FACULDADES promove	**FACULDADE PROMOVE DE SETE LAGOAS** Rua Dr. Pena, 35 – Centro – campus 1 Av. Prefeito Alberto Moura, 15 – Nova Cidade – campus 2	
TUTORIAL AULA 9		
CURSO: TECNOLÓGICO EM REDES DE COMPUTADORES		
DISCIPLINA: SERVIÇOS DE DIRETÓRIOS	**PROFESSOR**: MANASSÉS FERREIRA	
TURMA: 4° PERÍODO NOTURNO TURMA I	**DATA:** 09/10/2017	**NOTA:**
ALUNO:		

Instalação/Configuração do openldap via docker no linux (ubuntu server 16.04.3)

Primeiro passo: Instalar o docker-ce [1]

```
sudo apt-get update
sudo apt-get install linux-image-extra-$(uname -r) linux-image-extra-virtual

sudo apt-get update
sudo apt-get install apt-transport-https ca-certificates curl software-properties-common

curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -

sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu \
   $(lsb_release -cs) \
   stable"

sudo apt-get update
sudo apt-get install docker-ce

#para adicionar o usuário atual ao grupo docker, é necessário reiniciar a sessão
sudo usermod -aG docker $USER
```

Segundo passo: Instanciar o docker [2]

```
docker run --name meuLDAP --detach osixia/openldap:1.1.9
```

Para verificar os containers docker que estão rodando, execute: `docker ps -a`
Uma saída semelhante a seguir será produzida:

```
CONTAINER ID        IMAGE                  COMMAND                  CREATED             STATUS              PORTS               NAMES
575c68d784b2        osixia/openldap:1.1.9  "/container/tool/run"    51 seconds ago      Up 50 seconds       389/tcp, 636/tcp    meuLDAP
```

Para verificar as imagens de containers que estão disponíveis, execute: `docker images`
Uma saída semelhante a seguir será produzida:

```
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
osixia/openldap     1.1.9               0670342c5b82        2 months ago        180MB
```

Terceiro passo: Consulta ao servidor de diretórios - `ldapsearch` **autenticação simples**

```
docker exec meuLDAP
  ldapsearch -x
             -H ldap://localhost
             -b dc=example,dc=org
             -D "cn=admin,dc=example,dc=org"
             -w admin
```

A execução do `docker exec` acima realiza uma consulta à base LDAP -b dc=example,dc=org do container `meuLDAP` no host -H ldap://localhost usando o comando `ldasearch` em modo de autenticação simples indicado pelo parâmetro `-x`, com o usuário -D "cn=admin,dc=example,dc=org" e senha -w admin. Para que a senha fosse solicitada, poderia ser usado o parâmetro `-W`.

Verifique isso executando:

```
docker exec -i meuLDAP
  ldapsearch -x
             -H ldap://localhost
             -b dc=example,dc=org
             -D "cn=admin,dc=example,dc=org"
             -W
```

O parâmetro `-i` foi adicionado ao docker exec para que ocorra interação. A consulta resultante do comando acima será algo como:

```
# extended LDIF
#
# LDAPv3
# base <dc=example,dc=org> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# example.org
dn: dc=example,dc=org
objectClass: top
objectClass: dcObject
objectClass: organization
o: Example Inc.
dc: example

# admin, example.org
dn: cn=admin,dc=example,dc=org
objectClass: simpleSecurityObject
objectClass: organizationalRole
cn: admin
description: LDAP administrator
userPassword:: e1NTSEF9WmRDTVZlK2VJNkw5Z2szK0dHZnRHS0VGYSt0dHQ0bEY=

# search result
search: 2
result: 0 Success

# numResponses: 3
# numEntries: 2
```

Quarto passo: Consulta ao servidor de diretórios - `ldapsearch` **autenticação anônima**

```
docker exec meuLDAP
  ldapsearch -x
             -H ldap://localhost
             -b dc=example,dc=org
```

O parâmetro `-i` foi adicionado ao `docker exec` para que ocorra interação. A consulta resultante do comando acima será algo como:

```
# extended LDIF
#
# LDAPv3
# base <dc=example,dc=org> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# search result
search: 2
result: 32 No such object

# numResponses: 1
```

Quinto passo: Adicionar um entrada na árvore LDAP do servidor de diretórios – `ldapadd`

Inicialmente crie o arquivo `adicao.ldif` contendo o seguinte:

```
dn: uid=billy,dc=example,dc=org
uid: billy
cn: billy
sn: 3
objectClass: top
objectClass: posixAccount
objectClass: inetOrgPerson
loginShell: /bin/bash
homeDirectory: /home/billy
uidNumber: 14583102
gidNumber: 14564100
userPassword: {SSHA}j3lBh1Seqe4rqF1+NuWmjhvtAni1JC5A
mail: billy@example.org
gecos: Billy User
```

Copie o arquivo para o container, usando `docker cp adicao.ldif meuLDAP:/root/`

Em seguida execute:

```
docker exec meuLDAP
  ldapadd -x
          -D "cn=admin,dc=example,dc=org" -w admin
          -f /root/adicao.ldif
          -H ldap://localhost
          -ZZ
```

Realize uma nova consulta ldapsearch para verificar que a entrada foi adicionada.

Sexto passo: Excluir um entrada na árvore LDAP do servidor de diretórios – `ldapmodify`
Inicialmente crie o arquivo `exclusao.ldif` contendo o seguinte:

```
dn: uid=billy,dc=example,dc=org
changetype: delete
```

Copie o arquivo para o container, usando `docker cp exclusao.ldif meuLDAP:/root/`
Em seguida execute:

```
docker exec meuLDAP
  ldapmodify -x
             -D "cn=admin,dc=example,dc=org" -w admin
             -f /root/exclusao.ldif
             -H ldap://localhost
             -ZZ
```

Realize uma nova consulta ldapsearch para verificar que a entrada foi excluída.

Sétimo passo: Abrindo um shell no container – criando um hash com `slappasswd`

```
docker exec -ti meuLDAP bash
root@575c68d784b2:/# slappasswd -s teste
{SSHA}a0vZFAUlPCgzEplUbH6tZPldpJsNYaWr
```

Oitavo passo: Adicionar outra entrada na árvore LDAP – `ldapmodify`
Inicialmente crie o arquivo `leitor.ldif` contendo o seguinte:

```
dn: cn=leitor,dc=example,dc=org
changetype: add
cn: leitor
objectClass: simpleSecurityObject
objectClass: organizationalRole
userPassword: {SSHA}a0vZFAUlPCgzEplUbH6tZPldpJsNYaWr
description: LDAP read only user
```

Copie o arquivo para o container, usando `docker cp leitor.ldif meuLDAP:/root/`
Em seguida execute:

```
docker exec meuLDAP
  ldapmodify -x
             -D "cn=admin,dc=example,dc=org" -w admin
             -f /root/leitor.ldif
             -H ldap://localhost
             -ZZ
```

Realize uma consulta ldapsearch com o novo usuário.

```
docker exec -i meuLDAP
  ldapsearch -x
             -H ldap://localhost
             -b dc=example,dc=org
             -D "cn=leitor,dc=example,dc=org"
             -W
```

Nono passo: Concedendo privilégio de leitura via ACLs – `ldapmodify`

Inicialmente crie o arquivo `leitor-acl.ldif` contendo o seguinte:

```
dn: olcDatabase={1}hdb,cn=config
changetype: modify
delete: olcAccess
-
add: olcAccess
olcAccess: to attrs=userPassword,shadowLastChange by self write by dn="cn=admin,dc=example,dc=org" write by anonymous auth by * none
olcAccess: to * by self write by dn="cn=admin,dc=example,dc=org" write by dn="cn=leitor,dc=example,dc=org" read by * none
```

Copie o arquivo para o container, usando `docker cp leitor-acl.ldif meuLDAP:/root/`

Em seguida execute:

```
docker exec meuLDAP
  ldapmodify -x
             -D "cn=admin,dc=example,dc=org" -w admin
             -f /root/leitor-acl.ldif
             -H ldap://localhost
             -ZZ
```

Realize uma nova consulta ldapsearch com o usuário.

```
docker exec -i meuLDAP
  ldapsearch -x
             -H ldap://localhost
             -b dc=example,dc=org
             -D "cn=leitor,dc=example,dc=org"
             -W
```

Décimo passo: Interface gráficas para gerenciar LDAP: `phpldapmyadmin - apachedirectorystudio`

Usaremos o container com o phpldapadmin. [3]

```
#!/bin/bash -e
docker run --name ldap-service --hostname ldap-service --detach osixia/openldap:1.1.9

docker run --name phpldapadmin-service --hostname phpldapadmin-service --link ldap-service:ldap-host --env
PHPLDAPADMIN_LDAP_HOSTS=ldap-host -detach osixia/phpldapadmin:0.7.0

PHPLDAP_IP=$(docker inspect -f "{{ .NetworkSettings.IPAddress }}" phpldapadmin-service)

echo "Go to: https://$PHPLDAP_IP"
echo "Login DN: cn=admin,dc=example,dc=org"
echo "Password: admin"
```

O apache directory studio possui dependência com o java JRE. [4]

```
wget http://ftp.unicamp.br/pub/apache/directory/studio/2.0.0.v20170904-M13/ApacheDirectoryStudio-2.0.0.v20170904-M13-linux.gtk.x86_64.tar.gz
sudo apt-get install default-jre
tar zxvf ApacheDirectoryStudio-2.0.0.v20170904-M13-linux.gtk.x86_64.tar.gz
cd ApacheDirectoryStudio
./ApacheDirectoryStudio
```

Iremos configurar o acesso de ambas ferramentas ao servidor de diretórios configurado nos outros passos.

Referências:

[1] https://docs.docker.com/engine/installation/linux/docker-ce/ubuntu
[2] https://github.com/osixia/docker-openldap
[3] https://github.com/osixia/phpldapadmin
[4] http://directory.apache.org/studio