

Outils

Version 1.0.0

Niveau requis : 6/7



Initialisation Serveur VPS OVH-Debian 11- iRedMail – Drupal 9

Sommaire

I.	PREAMBULE.....	3
I.I.	OBJET.....	3
I.II.	PRE-REQUIS	3
I.III.	VERSIONS DU DOCUMENT	3
I.IV.	DOCUMENTS DE REFERENCE	3
II.	OVH – INITIALISATION DE DEBIAN 11.....	3
II.I.	RECOMMANDATION DU SERVEUR VPS	3
II.I.1	VPS : 2Go de RAM/ Disque SSD 20 Go, 1 Coeur	3
II.I.2	Choix de l'OS sur le serveur VPS : Debian 11.....	4
II.I.3	Réception du mail de connexion pour se connecter Putty sous Windows	4
II.II.	CONNEXION AVEC PUTTY.....	5
II.III.	ASTUCE, REDIRECTION DE TUNNEL DE PORT POUR MARIADB	6
II.IV.	CONNEXION	7
II.V.	CONFIGURATION DNS OVH POUR REDIRIGER VERS VOTRE FUTUR SERVEUR DE MAIL SUR LA MACHINE VPS	7
II.V.1	Configuration en mode Textuel du DNS (Attention anonymisé).....	7
II.VI.	MISE A NIVEAU DE DEBIAN DERNIERE VERSION	8
II.VII.	CONFIGURATION DU NOM DU HOSTNAME DE LA MACHINE.....	9
III.	INSTALLATION DU SERVEUR DE MAIL SUR DEBIAN 11 AVEC IREDMAIL	9
III.I.	INSTALLATION DE IREDMAIL 1.6.0 (DERNIERE VERSION ACTUELLE).....	9
III.II.	SECURISER LE SERVEUR DE MAIL AVEC UN CERTIFICAT RECONNU OFFICIEL GRATUIT « LET'S ENCRYPT TLS CERTIFICATE »	14
III.II.1	Obtenir l'outil certbot.....	14
III.II.2	Vérifier la bonne configuration	17
IV.	MISE DES SERVEUR AVEC LE CERTIFICAT OFFICIEL LET'S ENCRYPT	17
IV.I.	SITE INTERNET.....	17
IV.II.	SERVEUR DE MAIL	19
V.	MISE D'ESPACE DE SWAP	20
VI.	MISE D'UN SERVEUR DRUPAL 9 SUR LE SERVEUR DEBIAN	21
VI.I.	BASE DE DONNEES	21
VI.II.	SERVEUR NGINX.....	22
VII.	FIN DU DOCUMENT	31

I. Préambule

I.I. *Objet*

L'objet de ce document est de présenter l'installation chez OVH d'un Serveur Privé Virtuel (VPS) avec comme installation un OS Debian 11, Un Serveur de Mail géré par la solution gratuite iRedMail ainsi que la mise en place d'un site CMS Drupal 9.

Il faut compter pour une personne avertie une journée de travail.

I.II. *Pré-requis*

Avoir de solide connaissance dans le réseau, les parfeux, la configuration réseau des protocoles DNS ainsi qu'être à l'aise dans la base de données MariaDB et autonome dans l'installation d'un Drupal 9.

I.III. *Versions du document*

Version	Date	Auteur	Description
1.0.0	24/07/2022	Péquignat.eu	Création du document

I.IV. *Documents de référence*

#	Document	Version	Auteur(s)
[R1]	https://www.linuxbabe.com/mail-server/email-server-debian-11-iredmail	3 mai 2022	Xiao Guoan

II. OVH – Initialisation de Debian 11

II.I. *Recommandation du serveur VPS*

II.I.1 **VPS : 2Go de RAM/ Disque SSD 20 Go, 1 Coeur**

Mon serveur personnellement sert uniquement de site vitrine, peu fréquenté et de blog. Aussi j'ai dessus mis un serveur de mail, mais les performances pour mon utilisation ne sont pas cruciales. Pour une utilisation, plus intensive, il est recommandé de choisir plutôt à minima 3GO de RAM.

II.I.2 Choix de l'OS sur le serveur VPS : Debian 11

J'ai fait le choix d'avoir comme serveur vitrine du Drupal 9, aussi les recommandations actuelles de Drupal 9 nécessite d'avoir PHP 7.4 sur sa machine. D'où le passage à Debian 11 à la place de 10 depuis une installation d'usine d'OVH.



II.I.3 Réception du mail de connexion pour se connecter Putty sous Windows

Bonjour,

Votre VPS vient d'être installé sous le système d'exploitation / distribution Debian 11

PARAMETRES D'ACCES :

L'adresse IPv4 du VPS est : XXX.XXX.XX.XX

Le nom du VPS est : XXXXXX.ovh.net

Le compte administrateur suivant a été configuré sur le VPS :

Nom d'utilisateur : <user>

Mot de passe : <pass>

POUR BIEN DEMARRER:

Si vous vous connectez pour la première fois à un VPS, nous vous recommandons de consulter le guide suivant :

<https://docs.ovh.com/fr/vps/debuter-avec-vps/>

GESTION, FACTURATION, ASSISTANCE

Vous pouvez gérer votre VPS depuis votre espace client web à l'adresse suivante :

<https://www.ovh.com/manager/>

OBTENIR DE L'AIDE:

Pour vous accompagner dans la prise en main de votre VPS, nous mettons à votre disposition de nombreux guides d'utilisation :

<https://www.ovh.com/fr/support/knowledge/>

D'autre part, une importante communauté d'utilisateurs est accessible via notre forum et nos mailing-listes :

<https://www.ovh.com/fr/support/>

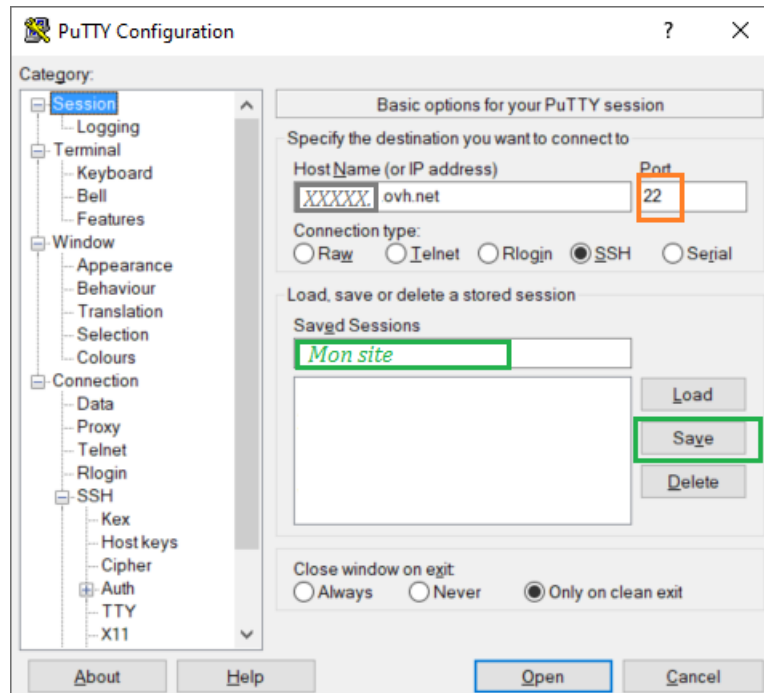
Merci de votre confiance,

L'équipe OVH

Je vous conseil de vous connecter dessus avec Putty.

Le port par défaut est le 22, à changer en suivant les instructions énoncées dans la vidéo présente sous <https://docs.ovh.com/fr/vps/debuter-avec-vps/>

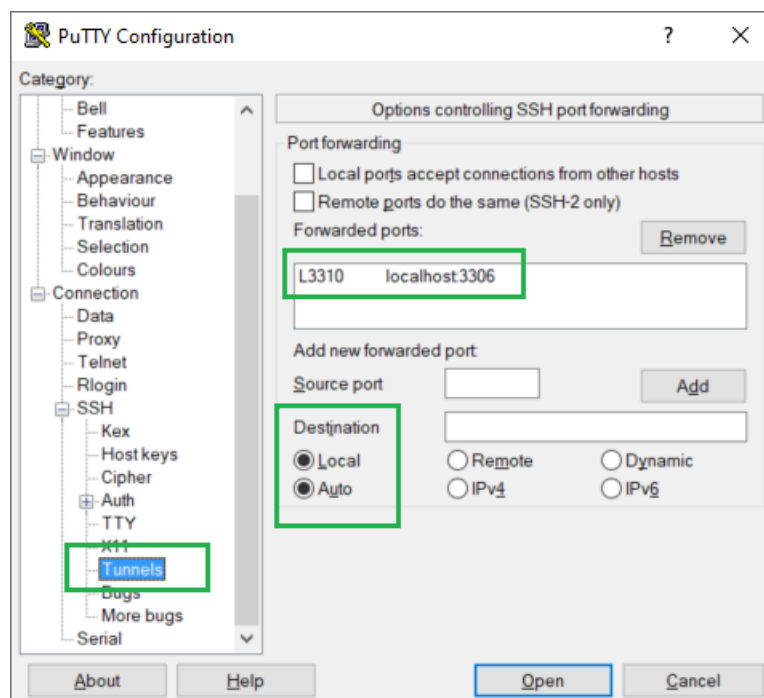
II.II. *Connexion avec Putty*



II.III. Astuce, redirection de Tunnel de port pour MariaDB

Une astuce permet de rediriger avec Putty le port local à votre machine par exemple de 3310 vers le port en local de MariaDB qui est par défaut 3306.

Pour cela aller dans



Après avoir fait cette manipulation, sauvegarder de nouveau la session de Putty pour conserver le changement. Cela vous sera utile pour initialiser à distance la base de données MariaDB avec comme client HeidiSQL.

II.IV. Connexion

Nous supposons que vous arrivez à vous connecter avec putty :

```
Linux mail.pequignat.eu 5.10.0-16-cloud-amd64 #1 SMP Debian 5.10.127-1 (2022-06-30) 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.

Last login: Sun Jul 24 08:09:01 2022 from 91.175.96.240
<user>@mail:~$
```

II.V. Configuration DNS OVH pour rediriger vers votre futur serveur de mail sur la machine VPS

II.V.1 Configuration en mode Textuel du DNS (Attention anonymisé)

```
$TTL 86400
@      IN SOA dns11.ovh.net. tech.ovh.net. (2022072404 86400 3600 3600000 86400)
      IN NS      dns11.ovh.net.
      IN NS      ns11.ovh.net.
      IN MX      1 mail.pequignat.eu.
      IN MX      100 mail.pequignat.eu.
      IN MX      5 mail.pequignat.eu.
      IN MX      4 mail.pequignat.eu.
      IN A       XXX.XXX.XX.XX
      IN AAAA    NNNN : NNNNN : NNNN : NNNN : NNNN : NNNN : NNNN : NNNN
```


Accepter le téléchargement si demandé.

II.VII. Configuration du nom du hostname de la machine

Ici dans cet exemple, j'ai mis mon propre serveur de mail soit mail.pequignat.eu. A vous de la configurer avec vos propres données.

```
sudo hostnamectl set-hostname mail.pequignat.eu
```

Edition du fichier de texte /etc/hosts

```
sudo nano /etc/hosts
```

Editer la première ligne avec :

```
127.0.0.1 mail.pequignat.eu localhost
```

Pour sauvegarder, faire CTRL+O, puis ENTER pour confirmer et pour fermer CTRL+X

Pour voir les changements :

```
hostname -f
```

III. Installation du serveur de Mail sur Debian 11 avec iRedMail

III.I. Installation de iRedMail 1.6.0 (dernière version actuelle)

```
wget https://github.com/iredmail/iRedMail/archive/1.6.0.tar.gz
```

Extraire :

```
tar xvf 1.6.0.tar.gz
```

Aller dans le répertoire

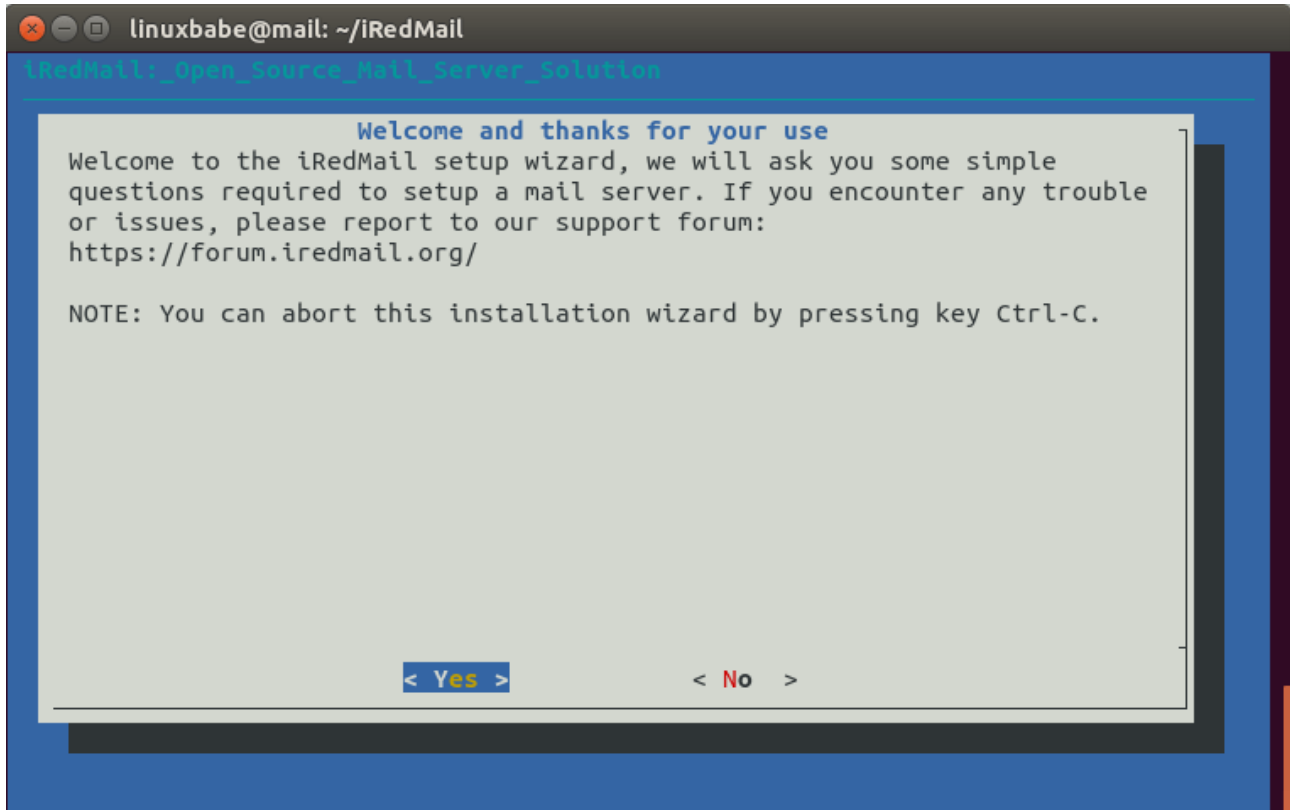
```
cd iRedMail-1.6.0/
```

Rendre exécutable l'installeur :

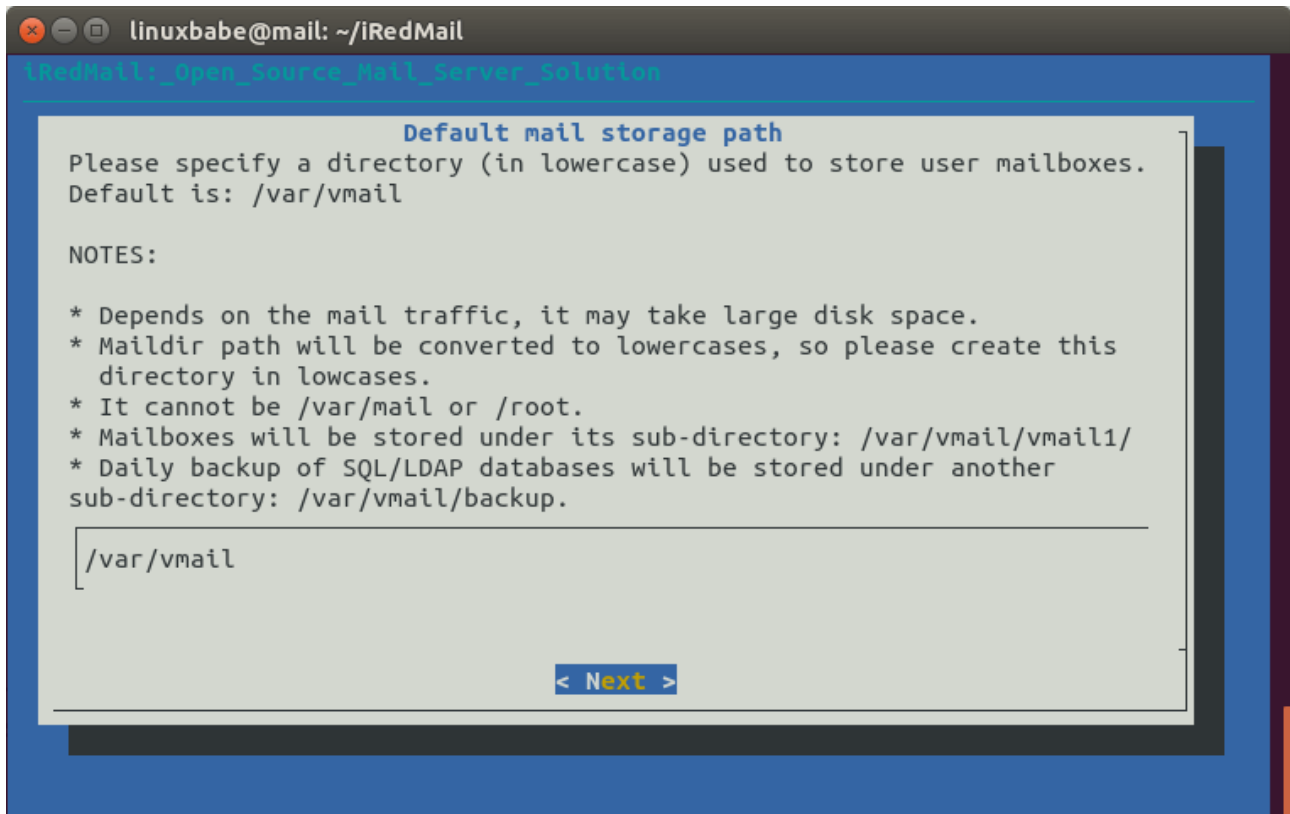
```
chmod +x iRedMail.sh
```

Lancer l'installation :

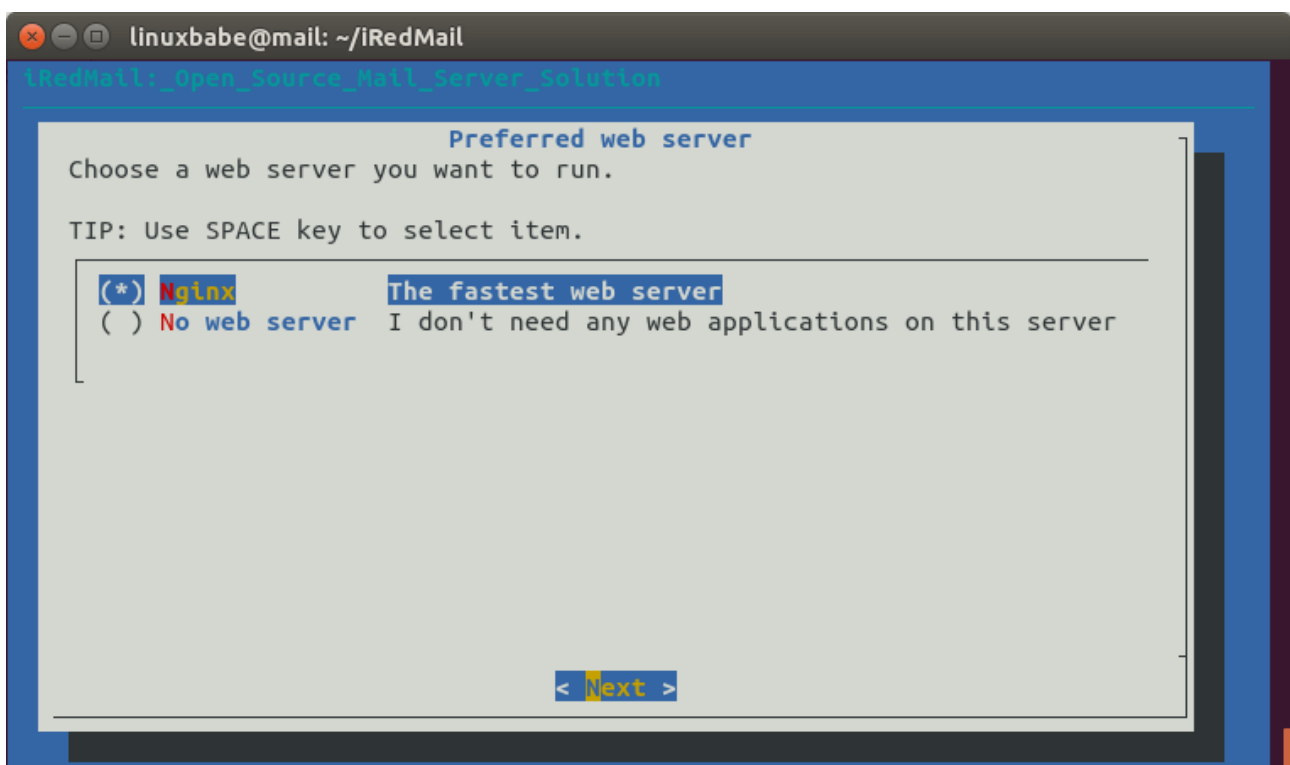
```
sudo bash iRedMail.sh
```



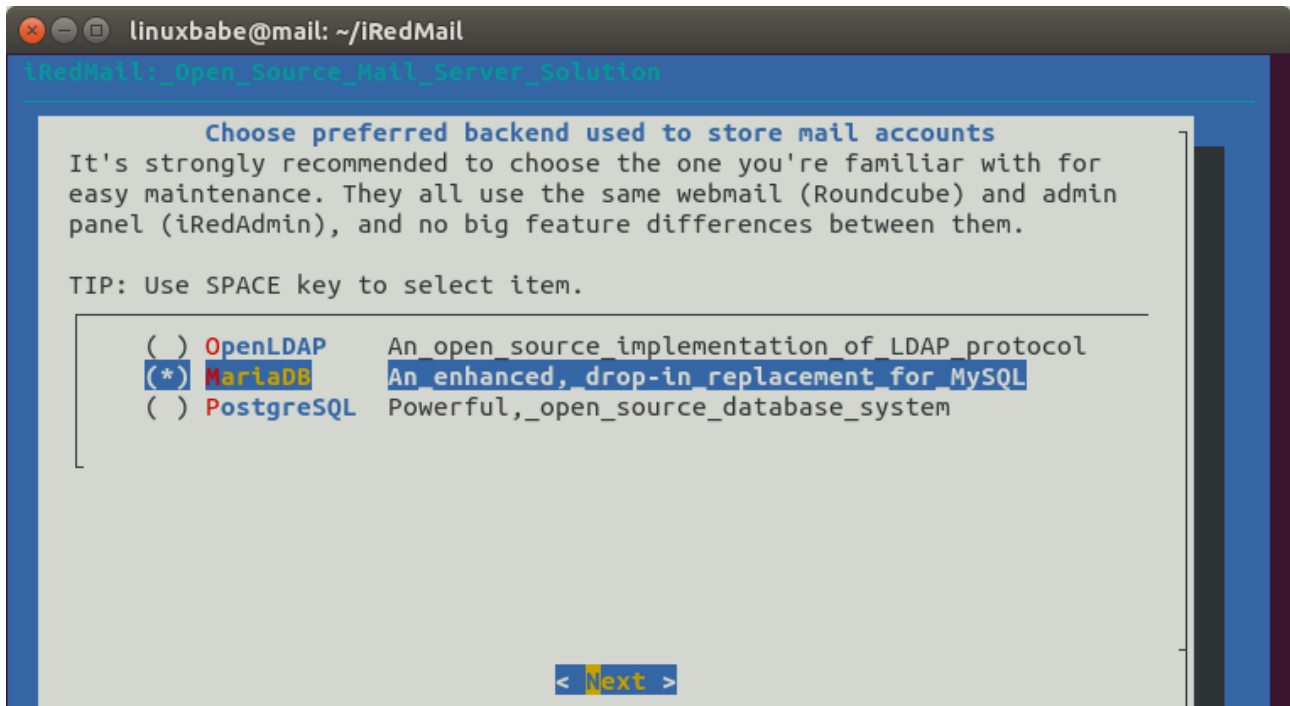
Cliquer sur Enter avec le focus sur le Yes



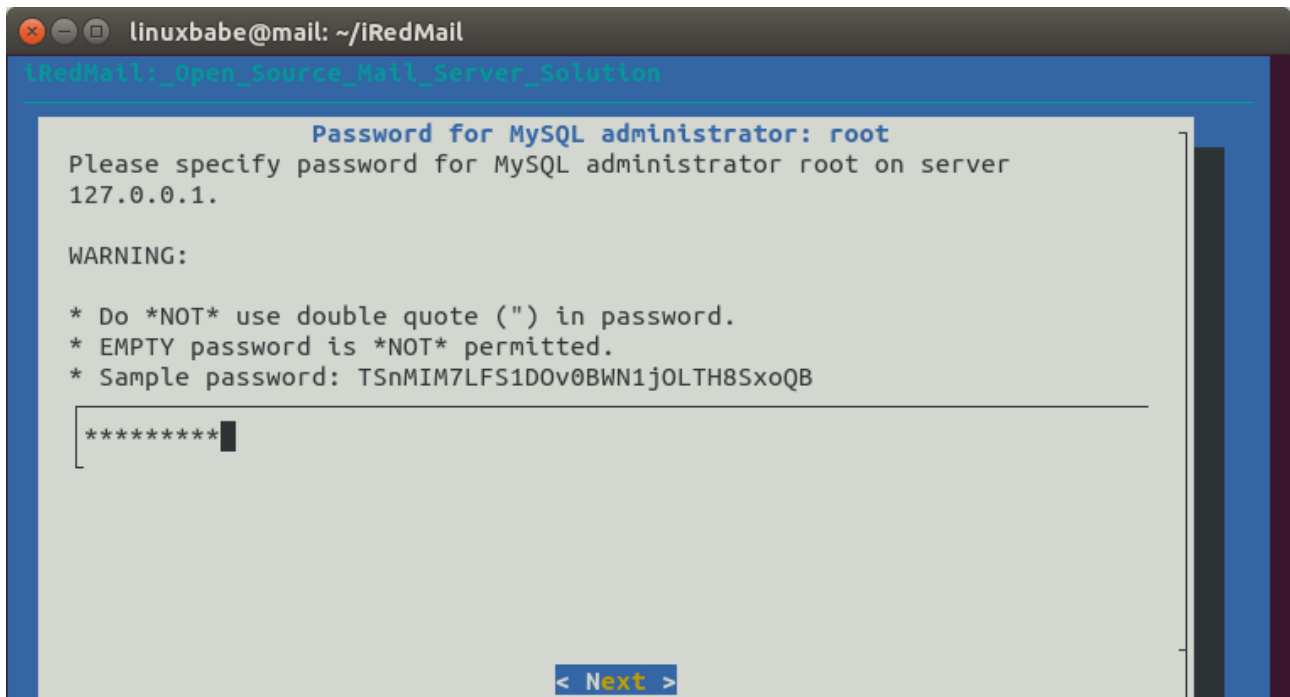
Laissez le choix par défaut, cliquer sur Enter



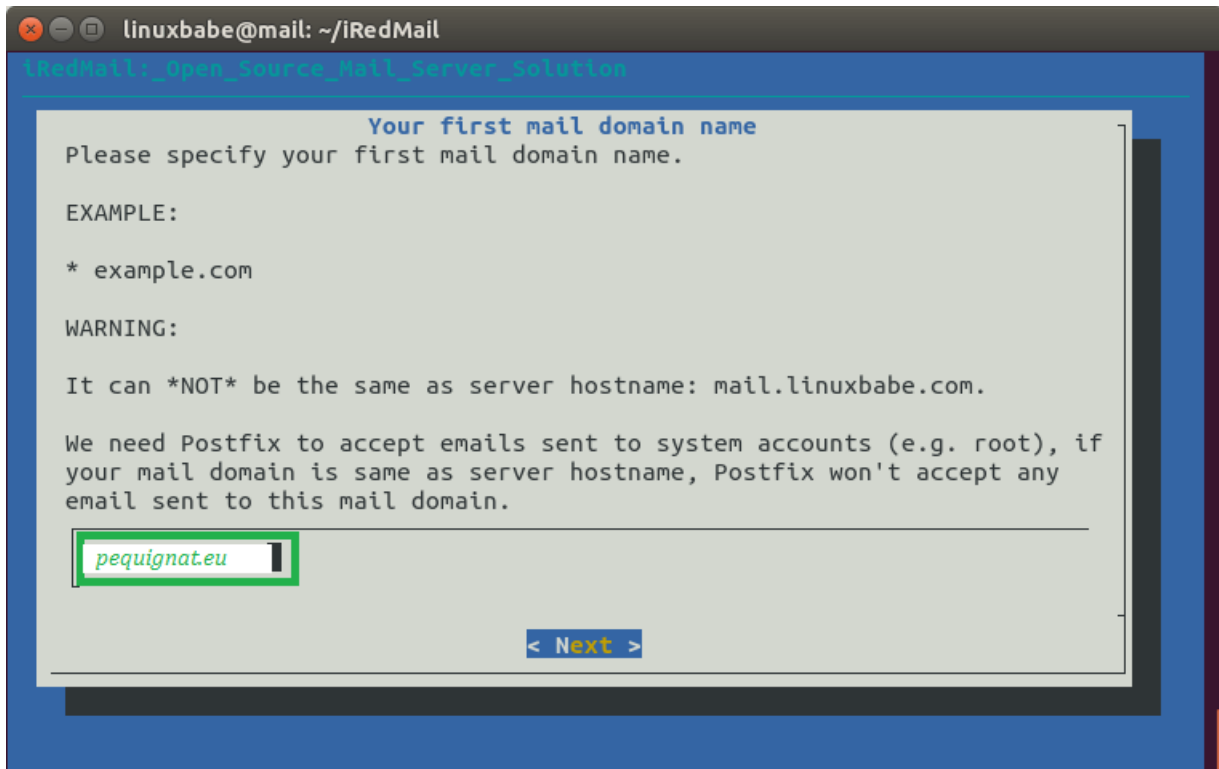
Sélectionnez le Serveur Nginx par la touche espace, puis Tab et Enter



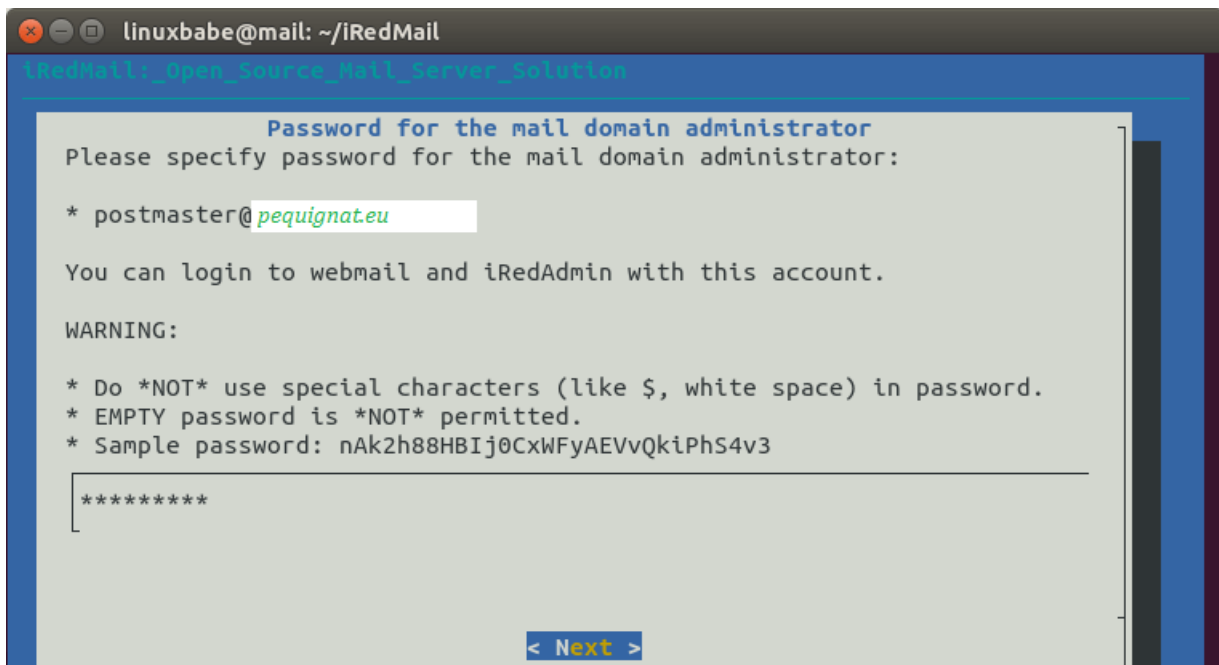
Avec la flèche du bas du clavier sélectionnez MariaDB avec la touche espace, puis Tab et Enter



Définir votre mot de passe pour l'utilisateur root de la base de données. Sera utile pour l'installation de Drupal 9 par la suite.



Mettez ici votre domaine racine du site internet. Pour moi : pequignat.eu



Choisissez un mot de passe complexe pour le compte qui administrera les mails.

```
iRedMail:_Open_Source_Mail_Server_Solution

Optional components
* DKIM signing/verification and SPF validation are enabled by default.
* DNS records for SPF and DKIM are required after installation.

Refer to below file for more detail after installation:
* /root/iRedMail/iRedMail.tips

[*] Roundcubemail Popular webmail built with PHP and AJAX
[ ] SGOgo Webmail, Calendar, Address book
[*] netdata Awesome system monitor
[*] iRedAdmin Official web-based Admin Panel
[*] Fail2ban Ban IP with too many password failures

< Next >
```

Faire suivant.

Lorsque qu'une demande de confirmation, tapez « Y »

A la question de savoir si vous voulez activer le parfeux sur le port que vous avez changé, tapez « Y »

Et pour redémarrer le parfeux : tapez « Y » aussi.

Un fichier « iRedMail.tips » sera généré sur le serveur contenant le rappel des informations de connexions.

Vous devez rebooter le serveur debian pour prise en compte. Attention, cela prend plusieurs minutes maintenant.

```
sudo shutdown -r now
```

III.II. *Sécuriser le serveur de mail avec un certificat reconnu officiel gratuit « Let's Encrypt TLS Certificate »*

III.II.1 **Obtenir l'outil certbot**

```
sudo apt install certbot
```

Par défaut, iRedMail est configuré avec un certificat auto signé, ce qui fait qu'il n'est pas reconnu par les clients comme fiable.

Pour cela nous allons utiliser Let's Encrypt qui fournit gratuitement un certificat valide pour 3 mois à renouveler.

```
sudo certbot certonly --webroot --agree-tos --email you@example.com -d mail.pequignat.eu -w /var/www/html/
```

```
-----  
Would you be willing to share your email address with the Electronic Frontier  
Foundation, a founding partner of the Let's Encrypt project and the non-profit  
organization that develops Certbot? We'd like to send you email about our work  
encrypting the web, EFF news, campaigns, and ways to support digital freedom.  
-----  
(Y)es/(N)o: n
```

A la question, si vous ne voulez pas recevoir des demandes de fonds, répondez « n ».

Mettre une adresse mail vous concernant valide pour le « you@exemple.com »

Si cela se passe bien, vous recevrez les fichiers de certificats dans le répertoire :

```
/etc/letsencrypt/live/mail.pequignat.eu/
```

IMPORTANT NOTES:

- Congratulations! Your certificate and chain have been saved at:
/etc/letsencrypt/live/mail.linuxbabe.com/fullchain.pem
Your key file has been saved at:
/etc/letsencrypt/live/mail.linuxbabe.com/privkey.pem
Your cert will expire on 2019-02-01. To obtain a new or tweaked version of this certificate in the future, simply run certbot again. To non-interactively renew *all* of your certificates, run "certbot renew"
- Your account credentials have been saved in your Certbot configuration directory at /etc/letsencrypt. You should make a secure backup of this folder now. This configuration directory will also contain certificates and private keys obtained by Certbot so making regular backups of this folder is ideal.
- If you like Certbot, please consider supporting our work by:

Donating to ISRG / Let's Encrypt: <https://letsencrypt.org/donate>
Donating to EFF: <https://eff.org/donate-le>

Vérifier que vous avez bien configuré les deux fichiers :

```
sudo nano /etc/nginx/sites-enabled/00-default.conf
```

```
#
# Note: This file must be loaded before other virtual host config files,
#
# HTTP
server {
    # Listen on ipv4
    listen 80;
    listen [::]:80;

    server_name _;

    # Redirect all insecure http:// requests to https://
    return 301 https://$host$request_uri;
}
```

```
sudo nano /etc/nginx/sites-enabled/00-default-ssl.conf
```

```
#
# Note: This file must be loaded before other virtual host config files,
#
# HTTPS
server {
    listen 443 ssl http2;
    listen [::]:443 ssl http2;
    server_name _;

    root /var/www/html;
    index index.php index.html;

    include /etc/nginx/templates/misc.tpl;
    include /etc/nginx/templates/ssl.tpl;
    include /etc/nginx/templates/iredadmin.tpl;
}
```



```
include /etc/nginx/templates/roundcube.tpl;
include /etc/nginx/templates/sogo.tpl;
include /etc/nginx/templates/netdata.tpl;
include /etc/nginx/templates/php-catchall.tpl;
include /etc/nginx/templates/stub_status.tpl;
}
```

III.II.2 Vérifier la bonne configuration

```
sudo nginx -t
```

```
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
```

Relance du serveur :

```
sudo systemctl reload nginx
```

```
sudo systemctl reload nginx
```

```
sudo certbot certonly --webroot --agree-tos --email you@example.com -d
mail.pequignat.eu -w /var/www/html/
```

IV. Mise en place du serveur avec le certificat officiel Let's Encrypt

IV.I. Site internet

Bien que le site soit reconnu comme fiable, le serveur lui-même n'est pas encore.

```
sudo nano /etc/nginx/templates/ssl.tpl
```

```
ssl_protocols TLSv1.2 TLSv1.3;

# Fix 'The Logjam Attack'.
ssl_ciphers EECDH+CHACHA20:EECDH+AESGCM:EDH+AESGCM:AES256+EECDH;
```

```
ssl_prefer_server_ciphers on;
ssl_dhparam /etc/ssl/dh2048_param.pem;

# Greatly improve the performance of keep-alive connections over SSL.
# With this enabled, client is not necessary to do a full SSL-handshake for
# every request, thus saving time and cpu-resources.
ssl_session_cache shared:SSL:10m;

# To use your own ssl cert (e.g. "Let's Encrypt"), please create symbol link to
# ssl cert/key used below, so that we can manage this config file with Ansible.
#
# For example:
#
# rm -f /etc/ssl/private/iRedMail.key
# rm -f /etc/ssl/certs/iRedMail.crt
# ln -s /etc/letsencrypt/live/<domain>/privkey.pem /etc/ssl/private/iRedMail.key
# ln -s /etc/letsencrypt/live/<domain>/fullchain.pem /etc/ssl/certs/iRedMail.crt
#
# To request free "Let's Encrypt" cert, please check our tutorial:
# https://docs.iredmail.org/letsencrypt.html
ssl_certificate /etc/letsencrypt/live/mail.pequignat.eu/fullchain.pem;
ssl_certificate_key /etc/letsencrypt/live/mail.pequignat.eu/privkey.pem;
```

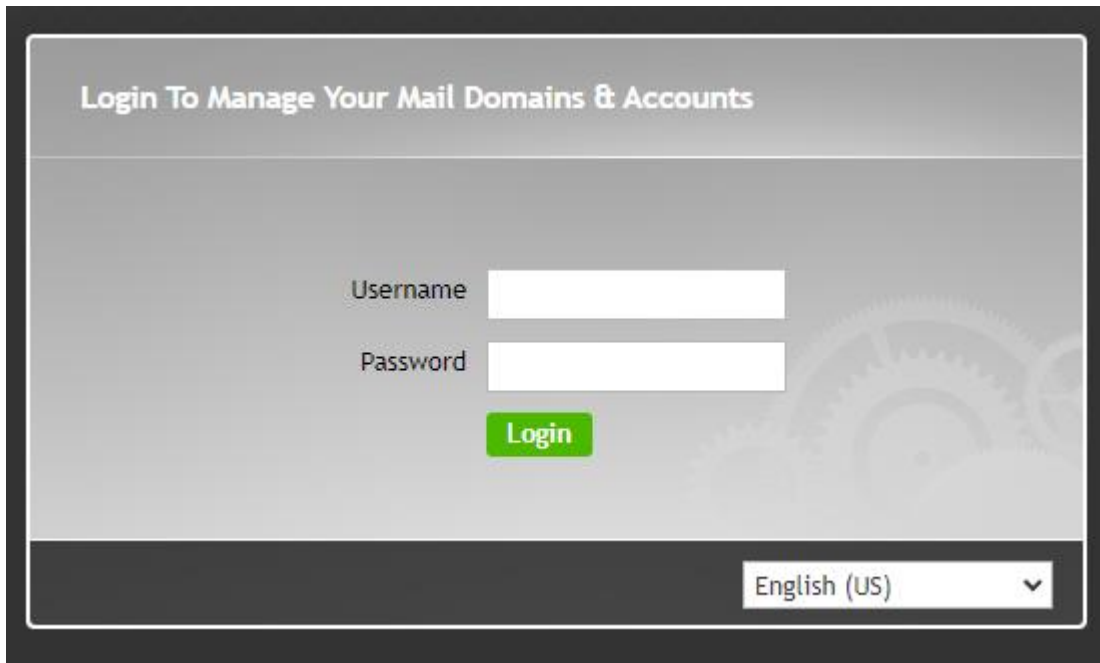
En rouge à changer

```
sudo nginx -t
```

```
sudo systemctl reload nginx
```

L'adresse est maintenant accessible et sécurisée dans le navigateur web :

<https://mail.pequignat.eu/iredadmin/>



IV.II. *Serveur de mail*

```
sudo nano /etc/postfix/main.cf
```

Changer aux ligne 95,96, 97

```
#  
# TLS settings.  
#  
# SSL key, certificate, CA  
#  
smtpd_tls_key_file = /etc/letsencrypt/live/mail.pequignat.eu/privkey.pem  
smtpd_tls_cert_file = /etc/letsencrypt/live/mail.pequignat.eu/cert.pem  
smtpd_tls_CAfile = /etc/letsencrypt/live/mail.pequignat.eu/chain.pem
```

Prendre en compte le changement de configuration :

```
sudo systemctl reload postfix
```

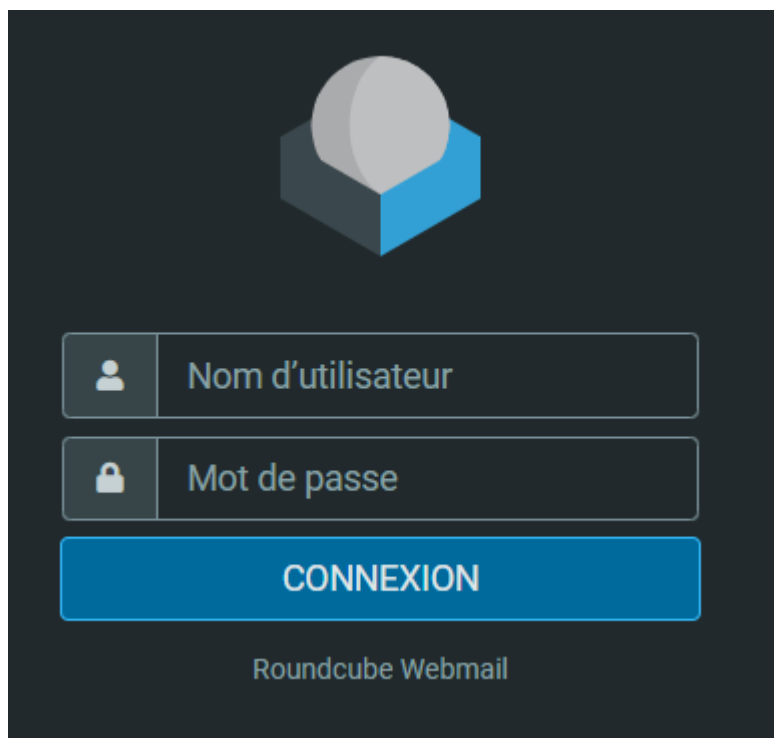
```
sudo nano /etc/dovecot/dovecot.conf
```

Ligne 47, 48

```
#ssl_ca = </path/to/ca  
ssl_cert = </etc/letsencrypt/live/mail.pequignat.eu/fullchain.pem  
ssl_key = </etc/letsencrypt/live/mail.pequignat.eu/privkey.pem
```

```
sudo systemctl reload dovecot
```

Vous pouvez maintenant tester la réception et l'envoi de mail avec l'éditeur de mail Roundcube :
<https://mail.pequignat.eu/mail/>



V. Mise d'espace de Swap

Afin de pallier une insuffisance de mémoire, car 2Go de RAM est peu. Il convient de rajouter un espace mémoire de Swap permettant avec des performances dégradées de permettre les traitements :

```
sudo fallocate -l 4G /swapfile
```

```
sudo chmod 600 /swapfile
```

```
sudo mkswap /swapfile
```

```
sudo swapon /swapfile
```

```
sudo nano /etc/fstab
```

Mettre à la fin du fichier :

```
/swapfile swap swap defaults 0 0
```

```
sudo systemctl daemon-reload
```

```
sudo systemctl restart clamav-daemon
```

VI. Mise d'un serveur Drupal 9 sur le Serveur Debian

VI.I. Base de données

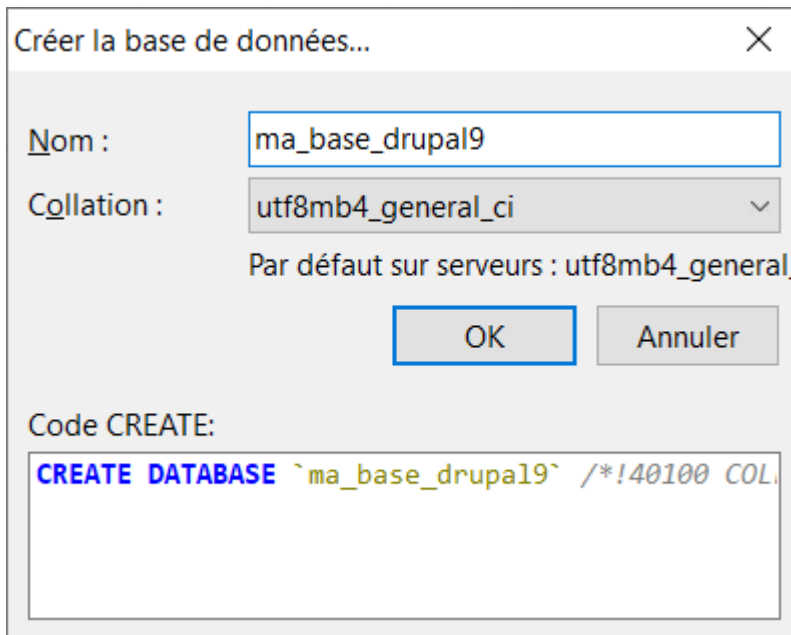
Avec un client comme HeidiSQL créer une base de données vide par le rebond en utilisant le port 3310 :

The screenshot shows the HeidiSQL configuration window with the following settings:

- Paramètres** (selected), Avancé, SSL, Statistiques
- Type de réseau : MariaDB or MySQL (TCP/IP)
- Library: libmariadb.dll
- Nom ou IP de l'hôte : localhost
- Demander les identifiants
- Utiliser l'identification Windows
- Utilisateur : root
- Mot de passe : [masked]
- Port : 3310
- Protocole client/serveur compressé
- Bases de données : Séparation par point-virgule
- Commentaire : [empty text area]

Créer la base de données ainsi que l'utilisateur qui aura accès à cette base.

Exemple :



Créer la base de données...

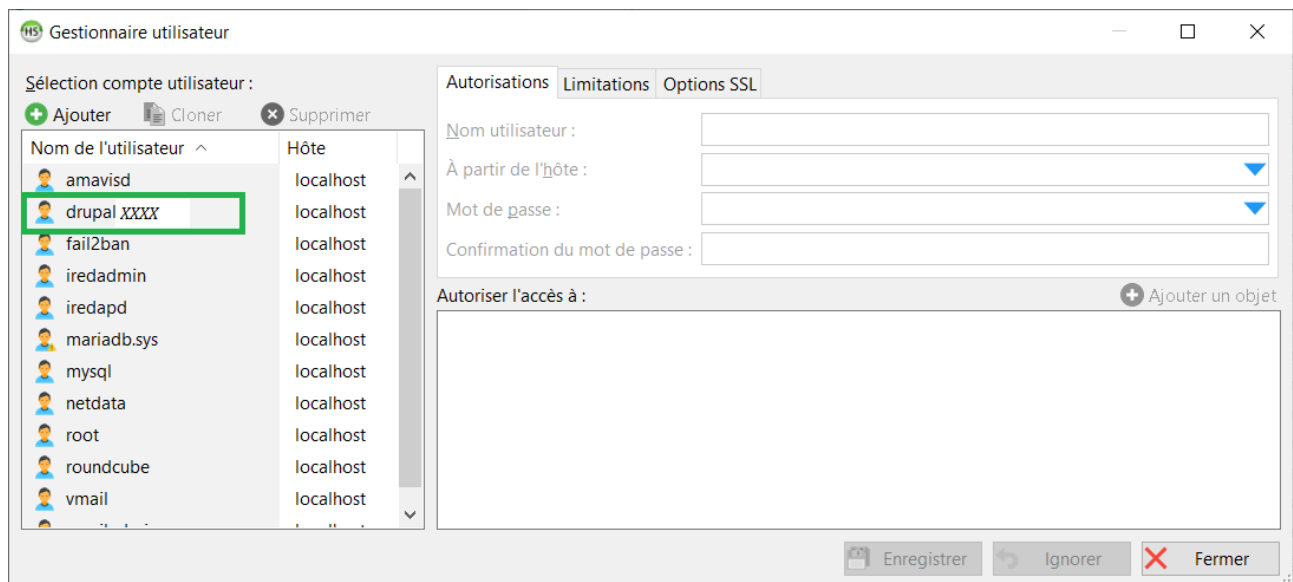
Nom :

Collation :

Par défaut sur serveurs : utf8mb4_general

Code CREATE:

```
CREATE DATABASE `ma_base_drupal9` /*!40100 COL...
```



Gestionnaire utilisateur

Sélection compte utilisateur :

Nom de l'utilisateur	Hôte
amavisd	localhost
drupal XXXX	localhost
fail2ban	localhost
iredadmin	localhost
iredapd	localhost
mariadb.sys	localhost
mysql	localhost
netdata	localhost
root	localhost
roundcube	localhost
vmail	localhost

Autorisations Limitations Options SSL

Nom utilisateur :

À partir de l'hôte :

Mot de passe :

Confirmation du mot de passe :

Autoriser l'accès à :

Associez les autorisations pour accéder complétement à la base drupalXXX

VI.II. Serveur Nginx

Créer un répertoire pour les données du site drupal dans « /var/www/drupal9 »

Vous pouvez aussi pour les fichiers privées, selon votre configuration créer un répertoire `private_drupal9` en frère.

```
/etc/nginx/sites-available$ sudo nano 01-www.conf
```

```
#
# Note: This file must be loaded before other virtual host config files,
#
# HTTP
server {
    # Listen on ipv4
    listen 80;
    listen [::]:80;

    server_name pequignat.eu;
    root /var/www/drupal9;

    # Redirect all insecure http:// requests to https://
    # force https-redirects
    if ($request_uri !~* "^/.well-known/(.*)") {
        return 301 https://$server_name$request_uri;
    }
}
```

```
sudo certbot certonly --webroot --agree-tos --email you@example.com -d
www.pequignat.eu -w /var/www/drupal9
```

```
sudo certbot certonly --webroot --agree-tos --email you@example.com -d
pequignat.eu -w /var/www/drupal9
```

```
/etc/nginx/sites-available$ sudo nano 01-www-ssl.conf
```

```
#
# Note: This file must be loaded before other virtual host config files,
#
#
```

```
# Le Vhost pequignat.eu
# #####
server {
    listen 443 ssl;
    listen [::]:443 ssl;
    server_name www.pequignat.eu;
    root /var/www/drupal9; ## <-- Your only path reference.

    location = /favicon.ico {
        log_not_found off;
        access_log off;
    }

    location = /robots.txt {
        allow all;
        log_not_found off;
        access_log off;
    }

    # Very rarely should these ever be accessed outside of your lan
    location ~* \.(txt|log)$ {
        allow 192.168.0.0/16;
        deny all;
    }

    location ~ \..*/.*\.php$ {
        return 403;
    }

    location ~ ^/sites/*/private/ {
        return 403;
    }

    # Block access to scripts in site files directory
    location ~ ^/sites/[^/]+/files/.*\.php$ {
        deny all;
    }
}
```



```
}

# Allow "Well-Known URIs" as per RFC 5785
location ~* ^/.well-known/ {
    allow all;
}

# Block access to "hidden" files and directories whose names begin with a
# period. This includes directories used by version control systems such
# as Subversion or Git to store control files.
location ~ (^|/)\. {
    return 403;
}

location / {
    # try_files $uri @rewrite; # For Drupal <= 6
    try_files $uri /index.php?$query_string; # For Drupal >= 7
}

location @rewrite {
    #rewrite ^/(.*)$ /index.php?q=$1; # For Drupal <= 6
    rewrite ^ /index.php; # For Drupal >= 7
}

# Don't allow direct access to PHP files in the vendor directory.
location ~ /vendor/.*\.php$ {
    deny all;
    return 404;
}

# Protect files and directories from prying eyes.
location ~*
\. (engine|inc|install|make|module|profile|po|sh|.*sql|theme|twig|tpl(\.php)?|xtra
pl|yaml) (~|\.sw[op]|\.bak|\.orig|\.save)>
    deny all;
    return 404;
}
```

```
}  
# In Drupal 8, we must also match new paths where the '.php' appears in  
# the middle, such as update.php/selection. The rule we use is strict,  
# and only allows this pattern with the update.php front controller.  
# This allows legacy path aliases in the form of  
# blog/index.php/legacy-path to continue to route to Drupal nodes. If  
# you do not have any paths like that, then you might prefer to use a  
# laxer rule, such as:  
# location ~ /\.php(/|)$ {  
# The laxer rule will continue to work if Drupal uses this new URL  
# pattern with front controllers other than update.php in a future  
# release.  
location ~ '\.php$|^/update.php' {  
    fastcgi_split_path_info ^(.+?\.php)(|/.*)$;  
    include snippets/fastcgi-php.conf;  
    fastcgi_pass 127.0.0.1:9999;  
}  
  
location ~* \.(js|css|png|jpg|jpeg|gif|ico|svg)$ {  
    try_files $uri @rewrite;  
    expires max;  
    log_not_found off;  
}  
  
# Fighting with Styles? This little gem is amazing.  
# location ~ ^/sites/*/files/imagecache/ { # For Drupal <= 6  
location ~ ^/sites/*/files/styles/ { # For Drupal >= 7  
    try_files $uri @rewrite;  
}  
  
# Handle private files through Drupal. Private file's path can come  
# with a language prefix.  
location ~ ^(/[a-z\-]+)?/system/files/ { # For Drupal >= 7  
    try_files $uri /index.php?$query_string;  
}
```

```
# Enforce clean URLs
# Removes index.php from urls like www.example.com/index.php/my-page -->
www.example.com/my-page
# Could be done with 301 for permanent or other redirect codes.
if ($request_uri ~* "(.*)index\.php/(.*)") {
    return 307 $1$2;
}

# SSL
ssl_protocols TLSv1.2;
ssl_ciphers EECDH+AESGCM:EDH+AESGCM:AES256+EECDH:AES256+EDH;
ssl_prefer_server_ciphers on;
ssl_dhparam /etc/ssl/dh2048_param.pem;
ssl_certificate /etc/letsencrypt/live/www.pequignat.eu/fullchain.pem;
ssl_certificate_key /etc/letsencrypt/live/www.pequignat.eu/privkey.pem;
# SSL
}

server {
    listen 443 ssl;
    listen [::]:443 ssl;
    server_name pequignat.eu;
    root /var/www/drupal9; ## <-- Your only path reference.

    location = /favicon.ico {
        log_not_found off;
        access_log off;
    }

    location = /robots.txt {
        allow all;
        log_not_found off;
```

```
        access_log off;
    }

    # Very rarely should these ever be accessed outside of your lan
    location ~* \.(txt|log)$ {
        allow 192.168.0.0/16;
        deny all;
    }

location ~ \..*/.*\.php$ {
    return 403;
}

location ~ ^/sites/.*/private/ {
    return 403;
}

# Block access to scripts in site files directory
location ~ ^/sites/[^/]+/files/.*\.php$ {
    deny all;
}

# Allow "Well-Known URIs" as per RFC 5785
location ~* ^/.well-known/ {
    allow all;
}

# Block access to "hidden" files and directories whose names begin with a
# period. This includes directories used by version control systems such
# as Subversion or Git to store control files.
location ~ (^|/)\. {
    return 403;
}

location / {
    # try_files $uri @rewrite; # For Drupal <= 6
```

```
    try_files $uri /index.php?$query_string; # For Drupal >= 7
}

location @rewrite {
    #rewrite ^/(.*)$ /index.php?q=$1; # For Drupal <= 6
    rewrite ^ /index.php; # For Drupal >= 7
}

# Protect files and directories from prying eyes.
location ~*
\. (engine|inc|install|make|module|profile|po|sh|.*sql|theme|twig|tpl(\.php)?|xmtm
pl|yaml) (~|\.sw[op]|\.bak|\.orig|\.save)>
    deny all;
    return 404;
}

# In Drupal 8, we must also match new paths where the '.php' appears in
# the middle, such as update.php/selection. The rule we use is strict,
# and only allows this pattern with the update.php front controller.
# This allows legacy path aliases in the form of
# blog/index.php/legacy-path to continue to route to Drupal nodes. If
# you do not have any paths like that, then you might prefer to use a
# laxer rule, such as:
# location ~ \.php(/|$) {
# The laxer rule will continue to work if Drupal uses this new URL
# pattern with front controllers other than update.php in a future
# release.
location ~ '\.php$|^/update.php' {
    fastcgi_split_path_info ^(.+?\.(php))(/.*)$;
    include snippets/fastcgi-php.conf;
    fastcgi_pass 127.0.0.1:9999;
}

location ~* \.(js|css|png|jpg|jpeg|gif|ico|svg)$ {
    try_files $uri @rewrite;
    expires max;
```

```
    log_not_found off;
}

# Fighting with Styles? This little gem is amazing.
# location ~ ^/sites/*/files/imagecache/ { # For Drupal <= 6
location ~ ^/sites/*/files/styles/ { # For Drupal >= 7
    try_files $uri @rewrite;
}

# Handle private files through Drupal. Private file's path can come
# with a language prefix.
location ~ ^(/[a-z\-\-]+)?/system/files/ { # For Drupal >= 7
    try_files $uri /index.php?$query_string;
}

# Enforce clean URLs

# Removes index.php from urls like www.example.com/index.php/my-page -->
www.example.com/my-page

# Could be done with 301 for permanent or other redirect codes.
if ($request_uri ~* "^(.*)index\.php/(.*)") {
    return 307 $1$2;
}

# SSL
ssl_protocols TLSv1.2;
ssl_ciphers EECDH+AESGCM:EDH+AESGCM:AES256+EECDH:AES256+EDH;
ssl_prefer_server_ciphers on;
ssl_dhparam /etc/ssl/dh2048_param.pem;
ssl_certificate /etc/letsencrypt/live/pequignat.eu/fullchain.pem;
ssl_certificate_key /etc/letsencrypt/live/pequignat.eu/privkey.pem;
# SSL
}
```

Pour activer les sites, faire un lien symbolique

Command : `ln -s <source> <destination>`

```
:/etc/nginx/sites-enabled$ ls -la
lrwxrwxrwx 1 root root 46 Jul 24 01:41 00-default-ssl.conf ->
/etc/nginx/sites-available/00-default-ssl.conf
lrwxrwxrwx 1 root root 42 Jul 24 01:41 00-default.conf -> /etc/nginx/sites-
available/00-default.conf
lrwxrwxrwx 1 root root 42 Jul 24 08:31 01-www-ssl.conf -> /etc/nginx/sites-
available/01-www-ssl.conf
lrwxrwxrwx 1 root root 38 Jul 24 08:26 01-www.conf -> /etc/nginx/sites-
available/01-www.conf
```

```
sudo nginx -t
```

```
sudo systemctl reload nginx
```

VII. Fin du document