LDAP azonosítás a gyakorlatban – egy esettanulmány (1. rész)
A kiszolgáló telepítése

Az LDAP telepítéséről sok dokumentáció jelent már meg a Linuxvilág oldalain és más helyeken is. Sokoldalúsága miatt én is elhatároztam, üzembe állítok egy ilyen nagy tudású rendszert. A beüzemelés közben persze számos nehézségbe ütköztem, ezért ez az írás talán segítséget jelent mindazok számára, akik hasonlóan nagy fába vágják a fejszéjüket...

Munkahelyemen előzőleg *NIS* azonosítást használtam. A *NIS* hibátlanul működött, de a következők miatt a csere mellett döntöttem:

1. Nem felel meg maradéktalanul napjaink biztonsági követelményeinek: *NIS* használata esetén a kódolt jelszóállományt a felhasználók egyszerűen lekérdezhetik.

2. Sok felhasználó esetén kényelmetlenné válik azoknak nyilvántartása a rendszerállományokban (*/etc/passwd* és */etc/shadow*). Munkahelyemen körülbelül 750 felhasználó, illetve munkaállomás fiók van, évente több mint 100 változik.

3. *Samba*-kiszolgálókkal a *NIS* együttműködése körülményes: Ha egyszerre többféle profiltípusú felhasználóink vannak (kötelező-, vándor- és helyi profil), ezek együttes kezelése úgy lehetséges, hogy a *Samba* kiszolgálónak a saját adatbázisára is szükség van, így duplán vannak nyilvántartva a felhasználók.

4. Ha a *Samba* a saját adatbázisát (is) használja, a *Samba PDC* (elsődleges tartományvezérlő) és *Samba BDC* (tartalék tartományvezérlő) között az adatbázis szinkronizálás nehezen oldható meg.

Az *LDAP* azonosítás mentes a fenti problémáktól, de bonyolultabb az üzembe helyezése. Az alábbiakban *Debian 3.1r1 „sarge”* verzión elvégzett telepítést mutatom be.

LDAP kiszolgáló telepítése

Az *OpenLDAP* kiszolgálót a *slapd* csomaggal lehet telepíteni. Mindjárt telepítettem az *ldap-utils* csomagot is, amelyben az *LDAP* adatbázis-kezelő parancsok vannak:

```
apt-get install slapd
➥ldap-utils
```

Telepítésekor a következőket kérdezte meg:

- A *DNS* tartomány nevét: *cegnev.hu* Ide természetesen mindenki a saját cégének *DNS* nevét gépelje. Ha szükséges, aldomain is megadható (például *egyseg.cegnev.hu*) A *DNS* névből állítja elő az *LDAP* kiszolgáló kiindulási pontját (*search base*) például *dc=cegnev,dc=hu* illetve *dc=egyseg,dc=cegnev,dc=hu*
- A szervezet nevét (tetszőleges név)
- Az *LDAP* adminisztrátor jelszavát
- Működjön-e a régi *LDAPv2* protokoll is (nem)

A kérdések megválaszolása után a *Debian* telepítette az *OpenLDAP* kiszolgálót, a válaszoknak megfelelően létrehozta az */etc/ldap/slapd.conf* beállítófájlt, és elindította a szolgáltatást. Az adatbázis fájljai a */var/lib/ldap/* könyvtárba kerültek. Az */etc/ldap/slapd.conf* beállító fájlt mindjárt módosítottam. A kiszolgáló, egy kérésre alapértelmezés szerint 500 bejegyzést jelenít meg maximum, utána bontja a kapcsolatot, így védekezve a túlterhelés ellen. Ez a beállítás nem jelent problémát amikor egy felhasználó bejelentkezik, vagy e-mail címeket keres. Amikor viszont valamennyi felhasználót szerettem volna megjeleníteni (megnéztem a `repquota -a` paranccsal kinek mennyi tárterülete van szabadon), csak az első 500 felhasználó neve jelent meg, a többinél pedig a felhasználó azonosító. Ezért a `sizelimit unlimited` sor beírásával kikapcsoltam ezt a korlátozást. Saját */etc/ldap/slapd.conf* állományom az *1. Listában* látható. A kiszolgálót leállítani, elindítani, illetve újraindítani a következő parancsokkal lehet:

```
/etc/init.d/slapd stop
/etc/init.d/slapd start
/etc/init.d/slapd restart
```

1. Lista A slapd.conf állomány tartalma

```
# 500 soros korlát kikapcsolása
sizelimit unlimited

# Ne muködjön LDAPv2
# allow bind_v2

# Schema fájlok definiálják az
adatszerkezetet
include         /etc/ldap/
➥schema/core.schema
include         /etc/ldap/
➥schema/cosine.schema
include         /etc/ldap/
➥schema/nis.schema
include         /etc/ldap/
➥schema/inetorgperson.schema

# Schema fájlok helyességének
az ellenorzését kapcsolja be
schemacheck     on

# Muködéshez szükséges fájlok
helye
pidfile         /var/run/
➥slapd/slapd.pid
argsfile        /var/run/
➥slapd.args

# Naplózási szint
(hibakereséskor kell magasabb
szintre állítani)
loglevel        0

# Modul jellemzok beállítása
modulepath /usr/lib/ldap
moduleload back_bdb

# Adatbázis beállítások
backend  bdb
checkpoint 512 30
database        bdb

# Adatbázis kiindulási
pontja(search base)
suffix          "dc=cegnev,
➥dc=hu"

# Adatfájlok elérési útja
directory       "/var/lib/ldap"

# Milyen index állományok
legyenek
index           objectClass eq

# Rögzítésre kerüljön-e minden
módosítás idopontja
lastmod         on

# Ez a beállítás csak akkor
kell, ha slave kiszolgáló is
lesz
# replogfile /var/lib/
➥ldap/replog


# Jogosutság a jelszavakhoz:
#  admin részére írási jog,
mindenkinek a sajátjára írási
jog
#  mindenki másnak tiltva a
hozzáférés
access to attrs=userPassword
        by dn="cn=admin,
➥dc=cegnev,dc=hu" write
        by anonymous auth
        by self write
        by * none

# Jogosultság a jelszavakon
kívüli adatokhoz
#  admin részére írási jog,
mindenki másnak olvasási jog
access to *
        by dn="cn=admin,
➥dc=cegnev,dc=hu" write
        by * read
```

Működését a következő módon ellenőriztem:

```
ldapsearch -x -b "dc=cegnev,
➥dc=hu" -h localhost
```

Természetesen a `dc=cegnev,dc=hu` kiindulási pontot értelemszerűen változtatva (pl. ha valakinek a telepítéskor beállított *DNS* név *penzugy.nagyceg.hu*, akkor: `dc=penzugy,dc=nagyceg,dc=hu`)

Ha a kiszolgáló nem lenne elérhető (`ldap:bind: Can't contact LDAP server (-1)`), annak egyik oka lehet, hogy a */etc/hosts.allow* fájlban nincs megfelelő engedély beállítva. Ekkor szövegszerkesztővel módosítsuk, hogy a szolgáltatás elérhető legyen. Ennél a beállításnál tapasztalatom szerint nem adható meg a gép neve, csak *IP* cím, vagy *IP* tartomány. A következő sor engedélyezi a kiszolgáló elérését a helyi gépen:

```
slapd: 127.0.0.1
```

Kiszolgáló elérésének engedélyezése helyi gépen és a ***10.0.1.0-10.0.1.255 IP*** című gépekről:

```
slapd: 127.0.0.1 10.0.1.
```

Hálózatban természetesen másik számítógépről is kereshető az adatbázis. Ekkor értelemszerűen az `ldapsearch` parancsban a `-h` után a gép *DNS* nevét, vagy *IP* címét kell megadni. Ne feledkezzünk meg az adott gép IP címére is engedélyezni az elérést!

LDAP segédprogramok beállítása

A */etc/ldap/ldap.conf* fájl módosításával lehetőség van arra, hogy a fenti `ldapsearch` parancsot és a többi adatbázis-kezelő parancsot is sokkal egyszerűbben lehessen használni. Minden olyan gépen célszerű elvégezni, melyről az adatbázist el szeretnénk érni. Szövegszerkesztővel beírtam a következő sorokat (természetesen a többi gépen a kiszolgálót futtató gép IP címét kell megadni a `HOST` után!):

```
HOST    127.0.0.1
BASE    dc=cegnev, dc=hu
```

A fenti módosítás után a következő egyszerű módon is elérhető az adatbázis:

```
ldapsearch -x
```

Biztonsági másolat készítése az adatbázisról

Leállítottam az *LDAP* kiszolgálót, majd a következő paranccsal biztonsági másolatot készítettem az adatbázisról:

```
slapcat -l ./root/
➥ldapbackup.ldif
```

Nagyméretű adatbázisnál célszerű tömöríteni a biztonsági másolatot:

```
slapcat | gzip > ./root/
➥ldapbackup.ldif.gz
```

A biztonsági másolat készítése nem kötelező, de ha van biztonsági másolat és valami baj történik az adatokkal, akkor egyszerűen helyre lehet állítani azokat, a kiszolgáló leállítása után:

```
slapadd -l ldapbackup.ldif
```

LDAP kiszolgáló futtatása megadott felhasználó nevében

A *Debian* az *LDAP* kiszolgálót a *root* felhasználó és csoport nevében futtatja alapértelmezés szerint. Ez nem biztonságos, célszerűbb egy olyan fiókot és csoportot létrehozni a kiszolgáló futtatásához aminek nincs semmi más jogosultsága. Leállítottam a kiszolgálót, majd létrehoztam *slapd* néven rendszer-fiókot és -csoportot:

```
adduser --system --group
➥--no-create-home slapd
```

A */etc/default/slapd* fájlban szövegszerkesztővel megadtam, hogy melyik felhasználó és csoport nevében fusson a kiszolgáló:

```
SLAPD_USER="slapd"
SLAP_GROUP="slapd"
```

A */etc/ldap/slapd.conf* fájlban módosítottam az argumentumfájl elérési útját olyan könyvtárra, melyre a *slapd* felhasználónak írási jogosultságot állítottam be:

```
argsfile /var/run/slapd/
➥slapd.args
```

Elvégeztem a szükséges fájl és könyvtár tulajdonos és jogosultság módosításokat, hogy a *slapd* felhasználó hozzáférjen ezekhez:

```
chown -R slapd.slapd /var/lib/
➥ldap
chmod 750 /var/lib/ldap
chown slapd.slapd /etc/ldap/
➥slapd.conf
chmod 640 /etc/ldap/slapd.conf
chown -R slapd.slapd /etc/ldap/
➥schema
chmod 640 /etc/ldap/schema/*
chown -R slapd.slapd /var/run/
➥slapd
chown -R slapd.slapd /var/
➥spool/slurpd
```

LDAP kiszolgálót elindítottam, majd ellenőriztem a működését a következő paranccsal:

```
ldapsearch -x
```

Ha a kiszolgálót az előbbiekben leírtak szerint parancssorból telepítettem, akkor rendben működött a *slapd* tulajdonos és csoport nevében. Azonban ha *dselect*-el telepítettem, csak a *root* nevében volt hajlandó működni. Az ok, hogy a *dselect* „ajánlottan” feltelepítette a *db2.4-util* csomagot is, amiben *BDB* adatbáziskezelő segédprogramok vannak. Ha a csomag fel van telepítve, akkor a *slapd* indító programja (*/etc/init.d/slapd*) egy *BDB* adatbázis visszaállító parancsot is lefuttat (`db_recover`) és a */var/lib/ldap* könyvtár fájljainak egy része vissza kerül a *root* tulajdonába, így a *slapd* felhasználó azokat nem tudja használni. Kétféle megoldást találtam. Az egyik lehetőség, az */etc/default/slapd* fájlban a helyreállítás kikapcsolása:

```
TRY_BDB_RECOVERY=no
```

Én azt a megoldást választottam, hogy az */etc/init.d/slapd* indítóprogramba megkerestem az adatbázis-visszaállító sort (140. sor körül) és utána szövegszerkesztővel beírtam egy tulajdonos-módosító parancsot:

```
echo -n "running BDB recovery"
for dbdir in $bdb_envs; do
  reason="`$DB_RECOVER_CMD -eh
  ➥$dbdir 2>&1`'  ||  \
    db_recover failed $dbdir
  # Új sor a slapd felhasználó
    tulajdonába adására
  /bin/chown -R slapd.slapd
  ➥/var/lib/ldap
done
```

Az LDAP admin jelszó módosítása

Hagyományos módon az *LDAP* adminisztrátor nevét és jelszavát az */etc/slapd.conf* fájlban szokás megadni, a `suffix` sora után beírva a következő sorokat:

```
suffix          "dc=jjsoft,
➥dc=hu"

rootdn "cn=admin,dc=cegnev,
➥dc=hu"
rootpw jelszo
```

Ilyen esetben az itt megadott jelszó átírásával és a kiszolgáló újraindításával lehet megváltoztatni a jelszót.
A másik lehetőség (ahogy az `ldapsearch -x` parancs kiadásakor látszik is), az *admin* fiók adatbázisban történő tárolása. Ennek módosításhoz létrehoztam szövegszerkesztővel egy *adminpasswdmod.ldif* fájlt a következő tartalommal:

```
dn: cn=admin,dc=cegnev,dc=hu
changetype: modify
replace: UserPassword
UserPassword: ujjelszo
```

Majd az `ldapmodify` paranccsal módosítottam a jelszót. A parancs kiadása után kétszer még bekéri az *admin* aktuális jelszavát:

```
ldapmodify -x -D "cn=admin,
➥dc=cegnev,dc=hu" -W -f
➥adminpasswdmod.ldif
```

Lehetőség van mind a két esetben, a jelszó titkosított megadására is. Titkosított jelszó a `slappasswd` paranccsal állítható elő, mely többféle titkosítást ismer. Én *SSHA* titkosítású jelszót állítottam elő, és a parancs kimenetét egy `jelszo` nevű fájlba írattam:

```
slappasswd -h {SSHA} > jelszo
```

A fájlból szövegszerkesztővel másoltam át, ezért az `adminpasswdmod.ldif` fájl utolsó sora a következő képen módosult:

```
UserPassword: {SSHA}Kaj3ZHujK
➥c9iE427TFhpoI8HnA
```

Adatok hozzáadása, módosítás és törlése

Ha valaki meglévő *UNIX*, illetve *Samba* fiókokat vagy csoportokat kíván hozzáadni, vagy meglévőket átvinni, akkor ezt a bekezdést ugorja át, ezeket a továbbiakban ismertetem. Ezt a lehetőséget én is csak a működőképesség tesztelésére használtam.
Mind a három esetben szövegszerkesztővel létre kell hozni egy megfelelő tartalmú *.ldif* fájlt, majd a megfelelő *LDAP* adatbáziskezelő paranccsal a fájl tartalmával a műveletet végrehajtani.
A hozzáadást a következő paranccsal végezhetjük el:

```
ldapadd -x -D "cn=admin,
➥dc=cegnev,dc=hu" -W -f
➥ldapadd.ldif
```

2. Lista Csoport és fiók hozzáadásánál az ldapadd.ldif fájl tartalma

```
#  Csoportok tárolására
alkalmas egység létrehozása
Groups néven
dn: ou=Groups,dc=cegnev,dc=hu
objectClass: organizationalUnit
ou: Groups

#  Felhasználó fiókok
tárolására alkalmas egység
létrehozása Users néven
dn: ou=Users,dc=cegnev,dc=hu
objectClass: organizationalUnit
ou: Users

#  users nevu csoport
létrehozása. A gidnumber értéke
a csoport azonosítója
dn: cn=users,ou=Groups,
➥dc=cegnev,dc=hu
objectClass: posixGroup
cn: users
gidnumber: 1100
description: Felhasznalok
➥csoportja

# nagypeter felhasználó fiók
hozzáadása, uidNumber értéke a
fiók azonosítója
dn: uid=nagypeter,
➥ou=Users,dc=cegnev,dc=hu
uid: nagypeter
cn: nagypeter
givenName: Peter
sn: nagypeter
objectClass: person
objectClass: organizational
➥Person
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: top
objectClass: shadowAccount
userPassword: {SSHA}A2414Ev0j
➥UGP$Ai8HbIlAsLwUZ4NtZtmTD1
shadowLastChange: 13016
shadowMax: 99999
shadowWarning: 7
loginShell: /bin/bash
uidNumber: 1000
gidNumber: 1100
homeDirectory: /home/nagypeter
gecos: Nagy Peter,,,
mail: nagypeter@cegnev.hu
```

Ha módosítani szeretnénk a felhasználó e-mail címét, akkor a ***ldapmodify.ldif*** fájl tartalma így néz ki:

```
dn: uid=nagypeter,
➥ou=Users,dc=cegnev,dc=hu
changetype: modify
replace: mail
mail: ujcim@cegnev.hu
```

A módosításra szolgáló parancs:

```
ldapmodify -x -D "cn=admin,
➥dc=cegnev,dc=hu" -W -f
➥ldapmodify.ldif
```

Ha törölni szeretnénk a felhasználót, akkor az ***ldapdelete.ldif*** fájl tartalma:

```
dn: uid=nagypeter,
➥ou=Users,dc=cegnev,dc=hu
changetype: delete
```

A törlést a következő paranccsal hajthatjuk végre:

```
ldapmodify -x -D "cn=admin,
➥dc=cegnev,dc=hu" -W -f
➥ldapdelete.ldif
```

A parancsok végrehajtódását a szokásos módon, az ***ldapsearch -x*** paranccsal lehet ellenőrizni.

Szolga LDAP kiszolgálók telepítése

A biztonság kedvéért létrehoztam ***slave LDAP*** kiszolgálókat is, melyek a ***master*** kiszolgálón lévő adatokról tárolnak másolatot. Így ha leáll a ***master*** kiszolgáló, akkor is működik a hálózat. A beállításokat úgy végeztem, hogy csak a ***master*** kiszolgálón lehessen adatokat módosítani. ***Slave*** kiszolgálót az elsődleges (***PDC***), a tartalék (***BDC***) ***Samba*** tartományvezérlőkre, valamint a levelező kiszolgálónkra telepítettem. Teljes egészében az előbb leírtak szerint végeztem telepítésüket, de a ***master*** kiszolgálón módosítottam a beállítást és ***slave*** kiszolgálókon is eltérés van.
A ***master*** kiszolgálón a következő módosításokat végeztem el.
Létrehoztam egy ***replicator*** nevű felhasználót, akinek a nevében a kiszolgálók közötti automatikus szinkronizálás (replikáció) fog végrehajtódni minden egyes alkalommal, ha a ***master*** kiszolgálón változás történik. Ha éppen nem működik valamelyik ***slave*** kiszolgáló a változás idején, akkor a ***master*** kiszolgáló megjegyzi a szükséges változtatást és akkor frissíti, amikor először elérhető lesz.
A felhasználót az előbbiekben ismertetett módon hoztam létre az `ldapadd` paranccsal.

A ***replicatoradd.ldif*** fájl tartalma:

```
dn: cn=replicator,
➥dc=cegnev,dc=hu
objectClass:
➥simpleSecurityObject
objectClass: organizationalRole
cn: replicator
description: LDAP replicator
userPassword: jelszo
```

A fájl tartalmát az adatbázishoz adtam:

```
ldapadd -x -D "cn=admin,
➥dc=cegnev,dc=hu" -W -f
➥replicatoradd.ldif
```

Létrehoztam a */var/log/ldap* könyvtárat:

```
mkdir /var/log/ldap
```

A ***master*** kiszolgálót leállítottam, majd módosítottam az */etc/ldap/slapd.conf* fájl tartamát:

```
# Beállítottam a replikációs
napló fájl helyét
replogfile   /var/log/ldap/
➥replog.log

# replikáció beállítása. Több
slave kiszolgáló esetén
#  mindegyiknek létre kell
hozni egy ilyen bejegyzést!
# A host= után a slave
kiszolgáló IP címe, vagy DNS
neve van, majd a portja
# Figyelem!!!! Itt a
titkosított jelszó nem muködik!
replicahost=192.168.1.101:389
  bindmethod=simple
  binddn="cn=replicator,
  ➥dc=cegnev,dc=hu"
  credentials=
  ➥replicator_jelszava
```

```
# Módosítottam még a
hozzáférési jogokat is úgy,
#   hogy a replicator is
olvashassa a jelszavakat
access to attrs=userPassword
  by dn="cn=admin,dc=cegnev,
  ➥dc=hu" write
  by dn="cn=replicator,
  ➥dc=cegnev,dc=hu" read
  by anonymous auth
  by self write
  by * none

access to *
        by dn="cn=admin,
        ➥dc=cegnev,dc=hu"
        ➥write
        by * read
```

A *master* kiszolgálót ezután újraindítottam.
Slave kiszolgálókon végzett beállítások a következők voltak:

A slave kiszolgálókon is létrehoztam a *replicator* nevű felhasználót, az előbbiekben leírt módon.
Kiszolgálót leállítottam, majd módosítottam az */etc/ldap/slapd.conf* fájl tartamát:

```
#  A replog sort nem szabad
bekapcsolni!

# Kinek a nevében történik a
replikálás és a master LDAP
címe, vagy DNS neve
updatedn  "cn=replicator,
➥dc=cegnev,dc=hu"
updateref
➥"ldap://192.168.1.100"
```

Módosítottam még a hozzáférési jogokat is úgy, hogy a *replicator* módosíthatja, az *admin* pedig csak olvashatja a jelszavakat:

```
access to attrs=userPassword
  by dn="cn=admin,dc=cegnev,
  ➥dc=hu" read
  by dn="cn=replicdator,
  ➥dc=cegnev,dc=hu" write
  by anonymous auth
  by self write
  by * none

access to *
        by dn="cn=replicator,
        ➥dc=cegnev,dc=hu"
        ➥write
        by * read
```

A kiszolgálót újraindítottam.
Fontos, hogy a *replikálás* beállítása előtt a *master* és a *slave* kiszolgálók adatbázisának a tartalma egyforma legyen!
Ez biztosítható olyan módon is, hogy az ismertetett `slapcat` paranccsal a *master* kiszolgálóról készített biztonsági másolatot a `slapadd` paranccsal a *slave* kiszolgálókra másoljuk.

A második részben a *Linux* kliensek beállítását mutatom be.

Jászberényi József

Szeret biciklizni, kirándulni, olvasni, sörözni és szabadban főzni.

A stratégiai játékoktól a műszaki CAD programokig sok minden érdekli. Legtöbbet szerverprogramokkal foglalkozik és néha mérgelődik.

(jaszberenyij@pattantyus-gyor.sulinet.hu)