
UNIVERZITET SINGIDUNUM



DEPARTMAN ZA POSTDIPLOMSKE STUDIJE

MASTER RAD

MASTER STUDIJSKI PROGRAM:

Napredne informaciono komunikacione tehnologije

TEMA:

**Zaštita podataka u WAN komunikaciji, implementirane
funkcionalnosti zaštite i mogućnosti backup-a
podataka u Informix bazi**

Mentor:
Prof. dr Mladen Veinović

Student: Miloš Kurteš
Br. indeksa: 54/2007

UNIVERZITET SINGIDUNUM



DEPARTMAN ZA POSTDIPLOMSKE STUDIJE

MASTER RAD

MASTER STUDIJSKI PROGRAM:

Napredne informaciono komunikacione tehnologije

TEMA:

**Zaštita podataka u WAN komunikaciji, implementirane
funkcionalnosti zaštite i mogućnosti backup-a
podataka u Informix bazi**

Mentor:
Prof. dr Mladen Veinović

Student: Miloš Kurteš
Br. indeksa: 54/2007

Zaštita podataka u WAN komunikaciji, implementirane funkcionalnosti zaštite i mogućnosti backup-a podataka u Informix bazi

Sažetak:

Master rad na temu zaštite podataka u WAN komunikaciji, implementirane funkcionalnosti zaštite i mogućnosti backup-a podataka u Informix bazi pisan je sa idejom da prikaže i ispita zaštita podataka koji se šalju sa radne stanice kroz transportni put WAN strukture, ugrađene mogućnosti Informix baze zaštite podataka kroz davanja prava pristupa i šifrovanja podataka kao i mogućnosti koje pruža IDS kada je u pitanju pravljenje sigurnosnih kopija baze podataka.

Testiranje tj praktično ispitivanje sa prikazanim primerima konfiguracionih fajlova i log fajlova, izvršeno je na operativnom informacionom sistemu firme Viator&Vektor Novi Sad d.o.o. Zaključci i komentari predstavljaju direktni proizvod neposrednih ispitivanja "na lokaciji".

Ključne reči:

WAN komunikacija, Informix baza podataka, IDS, Viator&Vektor Novi Sad d.o.o.

Data protection in WAN communication, native security functionalities and backup abilities in Informix database

Abstract:

This Master's thesis on data protection in WAN communication, native security functionalities and backup abilities in Informix database was written with the idea to demonstrate and test the protection of data sent from workstations through WAN structure transport route, incorporated ability of Informix database to protect data through granting access and data coding rights, and the possibilities provided by IDS in making database backup copies.

Testing, i.e. practical examination with shown examples of configuration and log files was conducted on the operative system of Viator&Vektor Novi Sad d.o.o., and the comments are the direct outcome of on-site hands-on examination.

Key words:

WAN communication, Informix database, IDS, Viator&Vektor Novi Sad d.o.o.

SADRŽAJ

1	Uvod.....	- 1 -
2	Tehnologije WAN povezivanja	- 2 -
2.1	Frame Relay.....	- 3 -
3	Metode istraživanja	- 7 -
4	Uvod u IPsec.....	- 8 -
5	Dizajn sistema.....	- 12 -
5.1	Transportni način rada	- 13 -
5.2	Tuneliranje	- 15 -
6	Protokoli implementacije	- 17 -
6.1	AH (Authentication Header) protokol.....	- 17 -
6.2	ESP (Encapsulated Security Payload) protokol	- 19 -
7	Uspostavljanje IPsec komunikacije	- 21 -
7.1	IKE protokol.....	- 21 -
7.1.1	IKE (Internet Key Exchange).....	- 22 -
7.1.2	Uspostavljanje IKE SA	- 22 -
7.1.3	Uspostavljanje IPsec SA.....	- 24 -
7.2	IPsec redundancija.....	- 24 -
7.3	Redundancija zaglavlja	- 24 -
7.4	Redundancija ispune.....	- 26 -
8	Implementacije IPsec protokola	- 28 -
8.1	Implementacijski problemi.....	- 28 -
8.1.1	NAT.....	- 28 -
8.1.2	Fragmentacija	- 29 -
8.2	Cisco i IPsec implementacija.....	- 29 -
8.3	WWAN i APN segment i IPsec implementacija	- 37 -
9	Izbor baze i modela pohranjivanja podataka	- 38 -
9.1	Relaciona baza - pregled	- 38 -
9.2	Sigurnost podataka u bazi - trenutno stanje	- 41 -
10	Istraživanje baze	- 42 -
11	Zaštita podataka u informix bazi.....	- 43 -
12	Autentikacija.....	- 44 -
12.1	Izbor vrsta autentikacije pri konekciji.....	- 44 -
12.1.1	Potpuno poverenje.....	- 44 -
12.1.2	Delimično poverenje	- 45 -
12.1.3	Bez poverenja	- 45 -
12.1.4	Predstavljanje kao drugi korisnik.....	- 46 -
12.1.5	Osnovna autentikacija OS šifre	- 47 -
12.2	Autentikacija pomoću zamenljivih modula (PAM).....	- 49 -
12.2.1	Osnovna konfiguracija	- 50 -
12.2.2	Upotreba autentikacije šifre.....	- 50 -
12.2.3	Upotreba autentikacije zahtev-odgovor.....	- 50 -
12.2.4	Zahtevi od aplikacije kod autentikacije zahtev-odgovor	- 50 -
12.3	Upotreba LDAP direktorijuma.....	- 51 -
13	Autorizacija.....	- 52 -

13.1	Pristup baziran po osnovu uloge - role (RBAC).....	52 -
13.2	Pristup baziran po osnovu oznake - labela (LBAC).....	54 -
14	Šifrovanje	57 -
14.1	Oblast šifrovanja	57 -
14.2	Šifrovanje i dešifrovanje	58 -
14.2.1	Osnovna tehnika šifrovanja i dešifrovanja.....	58 -
14.2.2	Povrat šifrovanih binarnih podataka.....	61 -
14.2.3	Indeksiranje šifrovanih podataka.....	62 -
14.3	Sakrivanje algoritma šifrovanja pomoću 'pogleda'	63 -
14.4	Pristupna šifra	65 -
14.4.1	Upotreba pristupne šifre	65 -
14.4.2	Šifrovanje pristupne šifre tokom transporta.....	66 -
14.4.3	Broj i oblik pristupnih šifri	67 -
14.4.4	Isticanje važnosti pristupne šifre	67 -
14.4.5	Upotreba asocijacije za pristupnu šifru	70 -
14.5	Veličina šifrovanog podatka	70 -
14.5.1	Određivanje veličine šifrovanog podatka	70 -
14.5.2	Greške kada je mali prostor za pohranu podatka.....	72 -
15	Backup podataka na Informix database serverima	74 -
15.1	Off-line backup na dnevnom nivou.....	74 -
15.2	On-line backup.....	75 -
15.3	Backup uz pomoć replikacije.....	76 -
16	Zaključak.....	78 -
17	Literatura.....	80 -

Slike

Slika 1.	Osnovna šema FR veze	3 -
Slika 2.	Metod "svaki sa svakim"	4 -
Slika 3.	Metod „zvezda“	4 -
Slika 4.	Šematski prikaz WAN "Viator&Vektor Novi Sad" d.o.o.....	5 -
Slika 5.	Položaj VPN tehnologije u odnosu na OSI model.....	6 -
Slika 6.	OSI referentni model.....	9 -
Slika 7.	Struktura IP paketa za prenos preko Ethernet mreže.....	10 -
Slika 8.	Struktura IPsec paketa podataka za prenos preko Ethernet mreže.....	10 -
Slika 9.	Prikaz sistema baziranog na IPsec protokolu	12 -
Slika 10.	AH u transportnom načinu rada	13 -
Slika 11.	ESP u transportnom načinu rada.....	14 -
Slika 12.	ESP + AH u transportnom načinu rada.....	14 -
Slika 13.	Tuneliranje	15 -
Slika 14.	Tuneliranje korištenjem AH.....	16 -
Slika 15.	Tuneliranje korištenjem ESP.....	16 -
Slika 16.	Standardni oblik IP datagrama.....	17 -
Slika 17.	AH zaglavlje.....	18 -
Slika 18.	ESP datagram	19 -
Slika 19.	Uspostavljanje SA veze pomoću IKE daemona.....	22 -
Slika 20.	WAN mreža sa tri routera	30 -
Slika 21.	Model baze na logičkom nivou - entiteti sa svojim vezama	39 -
Slika 22.	Model baze na logičkom nivou - šema relacija (sa nazivima atributa)-	40 -

Tabele

Tabela 1. Redundancija IPSec zaglavlja	- 25 -
Tabela 2. Prikaz prosečne redundancije IPsec zaglavlja u odnosu na dužinu datagrama	- 26 -
Tabela 3. Dužine ulaznih blokova podataka za uobičajene kriptografske algoritme i hash funkcije	- 26 -
Tabela 4. Redundancija ispune u najgorim slučajevima	- 27 -

Primeri

Primer 1. Konfiguracija Router-a br. 1	- 30 -
Primer 2. Konfiguracija Router-a br. 2	- 32 -
Primer 3. Konfiguracija Router-a br. 3	- 35 -
Primer 4. Konektovanje sa s=3.....	- 45 -
Primer 5. Konektovanje sa s=0.....	- 45 -
Primer 6. Konektovanje sa r=1, sadržaj .netrc fajla	- 47 -
Primer 7. Kreiranje korisničkih naloga na SuSE Linux-u	- 47 -
Primer 8. Pokušaji i dozvoljena konektovanja	- 47 -
Primer 9. RBAC tehnika	- 53 -
Primer 10. Sigurnosna komponenta polje.....	- 54 -
Primer 11. Sigurnosna komponenta set	- 55 -
Primer 12. Sigurnosna komponenta hijerarhijsko drvo	- 55 -
Primer 13. Kreiranje politike sigurnosti	- 55 -
Primer 14. Kreiranje oznake sigurnosti.....	- 55 -
Primer 15. Dodeljivanje oznake.....	- 56 -
Primer 16. Izuzeci.....	- 56 -
Primer 17. Osnovno šifrovanje i dešifrovanje podataka	- 58 -
Primer 18. Kompleksno šifrovanje i dešifrovanje	- 60 -
Primer 19. Dešifrovanje binarnih podataka.....	- 61 -
Primer 20. Šifrovani indeksi	- 62 -
Primer 21. 'Pogledi' nad tabelom u cilju sakrivanja algoritma za dešifrovanje..	- 64 -
Primer 22. Primer setovanja SQLHOSTS i CONCSM.CFG za šifrovanje pristupne šifre.....	- 66 -
Primer 23. Promena šifre.....	- 68 -
Primer 24. Višeredna izmena šifre	- 68 -
Primer 25. GETHINT() funkcija.....	- 70 -
Primer 26. Određivanje veličine šifrovanog podatka upotrebom funkcije LENGTH()	- 71 -
Primer 27. Greška – mali prostor za smeštaj podatka.....	- 72 -

1

Uvod

Preduzeće Viator&Vektor Novi Sad d.o.o. iz Novog Sada se bavi poslovima posredovanja u međunarodnoj špediciji što u biti predstavlja organizovanje posla od prvobitnog upita za traženjem/manipulacijom robe, organizovanja transporta od početne do krajnje tačke i obavljanje svih formalnosti koje su potrebne tokom transporta robe.

Bitni momenat u projektovanju celog WAN sistema predstavlja geografska razuđenost preduzeća i prisutne tehničke mogućnosti povezivanja na lokacijama. Osim direkcije u Novom Sadu preduzeće ima i svoje profitne centre u Subotici, Kikindi, Zrenjaninu, Vršcu, Beogradu, Pančevu, Sremskoj Mitrovici, Šidu, Bačkoj Palanci, Vrbasu, Preševu, Kragujevcu, Požegi, Šapcu i predstavništva na graničnim prelazima Horgoš, Kelebija, Batrovci, Tovarnik, Bogojevo, Sremska Rača, Preševo i na Aerodromu Beograd.

Dodatni zahtev je takođe i potreba obezbeđivanja stalne veze sa Upravom Carina radi podnošenja elektronskih dokumenata, veza sa poslovnim bankama radi obavljanja e-bankinga i stalna veza sa odabranim ISP-om radi konstantne poslovne internet komunikacije.

Poslovna procedura koja važi za sve vrste posla u preduzeću u delokrugu delatnosti i koja predstavlja operativnu okosnicu i sve što je prati, postavlja zahteve obezbeđivanja konekcije ka serverima u direkciji 24/7 zbog dežurnih službi na graničnim lokacijama. Jednom upisani podaci o konkretnoj pošiljci (uvoz->granica), koriste se u daljim fazama obrade pošiljke na drugim geografskim lokacijama i dopunjavaju podacima karakterističnim za datu fazu.

2 Tehnologije WAN povezivanja

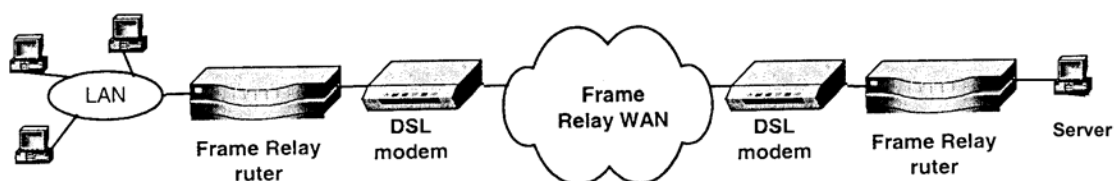
Lokacije su povezane u WAN zavisno od tehničkih mogućnosti na lokacijama.

- Primarno se sagledava mogućnost povezivanja udaljenih lokacija Frame Relay (FR) tehnologijom koja obezbeđuje visok nivo bezbednosti i sigurnosti u transportu podataka (davaoc usluga Telekom). Ovom tehnologijom je povezano 17 lokacija preduzeća i Uprava Carina. **ZA:** Veza je stabilna, bezbedna i žična. **PROTIV:** Izuzetno loš i skup tarifni sistem naplate, monopolističko ponašanje.
- Na udaljenim lokacijama gde nema tehničkih mogućnosti za FR uglavnom nema mogućnosti ni za ADSL pa se na tim lokacijama ostvaruje konekcija mobilnim APN-om (davaoc usluge Telenor). APN sam po sebi obezbeđuje dosta dobar nivo bezbednosti transporta podataka iako i pored toga za transport osetljivih podataka treba podići IPSec protokol. Ovom tehnologijom je povezano 8 lokacija. **ZA:** Dostupnost na celoj teritoriji Srbije (i u inostranstvu - Roaming). **PROTIV:** Naplata po količini prenosa u zbiru oba pravca (nije FLAT), brzina zavisi od tehničkih mogućnosti bazne stanice.
- Unutar grada Novog Sada izmeštena operativna mesta su povezana Wireless mrežom (uređaj Motorola Canopy, davaoc usluge eControl, Novi Sad) koja je po default-u obezbeđena protokolom proizvođača ali je nužno podoći IPSec protokol. Ovom tehnologijom su povezane 2 lokacije. **ZA:** Niska cena, stabilna veza. **PROTIV:** Uticaj okruženja na kvalitet prenosa.
- Poslovne potrebe moderne komunikacije zahtevaju 24/7 prisutnost na internetu za razmenu elektronske pošte i pribavljanje informacija o preduzećima sa zato sačinjenih site-ova državnih institucija. Za te potrebe je obezbeđena ADSL konekcija sa ISP-om (davaoc usluge Telekom, Neobee). **ZA:** Prihvatljiva cena. **PROTIV:** Mala upload brzina.

Na svakom kraju Frame Relay PVC-a je postavljen router proizvođača Cisco tipa 805 dok je centralni router tipa 1270. Frame Relay veza svojim konfigurisanjem predstavlja sigurni prenosni put između dva kraja PVC-a na nivou VPN zaštite bez kriptovanja.

2.1 Frame Relay

Bez obzira kako je rešena infrastruktura na lokaciji povezivanje se sprovodi na isti način. Sa svake strane veze treba da bude obezbeđen od strane korisnika FR-a Ruter koji se sa jedne strane priključuje na infrastrukturu (direktno na radnu stanicu, server, preko switch-a na LAN ...) a sa druge strane se priključuje na DSL modem koji se dobija na revers od strane Telekoma. Od DSL modema vodi veza (potrebna je jedna parica) prema Telekomu. Na korisniku je da obezbedi kućnu instalaciju od pristupnog telefonskog ormarića pa do mesta gde stoji uređaj u dotičnoj prostoriji.



Slika 1. Osnovna šema FR veze

Ostvarena veza između dve lokacije se naziva PVC i pojam PVC-a je dat u sledećoj definiciji:

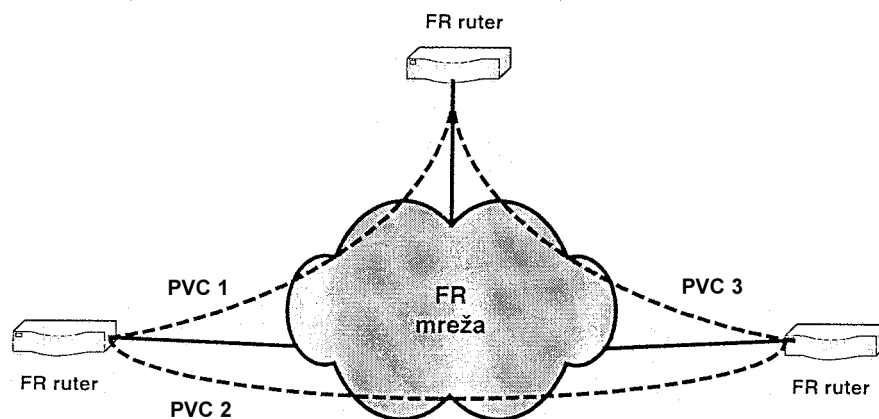
Permanentno virtuelno kolo postavlja mrežni operater (bez obzira da li se radi o privatnoj ili javnoj FR mreži) putem mrežnog sistema upravljanja. PVC je unapred definisana veza između dve strane ili dve krajnje tačke. Naknadno se može zahtevati proširenje propusnog opsega PVC-a ili uspostavljanje veze sa nekom drugom stranom. Permanentno virtuelno kolo je fiksni put. Iako se s vremena na vreme određene putanje kroz mrežu mogu menjati (recimo ukoliko se neki od linkova nađe u prekidu, a koristi se automatsko prerutiranje), početak i kraj veze se ne menjaju. Zbog toga ono podseća na iznajmljenu liniju sa kraja na kraj. PVC veza je popularna, jer predstavlja jeftinu i efektivnu alternativu iznajmljenim linijama. Nadgledanje PVC-a podrazumeva planiranje protoka, uz poznavanje parametara saobraćaja i iskorišćenja propusnog opsega. PVC veze nisu fleksibilne u slučajevima kada je potrebno uspostaviti vezu na kratak vremenski period, jer je za uspostavljanje PVC veza potrebno neko vreme (nekoliko radnih dana).

*Svaka PVC veza definiše se za samo jednog korisnika. Nije moguće pozajmljivati kapacitete drugih PVC veza (kada ove ne prenose saobraćaj) ili primati informacije iz drugih PVC veza. Na taj način se obezbeđuje integritet prenesenih podataka.*¹

Kod projektovanja potrebne linije potrebno je odrediti potrebnu brzinu protoka podataka pošto na osnovu toga određujemo parametre na osnovu kojih se

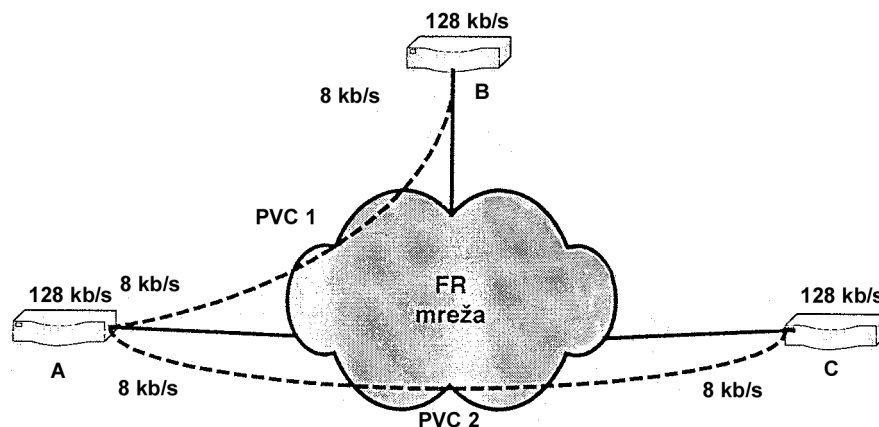
¹ Skripta „Javne mreže za prenos podataka“, FTN Novi Sad 2003, Dejan Nemec dipl. inž, Čedomir Stefanović dipl. inž., Dejan Vukobratović dipl. inž.

iznajmljuje dotični PVC. Karakteristike koje se pominju su CIR (garantovan protok od strane „Telekoma“) i brzina FR Porta na lokaciji. Najčešće se uzima za početak najniži CIR od 8 Kb/s i brzina porta od 128 Kb/s koja je naročito dovoljna za dislociranog korisnika. Ako se odmah u početku pušta četiri (4) i više konekcija koje koje „gledaju“ ka portu gde se nalazi baza podataka, tada je potrebno na portu gde se nalazi baza (čvorište), obezbediti brzinu od 2 Mb/s koje pokriva protok do šesnaest (16) navedenih konekcija. Ukoliko FR pravac nije opterećen, stvarni protok prelazi CIR-om definisan i biva ograničen zakupljenim protokom porta.



Slika 2. Metod "svaki sa svakim"

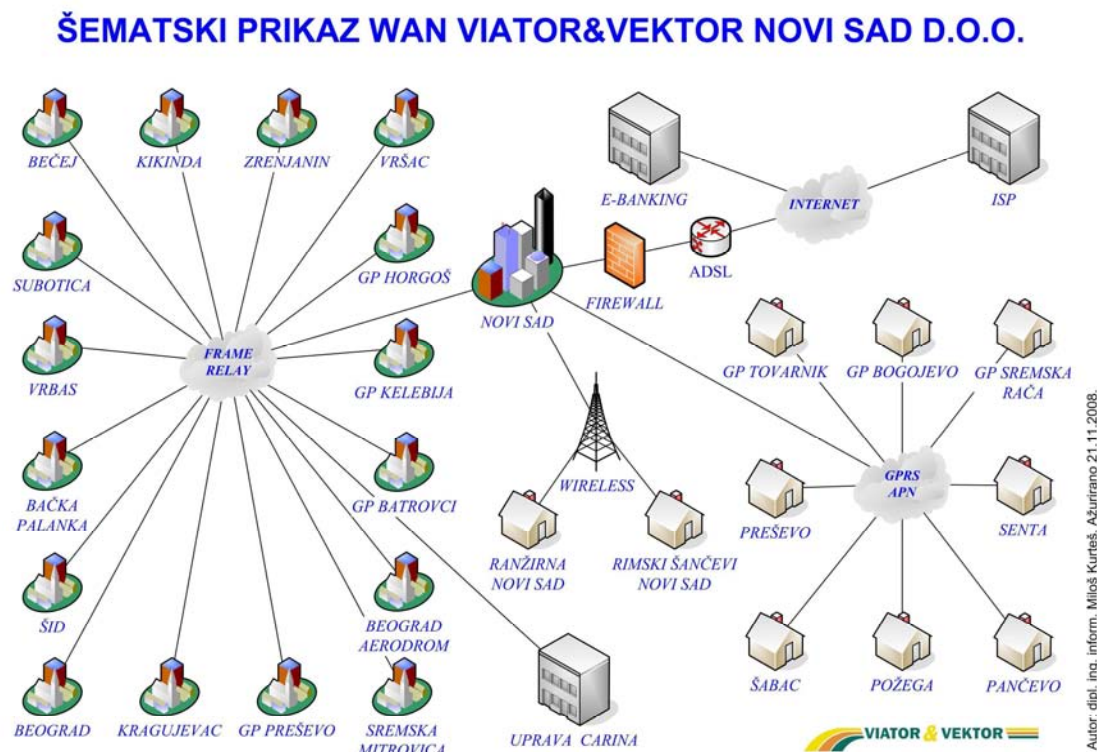
Kao što je prikazano na slici 2., povezivanje može da se ostvari u varijanti „svaki sa svakim“ što najčešće nije slučaj nego se povezuje u „zvezdu“ gde se obezbeđuje samo jedan port sa većim protokom, što je i finansijski isplativije.



Slika 3. Metod „zvezda“

Na slici 3. se vidi metod „zvezda“. Kod zakupa PVCa za mesečnu nadoknadu se plaća po tarifi Telekoma za definisani protok porta i dupli iznos za definisani CIR (sa svake strane PVC-a).

Kao što može da se vidi na sledećoj slici 4., poslovna WAN mreža preduzeća koje se posmatra predstavlja metod povezivanja tipa „zvezda“.



Slika 4. Šematski prikaz WAN Viator&Vektor Novi Sad d.o.o.

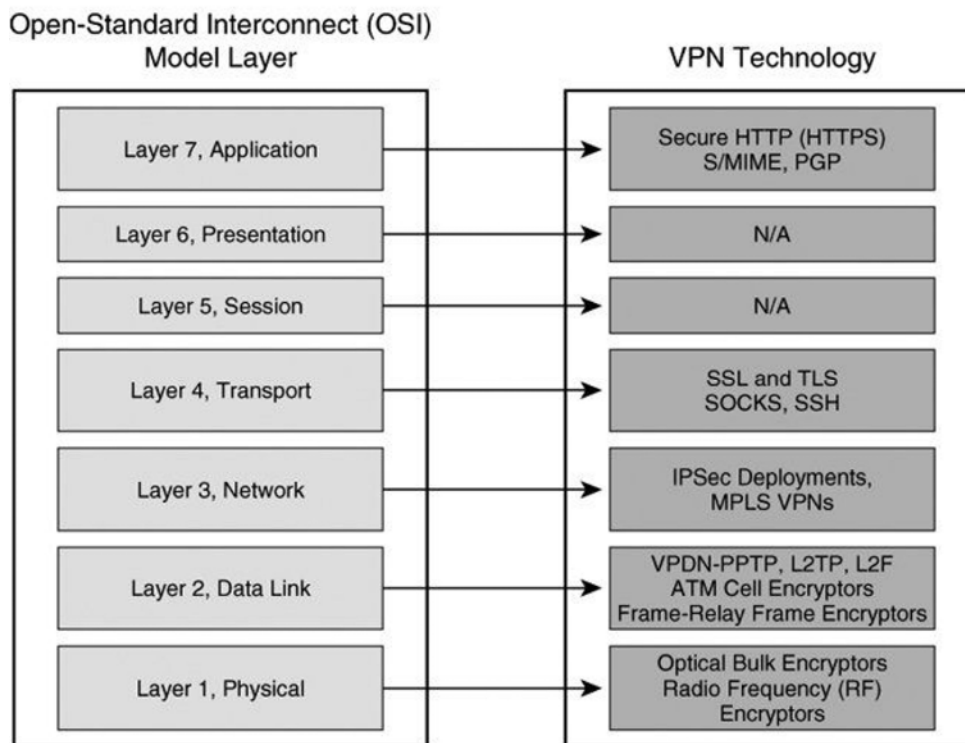
Na slici 4. uočavamo korišćenje više tehnologija pri prenosu podataka. Frame Relay je okosnica poslovne WAN mreže i već sam po sebi kao protokol drugog sloja OSI modela predstavlja po svojoj stukturi oblik tunelovanja i u praksi se ne koristi nikakav dalji metod zaštite podataka.

Ispitivanja radi, sagledaćemo IPsec mogućnosti Cisco router-a (treći sloj OSI modela) koje posedujemo sa obe strane svakog PVC-a unutar FR segmenta poslovne mreže i komentarisati dobijene rezultate.

WWAN (Wireless WAN) segment je obezbeđen u osnovi ugrađenim algoritmima u Canopy Motorola uređaje kojima se vrši primopredaja signala i tada VPN tunelom koji se sa jedne strane diže na Linux server mašini a sa druge na Windows XP klijentu.

APN (mobilna mreža) segment je takođe obezbeđen svojim osnovnim servisima ali i poder toga se koristi VPN (OpenVPN) za tuneliranje i na taj način se obezbeđuje uz kompresiju veća propusna moć takve veze i prisustvo u NAT-u firme. VPN tunel je otvoren sa centralne lokacije na Linux server mašini a sa druge na Windows XP klijentu.

Slika 5. prikazuje na kom mestu se nalaze odgovarajuće VPN tehnologije posmatrajući ih u odnosu na OSI referentni model



Slika 5. Položaj VPN tehnologije u odnosu na OSI model

3 Metode istraživanja

„Najjednostavnija je definicija „metodologije“ (grčki „methodos + logos“ = reč, govor, nauka o metodama znanstvenog istraživanja) da je to znanost o metodama znanstvenog istraživanja. U širem smislu metodologija je nauka o celokupnosti svih oblika i načina istraživanja pomoću kojih se dolazi do sistemskog i objektivnog naučnog znanja, ili naučna disciplina u kojoj se kritički ispituju i eksplicitno izlažu različite opšte i posebne naučne metode, naziva se metodologija „²

U ovom radu, da bi se postigli postavljeni ciljevi kao i što efikasniji rezultat koji se dobijaju povezivanjem i odgovarajućom kombinacijom više naučnih metoda, korištene su sledeće metode:

- metoda deskripcije,
- komparativna metoda,
- matematička metoda,
- metoda analize,
- metoda sinteze

Metoda deskripcije, definisana kao postupak jednostavnog opisivanja ili ocrtaavanja činjenica, procesa i predmeta, te njihovih empiriskih potvrđivanja odnosa i veza, ali bez naučnog tumačenja i objašnjavanja, je korištena kroz celi rad.

Komparativna metoda, koja podrazumeva upoređivanje istih ili srodnih činjenica, pojava, procesa i odnosa, odnosno utvrđivanje njihove sličnosti u ponašanju i intenzitetu kao i razlika među njima.

Matematička metoda, je naučni sistemski postupak, a sastoji se u primeni matematičke logike, simbola, brojnih matematičkih operacija. Kratko rečeno predstavlja matematički način zaključivanja.

Metoda analize, je postupak naučnog istraživanja i objašnjavanja stvarnosti putem raščlanjivanja složenih misaonih tvorevina (pojmovi, sudovi i zaključci) na njihove jednostavnije delove i elemente, kao i izračunavanje svakog dela (elementa) za sebe i u odnosu na druge delove, celine.

Metoda sinteze, je suprotna metoda od analize koja se takođe primjenjivala u izradi ovog rada.

² Prof.dr.Čekić Š.: „Osnovi metodologije i tehnologije izrade znanstvenog i stručnog djela“, Sarajevo 1999., str.33

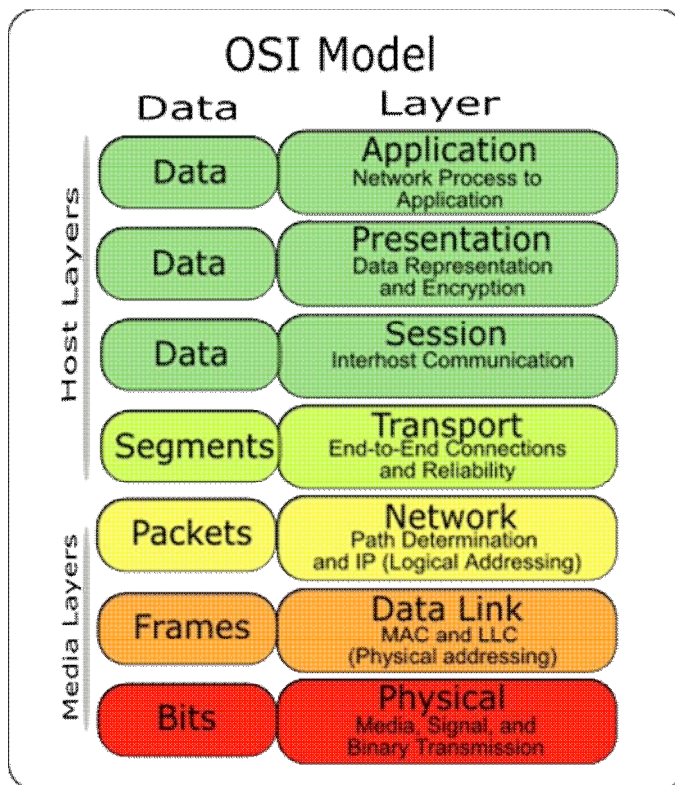
4 Uvod u IPsec

Većina računarskih mreža, uključujući i Internet kao globalnu svetsku mrežu, za komunikaciju koristi TCP/IP skup protokola. U osnovi se radi o dve vrste protokola, TCP protokolu, zaduženom za prenos podataka i IP protokolu, koji ima osnovnu ulogu usmeravanja paketa podataka kroz mrežu od izvorišnog do odredišnog računara. IP skup protokola pokazuje dobra komunikaciona svojstva u smislu skalabilnosti, prilagodljivosti i otvorenosti prema različitim računarskim arhitekturama i mrežnoj opremi, te je iz tog razloga široko rasprostranjen komunikacioni protokol.

Najveći nedostatak IP protokola je što ne pruža podršku za očuvanje tajnosti, autentičnosti i verodostojnosti prenesenih podataka. Prenos osetljivih podataka u IP mrežama potrebno je osigurati na višim nivoima komunikacije, u transportnim i aplikacijskim slojevima. Primjer sigurnosnog mehanizma koji deluje na nivou transportnog sloja je SSL protokol. Nedostatak takvog pristupa je što postoji veliki broj različitih protokola aplikacionog nivoa za koje je potrebno posebno razvijati sigurnosne mehanizme.

IPsec protokol (*IP Security protocol*) nastao je kao rezultat inicijative da se razvije jedinstveni sigurnosni mehanizam za zaštićeni prenos podataka u IP mrežama. Razvijen je od strane Internet Engineering Task Force (IETF) grupe i predstavlja proširenje osnovnog IP protokola.

IPsec protokol sastavni je deo mrežnog sloja OSI modela koji je referentni sistem za povezivanje. Osnovna osobina ovog modela je slojevitost gde su svi poslovi vezani za proces komuniciranja podeljeni u sedam (7) nivoa. Svaki nivo je nezavisan od ostalih nivoa u smislu da promena u jednom ne utiče na ostale nivoe.



Slika 6. OSI referentni model

1. Fizički sloj (*Physical*) je zadužen za definisanje električnih, mehaničkih i drugih specifikacija vezanih za vezu između dva mrežna uređaja. Odgovoran je za aktiviranje, održavanje i deaktiviranje veze između krajnjih tačaka linka i u slučaju prekida linka da o tome obavesti drugi nivo.
2. Sloj veze (*Data link*) definiše adresiranje fizičkog sloja, topologiju, brine o detekciji i otklanjanju grešaka na fizičkom nivou.
3. Sloj mreže (*Network*) obezbeđuje rutiranje i funkcije potrebne za uspostavljanje i raskidanje veze. U IP mrežama ovaj se sloj naziva IP sloj, a usmjeravanje paketa kroz mrežu obavlja se prema IP protokolu.
4. Transpotrni sloj (*Transport*) obezbeđuje transparentni transfer podataka od izvora do odredišta i zadužen je da se primljeni podaci proslede u istom stanju ka destinaciji.
5. Sloj sesije služi za organizovanje i sinhronizaciju razmene podataka između aplikacija.
6. Sloj prezentacije (*Prezentation*) prevodi podatke iz aplikativnog oblika u zajednički oblik uobičajen transport podataka preko mreže i obrnuto kada ti podaci pristižu u obrnutom pravcu.
7. Sloj aplikacije (*Application*) predstavlja interfejs prema krajnjem korisniku. Na ovom sloju se započinju i završavaju zahtevi.

IPsec protokol zadržava kompatibilnost s postojećim IP protokolom i omogućava mrežnoj opremi zasnovanoj na IP protokolu transparentnost u odnosu na IPsec protokol. Samo dva krajnja učesnika u komunikaciji, pošiljaoc i primaoc, moraju imati podršku za komunikaciju IPsec protokolom.

Mreža čvorišnih računara i rutera između krajnjih učesnika, zbog kompatibilnosti IPsec protokola s IP protokolom, ne mora biti opremljena programskom podrškom za IPsec protokol. Na slici 7. prikazana je struktura TCP/IP paketa podataka za komunikaciju preko Ethernet mreže.

Ethernet zaglavlje	IP zaglavlje	TCP zaglavlje	Podaci	Ethernet zaštita
--------------------	--------------	---------------	--------	------------------

Slika 7. Struktura IP paketa za prenos preko Ethernet mreže

Ethernet zaglavlje i Ethernet zaštitu koristi fizički sloj i sloj veze. Za pravilan rad mrežnog sloja i IP protokola bitno je IP zaglavlje. U IP zaglavlju su, između ostalog, upisane izvorišna i odredišna adresa IP paketa koje su potrebne za usmeravanje paketa kroz mrežu. Ostali delovi TCP/IP paketa, TCP zaglavlje i podaci, pripadaju višim slojevima i za IP sloj oni predstavljaju obične podatke koje je potrebno preneti kroz mrežu.

Da bi se zadržala kompatibilnost IPsec protokola s IP protokolom nužno je da u strukturi IPsec paketa podataka ostane očuvano IP zaglavlje. IPsec protokol stoga obavlja kriptografske akcije nad zaglavljem i podacima viših slojeva. Struktura IPsec paketa podataka za komunikaciju Ethernet mrežom prikazana je na slici 8. Polje s IPsec zaglavljem i podacima uključuje TCP zaglavlje, te sva ostala zaglavlja i podatke viših slojeva u kriptovanom obliku.

Ethernet zaglavlje	IP zaglavlje	IPsec zaglavlja i podaci	Ethernet zaštita
--------------------	--------------	--------------------------	------------------

Slika 8. Struktura IPsec paketa podataka za prenos preko Ethernet mreže

Kao što je već pomenuto internet protokol, IP, poznat kao IPv4, ne obezbeđuje bilo kakvu zaštitu tokom prenosa podataka. Čak i ne garantuje da je pošiljaoc autentičan. IPsec pokušava da to ispravi i da obezbedi sledeće osobine:

- **Tajnost**

Sigurnost da niko osim primaoca ne razumije poslane podatke.

- **Integritet**

Sigurnost da se podaci ne menjaju tokom transporta.

- **Autentifikacija**

Potpisivanjem podataka primaoc je siguran da je pošiljaoc verodostojan.

- **Neporecivost (zaštita protiv lažnog odgovora)**

Potrebni su nam načini da osiguramo da se transakcija može izvršiti samo jednom, izuzev ako ne postoji ovlaštenje da se ponovi

IPsec obezbeđuje tajnost, integritet, autentifikaciju i neporecivost kroz protokole AH (*Authentication header*), ESP (*Encapsulated security payload*) i IKE (*Internet Key Exchange*).

IPsec protokoli se razlikuju po strukturi i nameni IPsec zaglavlja i podataka. Razlikuju se tri vrste IPsec zaglavlja i podataka: zaglavlje za autentifikaciju, korisni kriptovani podaci i zaglavlje za razmenu ključeva.

IPsec protokoli omogućavaju stvaranje strukture sigurne virtualne privatne mreže (*Virtual Private Network - VPN*) izgrađene na infrastrukturi postojeće javne IP mreže, odnosno Interneta, u kojoj nisu zadovoljeni zahtevi u pogledu sigurnosti komunikacije.

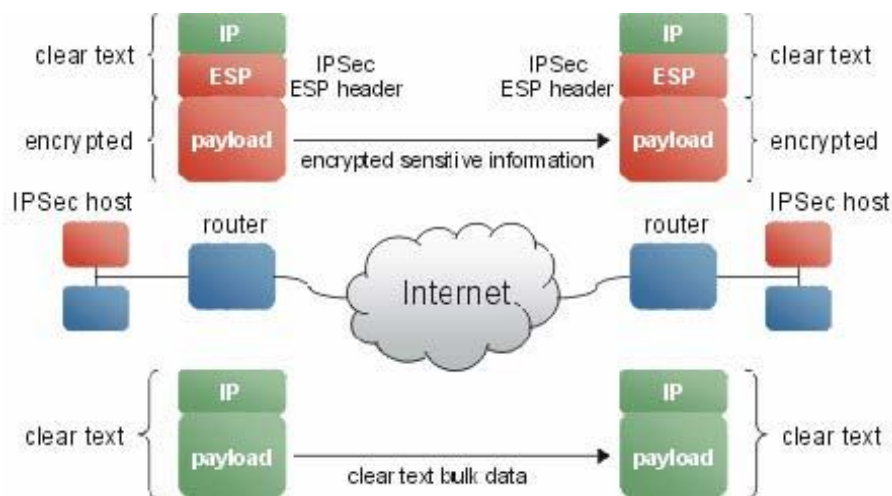
Za osiguravanje tajnosti, autentičnosti, verodostojnosti i neporecivosti prenesenih podataka, IPsec protokol koristi navedena tri protokola koji zajednički predstavljaju zaštitu od svih trenutno poznatih sigurnosnih napada.

5 Dizajn sistema

IPsec je dizajniran tako da zadovolji dva osnovna zadatka:

- tuneliranje paketa i
- transportni način rada.

U prvom slučaju nekoliko računara (ili cela jedna lokalna mreža) sakriva se iza jednog čvorišta te je kao takva nevidljiva ostatku mreže (a samim tim i zaštićena od napada). U drugom slučaju paketi se šalju između dva krajnja računara na mreži, pri čemu računar koje prima paket izvršava sigurnosne provjere pre isporučivanja paketa višim slojevima. U oba slučaja moguće je izgraditi Virtualne Privatne Mreže – VPN (*Virtual Private Network*), što je i osnovna ideja zaštite IPsec protokolima.



Slika 9. Prikaz sistema baziranog na IPsec protokolu

Pošto IP protokol osigurava uslugu komunikacionog kanala od kraja do kraja (*end-to-end*), zaštita kanala na istom nivou korištenjem IPsec-a omogućava mu nezavisnost obzirom na niže slojeve. To znači da komunikacioni uređaji na putu između dva entiteta ne moraju podržavati IPsec, što omogućava korištenje IPsec-a bez obzira na način implementacije fizičkog sloja i sloja veze.

S druge strane, ukoliko dva krajnja entiteta podržavaju IPsec, njegova upotreba je transparentna obzirom na više slojeve protokolnog steka. Aplikacije mogu koristiti sigurnu komunikaciju koju pruža IPsec, bez obzira na vlastitu funkcionalnost. Isto se odnosi i na protokole koji su implementirani u transportnom

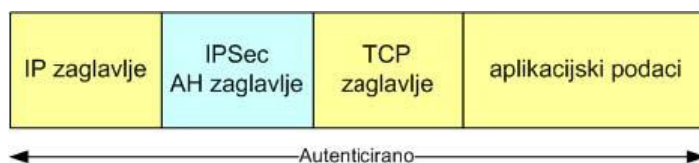
sloju, što znači da svi podaci koji se prenose korištenjem TCP i UDP protokola, isto kao i ICMP poruke, mogu koristiti sigurni komunikacioni kanal koji pruža IPsec.

Jednako kao i sigurnosni protokoli i načini rada su parametar politike koji se definišu o jezgru operativnog sistema (uspješna komunikacija će se upostaviti samo ako oba učesnika u komunikaciji imaju isti način rada).

5.1 Transportni način rada

Transportni način rada namenjen je prvenstveno za uspostavljanje sigurne komunikacije između entiteta, odnosno tzv. host-to-host komunikacije u privatnim LAN ili WAN računarnim mrežama. Za transportni način rada nužno je da obe krajnje tačke (izvor i odredište) podržavaju IPsec. Korištenjem AH i ESP protokola moguće je postići različite aspekte sigurne komunikacije.

AH – Ukoliko se u transportnom načinu rada koristi AH protokol, moguće je osigurati integritet, autentifikaciju i neporecivost, a AH zaglavlje se dodaje odmah iza IP zaglavlja (slika 10). U tom slučaju polje protokol u IP zaglavlju sadrži vrednost 51 (AH), dok polje sledeće zaglavlje u AH zaglavlju sadrži vrednost koja odgovara enkapsuliranom datagramu iz višeg sloja (npr. 6 za TCP datagram).



Slika 10. AH u transportnom načinu rada

Kako se iz slike može videti, AH u transportnom načinu rada provodi autentifikaciju, te osigurava integritet i neporecivost čitavog IP datagrama.

ESP – U transportnom načinu rada ESP osigurava integritet, autentifikaciju, neporecivost i tajnost podataka koji se prenose. Ukoliko se za IPsec koristi ESP, polje protokol u IP zaglavlju sadržaće vrednost 50 (ESP), a polje sledeće zaglavlje u ESP zaglavlju sadržaće vrednost koja odgovara enkapsuliranim podacima iz višeg sloja, isto kao i kod AH. Iza enkapsuliranih podataka ESP takođe dodaje ispunu, te opciono (ukoliko je u SA skupu sigurnosnih parametara navedena i autentifikacija) polje autentifikacijski podaci. Slika 11 Prikazuje ESP u transportnom načinu rada.



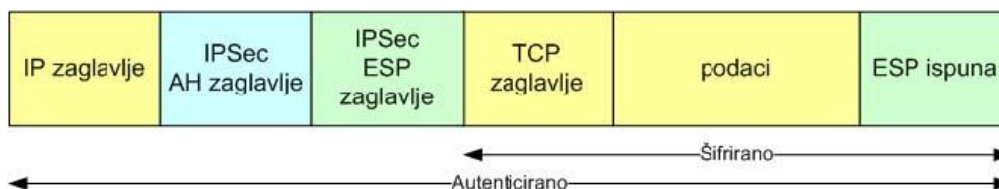
Slika 11. ESP u transportnom načinu rada

Kako se iz slike može videti, svi podaci višeg sloja, zajedno s ESP ispunom, su šifrirani. Također je uočljivo da, za razliku od AH koji autentifikuje čitav IP datagram, uključujući i IP zaglavlje, ESP autentifikuje vlastito zaglavlje i deo datagrama sa podacima, čime je teoretski ostavljena mogućnost neovlaštene modifikacije IP zaglavlja.

ESP + AH – Iz prethodnih razmatranja vidljivo je da AH osigurava integritet, autentifikaciju i neporecivost čitavog IP datagrama. Istovremeno, ESP može osigurati tajnost dela datagrama sa podacima, te opciono integritet, autentifikaciju i neporecivost za taj deo datagrama (uključujući i ESP zaglavlje) ali ne i za pripadajuće IP zaglavlje.

Ukoliko se želi postići maksimalni nivo zaštite, odnosno osigurati tajnost podataka i autentifikacija, integritet i neporecivost čitavog IP datagrama, moguće je koristiti ESP i AH zajedno. U tom slučaju polje protokol u IP zaglavlju sadržaće vrednost 51 (AH), nakon toga slediće AH zaglavlje, čije će polje sledeće zaglavlje sadržati vrednost 50 (ESP), nakon kojeg će slediti ESP zaglavlje u čijem će polju sledeće zaglavlje biti sadržana vrednost koja će označavati protokol višeg sloja čiji podaci su enkapsulirani u tako formiranom datagramu.

Slika 12 prikazuje IPsec u transportnom načinu rada ukoliko se istovremeno koriste ESP i AH. Ovde valja napomenuti da se prvo formira ESP deo paketa, odnosno šifrira se datagram transportnog sloja, te se formira pripadajuće ESP zaglavlje. Nakon toga se računa vrednost pripadajućeg AH zaglavlja i formira isto. U ovom slučaju ESP deo paketa ne sadrži opciono polje autentifikacioni podaci, pošto autentifikaciju, integritet i neporecivost osigurava AH.

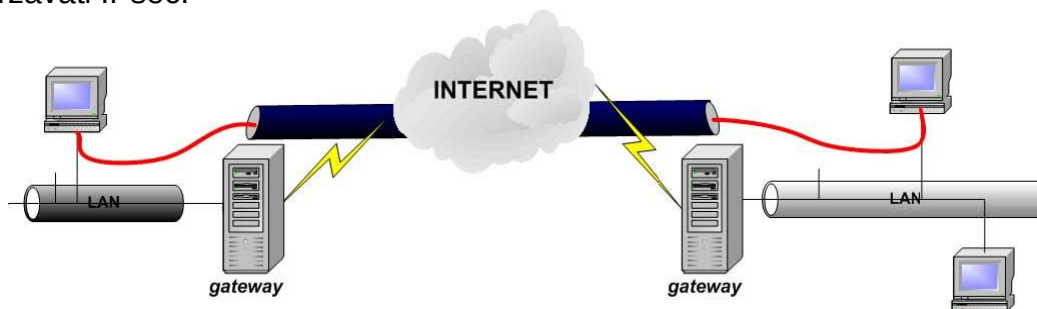


Slika 12. ESP + AH u transportnom načinu rada

5.2 Tuneliranje

Tuneliranje je drugi način rada ili druga funkcionalnost IPsec protokola. U tom načinu rada IPsec služi za uspostavljanje sigurne komunikacije između gateway uređaja na udaljenim mrežama (*gateway-to-gateway*), osiguravajući tako virtualnu privatnu komunikaciju, odnosno uspostavljajući VPN (*Virtual Private Network*) spoj između udaljenih lokacija. U ovom slučaju krajnji entiteti u komunikaciji ne moraju podržavati IPsec; čitava komunikacija za njih je potpuno transparentna jer sve operacije nužne za sigurnu komunikaciju korištenjem IPsec-a obavljaju gateway uređaji.

Gateway uređaji na udaljenim mrežama predstavljaju ulaznu, odnosno izlaznu tačku sigurnog komunikacionog kanala. Oni preko nesigurnog medija (Internet) formiraju sigurni tunel, zbog čega se ovaj način rada i zove tuneliranje (slika 13). Korištenje tunelskog načina rada također je moguće i u host-to-host ili host-to-gateway komunikaciji, no tada ponovno krajnji entiteti ili entitet moraju podržavati IPsec.



Slika 13. Tuneliranje

Za razliku od transportnog načina rada gdje se AH odnosno ESP zaglavlja dodaju unutar postojećeg IP datagrama, kod tuneliranja se formira potpuno novi IP datagram koji enkapsulira kompletni originalni IP datagram. U načelu komunikacija između dva entiteta funkcioniše na sledeći način:

1. Izvorni računar formira IP datagram i šalje ga preko lokalne mreže lokalnom gateway uređaju.
2. Gateway uređaj enkapsulira čitav originalni IP datagram u novi datagram te formira odgovarajuća AH odnosno ESP zaglavlja.
3. Tako formirani datagram se šalje preko uspostavljenog tunela do gateway uređaja na udaljenoj mreži koji uklanja dodatna zaglavlja, te po potrebi vrši dešifrovanje i proveru integriteta paketa.
4. Nakon toga originalni IP datagram se isporučuje ciljnom računaru.

Isto kao i u transportnom načinu rada, moguća je implementacija korištenjem AH i ESP.

AH – Ukoliko se želi osigurati samo integritet, autentifikacija i neporecivost poruka, a tajnost nije nužna, moguće je koristiti AH protokol. U tom slučaju originalni IP datagram, koji sadrži adresu krajnjeg odredišta, se enkapsulira u novi IP datagram kojem se dodaje odgovarajuće AH zaglavlje (slika 14). U ovom slučaju polje protokol novog IP zaglavlja koje sadrži adresu krajnje tačke IPsec tunela ima vrednost 51 (AH), dok polje sledeće zaglavlje unutar AH zaglavlja ima vrednost 4 (enkapsulirani IP).



Slika 14. Tuneliranje korištenjem AH

ESP – Ukoliko se osim autentifikacije, integriteta i neporecivosti želi osigurati i tajnost komunikacije, nužna je upotreba ESP protokola. Korištenjem ESP u tunelskom načinu rada, za razliku od transportnog načina, vrši se šifrovanje čitavog originalnog IP datagrama, a takođe je osigurana autentifikacija, integritet i neporecivost čitavog datagrama, pošto je sam datagram enkapsuliran u novi IP paket. U ovom slučaju polje protokol novog IP zaglavlja koje, isto kao i kod AH, sadrži adresu krajnje tačke IPsec tunela ima vrednost 50 (ESP), dok polje sledeće zaglavlje unutar ESP zaglavlja ima vrednost 4 (enkapsulirani IP).

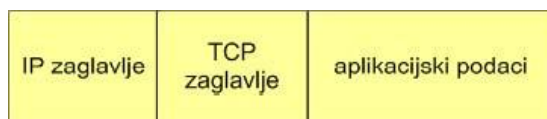


Slika 15. Tuneliranje korištenjem ESP

Kombinacija AH i ESP u tunelskom načinu rada nije predviđena.

6 Protokoli implementacije

IPsec se implementira korištenjem dva međusobno nezavisna protokola koji osiguravaju različite aspekte sigurnosti. AH (*authentication header*) osigurava integritet, autentifikaciju, neporecivost, dok ESP (*encapsulated security payload*) osim toga može osigurati i tajnost podataka koji se prenose. Oba protokola, AH i ESP, modifikuju standardni oblik IP datagrama (slika 16). U nastavku ovog poglavlja biće opisana oba protokola i njihova funkcionalnost.



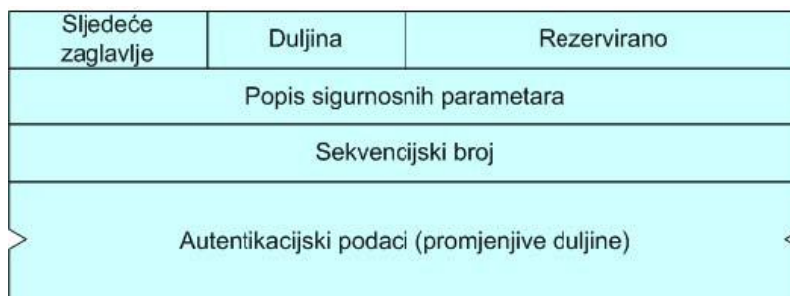
Slika 16. Standardni oblik IP datagrama

6.1 AH (Authentication Header) protokol

Zaglavlje za autentifikaciju (*Authentication Header - AH*) koristi se za proveru identiteta pošiljaoca podataka i otkrivanje namernih izmena nad podacima tokom prenosa mrežom. Izvodi se metodama digitalnog potpisivanja podataka. Zaglavlje za autentifikaciju ne čuva tajnost podataka i koristi se samo u slučajevima kada je bitna autentičnost i verodostojnost podataka, a ne i njihova tajnost.

AH protokol (IP protokol 51) osigurava autentifikaciju, integritet i neporecivost IP datagrama, ali ne može osigurati i tajnost. Protokolom je definisano vlastito (AH) zaglavlje koje se umeće između IP zaglavlja i IP podataka koji slede (slika 17).

Specifičnost AH jeste u tome što on, za razliku od ostalih protokola TCP/IP steka, ne enkapsulira podatke protokola kojima pruža uslugu. Slika 17. prikazuje AH zaglavlje, zajedno s pripadajućim poljima. Sva prikazana polja su obavezna, odnosno uvek su prisutna u AH zaglavlju. U nastavku su opisane njihove funkcije.



Slika 17. AH zaglavlje

Sljedeće zaglavlje (*next header*) – Sljedeće zaglavlje je 8-bitno polje koje identifikuje tip podataka koji sledi nakon AH zaglavlja. Polje može poprimiti vrednost iz definisanog skupa brojeva koji označavaju IP protokole (npr. 6 – TCP, 17 – UDP, 51 – ESP). U dokumentu RFC 3232, odnosno online bazi podataka (<http://www.iana.org>), dat je trenutno važeći skup brojeva, odnosno protokola.

Dužina (*payload length*) – Dužina je polje koje određuje dužinu AH zaglavlja. Dužina se računa kao dužina u 32-bitnim rečima umanjena za vrednost 2.

Rezervisano (*reserved*) – Ovo polje dužine 16 bita je rezervisano za buduće potrebe. Ono mora biti postavljeno na vrednost "0".

Popis sigurnosnih parametara (*security parameters index*) – Ovo polje dužine 32 bita sadrži proizvoljnu vrednost, koja uz IP adresu i sigurnosni protokol (u ovom slučaju AH) definiše jedinstveni skup sigurnosnih parametara (*security association* – SA) koji se koristi u sigurnoj komunikaciji između dva entiteta. SA skup sigurnosnih parametara definiše se prilikom uspostavljanja IPsec spoja. Vrednosti od 1 do 255 rezervisane su od IANA-e za buduću upotrebu.

Sekvencijski broj (*sequence number*) – Ovo polje dužine 32 bita služi za osiguranje od napada ponavljanjem paketa, a povećava se prilikom svakog slanja paketa koji ima identični SA skup sigurnosnih parametara. Pošiljaoc mora nužno generisati ovo polje, dok ga primaoc može ili ne mora interpretirati. Prilikom inicijacije komunikacije ovo polje se postavlja na vrednost "1".

Autentifikacijski podaci (*authentication data*) – Polje koje sadrži autentifikacijske podatke je varijabilne dužine. U njemu je sadržana ICV (*integrity check value*) vrednost na temelju koje se proverava integritet i autentičnost poruke. Dužina polja za autentifikacijske podatke mora biti celobrojni višekratnih 32-bitne reči. Ukoliko polje samo po sebi ne ispunjava taj uslov, dodaje se (proizvoljna) ispuna kojom se nadopunjava odgovarajući broj bitova.

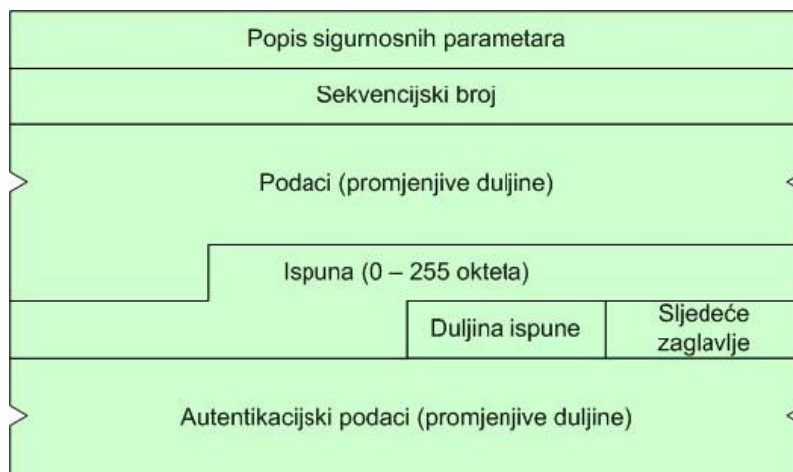
Vrednost ICV-a se računa na temelju svih polja IP zaglavlja koja se ne menjaju prilikom prenosa, čitavog AH zaglavlja (koje je za tu potrebu postavljeno na vrednost "0"), te svih podataka protokola višeg sloja. Algoritam koji se upotrebljava za računanje ICV-a, koji može biti autentifikacijski kod poruke (*message authentication code*) izračunat korištenjem simetričnih algoritama za

šifriranje (npr. DES) ili rezultat hash funkcija (npr. MD5 ili SHA-1), definiše se prilikom uspostavljanja komunikacije i deo je SA skupa sigurnosnih parametara.

6.2 ESP (Encapsulated Security Payload) protokol

IPsec paket u obliku korisnih kriptovanih podataka (*Encapsulating Security Payload ESP*) koristi se u slučajevima kada je bitno očuvati tajnost prenesenih podataka. Uz tajnost, ESP polje IPsec paketa čuva autentičnost i verodostojnost podataka. Za stvaranje ESP polja podataka koriste se metode jake kriptografije u sprezi s metodama digitalnog potpisivanja podataka.

ESP protokol (IP protokol 50) može osigurati autentifikaciju, integritet, neporecivost i tajnost podataka. Protokol takođe definiše sopstveno zaglavlje koje se umeće iza IP zaglavlja, te enkapsulira sve podatke protokola višeg sloja, dodajući pri tom završni slog u kojem mogu biti sadržani autentifikacijski podaci. Slika 18 prikazuje ESP datagram zajedno s pripadajućim poljima. U nastavku su opisane funkcije tih polja.



Slika 18. ESP datagram

Popis sigurnosnih parametara (*security parameters index*) – Popis sigurnosnih parametara je 32-bitno polje u kojem se, isto kao i kod AH, definiše jedinstveni SA skup sigurnosnih parametara (određen prilikom uspostavljanja komunikacije) koji se koristi u komunikaciji između dva entiteta. Kao i kod AH, vrednosti od 1 do 255 su rezervisane za buduću upotrebu.

Sekvencijski broj (*sequence number*) – Ovo polje dužine 32 bita, isto kao i kod AH, služi za osiguranje od napada ponavljanjem paketa, a povećava se prilikom svakog slanja paketa koji ima identični SA skup sigurnosnih parametara. Pošiljaoc mora nužno generisati ovo polje, dok ga primaoc može ili ne mora interpretirati. Prilikom inicijacije komunikacije ovo polje se postavlja na vrednost "0", za razliku od AH gdje je inicijalna vrednost tog polja "1".

Podaci i ispuna (*payload data*) – Ovo polje proizvoljne dužine sadrži deo IP paketa sa podacima i ispunu. Vrsta podataka koja se nalazi u delu za podatke definisana je poljem sledeće zaglavlje. Osim samih podataka, u tom polju mogu biti i eksplicitno sadržani podaci koji su nužni za kriptografsku sinhronizaciju (npr. inicijalizacijski vektor – IV), ukoliko to kriptografski algoritam koji se koristi zahteva (npr. DES u CBC načinu rada). Zavisno od načina rada kriptografskih protokola koji koriste IV inicijalizacijski vektor, on može biti sadržan na samom početku šifrovanog bloka podataka ili zasebno od šifrovanih podataka, što ovisi o konkretnim implementacijama algoritama. Ispuna se koristi iz dva razloga:

- o neki kriptografski algoritmi za šifrovanje koriste blokove fiksne dužine, te je deo paketa sa podacima potrebno dopuniti do odgovarajuće dužine,
- o zbog implementacionih razloga nužno je da dužina podataka ispune, te dva sledeća polja (dužina ispune i sledeće zaglavlje) zajedno daju celobrojni sadržalac 32-bitne reči, odnosno da je ta dužina poravnata na 4-okteta.

Dužina ispune (*payload length*) – Ovo 8-bitno polje definiše dužinu prethodno korištene ispune u oktetima. Dozvoljene vrednosti su od 0 do 255, s time da vrednost "0" označava da ispuna ne postoji.

Sledeće zaglavlje (*next header*) – Sledeće zaglavlje, ponovno kao i kod AH, je 8-bitno polje koje identifikuje tip podataka koji sledi nakon ESP zaglavlja. Polje može poprimiti vrednost iz definisanog skupa brojeva koji označavaju IP protokole.

Autentifikacijski podaci (*authentication data*) – Ovo polje proizvoljne dužine nije obavezno, a koristi se samo u slučaju da je u SA skupu sigurnosnih parametara određena usluga autentifikacije. U tom slučaju ovo polje sadrži ICV koji se računa za celi ESP datagram (ESP zaglavlje, deo za podatke i ispuna), ne uključujući pri tom samo polje namenjeno autentifikacijskim podacima, a njegova dužina zavisi od autentifikacijskog algoritma koji se koristi.

7

Uspostavljanje IPsec komunikacije

Korištenjem IPsec-a, odnosno AH i/ili ESP protokola, osigurava se integritet, autentifikacija, neporecivost i tajnost, no unutar IPsec-a nije implementiran nikakav mehanizam koji bi služio za uspostavljanje komunikacije, odnosno specifikaciju kriptografskih algoritama i funkcija koje će se koristiti u IPsec komunikaciji.

Za korištenje bilo kakvih kriptografskih metoda nužno je da entiteti u komunikaciji dogovore skup sigurnosnih parametara komunikacije (*Security Association – SA*). Taj skup sigurnosnih parametara uključuju dogovor kriptografskih metoda koje će se koristiti, način autentifikacije strana u komunikaciji, te razmenu kriptografskih ključeva nužnih za tako dogovorenu komunikaciju.

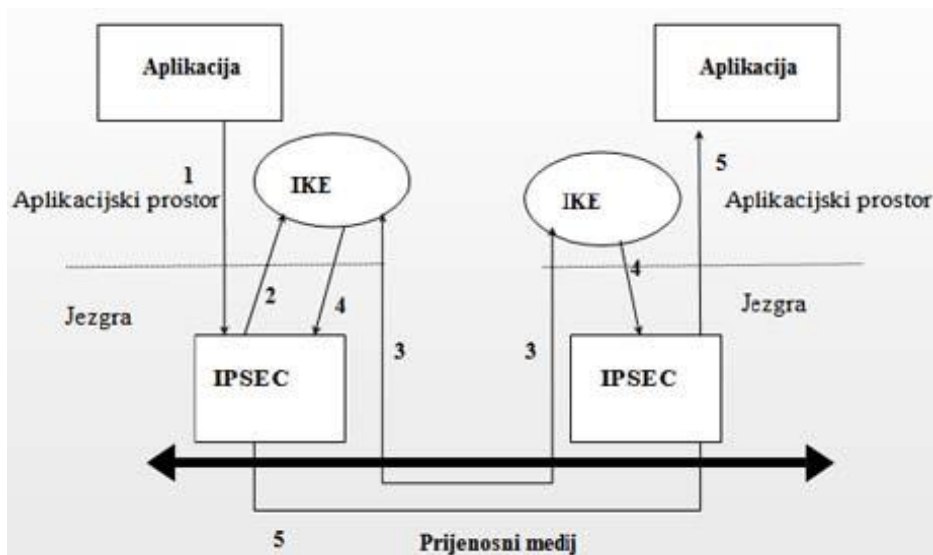
Postoji nekoliko načina na koje je moguće uspostaviti IPsec komunikaciju. Na prvom mestu, teoretski je moguće ručno podešavanje skupa sigurnosnih parametara, no to za bilo kakvu ozbiljniju primjenu nije prihvatljivo. Osim toga, postoji nekoliko formalnih metoda koje se koriste ili su predlagane za uspostavljanje IPsec komunikacije.

Photuris i SKIP (*Simple Key management for Internet Protocols*), protokoli bazirani na Diffie-Hellman razmeni ključeva mogu se koristiti za tu svrhu, no u širokoj primeni prihvaćen je ISAKMP (*InternetSecurity Association and Key Management Protocol*), odnosno IKE (*Internet Key Exchange*).

7.1

IKE protokol

Zaglavlje za razmenu ključeva koristi se kod protokola za razmenu ključeva (*Internet Key Exchange - IKE*) kao jednog od tri IPsec protokola. IKE je prilagodljiv protokol za dogovaranje autentifikacijskih metoda, kriptografskih algoritama i dužina ključeva, te za razmenu samih ključeva među učesnicima u komunikaciji.



Slika 19. Uspostavljanje SA veze pomoću IKE daemona

7.1.1 IKE (Internet Key Exchange)

IKE je standardni protokol za uspostavljanje sigurne IPsec komunikacije. Protokol je implementiran kombinovanjem nekoliko postojećih protokola; na prvom mestu ISAKMP, te Oakley i SKEME protokola.

ISAKMP protokol određuje infrastrukturu za autentifikaciju i razmenu ključeva. Protokol je oblikovan tako da je nezavisan o načinu razmene ključeva, odnosno može podržati razne metode za razmenu ključeva.

Oakley protokol opisuje načine razmene ključeva (*modes*), detalje i usluge koje svaki od tih načina pruža.

SKEME je također protokol koji opisuje način razmene ključeva i koji osigurava anonimnost, a baziran je na starijem Photuris protokolu.

Uspostava IPsec komunikacije korištenjem IKE protokola sastoji se od dve osnovne faze:

- o uspostavljanja IKE (ISAKMP) SA skupa sigurnosnih parametara i
- o uspostavljanja IPsec SA skupa sigurnosnih parametara korištenjem IKE SA. Uobičajeno je da se za IKE komunikaciju koristi UDP port 500.

7.1.2 Uspostavljanje IKE SA

Osnovna funkcija IKE SA skupa sigurnosnih parametara jeste osigurati autentifikaciju i sigurnost IKE prometa, a unutar tako uspostavljene komunikacije mogu se definisati višestruki IPsec SA.

Atributi koje uspostavljeni IKE SA mora sadržati su:

- o algoritam za šifriranje,
- o hash funkcija,
- o metoda autentifikacije i
- o Oakley grupa koja definiše Diffie-Hellman razmenu ključeva (RSA, elipsaste krive).

Metoda autentifikacije označava način na koji će se entiteti autentificirati u nadolazećoj komunikaciji. IKE podržava sledeće metode autentifikacije:

- o korišćenje digitalnih potpisa (RSA ili DSS),
- o korišćenje tajnog ključa (*preshared key*) i
- o korišćenje kriptografije temeljene na javnim ključevima (osnovna i modifikovana metoda).

Uspostavljanje IKE SA može se sprovesti na dva načina:

- o glavni način (*main mode*) i
- o agresivni način (*aggressive mode*).

Glavni način se koristi kada je nužna zaštita identiteta entiteta u komunikaciji. U glavnom načinu rada entiteti moraju razmijeniti 6 poruka da bi uspostavili IKE SA.

Agresivni način se može koristiti kada zaštita identiteta nije nužna, već je poželjna što veća brzina uspostavljanja komunikacije. Kod ovog načina potrebna je razmena svega 3 poruke između entiteta. Treba napomenuti da će, ukoliko se kod agresivnog načina koristi kriptografija temeljena na javnim ključevima, zaštita identiteta takođe biti osigurana. Za održavanje IKE komunikacije potrebno je generisati četiri različita ključa sesije (*session key*):

- o SKEYID – glavni ključ koji se koristi za generisanje ostalih ključeva,
- o SKEYID_d – ključ koji IKE SA koristi za šifrovanje poruka,
- o SKEYID_a – ključ koji IKE SA koristi za osiguranje integriteta i autentifikaciju poruka,
- o SKEYID_e – ključ koji služi za generisanje IPsec SA.

Pri generisanju ključeva koriste se i kolačići koje generišu entiteti u komunikaciji i koji predstavljaju hash vrednosti izračunate od identifikatora (IP adresa entiteta, port, protokol), vremenske značke (vreme pristupa) i tajne vrednosti poznate samo entitetu koji je generisao kolačić.

Metoda autentifikacije korišćenjem tajnog ključa ranjiva je na tzv. man-in-the-middle napade, što je inherentno svojstvo Diffie-Hellman algoritma za razmenu

ključeva, tako da se za sigurnu autentifikaciju preporučuje korištenje kriptografije temeljene na javnim ključevima, odnosno PKI.

Kod agresivnog načina broj poruka koje je potrebno razmeniti je smanjen, no kod ovog načina, osim što nije osigurana zaštita identiteta, moguće je izvođenje napada primjenom sile (*brute force*) jer se HASH vrednosti u ovom načinu prenose kao otvoreni tekst, a ne šifrovano kao kod normalnog načina.

7.1.3 Uspostavljanje IPsec SA

Druga faza IKE protokola služi za uspostavljanje IPsec SA skupa sigurnosnih parametara. Ova faza sprovodi se u tzv. brzom načinu (*quick mode*). Cela komunikacija koja se odvija kroz drugu fazu zaštićena je korištenjem prethodno uspostavljenog IKE SA skupa sigurnosnih parametara. Ova faza ustvari nije zasebna faza, već se koristi samo za generisanje IPsec SA na temelju ranije uspostavljenog IKE SA.

Nakon što se okonča druga faza IKE protokola, dva entiteta u komunikaciji su definisala IPsec SA skup sigurnosnih parametara i mogu uspostaviti siguran kanal za razmenu poruka.

7.2 IPsec redundancija

Jedno od pitanja koje se često postavlja jesu dodatni resursi koje IPsec zahteva. Zbog kriptografskih operacija koje su matematički zahtevne, korištenje IPsec-a zahteva dodatne procesorske resurse.

Osim zahteva za procesorskim resursima, IPSec takođe povećava ukupan mrežni promet, što je samo po sebi razumljivo ukoliko se IPSec datagrami posmatraju u odnosu na standardne IP datagrame.

Povećanje mrežnog prometa, odnosno redundancija (*overhead*) koju IPSec unosi, a što može rezultovati degradacijom mrežnih performansi, proizlazi iz dva funkcionalna razloga:

- o zbog dodatnih zaglavlja koja se mogu pojaviti u različitim načinima IPSec rada,
- o zbog kriptografskih algoritama koji se koriste, odnosno ispune (*padding*) koja je nužna za njihovo ispravno funkcionisanje.

7.3 Redundancija zaglavlja

Redundancija zaglavlja zavisi od načina rada IPSec-a, kao i od IPSec protokola koji se koriste. Korišćenje AH protokola (ukoliko se koriste propisane hash funkcije MD5 ili SHA-1) unosi redundanciju od 24 okteta od čega 12 okteta otpada na zaglavlje bez polja autentifikacijski podaci, dok preostalih 12 okteta (96 bita) otpada na to polje koje sadrži ICV vrednost generisanu od strane hash funkcija.

Ovdje treba napomenuti, iako MD5 daje izlazni rezultat dužine 128 bita, a SHA-1 160 bita, da se te vrednosti za potrebe IPSec-a svode na 96-bitni dužinu.

Ukoliko se koristi ESP protokol, redundancija zavisi od toga da li se ESP koristi samo za osiguravanje tajnosti, ili služi takođe za osiguranje integriteta, neporecivosti i autentifikacije poruke. Također, redundancija zavisi i od kriptografskog protokola koji se koristi. ESP zaglavlje samo po sebi dodaje 8 okteta. Nadalje, ukoliko se kriptografski algoritmi koriste u CBC načinu rada, zahtevaće se korišćenje IV, inicijalizacijskog vektora, čija dužina može biti do 16 okteta (8 okteta za DES i 3DES ili 16 okteta za AES).

Tu su još četiri okteta koji se odnose na polja dužina ispune i sledeće zaglavlje (dva okteta za polja i dva nužna okteta za poravnanje do 32-bitne reči), a ukoliko se ESP koristi za osiguranje integriteta, neporecivosti i autentifikacije, potrebno je dodati i 12 oktetnu ICV vrednost koja je u tom slučaju sadržana u polju autentifikacijski podaci na kraju ESP datagrama.

Konačno, ukoliko se IPSec koristi u tunelskom načinu rada treba dodati i 20 okteta, kolika je dužina novog IP zaglavlja. Tabela 1 prikazuje redundanciju koju unosi IPSec, zavisno od načina rada, protokola i kriptografskih algoritmima koji se koriste.

		Transportni način	Tuneliranje
AH	MD5, SHA-1	24 okteta	44 okteta
ESP	DES, 3DES, AES + MD5, SHA-1	24 okteta	44 okteta
	DES-CBC, 3DES CBC + MD5, SHA-1	32 okteta	52 okteta
	AES-CBC + MD5, SHA-1	40 okteta	60 okteta
AH+ESP	DES, 3DES, AES+ MD5, SHA-1	36 okteta	-
	DES-CBC, 3DES CBC + MD5, SHA-1	44 okteta	-
	AES-CBC + MD5, SHA-1	52 okteta	-

Tabela 1. Redundancija IPSec zaglavlja

Tabela 2 prikazuje prosečnu redundanciju koju IPSec zaglavlja dodaju u odnosu na standardni IP datagram. Iz tablice se lako može uočiti da je ta redundancija uopšteno nešto veća u tunelskom načinu rada, te da je značajna za kratke datagrame, dok kod većih datagrama postaje zanemariva.

Dužina paketa	Transportni način			Tuneliranje		
	ESP 3DES CBC + SHA-1, MD5	ESP AES CBC + SHA-1, MD5	AH SHA-1, MD5	ESP 3DES CBC + SHA-1, MD5	ESP AES CBC + SHA-1, MD5	AH SHA-1, MD5
46	70%	87%	52%	113%	130%	96%
512	6,3%	7,8%	4,7%	10,2%	11,7%	8,6%
1500	2,1%	2,7%	1,6%	3,5%	4%	2,9%

Tabela 2. Prikaz prosečne redundancije IPsec zaglavlja u odnosu na dužinu datagrama

7.4 Redundancija ispune

Redundancija ispune ovisi o IPsec protokolima koji se koriste, odnosno direktno o kriptografskim algoritmima i hash funkcijama koje su odabrane u SA skupu sigurnosnih parametara. Ta ispuna je nužna iz razloga što kriptografski algoritmi i hash funkcije kao ulaz koriste blokove fiksne dužine, čija dužina zavisi od specifičnog algoritma koji će se koristiti (tabela 3).

Kriptografski algoritam/hash funkcija	Dužina bloka
DES, 3DES	64 bita
AES	128 bita
MD5, SHA-1	512 bita

Tabela 3. Dužine ulaznih blokova podataka za uobičajene kriptografske algoritme i hash funkcije

Kod kriptografskih algoritama (DES, 3DES, AES) to konkretno znači da će svaki datagram imati ispunu takvu da IP datagram bude poravnat na 64 bitni odnosno 128 bitni blok. Kod hash funkcija (MD5 i SHA-1), zbog implementacijske specifičnosti, datagram će biti poravnat na 448 bitni blok. Razlog tome je što oba algoritma ulaznim podacima implicitno dodaju 64-bitni blok podataka, što skraćuje dužinu ulaznog bloka koji može biti procesuiran u hash funkciji. Tabela 4 prikazuje redundanciju ispune u najgorim slučajevima (kada je potrebna dužina ispune maksimalna). Uočljivo je da je redundancija kod manjih paketa značajna, dok kod većih paketa ta redundancija u relativnom učešću značajno opada.

Algoritam/ funkcija	Dužina bloka	Mali paketi (~50 okteta)	Prosečni paketi (~350 okteta)	Veliki paketi (~1500 okteta)	Dužina paketa je sadržalac bloka
DES, 3DES	64 bita	~15%	~2%	~0,5%	0%
AES	128 bita	~30%	~4%	~1%	0%
SHA-1, MD5	512 bita	~100%	~18%	~4%	0%

Tabela 4. Redundancija ispune u najgorim slučajevima

8 Implementacije IPsec protokola

Na Linux/Unix operativnim sistemima osnove IPsec protokola implementirane su u samo jezgro (*kernel*), dok je rad s IKE ključevima realizovan kroz ISAKMP/IKE sistem.

Projekat grupe FreeS/WAN – pluto je prva open source implementacija celokupnog IPsec sistema za Linux/Unix operativne sisteme. Ovaj projekat se sastoji od posebnih dodataka za samo jezgro napisanih u obliku programskih skripti koje se koriste za podešavanje parametara kojima se definiše da li se određeni paket propušta ili odbacuje. (parametri SAD - *Security Association Database* i SPD - *Security Policy Database* filtera).

Za Windows operativne sisteme postoje alati namenjeni Server 2000 i Server 2003 kojima se uređuje komunikacija IPsec protokolom na nivou domena koju ti serveri kontrolišu (Microsoft Domain Isolation). Takođe, kod MS Windows XP operativnog sistema s instaliranim SP2, korisnicima stoje na raspolaganju programi koji omogućavaju transparentu komunikaciju IPsec protokolom.

Od ostalih sistema za implementaciju IPsec protokola važno je još napomenuti i Ciscove programske pakete pomoću kojih i mrežni uređaji ove firme mogu komunicirati navedenim protokolom.

8.1 Implementacijski problemi

IPsec protokol, sam po sebi donosi neke probleme koje je ponekad, u specifičnim mrežnim okruženjima teško ili nemoguće riješiti. To se prvenstveno odnosi na korištenje NAT-a (*Network Address Translation*) kao i IP fragmentaciju koja se može pojaviti prilikom IPsec komunikacije.

8.1.1 NAT

Obzirom na poznata ograničenja adresiranja kod IPv4 protokola, velik broj lokalnih mreža koristi privatno adresiranje, a pristup Internetu, odnosno udaljenim lokacijama, implementira se korištenjem NAT-a (*Network Address Translation*) koji može biti statički ili dinamički, odnosno NAPT-a (*Network Address Port Translation*).

Ukoliko se bilo gde između entiteta koji žele uspostaviti IPsec komunikaciju upotrebljava NAT, korišćenje AH bilo u transportnom, bilo u tunelskom načinu rada, za TCP/UDP komunikaciju nije moguće pošto će korišćenje NAT-a rezultovati narušavanjem integriteta IP datagrama i uzrokovati njegovo odbacivanje na strani primaoca. Ukoliko se za IPsec koristi samo ESP, stvar je donekle drugačija. U transportnom načinu rada korišćenje NAT-a takođe rezultuje nemogućnošću IPsec komunikacije, no u tunelskom načinu ESP može funkcionisati.

Pri korišćenju NAT-a pažnju valja obratiti i na IKE/ISAKMP, jer autentifikacija temeljena na tajnom ključu koristi i kolačiće koji se generišu zavisno od IP adrese entiteta, što takođe rezultira gubitkom integriteta, te nemogućnošću uspostavljanja komunikacije. Ovaj nedostatak, za razliku od problema sa AH i ESP, može se rešiti korišćenjem drugih IKE autentifikacijskih metoda (korišćenje digitalnih potpisa ili kriptografije temeljene na javnim ključevima).

8.1.2 Fragmentacija

Pri korišćenju IPsec komunikacije, prvenstveno u tunelskom načinu rada, a isto tako i IKE/ISAKMP komunikacije, može doći do IP fragmentacije. U tom slučaju mogu se pojaviti dodatni problemi. Naime, neki firewall-ovi ili ruteri mogu biti konfigurisani tako da odbacuju IP fragmentovane datagrame, jer na taj način osiguravaju mreže od nekih oblika DoS (*Denial of Service*) napada (npr. Teardrop). Očiti problem kod toga jeste da će u tom slučaju i legitimni IPsec ili IKE paketi isto biti odbačeni, što će onemogućiti uspostavljanje sigurnog kanala između udaljenih entiteta.

8.2 Cisco i IPsec implementacija

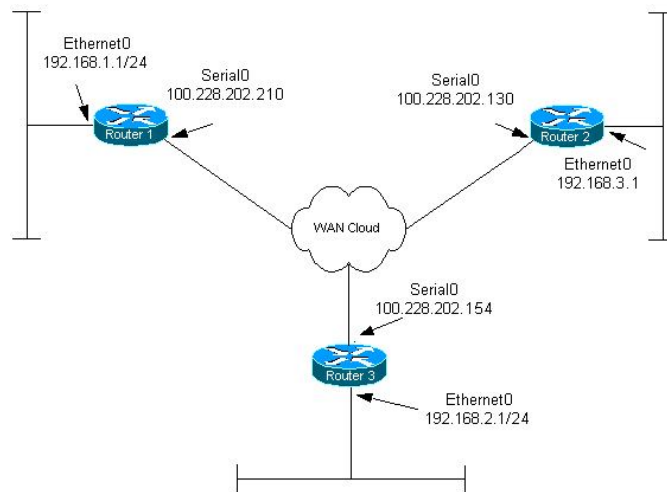
Setovanjem Cisco routera treba ispitati njihovo ponašanje u intranet poslovnoj mreži preduzeća. Za primer su upotrebljena tri routera po uputstvima za setovanje od strane proizvođača. Rad na routerima je obavljen neradnog dana (nedelja) kada su aktivne samo granične ispostave pa su se tako iste uključile u rad na primeru.

U primeru se ilustruju:

- Encapsulating Security Payload (ESP) protokol - samo Data Encryption Standard (DES)
- Pre-šerovani ključevi
- Lokalne mreže u domenu privatnih adresa iza svakog router-a:
 - o 192.168.1.0
 - o 192.168.2.0

- o 192.168.3.0
- isakmp polisa i crypto map konfiguracija

Saoubračaj sačinjenim tunelom je definisan access listom i mapom rutiranja



Slika 20. WAN mreža sa tri routera

Primer 1. Konfiguracija Router-a br. 1

Current configuration:

```

!
version 12.3
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname router1
!
boot-start-marker
boot-end-marker
!
!
clock timezone EST 0
no aaa new-model
ip subnet-zero
!
!
ip audit po max-events 100
no ftp-server write-enable
!
!--- Configure Internet Key Exchange (IKE) policy and
!--- pre-shared keys for each peer.
!
!--- IKE policy defined for peers.

```

```
!  
crypto isakmp policy 4  
authentication pre-share  
!  
!--- Pre-shared keys for different peers.  
!  
crypto isakmp key xxxxxx1234 address 100.228.202.154  
crypto isakmp key xxxxxx1234 address 100.228.202.130  
!  
!  
!--- IPsec policies:  
!  
crypto ipsec transform-set encrypt-des esp-des  
!  
!  
crypto map combined local-address Serial0  
!  
!--- Set the peer, transform-set and encryption traffic for tunnel  
peers.  
!  
crypto map combined 20 ipsec-isakmp  
    set peer 100.228.202.154  
    set transform-set encrypt-des  
    match address 106  
crypto map combined 30 ipsec-isakmp  
    set peer 100.228.202.130  
    set transform-set encrypt-des  
    match address 105  
!  
!  
interface Serial0  
    ip address 100.228.202.210 255.255.255.252  
    ip nat outside  
    serial restart-delay 0  
!  
!--- Apply the crypto map to the interface.  
!  
    crypto map combined  
!  
interface Ethernet0  
    ip address 192.168.1.1 255.255.255.0  
    ip nat inside  
!  
ip classless  
ip route 0.0.0.0 0.0.0.0 100.228.202.209  
no ip http server  
no ip http secure-server  
!  
!--- Define traffic for NAT.  
!
```

```
ip nat inside source route-map nonat interface Serial0 overload
!
!--- Access control list (ACL) that shows traffic to encrypt over the
tunnel.
!
access-list 105 permit ip 192.168.1.0 0.0.0.255 192.168.3.0
0.0.0.255
access-list 106 permit ip 192.168.1.0 0.0.0.255 192.168.2.0
0.0.0.255
!
!--- ACL to avoid the traffic through NAT over the tunnel.
!
access-list 150 deny ip 192.168.1.0 0.0.0.255 192.168.2.0 0.0.0.255
access-list 150 deny ip 192.168.1.0 0.0.0.255 192.168.3.0 0.0.0.255
!
!--- ACL to perform NAT on the traffic that does not go over the
tunnel.
!
access-list 150 permit ip 192.168.1.0 0.0.0.255 any
!
!--- Do not perform NAT on the IPSec traffic.
!
route-map nonat permit 10
  match ip address 150
!
control-plane
!
!
line con 0
line aux 0
line vty 0 4
!
!
end
```

Primer 2. Konfiguracija Router-a br. 2

Current configuration:

```
!
version 12.3
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname router2
!
boot-start-marker
boot-end-marker
!
```

```
!  
clock timezone EST 0  
no aaa new-model  
ip subnet-zero  
!  
!  
ip audit po max-events 100  
no ftp-server write-enable  
!  
!--- Configure IKE policy and pre-shared keys for each peer.  
!  
!--- IKE policy defined for peers.  
!  
crypto isakmp policy 4  
    authentication pre-share  
!  
!--- Pre-shared keys for different peers.  
!  
crypto isakmp key xxxxxx1234 address 100.228.202.154  
crypto isakmp key xxxxxx1234 address 100.228.202.210  
!  
!  
!--- IPsec policies.  
!  
crypto ipsec transform-set encrypt-des esp-des  
!  
!  
crypto map combined local-address Ethernet1  
!  
!--- Set the peer, transform-set and encryption traffic for tunnel  
peers.  
!  
crypto map combined 7 ipsec-isakmp  
    set peer 100.228.202.210  
    set transform-set encrypt-des  
    match address 105  
  
crypto map combined 8 ipsec-isakmp  
    set peer 100.228.202.154  
    set transform-set encrypt-des  
    match address 106  
!  
!  
!  
interface Ethernet0  
    ip address 192.168.3.1 255.255.255.0  
    ip nat inside  
!  
interface Ethernet1  
    ip address 100.228.202.130 255.255.255.224
```

```
ip nat outside
!
!--- Apply the crypto map to the interface.
!
crypto map combined
!
ip classless
ip route 0.0.0.0 0.0.0.0 100.228.202.129
no ip http server
no ip http secure-server
!
!--- Define traffic for NAT.
!
ip nat inside source route-map nonat interface Ethernet1 overload
!
!--- ACL shows traffic to encrypt over the tunnel.
!
access-list 105 permit ip 192.168.3.0 0.0.0.255 192.168.1.0
0.0.0.255
access-list 106 permit ip 192.168.3.0 0.0.0.255 192.168.2.0
0.0.0.255
!
!--- ACL to avoid the traffic through NAT over the tunnel.
!
access-list 150 deny ip 192.168.3.0 0.0.0.255 192.168.1.0 0.0.0.255
access-list 150 deny ip 192.168.3.0 0.0.0.255 192.168.2.0 0.0.0.255
!
!--- ACL to perform NAT on the traffic that does not go over the
tunnel.
!
access-list 150 permit ip any any
!
!--- Do not perform NAT on the IPSec traffic.
!
route-map nonat permit 10
match ip address 150
!
!
!
control-plane
!
!
line con 0
line aux 0
line vty 0 4
!
!
end
```

Primer 3. Konfiguracija Router-a br. 3

Current configuration:

```
!  
version 12.3  
service timestamps debug datetime msec  
service timestamps log datetime msec  
no service password-encryption  
!  
hostname router3  
!  
boot-start-marker  
boot-end-marker  
!  
!  
clock timezone EST 0  
no aaa new-model  
ip subnet-zero  
!  
!  
ip audit po max-events 100  
no ftp-server write-enable  
!  
!--- Configure IKE policy and pre-shared keys for each peer.  
!  
!--- IKE policy defined for peers.  
!  
crypto isakmp policy 4  
  authentication pre-share  
!  
!--- Pre-shared keys for different peers.  
!  
crypto isakmp key xxxxxx1234 address 100.228.202.210  
crypto isakmp key xxxxxx1234 address 100.228.202.130  
!  
!  
!--- IPSec policies:  
!  
crypto ipsec transform-set encrypt-des esp-des  
!  
!--- Set the peer, transform-set and encryption traffic for tunnel  
peers.  
!  
crypto map combined local-address Serial0  
crypto map combined 7 ipsec-isakmp  
  set peer 100.228.202.210  
  set transform-set encrypt-des  
  match address 106
```

```
crypto map combined 8 ipsec-isakmp
  set peer 100.228.202.130
  set transform-set encrypt-des
  match address 105
!
!
interface Serial0
  ip address 100.228.202.154 255.255.255.252
  ip nat outside
  serial restart-delay 0
!
!--- Apply the crypto map to the interface.
!
  crypto map combined
!
  interface Ethernet0
    ip address 192.168.2.1 255.255.255.0
    ip nat inside
!
ip classless
ip route 0.0.0.0 0.0.0.0 100.228.202.153
no ip http server
no ip http secure-server
!
!--- Define traffic for NAT.
!
ip nat inside source route-map nonat interface Serial0 overload
!
!--- ACL that shows traffic to encrypt over the tunnel.
!
access-list 105 permit ip 192.168.2.0 0.0.0.255 192.168.3.0
0.0.0.255
access-list 106 permit ip 192.168.2.0 0.0.0.255 192.168.1.0
0.0.0.255
!
!--- ACL to avoid the traffic through NAT over the tunnel.
!
access-list 150 deny ip 192.168.2.0 0.0.0.255 192.168.3.0 0.0.0.255
access-list 150 deny ip 192.168.2.0 0.0.0.255 192.168.1.0 0.0.0.255
!
!--- ACL to perform NAT on the traffic that does not go over the
tunnel.
!
access-list 150 permit ip 192.168.2.0 0.0.0.255 any
!
!--- Do not perform NAT on the IPSec traffic.
!
route-map nonat permit 10
  match ip address 150
!
```

```
!  
!  
control-plane  
!  
!  
line con 0  
line aux 0  
line vty 0 4  
  login  
!  
!  
End
```

Pre i nakon setovanja, ispitivanje protoka je vršeno prostim pingom sa jedne mašine iz lokalne mreže Router-a br. 1 na mašine u ostale dve lokalne mreže gde se pokazalo da je vreme pinga povećano za oko 10% što je prihvatljivo.

Ovo rešenje je dakle spremno u slučaju da se poslovna mreža funkcionalno proširi u Extranet povezivanjem sa firmom vlasnikom u Sloveniji.

8.3 WWAN i APN segment i IPsec implementacija

Implementacija IPsec-a na WWAN segmentu je primer velikog poremećaja u vidu usporavanja protoka. Kao što je već rečeno u poglavlju 1. osnovna zaštita predstavlja šifrovanje algoritmima obezbeđenim od strane Motorola u njihovim Canopy uređajima za primopredaju signala. Da bi sigurnost bila veća, podigao se IPsec VPN prema dve wireless lokacije. Tog momenta je nastupilo veliko kašnjenje paketa – do dva puta (2 x) u odnosu na do tada potrebno vreme. U prvom momentu je bila aktuelna pretpostavka da proces šifriranja i dešifriranja putem Motorolinih uređaja predstavlja zahtevan proces u kombinaciji sa podignutim VPN-om da bi se na kraju potvrdilo da problem prave instalacije IPsec-a na različitim operativnim sistema na dva transportna kraja. Sa strane čvorišta je podignut IPsec na Linux mašini a sa strane klijentskih mašina je na Windows XP. Potvrdu naših nalaza je nađena u stručnim prepiskama u blogovima na internetu.

Gore rečeno se može primeniti i na APN segment mreže gde su problemi koji se javljaju izraženiji zbog nestabilnosti brzine protoka.

9 Izbor baze i modela pohranjivanja podataka

U svakom segmentu poslovanja se evidentiraju podaci koji se obrađuju, izvode i prezentuju. Nakon toga dolazi do optimiziranja istorodnih podataka da ne bi bilo umnožavanja istih podataka unutar baze i da se svaki podatak jednom unese bez ponavljanja. To uslovljava smanjenje broja tabela, naročito u slučaju konsultativnih tabela (najčešće šifarnika).

Osnovna intencija je da se usvoji relacionala baza kao model ne samo zbog savremenosti nego što ona jedino može da odgovori zahtevima koji se postavljaju pred njom. U velikim sistemima (po broju korisnika, promena, razgranatosti ...) najčešće se ide na razne oblike struktuiranja baza i da bi se rasteretile baze a i sami serveri najčešće se formira distribuirana baza. Ona predstavlja da se određene grupe informacija drže na njihovom izvoru, konsultativne tabele i srž u centralnom serveru. Takođe postoji sistem sa replikacijom podataka koji na svim lokalitetima drži ažurne informacije kao i na centralnom serveru. Kao najuobičajenija se prihvata baza na jednom serveru.

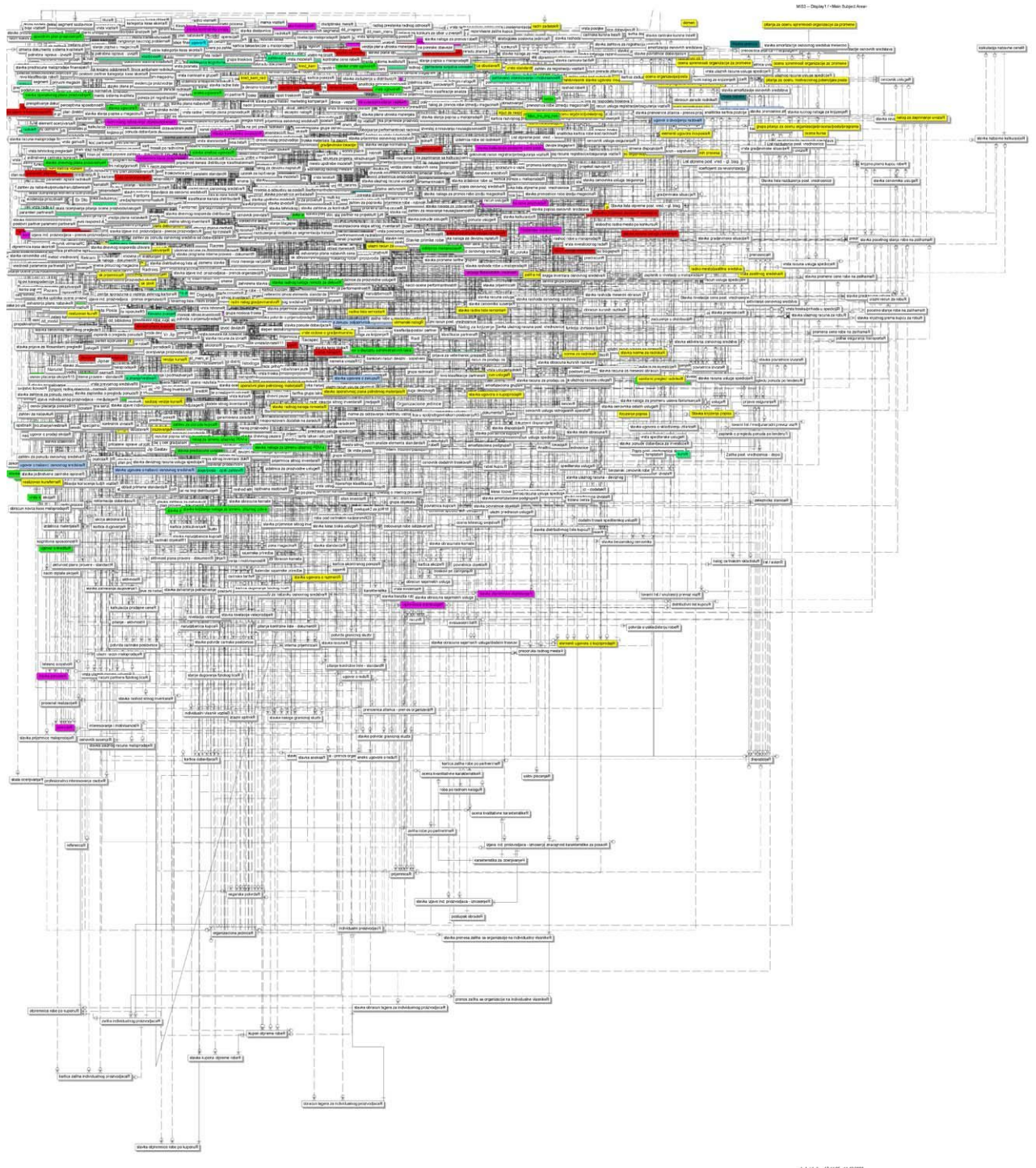
Kod nas je odabrana varijanta centralnog servera zbog rada 24/7 i potrebe ažuriranja poslovnih partnera sve vreme i manjih troškova u pogledu manjeg broja licenciranih baza. Izabrana je relacionala baza firme INFORMIX, verzija 9.2, koja je podignuta na dvoprocessorskom opteron serveru.

9.1 Relacionala baza - pregled

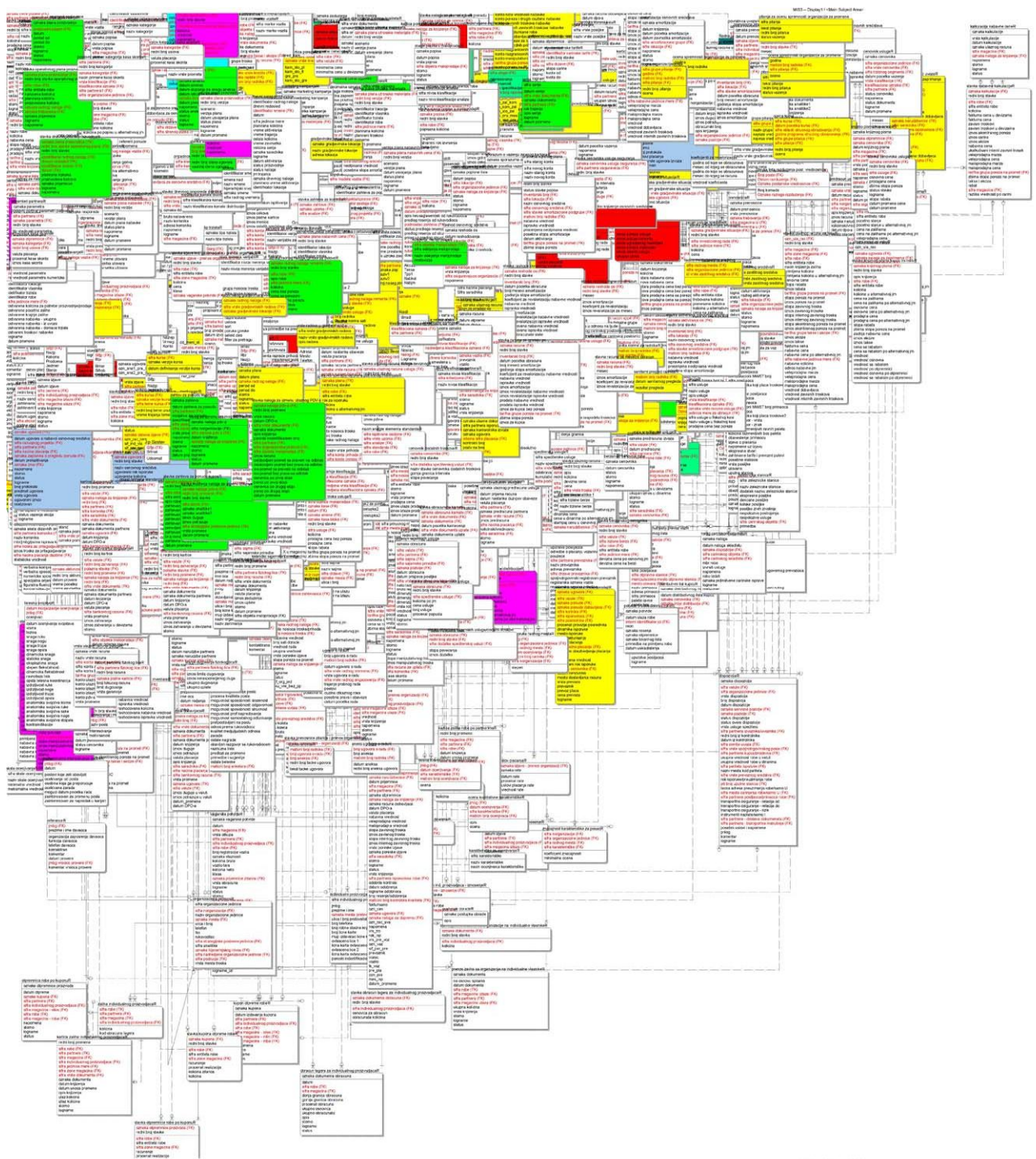
U razvoj IPS-a se ušlo sa firmom M&I Systems iz Novog Sada koji su imali već rešenja za standardne module opštih poslova koji su slični u svim branšama nadograđene sa specifičnim modulima koji su se bavili konkretnim delatnostima datih preduzeća. Tim pristupom je formirana baza koja se razvijala i povećavala sa novom delatnošću koja se obuhvatala u novoj firmi koja bi primenjivala dati IPS. U ovom modelu koji se prikazuje postoje 1393 tabele.

IPS je modularno koncipiran pa je tako i baza konstruisana gde je definisano koje relacije pripadaju kom nizu tabela. Povezivanjem u grupe održavanje baze se pojednostavljuje naročito kod velikih baza poput ove. U svakom slučaju se moraju koristiti ozbiljniji alati za modelovanje baza kao što je ERWin verzija 4.1 . Alatka poseduje mnoge automatizovane funkcije koje dozvoljavaju da se baza modelira do sitnih detalja, sa svim elementima meta podataka. Takođe takvi alati poseduju mogućnosti sačinjavanja pregleda i izveštaja potrebnih osobi koja se bavi administriranjem baze.

Grafički prikaz cele baze se može prikazati na fizičkom i logičkom nivou po osnovu prikaza relacija (entiteta) i po osnovu šema relacija sa nazivima atributa ili sa njihovim domenima što se da naslutiti u sledećim slikama koje su date samo zbog toga da bi se osetila veličina baze o kojoj se priča.



Slika 21. Model baze na logičkom nivou - entiteti sa svojim vezama



Slika 22. Model baze na logičkom nivou - šema relacija (sa nazivima atributa)

9.2

Sigurnost podataka u bazi - trenutno stanje

Sigurnost baze zavisi od celokupne zaštite sistema koja pokriva višestruke nivoe, počev od sigurnosti na Internetu i na mreži, pa sve do zaštite pojedinačnih polja u nekom dokumentu u bazi. Ovi nivoi su komplementarni - primena sigurnosnih mera na određenom nivou istovremeno pojačava sigurnost i iznad i ispod njega. Osim o sigurnosti mrežnih protokola kao i server i klijent operativnih sistema, zaštita obuhvata sledeće sigurnosne oblasti:

1. Sigurnost na mreži
2. Sigurnost na Internetu
3. Hijerarhijsko imenovanje servera i korisnika
4. Sigurnost servera
5. Sigurnost klijenta
6. Sigurnost Web browser klijenta
7. Sigurnost baze podataka
8. Sigurnost na nivou elementa dizajna baze podataka
9. Sigurnost mail sistema

Ono što će biti tema razmatranja ovog rada je poboljšanje sigurnosti podataka unutar Informix baze koristeći mogućnosti njenog RDBMS-a (sistema za upravljanjem relacionom bazom podataka).

Zatečeno stanje predstavlja pristup Informix bazi kroz jednoznačnu autentifikaciju na nivou aplikacije a kontrola pristupa pojedinih korisnika tabelama unutar baze se kontroliše unutar aplikacije po predhodno definisanim pravima korisnika zavisno od uloge u preduzeću.

Podaci u bazi su u sirovom (nešifrovanom) formatu i time su vidljivi pri pristupu bilo kojim SQL editorom koji može da čita Informix bazu.

10 Istraživanje baze

S obzirom da je Informix baza legalno kupljena i time omogućena tehnička podrška proizvođača, osnova istraživanja će biti kroz prikupljanje relevantnih podataka od proizvođača i eksperimentalno potvrđivanje tako donetih postavki.

Pomenuta Informix baza je pristupačna i moguće je bez ugrožavanja svakodnevnih aktivnosti preduzeća izvršiti ekperimentalno ispitivanje mogućnosti baze u smislu zaštite podataka osim postupka šifrovanja koji će biti testiran nad IBM-ovom Informix demo bazom po njihovim preporukama.

Drugi izvori osim proizvođača, vezani za datu temu, nisu prisutni ni na internetu ni van njega. Upotreba Informix baze u svetu je dosta raširena iako kod nas predstavlja retkost. Informix baza ima razvijen set karakteristika vezanih za sigurnost podataka koji se veoma retko koristi kod korisnika pa je stoga i dokumentacija sa poreklom od korisnika retka. Kao rezultat navedenog, nije nađen ni jedan dokument pisan na teritorijama Republika izraslih iz bivše SFRJ koji se bavi ovom tematikom.

Osnova na kojoj će se bazirati istraživanje će biti na osnovu dokumentacije koja se može naći na zvaničnom IBM site-u.

Informix baza je instalirana na OS SuSE Linux 10.0 .

11 Zaštita podataka u informix bazi

Zaštita podataka u bazi je krucijalna u današnjem svetu i važnost se iz dana u dan povećava kako se informatika uvači u svaki segment našeg življenja. Nemogućnost očuvanja tajnosti podataka u poslovnom procesu, kada se posmatra rad preduzeća, često dovodi do štete nastale usled zloupotrebe ukradenih podataka u raznim vidovima počevši od materijalne pa do gubitka klijenta. Ova tvrdnja važi i za ostale oblike delatnosti koje se oslanjaju na informatičku obradu podataka gde se šteta sagledava u zavisnosti od specifičnosti date delatnosti.

Kada se govori o zaštiti pojedinačnog podatka u bazi mora se voditi računa o proveru identiteta onoga ko pristupa, o upravljanju pristupom podacima i održavanjem sigurnosnog nivoa samog skladišta podataka (jednog ili više servera).

Sledeći procesi su neizostavno uključeni kada se govori zaštiti podataka:

- Autentikacija
- Autorizacija
- Šifrovanje

Sa svakim od ovih procesa bavićemo se detaljnije u daljem razmatranju.

12 Autentikacija

U ovom poglavlju ćemo ispitati kako Sistem za upravljanje bazom (DBMS) kontroliše autentikaciju korisnika i vrši prihvatanje i odbijanje konekcije. Metodi koje Informix nudi su određeni kroz niz setovanja instanci DBSERVERNAME i DBSERVERALIAS. Pošto su metodi za iste vrste pristupa isključivi, pri potrebi pristupa po više metoda koristi se višestruko setovanje instance DBSERVERALIAS.

Moguće metode su:

- Korisnički ID setovan u operativnom sistemu
- Korisnički ID setovan u operativnom sistemu - kriptovan tokom prenosa
- Pluggable Authentication Module (PAM) - autentikacijski metod obezbeđen od strane korisnika
- Lightweight Directory Access Protocol (LDAP) - pristup LDAP direktorijumu obezbeđen od strane korisnika

12.1 Izbor vrsta autentikacije pri konekciji

Pre nego što se proverí korisnički nalog i šifra pri pristupu bazi, DBMS mora predhodno da ima sačinjenu listu povrljivih korisnika i servisa. Koja od ovih lista se proverava određuje vrednost petog polja u svakom SQLHOSTS unosu. Navedeno polje može sadržati raznovrsne specifikacije ali one koje ćemo mi razmatrati su vrednosti koje se dodeljuju promenljivana r i s.

Vrednosti r koriste klijentske biblioteke i server se ne obazire na njih dok vrednosti s koristi server i one nisu vidljive za klijentske biblioteke..

Treba posebno napomenuti da vrednost s=6 predstavlja specijalni slučaj koji se koristi kod replikacije i koji tada zabranjuje pristup za sve osim što obezbeđuje replikacijsku nit između Informix server (IDS) instanci.

Ostale vrednosti ovih promenljivih će se razmotriti u daljem tekstu.

12.1.1 Potpuno poverenje

Ako su vrednosti za s 1 ili 3 pri zahtevu za konekciju, DBMS proverava listu koja se nalazi na sistemskom direktorijumu /etc u host.equiv fajlu da li su korisnik i/ili host na listi i u slučaju da jesu, konekcija se dopušta bez dalje provere. Šifra

nije potrebna da se navodi ali ako je navedena onda mora da bude odgovarajuća.

Iz razloga što je ova tehnika sa potpunim poverenjem, ona se koristi uglavnom u konektovanju između IDS-ova ili za rad u DB klasterima. Takođe se koristi kod izvršavanja distribuiranih upita eksternim IDS-ovima od strane predmetnog IDS-a.

U sledećem primeru 4, prikazan je sadržaj log fajla prilikom konekcije na bazu inis_07 na serveru tranns sa setovanom vrednošću s=3 bez navođenja korisnika i šifre za pristup.

<i>Primer 4. Konektovanje sa s=3</i>
informix@milos:~> dbaccess inis_07@tranns - Database selected. >

12.1.2

Delimično poverenje

Za vrednosti s=2 i s=3 konsultuju se drugačije liste poverljivih korisnika i sistema. Na Linux OS svaki korisnik na svom *home* direktorijumu poseduje .rhosts fajl koji je sličan fajlu /etc/hosts.equiv ali se odnosi samo na dotičnog korisnika.

Upotreba .rhosts se svodi na definisanje kojim distribuiranim sistemima korisnik može da pristupa kao potpuno poverljiv a kojima treba da upiše šifru pristupa. Ovo je dobar mehanizam u slučaju da želimo da ograničimo pristup korisnicima koji se nalaze tog momenta vam inicijalnog sistema.

Kada je s=3 tada se konsultuju obe liste dok kada je s=2 tada je onemogućen distribuirani upit pošto se konsultuje samo lista u fajlu .rhosts .

12.1.3

Bez poverenja

Dajući vrednost s=0, sistem pri konekciji zahteva korisnički nalog i šifru da bi se konekcija sprovedla uspešno. U primeru 5 se prikazuju dva pokušaja konekcije na bazu pri postavci vrednosti s=0, prvi put bez a drugi put sa unošenjem naloga i šifre i proverom uspešne konekcije.

<i>Primer 5. Konektovanje sa s=0</i>
informix@milos:~> dbaccess inis_07@tranns -

956: Client host or user informix@nile.itsosj.sanjose.ibm.com is not trusted by the server.

```
informix@milos:~> dbaccess - -
> connect to 'inis_07@tranns' user 'informix';
```

ENTER PASSWORD:

Connected.

```
> select first 3 * from kurs_lis_car_st;
```

ozn_kur_lis	sif_val	bro_obr_jed	car_kur
79/2008	941	1	1
79/2008	724	100	48,0423
79/2008	978	1	79,9357

3 row(s) retrieved.

```
>
```

```
> select first 3 * from car_obj;
```

sif_car_obj	naz_car_obj	vrs_car_obj	aktivan
11002	Beograd	CA	D
11029	Beograd Luka	CI	D
11037	Pancevo	CI	D

12.1.4

Predstavljane kao drugi korisnik

Klijentski program može da se predstavi kao drugi korisnik nego što je onaj koji se prijavljuje uz setovanje vrednosti $r=1$. Tada se fajl `.netrc` u korisnikovom *home* direktorijumu kontroliše. Ako su nalog i šifra upisani u navedenom fajlu, tada se oni koriste umesto korisnikovog identiteta. U slučaju da se upisuje nalog i šifra u izrazu za konektovanje, tada se `.netrc` ignoriše. Ovo setovanje mora da se postavi u `sqlhosts` fajlu na lokalnoj mašini.

Ovaj metod se ne preporučuje pošto podaci o korisniku stoje u običnom nezaštićenom tekst fajlu.

Postavljanje vrednosti $r=0$ onemogućava `.netrc` fajl i ovo setovanje se preporučuje da bi se sprečila zloupotreba ove metode za lažno predstavljanje.

Donji primer 6 prikazuje sadržaj `.netrc` fajla koji se koristi pri autentikaciji na

server.

Primer 6. Konektovanje sa r=1, sadržaj .netrc fajla

```
informix@nile:~> cat .netrc
machine ns03
    login nikola
    password nikola03
```

12.1.5

Osnovna autentikacija OS šifre

Informix (IDS) od svojih početaka koristi ovu autentikaciju. U operativnom sistemu na kojem je baza kreiraju se korisnici sa šiframa koje se tada koriste za kontrolu seta podataka koji se šalju kroz aplikaciju DBMS-u.

Ako OS vrati informaciju da korisnik sa datim nalogom (greška br. 951) ili šifrom (greška br. 952) ne postoji, tada DBMS odbija konekciju.

U našem slučaju, na OS SuSE Linux-u će se u primeru 7 kreirati dva korisnika i nakon toga u primeru 8 prikazati kako nije dovoljno samo kreirati korisnike nego i dodeliti im prava od strane database administratora (DBA).

Primer 7. Kreiranje korisničkih naloga na SuSE Linux-u

```
useradd -d /home/milos -p milos03 milos
useradd -d /home/nikola -p nikola03 nikola
```

Primer 8. Pokušaji i dozvoljena konektovanja

```
connect to 'inis_07' user 'marko' using 'marko03';
951: Incorrect password or user marko@dbase is not known on the
database

951: Incorrect password or user is not known on the database server.
Error in line 2
Near character position 1

connect to 'inis_07' user 'milos' using 'milos03';
387: No connect permission.

111: ISAM error: no record found.
Error in line 4
```

Near character position 1

connect to 'inis_07' user 'informix' using 'in4mix';

Connected.

grant resource to milos;
Permission granted.

grant connect to nikola;
Permission granted.

disconnect current;
Disconnected.

connect to 'inis_07' user 'nikola' using 'nikola03';
Connected.

select first 3 * from kurs_lis_car_st;
272: No SELECT permission.
Error in line 18
Near character position 31

disconnect current;
Disconnected.

connect to 'inis_07' user 'milos' using 'milos03';
Connected.

select first 3 * from kurs_lis_car_st;
272: No SELECT permission.
Error in line 24
Near character position 31

disconnect current;
informix Disconnected.

connect to 'inis_07' user 'informix' using 'in4mix';
Connected.

grant select on kurs_lis_car_st to milos;
Permission granted.

disconnect current;
Disconnected.

connect to 'inis_07' user 'milos' using 'milos03';
Connected.

select first 3 * from kurs_lis_car_st;

ozn_kur_lis	sif_val	bro_obr_jed	car_kur
79/2008	941	1	1
79/2008	724	100	48,0423
79/2008	978	1	79,9357

3 row(s) retrieved.

disconnect current;
informix Disconnected.

connect to 'inis_07' user 'informix' using 'in4mix';
Connected.

grant select on car_obj to nikola;
Permission granted.

disconnect current;
Disconnected.

connect to 'inis_07' user 'nikola' using 'nikola03';
Connected.

select first 3 * from car_obj;

sif_car_obj	naz_car_obj	vrs_car_obj	aktivan
11002	Beograd	CA	D
11029	Beograd Luka	CI	D
11037	Pancevo	CI	D

3 row(s) retrieved.

Prvi pokušaj konekcije pokušava da prijavi nepostojećeg korisnika "marko" za kojeg OS prijavljuje da je nepostojeći DBMS-u i koji to javlja. Sledeća prijava je za postojećeg korisnika ali brz prava pristupa što DBMS uredno javlja.

Posle davanja prava pristupa od strane DBA korisnika informix, sistem dozvoljava konekciju ali korisnici i dalje nemaju definisana prava na nivou baze. U dva navrata se daju prava nad dve tabele za dva korisnika što rezultira pristupom pomenutim tabelama.

12.2 Autentikacija pomoću zamenljivih modula (PAM)

Autentikacija pomoću zamenljivih modula (Pluggable Authentication Module

- PAM) je skup deljenih biblioteka koje omogućuju administratoru da izabere kako će aplikacije autentikovati korisnike. Način autentikacije i moduli za autentikaciju izabiru se preko konfiguracijskog fajla (pam.conf - linux). Bit PAM-a je odvajanje procesa razvoja aplikacija koje koriste autentikacijske mehanizme od razvoja samih šema. Korištenjem PAM biblioteka programi se mogu razvijati nezavisno o šemi autentikacije, a autentikacijski moduli se povezuju s njima tokom izvođenja (*run-time*). Moguće je menjati autentikacijske mehanizme koji se koriste ili nadograditi postojeće mehanizme bez potrebe za menjanjem aplikacija.

Na tržištu ima dosta proizvođača pomenutih modula, a koriste se zavisno od OS-a i odluke korisnika. U pojedimi slučajeva se nailazi i na PAM-ove koji su napisani od strane korisnika.

12.2.1 Osnovna konfiguracija

Unutar pomenutog seta biblioteka nalazimo rutine i uputstva kada koje od rutina spoljne aplikacije treba da koriste. Podešava se već spominjani parametar `s=4` koji govori da se koriste parametri za PAM.

12.2.2 Upotreba autentikacije šifre

Upotreba je slična kao i kod autentikacije OS šifre jedino što se DBMS ne obraća operativnom sistemu nego nekom drugom metodu.

12.2.3 Upotreba autentikacije zahtev-odgovor

Pri zahtevu za konekciju šalje se aplikaciji zahtev za odgovorom. Ako se obezbedi odgovarajući odgovor, konekcija je dopuštena. Ideja je slična upotrebi sistema lozinke i odgovora u vojsci.

12.2.4 Zahtevi od aplikacije kod autentikacije zahtev-odgovor

Ako se odabere ovaj metod treba svaku aplikaciju programirati da radi određene stvari drugačije od ostalih. Ako se aplikacije ne može preraditi u navedemon smislu mora se obezbediti korištenje drugačijih DBSERVERNAME ili DBSERVERALIAS-a.

12.3 Upotreba LDAP direktorijuma

U standardnoj instalaciji IDS-a se nalazi "kostur" LDAP (Lightweight Directory Access Protocol) modula. Od korisničke strane zavisi u mnogome kako je organizovan LDAP server i servisi.

13 Autorizacija

Nakon uspešne konekcije na server posle postupka autentikacije, nastupaju dalje kontrole nivoa pristupnih privilegija tokom SQL operacija. Upravljanje pristupom po osnovu privilegija se određuje za svakog korisnika.

13.1 Pristup baziran po osnovu uloge - role (RBAC)

Role predstavljaju set privilegija - prava pri pristupu resursima baze tokom izvršavanja određenog poslovnog procesa. Kada se definišu role, one mogu kao takve dodeljene korisniku i obično je jednostavnije nego određivati prava pristupa pojedinačno za svakog korisnika. Više rola može biti dodeljeno jednom korisniku i tada sam korisnik može da bira rolu koju će koristiti od dodeljenih korištenjem SET ROLE komande.

Svaki korisnik može imati dodeljenu podrazumevanu rolu koja mu se dodeljuje konekcijom na DBMS. Ako se želi naknadno dodeliti još koja rola, upotrebljava se već gore pomenuta komanda SET ROLE. Po pravilu podrazumevana rola sadrži samo set privilegija za SQL komandu select (samo pregled podataka). Da bi se dozvolila i izmena podataka, moraju se definisati dodatne role.

Najčešća upotreba podrazumevanih rola ograničavanje privilegija korisnicima van aplikacije kojom pristupaju bazi. Setuju se posebne role koje se dodeljuju aplikaciji i koje tada dozvoljavaju nesmetani rad, dok korisnici van aplikacije mogu da izvrše pregled podataka bez izmene, i to onih podataka koji su definisani kao javni i dozvoljeni kroz podrazumevanu rolu.

U primeru 9 ispitujemo ponašanje ove tehnike kroz sledeće korake:

1. Kreiranje dve role
2. Dodeljivanje roli korisniku
3. Konektovanje tog korisnika
4. Provera privilegija
5. Upotreba komande SET ROLE radi promene role
6. Prikaz promene privilegija

Primer 9. RBAC tehnika

Database selected.

create role izvestaj;
Role created.

create role svepriv;
Role created.

grant select on car_obj to izvestaj;
Permission granted.

grant select, insert, update, delete on car_obj to svepriv;
Permission granted.

grant default role izvestaj to milos;
Permission granted.

grant svepriv to milos;
Permission granted.

Database closed.

> connect to 'inis_07' user 'milos';

ENTER PASSWORD:
Connected.

select first 3 * from car_obj;

sif_car_obj naz_car_obj vrs_car_obj aktivan

11002	Beograd	CA		D	
11029	Beograd Luka		CI		D
11037	Pancevo	CI		D	

3 row(s) retrieved.

> update car_obj set aktivan = "N" where sif_car_obj = 11002;

273: No UPDATE permission.
Error in line 1
Near character position 15

> set role svepriv;

Role set.

```

>
> update car_obj set aktivan = "N" where sif_car_obj = 11002;

1 row(s) updated.

> set role izvestaj;

Role set.

> update car_obj set aktivan = "D" where sif_car_obj = 11002;

273: No UPDATE permission.
Error in line 1
Near character position 16
>

```

13.2 (LBAC)

Pristup baziran po osnovu oznake - labele

Kontrola pristupa bazirana na oznaci (LBAC) uveliko povećava kontrolu nad tim ko može pristupiti podacima. LBAC omogućava da se tačno odredi ko ima pristup za pisanje, a tko pristup za čitanje na pojedinačnim redovima i kolonama. LBAC sposobnost je vrlo podesiva: možete je konfigurisati da posebno odgovara postavljenoj sigurnosnoj okolini. Administrator sigurnosti, korisnik kojem je dodijeljeno DBSECADM ovlaštenje, obavlja čitavu LBAC konfiguraciju. Administrator sigurnosti konfigurira LBAC sistem kreiranjem politika sigurnosti.

Politika sigurnosti opisuje kriterijume koji se koriste za određivanje ko ima pristup određenim podacima. Samo jedna politika sigurnosti se može koristiti za zaštitu bilo koje tabele, ali različite tablele mogu biti zaštićene različitim politikama sigurnosti.

Politika sigurnosti je sadržana iz više sigurnosnih komponenti. Postoje tri tipa kakvih komponenti: polja, setovi i hijerarhijska drva.

Sigurnosna komponenta polje u sebi sadrži listu vrsta pristupa po sigurnosnom nivou koje na primer može biti: tajno, lokalno i javno, gde je hijerarhijski najdominantniji nivo.

Primer 10. Sigurnosna komponenta polje

```

CREATE SECURITY LABEL COMPONENT nivo ARRAY ['Tajno', 'Lokalno',
'Javno'];

```

Sigurnosna komponenta set predstavlja sigurnosne nivoe grupisane oko pojedinih operativnih grupa kao što je na primer proizvodnja, prodaja, marketing itd.

Primer 11. Sigurnosna komponenta set

```
CREATE SECURITY LABEL COMPONENT odeljenje SET {'Proizvodnja',
'Prodaja', 'Marketing'};
```

Sigurnosna grupa hijerarhijsko drvo predstavlja listu elemenata grupisanu u organizacione celine što preneseno na preduzeće predstavlja njegovu organizacionu šemu.

Primer 12. Sigurnosna komponenta hijerarhijsko drvo

```
CREATE SECURITY LABEL COMPONENT preduzece TREE ('Direktor' Root,
'Informatika' under 'Direktor', 'Finansije' under 'Direktor', 'Personalno'
under 'Direktor', 'Racunovodstvo' under 'Finansije', 'Investicije'
under 'Finansije');
```

Primer 13. Kreiranje politike sigurnosti

```
CREATE SECURITY POLICY kompanija COMPONENTS nivo, odeljenje,
preduzece WITH IDSLBACRULES;
```

Nakon kreiranja politike sigurnosti, administrator sigurnosti kreira oznake sigurnosti koje su deo te politike. Šta je značenje oznake sigurnosti određuje politika sigurnosti i može se konfigurisati da predstavlja kriterijum kojim se određuje ko ima pristup određenim stavkama podataka u procesu poslovanja

Primer 14. Kreiranje oznake sigurnosti

```
CREATE SECURITY LABEL kompanija.uprava COMPONENT nivo 'Tajno',
COMPONENT odeljenje 'Proizvodnja', 'Prodaja',
COMPONENT preduzece 'Informatika';
```

Nakon što se kreira, oznaka sigurnosti se može pridružiti pojedinim stupcima i redovima u tabeli da bi se zaštitili podaci koji su tamo sadržani. Podaci koji su zaštićeni od strane oznake sigurnosti nazivaju se zaštićeni podaci. Administrator sigurnosti omogućava korisnicima pristup zaštićenim podacima tako da im dodeljuje sigurnosne oznake. Kada korisnik pokuša pristupiti zaštićenim podacima ta oznaka sigurnosti korisnika upoređuje se s oznakom sigurnosti koja štiti podatke. Zaštitna sigurnosna oznaka će blokirati neke korisničke sigurnosne

oznake, dok druge neće blokirati.

Primer 15. Dodeljivanje oznake

```
GRANT SECURITY LABEL kompanija.uprava TO 'nikola' FOR READ ACCESS;  
GRANT SECURITY LABEL kompanija.uprava TO 'marko' FOR WRITE ACCESS;  
GRANT SECURITY LABEL kompanija.uprava TO 'milos' FOR ALL ACCESS;
```

Administrator sigurnosti takođe može dodijeliti izuzeća korisnicima. Izuzetak omogućuje pristup zaštićenim podacima kojima bi u suprotnom oznaka sigurnosti onemogućila.

Primer 16. Izuzeci

```
GRANT EXEMPTION ON RULE IDSLBACREADSET FOR kompanija TO 'milos';  
GRANT EXEMPTION ON RULE ALL FOR kompanija TO 'milos';
```

Zajedno se oznake sigurnosti i izuzeci nazivaju LBAC setom prava. Ako se želi pristupiti zaštićenim kolonama, a LBAC set prava ne dopuštaju pristup, tada pristup neće uspeti i dobiće se poruka greške. Ako se želi pročitati zaštićen red za koje LDAC ne dopušta da se pročita tada se DBMS ponaša kao da ti redovi ne postoje. Ovi redovi se ne mogu izabrati kao deo bilo kojeg SQL izraza koji se izvodi, uključujući SELECT, UPDATE i DELETE. Čak i agregatne funkcije ignorišu redove koje LBAC set prava ne daje da se pročitaju. Na primjer, funkcija COUNT(*) će vratiti broj samo onih redova za koje se ima pristup čitanja.

Sa LBAC kontrolom administratori imaju mogućnosti da kontrolišu pristup podacima u smislu čitanja i menjanja istih do nivoa reda i kolone zapisa u tabeli čime se dobija izuzetno precizan instrument zaštite podataka u bazi.

14 Šifrovanje

Za razliku od većine drugih relacionih baza, Informix poseduje set funkcija čija uloga se svodi na šifrovanje i dešifrovanje podataka smeštenih u bazi.

Ovakav pristup postavlja ove funkcije u deo fizičkog nivoa zaštite podataka kojim se sprečava pristup i čitljivost podataka vam ograničenja postavljenih u sistemu za upravljanje relacionom bazom podataka (RDBMS).

U današnje vreme razvoja višeslojno programiranih aplikacija i korišćenje internet pristupa kao veze za komunikaciju klijenata sa bazom, izuzetno je važno da se sigurnost podataka postavi na visok niv kroz različite pristupe unutar sistema zaštite.

Ako se organizuje smeštanje podataka na više servera po sistemu replikacije podataka, i ako se koriste javni (internet) putevi za prenos podataka, tada oba mogućnost daje visok nivo zaštite u prenosu. Kontrolu nad šifrovanjem i dešifrovanjem u tom slučaju imaju data serveri. Nakon uključivanja opcije šifrovanja, to sam sistem prepoznaje i drugi server izvršava potrebne radnje bez obzira da li je na njemu opcija šifrovanja/dešifrovanja aktivirana.

Naša intranet mreža, kao što je već pomenuto, strukturno u većini koristi Telekomovu Frame Relay strukturu koja je već po sebi zatvorena i zaštićena sistemom tuneliranja pa time zaštićeni podaci u prenosu nisu imali potrebu da se i dodatno šifriraju.

Preuzimanjem vlasništva nad bazom od strane IBM-a, došlo je do usklađivanja sa standardima koje je donela američka agencija za standarde u obradi informacija (Federal Information Processing Standards - FIPS) te je zamenjen interni sistem ta šifrovanje sa OpenSSL biblioteka na standardne IBM biblioteke za šifrovanje (ICC).

14.1 Oblast šifrovanja

Šifrovanje podataka se primanjuje na svaku pojedinačnu kolonu u svakoj tabeli u bazi. Sam proces se inicira unutar SQL izraza.

Sve vrste podataka u kolonama moraju biti šifrovane putem funkcija koje su obezbeđene od strane korisnika, jedino BLOB (binary large object) podaci koriste ugrađene funkcije.

Šifrovanje može da se primeni na deo ili celu kolonu isto tako i na deo ili sve redove unutar bilo koje kolone pa se time dobija uticaj na pojedine ćelije po izboru korisnika.

Primena ove mogućnosti može da se hipotetički predstavi i unutar našeg preduzeća. Po internim pravilnicima, plata je tajna i ne sme da se diskutuje o njenoj visini. I pored toga, dešava se da nezadovoljni zaposleni razmenjuju informacije o visini ličnog dohotka. U slučaju da je jedan od tih nezadovoljnih zaposlenih dovoljno informatički edukovan ili pripadnik službe za informatiku, dolazi do mogućnosti da po osnovu prava pristupa, ima uvid u plate posloводства. Uzmimo za primer da su te plate preko 100.000,00 dinara i da je potrebno takve podatke (preko tog iznosa) u svakom redu u određenoj koloni u datoj tabeli šifrovati. Postojeći set funkcija to omogućava.

Šifrovanje cele baze ili svih podataka unutar tabele putem jedne funkcije nije moguće, već je potrebno upotrebiti poglede ili sačiniti pojedinačne SQL izraze da bi se šifrovali podaci.

14.2 Šifrovanje i dešifrovanje

14.2.1 Osnovna tehnika šifrovanja i dešifrovanja

Osnovna tehnika je direktno odabiranje željene funkcije za šifrovanje kojom se obrađuje podatak.

U primeru 17 ažurira se tabela cust_calls u bazi stores9 koja već sadrži podatke, sa novom kolonom. Tada se puni kolona sa šifrovanim podatkom user_id reda i prikazuje sadržina tabele za nešifrovanu vrednost user_id, šifrovanu vrednost u vidljivom (sirovom) obliku i dešifrovanu vrednost šifrovane vrednosti user_id.

Upotreba funkcije SET ENCRYPTION PASSWORD uprošćava proces pošto se na taj način postavlja jedinstvena šifra pristupa kojom se izbegava greška pri svakom ukucavanju. Upotrebljena je funkcija ENCRYPT_AES(), pored koje postoji i funkcija ENCRYPT_TDES(). Obe su iste što se tiče argumenata koje koriste, razlika je jedino u algoritmima koje koriste.

Nedostatak korištenja ove osnovne tehnike je jednoznačnost šifre pristupa za veći broj procesa i korisnika.

Primer 17. Osnovno šifrovanje i dešifrovanje podataka

```
C:\idsdemos\stores9>dbaccess -e stores9 example1
```

Database selected.

```
alter table cust_calls add enc_id char(200);
Table altered.
```

```
set encryption password "erewhon" with hint "gulliver";
Encryption password set.
```

```
update cust_calls set enc_id = encrypt_aes(user_id);
7 row(s) updated.
select first 2
```

```
    user_id as plain_user_id
    , enc_id as enc_id
    , decrypt_char(enc_id) as decrypted_user_id
from cust_calls
;
```

```
plain_user_id      richc
enc_id v0AQAAAAEAW4Xcnj8bH292ATuXI6GllGA8yJJviU9JCUq9ngvJkEP+/BzwFhGSYw==
decrypted_user_id  richc
```

```
plain_user_id      maryj
enc_id 0gH8QAAAAEASv+pWbie/aTjzqckgdHCGhogtLyGH8vGCUq9ngvJkEP+/BzwFhGSYw==
decrypted_user_id  maryj
```

2 row(s) retrieved.

Database closed.

U sledećem primeru 18 je prikazan kompleksniji metod šifrovanja kojim u tabeli call_type u bazi stores9 šifriramo više kolona sa korištenjem kompleksne šeme funkcija za šifrovanje i samih šifara pristupa.

U tabelu se dodaju dve kolone i one se zajedno sa dve postojeće šifriraju. Koriste se dva različita algoritma za šifriranje za različite redove gde pojedini redovi imaju sopstvene jedinstvene šifre pristupa. Preostali redovi koriste predefinisane šifre pristupa postavljenu već pomenutom funkcijom SET ENCRYPTION PASSWORD. Ovaj deo treba da nam da do znanja da šifra pristupa koja je postavljena kroz pozivnu funkciju nadređuje predhodno postavljenu generalno postavljenu šifru pristupa.

Treba obratiti pažnju i biti izrazito oprezan pri korištenju ove tehnike pošto da bi se povratili podaci, potrebno je da se specifično navede odgovarajuća šifra pristupa za pojedini red. Posebno se ističe problem koji proizilazi da unutar jedne SELECT naredbe može da se definiše samo jedna šifra pristupa pa iz toga proizilazi da se mora primeniti toliko SELECT naredbi koliko ima različitih šifri pristupa. Ovo se prikazuje unutar prve dve SELECT naredbe u primeru 18. Da bi pristupili više redova mora se upotrebiti naredba UNION za pojedine rezultate.

Primer 18. Kompleksno šifrovanje i dešifrovanje

```
C:\idsdemos\stores9>dbaccess -e stores9 example2
```

Database selected.

```
ALTER TABLE call_type ADD enc_code char(99);  
Table altered.
```

```
ALTER TABLE call_type ADD enc_descr char(150);  
Table altered.
```

```
SET ENCRYPTION PASSWORD 'erewhon' WITH HINT 'gulliver';  
Encryption password set.
```

```
UPDATE call_type SET enc_code =  
CASE  
    WHEN call_code = 'B' then encrypt_aes('B', "Michael")  
    WHEN call_code = 'D' then encrypt_aes('D', "Terrell")  
    WHEN call_code = 'I' then encrypt_tdes('I', "Norfolk")  
    WHEN call_code = 'L' then encrypt_aes('L')  
    WHEN call_code = 'O' then encrypt_aes('O')
```

```
END;  
6 row(s) updated.
```

```
SET ENCRYPTION PASSWORD 'whosonfirst' WITH HINT 'abbotcostello';  
Encryption password set.
```

```
UPDATE call_type SET enc_descr = ENCRYPT_TDES(code_descr);  
6 row(s) updated.
```

```
SELECT DECRYPT_CHAR(enc_code, "Michael") AS decrypted_code  
FROM call_type  
WHERE call_code = 'B'
```

```
;  
decrypted_code B
```

1 row(s) retrieved.

```
SELECT DECRYPT_CHAR(enc_code) AS decrypted_code  
FROM call_type  
WHERE call_code = 'B'
```

```
;  
26008: The internal decryption function failed.  
Error in line 27  
Near character position 20
```

```
SELECT DECRYPT_CHAR(enc_code, "erewhon") AS decrypted_code
```

```

, DECRYPT_CHAR(enc_descr) AS decrypted_description
FROM call_type
WHERE call_code = 'L'
;
decrypted_code L
decrypted_descr+ late shipment
1 row(s) retrieved.

```

14.2.2

Povrat šifrovanih binarnih podataka

U predhodna dva primera (17 i 18), videli smo način kako se dešifriraju podaci uz upotrebu funkcije `decrypt_char()` koja po šifrovanom podatku tipa string zna kako da dešifruje podatak i vraća podatak iste dimenzije kakav je bio pre šifrovanja zbog čega ne treba posebno navoditi format podatka pri šifrovanju.

Posebno se jedino ponaša dešifrovanje binarnih - BLOB podataka koji se šifruju na isti način kao i druge vrste podataka. Da bi se dešifrovali BLOB podaci, upotrebljava se funkcija `DECRYPT_BINARY()` čija upotreba se prikazuje u sledećem (19) primeru gde se tabela `catalog` unutar baze `store9` proširuje sa dve nove kolone.

Primer 19. Dešifrovanje binarnih podataka

```
C:\idsdemos\stores9>dbaccess -e stores9 example9
```

```
Database selected.
```

```
ALTER TABLE catalog ADD adstuff BLOB;
Table altered.
```

```
ALTER TABLE catalog ADD enc_stuff BLOB;
Table altered.
```

```
UPDATE catalog
SET adstuff = FILETOBLOB ('c:\informix\gif','server')
WHERE catalog_num =10031
;
1 row(s) updated.
```

```
SET ENCRYPTION PASSWORD "Erasmus";
Encryption password set.
```

```
UPDATE catalog
SET enc_stuff = ENCRYPT_AES (adstuff)
WHERE catalog_num =10031
```

```
;
1 row(s) updated.

SELECT catalog_num
      , LTOFILE(DECRYPT_BINARY(enc_stuff), 'c:\informix\gif2','server')
FROM catalog
WHERE catalog_num = 10031
;
catalog_num 10031
(expression) c:\informix\gif2.0000000044be7076

1 row(s) retrieved.

Database closed.
```

14.2.3

Indeksiranje šifrovanih podataka

Kao se koriste šifrovani podaci za indeksiranje tabele tada ne postoji efikasan načina da se korektno uporede te vrednosti sa datom vrednošću unutar SQL upita jer šifrovanje podatka iz tabele nastaje pre njegovog smeštanja u bazu.

Postoji još jedan problem koji ćemo elaborirati u primeru 20. U pripremi za ovaj prikaz u bazi stores9 u tabeli stock, dodata je kolona e_status koja će sadržati šifrovani sadržaj kolone status koji je sadržan od vrednosti null i vrednosti 'A'. Time su određene samo dva različita stanja atributa u toj koloni (sa jednom vrednošću) što se korišćenjem naredbe DISTINCT i pokazalo. Posle šifrovanja i pri korišćenju iste naredbe nad šifrovanim podacima, dobijamo rezultat koji je jednak broju redova unutar tabele i koji nam govori da su i null stanja atributa (bez vrednosti) pri šifriranju dobila neku vrednost. To se dešava jer se pri svakom šifrovanju dobija druga vrednost i time indeks postaje neupotrebljiv za pronalaženje specifične vrednosti unutar šifrovane indexne kolone.

Ako je indeks neophodan, tada je rešenje da se pored šifrovane kolone ista vrednost pohrani kao nešifrovana u drugu kolonu i da se nad njom uvedu drugi načini zaštite opisani u predhodnim poglavljima iako to tada umanjuje vrednost zaštite zbog čega se prvenstveno i krenulo sa šifrovanjem podataka.

I na kraju moram spomenuti da se indeks može formirati nad šifrovanom kolonom bez ukakve poruke o grešci od strane sistema.

Primer 20. Šifrovani indeksi

```
kodiak:/home/informix $ dbaccess stores9 -
```

```
Database selected.
```

```
> info columns for stock;
```

Column name	Type	Nulls
stock_num	smallint	yes
manu_code	char(3)	yes
...		
status	char(1)	yes
...		
e_manucode	char(36)	yes
e_unit	char(35)	yes
e_desc	char(43)	yes
e_manuitem	char(43)	yes
e_status	char(35)	yes
e_bigger	char(35)	yes

```
>
```

```
> select count(distinct status) from stock;
      (count)
      1
```

```
1 row(s) retrieved.
```

```
> select count(distinct e_status) from stock;
      (count)
      44
```

```
1 row(s) retrieved.
```

```
> select count(*) from stock where status is not null;
      (count(*))
      44
```

```
1 row(s) retrieved.
```

```
> create index estat on stock(e_status);
Index created.
```

14.3 'pogleda'

Sakrivanje algoritma šifrovanja pomoću

Jedan od problema pri pristupu šifrovanim podacima nastaje i pri promeni SQL upita ako on sadrži i funkciju za dešifrovanje sa parametrima zbog mogućih grešaka pri ukucavanju, nesigurnog prenosa SQL upita do baze itd.

Ovo se prevazilazi sačinjavanjem 'pogleda' nad tabelom. Primenom ove tehnike sakriva se funkcija kojom šifriramo podatke. Rečeno se prikazuje u primeru 21 gde se prerađuje tabela cust_calls gde je kolona user_id šifrovana i gde se postojeći SQL izraz ne mora menjati. Preimenovana je tabela i napravljen 'pogled' sa predhodnim imenom te tabele u kojem je upotrebljena funkcija za dešifrovanje DECRYPT_CHAR da bi se dešifrovana vrednost kolone user_id pojavila na pravom mestu.

Ograničenja su u već pomenutim karakteristikama korištenja izraza SET ENCRYPTION PASSWORD. Da bi koristili višestruke pristupne šifre za višestruko šifrovanje i povrat tih podataka, moraju se koristiti stor procedure kojima bi se za povrat podataka pristupne šifre slale kao parametri.

Primer 21. 'Pogledi' nad tabelom u cilju sakrivanja algoritma za dešifrovanje

```
C:\idsdemos\stores9>dbaccess -e stores9 example7
```

```
Database selected.
```

```
RENAME TABLE cust_calls TO cust_calls_table;  
Table renamed.
```

```
CREATE VIEW cust_calls ( customer_num  
                        , call_dtime  
                        , user_id  
                        , call_code  
                        , call_descr  
                        , res_dtime  
                        , res_descr )
```

```
AS
```

```
SELECT customer_num  
      , call_dtime  
      , DECRYPT_CHAR (enc_id)  
      , call_code  
      , call_descr  
      , res_dtime  
      , res_descr
```

```
from cust_calls_table
```

```
;
```

```
View created.
```

```
SET ENCRYPTION PASSWORD "erewhon";  
Encryption password set.
```

```
select * from cust_calls;
```

```

customer_num      119
call_dtime        1998-07-01 15:00
user_id           richc
call_code         B
call_descr        Bill does not reflect credit from previous order
res_dtime         1998-07-02 08:21
res_descr         Spoke with Jane Akant in Finance. She found the error and is
send              ing new bill to customer

customer_num      121
call_dtime        1998-07-10 14:05
user_id           maryj
call_code         O
call_descr        Customer likes our merchandise. Requests that we stock
more types       of infant joggers. Will call back to place order.
res_dtime         1998-07-10 14:06
res_descr         Sent note to marketing group of interest in infant joggers
....

customer_num      110
call_dtime        1998-07-07 10:24
user_id           richc
call_code         L
call_descr        Order placed one month ago (6/7) not received.
res_dtime         1998-07-07 10:30
res_descr         Checked with shipping (Ed Smith). Order sent yesterday- we
were             waiting for goods from ANZ. Next time will call with delay if
                necessary.

7 row(s) retrieved.

Database closed.

```

14.4

Pristupna šifra

14.4.1

Upotreba pristupne šifre

Kako je već poneuto u predhodnom izlaganju, pomoću izraza SET ENCRYPTION PASSWORD se može postaviti predefinisana pristupna šifra ili se

može primeniti specifična pristupna šifra pojedinačno za svaku kolonu ili grupu kolona.

Korišćenjem pristupne šifre kroz SQL upit, predhodno predefinisana pristupna šifra SET ENCRYPTION PASSWORD nema više efekta. Ako se pri upitu ne obezbedi pristupna šifra a ista nije ni predhodno definisana SET ENCRYPTION PASSWORD izrazom, sistem vraća izveštaj o grešci.

14.4.2 Šifrovanje pristupne šifre tokom transporta

Korišćenje korisničkog naloga i pristupne šifre je uvek predstavljalo naprednu tehniku korišćenja pri autentifikaciji naročito sa pozicije kada se isti šifriraju unutar istog operativnog sistema.

U slučaju korišćenja istog seta podataka i za autentifikaciju pri pristupu bazi od strane aplikacije, korisnički nalog i pristupna šifra se transportuju unutar SQL upita ka DBMS-u kao prosti tekst što nalaže da se oni moraju u toj fazi zaštititi šifrovanjem

U konfiguracionim fajlovima sa obe strane transporta (klijent-server, server-server) moraju se postaviti isti parametri za šifrovanje. To se odnosi na SQLHOSTS i concsn.cfg fajl sa obe strane i koji se najčešće nalaze u \$INFORMIXDIR/etc direktorijumu. Concsn.cfg fajl se može nalaziti i na drugom mestu, naročito na klijentu ali se tada u setovaju okruženja u parametru INFORMIXCONCSM specificira putanja do njega.

Sa strane klijenta (aplikacije) treba oratiti pažnju na setovanje odgovarajućeg podatka za parametre DBSERVERNAME i DBSERVERALIAS. Pri višestukin korisničkim nalogima, neki korisnici mogu imati a drugi ne šifrovane podatke za autentifikaciju, zavisno od potreba i administracije sistema

U primeru 22 je prikazan primer setovanja za SQLHOSTS i CONCSM.CFG. U slučaju da setovanja nisu u redu sistem šalje poruku o grešci broj 930, "Cannot connect to database server (servername)."

Primer 22. Primer setovanja SQLHOSTS i CONCSM.CFG za šifrovanje pristupne šifre

```
sqlhosts entries:  
kod_auth1 onsoctcp kodiak kod_auth1 s=0  
kod_auth2 onsoctcp kodiak kod_auth2 r=1  
kod_auth3 onsoctcp koidak kod_auth3 csm=(SPWDSCSM)  
  
concsn.cfg entry;  
SPWDSCSM("/usr/informix/lib/csm/libixspw.so", "", "")
```

14.4.3

Broj i oblik pristupnih šifri

Za običnog korisnika administriranje sopstvene pristupne šifre je dovoljno traumatično, naročito ako ih ima više od jedne i ako se često menjaju po taktu koji daje sam sistem. Rad sa pristupnim šifarama je u mnogome još kompleksniji imajući u vidu broj pristupnih šifara koji u može preći i broj tabela u bazi koji danas predstavlja pozamašnu cifru u popularnim ERP rešenjima. Osnovna ideja vodilja je da se izbalansira između kompleksnosti šifara i njihovog broja i dovoljno obezbeđene sigurnosti podataka.

IDS ne postavlja nikakve posebne zahteve kada se tiče forme pristupne šifre, jedino što ograničava jeste da broj karaktera ne treba da ne bude manji od šest (6) i ne veći od stodvadesetiosam (128).

Sam proizvođač u svojoj dokumentaciji nudi četiri modela pri razmišljanju o načinu primene pristupnih šifara što rezultira različitim brojem pristupnih šifara na nivou baze:

1. Prvi pristup nas navodi na razmišljanje da su pristupne šifre potrebne samo za one tabele koje sadrže šifrovane podatke. Na taj način možemo imati pristupnih šifara koliko i tabela u bazi ali je to malo verovatno pošto nisu svi podaci takvog tipa da treba da se šifriraju tako da u ovom pristupu broj pristupnih šifara bi bio relativno mali..
2. Sledeći pristup se vezuje za primenu pristupnih šifara po osnovu svake uloge – role u okviru preduzeća tj kako smo u predhodnom pektu već pominjali, i time i u okviru baze. Pošto svaka rola ima tačno određene tabele koje koristi, s te pozicije ne bi bilo problema da se definiše konačan broj pristupnih šifara. Problem nastaje kada role međusobno imaju tabele koje se zajednički koriste i kada treba da se koristi veći broj pristupnih šifara u istim vremenskim momentima, tada može doći do problema u radu baze.
3. Razmišljanje o postavljanju pristupnih šifara za svaku šifrovanu kolonu pojedinačno ima smisla kod malih baza, dok u suprotnom to može biti izuzetno veliki broj pristupnih šifara.
4. Na kraju nas razmišljanje dovodi do osnove organizacionog projektovanja baze. IDS nam nudi opciju da određene tabele grupišemo funkcionalno u slojeve (layers) što je idealno za administraciju same baze. Takvo 'prirodno' grupisanje odvaja u posebnu grupu zajedničke tabele (problem u predlogu br. 2) i time se u grupi ostavljaju tabele vezane za određene grupe aktivnosti unutar preduzeća. Time se dobija konačan i dosta mali broj pristupnih šifara.

14.4.4

Isticanje važnosti pristupne šifre

IDS ne primorava korisnike da menjaju šifru niti obezbeđuje sistematizovan način na koji se to može uraditi. Ako se koristi autentifikacija operativnog sistema za rad sa bazom i taj sistem ima proceduru vrenemske promene šifre, potrebno je obezbediti način da se i dalje radi sa podacima sa novom šifrom. Bez adekvatne šifre podaci se ne mogu povući iz baze ni na koji način. Zbog toga je nužno da se obezbedi promena šifre pre njenog isteka..

Kao što je već pominjano, SET ENCRYPTION PASSWORD izraz može da koristi samo jednu šifru – staru ili novu – pa tako dolazimo do toga da se mora naći način kao uključiti i drugu šifru pri svakom UPDADE izrazu.

U primeru 23 se prikazuje način kako, ne upotrebljavajući izraz SET ENCRYPTION PASSWORD vršimo izmenu podataka.

<i>Primer 23. Promena šifre</i>
<pre>update call_type set enc_code =encrypt_aes(decrypt_char(enc_code, "Dennis"), "Erasm") where call_code = 'X';</pre>

Predhodni primer 23 nije praktičan u upotrebi zbog toga što se njime menja samo jedan red, dok je mnogo praktičnije primeniti metod kojim se menjaju šifre u svim zapisima u tabeli ili grupama tabela što je prikazano u primeru 24. Primer vrši izmenu u tabeli sales_rep stores9 baze podataka.

<i>Primer 24. Višeredna izmena šifre</i>
<pre>C:\idsdemos\stores9>dbaccess -e stores9 example10.sql Database selected. ALTER TABLE sales_rep ADD enc_name ROW(fname LVARCHAR, lname LVARCHAR); Table altered. SET ENCRYPTION PASSWORD "Ricardo" WITH HINT "Vancouver"; Encryption password set. UPDATE sales_rep SET enc_name = ROW(ENCRYPT_TDES(name.first), ENCRYPT_TDES(name.last)); 2 row(s) updated. SELECT * FROM sales_rep; rep_num 101 name ROW('Peter','Jones')</pre>

```

region_num 6
home_office t
sales      SET{ROW('1997-03','$47.22'),ROW('1997-04','$55.22')}
commission 0.03000
enc_name
    ROW('1lmkQAAAACAxE6fSstEGWCeDn/qDRyR0zZltX4YuOU9go5dXfYD/ZY=', '1C/
      QQAAAACA62TTuNf9q8csJ0hyVB8wQTZltX4YuOU9go5dXfYD/ZY=')

rep_num    102
name       ROW('John','Franklin')
region_num 6
home_office t
sales      SET{ROW('1997-03','$53.22'),ROW('1997-04','$18.22')}
commission 0.03000
enc_name
    ROW('1qvYQAAAACAfYXfzOuyd2CJJkqhpkLVPjZltX4YuOU9go5dXfYD/ZY=', '1hH
      sQAAAAEAz6TfNs9/MsWxEqEzlxMYtNwst7RWg8hzNmW1fhi45T2Cj1d9gP9lg==')

```

2 row(s) retrieved.

```

SELECT DECRYPT_CHAR(enc_name.fname) || ' ' ||
DECRYPT_CHAR(enc_name.lname) as NAME
FROM sales_rep

```

```

name      Peter Jones
name      John Franklin

```

2 row(s) retrieved.

```

SET ENCRYPTION PASSWORD "DenverLad" WITH HINT "Karl";
Encryption password set.

```

```

update sales_rep
set enc_name =
ROW(encrypt_tdes(decrypt_char(enc_name.fname, "Ricardo"))
    ,encrypt_tdes(decrypt_char(enc_name.lname, "Ricardo"))
    )
;

```

2 row(s) updated.

```

SELECT DECRYPT_CHAR(enc_name.fname) || ' ' ||
      DECRYPT_CHAR(enc_name.lname) as NAME
FROM sales_rep
;

```

```

name      Peter Jones
name      John Franklin

```

2 row(s) retrieved.

Database closed.

14.4.5 Upotreba asocijacija za pristupnu šifru

U svim sistemima gde postoji sistem autentifikacije postoji i metod koji omogućava pomoć pri pokušaju prisećanja potrebne šifre. I u IDS-ovom sistemu funkcija za šifrovanje omogućava da se pri setovanju šifre uključi i asocijacija na šifru koja se koristi prilikom potrebe da se šifra povraća. GETHINT() funkcija to radi i njena upotreba je prikazana u primeru 25.

U ovom primeru se odabire jedan zapis (red), ako obrađuju više redova i vrši prikaz njihovih asocijacija, svaka će biti prikazana. Skraćivanje postupka, ako se zna da je više redova sa istom šifrom, količinu prikaza možemo smanjiti korištenjem naredbe DISTINCT. Ako se koristi GETHINT() funkcija u slučaju kada asocijacija nije definisana, sistem vraća null vrednost.

Primer 25. GETHINT() funkcija

```
> select * from call_type where call_code = 'X';
```

call_code	X
code_descr	Test of a lengthy description.
enc_code	
enc_descr	1bZIQAAAAIAP4PoYtU4yma3rWOVDFEoDKmHK62JYMtIEJPPR3MGDXXiR/0mjlcMulDP 8EVcR0PsohMB0ZQaB2w=

1 row(s) retrieved.

```
> select gethint(enc_descr) from call_type where call_code = 'X';
```

(expression)

abbotcostello

1 row(s) retrieved.

14.5 Veličina šifrovanog podatka

14.5.1 Određivanje veličine šifrovanog podatka

Po svojoj prirodi, šifrovani podatak zahteva više prostora nego kada je nešifrovan, što se može relativno lako izračunati za podatke sa fiksnom dužinom. Za podatke sa promenljivom dužinom jedini način je da se proceni najduži podatak i prema tome da se odabere. U svakom slučaju, dužina šifrovanog podatka zavisi direktno od pristupne šifre komom se šifrira kao i od asocijacije ako se i ona koristi.

Ako se podatak smešta u prostor koji je manji od potrebnog, sistem neće reagovati ni upozorenjem niti prikazivanjem izveštaja o grešci. LENGTH funkcija obezbeđuje zgodan način kako možemo da direktno proračunamo potrebe za prostorom konkretnog podatka, što se i prikazuje u primeru 26.

Primer 26. Određivanje veličine šifrovanog podatka upotrebom funkcije LENGTH()

```
execute function length(encrypt_tdes('1234567890123456', 'simple password'));
```

(expression)

55

1 row(s) retrieved.

```
execute function length(encrypt_tdes('1234567890123456', 'simple password',  
'1234567890123456789012'));
```

(expression)

87

1 row(s) retrieved.

```
execute function length(encrypt_aes('1234567890123456', 'simple password'));
```

(expression)

67

1 row(s) retrieved.

```
execute function length(encrypt_aes('1234567890123456', 'simple password',  
'1234567890123456789012'));
```

(expression)

99

1 row(s) retrieved.

14.5.2**Greške kada je mali prostor za pohranu podatka**

IDS će dozvoliti da se upiše podatak bez obzira dali je on veći li ne od prostora gde se upisuje. Pri pokušaju isčitavanja će se prijaviti greška kod loše upisanog jer se stuktura šifrovanog podatka poremetila. Primer 27 pokazuje ponašanje sistema pri upisu podatka u kolonu od 87 byte-ova koja je dovoljno velika i upis od 86 byte-ova za čega nema dovoljno mesta. Kao što je rečeno a primer pokazuje, upis uspeva bez prijave greške u oba slučaja ali pri pokušaju isčitavanja podataka, kod loše upisanog se prijavljuje greška.

Primer 27. Greška – mali prostor za smeštaj podatka

```
C:\idsdemos\stores9>dbaccess -e stores9 example
```

```
Database selected.
```

```
alter table call_type drop enc_descr;  
Table altered.
```

```
ALTER TABLE call_type ADD enc_descr char(87);  
Table altered.
```

```
SET ENCRYPTION PASSWORD 'whosonfirst' with hint 'abbotcostello';  
Encryption password set.
```

```
update call_type set enc_descr = encrypt_tdes(code_descr) where call_code =  
"X";  
1 row(s) updated.
```

```
select decrypt_char(enc_code, "erewhon") as decrypted_code  
      , decrypt_char(enc_descr) as decrypted_description  
from call_type  
where call_code = 'X'  
;
```

```
decrypted_code  
decrypted_descrip+ Test of a lengthy description.
```

```
1 row(s) retrieved.
```

```
alter table call_type drop enc_descr;  
Table altered.
```

```
ALTER TABLE call_type ADD enc_descr char(86);  
Table altered.
```

```
SET ENCRYPTION PASSWORD 'whosonfirst' with hint 'abbotcostello';  
Encryption password set.
```

```
update call_type set enc_descr = encrypt_tdes(code_descr) where call_code =  
"X";
```

1 row(s) updated.

```
select decrypt_char(enc_code, "erewhon") as decrypted_code  
      , decrypt_char(enc_descr) as decrypted_description  
from call_type  
where call_code = 'X'  
;
```

26012: The internal base64 decoding function failed.

Error in line 26

Near character position 20

Database closed.

15 Backup podataka na Informix database serverima

Backup podataka je proces uzimanja i čuvanja kopije podataka sa database servera na neku vrstu medija ili drugog servera. U slučaju kvara, krađe, (ne)namernog brisanja podataka ... pomoću backupovanih podataka se izgubljeni podaci mogu vratiti. Proces vraćanja izgubljenih podataka korišćenjem backupa nazivamo restore.

U zavisnosti od toga koliki je prihvatljiv gubitak podataka, koliko je prihvatljivo vreme za restore, i kakve su mogućnosti klijenata, potrebno je odlučiti se za jednu od dole navedenih vrsta backupa. Praćenje i provera ispravnosti backupa je obaveza visokog nivoa.

Vrste backupa:

- Off-line backup na dnevnom nivou
- On-line backup
- Backup uz pomoć replikacije

15.1 Off-line backup na dnevnom nivou

Ovaj način backupa podrazumeva da se jednom dnevno (najčešće noću) kada niko od korisnika ne koristi server baze podataka, napravi export aktivnih kompletnih baza - snimanje u *.txt fajlove, sve se to kompresuje (zipuje) i prekopira na neki drugi računar i/ili medij.

Eventualni, maksimalni gubitak podataka u slučaju kvara database servera:

- je do 24 sata pošto se backup izvršava jednom dnevno.

Postupak i vreme potrebno za restore-a podataka:

- popravka servera (nekoliko minuta do nekoliko dana - zavisi od delova i od toga šta se pokvari)
- instalacija i podešavanje OS-a ako je potrebno (2 - 4 sata)
- instalacija i konfigurisanje Informix baze ako je potrebno (2 - 4 sata)
- import backupovanih baza (od 10 minuta do 2 sata po jednoj bazi podataka - zavisi koliko su velike baze)

Prednosti:

- ne zahteva poseban hardware (dodatni server)
- administracija je skoro svedena na nulu - sve se radi automatski
- export baze se može vrlo lako preneti na servere sa različitim operativnim sistemima i verzijama informixa

Mane:

- gubitak podataka u slučaju kvara na serveru je relativno velik. Sva dokumenta koja su unešena do 24 sata pre kvara su izgubljena.
- klijenti ne smeju da budu prijavljeni na bazu podataka dok se vrši backup, što može da potraje i po par sati ako su velike baze podataka, što drugim rečima znači da nije moguće na serveru raditi 24 sata dnevno (3 smene)

Opcija:

- Prenos exportovane baze na na Standby server posle pravljenja backupa, sa čime se skraćuje vreme potrebno za restore, jer se u ovom slučaju samo promeni IP adresa Standby servera i svi klijenti nastavljaju da rade na standby serveru sa podacima od trenutka od kada je prenešen poslednji backup. Za korišćenje ove opcije potrebno je da imate još jedan server - Standby.

15.2 On-line backup

Jednom nedeljno (ili po potrebi) se vrši level0 backup prilikom čega se backupuju svi dbspaceovi, znači sve baze sa stanjem od trenutka kada je pokrenut level0 backup. Na ovaj način se dobije jedan veliki fajl (ili više fajlova) kapaciteta od 2 do nekoliko desetina ili stotina GB-a u zavisnosti koliko baza je trenutno na serveru baze podataka, koji treba negde sačuvati van servera (tape uređaj, DVD-ovi, drugi server ...).

Posle ovoga se automatski vrši backup logical logova (fajlova sa opisom izvršenih transakcija) kada se log napuni, i logovi se kopiraju na neki medij/računar

Eventualni, maksimalni gubitak podataka u slučaju kvara database servera:

- svi podaci koji su unešeni pre nekoliko minuta pa do nekoliko sati, u zavisnosti od intenziteta rada. Samo transakcije u poslednjem logical log fajlu koji nije backupovan i prenešen na neki medijum su izgubljene.

Postupak i vreme potrebno za restore-a podataka:

- popravka servera (nekoliko minuta do nekoliko dana - zavisi od delova i od toga šta se pokvari)
- instalacija i podešavanje OS-a ako je potrebno (2 - 4 sata)

- instalacija i konfigurisanje Informix baze ako je potrebno (2 - 4 sata)
- restore podataka prvo level0 pa zatim redom svih logical logova, sve do poslednjeg backupovanog logical loga (od 10 minuta do nekoliko sati zavisi koliko su velike baze)

Prednosti:

- ne zahteva poseban hardware (dodatni server)
- korisnici mogu da rade na serveru dok se pravi On-line backup, tako da se na serveru može raditi 24 sata dnevno

Mane:

- potrebno je da čovek prati šta se dešava sa logovima, koliko se brzo pune, da li su backupovani ili ne, jer u slučaju da se logovi na backupuju kako treba, restore podataka će biti nemoguć

15.3 Backup uz pomoć replikacije

Postavi se HDR replikacija. Svi podaci koji se unose na Primarnom serveru se automatski prenose na Sekundarni server putem replikacije.

Eventualni, maksimalni gubitak podataka u slučaju kvara database servera:

- je od nekoliko sekundi do nekoliko minuta, gube se samo transakcije koje nisu komitovane, jer sve komitovane transakcije se odmah prenose na Sekundarni server

Postupak i vreme potrebno za restore-a podataka - tačnije Sekundarni server preuzima funkciju Primarnog:

- prebacivanje Sekundarnog servera da radi kao primarni server (jedna komanda - nekoliko sekundi)
- promena IP adrese i naziva instance informix-a, da bi setovanje na klijentima ostalo isto (do 10 minuta)

Prednosti:

- gubitak podataka je sveden na minimum - aktuelnu transakciju koja nije komitovana prilikom havarije
- korisnici mogu da rade na serveru non-stop 24 sata dnevno
- Sekundarni server je read-only, sadrži aktuelnu kopiju radne baze koja se može koristiti za izvlačenje zahtevnih izveštaja i na taj način osloboditi Primarni server

Mane:

- potrebno je da imate još jedan server - Sekundarni, koji je sličan po konfiguraciji Primarnom serveru
- morate da imate verziju Informix-a koja podržava HDR na oba servera (Enterprise, ili Workgroup + dodatak za HDR)
- Replikacija zapravo nije backup. Primer: Ako neko "slučajno" dropuje tabelu gl_knjiga, to isto će se replicirati i na Sekundarni server i nije moguće vratiti stanje pre te akcije. Zbog toga se replikacije kombinuje sa on-line backupom i/ili exportom baze

Opcija 1:

- export baze jednom dnevno na Sekundarnom serveru noću. Na ovan način možemo čuvati kompletan backup baze na dnevnom nivou, a sam export neće onemogućavati rad na Primarnom serveru pošto se vrši na sekundarnom serveru

Opcija 2:

- on-line backup na primarnom serveru, se može vršiti paralelno sa replikacijom i na taj način možemo imati sačuvane backupove baze u slučaju namernog brisanja od nekog datuma.

16

Zaključak

Sve veća potreba za sigurnošću računarskih mreža zahteva adekvatne elemente zaštite. Praksa je pokazala da zaštita na što nižem sloju OSI modela znatno poboljšava sigurnost prenosa podataka preko nesigurne računarske mreže. Ako se dodatno uzme u obzir kako je najznačajniji nivo OSI modela mrežni sloj lako se može zaključiti da se fokus sigurnosti treba napraviti na tom (trećem) sloju OSI modela.

IPsec protokoli su jednostavni i efikasni, a zaštita koju pružaju vrlo visoka. Zbog toga, i prethodno navedenih tendencija razvoja sigurnosti računarskih mreža, za očekivati je kako će u budućnosti sva mrežna komunikacija biti bazirana na IPsec protokolima.

IPsec je protokol kojem je cilj osigurati tajnost, integritet, autentifikaciju i neporecivost poruka koje razmenjuju entiteti u komunikaciji. Kako je iz prethodno izloženog vidljivo, sam protokol je prilično složen, a uspostavljanje komunikacije odvija se u više koraka.

Dodatni zahtev je taj da je za uspostavljanje IPsec kanala prethodno nužno uspostaviti IKE/ISAKMP komunikaciju. Osim toga, zbog same implementacije mogu se pojaviti problemi vezani uz NAT i fragmentaciju, što može onemogućiti uspostavljanje IPsec kanala. Postoje i neki sigurnosni problemi vezani uz korišćenje tajnih ključeva koji omogućavaju pojedine napade, no u realnim situacijama oni su veoma malo verovatni. Osim toga, ti problemi mogu se efikasno otkloniti korišćenjem kriptografije temeljene na javnim ključevima, odnosno upotrebom digitalnih sertifikata i PKI.

U realnim situacijama, često se dešava da se ne može osigurati interoperabilnost različitih IPsec implementacija, odnosno da je proizvođača teško konfigurisati da međusobno komuniciraju.

Bez obzira na sve te nedostatke, u većini situacija, pažljivim planiranjem i adekvatnom implementacijom, IPsec služi kao vrlo efikasno i robusno rešenje za uspostavljanje sigurne host-to-host komunikacije ili uspostavljanje VPN kanala između udaljenih lokacija.

O sigurnosti podataka u bazi podataka se mora voditi računa preventivno, pre moguće ekstenzivne situacije koja može da pokrene postupak ali već sa mogućim posledicama.

Bez obzira na sigurnost podataka u ostalim nivoima i kojima se podatak koristi i prenosi, osnovna zaštita podatka polazi od same baze podataka i njenog sistema za upravljanje bazom podataka (DBMS). Ako izabrana baza unutar svog DBMS-a već sadrži razrađen i usavršan sistem za obezbeđivanje sigurnosti podataka, to predstavlja viši nivo sigurnosti koji može da se iskoristi sa većom iskorišćenosti sistema i manjim angažovanjem nužnih resursa.

U predhodnom razmatranju smo videli da Informix baza ima velike mogućnosti u domenu zaštite podataka. Sve te mogućnosti, da bi se iskoristile, moraju biti inicirane sigurnosnom politikom preduzeća, ko može i šta da radi unutar IT okruženja i u postupku korištenja aplikacije koja pristupa podacima.

Ono što nije bila tema ovog rada, a tiče se zaštite podataka, i mora da se pomene jeste kontrola promene konfiguracije celog DBMS-a, alarmiranje i izveštavanje po promeni iste što informix takođe poznaje.

Dobra praksa a i zakonski akti o arhiviranju podataka predviđaju da se siguronosna kopija baze sačinjava svaki dan i da se smešta na geografski različitu lokaciju od one na kojoj se nalazi baza. U našem slučaju to se odvija postojećom Informix metodom snapshot-a koja pravi kopiju baze a predefinisani skriptovi tada taj fajl kopiraju na server na udaljenoj lokaciji unutar intraneta (drugi grad).

I pored ove osnovne zaštite unutar DBMS-a, da bi zaštita podataka bila kompletna, mora da se obezbedi siguran pristup radnoj stanici, aplikaciji kao i obezbeđivanje sigurnih prenosnih puteva između korisnika i baze podataka kao i pouzdane procedure pravljenja, provere i čuvanja siguronosnih kopija baze podataka.

17

Literatura

- [1.] Prof.dr.Čekić Š.: „*Osnovi metodologije i tehnologije izrade znanstvenog i stručnog djela*“, Sarajevo 1999.
- [2.] CERT.hr nacionalno središte za računalnu sigurnost, <http://www.cert.hr/>
- [3.] Hrvatska akademska i istraživačka mreža – CARNet, <http://www.carnet.hr/>
- [4.] *Uvod u IPsec*, Ana Kukec, FER-ZEMRIS, http://os2.zemris.fer.hr/ns/2006_kukec/akukec%20-%20SeminarOS2.pdf
- [5.] Cisco, <http://www.cisco.com/warp/public/105/IPSECpart1.html>
- [6.] Glossary for the Linux FreeSWAN project, http://www.freeswan.org/freeswan_snaps/CURRENT-SNAP/doc/glossary.html
- [7.] An Illustrated Guide to Ipsec, <http://www.unixwiz.net/techtips/iguide-ipsec.html>
- [8.] Internet FAQ Archives - Online Education, <http://www.faqs.org/faqs/>
- [9.] IETF - The Internet Engineering Task Force, <http://www.ietf.org/home.html>
- [10.] Koristenje IPsec (IP bezbjedonosni protokol), http://users.tmok.com/~sukovic/OpenBSD_serbian/faq/faq13.html
- [11.] *IBM - Data server - Informix - online processing*, <http://www-306.ibm.com/software/data/informix/>, April 2008.
- [12.] *Fakultet elektrotehnike i računarstva Zagreb*, <http://fer.hr/>, April 2008.
- [13.] *IV POŠTANSKO SAVETOVANJE - TEHNOLOŠKA BUDUĆNOST POŠTE, ZBORNIK RADOVA*, http://www.postfest.ptt.yu/savetovanje98/zbornik_radova.htm, April 2008.
- [14.] Prof.dr.Čekić Š.: „*Osnovi metodologije i tehnologije izrade znanstvenog i stručnog djela*“, Sarajevo 1999.