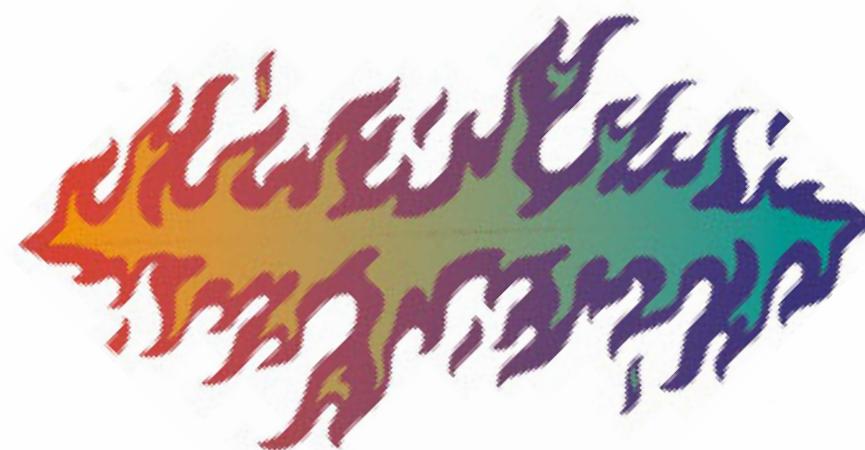
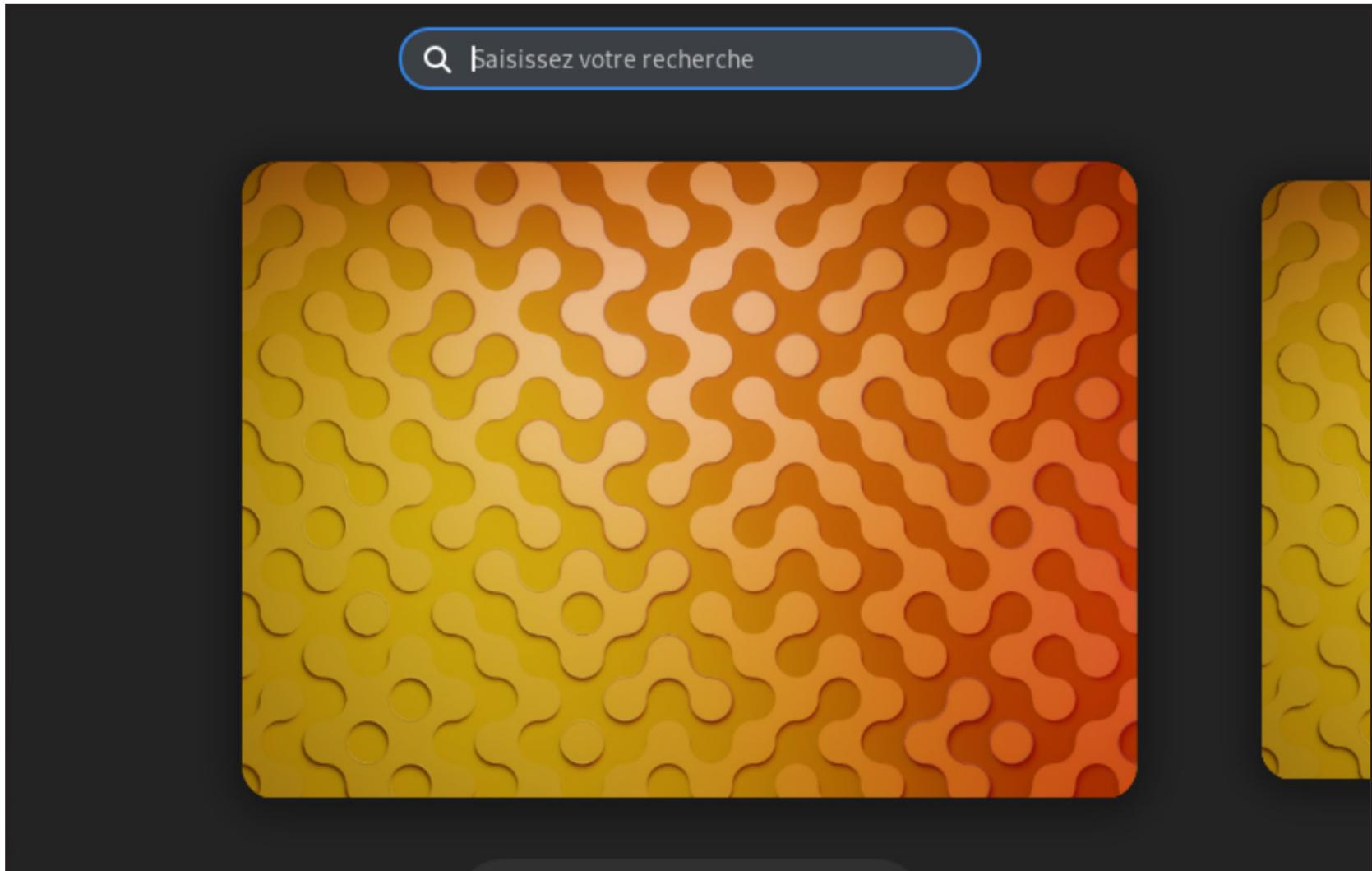


DDWS



CREATION VM INTERFACE GRAPHIQUE

01



Pour cette exercice j'utilise une VM debian avec VirtualBox. L'installation n'est pas bien différentes d'une autre VM .

INSTALLATION SERVEUR APACHE2

02

Pour l'installation du serveur Apache2, rien de plus simple.
Avant de commencer on oublie pas la commande :

sudo apt update && sudo apt upgrade

pour mettre les paquets à jour.
Il suffit ensuite d'entrer la commande :

sudo apt install apache2

```
Debian Graphique [En fonction] - Oracle VM VirtualBox
Activités Terminal 25 oct. 11:11 fr, ☰
barbz@debian:~
E: Impossible de verrouiller le répertoire /var/lib/apt/lists/
W: Problème de suppression du lien /var/cache/apt/pkgcache.bin - RemoveCaches (13: Permission non accordée)
W: Problème de suppression du lien /var/cache/apt/srcpkgcache.bin - RemoveCaches (13: Permission non accordée)
barbz@debian:~$ sudo apt update && sudo apt full-upgrade
[sudo] Mot de passe de barbz :
barbz n'est pas dans le fichier sudoers.
barbz@debian:~$ su
Mot de passe :
root@debian:/home/barbz# apt update && apt full-upgrade
Atteint :1 http://deb.debian.org/debian bookworm InRelease
Atteint :2 http://security.debian.org/debian-security bookworm-security InRelease
Atteint :3 http://deb.debian.org/debian bookworm-updates InRelease
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Tous les paquets sont à jour.
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Calcul de la mise à jour... Fait
0 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à jour.
```

```
barbz@debian:/etc/apache2/sites-available$ sudo apt install apache2
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
apache2 est déjà la version la plus récente (2.4.57-2).
0 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à jour.
barbz@debian:/etc/apache2/sites-available$
```

```

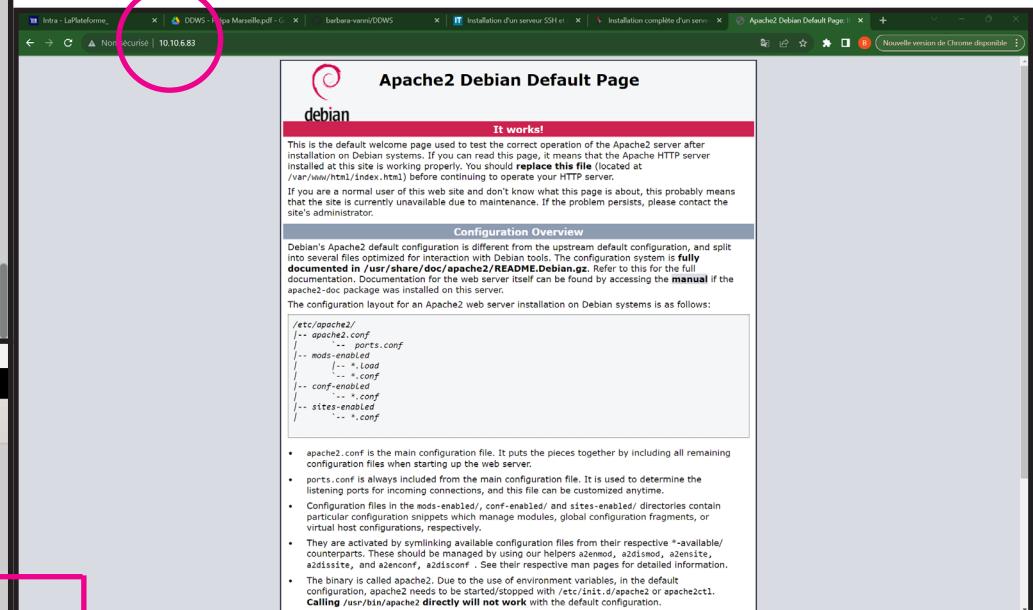
Activités Terminal 25 oct. 11:19 fr_
barbz@debian:~ 
root@debian:/home/barbz# sudo systemctl list-units --type=service | grep apache
apache2.service loaded active running The Apache HTTP Se
rver
root@debian:/home/barbz# systemctl restart apache2.service
root@debian:/home/barbz# systemctl status apache2.service
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; preset: enabled)
   Active: active (running) since Wed 2023-10-25 11:18:48 CEST; 13s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 3604 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
 Main PID: 3609 (apache2)
   Tasks: 56 (limit: 2285)
  Memory: 17.8M
    CPU: 49ms
   CGroup: /system.slice/apache2.service
           └─3609 /usr/sbin/apache2 -k start
              ├─3610 /usr/sbin/apache2 -k start
              ├─3611 /usr/sbin/apache2 -k start
              ├─3612 /usr/sbin/apache2 -k start

oct. 25 11:18:48 debian systemd[1]: Starting apache2.service - The Apache HTTP Server.>
oct. 25 11:18:48 debian apachectl[3608]: AH00558: apache2: Could not reliably determin>
oct. 25 11:18:48 debian systemd[1]: Started apache2.service - The Apache HTTP Server.
lines 1-18/18 (END)

Debian Graphique [En fonction] - Oracle VM VirtualBox
Activités Terminal 25 oct. 12:09 fr_
barbz@debian:~ 
root@debian:/home/barbz# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
  link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
  inet 127.0.0.1/8 scope host lo
    valid_lft forever preferred_lft forever
  inet6 ::1/128 scope host noprefixroute
    valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group def
ault qlen 1000
  link/ether 08:00:27:c9:f2:bb brd ff:ff:ff:ff:ff:ff
  inet 10.10.6.83/16 brd 10.10.255.255 scope global dynamic noprefixroute enp0s3
    valid_lft 537sec preferred_lft 537sec
  inet6 fe80::a00:27ff:fe9:f2bb/64 scope link noprefixroute
    valid_lft forever preferred_lft forever
root@debian:/home/barbz# hostname -I
10.10.6.83
root@debian:/home/barbz#

```

Dans un premier temps on active notre serveur en utilisant la commande :
systemctl status apache2.service



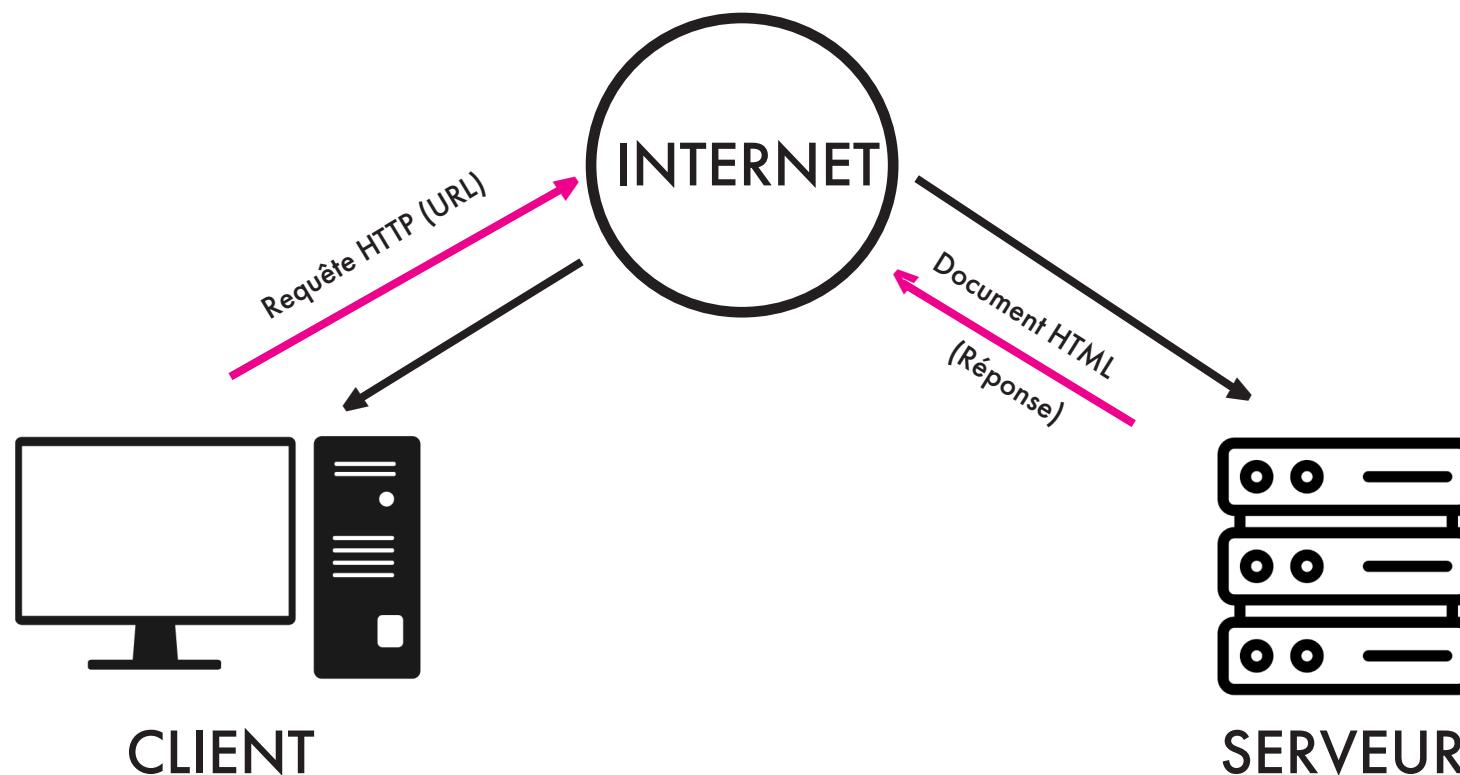
On vient ensuite récupérer l'adresse IP de notre VM et donc de notre serveur avec :
ip a

On ira ensuite rentrer cette adresse IP dans notre moteur de recherche pour voir apparaître la page Apache2

••• Qu'est-ce-qu'un serveur web?

Un **serveur web** est un logiciel ou un programme informatique qui gère les requêtes HTTP (Hypertext Transfer Protocol) provenant des clients, généralement des navigateurs Web, et renvoie les pages web correspondantes aux clients. Le rôle principal d'un serveur web est de stocker, traiter et diffuser des fichiers web aux utilisateurs finaux sur Internet.

Les serveurs web suivent un modèle client-serveur. Dans cette structure, un programme, également appelé client, demande une ressource ou un service à un autre programme, le serveur.



APACHE HTTP

- | AVANTAGES | INCONVENIENTS |
|---|---|
| <ul style="list-style-type: none">Très populaire et largement utilisé dans le monde entier.Grande communauté de support et de documentation.Hautement configurable.Prise en charge de modules tiers pour l'extension de fonctionnalités. | <ul style="list-style-type: none">Peut être plus complexe à configurer pour les débutants que d'autres serveurs.La configuration peut nécessiter la modification de nombreux fichiers. |

NGINX

- | AVANTAGES | INCONVENIENTS |
|--|---|
| <ul style="list-style-type: none">Conçu pour la performance et la rapidité, adapté aux sites à fort trafic.Excellente gestion de nombreuses connexions simultanées (événement-driven).Faible utilisation des ressources système. | <ul style="list-style-type: none">Moins flexible pour le traitement de scripts dynamiques par rapport à Apache.La configuration nécessite une approche différente de celle d'Apache. |

LITESPEED

- | AVANTAGES | INCONVENIENTS |
|---|---|
| <ul style="list-style-type: none">Haute performance, conçu pour rivaliser avec Nginx.Facile à configurer et dispose d'une interface utilisateur conviviale.Prise en charge de l'accélération des applications Web, notamment PHP. | <ul style="list-style-type: none">La version gratuite a des limitations par rapport à la version payante. |

CADDY

- | AVANTAGES | INCONVENIENTS |
|---|--|
| <ul style="list-style-type: none">Facilité d'utilisation, avec une configuration automatique via HTTPS.Gestion automatisée des certificats SSL grâce à Let's Encrypt.Prise en charge de la configuration via des fichiers Caddyfile faciles à lire. | <ul style="list-style-type: none">Moins flexible que certains serveurs pour des cas d'utilisation avancés. |

IIS

(Microsoft Internet Information Services)

AVANTAGES

- Intégré aux systèmes Windows Server.
- Prise en charge de technologies Microsoft telles que ASP.NET.
- Bonne intégration avec d'autres produits Microsoft.

INCONVENIENTS

- Limité aux environnements Windows.

CHEROKEE

AVANTAGES

- Interface Web conviviale pour la configuration.
- Prise en charge de nombreuses fonctionnalités, notamment la répartition de charge.
- Performant et évolutif.

INCONVENIENTS

- La documentation et la communauté sont moins étendues que celles d'Apache et de Nginx.

LIGHTTPD

AVANTAGES

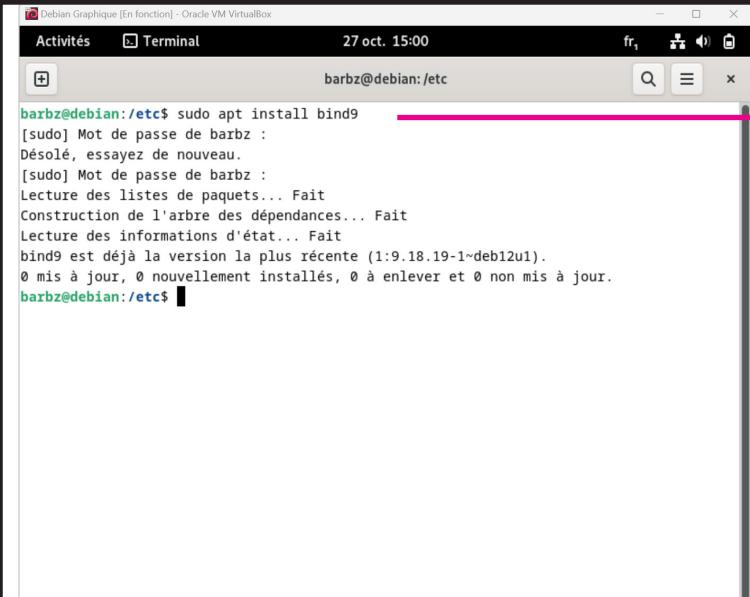
- Léger et conçu pour l'efficacité.
- Convient aux serveurs avec des ressources limitées.
- Prise en charge de la compression de contenu.

INCONVENIENTS

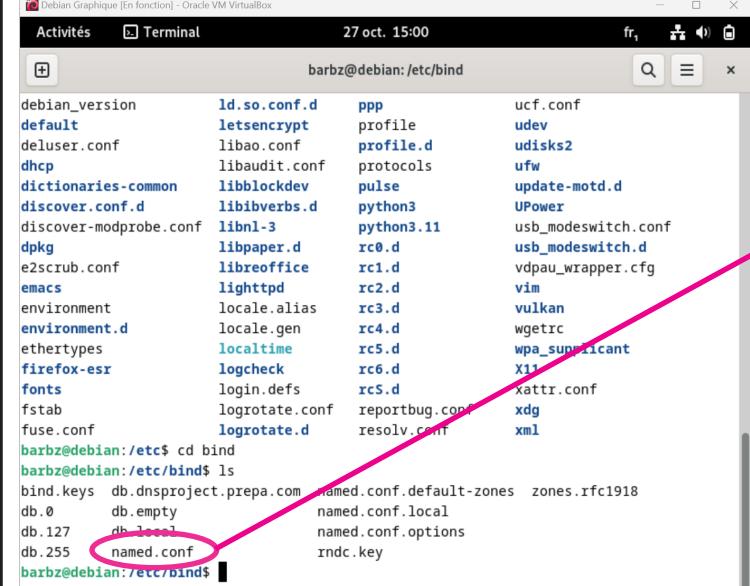
- Peut ne pas être aussi rapide que Nginx dans certaines situations.

MISE EN PLACE SERVEUR DNS

04

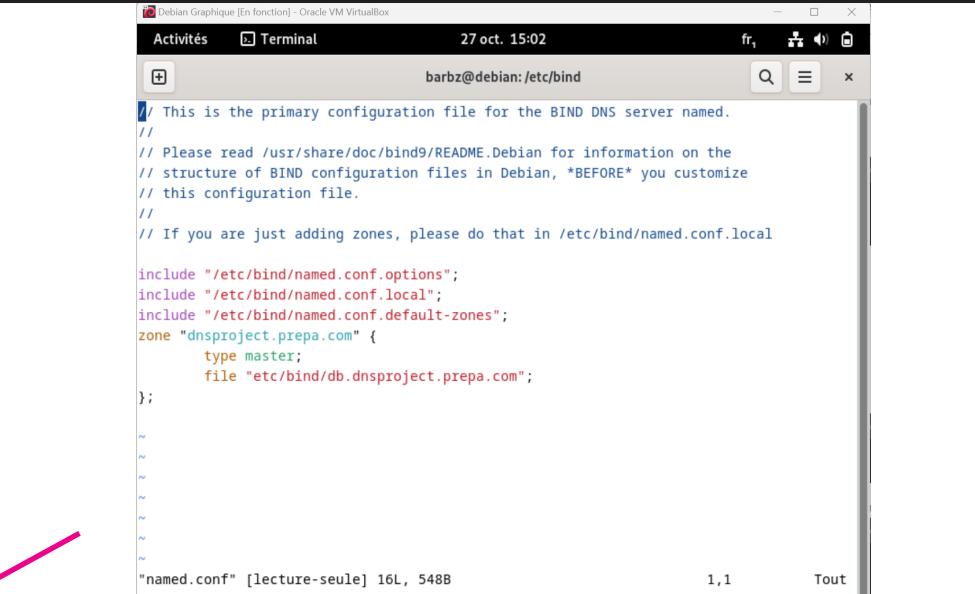


```
barbz@debian:/etc$ sudo apt install bind9
[sudo] Mot de passe de barbz :
Désolé, essayez de nouveau.
[sudo] Mot de passe de barbz :
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
bind9 est déjà la version la plus récente (1:9.18.19-1~deb12u1).
0 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à jour.
barbz@debian:/etc$
```



```
barbz@debian:/etc$ cd bind
barbz@debian:/etc/bind$ ls
bind.keys db.dnsproject.prepa.com named.conf.default-zones zones.rfc1918
db.0 db.empty named.conf.local named.conf.options rndc.key
db.127 db.local
db.255 named.conf
barbz@debian:/etc/bind$
```

On va d'abord installer bind9 qui est un logiciel pour serveur DNS. C'est un service TCP/IP permettant la correspondance entre un nom de domaine qualifié (FQDN : Fully Qualified Domain Name) et une adresse IP



```
// This is the primary configuration file for the BIND DNS server named.
//
// Please read /usr/share/doc/bind9/README.Debian for information on the
// structure of BIND configuration files in Debian, *BEFORE* you customize
// this configuration file.
//
// If you are just adding zones, please do that in /etc/bind/named.conf.local
include "/etc/bind/named.conf.options";
include "/etc/bind/named.conf.local";
include "/etc/bind/named.conf.default-zones";
zone "dnsproject.prepa.com" {
    type master;
    file "etc/bind/db.dnsproject.prepa.com";
};

named.conf [lecture-seule] 16L, 548B
```

Une fois Bind installé, on passe à la configuration de notre serveur DNS pour faire correspondre l'adresse IP de votre serveur au nom de domaine «dnsproject.prepa.com». Sans cet enregistrement DNS, les requêtes DNS pour «dnsproject.prepa.com» ne sauraient pas comment trouver votre serveur. Cette étape permet au serveur DNS de diriger les requêtes appropriées vers l'adresse IP correcte.

MISE EN PLACE SERVEUR DNS

04

Dans le dossier /etc/bind/ on crée un fichier db.dnsproject.prepa.com
Qui sera notre fichier permettant de définir notre «zone» et surtout d'associé l'adresse IP de notre machine à ce nom de domaine.

De sorte qu'on puisse dans un moteur de recherche rentrer le nom de domaine et arriver au même résultat qu'avec l'adresse IP

Apache2 Debian Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Debian systems. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should [replace this file](#) (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Debian's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Debian tools. The configuration system is **fully documented in `/usr/share/doc/apache2/README.debian.gz`**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the [manual](#) if the apache2-doc package was installed on this server.

The configuration layout for an Apache2 web server installation on Debian systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|   |-- ports.conf
|   |-- mods-enabled
|   |   |-- *.load
|   |   |-- *.conf
|   |-- conf-enabled
|   |   |-- *.conf
|   |-- sites-enabled
|   |   |-- *.conf
|   |
|   |-- sites-available
|       |-- *.conf
```

- `apache2.conf` is the main configuration file. It puts the pieces together by including all remaining configuration files when starting up the web server.
- `ports.conf` is always included from the main configuration file. It is used to determine the listening port for accepting connections, and this file can be customized anytime.
- Configuration files in the `mods-enabled`, `conf-enabled`, and `sites-enabled` directories contain particular configuration snippets which manage modules, global configuration fragments, or virtual host configurations, respectively.
- They are activated by symlinking available configuration files from their respective `*-available` counterparts. These should be managed by using our helpers `a2enmod`, `a2dismod`, `a2enconf`, `a2disconf`. See their respective man pages for detailed information.
- The binary is called `apache2`. Due to the use of environment variables, in the default configuration, `apache2` needs to be started/stopped with `/etc/init.d/apache2` or `apache2ctl`. Calling `/usr/bin/apache2` directly will not work with the default configuration.

••• Structure d'un nom de domaine

Prenons comme exemple le nom de domaine du site OVHcloud : www.ovhcloud.com

- **www** : correspond au domaine de troisième niveau (World Wide Web).
- **ovhcloud** : correspond au domaine de deuxième niveau, c'est le nom du site.
- **.com** : correspond au domaine de premier niveau, aussi appelé extension de nom de domaine (ou TLD, top-level domain).

••• Public ou Privé ?

Les termes «nom de domaine privé» et «nom de domaine public» ne sont pas des termes techniques standard en matière de noms de domaine. Cependant, ils sont souvent utilisés informellement pour décrire deux concepts différents liés aux noms de domaine.

- **Public** : Un nom de domaine public est un nom de domaine qui est enregistré auprès d'un registraire de noms de domaine et qui est accessible via Internet. Les noms de domaine publics sont utilisés pour identifier des sites web, des serveurs de messagerie, des serveurs de fichiers, etc., sur Internet. Ils sont enregistrés dans une base de données publique et sont accessibles à toute personne qui utilise Internet.
On obtient donc son nom de domaine en le soumettant à un registraire de son choix qui va ensuite l'approuver ou non.
- **Privé** : Le terme «nom de domaine privé» n'a pas de définition standard, mais il est parfois utilisé pour faire référence à un nom de domaine qui n'est pas destiné à être accessible au public. Cela pourrait signifier un nom de domaine utilisé à des fins de test en interne dans une organisation, ou un nom de domaine utilisé pour des communications internes. Ces noms de domaine ne sont pas destinés à être utilisés sur Internet de manière publique.

••• Les extensions des nom de domaine

Spécificités des extensions de nom de domaine (TLD, pour Top-Level Domain), il existe plusieurs «classes» de TLD:

TLD génériques : Ce sont des extensions de domaine couramment utilisées. Certaines des plus populaires incluent :

.com : L'extension de domaine la plus répandue, généralement utilisée pour des sites Web commerciaux.

.org : Généralement associée à des organisations à but non lucratif.

.net : Utilisée par divers types d'organisations.

.gov : Réservée aux entités gouvernementales aux États-Unis.

.edu : Réservée aux institutions éducatives.

TLD géographiques : Ils sont associés à des régions géographiques spécifiques. Par exemple :

.fr : Pour la France.

.uk : Pour le Royaume-Uni.

.ca : Pour le Canada.

TLD de premier niveau sponsorisé : Ils sont gérés par des organisations ou des gouvernements qui imposent des restrictions sur leur utilisation. Par exemple,

.aero est destiné à l'industrie de l'aviation, et .museum est pour les musées.

TLD de premier niveau restreint : Ce sont des TLD spécifiques qui ont des exigences d'admissibilité. Par exemple, .mil est réservé pour les entités militaires américaines, et .edu est généralement réservé pour les établissements d'enseignement postsecondaire.

Nouveaux TLD génériques : Au fil des ans, de nombreuses nouvelles extensions de domaine génériques ont été introduites, telles que .app, .blog, .guru, etc. Ces TLD visent à offrir plus de choix aux propriétaires de sites Web.

Personnalisation de TLD : Certains domaines permettent désormais une personnalisation supplémentaire, ce qui signifie que vous pouvez créer votre propre extension de domaine, mais cela a des restrictions et des coûts spécifiques.

The screenshot shows a web browser window with multiple tabs open. The active tab displays the Apache2 Debian Default Page. The page has a header with the Debian logo and the title "Apache2 Debian Default Page". A red banner at the top contains the text "It works!". Below the banner, there is explanatory text about the page's purpose and how to replace the default index file. A "Configuration Overview" section follows, featuring a file tree diagram and a bulleted list detailing the configuration layout.

Apache2 Debian Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Debian systems. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at /var/www/html/index.htm1) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Debian's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Debian tools. The configuration system is **fully documented in /usr/share/doc/apache2/README.Debian.gz**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the apache2-doc package was installed on this server.

The configuration layout for an Apache2 web server installation on Debian systems is as follows:

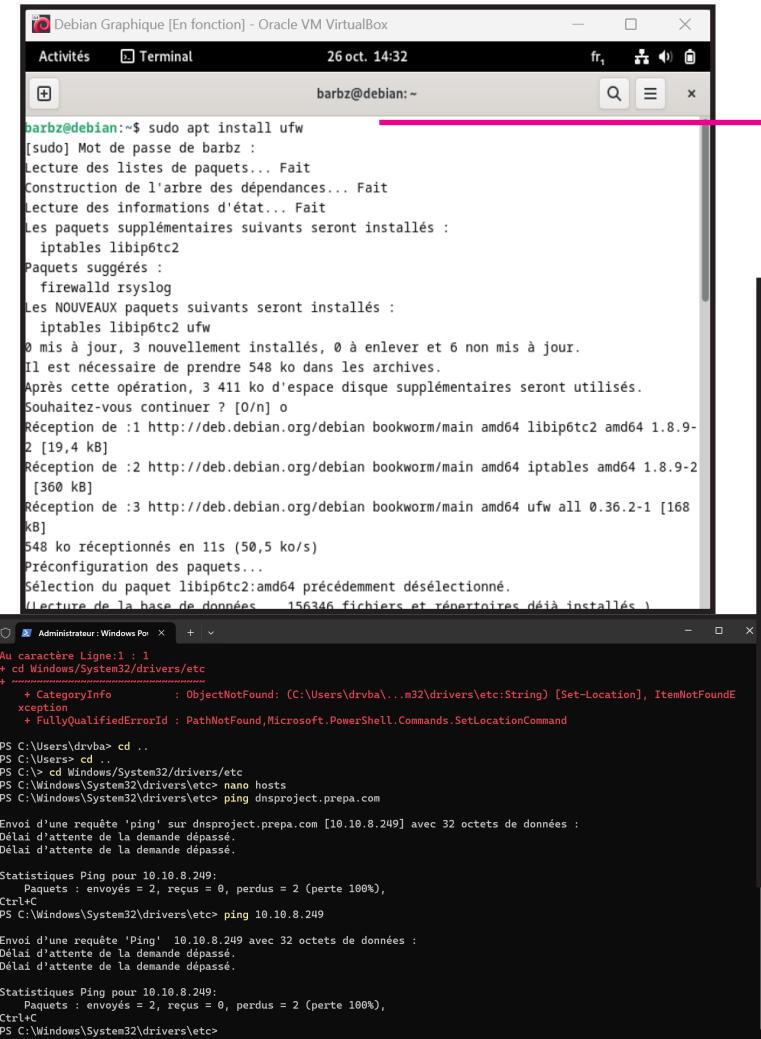
```
/etc/apache2/
|-- apache2.conf
|   '-- ports.conf
|-- mods-enabled
|   '-- *.Load
|   '-- *.conf
|-- conf-enabled
|   '-- *.conf
|-- sites-enabled
|   '-- *.conf
```

- apache2.conf is the main configuration file. It puts the pieces together by including all remaining configuration files when starting up the web server.
- ports.conf is always included from the main configuration file. It is used to determine the listening ports for incoming connections, and this file can be customized anytime.
- Configuration files in the mods-enabled/, conf-enabled/ and sites-enabled/ directories contain particular configuration snippets which manage modules, global configuration fragments, or virtual host configurations, respectively.
- They are activated by symlinking available configuration files from their respective *-available/ counterparts. These should be managed by using our helpers a2enmod, a2dismod, a2ensite, a2dissite, and a2enconf, a2disconf . See their respective man pages for detailed information.
- The binary is called apache2. Due to the use of environment variables, in the default configuration, apache2 needs to be started/stopped with /etc/init.d/apache2 or apache2ctl. **Calling /usr/bin/apache2 directly will not work** with the default configuration.

Comme vu précédemment, grâce à ce que l'on a fait dans le dossier etc/bind/ on a lié notre IP à ce nom de domaine.
Donc on a accès à notre page Apache2 en écrivant le nom de domaine, il est aussi possible de ping le serveur sur notre machine hôte avec le nom de domaine

METTRE EN PLACE UN PARE-FEU

07



```
barbz@debian:~$ sudo apt install ufw
[sudo] Mot de passe de barbz :
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 :
  iptables libip6tc2
Paquets suggérés :
  firewalld rsyslog
Les NOUVEAUX paquets suivants seront installés :
  iptables libip6tc2 ufw
0 mis à jour, 3 nouvellement installés, 0 à enlever et 6 non mis à jour.
Il est nécessaire de prendre 548 ko dans les archives.
Après cette opération, 3 411 ko d'espace disque supplémentaires seront utilisés.
Souhaitez-vous continuer ? [O/n] o
Réception de :1 http://deb.debian.org/debian bookworm/main amd64 libip6tc2 amd64 1.8.9-2 [19,4 kB]
Réception de :2 http://deb.debian.org/debian bookworm/main amd64 iptables amd64 1.8.9-2 [360 kB]
Réception de :3 http://deb.debian.org/debian bookworm/main amd64 ufw all 0.36.2-1 [168 kB]
548 ko réceptionnés en 11s (50,5 ko/s)
Préconfiguration des paquets...
 Sélection du paquet libip6tc2:amd64 précédemment désélectionné.
[... Lecture de la base de données... 156346 fichiers et répertoires déjà installés ]

Administrator: Windows Powershell
Au caractère Ligne:1 : 1
+ cd Windows\System32\drivers\etc
+
+ CategoryInfo          : ObjectNotFound: (C:\Users\drvba...m32\drivers\etc:String) [Set-Location], ItemNotFoundException
+ FullyQualifiedErrorId : PathNotFound,Microsoft.PowerShell.Commands.SetLocationCommand

PS C:\Users\drvba> cd ..
PS C:\Users> cd ..
PS C:\> cd Windows\System32\drivers\etc> nano hosts
PS C:\Windows\System32\drivers\etc> ping dnsproject.prepa.com

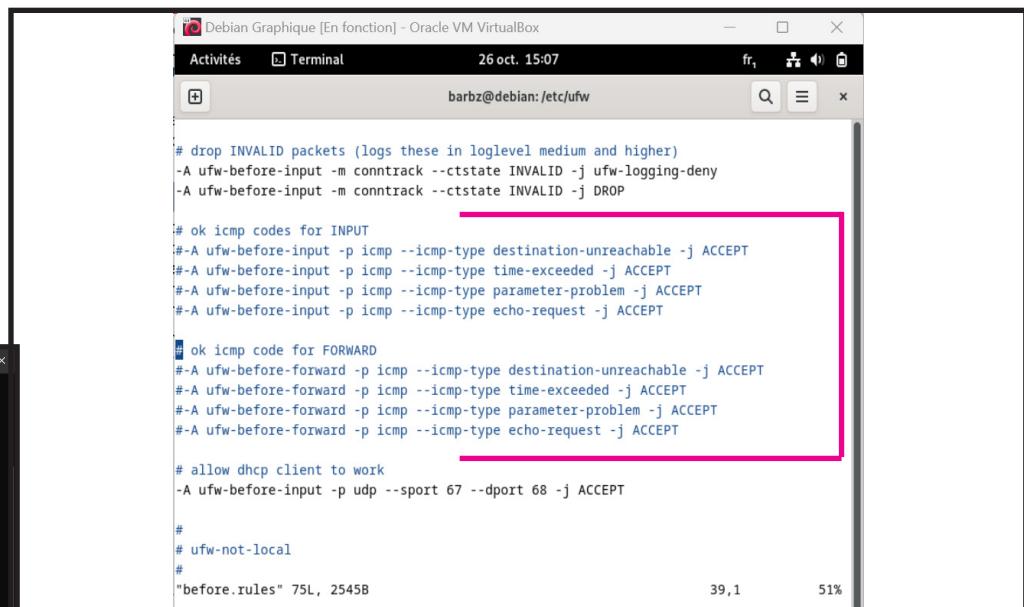
Envoi d'une requête 'ping' sur dnsproject.prepa.com [10.10.8.249] avec 32 octets de données :
Délai d'attente de la demande dépassé.
Délai d'attente de la demande dépassé.

Statistiques Ping pour 10.10.8.249:
 Paquets : envoyés = 2, reçus = 0, perdus = 2 (perte 100%),
Ctrl+C
PS C:\Windows\System32\drivers\etc> ping 10.10.8.249

Envoi d'une requête 'Ping' 10.10.8.249 avec 32 octets de données :
Délai d'attente de la demande dépassé.
Délai d'attente de la demande dépassé.

Statistiques Ping pour 10.10.8.249:
 Paquets : envoyés = 2, reçus = 0, perdus = 2 (perte 100%),
Ctrl+C
PS C:\Windows\System32\drivers\etc>
```

On va d'abord installer les paquets **ufw**, qui signifie «Uncomplicated Firewall», ce qui peut se traduire par «pare-feu simple» en français. C'est une interface utilisateur en ligne de commande pour **iptables**, le pare-feu netfilter de Linux. Une fois installé on l'active, pour le moment on peut toujours ping notre serveur. Pour empêcher les pings mes toujours avoir accès à la page web. Il faut commenter une partie du texte qui se trouve



```
# drop INVALID packets (logs these in loglevel medium and higher)
-A ufw-before-input -m conntrack --ctstate INVALID -j ufw-logging-deny
-A ufw-before-input -m conntrack --ctstate INVALID -j DROP

# ok icmp codes for INPUT
#-A ufw-before-input -p icmp --icmp-type destination-unreachable -j ACCEPT
#-A ufw-before-input -p icmp --icmp-type time-exceeded -j ACCEPT
#-A ufw-before-input -p icmp --icmp-type parameter-problem -j ACCEPT
#-A ufw-before-input -p icmp --icmp-type echo-request -j ACCEPT

# ok icmp code for FORWARD
#-A ufw-before-forward -p icmp --icmp-type destination-unreachable -j ACCEPT
#-A ufw-before-forward -p icmp --icmp-type time-exceeded -j ACCEPT
#-A ufw-before-forward -p icmp --icmp-type parameter-problem -j ACCEPT
#-A ufw-before-forward -p icmp --icmp-type echo-request -j ACCEPT

# allow dhcp client to work
-A ufw-before-input -p udp --sport 67 --dport 68 -j ACCEPT

#
# ufw-not-local
#
"before.rules" 75L, 2545B
```

dans le fichier **/etc/ufw/before.rules** qui nous permet de changer les règles de base. Ici j'ai commenté les lignes, mais on aurait pu remplacer le terme «ACCEPT» par «DROP».

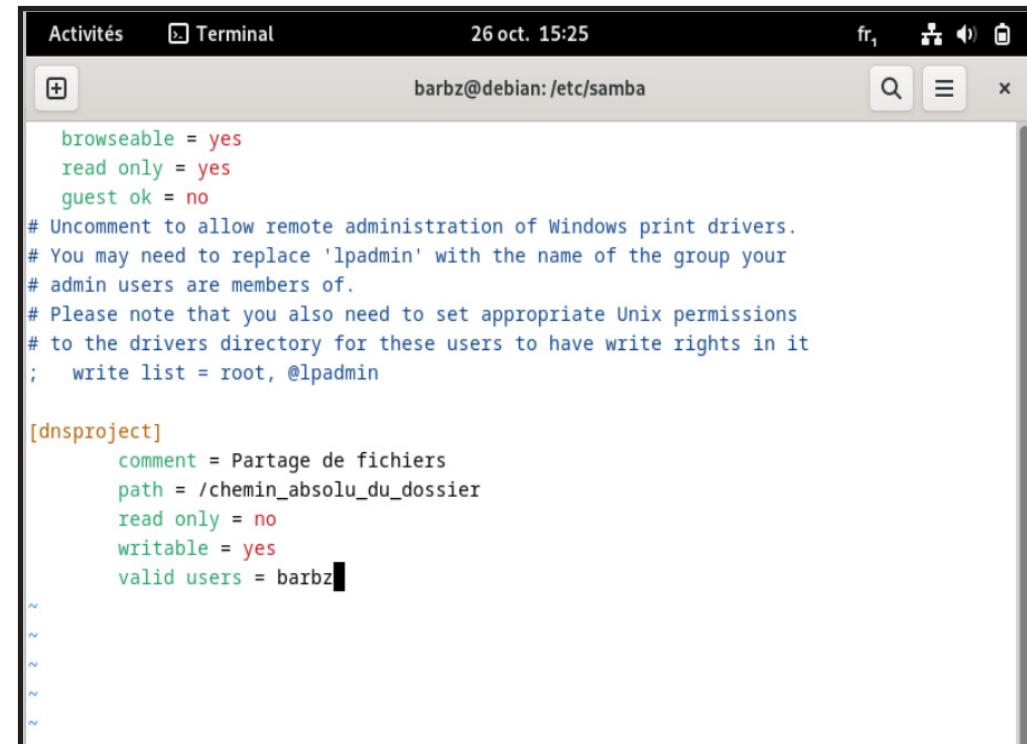
A ce moment là on ne pourra plus ping ni l'adresse IP, ni le nom de domaine, mais l'accès à la page web sera toujours possible.

Sans surprise, on à besoin de télécharger d'autres paquets pour pouvoir faire un dossier partagé entre notre VM et notre machine hôte.

Ici ce sera **samba** qui est un ensemble de logiciels open source qui permettent le partage de fichiers et d'imprimantes entre systèmes d'exploitation différents, en particulier entre des systèmes Windows et des systèmes basés sur Unix, tels que Linux. Samba permet de créer des serveurs de fichiers et d'impression compatibles avec les protocoles utilisés par les systèmes Windows, ce qui facilite le partage de ressources sur un réseau hétérogène.

Juste après l'installation il faut créer un dossier partagé dans l'emplacement de notre choix. Une fois notre dossier créé, on va ajouter les règles de partage dans le dossier /etc/samba.

La dernière chose à faire avant de restart samba et de voir si ça a marché, est d'autoriser les ports 139 et 445 pour notre pare-feu fraichement installé



```
browseable = yes
read only = yes
guest ok = no
# Uncomment to allow remote administration of Windows print drivers.
# You may need to replace 'lpadmin' with the name of the group your
# admin users are members of.
# Please note that you also need to set appropriate Unix permissions
# to the drivers directory for these users to have write rights in it
;   write list = root, @lpadmin

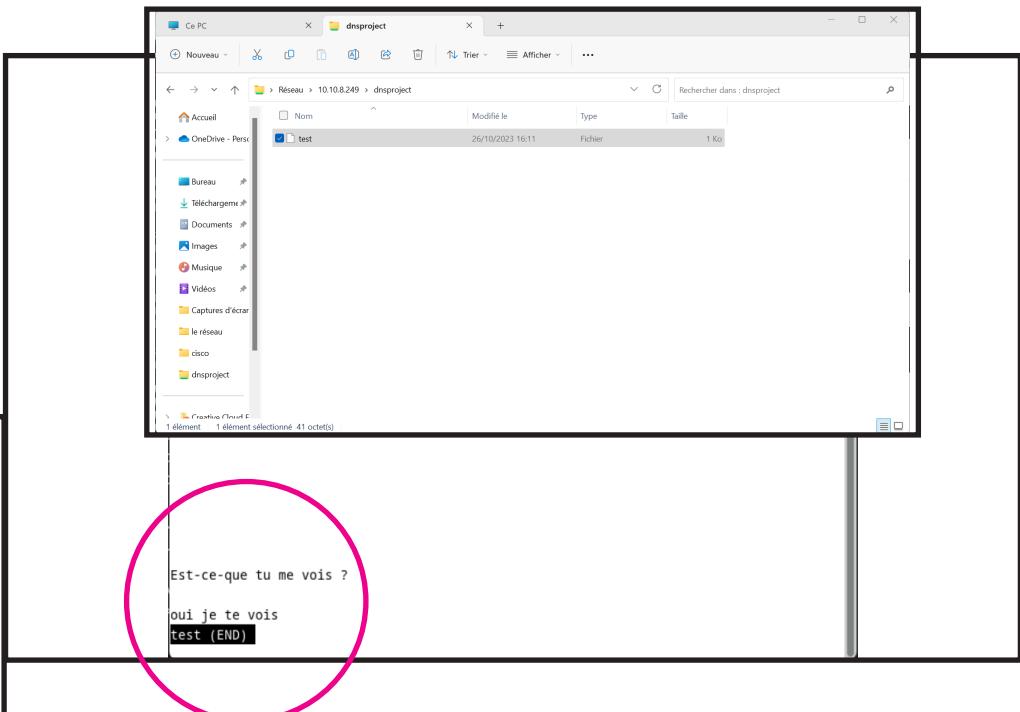
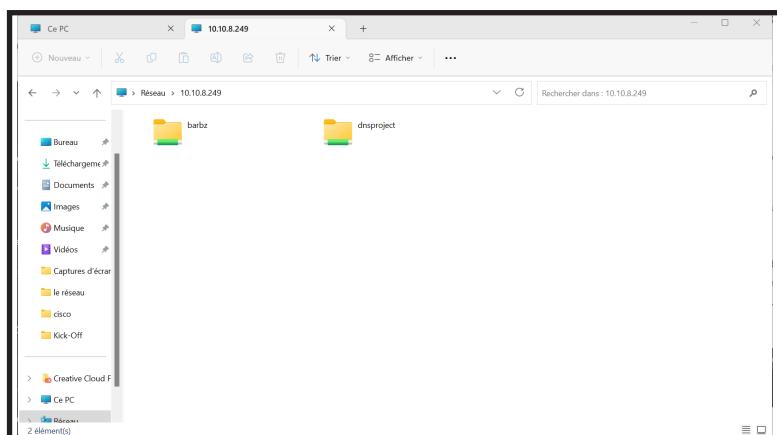
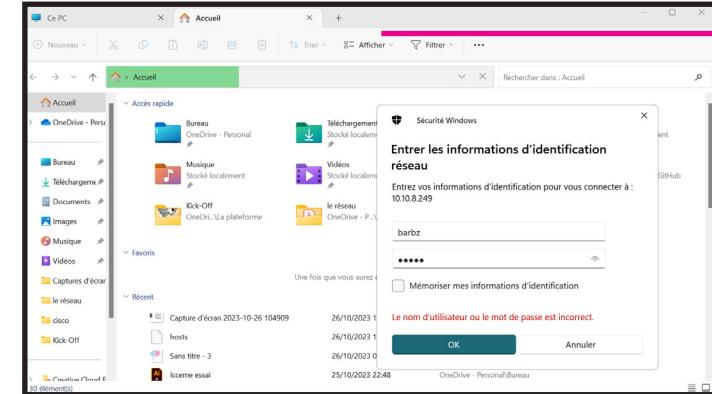
[dnsproject]
comment = Partage de fichiers
path = /chemin_absolu_du_dossier
read only = no
writable = yes
valid users = barbz
```

```
barbz@debian:/$ sudo ufw allow 139/tcp
Rule added
Rule added (v6)
barbz@debian:/$ sudo ufw allow 445/tcp
Rule added
Rule added (v6)
barbz@debian:/$
```

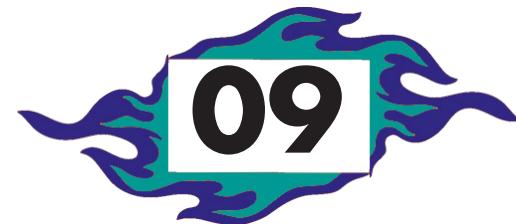
DOSSIER PARTAGÉ (BIS)

08

Dans mes capture d'écran on peut voir que je me connecte à l'utilisateur barbz, c'est exactement le même principe avec les utilisateurs Merry et Pippin !



Une fois que j'ai accès au dossier partagé, je peux créer un fichier .txt sur ma machine hôte et le retrouver dans ma VM. J'ai pu avoir une conversation super intéressante.



Dans cet exercice, on va devoir activer le HTTPS de notre serveur Apache !

Pour cela, on aura besoin de télécharger **OpenSSL** qui est une bibliothèque logicielle open source largement utilisée pour implémenter le protocole SSL/TLS (Secure Sockets Layer/Transport Layer Security) qui assure la sécurité des communications sur Internet.

Ensuite il va falloir enchaîner deux commandes qui vont permettre de créer deux fichiers : un **.crt** et **.key** (l'un étant le certificat et l'autre sa clé pour pouvoir y accéder)

1. `openssl genpkey -algorithm RSA -out moncertificat.key -aes256`
 2. `openssl rq -new -key moncertificat.key -x 509 -out moncertificat.crt`

Une fois ces deux commandes effectuées, plusieurs informations vont être demandées pour remplir le certificat et son «mot de passe»

Il faudra ajouter les informations

```

Debian Graphique [En fonction] - Oracle VM VirtualBox
Activités Terminal 27 oct. 13:31 fr_1
barbz@debian:/etc/apache2/sites-available

<VirtualHost *:80>
    ServerAdmin webmaster@dnsproject.prepa.com
    ServerName dnsproject.prepa.com
    DocumentRoot /var/www/html

    #ErrorLog ${APACHE_LOG_DIR}/error.log
    #CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>

<VirtualHost *:443>
    ServerName dnsproject.prepa.com
    DocumentRoot /var/www/html

    SSLEngine on
    SSLCertificateFile /etc/moncertificat.crt
    SSLCertificateKeyFile /etc/moncertificat.key
</VirtualHost>

```

The browser window shows the Apache2 Debian Default Page with the message "It works!". The address bar displays "https://dnsproject.prepa.com". A red circle highlights the "Non sécurisé" warning icon.

Il faudra ajouter le **VirtualHost 443** dans notre fichier de configuration avec les deux fichiers nouvellement créés (le .crt et le .key).

Et quand on ira sur la page web, on pourra voir que l'adresse à changer. On a désormais un **https** !

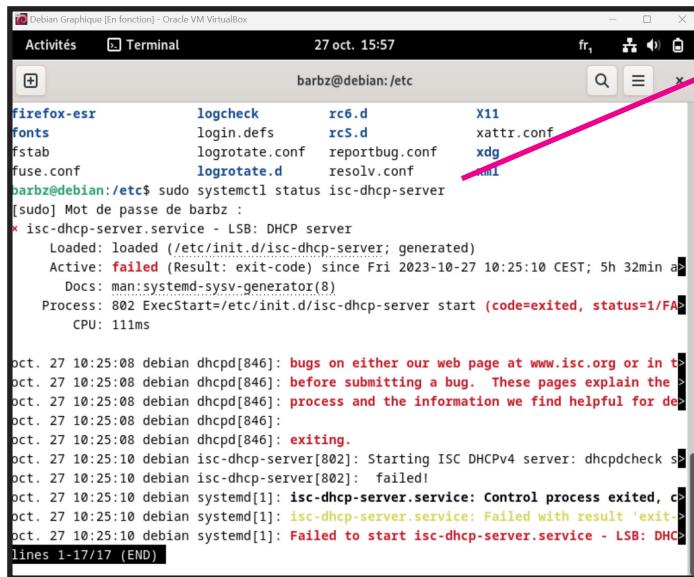
Pourquoi est ce que ça nous marque que c'est non sécurisé ?

Il s'avère que ce que nous avons fait est en réalité un certificat auto-signé. Etant donné qu'il n'est pas émis par une société spécialisée valide, il est considéré comme étant peu fiable et sûre. Généralement ils sont utilisés par les sociétés qui ont besoin d'un site interne et non d'un site public.

C'est pour cela que notre navigateur nous fait apparaître qu'il est non sécurisé

ALLER ENCORE PLUS LOIN : LE DHCP

10



```
barbz@debian:~$ sudo systemctl status isc-dhcp-server
[sudo] Mot de passe de barbz :
* isc-dhcp-server.service - LSB: DHCP server
  Loaded: loaded (/etc/init.d/isc-dhcp-server; generated)
  Active: failed (Result: exit-code) since Fri 2023-10-27 10:25:10 CEST; 5h 32min ago
    Docs: man:systemd-sysv-generator(8)
   Process: 802 ExecStart=/etc/init.d/isc-dhcp-server start (code=exited, status=1/FAIL)
      CPU: 11ms

oct. 27 10:25:08 debian dhcpcd[846]: bugs on either our web page at www.isc.org or in t>
oct. 27 10:25:08 debian dhcpcd[846]: before submitting a bug. These pages explain the >
oct. 27 10:25:08 debian dhcpcd[846]: process and the information we find helpful for de>
oct. 27 10:25:08 debian dhcpcd[846]:
oct. 27 10:25:08 debian dhcpcd[846]: exiting.
oct. 27 10:25:10 debian isc-dhcp-server[802]: Starting ISC DHCPV4 server: dhcpcdcheck s>
oct. 27 10:25:10 debian isc-dhcp-server[802]: failed!
oct. 27 10:25:10 debian systemd[1]: isc-dhcp-server.service: Control process exited, c>
oct. 27 10:25:10 debian systemd[1]: isc-dhcp-server.service: Failed with result 'exit>
oct. 27 10:25:10 debian systemd[1]: Failed to start isc-dhcp-server.service - LSB: DH<
lines 1-17/17 (END)
```



```
# range 10.17.224.10 10.17.224.250;
#
# pool {
#   deny members of "foo";
#   range 10.0.29.10 10.0.29.230;
# }
#
# Configuration du serveur DHCP
option domain-name "disproject.prepa.com";
option domain-name-servers 10.10.8.249;

subnet 10.10.8.0 netmask 255.255.255.0 {
  range 10.10.8.100 10.10.8.200;
  option routers 10.10.8.249;
  option subnet-mask 255.255.255.0;
  option broadcast-address 10.10.8.255;

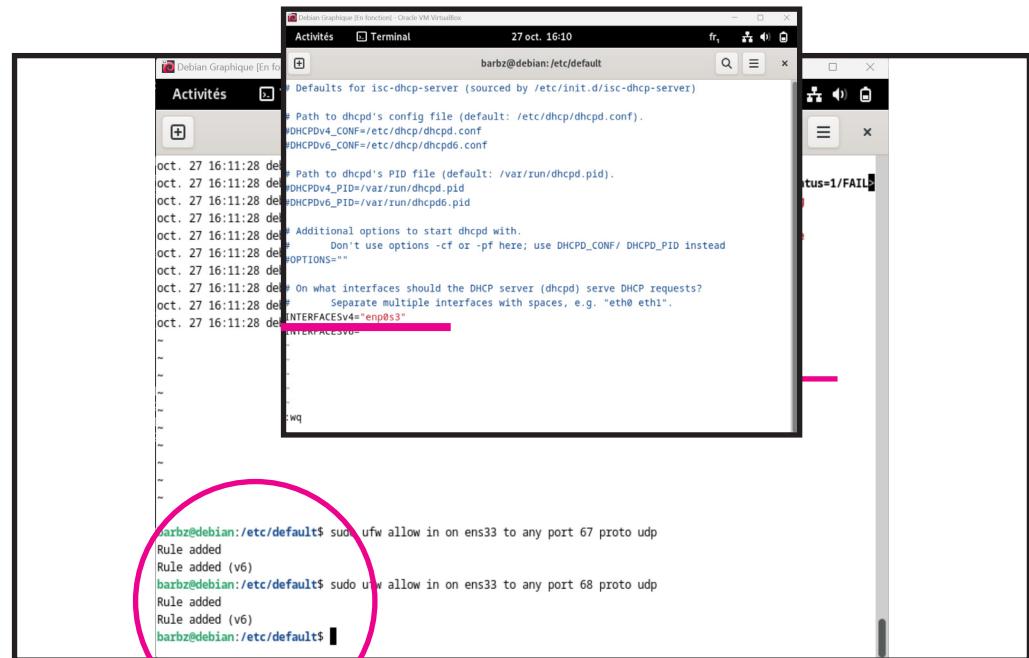
  default-lease-time 600;
  max-lease-time 7200;

  deny unknown-clients;
}

dhcpd.conf: 123L, 3866B
```

Pour notre serveur DHCP, il va falloir télécharger les paquets **isc-dhcp-server**.

Une fois installé il faut le paramétrier; ca se passera dans le dossier **dhcp.conf**



```
Defaults for isc-dhcp-server (sourced by /etc/init.d/isc-dhcp-server)

# Path to dhcpcd's config file (default: /etc/dhcpcd.conf).
#DHCPD_V4_CONF=/etc/dhcp/dhcpd.conf
#DHCPD_V6_CONF=/etc/dhcp/dhcpd6.conf

oct. 27 16:11:28 debian dhcpcd[846]: # Path to dhcpcd's PID file (default: /var/run/dhcpcd.pid).
oct. 27 16:11:28 debian dhcpcd[846]: #DHCPD_V4_PID=/var/run/dhcpcd.pid
oct. 27 16:11:28 debian dhcpcd[846]: #DHCPD_V6_PID=/var/run/dhcpcd6.pid
oct. 27 16:11:28 debian dhcpcd[846]: # Additional options to start dhcpcd with.
oct. 27 16:11:28 debian dhcpcd[846]: # Don't use options -cf or -pf here; use DHCPD_CONF/ DHCPD_PID instead
#OPTIONS=""

oct. 27 16:11:28 debian dhcpcd[846]: # On what interfaces should the DHCP server (dhcpcd) serve DHCP requests?
oct. 27 16:11:28 debian dhcpcd[846]: # Separate multiple interfaces with spaces, e.g. "eth0 eth1".
#INTERFACESV4="enp0s3"
#INTERFACESV6=""

:wq
```

Il ne faut pas oublier d'ouvrir les ports ufw 67 et 68, et de stopper NetworkManager (qui semblerait être l'entité qui gère le DHCP de notre machine virtuelle) pour éviter tout conflit !

Il faut aussi préciser notre interfacev4 qui est propre à chaque carte réseau. La mienne est enp0s3.

Maintenant que tout est configuré, on démarre le serveur DHCP avec la commande

systemctl start isc-dhcp-server.service

MERCI