UNIVERZITET U NOVOM SADU
FAKULTET TEHNIČKIH NAUKA
KATEDRA ZA TELEKOMUNIKACIJE
I OBRADU SIGNALA

Jedno rešenje realizacije programske podrške za zaštitu multimedijalnog sadržaja pomoću DTCP-IP protokola

kandidat
Rade Vulin

mentor
prof. dr. Ištvan Pap

April 2013

UNIVERZITET U NOVOM SADU ● FAKULTET TEHNIČKIH NAUKA
21000 NOVI SAD, Trg Dositeja Obradovića 6

KLJUČNA DOKUMENTACIJSKA INFORMACIJA

Redni broj, RBR:			
Identifikacioni broj, IBR:			
Tip dokumentacije, TD:		Monografska publikacija	
Tip zapisa, TZ:		Tekstualni štampani materijal	
Vrsta rada, VR:		Diplomski-master rad	
Autor, AU:		Rade Vulin	
Mentor, MN:		dr. Ištvan Pap, docent	
Naslov rada, NR:		Jedno rešenje realizacije programske podrške za zaštitu multimedijalnog sadržaja pomoću DTCP-IP protokola	
Jezik publikacije, JP:		Srpski	
Jezik izvoda, JI:		Srpski	
Zemlja publikovanja, ZP:		Srbija	
Uže geografsko područje, UGP:		Srbija, Novi Sad	
Godina, GO:		2013.	
Izdavač, IZ:		Autorski reprint	
Mesto i adresa, MA:		Trg Dositeja Obradovića 6, 21000 Novi Sad	
Fizički opis rada, FO: (poglavlja/strana/ citata/tabela/slika/grafika/priloga)		6 poglavlja, 47 strana, 31 slika, 8 tabela	
Naučna oblast, NO:		Elektrotehnika	
Naučna disciplina, ND:		Telekomunikacije	
Predmetna odrednica/Ključne reči, PO:		AES-128, AKE, DLNA, DTCP-IP, RTT.	
UDK			
Čuva se, ČU:		Biblioteci FTN–a, Novi Sad, Trg Dositeja Obradovića 6	
Važna napomena, VN:			
Izvod, IZ:		Rad prikazuje jedno rešenje realizacije programske podrške za zaštitu multimedijalnog sadržaja u kućnoj mreži. Programska podrška je zasnovana na DTCP-IP standardu koji je kreiran od strane DTLA za zaštitu sadržaja u kućnom okruženju.	
Datum prihvatanja teme, DP:			
Datum odbrane, DO:			
Članovi komisije, KO:	Predsednik:	dr Željen Trpovski	
	Član:	dr Vlado Delić	Potpis mentora
	Član, mentor:	dr Ištvan Pap	

UNIVERSITY OF NOVI SAD ● FACULTY OF TECHNICAL SCIENCES
21000 NOVI SAD, Trg Dositeja Obradovića 6

KEY WORDS DOCUMENTATION

Accession number, ANO:			
Identification number, INO:			
Document type, DT:		Monographic publication	
Type of record, TR:		Printed textual material	
Contents code, CC:		Diploma master	
Author, AU:		Rade Vulin	
Mentor, MN:		dr. Ištvan Pap, docent	
Title, TI:		One software solution for protection multimedia content with DTCP-IP protocol	
Language of text, LT:		Serbian	
Language of abstract, LA:		Serbian	
Country of publication, CP:		Serbia	
Locality of publication, LP:		Serbia, Novi Sad	
Publication year, PY:		2013.	
Publisher, PB:		Author's reprint	
Publication place, PP:		Trg Dositeja Obradovića 6, 21000 Novi Sad	
Physical description, PD: (chapters/pages/ref./tables/pictures/graphs/appendixes)		6 chapters, 47 pages, 31 pictures, 8 tables	
Scientific field, SF:		Electrical engineering	
Scientific discipline, SD:		Telecommunications	
Subject/Key words, S/KW:		AES-128, AKE, DLNA, DTCP-IP, RTT.	
UC			
Holding data, HD:		Library of Faculty of technical Sciences	
Note, N:			
Abstract, AB:		Paper presents one implementation of software solution for protection of multimedia content in home network. Softver solution is based on DTCP-IP standards, which are created by DTLA for content protection in home environment.	
Accepted by the Scientific Board on ASB:			
Defended on, DE:			
Defended Board, DB:	President:	dr Željen Trpovski	
	Member:	dr Vlado Delić	Mentor's sign
	Member, Mentor:	dr Ištvan Pap	

Sadržaj

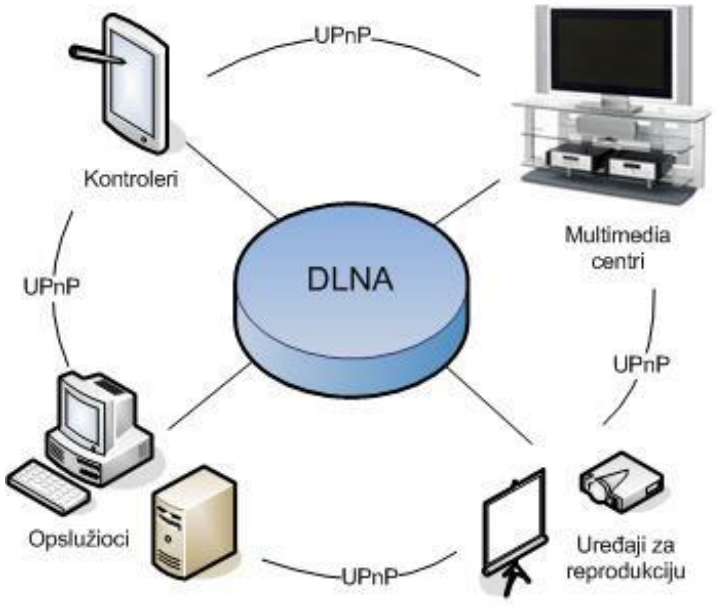
Slike

Tabele

Glava 1

Uvod

Sigurno najpoželjniji način praćenja multimedijalnog sadržaja predstavlja gledanje televizora. Postoje razni načini da se multimedijalni sadržaj sa personalnog računara prikaže na TV-u, međutim udaljavanje od TV prijemnika (tj. odlazak do računara i korišćenje miša i tastature) nije poželjno. Zahvaljujući DLNA (*Digital Living Network Alliance*) forumu [1], definisan je nov i savremen način deljenja, kontrole i reprodukcije multimedijalnog sadržaja u lokalnoj mreži, DLNA protokol stek.



Slika 1.1 DLNA okruženje

Međutim povećanjem zahteva potrošača došlo je do potrebe da se multimedijalni sadržaj zaštiti. Što više sadržaja ulazi u digitalni domen to potreba za zaštitom sadržaja postaje sve veća. Jedno od mogućih rešenja predstavlja upotreba DTCP (*Digital Transmission Content Protection*) protokola [2]. DTCP predstavlja DRM (*Digital Rights Management*), tehnologiju upravljanja digitalnim pravima, koja za cilj ima da ograniči razmenu sadržaja između kućnih digitalnih uređaja, uključujući DVD plejere i televizore, upotrebom šifrovanja sadržaja, koji se razmenjuje između uređaja.

U teoriji ova tehnologija dozvoljava razmenu multimedijalnog sadržaja između različitih uređaja, ukoliko ti uređaji podržavaju DTCP standard. DTCP standard se takođe naziva "5C" zaštita sadržaja, što upućuje na pet kompanija koje su kreirale DTCP, a to su *Hitachi, Intel, Matsushita, Sony* i *Toshiba*.

Standard je prvobitno predložen u februaru 1998. kada je "5C" predstavio CPTWG (*Copy Protection Technical Working Group*). Ovih pet kompanija naknadno u junu 1998. godine uspostavljaju DTLA (*Digital Transmission Licensing Administrator*) kako bi pojednostavili proceduru dobijanja licenci i kako bi promovisale DTCP metodu.

Rad prikazuje jedno rešenje realizacije programske podrške za zaštitu multimedijalnog sadržaja u kućnoj mreži. Programska podrška je zasnovana na DTCP-IP (*Digital Transmission Content Protection – Internet Protocol*) standardu koji je kreiran od strane DTLA kao DRM za zaštitu sadržaja u kućnom okruženju. Cilj rada predstavlja upoznavanje sa DTCP-IP protokolom i opis realizacije celokupne programske podrške.

Zaštita sadržaja upotrebom utvrđenog algoritma sprečava korišćenje sadržaja na način koji nije u skladu sa uslovima koji je propisao vlasnik sadržaja. Kriptografske tehnike predstavljaju osnovu zaštite sadržaja.

DTLA specifikacija opisuje ponašanje dva moguća uređaja a to su uređaj koji je izvor sadržaja (*Source*) i uređaj koji je potražuje sadržaj (*Sink*). Pre razmene sadržaja ovi uređaji prolaze kroz proces autentifikacije i razmene ključeva. Proces autentifikacije uređaja podrazumeva razmenu poruka između uređaja. Ove poruke nose informacije potrebne za autentifikaciju uređaja. Prilikom autentifikacije uređaji mogu da razmene RTT (*Round Trip Time*) i SRM (*System Renewability Messages*) poruke, ukoliko je potrebno.

Ukoliko su uređaji uspešno izvršili autentifikaciju i razmenu ključeva prelazi se na razmenu sadržaja. Sadržaj se šifruje na uređaju koji je izvor sadržaja, odnosno dešifruje na uređaju koji je odredište sadržaja. Šifrovanje odnosno dešifrovanje se vrši upotrebom AES-128 (*Advanced Encryption Standard*) standarda.

Programska podrška za zaštitu multimedijalnog sadržaja obezbeđuje sve potrebne funkcionalnosti koje propisuje DTCP-IP standard i realizovana je kroz DTCP-IP biblioteku. DTCP-IP biblioteka ima definisanu slojevitu strukturu. Sastoji se od tri osnovna sloja : API sloj, sloj radnih okvira i sloj servisa. Celokupna programska podrška razvijena je pomoću C programskog jezika i uspešno je prošla testiranje sa alatima koje je obezbedila DLNA organizacija.

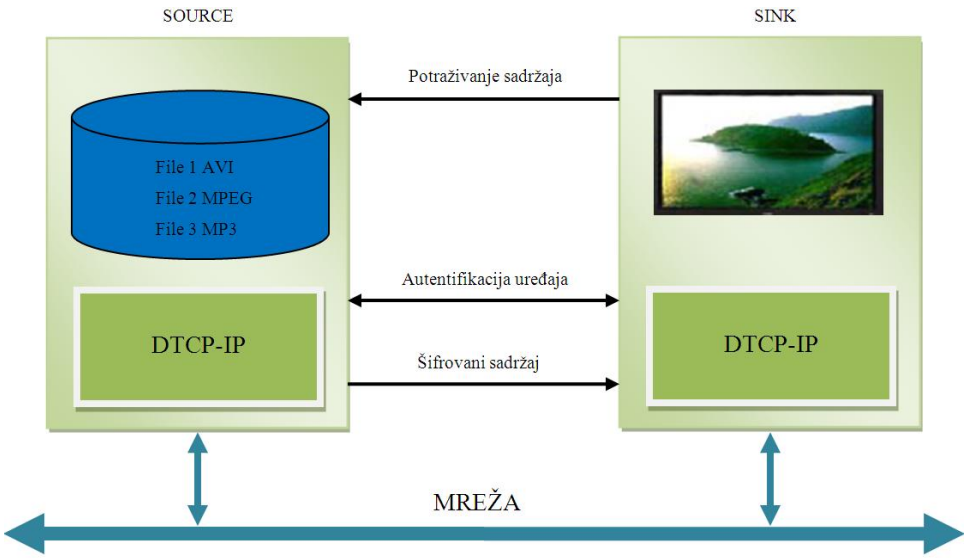
Glava 2

Povezivanje DTCP-IP-a sa DLNA

Digitalni mediji su veoma brzo usvojeni kao osnovni mediji za kućnu zabavu. Tokom poslednjih godina povećana je potreba za što kvalitetniji prenos zvuka i slike. DTCP predstavlja način zaštite multimedijalnog sadržaja koji se prenosi putem mrežnog povezivanja. DTCP sigurnosni okvir je kreiran tako da pruža mogućnost zaštite digitalnog sadržaja kao što su filmovi visokog kvaliteta, televizije koja naplaćuje svoje usluge ili muzike koja je zaštićena od neovlašćenog preuzimanja.

DTCP-IP predstavlja LPT (*Link Protection Technology*) koji je posebno prilagođen za prenos multimedijalnog sadržaja pomoću IP-a (*Internet Protocol*). DTCP-IP se koristi za potrebe DLNA tj. za zaštitu multimedijalnog sadržaja u kućnoj mreži. Implementacija DTCP-IP-a na DLNA uređaj omogućava bezbednu razmenu podataka između uređaja. Kao što DLNA sertifikovani uređaji mogu međusobno da komuniciraju i razmenjuju sadržaj, tako i DTCP-IP DLNA sertifikovani uređaji mogu međusobno da komuniciraju i razmenjuju sadržaj.

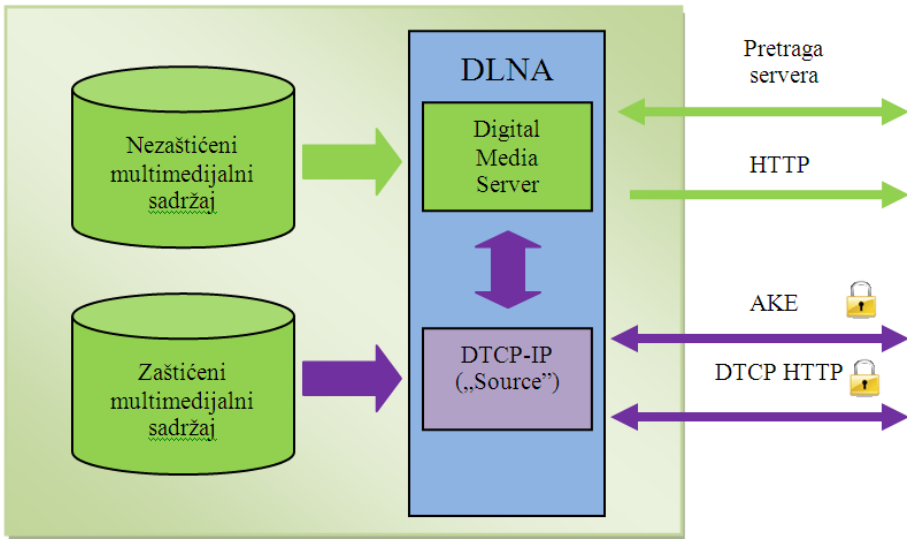
DTCP-IP tehnologija omogućava vlasnicima multimedijalnog sadržaja da zaštite sadržaj koji se prenosi sa uređaja koji predstavlja izvor multimedijalnog sadržaja (DMS – *Digital Media Server*) na kompatibilni uređaj koji je odredište sadžaja, što može da bude DMP (*Digital Media Player*) ili DMR (*Digital Media Renderer*). DMS je namenjen za deljenje multimedijalnog sadržaja DLNA kompatibilnim uređajima. DMP je namenjen za pretragu i reprodukciju multimedijalnog sadržaja deljenog na DLNA DMS kompatibilnim uređajima, dok je DMR uređaj namenjen za reprodukciju multimedijalnog sadržaja kontrolisan od strane udaljenog DLNA kompatibilnog uređaja.



Slika 2.1 DTCP-IP sistem dijagram

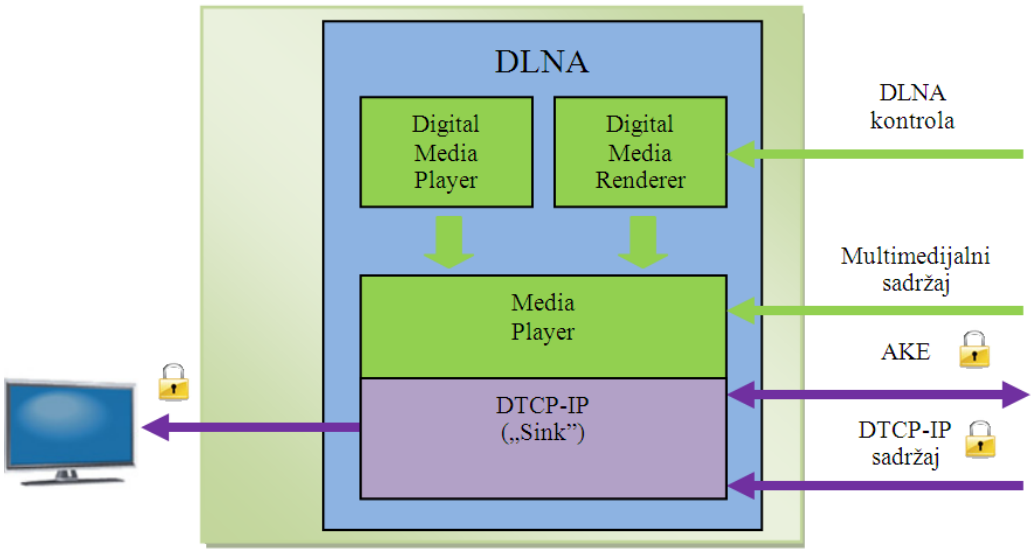
DTCP-IP tehnologija omogućava realizaciju zaštite multimedijalnog sadržaja na uređaju koji je izvor multimedijalnog sadržaja (*Source*) i realizaciju preuzimanja multimedijalnog sadržaja na uređaju koji je odredište sadržaja (*Sink*). Na uređaju koji je izvor sadržaja se vrši šifrovanje sadržaja, dok se na uređaju koji je odredište sadržaja vrši njegovo dešifrovanje. Uređaji međusobno komuniciraju preko mreže upotrebom internet protokola.

Pre same razmene sadržaja između uređaja, mora doći do potvrde autentičnosti uređaja i razmene ključeva odnosno AKE-a (*Authentication and Key Exchange*). Nakon AKE-a na uređaju koji je izvor sadržaja vrši se šifrovanje sadržaja, pakovanje u odgovarajući format i omogućava se njegovo preuzimanje. Na uređaju koji potražuje sadržaj, sadržaj se raspakuje i vrši se njegovo dešifrovanje (Slika 2.1).



Slika 2.2 Integracija DTCP-IP Source uređaja u DLNA okruženje

DMS uređaj mora da poseduje svoj HTTP server. Pre nego što DLNA DMP zatraži odgovarajući multimedijalni sadržaj od DMS-a on mora da izvrši pretragu DMS-a tj. da pretraži sadržaj koji DMS poseduje. Multimedijalni sadržaj može da bude nezaštićen ili zaštićen pomoću DTCP-IP-a. U slučaju da multimedijalni sadržaj nije zaštićen on se direktno razmenjuje upotrebom HTTP (*Hypertext Transfer Protocol*) protokola. Ukoliko je multimedijalni sadržaj zaštićen upotrebom DTCP-IP-a, pre same razmene sadržaja odvija se AKE procedura u kojoj se proverava autentičnost uređaja koji komuniciraju i razmene se neophodni ključevi. Zatim se multimedijalni sadržaj razmenjuje upotrebom HTTP protokola (Slika 2.2).



Slika 2.3 Integracija DTCP-IP Sink uređaja u DLNA okruženje

Na uređaju koji potražuje multimedijalni sadržaj vrši se prezentovanje tj. upotreba multimedijalnog sadržaja. Ukoliko je multimedijalni sadržaj zaštićen upotrebom DTCP-IP-a pre preuzimanja sadržaja mora da se izvrši AKE tj. potrebno je da se potvrdi autentičnost uređaja i razmene neophodni ključevi. Nakon AKE-a preuzima se zaštićeni multimedijalni sadržaj sa DMS-a koji se potom dešifruje u MP-u (*Media Player*). DTCP-IP mora biti ugrađen u MP. Nakon preuzimanja i dešifrovanja multimedijalni sadržaj može da se upotrebi tj. prezentuje (Slika 2.3).

Glava 3

Zaštita multimedijalnog sadržaja upotrebom DTCP-IP-a

3.1 Osnovna struktura zaštite sadržaja

Rešenje zaštite sadržaja koristi kombinaciju raznih tehnika i mehanizama za zaštitu sadržaja kako bi se sprečilo korišćenje sadržaja u suprotnosti sa uslovima propisanim od vlasnika sadržaja [3]. Mehanizmi zaštite imaju formu kriptografskih protokola tako da se sadržaj razmenjuje u šifrovanom obliku. Dešifrovanje sadržaja moguće je samo uz upotrebu odgovarajućih ključeva. Pristup ključevima i drugim potrebnim podacima, neophodnim za dešifrovanje zaštićenog sadržaja, imaju samo uređaji koji poseduju odgovarajuću licencu.

Licenca je pravni instrument za sprovođenje uslova pod kojima se obezbeđuje pristup sadržaju. Tehnike zaštite sadržaja su veoma efikasne i veoma je teško zaobići ih. Primena kriptografskih tehnika obezbeđuje osnov za uspešnu zaštitu sadržaja, dok efikasno licenciranje obezbeđuje izvršenje same zaštite.

DTLA specifikacija opisuje ponašanje dva moguća uređaja koji učestvuju u razmeni sadržaja. Opisuje se ponašanje uređaja koji je izvor sadržaja (*Source*) i uređaja koji je odredište sadržaja (*Sink*). Prvo se pristupa autentifikaciji uređaja a zatim se sadržaj na izvoru šifruje. Nakon toga moguće je da se sadržaj razmeni između uređaja koji je izvor sadržaja i uređaja koji je odredište sadržaja. Uređaj koji je odredište sadržaja dešifruje primljeni sadržaj i time je razmena sadržaja kompletirana. DTLA specifikacija opisuje četiri osnovna sloja u zaštiti sadržaja.

3.1.1 Kontrola kopiranja informacija

Kontrola kopiranja informacija se skraćeno naziva CCI (*Copy Control Information*) [4]. Vlasnicima sadržaja je potrebna mogućnost da odrede tj. specificiraju na koji način će se upotrebljavati zaštićeni sadržaj. Ovaj sistem zaštite sadržaja daje mogućnost kontrole kopiranja informacija na dva načina:

1. EMI (*Encryption Mode Indicator*) - Kontrola kopiranja informacija na osnovu vrednosti EMI parametra u zaglavlju paketa podataka.
2. Kontrola kopiranja informacija koja je ugrađena u sam multimedijalni sadržaj koji se razmenjuje. Ovaj način kontrole kopiranja informacija moguć je samo kod uređaja koji su sposobni da prepoznaju specifični format multimedijalnog sadržaja.

3.1.2 Autentifikacija uređaja i razmena ključeva

Autentifikacija i razmena ključeva skraćeno se naziva AKE (*Authentication and Key Exchange*). Pre same razmene sadržaja između uređaja, potrebno je da se izvrši autentifikacija povezanih uređaja i razmena ključeva. Sam proces autentifikacije i razmene ključeva biće detaljno opisan. Kako bi se ispunili strogi zahtevi industrije i realne potrebe personalnih računara i korisnika potrošačke elektronike, razvijena su dva načina autentifikacije, potpuna autentifikacija (*Full Authentication*) i ograničena autentifikacija (*Restricted Authentication*).

3.1.3 Šifrovanje i dešifrovanje sadržaja

Šifrovanje se koristi kako bi se sprečio neovlašćen pristup zaštićenom sadržaju. Dešifrovanje bi trebalo da bude moguće samo uređajima koji su kroz autentifikaciju potvrdili da poseduju licencu. Nakon autentifikacije i razmene ključeva uređaji su spremni da razmene sadržaj.

Na izvoru sadržaja vrši se šifrovanje sadržaja uz pomoć ključa za šifrovanje i odgovarajućeg algoritma šifrovanja, dok se na prijemnoj strani vrši dešifrovanje primljenog sadržaja uz pomoć identičnog ključa i algoritma šifrovanja koji se koristio na izvoru sadržaja.

3.1.4 Očuvanja integriteta sistema

Uređaji koji podržavaju potpunu autentifikaciju (*Full Authentication*) mogu da prime i obrade SRM (*System Renewability Messages*) poruke [4]. SRM poruke se razmenjuju u okviru AKE-a. Ove poruke kreira DTLA organizacija i distribuira ih kroz nove uređaje. Razmenom SRM poruka obezbeđuje se dugoročni integritet sistema, tj. vrši se opozivanje kompromitovanih uređaja. Na ovaj način se obezbeđuje da samo licencirani uređaji imaju pristup zaštićenom sadržaju i da se uređaji koji ne podržavaju uslove propisane DTLA specifikacijom izbace iz upotrebe.

3.2 DTCP-IP protokol

3.2.1 Potpuna autentifikacija i razmena ključeva

Da bi se osiguralo da samo licencirani uređaji imaju pristup zaštićenom sadržaju, potrebna su tehnička sredstva koja proveravaju ovlašćenja uređaja, tj. proveravaju da li uređaj poseduje licencu. Ovo se postiže tako što onaj koji daje licencu za tehnologiju zaštite sadržaja (DTLA), obezbedi tajne vrednosti koje su dostupne samo uređajima koji poseduju licencu. Ove tajne vrednosti se implicitno ili eksplicitno proveravaju u procesu pristupa zaštićenom sadržaju. Primer implicitne provere predstavlja izračunavanje i korišćenje tajne vrednosti koju jedino mogu da izračunaju usaglašeni licencirani uređaji. Licencirani uređaji ne moraju biti samo samostalni fizički uređaji kao što je televizor ili *set-top box* uređaji, to može biti i aplikacija koja radi na personalnom računaru.

Kao što je već pomenuto razvijena su dva načina autentifikacije, *potpuna* autentifikacija (*Full*) i *ograničena* autentifikacija (*Restricted*). Rad opisuje jedno rešenje realizacije programske podrške za zaštitu sadržaja u kom je realizovana podrška za *potpunu* autentifikaciju. Prema DTLA specifikaciji DTCP zaštita sadržaja mapirana na IP (*Internet Protocol*) podržava samo *potpunu* autentifikaciju.

Prilikom potpune autentifikacije koriste se *javni* i *privatni* ključ. Javni i privatni ključ koriste se u EC-DSA (*Elliptic Curve Digital Signature Algorithm*) algoritmu. EC-DSA algoritam se upotrebljava prilikom kreiranja potpisa na podatke i prilikom verifikacije potpisa. Takođe se koristi i EC-DH (*Elliptic Curve Diffie-Hellman*) algoritam za generisanje deljenog *autentifikacionog* ključa. Potrebno je napomenuti da određene konstante, parametre, ključeve i sertifikate kreira DTLA organizacija i prosleđuje ih na korišćenje licenciranim uređajima.

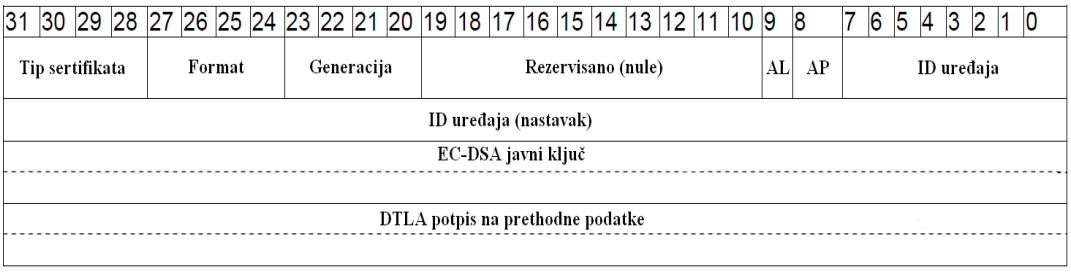
3.2.1.1 DTLA kompatibilni uređaji

Da bi neki uređaj koji podržava potpunu autentifikaciju bio DTLA kompatibilan potrebno je da poseduje jedinstven dodeljen ID (identifikacioni broj) uređaja. Takođe je potrebno da poseduje par EC-DSA ključeva, *privatni* i *javni* ključ, koje generiše DTLA. Kompatibilan uređaj mora da čuva *privatni* ključ i ne sme da dozvoli da on bude objavljen tj. otkriven.

Ukoliko dođe do otkrivanja privatnog ključa ID tog uređaja može da dospe u SRM listu i tada ovaj uređaj nije više DTLA kompatibilan. Kompatibilan uređaj takođe mora da poseduje i *sertifikat* kreiran od strane DTLA. *Sertifikat* se koristi prilikom autentifikacije uređaja. Kompatibilni uređaji moraju da poseduju i čuvaju neke dodatne konstante i ključeve koji su neophodni u procesu šifrovanja odnosno dešifrovanja sadržaja.

3.2.1.2 Sertifikat uređaja

DTLA dodeljuje sertifikat svakom sertifikovanom uređaju. Sertifikat svakog uređaja je jedinstven. Uređaj čuva *sertifikat* i koristi ga u procesu autentifikacije.



Slika 3.1 Struktura sertifikata uređaja

Sertifikat uređaja sadrži sledeća polja:

1. Tip sertifikata - trenutno jedina definisana vrednost ovog polja je 0 (nula).
2. Format sertifikata - ovo polje opisuje format sertifikata. Trenutno su definisana tri moguća formata:
    - Format 0 - format sertifikata za ograničenu autentifikaciju.
    - Format 1 - osnovi format sertifikata za potpunu autentifikaciju (format koji se koristi prilikom mapiranja DTCP-a na IP).
    - Format 2 - format proširenog sertifikata za potpunu autentifikaciju.
3. Generacija - Ovo polje označava maksimalnu generaciju SRM poruka koje podržava uređaj. Ovo polje može da ima sledeće vrednosti:
    - 0 - označava da uređaj podržava SRM poruke prve generacije.
    - 1 - označava da uređaj podržava SRM poruke druge generacije.
4. Rezervisano polje - rezervirani biti za buduću upotrebu. Sada su njihove vrednosti nula (0) i trenutno nemaju upotrebu.
5. AL (*Additional Localization*) polje - ovo polje se koristi kako bi se označilo da li uređaj ima mogućnost da izvrši AL („dodatna lokalizacija") proceduru. Prilikom ove procedure se proverava da li se povezani uređaji nalaze u kućnoj mreži.
6. AP polje - ovo polje se ne upotrebljava prilikom mapiranja DTCP-a na IP.
7. ID uređaja - identifikacioni broj uređaja koji dodeljuje DTLA organizacija.
8. EC-DSA javni ključ - *javni* ključ uređaja se koristi prilikom autentifikacije uređaja, odnosno prilikom verifikacije DTLA potpisa na podatke. *Javni* ključ je kreiran od strane DTLA organizacije i dodeljuje se svakom kompatibilnom uređaju.
9. DTLA potpis - *potpis* koji se daje na sve prethodne podatke u sertifikatu. Potpis se kreira na osnovu obrađenih podataka i vrednosti *privatnog* ključa.

3.2.1.3 Kriptografske funkcije

DTCP-IP funkcionalnosti su bazirane na kriptografskom sistemu eliptične krive (*ECC - Elliptic Curve Cryptography*). Prilikom autentifikacije uređaja, šifrovanja i dešifrovanja podataka, generisanja i provere potpisa koriste se odgovarajući kriptografski algoritmi odnosno kriptografske funkcije .

1. SHA-1 (*Secure Hash Algorithm*)
2. RND (*Random*) generator - generator slučajnog broja
3. EC-DH algoritam
4. EC-DSA algoritam

SHA-1 je algoritam koji se koristi u DSS-u (*Digital Signature Standard*), za generisanje digitalne poruke čija je veličina 160 bita.

Digitalna poruka se dobija na osnovu prosleđenih podataka SHA-1 algoritmu. Suštinski je algoritam sličan algoritmu provere ukupne sume. Prilikom potpune autentifikacije potreban je visoko kvalitetni generator slučajnog broja tj. RND generator.

ECC sistem se koristi kao osnova za EC-DH (*Diffie-Hellman*) i EC-DSA (*Digital Signature Algorithm*) algoritme. EC-DSA algoritam se koristi za kreiranje potpisa i za njegovu verifikaciju tj. proveru. Potpis se dobija na osnovu prosleđenih podataka, *privatnog* ključa i parametara eliptične krive. Suštinski predstavlja 320-bitnu vrednost. Verifikacija tj. provera potpisa se vrši na osnovu prosleđenih potpisanih podataka, *javnog* ključa i parametara eliptične krive.

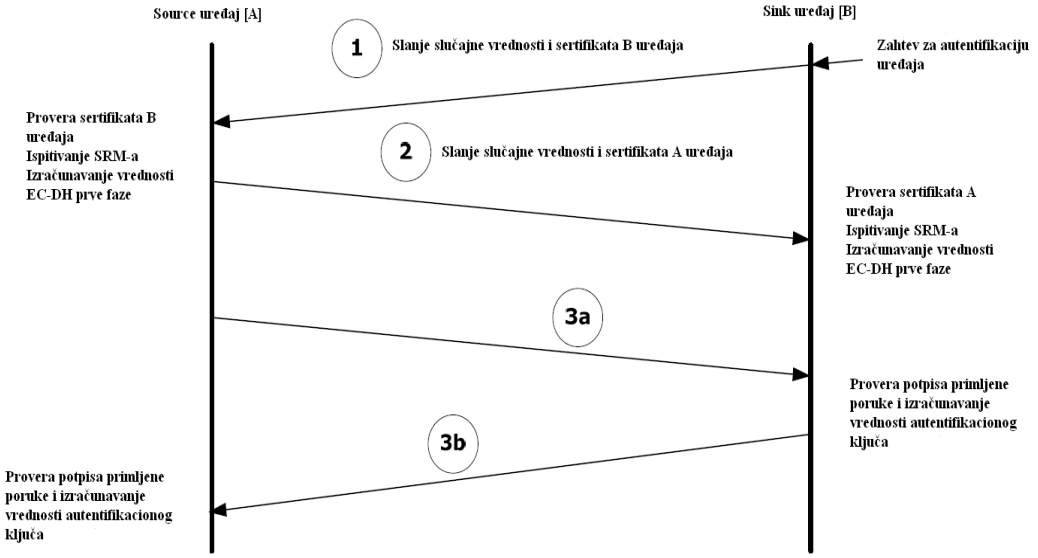
EC-DH algoritam se koristi kako bi se izračunala vrednost *Diffie-Hellman*-ove *prve faze*, a na osnovu vrednosti *prve faze*, računa se *tajna deljena* vrednost koja se strogo čuva i koristi dalje u algoritmu prilikom autentifikacije i razmene ključeva.

3.2.1.4 Razmena poruka prilikom autentifikacije i razmene ključeva

Kao što je već napomenuto autentifikacija i razmena ključeva (*AKE - Device Authentication and Key Exchange*) podrazumeva proveru autentičnosti povezanih uređaja pre nego što dođe do razmene informacija tj. sadržaja. DTLA specifikacija opisuje ponašanje dva moguća uređaja koji učestvuju u razmeni sadržaja. Opisuje se ponašanje uređaja koji je izvor sadržaja (*Source*) i uređaja koji je odredište sadržaja (*Sink*). Prilikom AKE-a dolazi do razmene poruka između ova dva uređaja (Slika 3.2).

Prilikom potpune autentifikacije AKE procedura započinje tako što uređaj koji je odredište sadržaja tj. *sink* uređaj šalje generisanu slučajnu vrednost i svoj *sertifikat* ka uređaju koji je izvor sadržaja. Slučajna vrednost se generiše upotrebom RND generatora slučajnog broja. *Sertifikat* uređaja mora biti potpisan na ispravan način upotrebom EC-DSA algoritma i DTLA *privatnog* ključa. Ovo ponašanje je posledica pokušaja *sink* uređaja da pristupi zaštićenom sadržaju. Uređaj koji je izvor sadržaja tj. *source* u svakom trenutku može da odbije zahtev *sink* uređaja ukoliko na bilo koji način zahtev postane diskutabilan odnosno neispravan.

Ukoliko je zahtev uređaja koji potražuje sadržaj uspešno primljen, uređaj koji je primio zahtev tj. izvor sadržaja šalje poruku odredišnom uređaju. Ova poruka takođe sadrži slučajnu vrednost izračunatu na uređaju koji je izvor sadržaja upotrebom RND generatora i *sertifikat* tog uređaja, koji je ispravno potpisan upotrebom EC-DSA algoritma i DTLA *privatnog* ključa. Ukoliko bilo koji od uređaja u bilo kom momentu ustanovi da je *sertifikat* drugog uređaja pogrešan ili se ustanovi da *potpis* nije korektan, proces autentifikacije se odmah prekida.



Slika 3.2 Protok komandi u AKE-u

Nakon razmene slučajnih vrednosti i sertifikata uređaja, svaki od uređaja proverava integritet *sertifikata* drugog uređaja kroz EC-DSA algoritam. To znači da se na svakoj strani proverava ispravnost *potpisa* na *sertifikat* upotrebom EC-DSA algoritma i DTLA *javnog* ključa. Ukoliko se ustanovi da su razmenjeni *sertifikati* uređaja ispravni tj. da su DTLA potpisi na *sertifikat* ispravni prelazi se na proveru SRM-a. SRM poruke kreira i isporučuje DTLA i one sadrže liste opozvanih uređaja. Na osnovu *sertifikata* svaki od uređaja odredi identifikacioni broj drugog uređaja i na osnovu identifikacionog broja i SRM poruke proveri se da li se uređaj nalazi u listi opozvanih uređaja. Ukoliko se uređaj nalazi u listi opozvanih uređaja proces autentifikacije se odmah prekida.Ukoliko su uređaji proverili SRM, i ukoliko se identifikacioni brojevi uređaja ne nalaze u SRM-u, prelazi se na razmenu poruka koje sadrže vrednost *prve faze* izračunate EC-DH algoritmom, verziju i generaciju SRM poruke koju sadrži svaki od uređaja i potpis na prethodne podatke. Kada se ove poruke prime na svakoj strani se proverava ispravnost potpisa upotrebom EC-DSA algoritma i DTLA *javnog* ključa. Ukoliko je potpis bilo koje poruke neispravan autentifikacija se prekida.

Verzija i generacija SRM poruke se koriste kako bi se ustanovilo da li neki od uređaja poseduje noviju SRM poruku. Ukoliko je to tačno uređaj sa novijom SRM porukom omogućava razmenu SRM poruke sa drugim uređajem ukoliko se to zatraži. Svaki od uređaja dalje računa *autentifikacioni ključ* (*K_{AUTH}-Authentication Key*) na osnovu primljene vrednosti *prve faze*. Autentifikacioni ključ-K_{AUTH} se koristi prilikom kreiranja ključeva za upravljanje sadržajem, tj. za kreiranje ključeva koji se kasnije koriste za šifrovanje i dešifrovanje sadržaja.

3.2.2 Upravljanje ključevima

Nakon uspešne autentifikacije uređaja prelazi se na razmenu ključeva. Uređaj koji je izvor sadržaja (*Source*) kreira ključ K_X (*Exchange Key*) ili ključ K_S (*Session Exchange Key*). Jedan od ova dva ključa se upotrebljava za izračunavanje K_C (*Content Key*) ključa koji se koristi prilikom šifrovanja odnosno dešifrovanja sadržaja.

3.2.2.1 K_X-ključ

Zajednički skup K_X ključeva se kreira između uređaja koji je izvor sadržaja i svih uređaja koji potražuju sadržaj od istog uređaja koji je izvor. Pri tom proces autentifikacije između tih uređaja mora biti završen.

Nakon izvršene autentifikacije uređaja prelazi se na izračunavanje K_X ključa. Uređaj koji je izvor sadržaja dodeljuje slučajnu vrednost za svaki K_X ključ koji se kreira. Slučajna vrednost se dobija pomoću RND generatora slučajne vrednosti.

Prilikom autentifikacije uređaja kreira se *tajna deljena* vrednost koju uređaji razmenjuju. Ova tajna vrednost naziva se autentifikacioni ključ (K_{AUTH}). Na osnovu dobijene slučajne vrednosti K_X i prethodno ustanovljenog autentifikacionog ključa (K_{AUTH}) kreira se vrednost K_{SX}. Na osnovu ove vrednosti i uz pomoć prethodno ustanovljenog autentifikacionog ključa (K_{AUTH}), koji poseduju oba uređaja, može se rekurzivnim postupkom doći do vrednosti K_X (*Exchange Key*) na uređaju koji je odredište sadržaja.

Uređaj koji je izvor sadržaja zatim šalje vrednost K_{SX} ka uređaju koji je odredište sadržaja. Kada uređaj koji je odredište sadržaja primi poslatu K_{SX} vrednost on je upotrebi kako bi uz pomoć nje i vrednosti prethodno ustanovljenog autentifikacionog ključa (K_{AUTH}), odredio slučajnu vrednost kreiranu na uređaju koji je izvor sadržaja. Dobijena vrednost predstavlja K_X (*Exchange Key*). K_X vrednost je veoma značajna jer se na osnovu ove vrednosti računa K_C ključ koji se koristi u algoritmu šifrovanja i dešifrovanja sadržaja.

Uređaj koji je izvor sadržaja ponavlja ovu proceduru za svaki novi zahtev za sadržajem. Nakon ovog postupka proces autentifikacije i razmene ključeva se završava. Na ovaj način uređaji su potvrdili autentičnost i razmenili ključeve koji su potrebni u daljem procesu razmene zaštićenog sadržaja.

3.2.2.2 K_S-ključ

K_S ključ je jedinstven za svaki povezani uređaj koji potražuje sadržaj od izvora sadržaja, za razliku od K_X ključa koji je zajednički za sve povezane uređaje. Podrška za K_S ključ nije obavezna i on se kreira ukoliko oba povezana uređaja imaju podršku za K_S ključ. K_S ključ se upotrebljava za prenos sadržaja koji ne može biti istovremeno poslat ka više uređaja. K_S ključ se istovremeno može koristiti i za prenos sadržaja za koji se inače upotrebljava K_X ključ.

3.2.2.3 K_C-ključ

K_C ključ (*Content Key*) se koristi prilikom šifrovanja odnosno dešifrovanja podataka koji se razmenjuju. Vrednost ovog ključa izračunava se na osnovu tri vrednosti:

1. K_X ključ.
2. N_C-slučajna vrednost dobijena pomoću RND generatora slučajnog broja na uređaju koji je izvor sadržaja.
3. Konstanti C_{A0}, C_{B1}, C_{B0}, C_{C1}, C_{C0} ili C_{D0} - koje odgovaraju EMI vednosti.

K_C se izračunava na sledeći način:

- K_C = J-AES(K_X, f[EMI], N_C);

C_A, C_B ili C_C konstante definiše DTLA i one predstavljaju vrednosti f[EMI] funkcije.

U zavisnosti od EMI vrednosti f[EMI] funkcija ima sledeće vrednosti:

1. C_{A0} ukoliko je EMI = Mod A0
2. C_{B0} ukoliko je EMI = Mod B0
3. C_{B1} ukoliko je EMI = Mod B1
4. C_{C0} ukoliko je EMI = Mod C0
5. C_{C1} ukoliko je EMI = Mod C1
6. C_{D0} ukoliko je EMI = Mod D0

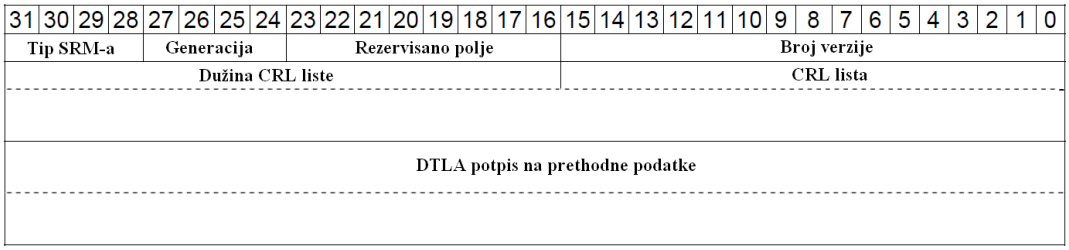
K_C ključevi (*Content Keys*) se koriste prilikom šifrovanja odnosno dešifrovanja sadržaja koji se razmenjuje. Nakon izvršene autentifikacije i razmene ključeva uređaji mogu da razmene sadržaj. Uređaj koji je izvor sadržaja vrši šifrovanje sadržaja odgovarajućim algoritmom i K_C ključem, dok se na prijemnoj strani vrši dešifrovanje sadržaja identičnim algoritmom i odgovarajućim K_C ključem. Kada izvor sadržaja počne sa slanjem sadržaja on pomoću RND generatora kreira 64-bitnu slučajnu vrednost kao početnu vrednost za izračunavanje K_C ključa. K_C ključ se kreira na osnovu slučajne vrednosti N_C, K_X ključa (*Exchange Key*) i vrednosti funkcije f[EMI]. Ove vrednosti se upotrebljavaju kao ulazni parametri u algoritmu koji kao izlazni parametar daje K_C (*Content Key*) ključ. Kada uređaj koji je odredište sadržaja pošalje zahtev ka uređaju koji je izvor sadržaja, izvor sadržaja šalje ka uređaju koji je odredište sadržaja kreiranu slučajnu vrednost N_C. Ova vrednost se na uređaju koji je odredište sadržaja koristi kako bi se izračunao K_C ključ. K_C ključ se računa na isti način kao i na uređaju koji je izvor sadržaja. Dobijeni ključ na obe strane mora biti identičan kako bi šifrovanje odnosno dešifrovanje sadržaja bilo uspešno.

3.2.3 Očuvanje integriteta sistema

Uređaji koji podržavaju potpunu autentifikaciju (*Full Authentication*) mogu da prime i obrade SRM (*System Renewability Messages*) poruke. Ove poruke kreira DTLA organizacija i distribuira ih kroz nove uređaje. Razmenom SRM poruka obezbeđuje se dugoročni integritet sistema, tj. vrši se opozivanje kompromitovanih uređaja.

3.2.3.1 Struktura SRM poruke

SRM poruke imaju precizno definisanu strukturu. One se sastoje od nekoliko polja koja opisuju osnovne osobine i sadržaj SRM poruke.



Slika 3.3 Struktura SRM poruke prve generacije

SRM poruke se sastoje iz sledećih polja:

1. Tip SRM poruke - ovo polje ima istu funkciju kao i kod sertifikata uređaja. Trenutno jedina definisana vrednost ovog polja je 0 (nula).
2. Generacija SRM poruke - ovim poljem se definiše generacija SRM poruke koju poseduje uređaj. Vrednost ovog polja definiše proširivost SRM mehanizma. Trenutno definisane vrednosti ovog polja su nula i jedan. Vrednost nula je rezervisana za SRM poruke prve generacije a vrednost jedan je rezervisana za SRM poruke druge generacije.
3. Rezervisano polje - može da se koristi ukoliko bude potrebe u budućnosti. Za sada vrednost ovog polja mora da bude jednaka nuli.
4. Broj verzije SRM poruke - definiše verziju SRM poruke koja je dostupna uređaju. Od vrednosti ovog polja zavisi da li će SRM poruka biti ažurirana. Ukoliko je uređaj povezan sa drugim uređajem koji poseduje SRM poruku novije generacije doćiće do ažuriranja SRM poruke.
5. Dužina CRL (*Certificate Revocation List*) liste - ovo polje definiše veličinu CRL liste u bajtovima. Veličina CRL liste obuhvata veličinu ovog polja, veličinu same CRL liste koja sadrži identifikacione brojeve kompromitovanih uređaja i veličinu DTLA potpisa.
6. CRL lista - lista identifikacionih brojeva uređaja koji su kompromitovani.
7. DTLA potpis - ovo polje sadrži DTLA potpis na prethodno definisane podatke.

Tip SRM poruke, generacija SRM poruke, rezervisano polje i broj verzije SRM poruke zajedno predstavljaju zaglavlje SRM poruke. Ažuriranjem SRM poruke lista se proširuje a zaglavlje ostaje isto uz ažuriranje generacije i verzije SRM poruke.

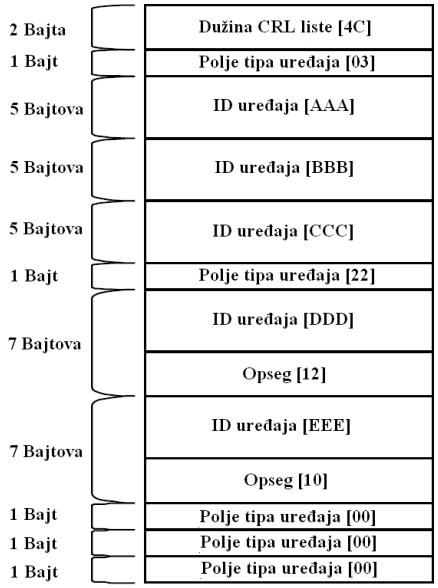
3.2.3.2 CRL lista

CRL lista identifikuje uređaje koji više nisu kompatibilni. CRL lista sadrži polje koje definiše dužinu CRL liste. Nakon ovog polja dolazi polje *tipa uređaja* koje definiše tip uređaja koji su opozvani i njihov broj. Ovo polje ima sledeću strukturu:

7	6	5	4	3	2	1	0
Tip uređaja			Identifikacioni broj uređaja				

Slika 3.4 Struktura polja koje definiše tip i broj opozvanih uređaja

U zavisnosti od tipa uređaja određuje se način definisanja opozvanih uređaja. Tip uređaja definiše vrednost tri bita naveće važnosti ovog polja. Ukoliko je vrednost ova tri bita nula svaki opozvani uređaj se opisuje samo identifikacionim brojem koji sadrži pet bajtova. Ukoliko je vrednost ova tri bita jedan, opozvani uređaji se opisuju identifikacionim brojem uređaja (pet bajtova) i opsegom (dva bajta), čija vrednost opisuje opseg uređaja koji su opozvani. Ostale vrednosti su rezervisane za neke buduće upotrebe.

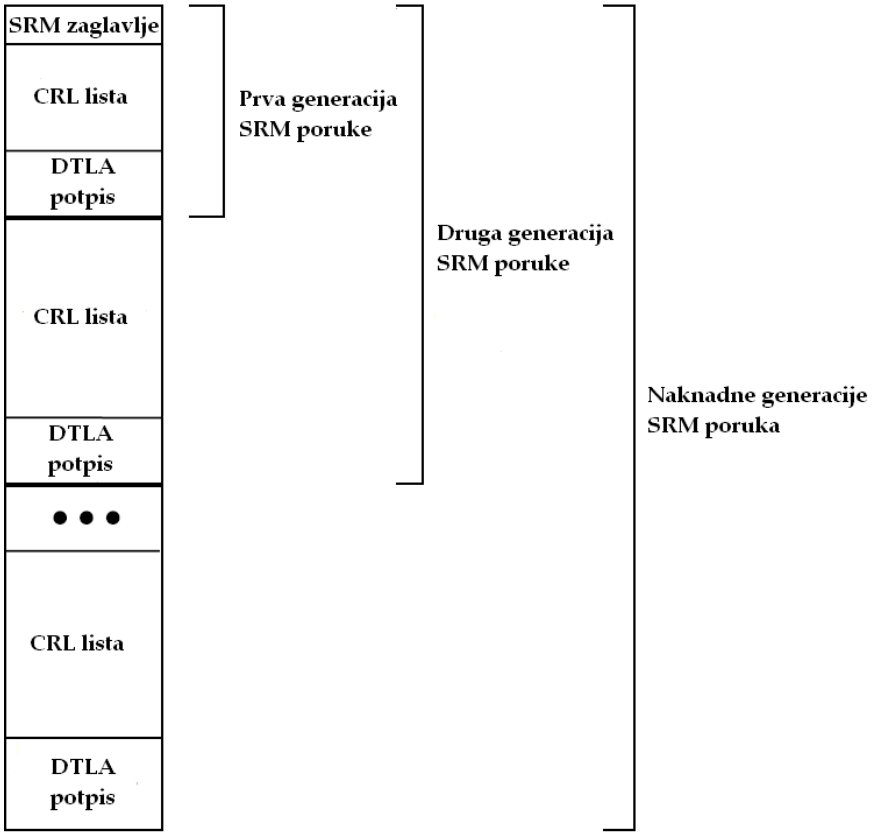


Slika 3.5 Primer CRL liste uređaja

U primeru CRL liste (Slika 3.5) opozvana su tri uređaja tipa nula, ovo je definisano u polju tipa uređaja. Identifikacioni brojevi opozvanih uređaja tipa nula su AAA, BBB i CCC. Zatim su opozvane dve grupe uređaja, čiji se identifikacioni brojevi nalaze u opsegu DDD - (DDD + 12_{16}) i opsegu EEE – (EEE + 10_{16}). CRL lista je dopunjena određenim brojem polja *tipa uređaja* vrednosti nula, kako bi veličina CRL liste u bitima bila poravnata na veličinu deljivu sa brojem 32.

3.2.3.3 Proširivost SRM poruke

SRM poruke moraju da se menjaju u zavisnosti od broja uređaja koji postaju nekompatibilni. Ovo je omogućeno time što su SRM poruke proširive. Svako sledeće proširenje SRM poruke tj. nova generacija SRM poruke, podrazumeva dodavanje nove SRM poruke na već postojeću SRM poruku, odnosno SRM generaciju koja je trenutno u upotrebi (Slika 3.6). Zaglavlje SRM poruke ostaje identično uz promenu generacije i verzije SRM poruke, dok se na trenutnu SRM poruku dodaje CRL lista koja sadrži identifikacione brojeve opozvanih uređaja. Prilikom svakog proširenja SRM poruke kreira se DTLA potpis, koji se kreira na osnovu svih podataka koji ulaze u SRM poruku.



Slika 3.6 Šema proširivosti SRM poruke

3.2.3.4 Ažuriranje SRM poruke

Ukoliko se prilikom potpune autentifikacije ustanovi da neki od kompatibilnih uređaja poseduje noviju SRM poruku u odnosu na SRM poruku koju poseduje drugi uređaj, dolazi do ažuriranja SRM poruke (podrazumeva se da su uređaji međusobno povezani). Razmena SRM poruka odvija se nakon završene autentifikacije uređaja i razmene ključeva.

Ažuriranje SRM poruke izvršava se na sledeći način:

1. Ispituje se broj verzije SRM poruke drugog kompatibilnog uređaja. Brojeve verzija SRM poruka uređaji razmene prilikom autentifikacije.
2. Ispituje se da li je broj verzije SRM poruke drugog kompatibilnog uređaja veći od broja verzije SRM poruke koju uređaj trenutno poseduje.
3. Ukoliko uređaj poseduje noviju SRM poruku proverava se ispravnost DTLA potpisa nove SRM poruke.
4. Ukoliko je SRM poruka ispravna i broj verzije nove SRM poruke veći od broja SRM poruke koju uređaj trenutno poseduje dolazi do zamene SRM poruke novom SRM porukom. Razmena SRM poruka se odvija nakon uspešne autentifikacije i razmene ključeva ukoliko je to potrebno.

3.2.4 Struktura komandi u AKE-u

3.2.4.1 AKE kontrolne komande

AKE kontrolne komande (Slika 3.7) se koriste za razmenu informacija prilikom AKE procedure (autentifikacija i razmena ključeva) [5].

	7	6	5	4	3	2	1	0
Tip[0]	0	0	0	0	0	0	0	1
Dužina[0]	Veličina_podataka							
Dužina[1]								
Kontrola[0]	Rezervisano				Ctip/odgovor			
Kontrola[1]	Kategorija (0000_2)				AKE_ID (0000_2)			
Kontrola[2]	AKE_ID zavisna polja							
Kontrola[3]								
Kontrola[4]								
Kontrola[5]								
Kontrola[6]	AKE_oznaka							
Kontrola[7]	Opcioni_broj				Status			
AKE_info[0-(N-1)]	AKE_info (podaci)							

Slika 3.7 Struktura DTCP-IP kontrolne komande

Prilikom razmene AKE komandi svaka AKE kontrolna komanda zahteva odgovor. Ukoliko je vrednost nekog od polja kontrolne komande nedefinisana vrednost, biće vraćen odgovor na kontrolnu komandu koji nosi informaciju da obrada komande tog formata nije realizovana.

AKE kontrolne komande koriste se prilikom razmene informacija za vreme procesa autentifikacije koji se odvija između uređaja koji je izvor sadržaja (*Source*) i uređaja koji je odredište sadržaja (*Sink*). Informacije se šalju u polju *podataka* i još se nazivaju *AKE_info*.

Kontrolna komanda sadrži specijalna polja koja se koriste kada se DTCP mapira na IP. To su polja *tip*, polja *dužina* i prvo *kontrolno* polje. Prilikom mapiranja DTCP-a na IP polje *tip* ima vrednost jedan. Polje *veličina_podataka* sadrži vrednost koja je jednaka broju bajtova koje zauzimaju *kontrolna* polja i *AKE_info* polje. Polje *rezervisano* ima vrednost nula.

Polje *ctip/odgovor* ima dvostruku ulogu. U strukturi komande vrednost ovog polja nosi informaciju koji tip komande se koristi, npr. da li se radi o kontrolnoj komandi, statusnoj komandi, komandi koja nosi neko obaveštenje ili o komandi kojom se vrši neko ispitivanje. U strukturi odgovora na komandu vrednost ovog polja nosi informaciju da li je komanda u procesu tranzicije, da li je prihvaćena, odbijena ili ne postoji realizacija odgovora na primljenu komandu.

Zatim sledi polje *kategorija* čija je vrednost nula. Vrednost *AKE_ID* polja opisuje format naredna četiri polja, a to su polja *podfunkcija*, *AKE_procedura*, *ključ_razmene* i *zavisnosti_podfunkcije*. Trenutno jedina definisana vrednost *AKE_ID* polja je nula, ostale vrednosti ovog polja mogu biti upotrebljene za neke buduće potrebe. Polje *AKE_oznaka* je jedinstvena oznaka koja se upotrebljava za razlikovanje niza AKE komandi koje pripadaju određenom autentifikacionom procesu. Pokretač autentifikacionog procesa može da odabere proizvoljnu vrednost polja *AKE_oznaka*. Vrednost ovog polja mora da bude različita za svaki pokrenuti proces autentifikacije.

Vrednost polja *opcioni_broj* opisuje broj razmenjenih komandi za vreme procesa autentifikacije. Uređaj koji pokreće proceduru autentifikacije postavlja vrednost ovog polja na jedan prilikom slanja kontrolne komande koja započinje proces autentifikacije. Vrednost ovog polja povećava se za jedan nakon razmene podniza komandi koja se odvija u istom procesu autentifikacije. Ukoliko uređaj ne podržava ovo polje njegova vrednost treba da bude nula.

Polje *status* se koristi kako bi se njime definisao status primljene komande. Kada se kreira odgovor na komandu u ovo polje se unosi vrednost koja nosi informaciju o razlogu odbijanja komande, ukoliko je komanda odbijena. Uređaj koji šalje komandu postavlja ovo polje na maksimalnu vrednost (sva četiri bita dobiju vrednost jedan). Ukoliko uređaj koji je primio poruku želi da odbije komadu u ovom polju se postavlja vrednost na osnovu koje se definiše razlog odbijanja kontrolne komande.

Maksimalna veličina komandi je 512 bajtova. Ukoliko je komanda dovoljno velika da baferi kontrolera ili ciljanog uređaja ne mogu da je prime prelazi se na deljenje komade. U ovom slučaju koristi se polje *preostali_blokovi*.

Kada je deljenje komande neophodno, polje *podataka* se deli u N blokova i svaki blok podataka se šalje jedan za drugim u potpuno odvojenim komandama. U ovom slučaju komanda je dovoljno mala da bafer kontrolera ili bafer ciljanog uređaja može da je primi. Ovi baferi moraju biti sposobni da prime komandu čije je polje *podataka* minimalne veličine 32 bajta. Veličina polja podataka u prvih N-1 deljenih komandi mora da bude jednaka i deljiva sa 16. Takođe veličina polja *podataka* mora biti veća ili jednaka 32 bajta.

Svaka od N komandi, koje su nastale deljenjem velike komande, ima jednaka sva polja osim polja *preostali blokovi*, *veličina podataka* i polja *podataka*. Za prvu komandu polje *preostali blokovi* ima vrednost N-1 i svaka sledeća komanda koja je uspešno poslata ima vrednost ovog polja umanjenu za jedan u odnosu na vrednost ovog polja u prethodnoj komandi. Kada ovo polje dostigne vrednost nula to znači da se šalje poslednja komanda u nizu. Ukoliko vrednost ovog polja nije korektna, uređaj koji primi poruku šalje odgovor kojim se poruka odbija.

Kada se kontrolna komanda deli i šalje kao niz više kontrolnih komandi svaka sledeća komanda se šalje samo ukoliko je prethodna komanda uspešno primljena i ukoliko je vraćen odgovor da je komanda uspešno primljena.

Polje *veličina podataka* definiše veličinu polja *podaci*. Takođe vrednost ovog polja u odgovoru na kontrolnu komandu mora biti ista kao i vrednost ovog polja u samoj kontrolnoj komandi, iako odgovor na kontrolnu komandu ne sadrži polje *podaci*. Polje *podaci* sadrži podatke koji se prenose u komandi. Odgovori na komande u kojima se komande odbijaju ne sadrže polje *podaci*.

3.2.4.2 AKE statusne komande

Statusne komande (Slika 3.8) se koriste za slanje statusa uređaja. Ove komande ne sadrže polje *podataka*.

<table>
<tr><th></th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td>Tip[0]</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>Dužina[0]</td><td colspan="8" rowspan="2">Veličina_podataka</td></tr>
<tr><td>Dužina[1]</td></tr>
<tr><td>Kontrola[0]</td><td colspan="4">Rezervisano</td><td colspan="4">Ctip/odgovor</td></tr>
<tr><td>Kontrola[1]</td><td colspan="4">Kategorija (0000_2)</td><td colspan="4">AKE_ID (0000_2)</td></tr>
<tr><td>Kontrola[2]</td><td colspan="8" rowspan="4">AKE_ID zavisna polja</td></tr>
<tr><td>Kontrola[3]</td></tr>
<tr><td>Kontrola[4]</td></tr>
<tr><td>Kontrola[5]</td></tr>
<tr><td>Kontrola[6]</td><td colspan="8">AKE_oznaka (FF_{16})</td></tr>
<tr><td>Kontrola[7]</td><td colspan="4">Opcioni_broj (F_{16})</td><td colspan="4">Status</td></tr>
</table>

Slika 3.8 Struktura DTCP-IP statusne komande

Statusna komanda sadrži specijalna polja koja se koriste kada se DTCP mapira na IP. To su polja *tip*, *veličina_podataka* i prvo *kontrolno* polje. Prilikom mapiranja DTCP-a na IP polje *tip* ima vrednost jedan. Polje *veličina_podataka* sadrži vrednost koja je jednaka broju bajtova koje zauzimaju *kontrolna* polja. Polje *rezervisano* ima vrednost nula.

Polje *ctip/odgovor* ima dvostruku ulogu. U strukturi komande vrednost ovog polja nosi informaciju koji tip komande se koristi, npr. da li se radi o kontrolnoj komandi, statusnoj komandi, komandi koja nosi neko obaveštenje ili o komandi kojom se vrši neko ispitivanje. U strukturi odgovora na komandu vrednost ovog polja nosi informaciju da li je komanda u procesu tranzicije, da li je prihvaćena, odbijena ili ne postoji realizacija odgovora na komandu.

Zatim sledi polje *kategorija* čija je vrednost nula. Vrednost *AKE_ID* polja opisuje format naredna četiri polja, to su polja *podfunkcija*, *AKE_procedura*, *ključ_razmene* i *zavisnosti_podfunkcije*. Trenutno jedina definisana vrednost *AKE_ID* polja je nula.

Polje *AKE_oznaka* i polje *opcioni_broj* uvek imaju maksimalnu vrednost u strukturi statusne komande. Polje *status* se koristi kako bi drugi uređaj dobio obaveštenje o stanju uređaja koji šalje komandu. Kada se kreira odgovor na komandu u ovo polje se unosi vrednost koja nosi informaciju o razlogu odbijanja komande, ukoliko je komanda odbijena.

3.2.4.3 AKE_ID zavisna polja

Vrednost *AKE_ID* polja opisuje format naredna četiri polja, to su polja *podfunkcija*, *AKE_procedura*, *ključ_razmene* i *zavisnosti_podfunkcije*. Trenutno jedina definisana vrednost *AKE_ID* polja je nula, u tom slučaju naredna četiri polja imaju strukturu kao na Slici 3.9.

	7	6	5	4	3	2	1	0
Operand[1]	Podfunkcija							
Operand[2]	AKE_procedura							
Operand[3]	Ključ_razmene							
Operand[4]	Zavisnosti_podfunkcije							

Tabela 3.1 Struktura AKE_ID zavisnih polja

Polje *podfunkcija* ima vrednost na osnovu koje se dobija informacija koju operaciju obavlja kontrolna komanda tj. koja podfunkcija se šalje. Vrednost najznačajnijeg bita polja *podfunkcija* označava da li kontrolna komanda ima polje *podaci* ili nema ovo polje.

Ukoliko je vrednost ovog bita nula kontrotrolna komanda ima polje *podaci* dok polje *veličina_podataka* nosi informaciju koliko bajtova zauzima polje *podaci.* Ukoliko je vrednost najznačajnijeg bita polja *podfunkcija* jedan, kontrolna komanda nema polje *podaci*. Skup mogućih podfunkcija koje se razmenjuju u AKE-u date su u tabeli 3.1. U strukturi statusne komande polje *podfunkcija* ima maksimalnu vrednost (FF_{16}).

Vrednost	AKE podfunkcije	Opis
01_{16}	CHALLENGE	Ovom funkcijom se šalje slučajna vrednost. Kada *sink* uređaj pošalje ovu podfukciju pokreće se AKE procedura.
02_{16}	RESPONSE	Ovom funkcijom se vraća vrednost izračunata na osnovu primljene slučajne vrednosti (*prva faza*).
03_{16}	EXCHANGE_KEY	Ovom funkcijom se salje K_X ključ ka *sink* uređaju.
04_{16}	SRM	Ovom porukom se salje SRM ukoliko je potrebno ažuriranje.
05_{16}	RESPONSE2	Ovom funkcijom se vraća vrednost izračunata na osnovu primljene slučajne vrednosti (*prva faza*) i jedinstvena vrednost koja služi za identifikovanje *Sink* uređaja.
CO_{16}	AKE_CANCEL	Ovom porukom se obaveštava uređaj da AKE procedura ne može biti nastavljena.
82_{16}	CAPABILITY_REQ	Ova poruka se koristi kako bi se utvrdile mogućnosti uređaja.

Tabela 3.2 AKE podfunkcije

Vrednost polja *AKE_procedura* opisuje o kom tipu AKE procedure se radi. Kada se DTCP mapira na IP moguća je samo potpuna autentifikacija. U opštem slučaju, uređaj koji pokreće autentifikaciju postavi vrednost ovog polja prilikom slanja prve kontrolne komande i svaka kontrolna komanda koja se razmeni u okviru te AKE procedure ima istu vrednost ovog polja. Ukoliko se radi o statusnim komandama, kada se ispituje status uređaja, uređaj koji šalje statusnu komandu postavi ovo polje na maksimalnu vrednost (FF_{16}). Kada se šalje odgovor na statusnu komandu u ovo polje se postavlja vrednost koja označava AKE procedure koje uređaj podržava (Tabela 3.2). Na osnovu statusne komande uređaj koji započinje autentifikaciju dobija informaciju koje AKE procedure podržava drugi uređaj i tada može da pokrene onu AKE proceduru koja je moguća.

Broj bita koji se postavi na vrednost jedan	AKE_procedura
0	Ograničena autentifikacija.
1	Poboljšana ograničena autentifikacija.
2	Potpuna autentifikacija.
3	Proširena potpuna autentifikacija.
4-7	Za buduće potrebe.

Tabela 3.3 Vrednosti polja *AKE_procedura*

Uređaj koji započinje proces autentifikacije prilikom slanja statusne komande,kojom se ispituje status uređaja, postavlja polje *ključ_razmene* na maksimalnu vrednost (FF_{16}). U odgovoru na statusnu komandu ovo polje sadrži vrednost koja definiše koji tip ključeva podržava ciljani uređaj (K_X, K_S…).

Kada se započne proces autentifikacije uređaj koji započinje autentifikaciju postavi vrednost ovog polja u kontrolnoj komandi i ona se ne menja dok se proces autentifikacije ne završi.

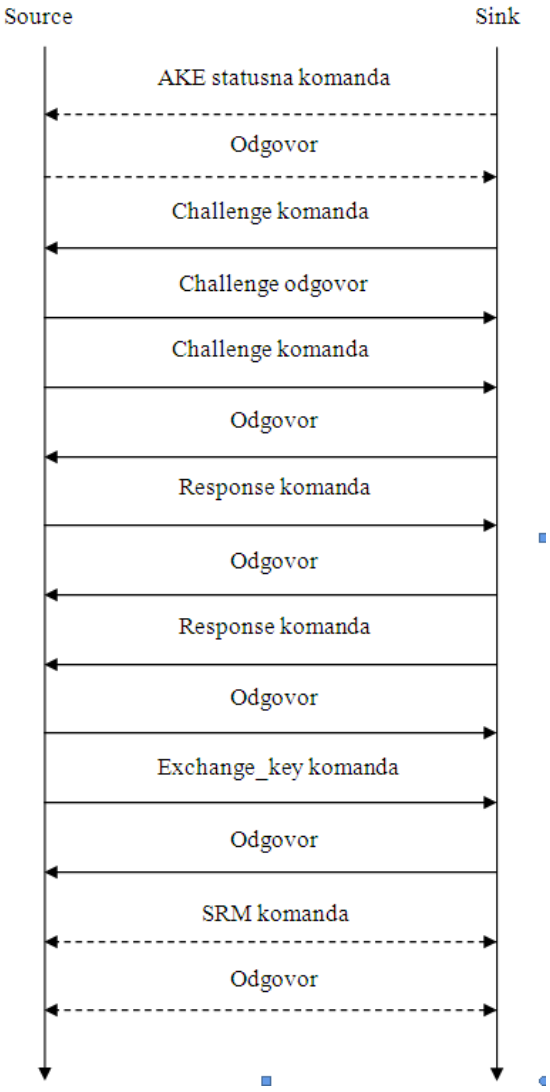
Broj bita koji se postavi na vrednost jedan	Ključ razmene
0	Zabranjeno.
1	Zabranjeno.
2	Zabranjeno.
3	K_X (*Exchange Key*) za AES-128.
4	Rezervisano za buduće potrebe.
5	K_S (*Session Exchange Key*) za AES-128.
6	K_R (*Remote Exchange Key*) za AES-128 (Upotrebljava se u *Remote Access* AKE-u koji nije podržan).
7	Rezervisano za buduće potrebe.

Tabela 3.4 Vrednost polja *ključ_razmene*

Polje *zavisnosti_podfunkcije* u statusnim komandama ima maksimalnu vrednost (FF_{16}), dok vrednost ovog polja u kontrolnim komandama definiše da li se radi o uređaju koji je izvor sadržaja ili o uređaju koji je odredište sadržaja.

3.2.5 Razmena komandi u AKE proceduri

AKE proceduru započinje uređaj koji potražuje sadržaj. Pre same autentifikacije uređaja moguće je da uređaj koji potražuje sadržaj pošalje *statusnu komandu* ka uređaju koji je izvor sadržaja (Slika 3.13). Uređaj koji je izvor sadržaja na *statusnu komandu* šalje *odgovor*. Na osnovu primljenog odgovora uređaj koji potražuje sadržaj dobija osnovne informacije o uređaju koji je izvor sadržaja i AKE proceduri koju ovaj uređaj podržava. Razmena *statusne komande* je moguća ali ne i obavezna.



Slika 3.9 Razmena komandi u AKE proceduri

U opštem slučaju AKE procedura započinje tako što *Sink* uređaj šalje ka *Source* uređaju *Challenge* komandu. *Challenge* komanda sadrži generisanu slučajnu vrednost i sertifikat uređaja koji šalje komandu. Slučajna vrednost se generiše upotrebom RND generatora slučajnog broja.

Sertifikat mora biti potpisan na ispravan način upotrebom EC-DSA algoritma i DTLA *privatnog* ključa. Ovo ponašanje je posledica pokušaja *Sink* uređaja da pristupi zaštićenom sadržaju. *Source* uređaj šalje ka *Sink* uređaju odgovor na komandu kojim obaveštava *Sink* uređaj o statusu komande. Zatim *Source* uređaj takođe šalje ka *Sink* uređaju *Challenge* komandu koja sadrži generisanu slučajnu vrednost i sertifikat *Sink* uređaja. Sertifikat takođe mora biti potpisan na ispravan način što se proverava nakon prijema komande. *Sink* uređaj šalje odgovor na primljenu *Challenge* komandu kojim obaveštava *Source* uređaj o statusu primljene komande.

Ukoliko su *Challenge* komande uspešno razmenjene AKE procedura se nastavlja tako što *Source* uređaj šalje *Response* komandu ka *Sink* uređaju. *Response* komanda sadrži vrednost *prve faze* izračunate EC-DH algoritmom, verziju i generaciju SRM poruke koju sadrži *Source* uređaj i potpis na prethodne podatke kreiran EC-DSA algoritmom i upotrebom *privatnog* ključa. Kada primi *Response* komandu *Sink* uređaj proverava ispravnost potpisa upotrebom EC-DSA algoritma i *javnog* ključa. Zatim se proverava verzija i generacija SRM poruke kako bi se ustanovilo da li neki od uređaja poseduje noviju verziju SRM poruke i da li je potrebno ažurirati SRM poruku. *Sink* uređaj šalje odgovor na primljenu komandu kojim obaveštava *Source* uređaj o statusu komande.

Ukoliko je komanda uspešno primljena, *Sink* uređaj takođe šalje *Response* komandu koja sadrži vrednost *prve faze* izračunate EC-DH algoritmom, verziju i generaciju SRM poruke koju sadrži *Sink* uređaj i potpis na prethodne podatke kreiran EC-DSA algoritmom i upotrebom *privatnog* ključa. *Source* uređaj takođe proverava ispravnost potpisa, upoređuje verziju i generaciju SRM poruka i šalje odgovor kojim obaveštava *Sink* uređaj o statusu primljene komande.

Sink i *Source* uređaj računaju autentifikacione ključeve (K_{AUTH} - *Authentication Key*) na osnovu vredosti primljenih *prvih faza*. Autentifikacioni ključevi moraju biti jednaki jer se upotrebljavaju prilikom kreiranja ključeva koji se koriste u algoritmu šifrovanja i dešifrovanja sadržaja.

Kada je *Challenge-Response* povezivanje uspešno završeno *Source* uređaj šalje ka *Sink* uređaju *Exchange_key* komandu koja sadrži slučajnu vrednost generisanu pomoću generatora slučajnog broja. Ova vrednost mora biti kodirana pomoću autentifikacionog ključa (K_{AUTH}). Kada *Sink* uređaj primi ovu komandu dekodira primljenu vrednost, takođe na osnovu autentifikacionog ključa, i dobija slučajnu vrednost kreiranu od strane *Source* uređaja. Ova vrednost se koristi prilikom kreiranja ključa koji se upotrebljava u algoritmu šifrovanja i dešifrovanja sadržaja.

Razmena SRM poruka je moguća ukoliko se poređenjem verzija i generacija SRM poruka, koje poseduju uređaji, ustanovi da neki od uređaja poseduje noviju SRM poruku.

Autentifikacija i razmena ključeva je ovim završena i postignuti su svi potrebni uslovi za razmenu zaštićenog sadržaja.

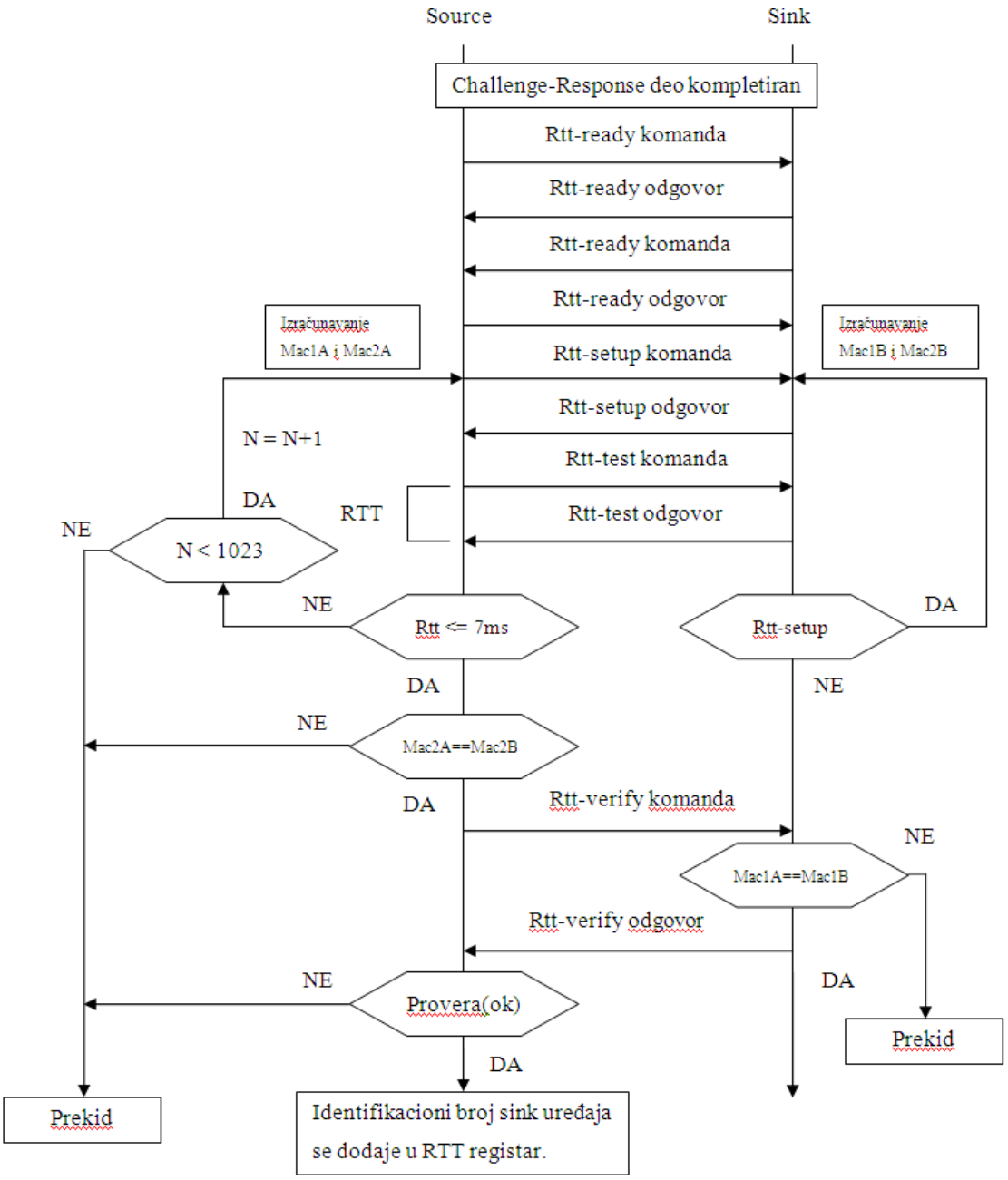
3.2.6 Razmena komandi u RTT proceduri

Uređaji koji su izvor zaštićenog sadržaja i uređaji koji potražuju zaštićeni sadržaj i pritom podržavaju potpunu autentifikaciju moraju da podržavaju i RTT (*Round Trip Time*) proceduru koji se još naziva i dodatna lokalizacija (*AL-Additional Localization*) [6]. Nakon realizacije *Challenge-Response* povezivanja, uređaj koji je izvor sadržaja treba da započne RTT proceduru ukoliko se identifikacioni broj uređaja koji potražuje sadržaj ne nalazi u RTT registru . Uređaj koji je izvor sadržaja dodaje identifikacioni broj uređaja koji potražuje sadržaj u RTT registar ukoliko je RTT procedura uspešno realizovana. Identifikacioni broj uređaja ostaje u RTT registru četrdeset časova nakon uspešno realizovane RTT procedure. Prilikom RTT procedure proverava se RTT period, koji mora da bude manji od sedam milisekundi. Ukoliko je RTT period manji od sedam milisekundi komunikacija između uređaja se nastavlja. Identifikacioni broj uređaja se postavlja u RTT registar i dolazi do razmene ključa, u protivnom komunikacija između uređaja se prekida. *Source* uređaj ažurira RTT registar prilikom slanja sadržaja i ukoliko je identifikacioni broj nekog uređaja duže od četrdeset časova u RTT registru on se automatski izbacuje iz RTT registra. Kada uređaj, čiji se identifikacioni broj nalazi u RTT registru nekog uređaja, zatraži sadržaj sa tog uređaja, RTT procedura se preskače.

Ukoliko je uspešno realizovana autentifikacija uređaja, uređaj koji je izvor sadržaja šalje komandu *Rtt-ready* ka odredištu sadržaja (Slika 3.14). Ovom komandom izvor sadržaja obaveštava uređaj koji je zatražio sadržaj da je spreman za RTT proceduru, time RTT procedura počinje. Uređaj koji potražuje sadržaj, ukoliko je spreman za RTT proceduru, šalje ka izvoru sadržaja takođe *Rtt-ready* komandu.

Kada su *Rtt-ready* komande uspešno razmenjene prelazi se na slanje *Rtt-setup* komande koja sadrži parametar N. Parametar N je na početku RTT procedure postavljen na vrednost nula i svakim sledećim pokretanjem RTT procedure vrednost ovog parametra se poveća za jedan. Vrednost ovog parametra se kreće u opsegu od 0-1023, što znači da je maksimalan broj ponavljanja RTT procedure 1024. RTT procedura se ponavlja ukoliko je RTT period veći od sedam milisekundi. Kada parametar N dostigne vrednost 1024 i ako je vrednost RTT perioda i dalje veća od sedam milisekundi komunikacija između uređaja se prekida. Ukoliko je međutim vrednost RTT perioda manja od sedam milisekundi RTT procedura se nastavlja.

Na osnovu vrednosti parametra N, koji je razmenjen u *Rtt-setup* komandama, i vrednosti autentifikacionog ključa K_{AUTH}, na uređaju koji je izvor sadržaja i uređaju koji je odredište sadržaja odvojeno se računaju MAC vrednosti. MAC vrednosti se na obe strane računaju upotrebom istog algoritma SHA-1 i istih vrednosti parametra N i autentifikacionog ključa K_{AUTH}. Jednakost MAC vrednosti proverava se dalje u RTT proceduri. *Rtt-test* komandama se razmenjuju MAC vrednosti i na svakoj strani se proverava njihova jednakost. Uređaj koji je izvor sadržaja šalje *Rtt-verify* komandu ka uređaju koji je odredište sadržaja ukoliko se ustanovi da su MAC vrednosti jednake. Kada uređaj koji je odredište sadržaja primi *Rtt-verify* poruku takođe proverava jednakost MAC vrednosti. Uređaj koji je odredište sadržaja šalje odgovor na *Rtt-verify* komandu koja sadrži OK vrednost. Ovom komandom se potvrđuje da su MAC vrednosti jednake. OK vrednost se računa na osnovu istih parametara i istim algoritmom kao i MAC vrednosti, jedina promena je što se za vrednost parametra N uzima trenutna vrednost ovog parametra uvećana za jedan.



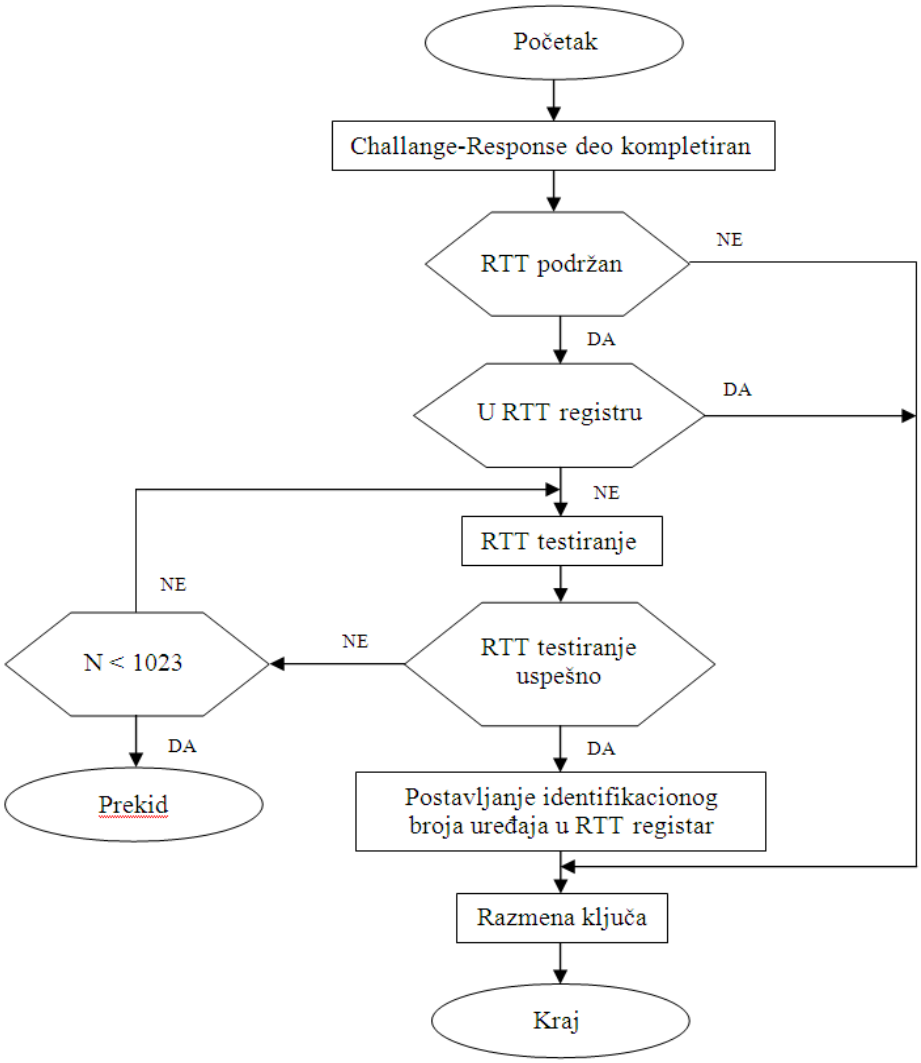
Slika 3.10 Razmena komandi u RTT proceduri

Kada uređaj koji je izvor sadržaja primi odgovor na *Rtt-verify* komandu koji nosi OK vrednost, na uređaju koji je izvor sadržaja se takođe izračunava OK vrednost i proverava se da li je jednaka OK vrednosti koja je primljena.

Ako su OK vrednosti jednake RTT procedura je uspešno realizovana i identifikacioni broj uređaja koji je odredište sadržaja se dodaje u RTT registar. Ukoliko se tokom RTT procedure ustanovi da MAC ili OK vrednosti nisu jednake RTT procedura se prekida.

3.2.6.1 RTT-AKE procedura

RTT-AKE procedura započinje isto kao i normalna AKE procedura. Uređaj koji je izvor sadržaja proverava vrednost AL (*Additional Localization*) polja u svom sertifikatu i ukoliko je vrednost ovog polja postavljena na jedan, vrši se provera vrednosti AL polja u sertifikatu uređaja koji je odredište sadržaja. Ako je vrednost AL polja u sertifikatu uređaja koji je odredište sadržja takođe postavljena na jedan oba uređaja podržavaju RTT proceduru (Slika 3.15). Nakon kompletiranja *Challenge-Response* povezivanja uređaj koji potražuje sadržaj prekida RTT-AKE proceduru ukoliko ne primi *Rtt-ready*, *Exchange-key* ili *Ake-cancel* poruku.



Slika 3.11 RTT-AKE procedura

Na samom početku RTT procedure uređaj koji je izvor sadržaja proverava da li se identifikacioni broj uređaja koji potražuje sadržaj nalazi u RTT registru. Ako je identifikacioni broj uređaja u RTT registru RTT procedura se preskače i odmah se prelazi na razmenu ključa, u suprotnom uređaj koji je izvor sadržaja započinje RTT proceduru.

RTT procedura se ponavlja maksimalno 1024 puta dok se ne dobije RTT period manji od sedam milisekundi. Ako nakon 1024 ponavljanja RTT period ne bude manji od sedam milisekundi RTT-AKE procedura se prekida. Ukoliko je RTT-AKE procedura uspešno završena identifikacioni broj uređaja koji potražuje sadržaj se postavi u RTT registar i tu ostaje narednih četrdeset časova. Zatim se prelazi na razmenu ključa i time je RTT-AKE uspešno završen tako da može da se pređe na šifrovanje sadržaja i njegovu razmenu.

3.2.7 Prenos zaštićenog sadržaja

Multimedijalni sadržaj pre samog slanja mora biti zaštićen. Zaštita sadržaja se obezbeđuje šifrovanjem sadržaja odgovarajućim algoritmom. Algoritam šifrovanja zahteva upotrebu odgovarajućih ključeva koji su dostupni samo uređajima koji su potvrdili autentičnost kroz proces autentifikacije. Prilikom zaštite multimedijalnog sadržaja upotrebom DTCP-IP protokola koristi se AES-128 (*Advanced Encryption Standard*) standard kao osnovni standard za šifrovanje odnosno dešifrovanje podataka.

AES-128 algoritmom vrši se šifrovanje nad blokovima podataka veličine 128 bita i pri tom se upotrebljava ključ za šifrovanje veličine 128 bita [7]. Potrebno je da ključ za šifrovanje i dešifrovanje budu identični kako bi se sadržaj uspešno dešifrovao. Prilikom šifrovanja odnosno dešifrovanja sadržaja upotrebljava se K_C (*Content Key*) ključ. K_C ključ se dobija na osnovu vrednosti koje su razmenjene prilikom autentifikacije i razmene ključa tj. AKE-a.

Nakon šifrovanja multimedijalnog sadržaja moguća je razmena sadržaja između uređaja tj. uređaj koji je izvor sadržaja može da preda šifrovani sadržaj uređaju koji potražuje sadržaj. Uređaj koji potražuje sadržaj mora uspešno da dešifruje sadržaj i time je razmena sadržaja uspešno kompletirana. Šifrovani sadržaj se pakuje u specifičan format (Slika 3.16) koji sadrži PCP (*Protected Content Packet*) zaglavlje i šifrovane podatke poravnate sa 0-15 bajtova.

PCP zaglavlje sadrži polja koja opisuju način šifrovanja podataka tj. vrednosti koje se koriste prilikom šifrovanja podataka [5]. Prilikom dešifrovanja podataka koriste se ove vrednosti kako bi se dešifrovanje uspešno izvršilo.

Polje *tip_paketa* ima vrednost postavljenu na nula (00_2). Polja *C_A2* i *C_A* određuju koji algoritam šifrovanja je upotrebljen prilikom šifrovanja podataka i uz pomoć kog ključa se vrši šifrovanje. Vrednost polja *E-EMI* definiše kontrolu kopiranja sadržaja. Polje N_C predstavlja slučajnu vrednost dobijenu pomoću generatora slučajnog broja. N_C i *E-EMI* vrednosti se koriste prilikom izračunavanja K_C ključa, koji se koristi u algoritmu šifrovanja. Polje *veličina_šifrovanih_podataka* definiše broj bajtova šifrovanih podataka koji se prenosi. Na osnovu vrednosti ovog polja prilikom dešifrovanja se određuje koliko podataka je validno a koliko pripada poravnanju.

Prenos zaštićenih odnosno šifrovanih podataka se vrši pomoću RTP (*Real-time Transport Protocol*) ili HTTP (*Hypertext Transfer Protocol*) protokola.

Ukoliko se sadržaj šalje pomoću RTP protokola uređaj koji je izvor zaštićenog sadržaja generiše slučajnu vrednost N_C pomoću generatora slučajnog broja. Na osnovu N_C vrednosti i E-EMI vrednosti računa se K_C. N_C vrednost se ažurira povećanjem za jedan. Minimalno vreme ažuriranja N_C vrednosti je 30 sekundi a maksimalno 120 sekundi. Kada se ažurira vrednost N_C menja se i K_C.

Ukoliko se sadržaj šalje pomoću HTTP protokola uređaj koji je izvor zaštićenog sadržaja takođe generiše slučajnu vrednost N_C pomoću generatora slučajnog broja. Ukoliko je veličina sadržaja koji se prenosi veća od 128 megabajta N_C vrednost se ažurira na svakih 128 megabajta podataka uvećanjem za jedan, samim tim i K_C se ažurira na svakih 128 megabajta obrađenih podataka.

Veličina PCP paketa može biti promenljiva ali se promenjena veličina PCP paketa mora upisati u PCP zaglavlje kako bi prilikom dešifrovanja bila poznata tačna veličina validnih podataka.

<table>
<tr><th></th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td>Zaglavlje[0]</td><td colspan="2">Tip_paketa</td><td>C_A2</td><td>C_A</td><td colspan="4">E-EMI</td></tr>
<tr><td>Zaglavlje[1]</td><td colspan="8">Oznaka_ključa_razmene</td></tr>
<tr><td>Zaglavlje[2]</td><td colspan="8" rowspan="8">N_C (Slučajna vrednost na osnovu koje se računa K_C)</td></tr>
<tr><td>Zaglavlje[3]</td></tr>
<tr><td>Zaglavlje[4]</td></tr>
<tr><td>Zaglavlje[5]</td></tr>
<tr><td>Zaglavlje[6]</td></tr>
<tr><td>Zaglavlje[7]</td></tr>
<tr><td>Zaglavlje[8]</td></tr>
<tr><td>Zaglavlje[9]</td></tr>
<tr><td>Zaglavlje[10]</td><td colspan="8" rowspan="4">Veličina_šifrovanih_podataka</td></tr>
<tr><td>Zaglavlje[11]</td></tr>
<tr><td>Zaglavlje[12]</td></tr>
<tr><td>Zaglavlje[13]</td></tr>
<tr><td>Podaci[0]</td><td colspan="8" rowspan="6">Šifrovani podaci (poravnati sa 0-15 bajtova)</td></tr>
<tr><td>Podaci[1]</td></tr>
<tr><td>Podaci[2]</td></tr>
<tr><td>-</td></tr>
<tr><td>-</td></tr>
<tr><td>Podaci[N-1]</td></tr>
</table>

Slika 3.12 Format PCP (Protected Content Packet) paketa

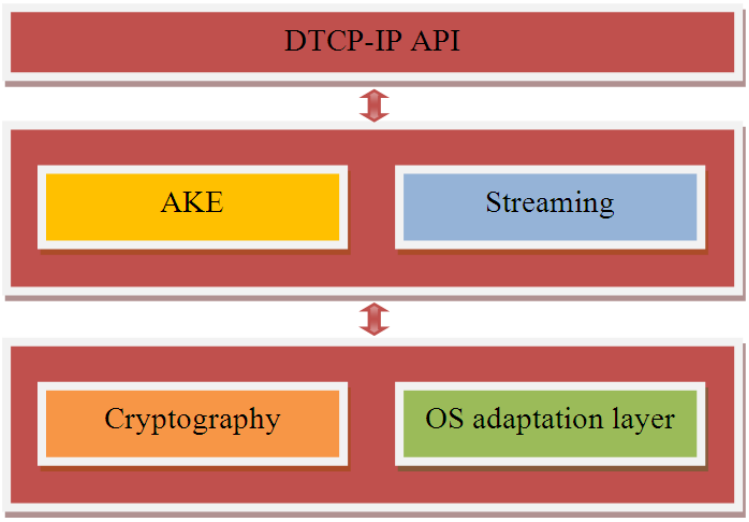
Prilikom pakovanja podataka u PCP zaglavlju se postave parametri pomoću kojih se vršilo šifrovanje podataka, zatim se pakuju šifrovani podaci poravnati sa 0-15 bajtova. Kada se tako sređeni paketi pošalju uređaj koji je odredište sadržaja raspakuje PCP zaglavlje i dobije parametre za dešifrovanje podataka. Kada se podaci dešifruju odgovarajućim parametrima prenos sadržaja je uspešno završen.

Glava 4

Opis i realizacija programske podrške

Programska podrška za zaštitu multimedijalnog sadržaja obezbeđuje sve potrebne funkcionalnosti koje propisuje DTCP-IP standard. Upotrebu ovih funkcionalnosti omogućava DTCP-IP biblioteka. Biblioteka je kreirana tako da je veoma lako ugraditi u već postojeće sisteme programske podrške. DTCP-IP biblioteka ima definisanu slojevitu strukturu koja je prikazana na slici 4.1.

Kompletna struktura je podeljena u tri glavna sloja. API sloj sadrži funkcionalnosti biblioteke koje su dostupne korisnicima. Zatim sloj radnih okvira (*Frameworks*), koji sadrži implementaciju specifičnih DTCP-IP funkcionalnosti. U ovom sloju su implementirana dva karakteristična radna okvira. "AKE" radni okvir koji sadrži implementaciju DTCP-IP AKE podrške i "Streaming" radni okvir, koji sadrži implementaciju podrške za HTTP (*Hypertext Transfer Protocol*) i RTP (*Real-time Transport Protocol*) strimovanje.



Slika 4.1 Struktura DTCP-IP biblioteke

Sloj servisa (*Services*) sadrži osnovne funkcionalnosti potrebne za rad prethodno definisanih radnih okvira. Implementirana su dva karakteristična servisa. Servis kriptografije (*Cryptography*) sadrži kriptografske primitive potrebne za DTCP-IP i OS adaptacioni sloj (*OS Adaptation Layer*) koji sadrži specifične servise operativnog sistema potrebne DTCP-IP biblioteci. Postoje tri grupe zavisnosti na osnovu kojih se izgrađuje biblioteka.

4.1 Standardna C biblioteka

Celokupna programska podrška napisana je pomoću C programskog jezika sa ciljem da bude lako prenosiva na druge platforme. Korišćena je standardna C biblioteka *libc*. Može se prevesti sa bilo kojim prevodiocem koji je C99 kompatibilan.

4.2 Operativni sistem

DTCP-IP biblioteka radi u višenitnom okruženju. Prilikom testiranja realizovana je na Linuks (*Linux*) operativnom sistemu, korišćene su biblioteke *libpthread*, *librt* i *libcrypto*. Međutim veoma jednostavno se može preneti na druge operativne sisteme koji imaju podršku za vremensku kontrolu, vremenske markere (rezolucija u milisekundama), niti i kritične sekcije. DTCP-IP biblioteka zahteva BSD (*Berkeley sockets*) utičnice za mrežnu (IP) komunikaciju između uređaja.

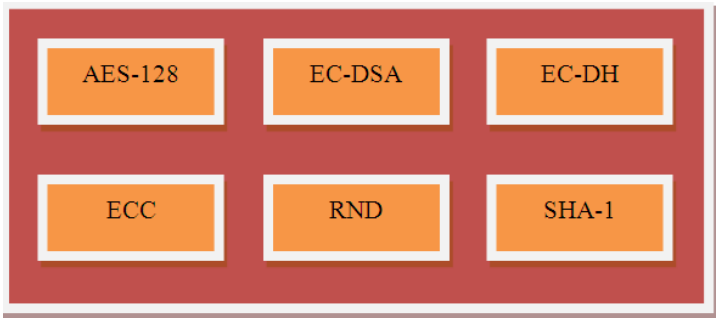
Slika 4.2 OS adaptacioni moduli

4.3 Kriptografija

DTCP-IP funkcionalnosti su bazirane na kriptografskom sistemu eliptične krive (*ECC - Elliptic Curve Cryptography*). Za ovu svrhu se koristi OpenSSL biblioteka (*libcrypto*), koja je statički uvezana sa ostatkom programske podrške. Minimalna zahtevana verzija OpenSSL biblioteke je 1.0.0 [8].

DTCP-IP biblioteka je dizajnirana tako da modul kriptografije može biti zamenjen sa kriptografskim modulom koji je kreirao sam korisnik biblioteke. Ukoliko korisnik želi da zameni modul kriptografije potrebno je da modul kriptografije koji je sam kreirao podržava sledeće kriptografske funkcionalnosti:

1. SHA-1
2. RNG
3. ECC
4. EC-DH
5. EC-DSA
6. AES-128

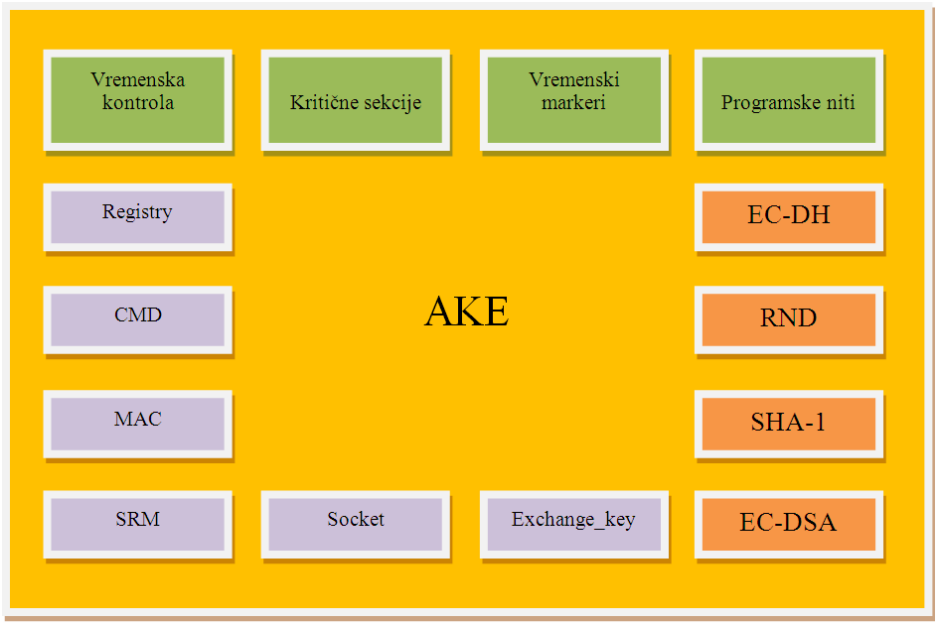


Slika 4.3 Kriptografski moduli

Ukoliko u sistemu postoji fizički realizovan AES-128 modul koji obavlja šifrovanje tj. dešifrovanje, tada je u programskoj podršci je implementirana podrška za njegovu upotrebu. Biblioteka ima male memorijske zahteve. Prevedena na x86, sa statički uvezanom OpenSSL bibliotekom zauzima 156 kB na disku.

4.4 AKE radni okvir

Radni okvir autentifikacije i razmene ključeva realizovan je kao zaseban programski modul. U okviru ovog programskog modula odvija se razmena komandi opisana u AKE proceduri. AKE modul se oslanja na pojedine podmodule koji sadrže realizaciju osnovnih funkcionalnosti koje zahteva AKE modul (Slika 4.4).



Slika 4.4 Struktura *AKE* programskog modula

4.4.1 CMD modul

CMD programski modul sadrži sve funkcionalnosti koje su potrebne za kreiranje i obradu komandi prilikom autentifikacije i razmene ključeva. Upotrebom funkcionalnosti *CMD* programskog modula moguće je kreirati kontrolne ili statusne komande. Kreiranje komandi podrazumeva kreiranje zaglavlja komandi i informacija koje se prenose komandom. Funkcionalnosti *CMD* programskog modula takođe pružaju mogućnost razmene kreiranih komandi, odnosno podržane su funkcionalnosti za obradu zaglavlja primljenih komandi i obradu informacija koje komande nose, bilo da se radi o statusnim ili kontrolnim komandama.

4.4.2 Socket modul

Socket modul sadrži funkcionalnosti potrebne za manipulaciju BSD utičnicama (*BSD sockets*) neophodnim za komunikaciju između uređaja prilikom autentifikacije i razmene ključeva. Kako bi slanje i primanje komandi bilo moguće potrebno je kreirati utičnicu, povezati je sa odgovarajućim portom i IP adresom. Kada je utičnica uspešno kreirana i povezana sa odgovarajućim portom i IP adresom preko nje je moguće slati i primati informacije, odnosno komande. Sve funkcionalnosti vezane za utičnice koje su potrebne prilikom autentifikacije i razmene ključeva realizovane su u *Socket* programskom modulu.

4.4.3 Registry modul

Prilikom potpune autentifikacije može da dođe do razmene RTT komandi, tj. može da se uključi RTT procedura ukoliko je to potrebno. Ukoliko oba povezana uređaja podržavaju RTT proceduru nakon uspostavljanja *Challenge-Response* povezivanja prelazi se na razmenu RTT poruka. Pre pokretanja RTT procedure uređaj koji je izvor zaštićenog sadržaja proveri da li se u RTT registru nalazi identifikacioni broj uređaja koji potražuje sadržaj. Ukoliko se identifikacioni broj uređaja nalazi u RTT registru RTT procedura se preskače u protivnom pokreće se RTT procedura. Ako se RTT procedura uspešno realizuje identifikacioni broj uređaja se smešta u RTT registar i tu ostaje sledećih četrdeset časova. Sve potrebne funkcionalnosti vezane za RTT registar kao što su postavljanje identifikacionog broja uređaja u registar, uklanjanje identifikacionog broja uređaja iz registra i provera registra, realizovane su *Registry* programskom modulu.

4.4.4 SRM modul

Potpuna autentifikacija uređaja predviđa i razmenu SRM poruka. Ažuriranje SRM poruka se događa ukoliko jedan od povezanih uređaja poseduje noviju SRM poruku. Sve funkcionalnosti koje su potrebna za obradu SRM poruka, učitavanje SRM poruka, poređenje verzija i generacija SRM poruka, kreiranje SRM liste uređaja na osnovu SRM poruke, pretraživanje SRM liste na osnovu identifikacionog broja uređaja, realizovanje su u *SRM* programskom modulu.

4.4.5 MAC modul

Prilikom potpune autentifikacije uređaja moguća je razmena RTT komandi tj. može da se uključi RTT procedura. Pojedine RTT komande razmenjuju MAC odnosno OK vrednosti. Nakon razmene MAC, odnosno OK vrednosti, vrši se njihovo poređenje. Sve funkcionalnosti vezane za kreiranje i poređenje MAC i OK vrednosti realizovane su u *MAC* programskom modulu.

4.4.6 Exchange_key modul

Prilikom potpune autentifikacije razmenjuje se vrednost kreirana pomoću generatora slučajnog broja, ova vrednost se naziva *ključ_razmene* (*Exchange Key*). Sve potrebne funkcionalnosti za kreiranje i manipulaciju ovom vrednošću realizovane su u *Exchange_key* programskom modulu.

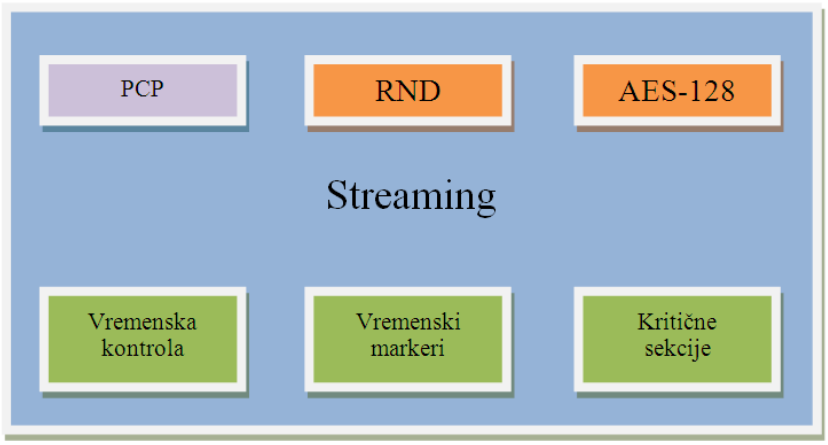
4.4.7 Kriptografski moduli i moduli OS adaptacionog sloja

Realizacija AKE programskog modula zahteva upotrebu kriptografskih alata. Kriptografski alati korišćeni kroz proces autentifikacije i razmene ključeva realizovani su u OpenSSL biblioteci. Za svaki kriptografski alat, koji se koristi prilikom autentifikacije i razmene ključeva, takođe je realizovan zaseban programski podmodul. Programski podmoduli EC-DH, EC-DSA, RND i SHA-1 sadrže realizaciju kriptografskih funkcionalnosti koje se koristi prilikom autentifikacije uređaja i razmene ključeva. Funkcionalnosti koje pruža EC-DH modul koriste se za kreiranja vrednosti *prve_faze* prilikom razmene komandi u AKE-u. Funkcionalnosti EC-DSA modula koriste se prilikom kreiranja i provere DTLA potpisa. RND modul pruža mogućnost kreiranja slučajne vrednosti prilikom uspostavljanja vrednosti *ključ_razmene.* Funkcionalnosti koje pruža SHA-1 modul imaju upotrebu prilikom kreiranja MAC i OK vrednosti u RTT proceduri.

Realizacija AKE programskog modula zahteva podršku za vremensku kontrolu, vremenske markere (rezolucija u milisekundama), niti i kritične sekcije, koji su realizovani kao podmoduli u OS adaptacionom sloju.

4.5 Streaming radni okvir

Streaming radni okvir realizovan je kao zaseban programski modul. Sadrži implementaciju programske podrške neophodne prilikom HTTP (*Hypertext Transfer Protocol*) i RTP (*Real-time Transport Protocol*) strimovanja podataka. U ovom programskom modulu realizovane su funkcionalnosti neophodne za šifrovanje i dešifrovanje podataka bilo da se radi o HTTP ili RTP strimovanju, kao i sve funkcionalnosti neophodne za kreiranje PCP paketa. PCP paketi se različito kreiraju u zavisnosti da li se radi o HTTP ili RTP strimovanju podataka, zbog toga su sve funkcionalnosti ovog programskog modula podeljene u dve grupe. Grupa funkcionalnosi koje su neophodne prilikom HTTP strimovanja i grupa funkcionalnosti koje su neophodne prilikom RTP strimovanja.



Slika 4.5 Struktura *Streaming* programskog modula

Streaming programski modul se oslanja na pojedine programske podmodule koji sadrže realizaciju osnovnih funkcionalnosti koje zahteva *Streaming* programski modul (Slika 4.5).

4.5.1 *PCP* modul

PCP programski modul sadrži realizaciju funkcionalnosti neophodnih za manipulacija PCP zaglavljima. Pod manipulacijom PCP zaglavljima podrazumeva se kreiranje PCP zaglavlja, čitanje PCP zaglavlja, promena vrednosti polja u PCP zaglavljima, ažuriranje PCP zaglavlja kao i pakovanje i raspakivanje PCP zaglavlja. Ovaj programski modul je jedinstven bilo da se o HTTP ili RTP strimovanju podataka.

4.5.2 Kriptografski moduli i moduli OS adaptacionog sloja

Prilikom realizacije *Streaming* programskog modula zahteva se upotreba kriptografskih alata. Potrebni kriptografski alati realizovani su u RND i AES-128 programskim podmodulima. Funkcionalnosti RDN programskog modula pružaju mogućnost kreiranja slučajne vrednosti prilikom ažuriranja PCP zaglavlja. AES-128 modul sadrži neophodne funkcionalnosti za šifrovanje odnosno dešifrovanje podataka.

Realizacija *Streaming* programskog modula zahteva podršku za vremensku kontrolu, vremenske markere (rezolucija u milisekundama) i kritične sekcije, koji su realizovani kao podmoduli u OS adaptacionom sloju.

Glava 5

Pristup testiranju i rezultati testiranja

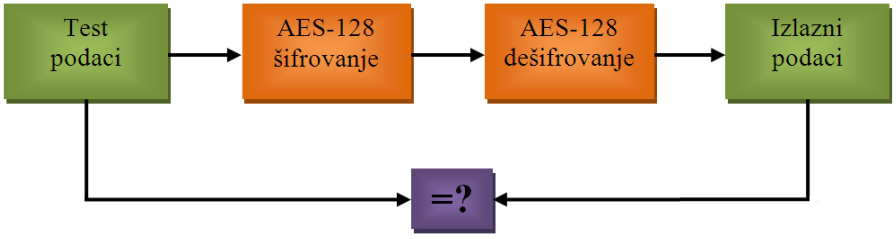
5.1 Testiranje na nivou modula

Većina modula programske podrške testirana je pojedinačno i za većinu modula je implementiran poseban test modul. Svi kriptografski alati koji se koriste prošli su testiranje na nivou modula. Kreirani su posebni testovi za SHA-1, AES-128, EC-DSA i EC-DH module, u okviru ovih testova indirektno su testirani ECC i RND programski moduli. Takođe su kreirani testovi za programske module SRM, MAC i *Registry*. Svi testovi su uspešno prošli proveru.

- SHA-1 test je kreiran kako bi se proverile funkcionalnosti SHA-1 algoritma (Slika 5.1). SHA-1 test sadrži primenu SHA-1 algoritma nad odgovarajućim skupom podataka (Test podaci). Kao rezultat primene SHA-1 algoritma dobija se niz podataka (Izlazni podaci) koji se potom porede sa referentnim nizom podataka i generiše se odgovarajući izveštaj. Ukoliko su dobijeni nizovi identični SHA-1 test je uspešan u suprotnom test nije uspešan.

Slika 5.1 Testiranje SHA-1 modula

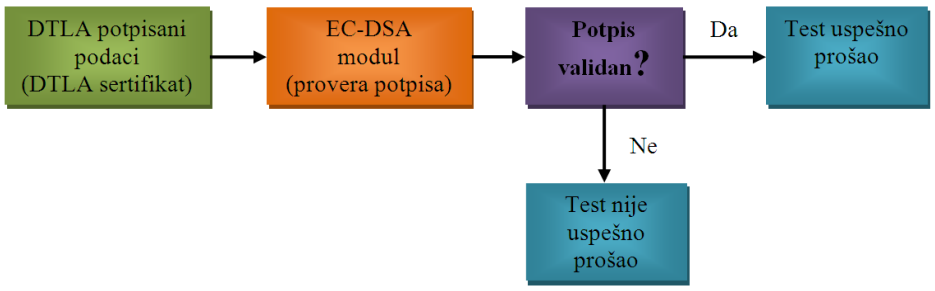
- AES-128 test kreiran je kako bi se proverile kriptografske funkcionalnosti tj. kako bi se proverila tačnost šifrovanja odnosno dešifrovanja podataka (Slika 5.2). Test sadrži primenu AES-128 algoritma šifrovanja nad referentnim nizom podataka (Test podaci). Šifrovanje i dešifrovanje se vrši pomoću odgovarajućeg AES konteksta koji se kreira na osnovu odgovarajućeg ključa.



Slika 5.2 Testiranje AES modula

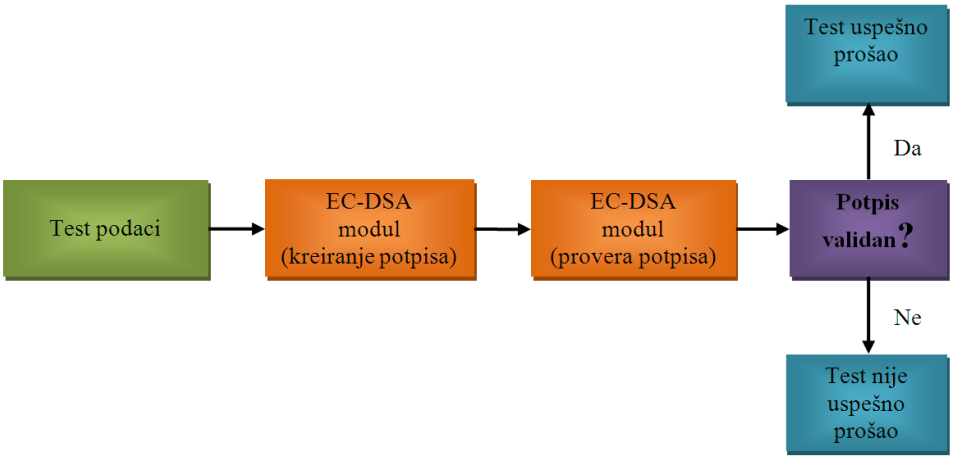
Kao rezultat šifrovanja dobija se niz šifrovanih podataka nad kojim se potom primeni AES-128 algoritam dešifrovanja podataka uz upotrebu identičnog AES-128 konteksta. Niz podataka dobijen dešifrovanjem podaka potom se poredi se sa referentnim nizom podataka i generiše se odgovarajući izveštaj. Ukoliko su nizovi podataka identični test je uspešan u suprotnom test nije uspešan.

- EC-DSA test kreiran je kako bi se proverile funkcionalnosti kreiranja i provere EC-DSA potpisa. Kao referentni skup podataka koristi se validan DTLA sertifikat, DTLA javni ključ i validan privatni ključ. U prvom delu testa kreira se EC-DSA kontekst uz upotrebu DTLA javnog ključa. Tako kreiran EC-DSA kontekst koristi se za proveru DTLA potpisa nad DTLA sertifikatom i generiše se odgovarajući izveštaj. Ukoliko se ustanovi da je potpis validan prvi deo testa je uspešan u suprotnom test nije uspešan (Slika 5.3).



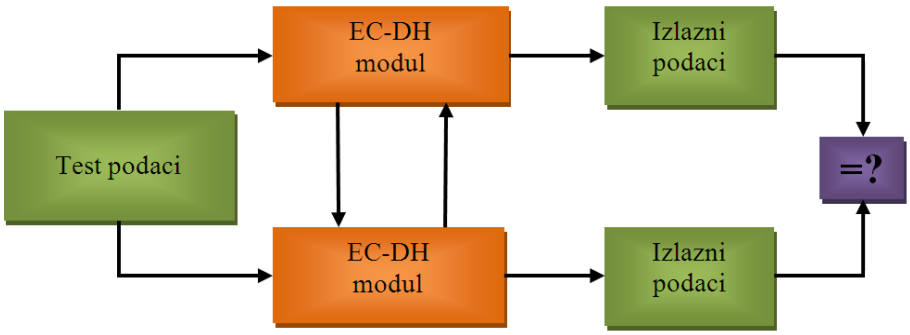
Slika 5.3 Testiranje EC-DSA modula (1)

Drugi deo EC-DSA testa sadrži kreiranje EC-DSA konteksta uz upotrebu privatnog i javnog ključa. Kreirani kontekst se koristi za kreiranje potpisa nad proizvoljnim skupom podataka a zatim se isti kontekst koristi za proveru dobijenog potpisa i generiše se odgovarajući izveštaj. Ukoliko je dobijeni potpis validan test je uspešan u suprotnom test nije uspešan (Slika 5.4).



Slika 5.4 Testiranje EC-DSA modula (2)

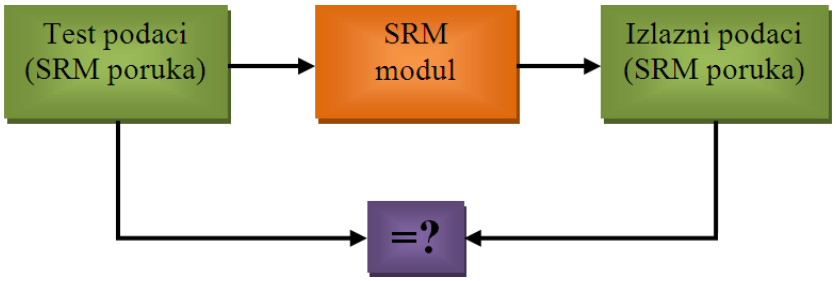
- EC-DH test je kreiran kako bi se proverile funkcionalnosti kreiranja EC-DH konteksta, kreiranja prve faze i autentifikacionog ključa na osnovu kreiranog EC-DH konteksta. U testu se kreiraju dva EC-DH konteksta na osnovu parametara identične eliptične krive. EC-DH konteksti se koriste kako bi se kreirale prve faze a zatim se kombinovanjem prve faze sa odgovaraujćim EC-DH kontekstom kreiraju dva zasebna autentifikaciona ključa. Poređenjem ova dva autentifikaciona kjluča proverava se da li je test uspešan ili ne, ukoliko su autentifikacioni ključevi jednaki test je uspešan u suprotnom test nije uspešan (Slika 5.5).



Slika 5.5 Testiranje EC-DH modula

Prilikom kreiranje prethodno navedenih testova koriste se funkcionalnosti EC i RND modula. Upotreba EC modula podrazumeva kreiranje EC konteksta koji se upotrebljava dalje u EC-DSA i EC-DH testovima, tako da su EC-DSA i EC-DH testovi obuhvatili testiranje EC i RND modula.

- SRM test je kreiran kako bi se proverile funkcionalnosti koje pruža SRM modul. Kako bi se proverila ispravnost SRM poruke kreira se EC-DSA kontekst na osnovu DTLA javnog ključa. Zatim se upotrebom EC-DSA konteksta proveri isravnost potpisa na niz podataka koji predstavljaju validnu SRM poruku.



Slika 5.6 Testiranje SRM modula

Ukoliko se ustanovi da je SRM poruka ispravno potpisana kreira se SRM lista na osnovu niza podataka koji predstavlja validnu SRM poruku. Svaki uređaj iz SRM poruke kreira se kao zaseban član SRM liste. Zatim se na osnovu dobijene liste inverzno kreira niz podataka. Poređenjem dobijenog niza podataka sa nizom podataka koji predstavlja validnu SRM poruku proverava se da li je test uspešan i generiše se odgovarajući izveštaj. Ukoliko su nizovi identični SRM test je uspešan u suprotnom test nije uspešan (Slika 5.6).

- MAC test je kreiran kako bi se proverile funkcionalnosti koje pruža MAC modul. MAC vrednost se kreira na osnovu referentnog autentifikacionog ključa upotrebom funkcionalnosti koje pruža MAC modul. Dobijena MAC vrednost se poredi sa referentnom MAC vrednošću i generiše se odgovarajući izveštaj. Ukoliko su MAC vrednosti jednake test je uspešan u suprotnom test nije upešan (Slika 5.7).

Slika 5.7 Testiranje MAC modula

- *Registry* test je kreiran kako bi se proverile funkcionalnosti koje pruža *Registry* modul. Kreira se registar uređaja kao dvostruko spregnuta lista. Zatim se u registar dodaje nekoliko uređaja, sa odgovarajućim podacima o uređajima, tu su pre svega identifikacioni broj uređaja i adresa uređaja. Funkcionalnosti *Registry* modula omogućavaju da se registar pretražuje na osnovu identifikacionog broja ili na osnovu adrese uređaja. U testu se proveravaju obe ove funkcionalnosti, pretražuje se registar na osnovu identifikacionog broja i na osnovu adrese uređaja. Zatim se jedan od uređaja uklanja iz registra uređaja. Sve ove funkcionalnosti se proveravaju u *Registry* testu i generiše se odgovarajući izveštaj. Ukoliko se neka od funkcionalnosti ne realizuje uspešno test nije uspešan u suprotnom *Registry* test je uspešan.

5.1.1 Rezultati testiranja na nivou modula

Testiranjem na nivou modula dobijeni su očekivani rezultati, svi testovi su uspešno prošli proveru (Tabela 5.1).

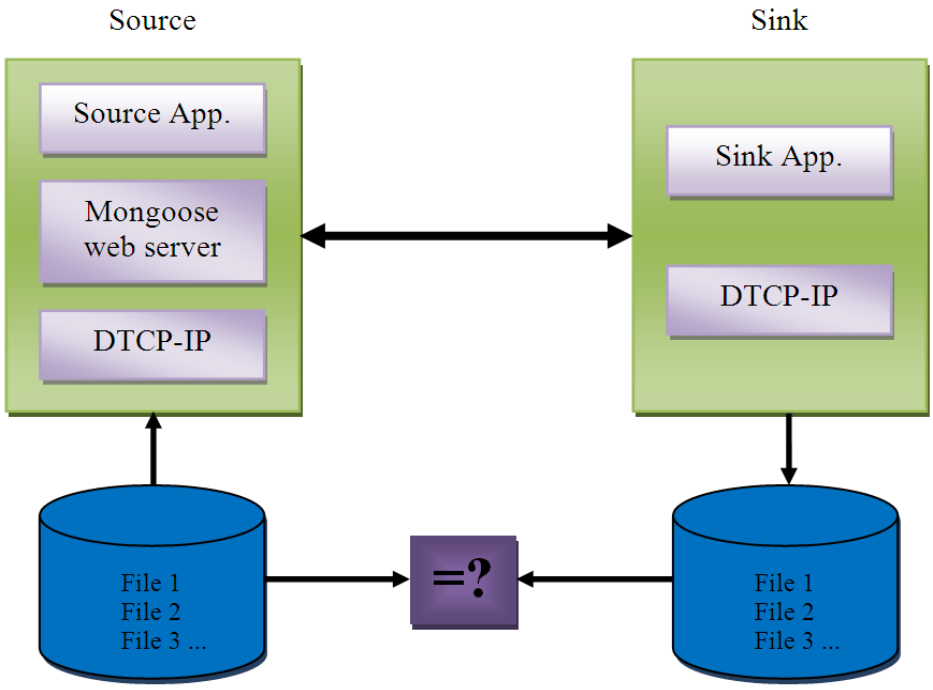
Naziv modul testa	Uspešan	Neuspešan
SHA-1	✓	
AES-128	✓	
EC-DSA	✓	
EC-DH	✓	
SRM	✓	
MAC	✓	
Registry	✓	

Tabela 5.1 Rezultati testiranja na nivou modula

5.2 Testiranje sistema

Određeni programski moduli testirani su pojedinačno kroz testiranje na nivou modula, dok su ostali programski moduli testrani kroz testove celokupnog sistema. U okviru ovih testova uključene su sve funkcionalnosti koje pruža DTCP-IP biblioteka.

Za testiranje celokupnog sistem kreirane su posebne test aplikacije *Sink* i *Source*. Slika 5.8 prikazuje arhitekturu prilikom testiranja celokupnog sistema. U okviru *Source* aplikacije kreira se uređaj koji ima svoj HTTP server (koristi se *Mongoose web server*) i predstavlja izvor multimedijalnog sadržaja - *Source* uređaj. U okviru *Sink* aplikacije kreira se uređaj koji potražuje multimedijalni saržaj - *Sink* uređaj. *Sink* uređaj se prvo poveže sa *Source* uređajem na osnovu odgovarajućeg porta i adrese. Kada se uređaji povežu *Sink* uređaj započinje AKE proceduru. AKE procedura je neophodna kako bi se potvrdila autentičnost uređaja i kako bi se izvršila razmena ključeva. Kada *Source* uređaj primi poruku kojom se započinje AKE procedura proverava se ispravnost primljene poruke. Ukoliko je primljena poruka ispravna AKE procedura se nastavlja u protivnom se prekida. Nakon uspešno realizovanog AKE-a *Source* uređaj prima HTTP zahtev na odgovarajućem portu. Na osnovu ovog HTTP zahteva potražuje se multimedijalni sadržaj. Kada primi HTTP zahtev *Source* uređaj vrši šifrovanje multimedijanog sadržaja i omogućava njegovo preuzimanje *Sink* uređaju. Kada *Sink* uređaj preuzme šifrovani sadržaj, vrši se njegovo dešifrovanje i skladištenje.



Slika 5.8 Testiranje sistema (*Sink* i *Source*)

Nakon preuzimanja i dešifrovanja multimedijalnog sadržaja može se proveriti uspešnost razmene sadržaja poređenjem jednakosti multimedijalnih fajlova koji su šifrovani na *Source* uređaju i multimedijalnih fajlova koji su dobijeni nakon dešifrovanja na *Sink* uređaju.

5.2.1 Rezultati testiranja sistema

Prilikom testiranja *Sink* i *Source* uređaja tj. prilikom testiranja sistema korišćen je multimedijalni sadržaj različitog formata i različitih veličina. Rezultati testiranja sistema za različite formate i veličine fajlova predstavljani su u tabeli 5.2.

Slike		**Video fajlovi**		**Audio fajlovi**	
Format	Rezultat	Format	Rezultat	Format	Rezultat
jpg	✓	mpeg	✓	wav	✓
bmp	✓	avi	✓	mp3	✓
png	✓	wmv	✓	wma	✓
				acc	✓

Tabela 5.2 Rezultati tesiranja sistema

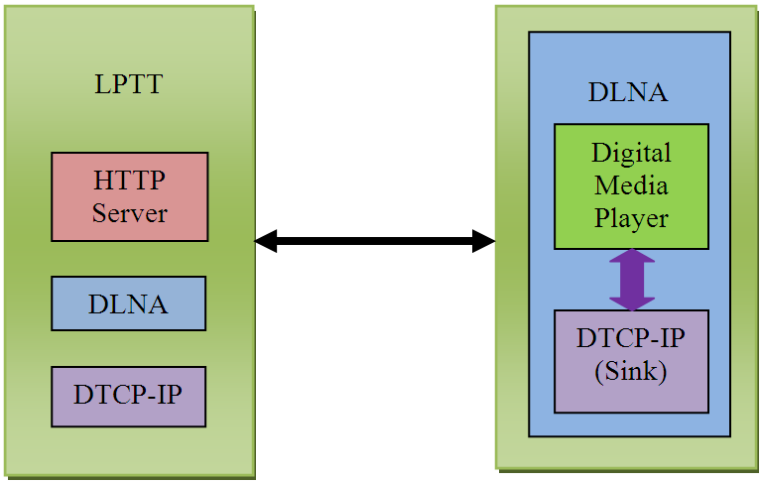
Prilikom testiranja *Sink* i *Source* uređaja korišćen je multimedijalni sadržaj različitih veličina, od svega nekoliko bajtova do nekoliko stotina megabajta. Svi fajlovi su uspešno razmenjeni tako da su *Sink* i *Source* testovi uspešno prošli testiranje čime je potvrđeno uspešno testiranje ostalih programskih modula za koje ne postoje pojedinačni modul testovi.

5.3 Testiranje LPTT alatom

Programska podrška zaštite multimedijalnog sadržaja pomoću DTCP-IP protokola prošla je testiranje LPTT (*Link Protection Test Tool*) alatom koji je obezbedila DLNA organizacija. LPTT alat je namenjen za testiranje DLNA programske podrške sa ugrađenom programskom podrškom za zaštitu multimedijalnog sadržaja upotrebom DTCP-IP protokola. LPTT alat je dostupan samo registrovanim DLNA članovima. Ovim testiranjem je potvrđeno da programska podrška za DTCP-IP ispunjava sve odrednice definisane DTLA specifikacijom.

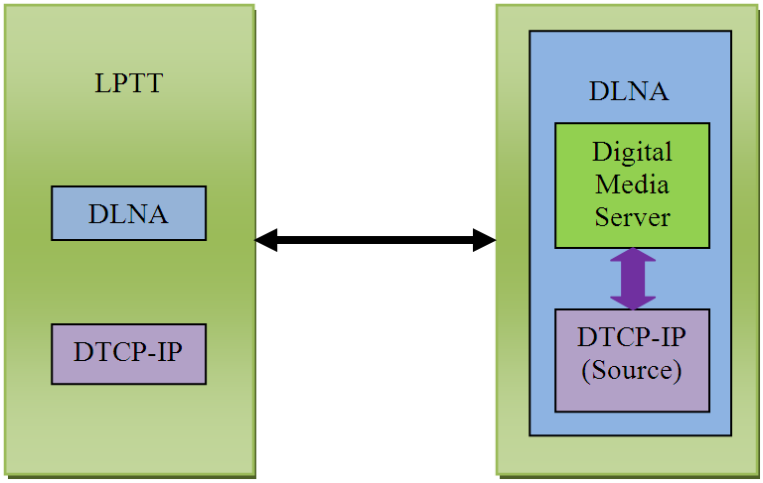
Prilikom testiranja programske podrške za zaštitu multimedijalnog sadržaja upotrebom DTCP-IP protokola koristi se grupa multimedijalnih fajlova koje je obezbedila DLNA organizacija. Ova grupa fajlova sadrži multimedijalne fajlove (video fajlovi, muzički fajlovi i slike) različitog formata i dimenzija.

LPTT alatom se odvojeno testiraju DMS (Slika 5.9) i DMP uređaj (Slika 5.10) sa ugrađenom programskom podrškom za zaštitu multimedijalnog sadržaja.



Slika 5.9 Testiranje DMP uređaja LPTT alatom

Prilikom testiranja DMS uređaja potrebno je da DMS uređaj takođe koristi multimedijalne fajlove koje je obezbedila DLNA organizacija i pre samog pokretanja LPTT alata kojim se testira DMS uređaj potreno je da DMS uređaj bude aktivan. LPTT alat se mora podesiti pre samog testiranja uređaja bilo da se radi o DMS ili DMP uređaju.



Slika 5.10 Testiranje DMS uređaja LPTT alatom

LPTT testovi sadrže nekoliko grupa testova. Svaka grupa testova sadrži po nekoliko podtestova. Za pojedine testove ne postoji podrška jer LPTT alat treba podesiti tako da se testiraju samo funkcionalnosti koje su podržane u realizaciji DTCP-IP programske podrške. Rezultati testiranja LPTT alatom su predstavljeni u tabelama 5.3 i 5.4.

5.3.1 Rezultati testiranja LPTT alatom

Naziv grupe testova	**Rezultati testiranja**
Kvalitet servisa (*QoS - Quality of service*)	Ova grupa testova sadrži jedan test koji je uspešno prošao testiranje.
Upravljanje medijima (*Media Management*)	Ova grupa testova sadrži tri testa, svi testovi su uspešno prošli testiranje.
Transport medija (*Media Transport*)	Ova grupa testova sadrži devet testova, osam testova iz ove grupe nisu podržani za trenutna podešavanja LPTT alata dok je jedan test uspešno prošao testiranje.
DTCP-IP	Ova grupa testova sadrži pet testova, tri testa iz ove grupe nisu podržana za trenutna podešavanja LPTT alata dok su dva testa uspešno prošla testiranje.

Tabela 5.3 Rezultati testiranja DMP uređaja LPTT alatom

Naziv grupe testova:	Rezultati testiranja:
Kvalitet servisa (*QoS - Quality of service*)	Ova grupa testova sadrži jedan test koji je uspešno prošao testiranje.
Upravljanje medijima (*Media Management*)	Ova grupa testova sadrži petnaest testova, deset testova iz ove grupe nisu podržani za trenutna podešavanja LPTT alata dok je pet testova uspešno prošlo testiranje.
Transport medija (*Media Transport*)	Ova grupa testova sadrži dvadeset i šest testova, dvadeset testova iz ove grupe nisu podržani za trenutna podešavanja LPTT alata dok je šest testova uspešno prošlo testiranje.
Uputstva tehnologije za zaštitu sadržaja	Ova grupa testova sadrži dva testa, oba testa su uspešno prošla testiranje.
DTCP-IP	Ova grupa testova sadrži devet testova, tri testa iz ove grupe nisu podržana za trenutna podešavanja LPTT alata dok je šest testova uspešno prošlo testiranje.

Tabela 5.4 Rezultati testiranja DMS uređaja LPTT alatom

Testovi za koje trenutno ne postoji podrška ne utiču na rezultate testiranja. Proširenjem funkcionalnosti programske podrške za zaštitu multimedijalnog sadržaja upotrebom DTCP-IP protokola, biće realizovane nove funkcionalnosti za koje trenutno ne postoji podrška i tada će biti moguće uključiti i ove testove prilikom testiranja. Iako trenutnom implementacijom nisu podržani sve mogućnosti koje testira LPTT alat, implementacija jeste kompletna imajući u vidu početne zahteve.

Glava 6

Zaključak

U radu je prikazana realizacija programske podrške za zaštitu multimedijalnog sadržaja pomoću DTCP-IP protokola. Kompletna struktura programske podrške je podeljena u tri osnovna sloja. API sloj sadrži funkcionalnosti biblioteke koje su dostupne korisnicima, sloj radnih okvira koji sadrži implementaciju specifičnih DTCP-IP funkcionalnosti i sloj servisa koji sadrži osnovne funkcionalnosti potrebne za rad radnih okvira. Realizacija u potpunosti ispunjava sve odrednice definisane DTLA specifikacijom, što je potvrđeno kroz testiranje na nivou modula, testiranje sistema kao i testiranje LPTT alatom. Realizovana programska biblioteka je upotrebljiva u proizvodima potrošačke elektronike, što je i bio jedan od ciljeva rada.

Celokupna programska podrška razvijena uz upotrebu C programskog jezika i posebna pažnja je posvećenja prilagođavanju programskog koda brojnim platformama. DTCP-IP funkcionalnosti su bazirane na kriptografskom sistemu eliptične krive. Za ovu svrhu se koristi OpenSSL biblioteka.

Dalji razvoj zasniva se na proširivanju programske podrške dodacima koje predviđa proširena DTLA specifikacija. Takođe se predviđa deljenje DTCP-IP biblioteke na dva konteksta: zaštićeni (izvršavaju se kriptografski algoritmi i ima pristup privatnom ključu) i nezaštićeni (podrška za mrežu). Između ova dva konteksta potrebno je napraviti jasno razgraničenje, tako da se samostalno prevode i na nedvosmislen način komuniciraju. Takođe se predviđa implementacija portabilnih kriptografskih primitiva. Algoritmi treba da se razvijaju kao C99 kompatibilni, da se izvršavaju što brže i da su nezavisni od operativnog sistema, čime bi se postigla maksimalna prenosivost. Ovim se eliminiše potreba za OpenSSL bibliotekom i obezbeđuje se kompaktnija programska podrška.

Dodatak A

Lista skraćenica

AES	Advanced Encryption Standard
AKE	Authentication and Key Exchange
AL	Additional Localization
CCI	Copy Control Information
CPTWG	Copy Protection Technical Working Group
CRL	Certificate Revocation List
DLNA	Digital Living Network Allience
DRM	Digital Rights Management
DSS	Digital Signature Standard
DTCP	Digital Transmission Content Protection
DTLA	Digital Transmission Licensing Administrator
ECC	Elliptic Curve Cryptography
EC-DH	Elliptic Curve Diffie-Hellman
EC-DSA	Elliptic Curve Digital Signature Algorithm
EMI	Encryption Mode Indicator
HTTP	Hypertext Transfer Protocol
LPT	Link Protection Technology
LPTT	Link Protection Test Tool
MP	Media Player
PCP	Protected Content Packet
RND	Random
RTP	Real-time Transport Protocol
RTT	Round Trip Time
SHA	Secure Hash Algorithm
SRM	System Renewability Messages

Literatura

[1] DLNA, Digital Living Network Alliance, http://www.dlna.org/

[2] DTCP, Digital Transmission Content Protection, http://en.wikipedia.org/wiki/Digital_Transmission_Content_Protection

[3] M. Reply, C.B.S. Traw, S. Balogh, M. Reed, "Conten Protection in the Digital Home", vol 6, 15. Nov, 2002, Intel Tehnology Journal http://www.intel.com/technology/itj/2002/volume06issue04/art05_protection/vol6iss4_art05.pdf

[4] DigitalTransmission Content Protection Specification (Informational version), vol 1, 14. Dec 2011 http://www.dtcp.com/documents/dtcp/info-20111214-dtcp-v1-rev-1-p-7.pdf

[5] DTCP, vol 1, Supplement F, Mapping DTCP to IP (Informational version), 12 Dec 2011, http://www.dtcp.com/documents/dtcp/info-20111214-dtcp-v1se-ip-rev-1-p-4-ed-1.pdf

[6] DTCP 1394 Additional Localization, vol 1, Supplement F (Informational version), 15 Jun 2011, http://www.dtcp.com/documents/dtcp/info-20070615-dtcp-v1sf-rev-1-p-0.pdf

[7] Advanced Encryption Standard, http://en.wikipedia.org/wiki /Advanced_Encryption_Standard

[8] OpenSSL, http://www.openssl.org/