HRVATSKA AKADEMSKA I ISTRAŽIVAČKA MREŽA
CROATIAN ACADEMIC AND RESEARCH NETWORK

Implementacija Kerberos protokola u Linux okruženjima

CCERT-PUBDOC-2005-10-136

CARNet CERT u suradnji s LS&S

Sigurnosni problemi u računalnim programima i operativnim sustavima područje je na kojem CARNet CERT kontinuirano radi.

Rezultat toga rada ovaj je dokument, koji je nastao suradnjom CARNet CERT-a i LS&S-a, a za koji se nadamo se da će Vam koristiti u poboljšanju sigurnosti Vašeg sustava.

CARNet CERT, www.cert.hr - nacionalno središte za **sigurnost računalnih mreža** i sustava.

LS&S, www.lss.hr - laboratorij za sustave i signale pri Zavodu za elektroničke sustave i obradbu informacija Fakulteta elektrotehnike i računarstva Sveučilišta u Zagrebu.

Sadržaj

1. Uvod

Proces autentikacije, odnosno provjere korisničkog identiteta, iznimno je važan element informacijske sigurnosti. Budući da predstavlja prvi korak prijave korisnika u sustav, sigurnosni zahtjevi koji se pred njega postavljaju prilično su visoki. Također, osim visoke razine sigurnosti, da bi bio upotrebljiv u praksi, proces autentikacije mora zadovoljavati i brojne druge zahtjeve (npr. praktičnost, financijska isplativost, jednostavnost održavanja i upravljanja i sl.). Kao primjer mogu se navesti biometrijski uređaji koji, usprkos visokoj razini sigurnosti koju nude, još uvijek nisu šire prihvaćeni kao mehanizam autentikacije.

Kerberos protokol jedan je od najpoznatijih protokola za autentikaciju korisnika. Protokol se odlikuje brojnim funkcionalnostima i prednostima, a jedna od najznačajnijih je svakako *Single Sign On* (SSO) funkcionalnost, koja korisnicima omogućuje da se samo jednom prijave u sustav i da nakon toga, u skladu sa svojim ovlastima, imaju pristup svim resursima u sustavu. Na taj način, korištenjem Kerberos protokola, uklanja se potreba za upravljanjem velikim brojem korisničkih računa i zaporki, a također se smanjuje i vrijeme potrebno za pristup pojedinim servisima. Dodatna prednost sa sigurnosne strane je ta što Kerberos protokol korisničke zaporke nikad ne šalje mrežom u čistom tekstualnom obliku, što ga čini otpornim na napade praćenjem i analizom mrežnog prometa (engl. *sniffing*).

Budući da novije inačice Windows operacijskih sustava Kerberos protokol koriste kao primarni protokol za autentikaciju korisnika, Kerberos se često pogrešno smatra Microsoftovim proizvodom. Kao što je detaljnije opisano u poglavlju 2.1, Kerberos protokol razvijen je još davne 1980 godine na Massachusetts Institute for Technology (MIT) institutu u sklopu poznatog Athena istraživačkog projekta. Iako je najveću popularnost protokol stekao nakon implementacije u Windows operacijskim sustavima, postoje implementacije Kerberos protokola i za druge operacijske sustave.

Dokument opisuje osnovne karakteristike i način rada Kerberos protokola, a detaljno je opisana i konkretna implementacija sustava u Linux okruženjima. Kao testna platforma korišten je Linux Debian operacijski sustav, iako se slični koncepti mogu primijeniti i na većinu drugih Linux/Unix operacijskih sustava.

2. Kerberos protokol

2.1. Povijest Kerberos protokola

Kerberos autentikacijski protokol razvijen je osamdesetih godina prošlog stoljeća na MIT (**M**assachusetts **I**nstitute of **T**echnology) institutu u okviru projekta pod nazivom Athena. Osim MIT-a, na projektu su sudjelovali i brojni stručnjaci iz DEC-a i IBM-a, koji su ujedno bili i glavni sponzori. Iako su primarni ciljevi ovog projekta bili daleko opsežniji i kompleksniji od samog Kerberos protokola, danas se slobodno može reći kako je Kerberos zasigurno jedan od značajnijih njegovih rezultata u tehničkom smislu. Osim Kerberos protokola u sklopu Athena projekta razvijene su i druge danas poznate tehnologije kao što je npr. X Window sustav za Linux/Unix operacijske sustave, a osim tehničkih dostignuća Athena projekt rezultirao je i velikim uspjehom u pogledu upotrebe računalnih sustava u akademskim zajednicama.

Prve tri inačice protokola bile su razvojne i koristile su se isključivo unutar MIT instituta. Sa inačicom v4 protokol je 24. siječnja 1989. godine dan u javnu uporabu, a uskoro se pojavila i inačica v5 s kojom su bili uklonjeni brojni nedostatci primijećeni kod ranijih inačica. Obzirom na brojne kvalitete i prednosti Kerberos protokola, isti je bio brzo prihvaćen u svijetu informacijskih tehnologija i danas je jedan od najpoznatijih protokola za mrežnu autentikaciju. Dodatnu popularnost Kerberos je stekao nakon što ga je Microsoft odlučio implementirati kao primarni autentikacijski mehanizam kod novijih inačica Windows operacijskih sustava, iako su poznati i brojni drugi proizvodi koji su djelomično ili u potpunosti bazirani na konceptima Kerberos protokola (SESAME, DCE i sl.).

2.2. Osnovne karakteristike

Kerberos se definira kao siguran, *single-sign-on* autentikacijski protokol baziran na centralnom autentikacijskom entitetu kojem svi drugi entiteti u informacijskom sustavu u potpunosti vjeruju (engl. *trusted entity*). Centralni autentikacijski entitet u Kerberos sustavu naziva se KDC poslužitelj (eng. ***K**ey **D**istribution **C**enter*), i predstavlja centralni repozitorij u kojem su pohranjeni autentikacijski parametri svih entiteta u Kerberos sustavu. Ulogu KDC poslužitelja može obavljati i više poslužitelja, kako bi se osigurala funkcionalnost sustava u slučaju nedostupnosti jednog od njih.

Kerberos protokol naziva se sigurnim zato jer zaporke računalnom mrežom nikad ne šalje u čistom tekstualnom obliku, već u tu svrhu koristi specijalne kriptirane poruke ograničenog perioda valjanosti– *tickets* (valjanost poruka tipično je od 8-24h). Ove poruke generira KDC poslužitelj na zahtjev korisnika koji želi pristupiti određenom resursu u Kerberos sustavu. Ovakav način rada Kerberos protokol čini idealnim autentikacijskim mehanizmom za računalne sustave u kojima se ne može vjerovati svim korisnicima. Budući da se autentikacijski parametri mrežom šalju kriptirani, protokol je zaštićen od napada praćenjem i analizom mrežnog prometa (eng. *sniffing*).

Singl-sign-on funkcionalnost podrazumijeva proces u kojem se korisnik samo jednom prijavljuje u sustav, nakon čega mu se omogućuje pristup svim mrežnim servisima koji podržavaju Kerberos protokol. Nakon inicijalne prijave u sustav, pristup svim mrežnim resursima za korisnika je u potpunosti transparentan, što znatno olakšava rad u distribuiranim mrežnim okruženjima. Također, iako je Kerberos prvenstveno autentikacijski protokol za autentikaciju, njegovom implementacijom znatno se olakšavaju i ostala dva procesa koja zajedno čine poznati tzv. "AAA (*Authentication*, *Authorization*, *Auditing*) koncept. Iz svih navedenih karakteristika može se zaključiti da Kerberos protokol osim svojih sigurnosnih svojstava donosi i druge pogodnosti, što je jedan od osnovnih razloga njegove iznimne popularnosti.

U nastavku poglavlja ukratko je opisan način rada Kerberos protokola. Opisane su osnovne komponente sustava, njihov značaj te postupak autentikacije korisnika.

2.3. Osnovni pojmovi

Uz Kerberos protokol vezani su brojni pojmovi i koncepti koji su specifični za ovaj sustav i ključni za njegovo razumijevanje. U nastavku će ukratko biti pojašnjeni neki od važnijih pojmova kako bi se olakšalo razumijevanje konkretne implementacije Kerberos sustava.

2.3.1. Kerberos realm i principali

Svaki entitet Kerberos sustava, bez obzira da li se radi o korisniku, računalu, mrežnom servisu, poslužitelju ili nečem trećem, opisan je sa odgovarajućim imenom u bazi KDC poslužitelja, koji se naziva **principal**. Svaki principal jedinstveno opisuje entitet u Kerberos sustavu i ima odgovarajuću strukturu definiranu specifikacijom protokola. Također, svaki principal u Kerberos sustavu posjeduje i odgovarajući tajni ključ koji je poznat samo KDC poslužitelju i entitetu o čijem se ključu radi. Ovaj tajni ključ koristi se za enkripciju poruka koje se razmjenjuju u postupku autentikacije.
Općenita struktura Kerberos principala je sljedeća:

```
identity/instance@realm
```

Značaj pojedinih elemenata je:

- ***identity*** – polje koje opisuje ime Kerberos entiteta (korisničko ime, mrežni servis, računalo i sl.). Ovo polje je obavezno za svaki principal objekt.
- ***instance*** – polje **instance** pobliže opisuje Kerberos entitet i može se shvatiti kao opis grupe kojoj entitet pripada. Za korisničke račune ovo polje može označavati grupu kojoj korisnik pripada, ili namjenu korisničkog računa kako bi se olakšala administracija sustava (slično kao i koncept grupa na operacijskim sustavima). Kod principala koji opisuju mrežne servise, **instance** dio sadrži ime računala na kojem je servis pokrenut. Na taj način razlikuju se identični servisi pokrenuti na različitim računalima. Ovo polje nije obavezno.
- ***realm*** – svaka zasebna instalacija Kerberos sustava definira jedinstveni **realm** koji opisuje sustav i koji se razlikuje od bilo kojeg drugog Kerberos okruženja. Prema konvenciji, Kerberos **realm** najčešće odgovara DNS imenu domene organizacije, s jedinom razlikom što se označava velikim slovima (npr. Kerberos **realm** za domenu example.hr označavao bi se kao EXAMPLE.HR)

Slijedi nekoliko konkretnih primjera Kerberos principala kako bi se dodatno pojasnili opisani koncepti:

- **`sjusic@EXAMPLE.HR`**. – korisnik `sjusic` u Kerberos realmu `EXAMPLE.HR`.
- **`sjusic/admin@EXAMPLE.HR`** – korisnik `sjusic` u realmu `EXAMPLE.HR` koji spada u administratorsku grupu.
- **`ssh/ssh.example.hr@EXAMPLE.HR`** – principal koji opisuje `ssh` servis na računalu `ssh.example.hr` u Kerberos realmu `EXAMPLE.HR`.
- **`host/test.example.hr@EXAMPLE.HR`** – principal koji opisuje računalo `test.example.hr` u Kerberos realmu `EXAMPLE.HR`.

Ovakav način kreiranja principala u Kerberos sustavu omogućuje jednoznačno definiranje i korištenje svih entiteta u Kerberos okruženju.

2.3.2. Key Distribution Center (KDC)

Kao što je već ranije spomenuto, KDC poslužitelj predstavlja jezgru Kerberos sustava i njegova dostupnost neophodna je za funkcionalnost cijelog sustava. Iako se KDC poslužitelj sastoji od tri različite komponente, sve su one najčešće integrirane u jedan program koji je pokrenut na odgovarajućem mrežnom poslužitelju. Tri komponente koje čine KDC poslužitelj su:

- Baza sa svim principalima unutar definiranog Kerberos realma s pripadajućim tajnim ključevima. Način na koji je implementirana baza sa ovim podacima ovisi o implementaciji sustava. Kod Microsoft sustava ovi se podaci čuvaju unutar Active Directory imenika, dok se kod Linux implementacija u tu svrhu koriste specijalizirane LDAP (eng. *Lightweight Directory Access Protocol*) baze.
- Authentication Server (AS).
- Ticket Granting Server (TGS).

Budući da KDC poslužitelj sadrži tajne ključeve svih korisnika sustava, posebnu je pažnju potrebno posvetiti njegovoj zaštiti. Kompromitiranje KDC poslužitelja u potpunosti ugrožava sigurnost cijelog Kerberos sustava, što predstavlja iznimno visok sigurnosni rizik. Budući da jedan Kerberos realm najčešće sadrži više KDC poslužitelja koji osiguravaju funkcionalnost sustava u slučaja pada jednog od njih, visoku razinu sigurnosti potrebo je održavati na svim poslužiteljima sustava. Također, vrlo je važna i međusobna sinkronizacija KDC poslužitelja kako bi se na svakom od njih nalazili najsvježiji podaci.

U nastavku poglavlja ukratko je opisana funkcija ranije spomenutih *Authentication Server* (AS) i *Ticket Granting Server* (TGS) poslužitelja.

- **Authentication Server**

Uloga autentikacijskog poslužitelja je da klijentima koji se žele prijaviti u Kerberos sustav izda odgovarajuću TGT (*Ticket Granting Ticket*) kartu. TGT karta generira se prilikom inicijalne prijave u sustav, nakon čega je klijenti lokalno pohranjuju i dalje koriste za pristup svim ostalim mrežnim resursima bez potrebe za ponovnim unosom korisničke zaporke.
Ukratko, postupak je sljedeći. Prilikom inicijalne prijave korisnika u sustav, AS poslužitelj generira odgovarajuću TGT kartu te je kriptira tajnim ključem (zaporkom), koji je poznat samo KDC poslužitelju i krajnjem korisniku. Ukoliko proces autentikacije uspješno dekriptira dobivenu kartu zaporkom koju je korisnik unio, proces autentikacije je uspješan i dobivena karta lokalno se pohranjuje kako bi se kasnije mogla iskoristiti za pristup ostalim mrežnim resursima. Više riječi o TGT karti i njezinoj ulozi u Kerberos sustavu biti će u nastavku dokumenta, zajedno s odgovarajućim praktičnim primjerima.

- **Ticket Granting Server**

Za razliku od AS poslužitelja, koji klijentima generira inicijalnu TGT kartu prilikom prijave u sustav, TGS poslužitelj zadužen je za izdavanje dodatnih karata za pristup ostalim mrežnim resursima. Za dobivanje odgovarajuće karte za pristup traženom resursu, klijent TGS poslužitelju prosljeđuje TGT kartu dobivenu od AS poslužitelja te ime resursa kojem želi pristupiti. Nakon što TGS provjeri da li je dobivena TGT karta valjana, klijentu se prosljeđuje TGS karta kojom je moguće ostvariti pristup zatraženom mrežnom resursu. Također, u nastavku dokumenta biti će detaljnije opisan koncept različitih Kerberos karata i načina njihove upotrebe.

2.3.3. Kerberos karte

Koncept karata na kojem se bazira Kerberos sustav jedinstven je za računalne sustave. No, usprkos ovom jedinstvenom konceptu, ideja karata vrlo je jednostavna i slična konceptu koji se često primjenjuje u svakodnevnom životu. Npr. Kerberos karta može se usporediti s vozačkom dozvolom. Centralni autoritet (MUP) izdaje odgovarajuću kartu (vozačku dozvolu) koja sadrži osnovne podatke o korisniku i samoj dozvoli (ime i prezime, vrijeme trajanja, identifikator i sl.). Kod Kerberos sustava centralni autoritet predstavlja KDC poslužitelj, dok je vozačka dozvola identična Kerberos karti koju KDC izdaje. Svaka karta je jedinstvena na svoj način i korisniku omogućuje pristup ostalim mrežnim resursima, kao što vozačka dozvola omogućuje upravljanje motornim vozilima. Ukoliko je karta valjana, korisniku se omogućuje pristup zatraženom resursu i obrnuto. Podaci koje sadrži svaka Kerberos karta navedeni su u nastavku:

- ime principala koji zahtjeva pristup,
- ime principala kojem se zahtjeva pristup,
- vremenska oznaka (engl. *timestamp*),
- vrijeme trajanja karte (engl. *lifetime*),
- lista IP adresa s kojih je moguća upotreba karte,
- tajni sjednički ključ za komunikaciju sa zatraženim resursom.

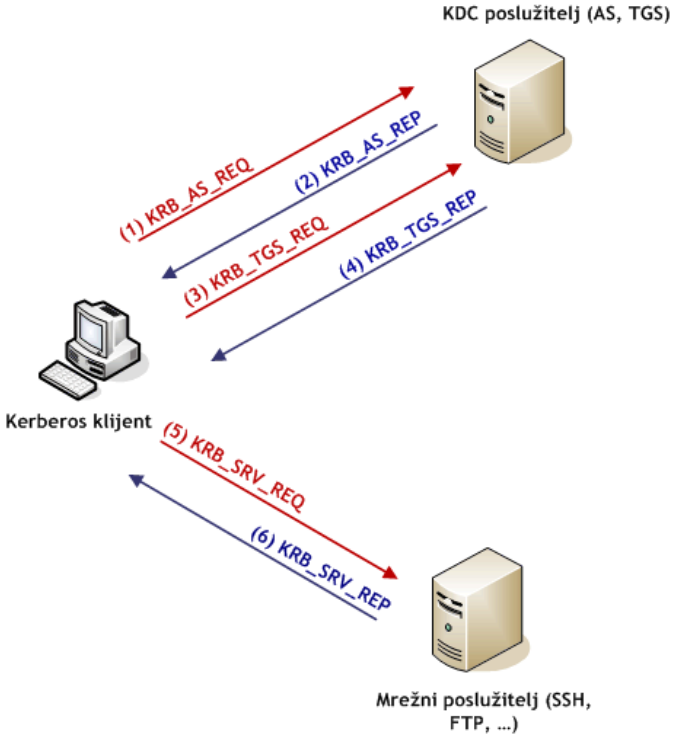
Kerberos karte imaju dvije osnovne funkcije, da se potvrdi identitet entiteta koji zahtjeva pristup određenom resursu, te da se uspostavi tajni sjednički ključ koji će se koristiti za enkripciju podataka tijekom komunikacije. Ovi pojmovi i koncepti će biti detaljnije opisani u nastavku dokumenta te pojašnjeni na konkretnim primjerima Kerberos autentikacije.
Vremenska oznaka i vrijeme trajanja karte također su bitni parametri Kerberos komunikacije. Njihovom upotrebom sustav se štiti od tzv. *replay* napada u kojem neovlašteni korisnik reproducira ranije zabilježeni mrežni promet s ciljem neovlaštenog pristupa sustavu. Za zaštitu od ovih napada Kerberos koristi dva mehanizma:

- Svaki zahtjev klijenta sadrži vremensku oznaku koju generira klijentsko računalo prilikom formiranja zahtjeva. Pri primanju zahtjeva KDC poslužitelj uspoređuje lokalno vrijeme sa vremenskom oznakom u primljenom zahtjevu i provjerava da li je vremenska razlika u skladu sa maksimalnim dozvoljenim odstupanjem (inicijalno 5 minuta). Ukoliko nije, zahtjev se odbija. Zbog toga je sinkronizacija sistemskih satova na svim sustavima vrlo važna za ispravno funkcioniranje Kerberos protokola.

- Sve Kerberos karte izdane od strane KDC poslužitelja sadrže vrijeme trajanja karte u kojem se ista može iskoristiti. Nakon što ovo vrijeme istekne karta više nije valjana i biti će generirana pogreška.

2.4. Kerberos komunikacija

Kerberos protokol najvećim se dijelom bazira na Needham-Schroeder autentikacisjkom protokolu koji je objavljen još davne 1978. godine. Iako su osnovni koncepti vrlo slični, kod Kerberos 4, a nakon toga i Kerberos 5 inačice, dodane su brojne napredne funkcionalnosti koje uklanjaju nedostatke spomenutog Needham-Schroeder protokola. Opis Kerberos komunikacije koji slijedi, vezan je uz Kerberos 5, posljednju inačicu protokola, iako su svi opisani koraci identični i kod inačice 4. Razlike između inačica 4 i 5, uglavnom su vezane uz proširenje funkcionalnosti koje dodatno olakšavaju i proširuju mogućnosti primjene Kerberos protokola (*Forwardable, Proxiable, Renewable, Postdated tickets,* korištenje ASN.1 tehnologije za opis protokola, modifikaciju formata Kerberos ticketa i sl.).
Na sljedećoj slici prikazana je razmjena poruka u Kerberos komunikaciji (Slika 1).



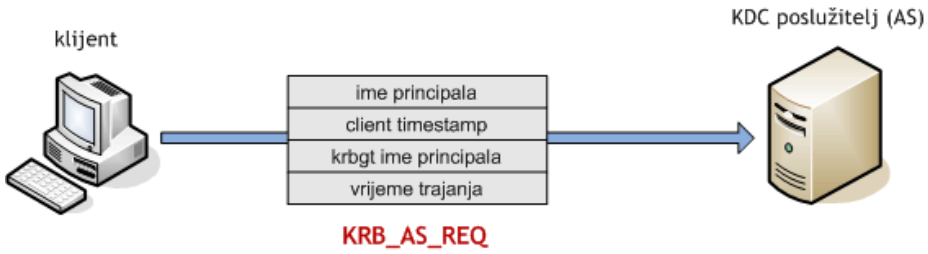
Slika 1: Razmjena Kerberos poruka

- **(1) KRB_AS_REQ zahtjev**

Postupak autentikacije korisnik inicira slanjem KRB_AS_REQ zahtjeva KDC (AS) poslužitelju. Ova poruka šalje se u čistom tekstualnom obliku (eng. *plain text*) i sadrži sljedeće elemente:

- ime principala Kerberos klijenta koji inicira zahtjev,
- vremensku oznaku (eng. *timestamp*) – lokalno vrijeme na strani klijenta,
- ime principala TGS poslužitelja (`krbtgt`),
- traženo vrijeme trajanja karte.

Izgled KRB_AS_REQ zahtjeva klijenta prikazan je u nastavku (Slika 2).



Slika 2: KRB_AS_REQ zahtjev

- **(2) KRB_AS_REP odgovor**

Pri primanju zahtjeva klijenta, AS poslužitelj u lokalnoj bazi provjerava postojanje navedenog klijentskog principala i ukoliko isti postoji vraća mu odgovor koji je kriptiran tajnim ključem koji KDC poslužitelj dijeli s istim korisnikom. Na ovaj način dobiveni odgovor može dekriptirati samo korisnik koji posjeduje odgovarajući tajni ključ, čime se poruka štiti od *sniffing* napada. Osim postojanja klijentskog principala, KDC poslužitelj također provjerava i vrijeme navedeno u dobivenom zahtjevu i uspoređuje ga sa lokalnim vremenom kako bi se sustav zaštitio od mogućnosti provođenja *replay* napada. Ukoliko je ova vremenska razlika veća od dozvoljene (oko 5 minuta), korisniku se vraća poruka o pogrešci.

KRB_AS_REP odgovor sastoji se od dva dijela. Prvi dio kriptiran je tajnim ključem korisnika (engl. *client key*) i sadrži sljedeće elemente:

- sjednički ključ koji će klijent u nastavku komunikacije koristiti za razmjenu poruka s TGS poslužiteljem (engl. *client-TGS session key*),
- ime principala TGS poslužitelja (`krbtgt`),
- vrijeme trajanja karte.

Dekriptiranjem prvog dijela poruke klijent dolazi do sjedničkog ključa kojeg će koristiti za enkripciju budućih poruka koje razmjenjuje s TGS poslužiteljem (npr. prilikom generiranja zahtjeva za pristupom određenom mrežnom resursu).

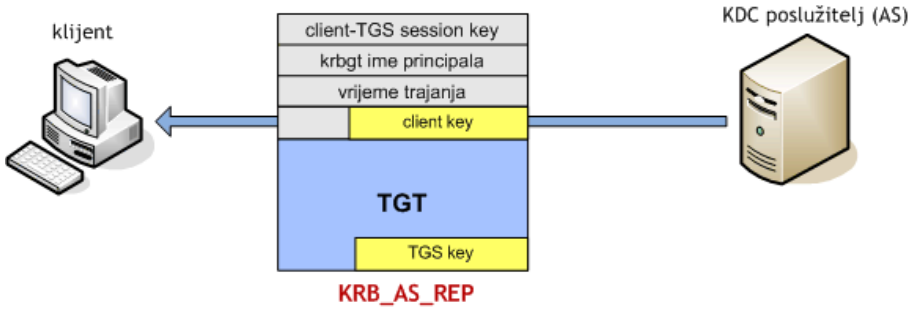
Drugi dio poruke sadrži ranije spomenutu TGT kartu koja je kriptirana tajnim ključem koji KDC poslužitelj dijeli s TGS poslužiteljem (engl. *TGS key*). To znači da ovaj dio poruke klijent nije u mogućnosti dekriptirati. Kriptiranu TGT kartu klijent će pohraniti u svoju lokalnu *cache* memoriju i iskoristit je prilikom sljedećih zahtjeva za pristupom ostalim mrežnim resursima u Kerberos sustavu.

TGT karta generira se prilikom inicijalne prijave korisnika u sustav i uz pomoću nje moguće je zatražiti pristup bilo kojem mrežnom resursu. Cijelo vrijeme dok je TGT karta valjana, klijent ne mora unositi korisničku zaporku za pristup ostalim mrežnim resursima unutar Kerberos sustava. Nakon što TGT karta istekne, klijent ponovo od AS poslužitelja mora zatražiti novu TGT kartu generiranjem novog KRB_AS_REQ zahtjeva.

Sadržaj kriptirane TGT karte je sljedeći:

- sjednički ključ koji će klijent koristiti za razmjenu poruka s TGS poslužiteljem (*client-TGS session key*),
- ime principala Kerberos klijenta,
- vrijeme trajanja karte,
- vremensku oznaku KDC poslužitelja,
- IP adresa klijenta (dobivena iz inicijalnog AS_REQ zahtjeva).

Struktura opisanog KRB_AS_REP paketa prikazana je na sljedećoj slici (Slika 3).



Slika 3: KRB_AS_REP odgovor

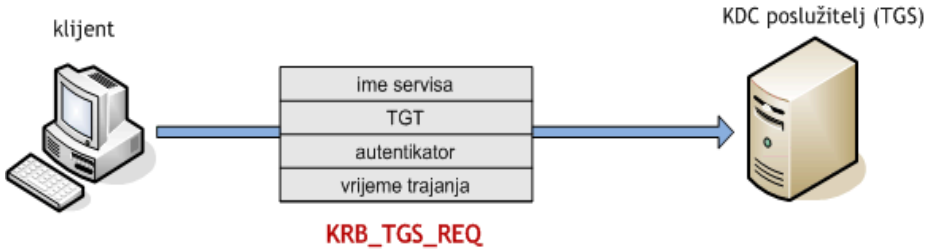
- **(3) KRB_TGS_REQ**

Nakon primanja KRB_AS_REP poruke, klijent svojim tajnim ključem (zaporkom koju je korisnik unio) pokušava dekriptirati prvi dio poruke koji sadrži sjednički ključ za komunikaciju s TGS poslužiteljem. Ukoliko je dekripcija uspješna, klijent u *cache* memoriju pohranjuje sjednički ključ i dobivenu TGT kartu. Treba napomenuti da u ovom trenutku klijent još uvijek nema pristup niti jednom mrežnom resursu unutar Kerberos sustava. On samo posjeduje TGT kartu i odgovarajuću sjednički ključ koji će mu omogućiti da od TGS poslužitelja zatraži pristup željenom resursu. Upravo je to zadatak KRB_TGS_REQ upita koji je opisan u koraku 3.

Zahtjev za pristup resursu sastoji se od tri dijela (Slika 4):

- ime principala resursa kojem klijent želi pristupiti (npr. SSH servis na udaljenom poslužitelju),
- traženo vrijeme trajanja karte,
- TGT karte pohranjene u prethodnom koraku,
- autentikatora.

Autentikator osigurava da je svaki zahtjev za pristup resursu jedinstven i potvrđuje da korisnik posjeduje tajni sjednički ključ dogovoren u ranijim fazama komunikacije.



Slika 4: KRB_TGS_REQ zahtjev

- **(4) KRB_TGS_REP**

Slično kao i u drugom koraku, pri primanju zahtjeva klijenta KDC poslužitelj formira odgovor koji će sadržavati novi sjednički ključ (engl. *client-service session key*) koji će klijent koristiti za razmjenu poruka sa resursom (poslužiteljem) kojem se zahtjeva pristup. Format ovog odgovora identičan je onome u koraku 2, samo što su vrijednosti unutar poruke različite. Dok je poruka 2 sadržavala TGT kartu i ključ koji klijent koristi za razmjenu poruka s KDC poslužiteljem, ova poruka sadržava sjednički ključ za razmjenu poruka sa zahtijevanim resursom (poslužiteljem) te TGS kartu za pristup istome. Poruka se također sastoji od dva dijela. Prvi dio kriptiran je sjedničkim ključem dogovorenim u koracima 1 i 2 između klijenta i KDC (AS) poslužitelja, i sastoji se od sljedećih elemenata:

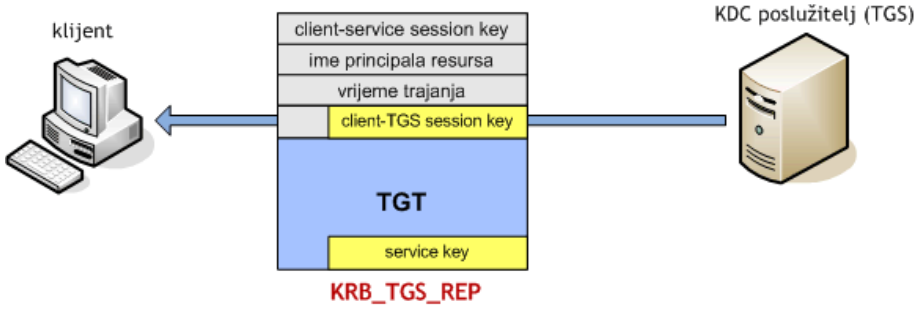
- ime principala resursa kojem klijent želi pristupiti (npr. SSH servis na udaljenom poslužitelju),
- vrijeme trajanja karte,
- sjednički ključ za razmjenu poruka sa resursom kojem se zahtjeva pristup (*client – service session key*),

Ovaj dio poruke mogu dekriptirati samo KDC (AS) poslužitelj i klijent, budući da su oni jedini koji poznaju ključ dogovoren u koracima 1 i 2.

Drugi dio poruke je TGS karta za pristup zatraženom resursu. Slično kao i TGT karta, ova ja karta kriptirana tajnim ključem koji dijele KDC poslužitelj i resurs (poslužitelj) (engl. *service key*) kojem je zatražen pristup. TGS karta sadrži sljedeće elemente:

- sjednički ključ za razmjenu poruka sa resursom kojem se zahtjeva pristup (*client – service session key*),
- ime principala klijenta,
- vrijeme trajanja karte,
- vremenska oznaka KDC poslužitelja,
- IP adresa klijenta.

Struktura opisanog KRB_TGS_REP zahtjeva prikazana je na sljedećoj slici (Slika 5).



Slika 5: KRB_TGS_REP odgovor

Nakon što klijent dekriptira primljenu poruku sjedničkim ključem iz 1 i 2 koraka, dobivenu TGS kartu i novi sjednički ključ također pohranjuje u *cache* memoriju. Pohranjene parametre klijent će koristiti u sljedećem koraku prilikom pristupa traženom resursu.

2.5. Sekundarni KDC poslužitelji

Osim primarnog KDC poslužitelja (eng. *master KDC*), u produkcijskim okruženjima svakako je preporučljivo korištenje jednog ili više sekundarnih KDC poslužitelja (eng. *slave KDC*), koji će preuzeti ulogu u slučaju nedostupnosti primarnog poslužitelja. Na ovaj način podiže se razina raspoloživosti sustava, budući da u Kerberos sustavu primarni KDC poslužitelj predstavlja tzv. *single point of failure* komponentu. Kada ne bi postojali sekundarni poslužitelji, nedostupnost primarnog KDC poslužitelja rezultirala bi nemogućnošću prijavljivanja u sustav svih korisnika, odnosno nemogućnošću uporabe mrežnih servisa u Kerberos sustavu.

Kod MIT implementacije Kerberos protokola specifično je da sekundarni poslužitelji mogu izdavati karte korisnicima, odnosno omogućiti im prijavu u sustav te pristup pojedinim servisima, međutim oni nisu u mogućnosti mijenjati podatke u Kerberos bazi. To znači da sekundarni KDC poslužitelji sadrže *read-only* kopiju podataka sa primarnog KDC poslužitelja te je sve promjene na bazi moguće raditi isključivo na primarnom poslužitelju.

3. Implementacija Kerberos protokola na Linux operacijskom sustavu

Implementacija Kerberos sustava, neovisno o kojem se okruženju i platformi radi, zahtjeva kvalitetno i detaljno planiranje. Planiranje sustava podrazumijeva korake kao što su određivanje opsega sustava, definiranje osnovnih parametara Kerberos protokola, određivanje kritičnih poslužitelja i njihove lokacije, određivanje servisa i aplikacija koji će koristiti Kerberos protokol, sigurnosne rizike i prijetnje sustavu i sl.

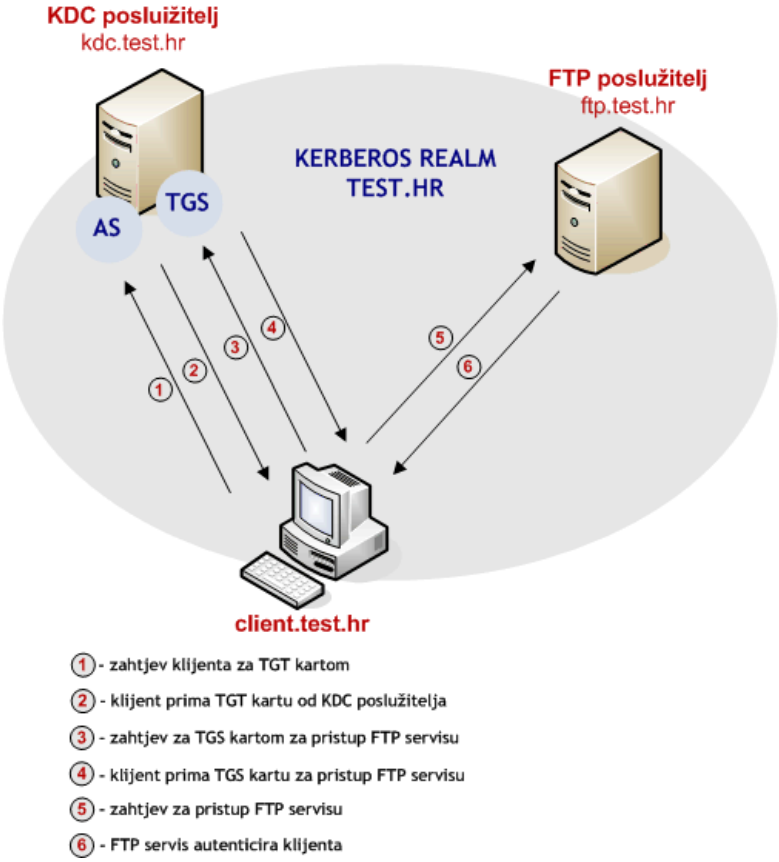
Posebnu pozornost treba posvetiti raspoloživosti Kerberos sustava, budući da o njemu ovisi velik broj korisnika i servisa. Nadzor i praćenje rada sustava također mora biti temeljito i redovito, a u slučaju problema sa radom moraju biti spremna rješenja koja će u kratkom vremenu omogućiti vraćanje sustava u funkcionalno stanje. Također, u obzir treba uzeti i zastarjele sustave koji mogu imati problema s korištenjem Kerberos protokola.

U nastavku dokumenta biti će opisan postupak implementacije i uspostave Kerberos v5 sustava na Linux operacijskom sustavu. Opisani su osnovni postupci instalacije sustava na Linux Debian

platformi, a na konkretnom primjeru opisan je i postupak podešavanja KDC poslužitelja te klijentskih računala unutar Kerberos okruženja.

3.1. Testno okruženje

U svrhu testiranja implementacije Kerberos sustava u Linux okruženju, implementiran je testni sustav prikazan na sljedećoj slici (Slika 6).



Slika 6: Testno okruženje

3.2. Vremenske postavke

Budući da je vremenska sinkronizacija između računala u Kerberos okruženju neophodna za uspješan rad sustava, preporučljivo je da se prije instalacije potrebnih programskih paketa na svim računalima podesi vrijeme korištenjem NTP servisa.

To je moguće postići korištenjem `ntpdate` programa kako je prikazano u nastavku:

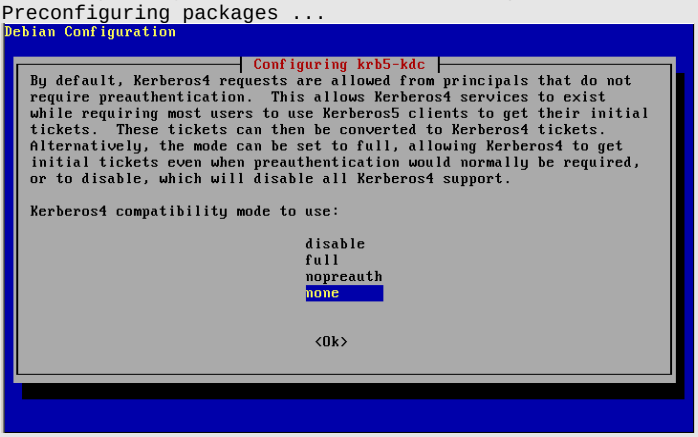
```
# ntpdate -v zg1.ntp.carnet.hr
13 Oct 19:13:05 ntpdate[1014]: ntpdate 4.2.0a@1:4.2.0a+stable-2-r Sun Jan  9
16:13:28 CET 2005 (1)
14  Oct  09:45:52  ntpdate[1014]:  step  time  server  161.53.160.4  offset
52366.767944 sec
```

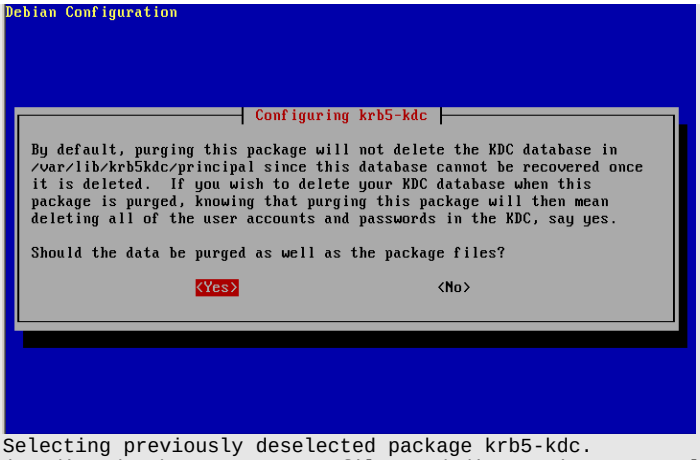
Istu naredbu preporučljivo je redovito pokretati putem `crond` poslužitelja, kako bi se osigurala stalna usklađenost vremenskih postavki na sustavu. Također, na taj način izbjeći će se brojni problemi i pogreške koje mogu nastati zbog vremenske razlike na pojedinim računalima u Kerberos sustavu.

3.3. Instalacija KDC poslužitelja

Postupak instalacije KDC poslužitelja na Debian operacijskim sustavima vrlo je jednostavan. Korištenjem `apt-get` naredbe potrebno je instalirati minimalno sljedeće pakete: `krb5-admin-server` i `krb5-kdc`. Postupak je sljedeći:

```
# apt-get install krb5-kdc
Reading Package Lists... Done
Building Dependency Tree... Done
krb5-kdc is already the newest version.
0 upgraded, 0 newly installed, 0 to remove and 5 not upgraded.
marlena:~# apt-get remove --purge krb5-kdc
Reading Package Lists... Done
Building Dependency Tree... Done
The following packages will be REMOVED:
  krb5-kdc*
0 upgraded, 0 newly installed, 1 to remove and 5 not upgraded.
Need to get 0B of archives.
After unpacking 315kB disk space will be freed.
Do you want to continue? [Y/n] Y
(Reading database ... 21463 files and directories currently installed.)
Removing krb5-kdc ...
Purging configuration files for krb5-kdc ...
dpkg - warning: while removing krb5-kdc, directory `/etc/krb5kdc' not empty
so not removed.
marlena:~# apt-get install krb5-kdc
Reading Package Lists... Done
Building Dependency Tree... Done
Suggested packages:
  krb5-admin-server
The following NEW packages will be installed:
  krb5-kdc
0 upgraded, 1 newly installed, 0 to remove and 5 not upgraded.
Need to get 0B/117kB of archives.
After unpacking 315kB of additional disk space will be used.
Preconfiguring packages ...
```



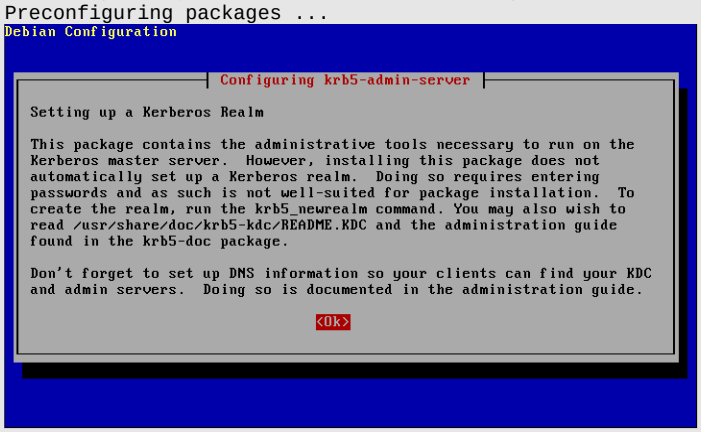



```
Selecting previously deselected package krb5-kdc.
(Reading database ... 21440 files and directories currently installed.)
Unpacking krb5-kdc (from .../krb5-kdc_1.3.6-3_i386.deb) ...
Setting up krb5-kdc (1.3.6-3) ...
#
```

Tijekom postupka instalacije `krb5-kdc` programskog paketa od korisnika se zahtjeva da definira potrebu za kompatibilnošću v4 verzije Kerberos protokola te da li je potrebno brisanje baze Kerberos korisnika ukoliko je ista prisutna na sustavu. Nakon odgovora na ova pitanja, postupak instalacije se nastavlja.

Sličan postupak potrebno je ponoviti i za `krb5-admin-server` programski paket:

```
# apt-get remove --purge krb5-admin-server
Reading Package Lists... Done
Building Dependency Tree... Done
The following packages will be REMOVED:
  krb5-admin-server*
0 upgraded, 0 newly installed, 1 to remove and 5 not upgraded.
Need to get 0B of archives.
After unpacking 274kB disk space will be freed.
Do you want to continue? [Y/n] Y
(Reading database ... 21480 files and directories currently installed.)
Removing krb5-admin-server ...
Stopping Kerberos Administration Servers: kadmind.
Purging configuration files for krb5-admin-server ...
marlena:~# apt-get install krb5-admin-server
Reading Package Lists... Done
Building Dependency Tree... Done
The following NEW packages will be installed:
  krb5-admin-server
0 upgraded, 1 newly installed, 0 to remove and 5 not upgraded.
Need to get 0B/95.2kB of archives.
After unpacking 274kB of additional disk space will be used.
Preconfiguring packages ...
```




```
Selecting previously deselected package krb5-admin-server.
(Reading database ... 21463 files and directories currently installed.)
```

```
Unpacking krb5-admin-server (from .../krb5-admin-server_1.3.6-3_i386.deb) ...
Setting up krb5-admin-server (1.3.6-3) ...
```

Nakon instalacije KDC poslužitelja i pripadajućih paketa moguće je krenuti sa podešavanjem

3.4. Podešavanje KDC poslužitelja

3.4.1. Uređivanje konfiguracijskih datoteka

Prvi korak podešavanja KDC poslužitelja je definiranje osnovnih parametara sustava, što je moguće postići uređivanjem `/etc/krb5.conf` i `/etc/krb5kdc/kdc.conf` konfiguracijskih datoteka. U nastavku je prikazan primjer ovih datoteka za testno okruženje opisano u ovom dokumentu. Sadržaj `/etc/krb5.conf` datoteke na KDC poslužitelju `kdc.test.hr` prikazan je u nastavku:

```
#
#     krb5.conf
#
# globalne postavke sustava
[libdefaults]
        default_realm = TEST.HR

# Parametri Kerberos realma
[realms]
TEST.HR = {
         kdc = kdc.test.hr
         admin_server = kdc.test.hr
         default_domain = test.hr
}

# Parametri koji opisuju vezu izmedu naziva domena i Kerberos realma
[domain_realm]
        test.hr = TEST.HR
        .test.hr = TEST.HR

# Biljezenje log zapisa
[logging]
        kdc = FILE:/var/log/krb5kdc.log
        admin_server = FILE:/var/log/kadmin.log
        default = FILE:/var/log/krb5lib.log
```

Navedeni parametri definiraju ime Kerebros realm okruženja, imena važnijih poslužitelja u sustavu, povezanost realma s nazivom domene, te lokacije na tvrdom disku na kojoj će se bilježiti log zapisi.

Slijedi uređivanje `/etc/krb5kdc/kdc.conf` datoteke (lokacija ove datoteke može varirati od sustava do sustava, ovisno o načinu instalacije paketa i distribuciji koja se koristi).

```
#
#      kdc.conf
#
[kdcdefaults]
        kdc_ports = 750,88

[realms]
TEST.HR = {
                database_name = /var/lib/krb5kdc/principal
                admin_keytab = FILE:/etc/krb5kdc/kadm5.keytab
                acl_file = /etc/krb5kdc/kadm5.acl
                key_stash_file = /etc/krb5kdc/stash
                kdc_ports = 750,88
                max_life = 10h 0m 0s
                max_renewable_life = 7d 0h 0m 0s
                master_key_type = des3-hmac-sha1
                supported_enctypes = des3-hmac-sha1:normal des-cbc-crc:normal
des:normal des:v4 des:norealm des:onlyrealm des:afs3
                default_principal_flags = +preauth
        }
```

Navedenim postavkama definiraju se mrežni portovi koje će KDC poslužitelj koristiti, lokacije važnijih datoteka na tvrdom disku te neki dodatni parametri sustava. Detaljnije informacije o parametrima `kdc.conf` datoteke moguće je naći unutar pripadajućih *man* stranica.

3.4.2. Inicijalizacija Kerberos baze

Nakon podešavanje osnovnih konfiguracijskih datoteka, moguće je pokrenuti postupak inicijalizacije Kerberos baze u kojoj će se nalaziti informacije o svim subjektima definiranog Kerberos realma. Postupak je sljedeći:

```
# kdb5_util create -s
Loading random data
Initializing database '/var/lib/krb5kdc/principal' for realm 'TEST.HR',
master key name 'K/M@TEST.HR'
You will be prompted for the database Master Password.
It is important that you NOT FORGET this password.
Enter KDC database master key: ********
Re-enter KDC database master key to verify: ********
```

Izvršavanje navedene naredbe rezultirati će inicijalizacijom Kerberos baze (`/var/lib/krb5kdc/principal`), pri čemu se od korisnika traže da navede tajni ključ kojim se kriptira Kerberos baza. Enkripcija Kerberos baze vrlo je važan korak, budući da ona sadrži podatke o svim principalima sustava i njezino kompromitiranje ugrozilo bi cijeli Kerberos sustav.

Opcija `–s` rezultirati će kreiranjem tzv. *stash* datoteke u kojoj će biti pohranjen tajni ključ kojim je baza kriptirana. Na ovaj način omogućuje se nesmetano pokretanje Kerberos poslužitelja prilikom ponovnog pokretanja sustava (eng. *reboot*), bez intervencije sistem administratora. Ukoliko ova datoteka ne bi postojala, prilikom svakog pokretanja sustava korisnik bi morao ručno unijeti odgovarajući tajni ključ što bi bilo vrlo nepraktično.

Sadržaj `/var/lib/krb5kdc` direktorija sa kreiranim datotekama prikazan je u nastavku (kao što je ranije spomenuto sadržaj navedenih datoteka je kriptiran tako da ih nije moguće pregledavati običnim uređivačem teksta).

```
# ls -al /var/lib/krb5kdc/
total 24
drwx------   2 root root 4096 Sep 21 15:03 .
drwxr-xr-x  14 root root 4096 Sep 21 14:14 ..
-rw-------   1 root root 8192 Sep 21 15:03 principal
-rw-------   1 root root 8192 Sep 21 15:03 principal.kadm5
-rw-------   1 root root    0 Sep 21 15:03 principal.kadm5.lock
-rw-------   1 root root    0 Sep 21 15:03 principal.ok
```

3.4.3. Dodavanje Kerberos principala

Sljedeći važan korak nakon inicijalnog kreiranja Kerberos baze je njeno popunjavanje principalima, odnosno subjektima koji će koristiti Kerberos sustav. U tu svrhu uobičajeno se koristi `kadmin` program, koji predstavlja sučelje prema KDC administracijskom poslužitelju. Međutim, spomenuti program odmah nakon inicijalizacije baze nije moguće koristiti budući da u bazu nisu dodani odgovarajući korisnički računi potrebni za administraciju sustava. Naime, za autentikaciju na KDC administracijski poslužitelj `kadmin` program koristi autentikacijske parametre iz Kerberos baze koji u ovom trenutku još nisu definirani. Kako bi se uklonio ovaj problem, na KDC poslužitelju dostupan je `kadmin.local` program koji izravno pristupa bazi bez potrebe za autentikacijom. Kako bi se omogućilo korištenje `kadmin` programa, potrebno je pomoću `kadmin.local` naredbe u Kerberos bazu dodati odgovarajući korisnički račun za administraciju Kerberos baze. Postupak je sljedeći:

```
#kadmin.local
Authenticating as principal root/admin@TEST.HR with password.

kadmin.local:  addprinc admin/admin
WARNING: no policy specified for admin/admin@TEST.HR; defaulting to no policy
Enter password for principal "admin/admin@TEST.HR":
Re-enter password for principal "admin/admin@TEST.HR":
Principal "admin/admin@TEST.HR" created.
```

Nakon ovog koraka u bazu je dodan principal pod nazivom `admin/admin` putem kojeg će se moći pristupiti KDC administarcijskom poslužitelju korištenjem `kadmin` programa. Također, unutar `/etc/krb5kdc/kadm5.acl` konfiguracijske datoteke potrebno je definirati ovlasti koje će ovaj korisnički račun imati. Budući da se radi o administratorskom računu, dane su mu potpune ovlasti.

```
*/admin@TEST.HR    *
```

Na sličan način moguće je precizno definirati ovlasti za sve korisnike Kerberos sustava.Za detaljnije informacije o sintaksi i načinu uređivanja `kadm.acl` konfiguracijske datoteke korisnici se upućuju na dokumentaciju Kerberos MIT programskog paketa [2].

Nakon što je dodan administratorski korisnički račun, administraciju Kerberos sustava moguće je provoditi korištenjem `kadmin` programa s bilo kojeg računala u Kerberos sustavu. Primjer:

```
# kadmin -p admin/admin
Authenticating as principal admin/admin with password.
Password for admin/admin@TEST.HR: *****
kadmin:  ?
Available kadmin requests:

add_principal, addprinc, ank
                         Add principal
delete_principal, delprinc
                         Delete principal
modify_principal, modprinc
                         Modify principal
change_password, cpw     Change password
get_principal, getprinc  Get principal
list_principals, listprincs, get_principals, getprincs
                         List principals
add_policy, addpol       Add policy
modify_policy, modpol    Modify policy
delete_policy, delpol    Delete policy
get_policy, getpol       Get policy
list_policies, listpols, get_policies, getpols
                         List policies
get_privs, getprivs      Get privileges
ktadd, xst               Add entry(s) to a keytab
ktremove, ktrem          Remove entry(s) from a keytab
lock                            Lock database exclusively (use with extreme
caution!)
unlock                   Release exclusive database lock
list_requests, lr, ?     List available requests.
quit, exit, q            Exit program.
```

Nakon uspješne autentikacije, aktivira se konzola `kadmin` programa putem koje je moguća daljnja administracija Kerberos sustava. Listu dostupnih naredbi moguće je dobiti unosom znaka upitnika (?), kao što je prikazano u prethodnom primjeru. U nastavku je prikazan primjer korištenja nekih od važnijih naredbi. Npr. listu svih korisnika u bazi moguće je dobiti zadavanjem naredbe `list_principals` (skraćeno `listprincs`), a nove principale (korisnike, računala, servise i sl.) moguće je dodati korištenjem naredbe `add_principal` (skraćeno `addprinc`).

```
kadmin:  listprincs
K/M@TEST.HR
admin/admin@TEST.HR
kadmin/admin@TEST.HR
kadmin/changepw@TEST.HR
kadmin/history@TEST.HR
krbtgt/TEST.HR@TEST.HR

kadmin:  addprinc sjusic
WARNING: no policy specified for sjusic@TEST.HR; defaulting to no policy
Enter password for principal "sjusic@TEST.HR": *****
Re-enter password for principal "sjusic@TEST.HR": *****
Principal "sjusic@TEST.HR" created.

kadmin:  addprinc test
WARNING: no policy specified for test@TEST.HR; defaulting to no policy
Enter password for principal "test@TEST.HR": *****
Re-enter password for principal "test@TEST.HR": *****
Principal "test@TEST.HR" created.

kadmin:  listprincs
K/M@TEST.HR
admin/admin@TEST.HR
kadmin/admin@TEST.HR
kadmin/changepw@TEST.HR
kadmin/history@TEST.HR
krbtgt/TEST.HR@TEST.HR
sjusic@TEST.HR
test@TEST.HR
kadmin:
```

Na danom primjeru prikazan je postupak dodavanja dva korisnika u Kerberos sustav (korisnici `test` i `sjusic`). Također, prilikom dodavanja svakog od korisnika potrebno je definirati i zaporku koja će se koristiti za generiranje tajnog ključa koji će korisnik dijeliti s KDC poslužiteljem.
Na sličan način potrebno je u bazu KDC poslužitelja dodati i ostale principale Kerberos sustava. U našem primjeru radi se o klijentu koji će koristi Kerberos servise (`client.test.hr`) te poslužitelju na kojem je pokrenut FTP poslužitelj sa Kerberos podrškom (`ftp.test.hr`). Naravno, u stvarnim informacijskim sustavima, isti postupak potrebno je provesti za sve entitete koji će biti dio Kerberos sustava.

```
kadmin:  addprinc -randkey host/ftp.test.hr
WARNING: no policy specified for host/ftp.test.hr@TEST.HR; defaulting to no
policy
Principal "host/ftp.test.hr@TEST.HR" created.

kadmin:  addprinc -randkey host/kdc.test.hr
WARNING: no policy specified for host/kdc.test.hr@TEST.HR; defaulting to no
policy
Principal "host/kdc.test.hr@TEST.HR" created.

kadmin:  addprinc -randkey host/client.test.hr
WARNING: no policy specified for host/client.test.hr@TEST.HR; defaulting to
no policy
Principal "host/client.test.hr@TEST.HR" created.

kadmin:  listprincs
K/M@TEST.HR
admin/admin@TEST.HR
host/client.test.hr@TEST.HR
host/ftp.test.hr@TEST.HR
host/kdc.test.hr@TEST.HR
kadmin/admin@TEST.HR
kadmin/changepw@TEST.HR
kadmin/history@TEST.HR
krbtgt/TEST.HR@TEST.HR
sjusic@TEST.HR
test@TEST.HR
kadmin:
```

Prilikom dodavanja principala koji opisuju računala i servise (npr. `host/ftp.test.hr` ili `ftp/ftp.test.hr`) potrebno je koristiti parametar `randkey` koji će omogućiti automatsko definiranje tajnog ključa bez potrebe za unosom zaporke. Budući da se o ovom slučaju radi o principalima koji se neće interaktivno prijavljivati u sustav, korištenje zaporke prilikom generiranja Kerberos karti bilo bi vrlo nepraktično.
Osim pojedinačnih računala i poslužitelja, u Kerberos bazu potrebno je dodati i sve principale koji opisuju servise koji će se koristiti unutar Kerberos sustava (npr. `ssh`, `ftp` i sl.). Za razliku od host principala koji imaju format `host/<ime racunala>` (npr. `host/ftp.test.hr`), principali za servise imaju oblik `servis/<ime racunala>` (npr. `ftp/ftp.test.hr`, `ssh/ftp.test.hr` i sl.).
U nastavku je prikazan primjer dodavanja principala za FTP servis pokrenut na poslužitelju `ftp.test.hr`.

```
kadmin:  addprinc -randkey ftp/ftp.test.hr
WARNING: no policy specified for ftp/ftp.test.hr@TEST.HR; defaulting to no
policy
Principal "ftp/ftp.test.hr@TEST.HR" created.
```

Tajni ključevi koji su generirani za pojedina računala i servise u ovome su trenutku pohranjeni u Kerberos bazi na KDC poslužitelju. Budući da se radi o tajnom parametru koji KDC poslužitelj dijeli sa pojedinim principalima u Kerberos okruženju, iste ključeve biti će potrebno prebaciti na odgovarajuća računala. Konkretno, u ovom primjeru potrebno je na računalo `ftp.test.hr` prebaciti ključeva za principale `ftp/ftp.test.hr` i `host/ftp.test.hr`. Na računalo `client.test.hr` potrebno je prebaciti samo ključ za principal `host/client.test.hr`, budući da njemu nije pokrenut niti jedan servis.

3.5. Podešavanje klijentskih računala

Svi klijenti i aplikacijski poslužitelji sa podrškom za Kerberos protokol mogu se smatrati Kerberos klijentima. Osim ranije opisanog podešavanja KDC poslužitelja, za uspješno prijavljivanje u sustav i korištenje pojedinih servisa potrebna su određena podešavanja na svim klijentima u Kerberos sustavu. Potrebno je instalirati određene programske biblioteke, kreirati potrebne konfiguracijske datoteke te učitati potrebne ključeve u *keytab* datoteke o kojima će više riječi biti u nastavku dokumenta. Naravno, u Kerberos bazu na KDC poslužitelju potrebno je dodati odgovarajuće principale koji opisuju svako klijentsko računalo i servis u Kerberos okruženju. Dodavanje principala u bazu moguće je obaviti ili lokalno na KDC poslužitelju korištenjem `kadmin.local` ili `kadmin` programa, ili udaljeno sa nekog od klijenata korištenjem `kadmin` aplikacije. U primjeru koji je korišten u svrhu izrade ovog dokumenta, principali za Kerberos klijentska računala i servise već su dodani u Kerberos bazu na KDC poslužitelju.

3.6. Instalacija klijentskih programskih paketa

Instalacija programskih paketa na Kerberos klijentima ovisi o funkciji koju pojedino računalo obavlja. Na računalu koje se isključivo koristi kao klijent za pristup ostalim servisima u Kerberos okruženju dovoljno je instalirati osnovne programske pakete koji će osigurati Kerberos podršku. Na Linux Debian sustavu koji je korišten kao testna platforma za izradu ovog dokumenta instalirani su sljedeći programski paketi `krb5-user`, `libkrb53` i `krb5-config`. Dodatno, na poslužitelju `ftp.test.hr` koji ujedno obavlja i funkciju FTP poslužitelja instaliran je i `krb5-ftpd` programski paket, dok je na klijentskom računalu instaliran paket `krb5-clients` koji sadrži različite klijentske aplikacije za pristup servisima sa ugrađenom Kerberos podrškom. Za prijavljivanje klijenata u sustav važna je biblioteka `libkrb53` te programski paket `krb5-user` koji sadrži programe kao što su `kinit`, `klist`, `kdestroy`, `ktutil` i `sl`.

Naravno, imena navedenih programskih paketa razlikuju se između pojedinih Linux distribucija.

Također, na klijentskim računalima moguće je instalirati i PAM modul za Kerberos autentikaciju, kako bi se omogućilo korištenje Kerberos protokola za različite se servise na sustavu. PAM modul za Kerberos autentikaciju na Linux Debian sustavu dolazi u paketu pod nazivom `libpam-krb5`, i moguće ga je jednostavno instalirati korištenjem apt.-get naredbe.

```
# apt-get install libpam-krb5
```

3.7. Podešavanje klijentskih računala

Podešavanje klijentskih računala prilično je jednostavno i svodi se na uređivanje `krb5.conf` konfiguracijske datoteke te učitavanje potrebnih ključeva u lokalnu `krb5.keytab` datoteku s KDC poslužitelja. `Krb5.conf` konfiguracijsku datoteku najjednostavnije je kopirati s KDC poslužitelja, budući da su u njoj navedeni svi potrebni parametri. Postupak prebacivanja i učitavanja ključeva nešto je složeniji i opisan je u nastavku.

Prebacivanje ključeva iz Kerberos baze na ostala računala u sustavu provodi se putem ranije spomenutih *keytab* datoteka. *Keytab* datoteka sadrži jedan ili više ključeva koji se koriste za komunikaciju s KDC poslužiteljem i mora biti prisutna na svim poslužiteljima u Kerberos sustavu. Razlog tome jasan je sam po sebi ukoliko se uzme u obzir sam način na koji radi Kerberos protokol. U opisu rada protokola navedeno je kako svaki principal u Kerberos sustavu s KDC poslužiteljem dijeli tajnu, odnosno tajni ključ, koji se koristi za enkripciju paketa koji se razmjenjuju s KDC poslužiteljem. U slučaju principala koji opisuju korisničke račune, ova "tajna" predstavljena je u obliku *hash* vrijednosti zaporke koju korisnik upisuje kada se želi prijaviti u sustav. Budući da principali koji opisuju pojedina računala i servise u Kerberos sustavu ne mogu interaktivno unositi zaporke prilikom komunikacije s KDC poslužiteljem, u tu svrhu se koriste keytab datoteke koje sadrže tajni ključ koji pojedini principal dijeli s KDC poslužiteljem. Upravo je zato na svim poslužiteljima koji sudjeluju u Kerberos komunikaciji potrebno podesiti odgovarajuće *keytab* datoteke koje će sadržavati tajne ključeve potrebne za komunikaciju s KDC poslužiteljem.

Budući da *keytab* datoteke imaju specifičan format i strukturu, s Kerberos programskim paketom dolazi specijalizirani `ktutil` program za njihovo uređivanje. U nastavku dokumenta prikazan je postupak prebacivanja ključeva iz Kerberos baze na KDC poslužitelju na pojedina računala u Kerberos sustavu.

U prvom koraku potrebno je iz Kerberos baze izvesti (engl. *export*) željene ključeve u privremene *keytab* datoteke koje će se kasnije sigurnim putem prebaciti na drugo računala. Pojam "sigurnim putem" vrlo je važan, budući da *keytab* datoteke sadrže tajne ključeve ključne za Kerberos komunikaciju te ih je kao takve potrebno i tretirati. U sljedećem primjeru iz baze su izvučeni ključevi za principale `host/ftp.test.hr`, `host/client.test.hr` i `ftp/ftp.test.hr`.

```
kadmin.local:  ktadd -k /tmp/host.ftp.keytab host/ftp.test.hr
kadmin.local:  ktadd -k /tmp/host.client.keytab host/client.test.hr
kadmin.local:  ktadd -k /tmp/ftp.ftp.keytab ftp/ftp.test.hr
```

Iz prikazanog primjera, može se vidjeti da se izvoženje tajnog ključa u *keytab* datoteku provodi korištenjem `ktadd` naredbe unutar `kadmin` programa. Izvršavanje prikazanih naredbi rezultirati će trima datotekama u `tmp` direktoriju koje sadrže tajne ključeve principala `host/ftp.test.hr`, `host/client.test.hr` i `ftp/ftp.test.hr`.

```
# ls –al
total 20
drwxrwxrwt   2 root root 4096 Oct  3 16:06 .
drwxr-xr-x  21 root root 4096 Jun 18 17:54 ..
-rw-------   1 root root  150 Oct  3 15:59 ftp.ftp.keytab
-rw-------   1 root root  152 Oct  3 15:59 host.client.keytab
-rw-------   1 root root  152 Oct  3 15:59 host.ftp.keytab
```

Navedene datoteke potrebno je "sigurnim putem" prenijeti na odgovarajuća računala. Tijekom testiranja u ovu je svrhu korišten *Secure Copy* (scp) protokol (sam postupak prebacivanja datoteka nije prikazan s obzirom da nije direktno vezan uz sam Kerberos protokol).

Nakon što su datoteke prebačene na pojedina računala, potrebno je iz njih izvesti ključeve te ih učitati u globalnu `keytab` datoteku sustava (`/etc/krb5.keytab`), koja će se koristiti prilikom komunikacije s KDC poslužiteljem. Postupak dodavanja ključa u navedenu datoteku provodi se korištenjem ranije spomenutog `ktutil` programa.

```
ktutil:  rkt host.ftp.keytab
ktutil:  rkt ftp.ftp.keytab
ktutil:  l
slot KVNO Principal
---- ---- ---------------------------------------------------------------------
--
   1    9     host/ftp.test.hr@TEST.HR
   2    9     host/ftp.test.hr@TEST.HR
   3    3      ftp/ftp.test.hr@TEST.HR
   4    3      ftp/ftp.test.hr@TEST.HR
ktutil:  wkt /etc/krb5
```

Naredbom `rkt` prvo su učitani ključevi iz `keytab` datoteka `host.ftp.keytab` i `ftp.ftp.keytab` datoteka, nakon čega su isti naredbom `wkt` zapisani u `/etc/krb5.keytab` datoteku. Naredbom `l` (list) moguće je vidjeti ključeve koji su učitani unutar `ktutil` programa. `Klist` naredbom moguće je provjeriti sadržaj datoteke `/etc/krb5.keytab` datoteke, odnosno da li su u njoj sadržani svi potrebni ključevi.

```
# klist -k
Keytab name: FILE:/etc/krb5.keytab
KVNO Principal
---- --------------------------------------------------------------------------
   9 host/ftp.test.hr@TEST.HR
   9 host/ftp.test.hr@TEST.HR
   3 ftp/ftp.test.hr@TEST.HR
   3 ftp/ftp.test.hr@TEST.HR
```

Identičan postupak potrebno je ponoviti i na svim ostalim računalima u Kerberos sustavu. U ovom primjeru potrebno je na računalo `klijent.test.hr` prebaciti *keytab* datoteku `host/klijent.test.hr` te `ktutil` programom učitati odgovarajuće ključeve u lokalnu `/etc/krb5.keytab` datoteku. Nakon što su u Kerberos bazu dodani svi potrebni principali i nakon što su na klijentskim računalima podešeni svi parametri i učitani odgovarajući ključevi unutar *keytab* datoteka, moguće je koristiti Kerberos autentikaciju za pristup servisima sa ugrađenom Kerberos podrškom.

3.7.1. Prijavljivanje u sustav

U nastavku poglavlja opisan je proces prijavljivanja u sustav korištenjem Kerberos protokola. Opisane su naredbe koje se pritom koriste te log zapisi sa KDC poslužitelja koji dodatno pojašnjavaju cijeli proces.

Postupak prijavljivanja u sustav inicira se korištenjem `kinit` naredbe. Nakon zadavanja `kinit` naredbe, od korisnika se traže da unese odgovarajuću korisničku zaporku.

```
client$ kinit sjusic
Password for sjusic@TEST.HR: *****
```

Ukoliko opisani korak prođe bez greške, korisnik se uspješno autenticirao kod KDC poslužitelja i od njega dobio TGT kartu koju će kasnije koristiti za pristup ostalim servisima. Korisničke karte uobičajeno su pohranjene u `tmp` direktoriju i moguće ih je pregledati korištenjem `klist` naredbe.

```
client$ klist
Ticket cache: FILE:/tmp/krb5cc_1003
Default principal: sjusic@TEST.HR

Valid starting     Expires            Service principal
10/13/05 11:03:24  10/13/05 21:03:24  krbtgt/TEST.HR@TEST.HR

Kerberos 4 ticket cache: /tmp/tkt1003
klist: You have no tickets cached
```

Dobiveni ispis prikazuje da je TGT karta za korisnika sjusic pohranjena u datoteci `/tmp/krb5cc_1003`. Log zapis na KDC poslužitelju također potvrđuje proces izdavanja TGT karte.

```
Oct 13 11:08:40 kdc krb5kdc[854](info): AS_REQ (7 etypes {18 17 16 23 1 3 2})
161.53.64.12: ISSUE: authtime 1129194520, etypes {rep=16 tkt=16 ses=16},
sjusic@TEST.HR for krbtgt/TEST.HR@TEST.HR
```

Nakon što korisnik posjeduje TGT kartu, moguć je pristup ostalim servisima u Kerberos okruženju. U našem primjeru prikazan je primjer autentikacije prema FTP poslužitelju sa ugrađenom Kerberos podrškom.

Korištenjem FTP klijentskog programa prikazan je postupak prijave na FTP poslužitelj na adresi ftp.test.hr.

```
client~$ ftp ftp.test.hr
Connected to ftp.test.hr.
220 ftp FTP server (Version 5.60) ready.
334 Using authentication type GSSAPI; ADAT must follow
KERBEROS_V4 accepted as authentication type
Kerberos V4 krb_mk_req failed: You have no tickets cached
Remote system type is UNIX.
Using binary mode to transfer files
ftp>
```

Postupak autentikacije uspješno je izvršen, pri čemu od FTP servis od korisnika nije tražio unos korisničkog imena niti zaporke. Cijeli proces za korisnika je potpuno transparentan, a autentikacija je provedena korištenjem Kerberos karata. Ponovno zadavanje `klist` naredbe pokazuje da je na klijentskom računalu osim ranije dobivene TGT karte, pohranjena i nova karta za pristup FTP servisu.

```
client~$ klist
Ticket cache: FILE:/tmp/krb5cc_1003
Default principal: sjusic@TEST.HR

Valid starting     Expires            Service principal
10/13/05 11:08:40  10/13/05 21:08:40  krbtgt/TEST.HR@TEST.HR
10/13/05 11:12:01  10/13/05 21:08:40  ftp/ftp.test.hr@TEST.HR
10/13/05 11:12:01  10/13/05 21:08:40  host/ftp.test.hr@TEST.HR


Kerberos 4 ticket cache: /tmp/tkt1003
klist: You have no tickets cached
```

Log zapisi na KDC poslužitelju potvrđuju i ovaj postupak.

```
Oct 13 11:12:01 kdc krb5kdc[854](info): TGS_REQ (7 etypes {18 17 16 23 1 3
2}) 161.53.64.12: ISSUE: authtime 1129194520, etypes {rep=16 tkt=16 ses=16},
sjusic@TEST.HR for ftp/ftp.test.hr@TEST.HR

Oct 13 11:12:01 kdc krb5kdc[854](info): TGS_REQ (7 etypes {18 17 16 23 1 3
2}) 161.53.64.12: ISSUE: authtime 1129194520, etypes {rep=16 tkt=16 ses=16},
sjusic@TEST.HR for host/ftp.test.hr@TEST.HR
```

Na sličan način moguće je pristupiti i svim drugim servisima u Kerberos sustavu, za koji postoje odgovarajući principali u Kerberos bazi i na kojima su pokrenuti servisi sa ugrađenom Kerberos podrškom.

Kerberos autentikaciju moguće je podesiti i korištenjem PAM modula, ranije spomenutog u poglavlju 3.6. U tom slučaju potrebno je u `/etc/pam.d/common-auth` datoteci unijeti sljedeće parametre.

```
auth    sufficient      pam_krb5.so
auth    required        pam_unix.so nullok_secure try_first_pass
```

Navedene linije omogućiti će da različite aplikacije na sustavu koriste Kerberos autentikaciju korištenjem `pam_krb5.so` PAM modula. Nakon prikazanih izmjena, interaktivno prijavljivanje u sustav putem `login` programa izgledati će:

```
Debian GNU/Linux testing/unstable ftp tty1

ftp login: sjusic
Password for sjusic@ZESOI.FER.HR: ******
# id
uid=1001(sjusic) gid=1001(sjusic) groups=1001(sjusic)
```

4. Zaključak

U dokumentu je dan detaljan pregled osnovnih značajki Kerberos protokola te postupak njegove implementacije u Linux okruženjima. Za razliku od novijih generacija Windows operacijskih sustava, gdje se Kerberos protokol koristi kao primarni autentikacijski mehanizam ugrađen u jezgru sustava i kao takav je potpuno transparentan za većinu korisnika, postupak implementacije u Linux okruženjima nešto je složeniji. Uspješna implementacija od sistem administratora zahtjeva određeno poznavanje načina rada Kerberos protokola, te iskustvo u radu s Linux/Unix operacijskim sustavima. Pritom treba napomenuti da je kao testna platforma korišten Linux Debian operacijski sustav te da su sva podešavanja sustava bila provedena ručno u skladu s dokumentacijom programskog paketa. Kod nekih komercijalnih Linux operacijskih sustava kao što su Linux Red Hat i Linux Novell, moguće je očekivati da je postupak implementacije sustava olakšan specijaliziranim alatima koji korisnika vode kroz proces instalacije i podešavanja sustava. No, usprkos nešto kompleksnijem procesu instalacije, nakon što je sustav uspješno uspostavljen, sve prednosti Kerberos protokola vrlo brzo dolaze do izražaja.

5. Reference

[1] Kerberos: The Network Authentication Protocol, http://web.mit.edu/kerberos/www/

[2] Kerberos Infrastructure HOWTO, http://www.linux.com/howtos/Kerberos-Infrastructure-HOWTO/index.shtml

[3] Kerberos – The definitive guide, O'Reilley