Documentation DDWS

JOB 1

Installer une VM :

* ouvrir VMware
* cliquer sur ouvrir une nouvelle VM
* lui allouer une quantitée spécifique de ressources
* nommer la machine virtuelle
* sélectionner la version du hardware à installer
* passer par l’ ISO pour installer debian
* se mettre en NAT
* attention à bien sélectionner GRUB pendant la config
* finir l’installation

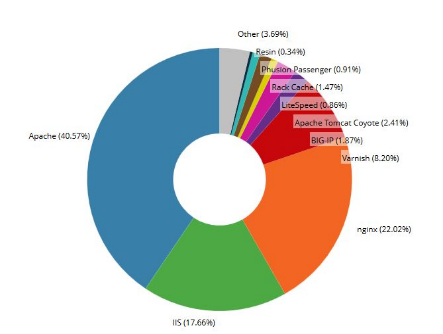
JOB 2

Pour installer Apache2 :

Sur le terminal

* faire un “sudo apt update”
* puis un “sudo apt install apache2”
* maintenant que apache2 est installé on règle les pare-feux “sudo ufw allow ‘Apache’”
* depuis le terminal on peut vérifier si le serveur web est bien fonctionnelle “sudo systemctl status apache2
* ou alors en tapant la commande “hostname -I” on obtient l’ip puis on la rentre sur notre barre de recherche soit  “http://ip\_du\_serveur”

JOB 3

J’ai commencé à me renseigner sur les différents serveurs web existant en commençant par chercher les plus communs et je suis donc tombé sur cette image : source : <https://www.projets-informatiques.com/>

On remarque que les 3 principaux sont, dans l’ordre décroissant Apache, Nginix et IIS, je vais donc parler de ceux-là et essayer de comprendre pourquoi ils sont le plus utilisés.

Apache : C’est un logiciel qui s’exécute sur un serveur permettant une connexion entre serveurs et navigateurs. Apache possède une architecture modulaire cela permet d'implémenter facilement une nouvelle fonctionnalité au système

**Avantages :**

* Open-source et gratuit même pour un usage commercial.
* Logiciel fiable et stable.
* Mise à jour régulière, correctifs de sécurité réguliers.
* Flexible grâce à sa structure basée sur des modules.
* Facile à configurer, adapté aux débutants.
* Plateforme-Cross (fonctionne sur les serveurs Unix et Windows).
* Fonctionne avec les sites WordPress.
* Grande communauté et support disponible en cas de problème.

**Inconvénients :**

* Problèmes de performances sur les sites web avec un énorme trafic.
* Trop d’options de configuration peuvent mener à la vulnérabilité de la sécurité.

sources : https://www.hostinger.fr/tutoriels/quest-ce-quapache-serveur-web-apache#Les\_avantages\_et\_inconvenients\_dApache

Nginx : est un serveur proxy inverse ce qui aide à équilibrer la charge de travail en répartissant les requêtes et en mettant en cache certain type de contenu. Comme Apache, Nginx à un architecture modulaire.

**Avantages :**

* Nginx évolue rapidement et de manière optimisée.
* Nginx ne subit aucuns problèmes de ralentissement due à un potentiel manque de RAM.
* Aucun refus des requêtes car une saturation lui est impossible.
* Nginx est conçu sur mesure pour accélérer le fonctionnement des serveurs privés virtuels (VPS).

**Inconvénients :**

* Ne prends pas en charge .htaccess
* Ne disposant pas de nombreux modules pour l'étendre, il y a donc moins de compatibilité avec les technologies tierces.
* un nombre de soutien et d’aide relativement réduit

sources : https://www.opportunites-digitales.com/guide-complet-du-serveur-nginx/

IIS : ou <<Internet Information Services>> est un progiciel conçu pour Windows Server.

**Avantages :**

* possède une excellente sécurité
* fournis avec Windows Server
* entièrement compatible avec Windows
* très facile d’utilisation

**Inconvénients :**

* payant
* des performances visiblement un peu en dessous de ces concurrents
* seulement monoplateforme

sources : https://fr.differbetween.com/article/pros\_and\_cons\_of\_iis

JOB 4

Il faut utiliser la commande “nano /etc/hosts” afin d’ajouter la ligne de commande avec l’ip et le nom de domaine de notre serveur cad : “192.168.184.133 dnsproject.prepa.com”.

Puis une fois cela fait, on entre dans le terminal “nano /etc/resolv.conf” pour rajouter “search dnsproject.prepa.com” sans oublié de mettre “#” devant les lignes déjà existantes.

Maintenant on remarque que si nous cherchons à accéder à notre serveur via navigateur on le peut désormais simplement en entrant son nom de domaine dans la barre de recherche.

JOB 5

Le nom de domaine est l’identifiant du site. Une fois un nom choisi, il faut le déposer afin qu’un organisme défini par la spécificité que nous avons choisi à notre site valide ou non notre proposition.

Les spécificités que l’on peut retrouver sur un nom de domaine sont les suivants :

gTLD : aussi appelé domaines génériques c’est à dire; com (pour les activités commerciales); .net (pour les entreprises); .org (pour les associations ou organisations non gouvernementales, etc.). On les utilise seulement à vocation international

ccTLD : à l’inverse, nous avons les ccTLD, aussi appelés codes de pays cad : .fr (France); .de (Allemagne); .it (Italie); .eu (Union européenne). Donc ces derniers sont à vocation nationale et/ou géographique.

JOB 6

Ce job a été réalisé en étant connecté au réseau WIFI en bridge. (10.10.29.226)

Dans **/etc/apache2/sites-enabled/000-default.conf**, ajouter la ligne:

*ServerName =* [*dnsproject.prepa.com*](http://dnsproject.prepa.com/)

Ajouter l’adresse du serveur dans **/etc/resolv.conf** :

Désactiver la réinitialisation automatique de **resolv.conf** :

Ajouter un fichier de configuration dans **/etc/NetworkManager/conf.d/** en **.conf** qui contiendra:

*[main]*

*dns=none*

Dans **/etc/resolv.conf**, ajouter les lignes:

*search* [*dnsproject.prepa.com*](http://dnsproject.prepa.com/)

*nameserver 10.10.29.226*

Dans **/etc/bind/named.conf.local**, ajouter les lignes:

*zone "*[*prepa.com*](http://prepa.com/)*" IN {*

*type master;*

*file "/etc/bind/prepa.forward.zone";*

*};*

*zone "29.10.10.in-addr.arpa" IN {*

*type master;*

*file "/etc/bind/prepa.reverse.zone";*

*};*

Créer les fichiers “**prepa.forward.zone**” et “**prepa.reverse.zone**” dans **/etc/bind**

Dans **prepa.forward.zone** :

*$TTL 604800*

*@   IN  SOA* [*prepa.com*](http://prepa.com/)*.* [*root.dnsproject.prepa.com*](http://root.dnsproject.prepa.com/)*. (*

*2     ; Serial*

*604800     ; Refresh*

*86400     ; Retry*

*2419200     ; Expire*

*604800 )   ; Negative Cache TTL*

*;*

*@   IN  NS* [*dnsproject.prepa.com*](http://dnsproject.prepa.com/)*.*

*dnsproject  IN  A   10.10.29.226*

*www IN  CNAME* [*dnsproject.prepa.com*](http://dnsproject.prepa.com/)*.*

Dans **prepa.reverse.zone** :

*$TTL 604800*

*@   IN  SOA* [*prepa.com*](http://prepa.com/)*.* [*root.dnsproject.prepa.com*](http://root.dnsproject.prepa.com/)*. (*

*2     ; Serial*

*604800     ; Refresh*

*86400     ; Retry*

*2419200     ; Expire*

*604800 )   ; Negative Cache TTL*

*;*

*@   IN  NS* [*dnsproject.prepa.com*](http://dnsproject.prepa.com/)*.*

*dnsproject  IN  A   10.10.29.226*

*58  IN  PTR* [*dnsproject.prepa.com*](http://dnsproject.prepa.com/)*.*

**prepa.forward.zone** va permettre au DNS de trouver l’ip du site web en fonction du nom de domaine, tandis que **prepa.reverse.zone** permet l’opération inverse.

Dans la configuration hôte, il faut modifier les paramètres DNS en ajoutant l’ip de la machine virtuelle comme DNS.

Comme le DNS de la VM n’est configuré que pour trouver un seul nom de domaine, il faut spécifier un forwarder sur l’hôte (sous Windows, entrer **8.8.8.8** ou **8.8.4.4** comme forwarder dans le champ “Autre DNS” ou “DNS Secondaire”, cela va permettre d’utiliser le serveur DNS public fourni par Google).

JOB 8 ( en NAT )

* sur nano /etc/network/interfaces :

auto lo

# The loopback network interface

iface lo inet loopback

#carte vers la box

auto ens33

iface ens33 inet static

address 192.168.0.1

network 192.168.0.0

netmask 255.255.255.0

gateway 192.168.0.254

#post-up iptables-restore < /etc/iptables.save # ( ligne à décommenter plus tard )

#carte vers le LAN

auto ens34

iface ens34 inet static

address 192.168.1.1

network 192.168.1.0

netmask 255.255.255.0

* prise en compte des modification : /etc/init.d/networking start
* modification du fichier /etc/sysctl.conf pour dé-commenter :

# Uncomment the next line to enable packet forwarding for IPv4 net.ipv4.ip\_forward=1

* prise en compte des modification : sysctl -p

* Mise en place du protocole NAT sur ens33

iptables -t nat -A POSTROUTING -o ens33 -j MASQUERADE

* Sauvegarde de cette modification :

iptables-save > /etc/iptables.save

* Édition de /etc/network/interfaces

Décommenter la ligne suivante :

post-up iptables-restore < /etc/iptables.save

* Au prochain démarrage du système :

iptables -L -t nat

* sur les VM reliés à la passerelle ajouter en Gateway 192.168.1.1

JOB 7

Pour installer les paquets DHCP :

cd /etc/bind

apt -get install isc-dhcp-server

ifconfig enp0s3 192.168.184.133

vim.tiny /etc/resolv.conf

serach dnsproject.prepa.com

name erver 192.168.184.133

systemctl restart bind9

nslookup www

vim.tiny /etc/default/isc-dhcp-server

cd

vim.tiny /etc/dhcp/dhcp.conf

On décommente :

subnet 192.168.1.0 netmask 255.255.255.0 {

range 192.168.1.2 192.168.1.50;

option domain-name-servers 192.168.1.1;

option domain-name "dnsproject.prepa.com";

option netbios-name-servers 192.168.1.1;

option routers 192.168.1.1;

option subnet-mask 255.255.255.0;

option broadcast-address 192.168.1.255;

default-lease-time 86400;

max-lease-time 676800;

}

JOB 9

* installer le pare-feu : apt-get update && apt-get install ufw
* On met en route le pare-feu : ufw enable
* Pour interdire le ping il faut commenter la ligne suivante dans le fichier /etc/ufw/before.rules :

 -A ufw-before-input -p icmp --icmp-type echo-request -j ACCEPT

* appliquer les changements : ufw disable && ufw enable

JOB 10

* installer Samba : apt install samba
* éditer fichier de configuration Samba /etc/samba/smb.conf :

[partage]

   comment = Partage de données

   path = /srv/partage

   guest ok = no

   read only = no

   browseable = yes

   valid users = @partage

* on redémarre samba : systemctl restart smbd\*
* Le groupe "partage" que nous avons déclaré dans la configuration n'existe pas. Nous allons créer le groupe avec la commande : groupadd partage
* créer les diverse utilisateurs avec la commande : adduser
* lui attribuer un mot de passe avec la commande : smbpasswd -a =”username”
* ajouter les utilisateurs aux groupes avec la commande : gpasswd -a “username” partage
* Le partage sera hébergé à l'emplacement "/srv/partage" du serveur.
* Créer le dossier : mkdir /srv/partage
* groupe “partage comme propriétaire du dossier : chgrp -R partage /srv/partage/
* ajouter droit écriture et lecture au groupe : chmod -R g+rw /srv/partage/

Pour aller plus loin :

* Activer le module SSL : a2enmod ssl
* Activer le module headers : a2enmod hearder ( le module hearders permet d’activer la directive HSTS. )
* redémarrer le service  : seystemctl restart apache2
* Création du répertoire qui contiendra le certificat SSL : mkdir /etc/ssl/dnsproject.prepa.com/
* Déclarer VirtualHosts pour le HTTPS; pour accéder à votre site en HTTPS il faut créer un VirtualHost dédié, pour cela :

Dans /etc/apache2/sites-available : dupliquer fichier de configuration “000-default.conf en 000-default-ssl.conf” avec :

cp 000-default.conf 000-default-ssl.conf

Modifiez le fichier dupliqué, remplacez le port 80 du VirtualHost par le port 443 : <VirtualHost \*:80> → <VirtualHost \*:443>

Ajoute les instructions suivantes avant le </VirtualHost>

SSLEngine On

SSLProtocol All -SSLv3 -SSLv2

SSLCipherSuite HIGH:!aNULL:!MD5:!ADH:!RC4:!DH

SSLHonorCipherOrder on

SSLCertificateFile "/etc/ssl/votre-domaine-fr/www.dnsproject.prepa.com.cer"

SSLCertificateKeyFile "/etc/ssl/votre-domaine-fr/www.dnsproject.prepa.com.key"

Header always set Strict-Transport-Security "max-age=15768000"

* activer VirtualHost : a2ensite nom-du-vhost-ssl
* redémarrer le service : systemctl restart apache2

Lorsque vous achetez un certificat SSL “traditionnel”, vous savez qu'il a été signé par une autorité de certification réputée. En revanche, un certificat auto-signé n'est pas signé par une autorité comme SSL ou TLS ; il est créé, mis en œuvre et signé par un développeur de logiciels tiers.

Le certificat est auto-signé, il n’est donc pas reconnu comme étant suffisamment sécurisé.