Mise en place d'un serveur GITEA avec login en LDAP

Dans cette documentation nous installerons une solution GITEA avec une liaison LDAP et un certificat SSL.

Cette documentation est réalisée dans le cadre d'un TP guidé, il peut donc y avoir d'autre méthode plus ou moins simple pour y parvenir. Pour mieux s'y retrouver cette documentation disposera de plusieurs screenshots illustrant les consignes.

Préambule

Je considère que vous avez déjà suivi la documentation “MONTAGE D'UN AD METTRE LIEN” et que vos utilisateurs sont déjà créés.

Nous considérons que vous êtes équipé de cette manière :

1. Une VM sous Windows Serveur 2k16 **[AD]**
2. Une VM sous Debian 11 vierge **[GITEA]**

Les allocations de matériel (CPU/RAM...) sont à allouer selon vos envies, attention à respecter la configuration minimale. C'est à dire :

Pour **GITEA** :

1. *2GB* de ram
2. *2* cœurs de CPU
3. *20GB* d'espace disque
4. *Debian 11*

Nos IP pour notre infrastructure seront :

1. [AD] : **10.192.43.10**
2. [GITEA] : **10.192.43.14**

Mot de passe par défaut sur toutes les sessions : **Not24get**

Rappel des deux commandes essentiels :

1. `ip a` (connaitre son adresse IP)
2. `nano /etc/network/interfaces` (configuration de l'interface réseau)

Conseil : Ajouter les deux machines dans un logiciel tel que mRemoteNG pour faciliter l'administration.

Installation de Gitea par le binary

Téléchargement du binary

```
wget -O gitea https://dl.gitea.io/gitea/1.17.3/gitea-1.17.3-linux-amd64
chmod +x gitea
```

Vérification GPG

```
apt install gpg
gpg --keyserver keys.openpgp.org --recv
7C9E68152594688862D62AF62D9AE806EC1592E2
gpg --verify gitea-1.17.3-linux-amd64.asc gitea-1.17.3-linux-amd64
```

Installation des dépendances

GIT

```
apt install git
```

Vérifier avec :

```
git --version
```

MariaDB

```
apt install mariadb-server
```

```
mysql_secure_installation
```

Suivez la procédure d'installation de MariaDB.

Vérifier la bonne exécution avec :

```
systemctl status mariadb
```

apache2

```
apt-get install apache2
```

Création utilisateur

Dans le système

snippet.sh

```
adduser \
   --system \
   --shell /bin/bash \
   --gecos 'Git Version Control' \
   --group \
   --disabled-password \
   --home /home/git \
   gitea
```

Dans le SGBD

```
mysql -uroot -p
```

snippet.sql

```
CREATE USER giteaDBuser IDENTIFIED BY 'Not24get';
CREATE DATABASE giteaDB DEFAULT CHARACTER SET utf8 DEFAULT COLLATE
utf8_general_ci;
GRANT ALL PRIVILEGES ON giteaDB.* TO giteaDBuser;
FLUSH PRIVILEGES;
```

- Tester la connexion :

```
mysql -u giteaDBuser -p -e "SHOW DATABASES;"
```

Création structure de dossier

Dossiers Gitea conf

```
mkdir -p /var/lib/giteadir/{custom,data,log}
chown -R gitea:gitea /var/lib/giteadir/
chmod -R 750 /var/lib/giteadir/
mkdir /var/lib/giteadir/custom/conf
ln -s /var/lib/giteadir/custom/conf /etc/gitea
chown root:gitea /etc/gitea
chmod 770 /etc/gitea
```

Permissions dossiers

```
chmod 750 /etc/gitea
chmod 640 /etc/gitea/app.ini
```

Définition des variables

snippet.sh

```
export GITEA_WORK_DIR=/var/lib/giteadir/
```

Copie de Gitea

snippet.sh

```
cp gitea /usr/local/bin/gitea
```

Essai de Gitea (exécution du binary)

- Changer d'utilisateur :

```
su gitea
```

- Exécuter Gitea

```
GITEA_WORK_DIR=/var/lib/giteadir/ /usr/local/bin/gitea web -c
/etc/gitea/app.ini
```

Reset mot de passe compte administrateur ''gitea''

En cas de perte du mot de passe voici la commande à utiliser :

```
gitea -c /etc/gitea/app.ini admin user change-password -u gitea -p
"motdepasse"
```

Création d'un service systemd

- Copier dans `/etc/systemd/system/gitea.service` :

snippet.sh

```
[Unit]
Description=Gitea (Git with a cup of tea)
After=syslog.target
After=network.target
Wants=mariadb.service
After=mariadb.service

[Service]
RestartSec=2s
Type=simple
User=gitea
Group=
WorkingDirectory=/var/lib/giteadir/
ExecStart=/usr/local/bin/gitea web --config /etc/gitea/app.ini
Restart=always
Environment=USER=gitea HOME=/home/git GITEA_WORK_DIR=/var/lib/giteadir

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

- Activer le service et démarrer le avec :

```
systemctl enable gitea
systemctl start gitea
systemctl status gitea
```

Configuration du serveur SQL sur Gitea

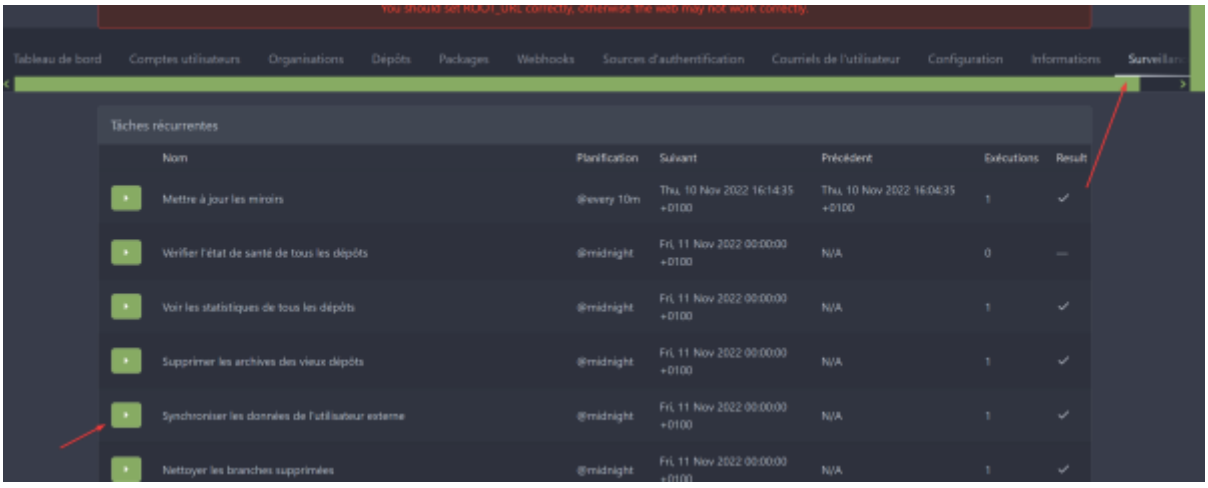
- Type : `mysql`
- Hôte : `127.0.0.1:3306`
- Nom : `giteaDB`
- Nom d'utilisateur : `giteaDBuser`

Ajout authentification LDAP

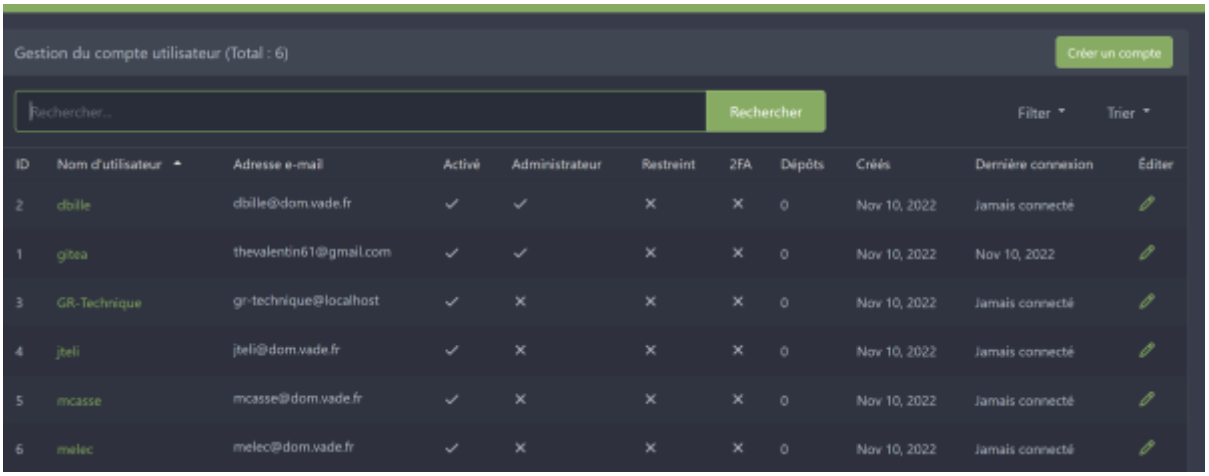
Type d'authentification LDAP (via BindDN)

- Hôte : 10.192.43.10
- Port : 389
- Bind DN : `CN=srv-gitea,OU=LDAP,OU=Utilisateurs,DC=dom,DC=vade,DC=fr`
- Bind mot de passe : `password`
- Utilisateur Search Base : `DC=dom,DC=vade,DC=fr`
- Filtre utilisateur : `(&(memberof:1.2.840.113556.1.4.1941:=cn=GG-Git-Users,ou=GG,ou=Groupes,dc=dom,dc=vade,dc=fr)(|(userPrincipalName=%[1]s)(sAMAccountName=%[1]s)))`
- Filtre administrateur : `(memberof=cn=GG-Git-Admins,ou=GG,ou=Groupes,dc=dom,dc=vade,dc=fr)`
- Attribut nom d'utilisateur : `sAMAccountName`
- Attribut prénom : `givenName`
- Attribut nom de famille : `sn`
- Attribut e-mail : `userPrincipalName`

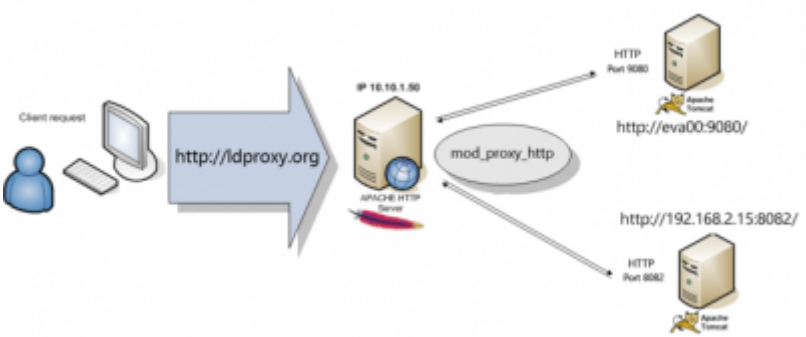
Une fois les champs remplis, mettre à jour manuellement la BDD en allant :



Voici les utilisateurs une fois synchronisés :



Reverse proxy avec apache et ajout d'un certificat SSL

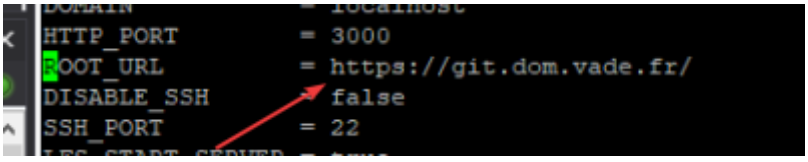


Modification configuration root_url (pour passer en https)

- Avant de lancer l'installation :

- Après avoir lancé l'installation :

```
nano /etc/gitea/app.ini
```



Création de la config apache2

- Activer l'option reverse_proxy :

```
a2enmod proxy proxy_http
service apache2 restart
```

- Créer la configuration dans les *sites-available* :

```
nano /etc/apache2/sites-available/git.dom.vade.fr.conf
```

snippet.sh

```
<VirtualHost *:80>
    ServerName git.dom.vade.fr
    Redirect permanent / https://git.dom.vade.fr/
</VirtualHost>
<VirtualHost *:443>
    ServerName git.dom.vade.fr
    ServerAdmin valentin@moimeme.fr

    ProxyPass / http://127.0.0.1:3000/
    ProxyPassReverse / http://127.0.0.1:3000/
    ProxyRequests Off
</VirtualHost>
```

- Activer la configuration :

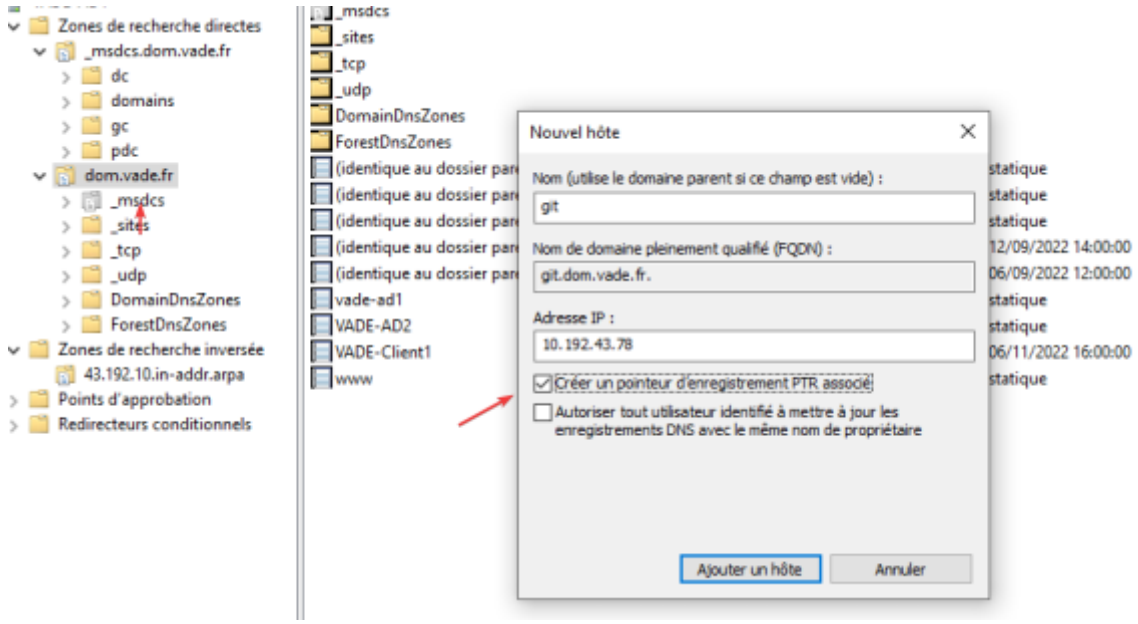
```
a2ensite git.dom.vade.fr.conf
systemctl apache2 reload
```

- Puis vérifier :

```
systemctl status apache2
```

Ajout de la règle CNAME dans le DNS

- Créer la règle DNS en recherche direct dans le serveur DNS :



Création certificat auto-signé sur git.dom.vade.fr

```
apt-get install openssl
```

```
openssl req -x509 -nodes -days 365 -newkey rsa:2048 -sha256 -out
/etc/apache2/server.crt -keyout /etc/apache2/server.key
```

Suivre la procédure pour générer le certificat, mettre en FQDN : `git.dom.vade.fr`.

- Ajout dans le `virtual-host:443` le certificat :

```
nano /etc/apache2/sites-available/git.dom.vade.fr.conf
```

```

SSLEngine on
SSLCertificateFile /etc/apache2/server.crt
SSLCertificateKeyFile /etc/apache2/server.key
```

- Activer le SSL :

```
a2enmod ssl
```

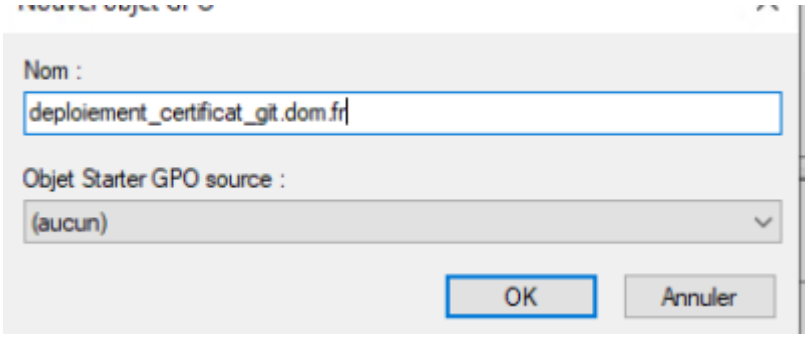
Puis redémarre le service :

```
systemctl restart apache2
```

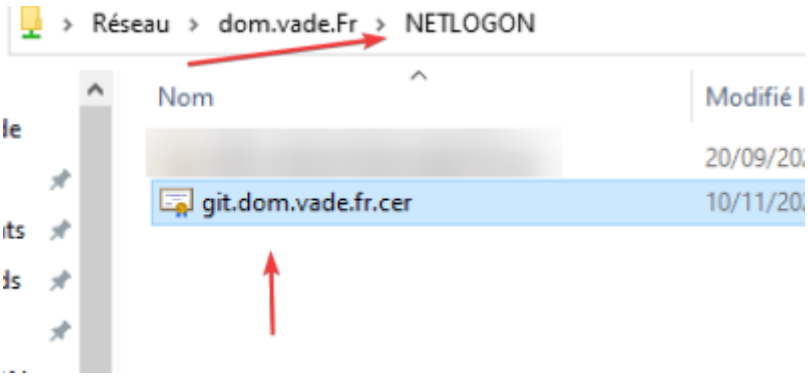
Déploiement GPO du certificat

Rappel, utilitaire de certificat sur windows : `certmgr.msc`

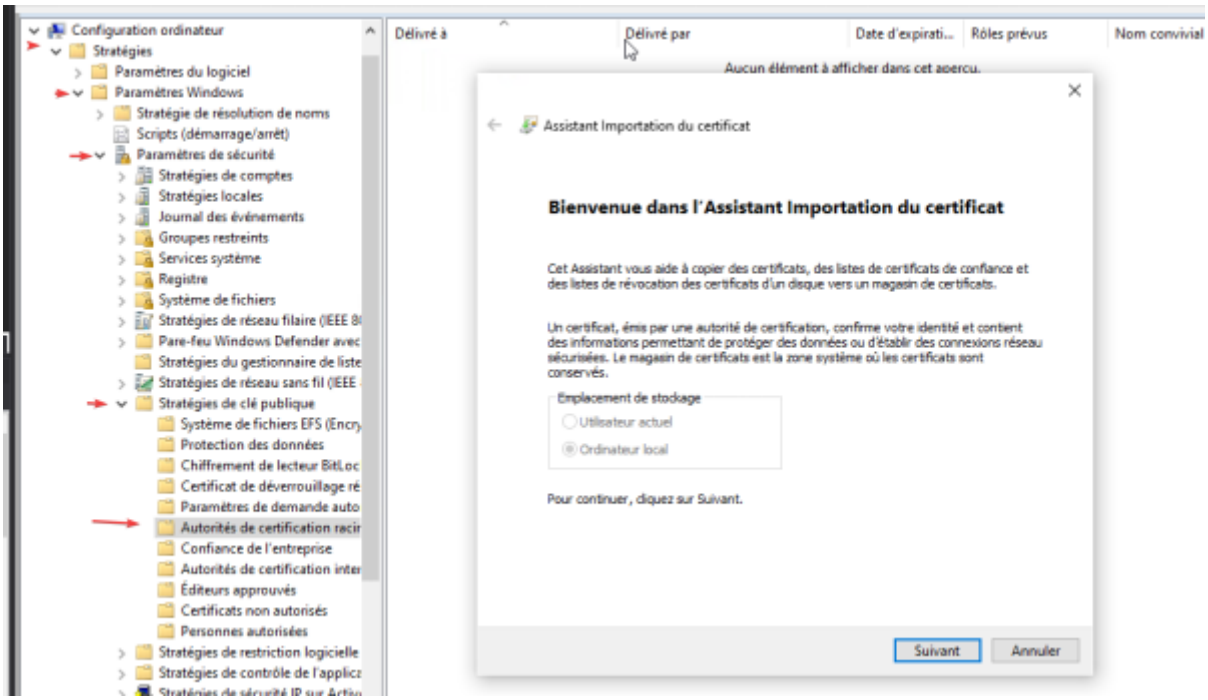
- Création de la GPO :



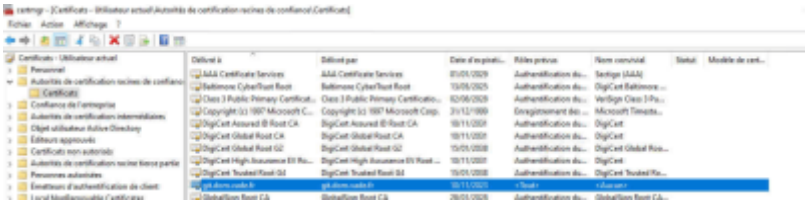
Mettre le certificat dans le NETLOGON du serveur :



- Ajout du certificat dans l'importation :



- Test du déploiement du certificat
  - Redémarrer le poste
  - Ouvrir la console MMC de gestion de certificat sur l'ordinateur local et aller sur Autorité de certification racines de confiance et vérifier la présence du certificat.



Le certificat est présent sur le client.

Essai HTTPS client windows

Mes sources

1. Tous les screens en raw : http://files.stoneset.fr/stoneset/images/doc_ad/?C=M;O=D
2. https://docs.gitea.io/en-us/install-from-binary/
3. https://rdr-it.com/gpo-deployer-un-certificat/