Job 01

Installation VM

I. Installation

Pour établir un serveur Apache2, la première étape consiste à déployer une machine virtuelle sous Debian 12 avec une interface graphique. Dans ce guide, j'utiliserai VMWare Workstation Pro pour cette configuration.

II. Accès SSH

Il est possible de configurer notre serveur web directement depuis la machine virtuelle ou d'établir une connexion SSH depuis notre machine hôte. J’ai choisi la configuration directe depuis la machine virtuelle.

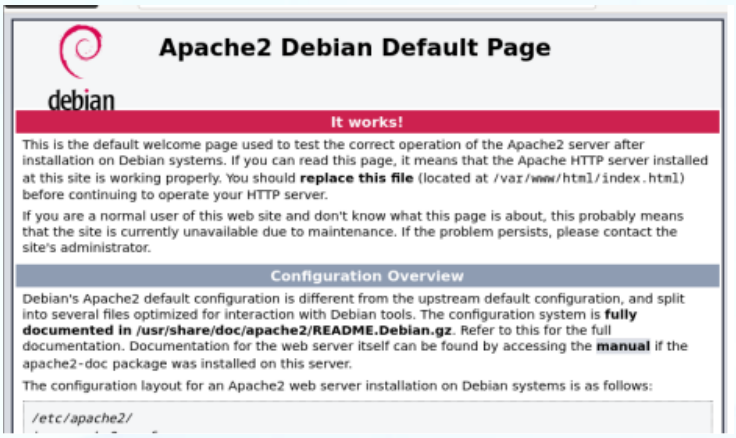
Pour commencer, nous devons vérifier que le service SSH est opérationnel en utilisant la commande `systemctl status ssh`. Une fois cette vérification effectuée, sur notre machine hôte, nous pouvons exécuter la commande suivante : `ssh 192.168.44.129`, en remplaçant l'adresse IP par celle de votre propre machine virtuelle. Ensuite, saisissez le mot de passe lorsque vous y êtes invité, et la connexion est établie.

Job 02

Installation du serveur Apache

I. Apache2

Pour installer Apache2, nous utilisons la commande `apt install apache2`. Une fois l'installation terminée, aucune configuration supplémentaire n'est requise, car les paramètres par défaut sont suffisants pour cet exercice. Pour visualiser la page web, il vous suffit d'entrer l'adresse IP de votre machine virtuelle dans le navigateur.



L'accès à la page web est possible aussi bien depuis notre machine virtuelle que depuis notre machine hôte.

Job 03

Serveurs web

Il existe plusieurs serveurs web parmi lesquels les cinq les plus couramment utilisés sont les suivants :

1. Apache HTTP Server (Apache):

- Description : Apache est l'un des serveurs web les plus anciens et largement adoptés. Il s'agit d'un logiciel open source avec une extensibilité grâce à l'utilisation de modules.
- Avantages : Stabilité éprouvée, documentation exhaustive, polyvalence, module de sécurité robuste.
- Inconvénients: Complexité potentielle pour les débutants, nécessité de redémarrages fréquents lors de modifications de la configuration.

2. Nginx:

- Description: Nginx se distingue par ses performances élevées et sa faible consommation de ressources. Souvent utilisé comme serveur proxy ou pour la répartition de charge.
- Avantages: Efficacité élevée, gestion efficace de nombreuses connexions simultanées, équilibrage de charge performant.
- Inconvénients: Courbe d'apprentissage abrupte pour les débutants, absence de support natif pour certains langages serveur, nécessité d'utiliser des proxies pour pallier cette limitation.

3. Microsoft Internet Information Services (IIS):

- Description: IIS est le serveur web de Microsoft, principalement utilisé sur les serveurs Windows.
- Avantages: Intégration native avec les systèmes Windows Server, prise en charge d'ASP.NET, facilité d'administration via une interface graphique conviviale.
- Inconvénients: Exclusivité aux environnements Windows, moins répandu en dehors de cet écosystème.

4. LiteSpeed:

- Description: Connu pour sa rapidité et sa fiabilité en matière de sécurité, LiteSpeed est compatible avec Apache et offre une transition aisée.
- Avantages: Performance exceptionnelle, transition facile depuis Apache, fonctionnalités de sécurité robustes.
- Inconvénient: Non open source, version gratuite avec fonctionnalités limitées.

5. Caddy:

- Description: Caddy se distingue par sa facilité d'installation et de configuration, avec un support natif du chiffrement HTTPS et une interface de gestion web.
- Avantages: Installation et configuration simples, support natif du chiffrement HTTPS, interface de gestion web.
- Inconvénients: Moins flexible que certains autres serveurs, performances légèrement inférieures dans certaines situations par rapport à des serveurs tels que Nginx.

Job 04

DNS (Système de Noms de Domaine)

I. Installation et configuration de Bind9

Pour mettre en place un serveur DNS, Bind9 peut être utilisé. Son installation s'effectue via la commande `apt install bind9`.

Une fois l'installation terminée, la configuration nécessite des ajustements dans différents fichiers afin de créer notre propre domaine.

Pour commencer, la configuration du fichier de zone directe est cruciale. Nous procédons en copiant le fichier `db.local` vers `direct` avec la commande `sudo cp /etc/bind/db.local /etc/bind/direct`.

Une zone directe, également appelée zone "forward" dans un serveur DNS, gère la résolution des noms de domaine vers des adresses IP. Plus concrètement, elle établit des correspondances entre des noms de domaine spécifiques et leurs adresses IP respectives. Dans notre contexte, elle associera le nom `dnsproject.prepa.com` à l'adresse IP de notre machine. À noter que les zones inverses effectuent l'opération inverse.

La prochaine étape implique la modification du fichier de zone directe à l'aide de la commande `sudo nano /etc/bind/direct`.

```
GNU nano 7.2                        /etc/bind/direct

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.
nsproject      IN      A       192.168.0.15
ww     IN      CNAME   dnsproject.prepa.com.
```

Après avoir configuré le fichier "direct", nous procéderons à sa copie dans le fichier "inverse" en utilisant la commande `sudo cp /etc/bind/direct /etc/bind/inverse`.

```
GNU nano 7.2                        /etc/bind/inverse

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      dsnproject.prepa.com.
nsproject      IN      A       192.168.0.15
0      IN      PTR     dnsproject.prepa.com.
```

Ensuite, il nous faut éditer le fichier "named.conf.local" situé dans `/etc/bind/`.

```
  GNU nano 7.2                  /etc/bind/named.conf.local
//
// Do any local configuration here
//

// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";
zone "prepa.com" IN {
        type master;
        file"/etc/bind/direct";
};
zone "9.10.10.in-addr-arpa" IN {
        type master;
        file"/etc/bind/inverse";
};
```

Une fois toutes ces étapes réalisées, nous devons configurer le fichier "resolv.conf" situé dans le dossier `/etc/`.

```
  GNU nano 7.2
# Generated by NetworkManager
search prepa.com
nameserver 192.168.0.15
```

Une fois réalisé ces étapes nous pouvons redémarrer le service bind9.

II. Test

Nous allons à présent tester la fonctionnalité de notre configuration sur notre machine virtuelle en effectuant un ping et en affichant notre site sur Firefox à partir de l'adresse `dnsproject.prepa.com`.

Job 05

Domaine public

Communément, l'acquisition d'un nom de domaine passe par le choix d'un prestataire de services de noms de domaine, tels que GoDaddy, Namecheap, ou Google Domains.

Ensuite, il est impératif de scruter la disponibilité du nom de domaine sur le site du fournisseur, car deux entités distinctes ne peuvent arborer le même nom de domaine.

Le choix de l'extension de domaine (Top-Level Domain), telle que .com, .net, .org, .fr, .io, etc., doit également être tranché. Chaque extension de domaine présente ses caractéristiques et restrictions uniques (voir fin du Job 05). La marche à suivre consiste ensuite à suivre attentivement les instructions du site du prestataire.

Une fois le nom de domaine enregistré, la configuration des enregistrements DNS s'impose. Ces enregistrements dirigent vers les serveurs de votre site web ou de votre service de messagerie.

Enfin, des frais d'enregistrement annuels doivent être acquittés pour préserver la propriété du nom de domaine.

Il existe divers types d'extensions de domaine (TLD) :

1. gTLD (Generic Top-Level Domain): Ces extensions, telles que .com, .org, .net, sont génériques et largement utilisées. Elles sont ouvertes à tous, sans restrictions particulières.

2. ccTLD (Country Code Top-Level Domain) : Ces extensions sont liées à des pays ou territoires, comme .fr (France), .uk (Royaume-Uni), .de (Allemagne), etc. Les ccTLD sont souvent soumis à des restrictions géographiques, exigeant parfois une présence ou une adresse dans le pays pour l'enregistrement.

3. TLD de Second Niveau Restreints : Certains TLD de second niveau, tels que .gov, .edu, sont réservés à des entités spécifiques comme les organismes gouvernementaux ou les établissements d'enseignement.

4. TLD de Second Niveau Génériques : Certains pays et registres offrent des TLD de second niveau génériques, accessibles à tous sans restriction particulière.

5. Nouvelles Extensions (gTLD) : Ces extensions, comme .app, .io, .blog, sont plus récentes et spécifiques. Elles peuvent parfois offrir des opportunités intéressantes pour des noms de domaine pertinents.

Job 06

DNS machine hôte

Pour effectuer ce job, il est recommandé de configurer l'adaptateur réseau de notre machine virtuelle en mode bridge ou host-only. En optant pour host-only, seules notre hôte et notre machine virtuelle pourront communiquer. Dans mon cas, je vais redémarrer la machine virtuelle en mode host-only.

I. Modifier le DNS de notre connexion Wi-Fi sur notre machine hôte

Afin d'utiliser le serveur DNS de notre machine hôte (ici Windows 11), nous devons accéder aux paramètres réseau, cliquer sur la configuration du Wi-Fi, puis dans la section DNS, choisir la configuration manuelle et entrer l'adresse IP de notre serveur DNS (notre machine virtuelle).

II. Test

À présent, nous pouvons tenter d'accéder à notre serveur web depuis le navigateur de notre machine hôte.

Job 07

Pare-feu ufw (firewall)

La désactivation des pings (ICMP) sur un serveur ou un pare-feu offre des avantages significatifs en matière de sécurité, tels que la réduction de la visibilité pour les attaquants et la minimisation de la surface d'attaque. Cette mesure de sécurité peut être particulièrement efficace contre les attaques DDoS, tout en améliorant potentiellement les performances. Cependant, il est essentiel de noter que cela peut rendre le dépannage plus complexe, influencer certains services réseau et ne garantit pas une protection totale.

I. Installation et configuration de ufw (uncomplicated firewall)

Commencez par l'installation de ufw à l'aide de la commande `sudo apt install ufw`. Une fois l'installation terminée, passons à la configuration.

Éditez le fichier `before.rules`, généralement situé dans `/etc/ufw`, en utilisant la commande `sudo nano /etc/ufw/before.rules`. Dans la section `# ok icmp codes for INPUT`, remplacez `ACCEPT` par `DROP`.

Ensuite, autorisez le trafic sur le port 80 pour le protocole HTTP et sur le port 443 pour le protocole HTTPS en utilisant respectivement les commandes `sudo ufw allow 80/tcp` et `sudo ufw allow 443/tcp`. Vérifiez les règles appliquées avec `sudo ufw status`.

Pour activer le pare-feu, exécutez la commande `sudo ufw enable`.

II. Test

Observez que les pings ne sont plus fonctionnels. Tous les paquets ICMP envoyés sont désormais rejetés. Toutefois, malgré cette restriction, la page web reste accessible.

Job 08

SMB (Server Message Block)

Pour établir un dossier partagé sur notre serveur, nous allons utiliser le protocole SMB (Server Message Block) afin de créer un partage réseau. Cela permettra aux autres membres de votre réseau d'accéder aux fichiers et de partager des données dans ce dossier.

I. Installation et configuration de Samba

Commencez par installer Samba en utilisant la commande `sudo apt install samba`.

Ensuite, créons un dossier qui servira de dossier partagé. Dans notre cas, j'ai créé ce dossier dans le répertoire home avec la commande `sudo mkdir /home/dossier-partage`.

Ensuite, attribuons tous les droits à tous les utilisateurs, bien que cela ne soit pas une pratique recommandée pour des raisons de sécurité. Utilisons la commande `sudo chmod -R 777 /home/dossier-partage`.

Passons à l'édition de la configuration de Samba. Utilisez la commande `sudo nano /etc/samba/smb.conf`.

```
[Partage]
   path = /home/dossier-partage
   available = yes
   valid users = dvdp
   read only = no
   browsable = yes
   public = yes
   writable = yes
```

Il ne reste plus qu'à configurer un mot de passe pour l'utilisateur avec sudo smbpasswd -a dvdp.

Ensuite, redémarrer le service avec systemctl restart samba.

II. Configuration ufw

Notre nouveau pare-feu empêche la connexion via le protocole SMB. Pour autoriser la connexion, ouvrons le port 139 et le port 445 avec les commandes sudo ufw allow 139/tcp et sudo ufw allow 445/tcp.

III. Test

Nous pouvons maintenant tester depuis le navigateur de fichiers de notre machine hôte.

Utilisons \10.10.11.178\Partage dans l'explorateur de fichiers.

Il nous sera alors demandé de nous connecter. Une fois fait, les fichiers se partagent correctement.