Homomorfni kriptosustavi

Zrilić, Tanja

SVEUČILIŠTE U ZAGREBU

PRIRODOSLOVNO–MATEMATIČKI FAKULTET

MATEMATIČKI ODSJEK

Tanja Zrilić

HOMOMORFNI KRIPTOSUSTAVI

Diplomski rad

Voditelj rada:
akademik Andrej Dujella

Zagreb, srpanj, 2018.

Ovaj diplomski rad obranjen je dana _____ pred ispitnim povjerenstvom u sastavu:

1. _____, predsjednik
2. _____, član
3. _____, član

Povjerenstvo je rad ocijenilo ocjenom _____.

Potpisi članova povjerenstva:

1. _____
2. _____
3. _____

Mojim roditeljima

Sadržaj

Uvod

Razvojem društva i napredovanjem tehnologije čovjek postaje sve više svjestan danih mogućnosti te iz dana u dan radi na novim pokušajima unaprjeđenja kako bi olakšao život u današnjem vremenu. Najvažniju ulogu u tome ima razmjena informacija i njezina sigurnost. Enkripcijom je omogućeno dvjema osobama komuniciranje preko nesigurnog komunikacijskog kanala na način da treća osoba, koja može nadzirati komunikacijski kanal, ne može razumjeti njihove poruke. U prošlosti to se očitovalo u pisanju šifriranih poruka dok danas u vremenu digitalne komunikacije primjena kriptografije je neizbježna. Informacijski sustavi nikad nisu u potpunosti sigurni, a izrazito je bitno sačuvati sigurnost informacija prilikom razmjene. Jedan od trendova koji raste posljednjih godina je računarstvo u oblacima (*eng. cloud computing*). Riječ je o preseljenju dijela usluga u oblake, to jest na servere koji se nalaze negdje u svijetu, a podacima možete pristupiti s bilo kojeg uređaja koji ima pristup Internetu. Najveća prepreka sve češćeg korištenja ovakvih sustava je mogućnost zloupotrebe ili krađe pohranjenih podataka. Kako bi se to izbjeglo podatke je potrebno zaštiti na način da ih se šifrira. Glavni problem koji se javlja kod skladištenja šifriranih podataka je nemogućnost njihove obrade unutar oblaka bez prethodnog dešifriranja. Postavlja se pitanje je li moguće obradu podataka prepustiti nekom drugom, a da mu pri tome ne dozvolimo pristup tim istim podacima? Problem su prvi put predstavili Ronald L. Rivest, Len Adleman i Michael L. Dertouzos još 1978. godine, ali do prvog potpunog rješenja, 30 godina kasnije, došao je Craig Gentry. Rješenje ovog problema leži u primjeni homomorfne enkripcije, koju neki autori još nazivaju i svetim gralom kriptografije. Ona nam omogućava obradu nad šifriranim podacima bez potrebe da se podaci prije toga dešifriraju, što znači da je rezultat također šifriran i može ga dešifrirati samo onaj tko posjeduje ključ za dešifriranje.

U prvom poglavlju ovog rada definirat ćemo homomorfizam i homomorfne kriptosustave pritom pazeći da prethodno definiramo sve pojmove koji će nam biti potrebni za spomenute definicije. Korištena literatura u ovom poglavlju je X. Yi, R. Paulet, E. Bertino, Homomorphic Encryption and Applications [10], članak V. Malidžan, Homomorfna enkripcija [5] i skripta B. Širola, Algebarske strukture [9].

U ovom radu cilj nam je obraditi djelomične i potpune homomorfne kriptosustave te precizno objasniti njihove razlike. U drugom poglavlju opisat ćemo što su to djelomični

homomorfni kriptosustavi te detaljno obraditi RSA, ElGamalov i Paillierov kriptosustav. Za svaki od spomenutih kriptosustava objasnit ćemo na koje načine generiramo ključeve, šifriramo i dešifriramo poruke. Također ćemo pokazati jesu li zadovoljena svojstva homomorfizma i kako se može narušiti sigurnost ovih kriptosustava. Na kraju ćemo navesti i primjere u kojima ćemo pokazati primjenu svega prethodno definiranog. Literatura korištena u ovom poglavlju je X. Yi, R. Paulet, E. Bertino, Homomorphic Encryption and Applications, A. Dujella, M. Maretić, Kriptografija [2] i doktorski rad D.Savić, Jedna klasa sustava zaštite u računarskom oblaku zasnovana na homomorfnim šiframa [8].

Treće poglavlje opisuje potpune homomorfne kriptosustave. Na početku ćemo definirati sve pojmove potrebne da bi iskazali što je to potpuni homomorfni kriptosustav. U kratkom povijesnom pregledu opisat ćemo razvoj potpunih homomorfnih kriptosustava kroz godine do danas. Na samom kraju definirat ćemo i objasniti jednostavnije donekle potpune homomorfne kriptosustave koji će nam služiti kao osnova za konačnu implementaciju potpuno homomorfnog kriptosustava koji je također opisan. Literatura korištena u ovom poglavlju je X. Yi, R. Paulet, E. Bertino, Homomorphic Encryption and Applications, Y.Lindell (Ed.), Tutorials on the Foundations of Cryptography [4], članci M. van Dijk,C. Gentry, V. Halevi i V. Vaikuntanathan, Fully homomorphic encryption over the integers[1], P.S Pisa, M. Abdalla i O.C.M.B Duarte, Somewhat homomorphic encryption scheme for arithmetic operations on large integers [7] i skripta B. Širola, Algebarske strukture.

Ovaj diplomski rad napravljen je u sklopu aktivnosti Projekta KK.01.1.1.01.0004 - Znanstveni centar izvrsnosti za kvantne i kompleksne sustave te reprezentacije Liejevih algebri.

Poglavlje 1

Homomorfizam

1.1 Definicija homomorfizma

Grupa, kao algebarska struktura, je osnovni pojam u matematici. Grupe se pojavljuju u analizi, u algebri, u teoriji brojeva, u algebarskoj geometriji i u mnogim drugim granama matematike.

Definicija 1.1.1. *Neprazan skup $G = (G, \circ)$, gdje je $\circ : G \times G \to G$ binarna operacija, zove se grupa ako vrijede sljedeća svojstva (ovdje govorimo i o aksiomima grupe):*

$$(x \circ y) \circ z = x \circ (y \circ z) \quad \forall x, y, z \in G \ (asocijativnost),$$

$$(\exists e \in G)\ e \circ x = x \circ e = x \quad \forall x \in G \ (neutralni\ element),$$

$$(\forall x \in G)(\exists! x^{-1} \in G)\ x \circ x^{-1} = x^{-1} \circ x = e \ (inverzni\ element).$$

Element e, ili e_G ako želimo posebno naglasiti da je riječ o grupi G, zove se neutralni element grupe, ili kraće neutral grupe. Za zadani $x \in G$, element $x^{-1} \in G$ koji zadovoljava gore navedeno treće po redu svojstvo, zove se inverzni element od x, ili kraće inverz od x. Ako još vrijedi i svojstvo

$$x \circ y = y \circ x \qquad \forall x, y \in G \ (komutativnost),$$

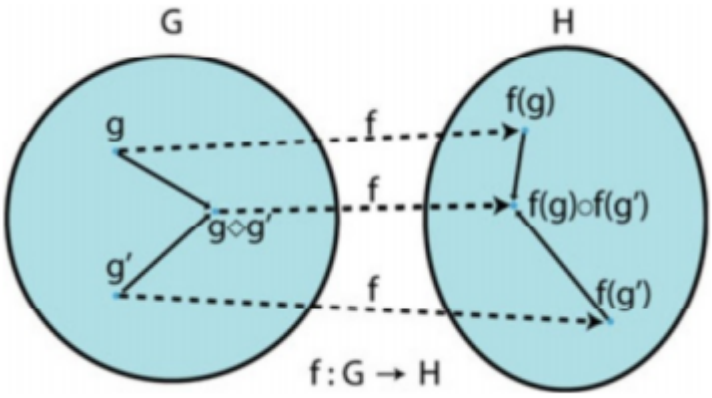
onda kažemo da je G komutativna grupa, a u suprotnom govorimo o nekomutativnoj grupi; jednako su u upotrebi i termini abelova grupa za komutativnu grupu, te neabelova grupa za nekomutativnu grupu.

U algebri, homomorfizam predstavlja preslikavanje između dvije algebarske strukture istog tipa koje čuva njihovu strukturu. Riječ homomorfizam dolazi iz grčkog jezika od riječi "*homos*" što znači isti i "*morphe*" što znači oblik.

Definicija 1.1.2. *Neka su* $(G, \circ)$ *i* $(H, \diamond)$ *dvije grupe. Preslikavanje* $f : G \to H$ *je homomorfizam grupa, ako "čuva strukturu", to jest, ako vrijedi*

$$f(x \circ y) = f(x) \diamond f(y) \qquad \forall x, y \in G.$$

Na slici 1.1. vidimo ilustrirani prikaz homomorfizma grupa.



Slika 1.1: Homomorfizam grupa

1.2 Definicija homomorfnog kriptosustava

Homomorfni kriptosustavi su oni sustavi čija funkcija šifriranja ima svojstvo homomorfizma i time čuva operacije nad šifratima unutar grupe.

Definicija 1.2.1. *Neka je* (P, C, K, E, D) *kriptosustav gdje su* P, C *konačni skupovi otvorenih tekstova i šifrata,* K *je prostor ključeva, a* E, D *algoritmi šifriranja i dešifriranja. Pretpostavimo da otvoreni tekstovi čine grupu* $(P, \diamond)$ *i šifrati čine grupu* $(C, \circ)$ *, tada je funkcija šifriranja* $e_k \in E$ *preslikavanje iz grupe* P *u grupu* C*, to jest* $E_k : P \to C$*, gdje je* $k \in K$ *tajni ključ (u simetričnim kriptosustavima) ili javni ključ (u asimetričnim kriptosustavima). Ako za sve* a *i* b *iz* P *i* k *iz* K *vrijedi*

$$E_k(a) \circ E_k(b) = E_k(a \diamond b) \tag{1.1}$$

kažemo da je kriptosustav homomorfan.

Jednostavnije rečeno definicija nam govori da je upotreba jednog binarnog operatora nad pojedinačno šifriranim otvorenim tekstovima sa istim javnim ključem isto što i šifriranje javnim ključem otvorenog teksta koji je rezultat primjene drugog binarnog operatora nad pojedinačnim otvorenim tekstovima. Promatranjem jednadžbe (1.1) uočavamo da binarni operatori $\circ$ i $\diamond$ smiju biti identični. U tom slučaju moguće je izračunati šifriranu kombinaciju pojedinačnih poruka bez korištenja originalnih poruka koje mogu ostati tajne. To nam omogućuje da korisnik može šifrirati podatke te ih poslati na obradu trećoj osobi u koju nema povjerenja. Treća osoba obradi podatke i rezultat vraća korisniku bez da je saznala informacije o podacima. Nakon što primi rezultat, korisnik dešifrira podatke i dobije rezultat u originalnom izdanju.

Prethodno opisani postupak primjenu najčešće pronalazi u komercijalnim sustavima za pohranu i obradu podataka koji su jako osjetljivi. Ti podaci mogu biti osobnog, medicinskog, financijskog tipa i slično te ne želimo da budu dostupni nikome osim korisnika.

Drugi primjer primjene homomorfne enkripcije je elektronsko glasovanje u kojem želimo da izborna administracija može prebrojavati glasove i objaviti konačne rezultate izbora bez da se dešifriraju pojedinačni glasovi te na taj način ugrozi identitet glasača. Također nam omogućuje i da vršimo šifrirane upite u web pretraživačima. Korisnik postavlja šifrirani upit, a pretraživač daje odgovor bez "gledanja" u upit u nešifriranom obliku.

Lako je zamijetiti da se homomorfna enkripcija primjenjuje u širokom području, a najčešće operacije nad šifratima su operacije zbrajanja i množenja pa tako razlikujemo aditivni i multiplikativni homomorfizam. Sustav za šifriranje koji istovremeno podržava i zbrajanje i množenje te samim time i sve algebarske operacije naziva se potpuni homomorfni kriptosustav.

Potpuni homomorfni kriptosustav do sada se koristio više u teoriji nego u praktičnoj primjeni te se nadamo da će se u bliskoj budućnosti to promijeniti. Srećom, pokazalo se da za mnoge primjene nije neophodan potpuni, već je sasvim dovoljan djelomični homomorfni kriptosustav.

U daljnjem tekstu prvo ćemo obraditi djelomične, a zatim i potpune homomorfne kriptosustave.

Poglavlje 2

Djelomični homomorfni kriptosustavi

Homomorfni kriptosustavi koji istovremeno ne podržavaju sve algebarske operacije već samo neke nazivaju se djelomični homomorfni kriptosustavi. Unatoč tome što nisu u potpunosti homomorfni, u praksi se pokazalo da je njihova primjena u rješavanju određenih problema sasvim prihvatljiva. U nastavku ćemo objasniti homomorfizam kod RSA, ElGamalovog i Pailierovog kriptosustava.

2.1 RSA

Prvi djelomični homomorfni kriptosustav koji ćemo obraditi ujedno je najpopularniji i najšire korišteni kriptosustav s javnim ključem koji su izumili Ron Rivest, Adi Shamir i Len Adleman 1977. godine. Njegova sigurnost zasnovana je na teškoći faktorizacije velikih prirodnih brojeva. Slijedi precizna definicija RSA kriptosustava.

Definicija 2.1.1. *Neka je $n = p \cdot q$, gdje su p i q prosti brojevi. Neka je $P = C = \mathbb{Z}_n$, te $K = \{(n, p, q, d, e) : n = pq,\ p, q\ prosti,\ de \equiv 1 (\mathrm{mod}\ n)\}$. Za $\mathcal{K} = (n, p, q, d, e) \in K$ definiramo*

$$e_{\mathcal{K}}(x) = x^e \bmod n \quad i \quad d_{\mathcal{K}}(y) = y^d \bmod n, \qquad x, y \in \mathbb{Z}_n. \tag{2.1}$$

Vrijednosti n i e su javne, a vrijednosti p, q i d su tajne.

RSA kriptosustav, kao i većina kriptosustava, sastoji se od tri koraka: generiranje ključa, šifriranje i dešifriranje.

Generiranje ključa

Tajno izaberemo dva velika prosta broja p i q od oko 150 znamenaka, tako da q ima nekoliko znamenaka više od p. To radimo tako da pomoću nekog generatora slučajnih brojeva generiramo prirodan broj m s traženim brojem znamenaka, a zatim korištenjem nekog

testa za testiranje prostosti tražimo prvi prosti broj veći ili jednak m. Zatim izračunamo $n = pq$ i $\varphi(n) = (p-1)(q-1) = n + 1 - p - q$, gdje je $\varphi(n)$ Eulerova funkcija, to jest broj pozitivnih cijelih brojeva manjih ili jednakih n, koji su relativno prosti sa n. Izaberemo na slučajan način broj e takav da je $e < \varphi(n)$ i $nzd(\varphi(n), e) = 1$, gdje je nzd oznaka za najveći zajednički djelitelj. To se može napraviti slično kao kod odabira p i q. Nakon toga tajno izračunamo d tako da je $de \equiv 1 (\mathrm{mod}\ \varphi(n))$, tj. $d \equiv e^{-1} (\mathrm{mod}\ \varphi(n))$. To se radi pomoću Euklidovog algoritma. Posljednji korak je da stavimo ključ za šifriranje (n, e) u javni direktorij. Dakle, javni ključ se sastoji od prirodnog broja n i javnog eksponenta e dok se tajni ključ sastoji od privatnog eksponenta d.

Šifriranje

Alice prenosi svoj javni ključ (n, e) Bobu, a kod sebe zadržava tajni ključ. Bob želi poslati Alice poruku m. Bob računa šifrat c koristeći se formulom $c = m^e \bmod n$. Nakon toga Bob šalje c Alice. Za efikasnost RSA kriptosustava, važna je činjenica da se modularno potenciranje može izvesti na efikasan način takozvanom metodom "kvadriraj i množi". Najprije e prikažemo u bazi 2:

$$e = e_0 + 2 \cdot e_1 + ... + 2^{s-1} \cdot e_{s-1} ,$$

a potom primjenimo sljedeći algoritam:

```
y = 1
za i = s-1, ...,1,0
      y = y^2 mod n
      ako je e_i = 1, onda y = y · m  mod n
```

Dešifriranje

Alice prima c koji joj je poslao Bob te korištenjem svog privatnog ključa d, iz formule $m \equiv c^d \bmod n$ dobiva m.

Svojstvo homomorfizma

Pokažimo da RSA šifriranje zadovoljava svojstvo homomorfizma.

$$E(m_1) \cdot E(m_2) = m_1^e \cdot m_2^e (\mathrm{mod}\ n) = (m_1 \cdot m_2)^e (\mathrm{mod}\ n) = E(m_1 \cdot m_2) \tag{2.2}$$

Očito je da RSA algoritam zadovoljava svojstvo multiplikativnog homomorfizma.

Primjer

Na početku Alice izabere dva prosta broja $p = 31$ i $q = 53$.
Zatim računa $n = pq = 31 \cdot 53 = 1643$ i $\varphi(1643) = 30 \cdot 52 = 1560$.

Proizvoljno izabere bilo koji broj e takav da vrijedi $1 < e < 1560$ *i* $nzd(1560, e) = 1$. U ovom slučaju neka je $e = 17$. Preostaje još izračunati d.

$$d \equiv e^{-1} (\text{mod } \varphi(n)) \equiv 17^{-1} (\text{mod } 1560) = 1193$$

Dakle, javni ključ od Alice je (1643, 17), a privatni $d = 1193$. Javni ključ prenosi Bobu dok privatni čuva kod sebe.
Neka je $m = 501$ poruka koju Bob želi poslati Alice. Bob uzima javni ključ od Alice i šifrat poruke m dobiva na sljedeći način:

$$c \equiv m^e (\text{mod } n) \equiv 501^{17} (\text{mod } 1643) = 738.$$

Bob šalje Alice $c = 738$.
Alice prima poruku i koristeći se svojim privatnim ključem dešifrira je preko formule:

$$m \equiv c^d (\text{mod } n) \equiv 738^{1193} (\text{mod } 1643) = 501.$$

Sigurnost

Sigurnost RSA kriptosustava leži u pretpostavci da je funkcija $e_K(x) = x^b (\text{mod n})$ jednosmjerna. Za funkciju f kažemo da je jednosmjerna ako je f lako, a f^{-1} teško izračunati. Ako je pritom f^{-1} lako izračunati ukoliko nam je poznat neki dodatni podatak (trapdoor - skriveni ulaz), onda f nazivamo osobna jednosmjerna funkcija. Dodatni podatak koji omogućava dešifriranje u RSA kriptosustavu je poznavanje faktorizacije $n = pq$ jer je tada lako izračunati $\varphi(n)$ i dobiti dekripcijski eksponent d pomoću proširenog Euklidovog algoritma. Trenutno najbrži algoritam za faktorizaciju treba $exp(O((\log n)^{1/3} (\log \log n)^{2/3}))$ operacija, tako da su brojevi od preko 300 znamenaka za sada sigurni od ovog napada. Dakle, nije poznat niti jedan polinomijalni algoritam za faktorizaciju. Ovdje ipak treba reći da je u nekim slučajevima n puno lakše faktorizirati, pa takve n-ove treba izbjegavati. Takav je slučaj npr. ako su p i q jako blizu jedan drugoga ili ako $p - 1$ i $q - 1$ imaju samo male proste faktore.

2.2 ElGamalov kriptosustav

ElGamalov kriptosustav je kriptosustav s javnim ključem kojeg je 1985. godine izmislio Taher ElGamal. Kriptosustav je zasnovan na teškoći računanja diskretnog logaritma u konačnim poljima. Problem diskretnog logaritma definiramo na sljedeći način:
Neka je $(G, \circ)$ *konačna grupa,* $g \in G$, $H = \{g^i : i \geq 0\}$ *grupa generirana s* g, *te* $y \in H$. *Treba naći jedinstveni cijeli broj* a *takav da je* $0 \leq a \leq |H| - 1$ *i* $g^a = y$, *gdje je* $g^a = g \circ g \circ ... \circ g$ (a *puta*). *Taj cijeli broj* a *se zove diskretni logaritam i označava se s* $\log_g y$.

U originalnom ElGamalovom kriptosustavu je $(G, \circ) = (\mathbb{Z}_p^*, \cdot_p)$, dok je g primitivni korijen modulo p. Za g kažemo da je primitivni korijen modulo p ako vrijedi svojstvo $\{g^i : i = 0, 1, ..., p-2\} = \mathbb{Z}_p^*$. U ovom slučaju to znači $H = G$.

Generiranje ključa

Ključ za ElGamalov kriptosustav generira se na sljedeći način:
Prvo izaberemo neki dovoljno velik prosti broj p za kojeg će problem diskretnog algoritma u $\mathbb{Z}_p^*$ biti težak i g koji je primitivni korijen modulo p. Mali Fermatov teorem nam kaže da ako je p prost broj i $s \in \mathbb{Z}$ takav da je $s \not\equiv 0 (\text{mod } p)$ tada vrijedi $s^{p-1} \equiv 1 (\text{mod } p)$. Za primitivni korijen g vrijedi da je g^{p-1} najmanja potencija od g koja je kongruentna 1 modulo p. Dakle, za provjeriti da je g primitivni korijen modulo p, dovoljno je provjeriti da niti jedan od brojeva $g^{(p-1)/q}$, gdje je q prosti faktor od $p-1$, nije kongruentan 1 modulo p. Nasumičnim odabirom biramo broj $a \in \mathbb{Z}_{p-1}$ i računamo y po formuli:

$$y \equiv g^a (\text{mod } p). \tag{2.3}$$

Dobivena uređena trojka (p, g, y) je javni, a broj a tajni ključ.

Šifriranje

Da bi šifrirao poruku m namjenjenu Alice, Bob dohvaća njezin javni ključ (p, g, y) i odabire proizvoljan broj $r \in \{1, ..., p-1\}$ te računa dva broja:

$$c_1 \equiv g^r \text{mod p}, \tag{2.4}$$

$$c_2 \equiv (m \cdot y^r) \text{mod p}. \tag{2.5}$$

Bobov šifrairani tekst m je uređeni par (c_1, c_2) kojeg šalje Alice.
Zamijetimo da ako znamo vrijednost od m vrlo lako možemo saznati i vrijednost od y^r. Zbog sigurnosti, za šifriranje svake nove poruke biramo novi broj r zbog čega r nazivamo prolazni ili jednokračni ključ .

Dešifriranje

Da bi dešifrirala šifrat (c_1, c_2) sa svojim privatnim ključem a, Alice treba redom izračunati:

$$x \equiv c_1^a (\text{mod } p), \tag{2.6}$$

$$m \equiv c_2 \cdot x^{-1} (\text{mod } p). \tag{2.7}$$

Pokažimo da je vrijednost desne strane od (2.7) uistinu kongruentna vrijednosti od m.

$$x^{-1} \cdot c_2 \equiv (c_1^a)^{-1} \cdot c_2 (\text{mod } p) \equiv (g^{ak})^{-1} \cdot (m \cdot (g^a)^k) (\text{mod } p)$$

$$\equiv (g^{ak})^{-1} \cdot m \cdot g^{ak} (\text{mod } p) \equiv m \ (\text{mod } p)$$

Svojstvo homomorfizma

Neka je (p, g, y) javni ključ u Elgamalovom kriptosustavu i neka je a tajni ključ. Tada je enkripcija poruke m jednaka:
$E(m, r) = (g^r, m \cdot y^r)$ pri čemu je $r \in \{1, ..., p-1\}$. Tada vrijedi :

$$\begin{aligned} E(m_1 \cdot m_2, r_1 + r_2) &= (g^{r_1+r_2}, m_1 \cdot m_2 \cdot y^{r_1+r_2}) = (g^{r_1}, m_1 \cdot y^{r_1}) \cdot (g^{r_2}, m_2 \cdot y^{r_2}) \\ &= E(m_1, r_1) \cdot E(m_2, r_2). \end{aligned} \tag{2.8}$$

Elgamalov kriptosustav također ima svojstvo multiplikativnog homomorfizma.

Primjer

Na početku Alice generira prosti broj p i $g \in \{1, ..., p-1\}$ primitivni korijen modulo p:

$$p = 2879$$
$$g = 2585$$

Znamo da je g dobro odabran jer vrijedi $g^{p-1} \equiv 1 (\text{mod } p)$.
Alice izabire proizvoljan broj a koji će biti njezin privatni ključ.

$$a = 35$$

Tada računa :

$$y = g^a = 2585^{35} \equiv 2733 (\text{mod } 2879).$$

Alice sada ima javni ključ (p, g, y) i šalje ga Bobu. Jedina osoba koja zna privatni ključ a je Alice.
Sada Bob stvori poruku $m = 82$ te izabire proizvoljnu vrijednost $r = 70$ i računa šifrat (c_1, c_2) gdje su:

$$\begin{aligned} c_1 &= g^r = 2585^{70} \equiv 1163 (\text{mod } 2879), \\ c_2 &= m \cdot y^r = 82 \cdot 2826^{70} \equiv 2298 (\text{mod } 2879). \end{aligned}$$

Alice šifrat može dešifrirati na sljedeći način:

$$c_2 / c_1^a = 2298 / 1163^{35} \equiv 2298 \cdot 1313 (\text{mod } 2879) \equiv 82 (\text{mod } 2879).$$

Sigurnost

Sigurnost ElGamalovog kriptosustava zasniva se na teškom ili gotovo nemogućem računu problema diskretnog algoritma. Prilikom šifriranja otvorenog teksta m ono što zapravo radimo je da "zamaskiramo" m množeći ga s y^r. Zamijetimo da onaj tko poznaje tajni eksponent a iz g^r može izračunati y^r i na taj način "ukloniti masku". Kako bi izbjegli takve slučajeve i eksponent a stvarno ostao tajan, prost broj p mora biti dovoljno velik da bi problem diskretnog algoritma u $\mathbb{Z}_p^*$ bio praktički nerješiv. Postavlja se pitanje koliko je to "dovoljno velik". Danas preporučena veličina za p iznosi oko 2048 bita, a red grupe, to jest broj $p-1$ trebao bi imati barem jedan veliki prosti faktor od barem 160 bitova. Najefikasniji poznati algoritmi za problem diskretnog logaritma u $\mathbb{Z}_p^*$ zasnovani su na tzv. "*index calculus metodi*" [2], a trebaju $exp(O((\log p)^{1/3}(\log\log p)^{2/3})$ operacija. Problem je po kompleksnosti jako sličan problemu faktorizacije pa su i metode koje se koriste za rješavanje tih problema također vrlo slične. Najbolja varijanta "*index calculus metode*" naziva se "*sito polja brojeva*". Očekivanih operacija za računanje diskretnog logaritma tom metodom je $exp(1.92(\ln p)^{1/3}(\ln\ln p)^{2/3})$, a algoritmi ovakve složenosti nazivaju se subeksponencijalni algoritmi. Grupa $\mathbb{Z}_p^*$ nije jedina grupa kod koje je potenciranje puno lakše od logaritmiranja. Od interesa su nam još i multiplikativne grupe konačnih polja karakteristike 2 i grupe eliptičkih krivulja nad konačnim poljima, kod kojih je razlika u težini ova dva problema još veća. Upravo zbog potrebe za što sigurnijim kriptosustavom, uz originalni, razvile su se i druge varijante ElGamalovog kriptosustava nad različitim grupama.

2.3 Paillierov kriptosustav

Za broj z kažemo da je n-ti ostatak modulo n^2 ako postoji $y \in \mathbb{Z}_{n^2}^*$ takav da vrijedi $z \equiv y^n (\mathrm{mod}\ n^2)$. Unatoč razvoju tehnologije računanje n-tog ostatka pomoću računala još uvijek je veoma kompliciran. 1999. godine Pascal Paillier došao je na ideju konstrukcije asimetričnog kriptosustava koji se zasniva na problemu rješavanja n-tog ostatka. Kao i do sada, opisat ćemo spomenuti kriptosustav u tri sljedeća koraka: generiranje ključa, šifriranje i dešifriranje.

Generiranje ključa

Izaberemo proizvoljno dva prosta broja p i q, neovisno jedan o drugome, tako da vrijedi $nzd(pq, (p-1)(q-1)) = 1$. Svojstvo je osigurano ukoliko su brojevi p i q iste duljine. Računamo:

$$n = pq,\ \lambda = nzv(p-1, q-1),$$

gdje je nzv oznaka za najmanji zajednički višekratnik .
Izaberimo proizvoljno g, gdje je $g \in \mathbb{Z}_{n^2}^*$.

Definicija 2.3.1. *Neka su a i n relativno prosti prirodni brojevi. Najmanji prirodni broj d sa svojstvom da je* $a^d \equiv 1 (\mathrm{mod}\ n)$ *zove se red od a modulo n.*

Treba vrijediti da n dijeli red od g modulo n^2. Ako je to zadovoljeno, tada postoji sljedeći multiplikativni inverz (za dokaz vidjeti [6]) :

$$\mu \equiv (L(g^\lambda \bmod n^2))^{-1} (\mathrm{mod}\ n), \tag{2.9}$$

gdje je funkcija L definirana kao:

$$L(u) = \frac{u-1}{n}. \tag{2.10}$$

Konačno, javni ključ je (n, g) dok je privatni ključ (λ, μ).

Ako koristimo p, q iste duljine, postoji jednostavniji način generiranja ključa:

$$g = n + 1,\ \lambda = \varphi(n),\ \mu \equiv \varphi(n)^{-1} (\mathrm{mod}\ n)$$

gdje je $\varphi(n) = (p-1)(q-1)$.

Šifriranje

Neka je m poruka koju treba šifrirati, takva da je $m \in \mathbb{Z}_n$. Izaberimo proizvoljan $r \in \mathbb{Z}_n^*$. Šifrat računamo na sljedeći način:

$$c \equiv g^m \cdot r^n (\mathrm{mod}\ n^2). \tag{2.11}$$

Dešifriranje

Neka je c šifrat koji trebamo dešifrirati, takav da je $c \in \mathbb{Z}_{n^2}^*$.
Otvoreni tekst računamo preko formule :

$$m \equiv L(c^\lambda \bmod n^2) \cdot \mu (\mathrm{mod}\ n).$$

Paillierov algoritam koristi činjenicu da je određene diskretne logaritme lagano izračunati. Na primjer, iz binomnog teorema:

$$(1+n)^x = \sum_{k=0}^{x} \binom{x}{k} n^k = 1 + nx + \binom{x}{2} n^2 + (...)$$

slijedi da je

$$(1+n)^x = 1 + nx (\mathrm{mod}\ n^2).$$

Stoga , ako je

$$y \equiv (1+n)^x (\text{mod } n^2),$$

tada je

$$x \equiv \frac{y-1}{n} (\text{mod } n).$$

Prema tome

$$L((1+n)^x (\text{mod } n^2)) \equiv x (\text{mod } n)$$

za svaki $x \in \mathbb{Z}_n$. Dakle, kada je $g = n+1$, imamo :

$$L(c^\lambda (\text{mod } n^2)) \cdot \mu = L((g^m r^n)^\lambda (\text{mod } n^2)) \cdot \lambda^{-1} = L((g^m \lambda (\text{mod } n^2)) \cdot \lambda^{-1}$$

$$= \lambda \cdot m \cdot \lambda^{-1} \equiv m (\text{mod } n).$$

Svojstvo homomorfizma

Značajno svojstvo Paillierovog kriptosustava je svojstvo homomorfizma. Neka su $E(m_1, pk) = g^{m_1} r_1^n (\text{mod } n^2)$ i $E(m_2, pk) = g^{m_2} r_2^n (\text{mod } n^2)$ dva šifrata, gdje su r_1, r_2 proizvoljno izabrani iz $\mathbb{Z}_n^*$.

• Aditivni homomorfizam

Dešifrirani produkt dva šifrata jednak je zbroju njihovih otvorenih tekstova.

$$D(E(m_1, pk) \cdot E(m_2, pk) (\text{mod } n^2)) \equiv m_1 + m_2 (\text{mod } n)$$

jer je

$$E(m_1, pk) \cdot E(m_2, pk) = (g^{m_1} r_1^n)(g^{m_2} r_2^n) (\text{mod } n^2)$$

$$= g^{m_1+m_2} (r_1 r_2)^n (\text{mod } n^2)$$

$$= E(m_1 + m_2, pk).$$

$$D(E(m_1, pk) \cdot g^{m_2} (\text{mod } n^2)) \equiv m_1 + m_2 (\text{mod } n)$$

jer je

$$\begin{aligned} E(m_1, pk) \cdot g^{m_2} &= (g^{m_1} r_1^n) g^{m_2} (\text{mod } n^2) \\ &= g^{m_1+m_2} r_1^n (\text{mod } n^2) \\ &= E(m_1 + m_2, pk). \end{aligned}$$

• Multiplikativni homomorfizam

Vrijedi :

$$D(E(m_1, pk)^{m_2} (\text{mod } n^2)) \equiv m_1 m_2 (\text{mod } n)$$

jer je

$$\begin{aligned} E(m_1, pk)^{m_2} &= (g^{m_1} r_1^n)^{m_2} (\text{mod } n^2) \\ &= g^{m_1 m_2} (r_1^{m_2})^n (\text{mod } n^2) \\ &= E(m_1 m_2, pk). \end{aligned}$$

Općenito, dešifriranjem šifriranog otvorenog teksta na potenciju konstante k dobit ćemo isto kao i umnoškom otvorenog teksta i konstante k

$$D(E(m_1, pk)^k (\text{mod } n^2)) \equiv k m_1 (\text{mod } n).$$

Međutim, za dvije šifrirane poruke Paillierovom enkripcijom nemoguće je izračunati njihov šifrirani produkt bez poznavanja privatnog ključa.

Primjer

Zbog lakšeg računanja, u primjeru ćemo koristiti male brojeve.
Neka je p=7 i q=11. Tada je

$$n = pq = 7 \cdot 11 = 77.$$

Sada izaberimo $g \in \mathbb{Z}_{n^2}^*$ tako da je red od g višekratnik od n u $\mathbb{Z}_n^2$. Ako proizvoljno uzmemo da je $g = 5652$, tada su zadovoljena sva potrebna svojstva jer je $|g| = 2310 = 30 \cdot 77$ *u* $\mathbb{Z}_{n^2}^*$. Prema tome javni ključ bit će uređeni par

$$(n, g) = (77, 5652).$$

Da bi šifrirali poruku m= 55, gdje je $m \in \mathbb{Z}_n$ proizvoljno odaberemo

$$r = 32,$$

gdje je r različit od nule i $r \in \mathbb{Z}_n$.

Računamo:

$$c \equiv g^m r^n (\text{mod } n^2) \equiv 5652^{55} \cdot 32^{77} (\text{mod } 5929) \equiv 1693 (\text{mod } 5929).$$

Da bi dešifrirali šifrat c, računamo

$$\lambda = nzv(6, 10) = 30.$$

Neka je $L(u) = (u - 1)/n$, računamo

$$\begin{aligned} k &= L(g^\lambda (\text{mod } n^2)) = L(5652^{30} (\text{mod } 5929)) \\ &= L(3928) = (3928 - 1)/77 = 3927/77 = 51. \end{aligned}$$

Izračunamo inverz od k, a zatim i vrijednost od μ

$$\mu \equiv k^{-1} (\text{mod } n) \equiv 51^{-1} \equiv 74 (\text{mod } 77).$$

Dalje imamo

$$\begin{aligned} m &\equiv L(c^\lambda \text{mod } n^2) \cdot \mu (\text{mod } n) \equiv L(1693^{30} (\text{mod } 5929)) \cdot 74 (\text{mod } 77) \\ &\equiv L(2542) \cdot 74 (\text{mod } 77) \equiv 33 \cdot 74 (\text{mod } 77) = 42. \end{aligned}$$

Sigurnost

Za kriptosustave s javnim ključem u kojima presretač nije u mogućnosti u polinomijalnom vremenu uočiti razliku između dva ili više šifrata, kažemo da su postigli semantičku sigurnost. Paillierov kriptosustav pruža semantičku sigurnost protiv napada odabrani otvoreni tekst (kriptoanalitičar ima mogućnost odabira teksta koji će biti šifriran, te može dobiti njegov šifrat). Postizanje semantičke sigurnosti nije lak posao jer se na ovaj način omogućuje zaštita od ciljeva koje presretač lako postiže. U Paillierovom kriptosustavu semantička sigurnost se postiže zahvaljujući teškoći rješavanja sljedećeg problema: za dane brojeve z i n postoji li y takav da vrijedi :

$$z \equiv y^n (\text{mod } n^2).$$

Međutim, zbog homomorfnog svojstva Paillierov kriptosustav nije zaštićen od napada odabranog šifrata (kriptoanalitičar je dobio pristup alatu za dešifriranje pa može odabrati šifrat te dobiti odgovarajući otvoreni tekst). Inače se u kriptografiji to ne smatra prednošću, ali kod određenih sustava kao što je tajno elektroničko glasovanje to svojstvo može biti neophodno.

Poglavlje 3

Potpuni homomorfni kriptosustavi

Homomorfna enkripcija je vrlo koristan alat koji ima velik broj primjena. Međutim, primjene su ograničene činjenicom da je moguće vršiti samo jednu operaciju nad otvorenim tekstom dok je šifriran. Ono što bi stvarno bilo korisno jest mogućnost izvršavanja zbrajanja i množenja istovremeno. To bi dopustilo više manipulacija nad otvorenim tekstom vršeći operacije nad šifratom. Zapravo bi to omogućilo da netko bez tajnog ključa može učinkovito izračunati funkciju nad otvorenim tekstom kada mu je poznat samo šifrat. U ovom poglavlju obradit ćemo algoritme potpune homomorfne enkripcije, koji omogućuju obavljanje jedne ili obje operacije, i zbrajanja i množenja otvorenog teksta dok je šifriran. Prva ideja potpuno homomorfnih kriptosustava bila je predstavljena od strane Rivesta pod imenom "*privatni homomorfizmi*". Problem konstruiranja algoritma sa spomenutim svojstvima ostao je neriješen sve do 2009. godine, kada Gentry prezentira svoja rješenja. Njegov algoritam dopušta matematičke operacije sa šifriranim podacima, a zatim dešifriranje rezultata daje isti odgovor kao analogni rad s nekodiranim podacima. Ovo poglavlje započet ćemo s uvodom modela potpunog homomorfnog kriptosustava, njegovom definicijom te konstrukcijom algoritama nad cijelim brojevima.

3.1 Definicija potpunog homomorfnog kriptosustava

Definicija 3.1.1. *Potpun homomorfni kriptosustav možemo smatrati kao homomorfizam prstena. U matematici, prsten je neprazan skup* $R = (R, +, \cdot)$ *za koji je za operacije zbrajanja* $+ : R \times R \to R$ *i množenja* $\cdot : R \times R \to R$ *ispunjeno sljedeće:*
(1) $(R, +)$ *je komutativna grupa, sa neutralom* $0 = 0_R$*;*
(2) $(R, \cdot)$ *je polugrupa, tj. množenje je asocijativno;*
(3) Vrijedi distributivnost "množenja prema zbrajanju", tj.

$$x \cdot (y + z) = x \cdot y + x \cdot z, \quad \forall x, y, z \in R,$$

$$(x + y) \cdot z = x \cdot z + y \cdot z, \quad \forall x, y, z \in R.$$

Homomorfizam prstena je funkcija između dva prstena koja čuva strukturu, tj. homomorfizam prstena je preslikavanje $f : R \to S$, R i S prsteni, koje je i aditivno i multiplikativno, tj. ako vrijedi $f(x + y) = f(x) + f(y)$ & $f(xy) = f(x) \cdot f(y)$ $\forall x, y \in R$ te ako je $f(1_R) = 1_S$.

Definicija 3.1.2. *Neka je (P, C, K, E, D) kriptosustav gdje su P, C konačni skupovi otvorenog teksta i šifrata, K je prostor ključeva, a E, D algoritmi šifriranja i dešifriranja. Pretpostavimo da otvoreni tekstovi čine prsten $(P, \oplus_P, \otimes_P)$ i šifrati čine prsten $(C, \oplus_C, \otimes_C)$, tada je funkcija šifriranja $e_k \in E$ preslikavanje iz prstena P u prsten C, tj. $E_k : P \to C$, gdje je $k \in K$ tajni ključ (u simetričnim kriptosustavima) ili javni ključ (u asimetričnim kriptosustavima). Ako za sve a, b iz P i k iz K vrijedi*

$$E_k(a) \oplus_C E_k(b) = E_k(a \oplus_P b) \tag{3.1}$$

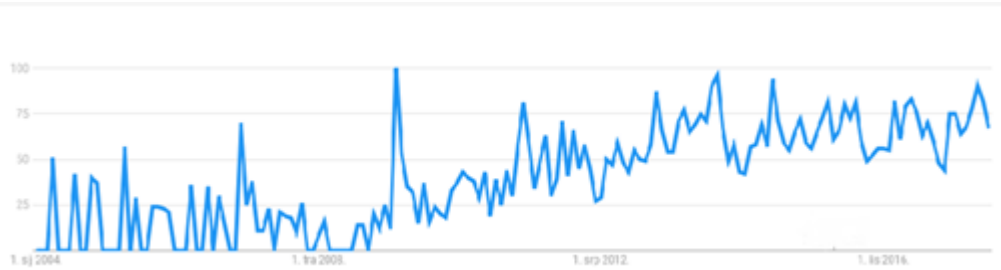
$$E_k(a) \otimes_C E_k(b) = E_k(a \otimes_P b) \tag{3.2}$$

kažemo da je kriptosustav potpuno homomorfan.

3.2 Povijesni razvoj potpuno homomorfnog kriptosustava

Craig Gentry autor je prvog algoritma potpuno homomorfnog kriptosustava koji je predstavljen od strane IBM-a 25. lipnja 2009. godine. Ovaj algoritam se bazira na djelomično homomorfnom kriptosustavu zasnovanom na idealnim rešetkama *(eng. ideal lattices)*. Dobra strana algoritma je da vrijeme obrade linearno ovisi samo o broju operacija koje se obavljaju, dok je mana ograničenje na mali broj operacija nad šifratima. Svaka operacija nad šifratima čini male izmjene (šumove), tako da na kraju šifrat postaje toliko oštećen da se više ne može dešifrirati. Uz to, podizanjem nivoa sigurnosti veličina šifrata i vrijeme obrade drastično se povećavaju pa algoritam postaje neefikasan. Sam Gentry procjenjuje da bi se korištenjem ovog algoritma za pretraživanje weba uz pomoć šifriranih riječi, potrebno vrijeme povećalo 10^{12} puta. Algoritam su kasnije pojednostavili Stehle Damien i Steinfeld Ron kao i Nigel Smart i Frederik Vercauteren kako bi smanjili nedostatke, ali i takvo pojednostavljenje je i dalje nepraktično za svakodnevnu primjenu.

U 2009. godini, Marten van Dijk, Craig Gentry, Shai Halevi i Vinod Vaikuntanathan predstavili su drugi algoritam potpunog homomorfnog kriptosustava, koji se koristi većim dijelom Gentryjevim algoritmom, ali ne zahtjeva idealne rešetke. Umjesto toga pokazali su kako homomorfni algoritam zasnovan na idealnim rešetkama može biti zamjenjen s jednostavnijim homomorfnim algoritmom koji koristi cijele brojeve. I ovaj algoritam je nažalost



Slika 3.1: Indeks pretrage za pojmom “homomorphic encryption”

relativno neefikasan za praktičnu primjenu pa su ga u više navrata pokušali prerađivati Jean-Sébastien Coron, Tancrède Lepoint, Avradip Mandal, David Naccache i Mehdi Tibouch čime završava prva generacija potpunih homomorfnih kriptosustava.

Druga generacija započinje u 2011. godini sa radom Zvika Brakerskija, Craig Gentryja i Vinod Vaikuntanathana koji su u dvije godine razvili više novih tehnika za kontroliranje šumova što je rezultiralo znatno efikasnijim potpuno homomorfnim algoritmima.

Treću generaciju započinje Gentry sa suradnicima 2013. godine predstavljajući drugačiji uzorak šuma. Algoritam zadovoljava asimetrično množenje, tj. $c_1 \otimes c_2$ rezultira drugačijim šifratom u odnosu na $c_2 \otimes c_1$ (oba izraza šifriraju isti produkt $b_1 \cdot b_2$). Druga, još važnija činjenica je da šum također raste asimetrično. Šum u lijevom umnošku ima veći utjecaj na rezultat nego šum u desnom umnošku. Kasnije su Brakerski i Vaikuntanathan pokazali da se asimetričnost može iskoristiti kao bi se usporio rast šuma na način da se konstruiraju sklopovi u kojima lijevi umnožak uvijek ima manji šum. To je u budućnosti omogućilo smanjenje broja parametara i kompliciranost samog algoritma. Međutim, algoritmi treće generacije nisu kompatibilni sa svojstvima optimizcije druge generacija, ali zato imaju veći domet.

Nažalost ni do današnjeg dana akademska zajednica nije došla do efikasnog potpuno homomorfnog kriptosustava koji bi imao praktično primjenjivu implementaciju u smislu brzine šifriranja i dešifriranja. Ipak, unatoč nedostatcima u implementaciji uočljiv je porast interesa koji akademska zajednica ulaže kako bi se unaprijedio razvoj homomrfnih kriptosustava. To možemo zaključiti i na osnovi slike 3.1. koja pokazuje indeks pretrage za pojmom “homomorphic encryption” u pretraživaču google.com posljednjih četrnaest godina.

3.3 Donekle potpuni homomorfni kriptosustav nad cijelim brojevima

Homomorfna enkripcija zasnovana na rešetkama, iako veoma zanimljiva s teoretske strane, vrlo je kompliciranа za objasniti. Iz toga razloga koristit ćemo kriptosustav koji je jednostavniji za razumjeti. U ovom poglavlju opisat ćemo simetrični i asimetrični donekle potpun homomorfni kriptosustav koji se koristi samo osnovnom modularnom aritmetikom, to jest zbrajanjem i množenjem cijelih brojeva.

Donekle potpuni homomorfni kriptosustav s privatnim ključem

Koji je najjednostavniji način na koji bismo mogli šifrirati poruku i pritom postići sigurnost? Cezarova šifra je jednostavna, ali nije sigurna. Vjeruje se da su asimetrični kriptosustavi s modularno eksponencijalnim funkcijama sigurni, ali računanje takvih funkcija nije jednostavna operacija. Kada bi trebali zaboraviti sve dosadašnje poznate kriptosustave i započeti ispočetka, dobar kandidat za jednostavni simetrični kriptosustav možda bi bio donekle potpuni homomorfni kriptosustav s privatnim ključem. Unatoč svojoj jednostavnosti algoritam sadrži vrlo složene funkcionalnosti.

Generiranje ključa

Privatni ključ p je neparan cijeli broj iz intervala $p \in [2^{\eta-1}, 2^{\eta}]$, gdje je η veličina privatnog ključa u bitovima.

Šifriranje

Šifriramo bit $m \in \{0, 1\}$ na način da definiramo šifrat kao cijeli broj čiji ostatak modulo p ima istu parnost kao i otvoreni tekst. Naime,

$$c = pq + 2r + m \tag{3.3}$$

gdje su q, r cijeli brojevi proizvoljno odabrani tako da vrijedi da je $2r$ po apsolutnoj vrijednosti manje od $p/2$.

Dešifriranje

Dobiveni šifrat c dešifriramo pomoću privatnog ključa p:

$$m \equiv (c(\text{mod } p))(\text{mod } 2). \tag{3.4}$$

Pokažimo da je vrijednost od m uistinu jednaka vrijednosti desne strane od (3.4).

$$\begin{aligned}(c(\text{mod } p))(\text{mod } 2) &\equiv ((pq + 2r + m)(\text{mod } p))(\text{mod } 2)\\ &\equiv (2r + m)(\text{mod } 2)\\ &\equiv m\end{aligned}$$

Primjer

Neka je p=17. Šifrirajmo poruku m=0

$$c = pq + 2r + m = 17 \cdot 3 + 2 \cdot 1 + 0 = 53,$$

gdje je $q = 3$ i $r = 1$.
Dobiveni šifrat je $c = 53$.
Kako bi dešifrirati c koristimo se formulom:

$$\begin{aligned}(c(\text{mod } p))(\text{mod } 2) &\equiv (53(\text{mod } 17))(\text{mod } 2)\\ &\equiv 2(\text{mod } 2) \equiv 0.\end{aligned}$$

Svojstvo potpunog homomorfizma

Neka su $c_1 = pq_1 + 2r_1 + m_1$ i $c_2 = pq_2 + 2r_2 + m_2$ dva dana šifrata. Izračunajmo njihov zbroj i umnožak.

$$c_1 + c_2 = (q_1 + q_2)p + 2(r_1 + r_2) + (m_1 + m_2) \tag{3.5}$$

$$\begin{aligned}c_1 \cdot c_2 = (pq_1q_2 + 2q_1r_2 + 2q_2r_1 + m_1q_2 + m_2q_1)p\\ + 2(2r_1r_2 + m_1r_2 + m_2r_1) + m_1 \cdot m_2\end{aligned} \tag{3.6}$$

Ako je

$$r_1 + r_2 < p/2$$

$$2r_1r_2 + m_1r_2 + m_2r_1 < p/2,$$

tada vrijedi

$$(c_1 + c_2(\text{mod } p))(\text{mod } 2) = m_1 + m_2$$

$$(c_1 \cdot c_2(\text{mod } p))(\text{mod } 2) = m_1m_2.$$

Uvjerimo se primjerom da ovo uistinu vrijedi. Neka je $p = 17$, $m_1 = 0$ i $m_2 = 1$.

$$c_1 = p \cdot 2 + 2 \cdot 2 + 0 = 38$$

$$c_2 = p \cdot 3 + 2 \cdot 1 + 1 = 54$$

gdje su $q_1 = 2$, $r_1 = 2$, $q_2 = 3$ i $r_2 = 1$.
Jednostavno je vidjeti da vrijedi

$$\begin{aligned}(c_1 + c_2(\text{mod } p))(\text{mod } 2) &\equiv (92(\text{mod } 17))(\text{mod } 2)\\ &\equiv 7(\text{mod } 2)\\ &\equiv 1 = 0 + 1 = m_1 + m_2\end{aligned}$$

$$\begin{aligned}(c_1 \cdot c_2(\text{mod } p))(\text{mod } 2) &\equiv (2052(\text{mod } 17))(\text{mod } 2)\\ &\equiv 12(\text{mod } 2)\\ &\equiv 0 = 0 \cdot 1 = m_1 \cdot m_2.\end{aligned}$$

Dakle, kriptosustav zadovoljava svojstvo potpunog homomorfizma.
Funkcija $f : B^n \to B$, gdje je $B = \{0, 1\}$ naziva se Booleova funkcija. Primjenom svojstva potpunog homomorfizma prilikom računanja Booleove funkcije $f(x_1, x_2, ..., x_n)$ gdje je c_i šifrat od x_i, za $i = 1, 2, ..., n$, iz (3.5) i (3.6) možemo primjetiti da

$$r_1 + r_2 \geq max(r_1, r_2),$$

$$2r_1r_2 + m_1r_2 + m_2r_1 \geq max(r_1, r_2).$$

To jest, veličina šuma komponente r u šifratu raste s povećanjem broja zbrajanja i množenja u Booleovoj funkciji. Jednom kada bude vrijedilo

$$r_1 + r_2 > p/2,$$

$$2r_1r_2 + m_1r_2 + m_2r_1 > p/2,$$

dešifrirana funkcija $f(c_1, c_2, ..., c_n)$ možda ne bude jednaka $f(x_1, x_2, ..., x_n)$. Stoga, ovaj kriptosustav je moguće koristiti samo prilikom računanja Booleove funkcije niskog stupnja nad šifriranim ulazom. Upravo je to razlog zbog kojeg se homomorfni kriptosustav naziva donekle potpun. Ukoliko izaberemo $r \approx 2^n$, $p \approx 2^{n^2}$ *i* $q \approx 2^{n^5}$ donekle potpun homomorfni kriptosustav može točno izračunati polinome stupnja $\approx n$ prije nego šum postane prevelik.

Sigurnost

Sigurnost ovog kriptosustava možemo svesti na problem približnog najvećeg zajedničkog djelitelja [3]. Opišimo spomenuti problem za slučaj kada imamo dva cijela broja. Dakle, imamo zadana dva cijela broja a i b. Njihov zajednički djelitelj d možemo pronaći u polinomnom vremenu pomoću Euklidovog algoritma. Ukoliko je d u nekom smislu "velik" moguće je dopustiti aditivne pogreške na nekom od brojeva a ili b, ili na oba, pa da još uvijek možemo ispravno odrediti njihov najveći zajednički djelitelj. Ukoliko postoji previše pogrešaka nastalih na brojevima a i b algoritam možda neće biti u mogućnosti prepoznati d kao najveći zajednički djelitelj. Ovaj problem detaljno je analizirao Howgrave-Graham [3]. Kako bi se zaštitili od napada na kriptosustav potrebno je prilikom šifriranja pametno odabrati parametre r i q. Za $r \approx 2^{\sqrt{\eta}}$ i $q \approx 2^{\eta^3}$ možemo reći da je kriptosustav siguran.

Donekle potpun homomorfni kriptosustav s javnim ključem

Donekle potpun homomorfni kriptosustav s privatnim ključem koristi privatni ključ p kako bi šifrirao poruku m. U ovom potpoglavlju objasnit ćemo donekle potpun homomorfni kriptosustav s javnim ključem koji dopušta šifriranje bez poznavanja privatnog

ključa. Kriptosustav koristi pet parametara, koji polinomijalno ovise o parametru λ kojim izražavamo razinu sigurnosti sustava. Prvi parametar η definira veličinu tajnog ključa u bitovima, drugi parametar γ predstavlja elemente javnog ključa, također u bitovima, a treći parametar τ je broj elemenata javnog ključa. Posljednja dva parametra ρ i ρ' predstavljaju duljinu šuma u bitovima u javnom ključu, odnosno šifratu.
Van Dijk sa suradnicima je izabrao sljedeće vrijednosti parametara $\eta, \gamma, \rho, \rho'$ i τ kako bi osigurao kriptosustav i spriječio napade:

$$\eta = \lambda^2,\ \gamma = \lambda^5,\ \rho = \lambda,\ \rho' = 2\lambda\ i\ \tau = \gamma + \lambda.$$

Generiranje ključa

Privatni ključ p je neparni cijeli broj, proizvoljno odabran iz intervala $[2^{\eta-1}, 2^{\eta}]$. Koristeći privatni ključ p generiramo javni ključ na sljedeći način:

$$x_i = pq_i + r_i, \tag{3.7}$$

gdje su $q_i \in \mathbb{Z} \cap [0, 2^{\gamma}/p)$ i $r_i \in \mathbb{Z} \cap (-2^{\rho}, 2^{\rho})$ proizvoljno odabrani, za svaki $i = 0, 1, ..., \tau$. Nakon toga, promjenimo oznake od x_i tako da x_0 bude najveći od njih. Također x_0 treba biti neparan, a $x_0 (\text{mod } p)$ paran. Ukoliko x_0 ne zadovoljava prethodne uvjete ponovno ispočetka ponovimo proces generiranja javnog ključa. Na kraju, javni ključ je

$$pk =< x_0, x_1, ..., x_{\tau} > .$$

Šifriranje

Da bi se šifrirala poruka $m \in \{0, 1\}$, odaberemo proizvoljan podskup $S \subseteq \{1, 2, ..., \tau\}$, gdje je τ broj elemenata u javnom ključu. Također odaberemo i proizvoljan cijeli broj $r \in (2^{-\rho'}, 2^{\rho'})$. Tada šifrat računamo kao

$$c = (m + 2r + 2\sum_{i \in S} x_i)(\text{mod } x_0). \tag{3.8}$$

Dešifriranje

Šifrat c dešifriramo pomoću privatnog ključa p i dobivamo poruku m na sljedeći način:

$$m = (c(\text{mod } p))(\text{mod } 2). \tag{3.9}$$

Podsjetimo da je

$$c(\text{mod } p) = c - p \cdot [c/p],$$

gdje je $[c/p]$ najveće cijelo od c/p. Budući da je p neparan, poruku m možemo dobiti i koristeći se formulom

$$\begin{aligned} m &= (c - p \cdot [c/p])(\text{mod } 2) \\ &= (c(\text{mod } 2)) \oplus ([c/p](\text{mod } 2)), \end{aligned}$$

gdje je $p(\text{mod } 2) = 1$. Ostatak pri djeljenju šifrata privatnim ključem iste je parnosti kao i šifrirana poruka m jer je $p \times q_i \equiv 0(\text{mod } p)$. Stoga, budući da je vrijednost $2 \sum r_i$ parna i ukoliko je šum od x_0 također paran, ostatak pri djeljenju s 2 eliminira sav šum koji je mogao nastati pri šifriranju. Ove činjenice omogućuju ispravno dešifriranje jer jedini način za uklanjanje šuma prilikom šifriranja je razmatranje parnosti.

Primjer

Neka je $p = 10001$ tajni ključ. Koristeći se tajnim ključem p konstruiramo javni ključ. Neka su

$$[q_0, q_1, q_2, q_3] = [36, 27, 34, 8],$$
$$[r_0, r_1, r_2, r_3] = [8, 6, 4, 3].$$

Javni ključ pk računamo kao vektor

$$[x_0, x_1, x_2, x_3] = [360044, 270033, 340038, 80011],$$

gdje je $x_i = q_i p + r_i$ i x_0 je najveći. Sada ćemo šifrirati dvije poruke $m_1 = 0$ i $m_2 = 1$. Pretpostavimo da je S=[1,3]. Proizvoljno izaberemo cijeli broj $r = 31$ i šifriramo poruku m_1 kao

$$\begin{aligned} c_1 &= m_1 + 2 \cdot r + 2 \cdot \sum_{i \in S} x_i \\ &= 0 + 2 \cdot 31 + 2 \cdot (270033 + 80011) \\ &= 700150. \end{aligned}$$

Zbog kompaktnosti, korisno je skratiti šifrat s x_0:

$$c_1' \equiv c_1(\text{mod } x_0) \equiv 340106(\text{mod } 360044).$$

Na isti način šifriramo i poruku m_2. Neka je $r = 11$ i $S = [2, 3]$.

$$\begin{aligned} c_2 &= m_2 + 2 \cdot r + 2 \cdot \sum_{i \in S} x_i \\ &= 1 + 2 \cdot 11 + 2 \cdot (340038 + 80011) \\ &= 840121 \end{aligned}$$

$$c_2' \equiv c_2 (\text{mod } x_0) \equiv 120033 (\text{mod } 360044).$$

Kao što je bilo i za očekivati, šifrati su ispravni. U to se možemo uvjeriti tako da ih dešifriramo i usporedimo s polaznim porukama m_1 i m_2.

$$c_1' = 340106 \equiv 72 (\text{mod } p) \equiv 0 (\text{mod } 2)$$

$$c_2' = 120033 \equiv 21 (\text{mod } p) \equiv 1 (\text{mod } 2)$$

Na kraju još pokažimo da primjer zadovoljava svojstvo potpunog homomorfizma.

$$c_1' + c_2' = 340106 + 120033 \equiv 93 (\text{mod } p) \equiv 1 (\text{mod } 2) = 0 + 1 = m_1 + m_2$$

$$c_1' \cdot c_2' = 340106 \cdot 120033 \equiv 1512 (\text{mod } p) \equiv 0 (\text{mod } 2) = 0 \cdot 1 = m_1 \cdot m_2$$

Sigurnost

Kao i kod donekle potpunog homomorfnog kriptosustava s privatnim ključem, sigurnost donekle potpunog homomorfnog kriptosustava s javnim ključem također se zasniva na problemu približnog najvećeg zajedničkog djelitelja. Neka je $\{x_0, x_1, ..., x_t\}$, gdje je $x_i = pq_i + r_i$ primjer u približnom NZD algoritmu. Poznati napadi na približni NZD problem za dva broja uključuju brute force napad ili napad uzastopnim pokušavanjem, verižne razlomke i Howgrave-Grahamov približni NZD algoritam.
Jednostavni brute-force napad za dva broja pokušava pogoditi r_1 i r_2 i provjeriti nagađanje pomoću NZD algoritma. Specijalno, za $r_1', r_2' \in (2^{-\rho}, 2^{\rho})$ imamo

$$x_1' = x_1 - r_1',\ x_2' = x_2 - r_2',\ p' = NZD(x_1',\ x_2').$$

Broj p' je moguće rješenje ukoliko je njegova veličina η bita. Ovom tehnikom gotovo sigurno možemo pronaći rješenje p. Ukoliko još kod izbora parametara izaberemo ρ koji je mnogo manji od η, velika je vjerojatnost da će rješenje biti jedinstveno. Vrijeme potrebno da se ovakav napad izvrši je oko $2^{2\rho}$.

3.4 Potpuni homomorfni kriptosustavi nad cijelim brojevima

U ovom poglavlju opisat ćemo konstrukciju potpuno homomorfnog kriptosustava koju je 2010. godine predstavio van Dijk sa suradnicima. Kriptosustav se bazira na prethodna dva kriptosustava opisana u poglavlju 3.3. i potiskivanju dešifriranih sklopova. U nastavku uzimamo u obzir samo kriptosustave koji su homomorfni s obzirom na Booleove sklopove

koji se sastoje od vrata za zbrajanje i množenje. Booleov sklop ili logički sklop oponaša djelovanje osnovnih logičkih operacija u računalu. Računalo pamti samo dva stanja, koja mi prikazujemo s 1 i 0. Svaki program može se prikazati u obliku logičkog sklopa, a svaki logički sklop, shodno disjunktivnoj normalnoj formi može se svesti na tri osnovne vrste logičkih sklopova, *logičko i , logičko ili* i *logičko ne*. S obzirom da vrijedi:

$$xor(x,1) = not(x)$$
$$not(and(not(x), not(y))) = !(!x\&!y) = xor(x,y)$$

zaključujemo da se svaki program može prikazati pomoću *logičkog i* i *logičkog isključivo ili*. Glavna motivacija autora ovog kriptosustava je koncept jednostavnosti, to jest pokazati da se nešto složeno kao potpuni homomorfni kriptosustav može postići korištenjem "osnovnih" tehnika.

Potisnuti kriptosustav

Kako bi definirali potisnuti kriptosustav uvodimo tri nova parametra u ovisnosti o λ. Definiramo

$$\kappa = \gamma\eta/\rho',\ \theta = \lambda\ i\ \Theta = \omega(\kappa \cdot \log\lambda).$$

Tajni ključ je $sk^* = p$, a javni ključ pk^* je isti kao kod donekle potpunog homomorfnog kriptosustava, ali mu još dodamo skup $y = \{y_1, y_2, ..., y_\Theta\}$ gdje su y_i racionalni brojevi iz $[0, 2)$ veličine κ bita, takvi da postoji prorijeđen podskup $S \subset \{1, ..., \Theta\}$ veličine θ tako da vrijedi $\sum_{i\in S} y_i \approx 1/p(\text{mod } 2)$. Po novom je tajni ključ zamjenjen vektorom koji pokazuje na podskup S.

Generiranje ključa

Tajni ključ $sk^* = p$ i javni ključ pk^* generiramo kao i prije. Još preostaje izgenerirati skup y. Neka je $x_p = [2^\kappa/p]$ te izaberemo proizvoljni Θ-bitni vektor $< s_1, s_2, ..., s_\Theta >$ sa Hammingovom udaljenosti (broj pozicija u kojima su odgovarajući simboli unutar dva broja iste duljine različiti.) θ i $S = \{i : s_i = 1\}$. Također biramo i proizvoljan cijeli broj $u_i \in \mathbb{Z} \cap [0, 2^{\kappa+1}), i = 1, 2, ..., \Theta$ takav da vrijedi $\sum_{i\in S} u_i = x_p(\text{mod } 2^{\kappa+1})$. Sada je $y_i = u_i/2^\kappa$ i na taj način dobijemo $y = \{y_1, y_2, ..., y_\Theta\}$. Dakle, svaki y_i je pozitivan broj manji od 2, sa κ bita iza binarne točke. Također vrijedi:

$$\sum_{i\in S} y_i(\text{mod}2) = (1/p) - \Delta_p,$$

za $|\Delta_p| < 2^{-\kappa}$ jer

$$\begin{aligned}\sum_{i\in S} y_i &= \sum_{i\in S} u_i/2^{\kappa} \\ &= (x_p + \alpha \cdot 2^{\kappa+1})/2^{\kappa} \\ &= x_p/2^{\kappa} + \alpha \cdot 2 \\ &= [2^{\kappa}/p]/2^{\kappa} + \alpha \cdot 2 \\ &= (1/p - \Delta/2^{\kappa}) + \alpha \cdot 2 \\ &= 1/p - \Delta_p (\text{mod } 2)\end{aligned}$$

gdje je $|\Delta| < 1$. Tajni ključ $sk = \vec{s}$, a javni ključ je $pk = (pk^*, y)$.

Šifriranje

Generiramo šifrat c^* kao i prije. Za svaki $i \in \{1, ..., \Theta\}$ računamo

$$z_i = c^* \cdot y_i (\text{mod } 2)$$

zadržavajući samo $n = [\log \theta] + 3$ bita nakon binarne točke za svaki z_i. Dobiveni šifrat je c^* i $\vec{z} =< z_1, ..., z_\Theta >$.

Dešifriranje

Pomoću tajnog ključa p dobiveni šifrat $(c^*, \vec{z})$ dešifriramo na sljedeći način:

$$m = (c^* - \sum_{i\in S} z_i)(\text{mod } 2). \tag{3.10}$$

Sigurnost

Sigurnost potisnutog kriptosustava kao i sigurnost donekle potpunog homomorfnog kriptosustava temelji se na približnom NZD problemu.
Uz to, dodavanje skupa $y = \{y_1, y_2, ..., y_\Theta\}$ u javni ključ navodi nas na pretpostavku koju je koristio Gentry, a koja se oslanja na problem zbrajanja prorijeđenih podskupova (*eng. sparse subset sum problem*). Problem zbrajanja skupova je veoma važan problem u teoriji složenosti i u kriptografiji. Problem se svodi na to da za dva dana skupa cijelih brojeva odredimo je li postoji neprazan podskup čiji je zbroj elemenata nula. Na primjer, za dani skup {-1, -3, -2, 5, 9}, odgovor je da postoji jer je zbroj elemenata podskupa {-3, -2, 5} jednak nuli. Ovaj problem je NP problem (nedeterminističko polinomijalni). NP problemi su poznati kao teško rješivi i unatoč intenzivnim naporima da se pronađu algoritmi u polinomijalnom vremenu za probleme u NP klasi, ovaj problem je još uvijek jedan od najvećih otvorenih pitanja u računarstvu. Ekvivalentan problem spomenutom problemu je za dani

skup cijelih brojeva i zadani cijeli broj s odrediti postoji li podskup čiji je zbroj elemenata jednak s. Kako bi izbjegli poznate napade na kriptosustav potrebno je izabrati dovoljno velik θ te Θ treba biti $\omega(\log \lambda)$ puta veći od duljine racionalnih brojeva u javnom ključu koja je jednaka κ.

Primjer

Neka je tajni ključ $p = 10001$ i $\kappa = 24$. Tada je

$$x_p = [2^{24}/p] = 1678.$$

Odaberimo proizvoljan 9-bitni vektor $\vec{s}$ s Hammingovom udaljenosti 3, $\vec{s} = \{0, 1, 0, 1, 0, 0, 0, 1, 0\}$ i neka je $S = \{2, 4, 8\}$. Sada biramo cijele brojeve $u_i \in \mathbb{Z} \cap [0, 2^{25}), i = 1, 2, ..., 9$ tako da vrijedi $\sum_{i \in S} u_i \equiv 1678 (\text{mod } 2^{25}) \equiv x_p (\text{mod } 2^{25})$. Neka je

$$\begin{aligned}
u_1 &= 281782, \\
u_2 &= 589103, \\
u_3 &= 1892147 \\
u_4 &= 1093482, , \\
u_5 &= 491831, \\
u_6 &= 487403, \\
u_7 &= 293813, \\
u_8 &= 31873525, \\
u_9 &= 5718711,
\end{aligned}$$

i uvjet je zadovoljen jer je $\sum_{i \in S} u_i = u_2 + u_4 + u_8 = 589103 + 1093482 + 31873525 \equiv 1678 (\text{mod } 2^{25}) \equiv x_p (\text{mod } 2^{25})$.
Računamo $y_i = u_i/2^{\kappa}, i = 1, 2, ..., 9$ i dobijemo sljedeće:

$$\begin{aligned}
y_1 &= 0.0167955, \\
y_2 &= 0.0351133, \\
y_3 &= 0.1127807, \\
y_4 &= 0.0651766, \\
y_5 &= 0.0293154, \\
y_6 &= 0.0290515, \\
y_7 &= 0.0175126, \\
y_8 &= 1.8998101, \\
y_9 &= 0.3408617,
\end{aligned}$$

gdje je $\sum_{i\in S} y_i = y_2+y_4+y_8 = 0.0351133+0.0651766+1.8998101 \equiv 2.0001 \approx 1/p(\text{mod } 2)$.

Dani šifrat $c^* = 300098$ je šifrirana 0. Da bi redešifrirali c^* računamo $z_i = c^* \cdot y_i(\text{mod } 2)$, $i = 1, 2, ..., 9$ i dobijemo:

$$\begin{aligned} z_1 &= 0.295959, \\ z_2 &= 1.4311034, \\ z_3 &= 1.2625086, \\ z_4 &= 1.3673068, , \\ z_5 &= 1.4929092, \\ z_6 &= 0.297047, \\ z_7 &= 1.4962348, \\ z_8 &= 1.2113898, \\ z_9 &= 1.9144466. \end{aligned}$$

Sada je $\vec{z} = \{z_1, z_2, ..., z_9\}$, a novi šifrat je $(c^*, \vec{z})$.
Kod dešifriranja računamo:

$$\begin{aligned} c^* - \sum_{i\in S} z_i &= 300098 - [z_2 + z_4 + z_8] \\ &= 300098 - [1.4311034 + 1.3673068 + 1.2113898] \\ &= 300098 - [4.0098] \\ &= 300094 \equiv 0(\text{mod } 2). \end{aligned}$$

Dešifrirani rezultat je isti kao i otvoreni tekst pa je primjer ispravan.

3.5 Bootstrap kriptosustav

Prilikom opisa donekle potpunih homomorfnih kriptosustava suočili smo se s problemom šuma koji se pojavljuje prilikom šifriranja i na taj način ograničava kriptosustav. Jedno od mogućih rješenja za smanjenje šuma upravo je *bootstrap*. Glavna ideja je šifrirati otvoreni tekst pomoću tajnog ključa, a zatim ga rešifrirati koristeći drugi par ključa. Jednom kad operacija dešifriranja otkloni šum šifrata, rezultirajući šum ovisit će o složenosti sklopa za šifriranje.
Homomorfni kriptosustav s javnim ključem $\mathcal{E}$ za razliku od standardnih kriptosustava može sadržavati četiri algoritma. Uz uobičajene algoritme za generiranje ključa, šifriranje i dešifriranje četvrti algoritam je algoritam procjene. Algoritam procjene na ulazu prima javni ključ pk, sklop C i t-torku šifrata $\vec{c} =< c_1, ..., c_t >$ (jedan za svaki ulazni bit od C), a na izlazu dobijemo novi šifrat c

Definicija 3.5.1. *Neka je $\mathcal{E}$ kriptosustav u kojem je dešifriranje implementirano pomoću sklopa koji ovisi samo o sigurnosnom parametru λ. To znači da za fiksiranu vrijednost sigurnosnog parametra, veličina tajnog ključa je uvijek ista što vrijedi i za veličinu šifrata. Za skup sklopova kažemo da je prošireni skup sklopova za dešifriranje ukoliko se sastoji od dva sklopa koja na ulazu primaju tajni ključ i dva šifrata. Prvi sklop dešifrira oba šifrata te zbraja bitove dobivenih otvorenih tekstova modulo 2, dok drugi sklop umjesto zbrajanja množi dobivene otvorene tekstove modulo 2. Dani skup označit ćemo sa $D_{\mathcal{E}}(\lambda)$.*

Definicija 3.5.2. *Neka je $\mathcal{E}$ homomorfni kriptosustav. Za $\mathcal{E}$ kažemo da je bootstrap ako njegovi prošireni sklopovi za dešifriranje prihvaćaju svaku vrijednost sigurnosnog parametra λ.*

Teorem 3.5.3. *Potisnuti kriptosustav je bootstrap.*

Dokaz teorema može se pronaći u [1].

Kako bi smanjili veličinu šifrata tijekom procjene, van Dijk i suradnici dodali su u javni ključ više elemenata oblika $x'_i = q'_i p + 2r_i$ gdje je r_i izabran kao i obično iz intervala $(2^{-\rho}, 2^{\rho})$, ali su q_i odabrani mnogo veći nego ostali elementi iz javnog ključa. Posebno, za $i = 0, 1, ..., \gamma$ $q'_i \in \mathbb{Z} \cap [2^{\gamma+i-1}/p, 2^{\gamma+i}/p)$, $r_i \in \mathbb{Z} \cap (2^{-\rho}, 2^{\rho})$, $x'_i = 2(q'_i \cdot p + r_i)$ dobivamo $x'_i \in [2^{\gamma+i}, 2^{\gamma+i+1}]$.

Tijekom algoritma procjene, svaki put kada imamo šifrat koji raste iznad 2^{γ}, prvo ga smanjimo sa modulo x'_{γ}, zatim sa modulo $x'_{\gamma-1}$ i tako dalje dok ne dođemo do x'_0. Sada ponovno imamo šifrat duljine koja nije veća od γ. Podsjetimo se da je za povećanje duljine šifrata dovoljna samo jedna operacija. Nakon bilo koje operacije šifrat ne može biti veći od $2x'_{\gamma}$ pa slijed modularnih smanjenja uključuje mala množenja sa x'_i, što znači da se stvaraju samo male količine šumova.

Koraci konstruiranja potpuno homomorfnog kriptosustava

Potpuni homomorfni kriptosustav dopušta izvođenje zbrajanja i množenja na šifratima, a prilikom dešifriranja rezultata dobivamo jednaku vrijednost kao da smo primjenili zbrajanje i množenje na otvorenim tekstovima. Sam kriptosustav provodi se u nekoliko koraka. Prvi korak je konstruiranje donekle potpuno homomorfnog kriptosustava koji podržava ograničen broj množenja. Drugi korak je potiskivanje procesa dešifriranja povezanog s proizvoljnim šifratom tako da se može izraziti kao polinom niskog stupnja u tajnom ključu. Zatim primjenimo ključnu ideju nazvanu *bootstrap* koja rezultira novim šifratom povezanim s istim otvorenim tekstom, ali sa smanjenom količinom šuma. Novi "osvježeni" šifrat kasnije koristimo u sljedećim homomorfnim operacijama. Konstantnim "osvježavanjem" šifrata, broj homomorfnih operacija postaje neograničen što rezultira potpuno homomorfnim kriptosustavom. Primarni otvoreni problem je poboljšati učinkovitost kriptosustava, u mjeri u kojoj je to moguće, uz očuvanje težine problema približnog NZD.

Bibliografija

[1] Dijk, M. van, C. Gentry, V. Halevi i V. Vaikuntanathan: *Fully homomorphic encryption over the integers*. Proceedings of Advances in Cryptology, EUROCRYPT'10, 24–43.

[2] Dujella, A. Maretić, M.: *Kriptografija*. Element, 2007.

[3] Howgrave-Graham: *Approximate integer common divisors*. CaLC '01, volume 2146 of Lecture Notes in Computer Science, Springer, stranice 51–66, 2001.

[4] Lindell (Ed.), Y.: *Tutorials on the Foundations of Cryptography*. Springer, 2017.

[5] Malidžan, V. i Đaković B.: *Homomorfna enkripcija*. `http://infoteh.etf.unssa.rs.ba/zbornik/2012/radovi/RSS-5/RSS-5-1.pdf`.

[6] Pettersen, N.: *Applications of Paillier s Cryptosystem.* Magistarska radnja, NTNU, 2016.

[7] Pisa, P.S, M. Abdalla i O.C.M.B Duarte: *Somewhat homomorphic encryption scheme for arithmetic operations on large integers*. U *Global Information Infrastructure and Networking Symposium (GIIS), 2012*, stranice 1–8. IEEE, 2012.

[8] Savić, D.: *Jedna klasa sistema zaštite u računarskom oblaku zasnovana na homomorfnim šiframa*. Disertacija, 2010.

[9] Širola, B.: *Algebarske strukture*. `https://web.math.pmf.unizg.hr/nastava/alg/predavanja.php`.

[10] Yi, X., R. Paulet i E. Bertino: *Homomorphic Encryption and Applications*. Springer, 2014.

Sažetak

Ovaj rad proučava homomorfne kriptosustave koji omogućavaju obavljanje matematičkih operacija nad šifriranim podacima bez potrebe da se podaci prije toga dešifriraju. Rad se može promatrati kroz tri osnovne cjeline. Prvi dio rada odnosi se na prvo poglavlje u kojem smo definirali homomorfizam i homomorfne kriptosustave.

Drugi dio rada opisuje djelomične homomorfne kriptosustave koji zadovoljavaju svojstvo homomorfizma samo nad nekim matematičkim operacijama.Takvih kriptosustava ima više, a obrađeni u ovom radu su RSA, ElGamalov i Paillierov kriptosustav. Svaki kriptosustav prikazan je na isti način u više koraka, a to su: generiranje ključa, šifriranje, dešifriranje, svojstvo homomorfizma i sigurnost. Na kraju svakog kriptosustava naveden je primjer u kojem je ilustrirano njihovo izvođenje.

Treći dio rada odnosi se na potpune homomorfne kriptosustave koji zadovoljavaju svojstvo homomorfizma nad svim matematičkim operacijama. Prvo ćemo definirati samo značenje potpunog homomorfnog kriptosustava, a zatim prikazati njegov povijesni razvoj. Na samom kraju pokazat ćemo kako preko jednostavnijih donekle potpunih doći do potpunih homomorfnih kriptosustava uz objašnjenja njihovih prednosti i nedostataka.

Summary

This thesis studies homomorphic encryption which allows specific types of computations to be carried out on ciphertexts and generate an encrypted result which, when decrypted, matches the result of operations performed on the plaintexts. It may be divided into three fundamental parts. The first part includes the first chapters where we have defined homomorphism and homomorphic encryption.

In the second part we describe partially homomorphic cryptosystems which allows only one operation is possible (usually addition or multiplication in the plaintext space) to be able to manipulate the plaintext by using only the ciphertext. There is a lot of partially homomorphic cryptosystems, but in this thesis processed are RSA, ElGamal and Paillier's cryptosystems. Every encryption scheme consists of the same components: the key generation, the encryption algorithm, the decryption algorithm, homomorphic property and security. At the end, each cryptosystem is confirmed with example which illustrate their execution.

The third part of the thesis refers to the fully homomorphic encryption which allows one to evaluate both addition and multiplication of plaintext, while remaining encrypted. First we described fully homomorphic encryption in general and introduce some of its properties. Then we gave a historical overview. At the end we showed connection between simple somewhat encryption sheme and fully homomorphic encryption sheme with all the advantages and disadvantages.

Životopis

Dana 29.listopada 1993. rođena sam u Šibeniku. Djetinjstvo vežem za Biograd na Moru, gdje sam živjela, odlazila u vrtić i završila osnovnu i srednju školu. Još tijekom školovanja u osnovnoj školi otkrila sam svoje zanimanje za matematiku. Od nižih razreda sudjelujem na državnim natjecanjima iz matematike, gdje ostvarujem svoj najbolji plasman, 3. mjesto. Opću gimanziju završila sam u Biogradu na Moru, a 2012. godine selim se u Zagreb i upisujem Prirodoslovno-matematički fakultet, smjer Matematika. Titulu sveučilišne prvostupnice stekla sam 2015. kada upisujem diplomski studij Matematike i računarstva na istom fakultetu. Zahvaljujući stečenom znanju tijekom studija još kao studentica započinjem s radom u kompaniji Ericsson Nikola Tesla .