Ecole Nationale Supérieure Polytechnique de Maroua

Cryptographie et Sécurité Informatique

Carte à Puce

Déployer ODOO sur un environnement IBM AIX(Unix) avec deux bases de donnes diffrentes (SQLPosgres et Informix), étudier et implmenter la fonctionnalité "Sigle Sign On" dans votre nouveau déploiement

Auteur

Oumar Djimé Ratou
Matricule 17Y402P

Enseignant

Mr. Charles Laoukoura

30 janvier 2019

Sommaires

Intoduction

Odoo est le logiciel d'entreprise tout-en-un le plus populaire au monde. Il offre une gamme d'applications professionnelles : CRM, site Web, commerce électronique, facturation, comptabilité, fabrication, entrepôt, gestion de projet, inventaire et bien plus encore, le tout parfaitement intégré. Nous verons dans cette article son importantce et comment l'installer dans un environnement Linux en l'ocurrence Centos 7[2] puise qu'on a pas pu se procuré de la distribution payant d'Unix. La version de odoo choisi ici est odoo 11[7].

1 Installation et configuration d'odoo 11

1.1 Prérequits

Pour que odoo s'installe normalement dans un système d'exploitation Linx ou autres, il faut au préalable installé les outils suivants[8] :

```
$ sudo yum update
$ sudo yum install epel-release
$ sudo yum install centos-release-scl
$ sudo yum install rh-python35
$ sudo yum install git gcc wget nodejs-less libxslt-devel
$ sudo yum install bzip2-devel openldap-devel libjpeg-devel
$ sudo yum install freetype-devel postgresql-devel
```

1.2 Création d'un utilisateur Odoo

Créez un nouvel utilisateur système et un groupe avec le répertoire de base /opt/odooqui exécutera le service Odoo :

```
$ sudo useradd -m -U -r -d /opt/odoo -s /bin/bash odoo
```

REMARQUE :

Vous pouvez nommer l'utilisateur comme bon vous semble, assurez-vous simplement de créer un utilisateur PostgreSQL [a] portant le même nom.

a. Gestion de base de donnée, on le verra dans la section suivante

1.3 Installation et configuration de PostgreSQL

Installez le serveur PostgreSQL et créez un nouveau cluster de base de données PostgreSQL :

```
$ sudo yum install postgresql-serversudo postgresql-setup initdb
```

Une fois l'installation terminée, activez et démarrez le service PostgreSQL :

```
$ sudo systemctl enable postgresqlsudo systemctl start postgresql
```

Créez un utilisateur PostgreSQL portant le même nom que l'utilisateur système créé précédemment, dans notre cas odoo :

```
$ sudo su - postgres -c "createuser -s odoo
```

1.4 Installation de Wkhtmltopdf

Le *wkhtmltoxpaquet* fournit un ensemble d'outils de ligne de commande open source pouvant rendre HTML au format PDF et à divers formats d'image. Pour imprimer des rapports PDF, vous aurez besoin de l' wkhtmltopdfoutil. La version recommandée pour Odoo est celle 0.12.1 qui n'est pas disponible dans les dépôts officiels CentOS 7[6].

Pour télécharger et installer la version recommandée, exécutez les commandes suivantes :

```
$ wget https://github.com/wkhtmltopdf/wkhtmltopdf/releases/download/\
0.12.1/wkhtmltox-0.12.1_linux-centos7-amd64.rpm
$ sudo yum localinstall wkhtmltox-0.12.1_linux-centos7-amd64.rpm
```

1.5 Installation et configuration d'Odoo 11

Nous allons installer Odoo à partir du référentiel GitHub afin de pouvoir mieux contrôler les versions et les mises à jour. Nous utiliserons également virtualenv, un outil permettant de créer des environnements Python isolés.

Avant de commencer le processus d'installation, assurez-vous de passer en mode utilisateur **odoo**.Pour ça on tape la commande suivante :

```
$ sudo su - odoo
```

Pour confirmer que vous êtes connecté en tant qu'utilisateur **odoo**, vous pouvez utiliser la commande suivante :

```
$ whoami
```

Maintenant, nous pouvons commencer par le processus d'installation, d'abord clonez l'odoo à partir du référentiel GitHub :

```
$ git clone https://www.github.com/odoo/odoo --depth 1 --branch 11.0\
        /opt/odoo/odoo11
```

Activez les collections de logiciels afin que nous puissions accéder aux fichiers binaires de Python 3.5 :

```
$ scl enable rh-python35 bash
```

Pour installez tous les modules Python requis, il nous faut crée un environnement virtuel, l'activez et ensuite l'installer en tapant les commandes successives suivantes :

```
$ cd /opt/odoopython3 -m venv odoo11-venv
$ source odoo11-venv/bin/activate
(env) $ pip3 install -r odoo11/requirements.txt
```

REMARQUE :

Si vous rencontrez des erreurs de compilation lors de l'installation, assurez-vous d'avoir installé toutes les dépendances requises répertoriées dans la section **prérequis**.

Une fois l'installation terminée, désactivez l'environnement et revenez à votre utilisateur sudo à l'aide des commandes suivantes :

```
(env) $ deactivate
$ exit
```

Si vous envisagez d'installer des modules personnalisés, il est préférable de les installer dans un répertoire séparé. Pour créer un nouveau répertoire pour les modules personnalisés, exécutez :

```
$ sudo mkdir /opt/odoo/odoo11-custom-addons
$ sudo chown odoo: /opt/odoo/odoo11-custom-addons
```

Ensuite, nous devons créer un fichier de configuration :

/etc/odoo11.conf

[options]
; C'est le mot de passe qui permet les opérations sur la base de données : $admin\_passord = motDePasseSuperAdmin$
$db\_host = False$
$db\_port = False$
$db\_user = odoo$
$db\_password = False$
$addons\_path = /opt/odoo/odoo11/addons$
; Si vous utilisez des modules personnalisés
;$addons\_path = /opt/odoo/odoo11/addons, /opt/odoo/odoo11 - custom - addons$

REMARQUE :

N'oubliez pas de changer **motDePasseSuperAdmin** par quelques chose de plus sécurisé et de règler **addons_path** si vous voulez utiliser des modules personnalisés.

1.6 Exécuter odoo en tant qu'un service

Pour exécuter odoo en tant que service, nous allons créer un fichier **odoo11.service** dans le répertoire **/etc/systemd/system/** avec le contenu suivant :

/etc/systemd/system/odoo11.service

[Unit]
Description=Odoo11
Requires=postgresql.service
After=network.target postgresql.service

[Service]
Type=simple
SyslogIdentifier=odoo11
PermissionsStartOnly=true
User=odoo
Group=odoo
ExecStart=/usr/bin/scl enable rh-python35 – /opt/odoo/odoo11-venv/bin/python3 /opt/odoo/odoo11/odoo-bin -c /etc/odoo11.conf
StandardOutput=journal+console

[Install]
WantedBy=multi-user.target

Avertissons le démon **systemd** que nous avons créé un nouveau fichier, démarrond le service Odoo et verifions l'état du sevice odoo en exécutant successivements les commandes suivantes :

```
$ sudo systemctl daemon-reload
$ sudo systemctl start odoo11
$ sudo systemctl status odoo11
```

Si tout s'est bien passé on aurra la sorite suivante :
et s'il n'ya pas d'erreur, vous pouvez activer le démarrage automatique du service Odoo au démarrage :

```
$ sudo systemctl enable odoo11
```

1.7 Test

On ouvre le navigateur et on tape **http ://nomDuDomaine ou AddresseIP :8069**, dans notre cas on a utilisé dans un prémier temps l'addresse **local 127.0.0.11** avec le port **8069** d'odoo et ça nous donne **http ://127.0.0.1 :8069**.

Le resultat est :

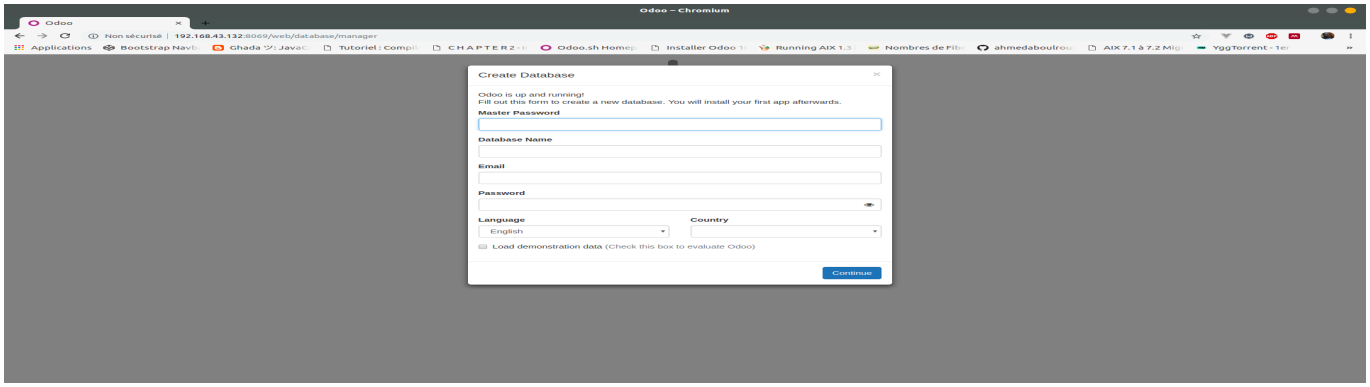

FIGURE 1 – Page de creation de base de donnée

2 Mise en place de la politique de sécurité

2.1 Mise en place de Single Sign On

2.1.1 Introduction

Compte tenu de la multiplication des applications web, les utilisateurs sont amenés à s'authentifier de nombreuse fois auprès de chacune de ces applications, en multipliant les couples identifiant/mot de passe à retenir.Ainsi pour les applications utilisant ce référentiel d'authentification, l'utilisateur peut utiliser un mot de passe unique ce qui correspond au système d'authentification Single Sign On en abrégé SSO [1].

1. SSO est Single Sign On

2.1.2 Présentation de SSO

Le Single Sign-On (SSO), appelé Signature Unique, est un mécanisme par lequel l'utilisateur n'a besoin de s'authentifier qu'une seule fois pour accéder à plusieurs ressources hétérogènes. Toute contrainte des authentifications à répétition est ainsi éliminée ; le SSO constitue un point d'entrée unique au système d'informations[3]. En somme, l'utilisateur s'authentifie une première fois ; il s'agit d'une authentification primaire. Puis la technologie SSO récupère le nécessaire pour les authentifications secondaires. Ce mécanisme évite ainsi à l'utilisateur de devoir lui- même s'authentifier de multiples fois

2.1.3 Objectif de SSO

— Simplifier la gestion de mot de passe pour les utilisateurs ;
— simplifier la gestion des données personnelles détenues par les différents services ;
— simplifier la définition et la mise en œuvre de politiques de sécurité.

2.1.4 Avantages de SSO

— la réduction de la fatigue de mot de passe : manque de souplesse liée à l'utilisation de différentes combinaisons de nom d'utilisateur et de mot de passe ;
— la réduction du temps passé à saisir le même mot de passe pour le même compte ;
— la réduction du temps passé en support informatique pour des oublis de mots de passe ;
— a centralisation des systèmes d'authentification ;
— la sécurisation à tous les niveaux d'entrée et de sortie d'accès aux systèmes sans sollicitation multiple des utilisateurs ;
— la centralisation des informations de contrôle d'accès pour les tests de conformités aux différentes normes[3].

2.1.5 Architechture de SSO

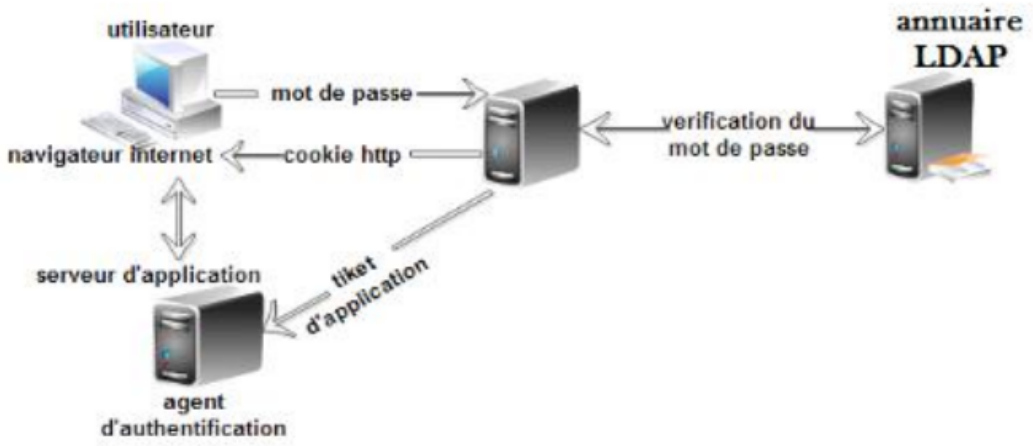

FIGURE 2 – Architecture de Singl Sign On

2.1.6 Étape d'installation et la mise en place

— La prémière étape est la mise en place d'un annaire cetralisé des utilisateurs (LDAP) ;
— La deuxième étape est de préparer le serveur a recevoir beaucoup de connexions.Il faut donc dupliquer l'annuaire sur d'autre serveur pour alléger l'accès serveur.
— La troixième étape est la mise en place de logiciel SSO[10]

2.2 génération de certificat auto-signé avec OpenSSL

Notre objectif c'est de chiffré les données qui transitent sur le web en clair, par ce que le protocole **http** n'est pas sécurisé. Donc il est question de rendre le **http** en **https** et pour cela il nous faut un certificat. Comme les certificats sont payant, nous allons tenté de mettre en place un certificat auto-signé [2].

2.2.1 Génération de la clé privé

La génération de la clé privé en utilisant le cyptosystème RSA se fait avec la commande suivante :

```
$  openssl genrsa -des3 -out server.key 2048
Generating RSA private key, 4096 bit long modulus
.....................................................++
.....................................++
e is 65537 (0x010001)
```

On peut vérifier le contenu du fichier **privates.key** s'il contient belle et bien la clé privé en tapant au console :

```
$ cat privates.key
-----BEGIN RSA PRIVATE KEY-----
MIIJKgIBAAKCAgEAxEmnYLUkdPnR8aD++UpLZT1Oqs7eA99l1piboDGDdh+fJP2z
7kkOoDZhBjTjvot/2hD/1vIjDoJ1dW8+TkGhpODcuIymZmjvzIp80zZcxQn2p63e
K388QnE5z6VJr27ON1/Cg7uaXG0vDpF692b53gqUgY+gQb1DsLIWRJmsZufGTCFe
LbslnF+MnT3ElcuT1Pl4QIN9fICE/Xgg84BLT8QNZugak7nA9h2PnH1LzcyzviPu
Mh8Jd5uCy1+yutKeJn6xVWyf8/tdGTqzpC2liOsf49hBFqIdM5Fg9XItg4kMAw==
-----END RSA PRIVATE KEY-----
```

2.2.2 Génération d'une demande de certificat CSR (Certified Signing Request)

Pour généréer une demande de certificat CSR, on tape la commande suivante :

```
$ openssl req -new -key server.key -out server.csr
```

2.2.3 Générer un certificat auto-signé avec notre CSR

Pour auto signé notre sertificat dans un fichier(server.crt), on tape la commande suivante[4] :

```
$ openssl x509 -in server.csr -out server.crt -req -signkey server.key -days 365
```

Ainsi au total on a trois 03 fichiers à savoir server.key qui contient notre clé secrète, server.csr qui contient notre demande de certificat qu'ona plus besoin pour le moment à moins qu'on veut généré un autre certificat et en fin server.crt qui est notre certificat auto signé.

2.3 Configurations

2.3.1 Installation et configuration de serveur apache ou httpd

Le serveur apache ou httpd[11] nous permet de mettre en place notre certificat auto signé. Pour l'installer sous Centos 7 ou Red Hat on tape la commande suivante :

```
$ sudo yum install httpd
```

2. C'est à dire nous qui allons signé

Après l'installation du serveur httpd, on crée un deux répertoires nommés successivement **private** pour stocker la clé privée et **certs** pour stocker le certificat dans le sous répertoire suivant /**etc**/**pki**/**tls**/.

```
$ sudo mkdir /etc/pki/tls/private
$ sudo mkdir /etc/pki/tls/certs
```

Ensuite, on deplace notre clé privée dans le répertoire /**etc**/**pki**/**tls**/**private**/ et le certificat dans le répertoire /**etc**/**pki**/**tls**/**certs**/ et on aura les chémins vers la clé et le certificat comme suit :

```
$  /etc/pki/tls/private/server.key
$  /etc/pki/tls/certs/server.crt
```

Maintenent il faut indiqué au serveur **httpd** où se trouve notre certificat et la clé privée. Pour faire cela on se déplace dans le fichier de configuration **ssl.conf** qui ce trouve dans le sous-répertoire /**etc**/**httpd**/**conf.d**/. On peut l'éditer avec un éditeur comme **vim, nano, gedit, emacs** ou autres. Dans mon cas j'utilise **vim** donc on tape la commande suivante[9] :

```
$ sudo vim /etc/httpd/conf.d/ssl.conf
```

Par la suite on repère la **ligne 100** et la **ligne 107** pour placer respectivement le chémin de certificat et de la clé comme suit :

```
SSLCertificateFile /etc/pki/tls/certs/server.crt
SLCertificateKeyFile /etc/pki/tls/private/server.key
```

Jusqu'ici on a pas indiqué au serveur httpd le port d'odoo par laquelle si on tape l'addresse **ip** ou le nom du domaine pour nous permettre d'accédé à la plateforme odoo. Pour faire cela on doit crée un fichier avec le même nom du domaine de notre serveur avec l'extension **.conf**.

```
$ sudo vim /etc/httpd/conf.d/odoocentos.com.conf
```

où, odoocentos.com est le nom de domaine de notre serveur odoo. Et on ajoute la conficguration suivant[1] :

```
<VirtualHost *:80>
        ServerName odoocentos.com
        ServerAlias www.odoocentos.com

        ProxyRequests Off
        <Proxy *>
                Order deny,allow
                Allow from all
        </Proxy>

        ProxyPass / http://odoocentos.com:8069/
        ProxyPassReverse / http://odoocentos.com:8069/

        <Location />
                Order allow,deny
                Allow from all
        </Location>
</VirtualHost>
```

NOTE :

Pour utiliser le nom de domaine **odoocentos.com** il faut bien sûre au préalable configuré le serveur DNS dans votre serveur Centos ou autres. Et le nom **odoocentos** n'est pas obligatoire, vous pouvez changé comme bon vous semble.

3 Prise en main

3.1 Création de la base de donnée

La prémière de chose c'est de créee une base de donnée, pour ce faire nous allons entrer les informations comme le montre la figure suivante :

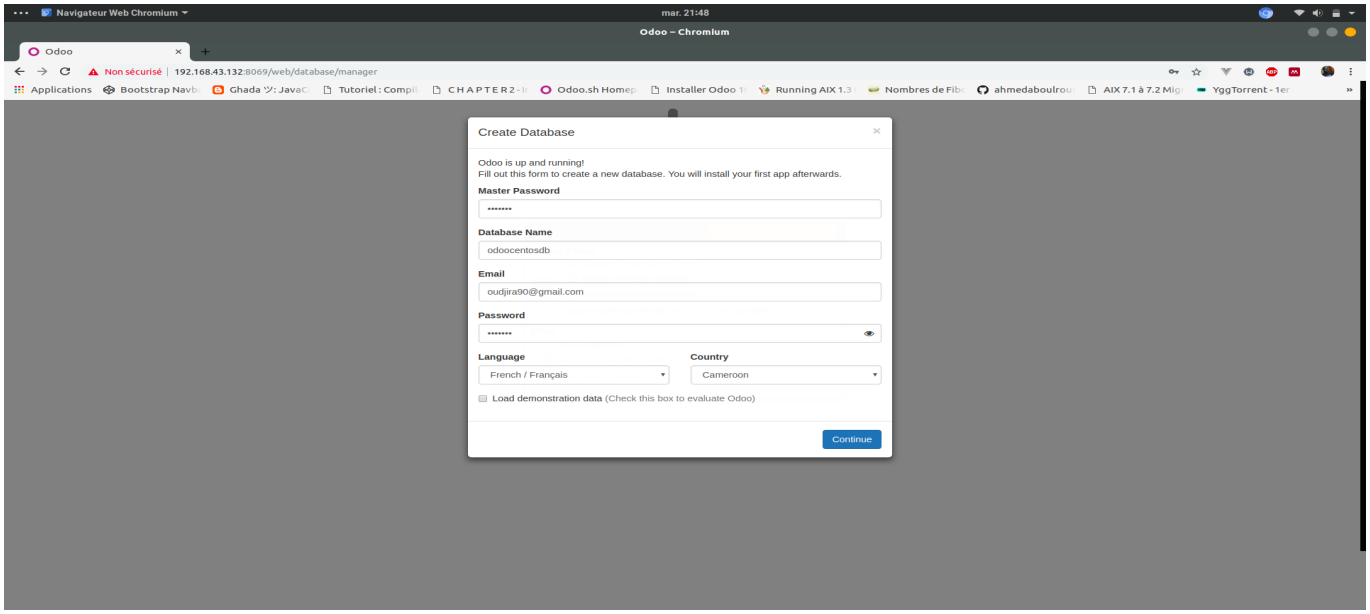

FIGURE 3 – Création de la base de donnée

Les informations sont **Master Password** qui est la mot de passe super admin [3] qu'on avait configurer au paravant, le nom de la base de donnée, l'email, le mot de passe de l'utilisateur qui est diéfférent de mot de passe super admin, ensuite on choisi la langue et le pays[5].

3. ce mot de passe nous permettra par la suite de pourvoir administrer notre base de donnée, c'est dire supprimé, modifier etc, qu'il ne faut pas confondre avec le password qui juste pour la connexion antérieure.

3.2 Tableau de board

Une fois la base de donnée créee on sera redirigé vers le tableau de bord ou dashboard en englais suivante :

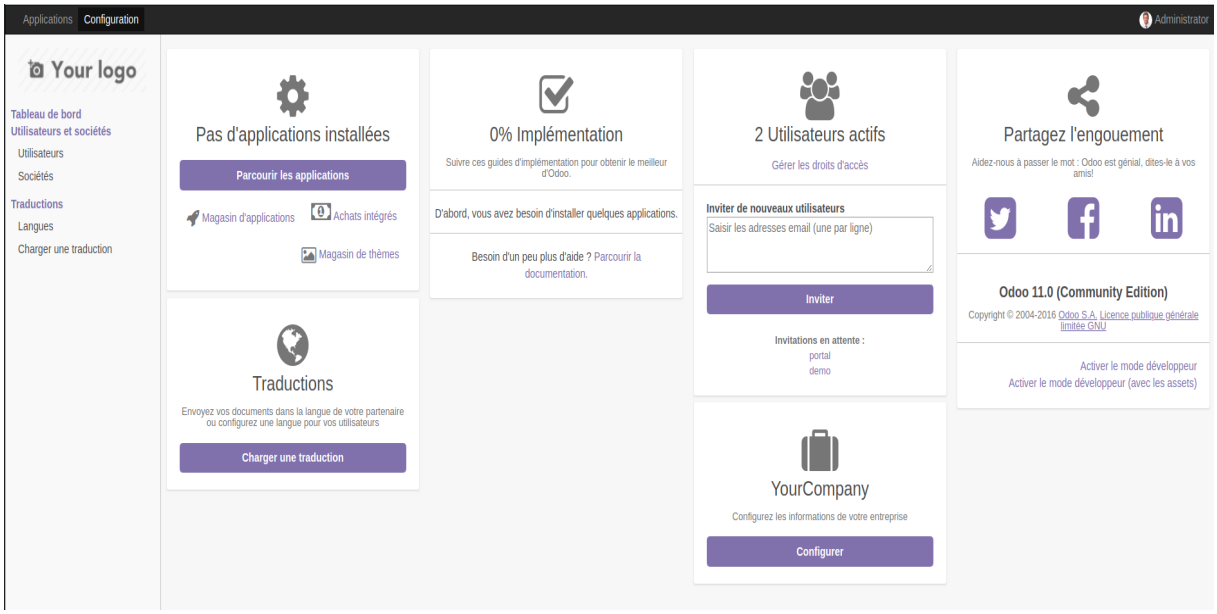

FIGURE 4 – Tableau de bord

Conclusion

Dans cette article on a découvert l'installation et la présentation de Odoo 11 sur CentOS 7[5] dans un environnement virtuel Python. En suite on a pu mettre une politique de sécurité pour chifré les données qui transite dans un canal peu sûre entre notre serveur et les différents utilisateurs.

Références

[1] NIX ARTISANS. Comment configurer une adresse ip statique sur centos 7 / rhel 7. 22/11/2017.

[2] Centos. Le projet centos. 16 Decembre 2018.

[3] cyrille Dufresnes. Sso. 11/03/2008.

[4] Digitcert. Apache : Create csr & install ssl certificate (openssl). Vue le 10 Janvier 2019.

[5] Red Hat. Red hat et l'open source. Vue le 22 Novembre 2018.

[6] Linuzixe. Comment configurer une adresse ip statique sur centos 7 / rhel 7. 9 octobre 2019.

[7] odoo. Repensons le futur du travai avec odoo. Vu le 20 decembre 2018.

[8] oDOO. Odoo cloud platform. *Enterprise grade release cycle.*, Vue le 11 Decembre 2018.

[9] Wira Soenaryo. How to use apache as reverse proxy on odoo. Novembre 8, 2015.

[10] SSO. scenari. Vue le 14/01/2019.

[11] SUREKHA TECHNOLOGIES. Configure apache web server with odoo. 8/20/2018.