

1. Analyse du projet

- **Comprendre les besoins** : Reviens sur chaque point du projet pour bien comprendre les attentes et les contraintes. Identifie les éléments que tu maîtrises et ceux pour lesquels tu devras peut-être faire des recherches.

2. Planification des tâches

- **Découpage en étapes** :
 1. **Création des VM** : Installer les deux VM Debian avec les configurations matérielles spécifiées.
 2. **Installation et configuration du serveur** :
 - Installer et configurer Nginx, PHP (versions 7.x et 8.x), MariaDB.
 - Configurer le serveur DHCP/DNS, FTP (SSL/TLS), LDAP.
 - Configurer les domaines comme spécifié (www8.starfleet.lan, etc.).
 3. **Sécurisation** :
 - Mise en place du pare-feu.
 - Création des certificats SSL.
 4. **Tests et validations** :
 - Tester les fonctionnalités web avec la VM cliente.
 - Tester la connexion au serveur FTP et LDAP.
 5. **Documentation** :
 - Procédure pour l'export des VM.
 - Notice d'installation et d'utilisation pour l'utilisateur final.
 6. **Extension du projet** :
 - Installation de Visual Studio Code Server.
 - Développement du script de sauvegarde automatique.
 7. **Finalisation** :
 - Publier sur GitHub.
 - Préparation de la présentation.

3. Gestion du temps

- **Estimation du temps pour chaque étape** : Précise combien de temps tu penses consacrer à chaque étape. Réserve du temps pour les imprévus.
- **Délais** : Fixe des dates limites pour chaque tâche.

4. Mise en œuvre

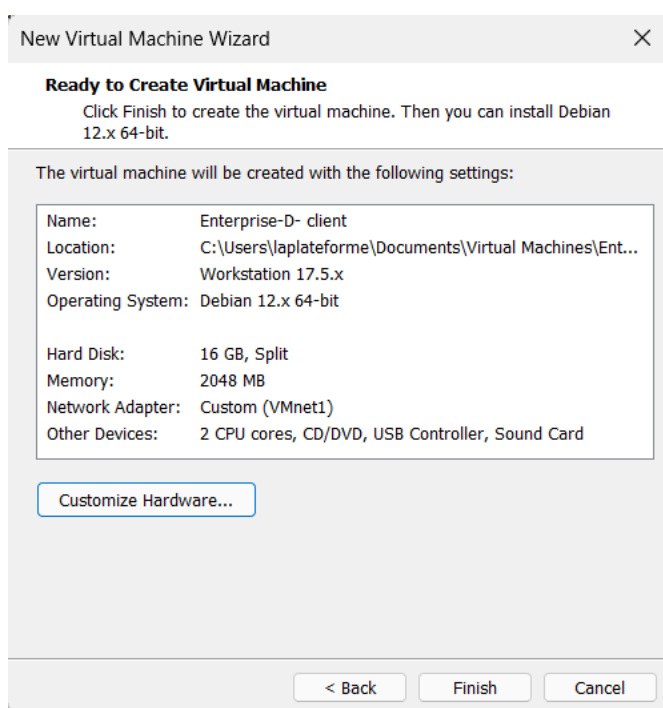
- **Suivi quotidien** : Travaille chaque jour sur une ou plusieurs tâches selon le temps disponible.
- **Révisions et ajustements** : Révise régulièrement ton avancement et ajuste ton planning si nécessaire.

5. Finalisation et Rendu

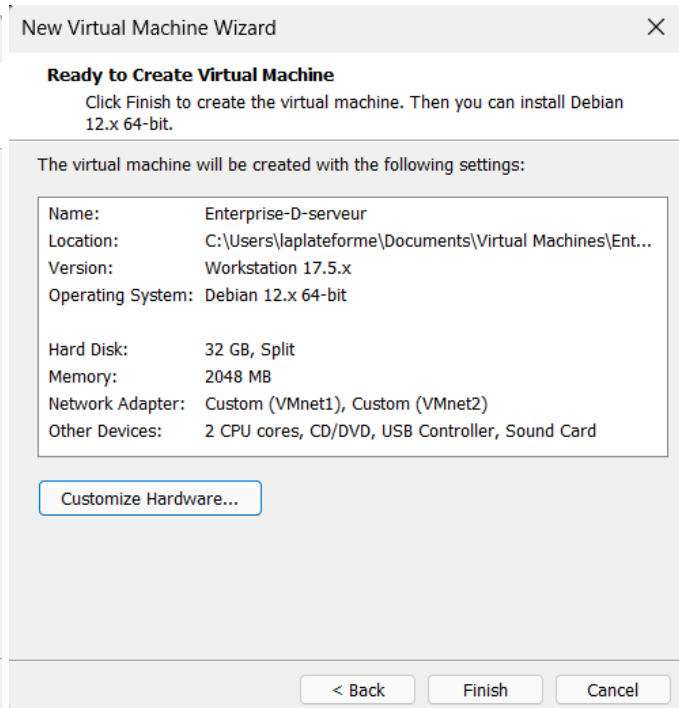
- **Vérifications finales** : Assure-toi que tout est documenté, testé et fonctionnel.
- **Mise en ligne** : Publie le projet sur GitHub en suivant les consignes données.
- **Préparation de la présentation** : Crée un support visuel pour présenter ton projet à l'équipe pédagogique.

paramétrer les VM

client



serveur



Configuration du serveur DHCP

pour installer le serveur DHCP lancer la commande
apt install isc-dhcp-server

```
root@server-web:~# apt install isc-dhcp-server
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les paquets supplémentaires suivants seront installés :
  policycoreutils selinux-utils
Paquets suggérés :
  policykit-1 isc-dhcp-server-ldap ieee-data
Les NOUVEAUX paquets suivants seront installés :
  isc-dhcp-server policycoreutils selinux-utils
0 mis à jour, 3 nouvellement installés, 0 à enlever et 0 non mis à jour.
Il est nécessaire de prendre 1 766 ko dans les archives.
Après cette opération, 7 818 ko d'espace disque supplémentaires seront utilisés.
Souhaitez-vous continuer ? [O/n]
```

pour la configuration de l'interface d'écoute on va indiquer quelle carte réseau nous utiliserons
la carte réseau sera là (ens34) (lan)

```
root@server-web:~# nano /etc/default/isc-dhcp-server_
```

```
GNU nano 7.2 /etc/default/isc-dhcp-server
# Defaults for isc-dhcp-server (sourced by /etc/init.d/isc-dhcp-server)

# Path to dhcpd's config file (default: /etc/dhcp/dhcpd.conf).
#DHCPDv4_CONF=/etc/dhcp/dhcpd.conf
#DHCPDv6_CONF=/etc/dhcp/dhcpd6.conf

# Path to dhcpd's PID file (default: /var/run/dhcpd.pid).
#DHCPDv4_PID=/var/run/dhcpd.pid
#DHCPDv6_PID=/var/run/dhcpd6.pid

# Additional options to start dhcpd with.
# Don't use options -cf or -pf here; use DHCPD_CONF/ DHCPD_PID instead
#OPTIONS=""

# On what interfaces should the DHCP server (dhcpd) serve DHCP requests?
# Separate multiple interfaces with spaces, e.g. "eth0 eth1".
INTERFACESv4="ens34"
INTERFACESv6=""
```

je ne suis pas sûr

Configuration d'attribution automatique d'adresse IP

```
root@server-web:~# nano /etc/dhcp/dhcpd.conf_
```

```

GNU nano 7.2 /etc/dhcp/dhcpd.conf
# to which a BOOTP client is connected which has the dynamic-bootp flag
# set.
#host fantasia {
#   hardware ethernet 08:00:07:26:c0:a5;
#   fixed-address fantasia.example.com;
#}

# You can declare a class of clients and then do address allocation
# based on that. The example below shows a case where all clients
# in a certain class get addresses on the 10.17.224/24 subnet, and all
# other clients get addresses on the 10.0.29/24 subnet.

#class "foo" {
#   match if substring (option vendor-class-identifier, 0, 4) = "SUNW";
#}

#shared-network 224-29 {
#   subnet 10.17.224.0 netmask 255.255.255.0 {
#       option routers rtr-224.example.org;
#   }
#   subnet 10.0.29.0 netmask 255.255.255.0 {
#       option routers rtr-29.example.org;
#   }
#   pool {
#       allow members of "foo";
#       range 10.17.224.10 10.17.224.250;
#   }
#   pool {
#       deny members of "foo";
#       range 10.0.29.10 10.0.29.230;
#   }
#}

subnet 192.168.116.0 netmask 255.255.255.0 {
    range 192.168.116.10 192.168.116.50;
    option routers 192.168.116.1;
    option subnet-mask 255.255.0.0;
    option domain-name-servers 192.168.116.1;
}

```

installation bind9

```
root@server-web:~# apt instal bind9
```

création du backup (sauvegarde du DNS)

```
root@server-web:~# cp /etc/bind/named.conf.local /etc/bind/named.conf.local.save
```

```

root@server-web:~# ls /etc/bind
bind.keys  db.127  db.empty  db.starfleet.lan  named.conf.default-zones  named.conf.local.save  rndc.key
db.0      db.255  db.local  named.conf        named.conf.local        named.conf.options    zones.rfc1918
root@server-web:~# _

```

création du répertoire starfleet.lan

```
root@server-web:~# cp /etc/bind/db.0 /etc/bind/db.starfleet.lan_
```

enregistrement dns en indiquant l'adresse ip du lan segmente qui est 172.16.128.33

```
root@server-web:~# nano /etc/bind/db.starfleet.lan_
```

```
GNU nano 7.2 /etc/bind/db.starfleet.lan
;
; BIND reverse data file for "this host on this network" zone
;
$TTL      604800
@         IN      SOA      localhost. root.localhost. (
                        1          ; Serial
                        604800     ; Refresh
                        86400      ; Retry
                        2419200    ; Expire
                        604800 )   ; Negative Cache TTL
;
#@        IN      NS       localhost.
@         IN      ns       dns.starfleet.lan.
dns       A       172.16.128.33
```

Il faut indiquer le chemin d'accès de db.starfleet.lan sur le répertoire named.conf.local

```
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 "starfleet.lan" {
//    type master;
//    file "/etc/bind/db.starfleet.lan";
//};
```

```
root@server-web:~# systemctl status bind9
• named.service - BIND Domain Name Server
   Loaded: loaded (/lib/systemd/system/named.service; enabled; preset: enabled)
   Active: active (running) since Mon 2024-09-16 00:09:56 CEST; 34min ago
     Docs: man:named(8)
   Main PID: 976 (named)
    Status: "running"
   Tasks: 8 (limit: 2264)
  Memory: 36.6M
     CPU: 44ms
   CGroup: /system.slice/named.service
           └─976 /usr/sbin/named -f -u bind

sept. 16 00:09:56 server-web named[976]: network unreachable resolving './DNSKEY/IN': 2001:500:9f::42#53
sept. 16 00:09:56 server-web named[976]: network unreachable resolving './NS/IN': 2001:500:9f::42#53
sept. 16 00:09:56 server-web named[976]: network unreachable resolving './DNSKEY/IN': 2001:500:2f::f#53
sept. 16 00:09:56 server-web named[976]: network unreachable resolving './NS/IN': 2001:500:2f::f#53
sept. 16 00:09:56 server-web named[976]: network unreachable resolving './DNSKEY/IN': 2001:503:ba3e::2:30#53
sept. 16 00:09:56 server-web named[976]: network unreachable resolving './NS/IN': 2001:503:ba3e::2:30#53
sept. 16 00:09:56 server-web named[976]: network unreachable resolving './DNSKEY/IN': 2001:500:2d::d#53
sept. 16 00:09:56 server-web named[976]: network unreachable resolving './NS/IN': 2001:500:2d::d#53
sept. 16 00:09:56 server-web named[976]: managed-keys-zone: Key 20326 for zone . is now trusted (acceptance timer compl
sept. 16 00:09:56 server-web named[976]: resolver priming query complete: success
root@server-web:~#
```

serveur Web Nginx en HTTPS.

commande installation NGINX

```
root@enterprise-serveur:/home/enterprise-serveur# apt-get install nginx -y
```

installation NGINX

```

root@entreprise-serveur:/home/entreprise-serveur# apt-get install nginx -y
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les paquets supplémentaires suivants seront installés :
  nginx-common
Paquets suggérés :
  fcgiwrap nginx-doc ssl-cert
Les NOUVEAUX paquets suivants seront installés :
  nginx nginx-common
0 mis à jour, 2 nouvellement installés, 0 à enlever et 0 non mis à jour.
Il est nécessaire de prendre 640 ko dans les archives.
Après cette opération, 1 696 ko d'espace disque supplémentaires seront utilisés.
Réception de :1 http://deb.debian.org/debian bookworm/main amd64 nginx-common all 1.22.1-9 [112 kB]
Réception de :2 http://deb.debian.org/debian bookworm/main amd64 nginx amd64 1.22.1-9 [527 kB]
640 ko réceptionnés en 0s (3 974 ko/s)
Préconfiguration des paquets...
Sélection du paquet nginx-common précédemment désélectionné.
(Lecture de la base de données... 34236 fichiers et répertoires déjà installés.)
Préparation du dépaquetage de .../nginx-common_1.22.1-9_all.deb ...
Dépaquetage de nginx-common (1.22.1-9) ...
Sélection du paquet nginx précédemment désélectionné.
Préparation du dépaquetage de .../nginx_1.22.1-9_amd64.deb ...
Dépaquetage de nginx (1.22.1-9) ...
Paramétrage de nginx-common (1.22.1-9) ...
Created symlink /etc/systemd/system/multi-user.target.wants/nginx.service → /lib/systemd/system/nginx.service.
Paramétrage de nginx (1.22.1-9) ...
Upgrading binary: nginx.
Traitement des actions différées (« triggers ») pour man-db (2.11.2-2) ...
root@entreprise-serveur:/home/entreprise-serveur#

```

version d'NGINX

```

root@entreprise-serveur:~# nginx -v
nginx version: nginx/1.22.1
root@entreprise-serveur:~#

```

en cours d'exécution

```

root@entreprise-serveur:~# systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/lib/systemd/system/nginx.service; enabled; preset: enabled)
   Active: active (running) since Fri 2024-09-06 11:55:56 CEST; 8min ago
     Docs: man:nginx(8)
   Process: 3010 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
   Process: 3011 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
  Main PID: 3035 (nginx)
    Tasks: 3 (limit: 2264)
   Memory: 2.5M
      CPU: 78ms
   CGroup: /system.slice/nginx.service
           └─3035 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
             └─3038 "nginx: worker process"
               └─3039 "nginx: worker process"

sept. 06 11:55:56 entreprise-serveur systemd[1]: Starting nginx.service - A high performance web server and a reverse pr
sept. 06 11:55:56 entreprise-serveur systemd[1]: Started nginx.service - A high performance web server and a reverse pr
root@entreprise-serveur:~# _

```

NGINX démarrera automatiquement au démarrage du système

```

root@enterprise-serveur:~# systemctl enable nginx
Synchronizing state of nginx.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable nginx
root@enterprise-serveur:~# systemctl is-enabled nginx
enabled
root@enterprise-serveur:~# _

```

démarrer le serveur NGINX

```

root@enterprise-serveur:~# systemctl start nginx
root@enterprise-serveur:~# systemctl status nginx
• nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/lib/systemd/system/nginx.service; enabled; preset: enabled)
   Active: active (running) since Sun 2024-09-08 22:35:15 CEST; 6s ago
     Docs: man:nginx(8)
   Process: 4245 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
   Process: 4246 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
  Main PID: 4247 (nginx)
    Tasks: 3 (limit: 2264)
   Memory: 2.3M
      CPU: 39ms
   CGroup: /system.slice/nginx.service
           └─4247 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
             └─4248 "nginx: worker process"
               └─4249 "nginx: worker process"

sept. 08 22:35:15 enterprise-serveur systemd[1]: Starting nginx.service - A high performance web server and a reverse pr
sept. 08 22:35:15 enterprise-serveur systemd[1]: Started nginx.service - A high performance web server and a reverse pr
root@enterprise-serveur:~# _

```

arrêter NGINX

```

root@enterprise-serveur:~# systemctl stop nginx
root@enterprise-serveur:~# systemctl status nginx
1: 1
♦ nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/lib/systemd/system/nginx.service; enabled; preset: enabled)
   Active: inactive (dead) since Sun 2024-09-08 22:16:21 CEST; 15min ago
     Docs: man:nginx(8)
   Process: 4205 ExecStop=/sbin/start-stop-daemon --quiet --stop --retry QUIT/5 --pidfile /run/nginx.pid (code=exited,
  Main PID: 3005 (code=exited, status=0/SUCCESS)
      CPU: 96ms

sept. 06 11:55:56 enterprise-serveur systemd[1]: Starting nginx.service - A high performance web server and a reverse pr
sept. 06 11:55:56 enterprise-serveur systemd[1]: Started nginx.service - A high performance web server and a reverse pr
sept. 08 22:16:21 enterprise-serveur systemd[1]: Stopping nginx.service - A high performance web server and a reverse pr
sept. 08 22:16:21 enterprise-serveur systemd[1]: nginx.service: Deactivated successfully.
sept. 08 22:16:21 enterprise-serveur systemd[1]: Stopped nginx.service - A high performance web server and a reverse pr
root@enterprise-serveur:~# _

```



```
bash: cd: trop d'arguments
root@enterprise-serveur:~# cd /var/www/html/
root@enterprise-serveur:/var/www/html# ls
index.nginx-debian.html
root@enterprise-serveur:/var/www/html#
```

commende pour entrer dans le fichier

```
root@enterprise-serveur:/var/www/html# cat index.nginx-debian.html
```

contenue du fichier index.nginx-debian.html

```
index.nginx-debian.html
root@enterprise-serveur:/var/www/html# cat index.nginx-debian.html
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
root@enterprise-serveur:/var/www/html#
```

configuration du site par default

```
GNU nano 7.2 /etc/nginx/sites-enabled/default
##
# You should look at the following URL's in order to grasp a solid understanding
# of Nginx configuration files in order to fully unleash the power of Nginx.
# https://www.nginx.com/resources/wiki/start/
# https://www.nginx.com/resources/wiki/start/topics/tutorials/config_pitfalls/
# https://wiki.debian.org/Nginx/DirectoryStructure
#
# In most cases, administrators will remove this file from sites-enabled/ and
# leave it as reference inside of sites-available where it will continue to be
# updated by the nginx packaging team.
#
# This file will automatically load configuration files provided by other
# applications, such as Drupal or Wordpress. These applications will be made
# available underneath a path with that package name, such as /drupal8.
#
# Please see /usr/share/doc/nginx-doc/examples/ for more detailed examples.
##

# Default server configuration
#
server {
    listen 80 default_server;
    listen [::]:80 default_server;

    # SSL configuration
    #
    # listen 443 ssl default_server;
    # listen [::]:443 ssl default_server;
    #
    # Note: You should disable gzip for SSL traffic.
    # See: https://bugs.debian.org/773332
    #
    # Read up on ssl_ciphers to ensure a secure configuration.
    # See: https://bugs.debian.org/765782
    #
    # Self signed certs generated by the ssl-cert package
    # Don't use them in a production server!
    #
    # include snippets/snakeoil.conf;

    root /var/www/html;

    # Add index.php to the list if you are using PHP
    index index.html index.htm index.nginx-debian.html;

    server_name _;
}
```

nano commande de configuration globale

```
root@enterprise-serveur:/var/www/html# cd /
root@enterprise-serveur:/# nano /etc/nginx/nginx.conf
```

fichier de configuration globale

```
GNU nano 7.2 /etc/nginx/nginx.conf
user www-data;
worker_processes auto;
pid /run/nginx.pid;
error_log /var/log/nginx/error.log;
include /etc/nginx/modules-enabled/*.conf;

events {
    worker_connections 768;
    # multi_accept on;
}

http {

    ##
    # Basic Settings
    ##

    sendfile on;
    tcp_nopush on;
    types_hash_max_size 2048;
    # server_tokens off;

    # server_names_hash_bucket_size 64;
    # server_name_in_redirect off;

    include /etc/nginx/mime.types;
    default_type application/octet-stream;

    ##
    # SSL Settings
    ##

    ssl_protocols TLSv1 TLSv1.1 TLSv1.2 TLSv1.3; # Dropping SSLv3, ref: P000LE
    ssl_prefer_server_ciphers on;

    ##
    # Logging Settings
    ##

    access_log /var/log/nginx/access.log;

    ##
    # Gzip Settings
    ##

    gzip on;

    [ Lecture de 83 lignes ]

```

Aide Quitter Écrire Lire fich. Chercher Remplacer Couper Coller Exécuter Justifier Emplacement Aller ligne Annuler Refaire Hand Cop.

dossier sites-enabled sites-available

```
root@enterprise-serveur:/# ls /etc/nginx/
conf.d      fastcgi_params  koi-win        modules-available  nginx.conf      scgi_params     sites-enabled  uwsgi_params
fastcgi.conf koi-utf         mime.types     modules-enabled    proxy_params    sites-available snippets       win-utf
root@enterprise-serveur:/#
```

créer un site avec NGINX