AMRES/RCUB

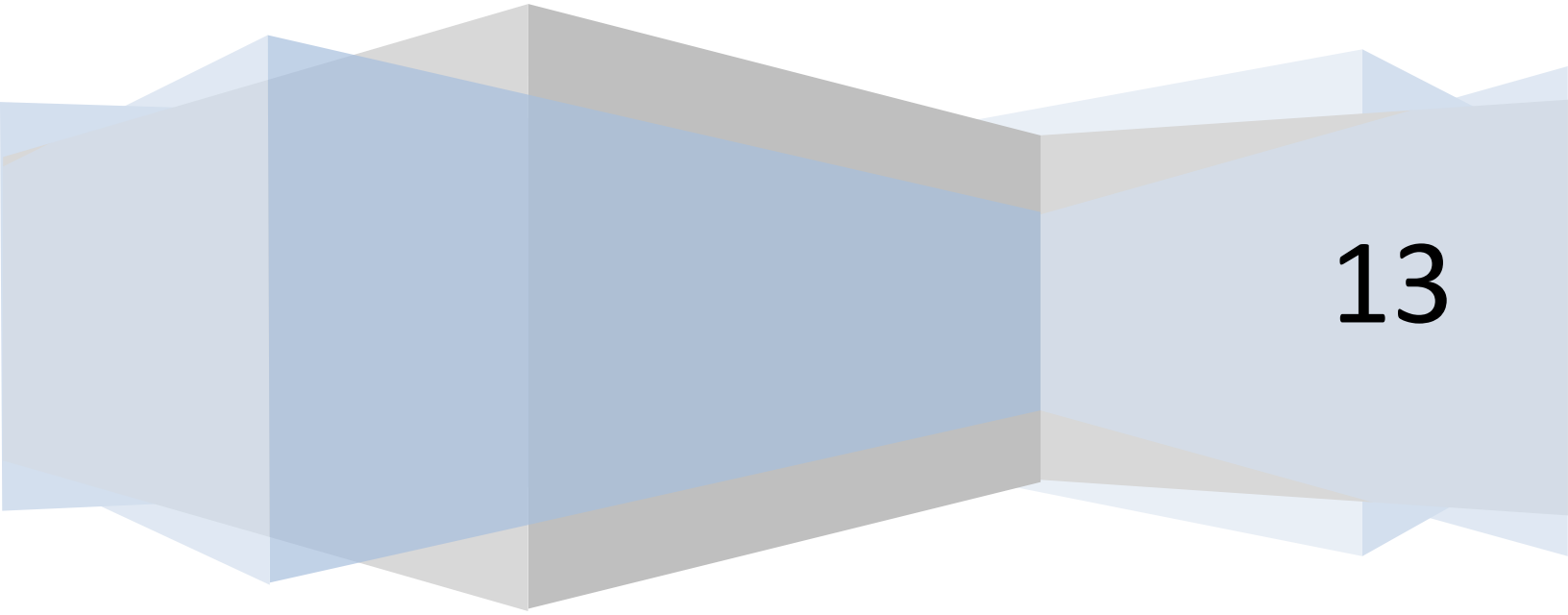
OpenLdap Instalacija i osnovna podešavanja

Žabljak IT konferencija - NA3T4 Géant Workshop

Ivan Ivanović

Jovana Palibrk

Marina Vermezović



13

Verzija 2

Sadržaj

OpenLdap Instalacija i osnovna podešavanja

Cilj ove laboratorijske vežbe je da se omogući administratorima uvid u rad sa OpenLdap aplikacijom koja će im pomoći prilikom kreiranja baze korisnika (zaposlenih/studenata). Nakon implementacije ovakvog sistema (baze podataka) administratorima će biti dat uvid u osnovne principe integracije OpenLdap baze podataka sa ostalim aplikacijama (Apache).

Laboratorijska vežba

Laboratorijska vežba obuhvata sledeće faze:

- Pokretanje CentOS operativnog sistema
- Instalacija OpenLdap aplikacije
- Podešavanja OpenLdap aplikacije
- Integracija OpenLdap aplikacije sa Apache serverom

Aplikacije i OS koji će se koristiti prilikom ove laboratorijske vežbe su:

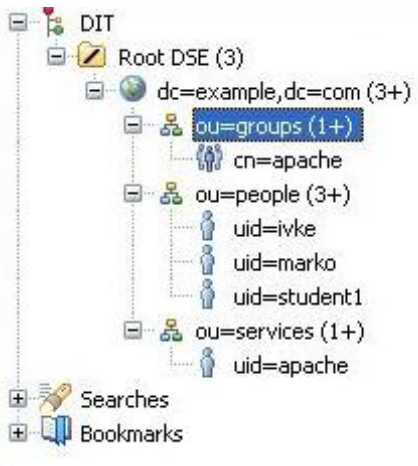
- CentOS 6.x – Operativni sistem
- OpenLdap
- Apache Directory Studio (Zahteva da se instalira Java)
- VmWare player
- Putty – SSH agent
- Apache web server

Sve prethodno navedene aplikacije i operativni sistem se mogu besplatno preuzeti sa Interneta.

Nakon instalacije i pokretanja VmWare player-a potrebno je selektovati ***Open a Virtual Machine*** opciju i pronaći prekopirani ***CentOS.vmx*** fajl. Markirati ga i selektovati ***OPEN*** opciju. Selektovati importovani operativni sistem i kliknuti na ***Play virtual machine***. Nakon pokretanja **obavezno** selektovati ***I moved it*** opciju.

Logički prikaz organizacije korisnika institucije

U vežbi će se napraviti Ldap baza za izmišljenu instituciju koja poseduje example.com domen.



Slika 4 – Struktura ldap baze koja će se napraviti u okviru vežbe

Cela struktura može da se napravi na više načina. Najlakše je pomoću ***Apache directory studio*** aplikacije. Pre kreiranja strukture na slici 4 potrebno je izvršiti početna podešavanja i pokrenuti OpenLdap server.

OS i parametri za SSH pristup

Pre početka instalacije i konfiguracije OpenLdap servera potrebno je pokrenuti operativni sistem na kome će se instalirati OpenLdap server. Za potrebe Laboratorijskih vežbi će se koristiti Vmware player. Potrebno je pokrenuti Vmware player i pokrenuti operativni sistem koji se zove ***CentOS6***.

Mrežna kartica na ***CentOS6*** serveru je podešena tako da dobija IP adresu putem dhcp protokola i informacije o IP adresi se može dobiti pomoću komande *ifconfig*. Pomoću putty aplikacije, koristeći ssh protokol potrebno je pristupiti serveru na IP adresi dobijenoj pomoću *ifconfig* komande. Root parametri za pristup serveru su:

- Username: root
- Password: openldap

Instalacija OpenLdap aplikacije

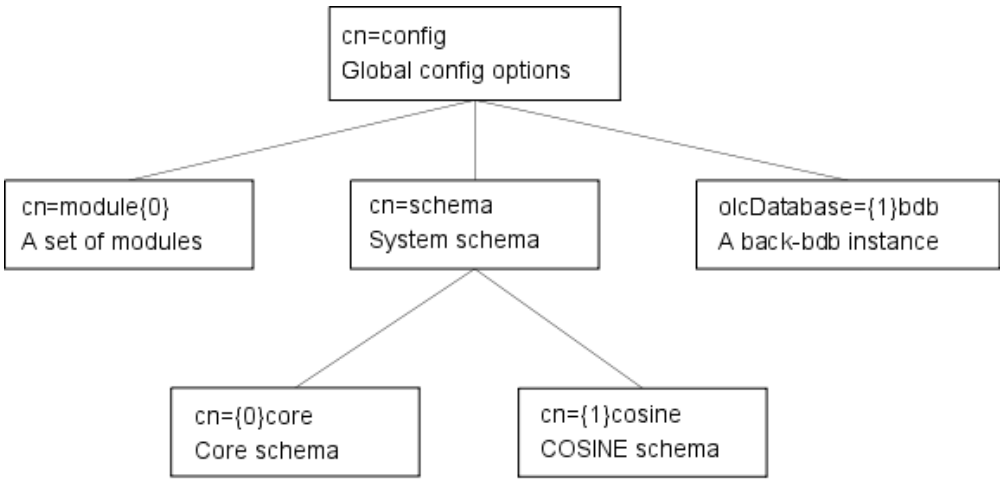
Openldap aplikacije se instalira pomoću sledeće komande:

yum install openldap-servers openldap-clients

Na sva ponuđena pitanja odgovoriti potvrdno sa **y**.
Softverski paket ***openldap-servers*** sadrži serversku OpenLdap aplikaciju a paket ***openldap-clients*** sadrži alate (ldapadd, slaptest,…..) koji se mogu koristiti za testiranje ispravnosti rada i popunjavanje baze OpenLdap serverske aplikacije. Komanda „***chkconfig slapd on“*** nam omogućuje da se OpenLdap demon (slapd process) pokrene prilikom restarta servera.
Nakon instalacije potrebno je podesiti osnovne parametre da bi se ispravno pokrenula OpenLdap aplikacija.

Osnovna konfiguracija

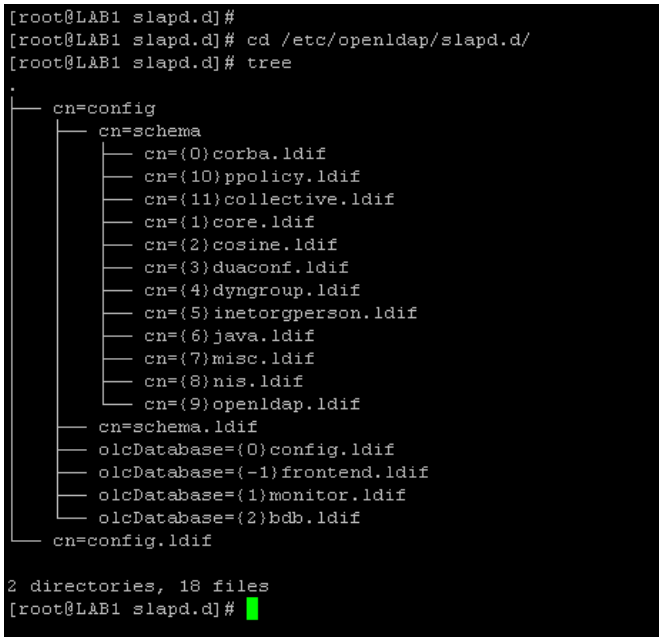
Slika 1 sadrži primer organizacije osnovnih globalnih konfiguracionih parametara za OpenLdap servis.



Slika 1 – Organizacija konfiguracionih parametara za OpenLdap

U poslednjim verzijama OpenLdap-a se struktura sa slike 1 direktno preslikava u fajl sistem organizaciju u

okviru linux direktorijuma u formi ldif fajlova i to je prikazano na slici 2.



Slika 2 – Osnovna konfiguracija u okviru linux fajl sistema

Kako je ldif forma nečitljiva i nezgodna za rad, osnovna konfiguracija će biti izvršena na drugi način. Prvo ćemo pronaći stari ***slapd.conf*** konfiguracioni fajl koji se u ranijim OpenLdap verzijama servisa koristio za konfigurisanje, zatim ćemo ga izmeniti prema našim potrebama i zatim ćemo ga konvertovati u željenu ldif formu.
Stari konfiguracioni fajl se može pronaći pomoću komande ***find / -name slapd.conf**** i rezultat pretrage je prikazan na slici 3.

Slika 3 – Pronalaženje starog slapd.conf konfiguracionog fajla

Fajl ***/usr/share/openldap-servers/slapd.conf.obsolete*** je potrebno prekopirati na lokaciju gde ćemo vršiti njegove izmene, recimo u ***/etc/openldap/*** direktorijum pošto njega većina linux sistema koristi za čuvanje konfiguracionih fajlova. Prilikom kopiranja, po uzoru na ranije verzije OpenLdapa, novi konfiguracioni fajl ćemo preimenovati u ***slapd.conf***.
Komanda za kopiranje je:

cp /usr/share/openldap-servers/slapd.conf.obsolete /etc/openldap/slapd.conf

Sledeća stavka je editovanje *slapd.conf* fajla i postavljanje željenih parametara. Prekopirani fajl sadrži default podešavanja i mi ćemo ih iskoristiti da bi konfigurisali OpenLdap aplikaciju. Nakon izvršenih izmena ćemo konvertovati slapd.conf konfiguracioni fajl u ldif formu. Naredna glava opisuje parametre u okviru *slapd.conf* fajla.

Primer default slapd.conf konfiguracionog fajla

```
#
# See slapd.conf(5) for details on configuration options.
# This file should NOT be world readable.
#

include         /etc/openldap/schema/corba.schema
include         /etc/openldap/schema/core.schema
include         /etc/openldap/schema/cosine.schema
include         /etc/openldap/schema/duaconf.schema
include         /etc/openldap/schema/dyngroup.schema
include         /etc/openldap/schema/inetorgperson.schema
include         /etc/openldap/schema/java.schema
include         /etc/openldap/schema/misc.schema
include         /etc/openldap/schema/nis.schema
include         /etc/openldap/schema/openldap.schema
include         /etc/openldap/schema/ppolicy.schema
include         /etc/openldap/schema/collective.schema
```

1

```
# Allow LDAPv2 client connections.  This is NOT the default.
allow bind_v2
```

2

```
# Do not enable referrals until AFTER you have a working directory
# service AND an understanding of referrals.
#referral     ldap://root.openldap.org

pidfile         /var/run/openldap/slapd.pid
argsfile        /var/run/openldap/slapd.args
```

3

```
# Load dynamic backend modules
# - modulepath is architecture dependent value (32/64-bit system)
# - back_sql.la overlay requires openldap-server-sql package
# - dyngroup.la and dynlist.la cannot be used at the same time

# modulepath /usr/lib/openldap
# modulepath /usr/lib64/openldap

# moduleload accesslog.la
# moduleload auditlog.la
# moduleload back_sql.la
# moduleload chain.la
# moduleload collect.la
# moduleload constraint.la
# moduleload dds.la
# moduleload deref.la
# moduleload dyngroup.la
# moduleload dynlist.la
```

```
# moduleload memberof.la
# moduleload pbind.la
# moduleload pcache.la
# moduleload ppolicy.la
# moduleload refint.la
# moduleload retcode.la
# moduleload rwm.la
# moduleload seqmod.la
# moduleload smbk5pwd.la
# moduleload sssvlv.la
# moduleload syncprov.la
# moduleload translucent.la
# moduleload unique.la
# moduleload valsort.la

# The next three lines allow use of TLS for encrypting connections using a
# dummy test certificate which you can generate by running
# /usr/libexec/openldap/generate-server-cert.sh. Your client software may balk
# at self-signed certificates, however.

TLSCACertificatePath /etc/openldap/certs
TLSCertificateFile "\"OpenLDAP Server\""
TLSCertificateKeyFile /etc/openldap/certs/password

# Sample security restrictions
#       Require integrity protection (prevent hijacking)
#       Require 112-bit (3DES or better) encryption for updates
#       Require 63-bit encryption for simple bind
# security ssf=1 update_ssf=112 simple_bind=64

# Sample access control policy:
#       Root DSE: allow anyone to read it
#       Subschema (sub)entry DSE: allow anyone to read it
#       Other DSEs:
#               Allow self write access
#               Allow authenticated users read access
#               Allow anonymous users to authenticate
#       Directives needed to implement policy:
# access to dn.base="" by * read
# access to dn.base="cn=Subschema" by * read
# access to *
#       by self write
#       by users read
#       by anonymous auth
#
# if no access controls are present, the default policy
# allows anyone and everyone to read anything but restricts
# updates to rootdn.  (e.g., "access to * by * read")
```

4

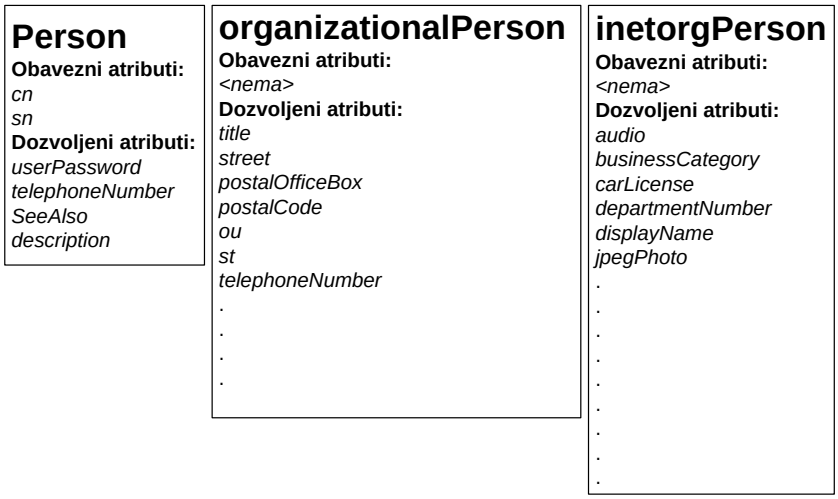
```
#
# rootdn can always read and write EVERYTHING!

# enable on-the-fly configuration (cn=config)
database config
access to *
      by dn.exact="gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth" manage
      by * none

# enable server status monitoring (cn=monitor)
database monitor
access to *
      by dn.exact="gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth" read
      by dn.exact="cn=Manager,dc=my-domain,dc=com" read
      by * none

########################################################################
# database definitions
########################################################################

database       bdb
suffix         "dc=my-domain,dc=com"
checkpoint     1024 15
rootdn         "cn=Manager,dc=my-domain,dc=com"
# Cleartext passwords, especially for the rootdn, should
# be avoided.  See slappasswd(8) and slapd.conf(5) for details.
# Use of strong authentication encouraged.
# rootpw               secret
# rootpw               {crypt}ijFYNcSNctBYg

# The database directory MUST exist prior to running slapd AND
# should only be accessible by the slapd and slap tools.
# Mode 700 recommended.
directory      /var/lib/ldap

# Indices to maintain for this database
index objectClass                       eq,pres
index ou,cn,mail,surname,givenname      eq,pres,sub
index uidNumber,gidNumber,loginShell    eq,pres
index uid,memberUid                     eq,pres,sub
index nisMapName,nisMapEntry            eq,pres,sub

# Replicas of this database
#replogfile /var/lib/ldap/openldap-master-replog
```

Pojašnjenje i konfiguracija pojedinih sekcija u okviru slapd.conf fajla

Bitni delovi konfiguracionog fajla su osenčeni i označeni rednim brojevima od 1 do 6. U daljem tekstu je dato objašnjenje za svaki od njih.

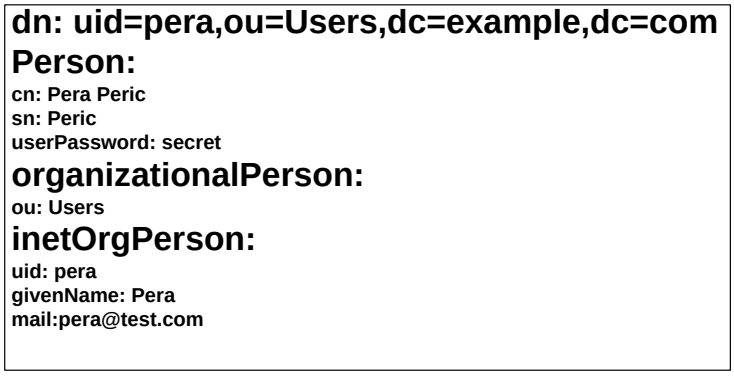
1. Include sekcija

Ovaj deo konfiguracije obuhvata šeme koje će se koristiti prilikom kreiranja unosa (npr. korisnika) u okviru OpenLdap aplikacije. Svaka od navedenih šema opisuje određene atribute (attributes) i klase objekata (objectclasses). Klase objekata grupišu atribute. Klase objekata mogu biti hijerarhijski organizovane, i u tom slučaju one nasleđuju osobine roditelja (parent objectclass). Primer je dat na slici 4.



Slika 4 – Primer nasleđivanja klasa objekata

Sve tri klase objekata na slici pripadaju ***core.schema*** šemi. Na slici 5 je dat primer jednog unosa u ldap bazu. Ovaj unos obuhvata atribute iz sve tri klase objekata. Klasa ***inetorgPerson*** nasleđuje atribute klase ***organizationalPerson*** a ona nasleđuje atribute klase ***Person***.



Slika 5 – Primer jednog unosa u OpenLdap bazi

U okviru šeme se takođe definišu i „obavezni“ atributi kao i mogućnost njihovog ponavljanja. (recimo korisnik mora da ima samo jedno ime i prezime ali može da ima/nema jedan/više brojeva telefona). U laboratorijskoj vežbi ćemo koristiti default šeme koje se već nalaze u konfiguracionom fajlu ali potrebno je znati da administratori institucija mogu da naprave svoje šeme i ubace ih u OpenLdap bazu. Postojeće default šeme su dizajnirane tako da pokriju većinu potrebnih atributa za organizaciju podataka u okviru institucije. Da bi kreirali bilo kakav unos u OpenLdap bazi potrebno je da taj unos sadrži bar jednu strukturnu objekt klasu. Kako kreiranje OpenLdap baze zahteva poštovanje niza pravila najlakši način da se to obavi je da se koristi neka druga aplikacija koja će voditi računa o tom skupu pravila. Jedna od takvih aplikacije je **Apache Directory studio** i nju ćemo koristiti za kreiranje šeme i ubacivanje korisnika.

2. Bind sekcija

Bind sekcija definiše koje verzije Ldap protokola su podržane od strane OpenLdap servera. Podrazumevana je verzija 3 a dodavanjem opcije *„allow bind_v2“* omogućuje se rad i sa starijom verzijom ldap protokola. Ostavićemo ovu sekciju nepromenjenu da bi omogućili starijim aplikacijama ,koje koriste ldap verziju 2, da komuniciraju sa našim serverom. Bind operacija predstavlja proces otvaranje sesije ka OpenLdap serveru i ovaj proces se koristi za autentifikaciju korisnika.

3. SLAPD parametri

Parametri u sekciji 3 definišu mesta na Linux operaivnom sistemu gde se čuvaju proces ID i dodatni parametri SLAPD procesa. Slapd proces predstavlja demon proces koji se koristi za OpenLdap na Linux serverskim platformama. Ovu sekciju nećemo menjati.

4. Sertifikati

Ovaj deo sekcije definiše mesta gde se čuvaju sertifikati koj se koriste prilikom korišćenja TLS protokola. Ova laboratorijska vežba ne obuhvata primer korišćenja sertifikata i enkripcije prilikom prenosa podataka pa ni ove parametre nećemo menjati. Njih je moguće naknadano promeniti ako administrator bude imao potrebe za korišćenjem enkripcije saobraćaja.

5. Config i Monitor baza

U okviru sekcije 5 se definiše baza sa osnovnim konfiguracionim parametrima i baza za monitoring. Kako ove baze sadrže osetljive podatke potrebno je i definisati pristup njima pomoću ***access list*** direktiva. Mi nećemo ništa menjati osim dela koji se tiče monitor baze. Promenićemo deo access liste

by dn.exact="cn=Manager,dc=my-domain,dc=com" read

u

by dn.exact="cn=admin,dc=example,dc=com" read

Na ovaj način smo dali pristup **admin** korisniku ***monitor*** bazi.

6. Baza korisnika

Sekcija 6 predstavlja skup informacija o bazi gde će se čuvati korisnici i ona predstavlja osnovu koju je potrebno konfigurisati pre unosa korisnika.

database bdb – predstavlja bekend bazu podataka gde će se čuvati unosi o korisnicima. Predefinisan je tip Oracle Berekeley DB (bdb), Podržan je i skup drugih DB (mysql, postgresql,ldif.....) Preporuka je da se koristi bdb i mi ćemo koristiti bdb bazu podataka. Prava pristupa bazi će biti objašnjena kasnije.

suffix "dc=my-domain,dc=com" – Predstavlja sufiks koji će se koristiti prilikom kreiranja osnovnog čvora u ldap strukturi. Mi ćemo ga promeniti u:

suffix "dc=example,dc=com"

Obično se za sufiks definiše domen institucije. (dc – Domain Component)

checkpoint 1024 15 – Definiše pravila kada se vrši provera unosa podataka iz memorije u bazu. Čim se u memoriji OpenLdap aplikacije nađe više od jednog megabajta (1024kB) ili je prošlo 15min od poslednjeg checkpoint-a potrebno je da se uradi checkpoint. Period izvršavanja checkpoint funkcije je 15 minuta. Ovaj parametar nećemo menjati jer u našem slučaju nećemo imati ogromne količine upisa a i čitanja iz baze.

rootdn "cn=Manager,dc=my-domain,dc=com" – Ovaj parametar predstavlja administratorski nalog koji ima najviše privilegije u okviru OpenLdap baze. Moramo ga promeniti tako da se podudara sa suffix parametrom. I za ovu bazi korisnik sa najvišim privilegijama je admin.

rootdn "cn=admin,dc=example,dc=com"

rootpw – Ovaj parametar se koristi da bi se definisao password za rootdn korisnika. Potrebno je koristiti ***slappasswd*** komandu da bi generisali password. Iz shell okruženja pokrenuti ***slappasswd*** komandu bez argumenata. Ukucati password ***test***. Rezultat izvršenja komande počinje sa {SSHA} direktivom koja govori OpenLdap serveru koji tip enkripcije se koristi. Moguće je koristiti i druge tipove enkripcije za skrivanje password-a. mi ćemo koristiti {SSHA}. Potrebno je ispod ***rootdn*** dodati:

rootpw {SSHA}ci1Ndw1DkG11loFkAr4EOuPLxKxG4B3Z

Ovde je korišćen password „test“.

directory /var/lib/ldap – Predstavlja direktorijum gde će se čuvati bdb baza podataka. Odavde se vidi da je konfiguraciona baza (u ldif formi) razdvojena od bdb (Oracle Berekeley DB) baze koja sadrži podatke o korisnicima.

index – Ovaj deo predstavlja skup parametara koji će se koristiti za indeksiranje podataka. Ima smisla koristiti ga prilikom rada sa ogromnom količinom podataka. U našem slučaju nema potrebe menjati ove parametre jer se ne koristi velika količina podataka. Indeksiranje se može vršiti za više atributa i to za više različitih načina pretrage u okviru ldap baze.

Konvertovanje u LDIF formu

Nakon završene konfiguracije potrebno je napraviti ldif konfiguracionu strukturu koju će OpenLdap koristiti. Pre konverzije u ldif formu potrebno je obrisati sve fajlove iz „***/etc/openldap/slapd.d/“*** foldera. Komanda za brisanje je:

rm –fr /etc/openldap/slapd.d/*

Zatim je potrebno izvršiti konverziju slapd.conf fajla u ldif formu.

slaptest –f /etc/openldap/slapd.conf –F /etc/openldap/slapd.d/

Ignorisati poruku upozoorenja ako je u formi:

bdb_db_open: database "dc=example,dc=com": db_open(/var/lib/ldap/id2entry.bdb) failed: No such file or directory (2).
backend_startup_one (type=bdb, suffix="dc=example,dc=com"): bi_db_open failed! (2)
slap_startup failed (test would succeed using the -u switch)

Nakon toga će se u folderu ***/etc/openldap/slapd.d/*** pojaviti željena ldif konfiguraciona struktura.
Pre pokretanja slapd demona potrebno je prekopirati ***DB_CONFIG.example*** fajl u folder gde će se nalaziti baza sa korisnicima. Takođe je potrebno promeniti mu ime u DB_CONFIG.

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

U ***DB_CONFIG*** fajlu se podešavaju parametri koji mogu da utiču na performanse rada OpenLdap baze. Mi ih nećemo menjati.

Pre pokretanja slapd porcesa proveriti da li je ldap korisnik vlasnik (owner) svih fajlova u ***/var/lib/ldap*** i ***/etc/openldap/slapd.d/*** direktorijumima. U slučaju da nije pomoću komanda

chown ldap.ldap /var/lib/ldap –R

chown ldap.ldap /etc/openldap/slapd.d/ -R

potrebno je promeniti vlasnika fajlova u ovim direktorijumima.

Pokrenuti slapd proces pomoću komande:

service slapd start

Sve prethodne komande, osim podešavanja slapd.conf fajla se nalaze u skripti ***reset.sh*** u /tmp direktorijumu ***(/tmp/reset.sh).*** Skriptu smo mi napravili da bi ubrzali rad na ovoj vežbi. Konverziju slapd.conf fajla u LDIF strukturu je moguće izvršiti pomoću ***/tmp/reset.sh*** komande. Ova skripta će se koristiti i kasnije kada se budu dodavale access liste i kada se javi potreba za editovanjem LDIF strukture, odnosno ponovo editovanje ***/etc/openldap/slapd.conf*** fajla i konverzija u LDIF formu.

Kreiranje OpenLdap DB strukture

Nakon pokretanja OpenLdap servera slapd deamon počinje da osluškuje ldap zahteve na TCP port 389. Proveriti da li je ovaj port pušten na iptables firewallu (Ugasiti firewall tokom vežbe pomoću komande ***service iptables stop*** ili dopustite prolaz zahtevima po tcp portu 389). Sledeći korak je kreiranje inicijalne strukture sa slike 4. Postoji više načina na koji se to može uraditi a mi ćemo pokriti dva načina. Prvi je pomoću ***ldapadd*** komande direktno iz shell okruženja. Ovaj način podrazumeva da već postoji definisana šema željene strukture baze.Drugi način je pomoću ***Apache Directory studia***. Preporučuje se drugi način.

Kreiranje pomoću *ldapadd* komande i predefinisane šeme baze

Koristiće se već predefinisana baza koja se nalazi u fajlu u /tmp folderu (/tmp/example.ldif). Komanda za dodavanje inicijalne strukture je:

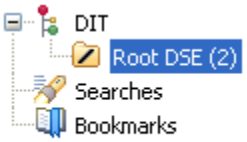
ldapadd -x -W -D 'cn=admin,dc=example,dc=com' -f /tmp/example.ldif

Potrebno je koristiti password koji je prethodno definisan za admin nalog u okviru slapd.conf fajla.

Kreiranje pomoću Apache Directory Studia (ADS)

- Pokrenuti ***ADS*** i selektovati opciju ***File->New->Ldap Browser->Ldap Connection***.
- Dati naziv konekciji, recimo TEST i u polje ***hostname*** upisati IP adresu OpenLdap servera i selektovati ***Next*** opciju.
- U polje „Bind DN or user" upisati ***cn=admin,dc=example,dc=com***.
- U polje „Bind password" upisati password koji je ranije definisan u slapd.conf fajlu i selektovati ***finish*** opciju.

Nakon toga će se otvoriti OpenLdap struktura kao na slici 6.



Slika 6 – OpenLdap struktura

Sledeći korak je dodavanje domain object klase, odnosno inicijalnog čvora ispod kojeg će se nalaziti naši podaci o korisnicima.

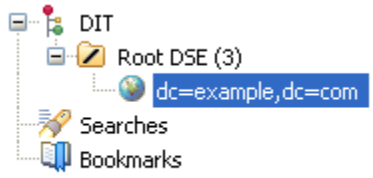
- Potrebno je desnim tasterom miša selektovati „Root DSE" čvor sa slike 6 i iz padajućeg menija selektovati ***New->New Context Entry*** opciju.
- U novom prozoru ostaviti selektovanu ***Create entry from scratch***opciju i zatim selektovati Next opciju.
- Zatim je potrebno pronaći ***domain*** klasu i dodati je pomoću opcije ***add***. Selektovati ***Next*** opciju. Za DN (Distinguished Name) upisati ***dc=example,dc=com*** i selektovati ***Next*** a potom ***Finish***.

Sledeći korak je dodavanje Organizacionih jedinica koje predstavljaju grupe korisnika (groups), korisnike (people) i servise (services).

Sve tri organizacione jedinice se konfigurišu na isti način samo je potrebno promeniti naziv organizacione jedinice

Dodavanje korisničkih naloga

- Desnim tasterom miša selektovati ***dc=example,dc=com*** čvor kao na slici 7.

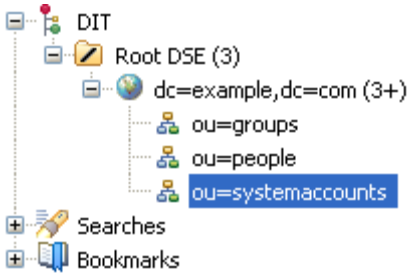


Slika 7 – Dodavanje Organizacionih jedinica

- Potom selektovati ***New->New Context Entry*** opciju. U novom prozoru ostaviti selektovanu ***Create entry from scratch***opciju i zatim selektovati Next opciju.
- Zatim je potrebno pronaći ***organizationalUnit*** klasu i dodati je pomoću opcije ***add*** i selektovati ***Next*** opciju.
- U novom prozoru za DN upisati ***ou=people,dc=example,dc=com.***

Isti postupak ponoviti za ***group*** i ***systemaccounts*** organizacione jedinice.

Nakon dodavanje sve tri organizacione jedinice dobiće se struktura kao na slici 8.

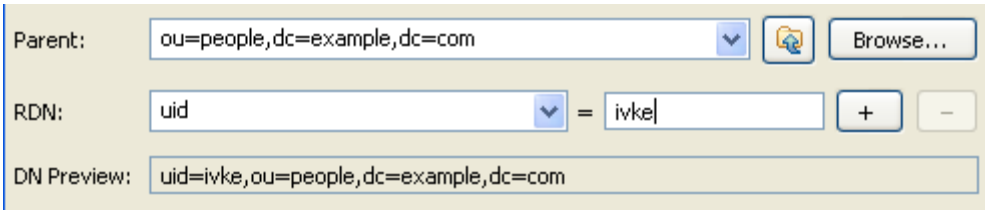


Slika 8 – Dodavanje organizacionih jedinica

Naredni korak je dodavanje korisnika.

- Desnim tasterom miša selektovati čvor „ou=people, dc=example,dc=com" i odabrati ***New->New Entry*** opciju.
- U novom prozoru ostaviti selektovanu ***Create entry from scratch*** opciju i zatim selektovati Next opciju.

- Zatim je potrebno pronaći ***inetOrgPerson*** klasu, dodati je pomoću opcije ***add,*** i selektovati ***Next*** opciju.
- Za RDN upisati ***uid*** a nakon znaka jednako upisati naziv korisnika. Slika 9 prikazuje ovaj proces.
- Selektovati Next opciju.

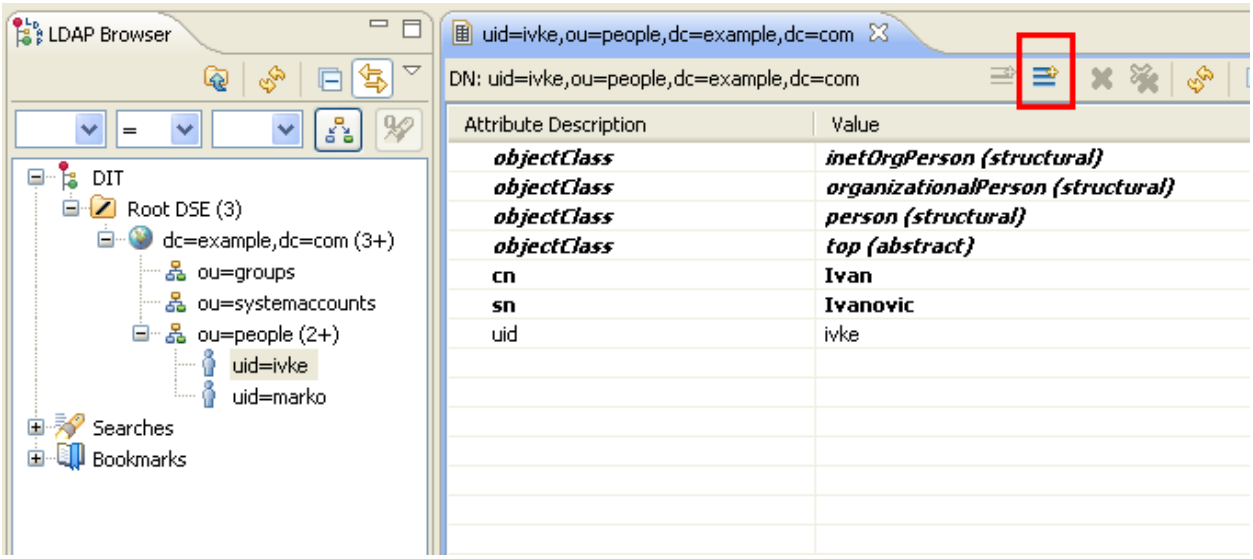


Slika 9 – Dodavanje korisnika

U novom prozoru će se pojaviti novokreirani korisnik sa svojim novim atributima. Primetiti da su cn (Common Name) i sn (Surname) atributi crveno obojeni. To znači da su to obavezni atributi i da ih je potrebno popuniti. Za ***cn*** se obično stavlja pravo ime korisnika a za ***sn*** njegovo prezime.Upisati ime i prezime. (Ivan Ivanovic)

Dodati još jednog korisnika koji ima ***uid*** atribut marko i ime (***cn***) i prezime (***sn***) Marko Markovic.

Da bi se korisnici uspešno autentifikovali (ili uspešno uradili bind proces) potrebno je da im se definiše atribut koj sadrži njihov password. Selektovati korisnika sa levim tasterom miša (***uid=ivke, ou=people, dc=example,dc=com***) i zatim odabrati opciju ***New Attribute*** kao na slici 10.



Slika 10 – Dodavanje userPassword atributa

Pronaći ***userPassword*** atribut i selektovati ***Finish*** opciju. Upisati novi password i selektovati OK opciju.

Svi korisnici u ***ou=people,dc=example,dc=com*** grani su jedinstveno definisani sa ***uid*** (User ID) atributom. Ne možemo imati dva korisnika sa istim ***uid*** atributom.

Dodavanje grupe

Za ***group*** organizacionu jedinicu je potrebno definisati grupu koja ima drugačije atribute nego korisnici i tu će se koristiti ***groupOfUniqueNames*** objekt klasa.

- Kao i u prethodnom slučaju potrebno je selektovati desnim tasterom miša ***ou=groups, dc=example, dc=com*** a potom odabrati ***New->New Entry*** opciju.
- Pronaći ***groupOfUniqueNames*** objekt klasu i dodati je pomoću opcije ***add***, kliknuti ***Next.***
- Za RDN upisati ***cn*** a nakon znake jednako ***apache*** (videti sliku 9). Selektovati ***Next***.
- Za ***uniqueMember*** atribut upisati ***uid=ivke,ou=people,dc=example,dc=com***.

Dodavanje sistemskih korisnika

Kao za slučaj dodavanja korisnika u ***ou=people,dc=example,dc=com*** granu dodati i novog korisnika u ***ou=systemaccounts, dc=example,dc=com*** granu.

- Objekt klasa novog korisnika u ***ou=systemaccounts,dc=example,dc=com*** grani je ***inetOrgPerson***.
- RDN je ***cn=apache***.
- **sn** atribut je takođe ***apache***
- Novom korisniku ***cn=apache,ou=systemaccounts,dc=example,dc=com*** je potrebno dodati ***userPassword*** atribut i postaviti password ***apache***.

Kontrola pristupa

Nakon inicijalnog podešavanja, svaki korisnički unos koji ima definisan ***userPassword*** može da se uloguje i da pročita **sve** atribute ostalih korisnika. To može da predstavlja sigurnosni propust i potrebno je nekako zaštititi određene atribute i korisnike. Za ove svrhe se koriste OpenLdap Access Liste. Access liste se proveravaju sekvencijalno sve dok se ne javi prvo poklapanje sa uslovom koji je definisan u njima.

Sledeći primer predstavlja jednostavnu kontrolu pristupa:

access to dn.base="" by * read

- Ovu direktivu je potrebno dodati da bi obezbedili svim korisnicima da mogu da provere (read – samo da pročitaju) osnovne funkcije koje naš OpenLdap podržava. (npr. verzija ldapa).

access to attrs=userPassword
by dn="cn=apache,ou= systemaccounts,dc=example,dc=com" write
by anonymous auth
by self write
by * none

- Ova direktiva kaže da:

- ***cn=apache,ou= systemaccounts,dc=example,dc=com*** može da menja ***userPassword*** atribute svim unosima koji poseduju ***userPassword*** atribut.
- Anonimni korisnici mogu da se priđu atributu ***userPasseword*** samo u cilju autentifikacije.
- Korinik koji poseduje ovaj atribut može da ga menja (promena sopstvenog passworda)
- Sve ostalo je zabranjeno

access to dn.subtree="ou=people,dc=example,dc=com"
by self write
by anonymous auth
by dn="cn=apache,ou= systemaccounts,dc=example,dc=com" read
by * none

Ova ACL direktiva kaže da:

- Samo korisnivci u ***ou=people,dc=example,dc=com*** grani mogu da menjaju svoje atribute.
- Anonimni korisnik može da se autentifikuje u ***ou=people,dc=example,dc=com*** grani***.***
- ***cn=apache,ou= systemaccounts,dc=example,dc=com*** može da vrši izmene svih atributa u celoj ***ou=people,dc=example,dc=com*** grani.
- Sve ostalo je zabranjeno.

access to dn.subtree="ou=groups,dc=example,dc=com"
by self write
by dn="cn=apache,ou=systemaccounts,dc=example,dc=com" read
by * none

Ova ACL direktiva kaže da:

- Prisup ispod grane ***ou=groups,dc=example,dc=com*** imaju samo korisnici u toj grani i ***cn=apache,ou=systemaccounts,dc=example,dc=com*** sistemski nalog.

Pomoću komande

ldapsearch -D "uid=ivke,ou=people,dc=example,dc=com" -b 'dc=example,dc=com' -W

se pokušava pročitati sve ispod ***dc=example,dc=com*** čvora sa korisničkim nalogom ***uid=ivke,ou=people,dc=example,dc=com***. Na osnovu rezultata komande se može zaključiti da obični korisnici mogu čitati sve iz ldap baze pa i password atribute drugih korisnika.

Da bi primenili ACL potrebno je da se ponovo izmeni ***/etc/openldap/slapd.conf*** fajl.

Ispod „database bdb“ direktive u slapd.conf fajlu prekopirati prethodne access liste.

access to dn.base="" by * read

access to attrs=userPassword
by dn="cn=apache,ou=systemaccounts,dc=example,dc=com" write
by anonymous auth
by self write

```
  by * none

access to dn.subtree="ou=people,dc=example,dc=com"
  by self write
  by anonymous auth
  by dn="cn=apache,ou=systemaccounts,dc=example,dc=com" read
  by * none

access to dn.subtree="ou=groups,dc=example,dc=com"
  by self write
  by dn="cn=apache,ou=systemaccounts,dc=example,dc=com" read
  by * none
```

Zatim je potrebno pokrenuti ***/tmp/reset.sh*** skriptu koja će ponovo generisati konfiguracione LDIF fajlove za OpenLdap. Sada će se i u njima nalaziti nove ACL.

Proveriti šta sledeće komande mogu da pročitaju?

ldapsearch -D "uid=ivke,ou=people,dc=example,dc=com" -b 'uid=marko,ou=people, dc=example,dc=com' -W

ldapsearch -D "cn=apache,ou=systemaccounts,dc=example,dc=com" –b 'ou=people,dc=example,dc=com' –W

Koja od njih će pročitati podatke iz naše baze?

Integracija sa Apache softverom

Instalirati Apache pomoću komande ***yum install httpd.***

Autentfikacija

Inicajlna podešavanja kod autentifikacije Apache aplikacije su data u daljem tekstu. Zbog sigurnosti koristićemo sistemski nalog ***cn=apache,ou=systemaccounts,dc=example,dc=com*** da bi proverili da li korisnik koji pokušava da se uloguje postoji u OpenLdap bazi.

Bitni delovi HTTP konfiguracije:

<Directory /var/www/html/> - Direktorijum kojem pristupamo
AuthLDAPURL "ldap://localhost:389/ou=people,dc=example,dc=com?uid?sub" – Lokacija gde tražimo korisnika koji želi da se autentifikuje, tražimo UID atribut.
Require valid-user – Uslov je da user postoji u OpenLdap bazi i da može da se uspešno autentifikuje.

Editovati /etc/httpd/conf/httpd.conf fajl i na kraju fajla dodati:

<Directory /var/www/html/>
AuthType Basic
AuthName "Secure Area"
AuthBasicProvider ldap
AuthzLDAPAuthoritative Off
AuthLDAPURL "ldap://localhost:389/ou=people,dc=example,dc=com?uid"
AuthLDAPBindDN "cn=apache,ou=systemaccounts,dc=example,dc=com"
AuthLDAPBindPassword "apache"
Require valid-user
Options Indexes FollowSymLinks
AllowOverride All
Order allow,deny
Allow from all
</Directory>

Restartovati apache server pomoću komande ***service httpd restart***. Pristupiti apache web serveru iz lokalnog web browsera. (U URL polje ukucati ip adresu viruelne mašine)
Nakon svake izmene u httpd.conf fajlu restartovati apache (service httpd restart), slapd (service slapd restart) i očistiti keš iz web browsera.

Autorizacija

Za autorizaciju korisnika postoje dva pristupa. Prvi je da se proverava da li je autentifikovani korisnik član ***cn=apache,ou=groups,dc=example,dc=com*** grupe. U slučaju Apache servisa će se proveravati ***uniqueMember*** atribut.

Primer:

<Directory /var/www/html/>
AuthType Basic
AuthName "Secure Area"
AuthBasicProvider ldap
AuthzLDAPAuthoritative Off
AuthLDAPURL "ldap://localhost:389/ou=people,dc=example,dc=com?uid?sub"
AuthLDAPBindDN "cn=apache,ou=systemaccounts,dc=example,dc=com"
AuthLDAPBindPassword "apache"
Require ldap-group cn=apache,ou=groups,dc=example,dc=com
Options Indexes FollowSymLinks
AllowOverride All
Order allow,deny
Allow from all
</Directory>

Probajte da se ulogujete sa dva različita korisnička naloga. Neka je samo jedan nalog član ***cn=apache,ou=groups,dc=example,dc=com*** grupe***.***
Drugi način je kada se proverava da li autentifikovan korisnik poseduje određeni atribut. Recimo da jedan korisnik u ***ou=people,dc=example,dc=com*** ima ***title=apacheuser*** atribut.

<Directory /var/www/html/>
AuthType Basic
AuthName "Secure Area"
AuthBasicProvider ldap
AuthzLDAPAuthoritative Off
AuthLDAPURL "ldap://localhost:389/ou=people,dc=example,dc=com?uid?sub"
AuthLDAPBindDN "cn=apache,ou=systemaccounts,dc=example,dc=com"
AuthLDAPBindPassword "apache"
Require ldap-attribute title=apacheuser
Options Indexes FollowSymLinks
AllowOverride All
Order allow,deny
Allow from all
</Directory>

Sada se iz ***ou=people,dc=example,dc=com*** grane može ulogovati samo korisnik koji poseduje ***title=apacheuser*** atribut***.***