

# HOLODECK

## Installation de la VM serveur

```
VM serveur x

Debian GNU/Linux 12 IKARUS tty1

IKARUS login: wass-sama
Password:
Linux IKARUS 6.1.0-38-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.147-1 (2025-08-02) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
wass-sama@IKARUS:~$ sudo
-bash: sudo : commande introuvable
wass-sama@IKARUS:~$ su -
Mot de passe :
root@IKARUS:~#
```

Avant toute chose faire en sorte que l'utilisateur soi dans le groupe sudo pour pas retaper sudo a chaque fois. Conformément aux ordres de StarFleet. Ensuite se déconnecter et se reconnecter.

```
root@IKARUS:~# apt install sudo
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les NOUVEAUX paquets suivants seront installés :
  sudo
0 mis à jour, 1 nouvellement installés, 0 à enlever et 0 non mis à jour.
Il est nécessaire de prendre 1 890 ko dans les archives.
Après cette opération, 6 199 ko d'espace disque supplémentaires seront utilisés.
Réception de :1 http://security.debian.org/debian-security bookworm-security/main amd64 sudo amd64 1.9.13p3-1+deb12u2 [1 890 kB]
1 890 ko réceptionnés en 0s (8 313 ko/s)
Sélection du paquet sudo précédemment désélectionné.
(Lecture de la base de données... 38680 fichiers et répertoires déjà installés.)
Préparation du dépaquetage de .../sudo_1.9.13p3-1+deb12u2_amd64.deb ...
Dépaquetage de sudo (1.9.13p3-1+deb12u2) ...
Paramétrage de sudo (1.9.13p3-1+deb12u2) ...
Traitement des actions différées (« triggers ») pour man-db (2.11.2-2) ...
Traitement des actions différées (« triggers ») pour libc-bin (2.36-9+deb12u10) ...
root@IKARUS:~# adduser wass-sama sudo
Ajout de l'utilisateur « wass-sama » au groupe « sudo » ...
Fait.
root@IKARUS:~#
```

Faire la commande Usermod -aG sudo wass-sama

Ensuite groups

```
wass-sama@IKARUS:~$ groups
wass-sama cdrom floppy sudo audio dip video plugdev users netdev
wass-sama@IKARUS:~$ sudo whoami
[sudo] Mot de passe de wass-sama :
root
wass-sama@IKARUS:~$
```

1/ Installer DNS :

Sudo apt update

Sudo apt install bind9 bind9-utils bind9-dnsutils -y

## 2. Configurer BIND9 comme serveur DNS local

sudo nano /etc/bind/named.conf.options

Ajoutez/éditez :

options {

directory "/var/cache/bind";

recursion yes; # autoriser la résolution récursive pour le LAN

allow-query { any; }; # permettre aux clients LAN de faire des requêtes

listen-on { 192.168.56.1; }; # adresse LAN de votre VM serveur

forwarders {

1.1.1.1; 8.8.8.8; # serveurs DNS publics (Cloudflare, Google)

};

};

## 3. Créer la zone starfleet.lan

sudo nano /etc/bind/named.conf.local

Ajoutez :

zone "starfleet.lan" {

type master;

file "/etc/bind/db.starfleet.lan";

};

## 4. Fichier de zone DNS

sudo nano /etc/bind/db.starfleet.lan

Exemple minimal :

\$TTL 604800

@ IN SOA ns.starfleet.lan. admin.starfleet.lan. (

2025091001 ; Serial

604800 ; Refresh

86400 ; Retry

2419200 ; Expire

604800 ) ; Negative Cache TTL

;

@ IN NS ns.starfleet.lan.

; Serveur DNS

ns IN A 192.168.56.1

; Sites web

www8 IN A 192.168.56.1

www7 IN A 192.168.56.1

php IN A 192.168.56.1

admin IN A 192.168.56.1

vscore IN A 192.168.56.1

## 5. Vérifier et relancer BIND9

sudo named-checkconf

sudo named-checkzone starfleet.lan /etc/bind/db.starfleet.lan

```
root@IKARUS:~# sudo nano /etc/bind/named.conf.local

root@IKARUS:~# sudo nano /etc/bind/db.starfleet.lan
root@IKARUS:~# udo named-checkconf
-bash: udo : commande introuvable
root@IKARUS:~# sudo named-checkconf
/etc/bind/named.conf.options:28: 'options' redefined near 'options'
root@IKARUS:~# sudo named-checkzone starfleet.lan /etc/bind/db.starfleet.lan
zone starfleet.lan/IN: loaded serial 2025091001
OK
root@IKARUS:~# |
```

Redémarrez :

sudo systemctl restart bind9

sudo systemctl enable bind9

## 6. Configurer le client (VM Debian GUI)

Sur la **VM cliente**, éditez /etc/resolv.conf (ou configurez via DHCP plus tard) :

nameserver 192.168.56.1

search starfleet.lan

Faire le Teste:

ping www8.starfleet.lan

ping php.starfleet.lan

**/2/ ensuite mettre en place d'un pare-feu pour autoriser uniquement les ports requis.**

Utilisez les commande : `sudo apt update` ensuite `sudo apt install ufw`

Avec **UFW** (Uncomplicated Firewall), c'est le plus simple. Pour un pare feu.

```
root@IKARUS:/home/wass-sama# apt install ufw
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 libnetfilter-contrack3 libnfnetlink0
Paquets suggérés :
  firewalld rsyslog
Les NOUVEAUX paquets suivants seront installés :
  iptables libip6tc2 libnetfilter-contrack3 libnfnetlink0 ufw
0 mis à jour, 5 nouvellement installés, 0 à enlever et 0 non mis à jour.
Il est nécessaire de prendre 603 ko dans les archives.
Après cette opération, 3 606 ko d'espace disque supplémentaires seront utilisés.
Souhaitez-vous continuer ? [O/n] o
```

- Mise en place d'un pare-feu pour autoriser uniquement les ports requis.

`sudo ufw allow 22/tcp`

`sudo ufw allow 80/tcp`

`sudo ufw allow 443/tcp`

```

root@IKARUS:/home/wass-sama# # Autoriser SSH
sudo ufw allow 22/tcp

# Autoriser HTTP
sudo ufw allow 80/tcp

# Autoriser HTTPS
sudo ufw allow 443/tcp
Skipping adding existing rule
Skipping adding existing rule (v6)
Skipping adding existing rule
Skipping adding existing rule (v6)
Skipping adding existing rule
Skipping adding existing rule (v6)

```

Il faut bien activer est ouvrir le port 22 sinon le SSH s'arrete.

```

root@IKARUS:/home/wass-sama# sudo ufw enable
Command may disrupt existing ssh connections. Proceed with operation (y|n)? y
Firewall is active and enabled on system startup
root@IKARUS:/home/wass-sama#

```

sudo ufw status verbose.

Cette commande permet de vérifier les ports s'ils ont bien ouvert ou fermé.

```

root@IKARUS:/home/wass-sama# sudo ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing), disabled (routed)
New profiles: skip

To Action From
--
22/tcp ALLOW IN Anywhere
80/tcp ALLOW IN Anywhere
443/tcp ALLOW IN Anywhere
22/tcp (v6) ALLOW IN Anywhere (v6)
80/tcp (v6) ALLOW IN Anywhere (v6)
443/tcp (v6) ALLOW IN Anywhere (v6)

```

On peut penser que ce sont des doublons mais non :

- 22/tcp - 80/tcp - 443/tcp: ce sont les règles pour IPv4
- 22/tcp (v6), 80/tcp (v6), 443/tcp (v6) → les mêmes règles appliquées pour IPv6

C'est UFW qui crée automatiquement une règle pour IPv4 et une pour IPv6.

Ensuite faire en sorte que le serveur Web doit être Nginx et en HTTPS.

```
root@IKARUS:/home/wass-sama# apt 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 :
```

Tester si nginx fonctionne.

```
wass-sama@IKARUS:~$ sudo systemctl restart nginx
wass-sama@IKARUS:~$ sudo 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 Wed 2025-09-03 16:13:24 CEST; 10s ago
     Docs: man:nginx(8)
  Process: 3920 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
  Process: 3921 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
 Main PID: 3922 (nginx)
    Tasks: 3 (limit: 2273)
   Memory: 2.3M
      CPU: 48ms
   CGroup: /system.slice/nginx.service
           └─3922 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
             └─3923 "nginx: worker process"
               └─3924 "nginx: worker process"
```

Mettre son utilisateur dans sudoers.

```
wass-sama@IKARUS: ~
GNU nano 7.2 /etc/sudoers

# Ditto for GPG agent
#Defaults:%sudo env_keep += "GPG_AGENT_INFO"

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root    ALL=(ALL:ALL) ALL
wass-sama    ALL=(ALL:ALL) ALL
# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "@include" directives:

@includedir /etc/sudoers.d
```

Installer dernière version Firefox sans sudo:

```
root@IKARUS:/home/wass-sama# apt install firefox-esr -y
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
firefox-esr est déjà la version la plus récente (128.14.0esr-1~deb12u1).
0 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à jour.
root@IKARUS:/home/wass-sama#
```

1/Ensuite installer nginx en cherchant la bonne commande

```
root@IKARUS:/home/wass-sama# apt install curl gnupg2 ca-certificates lsb-release
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
ca-certificates est déjà la version la plus récente (20230311+deb12u1).
```

2/Ajoutez ensuite la clé officielle de Nginx pour vérifier l'entièreté des paquets.

```
root@IKARUS:/home/wass-sama# curl -fsSL https://nginx.org/keys/nginx_signing.key | sudo gpg --dearmor -o /usr/share/keyrings/nginx-archive-keyring.gpg
root@IKARUS:/home/wass-sama# apt upgrade
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.
root@IKARUS:/home/wass-sama#
```

Ensuite vérifier s'il marche:

```
root@IKARUS:~# 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 Tue 2025-09-09 13:51:30 CEST; 8min ago
     Docs: man:nginx(8)
```

Faire : `sudo ufw allow 'Nginx Full'`

Ça ouvre automatiquement les port **(HTTP)** et **(HTTPS)**.

```
root@IKARUS:~# sudo ufw allow 'Nginx Full'
Rule added
Rule added (v6)
```

