

Création Docker OPENLDAP image OSIXIA / GITHUB avec réplication

```
#####  
creation fake Docker sur le serveur ldapPREPRD-02:  
LDAP_CID=$(docker run --restart always --name ldap-test --hostname "hostname" --env LDAP_BASE_DN="DC" --env LDAP_ORGANISATION="NOM" --env LDAP_DOMAIN="domaine" --env LDAP_ADMIN_PASSWORD=PASSWORD  
--env LDAP_TLS=false -v /data2/ldap:/var/lib/ldap -v /data2/slapd.d:/etc/ldap/slapd.d -p 1389:389 --detach osixia/openldap:1.5.0)  
#####  
creation Vrai Docker sur srv 1 :  
LDAP_CID=$(docker run --restart always --name ldap --hostname "hostname" --env LDAP_BASE_DN="DC" --env LDAP_ORGANISATION="NOM" --env LDAP_DOMAIN="domaine" --env LDAP_ADMIN_PASSWORD=PASSWORD  
--env LDAP_TLS=true --env LDAP_TLS_VERIFY_CLIENT=never --env LDAP_REPLICATION=true --env LDAP_REPLICATION_HOSTS="PYTHON2BASH:('ldap://hostname', 'ldap://hostname')"  
--env LDAP_TLS_CERT_FILENAME=crt --env LDAP_TLS_KEY_FILENAME=key --env LDAP_TLS_CA_CERT_FILENAME=crt -v /data/ldap:/var/lib/ldap -v /data/slapd.d:/etc/ldap/slapd.d --volume  
me /etc/pki/openldap/pprod-2:/container/service/slapd/assets/certs -p 389:389 -p 636:636 --detach osixia/openldap:1.5.0)  
creation Vrai Docker sur srv 2 :  
LDAP_CID=$(docker run --restart always --name ldap --hostname "hostname" --env LDAP_BASE_DN="DC" --env LDAP_ORGANISATION="NOM" --env LDAP_DOMAIN="domaine" --env LDAP_ADMIN_PASSWORD=PASSWORD  
--env LDAP_TLS=true --env LDAP_TLS_VERIFY_CLIENT=never --env LDAP_REPLICATION=true --env LDAP_REPLICATION_HOSTS="PYTHON2BASH:('ldap://hostname', 'ldap://hostname')"  
--env LDAP_TLS_CERT_FILENAME=crt --env LDAP_TLS_KEY_FILENAME=key --env LDAP_TLS_CA_CERT_FILENAME=crt -v /data/ldap:/var/lib/ldap -v /data/slapd.d:/etc/ldap/slapd.d --volume  
me /etc/pki/openldap/pprod-2:/container/service/slapd/assets/certs -p 389:389 -p 636:636 --detach osixia/openldap:1.5.0)
```

creation fake Docker sur le serveur : CREATION LDAP TEST SANS REPLICATION POUR TESTER un LDIF.

```
LDAP_CID=$(docker run --restart always --name ldap-test --hostname "hostname" --env LDAP_BASE_DN="DC" --env  
LDAP_ORGANISATION="NOM" --env LDAP_DOMAIN="domaine" --env LDAP_ADMIN_PASSWORD=PASSWORD  
--env LDAP_TLS=false -v /data2/ldap:/var/lib/ldap -v /data2/slapd.d:/etc/ldap/slapd.d -p 1389:389 --detach  
osixia/openldap:1.5.0)
```

```
#####  
#####  
#####
```

creation Vrai Docker sur srv 1 avec replication et sécurité certificat ect.. = c'est une commande
entiere a cp

```
LDAP_CID=$(docker run --restart always --name ldap --hostname "hostname" --env LDAP_BASE_DN="DC" --env  
LDAP_ORGANISATION="NOM" --env LDAP_DOMAIN="domaine" --env LDAP_ADMIN_PASSWORD=PASSWORD --  
env LDAP_TLS=true --env LDAP_TLS_VERIFY_CLIENT=never --env LDAP_REPLICATION=true --env
```

```
LDAP_REPLICATION_HOSTS="#PYTHON2BASH:['ldap://noomdulap','ldap://noomdulap']" --env
LDAP_TLS_CERT_FILENAME=cert.crt --env LDAP_TLS_KEY_FILENAME=cert.key --env
LDAP_TLS_CA_CERT_FILENAME=ca.crt -v /data/ldap:/var/lib/ldap -v /data/slapd.d:/etc/ldap/slapd.d --volume
/etc/pki/openldap/pprod-2:/container/service/slapd/assets/certs -p 389:389 -p 636:636 --detach
osixia/openldap:1.5.0)
```

creation Vrai Docker sur srv 2 avec replication sur le 1 et sécurité certificat ect.. = c'est une commande entiere a cp

```
LDAP_CID=$(docker run --restart always --name ldap --hostname "hostname" --env LDAP_BASE_DN="DC" --env
LDAP_ORGANISATION="NOM" --env LDAP_DOMAIN="domaine" --env LDAP_ADMIN_PASSWORD=PASSWORD --
env LDAP_TLS=true --env LDAP_TLS_VERIFY_CLIENT=never --env LDAP_REPLICATION=true --env
LDAP_REPLICATION_HOSTS="#PYTHON2BASH:['ldap://noomdulap','ldap://noomdulap']" --env
LDAP_TLS_CERT_FILENAME=cert.crt --env LDAP_TLS_KEY_FILENAME=cert.key --env
LDAP_TLS_CA_CERT_FILENAME=ca.crt -v /data/ldap:/var/lib/ldap -v /data/slapd.d:/etc/ldap/slapd.d --volume
/etc/pki/openldap/pprod-2:/container/service/slapd/assets/certs -p 389:389 -p 636:636 --detach
osixia/openldap:1.5.0)
```

Suppression du docker et des data docker, vite fait et efficace, vous pouvez après recréer le ldap dockeriser avec les commande du dessus :

```
##### suppression data + dockre
docker stop ""
docker rm ldap
rm -rf /data2/slapd.d/*
rm -rf /data2/ldap/*
```

```
#####
#####
#####
```

Outils en interface sur windows qui aide a rentrer des Idif ou troublehsoot des compte ect ...

```
vi LDAP/LdapAdminTool-7.8.x-win-x64-Setup.exe
```

commande test ldap et autres voir si il répond.

```
ldapsearch -x -H ldap://noomdulap -b dc=,dc= -D "cn=,dc=,dc=" -W
ldappasswd -D "cn=,dc=,dc=" -W -S "uid=mail@domaine.com,ou=,ou=,dc=,dc="
ldapwhoami -x -D "uid=mail@domaine.com,ou=,ou=,dc=,dc=" -W -v
```

```
ldapsearch -x -H ldap://noomduldap -b dc=,dc= -D "cn=,dc=,dc=" -W
ldapwhoami -x -D "uid=mail@domaine.com,ou=,ou=,dc=,dc=" -W -v
ldapwhoami -x -D "uid=,ou=,ou=,dc=,dc=" -W -v
```

grep -5ni 'test.test,' LdapExt-Prod-FINAL.ldif == pour rechercher les user ou erreur dans un contexte chaine de caractère sur du dn ou uid

commande test ldap et autres voir si il répond.

```
[admin] $ docker exec -it ldap /bin/bash

***** à l'interieur du docker *****

ldappasswd -D "cn=admin" -W -S "uid=,ou=,dc=,dc="
ldapwhoami -x -D "uid=,com,ou=,ou=,dc=,dc=" -W -v

***** another server searching docker ldap *****

ldapsearch -x -H ldap:// -b dc=,dc= -D "cn=admin,dc=,dc=" -W
ldapsearch -x -H ldap://10.105 -b dc=,dc= -D "cn=admin,dc=,dc=" -W

***** docker cmd ! *****
docker exec -it ldap /bin/bash


docker exec ldap ldapadd -x -D "cn=admin,dc=,dc=" -W -f /home/.ldif
docker cp ./ .ldif ldap:/home/.ldif

docker logs ldap
```

Debug docker logs :

3. Suivre les logs en temps réel en affichant d'abord les 50 dernières lignes :

```
bash
```

 Copier le code

```
docker logs --tail 50 -f 123abc456def
```

Créer un ldif :

```
ldapsearch -x -H ldap://localhost -D "cn=admin,dc=example,dc=com" -w admin_password -b
"dc=example,dc=com" "(objectClass=*)" > backup.ldif
```

Revision #4

Created 31 October 2024 17:29:12 by Cavallone

Updated 19 November 2024 16:33:12 by Cavallone