LDAP teadmiste omandamise õppematerjal

Sisukord

1 Mõisted

kataloogiteenus- üks IT infrastruktuuri teenustest, mille kaks peamist kasutusvaldkonda on kasutajainfo haldamine ja autentimisteenuse pakkumine.

X.500- kataloogiteenuse standard, mis standardiseeriti 1980ndate lõpus CCITT (Commite' Consultatif International de Telegraphique et Telephonique) poolt. 1990ndate alguses loodi selle põhjal LDAP[1]. See defineerib DAP (*Directory Access Protocol*) ehk kataloogiteenuse protokolli, et teenuse kliendid saaksid luua ühenduse kataloogiteenuse serveriga[2].

LDAP (*Lightweight Directory Access Protocol*)- hajutatud kataloogidele ligipääsemiseks loodud internetiprotokoll, mis on kooskõlas X.500 andme- ja teenusemudeliga[3].

atribuut (*attribute)*- on kataloogiteenuse väikseim informatsiooni sisaldav andmekoosseis, mis sisaldab ka metaandmeid. Näiteks kasutaja nimi.

objektiklass (*objectClass*)- eriline atribuudi tüüp, mis võimaldab LDAP kirjetes määratleda, millised atribuudid on kohustuslikud ja lubatud kasutada. Näiteks telefoniraamatu objektiklassis võiks kohustuslik olla sisestada ees- ja perenimi, telefoninumber, ning lubatud atribuutidena peaks saama sisestada e-mail, aadress ja lisainfo[2].

skeem (*Schema*)- skeem on objektiklasside ja atribuutide kogum, mis defineerib kataloogiteenuse andmebaasis kasutatavad atribuudid ja objektiklassid, ning andmeväljade kontrolltingimused.

Suhteline kirje nimi (RDN- *Relative Distinguished Name*)- kirje nimi, mis seob teda vahetu kõrgema taseme kataloogiga ja on unikaalne vaid selle kataloogi piires milles ta asub[4].

eristuv/kirje nimi (DN- *Distinguished Name*)- kirje täispikk nimi, mis koosneb kirje RDNist ja temast kõrgema taseme kataloogide eristuv/kirje nimest. DN näitab täpselt kirje asukohta kataloogipuus[4]. Kirje unikaalne tunnus on tema DN, kuna koosneb RDN-ist ja kataloogi asukohast kataloogipuus- teist täpselt samasugust jada ei saa olla, kuna kataloogi ei saa tekitada teist sama RDNiga kirjet. Sama RDN teises kataloogis aga muudab ära tagumise osa DN-ist (kirje asukoha kataloogipuus).

baas nimi (*BaseDN*)- LDAP baasi nimi, mille vastu päringud saadetakse.

juurnimi (*RootDN*)- kataloogi infopuu juurkasutaja

nimi (CN- *Common Name*)- objekti/kirje nimi, väärtus võib koosneda mitmest nimest. Kui objekti/kirje näol on tegemist inimesega siis tavaliselt sisaldab täisnime[5].

organisatsiooni üksus (OU- *Organizational Unit*)- on põhimõtteliselt grupp. Nendega saab tähistada näiteks firmasiseseid osakondi (müügiosakond, raamatupidamine jne).

LDIF (LDAP *Data Interchange Format*)- LDAP-i andmete vahetuse formaat

LDAP skoop (*Scope*) – skoop milles LDAP otsingut tehakse

Indeks (*Index*) - parameeter, mille abil kirjeid indekseeritakse

POSIX (*Portable Operating System Interface*) - UNIX-i porditav operatsioonisüsteemi liides. IEEE ja ISO standarite kogumik, mis määrab ära, kuidas programmid ja operatsioonisüsteemid teineteisega ühilduvad[6].

PAM (*Pluggable authentication module*) – Linuxis ja UNIX-is kasutatav autentimismoodul

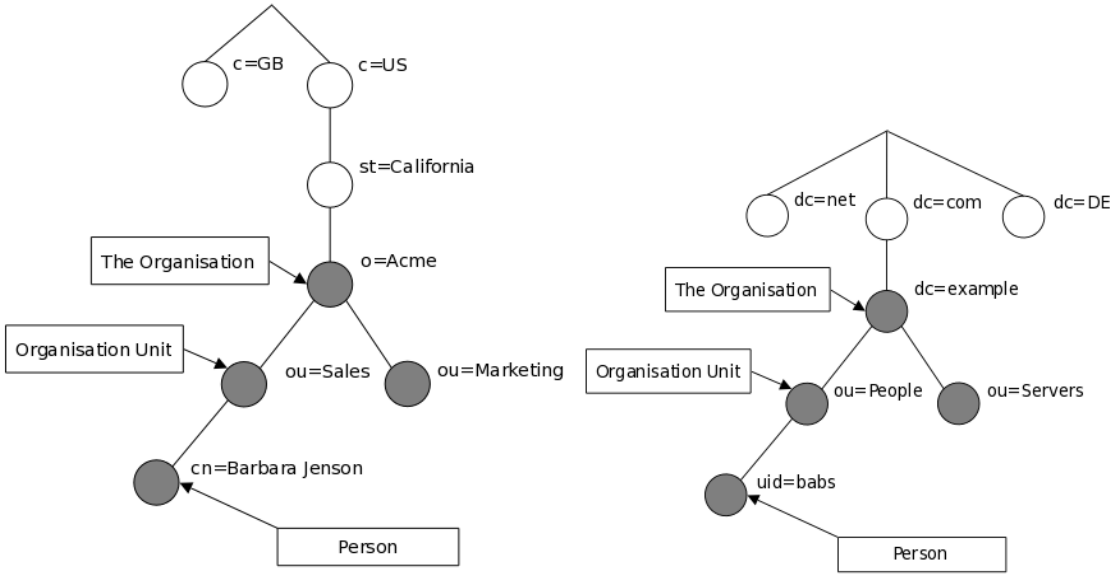
Viide (*referral*) - viitab teisele LDAP-i puule

Replikeerimine - andmete või muude ressursside koopiate hoidmine[7]

LDAP andmemudel (objektiklass, atribuut)- LDAP'i informatsiooni mudel baseerub kirjetel. Iga kirje on kogum atribuutidest, millel on **unikaalne kirje tunnus – *Distinguished Name*** (DN). Iga kirje atribuudil on tüüp ja üks või mitu väärtus. Tüübiks on tavaliselt mõni meeldejääv string, näiteks „mail“ e-maili tähenduses. Väärtuse süntaks sõltub atribuudi tüübist, mis on määratud andmemudeli skeemis (*Schema*)[2].

2 LDAP (puukujuline) struktuur

LDAP'is on kataloogi kirjed korraldatud hierarhilisel puukujulisel struktuuril. Joonisel 1 vasakpoolne kujutis on näide traditsioonilisest ja parempoolne Interneti domeeni nimedel paseeruvast struktuurist.



Joonis 1: LDAP puukujulise struktuuri kujud[2]

Traditsiooniline struktuur, mis (vasakpoolne kujutise näide) on korraldatud geograafiliste ja/või organisatsiooniliste piiridena. Puu tipus on riigi tunnus, seejärel piirkonna tunnus, millest iga järgnev sõõr täpsustab kirje asukohta asutuse struktuuris.

Alternatiivse struktuurina (parempoolne kujutise näide) on kasutusel Interneti domeeninimede põhised nimed, mis võimaldab kataloogiteenust leida DNS-i (*Domain Name Service*) kasutades

3 LDAP Serveriga ühendumine ja autentimine

LDAP serveriga ühendumise esimest operatsiooni nimetadakse *bind* (sidumine, ühendumine), mille käigus saadetakse kirje DN ja parool, mida kasutada autentimiseks. Sama operatsiooni ilma DN ja parooli saatmiseta nimetatakse *anonymous bind* (anonüümne sidumine/ühendumine). Ühendumise operatsioon ei võimalda otsingut, seetõttu peab autentimiseks kasutatav DN olema täpne eelnevalt loodud kirje DN.[8]

Järgnev õppematerjal põhineb OpenLDAP administraatori teejuhil[9].

LDAP serveriga ühendumiseks ja autentimiseks on olemas 2 meetodit: nn. „lihtne“(*simple*) ja SASL (*Simple Authentication and Security* Layer).

LDAP-i "lihtsal" meetodil on 3 töörežiimi:

- anonüümne (*anonymous*)- ligipääsureziim, mille korral ühendumisel ei sisestata kasutajanime ega parooli. Kasutaja saab ligi kirjetele, mille ligipääsusubjektiks on „anonymous“(ligipääsudest lähemalt „LDAP õiguste mudeli“ punktis);
- autentimata (*unauthenticated*)- ligipääsureziim, kus sisestatakse kasutajanimi, kuid mitte parooli ja seetõttu on kasutaja „anonymous“ õigustes. On mõned LDAP rakendused, mis ei vaja kasuaja autentimist, kuid mingite tegevuste jaoks on vajalik kasutajanimi. Vaikesätetes on see reziim väljalülitatud;
- autentitud kasutaja ja parooliga(*user/password authenticated*)- ligipääsureziim võimaldab ühenduda õiget kasutajanime ja parooli teades. Kasutaja saab ligi just tema kasutajanimele antud õigustest lähtuvalt.

Lihtsa meetodi nõrgaks küljeks on selle pealkuulatavus(näiteks paroolid võrgu pealtkuulamisel nähtava), seetõttu on soovituslik kasutada vaid tihedalt kontrollitud süsteemides või kui LDAP ühendus on turvatud võrgu tasemel (näitekst TLS- *Transport Layer Security* või Ipsec- *Internet Protocol Security*).

Simple Authentication and Security Layer (edaspidi SASL) on meetod, mis lisab autentimistoe ühenduspõhistele protokollidele sh. LDAP-ile. Protokoll sisaldab käsustikku identifitseerimaks ja autentimaks kasutajat serverile ja võimaldab turvakihis läbirääkimisi teiste protokollidega.[10]

SASL-iga koos saab kasutada mitmeid tööstuslikke autentimis mehhanismi standardeid[11]:

- GSSAPI kasutades koos Kerberos V-ga;
- DIGEST-MD5;
- PLAIN ja EXTERNAL sobivad kasutamikse ka TLS-ga (Transport Layer Security).

Täpsemalt turvalisestest ühendumisest ja autentimisest loe OpenLDAP'i administraatori teejuhist[11].

4 LDAP andmevahetuse formaat - LDIF

Järgnev osa põhineb OpenLDAP administraatori teejuhil[12].

LDAP andmevahetuse formaat (LDAP *Data Interchange Format* – LDIF) on kasutusel LDAP kirjete esitamiseks ja muudatuste salvestamiseks teksti vormi. LDAP tööriistad, nagu näiteks ldapadd ja ldapsearch, loevad ja kirjutavad LDIF kirjeid, ning ldapmodify vaid loeb LDIF kirjeid. Täpne tehniline spcifikatsioon on kirjas RFC2849 artiklis.

LDIF faili kirje põhivorm sisu:

Faili sisu	Seletus ja näide faili sisu kohta
#kommentaar	#(trellid) rea alguses võimaldab kommentaari lisada, programmid #alustatud rida ei loe!
dn: <kirjenimetus (*distinguished name)*>	dn: cn=Bjorn J Jensen,dc=example,dc=com „dn:..“ näitab kirje asukohta puus.
<atribuut|objektiklass|OID|suvand>: <väärtus>	cn: Bjorn Jensen Kirje esimene atribuut koos väärtusega. Atribuudi asemel võib olla ka objektiklass, OID või suvand (näiteks: „userCertificate;binary“).
<atribuut>: <väär tus>	Kirje järgmine atribuut/objektiklass/vms koos väärtusega. Üks tühik või tabulaatori karakterisitik rea alguses tähendab, et jätkatakse eelmist rida!
<atribuut>:: <väärtus>	Kui atribuudi väärtus koosneb mitteprinditavatest karakteristikutest (näiteks pilt) või algab tühiku, kooloni(':') või väiksem kui ('<') märgiga, siis tuleb atribuut väärtuse eraldajaks panna ühe kooloni (':') asemel kaks koolonit ('::') ja väärtus tuleb muuta *base64* kodeeringusse!!! Näiteks: jpegPhoto:: / 9j/4AAQSkZJRgABAAAAAQAB A4MChAODQ4SERATGCgaGBYWGDE jJR
	Tühirida, tähistab kirje lõppu ja teeb võimalikuks mitme kirje hoidmise ühes

	failis.
dn: <kirjenimetus (*distinguished name)*>	Algas järgmine kirje, mille esimesel real on kirje dn.

Näitefaili sisu:

Bjorn'i sissekanne

dn: cn=Bjorn J Jensen,dc=example,dc=com

cn: Bjorn J Jensen

cn: Bjorn Jensen

objectClass: person

sn: Jensen

Base64 kodeeringus JPEG foto

jpegPhoto:: /9j/4AAQSkZJRgABAAAAAQABAAD/2wBDABALD

A4MChAODQ4SERATGCgaGBYWGDEjJR0oOjM9PDkzODdASFxOQ

ERXRTc4UG1RV19iZ2hnPk1xeXBkeFxlZ2P/2wBDARESEhgVG

#eelmised 2 rida algavad tühikuga, programm loeb seetõttu viimast 3 rida ühe reana!!!

#Eelmine, tühi rida väljendab kirje lõppu!

#Algab järgmine, Jennifer'i kirje

dn: cn=Jennifer J Jensen,dc=example,dc=com

cn: Jennifer J Jensen

cn: Jennifer Jensen

objectClass: person

sn: Jensen

JPEG foto failile viitamisega

jpegPhoto:< file:///asukoht/failile.jpeg

5 Replikeerimine

Järgnev alapunkt põhineb OpenLDAP administraatori teejuhil[13].

Replikeerimine on andmete või muude ressursside koopiate hoidmine[7].

Replikeerimiseks on kolm põhjust[7]:

- Jõudluse suurendamine — puhverdamine, paralleelsete serverite kasutamine;
- Teenuse kättesaadavuse tõstmine — tõrke tekkimisel saame kasutada teist serverit;
- Tõrkekindlus — liiasusega süsteemi saame disainida tõrkekindlaks, et saavutada ka tõrgete korral mingeid garantiisid teenuse kohta

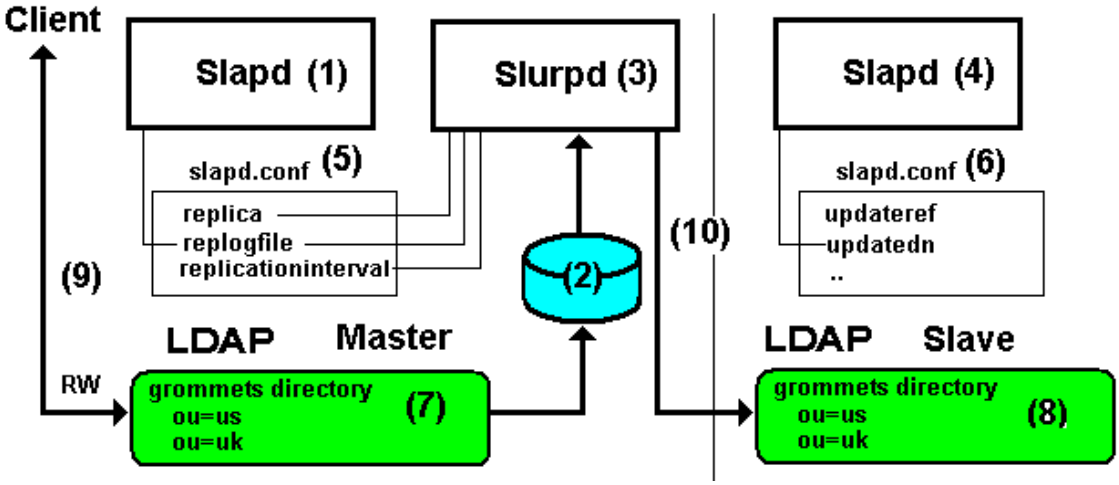
Andmete replikeerimise korraldamiseks on kataloogiteenustes kasutusel erinevaid rolle ja neist sõltuvaid tehnoloogiaid. Peamised rollid ja nende kirjeldus:

- ülem-server (*master server*) ja alam-serverite (*slave server*) rollid – on üks nn. ülem-server, kuhu saab klient teha uuendusi/muudatusi ja millest omakorda uuendatakse alam-servereid. Tegemist on jäiga struktuuriga, kus iga server on kindlaksmääratud rollis. OpenLDAP versioon 2.3 (kaasa arvatud) olid ainult eelnevalt kirjeldatud rollid. Kuna ainult ülem-serveris on võimalik muudatusi teha, on see ka kogu süsteemi nõrk koht;
- OpenLDAP võimaldab alates versioon 2.4 laiemat valikut replikeerimise

tehnoloogiaid ning lisaks eelnevatele on ka pakkuja ja tarbija (provider & consumer) rollid – pakkuja jagab kirje uuendust teisele serverile (tarbijale). Tarbija pärib pakkujalt andmeid. Sellise konfiguratsiooni puhul võib jällegi olla probleeme andmete uuendamisel: kui ühe kirje kahte atribuuti uuendatakse samal ajal erinevate väärtustega võib väärtus muutuda kasutamatuks.

5.1 OpenLDAP slurpd stiilis replikeerimine(ajalugu)

OpenLDAP slurpd stiilis replikeerimine (Sluprd Style replication) oli kasutuses kuni versioonini 2.4. Varajasemates versioonides kasutas OpenLDAP replikeerimisel slurpd'i(Standalone LDAP Update Replication Daemon) ja ajutist faili. Slurpd stiilis replikeerimine oli lükkamispõhine- ülem saatis muudatusi alluvatele.



Joonis 2: OpenLDAP slurpd stiilis replikeerimine[14]

Kui slapd (1), kus jookseb ülem (*master*) DIT (7) saab "muuda" käsu (9), uuendatakse DIT ja kirjutatakse ajatempliga transaktsiooni koopia ülema slapd.conf (5) failis defineeritud logifaili (2).[14]

Slurpd asendati mitmetel põhjustel[13]:

- see ei olnud usaldusväärne:
  - muudatused pidid toimuma kindlas järjekorras;

- võis kiiresti minna sünkroonist välja, mistõttu vajas käsitsi sekkumist;
- kui alluvserver oli tükk aega kättesaamatu võis edastatavate andmete maht kasvada määral, et see oli slurpd-i jaoks käsitsemiseks liiga suur;

- töötas ainult lükkamispõhises režiimis;
- alamserverite lisamiseks, tuli ülem peatada ning taaskäivitada;
- ülemaid tohtis olla ainult üks.

5.2 LDAP Sync Replication

LDAP sünkroniseerimise replikeerimise mootor *(LDAP Sync Replication engine) ehk syncrepl* on tarbijapoolne replikatsiooni mootor, mis võimaldab tarbija rollis oleval LDAP serveril teha varikoopia (*shadow copy*) kataloogipuu osast.

Syncrepl kasutab LDAP sisusünkroniseerimise protokolli (*LDAP Content Synchronization protocol)*, mis toetab nii tõmbamispõhist (pull-based) kui ka surumispõhist (push-based) sünkroniseerimist. Tõmbamispõhise replikatsiooni korral küsib klient pakkujalt perioodiliselt uuendusi. Surumispõhise replikatsiooni korral klient kuulab (*consumer listens*) pakkujalt reaalajas uuendusi. Kuna protokoll ei nõua ajaloo säilitamist siis pakkuja ei pea logi saadud uuenduste kohta.

Syncrepl jälgib replikeeritavate andmete olukorda küpsiste (cookies) abil. Kuna nii pakkuja kui ka tarbija hoiavad informatsiooni kehtivate andmete kohta, saab tarbija küsida pakkujalt andmeid, sooritamaks **inkrementaalset sünkroniseerimist**- küsitakse pakkujalt juurde andmed, mis on tarvilikud, et tarbijapoolne andmekoopia oleks ajakohane ja uuendatud.

5.3 Syncrepl režiimis replikeerimine

Syncrepl režiimis replikeerimine, nagu eelnevalt mainitud, on kasutuses alates versioonist 2.4. Selles on kõrvaldatud eelpool mainitud slurpd režiimis replikeerimise nõrgad kohad:

- sünkroniseerimine toimub nö iseseisvalt, ehk tarbija andmebaas võib olla

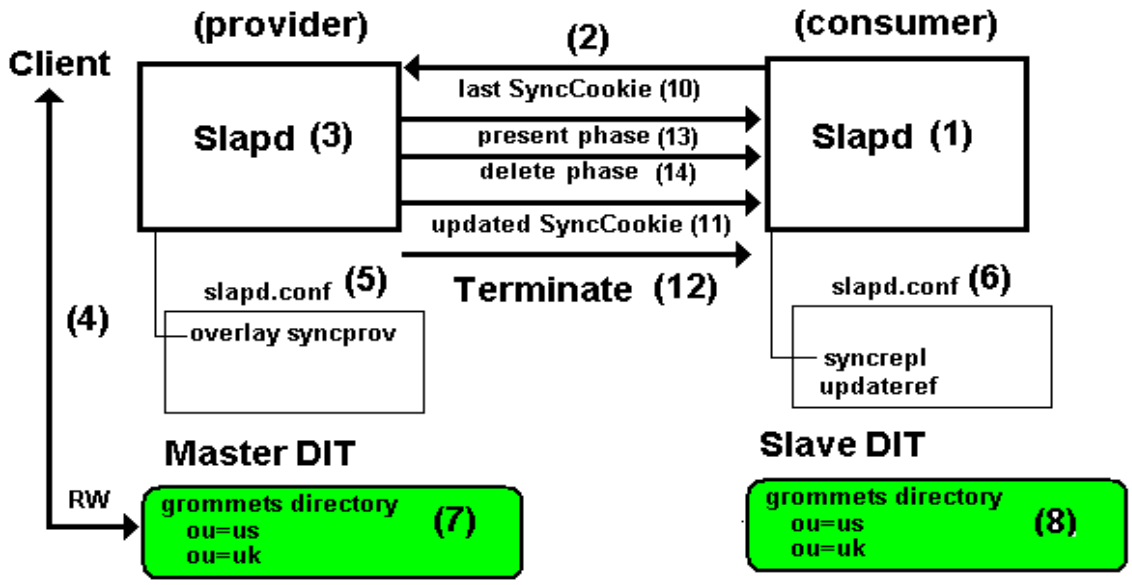
suvalises seisus ning sünkroniseerimine ja ajakohasuse säilitamine toimivad kõrvalise sekkumiseta:

  - muudatused võivad toimuda täielikult suvalises järjekorras;
  - käsitsi sekkumine pole enam tarvilik;
  - pole oluline kui kaua tarbija on kättesaamatu, sünkroniseerimine on ikkagi võimalik;

- syncrepl saab töötada nii lükkamispõhises kui surumispõhises režiimis;
- tarbijaid võib lisada suvalisel ajal, pakkujat see ei mõjuta;
- saab olla korraga rohkem kui üks pakkuja.

5.4 „Ainult värskenda“ (refreshOnly) režiim

„Ainult värskenda“ (refreshOnly) režiimis kasutatakse tõmbamispõhist sünkroniseerimist, mis puhul pole tarvis tarbija servereid jälgida ja ajaloo infot ei salvestata. Info, mis on pakkujal tarvis perioodiliste päringute töötlemiseks, on kirjas päringu enda sünkroniseerimise küpsises.



Joonis 3: Synrepl töörežiimi skeem[14]

Joonisel 3 on syncrepl töörežiimi skeem. Slapd server (1) mis tahab kataloogi infopuud (8) replikeerida (tarbija) on slapd.conf failis (6) seadistatud kasutama syncrepl-it. Selles defineeritakse ülemat kataloogi infopuu (*master* DIT) koopiat sisaldava slapd serveri (3) (pakkuja) asukoht (nimi).

Tarbija (1) algatab ühenduse (2) pakkujaga (3) - sünkroniseeritakse mõlemad kataloogi infopuud (DIT-id) ja ühendus katkestatakse (12). Tarbija (1) taastab perioodiliselt pakkujaga (3) ühenduse ja teeb uue sünkroniseerimise.

Pakkuja ise ei oma infot tarbijate kohta, seni kuni tarbija vastab pakkuja turvanõudmistele, edastatakse nõutud info. Sünkroniseerimise päring on sisuliselt laiendatud LDAP otsing, mis defineerib replikeerimise skoobi, sellest tulenevalt saab uuendada kas ainult osa või terve kataloogi infopuu (DIT).

Pakkuja ei pea säilitama tarbijapõhist olekuinformatsiooni. Selle asemel saadab pakkuja pärast igat sünkroniseerimist sünkroniseerimise küpsise (11) (SyncCookie). Küpsis sisaldab muudatuse järjekorra numbrit (change sequence number- contextCSN), mis on põhimõtteliselt viimase muudatuse sooritamise ajatempel, ehk siis sünkroniseerimise kontrollpunkt. Järgmine kord kui tarbija loob sessiooni, siis pannakse sellega kaasa ka viimane küpsis (10), mille pakkuja saatis, et oleks teada, mis ajast on uuendusi tarvis.

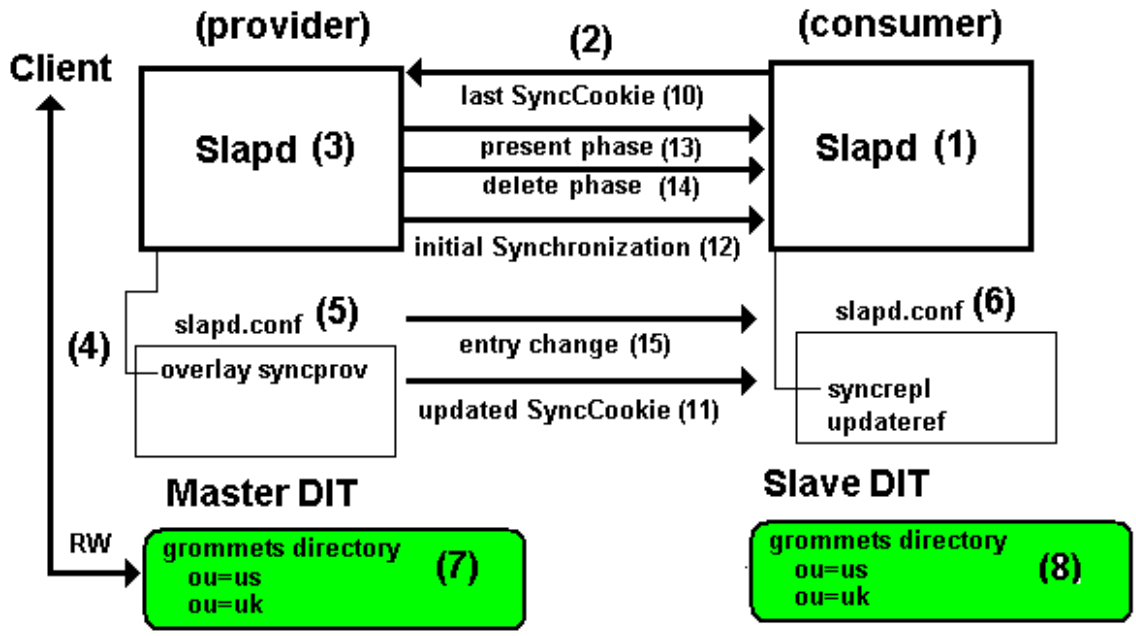
5.5 „Püsiv ühendus“ (refreshAndPersist) režiim

„Püsiv ühendus“ (refreshAndPersist) režiimis kasutatakse lükkamise põhist sünkroniseerimist. Pakkuja jälgib tarbija servereid, mis on nõudnud pidevat otsingut ja kui tema replikeeritav sisu muutub, saadab vajalikud uuendused.

Joonisel 4 on kirjeldatud püsiv ühendus režiimis replikeerimist. Slapd server (1), mis tahab replikeerida pakkuja serveri (3) kataloogi infopuud (7) on slapds.conf failis (6) seadistatud kasutama syncrepl-it. Selles defineeritakse ülemat kataloogi infopuu (*Master* DIT) koopiat sisaldava slapd serveri (3) (pakkuja) asukoht (nimi).

"Püsiv ühendus" tüüpi replikatsiooni puhul algatab ühenduse (2) pakkujaga (3) tarbija (1) - kataloogi infopuud sünkroniseeritakse (12) koheselt ja protsessi lõpus jääb ühendus püsima (sellest tulenevalt ka püsiv ühendus). Järgnevad pakkuja (3) muudatused (4)

edastatakse juba koheselt tarbijale (1).



Joonis 4: Püsiv ühendus režiimis replikeerimine[14]

Tarbija (1) avab sessiooni (2) pakkujaga (3) ja nõuab "püsiv ühendus" režiimis sünkroniseerimist. Suvaline arv tarbijaid võivad ühenduda pakkujaga ja nõuda sünkroniseerimist. Pakkuja ise ei oma infot tarbijate kohta, seni kuni tarbija vastab pakkuja turvanõudmistele, edastatakse nõutud info.

5.6 Delta-syncrepl replikeerimine (Delta-syncrepl replication)

Delta-syncrepl replikeerimine (Delta-syncrepl replication) on muudatuslogi-põhine syncrepl variant.

LDAP sync *replication* on objektipõhine replikatsiooni mehhanism, see tähendab, et kui replikeeritava objekti atribuudi väärtus pakkujal muutub, tõmbab iga tarbija ära terve muudetud objekti, ehk siis nii muudetud atribuudi väärtused kui ka muutmata atribuudi väärtused. Selle eeliseks on, et kui tehakse objektis mitu muudatust, siis ei pea säilitama muudatuste sooritamise täpset järjekorda- oluline on ainult lõpptulemus. Samas kui mitmes objektis muudetakse ainult väike väärtus, siis selle asemel, et tarbija saaks selle

muudetud väikese osa, peab ta töötlema kõik mitu objekti, mis on kindlasti ressursimahukam.

Delta-syncrepl on just ülal mainitud situatsioonide jaoks välja töötatud. Tarbija vaatab muudatuste logist järele, mis muudatusi tal tarvis on ning rakendab need oma andmebaasis. Kui replikatsioon on tühi või liiga sünkroonist väljas, kasutatakse tavapärast syncrepl replikeerimist, pärast mida lülitutakse taas delta-syncrepl režiimi.

5.7 N-Way Multi-Master replikatsiooni

N-Way Multi-Master replikatsiooni (N-*Way Multi-Master replication*) kasutatakse andmete replikeerimiseks mitme pakkuja puhul.

Head põhjused miks Multi-Master replikatsiooni kasutada:

- kui üks pakkuja tõrgub, teevad teised tööd edasi;
- väldib ühest katkemiskohta;
- pakkujad võivad geograafiliselt erinevates kohtades asuda;
- kuna ühe serveri tõrkudes töötavad teised edasi, on hea seda kasutada kõrge käideldavusega süsteemides.

Põhjused, mis arvatakse rääkivat Multi-Master replikatsiooni kasuks, kuid tegelikult seda siiski ei ole:

- see ei aita kaasa koormuse tasakaalustamisele;
- pakkujad peavad muudatusi kõikidele serveritele levitama, seega võrguliiklus ja kirjutamiskoormus jaotatakse samuti nagu ühe pakkuja korral;
- serveri kasutus ja jõudlus on heal juhul samad, mis ühe pakkujaga replikeerimisel. Ühe pakkujaga replikeerimisel saab samas indekseerimist seadistada teisiti, et optimeerida erinevaid mustrite kasutusi pakkuja ja tarbija vahel.

Multi-Master replikeerimist võiks kasutada kui on tarvis kõrgemat käideldavust ja on tarvis vältida ühest katkemiskohta.

5.8 Peegli režiimis replikeerimise

Peegli režiimis replikeerimise (MirrorMode replication) puhul seadistatakse kaks pakkujat sedasi, et nad replikeerivad teineteiselt (nagu Multi-Master replikatsiooni puhul) aga muudatused tehakse vaid ühte serverisse. Teist pakkujat kasutatakse kirjutamiseks ainult siis kui esimene ei ole kättesaadav.

5.9 Syncrepl puhverrežiimi

Syncrepl puhverrežiimi (Syncrepl Proxy Mode) võib olla tarvis kasutada kui võrgusätted takistavad tarbijal pakkujale ühenduste tegemist. Sel juhul seadistakse slapd-ldap puhver pakkuja juures, mis viitab tarbijale ning syncrepl'i mootor jookseb puhvri peal.

Kasutatud kirjandus

1: OpenLDAP Foundation, COSINE LDAP/X.500 Schema, 28.08.2011, http://tools.ietf.org/html/rfc4524
2: www.openldap.org, Administraotr Guid: Introduction, 20.05.2011, http://www.openldap.org/doc/admin24/intro.html
3: OpenLDAP Foundation, LDAP Technical Specification Road Map, 28.08.2011, http://www.ietf.org/rfc/rfc4510.txt
4: OpenLDAP Foundation, LDAP: Directory Information Models, 28.08.2011, http://www.rfc-editor.org/rfc/rfc4512.txt
5: eB2Bcom, LDAP: Schema for User Applications, 29.08.2011, http://www.rfc-editor.org/rfc/rfc4519.txt
6: www.vallaste.ee, E-teatmik, 1.03.2011, http://vallaste.ee/
7: Meelis Roos, Replikatsioon, 1.03.2011, kodu.ut.ee/~mroos/hs/2010k/repl-2x2.pdf
8: www.zyntrax.com, LDAP Glossary, 20.09.2011, http://www.zytrax.com/books/ldap/apd/index.html#
9: www.openldap.org, Authentication Methods, 20.05.2011, http://www.openldap.org/doc/admin24/guide.html#Authentication Methods
10: www.iana.com, Simple Authentication and Security Layer (SASL) Mechanisms, 20.10.2011, http://www.iana.org/assignments/sasl-mechanisms/sasl-mechanisms.xml
11: www.openldap.org, Using SASL, 20.05.2011, http://www.openldap.org/doc/admin24/guide.html#Using%20SASL
12: www.openldap.org, The LDIF text entry format, 20.05.2011, http://www.openldap.org/doc/admin24/guide.html#The%20LDIF%20text%20entry%20format
13: www.openldap.org, Replikeerimine, 20.05.2011, http://www.openldap.org/doc/admin24/guide.html#LDAP%20Sync%20Replication
14: www.zytrax.com, Replication & Referral, 20.09.2011, http://www.zytrax.com/books/ldap/ch7/