

Installation d'un VPS

Stéphane Apiou

Version 1.0, 2020-03-27

Table of Contents

| | |
|--|----|
| 1. Avant propos | 1 |
| 2. Choix du VPS | 3 |
| 3. Choix du registrar | 4 |
| 4. Se loguer root sur le serveur | 5 |
| 5. Gestion des mots de passe | 6 |
| 6. Configuration basique | 8 |
| 6.1. Mettre l'éditeur de votre choix | 8 |
| 6.2. Installation d'un repository pour <i>/etc</i> | 8 |
| 6.3. Mise à jour des sources de paquets Debian | 10 |
| 6.4. Installation des paquets de base | 11 |
| 6.5. Installer l'outil Debfooster | 11 |
| 6.6. Création d'un fichier keeper dans <i>/etc</i> | 12 |
| 6.7. Installation des mises à jours automatiques | 13 |
| 6.8. Vérification du nom de serveur | 13 |
| 6.9. Configurer une IPV6 | 15 |
| 6.10. Interdire le login direct en root | 15 |
| 6.11. Création d'une clé de connexion ssh locale | 16 |
| 6.12. Sudo sans mot de passe | 18 |
| 6.13. Installer l'outil dselect | 19 |
| 6.14. Ajouter un fichier de swap | 19 |
| 7. Installation initiale des outils | 21 |
| 7.1. Configuration de Postfix | 21 |
| 7.2. Configuration de MariaDB | 22 |
| 7.3. Configuration d'Apache | 24 |
| 7.4. Installation et Configuration de Mailman | 25 |
| 7.5. Configuration d'Awstats | 27 |
| 7.6. Configuration de Fail2ban | 27 |
| 7.7. Installation et configuration de PureFTPD | 28 |
| 7.8. Installation et configuration de phpmyadmin | 30 |
| 7.9. Installation et configuration de Roundcube | 34 |
| 7.10. Installation de Let's Encrypt | 35 |
| 7.11. Installation d'un scanner de vulnérabilités | 35 |
| 8. Installation d'un Panel | 37 |
| 8.1. Installation et configuration de ISPConfig | 37 |
| 8.2. Installation de Webmin | 39 |
| 9. Configuration d'un domaine | 42 |
| 9.1. Login initial | 42 |
| 9.2. Création de la zone DNS d'un domaine | 44 |

| | |
|--|----|
| 9.3. Activation de DNSSEC | 45 |
| 9.4. Exemple de configuration de domaine | 47 |
| 9.5. Création d'un sous domaine | 48 |
| 9.6. Création d'un site web | 49 |
| 9.7. Création d'un Site Vhost | 50 |
| 10. Associer des certificats reconnu à vos outils | 52 |
| 11. Surveillance du serveur avec Munin et Monit | 54 |
| 11.1. Note préliminaire | 54 |
| 11.2. Installation et configuration de Munin | 54 |
| 11.3. Activez les plugins de Munin | 58 |
| 11.4. Installer et configurer Monit | 58 |
| 12. Configuration de la messagerie | 62 |
| 12.1. Installation de rspamd à la place d' Amavis-new | 62 |
| 12.2. Création du serveur de messagerie | 68 |
| 12.3. Finaliser la sécurisation de votre serveur de mail | 68 |
| 12.4. Création de l'autoconfig pour Thunderbird et Android | 69 |
| 12.5. Création d'autodiscover pour Outlook | 71 |
| 12.6. Création d'une boîte mail | 74 |
| 12.7. Configuration de votre client de messagerie. | 75 |
| 12.8. Mise en oeuvre du site web de webmail | 75 |
| 12.9. Transfert de vos boîtes mails IMAP | 76 |
| 13. Installation de Joomla ou de Concrete5 | 78 |
| 13.1. Création du site web de Joomla | 78 |
| 13.2. Création de l'application Joomla | 78 |
| 14. Installation de Mediawiki | 80 |
| 14.1. Création du site web de Mediawiki | 80 |
| 14.2. Création de l'application Mediawiki | 80 |
| 15. Installation de Wordpress | 82 |
| 15.1. Création du site web de Wordpress | 82 |
| 15.2. Création de l'application Wordpress | 82 |
| 16. Micro Weber | 84 |
| 16.1. Création du site web de Microweber | 84 |
| 16.2. Création des bases de données | 84 |
| 16.3. Installation de Microweber | 85 |
| 17. Piwigo | 87 |
| 17.1. Création du site web de Piwigo | 87 |
| 17.2. Création des bases de données | 87 |
| 17.3. Installation de Piwigo | 88 |
| 18. Installation de Nextcloud | 89 |
| 18.1. Installation initiale | 89 |
| 18.2. Création du site web de Nextcloud | 89 |

| | |
|---|-----|
| 18.3. Création des bases de données | 90 |
| 18.4. Installation de Nextcloud | 91 |
| 19. Installation et configuration de Gitea | 92 |
| 19.1. Création du site web de Gitea | 92 |
| 19.2. Création des bases de données | 93 |
| 19.3. Téléchargez et installez Gitea | 94 |
| 19.4. Activer une connexion SSH dédiée | 95 |
| 20. Installation de Seafile | 97 |
| 20.1. Création du site web de Seafile | 97 |
| 20.2. Création de bases de données | 98 |
| 20.3. Téléchargez et installez Seafile | 99 |
| 20.4. Lancement initial | 100 |
| 20.5. Lancement automatique de Seafile | 101 |
| 21. Installation de Grafana | 104 |
| 21.1. Création du site web de Grafana | 104 |
| 21.2. Installation de Grafana | 105 |
| 21.3. Installation et configuration de Loki | 108 |
| 21.4. Installation et configuration de Promtail | 110 |
| 22. BorgBackup | 113 |
| 22.1. Introduction | 113 |
| 22.2. Installation du serveur de stockage | 113 |
| 22.3. Installation sur le serveur sauvegardé | 114 |
| 22.4. Effectuer un backup | 115 |
| 22.5. Lister les backups | 116 |
| 22.6. Vérifier un backup | 117 |
| 22.7. Restaurer un backup | 118 |
| 22.8. Supprimer vos vieux backups | 119 |
| 23. Installation d'un serveur de VPN Pritunl | 120 |
| 23.1. Création du site web de Pritunl | 120 |
| 23.2. Installation de Pritunl | 121 |
| 23.3. Configuration de Pritunl | 121 |
| 23.4. Se connecter au serveur de VPN | 123 |
| 23.5. Réparer une base Pritunl | 123 |
| 23.6. Mot de passe perdu | 124 |
| 24. Annexe | 125 |
| 24.1. Installation de Hestia | 125 |

Chapter 1. Avant propos

Ce document est disponible sur le site [ReadTheDocs](#) et sur [Github](#).

Cette documentation décrit la méthode que j'ai utilisé pour installer un serveur VPS sur la plateforme OVH. Elle est le résultat de très nombreuses heures de travail pour collecter la documentation nécessaire. Sur mon serveur, j'ai installé un Linux Debian 10. Cette documentation est facilement transposable pour des versions différentes de Debian ou à Ubuntu ou toute autre distribution basée sur l'un ou l'autre. En revanche si vous utilisez CentOS, il y aura des différences beaucoup plus importantes notamment liées au gestionnaire de paquets [yum](#), le nommage des paquets, les configurations par défaut et aux différences dans l'arborescence présente dans `/etc`.

Dans ce document, je montre la configuration de nombreux types de sites web et services dans un domaine en utilisant ISPConfig.

Sont installés:

- un panel [ISPConfig](#)
- un configurateur [Webmin](#)
- un serveur apache avec sa configuration let's encrypt et les plugins PHP, python et ruby
- un serveur de mail avec antispam, sécurisation d'envoi des mails et autoconfiguration pour Outlook, Thunderbird, Android.
- un webmail [roundcube](#),
- un serveur de mailing list [mailman](#),
- un serveur ftp et sftp sécurisé.
- un serveur de base de données et son interface web d'administration [phpmyadmin](#).
- des outils de sécurisation, de mise à jour automatique et d'audit du serveur
- un outil de Monitoring [Munin](#)
- un outil de Monitoring [Monit](#)
- un sous domaine pointant sur un site auto-hébergé (l'installation du site n'est pas décrite ici; Se référer à [Yunohost](#)),
- un site CMS sous [Joomla](#),
- un site CMS sous [Concrete5](#),
- un site WIKI sous [Mediawiki](#),
- un site [Wordpress](#),
- un site [Microweber](#),
- un site Photo sous [Piwigo](#),
- un site Collaboratif sous [Nextcloud](#),
- un site [Gitea](#) et son repository GIT,
- un serveur et un site de partage de fichiers [Seafile](#),

- un serveur [Grafana](#), [Prometheus](#), [Loki](#), Promtail pour gérer les statistiques et les logs du serveur,
- un serveur de sauvegardes [Duplicati](#)
- un serveur de VPN [Pritunl](#),

Dans ce document nous configurons un nom de domaine principal. Pour la clarté du texte, il sera nommé "example.com". Il est à remplacer évidemment par votre nom de domaine principal.

Je suppose dans ce document que vous savez vous connecter à distance sur un serveur en mode terminal, que vous savez vous servir de [ssh](#) pour Linux ou de [putty](#) pour Windows, que vous avez des notions élémentaires de Shell Unix et que vous savez vous servir de l'éditeur [vi](#). Si [vi](#) est trop compliqué pour vous, je vous suggère d'utiliser l'éditeur de commande [nano](#) à la place et de remplacer [vi](#) par [nano](#) dans toutes les lignes de commande.

Dans le document, on peut trouver des textes entourés de <texte>. Cela signifie que vous devez mettre ici votre propre texte selon vos préférences.

Le coût pour mettre en oeuvre ce type de serveur est relativement faible: * Compter 15-18€TTC/an pour un nom de domaine classique (mais il peut y avoir des promos) * Compter 5€TTC/mois pour un VPS de base (2Go de Ram, un coeur, 20Go de SSD). Une machine plus sérieuse sera à 15€/mois (8Go de Ram, 2 coeurs, 80Go de SSD).

Le budget est donc de 6-7€TTC/mois pour une offre d'entrée de gamme. Il faut plus sérieusement compter sur 16€/mois tout compris.

Chapter 2. Choix du VPS

Cette partie du guide s'adresse aux utilisateurs d'OVH. J'ai pour ma part choisi un serveur VPS SSD chez OVH avec 2Go de RAM. Au moment où j'écris ce document il possède un seul coeur et 20 Go de disque.

Choisissez d'installer une image Linux seule avec Debian 10. Une fois l'installation effectuée, vous recevez un Email sur l'adresse mail de votre compte OVH avec vos identifiants de login root. Ils serviront à vous connecter sur le serveur.

En vous loguant sur la [plateforme d'administration d'OVH](#), vous accéderez aux informations de votre serveur dans le menu Server → VPS. A cet endroit votre VPS doit y être indiqué.

En cliquant dessus un ensemble de menus doivent apparaitre pour administrer celui-ci. Vous y trouverez notamment:

- Son adresse <IP> et le nom de la machine chez OVH. Elle est du type "VPSxxxxxx.ovh.net".
- La possibilité de le redémarrer
- La possibilité de le réinstaller (avec perte complète de données)
- un KVM pour en prendre le controle console directement dans le navigateur
- un menu de configuration de reverse DNS (qui nous sera utile par la suite) pour définir le domaine par défaut
- le statut des services principaux (http, ftp, ssh ...)
- enfin des choix pour souscrire à un backup régulier, ajouter des disques ou effectuer un snapshot de la VM associée au VPS.

Chapter 3. Choix du registrar

Pour rappel, un registrar est une société auprès de laquelle vous pourrez acheter un nom de domaine sur une durée déterminée. Vous devrez fournir pour votre enregistrement un ensemble de données personnelles qui permettront de vous identifier en tant que propriétaire de ce nom de domaine.

Pour ma part j'ai choisi Gandi car il ne sont pas très cher et leur interface d'administration est simple d'usage. Vous pouvez très bien prendre aussi vos DNS chez OVH.

Une fois votre domaine enregistré et votre compte créé vous pouvez vous loguer sur la [plateforme de gestion de Gandi](#).

Allez dans Nom de domaine et sélectionnez le nom de domaine que vous voulez administrer. La vue générale vous montre les services actifs. Il faut une fois la configuration des DNS effectuée être dans le mode suivant:

- Serveurs de noms: Externes
- Emails: Inactif
- DNSSEC: Actif (cela sera activé dans une seconde étape de ce guide)

Vous ne devez avoir aucune boîte mail active sur ce domaine. A regardez dans le menu "Boîtes & redirections Mails". Vous devez reconfigurer les 'Enregistrements DNS' en mode externes. Dans le menu "serveurs de noms", vous devez configurer les serveurs de noms externe. Mettre 3 DNS:

- les deux DNS de votre domaine: ns1.<example.com> et ns2.<example.com>

Pour que tout cela fonctionne bien, ajoutez des Glue records:

- un pour ns1.<example.com> lié à l'adresse <IP> du serveur
- un pour ns2.<example.com> lié à l'adresse <IP> du serveur



Cette configuration du lien chez votre registrar des deux DNS de votre serveur n'est à faire qu'après avoir défini le premier domaine de votre serveur

Il y a la possibilité chez OVH d'utiliser un DNS secondaire. Je ne l'ai pas mis en oeuvre.



Avoir un DNS sur au moins deux machines distinctes est la configuration recommandée.

Le menu restant est associé à DNSSEC; nous y reviendrons plus tard.

Chapter 4. Se loguer root sur le serveur

A de nombreux endroit dans la documentation, il est demandé de se loguer root sur le serveur. Pour se loguer root, et dans l'hypothèse que vous avez mis en place un compte sudo:

1. De votre machine locale, loguez vous avec votre compte `<sudo_username>`. Tapez :

```
ssh <sudo_username>@<example.com> ①
```

- ① Mettez ici `<sudo_username>` par votre nom de login et `<example.com>` par votre nom de domaine ou son adresse IP. Au début votre nom de domaine acheté n'est pas encore configuré. Il faut donc utiliser le nom de machine (par exemple pour un VPS OVH: `VPSxxxxxx.ovh.net` ou pour un raspberry: `raspberrypi.local`) ou votre adresse IP.

ou utilisez putty si vous êtes sous Windows.

2. Tapez votre mot de passe s'il est demandé. Si vous avez installé une clé de connexion ce ne devrait pas être le cas.
3. Loguez-vous `root`. Tapez :

```
sudo bash
```

Un mot de passe vous est demandé. Tapez le mot de passe demandé.

4. Dans le cas contraire (pas de sudo créé et connexion en root directe sur le serveur):
 - a. Se loguer root sur le serveur distant. Tapez:

```
ssh root@<example.com> ①
```

- ① remplacer ici `<example.com>` par votre nom de domaine.

Tapez ensuite votre mot de passe root

Chapter 5. Gestion des mots de passe

A propos des mots de passe: il est conseillé de saisir des mots de passe de 10 caractères contenant des majuscules/minuscules/nombres/caractères spéciaux. Une autre façon de faire est de saisir de longues phrases. Par exemple: 'J'aime manger de la mousse au chocolat parfumée à la menthe'. Ce dernier exemple a un taux de complexité bien meilleur qu'un mot de passe classique. Il est aussi plus facile à retenir que 'Az3~1ym_a&'.

Cependant, si vous êtes en manque d'inspiration et que vous souhaitez générer des mots de passe, voici quelques méthodes:

1. En se basant sur la date. Tapez:

```
date +%s | sha256sum | base64 | head -c 32 ; echo ①
```

- ① remplacez 32 par la valeur qui vous convient pour générer un mot de passe d'une taille différente de 32 caractères

2. En se basant sur les nombres aléatoires système. Tapez l'une des deux lignes ci dessous :

```
tr -cd '[:graph:]' < /dev/urandom | head -c 32; echo ①  
tr -cd A-Za-z0-9 < /dev/urandom | head -c 32;echo ①
```

- ① remplacez 32 par la valeur qui vous convient pour générer un mot de passe d'une taille différente de 32 caractères

3. En utilisant Openssl. Tapez :

```
openssl rand -base64 32 | cut -c-32 ①
```

- ① remplacez 32 par la valeur qui vous convient pour générer un mot de passe d'une taille différente de 32 caractères

4. En utilisant gpg. Tapez :

```
gpg --gen-random --armor 1 32 | cut -c-32 ①
```

- ① remplacez 32 par la valeur qui vous convient pour générer un mot de passe d'une taille différente de 32 caractères

5. En utilisant pwgen pour générer des mots de passe qui suivent des règles de longueur et types de caractères.

- a. Pour installer l'outil, tapez:

```
apt install pwgen
```

b. Ensuite tapez :

```
pwgen -Bcny 32 -1 ①
```

① remplacez 32 par la valeur qui vous convient pour générer un mot de passe d'une taille différente de 32 caractères. La commande crée un mot de passe non ambiguë avec au moins une majuscule, une valeur numérique, un symbole.

6. En utilisant `apg` pour générer des mots de passe prononcables tel que: `7quiGrikCod+` (`SEVEN-qui-Grik-Cod-PLUS_SIGN`)

a. Pour installer l'outil, tapez:

```
apt install apg
```

b. Ensuite tapez :

```
apg
```

7. En utilisant `xkcdpass` pour générer des passphrases comme: `context smashup spiffy cuddly throttle landfall`

a. Pour installer l'outil, tapez:

```
apt install xkcdpass
```

b. Ensuite tapez :

```
xkcdpass
```

Chapter 6. Configuration basique

6.1. Mettre l'éditeur de votre choix

En fonction de vos préférences en terme d'éditeur, choisissez celui qui vous convient pour les outils utilisant un éditeur de façon automatique tels que `crontab`.

Pour les débutants, il est conseillé d'utiliser nano.

Loguez vous comme root et tapez:

```
update-alternatives --config editor
```

6.2. Installation d'un repository pour `/etc`

Si vous souhaitez gérer en gestion de configuration le contenu de votre répertoire `/etc`, installez `etckeeper`.

Cette installation est optionnelle.

1. Loguez vous comme root sur le serveur

2. Tapez :

```
apt update  
apt install etckeeper
```

3. Vous pouvez créer un repository privé dans le cloud pour stocker votre configuration de serveur (autre serveur privé de confiance ou repository privé `Gitlab` ou `Github`).

4. Ajoutez ce repository distant. Pour `Gitlab` et `Github`, une fois le repository créé, demandez l'affichage de la commande git pour une communication en ssh. Tapez ensuite sur votre serveur :

```
cd /etc  
git remote add origin git@github.com:username/etc_keeper.git ①
```

① remplacer l'url par celle qui correspond au chemin de votre repository

5. modifier le fichier de configuration de `etckeeper`. tapez:

```
vi /etc/etckeeper/etckeeper.conf
```

6. Recherchez la ligne contenant `PUSH_REMOTE` et ajoutez y tous les repositories distant sur lesquels vous souhaitez pousser les modifications. Pour notre configuration, mettez:

```
PUSH_REMOTE="origin"
```

7. Pour éviter demandes de mot de passe de la part de **github** ou **gitlab**, il est nécessaire de déclarer une clé publique sur leur site. Créez une clé sur votre serveur pour l'utilisateur root:

a. Créer un répertoire **/root/.ssh** s'il n'existe pas. tapez :

```
cd /root  
mkdir -p .ssh
```

b. Allez dans le répertoire. Tapez :

```
cd /root/.ssh
```

c. Générez vous clés. Tapez :

```
ssh-keygen -t rsa
```

d. Un ensemble de questions apparaît. Si un texte vous explique que le fichier existe déjà, arrêtez la procédure. Cela signifie que vous avez déjà créé une clé et que vous risquez de perdre la connexion à d'autres serveurs si vous en générez une nouvelle. Sinon, appuyez sur Entrée à chaque fois pour accepter les valeurs par défaut.

e. Allez sur **gitlab** ou **github** dans la rubriques "settings" et le menu "SSH keys". Ajoutez la clé que vous aurez affiché avec la commande suivante:

```
cat /root/.ssh/id_rsa.pub
```

8. Effectuez un premier push. Tapez:

```
cd /etc  
git push -u origin master
```

9. aucun mot de passe ne doit vous être demandé. Si ce n'est pas le cas, re-vérifier les étapes précédentes.

10. Lancer **etckeeper**. Tapez:

```
etckeeper commit
```

11. Tout le contenu de **/etc** est poussé sur le repository. Saisissez un commentaire.

12. C'est fait !

6.3. Mise à jour des sources de paquets Debian

1. [Loguez vous comme root sur le serveur](#)
2. Modifier la liste standard de paquets
 - a. Éditer le fichier `/etc/apt/sources.list`. Tapez:

```
vi /etc/apt/sources.list
```

- b. Dé-commenter les lignes débutant par `deb` et contenant le terme `backports`. Par exemple pour `#deb http://deb.debian.org/debian buster-backports main contrib non-free` enlever le `#` en début de ligne
 - c. Ajouter sur toutes les lignes les paquets `contrib` et `non-free` . en ajoutant ces textes après chaque mot `main` du fichier `source.list`
 - d. Le fichier doit ressembler à ceci:

```
deb http://deb.debian.org/debian buster main contrib non-free
deb-src http://deb.debian.org/debian buster main contrib non-free

## Major bug fix updates produced after the final release of the
## distribution.
deb http://security.debian.org/ buster/updates main contrib non-free
deb-src http://security.debian.org/ buster/updates main contrib non-free
deb http://deb.debian.org/debian buster-updates main contrib non-free
deb-src http://deb.debian.org/debian buster-updates main contrib non-free

## N.B. software from this repository may not have been tested as
## extensively as that contained in the main release, although it includes
## newer versions of some applications which may provide useful features.
deb http://deb.debian.org/debian buster-backports main contrib non-free
deb-src http://deb.debian.org/debian buster-backports main contrib non-free
```

3. Effectuer une mise à niveau du système
 - a. Mettez à jour la liste des paquets. Tapez:

```
apt update
```

- b. Installez les nouveautés. Tapez:

```
apt dist-upgrade
```

4. Effectuez du ménage. Tapez:

```
apt autoremove
```

6.4. Installation des paquets de base

1. [Loguez vous comme root sur le serveur](#)
2. Tapez:

```
apt install curl wget ntpdate apt-transport-https apt-listchanges apt-file apt-  
rdepends man
```

6.5. Installer l'outil Debfooster

L'outil **debfooster** permet de ne conserver que les paquets essentiels.

Cette installation est optionnelle.

Il maintient un fichier **keepers** présent dans **/var/lib/debfooster**

En répondant aux questions de conservations de paquets, **debfooster** maintient la liste des paquets uniques nécessaires au système. Tous les autres paquets seront supprimés.

1. [Loguez vous comme root sur le serveur](#)
2. Ajouter le paquet **debfooster**. Tapez :

```
apt install debfooster
```

3. Lancez **debfooster**. Tapez :

```
debfooster
```

4. Répondez au questions pour chaque paquet
5. Acceptez la liste des modifications proposées à la fin. Les paquets superflus seront supprimés

Ci dessous une petite liste de paquets à conserver sur une installation basique:

| | | | |
|-------------|------------|-------------|-----------------------------|
| aptitude | cloud-init | cloud-utils | curl |
| debfooster | etckeeper | euca2ools | gdbm-l10n |
| grub-pc | ifenslave | kbd | linux-image-cloud- amd64 |
| locales-all | most | ntp | openssh-server |
| screen | unscd | whiptail | |

6.6. Création d'un fichier keeper dans /etc

Vous pourriez être intéressé après l'installation de **debfooster** et de **etckeeper** de construire automatiquement un fichier qui contient la liste des paquets qui permettent de réinstaller le système:

1. [Loguez vous comme root sur le serveur](#)

2. Tapez:

```
vi /etc/etckeeper/pre-commit.d/35debfooster
```

3. Saisissez dans le fichier:

```
#!/bin/sh
set -e

# Make sure sort always sorts in same order.
LANG=C
export LANG

shellquote() {
    # Single quotes text, escaping existing single quotes.
    sed -e "s/'/'\"'\"'/g" -e "s/^/'/" -e "s/$/'/"
}

if [ "$VCS" = git ] || [ "$VCS" = hg ] || [ "$VCS" = bazaar ] || [ "$VCS" = darcs ];
then
    # Make sure the file is not readable by others, since it can leak
    # information about contents of non-readable directories in /etc.
    debfooster -q -k /etc/keepers
    chmod 600 /etc/keepers
    sed -i "1i\\# debfooster file" /etc/keepers
    sed -i "1i\\# Generated by etckeeper. Do not edit." /etc/keepers

    # stage the file as part of the current commit
    if [ "$VCS" = git ]; then
        # this will do nothing if the keepers file is unchanged.
        git add keepers
    fi
    # hg, bazaar and darcs add not done, they will automatically
    # include the file in the current commit
fi
```

4. Sauvez et tapez:


```
chmod 755 /etc/etckeeper/pre-commit.d/35debfooster
```

5. Exécutez maintenant `etckeeper`

```
etckeeper commit
```

6. Le fichier `keepers` est créé et sauvegardé automatiquement.

6.7. Installation des mises à jours automatiques

Si vous souhaitez installer automatiquement les paquets Debian de correction de bugs de sécurité, cette installation est pour vous.

Cette installation est optionnelle.



L'installation automatique de paquets peut conduire dans certains cas très rare à des dysfonctionnements du serveur. Il est important de regarder périodiquement les logs d'installation

Suivez la procédure suivante:

1. [Loguez vous comme root sur le serveur](#)
2. Tapez:

```
apt install unattended-upgrades
```

6.8. Vérification du nom de serveur

Cette partie consiste à vérifier que le serveur a un hostname correctement configuré.

1. [Loguez vous comme root sur le serveur](#)
2. vérifier que le hostname est bien celui attendu (c'est à dire configuré par votre hébergeur). Tapez :

```
cat /etc/hostname
```

Le nom du hostname (sans le domaine) doit s'afficher.

- a. Si ce n'est pas le cas, changer ce nom en éditant le fichier. Tapez :

```
vi /etc/hostname
```

Changez la valeur, sauvegardez et rebootez. Tapez :

```
reboot
```

b. [Loguez vous comme root sur le serveur](#)

3. Vérifier le fichier `hosts`. Tapez :

```
cat /etc/hosts
```

Si le fichier contient plusieurs lignes avec la même adresse de loopback en `127.x.y.z`, en gardez une seule et celle avec le hostname et le nom de domaine complet.

a. si ce n'est pas le cas, changer les lignes en éditant le fichier. Tapez:

```
vi /etc/hosts
```

b. Changez la ou les lignes, sauvegardez.



Le FQDN (nom de machine avant le nom de domaine) doit être déclaré avant le hostname simple dans le fichier `hosts`.

c. Rebootez. Tapez :

```
reboot
```

d. [Loguez vous comme root sur le serveur](#)

4. Vérifiez que tout est correctement configuré.

a. Tapez :

```
hostname
```

La sortie doit afficher le nom de host.

b. Tapez ensuite :

```
hostname -f
```

La sortie doit afficher le nom de host avec le nom de domaine.

6.9. Configurer une IPV6

OVH propose des adresses IPV6. Ces adresses sont indiquées sur le panneau de synthèse du VPS (Dashboard).

La résolution par DHCP ne semble pas fonctionner. Il faut donc configurer l'adresse à la main:

Suivez la procédure suivante:

1. [Loguez vous comme root sur le serveur](#)
2. Tapez:

```
vi /etc/network/interfaces
```

3. Ajoutez ces lignes à la fin:

```
iface eth0 inet6 static
address <IPV6_ADDRESS> ①
post-up /sbin/ip -6 route add <GW_ADDRESS> dev eth0 ②
post-up /sbin/ip -6 route add default via <GW_ADDRESS> dev eth0 ②
pre-down /sbin/ip -6 route del default via <GW_ADDRESS> dev eth0 ②
pre-down /sbin/ip -6 route del <GW_ADDRESS> dev eth0 ②
```

- ① Mettre ici l'adresse IPV6 proposée par OVH
- ② Mettre ici l'adresse IPV6 du gateway proposé par OVH

6.10. Interdire le login direct en root

Il est toujours vivement déconseillé d'autoriser la possibilité de se connecter directement en SSH en tant que root. De ce fait, notre première action sera de désactiver le login direct en root et d'autoriser le sudo. Respectez bien les étapes de cette procédure:

1. [Loguez vous comme root sur le serveur](#)
2. Ajoutez un utilisateur standard qui sera nommé par la suite en tant que <sudo_username>
 - a. Tapez :

```
adduser <sudo_username>
```

- b. Répondez aux questions qui vont être posées: habituellement le nom complet d'utilisateur et le mot de passe.
- c. Donner les attributs sudo à l'utilisateur <sudo_username>. Tapez :

```
usermod -a -G sudo <sudo_username>
```

- d. Dans une autre fenêtre, se connecter sur le serveur avec votre nouveau compte `<sudo_username>`:

```
ssh <sudo_username>@<example.com> ①
```

- ① remplacer ici `<sudo_username>` par votre login et `<example.com>` par votre nom de domaine

- e. une fois logué, tapez:

```
sudo bash
```

Tapez le mot de passe de votre utilisateur. Vous devez avoir accès au compte root. Si ce n'est pas le cas, revérifiez la procédure et repassez toutes les étapes.



Tout pendant que ces premières étapes ne donnent pas satisfaction ne passez pas à la suite sous peine de perdre la possibilité d'accéder à votre serveur.

1. Il faut maintenant modifier la configuration de sshd.

- a. Editez le fichier `/etc/ssh/sshd_config`, Tapez:

```
vi /etc/ssh/sshd_config
```

il faut rechercher la ligne: `PermitRootLogin yes` et la remplacer par:

```
PermitRootLogin no
```

- b. Redémarrez le serveur ssh. Tapez :

```
service sshd restart
```

2. Faites maintenant l'essai de vous re-loguer avec le compte root. Tapez :

```
ssh root@<example.com> ①
```

- ① Remplacer ici `<example.com>` par votre nom de domaine

3. Ce ne devrait plus être possible: le serveur vous l'indique par un message `Permission denied, please try again.`

6.11. Création d'une clé de connexion ssh locale

Pour créer une clé et la déployer:

1. Créez une clé sur votre machine locale (et pas sur le serveur distant!):

- a. Ouvrir un terminal
- b. Créer un répertoire `~/.ssh` s'il n'existe pas. tapez :

```
mkdir -p $HOME/.ssh  
chmod 700 ~/.ssh
```

c. Allez dans le répertoire. Tapez :

```
cd ~/.ssh
```

d. Générez vous clés. Tapez :

```
ssh-keygen -t rsa
```

e. Un ensemble de questions apparaît. Si un texte vous explique que le fichier existe déjà, arrêtez la procédure. Cela signifie que vous avez déjà créé une clé et que vous risquez de perdre la connexion à d'autres serveurs si vous en générez une nouvelle. Sinon, appuyez sur Entrée à chaque fois pour accepter les valeurs par défaut.

2. Sur votre PC local afficher la clé à l'écran. Elle sera copiée-collée par la suite:

```
cat /root/.ssh/id_rsa.pub
```

3. Déployez votre clé:

a. Loguez vous sur votre serveur distant. Tapez :

```
ssh <sudo_username>@<example.com> ①
```

① remplacer ici `<sudo_username>` par votre login et `<example.com>` par votre nom de domaine

Entrez votre mot de passe

b. Créer un répertoire `~/.ssh` s'il n'existe pas. tapez: :

```
mkdir -p $HOME/.ssh
```

c. Éditez le fichier `~/.ssh/authorized_keys` tapez:

```
vi ~/.ssh/authorized_keys
```

et coller dans ce fichier le texte contenu dans le votre fichier local `~/.ssh/id_rsa.pub`.
Remarque: il peut y avoir déjà des clés dans le fichier `authorized_keys`.

d. Sécurisez votre fichier de clés. Tapez :

```
chmod 600 ~/.ssh/authorized_keys
```

e. Sécurisez le répertoire SSH; Tapez :

```
chmod 700 ~/.ssh
```

f. Déconnectez vous de votre session

4. Vérifiez que tout fonctionne en vous connectant. Tapez :

```
ssh <sudo_username>@<example.com> ①
```

① remplacer ici `<sudo_username>` par votre login et `<example.com>` par votre nom de domaine

La session doit s'ouvrir sans demander de mot de passe.

6.12. Sudo sans mot de passe

Avant tout, il faut bien se rendre compte que cela constitue potentiellement une faille de sécurité et qu'en conséquence, le compte possédant cette propriété devra être autant sécurisé qu'un compte root. L'intérêt étant d'interdire le compte root en connexion ssh tout en gardant la facilité de se loguer root sur le système au travers d'un super-compte.

1. [Loguez vous comme root sur le serveur](#)

2. Ajoutez un groupe `sudonp` et y affecter un utilisateur. Tapez :

```
addgroup --system sudonp
```

a. Ajouter l'utilisateur :

```
usermod -a -G sudonp <sudo_username>
```

b. Éventuellement retirez l'utilisateur du groupe `sudo` s'il a été ajouté auparavant :

```
gpasswd -d <sudo_username> sudo
```

c. Éditez le fichier `sudoers`. Tapez :

```
vi /etc/sudoers
```

d. Ajouter dans le fichier la ligne suivante:

```
%sudonp ALL=(ALL:ALL) NOPASSWD: ALL
```

L'utilisateur `nom_d_utilisateur` pourra se logger root sans mot de passe au travers de la commande `sudo bash`

6.13. Installer l'outil dselect

L'outil `dselect` permet de choisir de façon interactive les paquets que l'on souhaite installer.

1. [Loguez vous comme root sur le serveur](#)
2. Ajouter le paquet `dselect`. Tapez :

```
apt install dselect
```

6.14. Ajouter un fichier de swap

Pour un serveur VPS ou Raspberry Pi de 2 Go de RAM, la taille du fichier de swap sera de 2 Go. Si vous avez beaucoup d'outils et de serveurs à installer il peut être nécessaire d'avoir 4 Go de RAM au total + 2 Go de swap.

Enfin pour un Raspberry PI 3 avec 1 Go de Ram, il faut ajouter 1 Go de swap.

Tapez :

1. [Loguez vous comme root sur le serveur](#)
2. Tout d'abord, si l'outil `dphys-swapfile` est installé et configuré sur la machine, commencez par désactiver le swap. Tapez:

```
dphys-swapfile uninstall
```

3. Pour installer un swap de 2Go, tapez:

```
cd /  
fallocate -l 2G /swapfile  
chmod 600 /swapfile  
mkswap /swapfile  
swapon /swapfile
```

4. Enfin ajoutez une entrée dans le fichier fstab. Tapez :

```
vi /etc/fstab
```

5. Ajoutez la ligne:

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

6. Enfin vous pouvez être tenté de limiter le swap (surtout utile sur les systèmes avec peu de RAM et du SSD. Tapez:

```
vi /etc/systctl.conf
```

7. Ajoutez ou modifiez la ligne:

```
vm.swappiness = 5
```

8. Le paramètre sera actif au prochain reboot

Chapter 7. Installation initiale des outils

La procédure d'installation ci-dessous configure ISPconfig avec les fonctionnalités suivantes: Postfix, Dovecot, MariaDB, rkHunter, Apache, PHP, Let's Encrypt, PureFTPD, Bind, Webalizer, AWStats, fail2Ban, UFW Firewall, PHPMyadmin, RoundCube.

Pour les systèmes ayant 2 Go de RAM ou plus, il est fortement conseillé d'installer les outils ci après : Amavisd, SPamAssassin, ClamAV, Mailman.

1. [Loguez vous comme root sur le serveur](#)
2. Changez le Shell par défaut. Tapez :

```
dpkg-reconfigure dash
```

A la question **utilisez dash comme shell par défaut** répondez **non**. C'est bash qui doit être utilisé.

3. Installation de quelques paquets debian. ;-)

a. Tapez :

```
apt install patch ntp postfix postfix-mysql postfix-doc mariadb-client mariadb-server openssl getmail4 rkhunter binutils dovecot-imapd dovecot-pop3d dovecot-mysql dovecot-sieve dovecot-lmtpd unzip bzip2 arj nomarch lzop cabextract p7zip p7zip-full unrar lrzip libnet-ldap-perl libauthen-sasl-perl clamav-docs daemon libio-string-perl libio-socket-ssl-perl libnet-ident-perl zip libnet-dns-perl libdbd-mysql-perl postgresql apache2 apache2-doc apache2-utils libapache2-mod-php php php-common php-gd php-mysql php-imap php-cli php-cgi libapache2-mod-fcgid apache2-suexec-pristine php-pear mcrypt imagemagick libruby libapache2-mod-python php-curl php-intl php-pspell php-recode php-sqlite3 php-tidy php-xmlrpc php-xsl memcached php-memcache php-imagick php-gettext php-zip php-mbstring memcached libapache2-mod-passenger php-soap php-fpm php-opcache php-apcu bind9 dnsutils haveged webalizer awstats geoip-database libclass-dbi-mysql-perl libtimedate-perl fail2ban ufw anacron
```

b. Pour les systèmes avec plus de mémoire tapez :

```
apt install amavisd-new spamassassin clamav clamav-daemon
```

4. Aux questions posées répondez:

- a. **Type principal de configuration de mail:** ← Sélectionnez **Site Internet**
- b. **Nom de courrier:** ← Entrez votre nom de host. Par exemple: mail.example.com

7.1. Configuration de Postfix

Suivez la procédure suivante:

1. [Loguez vous comme root sur le serveur](#)
2. Editez le master.cf file de postfix. Tapez :

```
vi /etc/postfix/master.cf
```

3. Ajoutez dans le fichier:

```
submission inet n - - - smtpd
-o syslog_name=postfix/submission
-o smtpd_tls_security_level=encrypt
-o smtpd_sasl_auth_enable=yes
-o smtpd_client_restrictions=permit_sasl_authenticated,reject

smtps inet n - - - smtpd
-o syslog_name=postfix/smtps
-o smtpd_tls_wrappermode=yes
-o smtpd_sasl_auth_enable=yes
-o smtpd_client_restrictions=permit_sasl_authenticated,reject
```

4. Sauvegardez et relancez Postfix:

```
systemctl restart postfix
```

5. Si vous avez installé **SpamAssassin**, désactiver **SpamAssassin** puisque **amavisd** utilise celui ci en sous jacent. Tapez :

```
systemctl stop spamassassin
systemctl disable spamassassin
```

7.2. Configuration de MariaDB

Suivez la procédure suivante:

1. [Loguez vous comme root sur le serveur](#)
2. Sécurisez votre installation MariaDB. Tapez :

```
mysql_secure_installation
```

Répondez au questions ainsi:

- a. **Enter current password for root:** ← Tapez Entrée
- b. **Set root password? [Y/n]:** ← Tapez Y

- c. **New password::** ← Tapez votre mot de passe root MariaDB
 - d. **Re-enter New password::** ← Tapez votre mot de passe root MariaDB
 - e. **Remove anonymous users? [Y/n]:** ← Tapez Y
 - f. **Disallow root login remotely? [Y/n]:** ← Tapez Y
 - g. **Remove test database and access to it? [Y/n]:** ← Tapez Y
 - h. **Reload privilege tables now? [Y/n]:** ← Tapez Y
3. MariaDB doit pouvoir être atteint par toutes les interfaces et pas seulement localhost.
4. Éditez le fichier de configuration. :

```
vi /etc/mysql/mariadb.conf.d/50-server.cnf
```

5. Commentez la ligne **bind-address**:

```
#bind-address            = 127.0.0.1
```

6. Modifiez la méthode d'accès à la base MariaDB pour utiliser la méthode de login native.

- a. Tapez :

```
echo "update mysql.user set plugin = 'mysql_native_password' where user='root';"  
| mysql -u root
```

7. Editez le fichier debian.cnf. Tapez :

```
vi /etc/mysql/debian.cnf
```

- a. Aux deux endroits du fichier où le mot clé **password** est présent, mettez le mot de passe root de votre base de données.

```
password = votre_mot_de_passe
```

8. Pour éviter l'erreur **Error in accept: Too many open files**, augmenter la limite du nombre de fichiers ouverts.

- a. Editer le fichier: :

```
vi /etc/security/limits.conf
```

- b. Ajoutez à la fin du fichier les deux lignes:

```
mysql soft nofile 65535
mysql hard nofile 65535
```

9. Créez ensuite un nouveau répertoire. Tapez:

```
mkdir -p /etc/systemd/system/mysql.service.d/
```

a. Editer le fichier limits.conf :

```
vi /etc/systemd/system/mysql.service.d/limits.conf
```

b. Ajoutez dans le fichier les lignes suivantes:

```
[Service]
LimitNOFILE=infinity
```

10. Redémarrez votre serveur MariaDB. Tapez :

```
systemctl daemon-reload
systemctl restart mariadb
```

11. vérifiez maintenant que MariaDB est accessible sur toutes les interfaces réseau. Tapez :

```
netstat -tap | grep mysql
```

12. La sortie doit être du type: `tcp6 0 0 [::]:mysql [::]:* LISTEN 13708/mysql`

13. Pour les serveur avec peu de ressources quelques éléments de tuning. Editez le fichier 50-server.cnf:

```
vi /etc/mysql/mariadb.conf.d/50-server.cnf
```

7.3. Configuration d'Apache

Suivez la procédure suivante:

1. [Loguez vous comme root sur le serveur](#)
2. Installez les modules Apache nécessaires. Tapez :

```
a2enmod suexec rewrite ssl proxy_http actions include dav_fs dav auth_digest cgi
headers actions proxy_fcgi alias spelling
```

3. Pour ne pas être confronté aux problèmes de sécurité de type [HTTPoxy](#), il est nécessaire de créer un petit module dans apache.

a. Éditez le fichier httpoxy.conf :

```
vi /etc/apache2/conf-available/httpoxy.conf
```

b. Collez les lignes suivantes:

```
<IfModule mod_headers.c>
    RequestHeader unset Proxy early
</IfModule>
```

4. Activez le module en tapant :

```
a2enconf httpoxy
systemctl restart apache2
```

5. Désactiver la documentation apache en tapant:

```
a2disconf apache2-doc
systemctl restart apache2
```

7.4. Installation et Configuration de Mailman

Suivez la procédure suivante:

1. [Loguez vous comme root sur le serveur](#)

2. Tapez :

```
apt-get install mailman
```

3. Sélectionnez un langage:

a. **Languages to support:** ← Tapez en (English)

b. **Missing site list :** ← Tapez Ok

4. Créez une mailing list. Tapez:

```
newlist mailman
```

5. ensuite éditez le fichier aliases :

```
vi /etc/aliases
```

et ajoutez les lignes affichées à l'écran:

```
## mailman mailing list
mailman:                "|/var/lib/mailman/mail/mailman post mailman"
mailman-admin:          "|/var/lib/mailman/mail/mailman admin mailman"
mailman-bounces:        "|/var/lib/mailman/mail/mailman bounces mailman"
mailman-confirm:        "|/var/lib/mailman/mail/mailman confirm mailman"
mailman-join:           "|/var/lib/mailman/mail/mailman join mailman"
mailman-leave:          "|/var/lib/mailman/mail/mailman leave mailman"
mailman-owner:          "|/var/lib/mailman/mail/mailman owner mailman"
mailman-request:        "|/var/lib/mailman/mail/mailman request mailman"
mailman-subscribe:      "|/var/lib/mailman/mail/mailman subscribe mailman"
mailman-unsubscribe:    "|/var/lib/mailman/mail/mailman unsubscribe mailman"
```

6. Exécutez :

```
newaliases
```

et redémarrez postfix :

```
systemctl restart postfix
```

7. Activez la page web de mailman dans apache :

```
ln -s /etc/mailman/apache.conf /etc/apache2/conf-enabled/mailman.conf
```

8. Redémarrez apache :

```
systemctl restart apache2
```

puis redémarrez le demon mailman :

```
systemctl restart mailman
```

9. Le site web de mailman est accessible

- a. Vous pouvez accéder à la page admin Mailman à <http://<server1.example.com>/cgi-bin/mailman/admin/>
- b. La page web utilisateur de la mailing list est accessible ici <http://<server1.example.com>/cgi-bin/mailman/listinfo/>.
- c. Sous <http://<server1.example.com>/pipermail/mailman> vous avez accès aux archives.

7.5. Configuration d' Awstats

Suivez la procédure suivante:

1. [Loguez vous comme root sur le serveur](#)
2. configurer la tache cron d'awstats: Éditez le fichier :

```
vi /etc/cron.d/awstats
```

Et commentez toutes les lignes:

```
#MAILTO=root
#*/10 * * * * www-data [ -x /usr/share/awstats/tools/update.sh ] &&
/usr/share/awstats/tools/update.sh
# Generate static reports:
#10 03 * * * www-data [ -x /usr/share/awstats/tools/buildstatic.sh ] &&
/usr/share/awstats/tools/buildstatic.sh
```

7.6. Configuration de Fail2ban

Suivez la procédure suivante:

1. [Loguez vous comme root sur le serveur](#)
2. Editez le fichier jail.local :

```
vi /etc/fail2ban/jail.local
```

Ajoutez les lignes suivantes:

```
[dovecot]
enabled = true
filter = dovecot
logpath = /var/log/mail.log
maxretry = 5

[postfix-sasl]
enabled = true
port = smtp
filter = postfix[mode=auth]
logpath = /var/log/mail.log
maxretry = 3
```

3. Redémarrez Fail2ban: :

```
systemctl restart fail2ban
```

7.7. Installation et configuration de PureFTPd

Suivez la procédure suivante:

1. [Loguez vous comme root sur le serveur](#)

2. Tapez: :

```
apt-get install pure-ftpd-common pure-ftpd-mysql
```

3. Éditez le fichier de conf: :

```
vi /etc/default/pure-ftpd-common
```

4. Changez les lignes ainsi:

```
STANDALONE_OR_INETD=standalone
VIRTUALCHROOT=true
```

5. Autorisez les connexions TLS. Tapez:

```
echo 1 > /etc/pure-ftpd/conf/TLS
```

6. Créez un certificat SSL.

a. Tapez :


```
mkdir -p /etc/ssl/private/
```

b. Puis créez le certificat auto signé. Tapez :

```
openssl req -x509 -nodes -days 7300 -newkey rsa:2048 -keyout  
/etc/ssl/private/pure-ftpd.pem -out /etc/ssl/private/pure-ftpd.pem
```

et répondez aux questions de la manière suivante:

- i. **Country Name (2 letter code) [AU]:** ← Entrez le code pays à 2 lettres
- ii. **State or Province Name (full name) [Some-State]:** ← Entrer le nom d'état
- iii. **Locality Name (eg, city) []:** ← Entrer votre ville
- iv. **Organization Name (eg, company) [Internet Widgits Pty Ltd]:** ← Entrez votre entreprise ou tapez entrée
- v. **Organizational Unit Name (eg, section) []:** ← Tapez entrée
- vi. **Common Name (e.g. server FQDN or YOUR name) []:** ← Enter le nom d'hôte de votre serveur. Dans notre cas: server1.example.com
- vii. **Email Address []:** ← Tapez entrée

c. Puis tapez :

```
chmod 600 /etc/ssl/private/pure-ftpd.pem
```

d. et redémarrez pure-ftpd en tapant: :

```
systemctl restart pure-ftpd-mysql
```

e. En Option: Activer les quotas si votre kernel le permet.

- Installez les paquets de gestion des quotas. Tapez:

```
apt install quota quotatool
```

- Editez **fstab**. Tapez:

```
vi /etc/fstab
```

- Inserez le texte ci dessous pour chaque directive de montage

```
UUID=45576b38-39e8-4994-b8c1-ea4870e2e614 / ext4 errors=remount-  
ro,usrjquota=quota.user,grpjquota=quota.group,jqfmt=vfsv0 0 1
```

- Pour le Raspberry, éditez le fichier rc.local pour créer /dev/root à chaque reboot:

```
ln -s /dev/mmlk0p7 /dev/root  
vi /etc/rc.local
```

- Ajoutez avant `exit 0`:

```
ln -s /dev/mmlk0p7 /dev/root
```

- Pour activer les quotas, tapez:

```
mount -o remount /  
quotacheck -avugm  
quotaon -avug
```

7.8. Installation et configuration de phpmyadmin

Suivez la procédure suivante:

1. [Loguez vous comme root sur le serveur](#)
2. Installez phpmyadmin. Exécutez:

```
mkdir /usr/share/phpmyadmin  
mkdir /etc/phpmyadmin  
mkdir -p /var/lib/phpmyadmin/tmp  
chown -R www-data:www-data /var/lib/phpmyadmin  
touch /etc/phpmyadmin/htpasswd.setup  
cd /tmp  
wget https://files.phpmyadmin.net/phpMyAdmin/4.9.0.1/phpMyAdmin-4.9.0.1-all-  
languages.tar.gz  
tar xzf phpMyAdmin-4.9.0.1-all-languages.tar.gz  
mv phpMyAdmin-4.9.0.1-all-languages/* /usr/share/phpmyadmin/  
rm phpMyAdmin-4.9.0.1-all-languages.tar.gz  
rm -rf phpMyAdmin-4.9.0.1-all-languages  
cp /usr/share/phpmyadmin/config.sample.inc.php  
/usr/share/phpmyadmin/config.inc.php
```

3. Créez votre chaîne aléatoire en base64. Tapez:

```
tr -dc A-Za-z0-9 < /dev/urandom | head -c${1:-32};echo;
```

4. Copiez le texte généré
5. Éditez le fichier :

```
vi /usr/share/phpmyadmin/config.inc.php
```

- a. Modifier l'entrée `blowfish_secret` en ajoutant votre propre chaîne de 32 caractères générée juste avant.
- b. Éditez le fichier: :

```
vi /etc/apache2/conf-available/phpmyadmin.conf
```

- c. Ajoutez les lignes suivantes:

```
# phpMyAdmin default Apache configuration

Alias /phpmyadmin /usr/share/phpmyadmin

<Directory /usr/share/phpmyadmin>
    Options FollowSymLinks
    DirectoryIndex index.php

    <IfModule mod_php7.c>
        AddType application/x-httpd-php .php

        php_flag magic_quotes_gpc Off
        php_flag track_vars On
        php_flag register_globals Off
        php_value include_path .
    </IfModule>

</Directory>

# Authorize for setup
<Directory /usr/share/phpmyadmin/setup>
    <IfModule mod_authn_file.c>
        AuthType Basic
        AuthName "phpMyAdmin Setup"
        AuthUserFile /etc/phpmyadmin/htpasswd.setup
    </IfModule>
    Require valid-user
</Directory>

# Disallow web access to directories that don't need it
<Directory /usr/share/phpmyadmin/libraries>
    Order Deny,Allow
    Deny from All
</Directory>
<Directory /usr/share/phpmyadmin/setup/lib>
    Order Deny,Allow
    Deny from All
</Directory>
```

6. Activez le module et redémarrez apache. Tapez :

```
a2enconf phpmyadmin
systemctl restart apache2
```

7. Créer la base de donnée phpmyadmin.

a. Tapez :

```
mysql -u root -p
```

puis entrer le mot de passe root

b. Créez une base phpmyadmin. Tapez :

```
CREATE DATABASE phpmyadmin;
```

c. Créez un utilisateur phpmyadmin. Tapez :

```
CREATE USER 'pma'@'localhost' IDENTIFIED BY 'mypassword'; ①
```

① **mypassword** doit être remplacé par un mot de passe choisi.

d. Accordez des privilèges et sauvez:

```
GRANT ALL PRIVILEGES ON phpmyadmin.* TO 'pma'@'localhost' IDENTIFIED BY  
'mypassword' WITH GRANT OPTION; ①
```

① **mypassword** doit être remplacé par un mot de passe choisi.

e. Flusher les privilèges:

```
FLUSH PRIVILEGES;
```

f. et enfin

```
EXIT;
```

8. Chargez les tables sql dans la base phpmyadmin:

```
mysql -u root -p phpmyadmin < /usr/share/phpmyadmin/sql/create_tables.sql
```

9. Enfin ajoutez les mots de passe nécessaires dans le fichier de config.

a. Tapez:

```
vi /usr/share/phpmyadmin/config.inc.php
```

b. Rechercher le texte contenant **controlhost** . Ci-dessous, un exemple:

```

/* User used to manipulate with storage */
$cfg['Servers'][$i]['controlhost'] = 'localhost';
$cfg['Servers'][$i]['controlport'] = '';
$cfg['Servers'][$i]['controluser'] = 'pma';
$cfg['Servers'][$i]['controlpass'] = 'mypassword'; ①

/* Storage database and tables */
$cfg['Servers'][$i]['pmadb'] = 'phpmyadmin';
$cfg['Servers'][$i]['bookmarktable'] = 'pma__bookmark';
$cfg['Servers'][$i]['relation'] = 'pma__relation';
$cfg['Servers'][$i]['table_info'] = 'pma__table_info';
$cfg['Servers'][$i]['table_coords'] = 'pma__table_coords';
$cfg['Servers'][$i]['pdf_pages'] = 'pma__pdf_pages';
$cfg['Servers'][$i]['column_info'] = 'pma__column_info';
$cfg['Servers'][$i]['history'] = 'pma__history';
$cfg['Servers'][$i]['table_uiprefs'] = 'pma__table_uiprefs';
$cfg['Servers'][$i]['tracking'] = 'pma__tracking';
$cfg['Servers'][$i]['userconfig'] = 'pma__userconfig';
$cfg['Servers'][$i]['recent'] = 'pma__recent';
$cfg['Servers'][$i]['favorite'] = 'pma__favorite';
$cfg['Servers'][$i]['users'] = 'pma__users';
$cfg['Servers'][$i]['usergroups'] = 'pma__usergroups';
$cfg['Servers'][$i]['navigationhiding'] = 'pma__navigationhiding';
$cfg['Servers'][$i]['savedsearches'] = 'pma__savedsearches';
$cfg['Servers'][$i]['central_columns'] = 'pma__central_columns';
$cfg['Servers'][$i]['designer_settings'] = 'pma__designer_settings';
$cfg['Servers'][$i]['export_templates'] = 'pma__export_templates';

```

① A tous les endroit ou vous voyez dans le texte ci dessus le mot **mypassword** mettez celui choisi. N'oubliez pas de dé-commenter les lignes.

7.9. Installation et configuration de Roundcube

Suivez la procédure suivante:

1. [Loguez vous comme root sur le serveur](#)

2. Tapez:

```
apt-get install roundcube roundcube-core roundcube-mysql roundcube-plugins
```

3. Répondez aux question

- Utiliser **dbconfig_common** ← Répondre **Oui**
- Mot de passe Mysql pour db Roundcube ← Tapez un mot de passe

4. Éditez le fichier php de roundcube: :

```
vi /etc/roundcube/config.inc.php
```

et définissez les hosts par défaut comme localhost

```
$config['default_host'] = 'localhost';  
$config['smtp_server'] = 'localhost';
```

5. Éditez la configuration apache pour roundcube: :

```
vi /etc/apache2/conf-enabled/roundcube.conf
```

et ajouter au début les lignes suivantes:

```
Alias /roundcube /var/lib/roundcube  
Alias /webmail /var/lib/roundcube
```

6. Redémarrez Apache:

```
systemctl reload apache2
```

7.10. Installation de Let's Encrypt

Suivez la procédure suivante:

1. [Loguez vous comme root sur le serveur](#)
2. Installez Let's Encrypt. Tapez:

```
cd /usr/local/bin  
wget https://dl.eff.org/certbot-auto  
chmod a+x certbot-auto  
./certbot-auto --install-only
```

3. Une façon alternative de l'installer est:

```
apt install python3-certbot-apache
```

7.11. Installation d'un scanner de vulnérabilités

Suivez la procédure suivante:

1. [Loguez vous comme root sur le serveur](#)

2. installer Git. Tapez :

```
apt install git
```

3. installer Lynis

a. Tapez :

```
cd  
git clone https://github.com/CISOfy/lynis
```

b. Exécutez :

```
cd lynis;./lynis audit system
```

4. L'outil vous listera dans une forme très synthétique la liste des vulnérabilités et des améliorations de sécurité à appliquer.

Chapter 8. Installation d'un Panel

Il existe plusieurs type de panel de contrôle pour les VPS. La plupart sont payant.

Pour citer les plus connus:

- payant: cPanel (leader du type), Plesk
- gratuit: Yunohost (un excellent système d'autohébergement packagé) , Ajenti, Froxlor, Centos web panel, Webmin et Usermin, ISPConfig, HestiaCP, VestaCP ,

Ci après nous allons en présenter 3 différents (ISPConfig, Webmin et HestiaCP). Ils sont incompatibles entre eux.

On peut faire cohabiter ISPConfig et Webmin en prenant les précautions suivantes:

- ISPConfig est le maitre de la configuration: toute modification sur les sites webs, mailboxes et DNS doit impérativement être effectuées du coté d'ISPConfig
- Les modifications réalisées au niveau de webmin pour ces sites webs, mailboxes et DNS seront au mieux écrasées par ISPConfig au pire elles risquent de conduire à des incompatibilités qui engendreront des dysfonctionnement d'ISPConfig (impossibilité de mettre à jour les configurations)
- Le reste des modifications peuvent être configurées au niveau de webmin sans trop de contraintes.

Pour rappel, HestiaCP (tout comme VestaCP) sont incompatibles d'ISPConfig et de Webmin. Ils doivent être utilisés seuls

8.1. Installation et configuration de ISPConfig

ISPConfig est un système de configuration de sites web totalement compatible avec Webmin.

Pour installer ISPConfig, vous devez suivre la procédure ci-dessous. ISPConfig 3.1 a été utilisé dans ce tutoriel.

1. [Loguez vous comme root sur le serveur](#)
2. Tapez:

```
cd /tmp
```

3. Cherchez la dernière version d'ISPConfig sur le site [ISPConfig](#)
4. Installez cette version en tapant: :

```
wget <la_version_a_telecharger>.tar.gz
```

5. Décompressez la version en tapant: :

```
tar xzf <la_version>.tar.gz
```

6. Enfin allez dans le répertoire d'installation: :

```
cd ispconfig3_install/install/
```

7. Lancez l'installation: :

```
php -q install.php
```

et répondez aux questions:

- a. Select language (en,de) [en]: ← Tapez entrée
- b. Installation mode (standard,expert) [standard]: ← Tapez entrée
- c. Full qualified hostname (FQDN) of the server, eg server1.domain.tld [server1.example.com]: ← Tapez entrée
- d. MySQL server hostname [localhost]: ← Tapez entrée
- e. MySQL server port [3306]: ← Tapez entrée
- f. MySQL root username [root]: ← Tapez entrée
- g. MySQL root password []: ← Enter your MySQL root password
- h. MySQL database to create [dbispconfig]: ← Tapez entrée
- i. MySQL charset [utf8]: ← Tapez entrée
- j. Country Name (2 letter code) [AU]: ← Entrez le code pays à 2 lettres
- k. State or Province Name (full name) [Some-State]: ← Entrer le nom d'état
- l. Locality Name (eg, city) []: ← Entrer votre ville
- m. Organization Name (eg, company) [Internet Widgits Pty Ltd]: ← Entrez votre entreprise ou tapez entrée
- n. Organizational Unit Name (eg, section) []: ← Tapez entrée
- o. Common Name (e.g. server FQDN or YOUR name) []: ← Enter le nom d'hôte de votre serveur. Dans notre cas: server1.example.com
- p. Email Address []: ← Tapez entrée
- q. ISPConfig Port [8080]: ← Tapez entrée
- r. Admin password [admin]: ← Tapez entrée
- s. Do you want a secure (SSL) connection to the ISPConfig web interface (y,n) [y]: ← Tapez entrée
- t. Country Name (2 letter code) [AU]: ← Entrez le code pays à 2 lettres
- u. State or Province Name (full name) [Some-State]: ← Entrer le nom d'état

- v. **Locality Name** (eg, city) []: ← Entrez votre ville
- w. **Organization Name** (eg, company) [Internet Widgits Pty Ltd]: ← Entrez votre entreprise ou tapez entrée
- x. **Organizational Unit Name** (eg, section) []: ← Tapez entrée
- y. **Common Name** (e.g. server FQDN or YOUR name) []: ← Enter le nom d'hôte de votre serveur.
Dans notre cas: server1.example.com
- z. **Email Address** []: ← Tapez entrée

8. Sécurisez Apache

- a. Il est maintenant recommandé de désactiver les protocoles TLS 1.0 et TLS 1.1. Ce n'est pas la configuration par défaut d'ISPconfig
- b. [Loguez vous comme root sur le serveur.](#)
- c. Copier le fichier **vhost.conf.master** dans la zone custom

```
cp /usr/local/ispconfig/server/conf/vhost.conf.master  
/usr/local/ispconfig/server/conf-custom/vhost.conf.master
```

- d. Editer le fichier dans la zone custom. Tapez:

```
vi /usr/local/ispconfig/server/conf-custom/vhost.conf.master
```

- e. Remplacez la ligne **SSLProtocol All** par:

```
SSLProtocol All -SSLv2 -SSLv3 -TLSv1 -TLSv1.1
```

- 9. L'installation est terminée. Vous accédez au serveur à l'adresse: <https://example.com:8080/>.



Lors de votre première connexion, votre domaine n'est pas encore configuré. Il faudra alors utiliser le nom DNS donné par votre hébergeur. Pour OVH, elle s'écrit VPSxxxxxxx.ovh.net

- 10. Loguez vous comme admin et avec le mot de passe que vous avez choisi. Vous pouvez décider de le changer au premier login



Si le message "Possible attack detected. This action has been logged.". Cela signifie que vous avez des cookies d'une précédente installation qui sont configurés. Effacer les cookies de ce site de votre navigateur.

8.2. Installation de Webmin

Webmin est un outil généraliste de configuration de votre serveur. Son usage peut être assez complexe mais il permet une configuration plus précise des fonctionnalités.

1. Loguez vous comme root sur le serveur

2. Ajoutez le repository Webmin

a. allez dans le répertoire des repositories. Tapez :

```
cd /etc/apt/sources.list.d
```

b. Tapez :

```
echo "deb http://download.webmin.com/download/repository sarge contrib" >>  
webmin.list
```

c. Ajoutez la clé. Tapez :

```
curl -fsSL http://www.webmin.com/jcameron-key.asc | sudo apt-key add -
```

Le message OK s'affiche

3. Mise à jour. Tapez :

```
apt update
```

4. Installation de Webmin. Tapez :

```
apt install webmin
```

Débloquez le port 10000 dans votre firewall

a. Allez sur le site ispconfig <https://example.com:8080/>

b. Loguez-vous et cliquez sur la rubrique **System** et le menu **Firewall**. Cliquez sur votre serveur.

c. dans la rubrique **Open TCP ports:**, ajoutez le port 10000

d. Cliquez sur **save**

5. Connectez vous avec votre navigateur sur l'url <https://<example.com>:10000>. Un message indique un problème de sécurité. Cela vient du certificat auto-signé. Cliquez sur 'Avancé' puis 'Accepter le risque et poursuivre'.

6. Loguez-vous **root**. Tapez le mot de passe de **root**. Le dashboard s'affiche.

7. Restreignez l'adressage IP

a. Obtenez votre adresse IP en allant par exemples sur le site <https://www.showmyip.com/>

b. Sur votre URL Webmin ou vous êtes logué, allez dans Webmin → Webmin Configuration

- c. Dans l'écran choisir l'icône **Ip Access Control**.
 - d. Choisissez **Only allow from listed addresses**
 - e. Puis dans le champ **Allowed IP addresses** tapez votre adresse IP récupérée sur showmyip
 - f. Cliquez sur **Save**
 - g. Vous devriez avoir une brève déconnexion le temps que le serveur Webmin redémarre puis une reconnexion.
8. Si vous n'arrivez pas à vous reconnecter c'est que l'adresse IP n'est pas la bonne. Le seul moyen de se reconnecter est de:
- a. **Loguez vous comme root sur le serveur**
 - b. Éditez le fichier `/etc/webmin/miniserv.conf` et supprimez la ligne `allow= ...`
 - c. Tapez :
- ```
service webmin restart
```
- d. Connectez vous sur l'url de votre site Webmin. Tout doit fonctionner
9. Passez en Français. Pour les personnes non anglophone. Les traductions française ont des problèmes d'encodage de caractère ce n'est donc pas recommandé. La suite de mon tutoriel suppose que vous êtes resté en anglais.
- a. Sur votre url Webmin ou vous êtes logué, allez dans Webmin → Webmin Configuration
  - b. Dans l'écran choisir l'icône **Language and Locale**.
  - c. Choisir **Display Language** à **French (FR.UTF-8)**

# Chapter 9. Configuration d'un domaine

Cette configuration est réalisée avec le Panel ISPConfig installé dans le chapitre précédent. L'étape "login initial" n'est à appliquer qu'une seule fois. Une fois votre premier domaine configuré, vous pourrez vous connecter à ISPconfig en utilisant ce domaine à l'adresse: <https://example.com:8080/>.

## 9.1. Login initial



Cette procédure n'est à appliquer que lorsqu'aucun domaine n'est encore créé.

Vous devrez tout d'abord vous connecter sur le serveur ISPConfig. Comme vous n'avez pas encore configuré de nom de domaine, vous devrez vous connecter de prime abord sur le site <http://vpsxxxxxx.ovh.net:8080/> pour un vps chez ovh par exemple ou sur <http://raspberrypi.local:8080/> pour un Raspberry.

Utiliser le login: Admin et le mot de passe que vous avez configuré lors de l'installation d'ISPConfig

1. Aller dans la rubrique **System**
  - a. Dans le menu **Main config**
    - i. Dans l'onglet **Sites**, configurer:
      - A. **Create subdomains as web site:** ← Yes
      - B. **Create aliasdomains as web site:** ← Yes
    - ii. Dans l'onglet **Mail** :
      - A. **Administrator's e-mail** : ← adresse mail de l'administrateur. par exemple [admin@example.com](mailto:admin@example.com)
      - B. **Administrator's name** : ← nom de l'administrateur
  - b. Dans le menu **Firewall**
    - i. Cliquez sur **Add Firewall Record**
    - ii. Acceptez les valeurs par défaut en cliquant sur **Save**



Il est possible de basculer le site ISPConfig entièrement en Français. J'ai pour ma part gardé la version anglaise du site. Vous trouverez donc tous les libellés dans la suite de la documentation en anglais.

2. Aller dans la rubrique **DNS**
  - a. Dans le menu **Template**
    - i. Cliquez sur **Add new record**
    - ii. Remplissez les champs comme ci-après:
      - **Name** ← Tapez **Template IPV4 autoNS**
      - **Fields** ← Cochez **Domain, IP Address, Email, DKIM, DNSSEC**
      - **Template** ← remplissez comme ci dessous:

```

[ZONE]
origin={DOMAIN}.
ns=ns1.{DOMAIN}.
mbox={EMAIL}.
refresh=7200
retry=540
expire=604800
minimum=3600
ttl=3600

[DNS_RECORDS]
A|{DOMAIN}.|{IP}|0|3600
A|www|{IP}|0|3600
A|mail|{IP}|0|3600
A|autoconfig|{IP}|0|3600
A|autodiscover|{IP}|0|3600
A|webmail|{IP}|0|3600
A|ns1|{IP}|0|3600
CNAME|ftp|{DOMAIN}|0|3600
CNAME|smtp|{DOMAIN}|0|3600
CNAME|pop3|{DOMAIN}|0|3600
CNAME|imap|{DOMAIN}|0|3600
SRV|_pop3._tcp|0 0 .|0|3600
SRV|_imap._tcp|0 0 .|0|3600
SRV|_pop3s._tcp|1 995 mail.{DOMAIN}|0|3600
SRV|_imaps._tcp|1 993 mail.{DOMAIN}|0|3600
SRV|_submission._tcp|1 465 mail.{DOMAIN}|0|3600
SRV|_autodiscover._tcp|1 443 autodiscover.{DOMAIN}|0|3600
NS|{DOMAIN}.|ns1.{DOMAIN}.|0|3600
MX|{DOMAIN}.|mail.{DOMAIN}.|10|3600
TXT|{DOMAIN}.|v=spf1 mx a ~all|0|3600

```

iii. Cliquez sur **Save**

iv. Cliquez sur **Add new record**

v. Remplissez les champs comme ci-après:

- **Name** ← Tapez **Template IPV6 autoNS**
- **Fields** ← Cochez **Domain, IP Address, IPV6 Address, Email, DKIM, DNSSEC**
- **Template** ← remplissez comme ci dessous:

```

[ZONE]
origin={DOMAIN}.
ns=ns1.{DOMAIN}.
mbox={EMAIL}.
refresh=7200
retry=540
expire=604800
minimum=3600
ttl=3600

[DNS_RECORDS]
A|{DOMAIN}.|{IP}|0|3600
A|www|{IP}|0|3600
A|mail|{IP}|0|3600
A|autoconfig|{IP}|0|3600
A|autodiscover|{IP}|0|3600
A|webmail|{IP}|0|3600
A|ns1|{IP}|0|3600
AAAA|{DOMAIN}.|{IPV6}|0|3600
AAAA|www|{IPV6}|0|3600
AAAA|mail|{IPV6}|0|3600
AAAA|autoconfig|{IPV6}|0|3600
AAAA|autodiscover|{IPV6}|0|3600
AAAA|webmail|{IPV6}|0|3600
AAAA|ns1|{IPV6}|0|3600
CNAME|ftp|{DOMAIN}|0|3600
CNAME|smtp|{DOMAIN}|0|3600
CNAME|pop3|{DOMAIN}|0|3600
CNAME|imap|{DOMAIN}|0|3600
SRV|_pop3._tcp|0 0 .|0|3600
SRV|_imap._tcp|0 0 .|0|3600
SRV|_pop3s._tcp|1 995 mail.{DOMAIN}|0|3600
SRV|_imaps._tcp|1 993 mail.{DOMAIN}|0|3600
SRV|_submission._tcp|1 465 mail.{DOMAIN}|0|3600
SRV|_autodiscover._tcp|1 443 autodiscover.{DOMAIN}|0|3600
NS|{DOMAIN}.|ns1.{DOMAIN}.|0|3600
MX|{DOMAIN}.|mail.{DOMAIN}.|10|3600
TXT|{DOMAIN}.|v=spf1 mx a ~all|0|3600

```

## 9.2. Création de la zone DNS d'un domaine

1. Allez dans **DNS**
  - a. Cliquez sur **Add dns-zone**
  - b. Cliquez sur **Dns zone wizard**
  - c. Choisir le template **IPV4 autoNS** ou **IPV6 autoNS** selon que vous soyez IPV4 ou IPV4+V6
  - d. Remplissez les champs:



- **Domain :** ← tapez le nom de votre domaine [example.com](#)
- **IP Address:** ← prendre l'adresse IPV4 du serveur sélectionnée
- **IPV6 Address:** ← prendre l'adresse IPV6 du serveur sélectionnée
- **Email:** ← votre Email valide exemple [admin@example.com](#)
- **DKIM:** ← Yes



Si votre serveur est chez vous, il est probablement installé derrière un routeur ADSL configuré au préalable avec une DMZ qui pointe sur ce serveur. Dans ce cas, vous ne devrez pas indiquer l'adresse IP locale de votre serveur mais l'adresse IP de votre routeur ADSL telle qu'elle est vue sur internet. On suppose aussi que cette adresse IP est statique et non pas allouée dynamiquement par l'opérateur.

e. Cliquez sur **Create DNS-record**

Attendez quelques minutes le temps que les enregistrements DNS se propagent et faites une essai de votre nom de domaine sur le site [ZoneMaster](#).

Dans le champ Nom de domaine saisissez votre nom de domaine et tapez sur check. Tout doit être OK sauf pour les serveurs de noms ns1 et ns2. Si ce n'est pas le cas, votre nom de domaine doit être mal configuré chez votre registrar. Il vous faut vérifier la configuration initiale.



Zonemaster a bien repéré que l'on a essayé de mettre des noms de host différents pour les serveurs de DNS. Ils ont cependant tous la même adresse IP. Cela apparaît comme une erreur suite au test. De la même manière, il indique dans la rubrique connectivité qu'il n'y a pas de redondance de serveur DNS. Une manière de corriger ce problème est de définir un DNS secondaire chez OVH en utilisant le service qu'ils mettent à disposition.

Vous pouvez maintenant essayer les différents Hostname munis de leur nom de domaine dans votre navigateur. Par exemple: <http://webmail.example.com>

Ils doivent afficher une page web basique (Apache2, ou de parking). Si ce n'est pas le cas revérifier la configuration du DNS dans ISPConfig.

## 9.3. Activation de DNSSEC

Vous pouvez maintenant activer DNSSEC afin d'augmenter la sécurité de résolution de nom de domaine:

1. Allez dans la rubrique **DNS**
  - a. puis dans le menu **Zones**
  - b. choisissez la zone correspondant à votre domaine
  - c. dans l'onglet **DNS Zone** allez tout en bas et activer la coche **Sign Zone (DNSSEC)**
  - d. cliquez sur **Save**

- e. Une fois fait, retourner dans le même onglet. La boîte `DNSSEC DS-Data for registry` contient les informations que vous devez coller dans le site web de votre registrar pour sécuriser votre zone.
- f. Gardez cette fenêtre ouverte dans votre navigateur et ouvrez un autre onglet sur le site de votre registrar.

Si vous êtes chez [Gandi](#), il vous faut:

1. Sélectionner le menu **nom de domaine**
2. Choisir votre nom de domaine "example.com"
3. Allez dans l'onglet DNSSEC. Il doit permettre d'ajouter des clés puisque vous fonctionnez avec des DNS externes.
4. Effacez éventuellement toutes les clés si vous n'êtes pas sûr de celles-ci.
5. puis cliquez sur **Ajouter une clé externe**
  - a. Sélectionnez d'abord le flag **257 (KSK)**. puis l'algorithme **7 (RSASHA1-NSEC3-SHA1)**
  - b. Collez ensuite la clé de votre site ISPConfig. Elle doit ressembler à cela:

```
example.com. IN DNSKEY 257 3 7
AwEAAcs+xTC5GlyC8CSufM9U7z5uazLNmNP3vG2txzNIGM1VJHWCpRYQVZjsBZqx5vZu0FBwp0F6cpF8
YdW9QibZc82UAeIYAstgRSwnCLYsIV+3Zq0NpCcnGtKPLknxxZuN3MD5tARKxBM5c5fME0NgMU+kcx4x
aTVm2Go6bEeFuhgNfRogzXKqLV6h2bMCajudfJbbTbJlehyM2YegLI+yYCpYr6b+jWHorRoUVDJ410PX
Ltz2s8wtycyINpZsdmLNJhNNaeGqOok3+c5uazLNmNP3vG2txzNIGLM1VJHWCpRYQVZjsBZkqx5vZu0F
Bgwp0F6cpF8YdW9QibZc82UAeIYAstKgRSwnCLYsIV+3Zq0NpCcnGtKPLkn
```

- c. Cliquez sur **Ajouter**
- d. Entrez la deuxième clé. Cliquez sur **Ajouter une clé externe**
- e. Sélectionnez d'abord le flag **256 (ZSK)**. puis l'algorithme **7 (RSASHA1-NSEC3-SHA1)**
- f. Collez ensuite la clé de votre site ISPConfig. Elle doit ressembler à cela:

```
example.com. IN DNSKEY 256 3 7
AwEAAcs+xTC5GlyC8CSufM9U7z5uazLNmNP3vG2txzNIGM1VJHWCpRYQVZjsBZqx5vZu0FBwp0F6cpF8
YdW9QibZc82UAeIYAstgRSwnCLYsIV+3Zq0NpCcnGtKPLknxxZuN3MD5tARKxBM5c5fME0NgMU+kcx4x
aTVm2Go6bEeFuhgNfRogzXKqLV6h2bMCajudfJbbTbJlehyM2YegLI+yYCpYr6b+jWHorRoUVDJ410PX
Ltz2s8wtycyINpZsdmLNJhNNaeGqOok3+c5uazLNmNP3vG2txzNIGLM1VJHWCpRYQVZjsBZkqx5vZu0F
Bgwp0F6cpF8YdW9QibZc82UAeIYAstKgRSwnCLYsIV+3Zq0NpCcnGtKPLkn
```

- g. Cliquez sur **Ajouter**
- h. Les deux clés doivent maintenant apparaître dans l'onglet **DNSSEC**
- i. Vous devez attendre quelques minutes (une heure dans certains cas) pour que les clés se propagent. Pendant ce temps vous pouvez avoir quelques problèmes d'accès à vos sites webs
- j. Allez sur le site [DNSSEC Analyzer](#).

k. Entrez votre nom de domaine "example.com" et tapez sur "entrée".

Le site doit afficher pour les différentes zones le statut des certificats. Tout doit être au vert. Si ce n'est pas le cas, réessayer dans une heure. S'il y a encore des problèmes vérifiez votre configuration dans ISPConfig, chez votre registrar (rubrique DNSSEC) ou regardez les logs d'ISPConfig sur votre serveur pour y débusquer une erreur.



Une erreur classique est de croiser les certificats avec leurs types. Vérifiez bien que vous avez mis les bons certificats avec les bons types.



Une fois que vous activez DNSSEC, vous pourriez faire face au problème suivant: les nouveaux enregistrements que vous renseignez ne sont pas actifs. Une analyse des logs montre que la commande `dnssec-signzone` retourne l'erreur **fatal: 'example.com': found DS RRset without NS RRset**. Cela signifie que vous avez saisi une ou deux entrées DS dans vos enregistrements. Il faut les supprimer pour que tout redevienne fonctionnel.

## 9.4. Exemple de configuration de domaine

Une fois la configuration terminée, les différents enregistrements du domaine ressemblent à l'exemple ci-dessous. Il peut y avoir des enregistrements supplémentaires pour les configurations SPF, DKIM et Let's encrypt.

|                    |      |       |          |                         |
|--------------------|------|-------|----------|-------------------------|
| example.com.       | 3600 | A     |          | 1.2.3.4                 |
| www                | 3600 | A     |          | 1.2.3.4                 |
| mail               | 3600 | A     |          | 1.2.3.4                 |
| ns1                | 3600 | A     |          | 1.2.3.4                 |
| ns2                | 3600 | A     |          | 1.2.3.4                 |
| webmail            | 3600 | A     |          | 1.2.3.4                 |
| autoconfig         | 3600 | A     |          | 1.2.3.4                 |
| autodiscover       | 3600 | A     |          | 1.2.3.4                 |
| ftp                | 3600 | CNAME |          | example.com.            |
| smtp               | 3600 | CNAME |          | mail.example.com.       |
| pop3               | 3600 | CNAME |          | mail.example.com.       |
| imap               | 3600 | CNAME |          | mail.example.com.       |
| example.com.       | 3600 | NS    |          | ns1.example.com.        |
| example.com.       | 3600 | NS    |          | ns2.example.com.        |
| example.com.       | 3600 | MX    | 10       | mail.example.com.       |
| _pop3s._tcp        | 3600 | SRV   | 10 1 995 | mail.example.com.       |
| _imaps._tcp        | 3600 | SRV   | 0 1 993  | mail.example.com.       |
| _submission._tcp   | 3600 | SRV   | 0 1 465  | mail.example.com.       |
| _imap._tcp         | 3600 | SRV   | 0 0 0    | .                       |
| _pop3._tcp         | 3600 | SRV   | 0 0 0    | .                       |
| _autodiscover._tcp | 3600 | SRV   | 0 0 443  | autoconfig.example.com. |
| example.com.       | 3600 | TXT   |          | "v=spf1 mx a ~all"      |

## 9.5. Création d'un sous domaine

Supposons que vous êtes en train de créer un sous domain nommé `sub.example.com` . Dans ce sous domaines vous allez créer un ensemble de site web par exemple `mail.sub.example.com` ou `blog.sub.example.com`.

Un cas assez classique est que ce sous domaine est délégué à une machine tierce.

Par exemple: `example.com` est installé sur un VPS quelque part sur internet et `sub.example.com` est hébergé chez vous sur votre Raspberry.

On suppose que votre domain a été configuré en suivant la procédure du chapitre précédent.

Rien de bien sorcier pour votre sous domaine: Vous devez le créer sur votre Raspberry selon la même procédure mais avec le nom du sous domaine (`sub.example.com` donc).

Vous aurez des actions complémentaires à effectuer sur votre domaine:

1. Allez dans **DNS** de votre serveur de domaine principal
2. Sélectionner le menu **Zones** puis le domaine `example.com`
3. Choisissez l'onglet **Records** et créez:
  - un enregistrement de type **NS** avec une **Zone** ← `sub.example.com.` et un **nameserver Hostname** ← `ns1.sub.example.com.`
  - un enregistrement de type **NS** avec une **Zone** ← `sub.example.com.` et un **nameserver Hostname** ← `ns2.sub.example.com.`
  - un enregistrement de type **NS** avec une **Zone** ← `sub.example.com.` et un **nameserver Hostname** ← `ns3.example.com.` .

Ce dernier type d'enregistrement se nomme un Glue record pour faire le lien vers le serveur secondaire.

- un enregistrement de type **A** avec un **Hostname** ← `ns3` et une **IP-address** ← Adresse IP de votre routeur ADSL ou est connecté le Raspberry.
- Si vous ne la connaissez pas, tapez dans un terminal texte:

```
wget -q0- http://ipecho.net/plain; echo
```

Ce dernier enregistrement en complétant le Glue record fait le lien avec l'adresse IP de `sub.example.com`

4. Si vous avez activé DNSSEC sur votre serveur DNS de `sub.example.com` vous devrez récupérer les entrées DS du champ **DNSSEC DS-Data for registry** de votre domaine `sub.example.com` et créer dans votre domaine `example.com` les deux entrées suivantes:
  - un enregistrement de type **DS** avec une **Zone** ← `sub.example.com` et un champ **data** contenant `xxxxx 7 1 <votre_digest_recupérée>`
  - un enregistrement de type **DS** avec une **Zone** ← `sub.example.com` et un champ **data** contenant

xxxxx 7 2 <votre\_digest\_recupérée>

5. Allez sur le site [DNSSEC Analyzer](#).
6. Entrez votre nom de domaine "sub.example.com" et tapez sur "entrée".

Le site doit afficher pour les différentes zones le statut des certificats. Tout doit être au vert. Si ce n'est pas le cas, réessayer dans une heure. S'il y a encore des problèmes vérifiez votre configuration dans ISPConfig de votre domaine et de votre sous-domaine, chez votre registrar (rubrique DNSSEC) ou regardez les logs d'ISPConfig sur votre serveur pour y débusquer une erreur.

## 9.6. Création d'un site web

Dans la suite le site web sera nommé "example.com".

Vous devez avoir avant tout défini le "record" DNS associé au site.

1. Aller dans "Sites"
  - a. Aller dans le menu "Website" pour définir un site web
    - i. Cliquez sur "Add new website"
    - ii. Saisissez les informations:
      - **Domain:** ← mettre `example.com`
      - **Auto-subdomain:** ← sélectionner `www` ou `*` si l'on veut un certificat let's encrypt wildcard
      - **SSL:** ← yes
      - **Let's Encrypt:** ← yes
      - **Php:** ← Sélectionnez `php-fpm`
      - Sélectionnez éventuellement aussi les coches `Perl`, `Python`, `Ruby` en fonction des technologies déployées sur votre site. Cela est indiqué dans la procédure d'installation du site.
    - iii. Dans l'onglet `redirect` du même écran
      - **SEO Redirect:** ← Sélectionner `domain.tld ⇒ www.domain.tld`
      - **Rewrite http to https:** ← yes
    - iv. Dans l'onglet `Statistics` du même écran
      - **Set Webstatistics password:** ← saisissez un mot de passe
      - **Repeat Password:** ← ressaisissez le mot de passe
    - v. Dans l'onglet `Backup` du même écran
      - **Backup interval:** ← saisir `weekly`
      - **Number of backup copies:** ← saisir `1`
    - vi. Dans l'onglet `Options`, il peut être utile pour certains types de site qui sont des redirections d'autres sites de saisir dans la zone **Apache Directives:**

```

ProxyPass "/.well-known/acme-challenge" http://localhost:80/.well-known/acme-challenge
ProxyPassReverse "/.well-known/acme-challenge" http://localhost:80/.well-known/acme-challenge
RewriteRule ^/.well-known/acme-challenge - [QSA,L]

redirect from server
#

SetEnvIf Authorization "(.*)" HTTP_AUTHORIZATION=$1
ProxyPass / http://localhost[:port_number_if_any]/[path_if_any]
ProxyPassReverse / http://localhost[:port_number_if_any]/[path_if_any]

```

2. Vous pouvez maintenant tester la qualité de la connexion de votre site en allant sur: [SSL Server Test](#). Saisissez votre nom de domaine et cliquez sur [Submit](#). Votre site doit au moins être de [Grade A](#).

## 9.7. Création d'un Site Vhost

Dans la suite le sous-domaine sera nommé "mail.example.com".

Vous devez avoir avant tout défini le "record" DNS associé au site. Vous ne pouvez définir un sous-domaine que si vous avez défini le site web racine auparavant.

1. Aller dans "Sites"
  - a. Aller dans le menu "Subdomain(vhost)" pour définir un sous-domaine
    - i. Cliquez sur "Add Subdomain" pour un nouveau sous domaine
    - ii. Saisissez les informations:
      - **Hostname:** ← saisir [mail](#)
      - **Domain:** ← mettre [example.com](#)
      - **web folder:** ← saisir [mail](#)
      - **Auto-subdomain:** ← sélectionner [www](#) ou [\\*](#) si l'on veut un certificat let's encrypt wildcard
      - **SSL:** ← yes
      - **Let's Encrypt:** ← yes
      - **Php:** ← Sélectionnez [php-fpm](#)
      - Sélectionnez éventuellement aussi les coches [Perl](#), [Python](#), [Ruby](#) en fonction des technologies déployées sur votre site. Cela est indiqué dans la procédure d'installation du site.
    - iii. Dans l'onglet [redirect](#) du même écran
      - **Rewrite http to https:** ← yes
    - iv. Dans l'onglet [Statistics](#) du même écran

- **Set Webstatistics password:** ← saisissez un mot de passe
  - **Repeat Password:** ← ressaisissez le mot de passe
- v. Dans l'onglet **Options**, il peut être utile pour certains types de site qui sont des redirections d'autres sites de saisir dans la zone **Apache Directives**:

```
ProxyPass "/.well-known/acme-challenge" http://localhost:80/.well-known/acme-challenge
ProxyPassReverse "/.well-known/acme-challenge" http://localhost:80/.well-known/acme-challenge
RewriteRule ^/.well-known/acme-challenge - [QSA,L]

redirect from server
#

SetEnvIf Authorization "(.*)" HTTP_AUTHORIZATION=$1
ProxyPass / http://localhost[:port_number_if_any]/[path_if_any]
ProxyPassReverse / http://localhost[:port_number_if_any]/[path_if_any]
```

2. Vous pouvez maintenant tester la qualité de la connexion de votre site en allant sur: [SSL Server Test](#). Saisissez votre nom de domaine et cliquez sur **Submit**. Votre site doit au moins être de **Grade A**.

# Chapter 10. Associer des certificats reconnu à vos outils

Cette action est à effectuer une fois que vous avez créé votre domaine principal et que vous avez généré vos premiers certificats let's encrypt dans ISPConfig, vous pouvez maintenant, affecter ce certificat aux services de base:

1. Vous devez avoir créé au préalable un site pour les domaines example.com et mail.example.com
2. [Loguez vous comme root sur le serveur](#)
3. Liez le certificat d'ISPconfig avec celui du domaine crée.

◦ Tapez :

```
cd /usr/local/ispconfig/interface/ssl/
mv ispserver.crt ispserver.crt-$(date +"%y%m%d%H%M%S").bak
mv ispserver.key ispserver.key-$(date +"%y%m%d%H%M%S").bak
ln -s /etc/letsencrypt/live/example.com/fullchain.pem ispserver.crt ①
ln -s /etc/letsencrypt/live/example.com/privkey.pem ispserver.key ①
cat ispserver.{key,crt} > ispserver.pem
chmod 600 ispserver.pem
systemctl restart apache2
```

① remplacer <example.com> par votre nom de domaine

4. Liez le certificat Postfix et Dovecot avec celui de let's encrypt

◦ Tapez :

```
cd /etc/postfix/
mv smtpd.cert smtpd.cert-$(date +"%y%m%d%H%M%S").bak
mv smtpd.key smtpd.key-$(date +"%y%m%d%H%M%S").bak
ln -s /etc/letsencrypt/live/mail.example.com/fullchain.pem smtpd.cert ①
ln -s /etc/letsencrypt/live/mail.example.com/privkey.pem smtpd.key ①
service postfix restart
service dovecot restart
```

① remplacer <example.com> par votre nom de domaine

5. Liez le certificat pour Pureftd

◦ Tapez :

```
cd /etc/ssl/private/
mv pure-ftp.pem pure-ftp.pem-$(date +"%y%m%d%H%M%S").bak
ln -s /usr/local/ispconfig/interface/ssl/ispserver.pem pure-ftp.pem
chmod 600 pure-ftp.pem
service pure-ftp-mysql restart
```



## 6. Création d'un script de renouvellement automatique du fichier pem

### a. Installez incron. Tapez :

```
apt install -y incron
```

### b. Créez le fichier d'exécution périodique. Tapez :

```
vi /etc/init.d/le_ispc_pem.sh
```

et coller dans le fichier le code suivant:

```
#!/bin/sh
BEGIN INIT INFO
Provides: LE ISPSERVER.PEM AUTO UPDATER
Required-Start: $local_fs $network
Required-Stop: $local_fs
Default-Start: 2 3 4 5
Default-Stop: 0 1 6
Short-Description: LE ISPSERVER.PEM AUTO UPDATER
Description: Update ispserver.pem automatically after ISPC LE SSL certs are
renewed.
END INIT INFO
cd /usr/local/ispconfig/interface/ssl/
mv ispserver.pem ispserver.pem-$(date +"%y%m%d%H%M%S").bak
cat ispserver.{key,crt} > ispserver.pem
chmod 600 ispserver.pem
chmod 600 /etc/ssl/private/pure-ftpd.pem
service pure-ftpd-mysql restart
service monit restart
service postfix restart
service dovecot restart
service apache2 restart
exit 1
```

### c. Sauvez et quittez. Tapez ensuite:

```
chmod +x /etc/init.d/le_ispc_pem.sh
echo "root" >> /etc/incron.allow
incrontab -e.
```

et ajoutez les lignes ci dessous dans le fichier:

```
/etc/letsencrypt/archive/example.com/ IN_MODIFY /etc/init.d/le_ispc_pem.sh ①
```

① Remplacer example.com par votre nom de domaine.

# Chapter 11. Surveillance du serveur avec Munin et Monit

## 11.1. Note préliminaire

Installez tout d'abord les paquets indispensables pour faire fonctionner Munin avec Apache puis activez le module fcgid:

```
apt-get install apache2 libcgi-fast-perl libapache2-mod-fcgid
a2enmod fcgid
```

## 11.2. Installation et configuration de Munin

Suivez les étapes ci-après:

1. Installer le paquet Munin:

```
apt-get install munin munin-node munin-plugins-extra logtail libcache-cache-perl
```

2. Votre configuration de Munin va utiliser une base de données MariaDB. Vous devez activer quelques plugins. Tapez:

```
cd /etc/munin/plugins
ln -s /usr/share/munin/plugins/mysql_ mysql_
ln -s /usr/share/munin/plugins/mysql_bytes mysql_bytes
ln -s /usr/share/munin/plugins/mysql_innodb mysql_innodb
ln -s /usr/share/munin/plugins/mysql_isam_space_ mysql_isam_space_
ln -s /usr/share/munin/plugins/mysql_queries mysql_queries
ln -s /usr/share/munin/plugins/mysql_slowqueries mysql_slowqueries
ln -s /usr/share/munin/plugins/mysql_threads mysql_threads
```

3. Créez la base de données MariaDB de Munin. Tapez:

```
mysql -p
```

4. Tapez le mot de passe mysql de root , puis dans mysql tapez:

```
CREATE SCHEMA munin_innodb;
USE munin_innodb
CREATE TABLE something (anything int) ENGINE=InnoDB;
GRANT SELECT ON munin_innodb.* TO 'munin'@'localhost' IDENTIFIED BY 'munin';
FLUSH PRIVILEGES;
EXIT;
```

5. Editez ensuite le fichier de configuration de Munin. Tapez:

```
vi /etc/munin/munin.conf
```

6. Décommentez les lignes débutant par: `bdir`, `htmldir`, `logdir`, `rundir`, and `tmpdir`. Les valeurs par défaut sont correctes.
7. Munin utilisera l'adresse `munin.example.com`. Toujours dans le fichier de configuration de munin, remplacer la directive `[localhost.localdomain]` par `[munin.example.com]`.
8. Un fois les commentaires enlevés et la ligne modifiée, le fichier de configuration doit ressembler à celui-ci:

```
Example configuration file for Munin, generated by 'make build'
The next three variables specifies where the location of the RRD
databases, the HTML output, logs and the lock/pid files. They all
must be writable by the user running munin-cron. They are all
defaulted to the values you see here.
#
dbdir /var/lib/munin
htmldir /var/cache/munin/www
logdir /var/log/munin
rundir /var/run/munin
Where to look for the HTML templates
#
tmpldir /etc/munin/templates
Where to look for the static www files
#
#staticdir /etc/munin/static
temporary cgi files are here. note that it has to be writable by
the cgi user (usually nobody or httpd).
#
cgitmpdir /var/lib/munin/cgi-tmp

(Exactly one) directory to include all files from.
includedir /etc/munin/munin-conf.d
[...]
a simple host tree
[server1.example.com]
 address 127.0.0.1
 use_node_name yes
[...]
```

9. Activez Munin dans Apache. Tapez:

```
a2enconf munin
```

10. Editez le fichier munin.conf d'Apache:

```
vi /etc/apache2/conf-enabled/munin.conf
```

11. Nous allons maintenant activer le module Munin dans Apache et définir une authentification basique.

12. Modifiez le fichier pour qu'il ressemble à celui ci-dessous:

```
ScriptAlias /munin-cgi/munin-cgi-graph /usr/lib/munin/cgi/munin-cgi-graph
Alias /munin/static/ /var/cache/munin/www/static/

<Directory /var/cache/munin/www>
 Options FollowSymLinks SymLinksIfOwnerMatch
 AuthUserFile /etc/munin/munin-htpasswd
 AuthName "Munin"
 AuthType Basic
 Require valid-user

</Directory>

<Directory /usr/lib/munin/cgi>
 AuthUserFile /etc/munin/munin-htpasswd
 AuthName "Munin"
 AuthType Basic
 Require valid-user
 Options FollowSymLinks SymLinksIfOwnerMatch
 <IfModule mod_fcgid.c>
 SetHandler fcgid-script
 </IfModule>
 <IfModule !mod_fcgid.c>
 SetHandler cgi-script
 </IfModule>
</Directory>

***** SETTINGS FOR CGI/CRON STRATEGIES *****

pick _one_ of the following lines depending on your "html_strategy"
html_strategy: cron (default)
Alias /munin /var/cache/munin/www
html_strategy: cgi (requires the apache module "cgid" or "fcgid")
#ScriptAlias /munin /usr/lib/munin/cgi/munin-cgi-html
```

13. Créez ensuite le fichier de mot de passe de munin:

```
htpasswd -c /etc/munin/munin-htpasswd admin
```

14. Tapez votre mot de passe

15. Redémarrez apache. Tapez:

```
service apache2 restart
```

16. Redémarrez Munin. Tapez:

```
service munin-node restart
```

17. Attendez quelques minutes afin que Munin produise ses premiers fichiers de sortie. et allez ensuite sur l'URL: <http://example.com/munin/>.

## 11.3. Activez les plugins de Munin

Dans Debian 10, tous les plugins complémentaires sont déjà activés. Vous pouvez être tenté de vérifier:

1. Pour vérifier que la configuration est correcte. Tapez:

```
munin-node-configure --suggest
```

2. Une liste de plugins doit s'afficher à l'écran. La colonne **used** indique que le plugin est activé. La colonne **Suggestions** indique que le serveur fait fonctionner un service qui peut être monitoré par ce module. Il faut créer un lien symbolique du module de **/usr/share/munin/plugins** dans **/etc/munin/plugins** pour l'activer.
3. Par exemple pour activer les modules `apache_*`:

```
cd /etc/munin/plugins
ln -s /usr/share/munin/plugins/apache_accesses
ln -s /usr/share/munin/plugins/apache_processes
ln -s /usr/share/munin/plugins/apache_volume
rm /usr/share/munin/plugins/mysql_
```

4. Redémarrez ensuite le service Munin. Tapez:

```
service munin-node restart
```

## 11.4. Installer et configurer Monit

Pour installer et configurer Monit, vous devez appliquer la procédure suivante:

1. Tapez:

```
apt install monit
```

2. Maintenant nous devons éditer le fichier **monitrc** qui définira les services que l'on souhaite monitorer. Il existe de nombreux exemples sur le web et vous pourrez trouver de nombreuses configurations sur <http://mmonit.com/monit/documentation/>.
3. Editez le fichier `monitrc`. Tapez:

```
cp /etc/monit/monitrc /etc/monit/monitrc_orig
vi /etc/monit/monitrc
```

4. Le fichier contient déjà de nombreux exemples. Nous configurer une surveillance de sshd, apache, mysql, proftpd, postfix, memcached, named, ntpd, mailman, amavisd, dovecot. Monit sera activé sur le port 2812 et nous allons donner à l'utilisateur admin un mot de passe. Le certificat HTTPS sera celui généré avec let's encrypt pour le site ISPConfig. Collez le contenu ci dessous dans le fichier monitrc:

```
set daemon 60
set logfile syslog facility log_daemon
set mailserver localhost
set mail-format { from: monit@fpvview.site }
set alert stef@fpvview.site
set httpd port 2812 and
 SSL ENABLE
 PEMFILE /usr/local/ispconfig/interface/ssl/ispserver.pem
 allow admin:"my_password" ①

check process sshd with pidfile /var/run/sshd.pid
 start program "/usr/sbin/service ssh start"
 stop program "/usr/sbin/service ssh stop"
 if failed port 22 protocol ssh then restart
 if 5 restarts within 5 cycles then timeout

check process apache with pidfile /var/run/apache2/apache2.pid
 group www
 start program = "/usr/sbin/service apache2 start"
 stop program = "/usr/sbin/service apache2 stop"
 if failed host localhost port 80 protocol http
 and request "/monit/token" then restart
 if cpu is greater than 60% for 2 cycles then alert
 if cpu > 80% for 5 cycles then restart
 if totalmem > 500 MB for 5 cycles then restart
 if children > 250 then restart
 if loadavg(5min) greater than 10 for 8 cycles then stop
 if 3 restarts within 5 cycles then timeout

#

NOTE: Replace example.pid with the pid name of your server, the name depends on
the hostname
#

check process mysql with pidfile /var/run/mysqld/mysqld.pid
 group database
 start program = "/usr/sbin/service mysql start"
```

```

stop program = "/usr/sbin/service mysql stop"
if failed host 127.0.0.1 port 3306 then restart
if 5 restarts within 5 cycles then timeout

check process pureftpd with pidfile /var/run/pure-ftpd/pure-ftpd.pid
start program = "/usr/sbin/service pure-ftpd-mysql start"
stop program = "/usr/sbin/service pure-ftpd-mysql stop"
if failed port 21 protocol ftp then restart
if 5 restarts within 5 cycles then timeout

check process postfix with pidfile /var/spool/postfix/pid/master.pid
group mail
start program = "/usr/sbin/service postfix start"
stop program = "/usr/sbin/service postfix stop"
if failed port 25 protocol smtp then restart
if 5 restarts within 5 cycles then timeout

check process memcached with pidfile /var/run/memcached/memcached.pid
start program = "/usr/sbin/service memcached start"
stop program = "/usr/sbin/service memcached stop"
if failed host 127.0.0.1 port 11211 then restart

check process named with pidfile /var/run/named/named.pid
start program = "/usr/sbin/service bind9 start"
stop program = "/usr/sbin/service bind9 stop"
if failed host 127.0.0.1 port 53 type tcp protocol dns then restart
if failed host 127.0.0.1 port 53 type udp protocol dns then restart
if 5 restarts within 5 cycles then timeout

check process ntpd with pidfile /var/run/ntpd.pid
start program = "/usr/sbin/service ntp start"
stop program = "/usr/sbin/service ntp stop"
if failed host 127.0.0.1 port 123 type udp then restart
if 5 restarts within 5 cycles then timeout

check process mailman with pidfile /var/run/mailman/mailman.pid
group mail
start program = "/usr/sbin/service mailman start"
stop program = "/usr/sbin/service mailman stop"

check process amavisd with pidfile /var/run/amavis/amavisd.pid
group mail
start program = "/usr/sbin/service amavis start"
stop program = "/usr/sbin/service amavis stop"
if failed port 10024 protocol smtp then restart
if 5 restarts within 5 cycles then timeout

check process dovecot with pidfile /var/run/dovecot/master.pid
group mail
start program = "/usr/sbin/service dovecot start"
stop program = "/usr/sbin/service dovecot stop"

```



```
if failed host localhost port 993 type tcpssl sslauto protocol imap then restart
if 5 restarts within 5 cycles then timeout
```

① remplacez my\_password par votre mot de passe

5. La configuration est assez claire à lire. pour obtenir des précisions, référez vous à la documentation de monit <http://mmonit.com/monit/documentation/monit.html>.
6. Dans la configuration pour apache, la configuration indique que monit doit aller chercher sur le port 80 un fichier dans `/monit/token`. Nous devons donc créer ce fichier. Tapez:

```
mkdir /var/www/html/monit
echo "hello" > /var/www/html/monit/token
```

7. Tapez :

```
service monit restart
```

8. Pour monitorer le statut des process en ligne de commande, tapez:

```
monit status
```

9. Débloquez le port 2812 dans votre firewall
  - a. Allez sur le site ispconfig <https://example.com:8080/>
  - b. Loguez-vous et cliquez sur la rubrique **System** et le menu **Firewall**. Cliquez sur votre serveur.
  - c. dans la rubrique **Open TCP ports:**, ajoutez le port 2812
  - d. Cliquez sur **save**
10. Maintenant naviguez sur le site <https://example.com:2812/>
11. Rentrez le login **admin** et votre mot de passe **my\_password**. Monit affiche alors les informations de monitoring du serveur.

# Chapter 12. Configuration de la messagerie

## 12.1. Installation de rspamd à la place d' Amavis-new

**rspamd** est réputé de meilleure qualité que **Amavis** dans la chasse aux spams. Vous pouvez décider de l'installer à la place d'Amavis. Cette installation reste optionnelle.

Suivez la procédure suivante:

1. [Loguez vous comme root sur le serveur](#)
2. Installez les paquets debian. tapez:

```
apt-get install rspamd redis-server
```

3. Activez l'apprentissage automatique

```
echo "autolearn = true;" > /etc/rspamd/local.d/classifier-bayes.conf
echo 'backend = "redis";' >> /etc/rspamd/local.d/classifier-bayes.conf
echo "new_schema = true;" >> /etc/rspamd/local.d/classifier-bayes.conf
echo "expire = 8640000;" >> /etc/rspamd/local.d/classifier-bayes.conf
```

4. Activez Redis dans la configuration de Rspamd. Tapez:

```
echo 'servers = "127.0.0.1";' > /etc/rspamd/local.d/redis.conf
```

5. Fixer des métriques assez élevées pour analyser les spams

```
echo "actions {" > /etc/rspamd/local.d/metrics.conf
echo 'add_header = 5;' >> /etc/rspamd/local.d/metrics.conf
echo "greylist = 25;" >> /etc/rspamd/local.d/metrics.conf
echo "reject = 50;" >> /etc/rspamd/local.d/metrics.conf
echo "}" >> /etc/rspamd/local.d/metrics.conf
```

6. Augmentez la taille de l'historique de Rspamd, activez la compression.

```
echo "nrows = 2500;" > /etc/rspamd/local.d/history_redis.conf
echo "compress = true;" >> /etc/rspamd/local.d/history_redis.conf
echo "subject_privacy = false;" >> /etc/rspamd/local.d/history_redis.conf
```

7. Activez la mise à jour automatique de rspamd

```
echo 'enabled = true;' > /etc/rspamd/local.d/redis.conf
```

8. Enrichissez les headers des mails spams. Tapez:

```
vi /etc/rspamd/local.d/milter_headers.conf
```

9. inserez le texte suivant:

```
local.d/milter_headers.conf:

Options

Add "extended Rspamd headers" (default false) (enables x-spamd-result, x-rspamd-
server & x-rspamd-queue-id routines)
extended_spam_headers = true;

List of headers to be enabled for authenticated users (default empty)
authenticated_headers = ["authentication-results"];

List of headers to be enabled for local IPs (default empty)
local_headers = ["x-spamd-bar"];

Set false to always add headers for local IPs (default true)
skip_local = true;

Set false to always add headers for authenticated users (default true)
skip_authenticated = true;

Routines to use- this is the only required setting (may be omitted if using
extended_spam_headers)
use = ["x-spamd-bar", "x-spam-level", "authentication-results"];

this is where we may configure our selected routines
routines {
 # settings for x-spamd-bar routine
 x-spamd-bar {
 # effectively disables negative spambar
 negative = "";
 }
 # other routines...
}
custom {
 # user-defined routines: more on these later
}
```

10. Créez un mot de passe. Tapez:

```
rspamadm pw
```

11. Entrez votre mot de passe. Une hashphrase est générée.
12. Copiez la.
13. Remplacez celle déjà présente dans `/etc/rspamd/local.d/worker-controller.inc`

```
vi /etc/rspamd/local.d/worker-controller.inc
```

14. Remplacez le texte entre guillemets sur la ligne `password = "$2$g95yw.....dq3c5byy";` par le texte copié.
15. Sauvez
16. Redémarrez Rspamd

```
systemctl restart rspamd
```

17. Loguez vous dans ISPConfig
18. Activer Rspamd dans ISPConfig
  - a. Allez dans la rubrique `system` → menu `Server Config` → Sélectionnez votre serveur → Onglet `Mail`
  - b. Dans le champ `Content Filter`, sélectionnez `Rspamd`
  - c. Dans le champ `Rspamd Password`, tapez votre mot de passe
  - d. Cliquez sur `Save`
  - e. Revenez dans la rubrique `system` → menu `Server Config` → Sélectionnez votre serveur → Onglet `Mail`
  - f. Vous pouvez voir le mot de passe de connexion au serveur web Rspamd.
19. Rendre le site rspamd accessible dans un host
20. Activez le module proxy dans apache

```
a2enmod proxy
systemctl restart apache2
```

21. Allez dans la rubrique `DNS`, sélectionnez le menu `Zones`, Sélectionnez votre Zone, Allez dans l'onglet `Records`.
  - a. Cliquez sur `A` et saisissez:
    - `Hostname:` ← Tapez `rspamd`
    - `IP-Address:` ← Double cliquez et sélectionnez l'adresse IP de votre serveur
  - b. Cliquez sur `Save`
22. Créer un `sub-domain (vhost)` dans le configurateur de `sites`.
  - a. Lui donner le nom `rspamd`.
  - b. Le faire pointer vers le web folder `rspamd`.

- c. Activer let's encrypt ssl
- d. Activer **Fast CGI** pour PHP
- e. Laisser le reste par défaut.
- f. Dans l'onglet Options:
- g. Dans la boîte **Apache Directives**: saisir le texte suivant:

```
ProxyPass "/.well-known/acme-challenge" http://localhost:80/.well-known/acme-
challenge
ProxyPassReverse "/.well-known/acme-challenge" http://localhost:80/.well-
known/acme-challenge
RewriteRule ^/.well-known/acme-challenge - [QSA,L]

rspamd httpserver

SetEnvIf Authorization "(.*)" HTTP_AUTHORIZATION=$1
ProxyPass / http://localhost:11334/
ProxyPassReverse / http://localhost:11334/
```

- 23. en pointant sur le site `rspamd.example.com`, et en utilisant le mot de passe saisi plus haut vous pouvez accéder aux fonctions de l'outil.
- 24. Activer l'apprentissage par déplacement
  - a. Couplé avec Dovecot, Rspamd nous propose de pouvoir apprendre également en fonction des actions des utilisateurs. Si un mail est déplacé vers le répertoire Junk, il sera appris comme tel et au contraire, s'il est sorti du répertoire Junk vers autre chose que la corbeille, il sera appris comme Ham.
  - b. Editez le fichier `Dovecot.conf` (remarques `ISPConfig` n'utilise pas aujourd'hui le contenu du répertoire `conf.d`). Tapez:

```
vi /etc/dovecot/dovecot.conf
```

- c. Insérez dans le groupe plugin et le protocol imap déjà existants dans le fichier :

```
plugin {
 sieve_plugins = sieve_imapsieve sieve_extprograms

 imapsieve_mailbox1_name = Junk
 imapsieve_mailbox1_causes = COPY
 imapsieve_mailbox1_before = file:/etc/dovecot/sieve/report-spam.sieve

 imapsieve_mailbox2_name = *
 imapsieve_mailbox2_from = Junk
 imapsieve_mailbox2_causes = COPY
 imapsieve_mailbox2_before = file:/etc/dovecot/sieve/report-ham.sieve

 sieve_pipe_bin_dir = /etc/dovecot/sieve

 sieve_global_extensions = +vnd.dovecot.pipe
}

protocol imap {
 mail_plugins = quota imap_quota imap_sieve
}
```

d. Redémarrez dovecot. Tapez:

```
service dovecot restart
```

e. Créez un répertoire sieve et éditez report-ham.sieve. Tapez:

```
mkdir -p /etc/dovecot/sieve/
vi /etc/dovecot/sieve/report-ham.sieve
```

f. Insérez le texte suivant:

```
require ["vnd.dovecot.pipe", "copy", "imapsieve", "environment", "variables"];

if environment :matches "imap.mailbox" "*" {
set "mailbox" "${1}";
}

if string "${mailbox}" "Trash" {
stop;
}

if environment :matches "imap.email" "*" {
set "email" "${1}";
}

pipe :copy "train-ham.sh" ["${email}"];
```

g. Editez report-spam.sieve. Tapez:

```
vi /etc/dovecot/sieve/report-spam.sieve
```

h. Insérez le texte suivant:

```
require ["vnd.dovecot.pipe", "copy", "imapsieve", "environment", "variables"];

if environment :matches "imap.email" "*" {
set "email" "${1}";
}

pipe :copy "train-spam.sh" ["${email}"];
```

i. Créez les scripts et rétablissez les droits et permissions. Compilez les règles. Tapez:

```
echo "exec /usr/bin/rspamc learn_ham" > /etc/dovecot/sieve/train-ham.sh
echo "exec /usr/bin/rspamc learn_spam" > /etc/dovecot/sieve/train-spam.sh
sievec /etc/dovecot/sieve/report-ham.sieve
sievec /etc/dovecot/sieve/report-spam.sieve
chmod +x /etc/dovecot/sieve/train-*
chown -R vmail:vmail /etc/dovecot/sieve
```

j. Redémarrez dovecot. Tapez:

```
service dovecot restart
```

k. Lorsque vous déplacez un mail du répertoire Inbox vers le répertoire Junk ou vice-versa, les fichiers `/var/log/mail.log` et `/var/log/rspamd/rspamd.log` doivent montrer les actions de

recalcul des spams.

25. Enfin, vous pouvez désactiver amavisd si vous le souhaitez. tapez:

```
systemctl stop amavisd-new
systemctl disable amavisd-new
```

## 12.2. Création du serveur de messagerie

Pour créer un serveur de messagerie:

1. Assurez vous d'avoir créé le domaine DNS. Si ce n'est pas le cas déroulez tout d'abord la procédure de [création de domaines](#)
2. Aller dans la rubrique **Email**. Sélectionnez ensuite le menu **Domain**
3. Cliquez sur **Add new Domain**
4. Saisissez le nom de domaine.
5. Cliquez sur **DomainKeys Identified Mail (DKIM)**
6. Cliquez sur **enable DKIM**
7. Cliquez sur **Generate DKIM Private-key**
8. Une fois cela fait, retourner dans la gestion des **Records** de domaine et activer le type DMARC
9. Garder le paramétrage par défaut et sauvegardez.
10. Faites de même pour les enregistrements SPF mais sélectionnez le mécanisme softfail.
11. Votre serveur est créé et protégé Contre les spams (entrants et sortants).

## 12.3. Finaliser la sécurisation de votre serveur de mail

Afin de mieux sécuriser votre serveur de mail, appliquez les opérations suivantes:

1. [Loguez vous comme root sur le serveur](#)
2. editez le fichier main.cf

```
vi /etc/postfix/main.cf
```

3. Rechercher **myhostname** et remplacer le texte par:

```
myhostname = mail.example.com ①
```

① Remplacer example.com par votre nom de domaine.

4. Redémarrez Postfix. Tapez:



```
service postfix restart
```

5. Vous pouvez le tester en allant sur le site [MxToolbox](#).
  - Entrez le nom de host de votre serveur de mail: mail.example.com .
  - cliquez sur **test Email Server**
  - Tout doit être correct sauf éventuellement le reverse DNS qui doit être configuré pour pointer vers mail.example.com .

## 12.4. Création de l'autoconfig pour Thunderbird et Android

La procédure est utilisé par Thunderbird et Android pour configurer automatiquement les paramètres de la messagerie.

Appliquez la procédure suivante:

1. Créer un **sub-domain (vhost)** dans le configurateur de sites.
  - a. Lui donner le nom **autoconfig**.
  - b. Le faire pointer vers le web folder **autoconfig**.
  - c. Activer let's encrypt ssl
  - d. Activer **php-FPM**
  - e. Laisser le reste par défaut.
  - f. Dans l'onglet Options:
  - g. Dans la boîte **Apache Directives**: saisir le texte suivant:

```
AddType application/x-httpd-php .php .php3 .php4 .php5 .xml

CheckSpelling On
CheckCaseOnly Off
```

- h. Sauver.
2. **Loguez vous comme root sur le serveur**
3. Dans le répertoire **/var/www/autoconfig.example.com/autoconfig/** créer un répertoire mail. Lui donner les permissions 755 et affecter les mêmes possesseurs que pour autres fichiers du répertoire. Tapez:

```
mkdir -p /var/www/autoconfig.example.com/autoconfig/mail
chmod 755 /var/www/autoconfig.example.com/autoconfig/mail
chown web1:client0 /var/www/autoconfig.example.com/autoconfig/mail ①
```

① remplacer web1:client0 par les permissions du répertoire **/var/www/autoconfig.example.com**

a. A l'intérieur de ce répertoire, Editez un fichier `config-v1.1.xml`. Tapez:

```
vi /var/www/autoconfig.example.com/autoconfig/mail/config-v1.1.xml
```

4. Y coller:

```
<?php
header('Content-Type: application/xml');
?>
<?xml version="1.0" encoding="UTF-8"?>

<clientConfig version="1.1">
 <emailProvider id="example.com"> ①
 <domain>example.com</domain>
 <displayName>Example Mail</displayName> ②
 <displayShortName>Example</displayShortName> ③
 <incomingServer type="imap">
 <hostname>mail.example.com</hostname> ①
 <port>993</port>
 <socketType>SSL</socketType>
 <authentication>password-cleartext</authentication>
 <username>%EMAILADDRESS%</username>
 </incomingServer>
 <incomingServer type="pop3">
 <hostname>mail.example.com</hostname> ①
 <port>995</port>
 <socketType>SSL</socketType>
 <authentication>password-cleartext</authentication>
 <username>%EMAILADDRESS%</username>
 </incomingServer>
 <outgoingServer type="smtp">
 <hostname>mail.example.com</hostname> ①
 <port>465</port>
 <socketType>SSL</socketType>
 <authentication>password-cleartext</authentication>
 <username>%EMAILADDRESS%</username>
 </outgoingServer>
 <outgoingServer type="smtp">
 <hostname>mail.example.com</hostname> ①
 <port>587</port>
 <socketType>STARTTLS</socketType>
 <authentication>password-cleartext</authentication>
 <username>%EMAILADDRESS%</username>
 </outgoingServer>
 </emailProvider>
</clientConfig>
```

① mettre à la place de example.com votre nom de domaine

② mettre ici votre libellé long pour votre nom de messagerie

③ mettre ici un libellé court pour votre nom de messagerie

5. Donner la permission en lecture seule et affecter les groupes d'appartenance. Tapez:

```
chmod 644 /var/www/autoconfig.example.com/autoconfig/mail/config-v1.1.xml
chown web1:client0 /var/www/autoconfig.example.com/autoconfig/mail/config-v1.1.xml
```

①

① remplacer web1:client0 par les permissions du répertoire `/var/www/autoconfig.example.com`

## 12.5. Création d'autodiscover pour Outlook

Outlook utilise un autre mécanisme pour se configurer automatiquement. Il est basé sur l'utilisation du nom de sous-domaine `autodiscover`.

Appliquez la procédure suivante:

1. Créer un `sub-domain (vhost)` dans le configurateur de sites.

a. Lui donner le nom `autodiscover`.

b. Le faire pointer vers le web folder `autodiscover`.

c. Activer `let's encrypt ssl`

d. Activer `php-FPM`

e. Laisser le reste par défaut.

f. Dans l'onglet Options:

g. Dans la boîte `Apache Directives`: saisir le texte suivant:

```
CheckSpelling On
CheckCaseOnly On
RewriteEngine On
ProxyPass "/" http://autoconfig.example.com/ ①
ProxyPassReverse "/" http://autoconfig.example.com/ ①
RewriteRule ^/ - [QSA,L]
```

① remplacer `example.com` par votre nom de domaine

h. Sauver.

2. `Loguez vous comme root sur le serveur`

3. Dans le répertoire `/var/www/autoconfig.example.com/autoconfig/`, créer un répertoire `Autodiscover`. Lui donner les permissions 755 et affecter les mêmes possesseurs que pour autres fichiers du répertoire. Tapez:

```
mkdir -p /var/www/autoconfig.example.com/autoconfig/Autodiscover/
chmod 755 /var/www/autoconfig.example.com/autoconfig/Autodiscover/
chown web1:client0 /var/www/autoconfig.example.com/autoconfig/Autodiscover/ ①
```

- ① remplacer web1:client0 par les permissions du répertoire `/var/www/autoconfig.example.com`
- a. A l'intérieur de ce répertoire, Editez un fichier `Autodiscover.xml`. Tapez:

```
vi /var/www/autoconfig.example.com/autoconfig/Autodiscover/Autodiscover.xml
```

4. Y coller:

```

<?php
 $raw = file_get_contents('php://input');
 $matches = array();
 preg_match('/<EmailAddress>(.*?)</EmailAddress>/', $raw, $matches);
 header('Content-Type: application/xml');
?>
<Autodiscover
xmlns="http://schemas.microsoft.com/exchange/autodiscover/responseschema/2006">
 <Response
xmlns="http://schemas.microsoft.com/exchange/autodiscover/outlook/responseschema/2006a">
 <User>
 <DisplayName>Example Mail</DisplayName> ②
 </User>
 <Account>
 <AccountType>email</AccountType>
 <Action>settings</Action>
 <Protocol>
 <Type>IMAP</Type>
 <Server>mail.example.com</Server> ①
 <Port>993</Port>
 <DomainRequired>off</DomainRequired>
 <SPA>off</SPA>
 <SSL>on</SSL>
 <AuthRequired>on</AuthRequired>
 <LoginName><?php echo $matches[1]; ?></LoginName>
 </Protocol>
 <Protocol>
 <Type>SMTP</Type>
 <Server>mail.example.com</Server> ①
 <Port>465</Port>
 <DomainRequired>off</DomainRequired>
 <SPA>off</SPA>
 <SSL>on</SSL>
 <AuthRequired>on</AuthRequired>
 <LoginName><?php echo $matches[1]; ?></LoginName>
 </Protocol>
 </Account>
 </Response>
</Autodiscover>

```

① mettre à la place de example.com votre nom de domaine

② mettre ici votre libellé long pour votre nom de messagerie

5. Changez les permissions comme pour le répertoire

```
chmod 644 /var/www/autoconfig.example.com/autoconfig/Autodiscover/Autodiscover.xml
chown web1:client0
/var/www/autoconfig.example.com/autoconfig/Autodiscover/Autodiscover.xml ①
```

① remplacer web1:client0 par les permissions du répertoire `/var/www/autoconfig.example.com`

6. Pointer votre navigateur sur le site <https://autodiscover.example.com/Autodiscover/Autodiscover.xml>.
7. Le contenu du fichier xml doit s'afficher
8. Vous pouvez faire aussi un test sur le [Testeur de connectivité Microsoft](#).
  - a. choisissez: **Découverte automatique Outlook**
  - b. cliquez sur **suivant**
  - c. Entrez votre adresse de courrier: `user@example.com`, un domain: `example\user`, un mot de passe tiré au hasard, Cochez les deux cases en dessous.
  - d. Cliquez sur **effectuer un test**
  - e. Le résultat doit être: **Test de connectivité réussi**

## 12.6. Création d'une boîte mail

Pour créer une boîte de messagerie:

1. Aller dans la rubrique **Email**. Sélectionnez ensuite le menu **Email Mailbox**
2. Cliquez sur **Add new Mailbox**
3. Remplissez les champs suivants:
  - a. **Name**: ← mettez votre prénom et votre nom
  - b. **Email**: ← mail\_name @ example.com
  - c. **Password**: ← saisissez un mot de passe ou générez en un
  - d. **Repeat Password** ← saisissez une deuxième fois votre mot de passe
  - e. **Quota (0 for unlimited)**: ← mettez éventuellement un quota ou laissez 0 pour illimité.
  - f. **Spamfilter**: ← Sélectionnez **Normal**
4. Dans l'onglet Backup:
  - a. **Backup interval**: Sélectionnez **Daily**
  - b. **Number of backup copies**: Sélectionnez 1
5. Cliquez sur **Save**



Notez que si vous créez une adresse mail nommée `mail_name@example.com`, vous pouvez utiliser toutes les variantes (nommées tag) derrière le caractère "+". Ainsi `mail_name+nospam@example.com` sera bien redirigé vers votre boîte et l'extension +nospam vous permettra de trier automatiquement les mails que vous ne voulez pas recevoir.



Il est possible de changer ce caractère spécial en le modifiant dans le fichier `/etc/postfix/main.cf` sur la ligne commençant par `recipient_delimiter`.

## 12.7. Configuration de votre client de messagerie.

Saisir l'adresse mail et votre mot de passe doit suffire pour configurer automatiquement votre client de messagerie.

Si vous avez besoin de configurer votre client manuellement, voici les informations à saisir:

| Paramètre              | Valeur           |
|------------------------|------------------|
| Type de serveur        | IMAP             |
| Nom de serveur IMAP    | mail.example.com |
| Nom d'utilisateur IMAP | user@example.com |
| Port IMAP              | 993              |
| Sécurité IMAP          | SSL/TLS          |
| Authentification IMAP  | Normal Password  |
| Nom de serveur SMTP    | mail.example.com |
| Nom d'utilisateur SMTP | user@example.com |
| Port SMTP              | 465              |
| Sécurité SMTP          | SSL/TLS          |
| Authentification SMTP  | Normal Password  |

## 12.8. Mise en oeuvre du site web de webmail

On suppose que vous avez installé roundcube lors de la procédure d'installation initiale et que vous avez déjà créé le host mail.example.com.

Il vous reste à appliquer la procédure suivante:

1. Créer un **sub-domain (vhost)** dans le configurateur de sites.
  - a. Lui donner le nom **mail**.
  - b. Le faire pointer vers le web folder **mail**.
  - c. Activer let's encrypt ssl
  - d. Activer **Fast CGI** pour PHP
  - e. Laisser le reste par défaut.
  - f. Dans l'onglet Options:
  - g. Dans la boîte **Apache Directives**: saisir le texte suivant:

```

ProxyPass "/.well-known/acme-challenge" http://localhost:80/.well-known/acme-
challenge
ProxyPassReverse "/.well-known/acme-challenge" http://localhost:80/.well-
known/acme-challenge
RewriteRule ^/.well-known/acme-challenge - [QSA,L]

roundcube httpserver

SSLProxyEngine On
SSLProxyCheckPeerCN Off
SSLProxyCheckPeerName Off
SSLProxyVerify none

SetEnvIf Authorization "(.*)" HTTP_AUTHORIZATION=$1
ProxyPass / https://localhost:8080/webmail/
ProxyPassReverse / https://localhost:8080/webmail/
ProxyPreserveHost On

```

2. C'est fait, vous pouvez accéder à Roundcube directement sur <https://mail.example.com>

## 12.9. Transfert de vos boites mails IMAP

Si vous faites une migration d'un ancien serveur vers un nouveau serveur vous souhaitez probablement migrer aussi vos boites mail.

La procédure ci dessous est à appliquer pour chaque compte mail IMAP. Elle peut facilement être scriptée.

Suivez la procédure suivante:

1. [Loguez vous comme root sur le serveur](#)
2. Téléchargez imapsync du repository. Tapez:

```

wget https://raw.githubusercontent.com/imapsync/imapsync/master/imapsync
chmod 755 imapsync

```

3. Installez les packages perls éventuellement manquants:

```

apt install libregexp-common-perl libfile-tail-perl libsys-meminfo-perl libunicode-
string-perl libmail-imapclient-perl libio-tee-perl libio-socket-inet6-perl libfile-
copy-recursive-perl

```

4. Créez deux fichiers temporaires qui contiennent les mots de passe du 1er et 2eme serveur. Tapez:



```
echo "passwdsrc" > secretsrc ①
echo "passwddst" > secretdst ②
chmod 600 secretsrc
chmod 600 secretdst
```

① passwdsrc est à remplacer par le mot de passe du compte sur le serveur source

② passwddst est à remplacer par le mot de passe du compte sur le serveur destination

5. Nous pouvons maintenant lancer la commande. Tapez:

```
./imapsync --host1 imap.examplesrc.com --user1 usersrc@examplesrc.com --passfile1
/etc/secretsrc --host2 imap.exampledst.com --user2 userdst@example dst.com
--passfile2 /etc/secretdst
```

6. Un fois la synchronisation effectuée, vous pouvez supprimer le fichier des mots de passe. tapez:

```
rm secretsrc
rm secretdst
```

# Chapter 13. Installation de Joomla ou de Concrete5

Joomla est un CMS très connu écrit en PHP. Il est fréquemment mis à jour et inclut une foule de plugins Concrete5 est un autre CMS assez connu avec un design plus moderne.

L'installation s'effectue à 100% avec ISPConfig. Dans la procédure ci dessous qui est taillée pour Joomla, vous pouvez l'appliquer à l'identique pour concrete5 en remplaçant les textes joomla par concrete5.

## 13.1. Création du site web de Joomla

Appliquez les opérations suivantes Dans ISPConfig:

1. Allez dans la rubrique **DNS**, sélectionnez le menu **Zones**, Sélectionnez votre Zone, Allez dans l'onglet **Records**.
  - a. Cliquez sur **A** et saisissez:
    - **Hostname:** ← Tapez **joomla**
    - **IP-Address:** ← Double cliquez et sélectionnez l'adresse IP de votre serveur
  - b. Cliquez sur **Save**
2. Créer un **sub-domain (vhost)** dans le configurateur de sites.
  - a. Lui donner le nom **joomla**.
  - b. Le faire pointer vers le web folder **joomla**.
  - c. Activer let's encrypt ssl
  - d. Activer **PHP-FPM** pour PHP
  - e. Laisser le reste par défaut.

## 13.2. Création de l'application Joomla

Appliquez les opérations suivantes dans ISPConfig:

1. Allez dans la rubrique **Sites**, le menu **Update Packagelist**.
2. Cliquez sur **Update Packagelist**
3. Allez dans la rubrique **Sites**, le menu **Available packages**.
4. Faites une recherche par **Name**. Tapez **joomla**
5. Cliquez sur le package **joomla**
6. Cliquez sur **Install this package**
7. Remplissez tous les champs:
  - **Install location:** ← choisissez votre domain (**example.com**) et laissez vide le chemin.

- **New database password** ← gardez ce qui est rempli
- **Administrator's login** ← gardez ce qui est rempli: **admin**
- **Password** et **Repeat Password** ← Tapez votre mot de passe
- **Default site language:** ← choisissez **French**
- **I accept the license** ← cochez la case

8. Cliquez sur **Install**

9. Pointez votre navigateur sur <https://example.com/> et loguez vous **admin** avec votre mot de passe saisi, c'est fait !

10. N'oubliez pas d'administrer le site et de le mettre à jour avec la dernière version de Joomla.

# Chapter 14. Installation de Mediawiki

Mediawiki est le portail wiki mondialement connu et utilisé notamment pour le site wikipedia.

L'installation s'effectue à 100% avec ISPConfig.

## 14.1. Création du site web de Mediawiki

Appliquez les opérations suivantes Dans ISPConfig:

1. Allez dans la rubrique **DNS**, sélectionnez le menu **Zones**, Sélectionnez votre Zone, Allez dans l'onglet **Records**.
  - a. Cliquez sur **A** et saisissez:
    - **Hostname:** ← Tapez **mediawiki**
    - **IP-Address:** ← Double cliquez et sélectionnez l'adresse IP de votre serveur
  - b. Cliquez sur **Save**
2. Créer un **sub-domain (vhost)** dans le configurateur de sites.
  - a. Lui donner le nom **mediawiki**.
  - b. Le faire pointer vers le web folder **mediawiki**.
  - c. Activer let's encrypt ssl
  - d. Activer **PHP-FPM** pour PHP
  - e. Laisser le reste par défaut.

## 14.2. Création de l'application Mediawiki

Appliquez les opérations suivantes dans ISPConfig:

1. Allez dans la rubrique **Sites**, le menu **Update Packagelist**.
2. Cliquez sur **Update Packagelist**
3. Allez dans la rubrique **Sites**, le menu **Available packages**.
4. Faites une recherche par **Name**. Tapez **mediawiki**
5. Cliquez sur le package **mediawiki**
6. Cliquez sur **Install this package**
7. Remplissez tous les champs:
  - **Install location:** ← choisissez votre domain (**example.com**) et laissez vide le chemin.
  - **New database password** ← gardez ce qui est rempli
  - **Administrator's login** ← gardez ce qui est rempli: **admin**
  - **Password** et **Repeat Password** ← Tapez votre mot de passe
  - **Default site language:** ← choisissez **French**

◦ I accept the license ← cochez la case

8. Cliquez sur **Install**

9. Pointez votre navigateur sur <https://example.com/> et loguez vous **admin** avec votre mot de passe saisi, c'est fait !

10. N'oubliez pas d'administrer le site et de le mettre à jour avec la dernière version de Mediawiki.

# Chapter 15. Installation de Wordpress

Wordpress est un CMS très connu écrit en PHP. Il est fréquemment mis à jour.

L'installation s'effectue à 100% avec ISPConfig.

## 15.1. Création du site web de Wordpress

Appliquez les opérations suivantes Dans ISPConfig:

1. Allez dans la rubrique **DNS**, sélectionnez le menu **Zones**, Sélectionnez votre Zone, Allez dans l'onglet **Records**.
  - a. Cliquez sur **A** et saisissez:
    - **Hostname:** ← Tapez **wordpress**
    - **IP-Address:** ← Double cliquez et sélectionnez l'adresse IP de votre serveur
  - b. Cliquez sur **Save**
2. Créer un **sub-domain (vhost)** dans le configurateur de sites.
  - a. Lui donner le nom **wordpress**.
  - b. Le faire pointer vers le web folder **wordpress**.
  - c. Activer let's encrypt ssl
  - d. Activer **PHP-FPM** pour PHP
  - e. Laisser le reste par défaut.

## 15.2. Création de l'application Wordpress

Appliquez les opérations suivantes dans ISPConfig:

1. Allez dans la rubrique **Sites**, le menu **Update Packagelist**.
2. Cliquez sur **Update Packagelist**
3. Allez dans la rubrique **Sites**, le menu **Available packages**.
4. Faites une recherche par **Name**. Tapez **wordpress**
5. Cliquez sur le package **wordpress**
6. Cliquez sur **Install this package**
7. Remplissez tous les champs:
  - **Install location:** ← choisissez votre domain (**example.com**) et laissez vide le chemin.
  - **New database password** ← gardez ce qui est rempli
  - **Administrator's login** ← gardez ce qui est rempli: **admin**
  - **Password** et **Repeat Password** ← Tapez votre mot de passe
  - **Default site language:** ← choisissez **French**

◦ I accept the license ← cochez la case

8. Cliquez sur **Install**

9. Pointez votre navigateur sur <https://example.com/> et loguez vous **admin** avec votre mot de passe saisi, c'est fait !

10. N'oubliez pas d'administrer le site et de le mettre à jour avec la dernière version de Wordpress.

# Chapter 16. Micro Weber

Microweber est un système de gestion de contenu et un constructeur de sites web Open Source. Il est basé sur le langage de programmation PHP et le framework web Laravel 5, utilisant le glisser-déposer et permettant aux utilisateurs de créer rapidement du contenu, tout en programmant et en gérant plusieurs affichages. Il dispose d'une fonction d'édition en direct qui permet aux utilisateurs de visualiser leurs modifications telles qu'elles apparaîtraient.

## 16.1. Création du site web de Microweber

Appliquez les opérations suivantes Dans ISPConfig:

1. Allez dans la rubrique **DNS**, sélectionnez le menu **Zones**, Sélectionnez votre Zone, Allez dans l'onglet **Records**.
  - a. Cliquez sur **A** et saisissez:
    - **Hostname:** ← Tapez **microweber**
    - **IP-Address:** ← Double cliquez et sélectionnez l'adresse IP de votre serveur
  - b. Cliquez sur **Save**
2. Créer un **sub-domain (vhost)** dans le configurateur de sites.
  - a. Lui donner le nom **microweber**.
  - b. Le faire pointer vers le web folder **microweber**.
  - c. Activer let's encrypt ssl
  - d. Activer **PHP-FPM** pour PHP
  - e. Laisser le reste par défaut.
  - f. Cliquez sur **Save**
3. **Loguez vous comme root sur le serveur**

## 16.2. Création des bases de données

Appliquez les opérations suivantes dans ISPConfig :

1. Créez une base de données mysql. Aller dans le menu **Database** pour définir un utilisateur MariaDB
2. Aller dans la rubrique **Sites**
  - a. Aller dans le menu **Database users** pour définir un utilisateur MariaDB
    - i. Cliquez sur **Add new User** pour créer un nouvel utilisateur
    - ii. Saisissez les informations:
      - **Database user:** ← saisir votre nom d'utilisateur **microweber** par exemple
      - **Database password:** ← saisir un mot de passe ou en générer un en cliquant sur le bouton



- **Repeat Password:** ← saisir de nouveau le mot de passe
- b. Cliquez sur **save**
- c. Cliquez sur **Add new Database** pour créer une nouvelle base de données
- d. Saisissez les informations:
  - **Site:** ← sélectionner le site **example.com**
  - **Database name:** ← Saisissez le nom de la base de données **microweber**
  - **Database user:** ← Saisir ici le nom d'utilisateur créé: **cxmicroweber**. x: est le numéro de client.
- e. Cliquez sur **save**

## 16.3. Installation de Microweber

Suivez la procédure suivante:

1. **Loguez vous comme root sur le serveur**
2. Tapez:

```
cd /var/www/microweber.example.com/microweber
wget https://raw.githubusercontent.com/microweber-
dev/webinstall/master/webinstall.php
```

3. Un fois téléchargé, faites pointer votre navigateur vers <http://microweber.example.com/netinstall.php>
4. Indique **.** comme répertoire d'installation et cliquez sur **Télécharger et décompresser Piwigo**
5. Une fois le téléchargement terminé cliquez sur **Installer Microweber**. Rechargez la page si besoin.
6. Répondez aux questions suivantes:
  - **Hote** ← Laissez **localhost**
  - **Utilisateur** ← entrez **cxmicroweber**. x est le numero de client; habituellement c'est 0
  - **Mot de passe** ← Tapez votre mot de passe
  - **Nom de la Base de données** ← entrez **cxmicroweber**. x est le numero de client; habituellement c'est 0
  - **Préfix des noms de tables** ← Laissez le champ vide
  - **Nom d'utilisateur** ← tapez **admin**
  - **Mot de passe** ← Tapez votre mot de passe
  - **Mot de passe [confirmer]** ← Tapez votre mot de passe
  - **Adresse e-mail** ← Tapez votre adresse mail d'administrateur
7. Tapez **Démarrer l'installation**

8. Vous êtes redirigé sur le site Microweber ou vous pourrez vous loguer et commencer à utiliser l'outil

# Chapter 17. Piwigo

Piwigo est une application web pour gérer votre collection de photos, et autres médias. Doté de puissantes fonctionnalités, il gère des galeries partout dans le monde. Elle est écrite en PHP et nécessite une base de données MySQL.

Piwigo était auparavant connu sous le nom PhpWebGallery.

## 17.1. Création du site web de Piwigo

Appliquez les opérations suivantes Dans ISPConfig:

1. Allez dans la rubrique **DNS**, sélectionnez le menu **Zones**, Sélectionnez votre Zone, Allez dans l'onglet **Records**.
  - a. Cliquez sur **A** et saisissez:
    - **Hostname:** ← Tapez **piwigo**
    - **IP-Address:** ← Double cliquez et sélectionnez l'adresse IP de votre serveur
  - b. Cliquez sur **Save**
2. Créer un **sub-domain (vhost)** dans le configurateur de sites.
  - a. Lui donner le nom **piwigo**.
  - b. Le faire pointer vers le web folder **piwigo**.
  - c. Activer let's encrypt ssl
  - d. Activer **PHP-FPM** pour PHP
  - e. Laisser le reste par défaut.
  - f. Cliquez sur **Save**
3. **Loguez vous comme root sur le serveur**

## 17.2. Création des bases de données

Appliquez les opérations suivantes dans ISPConfig :

1. Créez une base de données mysql. Aller dans le menu **Database** pour définir un utilisateur MariaDB
2. Aller dans la rubrique **Sites**
  - a. Aller dans le menu **Database users** pour définir un utilisateur MariaDB
    - i. Cliquez sur **Add new User** pour créer un nouvel utilisateur
    - ii. Saisissez les informations:
      - **Database user:** ← saisir votre nom d'utilisateur **piwigo** par exemple
      - **Database password:** ← saisir un mot de passe ou en générer un en cliquant sur le bouton

- **Repeat Password:** ← saisir de nouveau le mot de passe
- b. Cliquez sur **save**
- c. Cliquez sur **Add new Database** pour créer une nouvelle base de données
- d. Saisissez les informations:
  - **Site:** ← sélectionner le site **example.com**
  - **Database name:** ← Saisissez le nom de la base de données **piwigo**
  - **Database user:** ← Saisir ici le nom d'utilisateur créé: **cxpiwigo**. x: est le numéro de client.
- e. Cliquez sur **save**

## 17.3. Installation de Piwigo

Suivez la procédure suivante:

1. **Loguez vous comme root sur le serveur**
2. Tapez la commande suivante:

```
cd /var/www/piwigo.example.com/piwigo
wget http://piwigo.org/download/dlcounter.php?code=netinstall -O piwigo-netinstall.php
```

1. Un fois téléchargé, faites pointer votre navigateur vers <http://piwigo.example.com/piwigo-netinstall.php>
2. Indique **.** comme répertoire d'installation et cliquez sur **Télécharger et décompresser Piwigo**
3. Une fois le téléchargement terminé cliquez sur **Installer Piwigo**. Rechargez la page si besoin.
4. Répondez aux questions suivantes:
  - **Hote** ← Laissez **localhost**
  - **Utilisateur** ← entrez **cxpiwigo**. x est le numero de client; habituellement c'est 0
  - **Mot de passe** ← Tapez votre mot de passe
  - **Nom de la Base de données** ← entrez **cxpiwigo**. x est le numero de client; habituellement c'est 0
  - **Préfix des noms de tables** ← Laissez le champ vide
  - **Nom d'Utilisateur** ← tapez **admin**
  - **Mot de passe** ← Tapez votre mot de passe
  - **Mot de passe [confirmer]** ← Tapez votre mot de passe
  - **Adresse e-mail** ← Tapez votre adresse mail d'administrateur
5. Tapez **Démarrer l'installation**
6. Vous êtes redirigé sur le site piwigo ou vous pourrez vous loguer et commencer à utiliser l'outil

# Chapter 18. Installation de Nextcloud

NextCloud est un serveur d'hébergement et de partage de fichiers gratuit et open source, fork du projet ownCloud. Il est très similaire aux autres systèmes de partage de fichiers des services comme Google Drive, Dropbox et iCloud ou Seafile. NextCloud vous permet de stocker des fichiers, des documents, des photos, des films et des vidéos à partir de la centrale l'emplacement. Avec NextCloud, vous pouvez partager des fichiers, des contacts et tout autre les médias avec vos amis et vos clients. NextCloud s'intègre avec le courrier, calendrier, contacts et autres fonctionnalités qui aideront vos équipes à obtenir leur travail est plus rapide et plus facile. Vous pouvez installer le client NextCloud sur un ou plusieurs PC pour synchroniser les fichiers avec votre serveur Nextcloud. Des clients sont disponibles pour la plupart des systèmes d'exploitation, y compris Windows, macOS, FreeBSD, et Linux.

## 18.1. Installation initiale

NextCloud est écrit en PHP et utilise une base de données MariaDB pour stocker ses données.

Pour installer, Suivez la procédure suivante:

1. [Loguez vous comme root sur le serveur](#)
2. Installez quelques paquets de base. Tapez:

```
apt-get install php-cgi php-curl
```

3. Une fois installé, éditez le fichier php.ini pour changer quelques limitations. Tapez:

```
vi /etc/php/7.3/apache2/php.ini
```

1. Cherchez les champs ci dessous et changez les valeurs comme suit:

```
memory_limit = 512M
upload_max_filesize = 500M
post_max_size = 500M
max_execution_time = 300
date.timezone = Asia/Kolkata
```

2. Sauvez et redémarrez apache. Tapez:

```
systemctl restart apache2
```

## 18.2. Création du site web de Nextcloud

Appliquez les opérations suivantes Dans ISPConfig:

1. Allez dans la rubrique **DNS**, sélectionnez le menu **Zones**, Sélectionnez votre Zone, Allez dans l'onglet **Records**.
  - a. Cliquez sur **A** et saisissez:
    - **Hostname:** ← Tapez **nextcloud**
    - **IP-Address:** ← Double cliquez et sélectionnez l'adresse IP de votre serveur
  - b. Cliquez sur **Save**
2. Créer un **sub-domain (vhost)** dans le configurateur de sites.
  - a. Lui donner le nom **nextcloud**.
  - b. Le faire pointer vers le web folder **nextcloud**.
  - c. Activer let's encrypt ssl
  - d. Activer **PHP-FPM** pour PHP
  - e. Laisser le reste par défaut.
  - f. Cliquez sur **Save**

## 18.3. Création des bases de données

Appliquez les opérations suivantes dans ISPConfig :

1. Créez une base de données mysql. Aller dans le menu **Database** pour définir un utilisateur MariaDB
2. Aller dans la rubrique **Sites**
  - a. Aller dans le menu **Database users** pour définir un utilisateur MariaDB
    - i. Cliquez sur **Add new User** pour créer un nouvel utilisateur
    - ii. Saisissez les informations:
      - **Database user:** ← saisir votre nom d'utilisateur **nextcloud** par exemple
      - **Database password:** ← saisir un mot de passe ou en générer un en cliquant sur le bouton
      - **Repeat Password:** ← saisir de nouveau le mot de passe
  - b. Cliquez sur **save**
  - c. Cliquez sur **Add new Database** pour créer une nouvelle base de données
  - d. Saisissez les informations:
    - **Site:** ← sélectionner le site **example.com**
    - **Database name:** ← Saisissez le nom de la base de données **nextcloud**
    - **Database user:** ← Saisir ici le nom d'utilisateur créé: **cxnextcloud**. x: est le numéro de client.
  - e. Cliquez sur **save**

## 18.4. Installation de Nextcloud

Suivez la procédure suivante:

1. [Loguez vous comme root sur le serveur](#)
2. Tapez la commande suivante:

```
cd /var/www/nextcloud.example.com/nextcloud
wget https://download.nextcloud.com/server/installer/setup-nextcloud.php
```

1. Un fois téléchargé, faites pointer votre navigateur vers <http://nextcloud.example.com/setup-nextcloud.php>
2. Indique `.` comme répertoire d'installation et cliquez sur **Next**
3. Une fois le téléchargement terminé cliquez sur **Next**. Rechargez la page si besoin.
4. Répondez aux questions suivantes:
  - **Login Admin** ← tapez **admin**
  - **Password Admin** ← Tapez votre mot de passe
  - ouvrez **Stockage et base de données**
  - **Configurer la base de données** ← cliquez sur **MariaDB**
  - **Utilisateur de la Base de données** ← entrez **cxnextcloud**. x est le numero de client; habituellement c'est 0
  - **Password de la Base de données** ← Tapez votre mot de passe
  - **Nom de la Base de données** ← entrez **cxnextcloud**. x est le numero de client; habituellement c'est 0
  - **nom du serveur** ← Laissez **localhost**
5. Tapez **Next**
6. Vous êtes redirigé sur le site nextcloud ou vous pourrez vous loguer et commencer à utiliser l'outil

# Chapter 19. Installation et configuration de Gitea

Gitea est un système simple d'hébergement de code basé sur Git. C'est un fork de Gogs. Il montre des fonctionnalités similaires à gitlab ou github tout en gardant un code plus simple.

## 19.1. Création du site web de Gitea

Appliquez les opérations suivantes Dans ISPConfig:

1. Allez dans la rubrique **DNS**, sélectionnez le menu **Zones**, Sélectionnez votre Zone, Allez dans l'onglet **Records**.
  - a. Cliquez sur **A** et saisissez:
    - **Hostname:** ← Tapez **gitea**
    - **IP-Address:** ← Double cliquez et sélectionnez l'adresse IP de votre serveur
  - b. Cliquez sur **Save**
2. Créer un **sub-domain (vhost)** dans le configurateur de sites.
  - a. Lui donner le nom **gitea**.
  - b. Le faire pointer vers le web folder **gitea**.
  - c. Activer let's encrypt ssl
  - d. Activer **Fast CGI** pour PHP
  - e. Laisser le reste par défaut.
  - f. Dans l'onglet Options:
  - g. Dans la boîte **Apache Directives:** saisir le texte suivant:

```
ProxyPass "/.well-known/acme-challenge" http://localhost:80/.well-known/acme-
challenge
ProxyPassReverse "/.well-known/acme-challenge" http://localhost:80/.well-
known/acme-challenge
RewriteRule ^/.well-known/acme-challenge - [QSA,L]

gitea httpserver
#

SetEnvIf Authorization "(.*)" HTTP_AUTHORIZATION=$1
ProxyPass / http://localhost:3000/ ①
ProxyPassReverse / http://localhost:3000/ ①
```

① mettez le nom de votre domaine à la place de example.com

- h. Cliquez sur **Save**
3. **Loguez vous comme root sur le serveur**



4. Créez un utilisateur **Gitea**. Tapez:

```
adduser --system --disabled-password --group --shell /bin/bash --home /home/gitea
gitea
```

5. Créez la structure de répertoire de **Gitea**. Tapez:

```
mkdir -p /var/lib/gitea/{data,log} /etc/gitea /run/gitea
```

6. Donnez les bonnes permissions aux répertoires. Tapez:

```
chown -R gitea:gitea /var/lib/gitea
chown -R gitea:gitea /run/gitea
chown -R root:gitea /etc/gitea
chmod -R 750 /var/lib/gitea
chmod 770 /etc/gitea
```

## 19.2. Création des bases de données

Appliquez les opérations suivantes dans ISPConfig :

1. Créez une base de données mysql. Aller dans le menu **Database** pour définir un utilisateur MariaDB
2. Aller dans la rubrique **Sites**
  - a. Aller dans le menu **Database users** pour définir un utilisateur MariaDB
    - i. Cliquez sur **Add new User** pour créer un nouvel utilisateur
    - ii. Saisissez les informations:
      - **Database user:** ← saisir votre nom d'utilisateur **gitea** par exemple
      - **Database password:** ← saisir un mot de passe ou en générer un en cliquant sur le bouton
      - **Repeat Password:** ← saisir de nouveau le mot de passe
  - b. Cliquez sur **save**
  - c. Cliquez sur **Add new Database** pour créer une nouvelle base de données
  - d. Saisissez les informations:
    - **Site:** ← sélectionner le site **example.com**
    - **Database name:** ← Saisissez le nom de la base de données **gitea**
    - **Database user:** ← Saisir ici le nom d'utilisateur créé: **cxgitea**. x: est le numéro de client.
  - e. Cliquez sur **save**

## 19.3. Téléchargez et installez Gitea

Appliquez les opérations suivantes:

1. [Loguez vous comme root sur le serveur](#)
2. Téléchargez gitea du [site de chargement](#). Tapez pour un système 64 bits:

```
wget https://dl.gitea.io/gitea/master/gitea-master-linux-amd64 -O
/usr/local/bin/gitea
chmod 755 /usr/local/bin/gitea
```

3. Créez maintenant une entrée pour le launcher systemd. Tapez:

```
vi /etc/systemd/system/gitea.service
```

4. y Coller le texte suivant:

```
[Unit]
Description=Gitea (Git with a cup of tea)
After=syslog.target
After=network.target
Requires=mysql.service
[Service]
Type=simple
User=gitea
Group=gitea
WorkingDirectory=/var/lib/gitea/
RuntimeDirectory=gitea
ExecStart=/usr/local/bin/gitea web -c /etc/gitea/app.ini
Restart=always
Environment=USER=gitea HOME=/home/gitea GITEA_WORK_DIR=/var/lib/gitea
[Install]
WantedBy=multi-user.target
```

5. Recharge la base de systemd. Tapez:

```
systemctl daemon-reload
```

6. Activez et démarrez Gitea. Tapez:

```
systemctl enable gitea.service
systemctl start gitea.service
```

7. Ouvrez votre navigateur sur l'url: <https://gitea.example.com/install> et remplissez les paramètres

comme ci-après :

- **Type de base de données:** ← Sélectionnez **MySQL**
- **Nom d'utilisateur:** ← Tapez **cgitea**
- **Mot de passe:** ← Tapez le mot de passe saisi lors de la création de la base
- **Nom de base de données:** ← Tapez **cgitea**
- **Titre du site:** ← mettez une titre de votre choix
- **Emplacement racine des dépôts:** ← saisissez **/home/gitea/gitea-repositories**
- **Répertoire racine Git LFS:** ← Tapez **/var/lib/gitea/data/lfs**
- **Exécuter avec le compte d'un autre utilisateur :** ← Tapez **gitea**
- **Domaine du serveur SSH:** ← Tapez votre domaine. exemple : **gitea.example.com**
- **Port du serveur SSH:** ← Tapez **22**
- **Port d'écoute HTTP de Gitea:** ← Tapez **3000**
- **URL de base de Gitea:** ← Tapez l'URL de votre domaine. Exemple: **https://gitea.example.com**
- **Chemin des fichiers log:** ← Tapez **/var/lib/gitea/log**
- **Hôte SMTP:** ← Tapez **localhost**
- **Envoyer les e-mails en tant que:** ← Tapez **gitea@gitea.example.com**
- **Exiger la confirmation de l'e-mail lors de l'inscription:** ← cochez la case
- **Activez les notifications par e-mail:** ← cochez la case
- **Désactiver le formulaire d'inscription:** ← cochez la case
- **Masquer les adresses e-mail par défaut:** ← cochez la case

8. Laissez le reste et cliquez sur **Install Gitea**.

9. Restreignez les permissions sur le fichier de configuration de gitea. Tapez:

```
chmod 750 /etc/gitea
chown root:gitea /etc/gitea/app.ini
chmod 640 /etc/gitea/app.ini
```

10. Redémarrez **gitea**.

11. **Loguez vous comme root sur le serveur**

12. Tapez:

```
systemctl restart gitea.service
```

## 19.4. Activer une connexion SSH dédiée

En option, vous pouvez avoir envie de dédier une connexion SSH pour Gitea:

1. Loguez vous comme root sur le serveur
2. Éditez le fichier de configuration. Tapez:

```
vi /etc/gitea/app.ini
```

3. Trouvez les lignes suivantes et les remplacer dans le fichier. Chercher et remplacez:

```
START_SSH_SERVER = true
SSH_PORT = 2222 ①
```

① mettez ici le numéro de port que vous souhaitez

4. Débloquez le port 2222 dans votre firewall
  - a. Allez sur le site ispconfig <https://example.com:8080/>
  - b. Loguez-vous et cliquez sur la rubrique **System** et le menu **Firewall**. Cliquez sur votre serveur.
  - c. dans la rubrique **Open TCP ports:**, ajoutez le port 222
  - d. Cliquez sur **save**
5. Redémarrez **gitea**. Tapez:

```
systemctl restart gitea.service
```

6. Enjoy !

# Chapter 20. Installation de Seafile

Seafile est un système de partage de fichier simple et efficace écrit en Python. Il existe des clients de connexion pour Windows, Linux, Android, IOS.

Cette installation est optionnelle.

## 20.1. Création du site web de Seafile

Appliquez la procédure suivante:

1. Allez dans la rubrique **DNS**, sélectionnez le menu **Zones**, Sélectionnez votre Zone, Allez dans l'onglet **Records**.
  - a. Cliquez sur **A** et saisissez:
    - **Hostname:** ← Tapez **seafile**
    - **IP-Address:** ← Double cliquez et sélectionnez l'adresse IP de votre serveur
  - b. Cliquez sur **Save**
2. Créer un **sub-domain (vhost)** dans le configurateur de sites.
  - a. Lui donner le nom **seafile**.
  - b. Le faire pointer vers le web folder **seafile**.
  - c. Activer let's encrypt ssl
  - d. Activer **Fast CGI** pour PHP
  - e. Laisser le reste par défaut.
  - f. Dans l'onglet Options:
  - g. Dans la boîte **Apache Directives:** saisir le texte suivant:

```

Alias /media {DOCRROOT}/private/seafile/seafile-server-latest/seahub/media
RewriteEngine On

<Location /media>
Require all granted
</Location>

Alias /.well-known {DOCRROOT}/private/seafile/.well-known
RewriteEngine On

<Location /.well-known>
Require all granted
</Location>

ProxyPass "/.well-known/acme-challenge" http://localhost:80/.well-known/acme-
challenge
ProxyPassReverse "/.well-known/acme-challenge" http://localhost:80/.well-
known/acme-challenge
RewriteRule ^/.well-known/acme-challenge - [QSA,L]

seafile httpserver
#
ProxyPass /seafhttp http://localhost:8092
ProxyPassReverse /seafhttp http://localhost:8092
RewriteRule ^/seafhttp - [QSA,L]
#
seahub
#
SetEnvIf Authorization "(.*)" HTTP_AUTHORIZATION=$1
ProxyPass / http://localhost:8090/
ProxyPassReverse / http://localhost:8090/

```

## 20.2. Création de bases de données

1. Loguez vous sur ISPConfig
2. Aller dans la rubrique **Sites**
  - a. Aller dans le menu **Database users** pour définir un utilisateur MariaDB
    - i. Cliquez sur **Add new User** pour créer un nouvel utilisateur
    - ii. Saisissez les informations:
      - **Database user:** ← saisir votre nom d'utilisateur **seafile** par exemple
      - **Database password:** ← saisir un mot de passe ou en générer un en cliquant sur le bouton
      - **Repeat Password:** ← saisir de nouveau le mot de passe
  - b. Aller dans le menu **Database** pour définir les bases de données

- c. Appliquer l'opération ci après 3 fois d'affilée pour créer les trois bases suivantes: **ccnetdb**, **seafiledb**, **seahubdb**
- Cliquez sur **Add new Database** pour créer une nouvelle base de données
  - Saisissez les informations:
    - Site:** ← sélectionner le site **example.com**
    - Database name:** ← Saisissez le nom de la base de données
    - Database user:** ← Saisir ici le nom d'utilisateur créé: **cxseafile**. x: est le numéro de client.
  - Cliquez sur **save**
- d. Les trois bases de données doivent apparaître dans la liste des bases

## 20.3. Téléchargez et installez Seafile

Appliquez la procédure suivante:

1. [Loguez vous comme root sur le serveur](#)
2. Installez quelques paquets Debian complémentaires. Tapez:

```
apt-get install python2.7 python-setuptools python-simplejson python-pil python-mysqldb python-flup
```

3. Je préfère faire tourner mes serveurs dans le répertoire privé plutôt que dans le répertoire web pour des questions de sécurité. Tapez:

```
cd /var/www/seafile.example.com/private
mkdir seafile
cd seafile
wget https://download.seadrive.org/seafile-server_7.0.5_x86-64.tar.gz
tar zxvf seafile-server_7.0.5_x86-64.tar.gz
mkdir installed
mv seafile-server_* installed
cd seafile-server-*
./setup-seafile-mysql.sh
cd ../..
chown -R web1:client0 seafile ①
```

- ① choisissez le user et le groupe de votre site web. Ces informations sont consultables dans ISPConfig en consultant les informations du Web Domain → onglet **Options** → champs Linux User et Linux Group.

4. A ce moment, vous devez répondre à un certain nombre de questions.
5. Choisissez le mode de configuration 2) pour indiquer vous même les informations sur les bases de données créées.

6. Vous devrez ensuite donner le nom d'utilisateur pour la base de données, le mot de passe ainsi que le nom des 3 bases de données.
7. Si tout est saisi correctement le programme doit donner une synthèse de ce qui a été configuré

## 20.4. Lancement initial

Nous allons effectuer un premier lancement du serveur Seafile:

1. allez dans le répertoire contenant les configurations et éditez `gunicorn.conf`. Tapez:

```
cd /var/www/seafile.example.com/private/seafile/conf
vi gunicorn.conf
```

2. Repérez le texte `bind=` et mettez un numéro de port 8090 à la place de 8000. Comme ceci:

```
bind = "127.0.0.1:8090"
```

3. Editez le fichier `seafile.conf`. Tapez:

```
vi seafile.conf
```

4. mettez un port 8092 au lieu du port 8080 saisi pour l'entrée `fileserver`. Le fichier doit contenir ceci:

```
[fileserver]
port = 8092
```

5. Editez le fichier `ccnet.conf`. Tapez:

```
vi ccnet.conf
```

6. modifier l'entrée `SERVICE_URL`. Le fichier doit contenir ceci:

```
SERVICE_URL = https://seafile.example.com
```

7. Editez le fichier `seahub_settings.py`. Tapez:

```
vi seahub_settings.py
```

8. modifier l'entrée `FILE_SERVER_ROOT`. Le fichier doit contenir ceci:



```
FILE_SERVER_ROOT = 'https://seafile.example.com/seafhttp'
```

9. Démarrez Seafile. Tapez:

```
sudo -u web1 ./seafile.sh start ①
sudo -u web1 ./seahub.sh start 8090 ①
```

① remplacer le nom de user web1 par celui correspondant à celui du site web installé (indiqué dans le champ **Options** → `linux user` du web domain). (Si vous n'avez qu'un site, web1 est le bon).

10. Débloquez le port 8090 et 8092 dans votre firewall

- Allez sur le site ispconfig <https://example.com:8080/>
- Loguez-vous et cliquez sur la rubrique **System** et le menu **Firewall**. Cliquez sur votre serveur.
- dans la rubrique **Open TCP ports:**, ajoutez le port 8090 et 8092
- Cliquez sur **save**

11. Faites pointer votre navigateur sur <https://seafile.example.com>

12. La page de login de Seafile doit s'afficher

## 20.5. Lancement automatique de Seafile

Afin de s'assurer que Seafile tourne en permanence, on doit créer un script de lancement automatique de Seafile:

1. Créer un script de lancement automatique. Tapez:

```
cd /var/www/seafile.example.com/private/seafile
touch startseafile.sh
chmod +x startseafile.sh
vi startseafile.sh
```

2. Coller le texte suivant de le fichier ouvert:

```
#!/bin/bash
```

```
Change the value of "seafile_dir" to your path of seafile installation
```

```
seafile_dir=/var/www/seafile.example.com/private/seafile ①
```

```
script_path=${seafile_dir}/seafile-server-latest
```

```
seafile_init_log=${seafile_dir}/logs/seafile.init.log
```

```
seahub_init_log=${seahub_dir}/logs/seahub.init.log
```

case "\$1" in

```
start)
```

```
${script_path}/seafile.sh start >> ${seafile_init_log}
```

```
${script_path}/seahub.sh start 8090 >> ${seahub_init_log}
```

//

```
restart)
```

```
${script_path}/seafile.sh restart >> ${seafile_init_log}
```

```

$ cd /usr/local/seahub
$./seahub.sh restart 8090 >> ${seahub_init_log}

```

//

```
stop)
```

```

${script_path}/seahub.sh stop >> ${seahub_init_log}

```

```
${script_path}/seafile.sh stop >> ${seafile_init_log}
```

$$\begin{matrix} \cdot & \cdot \\ // & // \end{matrix}$$

\*)

```
echo "Usage: /etc/init.d/seafile {start|stop|restart}"
```

exit 1

//

esac

① remplacer `example.com` par votre nom de domaine

### 3. Créer un job cron dans ISPConfig pour démarrer Seafile au démarrage

a. Allez dans la rubrique **Sites** puis dans le menu **Cron Jobs**. Cliquez sur **Add cron Job**. Saisissez les champs:

- **Parent Website:** ← `mettre example.com`
- **Minutes:** ← `mettre *`
- **Hours:** ← `mettre *`
- **Days of month:** ← `mettre *`
- **Months:** ← `mettre @reboot`
- **Days of week:** ← `mettre *`
- **Command**                      **to**                      **run:**                      ←                      **mettre**  
`/var/www/seafile.example.com/private/seafile/startseafile.sh start`

#### 4. Créer un second job cron dans ISPConfig pour redémarrer Seafile tous les jours

a. Allez dans la rubrique **Sites** puis dans le menu **Cron Jobs**. Cliquez sur **Add cron Job**. Saisissez les champs:

- Parent Website: ← mettre `example.com`

- **Minutes:** ← mettre 45
- **Hours:** ← mettre 20
- **Days of month:** ← mettre \*
- **Months:** ← mettre \*
- **Days of week:** ← mettre \*
- **Command**                      **to**                      **run:**                      ←                      **mettre**  
     /var/www/seafile.example.com/private/seafile/startseafile.sh restart

5. Arrêtez le serveur précédemment lancé en tant que root. Tapez:

6. Enjoy !

# Chapter 21. Installation de Grafana

Grafana est un logiciel de visualisation et d'analyse à code source ouvert. Il vous permet d'interroger, de visualiser, d'alerter et d'explorer vos mesures, quel que soit l'endroit où elles sont stockées. En clair, il vous fournit des outils pour transformer vos données de base de données de séries chronologiques (TSDB) en de magnifiques graphiques et visualisations. Grafana s'appuie sur Prometheus afin d'obtenir des métriques. Loki est aussi installé pour réaliser une analyse précise des fichiers de logs.

Cette installation est optionnelle puisque Munin est déjà installé sur votre système.

## 21.1. Création du site web de Grafana

Appliquez la procédure suivante:

1. Allez dans la rubrique **DNS**, sélectionnez le menu **Zones**, Sélectionnez votre Zone, Allez dans l'onglet **Records**.
  - a. Cliquez sur **A** et saisissez:
    - **Hostname:** ← Tapez **grafana**
    - **IP-Address:** ← Double cliquez et sélectionnez l'adresse IP de votre serveur
  - b. Cliquez sur **Save**
2. Créer un **sub-domain (vhost)** dans le configurateur de sites.
  - a. Lui donner le nom **grafana**.
  - b. Le faire pointer vers le web folder **grafana**.
  - c. Activer let's encrypt ssl
  - d. Activer **Fast CGI** pour PHP
  - e. Laisser le reste par défaut.
  - f. Dans l'onglet Options:
  - g. Dans la boîte **Apache Directives:** saisir le texte suivant:

```
ProxyPass "/.well-known/acme-challenge" http://localhost:80/.well-known/acme-
challenge
ProxyPassReverse "/.well-known/acme-challenge" http://localhost:80/.well-
known/acme-challenge
RewriteRule ^/.well-known/acme-challenge - [QSA,L]

grafana httpserver
#

SetEnvIf Authorization "(.*)" HTTP_AUTHORIZATION=$1
ProxyPass / http://localhost:3000/
ProxyPassReverse / http://localhost:3000/
```

## 21.2. Installation de Grafana

1. [Loguez vous comme root sur le serveur](#)

2. Tapez:

```
echo "deb https://packages.grafana.com/oss/deb stable main"
>>/etc/apt/sources.list.d/grafana.list
wget -q -O - https://packages.grafana.com/gpg.key | sudo apt-key add -
```

3. Installez les paquets. Tapez:

```
apt update
apt install grafana prometheus prometheus-mysqld-exporter prometheus-apache-
exporter prometheus-bind-exporter prometheus-process-exporter
```

4. Editez la configuration de Prometheus. Tapez:

```
vi /etc/prometheus/prometheus.yml
```

5. Ajoutez les lignes suivantes:

```

- job_name: 'prometheus'

 # Override the global default and scrape targets from this job every 5 seconds.
 scrape_interval: 5s
 scrape_timeout: 5s

 # metrics_path defaults to '/metrics'
 # scheme defaults to 'http'.

 static_configs:
 - targets: ['localhost:9090']

- job_name: node
 # If prometheus-node-exporter is installed, grab stats about the local
 # machine by default.
 static_configs:
 - targets: ['localhost:9100']

- job_name: dns-master
 static_configs:
 - targets: ['localhost:9119']
 labels:
 alias: dns-master

- job_name: apache
 static_configs:
 - targets: ['localhost:9117']

- job_name: process
 static_configs:
 - targets: ['localhost:9256']

- job_name: mysql
 static_configs:
 - targets: ['localhost:9104']

```

6. Editez la configuration de **prometheus-process-exporter**. Tapez:

```
vi etc/default/prometheus-process-exporter
```

7. Ajoutez les lignes suivantes:

```
ARGS="-procnames postgres,dovecot,apache2,sshd,php-fpm7.3,rspamd,named,mysqld"
```

8. Editez la configuration de **prometheus-mysqld-exporter**. Tapez:

```
vi etc/default/prometheus-mysqld-exporter
```

9. Ajoutez les lignes suivantes:

```
ARGS='--config.my-cnf /etc/mysql/debian.cnf
--collect.info_schema.tables.databases="*" --collect.auto_increment.columns
--collect.perf_schema.file_instances.filter=".*" --collect.info_schema.tablestats'
```

10. Ajuster les permissions du fichier de conf de mysql pour donner l'accès à prometheus. Tapez:

```
chmod 644 /etc/mysql/debian.cnf
```

11. Ajustez la configuration de bind pour servir des statistiques. Tapez:

```
vi /etc/bind/named.conf
```

12. Ajouter dans le fichier:

```
statistics-channels {
 inet 127.0.0.1 port 8053 allow { 127.0.0.1; };
};
```

13. Activez dans mysql quelques statistiques. Tapez:

```
mysql -p
```

14. tapez votre mot de passe root pour mysql. puis taper:

```
INSTALL PLUGIN QUERY_RESPONSE_TIME_AUDIT SONAME 'query_response_time.so';
INSTALL PLUGIN QUERY_RESPONSE_TIME SONAME 'query_response_time.so';
INSTALL PLUGIN QUERY_RESPONSE_TIME_READ SONAME 'query_response_time.so';
INSTALL PLUGIN QUERY_RESPONSE_TIME_WRITE SONAME 'query_response_time.so';
SET GLOBAL query_response_time_stats=ON;
SET GLOBAL userstat=ON;
```

15. Redémarrez les services. Taper:

```
service prometheus restart
service prometheus-mysqld-exporter restart
service prometheus-process-exporter restart
```

## 21.3. Installation et configuration de Loki

Pour installer Loki, appliquez la procédure suivante:

1. [Loguez vous comme root sur le serveur](#)
2. Allez sur le site de [loki](#) et repérez la dernière version à charger.
3. Tapez:

```
cd /usr/local/bin
curl -fSL -o loki.gz https://github.com/grafana/loki/releases/download/v1.4.1/loki-
linux-amd64.zip
gunzip loki.gz
chmod a+x loki
```

4. Créez le fichier de configuration de loki

```
vi /etc/config-loki.yml
```

5. Ajoutez le texte ci dessous dans le fichier

```
auth_enabled: false

server:
 http_listen_port: 3100
 log_level: "warn"

ingester:
 lifecycler:
 address: 127.0.0.1
 ring:
 kvstore:
 store: inmemory
 replication_factor: 1
 final_sleep: 0s
 chunk_idle_period: 5m
 chunk_retain_period: 30s

schema_config:
 configs:
 - from: 2010-01-01
 store: boltdb
 object_store: filesystem
 schema: v9
 index:
 prefix: index_
 period: 168h
```



```
storage_config:
 boltdb:
 directory: /tmp/loki/index

 filesystem:
 directory: /tmp/loki/chunks

limits_config:
 enforce_metric_name: false
 reject_old_samples: true
 reject_old_samples_max_age: 168h

chunk_store_config:
 max_look_back_period: 0

table_manager:
 chunk_tables_provisioning:
 inactive_read_throughput: 0
 inactive_write_throughput: 0
 provisioned_read_throughput: 0
 provisioned_write_throughput: 0
 index_tables_provisioning:
 inactive_read_throughput: 0
 inactive_write_throughput: 0
 provisioned_read_throughput: 0
 provisioned_write_throughput: 0
 retention_deletes_enabled: false
 retention_period: 0
```

6. Débloquez le port 3100 dans votre firewall
  - a. Allez sur le site ispconfig <https://example.com:8080/>
  - b. Loguez-vous et cliquez sur la rubrique **System** et le menu **Firewall**. Cliquez sur votre serveur.
  - c. dans la rubrique **Open TCP ports:**, ajoutez le port 3100
  - d. Cliquez sur **save**
7. Testez maintenant la configuration de Loki. Tapez:

```
loki -config.file /etc/config-loki.yml
```

8. Ouvrez un navigateur et visitez: <http://example.com:3100/metrics>
9. Maintenant arrêtez Loki en tapant **CTRL-C**.
10. Bloquez par sécurité le port 3100 dans votre firewall
  - a. Allez sur le site ispconfig <https://example.com:8080/>
  - b. Loguez-vous et cliquez sur la rubrique **System** et le menu **Firewall**. Cliquez sur votre serveur.
  - c. dans la rubrique **Open TCP ports:**, Supprimer le port 3100

d. Cliquez sur **save**

11. Configurez un service Loki afin de le faire tourner en arrière plan. Tapez:

```
vi /etc/systemd/system/loki.service
```

12. Ajoutez le texte ci dessous et sauvez:

```
[Unit]
Description=Loki service
After=network.target

[Service]
Type=simple
ExecStart=/usr/local/bin/loki -config.file /etc/config-loki.yml

[Install]
WantedBy=multi-user.target
```

13. Maintenant lancez le service et vérifiez que tout est fonctionnel. Tapez: Now start and check the service is running.

```
sudo service loki start
sudo service loki status
```

## 21.4. Installation et configuration de Promtail

Installez maintenant Promtail:

1. [Loguez vous comme root sur le serveur](#)
2. Tapez:

```
cd /usr/local/bin
curl -fSL -o promtail.gz
https://github.com/grafana/loki/releases/download/v1.4.1/promtail-linux-amd64.zip
gunzip promtail.gz
chmod a+x promtail
```

3. Créez la configuration de Promtail. Tapez:

```
mkdir -p /var/log/journal
vi /etc/config-promtail.yml
```

4. Et ajoutez le texte suivant puis sauvez:

```
server:
 http_listen_port: 9080
 grpc_listen_port: 0

positions:
 filename: /tmp/positions.yaml

clients:
 - url: http://127.0.0.1:3100/api/prom/push

scrape_configs:
 - job_name: system
 static_configs:
 - targets:
 - localhost
 labels:
 job: varlogs
 __path__: /var/log/{*.log,*/*.log}
```

5. Débloquez le port 9080 dans votre firewall
  - a. Allez sur le site ispsconfig <https://example.com:8080/>
  - b. Loguez-vous et cliquez sur la rubrique **System** et le menu **Firewall**. Cliquez sur votre serveur.
  - c. dans la rubrique **Open TCP ports:**, ajoutez le port 9080
  - d. Cliquez sur **save**
6. testez que Promtail fonctionne. Tapez:

```
promtail -config.file /etc/config-promtail.yml
```

7. Ouvrez un navigateur et visitez: <http://example.com:9080>
8. Maintenant arrêtez Promtail en tapant **CTRL-C**.
9. Bloquez par sécurité le port 9080 dans votre firewall
  - a. Allez sur le site ispsconfig <https://example.com:8080/>
  - b. Loguez-vous et cliquez sur la rubrique **System** et le menu **Firewall**. Cliquez sur votre serveur.
  - c. dans la rubrique **Open TCP ports:**, Supprimer le port 9080
  - d. Cliquez sur **save**
10. Configurez un service Promtail afin de le faire tourner en arrière plan. Tapez:

```
vi /etc/systemd/system/promtail.service
```

11. Ajoutez le texte ci dessous et sauvez:

```
[Unit]
Description=Promtail service
After=network.target

[Service]
Type=simple
ExecStart=/usr/local/bin/promtail -config.file /etc/config-promtail.yml

[Install]
WantedBy=multi-user.target
```

12. Maintenant lancez le service et vérifiez que tout est fonctionnel. Tapez:

```
sudo service promtail start
sudo service promtail status
```

13. Allez sur votre site grafana <http://grafana.example.com> et ajoutez une source de données de type loki
14. Mettez l'URL suivante: <http://127.0.0.1:3100> . Laissez tout le reste tel quel.
15. vous pouvez maintenant explorer vos logs en utilisant le menu explore sur la gauche. Dans la zone texte "Log Labels" essayez ces exemples un à un:

```
{job="varlogs"}
```

# Chapter 22. BorgBackup

BorgBackup est un système de backup simple mais offrant des fonctionnalités avancées telles que le backup incrémental, la déduplication de données, la compression, l'authentification, l'encryption.

Borg backup est un système de backup offsite. Cela signifie que vous devez avoir accès à un espace de stockage sur un autre site pour effectuer cette sauvegarde.

Pour le moment, BorgBackup n'utilise pas de mécanisme de type RClone et il n'est donc pas encore possible de sauvegarder sur google drive ou autres espaces partagés.

## 22.1. Introduction

BorgBackup permet de stocker des backups sur un serveur distant. Nous nommerons le serveur sur lequel les sauvegardes seront stockées : serveur de stockage et identifié par <storing\_srv>. Nous nommerons le serveur qu'il faut sauvegarder: serveur sauvegardé et identifié par <example.com>

## 22.2. Installation du serveur de stockage

Il est préférable pour des questions de sécurité de créer un compte utilisateur spécifique.

Suivez la procédure suivante:

1. [Loguez vous comme root sur <storing\\_srv>.](#)
2. Tapez:

```
apt install borgbackup
```

3. [Générez un mot de passe long](#)
4. Créez un compte utilisateur. Tapez:

```
adduser borgbackup
```

5. Copiez-collez le mot de passe généré lorsqu'il est demandé
6. Créer un répertoire `~/ssh` s'il n'existe pas. tapez :

```
mkdir -p $HOME/.ssh
chmod 700 ~/.ssh
```

7. Allez dans le répertoire. Tapez :

```
cd ~/.ssh
```

8. Générez vous clés. Tapez :

```
ssh-keygen -t rsa
```

9. Un ensemble de questions apparaît. Si un texte vous explique que le fichier existe déjà, arrêtez la procédure. Cela signifie que vous avez déjà créé une clé et que vous risquez de perdre la connexion à d'autres serveurs si vous en générez une nouvelle. Sinon, appuyez sur Entrée à chaque fois pour accepter les valeurs par défaut.

10. Créez maintenant le répertoire pour recevoir les sauvegardes

```
cd
mkdir borgbackup
chmod 700 borgbackup
```

## 22.3. Installation sur le serveur sauvegardé

Suivez la procédure suivante:

1. [Loguez vous comme root sur <example.com>](#).

2. Tapez:

```
apt install borgbackup
```

3. Copiez la clé publique de root sur le <storing\_srv>. Tapez:

```
ssh-copy-id -i ~/.ssh/id_*.pub borgbackup@<storing_srv>
```

4. Coller le mot de passe généré plus haut lorsqu'il est demandé

5. Affichez votre adresse IP. tapez:

```
wget -q0- http://ipecho.net/plain; echo
```

6. Faites un essai de connexion en tapant:

```
ssh borgbackup@<storing_srv>
```

7. Aucun mot de passe ne doit être demandée et vous devez être connecté en tant que borgbackup sur le <storing\_srv>

8. Si vous êtes très attaché à la sécurité, vous pouvez restreindre l'accès au seul serveur <example.com>. Tapez sur la ligne de commande du <storing\_srv> :

```
vi ~/.ssh/authorized_keys
```

9. Ajoutez en première ligne du fichier :

```
from="SERVERIPADDRESS",command="borg serve --restrict-to-path
/home/borgbackup/borgbackup/",no-pty,no-agent-forwarding,no-port-forwarding,no-X11-
forwarding,no-user-rc ①
```

① remplacez SERVERIPADDRESS par l'adresse IP affichée plus tôt.

10. Fusionnez cette ligne avec la suivante qui démarre par ssh en prenant bien garde de laisser un espace entre no-user-rc et ssh-rsa
11. Déconnectez vous en tapant :

```
exit
```

12. De retour sur le serveur <example.com>
13. [Créez un mot de passe pour le dépôt borg backup.](#)
14. Puis tapez:

```
export BORG_PASSPHRASE='mot_passe' ①
```

① mot\_passe doit être remplacé par celui généré plus haut

15. Initialisez le dépôt borg. Tapez:

```
borg init -e repokey-blake2 borgbackup@<storing_srv>:/home/borgbackup/borgbackup/
```

16. Tout est maintenant prêt pour faire un backup

## 22.4. Effectuer un backup

Nous allons créer tout d'abord un script de backup pour sauvegarder tout le serveur sauf les répertoires système:

1. [Loguez vous comme root sur <example.com>.](#)
2. Tapez:

```
vi /usr/local/bin/borgbackup.sh
```

3. Insérez dans le fichier le texte suivant:

```
#!/bin/sh
export BORG_PASSPHRASE='mot_passe' ①
cd / && borg create --stats --progress --compress zstd
borgbackup@<storing_srv>:/home/borgbackup/borgbackup/::`hostname`-`date +%Y-%m-%d-%H-%M-%S` ./ --exclude=dev --exclude=proc --exclude=run --exclude=root/.cache/
--exclude=mnt/borgmount --exclude=sys --exclude=swapfile --exclude=tmp && cd ②
```

① mot\_passe doit être remplacé par celui généré plus haut

② si votre machine est assez puissante, vous pouvez remplacer l'algorithme de compression zstd par un algorithme lz4 (rapide) ou lzma (très lent mais performant en taille).

4. changez les permissions du script. Tapez:

```
chmod 700 /usr/local/bin/borgbackup.sh
```

5. vous pouvez maintenant effectuer une première sauvegarde en tapant:

```
/usr/local/bin/borgbackup.sh
```

6. vous pouvez ensuite planifier votre backup à 1h du matin. Tapez:

```
crontab -e
```

7. Inserez ensuite le texte suivant:

```
Backup via Borg to backup server
00 02 * * * /usr/local/bin/borgbackup.sh
```

## 22.5. Lister les backups

Nous allons créer un script de listage :

1. [Loguez vous comme root sur <example.com>](#).

2. Tapez:

```
vi /usr/local/bin/borglist.sh
```

3. Insérez dans le fichier le texte suivant:



```
#!/bin/sh
export BORG_PASSPHRASE='mot_passe' ①
borg list -v borgbackup@<storing_srv>:/home/borgbackup/borgbackup/
```

① mot\_passe doit être remplacé par celui généré plus haut.

4. changez les permissions du script. Tapez:

```
chmod 700 /usr/local/bin/borglist.sh
```

5. vous pouvez maintenant lister vos backup en tapant:

```
/usr/local/bin/borglist.sh
```

## 22.6. Vérifier un backup

Nous allons créer un script de vérification :

1. [Loguez vous comme root sur <example.com>](#).
2. Tapez:

```
vi /usr/local/bin/borgcheck.sh
```

3. Insérez dans le fichier le texte suivant:

```
#!/bin/sh
export BORG_PASSPHRASE='mot_passe' ①
borg check borgbackup@<storing_srv>:/home/borgbackup/borgbackup/::$1
```

① mot\_passe doit être remplacé par celui généré plus haut.

4. changez les permissions du script. Tapez:

```
chmod 700 /usr/local/bin/borgcheck.sh
```

5. vous pouvez maintenant vérifier un de vos backup en tapant:

```
/usr/local/bin/borgcheck.sh <nom_de_sauvegarde> ①
```

① le nom de sauvegarde est récupéré en utilisant la commande borglist.sh

## 22.7. Restaurer un backup

Nous allons créer un script de montage sous forme de système de fichier :

1. [Loguez vous comme root sur <example.com>](#).
2. Tapez:

```
vi /usr/local/bin/borgmount.sh
```

3. Insérez dans le fichier le texte suivant:

```
#!/bin/sh
mkdir -p /mnt/borgbackup
export BORG_PASSPHRASE='mot_passe' ①
borg mount borgbackup@<storing_srv>:/home/borgbackup/borgbackup/ /mnt/borgbackup
```

① mot\_passe doit être remplacé par celui généré plus haut.

4. changez les permissions du script. Tapez:

```
chmod 700 /usr/local/bin/borgmount.sh
```

5. vous pouvez maintenant monter vos backups et effectuer des opérations de fichiers. Tapez:

```
/usr/local/bin/borgmount.sh
```

6. Pour créer un script pour démonter les backups. Tapez:

```
vi /usr/local/bin/borgumount.sh
```

7. Insérez dans le fichier le texte suivant:

```
#!/bin/sh
umount /mnt/borgbackup
rmdir /mnt/borgbackup
```

8. vous pouvez maintenant demonter vos backups. Tapez:

```
/usr/local/bin/borgumount.sh
```

## 22.8. Supprimer vos vieux backups

Nous allons créer un script de ménage des backups :

1. [Loguez vous comme root sur <example.com>](#).

2. Tapez:

```
vi /usr/local/bin/borgprune.sh
```

3. Insérez dans le fichier le texte suivant:

```
#!/bin/sh
export BORG_PASSPHRASE='mot_passe' ①
borg prune --stats --progress borgbackup@<storing_srv>:/home/borgbackup/borgbackup/
--prefix 'hostname' - --keep-daily=7 --keep-weekly=4 --keep-monthly=12 ②
```

① mot\_passe doit être remplacé par celui généré plus haut.

② Le nettoyage des sauvegardes va conserver 7 sauvegardes journalières, 4 à la semaine et 12 au mois

4. changez les permissions du script. Tapez:

```
chmod 700 /usr/local/bin/borgprune.sh
```

5. vous pouvez maintenant effectuer du ménage:

```
/usr/local/bin/borgprune.sh
```

pour installer borg backup en mode rescue afin d'avoir la dernière version. apt install python3-pip libssl-dev cython3 gcc g++ libpython-dev libacl1-dev pip3 install borgbackup pip3 install rdiff-backup

# Chapter 23. Installation d'un serveur de VPN Pritunl

Pritunl est un serveur VPN basé sur OpenVPN.

## 23.1. Création du site web de Pritunl

Appliquez la procédure suivante:

1. Allez dans la rubrique **DNS**, sélectionnez le menu **Zones**, Sélectionnez votre Zone, Allez dans l'onglet **Records**.
  - a. Cliquez sur **A** et saisissez:
    - **Hostname:** ← Tapez **pritunl**
    - **IP-Address:** ← Double cliquez et sélectionnez l'adresse IP de votre serveur
  - b. Cliquez sur **Save**
2. Créer un **sub-domain (vhost)** dans le configurateur de sites.
  - a. Lui donner le nom **pritunl**.
  - b. Le faire pointer vers le web folder **pritunl**.
  - c. Activer let's encrypt ssl
  - d. Activer **Fast CGI** pour PHP
  - e. Laisser le reste par défaut.
  - f. Dans l'onglet Options:
  - g. Dans la boîte **Apache Directives**: saisir le texte suivant:

```
ProxyPass "/.well-known/acme-challenge" http://localhost:80/.well-known/acme-
challenge
ProxyPassReverse "/.well-known/acme-challenge" http://localhost:80/.well-
known/acme-challenge
RewriteRule ^/.well-known/acme-challenge - [QSA,L]

Pritunl httpserver
#
 SSLProxyEngine On
 SSLProxyCheckPeerCN Off
 SSLProxyCheckPeerName Off
 SSLProxyVerify none

SetEnvIf Authorization "(.*)" HTTP_AUTHORIZATION=$1
ProxyPass / https://localhost:8070/
ProxyPassReverse / https://localhost:8070/
ProxyPreserveHost On
```

## 23.2. Installation de Pritunl

Veillez suivre la procédure suivante:

1. [Loguez vous comme root sur le serveur](#)
2. Ajoutez des repositories Debian. Tapez:

```
tee /etc/apt/sources.list.d/mongodb-org.list << EOF
deb http://repo.mongodb.org/apt/debian buster/mongodb-org/4.2 main
EOF
tee /etc/apt/sources.list.d/pritunl.list << EOF
deb http://repo.pritunl.com/stable/apt buster main
EOF
apt-get install dirmngr
apt-key adv --keyserver hkp://keyserver.ubuntu.com --recv
E162F504A20CDF15827F718D4B7C549A058F8B6B
apt-key adv --keyserver hkp://keyserver.ubuntu.com --recv
7568D9BB55FF9E5287D586017AE645C0CF8E292A
apt-get update
apt-get --assume-yes install pritunl mongodb-org
```

3. Pritunl utilise en standard le port 80 et 443. Ces deux ports sont utilisés dans notre configuration par le serveur apache
4. On commence par arrêter apache. Tapez:



Plus aucun site web ne sera servit. Danger donc.

```
systemctl stop apache2
```

5. Démarrez Mongodb ainsi que Pritunl. Tapez:

```
systemctl start mongod pritunl
systemctl enable mongod pritunl
```

## 23.3. Configuration de Pritunl

Votre service Pritunl est actif. Vous devez maintenant le configurer pour qu'il fonctionne:

1. pointez votre navigateur sur le site web de Pritunl: <https://example.com>
2. Accepter le certificat non sécurisé. La page de setup de Pritunl s'affiche.
3. Obtenez la clé d'activation. Tapez:

```
pritunl setup-key
```

4. copier la clé dans la page web. Cliquez sur **Save**
5. La page web s'affiche en erreur. Pas d'inquiétude à avoir.
6. Arrêtez le serveur Pritunl. Tapez:

```
systemctl stop pritunl
```

7. Configurez le serveur pour qu'il n'utilise plus le port 80 et le port 443

```
pritunl set app.server_port 8070
pritunl set app.redirect_server false
```

8. Redémarrez apache et pritunl

```
systemctl start apache2
systemctl start pritunl
```

9. Pointez maintenant votre navigateur sur le site <https://pritunl.example.com> . La page de login de pritunl doit s'afficher. Si ce n'est pas le cas, revérifier votre configuration de site web dans ISPConfig et que le port 8070 est bien activé.
10. Sur le serveur, tapez:

```
pritunl default-password
```

11. Entrez dans la page web la valeur de **username** et de **password** affichés dans le terminal.
12. Une boîte de dialogue **initial setup** s'affiche. Ne changez rien mais tapez votre mot de passe.
13. Vous êtes maintenant connecté sur le site web.
14. Cliquez sur l'onglet **Users**
  - a. Cliquez sur **Add Organization**
  - b. Entrez votre nom d'organisation. Par exemple **Personnel**
  - c. Cliquez sur **Add**
  - d. Cliquez sur **Add User**
  - e. Remplissez les champs:
    - **`Name:`** ← Tapez votre nom de login (pas de caractère accentué pas d'espace)
    - **`Select an organization:`** ← sélectionnez votre organisation
    - **`Email:`** ← Tapez votre adresse Email
    - **Pin:** ← entrez votre code Pin (que des nombres; au moins 6 chiffres)

- f. Cliquez sur **Add**
- 15. Allez sur l'onglet **Servers**
  - a. Cliquez sur **Add Server**
  - b. Remplissez les champs:
    - **Name:** ← donnez un nom à votre serveur (pas de caractère accentué pas d'espace)
    - laissez le reste tel quel mais notez bien le numéro de port UDP indiqué
  - c. Cliquez sur **Add**
  - d. Cliquez sur **Attach Organization**
  - e. Sélectionnez le **server** et l'**organization**.
  - f. Cliquez sur **Attach**
- 16. Débloquez le port VPN dans votre firewall
  - a. Allez sur le site ispconfig <https://example.com:8080/>
  - b. Loguez-vous et cliquez sur la rubrique **System** et le menu **Firewall**. Cliquez sur votre serveur.
  - c. dans la rubrique **Open UDP ports:**, ajoutez le port UDP du VPN que vous avez noté.
  - d. Cliquez sur **save**
- 17. Retourner dans l'interface de Pritunl. retournez sur l'onglet **Servers**
  - a. Cliquez sur **Start server**
- 18. Votre serveur de VPN est opérationnel.

## 23.4. Se connecter au serveur de VPN

Comme Pritunl est compatible OpenVPN n'importe quel logiciel compatible OpenVPN peut être utilisé. Pritunl fournit un **client** compatible pour Linux, macOS, and Windows.

Pour se connecter à l'aide du client, vous devez charger un fichier de configuration qui est téléchargeable dans l'onglet utilisateur du serveur web. Ce fichier est à importer dans le logiciel client de Pritunl. Une fois fait, un compte apparaît dans le logiciel client. Vous pourrez vous connecter en cliquant sur le bouton **Connect** du compte utilisateur.

## 23.5. Réparer une base Pritunl

Si jamais votre base est corrompue, vous pourrez la réparer en tapant:

```
systemctl stop pritunl
pritunl repair-database
systemctl start pritunl
```

## 23.6. Mot de passe perdu

Vous pouvez re-générer un mot de passe en tapant:

```
pritunl reset-password
```



# Chapter 24. Annexe

## 24.1. Installation de Hestia

Hestia est basé sur VestaCP. C'est une alternative opensource et plus moderne de cet outil. La documentation est proposée ici: <https://docs.hestiacp.com/>

Attention Hestia n'est pas compatible de Webmin dans le sens que webmin est incapable de lire et d'interpréter les fichiers créés par Hestia.

De même, Hestia est principalement compatible de PHP. Si vous utilisez des système web basés sur des applicatifs écrits en Python ou en Ruby, la configuration sera à faire à la main avec tous les problèmes de compatibilité que cela impose.

Pour installer:

1. [Loguez vous comme root sur le serveur](#)
2. Télécharger le package et lancez l'installateur

a. Tapez :

```
wget https://raw.githubusercontent.com/hestiacp/hestiacp/release/install/hst-install.sh
```

b. Lancez l'installateur. Tapez :

```
bash hst-install.sh -g yes -o yes
```

- c. Si le système n'est pas compatible, HestiaCP vous le dira. Sinon, il vous informe de la configuration qui sera installée. Tapez **Y** pour continuer.
  - d. Entrez votre adresse mail standard et indépendante du futur serveur qui sera installé. ce peut être une adresse gmail.com par exemple.
3. Hestia est installé. Il est important de bien noter le mot de passe du compte admin de Hestia ainsi que le numéro de port du site web