一、ldap 服务器的搭建

1.1. 环境准备

A. linux 环境，以 centos7 为例

B.关闭防火墙

systemctl stop firewalld.service #停止 firewall

systemctl disable firewalld.service #禁止 firewall 开机启动

firewall-cmd --state #查看默认防火墙状态（关闭后显示 notrunning，开启后显示 running）

C.关闭网络状态

由于 network 和 NetworkManager 服务会出现冲突，而且 NetworkManager 通常会比较先启动，所以为了防止 NetworkManager 的启动导致我们直接配置的网络环境失效，我们需要禁用它!

systemctl stop NetworkManager #临时关闭

systemctl disable NetworkManager #永久关闭网络管理命令

D.关闭 seLinux（安全子系统）

临时设置

setenforce 1 成为 permissive 模式

setenforce 0 成为 enforcing 模式

永久设置（设置后需要重启才能生效）

vi /etc/selinux/config

将 SELINUX=enforcing 改为 SELINUX=disabled

二、openLDAP 安装部署

2.1. openldap 服务端必要软件安装

1.使用 yum 命令安装

```
# yum install -y openldap openldap-clients openldap-servers compat-openldap openldap-devel
```

2. 安装 libdb 相关依赖

```
# yum -y install libdb.x86_64 libdb-devel.x86_64
```

3. 复制一个默认配置到指定目录下，并授权，这一步一定要做，然后再启动服务，不然生成密码时会报错

```
# cp /usr/share/openldap-servers/DB_CONFIG.example /var/lib/ldap/DB_CONFIG
```

4. 授权给 ldap 用户，此用户 yum 安装时便会自动创建

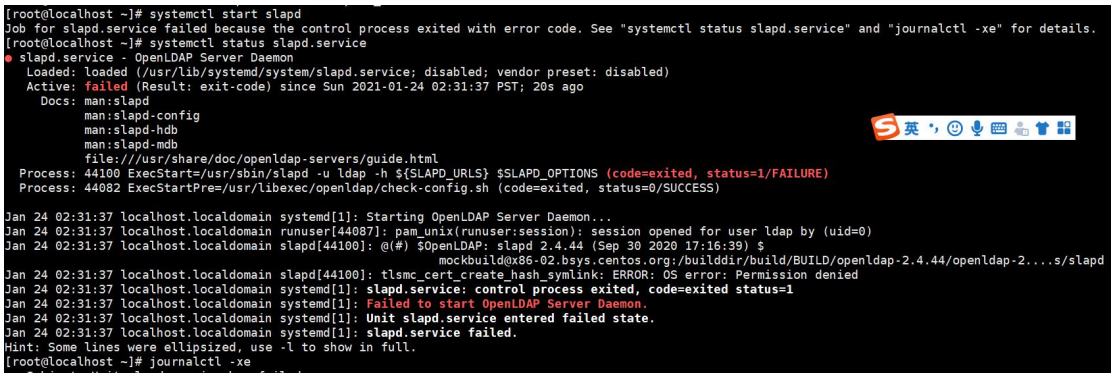
```
# chown -R ldap. /var/lib/ldap/DB_CONFIG
```

5. 启动 ldap server 服务，先启动服务，配置后面再进行修改

```
# systemctl start slapd
# systemctl enable slapd
```

若是不关闭 NetworkManager 以及 SELinux，在执行 systemctl start slapd 时会报错 ：

Job for slapd.service failed because the control process exited with error code. See "systemctl status slapd.service" and "journalctl -xe" for details.



6. 查看状态，正常启动则 OK

```
# systemctl status slapd
```

```
[root@localhost ~]# systemctl status slapd
● slapd.service - OpenLDAP Server Daemon
   Loaded: loaded (/usr/lib/systemd/system/slapd.service; enabled; vendor preset: disabled)
   Active: active (running) since Sun 2021-01-24 02:40:48 PST; 1min 30s ago
     Docs: man:slapd
           man:slapd-config
           man:slapd-hdb
```

2.2. 修改管理者密码

1. 生成管理员密码，记录下这个密码，后面需要用到

slappasswd -s 123456

```
[root@localhost myldap]# slappasswd -s 123456
{SSHA}oXLjRIdeul1EwV8Bmhq9lAeb+7GvnnRu
```

2. 新增修改密码文件，ldif 为后缀，不要在/etc/openldap/slapd.d 目录下创建类似文件，生成的文件为需要通过命令去动态修改 ldap 现有配置，如在/opt 下新建自己的目录 myldap，在自己目录下创建文件

cd /opt
mkdir myldap
vim changepwd.ldif
dn: olcDatabase={0}config,cn=config
changetype: modify
add: olcRootPW
olcRootPW:{SSHA}oXLjRIdeul1EwV8Bmhq9lAeb+7GvnnRu

```
dn: olcDatabase={0}config,cn=config
changetype: modify
add: olcRootPW
olcRootPW: {SSHA}oXLjRIdeul1EwV8Bmhq9lAeb+7GvnnRu
~
```

3. 执行命令，修改 ldap 配置，通过-f 执行文件

ldapadd -Y EXTERNAL -H ldapi:/// -f changepwd.ldif

执行修改命令后，有如下输出则为正常：

```
[root@node3 ~]# ldapadd -Y EXTERNAL -H ldapi:/// -f changepwd.ldif
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
modifying entry "olcDatabase={0}config,cn=config"
```

查看 olcDatabase={0}config 内容，

cat /etc/openldap/slapd.d/cn\=config/olcDatabase\=\{0\}config.ldif，新增了一个 olcRootPW 项

```
creatorsName: cn=config
createTimestamp: 20210124030467
olcRootPW:: e1NTSEF9b1hMalJJZGV1bDFFd1Y4Qm1ocTlsQWViKzdHdm5uUnU=
entryCSN: 20210124104553.005588Z#000000#000#000000
modifiersName: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
modifyTimestamp: 20210124104553Z
```

2.3. 导入基本 schema

我们需要向 ldap 中导入一些基本的 schema。这些 schema 文件位于/etc/openldap/schema/目录中，schema 控制着条目拥有哪些对象类和属性，可以自行选择需要的进行导入。依次执行下面的命令，导入基础的一些配置，我这里将所有的都导入一下，其中 core.ldif 是默认已经加载了的，不用导入。

```
# ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/cosine.ldif
# ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/nis.ldif
# ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/inetorgperson.ldif
# ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/collective.ldif
# ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/corba.ldif
# ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/duaconf.ldif
# ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/dyngroup.ldif
# ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/java.ldif
# ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/misc.ldif
# ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/openldap.ldif
# ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/pmi.ldif
# ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/ppolicy.ldif
```

```
[root@localhost myldap]# ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/corba.ldif
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
adding new entry "cn=corba,cn=schema,cn=config"

[root@localhost myldap]# ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/duaconf.ldif
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
adding new entry "cn=duaconf,cn=schema,cn=config"

[root@localhost myldap]# ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/dyngroup.ldif
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
adding new entry "cn=dyngroup,cn=schema,cn=config"

[root@localhost myldap]# ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/openldap/schema/java.ldif
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
adding new entry "cn=java,cn=schema,cn=config"
```

2.4. 修改域名

修改域名，新增 changedomain.ldif，这里自定义的域名为 node3.com，管理员用户账号为 admin。如果要修改，则修改文件中相应的 dc=node3.com,dc=com 为自己的域名。

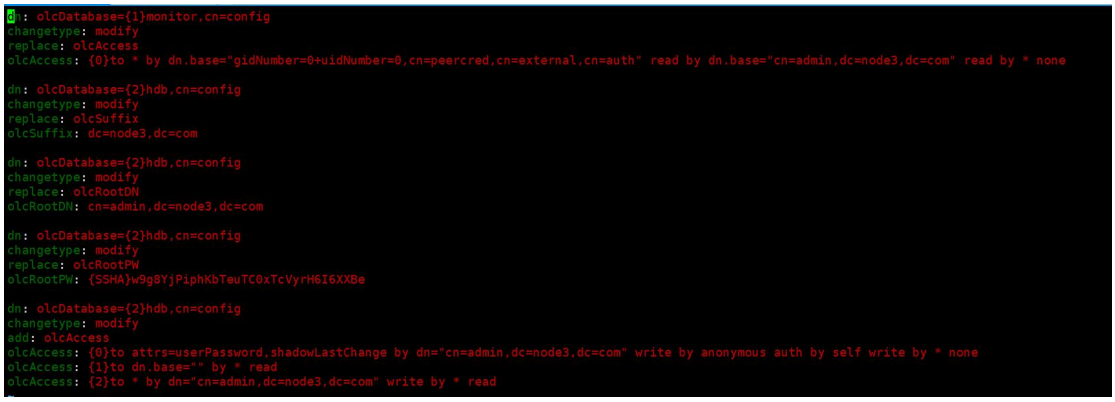
```
# vim changedomain.ldif
dn: olcDatabase={1}monitor,cn=config
changetype: modify
replace: olcAccess
olcAccess: {0}to * by dn.base="gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth" read by dn.base="cn=admin,dc=node3,dc=com" read by * none

dn: olcDatabase={2}hdb,cn=config
changetype: modify
replace: olcSuffix
olcSuffix: dc=node3,dc=com

dn: olcDatabase={2}hdb,cn=config
changetype: modify
replace: olcRootDN
olcRootDN: cn=admin,dc=node3,dc=com

dn: olcDatabase={2}hdb,cn=config
changetype: modify
replace: olcRootPW
olcRootPW: {SSHA}w9g8YjPiphKbTeuTC0xTcVyrH6I6XXBe

dn: olcDatabase={2}hdb,cn=config
changetype: modify
add: olcAccess
olcAccess: {0}to attrs=userPassword,shadowLastChange by dn="cn=admin,dc=node3,dc=com" write by anonymous auth by self write by * none
olcAccess: {1}to dn.base="" by * read
olcAccess: {2}to * by dn="cn=admin,dc=node3,dc=com" write by * read
```



执行命令，修改配置

ldapmodify -Y EXTERNAL -H ldapi:/// -f changedomain.ldif

最后这里有 5 个修改，所以执行会输出 5 行表示成功

```
[root@localhost myldap]# ldapmodify -Y EXTERNAL -H ldapi:/// -f changedomain.ldif
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
modifying entry "olcDatabase={1}monitor,cn=config"

modifying entry "olcDatabase={2}hdb,cn=config"

modifying entry "olcDatabase={2}hdb,cn=config"

modifying entry "olcDatabase={2}hdb,cn=config"

modifying entry "olcDatabase={2}hdb,cn=config"
```

至此，配置修改完了。

后 2.5 功能根据需求看是否需要：很多场景下，我们需要快速的查询某一个用户是属于哪一个或多个组的（member of），就会用到 memberof

2.6-2.7 为添加了一个组织两个组以及一个用户。

2.5. 启用 memberof 功能

新增 add-memberof.ldif，#开启 memberof 支持并新增用户支持 memberof 配置

```
# vim add-memberof.ldif
dn: cn=module{0},cn=config
cn: module{0}
objectClass: olcModuleList
objectclass: top
olcModuleload: memberof.la
olcModulePath: /usr/lib64/openldap

dn: olcOverlay={0}memberof,olcDatabase={2}hdb,cn=config
objectClass: olcConfig
objectClass: olcMemberOf
objectClass: olcOverlayConfig
objectClass: top
olcOverlay: memberof
olcMemberOfDangling: ignore
olcMemberOfRefInt: TRUE
olcMemberOfGroupOC: groupOfUniqueNames
olcMemberOfMemberAD: uniqueMember
olcMemberOfMemberOfAD: memberOf
```

```
dn: cn=module{0},cn=config
cn: module{0}
objectClass: olcModuleList
objectclass: top
olcModuleload: memberof.la
olcModulePath: /usr/lib64/openldap

dn: olcOverlay={0}memberof,olcDatabase={2}hdb,cn=config
objectClass: olcConfig
objectClass: olcMemberOf
objectClass: olcOverlayConfig
objectClass: top
olcOverlay: memberof
olcMemberOfDangling: ignore
olcMemberOfRefInt: TRUE
olcMemberOfGroupOC: groupOfUniqueNames
olcMemberOfMemberAD: uniqueMember
olcMemberOfMemberOfAD: memberOf
```

新增 refint1.ldif 文件

```
# vim refint1.ldif
dn: cn=module{0},cn=config
add: olcmoduleload
olcmoduleload: refint
```

```
dn: cn=module{0},cn=config
add: olcmoduleload
olcmoduleload: refint
```

新增 refint2.ldif 文件

```
# vim refint2.ldif
dn: olcOverlay=refint,olcDatabase={2}hdb,cn=config
objectClass: olcConfig
objectClass: olcOverlayConfig
objectClass: olcRefintConfig
objectClass: top
olcOverlay: refint
olcRefintAttribute: memberof uniqueMember    manager owner
```

```
dn: olcOverlay=refint,olcDatabase={2}hdb,cn=config
objectClass: olcConfig
objectClass: olcOverlayConfig
objectClass: olcRefintConfig
objectClass: top
olcOverlay: refint
olcRefintAttribute: memberof uniqueMember  manager owner
```

依次执行下面命令，加载配置，顺序不能错

```
# ldapadd -Q -Y EXTERNAL -H ldapi:/// -f add-memberof.ldif
# ldapmodify -Q -Y EXTERNAL -H ldapi:/// -f refint1.ldif
```

ldapadd -Q -Y EXTERNAL -H ldapi:/// -f refint2.ldif

```
[root@localhost myldap]# ldapadd -Q -Y EXTERNAL -H ldapi:/// -f add-memberof.ldif
adding new entry "cn=module{0},cn=config"

adding new entry "olcOverlay={0}memberof,olcDatabase={2}hdb,cn=config"

[root@localhost myldap]# ldapmodify -Q -Y EXTERNAL -H ldapi:/// -f refint1.ldif
modifying entry "cn=module{0},cn=config"

[root@localhost myldap]# ldapadd -Q -Y EXTERNAL -H ldapi:/// -f refint2.ldif
adding new entry "olcOverlay=refint,olcDatabase={2}hdb,cn=config"
```

2.6. 创建 node3 组织

在上述基础上，我们来创建一个 node3 company 的组织，node3 为域名，并在其下创建一个 admin 的组织角色(该组织角色内的用户具有管理整个 LDAP 的权限)和 People 和 Group 两个组织单元：

```
# vim node3.ldif
dn: dc=node3,dc=com
dc: node3
objectClass: top
objectClass: domain
o: node3

dn: cn=admin,dc=node3,dc=com
objectClass: organizationalRole
cn: admin
description: LDAP admin

dn: dc=hdp,dc=node3,dc=com
changetype: add
dc: hdp
objectClass: top
objectClass: dcObject
objectClass: organization
o: hdp

dn: ou=People,dc=hdp,dc=node3,dc=com
ou: People
objectClass: organizationalUnit

dn: ou=Group,dc=hdp,dc=node3,dc=com
ou: Group
objectClass: organizationalUnit
```

```
dn: dc=node3,dc=com
dc: node3
objectClass: top
objectClass: domain
o: node3

dn: cn=admin,dc=node3,dc=com
objectClass: organizationalRole
cn: admin
description: LDAP admin

dn: dc=hdp,dc=node3,dc=com
changetype: add
dc: hdp
objectClass: top
objectClass: dcObject
objectClass: organization
o: hdp

dn: ou=People,dc=hdp,dc=node3,dc=com
ou: People
objectClass: organizationalUnit

dn: ou=Group,dc=hdp,dc=node3,dc=com
ou: Group
objectClass: organizationalUnit
```

执行命令，添加配置，这里要注意修改域名为自己配置的域名，然后需要输入上面我们生成的密码

```
# ldapadd -x -D cn=admin,dc=node3,dc=com -W -f node3.ldif
```

添加结果为：

```
[root@localhost myldap]# ldapadd -x -D cn=admin,dc=node3,dc=com -W -f node3.ldif
Enter LDAP Password:
adding new entry "dc=node3,dc=com"

adding new entry "cn=admin,dc=node3,dc=com"

adding new entry "dc=hdp,dc=node3,dc=com"

adding new entry "ou=People,dc=hdp,dc=node3,dc=com"

adding new entry "ou=Group,dc=hdp,dc=node3,dc=com"
```

注：这里的 LDAP 密码为节点 2.2 配置的管理者密码

通过以上的所有步骤，我们就设置好了一个 LDAP 目录树：其中基准 dc=node3,dc=com 是该树的跟节点，其下有一个管理域 cn=admin,dc=node3,dc=com 和一个组织单元 dc=hdp,dc=node3,dc=com，其下有两个子属性 ou=People,dc=hdp,dc=node3,dc=com 及 ou=Group,dc=hdp,dc=node3,dc=com。

2.7. 创建新用户和新用户组的 ldif 文件

先生成一个密码 123456：

```
# slappasswd -s 123456
```

```
[root@localhost myldap]# slappasswd -s 123456
{SSHA}VmVEHNuPCzbblXTHBrsbXC0ts/NyIi5+
```

创建新用户的 ldif 文件

```
# vim ldapuser.ldif

#这里 testUser 用户，我将其加入到 testgroup 组中

# create new
# replace to your own domain name for "dc=***,dc=***" section
dn: uid=testldap,ou=People,dc=hdp,dc=node3,dc=com
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: shadowAccount
uid: testldap
cn: testgroup
sn: test
userPassword: {SSHA}32S2uLFahPZMqMzVYhT8fOKOp8RzremG
loginShell: /bin/bash
uidNumber: 2000
gidNumber: 3000
homeDirectory: /home/testldap


#这是添加一个用户组名为 testgroup 的 cn，在名为 Group 的 ou 下

dn: cn=testgroup,ou=Group,dc=hdp,dc=node3,dc=com
objectClass: posixGroup
cn: testgroup
gidNumber: 3000
memberUid: testldap
```

```
dn: uid=testldap,ou=People,dc=hdp,dc=node3,dc=com
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: shadowAccount
uid: testldap
cn: testgroup
sn: test
userPassword: {SSHA}32S2uLFahPZMqMzVYhT8fOKOp8RzremG
loginShell: /bin/bash
uidNumber: 2000
gidNumber: 3000
homeDirectory: /home/testldap

#这是添加一个用户组名为testgroup的cn，在名为Group的ou下
dn: cn=testgroup,ou=Group,dc=hdp,dc=node3,dc=com
objectClass: posixGroup
cn: testgroup
gidNumber: 3000
memberUid: testldap
```

向 openldap 服务端添加新用户 testldap

ldapadd -x -D cn=admin,dc=node3,dc=com -W -f ldapuser.ldif

为该用户修改密码为 123456 命令为：

ldappasswd -x -H ldap://192.168.1.107:389 -D "cn=admin,dc=node3,dc=com" -W "uid=testldap,ou=People,dc=hdp,dc=node3,dc=com" -s 123456

至此，添加了一个组织，两个组一个用户。openLDAP 配置到此结束。

三、第三方工具搭建（phpldapadmin）

3.1.安装 PHP 环境及依赖

yum -y install httpd php php-ldap php-gd php-mbstring php-pear php-bcmath php-xml

```
[root@localhost myldap]# yum -y install httpd php php-ldap php-gd php-mbstring php-pear php-bcmath php-xml
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: mirrors.163.com
 * extras: mirrors.163.com
 * updates: mirrors.aliyun.com
Resolving Dependencies
--> Running transaction check
---> Package httpd.x86_64 0:2.4.6-97.el7.centos will be installed
--> Processing Dependency: httpd-tools = 2.4.6-97.el7.centos for package: httpd-2.4.6-97.el7.centos.x86_64
--> Processing Dependency: /etc/mime.types for package: httpd-2.4.6-97.el7.centos.x86_64
--> Processing Dependency: libaprutil-1.so.0()(64bit) for package: httpd-2.4.6-97.el7.centos.x86_64
--> Processing Dependency: libapr-1.so.0()(64bit) for package: httpd-2.4.6-97.el7.centos.x86_64
---> Package php.x86_64 0:5.4.16-48.el7 will be installed
--> Processing Dependency: php-common(x86-64) = 5.4.16-48.el7 for package: php-5.4.16-48.el7.x86_64
--> Processing Dependency: php-cli(x86-64) = 5.4.16-48.el7 for package: php-5.4.16-48.el7.x86_64
```

3.2.下载最新的 phpldapadmin 安装包并配置

wget https://nchc.dl.sourceforge.net/project/phpldapadmin/phpldapadmin-php5/1.2.3/phpldapadmin-1.2.3.tgz

```
[root@localhost myldap]# wget https://nchc.dl.sourceforge.net/project/phpldapadmin/phpldapadmin-php5/1.2.3/phpldapadmin-1.2.3.tgz
--2021-01-24 03:02:04--  https://nchc.dl.sourceforge.net/project/phpldapadmin/phpldapadmin-php5/1.2.3/phpldapadmin-1.2.3.tgz
Resolving nchc.dl.sourceforge.net (nchc.dl.sourceforge.net)... 140.110.96.69, 2001:e10:ffff:1f02::17
Connecting to nchc.dl.sourceforge.net (nchc.dl.sourceforge.net)|140.110.96.69|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1115707 (1.1M) [application/octet-stream]
Saving to: 'phpldapadmin-1.2.3.tgz'

100%[=============================================================================================================>] 1,115,707   12.5KB/s   in 74s

2021-01-24 03:03:20 (14.6 KB/s) - 'phpldapadmin-1.2.3.tgz' saved [1115707/1115707]
```

将下载的压缩包解压至/var/www/html 目录下

#tar -zxvf phpldapadmin-1.2.3.tgz
#mv phpldapadmin-1.2.3 /var/www/html/phpldapadmin
ll /var/www/html/

```
[root@localhost myldap]# ll /var/www/html/
total 0
drwxrwxr-x 11 root root 208 Sep 30  2012 phpldapadmin
```

进入/var/www/html/phpldapadmin/conf 目录下，并将 config.php.example 重命名为 config.php

文件

```
[root@localhost config]# ll
total 28
-rw-rw-r-- 1 root root 24935 Sep 30  2012 config.php.example
[root@localhost config]# cp config.php.example config.php
[root@localhost config]# ll
total 56
-rw-r--r-- 1 root root 24935 Jan 24 03:05 config.php
-rw-rw-r-- 1 root root 24935 Sep 30  2012 config.php.example
```

编辑 config.php，将 OpenLDAP 的信息添加到该配置文件中

$servers->newServer('ldap_pla');
$servers->setValue('server','name','LDAP Server');
$servers->setValue('server','host','192.168.0.111');
$servers->setValue('server','port',389);
$servers->setValue('server','base',array('dc=node3,dc=com'));
$servers->setValue('login','auth_type','cookie');
$servers->setValue('login','bind_id','cn=admin,dc=node3,dc=com');
$servers->setValue('login','bind_pass','');
$servers->setValue('server','tls',false);

```
$servers->newServer('ldap_pla');
$servers->setValue('server','name','LDAP Server');
$servers->setValue('server','host','192.168.0.111');
$servers->setValue('server','port',389);
$servers->setValue('server','base',array('dc=node3,dc=com'));
$servers->setValue('login','auth_type','cookie');
$servers->setValue('login','bind_id','cn=admin,dc=node3,dc=com');
$servers->setValue('login','bind_pass','');
$servers->setValue('server','tls',false);
```

主要配置 LDAP 的服务器地址，Base DN，管理员账号及密码（可选择性的配置）

配置完成后启动 httpd 服务

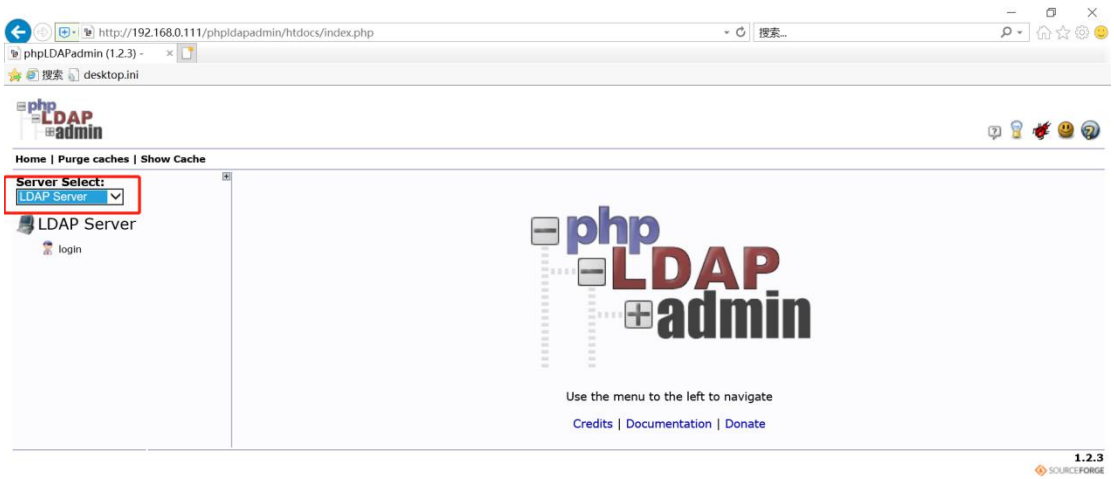
systemctl restart httpd
systemctl status httpd

```
[root@localhost config]# systemctl restart httpd
[root@localhost config]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)
   Active: active (running) since Sun 2021-01-24 03:11:05 PST; 9s ago
     Docs: man:httpd(8)
           man:apachectl(8)
 Main PID: 2047 (httpd)
   Status: "Total requests: 0; Current requests/sec: 0; Current traffic:   0 B/sec"
   CGroup: /system.slice/httpd.service
           ├─2047 /usr/sbin/httpd -DFOREGROUND
           ├─2049 /usr/sbin/httpd -DFOREGROUND
           ├─2050 /usr/sbin/httpd -DFOREGROUND
           ├─2051 /usr/sbin/httpd -DFOREGROUND
           ├─2052 /usr/sbin/httpd -DFOREGROUND
           └─2053 /usr/sbin/httpd -DFOREGROUND
```

四、phpldapadmin 访问及使用

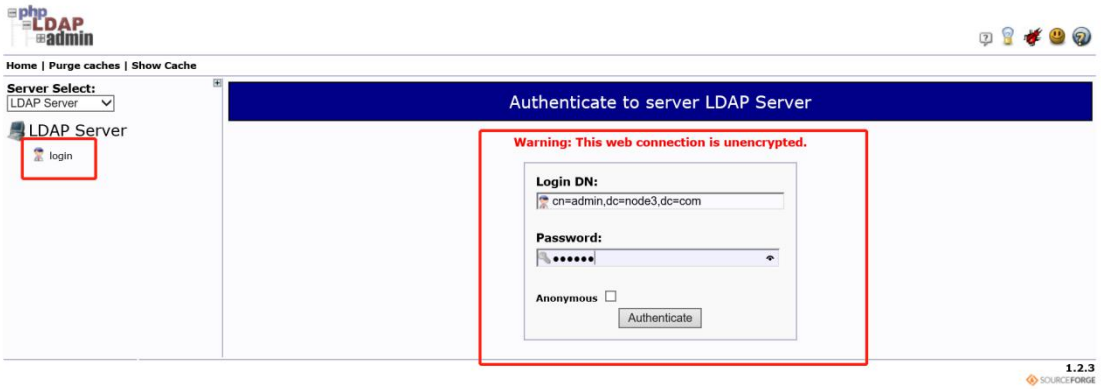
1.在浏览器输入 http://192.168.0.111/phpldapadmin/访问

2.点击左侧下拉菜单“Server Select”选择我们配置的 LDAP 服务



3.点击“登录”，配置文件中配置了管理员的账号所以默认显示为管理员账号

4.输入管理员密码进行认证，登录成功后显示如下界面：



5.以下为配置文件添加的域、组以及用户

Phpldapadmin 的使用参考链接：

https://mp.weixin.qq.com/s?__biz=MzI4OTY3MTUyNg==&mid=2247492103&idx=1&sn=1f9ee7117d9f2c4db5847e3bf8a5a6c8&chksm=ec29320edb5ebb18ac915d5e69073e035cf379130c96d9aa032eaf2b15de4962f7e924dfe2f1&scene=21#wechat_redirect

五、永洪 BI 配置

5.1. 权限配置

5.2. LDAP 配置

5.3. 结果展示