

LDAP a Kerberos

Jiří Sitera

Centrum informatizace a výpočetní techniky,
Laboratoř počítačových systémů,
Západočeská univerzita v Plzni,
e-mail: `sitera@civ.zcu.cz`

Technická zpráva CESNET č. 18/2002, 17. prosince 2002

1 Úvod

Tento dokument se zabývá některými konkrétními aspekty adresářových služeb a jejich nasazení ve výpočetním prostředí. Proto předpokládá základní znalosti z oblasti problematiky adresářových služeb a mechanismů (služeb) pro bezpečnou komunikaci, identifikaci a autentizaci. Následující text v této kapitole má sloužit jako vodítko pro případné získání informací z výše uvedených oblastí.

Cílem tohoto dokumentu je seznámit čtenáře s možnostmi využití bezpečnostní infrastruktury založené na technologii Kerberos ve světě adresářových služeb (LDAP).

1.1 LDAP

Základní informace o LDAPu formou úvodu do problematiky poskytl autor tohoto textu v samostatné publikaci[2], na kterou si dovolí na tomto místě laskavého čtenáře pouze odkázat.

1.2 Kerberos

Kerberos je autentizační služba založená na principu důvěryhodné třetí strany. Pochází z akademického prostředí, dnes je ale v podstatě standardem v oblasti autentizačních služeb distribuovaného výpočetního prostředí, na jeho implementaci jsou dokonce založena řešení firmy Microsoft počínaje Windows 2000. Další informace a odkazy byly shrnuty v článku [4].

2 Adresářové služby a bezpečnost

V dalším textu se budeme zabývat pouze adresářovými službami v „lightweight“ pojetí, tj. LDAPem. U adresářových služeb založených na plnohodnotném standardu X.500 či proprietárním řešení (například Novell NDS) je situace poněkud jiná.

2.1 LDAP a autentizace

LDAP vznikl jako maximálně jednoduchý prostředek pro přístup k datům uloženým v adresářových službách. Poté – co se stal velmi úspěšným a standardním – se k němu začala přidávat dodatečná řešení zajišťující další funkcionalitu. Díky dostatečně otevřenému návrhu LDAP¹ toto „nabalování funkcionality“ posouvá vpřed a přirozenou cestou se z něj stává plnohodnotný nástroj pro realizaci adresářových služeb.

Každá relace LDAP (klient – server) začíná operací BIND, jejíž součástí je autentizace. Takto navázané spojení pak dále poskytuje rámec pro další operace čtení či zápisu a to vše v kontextu provedené autentizace.

Základní možnosti autentizace:

- *Žádná autentizace* – anonymní přístup.
- *Jednoduchá autentizace (simple bind)* – klasická autentizace uživatele jménem a heslem. V základní podobě se heslo v rámci protokolu LDAP přenáší v otevřené formě a kontroluje proti heslu uloženému v LDAP databázi.
- *Externí autentizace* – zde se předpokládá vazba na vnější autentizační technologii, např. TLS (SSL) a v současné implementaci zejména SASL (viz níže).

Anonymní přístup se používá poměrně často, protože základní a běžné LDAP služby mají charakter veřejných služeb jen pro čtení (telefonní seznam, apod.). I zde však byla běžná potřeba autentizace uživatele a to zejména z důvodu řízení přístupu k datům (např. některá data z telefonního seznamu jsou dostupná pouze interním zaměstnancům) a možnosti aktualizace dat (např. možnost modifikovat některé informace u „svoji“ položky v telefonním seznamu).

Jednou z reálně použitelných variant autentizace k LDAPu je použití jednoduché autentizace v kombinaci se zabezpečením komunikačního kanálu. Přidáním SSL podpory (někdy označované jako server-side autentizace) k základní LDAP autentizaci heslem, zařídíme nejjednodušší a přitom poměrně kvalitní bezpečnost autentizace k LDAPu. Zde je potřeba nezapomenout na zamezení (ze strany serveru) možnosti autentizovat se bez použití SSL, což je opatření zcela nezbytné pro udržení reálné bezpečnosti.

V tomto případě je situace obvykle zcela jednoduchá a přehledná. V adresářovém stromu jsou položky odpovídající entitě uživatel, přičemž každá položka obsahuje uživatelské heslo (jako jeden atribut, v lépe či hůře zabezpečené formě, obvykle však jako výsledek jednosměrné funkce). DN této položky slouží jako identita uživatele (identifikující a autentizující se entity). Autentizace na úrovni LDAPu tedy probíhá na základě DN (a hesla), nikoli uživatelského jména samého.

Zde je potřeba si uvědomit, že LDAP vychází z norem X.500, proto i způsob popisu identity entity pomocí DN. Odtud je zřejmé i přímé propojení X.509 subjektu certifikátu a uživatelského DN LDAPu. Jedná se o tutéž věc, tentýž ideový zdroj.

2.1.1 TLS/SSL X.509 autentizace

X.509 autentizace (digitální certifikáty, či PKI) je pro LDAP přirozenou autentizační metodou a to z hlediska jeho silné ideové vazby na normy X.500, nikoli však z hlediska jeho koncepce

¹Většina diskutovaných vlastností včetně vlastních mechanismů rozšiřitelnosti je součástí normy LDAP v3 (RFC 2251). Předcházející (norma) LDAP se označuje jako LDAP v2 a z dnešního pohledu ji lze považovat za historickou. V tomto textu pod zkratkou LDAP rozumíme LDAP v3.

velmi jednoduchého přístupového protokolu. Nicméně v okamžiku, kdy se začal používat TLS protokol pro „obalení“ LDAP komunikace, bylo poměrně přímočaré přemýšlet i o využití autentizace klienta certifikátem (client-side autentizace). Tuto technologii používala řada komerčních LDAP implementací jako primární podporovaný silný autentizační mechanismus (obvykle označovaný jako „EXTERNAL“). Při správném použití v intencích všech zásad PKI poskytuje tato technologie samozřejmě kvalitní vzájemnou autentizaci klient-server (klient se prokáže serveru a také klient může ověřit autenticitu zdroje dat)². To vše však za cenu vyplývající z potřeby PKI infrastruktury.

Existuje i opačná strana mince, totiž vazba PKI infrastruktury na LDAP (využití LDAPu jako částí PKI). Vzhledem k výše uvedeným kořenům a souvislostem nepřekvapí, že je LDAP primární technologií pro publikaci některých informací v PKI prostředí, zejména pak publikaci vydaných digitálních certifikátů a seznamu jejich revokací.

2.1.2 SASL

Jak je nastíněno již v [2], SASL (*Simple Authentication and Security Layer*) je v kontextu LDAPu používán jako jednotné rozhraní k bezpečnostní infrastruktuře. Zejména v rámci projektu OpenLDAP došlo postupně ke zřetelnému vývoji tímto směrem a v současné době jsou všechny dříve (přímo) implementované mechanismy autentizace (včetně Kerbera) považovány za zastaralé a jsou podporovány pouze prostřednictvím SASL rozhraní. Zjednodušeně lze říci, že v rámci operace bind (navázání spojení) je k dispozici identita uživatele, jméno autentizačního mechanismu a *credentials*, neboli vlastní důkaz identity ve formě dat a vše ostatní je schováno pod SASL API.

Použitá konkrétní implementace SASL (zde Cyrus SASL) poskytuje některé mechanismy prokázání identity, jejichž jména jsou standardizována a dochází k předání informací o podpoře jednotlivých mechanismů mezi komunikujícími entitami (není přímo vlastností SASL, ale realizuje se standardizovanou cestou přes LDAP).

Z našeho pohledu je důležité, že SASL poskytuje několik standardně bezpečných mechanismů pro autentizaci heslem na nezabezpečeném kanálu (protokoly typu výzva/odpověď, implicitní je mechanismus CRAM-MD5) a přístup k autentizaci Kerberem prostřednictvím rozhraní GSSAPI (obecné rozhraní (API) nad autentizačními službami (obvykle) Kerbera, které mimo jiné dovoluje transparentní přístup k různým implementacím Kerbera).

2.1.3 Mapování identit

Jak již bylo výše naznačeno, v procesu identifikace a autentizace je klíčovou entitou identita uživatele. Ta je však závislá na použité autentizační technologii (mechanismu). Pro účely dalšího zpracování se tato identita mapuje na identitu primárně používanou LDAP serverem (např. uživatelské jméno na DN, či jméno Kerberos principalu na DN). Při tomto mapování lze obvykle realizovat i přemapování DN na DN např. při autentizaci digitálním certifikátem různých certifikačních autorit.

²Je vhodné poznamenat (na okraj), že LDAP v3 poskytuje podporu pro spolupráci s TLS/SSL protokolem na lepší úrovni než je pouhé „neviditelné“ obalení komunikace, což je nezbytné pro zajištění popisované funkcionality (více viz extended operace StartTLS).

2.2 LDAP jako autentizační služba

LDAP se, poměrně paradoxně, stal také sám oblíbenou autentizační službou. Vzhledem ke své jednoduchosti a snadné implementovatelnosti a zejména velmi rozšířenému standardnímu API, se dostal do mnoha SW produktů jako standardní rozhraní pro „zasunutí“ externí autentizace.

Zde se využívá jednoduché autentizace jménem a heslem – z LDAP API se zavolá operace BIND a pokud skončí úspěchem je identita uživatele potvrzena. Velmi jednoduché a pro základní využití (náhrada ověření hesla proti nějaké lokální databázi) zcela vyhovující. Existuje řada nástrojů pro jednoduchou implementaci, zejména PAM moduly a moduly do různých SW produktů. Nevýhodou je zejména funkcionality (bez SSO) a potenciální bezpečnostní problematičnost – v případě použití čistého LDAP protokolu jde heslo po síti v otevřené podobě.

Protokol LDAP nebyl k takovému účelu navržen a v této oblasti je jeho rozšiřitelnost problematická. Jako autentizační služba pro distribuované prostředí se jistě nedá příliš doporučit. Základní bezpečnostní slabiny se dají eliminovat – vlastní LDAP komunikace může být obalena SSL kanálem, tudíž je přenášené heslo chráněno. Vhodnou kombinací s PKI (tj. zabezpečení SSL kanálem je podepřeno důvěryhodným certifikátem serveru), lze dosáhnout i vzájemné autentizace, kdy je klientu prokázána důvěryhodnost LDAP serveru. Funkcionality však rozšířit směrem např. k SSO autentizační službě není možné.

Často se můžeme setkat s LDAPem jako náhradou NIS, kdy je součástí i jeho použití pro autentizaci (jako náhrada distribuce /etc/shadow, nebo dokonce položek hesla v /etc/passwd). Zde proti běžným implementacím NIS nic neztrácíme, naopak získáváme (zvláště pokud zajistíme důvěryhodnost LDAP serveru přes PKI). Obecně však lze jen doporučit LDAP používat (ostatně stejně jako NIS) pouze k distribuci informace o uživatelských kontech a dalších konfiguracích a kombinovat ho pro účely autentizace s jinou pro to určenou technologií (např. X.509 nebo s Kerberem, což je tématem tohoto textu).

Vzhledem k výše uvedenému lze snad smysluplně uvažovat o LDAP autentizaci pouze jako okrajovém a doplňkovém mechanismu pro některé nemodifikovatelné aplikace, u nichž je alespoň LDAP implementován. Pro tento účel je například k dispozici možnost nakonfigurovat OpenLDAP server jako jakousi LDAP/Kerberos bránu³.

2.3 LDAP a Kerberos

Jak bylo již výše naznačeno, vhodným přístupem nasazování LDAPu jako informační infrastruktury distribuovaného výpočetního prostředí je jeho napojení na zvolenou autentizační službu tohoto prostředí. Současné implementace LDAPu dovolují takové napojení jednoduše realizovat.

V současnosti je jednou z nejrozšířenějších a obecně přijatých autentizačních služeb rozsáhlého distribuovaného prostředí Kerberos. Integrace LDAP služeb do prostředí využívajících Kerberos je hlavním tématem tohoto textu. Zbývá jen dodat, že Kerberos lze doporučit i v případě nasazování LDAPu do prostředí bez autentizační služby, neboť takový přístup vytvoří dobré předpoklady pro další rozvoj jiných služeb a jeho prvotní nasazení je relativně

³OpenLDAP dovoluje realizovat operaci simple bind s ověřením (jménem a heslem) vůči Kerberovi. Není zde ani problém mít k dispozici pro jednu identitu (LDAP položku) jak možnost ověření přes simple bind jménem a heslem a přes SASL Kerberos lístkem, podle toho čeho je klient schopen.

jednoduché (v porovnání např. s PKI, nikoli s použitím LDAPu pro autentizaci, což však lze ve většině případů nedoporučit – viz výše).

2.3.1 LDAP jako kerberizovaná služba

V kerberizovaném prostředí je LDAP služba jako každá jiná. Uživatel má svoje pověření, na jehož základě získá pověření ke službě a tímto se prokáže LDAP serveru. Během autentizačního procesu může dojít k vzájemnému ověření identity, tj. uživatel si ověří, že LDAP služba je skutečně autentická. Následná komunikace (přenos dat) může být prostředky Kerbera také chráněna proti odposlechu.

Výše uvedené procesy jsou prováděny pro uživatele transparentně (na základě počátečního pověření uživatele), SSO (Single Sign-On) funkcionalita Kerberos autentizace může být s výhodou využita v případě, že je LDAP služba distribuována na několik serverů a k autentizaci musí ve skutečnosti pro některé LDAP dotazy docházet několikrát.

2.3.2 Kerberos identita

V rámci autentizačního procesu je ověřeno, že entita komunikující s LDAP serverem je entita označená v Kerberos světě jednoznačnou identifikací – jménem principalu. Tato identita se nějakým způsobem promítá do světa LDAPu. Konkrétně v implementaci OpenLDAP se vyjadřuje formou virtuálního DN ve tvaru:

```
uid=<principal>,cn=<realm>,cn=<mechanismus>,cn=auth
```

kde mechanismus je GSSAPI, tj. například principal sitera@ZCU.CZ se pro účely autentizace a autorizace v OpenLDAP serveru odkazuje jako:

```
uid=sitera,cn=zcu.cz,cn=gssapi,cn=auth
```

Tuto identitu (nemapované autentizační DN) lze přímo používat při definici autorizační informace (ACL), nebo ji lze přemapovat na existující DN (položky uživatele). Toto přemapování může být jednoduchou substitucí v DN (je-li cílové DN položky uživatele založeno na uživatelském jméně resp. krb principalu), nebo se může jednat o výsledek libovolné prohledávací operace (např. pokud je informace o UID či přímo Kerberos principalu uvedena u uživatelské položky v nějakém atributu).

2.4 LDAP a autorizace

Vztah LDAPu k autorizaci lze rozdělit na dvě části. První je použití LDAPu jako autorizační služby, druhou autorizace (řízení) přístupu k informacím v LDAP stromu na základě autentizace uživatele. Bohužel v obou těchto oblastech panuje značná nejasnost, neexistuje žádný standard a tím ani běžně používaná jednotná autorizační služba (ostatně odtud plyne i sama existence těchto dvou oblastí). První oblast ponechme pro naše účely stranou. Lze zde najít některé jasné příklady použití (např. modul pro Apache WEB server), ale většinou se používá spíše distribuce informací o příslušnosti uživatelů ke skupinám, žádné skutečné autorizační schéma (jako třeba v DCE CDS) se zde nevyskytuje.

V druhé skupině je situace jasnější, každá LDAP implementace má nějaké mechanismy pro řízení přístupu k jednotlivým položkám. Bohužel však každá jiná. Vždy se jedná o nějakou formu ACL (*Access Control List*) seznamů definujících přístupová práva (resp. práva

k jednotlivým operacím) podobně jako u souborového systému. Implementace se liší nejen flexibilitou ale i granularitou řízení přístupu. Komerční implementace (Active Directory, Novell e-NDS, Oracle Internet Directory, apod.) poskytují obvykle velmi jemnou granularitu a velkou flexibilitu možných definic přístupových práv.

OpenLDAP implementace se v tomto ohledu vyznačuje tím, že standardní způsob definice přístupových práv je konfigurační soubor adresářového serveru, což je v porovnání s uložením těchto ACL přímo v adresářovém serveru (jak to ostatně má většina jiných implementací) poněkud nešťastné⁴.

Pokud se však vrátíme k našemu tématu, musíme konstatovat, že při použití Kerberos autentizace je záležitost autorizace stejná jako při jiných autentizačních mechanismech. Jediný rozdíl je v odlišném jmenném prostoru identifikace entit a jejich případném mapování (viz výše)⁵.

3 LDAP a Kerberos jako infrastruktura výpočetního prostředí

Jak již bylo výše naznačeno, LDAP s Kerberos autentizací lze považovat za výhodnou a dnes již bezproblémově dostupnou a standardní kombinaci pro distribuované výpočetní prostředí. Mezi jedno z využití patří infrastruktura nahrazující NIS, resp. základní distribuovaný management stanic ve výpočetním prostředí. Kerberos zde slouží jako distribuovaná autentizační služba a LDAP poskytuje informace o uživatelských kontech, skupinách uživatelů a některých dalších konfiguračních záležitostech. Jak již bylo výše naznačeno, proti NIS nebo jeho přímé náhradě LDAPem (kdy LDAP slouží i pro autentizaci) je nasazení Kerbera velkým přínosem. Přitom vše potřebné je v dnešní době k dispozici ve standardních SW balíčcích a tento přístup je podporován hlavními distribucemi Linuxu i většinou ostatních Unixových OS.

3.1 Hlavní komponenty

- *LDAP RFC2307 služba* – adresářová služba poskytovaná libovolnou implementací LDAPu, kde jsou data o existujících uživateli uložena dle RFC2307. RFC určuje konkrétní schéma (názvy a strukturu LDAP položek), toto schéma se dá ovšem jako každé jiné rozšiřovat, neznamená to tedy zásadní omezení, pouze jakési povinné minimum potřebné k zachování kompatibility⁶.

RFC2307 pokrývá více informací, než jen informace o existujících uživatelských kontech – odpovídá funkcionalitě NIS. Za základní zde lze považovat informace o skupinách uživatelů, které je jednoznačně vhodné definovat globálně pro výpočetní prostředí a šířit je všem stanicím (viz dále).

⁴Na druhou stranu se jedná primárně pouze o způsob prezentace těchto informací a přístupu k nim. Vlastní ACL mohou být uložena odděleně (nejen formou speciálních položek a atributů adresářového stromu), ale přístup k nim by měl být možný nějakou formou přes standardní LDAP API. A to zatím není u OpenLDAPu standardně možné.

⁵Mohou nepochybně existovat i jiné drobné rozdíly technického charakteru, např. atribut určující Kerberos principal položky uživatele musí být veřejně čitelný, pokud na základě jeho hodnoty chceme provádět mapování identity apod.

⁶Ve skutečnosti většina klientů NSSwitch poskytuje možnost konfigurace do té míry, že je lze přizpůsobit používání schématu, které není v souladu s RFC2307. Pokud budujeme novou službu, je však, pokud k tomu nemáme zásadní důvod, takový postup nevhodný – lépe přizpůsobit naše schéma, než všechny klienty (pokud není důvodem to, že stavíme na nějakém „hotovém“ základě, např. Active Directory či Novell NDS[8]).

- *Kerberos služba* – standardní autentizační služba poskytovaná libovolnou implementací⁷.
- *NSS Name Service Switch funkcionalita klientů* – NSS je vlastnost knihovny `libc` umožňující vytvářet moduly, které implementují volání typu `getpwnam()`, `getpwent()` apod. Jedná se v podstatě o zobecnění navazující na různé implementace NIS, které dovoluje používat informace z lokálních souborů (např. `/etc/passwd`, NIS, DNS a dalších zdrojů na základě jednotné konfigurace a rozhraní⁸.

NSS pochází od firmy Sun a OS Solaris. V současné době je k dispozici minimálně na Linuxu (GNU C Library) Solarisu a FreeBSD. Existuje také řada proprietárních řešení s podobnou funkcionalitou, například na SGI IRIXu.

NSS.LDAP je implementace modulu NSS pro LDAP. Předpokládá RFC2307 uložení dat, ale je velmi flexibilní. Lze jej použít a nakonfigurovat prakticky pro každé LDAP schéma poskytující potřebné informace⁹. Jedná se o komerčně podporovaný open source produkt standardně dostupný ([15]) ve všech hlavních linuxových distribucích.

Standardní implementace NSS má jednu nevýhodu¹⁰: pro každé relevantní volání `libc` je znovu prováděna příslušná procedura získání informace, což v případě rozsáhlého výpočetního prostředí s mnoha klienty může znamenat poměrně značný počet LDAP dotazů. Kromě toho některé dotazy mohou být vzhledem k syntaxi příslušného `libc` rozhraní poměrně časově a přenosově náročné¹¹. Proto je vhodné použít `nscd` (*name service cache daemon*), který zařizuje konfigurovatelné kešování NSS dotazů. Tento démon je primárně určen pro kešování dotazů do lokálních souborů (pro rozsáhlé UNIXové instalace s velkým počtem uživatelů či skupin), ale vzhledem k transparentnímu napojení NSS knihovny jej lze využít i v případě LDAP implementace. Jediným rozdílem je, že běžně používá `nscd` pro invalidaci keše detekci změny lokálního souboru, což není v případě LDAP NSS účinné. Pak je funkcionalita řízena pouze nastaveným TTL časem (keše je možno invalidovat ručně a to i selektivně, teoreticky tedy i na základě nějakého oznamovacího mechanismu).

- *Klientský Kerberos SW* – podpora Kerbera je dnes již standardní u většiny Linuxových distribucí a je k dispozici pro všechny ostatní klony Unixu. Jedná se zejména o uživatelské nástroje a PAM modul. PAM subsystém zajišťuje konfigurovatelnou a flexibilní podporu různých mechanismů autentizace (a řady dalších věcí) ve všech důležitých systémových komponentách a aplikacích. Linux podporuje PAM moduly, přičemž je-

⁷Zde je myšleno zejména MIT a Heimdal, lze však úspěšně použít i např. MS W2k KDC, je-li tato technologie jádrem našeho výpočetního prostředí, což ovšem obecně nelze příliš doporučovat.

⁸Rozhraním pro aplikace jsou volání `libc`, NSS je rozhraní `libc` na vlastní implementace.

⁹Existuje vyzkoušená konfigurace/podpora např. pro Active Directory nebo Novell NDS.

¹⁰Je zde samozřejmě ještě jiná nevýhoda, totiž závislost na síťové službě pro tak základní funkci systému. Zde je potřeba říci dvě věci: Za prvé konfigurace je flexibilní a dovoluje ponechat lokální služby (soubory) jako zálohu poskytující základní potřebná data. Za druhé, uvažujeme-li využití řešení založeného na NSS LDAP jako alternativu k NIS, pak se jedná jistě o lepší alternativu i z pohledu dostupnosti, neboť LDAP lze přirozeným způsobem stavět jako zálohovanou službu a NSS LDAP knihovna takovou konfiguraci přímo podporuje.

¹¹Například pro převedení čísla skupiny na její jméno (provádí každý příkaz `ls -l`) je třeba použít volání `getgrgid()`, které vrátí strukturu obsahující všechny položky ve skupině, vzít si z této struktury název skupiny a zbytek zahodit. Má-li skupina netriviální počet členů, dochází při každém volání k přenosu jmen všech těchto členů po síti, k alokaci paměťové struktury a její následné dealokaci. Poznamenejme, že kešovací démon řeší pouze část tohoto problému.

jich konfigurací lze ovlivňovat většinu aplikací a systémových komponent ve kterých dochází k autentizaci.

V případě OS bez PAM modulů lze použít modifikovaný `login` a ostatní PAM funkcionalitou ovlivňované systémové komponenty (jsou k dispozici v Kerberos distribuci). U MS W2k je Kerberos klient součástí standardního autentizačního systému a lze jej dokumentovaným způsobem nakonfigurovat jako klienta (ne MS) Kerberos realmu.

Výše uvedené komponenty lze s úspěchem kombinovat tak, že poskytují infrastrukturu pro údržbu stanic ve výpočetním prostředí. Na rozdíl od NIS je toto řešení bezpečné a flexibilní. Doplněním o OpenAFS jako služby poskytující sdílený diskový prostor pro uživatele (včetně domovských adresářů – jejich umístění lze jednoduše distribuovat přes LDAP a není třeba je zakládat lokálně na jednotlivých stanicích) získáme komplexní prostředí, jehož velkou výhodou je, že jej lze postavit poměrně jednoduše (ve velmi základní konfiguraci) pro zajištění funkcionality několika stanic a postupně rozvíjet až po velmi rozsáhlé heterogenní a geograficky rozptýlené prostředí.

Jako další komponentu lze uvažovat využití standardního PAM modulu pro základní autorizaci (`pam_access`). Tento modul dovoluje řídit přístup ke stanici na základě členství ve skupině, přičemž toto členství je (pro tento modul transparentně) v naší konfiguraci určováno LDAP službou (LDAP distribuuje na všechny stanice informaci o všech uživatelských účtech plus skupiny kterými se omezuje přihlášení na jednotlivé stanice dle potřeb jejího správce).

4 OpenLDAP a Kerberos

Většina informací v předcházejícím textu, zejména pak všechny příklady, byly vázány na konkrétní implementaci – OpenLDAP. Bohužel většina těchto věcí (mapování autentizačních identit, autorizace, atd.) není standardizována a zaměření se na konkrétní implementaci je nezbytné. Otevřenou implementaci OpenLDAP lze však v současné době považovat za referenční implementaci a proto není příliš problematické vycházet z konkrétních údajů.

V následující kapitole se zaměříme na některé více technické aspekty nasazení OpenLDAPu jako Kerberos služby.

4.1 Potřebné komponenty a jejich konfigurace

Jak již bylo výše naznačeno podpora Kerbera v OpenLDAPu prošla několika fázemi. V současné době je Kerberos podporován jako jeden z autentizačních mechanismů SASL knihovny. Implementací SASL knihovny je Cyrus SASL. Více o potřebném SW a jeho konfiguraci viz [10].

Popis kompilace (včetně nuancí s verzemi a potřebnými záplatami) všech těchto komponent v minulosti vydal na poměrně rozsáhlou a (možná) zajímavou kapitolu. Dnes lze již téměř bez uzardění říci, že se jedná o standardně podporovanou věc – v důležitých distribucích Linuxu (zejména RedHat a Debian) jsou všechny potřebné komponenty ve standardních balících k dispozici a to implicitně s podporou všech potřebných technologií. Zajímavá je v tomto směru distribuce RedHat, která, ač poměrně konzervativní, poskytuje standardně již dva roky podporu všech výše jmenovaných technologií a téměř ve všech relevantních SW balících RedHatu je zahrnuta standardně podpora Kerbera (včetně např. CVS či Pine).

Nebo-li stačí nainstalovat¹² SASL (je implicitně, pouze chybí modul pro mechanismus GSSAPI (`cyrus-sasl-gssapi`) a pro kompilaci OpenLDAPu také `cyrus-sasl-devel`) a Kerberos (implicitně nainstalována klientská podpora, chybí development) naprosto standardními prostředky a je možno zkompileovat OpenLDAP s podporou Kerbera. Navíc je tento také k dispozici jako standardní balík (většinou však v poněkud starší verzi). Může být vhodné také zajistit podporu TLS/SSL, v kerberizovaném prostředí to však obvykle není nezbytné.

4.2 Základní konfigurace

Předpokládejme Kerberos a všechny s ním související utility správně nakonfigurovány¹³. Potom by výše uvedeným způsobem získané LDAP utility (řádkové příkazy) měly fungovat automaticky s GSSAPI autentizací (pokud ji server akceptuje – viz dále). Tato autentizace je implicitní, jednoduchou autentizací je třeba vynutit přepínačem `-x`. Příklad použití viz dále.

Na straně LDAP serveru také není v zásadě potřeba žádných zvláštních (Kerberos specifických) konfigurací. Pokud je přeložen s SASL podporou a je k dispozici GSSAPI mechanismus, je implicitně použit. Jako u každé Kerberos služby, je však potřeba uložit lokálně do souboru `krb5.keytab` klíč principálu pro LDAP službu (`ldap/<plné jméno stroje>`). V případě, že chceme provozovat LDAP server proces pod jiným uživatelem, než je `root`, je potřeba vytvořit samostatný `krb5.keytab` soubor a přidělit mu příslušná přístupová práva a nakonfigurovat LDAP server k používání tohoto souboru (`KRB5_KTNAME`).

Poznamenejme na tomto místě, že PAM modul `pam_ldap` slouží pro LDAP autentizaci (kap. 2.2). V našem pojetí je zajímavý jen `nss_ldap`, což není PAM modul, ale NSS modul a `pam_krb5` – je Kerberos PAM modul (slouží k zajištění autentizace viz popis výše, ale nemá nic společného s LDAPem).

4.3 Hlavní funkce a její ověření

Příkazem:

```
ldapsearch -x -H ldap://<jméno serveru> -b ''  
-s base 'supportedSASLMechanisms'
```

ověříme funkčnost na straně serveru. Tento dotaz nepoužívá žádnou autentizaci (pouze `simple bind` s prázdným heslem, tj. anonymní přístup) a je ekvivalentní operaci, kterou provádí klient pro zjištění serverem podporovaných SASL mechanismů. Pokud tento příkaz skončí s chybou, je velmi pravděpodobně problém mimo SASL/Kerberos. Pokud se ve výstupu tohoto programu vyskytuje řádek:

```
supportedSASLMechanisms: GSSAPI
```

je ze strany serveru k dispozici SASL autentizace s Kerberos mechanismem. Pokud tomu tak není, zkontrolujte instalaci SASL (přítomnost modulu pro GSSAPI¹⁴) a `krb5.keytab` položky.

¹²Zkušenosti vycházejí zejména z RedHatu, ale jsou snadno aplikovatelné na Debian.

¹³Balík `kerberos-workstation`.

¹⁴Jak SASL knihovna, tak jednotlivé mechanismy jsou realizovány dynamickou knihovnou – mezi časté a špatně detekovatelné problémy patří nemožnost najít tyto knihovny dynamickým loaderem při běhu LDAP serveru.

Příkazem:

```
ldapsearch -H ldap://<jméno serveru> -b '' -s base 'objectclass=*
```

ověříme funkcionální na obou stranách, tj. nejdříve klientovi. Možné potíže:

- Není k dispozici Kerberos lístek uživatele. Chybové hlášení:

```
ldap_sasl_interactive_bind_s: Local error
```

- Chybí kerberos identita služby. Klient musí úspěšně získat lístek pro službu (`ldap/<plně jméno LDAP serveru>`) – viz klist.

Úspěšný výsledek by měl znamenat dvě věci. Za prvé došlo k vzájemné autentizaci klient/server na základě důvěry v Kerbera a můžeme tedy důvěřovat v to, že náš klient skutečně komunikuje s žádaným serverem. Za druhé došlo k autentizaci klienta na základě Kerberos lístku. Tato autentizace se nevyužila k autorizaci, neboť dotazovaná data jsou čitelná všem, lze jí však využít a to na základě nastavení přístupových práv na straně serveru (identita se ověřila a přenesla).

4.4 Další konfigurace

4.4.1 Základní konfigurace klienta

Konfigurace LDAP služeb na straně klienta se provádí ve dvou základních konfiguračních souborech. Jejich jména a umístění uvedeme pro případ standardní instalace RedHat.

- Konfigurační soubor LDAP nástrojů – součást distribuce OpenLDAPu, funguje podobně jako konfigurační soubory jiných utilit, např. SSH. Existuje jeden pro celý systém, který udává implicitní informace, uživatel má možnost definovat si svoje konfigurace ve vlastním konfiguračním souboru a konečně i explicitně při volání vlastních utilit (přepínači příkazové řádky).

Příkladem je implicitní LDAP server a parametry připojení (báze, zabezpečení, umístění uživatelského X.509 certifikátu apod.).

Systémový soubor je `/etc/openldap/ldap.conf`, uživatel může umístit svoji konfiguraci do `.ldaprc` ve svém domovském adresáři.

- Konfigurační soubor NSS LDAP modulu – systémový konfigurační soubor určující chování NSS LDAP modulu. Zde je třeba nastavit parametry připojení k LDAP serveru (minimálně jeho jméno a bázi dat), odkud se bude používat RFC2307 informace. Soubor se jmenuje `/etc/ldap.conf`. Poznámka: Tento soubor slouží i pro konfiguraci PAM modulu `pam_ldap`, čili autentizace uživatelů přes LDAP.

Soubor `/etc/nsswitch.conf` určuje, pro které systémové objekty se použije NSS LDAP a v jakém pořadí vzhledem k jiným službám (relevantní jsou v našem případě lokální soubory – klíčové slovo `files`). Více viz příslušná dokumentace¹⁵.

¹⁵Problematika přesahuje rámec tohoto dokumentu, proto je potřeba získat informace jiným způsobem. Uvedme snad jen, že správnou funkci NSS LDAP ověříme utilitou `getent`.

4.4.2 Replikace s použitím Kerberos autentizace

Standardní replikace v OpenLDAPu (single-master) je realizována následovně: Master server proces (`slapd`) je nakonfigurován tak, že vytváří tzv. replikační log, obsahující standardní LDIF příkazy popisující změny, které je třeba provést na replikách. Replikační proces (`slurpd`) běží na stroji s master replikou, čte tento soubor a provádí změny na slave replikách standardním LDAP protokolem. Slave replika je tedy běžný LDAP server proces, pouze jeho konfigurace dovoluje změny výhradně identitě master replikačního procesu. Je zde možné/vhodné také nakonfigurovat tzv. *update referral*, tj. odkaz na master repliku, který bude vrácen klientovi při pokusu o modifikaci slave repliky. Z tohoto popisu (více o replikaci viz dokumentace, např. [10]) je patrné, že autentizaci a zabezpečení komunikace při replikaci je třeba realizovat mezi `slurpd` a slave `slapd`.

Pro zabezpečení replikace přes Kerberos je tedy nutné vytvořit identitu replikačního procesu, zajistit běh tohoto procesu pod vytvořenou identitou a správně nakonfigurovat obě komunikující komponenty.

Uvedme nyní stručně jednotlivé kroky:

- *Vytvoření replikační identity* – vytvoříme Kerberos identitu (principal), kterou pojmenujeme např. `ldapmgr/ldap-master.zcu.cz@ZCU.CZ`, kde `ldap.zcu.cz` je jméno master ldap serveru a `ZCU.CZ` realm. Její klíč uložíme na master server (formou souboru `keytab`)¹⁶ a zajistíme běh replikačního procesu `slurpd` pod touto identitou. Nesmíme zapomenout na obnovování platnosti identity. K tomu stačí spustit replikační proces se speciální Kerberos keší a jednoduchým pravidelně spouštěným skriptem zajistit (např. pomocí `kinit -k`) obnovení Kerberos lístku.
- *Konfigurace master `slapd` a `slurpd`* – např. (příslušná část `slapd.conf`, konfiguračního souboru, který používá jak master `slapd`, tak `slurpd`):

```
replica host=ldap2.zcu.cz:389
        bindmethod=sasl saslmech=GSSAPI
        authcId=ldapmgr/ldap-master.zcu.cz@ZCU.CZ
```

určuje, že bude vytvářen replikační log pro stroj `ldap2.zcu.cz` a `slurpd` jej bude na tento stroj protokolem LDAP s autentizací SASL a mechanismem GSSAPI pod replikační Krb identitou přenášet.

- *Konfigurace slave `slapd`* – konfigurace ACL tak, aby zapisovat mohla pouze replikační identita.

```
access to *
        by dn="uid=ldapmgr/ldap-master.zcu.cz" write
        by * read
```

¹⁶Je třeba rozhodnout pod jakým uživatelem replikační proces poběží a vytvořit samostatný soubor s klíčem, je-li potřeba.

V tomto příkladu mohou číst všichni všechno, ale obecně je třeba ještě přidat odpovídající ACL pro čtení. Z hlediska replikace jsou důležitá pouze práva pro zápis.

Konfigurace update refferalu na master repliku se provede následovně:

```
updatedn "ldapmgr/ldap-master.zcu.cz@ZCU.CZ"  
updateref ldap://ldap-master.zcu.cz/
```

kde `updatedn` určuje identitu, které není odkaz vrácen.

4.5 Nemusí to být tak složité (začátky)

Výše uvedená instalace a konfigurace (převážně) na straně klienta může být v dnešní době relativně jednoduchá a zvládnutelná bez hlubších znalostí. O tom, že je v moderních Linuxových distribucích snadné (začít) pracovat s výše (převážně) glorifikovanými technologiemi, by měl laskavého čtenáře přesvědčit projekt Boxed Penguin ([14]), který si klade za cíl dát k dispozici „instantní“ infrastrukturu odpovídající té, jež byla popsána v kap. 3 a to vše prostřednictvím sady utilit a (auto)konfiguračních nástrojů pro Debian. Tento projekt sice patrně ustrnul kdesi na své cestě vpřed, ale jeho smysl spočívá v tom, že ukazuje cestu. Cestu k integrovanému a s relativně velmi malým úsilím nastartovatelnému řešení pro malé až střední distribuované výpočetní prostředí založené na Linuxu (v tomto případě Debianu) a technologiích jako je Kerberos, LDAP a AFS.

4.6 Vývojové prostředí pro Perl s podporou Kerbera

Knihovnu pro přístup k adresářovým službám z Perlu `NET::LDAP` (viz také [1]) je možné zcela transparentně použít i s Kerberos autentizací (přes SASL a GSSAPI). Je k tomu třeba použít modul `Authen::SASL::GSSAPI`. Práce s daty v LDAPu se neliší od běžného stavu, rozdíl je pouze v připojování k LDAP serveru (viz následující příklad).

```
use Net::LDAP;                # LDAP module  
use Authen::SASL;            # Cyrus SASL module interface for GSSAPI  
  
#define $ldapHost and $ldapPort here  
  
#create LDAP connection  
$conn=new Net::LDAP($ldapHost, port=>$ldapPort);  
  
# GSSAPI user auth (actual principal <-> ldap service principal)  
# also does mutual authenticity check  
$sasl = Authen::SASL->new('GSSAPI',  
                           fqdn =>$ldapHost,  
                           service =>'ldap',  
                           user =>'');  
  
# bind using sasl credentials  
$conn->bind(DN=>'', sasl => $sasl, version=>3) ||  
die "Could't connect to $ldapHost ($mesg->code, $mesg->error)"
```

```

# do any LDAP operation, for example search

$msgg=$conn->search(
    base=>$ldapBase,
    scope=>'sub',
    filter=>$ldapFilter
);

# process result(s)

#close the connection
$conn->unbind();

```

5 Některá praktická využití bezpečného LDAPu

5.1 Informační infrastruktura *META Centra*

Využití LDAPu v rámci projektu *META Centrum* je předmětem samostatného dokumentu [1]. LDAP byl v rámci *META Centra* používán ještě v době, kdy jeho napojení na Kerberos autentizaci nebylo bezproblémové (Kerberos je základní autentizační technologií *META Centra* již od jeho vzniku) a LDAP zde sloužil pouze pro anonymní přístup k datům (pouze pro čtení). Proto se z počátku používal pouze zcela základní LDAP protokol bez zabezpečení. V současné době je Kerberos pro LDAP službu plně využíván a vzhledem k výše uvedeným skutečnostem (LDAP je rozhraní pro přístup k datům převážně pouze ke čtení) je důležitá zejména jeho funkcionality zajišťující vzájemné ověření klient/server.

5.1.1 Ověření autenticity zdroje dat

Některé základní systémové informace, jako je třeba seznam uživatelů, jsou sice veřejné, ale pro některé komponenty je zcela zásadní ověřit si autenticitu těchto dat (představme si například komponentu, která vytváří nová uživatelská konta). Zde lze s výhodou využít vzájemnou autentizaci klient/server pomocí Kerbera. Ačkoli server nepotřebuje znát identitu klienta, neboť informace jsou přístupné všem, je v rámci autentizace ověřena i identita serveru, což je v našem případě zásadní.

5.1.2 Řízení přístupu k datům

Některé lokální systémové informace (například Unix uid uživatelů, které je lokální vůči buňce *META Centra*) jsou uloženy primárně přímo v LDAPu, přičemž jejich změny provádí systémové skripty (přidělují lokální uid novým uživatelům). Tyto skripty běží pod vlastní Kerberos identitou a autentizují se pomocí ní k LDAP serveru.

5.2 Projekt Pleiades – informační infrastruktura ORIONu

ORION je distribuované výpočetní prostředí založené na Kerberos autentizaci, AFS souborovém systému a managementu a konfiguračních nástrojích vycházejících z projektu MIT Athena. Jmennou službou je zde tedy Hesiod, řada dalších informací (např. `/etc/passwd`)

je distribuována vlastními nástroji formou souborů. Mezi aktivity směřující k využití LDAPu patří zejména jeho použití pro získávání konfiguračních informací strojů ([5]) a projekt OPERONORION, který je postaven nad výše popsanou infrastrukturou výpočetního prostředí založenou na LDAPu (rfc2307 a NSSwitch), Kerberem a AFS.

5.3 Globus MDS, Datagrid

Projekt Globus ([11]) je od svého začátku směřován k dosažení co nejjednoduššího připojení zdrojů do celku, pro propojení jednotlivých uzlů byla zvolena autentizační technologie založená na principech PKI. Informační infrastruktura (MDS – *Monitoring and Discovery Service*)[12] je založena na LDAPu, vznikla nejdříve jako zcela anonymní služba bez autentizace. Dnes však již MDS používá standardní autentizační knihovnu Globusu – tzv. GSI (*Grid Security Infrastructure*)[13], která funkčně odpovídá PKI a slouží jak k autentizaci koncových klientů, tak k vzájemné autentizaci jednotlivých komponent MDS¹⁷. Kerberos se používá pouze jako interní autentizační služba jednotlivých uzlů. Projekt Datagrid používá Globus MDS.

¹⁷GSI je kupodivu použito jako SASL GSSAPI mechanismus OpenLDAPu, nad kterým je současná MDS vystavěna. S Kerberem to však nemá nic společného, GSI knihovny pro GSSAPI realizují toto API nad X.509 certifikáty.

Odkazy a literatura

- [1] Jiří Sitera, *Využití adresářových služeb (LDAP) v projektu META Centrum*, jako technická zpráva TEN 1/2001
<http://www.cesnet.cz/doc/techzpravy/2001/02/>
<http://home.zcu.cz/projekty/lps/ldap/projekt/www/papers/MetaLDAP.ps>
- [2] Jiří Sitera, *Adresářové služby – úvod do problematiky*, jako technická zpráva TEN 4/2000
<http://www.cesnet.cz/doc/techzpravy/2000-4/>
- [3] Jiří Sitera, *Adresářové služby jako informační infrastruktura distribuovaného výpočetního prostředí*, sborník konference EurOpen.CZ, listopad 1999, ISBN 80-902715-0-2.
- [4] Luboš Kejzlar, Jiří Sitera, *Prokazování identity v rozsáhlém výpočetním prostředí*, sborník XVIII. konference EurOpen.CZ Dolní Malá Úpa, 2001, ISBN 80-902715-8-8.
- [5] Jiří Sitera, *Using LDAP as service for getting workstation configuration information in distributed computing environment*, University of west Bohemia proceedings 1999, ISBN 80-7082-617-7.
- [6] Projekt Pleiades – domovská stránka,
<http://home.zcu.cz/projekty/lps/ldap>
- [7] Jiří Sitera, *Skriptovací jazyky a jejich využití pro přístup k adresářovým službám protokolem LDAP*,
<http://home.zcu.cz/projekty/lps/ldap/projekt/www/papers/skriptLDAP.ps>
- [8] Norbert Klasen, *Directory Services for Linux in comparison with Novell NDS and Microsoft Active Directory*, Diploma thesis, Aachen, Germany, August 2001.
- [9] The OpenLDAP Project, *Open source suite of LDAP applications and development tools*,
<http://www.openldap.org>
- [10] The OpenLDAP Project, *OpenLDAP Administrator's Guide*,
<http://www.openldap.org/doc/>
- [11] The Globus Project, <http://www.globus.org/>
- [12] The Globus Project, *Monitoring and Discovery Service*, <http://www.globus.org/mds/>
- [13] The Globus Project, *Grid Security Infrastructure*, <http://www.globus.org/security/>
- [14] *The Boxed Penguin – an Instant Infrastructure Prototype*,
<http://www.boxedpenguin.com>
- [15] Padl Software, *nss_ldap module*, http://www.padl.com/OSS/nss_ldap.html