HRVATSKA AKADEMSKA I ISTRAŽIVAČKA MREŽA
CROATIAN ACADEMIC AND RESEARCH NETWORK

Bluetooth sigurnost

CCERT-PUBDOC-2005-04-118

CARNet CERT u suradnji s LS&S

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

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

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

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

Sadržaj

1. UVOD 4

2. BLUETOOTH TEHNOLOGIJA 5

2.1. MODELI KORIŠTENJA 5
2.2. NAČINI POVEZIVANJA 5
2.3. BLUETOOTH ARHITEKTURA 6
2.3.1. Bluetooth Radio sloj 7
2.3.2. *Baseband* sloj 7
2.3.3. Format Bluetooth paketa 8

3. SIGURNOST BLUETOOTH KOMUNIKACIJE 8

3.1. GENERATORI SLUČAJNIH BROJEVA 9
3.2. UPRAVLJANJE KLJUČEVIMA 9
3.2.1. Tipovi ključa veze 9
3.2.2. Generiranje ključa i inicijalizacija 10
3.3. AUTENTIKACIJA 13
3.4. ENKRIPCIJA 14
3.4.1. Pregovaranje o duljini ključa enkripcije 16
3.5. ALGORITMI AUTENTIKACIJE I GENERIRANJA KLJUČEVA 16
3.5.1. Algoritam autentikacije E_1 16
3.5.2. Algoritam generiranja ključeva za autentikaciju E_2 16
3.5.3. Algoritam generiranja ključa enkripcije E_3 17
3.6. SIGURNOSNI PROBLEMI 18

4. ZAKLJUČAK 18

5. REFERENCE 18

1. Uvod

Od samog početka računalne industrije, kablovi su primarno bili korišteni za međusobno povezivanje računala i raznih perifernih uređaja. U tom smjeru s vremenom su razvijene i brojne sigurnosne kontrole za zaštitu podataka koji se prenose žičanim računalnim mrežama kako bi se osigurala njihova povjerljivost, integritet i raspoloživost.

Međutim, s vremenom su se također pojavila i određena ograničenja žičanih mreža te se i računalna industrija okrenula razvijanju okruženja u kojem će se komunikacija odvijati bez fizičke povezanosti. Bežično umrežavanje posljednjih je godina iznimno dobilo na popularnosti, a predviđanja su da će u budućnosti ovaj trend dodatno rasti. No, osim brojnih prednosti koje bežične tehnologije dosnose u odnosu na tradicionalne žičane mreže, treba voditi računa i o sigurnosnim problemima koji se mogu javiti kao posljedica njihovog korištenja.

Jedna od tehnologija za bežično povezivanje uređaja na kretkim udaljenostima je i Bluetooth tehnologija, o kojoj će biti nešto više riječi u ovom dokumentu. Treba napomenuti da ciljevi ove tehnologije nisu bili usmjereni prema razvijanju novih standarda za implementaciju lokalnih bežičnih računalnih mreža (WLAN), kojih je trenutno već dosta na tržištu, a mnoge se još razvijaju. Dok je WLAN tehnologija namijenjena spajanju velikog broja korisnika, Bluetooh se orijentira na spajanje mobilnih uređaja privatnim vezama. Bluetooth tehnologija, uz ostale bežične tehnologije, otvara nova područja primjene osobnih računala, ručnih računala i ostalih prijenosnih uređaja. Novim područjima primjene otvaraju se i nove sigurnosne ranjivosti, koje su posebno osjetljivije upravo zbog svojstava bežičnog medija. Iz tog razloga je sigurnosti Bluetooth tehnologije potrebno posvetiti dodatnu pažnju. U ovom dokumentu opisana je Bluetooth bežična tehnologija, s posebnim naglaskom na njezine sigurnosne aspekte.

2. Bluetooth tehnologija

Bluetooth je tehnologija namijenjena bežičnim komunikacijama za kratke domete. Osnovna ideja je zamijeniti kablove mobilnih uređaja radio valovima. Bluetooth je razvila grupa poznata pod imenom Bluetooth Special Interest Group (SIG), koja je formirana u Svibnju 1998. godine. Članovi koji su osnovali grupu su Ericsson, Nokia, Intel, IBM i Toshiba. Od tada su se sve veće telekomunikacijske kompanije uključile u rad Bluetooth SIG grupe, te danas ona ima više od 3000 članova koji koriste i podržavaju ovu tehnologiju u svojim uređajima. Osnovne odlike ove tehnologije su robustnost, jednostavnost, niska potrošnja energije i niska cijena.

Bluetooth tehnologija korisnicima pokušava pružiti istu cijenu, sigurnost i funkcionalnost koje su dostupne putem fizičke povezanosti uz dodatne prednosti koje donosi bežično povezivanje. Osnovni zahtjevi definirani su na slijedeći način:

- tehnologija mora pružati sigurnost poput one koja je dostupna putem kablova (podržavanje autorizacije na podatkovnom i aplikacijskom sloju, podržavanje autentikacije i enkripcije),
- mora biti proizvedena za približno istu cijenu,
- mora biti u mogućnosti povezivati različite uređaje koji su dostupni mobilnim korisnicima,
- mora podržavati prijenos podataka koji je konzistentan s potrebama mobilnih korisnika,
- mora podržavati više istovremenih i privatnih konekcija,
- mora podržavati tipove podataka kojima se koriste mobilni korisnici (glas i podaci),
- mora imati nisku potrošnju energije i kompaktnost zbog upotrebe u malim mobilnim uređajima u koje tehnologija treba biti ugrađena.

2.1. Modeli korištenja

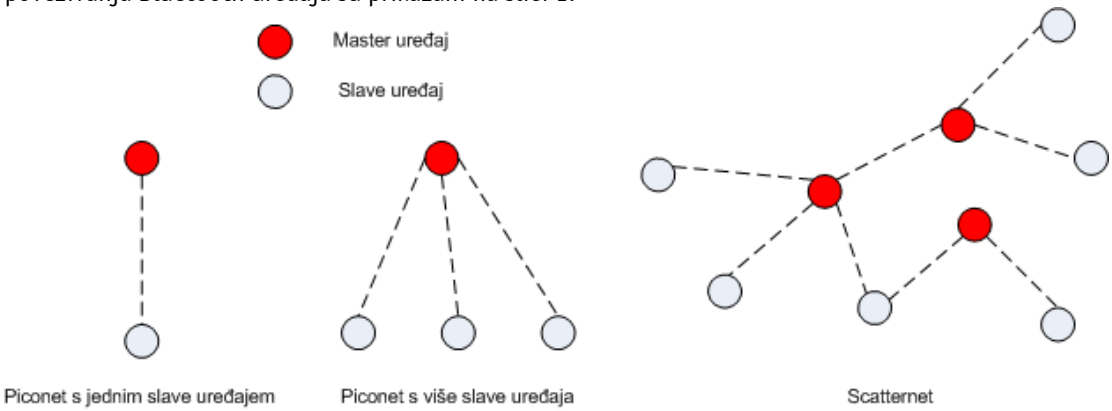
Osnovna namjena Bluetooth tehonologije je povezivanje različitih uređaja putem radio valova. Korištenje ove tehnologije se može podijeliti u tri osnovne kategorije:

- **pristupna točka za glas/podatke** – ovaj model uključuje povezivanje računala s komunikacijskim uređajima putem sigurne bežične veze. Jedan od primjera je spajanje prijenosnog računala s ugrađenom Bluetooth tehnologijom s mobilnim telefonom koji koristi Bluetooth za pristup elektroničkoj pošti. U ovom slučaju mobilni telefon predstavlja osobnu pristupnu točku.
- **spajanje perifernih uređaja** – ova kategorija obuhvaća međusobno povezivanje raznih perifernih uređaja. Ovdje su uključene tipkovnice, miševi ili igraće palice koje rade preko bežične veze. Ovakvi uređaji mogu biti korišteni na više načina. Npr. Bluetooth slušalice mogu biti korištene za spajanje s Bluetooth pristupnom točkom radi pristupa uredskom telefonu i multimedijalnim funkcijama prijenosnog računala. Te iste slušalice mogu služiti i kao sučelje za mobilnim telefonom.
- **Personal Area Networking** – ovaj model se fokusira na *ad-hoc* računalne mreže koje nastaju korištenjem Bluetooth veza.

2.2. Načini povezivanja

Bluetooth sustav podržava konekcije od točke do točke (*engl. point-to-point*), gdje se spajaju samo dva Bluetooth uređaja, ili konekcije koje spajaju jednu točku s više točaka (*point-to-multipoint*). Dva ili više uređaja koji dijele zajednički kanal stvaraju *ad-hoc* mrežu zvanu *piconet*. U *piconetu* jedan Blutooth uređaj ima ulogu *master* uređaja, dok preostali uređaji, pa makar i to bio još samo jedan, preuzimaju ulogu *slave* uređaja. *Master* uređaj je taj koji određuje postavke vezane uz frekvenciju i kanal po kojima se sinkroniziraju svi ostali uređaji u *piconetu*. U jednom *piconetu* moguće je aktivno spojiti do 8 uređaja. Od tih osam uređaja, samo jedan je *master*, dok je preostalih sedam *slave* uređaja. Svaki uređaj u *piconetu* posjeduje MAC adresu od tri bita. Moguće je postojanje i više *slave* uređaja, ali oni ne sudjeluju aktivno u komunikaciji, te se nalaze u tzv. parkirnom stanju (*engl. Parked State*). Oni su sinkronizirani s *master* uređajem, ali ne posjeduju MAC adresu za taj *piconet*, te iz tog razloga ne mogu sudjelovati u razmjeni podataka. Za aktivne i parkirane *slave* uređaje pristup kanalu kontrolira *master* uređaj.

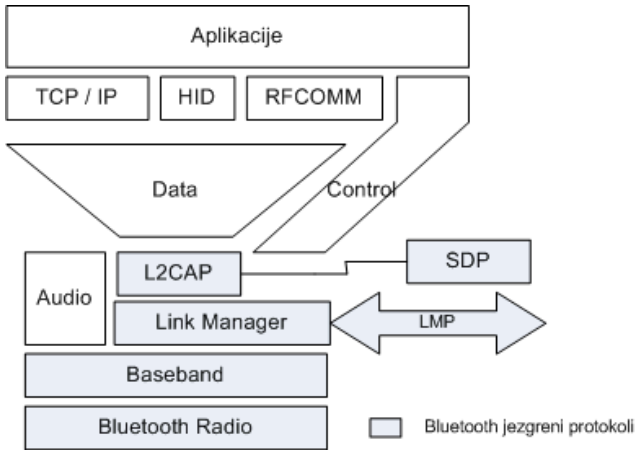
Više piconetova koji djeluju u istom području pokrivanja signala tvori *scatternet*. Svaki *piconet* i dalje zadržava svoju samostalnost, te posjeduje samo jedan *master* uređaj i zasebni kanal. *Piconeti* međusobno povezani u *scatternet* nisu niti vremenski, niti frekvencijski sinkornizirani. *Slave* uređaji mogu sudjelovati u različitim *piconetovima* pomoću multipleksiranja vremenske podjele. *Master* uređaj jednog piconeta može funkcionirati kao *slave* uređaj u nekom drugom *piconetu*. Mogući načini povezivanja Bluetooth uređaja su prikazani na slici 1.



Slika 1: Načini povezivanja Bluetooth uređaja

2.3. Bluetooth arhitektura

Bluetooth tehnologija se dijeli na dvije specifikacije: specifikacije jezgrenih protokola i specifikacije profila. Specifikacija jezgre opisuje kako tehnologija funkcionira, dok se specifikacija profila orijentira na način izgradnje uređaja koji koriste ovu tehnologiju za komunikaciju. Na slici 2 je prikazana Bluetooth arhitektura s postojećim slojevima. Ovaj dokument se orijentira na jezgrene tehnologije, odnosno na niže slojeve Bluetooth arhitekture.



Slika 2: Bluetooth arhitektura

Jezgreni Bluetooth protokoli:

- *Bluetooth Radio* rješava fizički sloj.
- *Baseband* sloj rješava radijsku vezu dva uređaja.
- LMP (*Link Management Protocol*) uspostavlja i prekida komunikaciju na radijskoj vezi između dva uređaja.
- L2CAP (*Logical Link Control and Adaptation Protocol*) prilagođava protokole viših slojeva.
- SDP (*Service Discovery Protocol*), protokol za otkrivanje usluga omogućuje odabir raspoloživih usluga.

2.3.1. Bluetooth Radio sloj

Bluetooth Radio sloj predstavlja fizički sloj Bluetooth arhitekture, a namjena mu je implementacija sučelja fizičke komunikacije. Bluetooth je radio veza kratkog dometa koja djeluje na besplatnom ISM (*engl. Industrial, Scientific, Medicine*) frekvencijskom području od 2.4 GHz-a (2400MHz-a do 2483.5MHz-a). Radi se o području globalno dostupnom i slobodnom za uporabu bez licence. Točna lokacija i širina pojasa se razlikuju od zemlje do zemlje. Točan raspon frekvencija za pojedine zemlje naveden je u sljedećoj tablici.

Zemlja	Raspon frekvencija	Radio kanali
Europa i SAD	2400 - 2483.5 MHz	f = 2402 + k MHz, k= 0,...,78
Japan	2471 - 2497 MHz	f = 2473 + k MHz, k= 0,...,22
Španjolska	2445 - 2475 MHz	f = 2449 + k MHz, k= 0,...,22
Francuska	2446.5 - 2483.5 MHz	f = 2454 + k MHz, k= 0,...,22

Iz tablice je vildjivo da u Japanu, Španjolskoj i Francuskoj postoje po 23 radio kanala, dok je u ostatku Svijeta dostupno 79 kanala. Bluetooth kanal je dizajniran tako da koristi skakanje po frekvenciji (*engl. Frequency Hopping*), čime se umanjuju smetnje šuma i slabog signala. Kanal je stoga definiran pseudo slučajnim nizom skakanja između 23, odnosno 79 radio kanala. Taj niz je jedinstven za svaki *piconet*, a određuje ga adresa master uređaja. Kanal je podijeljen u vremenske odsječke. Svaki vremenski odsječak je duljine 625 mikrosekundi, te je numeriran prema procesorskom taktu piconet *master* uređaja. Vremenski odsječak odgovara frekvenciji skokova između radio kanala. Uobičajena brzina skakanja iznosi 1600 skokova u sekundi. Svi Bluetooth uređaji koji sudjeluju u *piconetu* su sinkronizirani s kanalom i vremenski i skokovima.

Bluetooth standardom su definirane tri klase snage prijenosa. Svi Bluetooth uređaji podržavaju samo jednu od ovih opcija, a najveći broj uređaja koristi snagu za kratki domet. U slijedećoj tablici su navedene sve klase snage prijenosa.

Klasa snage	Izlazna snaga	Domet signala
Klasa 1	100 mW (20 dBm)	~100 m
Klasa 2	2.5 mW (4 dBm)	~20 m
Klasa 3	1 mW (0 dBm)	~10 m

2.3.2. *Baseband* sloj

Baseband sloj služi za regulaciju vremena i redoslijeda prijenosa fizičkih bitova bežičnom vezom od jednog Bluetooth uređaja do drugog. Komunikacija se može obavljati asinkrono ili sinkrono. Na jednom kanalu mogu biti podržana do tri sinkrona (glasovna) podatkovna kanala, ili jedan sinkroni i jedan asinkroni podatkovni kanal. Svaki sinkroni kanal podržava prijenos brzine 64 Kb/s, što je sasvim dovoljno za prijenos glasa. Asinkroni kanal može prenositi do 823.2 Kb/s u jednom smjeru i 57.6 Kb/s u drugom smjeru. Asinkrona konekcija može podržavati i prijenos od 432.6 Kb/s u oba smjera ukoliko je veza simetrična.

Bluetooth sustav se sastoji od radio jedinice (*engl. Radio Unit*), kontrolne jedinice za vezu (*engl. Link Controller*) , te podrške za upravljanje vezom (*engl. Link Manager*). Opisani sustav prikazan je na slici 3.

Slika 3: Funkcijski blokovi Bluetooth sustava

Radio jedinica i kontrolna jedinica su ostvareni sklopovski kao radio čip i kontroler. Upravljanje vezom realizirano je programski i zaduženo je za ostvarivanje i podešavanje veze, autentikaciju te konfiguraciju ostalih protokola. Radio jedinica je zadužena za slanje i primanje radio valova, dok *Link Controller* i *Link Manager* zajedno obavljaju slijedeće zadatke:

- slanje i primanje podataka,
- ostvarivanje veza,
- autentikacija,
- pregovaranje i ostvarenje tipa veze,

- određivanje tipa okvira svakog paketa,
- prebacivanje uređaja iz jednog načina rada u drugi.

Kako je već spomenuto, paketi se šalju u vremenskim odsječcima. U komunikaciji u kojoj *master* i *slave* uređaji naizmjence odašilju pakete, koristi se pristup u kojem *master* uređaj šalje pakete samo na parne vremenske odsječke, dok *slave* uređaj odašilje na neparne vremenske odsječke. Paket počinje s početkom odsječka, a njegova duljina može biti do pet vremenskih odjsečaka. Za vrijeme trajanja paketa frekvencija skakanja ostaje konstantna. Frekvencija skakanja se određuje na temelju trenutnog stanja procesorskog takta. Ukoliko se radi o paketu koji se proteže kroz više vremenskih odsječaka, frekvencija se određuje iz procesorskog takta u trenutku prvog odsječka paketa. Nakon što završi slanje takvog paketa, frekvencija skakanja za slijedeći paket se opet određuje iz trenutnog stanja procesorskog takta.

Između *master* i *slave* uređaja moguće je uspostaviti dva tipa veze:

- sinkrona veza (SCO)
- asinkrona veza (ACL)

SCO veza je simetrična *point-to-point* veza između *master* i *slave* uređaja u *piconetu*. SCO veza se održava korištenjem rezerviranih vremenskih odsječaka u točno definiranim intervalima. *Master* uređaj podržava postojanje tri SCO veze odjednom, bilo sa jednim *slave* uređajem, ili sa tri različita *slave* uređaja. *Slave* uređaj podržava tri SCO veze s istim *master* uređajem, ili po jednu SCO vezu s dva različita *master* uređaja. U komunikaciji se SCO paketi izmjenjuju samo u rezerviranim odsječcima, s tim da se odsječci dijele na *master-to-slave* i *slave-to-master* odsječke. Iz samog naziva jasno je o kakvoj se komunikaciji radi i za koji uređaj je odsječak rezeviran. U *master-to-slave* intrervalima *master* uređaj šalje SCO paket *slave* uređaju. *Slave* uređaju je dozvoljen odgovor SCO paketom u slijedećem *slave-to-master* odsječku, osim ako nije adresiran još jedan *slave* uređaj u posljednjem *master-to-slave* odsječku.

ACL komunikacija se odvija u svim ostalim odsječcima koji nisu rezervirani za SCO vezu. ACL veza omogućuje komunikaciju *master* uređaja sa svim ostalim aktivnim *slave* uređajima u piconetu. Između *master* i *slave* uređaja može postojati samo jedna ACL veza. Za većinu ACL paketa koristi se ponovno slanje kako bi se osigurao integritet podataka. ACL paketi koji nisu namijenjeni određenom *slave* uređaju smatraju se *broadcast* paketima, te ih svi *slave* uređaji čitaju.

2.3.3. Format Bluetooth paketa

Svi podaci u Bluetooth komunikaciji se izmjenjuju kroz pakete. Format osnovnog paketa je prikazan na slici 4.

Slika 4: Standardni format Bluetooth paketa

Paketi se sastoje od tri dijela:

- pristupni kôd – polje duljine 72 bita namijenjeno sinkronizaciji,
- zaglavlje – polje duljine 54 bita koje sadrži informacije namijenjene za kontrolu veze,
- sadržaj paketa – može biti duljine od 0 do 2745 bita.

3. Sigurnost Bluetooth komunikacije

Povjerljivost podataka i zaštita korisnika mora se ostvariti sigurnosnim kontrolama i na fizičkom i na aplikacijskom sloju. Mjere koje sustav poduzima za zaštitu na aplikacijskom sloju se razlikuju od aplikacije do aplikacije, te ovise o njenoj namjeni. Autentikacija i enkripcija koja se provodi na fizičkom sloju ostvarena je na isti način za sve Bluetooth uređaje.

Pružanje sigurnosti na fizičkom sloju se obavlja kroz korištenje četiri entiteta:

- javna adresa uređaja, BD_ADDR – 48 bita
- tajni ključ za autentikaciju – 128 bita
- tajni ključ enkripcije – 8 do 128 bita
- slučajno generirana vrijednost, RAND – 128

Adresa Bluetooth uređaja je jedinstvena 48-bitna IEEE adresa. Ove adrese su javne, te se mogu doznati kroz MMI (*engl. Man Machine Interface*) komunikaciju, ili automatski korištenjem upitne rutine Bluetooth uređaja.
Tajni ključevi se generiraju tijekom inicijalizacije. Ključ enkripcije se dijelom generira iz autentikacijskog ključa za vrijeme procesa autentikacije. Algoritam autentikacije uvijek koristi 128-bitni ključ, dok se za enkripciju koristi ključ čija duljina može biti od 1 do 16 okteta. Duljina ključa enkripcije je promijenjiva iz dva razloga. Prvi su različiti zakoni kojima podliježu kriptografski algoritmi u pojedinim zemljama. Drugi razlog je da bi se omogućile buduće nadogradnje sigurnosnih mjera bez potrebe za redizajniranjem algoritama i sklopovskih rješenja. Povećanje efektivne duljine ključa je najednostavniji način borbe protiv sve veće procesorske snage.
Autentikacijski ključ ostaje isti sve dok aplikacija koja se izvršava na Blutooth uređaju ne zatraži promjenu ključa. Za razliku od njega, kriptografski ključ se ponovno generira pri svakom ponovnom aktiviranju enkripcije.
RAND vrijednost se generira generatorima slučajnih brojeva ugrađenima u Bluetooth uređaje. Ova vrijednost nije stalan parametar, već se često mijenja.

3.1. Generatori slučajnih brojeva

Svaki Bluetooth uređaj posjeduje generator slučajnih brojeva. Slučajni brojevi se često koriste za potrebe provođenja sigurnosnih rutina, kao što su *challenge-response* procedure ili generiranje tajnih ključeva. Kada se govori o slučajnim brojevima, vrlo često se zapravo radi o pseudo-slučajnim brojevima. Jedno od svojstava koje opisuje računalo, a to se može proširiti općenito i na elekroničke uređaje, je determinizam. Računalo može biti samo u konačnom broju stanja. Taj broj je uistinu velik, ali je još uvijek konačan. Za svaki ulaz, izlaz koje daje računalo je deterministička funkcija ulaza i trenutnog stanja računala. Svaki generator slučajnih brojeva koji se odvija na elektroničkom uređaju, a samim tim i na Bluetooth uređaju, ne proizvodi uistinu slučajan niz. Zahtjevi koji se postavljaju na generatore slučajnih brojeva u Bluetooth uređajima su da se generirana vrijednost ne ponovi za vrijeme trajanja autentikacijskog ključa, te da se sljedeći generirani broj ne može predvidjeti.
Generator slučajnih brojeva se u Bluetooth procedurama koristi u više navrata, ti slučajevi biti će opisani u nastvaku dokumenta.

3.2. Upravljanje ključevima

Veličina ključa enkripcije mora biti tvornički prisutna vrijednost, te varira od uređaja do uređaja. Bluetooth ne prihvaća ključ enkripcije s viših slojeva, kako bi se izbjegli pokušaji korisnika da se promijeni dozvoljena veličina ključa.
Postoji više tipova autentikacijskog ključa, a koji će se od njih koristiti u komunikaciji ovisi o tipu aplikacije. Procedure za promjenu ovog ključa su također definirane, korištena procedura ovisi o korištenom tipu. Detalje je moguće pronaći u poglavlju 3.2.2.6. Autentikacijski ključ često se naziva i ključ veze (*engl. Link Key*).

3.2.1. Tipovi ključa veze

Ključ veze je 128-bitna slučajna vrijednost koja se dijeli između dva ili više entiteta koji sudjeluju u komunikaciji, te je temelj svih sigurnosnih transakcija između ovih entiteta.
Ključ veze može biti polutrajan ili privremen. Polutrajni ključ se pohranjuje u *non-volatile* memoriju, kako bi bio dostupan i nakon ponovnog pokretanja uređaja. Polutrajni ključ se može ponovno koristiti i nakon što trenutna sjednica završi. Pod pojmom sjednice podrazumijeva se vrijeme koje je pojedini uređaj proveo kao sastavni dio nekog *piconeta*. Nakon što se polutrajni ključ definira, moguće ga je koristiti za sve slijedeće sjednice između Bluetooth uređaja koji ga dijele. Privremeni ključ je kraćeg trajanja, te je ograničen samo na trenutnu sjednicu u kojoj je generiran.
Definirana su četiri tipa ključa veze kako bi se zadovoljile potrebe različitih vrsta aplikacija:

- kombinacijski ključ K_{AB} – *combination key*
- jedinični ključ K_A – *unit key*
- privremeni glavni ključ K_{master} – *temporary key*
- inicijalizacijski ključ K_{init} – *initialization key*

Za Bluetooth uređaj, K_{AB} i K_A ključevi funkcijski nisu različiti, razlika je tek u načinu njihovog generiranja. Jedinični ključ K_A se zasebno generira na uređaju A, te je samo on i ovisan o njemu. K_A se generira pri instalaciji Bluetooth uređaja, te se vrlo rijetko mijenja.
Kombinacijski ključ K_{AB} se izračunava iz informacija dobivenih iz dva uređaja, A i B, te je stoga uvijek zavisan o njima. Kombinacijski ključ se generira za svaku novu kombinaciju dva Bluetooth uređaja.
Koji ključ će biti korišten ovisi o aplikaciji ili uređaju. Jedinični ključ će se koristiti u slučajevima kada Bluetooth uređaj posjeduju malo memorije za pohranu ključa, ili kada se radi o ključu koji je instaliran na uređaju koji mora biti dostupan velikom broju korisnika. Kombinacijski ključ se koristi u aplikacijama koje zahtjevaju višu razinu sigurnosti.
Privremeni glavni ključ K_{master} se koristi samo tijekom trenutne sjednice. On zanijenjuje originalni ključ veze samo privremeno, a koristi se u situacijama kada *master* uređaj želi istovremeno poslati podatke više od dva *slave* uređaja korištenjem istog ključa enkripcije.
Inicijalizacijski ključ se koristi kao ključ veze tijekom inicijalizacije, dok kombinacijski ili jedinični ključ nisu definirani i izmjenjeni, ili u slučajevima kada dođe do gubitka ključa veze. Inicijalizacijski ključ štiti izmjenu inicijalizacijskih parametara. Ovaj ključ se generira iz adrese uređaja BD_ADDR, PIN kôda i slučajno generiranog broja. PIN kôd može biti konstantan broj koji dolazi uz Bluetooth uređaj, ili može zasebno biti odabran od strane korisnika, te potom unesen u oba uređaja koja se žele povezati. Za drugu proceduru potrebno je postojanje sučelja na uređajima preko kojeg je moguć korisnički unos, kao što su telefoni ili prijenosna računala. Unos PIN kôda je sigurniji nego korištenje fiksne PIN vrijednosti, te bi se uvijek trebao koristiti kad god je to moguće.
Vrlo često PIN je kratak niz brojeva, te se tipično sastoji od 4 decimalne znamenke. To može biti dovoljno za zadovoljavajući stupanj sigurnosti, iako postoje mnoge, osjetljivije sitaucije u kojima ovako kratak PIN neće biti dovoljan. Iz tog razloga podržana duljina PIN kôda se kreću u rasponu od 1 do 16 okteta. Za sigurnosno osjetljivije transakcije ne preporučuje se ručna razmjena ključeva, kao što je korisnička interakcija, već razmjena putem nekog programskog rješenja na aplikacijskoj razini, kao što je npr. Diffie-Hellman procedura.

3.2.2. Generiranje ključa i inicijalizacija

Ključevi veze moraju biti generirani i razdijeljeni između Bluetooth uređaja kako bi se mogli koristiti u autentikacijskoj proceduri. Ključevi veze moraju biti tajni, tako da ne mogu biti razmijenjeni kroz jednostavne rutine upita, kao što je to slučaj s Bluetooth adresama. Izmjena ključeva se odvija u fazi inicijalizacije, koja se mora provesti odvojeno za svake dvije jedince koje žele koristiti autentikaciju i kriptiranje. Inicijalizacijska procedra se sastoji od slijedećih koraka:

- generiranje inicijalizacijskog ključa,
- autentikacija,
- generiranje ključa veze,
- izmjena ključa veze,
- generiranje ključa enkripcije u svakoj jedinici.

Nakon inicijalizacijske procedure započinje komunikacija ili se veza prekida.
Za svaku novu konekciju između uređaja A i B za autentikaciju se koristi već uspostavljeni ključ veze. Za svako ponovno aktiviranje enkripcije iz aktualnog ključa veze kreira se novi ključ za enkripciju. Ukoliko ključ veze ne postoji za određeni par uređaja, *Link Manager* automatski započinje inicijalizacijsku proceduru.

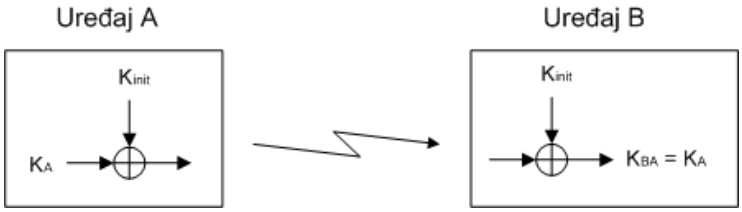
3.2.2.1. Generiranje inicijalizacijskog ključa

Inicijalizacijski ključ K_{init} generira se pomoću algoritma E_{22}. Algoritam koji služi za generiranje ključa je u stvari E_2 algoritam, ali kako posjeduje dva načina rada, uobičajeno je da se dijeli na E_{21} i E_{22} algoritme. Način rada E_2 algoritma ukratko je opisan u poglavlju 3.5.2.
Inicijalizacijski ključ se generira iz adrese uređaja koja zahtjeva uspostavu veze (*engl. Claimant*), PIN kôda, duljine PIN kôda (u oktetima), i slučajno generiranog broja od strane uređaja koji potvrđuje vezu (*engl. Verifier*). 128-bitni izlaz će biti korišten za izmjenu ključa tijekom generiranja ključa veze. Inicijalizacijski ključ se također koristi za autentikaciju dva uređaja koji nemaju već potvrđeni ključ veze. Nakon što se obavi izmjena ključa veze, incijalizacijski ključ se odbacuje.

Pri generiranju inicijalizacijskog ključa, PIN se proširuje BD_ADDR adresom *claimant* uređaja. Maksimalna duljina PIN kôda u algoritmu ne može biti veća od 16 okteta, te je stoga moguće da se ne iskoriste svi okteti BD_ADDR vrijednosti. Ova procedura osigurava da K_{init} ovisi o identitetu uređaja koji se pokušava povezati.

3.2.2.2. Generiranje jediničnog ključa

Jedinični ključ se generira kada se Bluetooth uređaj pokrene po prvi put. Jedinični ključ se generira putem E_{21} algoritma. I ovdje se radi o E_2 algoritmu, ali u prvom načinu rada.
Kada se jedinični ključ jednom generira, pohranjuje se u *non-volatile* memoriju, te se gotovo nikad ne mijenja. Tijekom inicijalizacije aplikacija odlučuje čiji će se jedinični ključ iskoristiti kao ključ veze. Najčešće je to ključ uređaja koji ima ograničenu memoriju, te je dovoljna da pamti samo svoj jedinični ključ. Jedinični ključ uređaja se prenosi drugom sudioniku, koji pohranjuje taj ključ kao ključ veze za komunikaciju s tim Bluetooth uređajem. Primjer u kojem ključ uređaja A služi kao ključ veze je prikazan na slici 5. K_A se prenosi uređaju B, koji pohranjuje taj ključ kao ključ veze K_{BA}. K_{BA} će i ubuduće uvijek biti korišten za komunikaciju uređaja A i B.



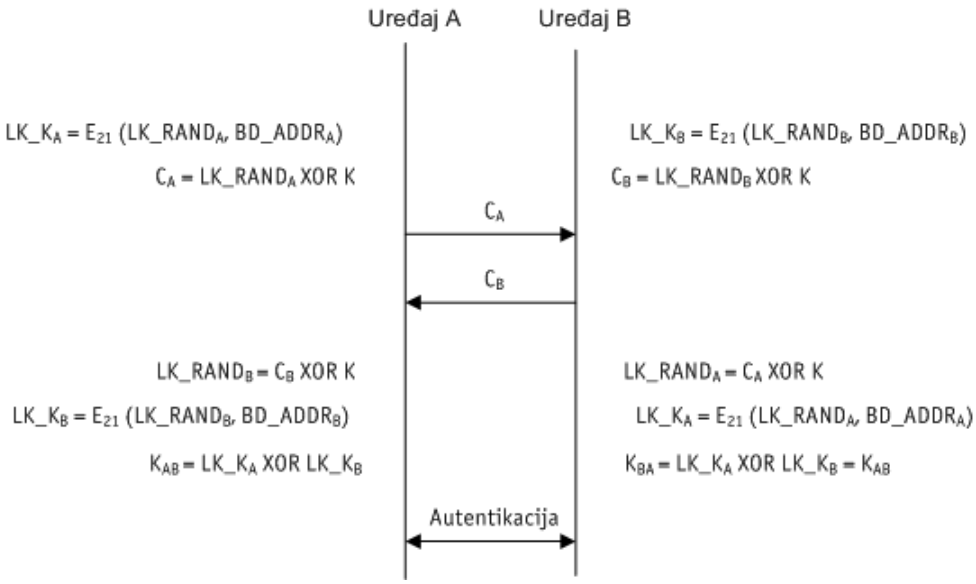
Slika 5: Generiranje jediničnog ključa

3.2.2.3. Generiranje kombinacijskog ključa

Kombinacijski ključ je kombinacija dva broja generirana u uređajima A i B. Oba uređaja prvo generiraju slučajne brojeve, LK_RAND_A i LK_RAND_B. Algoritmom E_{21} se iz generiranih slučajnih brojeva i adrese uređaja generiraju dva nova slučajna broja:

$$LK\_K_A = E_{21}(LK\_RAND_A, BD\_ADDR_A)$$
$$LK\_K_B = E_{21}(LK\_RAND_B, BD\_ADDR_B)$$

LK_K_A je doprinos ključu veze uređaja A, dok je LK_K_B doprinos uređaja B. Slučajni brojevi LK_RAND_A i LK_RAND_B se izmjene, ali uz prethodno izvršavanje XOR operacije s trenutnim ključem veze K. Ukoliko se ova procedura obavlja za vrijeme inicijalizacije, trenutni ključ veze je inicijalizacijski ključ. Uređaj A šalje $K \oplus LK\_RAND_A$ uređaju B, dok uređaj B šalje $K \oplus LK\_RAND_B$. Pri primitku ovih poruka, svaka strana izračuna slučajne brojeve, te dodatno izračuna doprinos onog drugog kombinacijskom ključu. Ovo je moguće jer obje strane znaju adresu Bluetooth uređaja s kojim komuniciraju.
Nakon što su svi podaci razmjenjeni oba uređaja iz LK_K_A i LK_K_B generiraju 128-bitni ključ veze. Operacija kojom se kombiniraju ove dvije vrijednosti je jednostavno zbrajanje po modulu 2. Nakon uspješne razmjene novog kombinacijskog ključa, stari ključ veze se odbacuje. Primjer upravo opisane procedure prikazan je na slici 6.



Slika 6: Generiranje kombinacijskog ključa

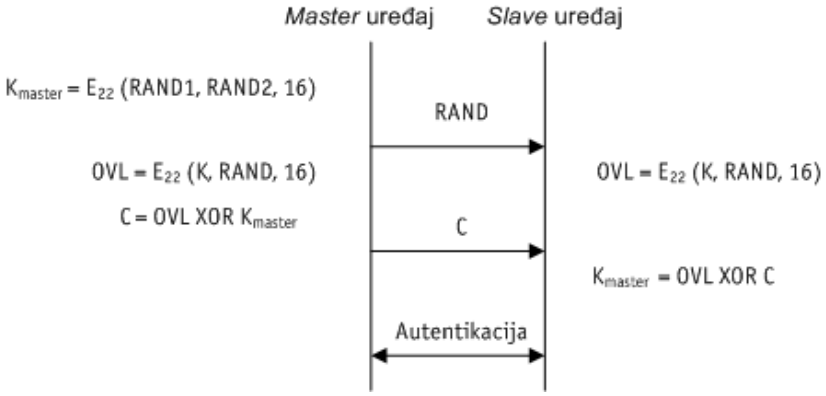
3.2.2.4. Generiranje privremenog glavnog ključa

U konfiguracijama *piconeta* u kojima je na *master* uređaj priključeno više *slave* uređaja, za svaki od njih se koristi drugačiji ključ enkripcije. Ovome su razlog razlog različiti ključevi veze. U slučajevima kada aplikacija mora poslati iste podatke na više *slave* uređaja, svaki *slave* uređaj se mora obraditi individualno. Ovo može uzrokovati neželjeni gubitak kapaciteta *piconeta*. Bluetooth jedinica nije u mogućnosti promijeniti ključ enkripcije u stvarnom vremenu. Iz tog razloga *master* uređaj ne može koristiti jedan ključ enkripcije *broadcast* poruka, a drugi za poruke namijenjene isključivo pojedinom *slave* uređaju. Za ovakve potrebe koristi se alternativni pristup. *Master* uređaj naređuje odabranim *slave* uređajima da počnu koristiti novi ključ veze. Na taj način svi odabrani *slave* uređaji će koristiti i isti ključ enkripcije generiran iz novog dijeljenog ključa veze. Nakon toga *master* uređaj može *broadcast* podatke slati u kriptiranom obliku. Ovakav ključ, koji je samo privremen, se naziva privremeni glavni ključ.

Master uređaj stvara novi ključ veze iz dva 128-bitna slučajna broja pomoću algoritma E_{22}. Razlog korištenja E_{22} algoritma, a ne direktno generiranog slučajnog broja, je zbog vrlo česte pojave slabih implementacija generatora slučajnih brojeva u Bluetooth uređajima. Izlaz ove procedure daje 128 bitni privremeni glavni ključ.

Master uređaj generira i treći slučajni broj RAND, koji šalje *slave* uređaju. Pomoću E_{22} algoritma te RAND vrijednosti i trenutnog ključa veza kao ulaza u algoritam, oba uređaja izračunaju 128-bitni rezultat. *Master* uređaj izvršava XOR operaciju između tako dobivenog rezultata i novog ključa veze, te to šalje *slave* uređaju. *Slave* uređaj jednostavnom XOR operacijom dolazi do novog ključa veze. Također, moguće je izvršiti ponovnu autentikaciju korištenjem novog ključa veze, kako bi se potvrdila uspješnost transakcije. Kad se za tim pokaže potreba, *master* uređaj može sve sudionike obavijestitio povratku na stare ključeve veze.

Slijed poruka i operacija pri generiranju privremenog glavnog ključa je prikazan na slici 7.



Slika 7: Generiranje privremenog glavnog ključa

3.2.2.5. Generiranje ključa enkripcije

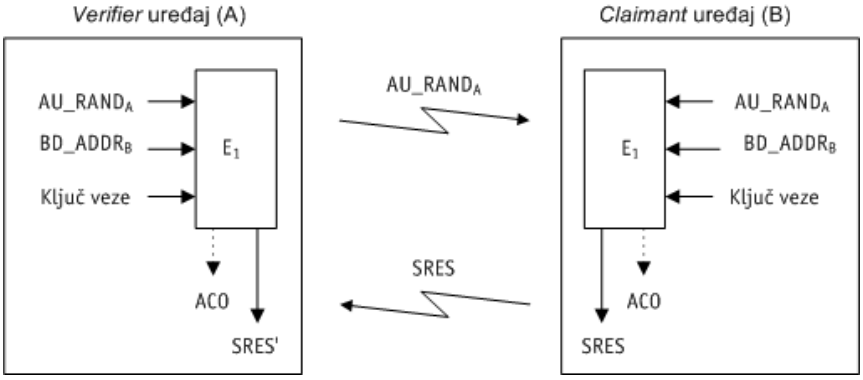
Ključ enkripcije K_C generira se pomoću algoritma E_3. Detalji ovog algoritma su opisani u poglavlju 3.5.3. Kao ulaz u ovaj algoritam koriste se trenutni ključ veze, 128-bitni slučajno generirani broj, te 96-bitna vrijednost COF (*engl. Ciphering OFfset Number*). COF vrijednost se određuje na dva načina. Ako je trenutni ključ veze privremeni glavni ključ, tada se COF izračunava iz BD_ADDR adrese *master* uređaja. U svim ostalim slučajevima COF vrijednost se postavlja na ACO vrijednost dobivenu tijekom autentikacije.
Kada *Link Manager* aktivira kriptiranje vrši se izravni poziv E_3 algoritma. Iz ovog je jasno da se ključ enkripcije mijenja pri svakom ulasku Bluetooth uređaja u kriptografski način rada.

3.2.2.6. Promjena ključa veza

U određenim okolnostima poželjno je promijeniti postojeći ključ veze. Ključ veze koji se bazira na jediničnom ključu se također može promijeniti, ali nešto teže nego što je slučaj s kombinacijskim ključem. Promjena jediničnog ključa je manje poželjna alternativa, jer više uređaja može dijeliti taj isti ključ kao ključ veze. Promjena jediničnog ključa zahtjeva reinicijalizaciju svih uređaja koji su se već povezivali. U nekim slučajevima, kao što je npr. zabrana pristupa jedinicama kojima je pristup već dodijeljen, ovo može biti i poželjan scenarij.
Prilikom promjene kombinacijskog ključa, dovoljno je samo ponovno pokrenuti proceduru generiranja kombinacijskog ključa. U takvom slučaju trenutni kombinacijski ključ koji se želi odbaciti se koristi kao ključ veze. Ova procedura se može provesti u bilo kojem trenutku nakon što započnu autentikacija i enkripcija podataka. Kombinacijski ključ se može mijenjati pri svakoj novoj konekciji. Ovo samo može povećati sigurnost sustava, s obzirom da nakon svake sjednice stari ključevi više nisu valjani.

3.3. Autentikacija

Bluetooth autentikacija se provodi kroz *challenge-response* mehanizam. U njemu se provjerava poznavanje tajnog ključa uređaja koji želi uspostaviti vezu (*engl. Claimant*) s uređajem koji potvrđuje vezu (*engl. Verifier*). Autentikacija će biti uspješna samo u slučaju ako dva uređaja dijele isti tajni ključ K. *Challenge-response* mehanizam se provodi tako da *verifier* uređaj traži od *claimant* uređaja da izvrši algoritam E1 nad slučajno generiranim brojem AU_RAND_A. Rezultat ove operacije je SRES vrijednost koju *claimant* uređaj šalje *verifier* uređaju. Ovaj postupak je prikazan na slici 8.



Slika 8: Challenge-response mehanizam u Bluetooth autentikaciji

Na slici je vidljivo da je ulaz u E_1 algoritam AU_RAND_A vrijednost, adresa *claimant* uređaja BD_ADDR, te zajednički ključ veze K.

Uređaj koji potvrđuje vezu ne mora nužno biti *master* uređaj. Aplikacija određuje tko se autenticira kome. Određene aplikacije zahtjevaju samo jednosmjerne autentikacije, dok se u nekima izvršava obostrana autentikacija. *Link Manager* je zadužen za koordinaciju autentikacijskih postavki koje je proslijedila aplikacija kako bi se odredilo u kojem smjeru se autentikacija provodi. Pri obostranoj autentikaciji oba uređaja generiraju svoje slučajne brojeve te od druge jedinice zahtjevaju generiranje SRES vrijednosti.

Nakon što je autenifikacija uspješno provedena, dio rezultata E_1 algoritma je i ACO vrijednost koja se pohranjuje kako bi se iskoristila u procesu enkripcije podataka.

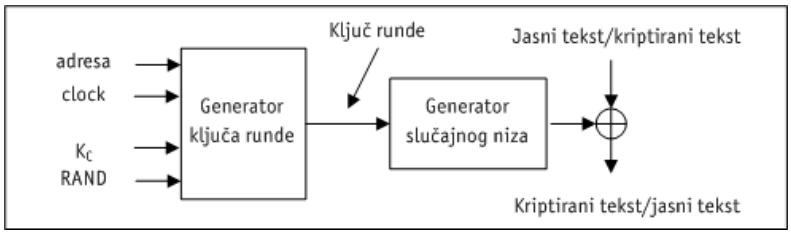
Pri neuspjelim pokušajma autentikacije mora proći određeni vremenski interval prije novog pokušaja. Za svaki ponovni neuspjeh sa iste Bluetooth adrese vrijeme čekanja se povećava eksponencijalno. Vremenski interval čekanja mora posjedovati i svoj maksimum, čiji iznos ovisi o implementaciji. Vremenski interval čekanja se vraća na minimum nakon što neko vrijeme nema neuspjelih pokušaja autentikacije s određene Bluetooth adrese. Ova jednostavna procedura spriječava ponavljanje pokušaja autentikacije s velikim brojem različitih ključeva. Bluetooth uređaji bi morali posjedovati listu individualnih vremenskih intervala svake jedinice koja se pokušala povezati. Veličina ove liste mora biti ograničena na N posljednjih uređaja s kojima je uspostavljen kontakt. Vrijednost N će ovisiti o dostupnoj memoriji i korisničkom okruženju.

3.4. Enkripcija

Bluetooth tehnologija pruža zaštitu korisničkih podataka korištenjem enkripcije. Samo sadržaj paketa se kriptira, dok se pristupni kôd paketa i njegovo zaglavlje nikad ne kriptiraju. Kriptiranje sadržaja paketa obavlja se algoritmom kriptiranja toka podataka E_0. Algoritam kriptiranja toka podataka E_0 se sastoji od 3 dijela. Prvi dio je zadužen za inicijalizaciju, u njemu se generira ključ za sadržaj svakog pojedinog paketa. Drugi dio generira niz slučajnih bitova, dok se u trećem vrši enkripcija, odnosno dekripcija. Enkripcija se vrši kao i u svakom drugom algoritmu kriptiranja toka podataka, XOR operacijom nad ulaznim i generiranim nizom.

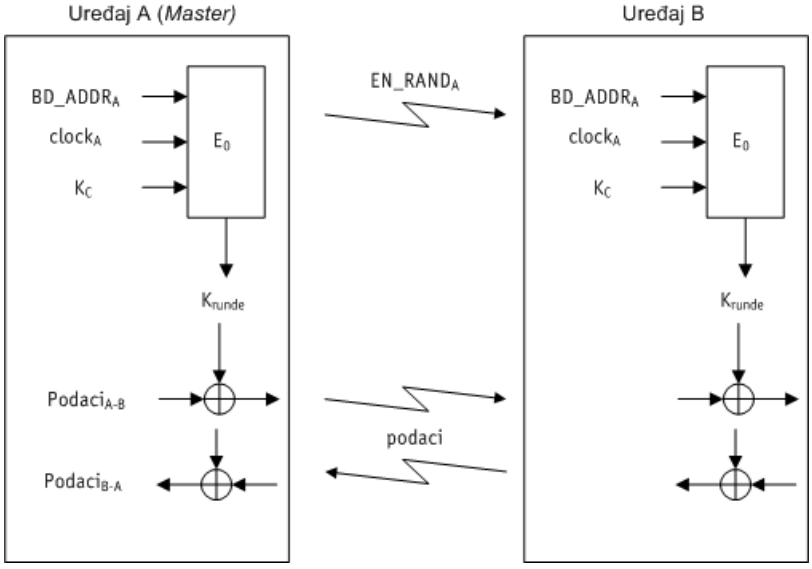
Svaki novi paket predstavlja novu rundu enkripcije, te se ključ koji se generira za svaki novi paket može nazvati i ključ runde. Generator ključa runde je vrlo jednostavan, sastoji se od 4 posmična registra s linearnom povratnom funkcijom. Ovi registri se popunjavaju određenom kombinacijom ulaznih bitova kako bi se generirao ključ. Glavni dio sustava enkripcije je drugi dio. Niz slučajnih bitova se generira metodom koju su osmislili Massey i Rueppel. Ova metoda je detaljno provjerena, te postoje dobre procjene njene snage s obzirom na trenutno poznate metode kriptoanalize.

Princip kriptiranja je prikazan na slici 9.



Slika 9: Princip kriptiranja

Kriptiranje sadržaja paketa se obavlja prije FEC (*engl. Forward Error Correction*) kodiranja, ali nakon što su dodani CRC (*engl. Cyclic Redundancy Check*) bitovi. FEC kodiranje služi za poboljšanje robustnosti pri prijenosu podataka, dok se CRC botovi dodaju radi otkrivanja nastalih pogrešaka. Svaki sadržaj paketa se kriptira zasebno. Algoritam E_0 za ulaz uzima adresu *master* uređaja, 26 bita sata stvarnog vremena *master* uređaja, te ključ enkripcije K_C. K_C se generira algoritmom E_3, za što su mu potrebni trenutni ključ veze, COF vrijednost, te slučajni broj EN_RAND_A kojeg generira *master* uređaj prije početka enkripcije. EN_RAND_A vrijednost se prenosi u čistom tekstualnom obliku te se smatra javno poznatom vrijednošću. Blokovski opis procedure enkripcije je prikazan na slici 10.



Slika 10: Procedura kriptiranja

U E0 algoritmu, ključ enkripcije K_C se transformira u K_{runde}, koji služi kao ključ samo za jedan paket. Za svaki vremenski odsječak, sat se uvećava. Na početku svakog novog paketa E_0 algoritam se reinicijalizira. S obzirom da se koristi i vrijednost sata, barem jedan bit će biti različit za dva uzastopna prijenosa. Iz ovog proizlazi da je niz generiranih slučajnih bitova drugačiji nakon svake reinicijalizacije. Za pakete koji se protežu preko više vremenskih odsječaka, za cijeli paket se koristi vrijednost sata na prvom odsječku.

Procedura kriptiranja je simetrična. Dekriptiranje se obavlja na isti način korištenjem istog ključa.

U ovisnosti koji ključ veze se koristi, *slave* uređaj razlikuje više načina rada kriptiranja. *Slave* uređaj koji koristi jedinični ili kombinacijski ključ (polutrajni ključ) može primati samo pakete namijenjene isključivo njemu. Za *broadcast* poruke slave uređaj će pretpostaviti da se ne koristi enkripcija. Za *slave* uređaj koji posjeduje privremeni glavni ključ postoje tri kombinacije *broadcast* i individualnog prometa. Ove kombinacije su prikazane u tablici.

Broadcast promet	Individualni promet
Bez enkripcije	Bez enkripcije
Bez enkripcije	Enkripcija, K_{master}
Enkripcija, K_{master}	Enkripcija, K_{master}

3.4.1. Pregovaranje o duljini ključa enkripcije

Svaki Bluetooth uređaj posjeduje parametar koji definira maksimalnu duljinu ključa L_{max}, $1 \leq L_{max} \leq 16$ (izražen brojem okteta). Svaka aplikacija definira L_{min}, minimalnu prihvatljivu duljinu ključa enkripcije. Prije samog generiranja ključa enkripcije, mora se provesti dogovor o duljini ključa koja će biti upotrijebljena.

- *Master* uređaj šalje predloženu vrijednost slave $L_{sug}^{(M)}$ *slave* uređaju. Ova vrijednost je inicijalno postavljena na $L_{max}^{(M)}$
- Ako je $L_{min}^{(S)} \leq L_{sug}^{(M)}$, i *slave* uređaj podržava predloženu duljinu, *slave* uređaj potvrđuje dogovorenu vrijednost. Pregovaranje je završeno i duljina ključa je dogovorena.
- Ako oba uvjeta nisu ispunjena *slave* uređaj šalje svoj prijedlog $L_{sug}^{(S)} < L_{sug}^{(M)}$ *master* uređaju. Ova treba biti najdulja od svih podržanih vrijednosti *slave* uređaja
- *Master* uređaj provjera da li podržava novu predloženu vrijednost.
- Procedura se nastavlja dok se ne dogovori duljina ključa ili jedna strana ne prekine proces pregovaranja. Prekid je moguć u slučaju da jedna od strana ne podržava L_{sug} i kraće duljine ključa, ili ako je $L_{sug} < L_{min}$ u jednom od uređaja. U slučaju prekida, enkripcija komunikacije nije moguća.

Mogućnost da se ne uspije uspostaviti sigurna veza je neizbježna s obzirom da se aplikaciji prepušta odluka o prihvaćanju ili odbijanju duljine ključa. Bez ove predostrožnosti, zlonamjerna Bluetooth jedinica bi mogla uspostaviti slabu zaštitu definiranjem jako kratke maksimalne duljine ključa.

3.5. Algoritmi autentikacije i generiranja ključeva

U ovom poglavlju ukratko su opisani algoritmi koji služe kao podrška Bluetooth sigurnosnim zahtjevima autentikacije i generiranja ključa.

3.5.1. Algoritam autentikacije E_1

E_1 algoritam u svom radu koristi algoritam kriptiranja SAFER+. SAFER+ je poboljšana verzija postojećeg 64-bitnog blok algoritma SAFER-SK 128, te je dostupan besplatno. SAFER+ uz 128-bitni ulaz i 128 bitni ključ daje 128-bitni izlaz, te se označava s A_r:

$$A_r: \{0,1\}^{128} \times \{0,1\}^{128} \rightarrow \{0,1\}^{128}$$

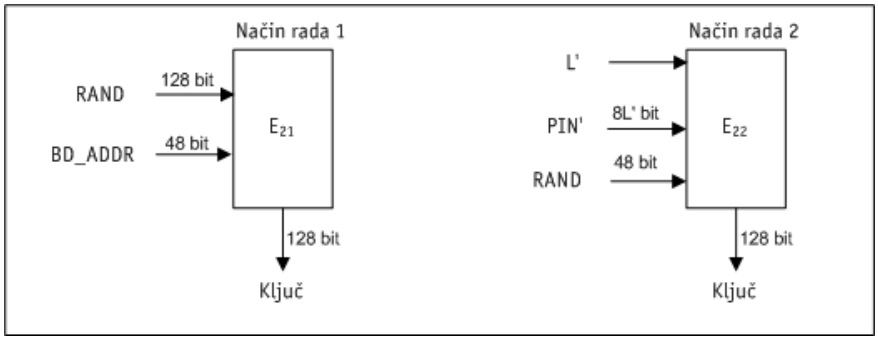
Algoritam E_1 je definiran na slijedeći način:

$$E_1: \{0,1\}^{128} \times \{0,1\}^{128} \times \{0,1\}^{48} \rightarrow \{0,1\}^{32} \times \{0,1\}^{96}$$
$$(K, RAND, BD\_ADDR) \rightarrow (SRES, ACO),$$

gdje je SRES = *Hash* (K, RAND, BD_ADDR, 6)[0...3], odnosno prva četiri okteta rezultata funkcije sažimanja koja u svom radu koristi A_r funkciju, te njezinu modifikaciju A_r'. A_r' funkcija se razlikuje od A_r funkcije samo po tome što se ulaz prve runde zbraja s ulazom treće runde. Detalji SAFER+ algoritma izlaze van predviđenog opsega ovog dokumenta, te ovdje neće biti analizirani.

3.5.2. Algoritam generiranja ključeva za autentikaciju E_2

E_2 algoritam služi za generiranje ključa koji je potreban za provođenje postupka autentikacije, odnosno za generiranje ključa veze. Procedura generiranja ovog ključa je blokovski prikazana na slici 11.



Slika 11: Procedura generiranja ključa autentikacije

Algoritam E_2 ima dva načina rada. U prvom načinu rada E_2 generira 128-bitni ključ veze iz 128-bitnog slučajnog broja i 48-bitne adrese. Ovaj način rada se koristi za generiranje jediničnih i kombinacijskih ključeva. U drugom načinu rada algoritam generira 128-bitni ključ veze iz 128-bitnog slučajnog broja i korisničkog PIN kôda duljine L okteta. Drugi način rada služi za generiranje inicijalizacijskih ključeva, te kad god je potreban privremeni glavni ključ.
Pri generiranju inicijalizacijskog ključa, PIN se proširuje adresom BD_ADDR uređaja koji želi uspostaviti vezu. Proširivanje uvijek počinje s najmanje značajnim oktetom adrese, koji slijedi odmah iz najznačajnijeg okteta PIN kôda. Duljina PIN kôda u algoritmu ne može biti veća od 16 okteta, tako da je moguće da se ne iskoriste svi okteti BD_ADDR vrijednosti.
Algoritam generiranja ključeva također koristi ranije spomenutu kriptografsku funkciju A_r'. E_2 u prvom načinu rada se označava kao E_{21}:

$$E21: \{0,1\}^{128} \times \{0,1\}^{48} \rightarrow \{0,1\}^{128}$$
$$(RAND, BD\_ADDR) \rightarrow A_r'(X, Y),$$
$$X = RAND[0...14] \cup (RAND[15] \oplus 6)$$
$$Y = \bigcup_{i=0}^{15} BD\_ADDR[i \bmod 6]$$

U drugom načinu rada koristi se proširenje korisničkog PIN kôda. Njegova duljina u oktetima označava se s L. Proširivanje se definira:

$$PIN' = PIN[0...L-1] \cup BD\_ADDR[0...\min\{5,15-L\}], \text{ za } L<16$$
$$PIN' = PIN[0...L-1], \text{ za } L=16$$

Tada je E_2 algoritam u drugom načinu rada definiran kao:

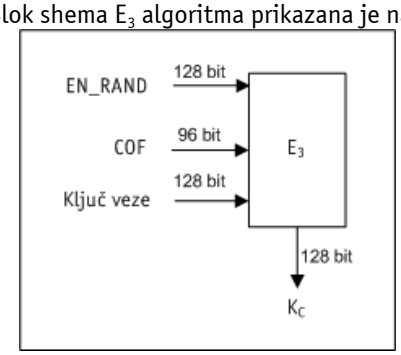
$$E_2: \{0,1\}^{128} \times \{0,1\}^{128} \times \{1,2,...,16\} \rightarrow \{0,1\}^{128}$$
$$(PIN', RAND, L') \rightarrow Ar'(X, Y),$$
$$X = \bigcup_{i=0}^{15} PIN'[i \bmod L']$$
$$Y = RAND[0...14] \cup (RAND[15] \text{ XOR } L'),$$
L' duljina u oktetima PIN' vrijednosti

3.5.3. Algoritam generiranja ključa enkripcije E_3

Ključ enkripcije se generira pomoću algoritma E_3. Algoritma E_3 kao građevni element koristi funkciju A_r':

$$E_3: \{0,1\}^{128} \times \{0,1\}^{128} \times \{0,1\}^{96} \rightarrow \{0,1\}^{128}$$
$$(K, RAND, COF) \rightarrow Hash(K, RAND, COF, 12)$$
$$COF = BD\_ADDR_A \cup BD\_ADDR_B, \text{ za privremeni glavni ključ}$$
$$COF = ACO, \text{ inače}$$

Hash funkcija je ista ona funkcija ranije spomenuta u poglavlju 3.5.1. Dobiveni ključ je duljine 128 bita. Prije upotrebe ključa enkripcije izvršite će se njegovo skraćivanje na dogovorenu duljinu kako je već opisano u poglavlju 3.4.1. Blok shema E_3 algoritma prikazana je na slici 12.



Slika 12: Procedura generiranja ključa enkripcije

3.6. Sigurnosni problemi

Algoritam kriptiranja toka podataka E_0 je u nekim slučajevima ranjiv na *divide-and-conquer* napade. Ključ enkripcije može biti otkriven samo kada je generirani niz slučajnih brojeva duži od periode najkraćeg posmičnog registra s povratnom funkcijom. Perioda posmičnog registra je duljina izlaznog niza prije nego što se niz počne ponavljati. Ovaj propust je ipak vezan samo uz kriptografski algoritam E_0, jer se u Bluetooth specifikaciji definirao način zaobilaska ovog problema. Spomenuta *divide-and-conquer* metoda zahtjeva pristup generiranom nizu slučajnih brojeva uz različite ulazne podatke. Ovo se u Bluetooth sustavu ne može dogoditi zbog vrlo visoke frekvencije reinicijalizacije. Kako se za svaki paket generira drugačiji ključ runde, svaki generirani niz slučajnih brojeva je nezavisan i toliko kratak da se ne zadovoljavaju uvjeti potrebni za provođenje napada.

Snaga inicijalizacijskog ključa ovisi isključivo o PIN kôdu. Algoritam generiranja inicijalizacijskog ključa E_{22} generira ključ iz PIN kôda, njegove duljine i slučajnog broja koji se u jasnom obliku prenosi radio valovima. Jedini tajni entitet je PIN kôd. Kako se većina PIN kôdova sastoji od svega 4 znamenke, dovodi se u pitanje vjerodostojnost inicijalizacijskog ključa.

Korištenje jediničnog ključa ne osigurava dovoljnu tajnost podataka. Uređaj koji koristi svoj jedinični ključ kao ključ veze, podložan je prisluškivanju od strane svih uređaja s kojima je već ranije imao uspostavljenu vezu, te stoga s njim dijele ključ veze. Taj ključ veze im omogućuje generiranje ključa enkripcije, pomoću kojeg mogu prisluškivati kriptiranu komunikaciju uređaja A koja nije namijenjena njima. Ovo je moguće jer su sve informacije, osim ključa veze, za autentikaciju i enkripciju javne. Kako se ključ veze dijeli s više uređaja, oni su uz lažiranje svoje adrese u mogućnosti prisluškivati promet uređaja koji koristi svoj jedinični ključ kao ključ veze.

Adrese Bluetooth uređaja, koje su jedinstvene za svaki uređaj, također mogu predstavljati potencijalni problem. Nakon što se ustanovi da adresa određenog uređaja pripada određenom korisniku, moguće je pratiti sve njegove transakcije, čime se narušava privatnost korisnika.

Čini se da je sigurnost Bluetooth sustava dovoljna za manje aplikacije, ali prijenos osjetljivih informacija putem Bluetooth veze bi ipak trebalo izbjegavati.

4. Zaključak

Bluetooth je radio sustav namijenjen povezivanju mobilnih uređaja putem sigurnih *ad-hoc* veza. U definiciju samog standarda uloženo je mnogo truda kako bi se razvio sustav koji omogućuje komunikaciju različitih tipova uređaja uz ispunjavanje svih zahtjeva mobilnih korisnika. Bluetooth sustav definira načine autentikacije i kriptiranja na fizičkom sloju. Ovakav pristup pruža zadovoljavajuću razinu sigurnosti za malene *ad-hoc* mreže, međutim, Bluetooth sigurnost je još uvijek nedostatna za ozbiljnije i osjetljivije aplikacije, kao što su novčane transakcije i prijenos drugih osjetljivih podataka. Važno je napomenuti da se prilikom dizajniranja Bluetooth specifikacije nije niti predviđala njegova namjena za kritične aplikacije sa stanovišta sigurnosti. Sigurnost se može unaprijediti korištenjem dodatne zaštite i na aplikacijskom sloju, ali ovi zahtjevi nisu propisani Bluetooth specifikacijom.

Bluetooth je standard koji je vrlo raširen i dobro prihvaćen. Broj korisnika i proizvođača koji ga podržava i koristi je sve veći. Osim dobre podrške i pokrivenosti, definirane su i zadovoljavajuće sigurnosne procedure, tako da se može zaključiti da je Bluetooth dobar izbor za bežično povezivanje mobilnih uređaja na kratkim udaljenostima.

5. Reference

[1] The Official Bluetooth Website, http://www.bluetooth.com/
[2] Bluetooth Security, http://www.bluetooth.com/upload/24Security_Paper.PDF
[3] Bluetooth Security, http://www.niksula.cs.hut.fi/~jiitv/bluesec.html