements Windows, moins flexible pour non-Microsoft.
Domino (IBM)	Large éventail de services, intégration avec produits IBM.	Principalement pour environnements IBM, complexité accrue par rapport à d'autres.
Google Web Server	Hautes performances, conçu pour les besoins spécifiques de Google.	Non disponible en dehors de Google, non open source.

# JOB 4.1 / Mettez en place un DNS

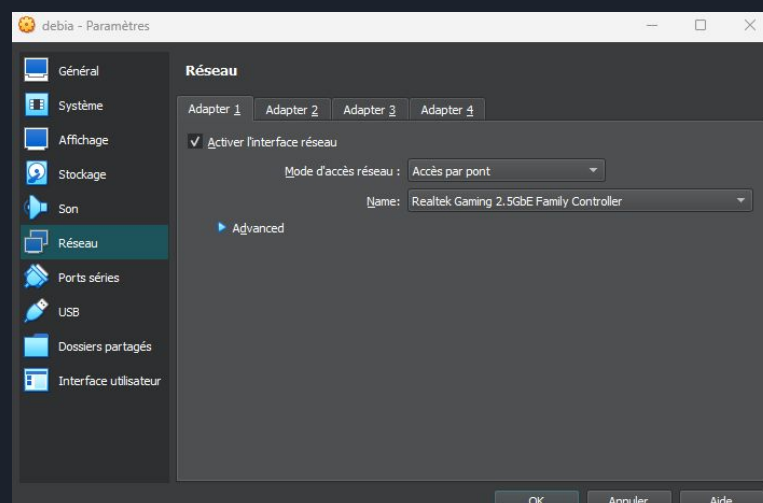
Dans un premier temps, j'ai procédé à l'installation des paquets bind9, bind9utils, dnsutils en utilisant ufw, ainsi que Samba à l'avance pour les tâches n°7 et n°8. Cette préparation était nécessaire, car en mode bridged, le téléchargement de ces paquets aurait pu poser problème.

bash

Copy code

```
$ sudo apt install bind9 bind9utils dnsutils ufw samba
```

Dans Virtual Box, j'ai configuré le réseau en mode "bridged,"



j'ai utilisé la commande "hostname -I" pour trouver mon adresse IP. Ensuite, je me suis dirigé vers le répertoire /etc/bind en utilisant la commande "cd" pour accéder aux fichiers de configuration de Bind. J'ai copié le contenu de la zone de base de données (db.local) dans un fichier nommé 'direct'. Cette étape configure le serveur DNS en établissant les enregistrements nécessaires pour la résolution des noms de domaine

## JOB 4.2 / Mettez en place un DNS

J'ai employé l'éditeur de texte 'nano' pour apporter des modifications au fichier 'direct', y incluant mon adresse IP et mon domaine. Cette action permet de personnaliser la configuration DNS en liant mon adresse IP à mon domaine, simplifiant ainsi la résolution de ce dernier vers l'adresse IP correspondante.

```
#
; BIND data file for local loopback interface
;
$TTL      604800
@         IN      SOA      prepa.com dnsproject.prepa.com. (
                           2           ; Serial
                           604800      ; Refresh
                           86400       ; Retry
                           2419200     ; Expire
                           604800 )   ; Negative Cache TTL
;
@         IN      NS       dnsproject.prepa.com.
dnsproject IN      A        10.10.7.177
www       IN      CNAME    dnsproject.prepa.com.
```

J'ai employé l'éditeur de texte 'nano' pour apporter des modifications au fichier 'direct' en y incluant mon adresse IP et mon domaine. Cette action personnalise la configuration DNS en reliant mon adresse IP à mon domaine, simplifiant ainsi la résolution de ce domaine vers l'adresse IP correspondante. Après avoir modifié le fichier "direct", j'ai copié son contenu dans un fichier nommé "inverse". La seule modification que j'ai effectuée dans ce fichier était à la dernière ligne, où j'ai ajusté son contenu pour qu'il ressemble à ceci : "213 IN PTR dnsproject.prepa.com." Cette adaptation permet d'associer l'adresse IP 213 à mon domaine "dnsproject.prepa.com" pour une résolution inversée.

# JOB 4.3 / Mettez en place un DNS

```
; BIND data file for local loopback interface
;
$TTL      604800
@         IN      SOA     prepa.com dnsproject.prepa.com. (
                        2      ; Serial
                        604800 ; Refresh
                        86400  ; Retry
                        2419200 ; Expire
                        604800 ) ; Negative Cache TTL
;
@         IN      NS      dnsproject.prepa.com.
dnsproject IN      A       10.10.7.177
213       IN      PTR     dnsproject.prepa.com.
```

Après avoir enregistré ces modifications dans le fichier "inverse", je suis passé au fichier "named.conf.local". J'y ai effectué des ajustements pour intégrer mon adresse IP, son équivalent inversé (reverse IP), ainsi que mon domaine. Ces actions permettent de configurer le serveur DNS afin qu'il identifie et réponde aux requêtes associées à mon domaine. De plus, j'ai pris en considération le fichier "resolv.conf" pour paramétrer les résolutions DNS sur mon appareil en utilisant mon adresse IP et mon nom de domaine "prepa.com". Cela garantit que mon appareil utilise correctement le serveur DNS que j'ai configuré pour résoudre les noms de domaine.

Après avoir effectué toutes ces étapes, j'ai redémarré le service Bind9 et utilisé la commande "ping" avec mon domaine "dnsproject.prepa.com" pour vérifier que ma configuration DNS fonctionnait correctement.



# JOB 5 / Nom de domaine

Comment peut-on acquérir un nom de domaine public ?

Pour obtenir un nom de domaine public, il suffit de sélectionner un bureau d'enregistrement de noms de domaine, vérifier la disponibilité du nom souhaité, choisir l'extension appropriée, l'ajouter au panier, effectuer le paiement en fournissant vos informations de contact, configurer les serveurs de noms DNS, et assurer le renouvellement régulier du nom de domaine pour le maintenir actif. Ces étapes simples permettent d'acquérir un nom de domaine pour votre site web ou projet en ligne.

Quelles particularités peut-on trouver avec certaines extensions de nom de domaine ?

Certaines extensions de nom de domaine possèdent des spécificités particulières. Par exemple, les extensions de pays (ccTLDs) sont associées à des régions et peuvent comporter des restrictions géographiques. Les extensions thématiques (nTLDs) ciblent des domaines spécifiques, comme .app pour les applications. Les extensions premium peuvent être plus onéreuses en raison de leur attrait particulier. Comprendre ces particularités est essentiel pour choisir l'extension la mieux adaptée en fonction de vos besoins.



# JOB 6 / Connectez votre hôte au nom de domaine

Tout en suivant attentivement les étapes que j'ai mentionnées pour le Job n°4, il est désormais nécessaire de mettre à jour les paramètres DNS de votre machine hôte. (10.10.0.1 représente le DNS de La Plateforme)

**Modifier les paramètres DNS du réseau**

Manuel

**IPv4**

☒ Activé

DNS préféré

10.10.0.1

DNS sur HTTPS

Désactivé

**Autre DNS**

10.10.7.177

DNS sur HTTPS

Désactivé

**IPv6**

☐ Désactivé

Enregistrer Annuler

# JOB 7 / pare-feu en utilisant ufw

Suite à la mise en place du pare-feu UFW lors du Job 4, j'ai initié la configuration des politiques par défaut pour le pare-feu. J'ai défini la politique de refus entrant en utilisant la commande `'ufw default deny incoming'`. Cette politique indique que par défaut, toutes les connexions entrantes sont rejetées, sauf celles pour lesquelles j'ai spécifiquement établi des règles autorisant le trafic entrant. Ensuite, j'ai configuré la politique de refus sortant en utilisant `'ufw default deny outgoing'`. Cela signifie que par défaut, toutes les connexions sortantes sont également rejetées, à moins que je ne définisse des règles spécifiques permettant le trafic sortant.

Afin de permettre le trafic entrant sur certains ports, j'ai mis en place des règles spécifiques. Par exemple, j'ai utilisé la commande `'ufw allow 80/tcp'` pour autoriser le trafic TCP sur le port 80. Cette action est généralement appliquée pour permettre le trafic HTTP entrant, permettant ainsi aux utilisateurs d'accéder à des sites web.

De plus, j'ai permis le trafic sur les ports 139 et 445 à l'aide des commandes `'ufw allow 139/tcp'` et `'ufw allow 445/tcp'`. Ces ports sont associés au protocole SMB (Server Message Block) et sont utilisés pour le partage de fichiers et d'imprimantes. En autorisant le trafic sur ces ports, j'ai préparé notre système pour l'éventuelle mise en place de services de partage de fichiers SAMBA.

J'ai édité le fichier `before.rules` situé dans `/etc/ufw/` en utilisant la commande `'nano /etc/ufw/before.rules'`. À la ligne `"# ok icmp codes for INPUT"`, j'ai remplacé toutes les occurrences de `"ACCEPT"` par `"DROP"`.

Cette modification a pour effet de bloquer par défaut les paquets ICMP entrants, renforçant la sécurité en limitant l'accès aux informations via ces paquets. Cela contribue à réduire le risque d'attaques basées sur les paquets ICMP, améliorant ainsi la sécurité du réseau.

Enfin, j'ai activé le pare-feu UFW en utilisant la commande `"ufw enable"` pour finaliser la configuration.

# JOB 8 / dossier partagé sur le réseaux

Passons maintenant à la configuration de Samba. Pour commencer, j'ai établi un dossier nommé "Partage" destiné à être un répertoire partagé entre le serveur et ma machine hôte ou d'autres machines virtuelles. J'ai utilisé la commande `"sudo mkdir /home/Partage"` pour le créer.

Ensuite, j'ai modifié la configuration de Samba en accédant au fichier `smb.conf` situé dans `/etc/samba`. J'ai utilisé la commande `"nano /etc/samba/smb.conf"` pour y accéder et y apporter les modifications nécessaires. J'ai simplement ajouté le contenu présent dans mon image à la fin du fichier.

Cela permet de configurer le dossier partagé de manière à ce qu'il soit accessible depuis d'autres machines, facilitant ainsi le partage de fichiers entre le serveur, ma machine hôte ou d'autres machines virtuelles.

Ensuite, j'ai créé un utilisateur et défini son mot de passe pour SMB afin d'accéder au dossier partagé. J'ai utilisé la commande `"smbpasswd -a [nom d'utilisateur]"` et saisi le mot de passe correspondant. Ensuite, j'ai accordé les droits appropriés pour interagir avec le fichier en utilisant `'chmod -R 777 /home/Partage'`. Cela garantit que l'utilisateur dispose des autorisations nécessaires pour accéder et gérer les fichiers dans le dossier "Partage".

Après avoir redémarré le service `smbd` à l'aide de la commande `'service smbd restart'`, le partage de fichiers devrait fonctionner conformément à ce qui a été configuré. Ce redémarrage permet d'appliquer les modifications que nous avons apportées à la configuration de Samba.

Une fois ces étapes effectuées, l'utilisateur que nous avons créé précédemment devrait pouvoir accéder au dossier Partage avec les autorisations appropriées, favorisant un partage de fichiers fluide et sécurisé.