Pehelysúlyú könyvtárelérési protokoll

Azonosítsuk Linux-rendszerünk felhasználóit a könnyen alakítható LDAP segítségével.

Az Interneten könnyűszerrel bukkanhatunk olyan nagy adatmennyiségre, amely gyakorta igen különböző könyvtárszerkezetbe van rendezve. A könyvtárak szabványosítására tett első próbálkozás az X.500 volt, amely a címlistát az X.400 levelezőrendszerhez tette elérhetővé. Ebből a könyvtárból az adatokat a DAP (Directory Access Protocol) protokoll segítségével lehetett kinyerni, ez azonban meglehetősen nehézsúlyú volt.

A későbbiekben számos más könyvtárat is kifejlesztettek, és ezek – bár néha meglehetősen különbözőek voltak – többnyire az X.500 szabványban gyökereztek. Kijött azonban egy új adatelérési protokoll is: a pehelysúlyú könyvtárelérési protokoll (Lightweight Directory Access Protocol – LDAP), amelyet csaknem az összes jelenlegi számítógépfajtán fel lehet használni szinte bármely X.500-zal együttműködő könyvtár adatainak az eléréséhez.

Eme pehelysúlyú könyvtárelérési protokoll használatát mutatjuk be ebben a cikkben. Elsőként azt vizsgáljuk, miképpen lehet olyan könyvtárszerkezetet összeállítani, amelyből az adatokat LDAP segítségével lehet elérni, majd rátérünk a protokoll Linux alatti használatának ismertetésére. Végezetül azzal foglalkozom, hogy miként alkalmazhatjuk az LDAP-könyvtárat akár rendszerünk felhasználóinak azonosítására is, ez ugyanis a NIS (Network Information Service) mellett érdekes választásnak ígérkezik.

A szakszókincs használata

Az LDAP elsődleges célja az X.500-megfelelő könyvtárszerkezetek adatainak az elérése. Ez a könyvtár tulajdonképpen rendszerbe szervezett adatbázis, amelyben különböző típusú adatok érhetők el. Gyakran használják nevek és címek tárolására, de ugyanilyen kényelmesen lehet benne a hálózati erőforrások adatait vagy bármi mást, például a linuxos gépek felhasználóinak azonosítására szolgáló adatokat is tárolni. A könyvtárrendszert a tárolóobjektumok alkotják, amelyeket gyakran könyvtárelemeknek is neveznek (Directory Components – DC). Ezeket a DC-ket akár a DNS-rendszerek tartományaival is össze lehet kapcsolni. Az LDAP-tároló akár DNS-tartományokhoz is láncolható; és ha a cégünk már rendelkezik bejegyzett DNS-tartománynévvel – például ***azlan.com*** –, az LDAP szervezési egységeket akár nevekhez is rendelhetjük, valahogy így:

```
ou=training, dc=azlan, dc=com.
```

A tárolóobjektumok tartalmazzák a levélobjektumokat, amelyeket gyakran egyszerűen csak bejegyzésnek hívnak, mert valóban csak azok az LDAP-adatbázisban. Egy ilyen levélre példa a felhasználó és a hozzárendelt levélcím, illetve minden más adat, amelynek segítségével az adott felhasználót azonosíthatjuk a gépünkön. Minden egyes ilyen bejegyzésnek egyedi neve van, amelyet megkülönböztetett névnek (Distinguished Name – DN) nevezünk. Például a bejegyzett tartománynévvel rendelkező ***Azlan*** részlegnél dolgozó ***Paul*** nevű felhasználó a következő megkülönböztetett nevet kapja `cn=Paul`, `ou=training`, `dc=azlan`, `dc=com`. Emellett a bejegyzésnek egy általános neve (Common Name – CN) is létezik, amely az objektum tárolójában egyedi azonosító – ez például a vezetéknév lehet.

Ezen objektumok tulajdonságai adják meg az objektumhoz rendelt adathalmazt; a felhasználói objektum tulajdonsága lehet például a cím és a jelszó. Ha a Linux-felhasználókat az LDAP segítségével szeretnénk azonosítani, nagyon fontos, hogy ezeknek a tulajdonságoknak pontos értékeket adjunk, például a */etc/passwd* fájl ***UID*** mezőjének az értékeit, amennyiben a Linux-erőforrásokat el szeretnénk érni. A bejegyzések pontos meghatározásai, vagyis az, hogy hol helyezkedjenek el a könyvtárban, és milyen tulajdonságok csatlakozzanak hozzájuk, a sémában jelenik meg. Linux alatt a bejegyzések a ***slapd.oc.conf***, a tulajdonságok pedig a ***slapd.at.conf*** fájlban találhatók.

Az LDAP-adatbázisok általános adatlekérdező nyelve az LDAP adatcsere-formátum (LDIF). Amennyiben ezt szeretnénk használni, nagyon fontos, hogy az egyes bejegyzések kötelezően kitöltendő mezőit ne hagyjuk üresen; ellenkező esetben objektummeghatározáskor kellemetlen hibaüzeneteket kaphatunk. A kötelező tulajdonságokat a ***spad.oc.conf*** tartalmazza.

OpenLDAP

A Linux alatt talán a legtöbbet használt LDAP-változat az ***OpenLDAP*** (➲ http://www.openldap.org). LDAP-megfelelő üzleti könyvtárak is beszerezhetők, például a ***Novell eDirectory*** vagy a ***Netscape's directory***. Linuxon az OpenLDAP telepítése után (amely gyakran az alapértelmezett kiszolgálótelepítés része) néhány fájl másolódik a rendszerre. Mielőtt azonban belemerülnénk a tényleges beállítások taglalásába, vessünk egy gyors pillantást a felmásolt fájlokra.

Az LDAP-telepítés legfontosabb állománya egy önálló LDAP démon: a `slapd`. Ezt kell elindítanunk az LDAP-rendszer használatba vételéhez. Ha a hálózaton egynél több LDAP-kiszolgálót használunk, és az adatokat mindkét rendszeren meg szeretnénk ismételni, a `slurpd`-re is szükségünk lesz, ugyanis ez másolja le az adatokat a LDAP-főkiszolgálóról egy vagy több LDAP-alkiszolgálóra.

Az LDAP-kiszolgáló beállításához természetesen néhány beállításfájlt át kell szerkesztenünk. A legtöbb ilyen fájl a */etc/openldap* könyvtárban található, azonban nem árt, ha odafigyelünk, mert ugyanazok a fájlok néha más könyvtárakban is jelen lehetnek, például a */etc*-ben, és ez megnehezíti a helyes beállítást. Ha a

/etc/openldap fájljai egynél több helyen vannak jelen, én többnyire csak a /etc/openldap könyvtár fájljait hagyom meg, az összes többi helyre csak hivatkozásokat teszek.
A legfontosabb beállításfájl a ***slapd.conf***. A `slapd` szinte összes tulajdonságát itt állíthatjuk be, ezt fájlt azonban még a `slapd` futtatása előtt át kell szerkesztenünk. A slapd.conf mellett található még két másik fájl is, amelyek a sémát tartalmazzák: a slapd.oc.conf és a slapd.at.conf. A legtöbb esetben ezekkel nem kell foglalkoznunk, hagyjuk érintetlenül őket úgy, ahogy vannak. Előfordulhat viszont, hogy a hálózat működtetéséhez mégis át kell szerkesztenünk őket. A legutolsó beállításfájl az ***ldap.conf*** – ez egy kicsi, de annál fontosabb állomány, amelyet az LDAP-ügyfél használ annak a kiszolgálónak az azonosítására, amelyikről az adatokat le kell töltenie.
A beállításfájlokon kívül néhány parancs is található itt, amelyekkel adatokat gyűjthetünk a könyvtárunkba, illetve meggyőződhetünk róla, hogy van-e benne egyáltalán valami. Az `ldapadd` segítségével adatokat adhatunk a könyvtárhoz, az `ldapmodify` a már létező bejegyzések tulajdonságait módosítja, végül az `ldapsearch` használatával egy adott bejegyzést kereshetünk meg. Az LDIF-fájlok révén a fenti néhány és pár további parancs segítségével kezelni tudjuk könyvtárunk adatait.
Végezetül azokról a fájlokról beszélünk, amelyekre akkor lesz szükségünk, ha a rendszerünkkel LDAP-kiszolgálóhoz szeretnénk csatlakozni. Ezek más programcsomagok részei, így nem biztos, hogy a rendszeren telepítve vannak. A névkiszolgálás-kapcsoló által használt modul neve ***nss_ldap***. A ***/etc/nsswitch.conf*** fájlban kell megadnunk, hogy az adatokat ezentúl a LDAP-könyvtárból vegye és ne a ***/etc/passwd***-fájlból. A másik fontos csomag a ***pam_ldap***. Ez az a Linux alatti modul, amelynek segítségével a felhasználók az LDAP-adatbázisban beilleszthető azonosító modulokkal (Pluggable Authentication Modules vagy PAM) azonosíthatók. Az LDA-könyvtárat könnyű beállítani a rendszerhez, mindössze négy lépés szükséges hozzá.

1. A program telepítése

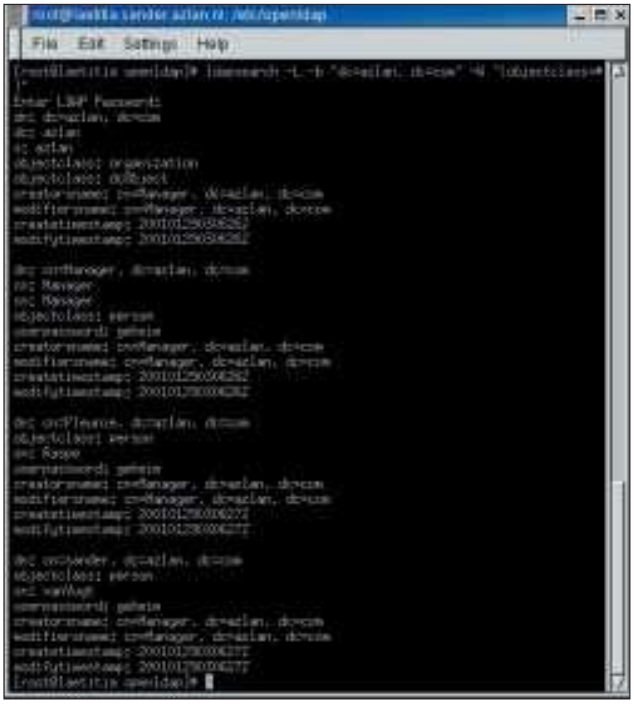
Ha az LDAP telepítése az alapértelmezett kiszolgálótelepítés folyamán még nem történt meg , a ➲ http://www.openldap.org címről vagy valamelyik tükörkiszolgálóról letölthetjük és telepíthetjük. A telepítéshez először tömörítsük ki:

```
tar -zxvf openldap-stable-xxxxx.tgz
```

ahol az xxxxx a letöltött fájl változatszáma. Lépjünk be az előzőleg létrejött könyvtárba, majd futtassuk le a `configure` parancsfájlt. Ez a parancsfájl ellenőrzi, hogy az LDAP telepítéséhez az összes feltétel rendelkezésre áll-e a rendszeren. Futtassuk a `make`-et; először el kell készítenünk a függőségeket a `make depend` segítségével, majd a programot egy egyszerű `make` utasítással le kell fordítani. Ezután futtassuk le a `make`-et a ***/test*** könyvtárban, ezzel ellenőrizve, hogy mindent helyesen fordítottunk-e. Végül a programot ténylegesen feltelepíthetjük a rendszerre: gépeljük be a `make install` parancsot abban a könyvtárban, amely a kicsomagoláskor keletkezett.

2. A slapd.conf beállításfájl átszerkesztése

A telepítést követően a /etc/openldap könyvtárban egy példát találhatunk a slapd.conf beállításfájlra. Ezt a saját rendszerünknek megfelelően át kell szerkesztenünk. Kezdetben nem szükséges túlbonyolítanunk fájlt; szerkesszük az *1. listához* (40. oldal) hasonlóra.
Vessünk egy pillantást a legfontosabb sorokra! Az első két sor feladata, hogy két másik beállításfájlt fűzzön be. Jelen esetben ezek a sémafájlok, amelyeket ugyan nem módosítottunk, de ettől még a `slapd`-nek meg kell mondani, hogy hol is találhatja meg őket. A `schemacheck = off` sor szintén nem túl érdekes – ez közli a `slapd`-vel, hogy szükségtelen ellenőriznie a sémát. A következő két sor ugyancsak külső fájlokra mutat: a ***slapd.pid***-re, amely a `slapd` által használt PID-számot tartalmazza, és a ***slapd.args***-ra, amely a `slapd` elindításakor megadott értékeket foglalja magában. Ezután egy olyan sort találunk, amely a felhasznált adatbázis típusát határozza meg. A megadható értékek: `ldbm`, `shell` és `passwd`, a legáltalánosabb az `ldbm`.



ldapsearch

Ezt három fontos bejegyzés követi. Az első a `"suffix dc=azlan, dc=com"` kezdetű sor, ez a bejegyzés határozza meg azt az általános tárolót, amelyben a `slapd` működni fog. Esetemben ez azonos annak a cégnek a DNS-nevével, ahol dolgozom. Ezt követi annak az azonosítója, akinek lehetősége lesz karbantartást vagy módosítást végrehajtani a teljes megkülönböztető névvel megadott adatbázison. A harmadik sor adja meg a karbantartó jelszavát; ez mint láthatjuk, egyszerű szövegként szerepel, ami azonban nem biztonságos, de erről még a későbbiek során szót ejtünk.
A `directory /usr/local/var/openldap-ldbm` sor azt a könyvtárat adja meg, ahol a LDAP-adatbázis telepítve van. Ellenőrizzük, hogy `700`-as módban van-e, valamint a `slapd`-folyamat tulajdonosa képes-e olvasni és írni.
E sorok után jó néhány olyan beállítás következik, amelyek ugyan nem feltétlenül szükségesek, de igen hasznosak lehetnek. Az első a `lastmod on`, amely nyomon követi azokat a felhasználókat, akik változtatásokat végeztek az objektumon. Erre a célra a `modifiersName`, a `modifyTimestamp`, a `creatorsName` és a `createTimestamp` tulajdonságokat használja. A következő beállítások segítségével indexelést végezhetünk. Az OpenLDAP sajnálatos módon nem éppen a leggyorsabb a fellelhető LDAP-könyvtárak között, ezért nem árt, ha néhány indexfájl segítségével kicsit felpörgetjük.
A `loglevel 64` beállítás, amely kiterjedt naplózást eredmé-

1. lista slapd.conf példa

```
include /etc/openldap/slapd.at.conf

include /etc/openldap/slapd.oc.conf
schemacheck off

pidfile /var/slapd.pid
argsfile /var/slapd.args

database ldbm
suffix "dc=azlan, dc=com"
rootdn "cn=manager, dc=azlan, dc=com"
rootpw secret

directory       /var/openldap-ldbm

lastmod on
index cn,uid

loglevel 64

defaultaccess   read

access to attr=userpassword
       by self write
       by dn="cn=manager, dc=azlan, dc=com"
             write
       by * compare
```

nyez, nagyon hasznos szolgáltatás lehet, ha gyorsabb működésre szeretnénk ösztökélni. Ennek legkisebb értéke `1`, a legnagyobb pedig 256. E két érték közt használhatjuk még a `2`, `4`, `8`, `16`, `32`, `64` és a `128` beállítást.
Végül néhány, a könyvtár elérési jogosultságára vonatkozó bejegyzést találunk. Az alapértelmezett `read` azt jelenti, hogy mindenki mindent olvashat, beleértve a jelszavakat is. A négy `access to attr=userpassword` kezdetű sor tartalmazza azokat a meghatározásokat, amelyek eldöntik, ki és mit tehet a jelszókkalaz adott könyvtárban. Az első sor engedélyezi, hogy mindenki módosíthassa a saját jelszavát. A rendszergazdának bármely jelszó átírására lehetősége nyílik, a felhasználók azonban csak olvasni tudják őket (természetesen, hiszen ez mindenképp szükséges ahhoz, hogy be tudjanak jelentkezni a rendszerbe).

3. A „slapd" indítása

Ha az ***slapd.conf*** fájlt kedvünkre átszerkesztettük, a következő lépés az `slapd` LDAP-démon indítása lesz. Ehhez az is elég, ha egyszerűen begépeljük, a `dn` kapcsolóval (ahol az `n` a kívánt hibakeresési szintet jelenti) azonban akár arra is utasíthatjuk, hogy minden nyomkövető üzenetet mutasson meg.

4. Adjunk adatokat a könyvtárhoz!

Most már áttérhetünk a következő lépésre, az adatok könyvtárba töltésére. Ebben a példában néhány egyszerű adatot fogunk beilleszteni, ehhez viszont előbb egy LDIF-fájlt kell szerkesztenünk a 2. listában (23. CD Magazin/OpenLDAP könyvtárban) láthatóhoz hasonló tartalommal.
Ha elkészítettünk egy a 2. listához hasonló fájlt, amelyet, tegyük fel, *~/users.ldif*-nek neveztünk, a következő paranccsal adhatjuk a könyvtárhoz:

```
ldapadd -D "cn=manager, dc=azlan, dc=com"
➥-W < ~/users.ldit
```

Meg kell adnunk a jelszót, amely azonos azzal, amit az slapd.conf-ban a rendszergazda-bejegyzéshez megadtunk. Ha minden jól ment, már képesek vagyunk adatot adni a könyvtárhoz.
Számos hibát azáltal is egyszerűen kiküszöbölhetünk, hogy ellenőrizzük, fut-e egyáltalán az `slapd` (igen, már kellene futnia), illetve nincs-e valahol a beállítás- vagy LDIF-fájlokban felesleges szóköz.

5. Nézzük meg, működnek-e a dolgok

Ha az adatot már a könyvtárhoz adtuk, a következő parancsokkal győződhetünk meg arról, hogy a rendszer működik-e:

```
ldapsearch -L -b "dc=azlan, dc=com"
➥-W "(objectclass=*)"
```

Eredményképp a könyvtárba helyezett összes adatot vissza kell kapnunk (lásd a *képen*).
Ha egyszer eljutottunk idáig, már elég sok mindent megtehetünk a könyvtárban. Többek között foghatjuk a böngészőnket és megtekinthetjük az LDAP-könyvtár adatait. És ez még csak nem is a legérdekesebb rész! További lehetőségként Linux-ügyfelünket úgy is beállíthatjuk , hogy az azonosítást többé ne a helyi jelszó- és árnyékfájlokon, hanem az LDAP-kiszolgálón keresztül végezze, így a felhasználók felügyeletét egyetlen pontba gyűjthetjük, s nem kell a saját jelszófájllal rendelkező számítógépek százaival foglalkoznunk. Ezt a következőképpen valósíthatjuk meg.

1. Telepítsük a programot

Mielőtt az ügyfelünket LDAP-kiszolgálón keresztüli azonosításhoz állítanánk be, bizonyosodjunk meg afelől, hogy minden szükséges program fel van-e telepítve. Ha RPM-alapú rendszert használunk, az ***openldap***, ***auth_ldap*** és ***nss_ldap*** csomagoknak kell fent lenniük. Ezt az `rpm -q` *`csomagnØv`* utasítással könnyen ellenőrizhetjük. Amennyiben nincsenek meg, a ➲ http://rpmfind.com címen megtalálhatjuk őket.

2. Az „ldap.conf" szerkesztése

A rendszereken gyakorta két ***ldif.conf*** nevű fájl is található. Az első a */etc* könyvtárban lelhető fel, és az `nss_ldap`, illetve a `pam_ldap` használja a szükséges adatok meghatározásához. A másik a */etc/openldap*-ban helyezkedik el, és az `ldapadd`, valamint az `ldapsearch` eszközök használják annak meghatározására, hogy melyik tárolón dolgozzanak. Amint már korábban is tettük: töröljük az egyiket, majd a munkánk leegyszerűsítése végett a helyére készítsünk közvetett hivatkozást a másik példányra. Ha elkészültünk, a szükséges adatokat beilleszthetjük. Egy egyszerű beállításhoz mindössze két sor szükséges:

```
BASE    dc=azlan, dc=com
HOST    laetitia.azlan.com
```

Az első sor határozza meg az alapértelmezett tárolót, ahol az ügyfél az adatokat keresni fogja, a második sor ad nevet az LDAP-kiszolgálónknak. Természetesen rendszerünknek képesnek kell lennie kiértékelni (resolve) ezt a nevet valamilyen DNS vagy hasonló eszközön keresztül, vagy az IP-címet kell használnunk.

3. Az „nsswitch.conf” szerkesztése

Következő lépésként meg kell adnunk a névszolgáltatás-kapcsolónak, hogy hol keresse az adatokat. Ezt a */etc/nsswitch.conf* fájl szerkesztésével tehetjük meg. A fájl a következő sorokat tartalmazhatja:

```
passwd: files ldap
shadow: files ldap
group: files ldap
```

Rendszerünk elsőként a fenti sorok segítségével kísérli meg helyi jelszófájlokon keresztül a felhasználók azonosítását, majd ha így nem jár sikerrel, az LDAP-adatbázison keresztüli azonosítással próbálkozik. Ha tehát a felhasználó a */etc/passwd* fájlban már létezik és a */etc/shadow*-ban jelszóval is rendelkezik, akkor az LDAP-rendszer használatára nem kerül sor.

4. PAM-beállítások szerkesztése

Ezután vessünk egy pillantást a PAM-fájlra. A legtöbb Linux-terjesztésben minden program ezt a korszerű módszert használja, ha csak egy csepp köze is van a felhasználóazonosításhoz. Ezt használják többek közt a `login`, az `ftp`, az `su`, az `ssh`, a `passwd` és további programok is. A PAM jelenlegi változataiban minden egyes ilyen fájlnak létezik beállításfájlja, amelyek alapesetben a */etc/pam.d* könyvtárban találhatók. Ezekben a beállításfájlokban adhatjuk meg, hogy az illető modul mely PAM-modulokat használhatja.
Amennyiben azt szeretnénk, hogy a ***login*** folyamat az azonosítást az LDAP-n keresztül végezze, a megfelelő beállításfájlnak valahogy úgy kell kinéznie, ahogyan azt a 3. listában (23. CD Magazin/OpenLDAP könyvtárban) láthatjuk.
Nézzük meg egy kicsit részletesebben! A felhasználó- és a jelszóadatokat négy folyamatban használjuk. Az első az azonosítás, ezt a PAM-fájlban az `auth` jelképezi. Ez a folyamat enged be minket a rendszerbe, illetve a feladatai közé tartozik a jelszó ellenőrzése is. A következő az `account` (felhasználói név), amely ellenőrzi, hogy a felhasználónak van-e valamilyen korlátozása, ami miatt nem jelentkezhet be a rendszerbe. Ezután következik a `password` (jelszó), amelyet akkor használunk, ha a jelszavunkat meg szeretnénk változtatni. Végül a `session` (munkamenet) határozza meg azokat a feladatokat, amelyeket akkor kell végrehajtani, ha egy olyan rendszeren szeretnénk más erőforrásokat használni, ahol már korábban azonosítottuk magunkat.
Minden ilyen modulnak külön jól meghatározott feladata adódik. Ezeket a feladatokat a PAM-modulok tartalmazzák, amelyek közül az egyik legfontosabb a ***pam_unix.so***. Ez a modul felügyeli a hagyományos jelszó-, illetve árnyékjelszó-azonosítás folyamatát és többnyire elengedhetetlen a rendszer eléréséhez. Ha azonban LDAP-t használunk, az is megfelel, ha az LDAP enged be, ezért a `pam_unix` sor meghívása előtt valahol léteznie kell egy sornak, amely a `pam_ldap`-t hívja meg. Ez nem kötelező (valószínűleg akkor is el szeretnénk érni a rendszert, ha az LDAP-kiszolgáló leáll), de elégséges. Következésképp ha a `pam_ldap` segítségével azonosítjuk magunkat, már nem szükséges a `pam_unix`-hoz is fordulnunk. A két nagyobb modul mellett még néhány kisebb modul is létezik, amelyek tárgyalásától most terjedelmi okok miatt eltekintünk.

5. Felhasználók létrehozása az összes szükséges tulajdonsággal

Miután megtettük a négy bevezető lépést, számítógépünk készen áll az LDAP-könyvtáron keresztüli azonosításra. Vajon a könyvtárunk szintén készen áll? Könyvtárunk azonosításhoz történő felkészítéséhez az összes szükséges felhasználói tulajdonságot be kell illeszteni, mindazokat, amelyek egyébként a */etc/passwd* és */etc/shadow* fájlokban megtalálhatók. Helyesen előállított felhasználói azonosítók nélkül miként használhatnánk értékes erőforrásainkat és e fájlok egyéb hasznos képességeit? Az adat megszerzéséhez olyan Perl-parancsfájlokat használhatunk fel, amelyeket kifejezetten a gépünkön található fájlok adatainak kigyűjtésére és LDAP-adatbázisba vitelére fejlesztettek ki, esetleg saját LDIF-fájlt is készíthetünk, amellyel felvihetjük a kívánt felhasználókat.
Ha az önműködő megoldás mellett döntünk, több Perl-parancsfájlból is választhatunk a ➲ http://www.padl.org-on. Olyan parancsfájlok is találhatók itt, amelyek majdnem minden beállítást össze képesek gyűjteni, legyen az NIS-adatbázis jelszófájl, a hostfájlunk, a networkfájlunk stb. Mielőtt azonban használnánk őket, az általános beállításfájlt, a ***migrate_common.ph***-t át kell szerkesztenünk. Ennél néhány olyan értéket kell megváltoztatnunk, amely az adat létrehozásának helyét határozza meg. Különösen fontos a `DEFAULT_MAIL_DOMAIN` és a `DEFAULT_BASE`; ezek határozzák meg azt a DNS-tartományt, ahol a felhasználók címei találhatók, illetve azt az LDAP-tárolót, ahol a felhasználókat létre kell hozni. Ha ez megtörtént, elkezdhetjük a bevitelt. Minden egyes adathoz egy-egy külön parancsfájlt találunk; a legérdekesebb közülük talán a `migrate_all_online.sh`, amely az összes hálózati adatot, illetve a `migrate_passwd.pl`, amely a felhasználókat gyűjti össze a rendszeren.
A másik lehetőség egy saját LDIF-fájl készítése, ennek tartalmát az `ldapadd` segítségével az adatbázishoz kell adnunk. Ne feledjük, az összes jogosultságot be kell állítanunk! A 4. listában (23. CD Magazin/OpenLDAP könyvtárban) bemutatott példán láthatjuk, hogyan valósíthatjuk ezt meg a legegyszerűbben.
Ennek a módszernek két hátránya van: az egyik, amikor a felhasználót az LDIF-adatbázisban létrehozzuk, de a felhasználói-könyvtár nem készül el önműködően (igaz, létezik egy ***pam_mkhomedir.so*** nevű PAM-modul, amely ezt a nehézséget is megszünteti). A másik gond a felhasználók jelszavaival kapcsolatos, sajnos ugyanis nincs rá jó módszer, hogy az adatbázisba titkosítva rejthessük el őket. Nem túl elegáns megoldásként javasolhatjuk, hogy a felhasználót hozzuk létre a /etc/passwd és /etc/shadow fájlokban, adjunk neki jelszót, majd a titkosított karaktersorozatot másoljuk ki a /etc/shadow-ból, és rakjuk be az LDIF-fájlba.
Ezután nem maradt más hátra, próbáljuk ki a rendszert: töröljük a felhasználót a helyi fájlokból, nyissunk meg egy bejelentkezési ablakot, és kíséreljünk meg bejelentkezni – ha minden jól ment, sikerrel járunk.

Sander van Vugt
(sander.van.vugt@azlan.nl) Hollandiában él. Linux-, Novell- és Nortel-szakoktatóként az Azlan Trainingnél dolgozik, és már jó néhány könyvet és cikket írt a Linuxról.

Kapcsolódó címek

OpenLDAP ➲ http://www.openldap.org
RPM Find ➲ http://www.rpmfind.com
Perl Scripts ➲ http://www.padl.org