Multicast sigurnost i zaštita autorskih prava

Vuković, Tihana

Master's thesis / Diplomski rad

2023

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, School of Applied Mathematics and Informatics / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet primijenjene matematike i informatike**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:126:907118

Sveučilište J. J. Strossmayera u Osijeku
Fakultet primijenjene matematike i informatike
Sveučilišni diplomski studij matematike
Smjer: Financijska matematika i statistika

Tihana Vuković

Multicast sigurnost i zaštita autorskih prava

Diplomski rad

Osijek, 2023.

Sveučilište J. J. Strossmayera u Osijeku
Fakultet primijenjene matematike i informatike
Sveučilišni diplomski studij matematike
Smjer: Financijska matematika i statistika

Tihana Vuković

Multicast sigurnost i zaštita autorskih prava

Diplomski rad

Mentor: prof. dr. sc. Ivan Matić

Osijek, 2023.

Sadržaj

1 Uvod

Od samih početaka ljudske povijesti i razvoja međusobne komunikacije paralelno dolazi i do razvoja težnje ka sigurnom načinu razmjene informacija. Jedna od najpoznatijih šifri, Cezarova šifra, nastala je još u doba antičkog Rima, gdje ju je sami Cezar koristio za razmjenu poruka sa svojim bliskim suradnicima. U današnje, nešto modernije doba, svjedočimo promptnom razvoju tehnologije, a time i kanala komunikacije gdje se razmjena informacija događa gotovo u trenutku. Samim time, javila se mogućnost i istodobnog slanja informacija većem broju korisnika, koja u sigurnosnom smislu predstavlja svojevrsni imperativ. Najčešći primjeri današnjice su streaming platforme poput Netflixa, HBO-a, MAXtv-a i dr., ali i sama komunikacija putem društvenih mreža i grupnih videopoziva. Smisao je ostao isti - poslati poruku korisniku i pritom onemogućiti razumijevanje iste korisniku kojemu ona nije namijenjena - no, učiniti to za više korisnika istovremeno.

Upravo tom tematikom bavimo se u ovom diplomskom radu koji je sadržajno podijeljen u tri glavna dijela. U prvom dijelu rada definiramo pojam *multicast* i dvije najčešća vrste takvog emitiranja. Zatim se bavimo pitanjem sigurnosti i nudimo odgovor na dva načina - enkripcijom odašiljanja te ponovnim generiranjem ključa. Enkripcija odašiljanja pokušava odgovoriti na ovaj izazov koristeći BES i generalizirani BES algoritam, čije dobre i loše strane ističemo te ih ilustriramo pokojim primjerom. Drugi način osiguravanja tajnosti poruke je ponovno generiranje ključa, koje promatramo kroz tri sheme: Blacklisting shemu, Naor-Pinkas shemu te shemu hijerarhije ključeva, uz odgovarajuće primjere.

Drugi dio rada namijenjen je zaštiti autorskih prava, još jednom učestalom problemu internetskog doba. Svi smo mi zasigurno barem jednom upalili film ili seriju putem besplatne web stranice, no jesmo li se ikada zapitali na koji način su vlasnici te stranice došli do tog sadržaja? Ovim pitanjem bavimo se u drugom poglavlju, gdje se osvrćemo na distribuciju podataka pomoću otiska prsta, odnosno kodnih riječi te identifikaciju familije korisnika koja pokušava konstruirati piratsku kopiju podataka za daljnje plasiranje.

U konačnici, treće poglavlje je rezervirano za praćenje nezakonito redistribuiranih ključeva. U određenom kodu, skup ključeva piratskog dekodera može se smatrati kodnom riječju, a ukoliko se on može pratiti, moguće ga je zaplijeniti i naći krivca. Ovo poglavlje posvećeno je upravo pronalasku kodova za koje možemo identificirati roditeljske kodove.

2 Multicast sigurnost

Kako bismo uopće mogli govoriti o sigurnosti, prije svega se upoznajmo s pojmom *multicast*, anglizmom koji je toliko dobro međunarodno prihvaćen, da se najčešće ni ne prevodi. Prema Institutu za hrvatski jezik i jezikoslovlje [5], multicast u duhu hrvatskog jezika prevodimo kao *višesmjerno odašiljanje*, odnosno *istodobno slanje neke poruke kroz mrežu dviju radnih stanica ili više njih.* U kontekstu naše teme, najjednostavnije rečeno, radi se o poruci koja ima više različitih naznačenih primatelja. Dakle, promatramo mrežu korisnika u kojoj je moguće istodobno emitirati poruke prema svim korisnicima te nas zanima sigurnost takve mreže. Navedimo dva najčešća primjera ovakvog emitiranja:

Emitiranje iz jednog izvora, u kojemu jedan entitet emitira informacije mreži korisnika koju ponekad nazivamo i grupa (primjerice, pretplatničke televizije, poput MAX-TV, Iskon TV, Eon TV itd.). Ovakve grupe su najčešće dugovječne, ali i vrlo dinamične, što znači da mnoštvo korisnika s vremenom ulazi i izlazi iz grupe te su nam potrebni algoritmi za uključivanje i isključivanje korisnika. U situaciji izlaska korisnika iz grupe koristimo algoritam opoziva ključa kako bismo obrisali ili onemogućili korištenje njegova ključa, uz ažuriranje ključeva ostalih korisnika koji pripadaju grupi. Jednostavan primjer ovakvog emitiranja vezan uz pretplatničke televizije je primjerice kupovina određenog sportskog kanala za vrijeme trajanja sezone Svjetskog skijaškog kupa. Pretplatnik dobije jedinstveni PIN (u kontekstu teme: ključ) koji mu omogućuje pristup tome kanalu, a po završetku pretplate njegov PIN se onemogućuje. U ovakvom tipu emitiranja, cilj nam je postići povjerljivost te autentičnost emitirane poruke.

Virtualna konferencija, u kojoj pretpostavljamo da jedan dio korisnika veće grupe želi održati virtualni sastanak, primjerice online videopoziv ili sastanak odbora formiranog od određenih članova grupe. Za razliku od emitiranja iz jednog izvora, ova grupa je kratkovječna, ali i statična. Glavni cilj ovakvog tipa sastanka je omogućiti bilo kojem članu sastanka da šalje informacije svim ostalim članovima. Povjerljivost pri ovoj vrsti odašiljanja postižemo uspostavljanjem privremenog ključa sastanka koji je poznat samo prisutnim članovima. Ovisno o vrsti sastanka, ponekad može biti poželjno odraditi provjeru autentičnosti (recimo, kod online sastanaka pri izvođenju online nastave) ili nam može biti bitna anonimnost članova (ukoliko, primjerice, provodimo tajno online glasovanje). U svakom slučaju, potrebno je osigurati da je pošiljatelj poruke uvijek član navedene grupe. Ovakav način komuniciranja posebno je dobio na popularnosti tijekom i nakon COVID pandemije te je tada osobito bitno bilo osigurati povjerljivost pri korištenju online platformi za komunikaciju poput Zoom-a, Google Meets-a, BigBlueButton-a i ostalih.

2.1 Enkripcija odašiljanja

Uvedimo najprije oznake koje ćemo koristiti u ovom poglavlju. Neka je $\mathcal{U}$ mreža koja se sastoji od n korisnika, P podskup (ponekad se naziva i privilegirani podskup) korisnika mreže $\mathcal{U}$, a $\mathcal{M}$ poruka koja treba biti emitirana. Prema *Shemi šifriranja emitiranjem*, pouzdano tijelo želi emitirati šifriranu (enkriptiranu) poruku podskupu P mreže korisnika $\mathcal{U}$. Recimo, unutar već spomenutih pretplatničkih televizija, film može biti šifriran pomoću blok šifre s tajnim ključem K, odnosno

$$y = e_K(\mathcal{M}), \quad \text{pri čemu je} \quad e_K \colon \mathcal{P} \to \mathcal{E},\ e_K(x) = x^e \mod n, \quad x \in \mathcal{P}$$

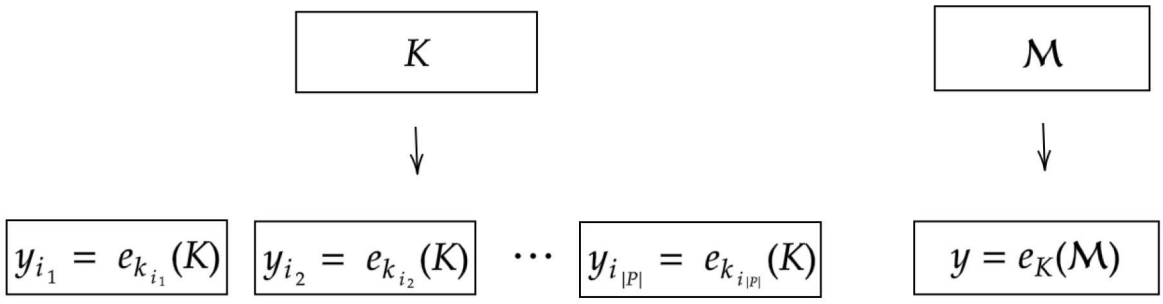
gdje je $\mathcal{P}$ skup svih otvorenih tekstova, a $\mathcal{E}$ skup svih šifrata. Jasno je da je u ovakvoj definiciji $\mathcal{M} \subset \mathcal{P}$. Korisnik $\mathcal{U}_i$ koji nije u povlaštenom skupu korisnika $\mathcal{P}$ može imati pristup emitiranju, ali ne smije biti u mogućnosti izračunati ključ šifriranja K. Također, treba imati na umu da skup $\mathcal{P}$ općenito nije unaprijed poznat prije postavljanja sheme te ju stoga možemo koristiti na više povlaštenih skupova tijekom određenog vremenskog razdoblja.

Prva shema na koju ćemo se malo detaljnije osvrnuti je *Trivijalna shema šifriranja emitiranjem.* U setup fazi, pouzdano tijelo daje ključ k_i svakom pojedinom korisniku $\mathcal{U}_i \in \mathcal{U}$. Za svakog korisnika $\mathcal{U}_i \in \mathcal{P}$, pouzdano tijelo šifrira K računajući $y_i = e_{k_i}(K)$. Svaki korisnik iz povlaštenog skupa P može dešifrirati y_i kako bi došao do ključa K te potom njime doći do poruke $\mathcal{M}$, ali korisnik izvan povlaštenog skupa ne može dešifrirati nijedan y_i, a time ni y u konačnici. U nastavku navodimo opisani kriptosustav.

Trivijalna shema šifriranja emitiranjem (BES)

- **Preraspodjela ključeva:** Pouzdano tijelo svakom od korisnika $U_i \in \mathcal{U}$ daje ključ k_i.
- **Šifriranje ključa:** Neka je P skup povlaštenih korisnika. Pouzdano tijelo šifrira K ključem k_i, za svaki i takav da je $U_i \in P$.
- **Šifriranje poruke:** Poruka $\mathcal{M}$ je šifrirana ključem K, odnosno $y = e_K(\mathcal{M})$. Emitiranje b_P se sastoji od P, y i liste šifriranih ključeva:

$$b_P = \{e_{k_i}(K) : U_i \in P\}.$$



Slika 1: Shema trivijalnog šifriranja emitiranjem

b_P je $|P|-$torka šifriranih ključeva, pa je širenje emitirane poruke jednako $|P|$. Zbog korištenja samo jednog ključa po korisniku, ova shema koristi malo prostora za pohranu te je odlikuje visoka sigurnost jer korisnici koji nisu u povlaštenom skupu ne mogu izračunati K. Ono što je mana ovakve sheme je visok stupanj širenja poruka. Cilj nam je naći dobar kompromis između ovih svojstava; primjerice, možemo tolerirati smanjenje razine sigurnosti i povećanje pohrane, ukoliko je stupanj proširenja emitirane poruke niži. To nas dovodi do generalizacije dosadašnjeg postupka koja će nam omogućiti veću efikasnost.

Pretpostavimo da je $P \subseteq \mathcal{U}$ povlašteni skup, r i v cijeli brojevi i sa w označimo maksimalnu veličinu familije korisnika, odnosno w je sigurnosni parametar za BES. Glavna razlika *Generalizirane sheme šifriranja emitiranjem* u odnosnu na Trivijalnu shemu je podjela ključa K na dijelove, a svaki taj dio je ujedno i šifriran te se takav odašilje. Za više detalja o učinkovitim konstrukcijama generalizirane sheme šifriranja emitiranjem koristeći Fiat-Naor raspodjelu ključeva može se pogledati u [3, Section 10.4.1].

Generalizirana shema šifriranja emitiranja

- **Preraspodjela ključeva:** Pouzdano tijelo svakom od korisnika $U_i \in \mathcal{U}$ dijeli materijal za ključ za v shema preraspodjele ključa K.
- **Tajno djeljenje:** Pouzdano tijelo bira tajni ključ K i razdvaja ga na dijelove $s_1, s_2, \ldots, s_v$ koristeći (r, v)-graničnu shemu.
- **Šifriranje/dešifriranje dijelova ključa:** Neka je P skup povlaštenih korisnika. Za $1 \leq i \leq v$, pouzdano tijelo šifrira s_i ključem k_i, tako da vrijedi:
  1) Svaki korisnik $U_j \in P$ može izračunati barem r od $k_1, \ldots, k_v$ ključeva (s obzirom da U_j može dešifrirati r dijelova ključa K i zatim rekonstruirati K).
  2) Svaki skup korisnika F, takav da $F \cap P = \emptyset$ and $|F| \leq w$ može izračunati najviše $r-1$ ključeva (s obzirom da F može dešifrirati najviše $r-1$ dijelova od K te ne može dobiti nikakve podatke o ključu).
- **Šifriranje poruke:** Poruka $\mathcal{M}$ je šifrirana ključem K, odnosno $y = e_K(\mathcal{M})$. Emitiranje b_P se sastoji od P, y i liste šifriranih ključeva:

$$b_P = \{e_{k_1}(s_1), \ldots, e_{k_v}(s_v)\}.$$

Označimo sa $1 - KDP$ (*engl. key distribution pattern*) ključeve koji su sigurni od uljeza te konstruiramo takvih v ključeva.
Neka je M matrica dimenzije $v \times n$ koja sadrži vrijednosti iz skupa $\{0, 1\}$ i označava koji je korisnik povezan s određenim KDP-em, koje ćemo označiti s $\mathcal{F}_1, \ldots, \mathcal{F}_v$.

Kažemo da korisniku U_j pripada ključ $\mathcal{F}_i$ ako i samo ako je $M[i, j] = 1$.
Drugim riječima,

$$korisnik(i) = \{U_j \colon M[i, j] = 1, 1 \leq i \leq v\}, \tag{1}$$

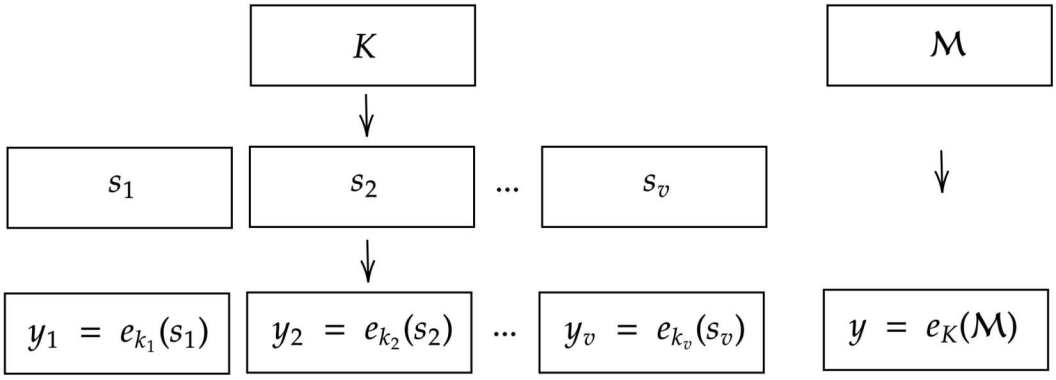
$$shema(j) = \{i \colon M[i, j] = 1, 1 \leq j \leq n\}.$$

Kako bismo se zaista upoznali s ovim procesom, preostaje nam navesti algoritme šifriranja i dešifriranja generaliziranog BES algoritma.

Algoritam šifriranja generaliziranog BES-a

1: **procedure**
2: **Ulaz**: Ključ K, povlašteni skup P i poruka $\mathcal{M}$.
3: Pouzdano tijelo koristi algoritam za generiranje podjele (r, v)-sheme kako bi konstruirali v podjela ključa K, u oznaci $s_1, \ldots, s_v$.

4: Za $1 \leq i \leq v$, pouzdano tijelo računa grupni ključ k_i za podskup korisnika iz $P \cap korisnik(i)$ u shemi $\mathcal{F}_i$.
5: Za $1 \leq i \leq v$, pouzdano tijelo računa $b_P = e_{k_i}(s_i)$.
6: Pouzdano tijelo računa $y = e_K(\mathcal{M})$.
7: **end procedure**
8: **Izlaz:** Šifrirana poruka y i emitiranje (prijenos) b_P.



Slika 2: Shema generaliziranog šifriranja emitiranjem

Sami algoritam šifriranja zahtjeva da bude zadovoljeno $|korisnik(j)| = r$, za svaki $j, 1 \leq j \leq n$, odnosno svakom korisniku pridružujemo točno r od v shema.

Algoritam dešifriranja generaliziranog BES-a

1: **procedure**
2: **Ulaz**: Povlašteni skup P, šifrirana poruka y i prijenos b_P.
3: Korisnik U_j konstruira grupu ključeva k_i za podskup korisnika iz $P \cap korisnik(i)$ u shemi $\mathcal{F}_i$, za svaki $i \in shema(j)$.
4: U_j računa $s_i = d_{k_i}(b_i)$, za svaki $i \in shema(j)$.
5: U_j koristi algoritam za rekonstrukciju podjele (r, v)-sheme kako bi izračunao K iz r dijelova u skupu $\{s_i : i \in shema(j)\}$.
6: U_j računa $\mathcal{M} = d_K(y)$.
7: **end procedure**
8: **Izlaz:** Dešifrirana poruka $\mathcal{M}$.

Kada govorimo o dekripciji poruke koju je emitirao korisnik iz povlaštenog skupa P, vidimo da je govorimo o proceduri koja je izravna. Ono o čemu još moramo voditi računa jest sigurnost pri izvedbi procedure generaliziranog BES-a.

Pretpostavimo da je F familija korisnika koja je disjunktna skupu korisnika P. Kako bi prethodno opisana procedura bila sigurna od napada, familija F može biti u mogućnosti izračunati najviše $r - 1$ od ukupno v ključeva. To znači da F može dešifrirati najviše $r - 1$ podjela, a kako je granična vrijednost ovog algoritma r, familija F ne može doći do informacije o ključu K. Matrica M određuje koje od ključeva $k_1, \ldots, k_v$ ova familija može izračunati, stoga nam je važno da matrica incidencije zadovoljava određena svojstva.

Općenito, kako bi postavilo shemu $\mathcal{F}_i$, pouzdano tijelo dijeli ključeve na sljedeći način:

1) Svakom korisniku u skupu $korisnik(i)$ dan je ključ l_i,

2) Za svaki $U_j \in korisnik(i)$ ključ $l_{i,j}$ je dan svakom korisniku iz skupa $korisnik(i) \setminus \{U_j\}$.

Tada grupni ključ za korisnike iz skupa $korisnik(i) \cap P$ unutar familije $\mathcal{F}_i$ računamo iz izraza u nastavku:

$$k_i = l_i + \sum_{\{j\colon U_j \in korisnik(i)\setminus P\}} l_{i,j}. \tag{2}$$

Niti jedan pojedinac koji nije u presjeku $korisnik(i) \cap P$ ne može izračunati ključ k_i, ali dva ili više korisnika iz skupa $korisnik(i) \setminus P$ mogu izračunati k_i. Iz toga slijedi da familija F može izračunati grupni ključ k_i ako i samo ako je $|F \cap korisnik(i)| \geq 2$.

Sada pokažimo kako funkcionira generalizirani BES algoritam na primjeru.

Primjer 2.1 (vidjeti [3, Example 14.1]). *Pretpostavimo da imamo sedam korisnika unutar mreže i želimo konstruirati generalizirani BES koji koristi također sedam KDP-ova ($n = 7, v = 7$). Zadana je sljedeća matrica incidencije:*

$$M = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Iz matrice vidimo da je svaki korisnik povezan s ukupno tri sheme ($r = 3$) te da je svaki KDP definiran na podskupu koji čine tri od ukupno sedam korisnika. U fazi postavljanja, konstruiramo sedam KDP-eva tako da svaki $\mathcal{F}_i, i \in \{1, \ldots, 7\}$, ima ukupno četiri ključa, a svaki korisnik iz skupa $korisnik(i)$ prima tri od ta četiri ključa.

Kako bismo lakše shvatili princip raspodjele ključeva, pretpostavimo da skup $korisnik(i)$ sadrži tri elementa (korisnika) koji su redom α, β i γ. Ključ l_i tada je dan svim trima navedenim korisnicima. Ključ $l_{i,\alpha}$ dan je korisnicima β i γ, ključ $l_{i,\beta}$ dan je korisnicima α i γ, dok je ključ $l_{i,\gamma}$ dan je korisnicima α i β. Ova procedura ponovljena je za svih sedam shema.

Iz matrice M tako vidimo da ključ l_1 dobivaju korisnici U_1, U_2 i U_4. Zatim, ključ $l_{1,1}$ dan je korisnicima U_2 i U_4, ključ $l_{1,2}$ korisnicima U_1 i U_4, a korisnici U_1 i U_2 dobivaju ključ $l_{1,4}$. Analognim postupkom dijelimo preostale ključeve. U Tablici 1 u nastavku dan je popis svih korisnika i ključeva koji su im dani.

Pretpostavimo da pouzdano tijelo želi emitirati poruku povlaštenom skupu $P = \{U_1, U_2, U_3\}$. Za $i = 1$, prema formuli (1),

$$korisnik(1) \cap P = \{U_j\colon M[1, j] = 1\} \cap \{U_1, U_2, U_3\} = \{U_1, U_2, U_4\} \cap \{U_1, U_2, U_3\} = \{U_1, U_2\}.$$

Grupni ključ k_1 tada računamo prema formuli (2) i dobivamo:

$$k_1 = l_1 + \sum_{\{j\colon U_j \in korisnik(1)\setminus P\}} l_{1,j} = l_1 + \sum_{\{j\colon U_j \in \{U_1,U_2,U_4\}\setminus\{U_1,U_2,U_3\}\}} l_{1,j} = l_1 + \sum_{\{j\colon U_j \in \{U_4\}\}} l_{1,j} = l_1 + l_{1,4}.$$

$\mathbf{U_1}$	$\mathbf{U_2}$	$\mathbf{U_3}$	$\mathbf{U_4}$	$\mathbf{U_5}$	$\mathbf{U_6}$	$\mathbf{U_7}$
l_1	l_1	l_2	l_1	l_2	l_3	l_4
$l_{1,2}$	$l_{1,1}$	$l_{2,2}$	$l_{1,1}$	$l_{2,2}$	$l_{3,3}$	$l_{4,4}$
$l_{1,4}$	$l_{1,4}$	$l_{2,5}$	$l_{1,2}$	$l_{2,3}$	$l_{3,4}$	$l_{4,5}$
l_5	l_2	l_3	l_3	l_4	l_5	l_6
$l_{5,5}$	$l_{2,3}$	$l_{3,4}$	$l_{3,3}$	$l_{4,4}$	$l_{5,1}$	$l_{6,2}$
$l_{5,6}$	$l_{2,5}$	$l_{3,6}$	$l_{3,6}$	$l_{4,7}$	$l_{5,5}$	$l_{6,6}$
l_7	l_6	l_7	l_4	l_5	l_6	l_7
$l_{7,3}$	$l_{6,6}$	$l_{7,1}$	$l_{4,5}$	$l_{5,1}$	$l_{6,2}$	$l_{7,1}$
$l_{7,7}$	$l_{6,7}$	$l_{7,7}$	$l_{4,7}$	$l_{5,6}$	$l_{6,7}$	$l_{7,3}$

Tablica 1: Raspodjela ključeva

i	$korisnik(i) \cap P$	k_i
1	$\{U_1, U_2\}$	$l_1 + l_{1,4}$
2	$\{U_2, U_3\}$	$l_2 + l_{2,5}$
3	$\{U_3\}$	$l_3 + l_{3,4} + l_{3,6}$
4	$\emptyset$	$l_4 + l_{4,4} + l_{4,5} + l_{4,7}$
5	$\{U_1\}$	$l_5 + l_{5,5} + l_{5,6}$
6	$\{U_2\}$	$l_6 + l_{6,6} + l_{6,7}$
7	$\{U_1, U_3\}$	$l_7 + l_{7,7}$

Tablica 2: Popis grupnih ključeva

Na potpuno analogan način dolazimo i do preostalih ključeva $k_2, \ldots, k_7$, koji su dani u Tablici 2.

Neka su $s_1, \ldots, s_7$ dijelovi tajnog ključa K. Prijenos $b_P = (b_1, \ldots, b_7)$ se tada sastoji od:

$$\begin{aligned} b_1 &= e_{l_1+l_{1,4}}(s_1) \\ b_2 &= e_{l_2+l_{2,5}}(s_2) \\ b_3 &= e_{l_3+l_{3,4}+l_{3,6}}(s_3) \\ b_4 &= e_{l_4+l_{4,4}+l_{4,5}+l_{4,7}}(s_4) \\ b_5 &= e_{l_5+l_{5,5}+l_{5,6}}(s_5) \\ b_6 &= e_{l_6+l_{6,6}+l_{6,7}}(s_6) \\ b_7 &= e_{l_7+l_{7,7}}(s_7). \end{aligned}$$

Dakle, svaki od korisnika U_1, U_2 i U_3 može izračunati tri ključa: U_1 može izračunati ključeve k_1, k_5 i k_7; U_2 može izračunati k_1, k_2 i k_6; U_3 može izračunati ključeve k_2, k_3 i k_7.

Osvrnimo se sada na sigurnost ovog algoritma i pokažimo da je ovako konstruirani algoritam siguran u slučaju napada familije veličine $w = 2$.

Pretpostavimo da imamo bilo koji dvočlani podskup korisnika $\{U_j, U_{j'}\}$. Tada postoji točno jedan KDP $\mathcal{F}_i$ tako da je $\{U_j, U_{j'}\} \subseteq korisnik(i)$. Neka je $F = \{U_j, U_{j'}\}$ proizvoljna dvočlana familija disjunktna povlaštenom skupu P. Slijedi da F može izračunati samo jedan od sedam grupnih ključeva, stoga F može dešifrirati samo jednu od sedam šifriranih podjela.

Korištena granična vrijednost pri tajnoj shemi raspodjele je $r = 3$, pa familija nema dovoljno dijelova da bi mogla otkriti tajnu poruku.

Vratimo se sada na svojstva koja matrica M mora zadovoljavati u općenitom slučaju kako bi navedeni algoritam jamčio sigurnost. M treba biti matrica incidencije tipa $v \times n$ koja u svakom stupcu ima točno r jedinica. Neka je $(\mathcal{U}, \mathcal{A})$ hipergraf[1] uređen n-torkom podskupova od $korisnik(i)$, u kojoj su blokovi formirani po retcima matrice M. Tada $(\mathcal{U}, \mathcal{A})$ sadrži točno n točaka (koje predstavljaju korisnike) i v blokova (koji predstavljaju sheme) te se svaka točka nalazi u točno r blokova.
Pretpostavimo da prethodno spomenuta uređena n-torka podskupova ima svojstvo da se svaki par točaka pojavljuje u najviše λ blokova, $\lambda \in \mathbb{Z}$. Tada familija F veličine w može izračunati najviše $\lambda\binom{w}{2}$ grupnih ključeva, jer postoji $\binom{w}{2}2-$ podskupova od F, od kojih svaki može izračunati najviše λ grupnih ključeva. Dakle, generalizirani BES je siguran od familije veličine w ako je $r > \lambda\binom{w}{2}$.

Općenito, želimo konstruirati matricu incidencije M, pri čemu zahtjevamo da pripadna uređeni par $(\mathcal{U}, \mathcal{A})$ zadovoljava:

1) $|\mathcal{U}| = n$ (postoji n točaka),

2) $|\mathcal{A}| = v$ (postoji v blokova),

3) svaka točka sc pojavljujc u točno r blokova,

4) svaki par točaka se pojavljuje u najviše λ blokova,

5) $r > \lambda\binom{w}{2}$.

Sustav koji zadovoljava svojstva 1) $-$ 4) označit ćemo sa (n, v, r, λ)-BKDP, odnosno (n, v, r, λ)-obrascem distribucije emitiranih ključeva. Sljedeći teorem govori nam o tome pod kojim je uvjetima (n, v, r, λ)-BKDP generalizirani BES.

Teorem 2.1 (vidjeti [3, Theorem 14.1]). *Pretpostavimo da je (n, v, r, λ)-BKDP i neka je w prirodan broj takav da je $\binom{w}{2} < \frac{r}{\lambda}$. Tada postoji shema enkripcije prijenosom za mrežu od n korisnika koja zadovoljava sljedeća svojstva:*

1. *šifrirana poruka je sigurna od familije korisnika veličine w,*

2. *širenje poruke je jednako v,*

3. *svaki korisnik je dužan pohraniti najviše $r + \lambda(n-1)$ ključeva.*

Dokaz. Neka je $(\mathcal{U}, \mathcal{A})$ (n, v, r, λ)-BKDP. S obzirom da smo svojstva 1. i 2. dosad već diskutirali, preostaje nam dokazati svojstvo 3. Neka je U_j proizvoljni korisnik. Za svaki $i \in shema(j)$, korisnik U_j prima ključeve iz $\mathcal{F}_i$. Preciznije, U_j prima točno $|korisnik(i)|$ ključeva iz sheme $\mathcal{F}_i$ za $i \in shema(j)$. Ukupan broj ključeva koji prima U_j iskazan u terminu uređene n-torke podskupova $(\mathcal{U}, \mathcal{A})$ je

$$\sum_{\{i\colon i \in shema(j)\}} |korisnik(i)|.$$

[1]Prema [4], hipergraf H sastoji se od skupa vrhova $V(H)$ i familije podskupova $E(H)$ od skupa $V(H)$, čije elemente nazivamo hiperbridovima. Iz definicije slijedi da hiperbrid može spajati više od dva vrha.

Kako za $(\mathcal{U}, \mathcal{A})$ vrijedi da se svaka točka pojavljuje u točno r blokova, vrijedi

$$\sum_{\{i\colon i \in shema(j)\}} 1 = r.$$

S obzirom da postoji ukupno n točaka (korisnika) i da se svaki par točaka pojavljuje u najviše λ blokova, slijedi

$$\sum_{\{i\colon i \in shema(j)\}} |korisnik(i) - 1| \leq \lambda(n-1).$$

Iz prethodna dva izraza dobivamo

$$\sum_{\{i\colon i \in shema(j)\}} |korisnik(i)| \leq r + \lambda(n-1).$$

□

Sljedeći teorem predstavlja nam nešto drugačiju konstrukciju BKDP-a, zasnovanu na polinomima, čiji parametri tvore još efikasnije sheme.

Teorem 2.2 (vidjeti [3, Theorem 14.2]). *Pretpostavimo da je q prost broj i neka je d cijeli broj za kojeg vrijedi $2 \leq d \leq q$. Tada postoji $(q^d, q^2, q, d-1)$-BKDP.*

Dokaz. Konstruiramo matricu incidencije ovog BKDP-a. Stupci matrice M su označeni d-torkama $(a_0, \ldots, a_{d-1}) \in (\mathbb{Z}_q)^d$ (d-torka odgovara polinomu u prstenu polinoma $\mathbb{Z}_q[x]$ stupnja najviše $d-1$). Retci matrice M su označeni uređenim parovima $(x, y) \in (\mathbb{Z}_q)^2$. Elementi matrice su definirani sljedećim izrazom:

$$M((x,y),(a_0,\ldots,a_{d-1})) = 1 \iff \sum_{i=0}^{d-1} a_i x^i \equiv y \,(\mathrm{mod}\ q). \tag{3}$$

Kako svaki polinom za bilo koji x daje jedinstveni y, vrijedi $r = q$. Preostaje izračunati parametar λ. Pretpostavimo da imamo dva stupca matrice M, u oznaci $a = (a_0, \ldots, a_{d-1})$ i $a' = (a'_0, \ldots, a'_{d-1})$. Ova dva stupca definiraju dva polinoma koje će redom označiti s $a(x)$ i $a'(x)$. Želimo ograničiti odozgo $\lambda_{a,a'}$,

$$\lambda_{a,a'} = |\{(x,y)\colon a(x) = a'(x) = y\}| = |\{(x,y)\colon a(x) = a'(x)\}|.$$

Zbog činjenice da vrijednosti u d točaka određuju jedinstven polinom stupnja najviše $d-1$, jasno je da $\lambda_{a,a'} \leq d-1$, pa možemo uzeti $\lambda = d-1$. □

Rezimirajmo ukratko što nam je sve potrebno za polinomijalnu konstrukciju BKDP-a: neka je q prost broj i neka je $d < q$. Tada postoji matrica incidencije za BKDP takav da je $n = q^d, v = q^2, r = q$ i $\lambda = d-1$. Prema dosad pokazanim tvrdnjama o sigurnosti BES-a, ovakav BES je siguran od familije korisnika veličine najviše w ako je $q > (d-1)\binom{w}{2}$.

Primjerice, uzmimo u obzir familiju korisnika veličine $w = 3$ i želimo postići sigurnost u ovakvoj shemi. Za bilo koji prost broj q, iz prethodno navedenog izraza slijedi da je

$$d < 1 + \frac{q}{\binom{w}{2}},$$

pa za potrebe primjera možemo uzeti $d = \lfloor 1 + \frac{q}{3} \rfloor$. Možemo konstruirati generalizirani BES za $n = q^d$ korisnika, pri čemu je $v = q^2$. U sljedećoj tablici prikazan je izračunati broj korisnika za nekoliko prostih brojeva.

Ono što ovdje treba uočiti je da broj korisnika mreže n zaista brzo raste kako raste q. S druge strane, veličina prijenosa ovisi o v, koji raste dosta sporije u odnosu na broj korisnika.

q	d	v	n
5	2	25	25
11	4	121	14 641
17	6	289	24 137 569

Tablica 3: Parametri BKDP-a za nekoliko prostih brojeva q

Primjer 2.2 (vidjeti [3, Example 14.2]). *S obzirom da za $q = 2$ imamo samo 2 korisnika u mreži, a za $q = 5$ broj korisnika raste već na $n = 25$, promotrimo BKDP za $q = 3$. Dakle, konstruirajmo matricu incidencije za $(9, 9, 3, 1)$-BKDP. Znamo da je $d = \lambda + 1 = 2$, $v = q^2 = 9$ te $n = q^d = 9$. Retci matrice M su ovdje uređeni parovi $(x, y) \in (\mathbb{Z}_3)^2$, dok su stupci koeficijenti polinoma iz $\mathbb{Z}_3[x]$ najvećeg mogućeg stupnja 2, u oznaci (a_0, a_1). Svi mogući takvi polinomi su redom:*

$$\begin{aligned}
p_1(x) &= 0 && ((a_0, a_1) = (0, 0)),\\
p_2(x) &= 1 && ((a_0, a_1) = (1, 0)),\\
p_3(x) &= 2 && ((a_0, a_1) = (2, 0)),\\
p_4(x) &= x && ((a_0, a_1) = (0, 1)),\\
p_5(x) &= 1 + x && ((a_0, a_1) = (1, 1)),\\
p_6(x) &= 2 + x && ((a_0, a_1) = (2, 1)),\\
p_7(x) &= 2x && ((a_0, a_1) = (0, 2)),\\
p_8(x) &= 1 + 2x && ((a_0, a_1) = (1, 2)),\\
p_9(x) &= 2 + 2x && ((a_0, a_1) = (2, 2)).
\end{aligned}$$

Koristeći izraz (3), popunjavamo matricu M nulama i jedinicama. Navodimo primjer računanja za uređene parove $(0, 0)$ te $(1, 2)$, a preostalih 7 parova računamo na analogan način.

- $(x, y) = (0, 0)$*:*

  $M((0,0),(0,0)) = 1$, *jer je* $0 + 0 \cdot 0 = 0 \equiv 0 \pmod 3$
  $M((0,0),(1,0)) = 0$, *jer je* $1 + 0 \cdot 0 = 1 \equiv 1 \pmod 3$
  $M((0,0),(0,1)) = 1$, *jer je* $0 + 1 \cdot 0 = 0 \equiv 0 \pmod 3$
  $M((0,0),(2,1)) = 0$, *jer je* $2 + 1 \cdot 0 = 2 \equiv 2 \pmod 3$
  $\vdots$

- $(x, y) = (1, 2)$*:*

  $M((1,2),(0,0)) = 0$, *jer je* $0 + 0 \cdot 1 = 0 \equiv 0 \pmod 3$
  $M((1,2),(2,0)) = 1$, *jer je* $2 + 0 \cdot 1 = 2 \equiv 2 \pmod 3$
  $M((1,2),(2,1)) = 0$, *jer je* $2 + 1 \cdot 1 = 3 \equiv 0 \pmod 3$
  $M((1,2),(0,2)) = 1$, *jer je* $0 + 2 \cdot 1 = 2 \equiv 2 \pmod 3$
  $\vdots$

U konačnici dolazimo do sljedeće matrice, koju ćemo radi jednostavnosti i jasnoće prikazati kao tablicu:

(x,y)	$(0,0)$	$(1,0)$	$(2,0)$	$(0,1)$	$(1,1)$	$(2,1)$	$(0,2)$	$(1,2)$	$(2,2)$
$(0,0)$	1	0	0	1	0	0	1	0	1
$(0,1)$	0	1	0	0	1	0	0	1	0
$(0,2)$	0	0	1	0	0	1	0	0	1
$(1,0)$	1	0	0	0	0	1	0	1	0
$(1,1)$	0	1	0	1	0	0	0	0	1
$(1,2)$	0	0	1	0	1	0	1	0	0
$(2,0)$	1	0	0	0	1	0	0	0	1
$(2,1)$	0	1	0	0	0	1	1	0	0
$(2,2)$	0	0	1	1	0	0	0	1	0

Tablica 4: Matrica incidencije $(9,9,3,1)$-BKPD sheme

Valja napomenuti da na mnoge načine ovu priču možemo unaprijediti. Jedan od njih je svakako koristeći *Ramp-shemu*, čime smanjujemo širenje poruke dok istovremeno ostajemo na dosadašnjoj razini sigurnosti. Zainteresirani čitatelj o njima može više saznati u [3, str. 528.]

2.2 Ponovno generiranje ključa

U ovom potpoglavlju promatramo dugovječnu dinamičnu grupu, u oznaci $\mathcal{U}$, koja prima prijenos iz jednog izvora. Pouzdano tijelo želi emitirati poruku svim korisnicima unutar grupe, pri čemu korisnici mogu ulaziti i izlaziti iz grupe tijekom prijenosa. U ovakvoj situaciji koristimo *Shemu ponovnog generiranja ključa.* Komunikacija s grupom je šifrirana jednim grupnim ključem i svaki korisnik ima njegovu kopiju. Korisnici također mogu posjedovati dodatne dugotrajne ključeve (LL-ključeve, engl. *long-lived keys*) koji se koriste za ažuriranje sustava kako se grupa razvija tijekom vremena.

Sustav se postavlja u fazi preraspodjele ključeva tijekom koje pouzdano tijelo distribuira LL-ključeve i početni grupni ključ korisnicima unutar mreže. Pri pridruživanju novog korisnika grupi, taj korisnik dobiva kopiju trenutnog grupnog ključa, zajedno s odgovarajućim LL-ključevima. Ovaj proces se naziva *operacija pridruživanja korisnika.* U slučaju da korisnik U napušta grupu, izvršava se *operacija opoziva korisnika* koja uspostavlja novi grupni ključ za preostale korisnike grupe $\mathcal{U} \setminus \{U\}$ te ovaj proces nazivamo ponovnim generiranjem ključa. Također, nakon operacije opoziva korisnika može biti potrebno ažurirati LL-ključeve.

Kriteriji koji se koriste za procjenu shema ponovnog generiranja ključa su:

- **Složenost komunikacije i pohrane** - uključuje veličinu prijenosa potrebnih za ažuriranje ključa te veličinu i broj tajnih LL-ključeva koje korisnici moraju pohraniti.
- **Sigurnost** - uglavnom uzimamo u obzir sigurnost protiv opozvanih korisnika i familija takvih korisnika. Opasnost se javlja zbog činjenice da opozvani korisnik ima više informacija o grupi nego korisnik koji nikada nije bio u njoj.
- **Fleksibilnost opoziva korisnika** - neki slučajevi mogu zahtjevati opozivanje jednog po jednog korisnika, dok neke sheme mogu opozivati više korisnika u isto vrijeme (do određenog broja korisnika), što je praktičnije jer u tom slučaju korisnici rjeđe moraju ažurirati svoje ključeve.

- **Fleksibilnost korisničkog pridruživanja** - u nekim sustavima može se dogoditi da se bilo koji broj novih korisnika može lako dodati u sustav. S druge strane, moguće je da se cijeli sustav mora ponovno pokrenuti kako bi se moglo dodati nove korisnike (ovakav sustav smatrali bismo jednokratnim).

- **Učinkovitost ažuriranja LL-ključeva** - ovdje također postoji više mogućnosti; LL-ključevi mogu biti statični i ne zahtjevati ažuriranje, a moguće je i da zahtijevaju ažuriranje učinkovitom operacijom, recimo emitiranjem. U najgorem slučaju, cijeli sustav bi morao biti ponovno postavljen nakon opoziva korisnika i to bi značilo da sustav ne prihvaća opoziv.

Jedan od mogućih pristupa ovom problemu bilo bi korištenje shema šifriranjem emitiranjem obrađenim u prethodnom potpoglavlju, međutim one su namijenjene prvenstveno za emitiranje proizvoljnom podskupu korisnika u velikoj grupi. Ovdje imamo situaciju u kojoj obično opozivamo mali broj korisnika iz grupe, pa bi povlašteni skup najčešće uključivao gotovo sve korisnike u grupi. Postoje sheme koje su prilagođene ponovnom generiranju ključa te su puno učinkovitije u tome od općih rješenja sličnih enkripciji prijenosom. U nastavku ćemo detaljnije opisati tri takva pristupa:

- *Blacklisting sheme,* koje koriste $(1, w)$-nepokrivajuće familije,

- *Naor-Pinkas sheme,* bazirane na graničnim shemama,

- *Hijerarhija ključeva,* shema temeljena na stablima.

2.2.1 Blacklisting shema

Definicija 2.1. Kažemo da je $(\mathcal{U}, \mathcal{A})$ (t, w)-nepokrivajuća familija (engl. *cover-free family*) ako za bilo koja dva disjunktna podskupa blokova P, $F \subseteq \mathcal{A}$, vrijedi

$$\bigcap_{A_i \in P} A_i \not\subseteq \bigcup_{A_j \in F} A_j,$$

gdje je $|P| = t, |F| = w$.

Drugim riječima, presjek t blokova nikad nije prekriven unijom preostalih w blokova. Za konstrukciju ove sheme potreban nam je $(1, w)$-CFF, što znači da niti jedan blok nije podskup preostalih w blokova. Svaki blok je skup ključeva danih korisnicima unutar mreže, odnosno imamo n korisnika te svaki od njih prima podskup LL-ključeva, kojih je ukupno v.

Pretpostavimo sada da je F podskup korisnika koje želimo opozvati i da je $|F| = w$. Pouzdano tijelo na slučajan način bira K', što predstavlja novi ključ za preostale korisnike u grupi $\mathcal{U} \setminus F$. Za svaki $i \notin \bigcup_{U_j \in F} ključevi(U_j)$, pouzdano tijelo računa $y_i = e_{k_i}(K')$ i emitira y_i. Ovaj postupak nazivamo *Blacklisting shema.* Čak i ako korisnici iz F kombiniraju informacije koje imaju, niti jedan od njih ne može izračunati K' jer nije šifriran koristeći ključeve u posjedu članova familije F.

Iz definicije $(1, w)$-CFF-a, vrijedi

$$ključevi(U_h) \not\subseteq \bigcup_{U_j \in F} ključevi(U_j).$$

Dakle, za svakog korisnika $U_h \notin F$ postoji ključ u posjedu korisnika U_h koji je korišten pri šifriranju K', pa U_h može izračunati K' dešifriranjem prikladnog šifrata.

Primjer 2.3 (vidjeti [3, Example 14.6]). *Promotrimo slučaj* $(1,2)$*-CFF-a* $(\mathcal{X}, \mathcal{B})$*, gdje su*

$$\mathcal{X} = \{1, \ldots, 7\}, \quad \mathcal{B} = \{\{2,3,5\}, \{3,4,6\}, \{4,5,7\}, \{1,5,6\}, \{2,6,7\}, \{1,3,7\}, \{1,2,4\}\}.$$

Drugim riječima, korisnik U_1 *dobiva ključeve* k_2, k_3 *i* k_5*,* U_2 *dobiva ključeve* k_3, k_4 *i* k_6 *itd. Ovdje koristimo Shemu crne liste s parametrom* $w = 2$*, primjerice pretpostavimo da želimo opozvati korisnike* U_2 *i* U_5*. U tom slučaju, novi grupni ključ* K' *šifriramo bez korištenja ključeva* $\{k_3, k_4, k_6\} \cup \{k_2, k_6, k_7\}$ *te se emitiranje sastoji od dvije vrijednosti,* $y_1 = e_{k_1}(K')$ *i* $y_5 = e_{k_5}(K')$*. Svi preostali korisnici mogu dešifrirati jednu od ovih dviju vrijednosti, jer* U_1, U_3 *i* U_4 *imaju ključ* k_5*, dok korisnici* U_6 *i* U_7 *imaju ključ* k_1*. Međutim, niti* U_2 *niti* U_5 *nemaju ključeve* k_1 *ili* k_5 *te oni nisu u mogućnosti izračunati novi grupni ključ* K'*.*

Ova shema ima svojstvo da su LL-ključevi statični te je složenost prijenosa $O(\log n)$. Ukoliko je to potrebno, korisnici mogu biti opozvani u fazama tijekom nekog vremena. Recimo da želimo opozvati korisnike u F_i u fazi i, $1 \leq i \leq T$. Pretpostavljamo da je $|F_1| + |F_2| + \cdots + |F_T| \leq w$ i da su grupe $F_1, F_2, \ldots, F_T$ disjunktne. Sa K_i označimo grupni ključ u fazi i. Šifrati T grupa ključeva su emitirani na način koji je naveden u sljedećem algoritmu.

Opoziv $(F_1, \ldots, F_T; K_1, \ldots, K_T)$

1: **procedure**
2: $F \leftarrow \emptyset$
3: **for** $i \leftarrow 1$ to T **do**
4: $\begin{cases} F \leftarrow F \cup \{F_i\} \\ \text{u fazi } i\text{, emitiraj grupni ključ } K_i \text{ skupu } \mathcal{U} \setminus F. \end{cases}$
5: **end for**
6: **end procedure**

Koristeći ovu shemu, ne nalazimo prikladan način za ažuriranje LL-ključeva, pa shema mora biti konfigurirana nakon opoziva svih w korisnika.

2.2.2 Naor-Pinkas shema

Ova shema ponovnog generiranja ključa temelji se na Shamir graničnoj shemi, o kojoj se više može pronaći u [3, Section 13.1]. *Naor-Pinkas shema* distribuira dijelove novog grupnog ključa K' tijekom faze postavljanja sustava. Ključ K' je unaprijed postavljen, a emitiranje služi za njegovu aktivaciju. Nakon emitiranja svaki korisnik koji nije u grupi F ima $w+1$ dijelova, što znači da može izračunati novi ključ K'. Suprotno tome, korisnici u F ne mogu izračunati ključ jer posjeduju najviše w dijelova ključa.

S obzirom da svaki korisnik dobiva jedan dio novog ključa, pohranjuje $O(1)$ informacija, pa je cijeli prijenos složenosti tipa $O(w)$. Moguće je opozvati $w' < w$ korisnika emitiranjem dijelova od opozvanih w' korisnika, zajedno s $w - w'$ novo kreiranih dijelova ključa koji ne odgovaraju nijednom dosadašnjem korisniku.

Naor-Pinkas shema

- **Faza postavljanja:** Neka je $|U| = n$. Pouzdano tijelo konstruira n podjela novog grupnog ključa K', u oznaci $y_1, \ldots, v_n$, koristeći Shamir $(w+1, n)$- graničnu shemu. Svakom korisniku U_i dan je dio y_i.
- **Opoziv:** Neka je $|F| = w$ skup korisnika koje želimo opozvati. Pouzdano tijelo emitira w podjela y_i korisnicima $U_i \in F$.

Klasična Naor-Pinkas shema ne dozvoljava opoziv korisnika u fazama, međutim postoji verzija ove sheme koja dopušta opoziv više korisnika istovremeno, do maksimalno w opozvanih korisnika. Prije nego što detaljnije promotrimo ovu poboljšanu verziju, definirajmo grupu $\mathbb{Z}_p^*$.

Neka je $n \in \mathbb{N}$. Definirajmo $\mathbb{Z}_p^* = \{1, 2, \ldots, p-1\}$ te na tom skupu uvodimo operaciju množenja modulo n:

$$a \cdot_p b = a \cdot b \mod p, \quad \cdot_p \colon \mathbb{Z}_p \times \mathbb{Z}_p \to \mathbb{Z}_p.$$

Može se pokazati [1, Zadatak 1.1.8.] da je $(\mathbb{Z}_p^*, \cdot_p)$ komutativna grupa, ukoliko je p prost broj.

Pretpostavimo da su ključevi i njegovi dijelovi u graničnoj shemi definirani unutar podgrupe G od $\mathbb{Z}_p^*$ reda q, pri čemu je q prost broj. Neka je α generator podgrupe G. Pouzdano tijelo dijeli ključ $K \in \mathbb{Z}_q$ koristeći Shamir $(w+1, n)$-graničnu shemu u $\mathbb{Z}_q$. Tada ključ K možemo izračunati koristeći formulu

$$K = \sum_{j=1}^{w+1} b_j y_{i_j} \mod q,$$

gdje su b_j interpolacijski koeficijenti definirani na sljedeći način:

$$b_j = \prod_{\substack{1 \le k \le w+1, \\ k \ne j}} \frac{x_{ik}}{x_{ik} - x_{ij}}.$$

Tada je

$$\alpha^K = \prod_{j=1}^{w+1} \alpha^{b_j y_{i_j}} \mod p,$$

te za svaki r vrijedi

$$\alpha^{rK} = \prod_{j=1}^{w+1} \alpha^{r b_j y_{i_j}} \mod p.$$

Pretpostavimo da pouzdano tijelo emitira α^r zajedno s w dijelova ključa, odnosno $\gamma_j = \alpha^{r y_{t_j}}, 1 \le j \le w$. Korisnik koji nije opozvan, u oznaci $U_{i_{w+1}}$, može izračunati svoj vlastiti dio prema formuli $\gamma_{w+1} = \alpha^{r y_{t_{w+1}}} = (\alpha^r)^{y_{t_{w+1}}}$. Tada on može izračunati α^{rK} pomoću formule

$$\alpha^{rK} = \prod_{j=1}^{w+1} (\alpha^{r y_{t_j}})^{b_j} \mod p,$$

što predstavlja novi grupni ključ. Slučajnim odabirom vrijednosti r za svaki novi prijenos možemo generirati seriju novih ključeva. I u ovoj shemi familija koja sadrži više od w opozvanih korisnika može izračunati grupni ključ, stoga treba paziti da broj opozvanih korisnika ne prelazi w.

Primjer 2.4 (vidjeti [3, Example 14.7]). *Pretpostavimo da je* $q = 503$ *i* $p = 6q + 1 = 3019$ *te neka je* $\alpha = 64$. *Tada je* $\alpha^q \equiv 64^{503} \equiv 1 \bmod 3019$ *pa je* α *reda* q. *Također, pretpostavimo da je* $w = 2$ *i da pouzdano tijelo bira tajnu polinomijalnu kongruenciju*

$$a(x) = 109 + 215x + 307x^2 \mod 503.$$

Ključ je jednak slobodnom koeficijentu polinoma, odnosno $K = 109$.

Pretpostavimo da imamo pet korisnika u mreži i da pouzdano tijelo pridružuje korisnicima javne $x-$*koordinate* $x_1 = 15, x_2 = 30, x_3 = 45, x_4 = 60$ *i* $x_5 = 75$. *Računamo dijelove ključa koje dobiva svaki korisnik prema formuli* $y_{i_j} = a(x_{i_j})$. *Za prvog korisnika tada imamo:*

$$a(15) = 109 + 215 \cdot 15 + 307 \cdot (15)^2 \mod 503 \equiv 480 \mod 503,$$

pa je $y_1 = 480$. *Na isti način imamo* $y_2 = 173, y_3 = 194, y_4 = 40$ *i* $y_5 = 214$.
Pouzdano tijelo zatim bira slučajan broj, recimo $r = 423$, *te je tada novi grupni ključ jednak*

$$\alpha^{rK} \mod p \equiv 64^{423 \cdot 109} \mod 3019 = 2452.$$

Pretpostavimo da želimo opozvati korisnike U_2 *i* U_4. *Tada pouzdano tijelo emitira sljedeće informacije:*

$$\begin{aligned}
\alpha^r \mod p &\equiv 64^{423} \mod 3019 = 1341, \\
\alpha^{ry_2} \mod p &\equiv 64^{423 \cdot 173} \mod 3019 = 2457, \\
\alpha^{ry_4} \mod p &\equiv 64^{423 \cdot 40} \mod 3019 = 24.
\end{aligned}$$

Još nam preostaje uvjeriti se da, primjerice, korisnik U_1 *može izračunati novi ključ. Najprije* U_1 *računa svoj dio ključa*

$$1341^{480} \mod 3019 = 701.$$

Zatim računa interpolacijske koeficijente:

$$\begin{aligned}
\frac{x_2}{x_2 - x_1} \cdot \frac{x_4}{x_4 - x_1} \mod q &\equiv \frac{30}{30 - 15} \cdot \frac{60}{60 - 15} \mod 503 = 338, \\
\frac{x_1}{x_1 - x_2} \cdot \frac{x_4}{x_4 - x_2} \mod q &\equiv \frac{15}{15 - 30} \cdot \frac{60}{60 - 30} \mod 503 = 501, \\
\frac{x_1}{x_1 - x_4} \cdot \frac{x_2}{x_2 - x_4} \mod q &\equiv \frac{15}{15 - 60} \cdot \frac{30}{30 - 60} \mod 503 = 168.
\end{aligned}$$

Novi grupni ključ U_1 *računa iz izraza:*

$$701^{338} \cdot 2457^{501} \cdot 24^{168} \mod 3019 = 2452,$$

što i je vrijednost novog ključa.

2.2.3 Hijerarhija ključeva

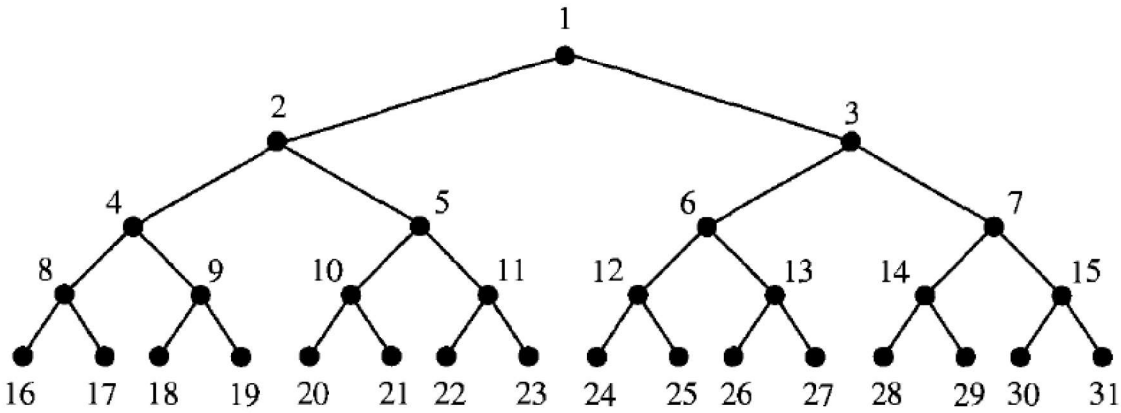
S obzirom da se ova shema temelji na binarnim stablima, recimo prvo pokoju riječ o istima. Binarno stablo se rekurzivno definira kao konačan skup čvorova koji je ili prazan, ili se sastoji od korijena s dva posebna podstabla, lijevim i desnim, koji su također binarna stabla. Za odnose između čvorova unutar stabla uobičajeno je koristiti obiteljsku terminologiju, pa je tako čvor X koji je spojen izravno sa čvorovima Y_1 i Y_2 roditelj tih dvaju čvorova, dok su

oni njegova djeca. Y_1 nazivamo lijevo dijete, Y_2 desno dijete, a njih dvojica su međusobno braća. Čvor koji nema djece nazivamo listom. Dubina (visina) stabla d je maksimalan broj čvorova na grani stabla.

Pretpostavimo da broj korisnika n zadovoljava nejednakost $2^{d-1} < n \leq 2^d$. Konstruirajmo binarno stablo $\mathcal{T}$ dubine d koje ima točno n listova. Sve razine stabla su popunjene, osim eventualno zadnje, te n listova predstavlja n korisnika. Za svakog korisnika U, neka U ujedno označava i list koji odgovara tome korisniku. Postoji ključ koji je povezan sa svakim čvorom stabla $\mathcal{T}$, odnosno postoje različiti ključevi za listove i za unutarnje čvorove.

Neka je $k(X)$ ključ čvora X. Tada je $k(R)$ grupni ključ, pri čemu sa R označavamo korijen stabla. Svakom korisniku je dano $d+1$ ključeva koji odgovaraju čvorovima koji leže na jedinstvenom putu od korisnika U do korijena stabla R. Stoga svaki korisnik ima $O(\log n)$ ključeva.

Primjerice, pretpostavimo da imamo binarno stablo dubine $d = 4$ sa $n = 16$ korisnika u mreži, čiji su čvorovi označeni $1, 2, \ldots, 2^{d+1} - 1 = 31$. Korisnici se nalaze na posljednjih 16 čvorova, stoga zauzimaju čvorove $16, \ldots, 31$. Grupni ključ je tada $k(1)$, dok korisnik U_{18} ima ključeve $k(1), k(2), k(4), k(9)$ i $k(18)$.



Slika 3: Primjer binarnog stabla sa 16 korisnika i dubinom $d = 4$.

Za potrebu ove sheme, pretpostavimo da su čvorovi u stablu $\mathcal{T}$ raspoređeni tako da zadovoljavaju sljedeća pravila:

1) Za $0 \leq l \leq d - 1$, ukupno 2^l čvorova na dubini l je označeno s $2^l, 2^l + 1, \ldots, 2^{l+1} - 1$, redom.

2) n listova stabla $\mathcal{T}$ dobiva različite oznake iz skupa $\{2^d, 2^d + 1, \ldots, 2^{d+1} - 1\}$.

3) Roditelj čvora j ($j \neq 1$) je čvor $\lfloor \frac{j}{2} \rfloor$.

4) Lijevo dijete čvora j je čvor $2j$, dok je desno dijete čvor $2j + 1$, ukoliko postoje.

5) Brat čvora j ($j \neq 1$) je čvor $j + 1$, ukoliko je j paran broj; $j - 1$ ukoliko je j neparan broj, ukoliko bratski čvor postoji.

Može se provjeriti da su ova svojstva zadovoljena za cijelo stablo prikazano na slici 3, a mi ćemo ovdje radi ilustracije istaknuti promatrana svojstva samo za neke čvorove.

1) Za $0 \leq l \leq 3$ biramo npr. $l = 2$ te na toj dubini imamo $2^2 = 4$ čvora redom označena s $2^2, 2^2 + 1, 2^2 + 2, 2^3 - 1$, odnosno $4, 5, 6, 7$.

2) Ukupno $n = 16$ listova ima oznake $2^4, 2^4+1, 2^4+2, \ldots, 2^5-1$, odnosno $16, 17, 18, \ldots, 31$.

3) Promotrimo roditelja čvora koji predstavlja već spomenutog korisnika U_18. Njegov roditelj je čvor $\lfloor \frac{18}{2} \rfloor = 9$.

4) Lijevo dijete čvora 9 je čvor $2 \cdot 9 = 18$, dok je $2 \cdot 9 + 1 = 19$ desno dijete ovog čvora.

5) Brat čvora 9 je čvor $9 - 1 = 8$, dok je $18 + 1 = 19$ brat čvora 18.

Ključevi, koji odgovaraju čvorovima u $\mathcal{T}$, mogu biti pohranjeni u obliku polja koje ima elemente $K[1], \ldots, K[2^{d+1} - 1]$. Pretpostavimo da želimo opozvati korisnika U i neka $\mathcal{P}(U)$ označava skup čvorova koji se nalaze na jedinstvenom putu od lista U do korijena R. Pri opozivu je nužno promijeniti odgovarajuće ključeve za d čvorova iz skupa $\mathcal{P}(U) \setminus \{U\}$ - neka je $k'(X)$ novi ključ za svaki čvor $X \in \mathcal{P}(U) \setminus \{U\}$. Označimo sa $\boldsymbol{sib}(X)$ bratski čvor čvora X te sa $\boldsymbol{par}(X)$ roditelja čvora X. Pouzdano tijelo emitira sljedećih $2d - 1$ stavki:

1) $e_{k_{\boldsymbol{sib}(U)}}(k'(\boldsymbol{par}(U)))$,

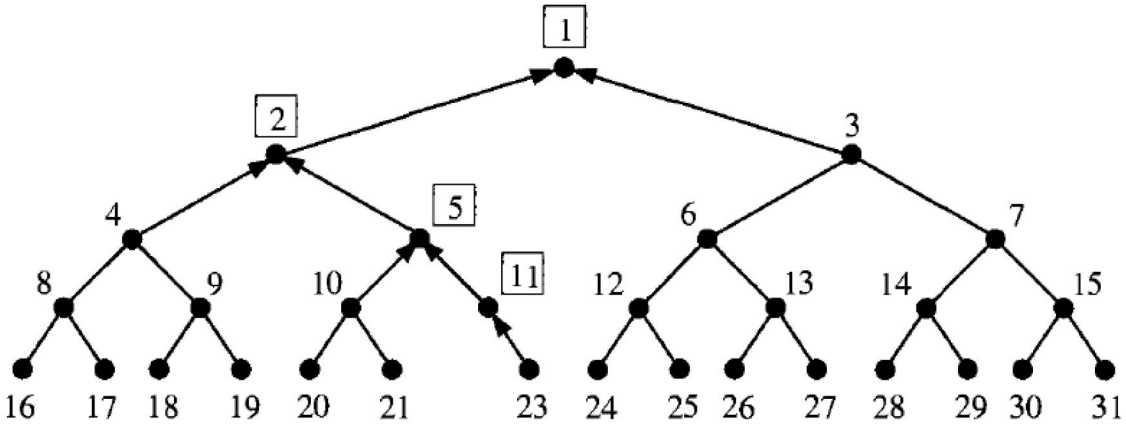
2) $e_{k_{\boldsymbol{sib}(X)}}(k'(\boldsymbol{par}(X)))$ i $e_{k'_{\boldsymbol{sib}(X)}}(k'(\boldsymbol{par}(X)))$, za svaki čvor $X \in \mathcal{P}(U), X \neq U, R$.

Emitiranje dozvoljava bilo kojem neopozvanom korisniku V da ažurira ključeve u čvorovima koji se nalaze u presjeku $\mathcal{P}(U) \cap \mathcal{P}(V)$. Pokažimo na primjeru binarnog stabla sa slike 3 kako funkcionira ova shema.

Primjer 2.5 (vidjeti [3, Example 14.9]). *Pretpostavimo da pouzdano tijelo želi opozvati korisnika $U = 22$. Skup $\mathcal{P}(U)$ tada je jednak $\{22, 11, 5, 2, 1\}$. Pouzdano tijelo stvara nove ključeve za sve čvorove iz tog skupa osim čvora korisnika kojeg ćemo opozvati, odnosno k'_{11}, k'_5, k'_2 i k'_1. Braća čvorova iz skupa $\mathcal{P}(U)$ su čvorovi $\{23, 10, 4, 3\}$, redom. Emitiranje se zbog toga sastoji od:*

$$e_{k(23)}(k'(11)),\ e_{k(10)}(k'(5)),\ e_{k(4)}(k'(2)),\ e_{k(3)}(k'(1)),\ e_{k'(11)}(k'(5)),\ e_{k'(5)}(k'(2)),\ e_{k'(2)}(k'(1)).$$

Razmislimo kako bi korisnik $U = 23$ ažurirao svoj ključ. Najprije može iskoristiti svoj ključ $k(23)$ kako bi dešifrirao $e_{k(23)}(k'(11))$ i tako izračunao $k'(11)$, kojeg zatim koristi za računanje $k'(5)$. Zatim dobiveni $k'(5)$ koristi za računanje $k'(2)$, dok njega u konačnici koristi za dobivanje $k'(1)$. Na slici 4 je prikazano šifriranje novih ključeva smjerom strelica te opoziv korisnika $U = 22$. Korisnici koji primaju nove ključeve označeni su kvadratnim obrubom.



Slika 4: Ažuriranje binarnog stabla pri opozivu korisnika 22.

Svaki korisnik pohranjuje $O(\log n)$ ključeva, pa je i prijenos te veličine. Ove količine podataka su veće nego kod prethodne dvije promatrane sheme, međutim, budući da se LL-ključevi ažuriraju svaki put kada je korisnik opozvan, ne postoji ograničenje broja korisnika koji se mogu opozvati tijekom vremena. Drugim riječima, bilo koji broj korisnika može biti opozvan bez utjecaja na sigurnost sustava.

Moguće je napraviti istovremeni opoziv više od jednog korisnika, ali je nešto kompliciranije. Nove korisnike je moguće dodati u ovu hijerarhiju kad god je broj trenutnih korisnika manji od 2^d, pridružujući ga krajnjem lijevom slobodnom listu stabla. Nakon što broj korisnika premaši 2^d, dodaje se nova razina čvorova, što automatski povećava dubinu stabla za 1 i omogućuje udvostručenje broja korisnika.

3 Zaštita autorskih prava

Zaštita od kršenja autorskih prava važan je i težak izazov u doba u kojem internet igra jednu od glavnih uloga u mnogim sferama života. Digitalni sadržaj dostupan je gotovo svima, lako se može kopirati i prenositi putem računalnih mreža. Sadržaj može biti šifriran prije prijenosa - na primjer, ranije spomenuta enkripcija pomoću BES algoritma štiti šifrirani sadržaj te ga neovlašteni korisnici ne mogu dešifrirati. Međutim, kako bi sadržaj bio razumljiv krajnjem korisniku kojem je namijenjen, mora se prethodno dešifrirati. Nakon što se sadržaj dešifrira, pojavljuje se opasnost od potencijalnog kopiranja i nedozvoljenog širenja. Najjednostavniji primjer s kojim je upoznata većina korisnika internetske mreže je svakako skidanje i distribucija podataka putem torrent poslužitelja, ali i mnogih drugih internetskih stranica za online besplatno gledanje sadržaja.

Potencijalni način zaštite nude algoritmi koji omogućuju praćenje sadržaja do korisnika kojemu je namijenjen. Prije nego što opišemo neke od načina praćenja, navodimo primjer dvaju karakterističnih načina kršenja autorskih prava.

Nezakonita redistribucija sadržaja - Nakon što stigne do ovlaštenog korisnika, šifrirani sadržaj uvijek se nepromjenjivo dešifrira te se tada može kopirati i proslijediti drugima, recimo već spomenutim ilegalnim piratskim emitiranjem.

Nezakonita redistribucija ključeva - Ukoliko je sadržaj šifriran, mora postojati mehanizam koji krajnjem korisniku omogućuje dešifriranje sadržaja. Ključevi korišteni za dešifriranje mogu se kopirati i distribuirati drugim korisnicima. Još jedna od opcija je kombiniranje ključeva kako bi se stvorio novi piratski dekoder, koji se potom može koristiti za ilegalno dešifriranje sadržaja.

3.1 Otisak prsta

Osvrnimo se najprije na redistribuciju nezakonitog sadržaja. Pretpostavimo da svaka kopija nekog digitalnog podatka, u oznaci D, sadrži jedinstveni otisak prsta F. Na primjer, u jednom megabajtu binarnih podataka otisak prsta može se sastojati od 100 "posebnih" bitova "skrivenih" u podacima tako da ih se teško može razotkriti. Ponekad se postupak ugrađivanja skrivenih podataka za identifikaciju naziva vodenim žigom, engl. *watermarking*.

U ovom scenariju, dobavljač (s obzirom da je riječ o otiscima prstiju, razumno je pretpostaviti da se ovdje radi o policijskoj službi) posjeduje bazu podataka koja prati sve različite otiske prstiju, kao i prave vlasnike odgovarajućih kopija podataka D. Bilo koja točna kopija podataka može se upariti s njezinim vlasnikom. Nažalost, ovaj pristup ima i svoje loše strane. Primjerice, ako se otisak prsta može lako prepoznati, tada ga je moguće modificirati ili uništiti, što povlači nemogućnost ulaska u trag podacima. Druga prijetnja je mogućnost prepoznavanja otisaka prstiju ili dijelova otisaka prstiju od strane familije korisnika, čak i ako pojedinačni korisnik to ne može učiniti, te stvaranje nove kopije podataka s uništenim otiskom prsta.

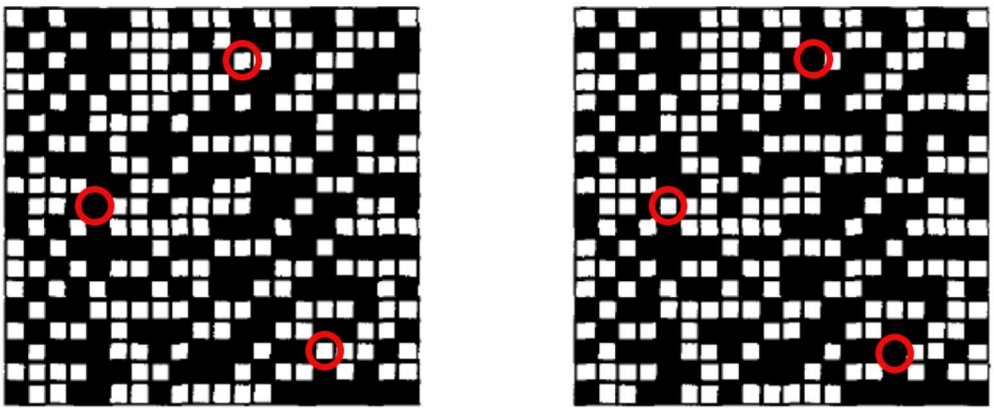
Modelirajmo dosad opisani problem pomoću preciznijeg matematičkog modela. Pretpostavimo da se svaka kopija podataka sastoji od sadržaja C čija je veličina L bitova i

l-bitnog otiska prsta F. Tada su podaci oblika $D = (C, F)$. Svi podaci su reprezentirani nekim fiksnom alfabetom, primjerice binarni podaci koriste alfabet $\{0, 1\}$. Pretpostavit ćemo da sve kopije podataka imaju isti sadržaj, ali različite otiske prstiju, pa imamo $D_1 = (C, F_1)$, $D_2 = (C, F_2)$ itd.

Za označavanje položaja na kojem se pojavljuje otisak prsta koristit ćemo termin "bitovi otiska prsta". Sama riječ "bitovi" upućuje na to da su podaci binarne forme, no ovaj termin koristit ćemo i u slučajevima kada podaci nisu definirani nad binarnim alfabetom. Stoga, pretpostavimo da se bitovi otiska prsta $b_{i_1}, \ldots, b_{i_l}$ pojavljuju uvijek na istom tajnom položaju u svim kopijama podataka. Ovaj problem se obično proučava pod *pretpostavkom označavanja* (engl. *marking assumption*), koju navodimo u nastavku:

Neka je dan određeni broj kopija podataka, primjerice $D_1, D_2, \ldots, D_w$. Jedini bitovi koje familija korisnika može detektirati kao bitove otiska prsta su bitovi b za koje vrijedi $D_i[b] \neq D_j[b]$, za neke i, j.

Drugim riječima, pretpostavljamo da su otisci prstiju skriveni dovoljno dobro da familija korisnika ne može identificirati nijedan određeni bit kao bit otiska prsta, osim u slučaju da posjeduje dvije kopije podataka u kojima dotični bit ima različite vrijednosti. Slika 5 ilustrira ideju koja stoji iza pretpostavke označavanja. Ovaj dijagram sadrži dvije mreže sastavljene od crnih i bijelih kvadrata, pri čemu je svaki kvadrat jedan piksel. Može se provjeriti da postoje točno tri piksela (označena crvenim krugovima) u kojima se dvije mreže razlikuju. Prema pretpostavci označavanja, samo se ova tri piksela mogu prepoznati kao bitovi otiska prsta.



Slika 5: Primjer označavanja crnom i bijelom bojom

Uzimajući u obzir da vrijedi pretpostavka označavanja, promotrimo vrste napada koje familija korisnika može izvesti. Pretpostavka označavanja implicira da je stvarni sadržaj irelevantan, a problem se svodi na proučavanje kombinatornih svojstava skupa otisaka prstiju. Kao što je prethodno opisano, pretpostavimo da je dano w kopija podataka i da se

neki bitovi mogu identificirati kao bitovi otiska prsta. Tada je moguće konstruirati novu "piratsku" kopiju podataka, postavljanjem vrijednosti identificiranih bitova otiska prsta iz jedne od kopija podataka na proizvoljan način. Dobivamo novi podatak $D' = (C, F')$, gdje je F' novostvoreni hibridni otisak prsta. Glavno je pitanje može li se hibridni otisak pratiti ako su otisci prsta konstruirani na odgovarajući način.
Radi potpunosti, u nastavku dajemo preciznu definiciju hibridnog otiska prsta.

Definicija 3.1. (l, n, q)-*kod* je podskup $\mathcal{C} \subseteq \mathcal{Q}^l$ takav da je $|\mathcal{Q}| = q$ i $|\mathcal{C}| = n$. Ovo povlači da imamo n *kodnih riječi*, svaka od kojih je uređena l-torka elemenata iz alfabeta $\mathcal{Q}$. Kodna riječ isto je što i otisak prsta. Neka je $\mathcal{C}_0 \subseteq \mathcal{C}$. Definiramo $\boldsymbol{desc}(\mathcal{C}_0)$ tako da sadrži sve uređene $l-$torke $\boldsymbol{f} = (f_1, \ldots, f_l)$ takve da za svaki $i = 1, \ldots, l$ postoji neki $(c_1, \ldots, c_l) \in \mathcal{C}_0$ za koji je $f_i = c_i$. Skup $\boldsymbol{desc}(\mathcal{C}_0)$ sastoji se od svih hibridnih otisaka prstiju koji se mogu konstruirati koristeći otiske iz skupa $\mathcal{C}_0$ i to zovemo *kod potomak* od $\mathcal{C}_0$ (engl. *descendant code*). Za svaki $\boldsymbol{c} \in \mathcal{C}_0$ i za svaki $\boldsymbol{f} \in \boldsymbol{desc}(\mathcal{C}_0)$, kažemo da je $\boldsymbol{c}$ *roditelj* od $\boldsymbol{f}$ u $\boldsymbol{desc}(\mathcal{C}_0)$.

Promotrimo prethodno opisano na jednom kratkom primjeru.

Primjer 3.1. *Neka je* $\mathcal{C}_0 = \{(1,1,2), (2,3,2)\}$. *Vidimo da u kodu potomku prva koordinata može biti* 1 *ili* 2, *druga koordinata može biti* 1 *ili* 3, *dok je treća koordinata* 2. *Slijedi da je*

$$\boldsymbol{desc}(\{(1,1,2), (2,3,2)\}) = \{(1,1,2), (1,3,2), (2,1,2), (2,3,2)\}.$$

U ovom slučaju, $\boldsymbol{desc}(\mathcal{C}_0)$ *sastoji se od dvije originalne kodne riječni i dva nova hibridna otiska prstiju.*

Za bilo koji $w \geq 2$, kod w-potomak od $\mathcal{C}$, u oznaci $\boldsymbol{desc}_w(\mathcal{C}_0)$, sastoji se od skupa l-torki danog u nastavku:

$$\boldsymbol{desc}_w(\mathcal{C}) = \bigcup_{\mathcal{C}_0 \subseteq \mathcal{C},\, |\mathcal{C}_0| \leq w} \boldsymbol{desc}_w(\mathcal{C}_0).$$

Kod w-potomak sastoji se od svih hibridnih otisaka prstiju koje može napraviti familija korisnika veličine najviše w.

3.2 Identificiranje familije korisnika

U ovom potpoglavlju zanima nas obrnuti postupak u odnosnu na prethodno potpoglavlje, odnosno pokušavamo identificirati familiju korisnika koja je konstruirala hibridni otisak prsta (engl. *Identifiable parent property - IPP*).

Pretpostavimo da je $\boldsymbol{f} \in \boldsymbol{desc}_w(\mathcal{C})$. Skup sumnjivih familija korisnika za $\boldsymbol{f}$ definiramo na sljedeći način:

$$\boldsymbol{susp}_w(\boldsymbol{f}) = \{\mathcal{C}_0 \subseteq C \colon |\mathcal{C}_0| \leq w,\ \boldsymbol{f} \in \boldsymbol{desc}(\mathcal{C})\}.$$

Ovako definiran skup $\boldsymbol{susp}_w(\boldsymbol{f})$ sastoji se od svih familija korisnika veličine najviše w koje su mogle kreirati hibridni otisak prsta $\boldsymbol{f}$. U najboljem slučaju, ovaj skup sadržavao bi samo jednu familiju korisnika te bismo u tom slučaju imali neke dokaze da je ovaj podskup zaista kreirao hibridni otisak prsta.

U slučaju da se $\boldsymbol{susp}_w(\boldsymbol{f})$ sastoji od više od jednog skupa, još uvijek možemo izvući neke korisne informacije promatrajući skupove u $\boldsymbol{susp}_w(\boldsymbol{f})$. Recimo, pretpostavimo da postoji kodna riječ $\boldsymbol{c} \in \mathcal{C}$ takva da je $\boldsymbol{c} \in \mathcal{C}_0$, za sve $\mathcal{C}_0 \in \boldsymbol{susp}_w(\boldsymbol{f})$. Svaka takva kodna riječ može se identificirati kao kriva (pod pretpostavkom da je familija veličine najviše w), čak i ako nismo

u mogućnosti identificirati čitavi podskup krivih kodnih riječi. Ovo svojstvo matematički zapisujemo kao:

$$\bigcap_{\mathcal{C}_0 \in \boldsymbol{susp}_w(\boldsymbol{f})} \mathcal{C}_0 \neq \emptyset. \tag{4}$$

Kažemo da je $\mathcal{C}_0$ *w-IPP kod* ukoliko (4) vrijedi za svaki $\boldsymbol{f} \in \boldsymbol{desc}_w(\mathcal{C})$. Osim toga, ako u w-IPP kodu vrijedi

$$\boldsymbol{c} \in \bigcap_{\mathcal{C}_0 \in \boldsymbol{susp}_w(\boldsymbol{f})} \mathcal{C}_0,$$

onda je $\boldsymbol{c}$ *identificirani roditelj* od $\boldsymbol{f}$.

Pogledajmo kako to izgleda na primjerima.

Primjer 3.2 (vidjeti [3, Example 14.10]). *Neka je $(3, 6, 3)$-kod, što znači da imamo ukupno 6 kodnih riječi ($n = 3$) , od kojih je svaka uređena trojka ($l = 3$) iz alfabeta $\{0, 1, 2\}$ ($q = 3$). Uzmimo u obzir familiju veličine najviše $w = 2$:*

$$\mathcal{C} = \{\boldsymbol{c}_1, \boldsymbol{c}_2, \boldsymbol{c}_3, \boldsymbol{c}_4, \boldsymbol{c}_5, \boldsymbol{c}_6\} = \{(0,1,1), (1,0,1), (1,1,0), (2,0,2), (1,0,2), (2,1,0)\}.$$

Tada je

$$\begin{aligned}\boldsymbol{desc}_2(\mathcal{C}) =& \{(0,0,0), (0,0,1), (0,0,2), (0,1,0), (0,1,1), (0,1,2), (1,0,0), (1,1,1), (1,1,2),\\ & (1,0,2), (1,1,0), (1,0,1), (2,0,0), (2,0,1), (2,0,2), (2,1,0), (2,1,1), (2,1,2)\}.\end{aligned}$$

Neka je $\boldsymbol{f}_1 = (1,1,1)$ hibridni otisak prsta. Skup sumnjivih familija korisnika koji su mogli kreirati $\boldsymbol{f}_1$ jednak je

$$\boldsymbol{susp}_2(\boldsymbol{f}_1) = \{\{\boldsymbol{c}_1, \boldsymbol{c}_2\}, \{\boldsymbol{c}_1, \boldsymbol{c}_3\}, \{\boldsymbol{c}_2, \boldsymbol{c}_3\}, \{\boldsymbol{c}_1, \boldsymbol{c}_5\}, \{\boldsymbol{c}_2, \boldsymbol{c}_6\}\}.$$

Primjerice, kombinirajući kodne riječi $\boldsymbol{c}_1$ i $\boldsymbol{c}_2$ možemo doći do $\boldsymbol{f}_1 = (1,1,1)$ itd. Navedeni hibridni otisak prsta $\boldsymbol{f}_1$ ne zadovoljava svojstvo (4) pa kod nije 2-IPP kod.
Sada promotrimo $\boldsymbol{f}_2 = (0,1,2)$. U ovom slučaju je

$$\boldsymbol{susp}_2(\boldsymbol{f}_2) = \{\{\boldsymbol{c}_1, \boldsymbol{c}_4\}, \{\boldsymbol{c}_1, \boldsymbol{c}_5\}\}.$$

S obzirom da $\boldsymbol{f}_2$ zadovoljava (4) jer se u presjeku nalazi $\boldsymbol{c}_1$, on je ujedno i identificirani roditelj od $\boldsymbol{f}_2$, pod pretpostavkom da je familija veličine najviše 2 kreirala $\boldsymbol{f}_2$.

Primjer 3.3 (vidjeti [3, Example 14.11]). *Neka je $(3, 7, 5)$ $2-$IPP kod dan u nastavku:*

$$\mathcal{C} = \{\boldsymbol{c}_1, \boldsymbol{c}_2, \boldsymbol{c}_3, \boldsymbol{c}_4, \boldsymbol{c}_5, \boldsymbol{c}_6, \boldsymbol{c}_7\} = \{(0,0,0), (0,1,1), (0,2,2), (1,0,3), (2,0,4), (3,3,0), (4,4,0)\}.$$

Pretpostavimo da je $\boldsymbol{f} = (f_1, f_2, f_3)$ hibridni otisak prsta kreiran od strane familije korisnika veličine $w = 2$. Ukoliko je bilo koja od koordinata u $\boldsymbol{f}$ različita od 0, tada je moguće identificirati barem jednog roditelja od $\boldsymbol{f}$:

$$\begin{array}{llll} f_1 = 1 \implies \boldsymbol{c}_4; & f_1 = 2 \implies \boldsymbol{c}_5; & f_1 = 3 \implies \boldsymbol{c}_6; & f_1 = 4 \implies \boldsymbol{c}_7 \\ f_2 = 1 \implies \boldsymbol{c}_2; & f_2 = 2 \implies \boldsymbol{c}_3; & f_2 = 3 \implies \boldsymbol{c}_6; & f_2 = 4 \implies \boldsymbol{c}_7 \\ f_3 = 1 \implies \boldsymbol{c}_2; & f_3 = 2 \implies \boldsymbol{c}_3; & f_3 = 3 \implies \boldsymbol{c}_4; & f_3 = 4 \implies \boldsymbol{c}_5. \end{array}$$

U konačnici, ako je $\boldsymbol{f} = (0,0,0)$, tada je $\boldsymbol{c}_1$ identificirani roditelj.

Više o samoj konstrukciji 2-IPP kodova i efikasnim algoritmima za identifikaciju roditelja pomoću hash funkcija zainteresirani čitatelj može pronaći u [3].

4 Praćenje nezakonito preraspodjeljenih ključeva

Pretpostavimo da je svaki korisnik u mreži dobio "dekodersku kutiju" koja omogućuje dešifriranje šifriranih prijenosa. Dakle, imamo shemu šifriranja emitiranja u kojoj svaki korisnik može dešifrirati emitiranje. Općenito, svaka kutija dekodera sadrži različitu kolekciju ključeva. U tom slučaju, familija w korisnika kojima prijenos nije namijenjen može stvoriti piratski dekoder kombinirajući ključeve iz svojih kutija dekodera te će svaki takav piratski dekoder moći dešifrirati prijenos.

Skup ključeva u svakoj kutiji dekodera može se smatrati kodnom riječi u određenom kodu, a ključevi u piratskom dekoderu mogu se smatrati kodnom riječi u w-potomku koda. Ako se kod može pratiti (primjerice, ako zadovoljava svojstvo w-IPP), tada se piratski dekoder može pratiti do barem jednog člana familije korisnika koja ga je stvorila. Dakle, ako je piratski dekoder zaplijenjen, onda se može utvrditi barem jedan od krivaca.

Promotrimo shemu šifriranja prijenosa koju ćemo koristiti. Ova shema je jednostavnija verzija BES sheme opisane u potpoglavlju 2.1 jer će bilo koji dovoljno velik podskup odgovarajućih ključeva moći dešifrirati prijenos.
Pouzdano tijelo bira l skupova ključeva, u oznaci $\mathcal{K}_1, \ldots, \mathcal{K}_l$, pri čemu se svaki skup ključeva sastoji od q ključeva odabranih iz $\mathbb{Z}_m$, pri čemu je m prost broj, te pišemo

$$\mathcal{K}_i = \{k_{i,j} \colon 1 \leq i \leq l,\, 1 \leq j \leq q\}.$$

Dekoderska kutija sadrži l ključeva, po jedan iz svakog skupa $\mathcal{K}_i$. Tajni ključ $K \in \mathbb{Z}_m$, koji je korišten za šifriranje sadržaja prijenosa $\mathcal{M}$, podijeljen je na l dijelova koristeći (l, l)-graničnu shemu. U ovoj shemi pouzdano tijelo na slučajan način nezavisno bira $l-1$ elemenata iz $\mathbb{Z}_m$ u oznaci $s_1, \ldots, s_{l-1}$. Nakon toga računa

$$s_l = K - \sum_{i=1}^{l-1} s_i \bmod m.$$

Zatim pouzdano tijelo distribuira korisnicima dijelove ključa s_i, $1 \leq i \leq l$. Dakle, vrijedi

$$\sum_{i=1}^{l} s_i = K \bmod m.$$

Ključ K koristi se za šifriranje poruke $\mathcal{M}$, a svaki $k_{i,j}$ koristi se za šifriranje s_i, $1 \leq i \leq l$. Prijenos se tada sastoji od šifrata $y = e_K(\mathcal{M})$ te skupa šifriranih ključeva

$$\{e_{k_{i,j}}(s_i) \colon 1 \leq i \leq l,\, 1 \leq j \leq q\}.$$

Neka je $\mathcal{C}$ skup svih kodnih riječi koje odgovaraju svim dekoderskim kutijama unutar sheme. Ključevi u piratskom dekoderu daju kodnu riječ koja se nalazi u w-potomku koda, $\boldsymbol{desc}_w(\mathcal{C})$. Postoji posebna klasa w-IPP kodova koji imaju vrlo efikasne algoritme namijenjene praćenju, a temelje se na ideji "dekodiranja najbližeg susjeda".

Neka $\boldsymbol{dist}(\boldsymbol{c}, \boldsymbol{d})$ označava Hammingovu udaljenost između dva vektora $\boldsymbol{c}, \boldsymbol{d} \in Q^2$, pišemo

$$\boldsymbol{dist}(\boldsymbol{c}, \boldsymbol{d}) = |\{i \colon \boldsymbol{c}_i \neq \boldsymbol{d}_i\}].$$

Drugim riječima, Hammingova udaljenost jednaka je broju pozicija u kojima su elementi dvaju vektora različiti. Za $\boldsymbol{f} \in \boldsymbol{desc}_w(\mathcal{C})$, *najbliži susjed* od $\boldsymbol{f}$ je bilo koja kodna riječ $\boldsymbol{c} \in \mathcal{C}$

takva da je udaljenost $\boldsymbol{dist}(\boldsymbol{f}, \boldsymbol{c})$ najmanja moguća. Označit ćemo bilo kojeg najbližeg susjeda (engl. *nearest neighbour*) od $\boldsymbol{f}$ s $\boldsymbol{nn}(\boldsymbol{f})$ i imati na umu da najbliži susjed ne mora biti jedinstven. Računanje najbližeg susjeda od $\boldsymbol{f}$ naziva se *dekodiranje najbližeg susjeda.*

Za kod $\mathcal{C}$ kažemo da je *w-TA kod* (engl. *trusted authority*, u ovom radu: pouzdano tijelo) ako sljedeće svojstvo vrijedi za sve $\boldsymbol{f} \in \boldsymbol{desc}_w(\mathcal{C})$:

$$\boldsymbol{nn}(\boldsymbol{f}) \in \bigcap_{\mathcal{C}_0 \in \boldsymbol{susp}_w(\boldsymbol{f})} \mathcal{C}_0.$$

Konkretno, w-TA kod je w-IPP kod u kojemu dekodiranje najbližeg susjeda uvijek daje roditelja koji se može identificirati. U svrhu ilustracije navedimo sljedeći primjer.

Primjer 4.1 (vidjeti [3, Example 14.14])**.** *Neka je dan sljedeći* $(5, 16, 4)$*-kod:*

$$\begin{aligned}\mathcal{C} =&\{\boldsymbol{c}_1, \boldsymbol{c}_2, \boldsymbol{c}_3, \boldsymbol{c}_4, \boldsymbol{c}_5, \boldsymbol{c}_6, \boldsymbol{c}_7, \boldsymbol{c}_8, \boldsymbol{c}_9, \boldsymbol{c}_{10}, \boldsymbol{c}_{11}, \boldsymbol{c}_{12}, \boldsymbol{c}_{13}, \boldsymbol{c}_{14}, \boldsymbol{c}_{15}, \boldsymbol{c}_{16}\}\\ =&\{(1,1,1,1,1), (1,2,2,2,2), (1,3,3,3,3), (1,4,4,4,4), (2,1,2,3,4), (2,2,1,4,3),\\ &(2,3,4,1,2), (2,4,3,2,1), (3,1,4,2,3), (3,2,3,1,4), (3,3,2,4,1), (3,4,1,3,2),\\ &(4,1,3,4,2), (4,2,4,3,1), (4,3,1,2,4), (4,4,2,1,3)\}.\end{aligned}$$

Može se pokazati da je navedeni kod 2-TA kod, što povlači da pronalaskom najbližeg susjeda možemo identificirati kodove roditelje. Primjerice, vektor $\boldsymbol{f} = (2,3,2,4,4)$ *nalazi se u 2-potomku od* $\mathcal{C}$*. Ukoliko pogledamo Hammingovu udaljenost vektora* $\boldsymbol{f}$ *i svake pojedine kodne riječi iz* $\mathcal{C}$*, vidimo da je*

$$\boldsymbol{dist}(\boldsymbol{f}, \boldsymbol{c}_5) = \boldsymbol{dist}(\boldsymbol{f}, \boldsymbol{c}_{11}) = 2, \qquad \boldsymbol{dist}(\boldsymbol{f}, \boldsymbol{c}_i) \geq 3,\ i \neq 5, 11.$$

Dakle, kodne riječi $\boldsymbol{c}_5$ *i* $\boldsymbol{c}_{11}$ *su identificirani roditelji od* $\boldsymbol{f}$*.*

Jedan od dovoljnih uvjeta da bi kod bio w-TA kod je da ima veliku minimalnu udaljenost između različitih kodnih riječi. Stoga definiramo

$$\boldsymbol{dist}(\mathcal{C}) = \min\{\boldsymbol{dist}(\boldsymbol{c}, \boldsymbol{d}) \colon \boldsymbol{c}, \boldsymbol{d} \in \mathcal{C}, \boldsymbol{c} \neq \boldsymbol{d}\}.$$

Sljedeći teorem daje nam korisnu karakterizaciju TA-kodova povezanu s netom definiranom Hammingovom udaljenosti koda $\mathcal{C}$.

Teorem 4.1 (vidjeti [3, Theorem 14.6])**.** *Pretpostavimo da je* $\mathcal{C}$ (l, n, q)*-kod za koji vrijedi*

$$\boldsymbol{dist}(\mathcal{C}) > l\Big(1 - \frac{1}{w^2}\Big).$$

Tada je $\mathcal{C}$ *w-TA kod.*

Dokaz. Označimo s $d = \boldsymbol{dist}(\mathcal{C})$. Za bilo koja dva vektora $\boldsymbol{c}, \boldsymbol{d}$ definiramo

$$\boldsymbol{match}(\boldsymbol{c}, \boldsymbol{d}) = l - \boldsymbol{dist}(\boldsymbol{c}, \boldsymbol{d}).$$

Pretpostavimo da je $\boldsymbol{c} = \boldsymbol{nn}(\boldsymbol{f})$ te da je $\mathcal{C}_0 \in \boldsymbol{susp}_w(\boldsymbol{f})$. Treba pokazati da je $\boldsymbol{c} \in \mathcal{C}_0$. Kako je $\boldsymbol{f} \in \boldsymbol{desc}(\mathcal{C}_0)$, vrijedi

$$\sum_{\boldsymbol{c}' \in \mathcal{C}_0} \boldsymbol{match}(\boldsymbol{f}, \boldsymbol{c}') \geq l.$$

Iz $|\mathcal{C}_0| \leq w$ slijedi da postoji kodna riječ $\boldsymbol{c'} \in \mathcal{C}_0$ takva da je

$$\boldsymbol{match}(\boldsymbol{f}, \boldsymbol{c'}) \leq \frac{l}{w},$$

a s obzirom da je $\boldsymbol{c}$ najbliži susjed od $\boldsymbol{f}$, vrijedi

$$\boldsymbol{match}(\boldsymbol{f}, \boldsymbol{c}) \leq \frac{l}{w}.$$

Neka je sada $\boldsymbol{b} \in \mathcal{C} \setminus \mathcal{C}_0$. Jer je $\boldsymbol{f} \in \boldsymbol{desc}(\mathcal{C}_0)$, imamo

$$\boldsymbol{match}(\boldsymbol{f}, \boldsymbol{b}) \leq \sum_{\boldsymbol{c'} \in \mathcal{C}_0} \boldsymbol{match}(\boldsymbol{c'}, \boldsymbol{b}) \leq w(l-d).$$

Iz pretpostavke teorema $d > l(1 - \frac{1}{w^2})$ dobivamo $w(l-d) < \frac{l}{w}$, pa je

$$\boldsymbol{match}(\boldsymbol{f}, \boldsymbol{b}) < \boldsymbol{match}(\boldsymbol{f}, \boldsymbol{c}),$$

za sve kodne riječi $\boldsymbol{b} \notin \mathcal{C}_0$. Dakle $\boldsymbol{c} \in \mathcal{C}_0$ te je dani kod w-TA kod. □

U nastavku opisujemo način konstrukcije inačice w-TA kodova, poznatijih pod imenom Reed-Solomon kodovi.

Pretpostavimo da je $t < q$, pri čemu je q prost broj. Neka se skup $\mathcal{P}(q,t)$ sastoji od svih polinoma $a(x) \in \mathbb{Z}_q[x]$ stupnja najviše $t-1$. Za prirodan broj $l < q$ definiramo

$$\mathcal{C}(q,l,t) = \{(a(0), a(1), \ldots, a(l-1)) \colon a(x) \in \mathcal{P}(q,t)\}.$$

Tvrdimo da je $\mathcal{C} = \mathcal{C}(q,l,t)$ (l, q^t, q)-kod takav da je $\boldsymbol{dist}(\mathcal{C}) = l - t + 1$. Uočimo da se bilo koja dva različita polinoma stupnja najviše $t-1$ mogu podudarati u najviše $t-1$ točaka.

Pretpostavimo da je

$$t = \left\lceil \frac{l}{w^2} \right\rceil,$$

što prema definiciji funkcije "strop" povlači da je

$$t < \frac{l}{w^2} + 1.$$

Stoga je

$$\boldsymbol{dist}(\mathcal{C}) = l - t + 1 < l - \left(\frac{l}{w^2} + 1\right) - 1 = l - \frac{l}{w^2} - 1 + 1 = l - \frac{l}{w^2} = l\left(1 - \frac{1}{w^2}\right),$$

odnosno

$$\boldsymbol{dist}(\mathcal{C}) < l\left(1 - \frac{1}{w^2}\right).$$

Prema teoremu 4.1, $\mathcal{C}$ je w-TA kod za koji je $n = q^t = q^{\left\lceil \frac{l}{w^2} \right\rceil}$. Prethodnim postupkom konstrukcije dokazali smo tvrdnju koju možemo sažeti u sljedeći teorem:

Teorem 4.2 (vidjeti [3, Theorem 14.7]). *Neka je q prost broj, $l \leq q$ i $w \geq 2$ prirodan broj. Tada postoji $(l, q^{\left\lceil \frac{l}{w^2} \right\rceil}, q)$-kod koji je w-TA kod.*

Literatura

[1] D. Brajković, *Algebra kroz primjere, priručnik za vježbe*, Sveučilište Josipa Jurja Strossmayera - Odjel za matematiku, Osijek, 2018.

[2] S. Lipschutz, M.L. Lipson, *Discrete Mathematics (3ed)*, McGraw Hill Professional, 2007.

[3] D. Stinson, *Cryptography Theory and Practice (3ed)*, Chapman and Hall/CRC, Boca Raton, 2006.

[4] D. Veljan, *Kombinatorna i diskretna matematika*, Algoritam, Zagreb, 2001.

[5] Struna - Hrvatsko strukovno nazivlje, dostupno na: \http://struna.ihjj.hr/naziv/visesmjerno-odasiljanje/1424/#naziv

Sažetak

U ovom radu upoznali smo se s pojmom multicast te predstavili neke od načina kako osigurati sigurnost prilikom višesmjernog odašiljanja. Razmotrili smo BES algoritam i generalizirani BES algoritam kao potencijalne načine osiguravanja sigurnosti enkriptiranjem odašiljanja. Kao drugu opciju zaštite naveli smo proces ponovnog generiranja ključa u vidu kojeg smo opisali Blacklisting shemu, Naor-Pinkas shemu te shemu hijerarhije ključeva. U drugom dijelu rada naglasak je stavljen na zaštitu autorskih prava i praćenje ilegalno distribuiranog sadržaja kroz problem otiska prsta te identificiranje familije korisnika koja pokušava ilegalno doći do sadržaja. Rad smo potkrijepili konkretnim primjerima provedbe navedenih algoritama i shema.

Ključne riječi

multicast, BES algoritam, generalizirani BES algoritam, ponovno generiranje ključa, LL-ključ, Blacklisting shema, Naor-Pinkas shema, hijerarhija ključeva, otisak prsta, IPP-kod

Multicast security and copyright protection

Summary

In this paper, we were introduced to the term multicast and presented some of the ways to ensure security during multicast transmission. We considered the BES algorithm and the generalized BES algorithm as potential ways to ensure security by encrypting transmissions. As another protection option, we have listed the process of re-generating the key in the form of which we have described the Blacklisting scheme, the Naor-Pinkas scheme and the key hierarchy scheme. In the second part of the paper, the emphasis is on the copyrights protection and the tracing of illegally distributed content through the problem of fingerprinting and identifying the family of users who are trying to access the content illegally. We supported the work with concrete examples of application of the mentioned algorithms and schemes.

Keywords

multicast, BES algorithm, generalised BES algorithm, re-keying, LL-keys, Blacklisting scheme, Naor-Pinkas scheme, key hierarchy, fingerprinting, IPP-code

Životopis

Rođena sam 28. kolovoza 1998. godine u Slavonskom Brodu. Osnovnu školu Ivana Meštrovića u Vrpolju završila sam 2013. godine te sam iste godine upisala općи smjer u Gimnaziji A.G. Matoša u Đakovu. Tijekom osnovnoškolskog i srednjoškolskog obrazovanja sudjelovala sam na više županijskih natjecanja iz hrvatskog jezika, tehničke kulture, geografije i fizike. Prijediplomski studij matematike na bivšem Odjelu za matematiku upisala sam 2017. godine, a završila 2020. godine završnim radom "Prebrojivost skupova" pod mentorstvom izv. prof. dr. sc. Dragane Jankov-Maširević, na temelju kojeg je kasnije nastao stručni članak "Ekvipotentnost skupova: kako smjestiti beskonačno mnogo osoba u već pun Hilbertov hotel". Diplomski studij matematike, smjer Financijska matematika i statistika upisala sam 2020. godine. U akademskoj godini 2020./2021. postala sam članica, a u sljedećoj akademskoj godini predsjednica Studentskog zbora Odjela za matematiku te sam u godini 2022./2023. dobila Pohvalu za izvannastavne aktivnosti. U srpnju i rujnu 2022. godine odradila sam studentsku praksu u Hrvatskoj agenciji za poljoprivredu i hranu u Osijeku, gdje sam zajedno s kolegama razvila program za procjenu izloženosti potrošača raznim kontaminantima u hrani. Za vrijeme studiranja radila sam kao online instruktor na platformi Einstrukcije gdje sam podučavala matematiku i statistiku.