UNIVERZITET CRNE GORE

Prirodno-matematički fakultet Podgorica

Marijana Jakić

Elgamal algoritam

Specijalistički rad

Podgorica, 2015.

UNIVERZITET CRNE GORE

Prirodno-matematički fakultet Podgorica

Elgamal algoritam

Specijalistički rad

Kriptografija

Mentor: dr Vladimir Božović

Marijana Jakić

Matematika i računarske nauke

Podgorica, maj 2015.

Apstrakt

U znamenitom radu koji je objavljen 1985. godine, u IEEE Transactions on Information Theory, Taher Elgamal je predstavio novi asimetrični algoritam enkripcije. Elgamalov kriptografski algoritam je matematički zasnovan na protokolu koji su 1976. godine predstavili Whitfield Diffie i Martin Hellman. Kao i Difi-Helman protokol, Elgamal enkripcija počiva na kompleksnosti problema diskretnog logaritma. Elgamal algoritam je ostvario značajan uticaj na razvoj kriptografije. Poslužio je kao model za razvoj mnogih drugih algoritama, među kojima je i Pretty Good Privacy kriptosistem.

Abstract

In the famous work, which was published in 1985, in IEEE Transactions on Information Theory, Taher Elgamal has introduced a new asymmetric encryption algorithm. Elgamal cryptographic algorithm is mathematically based on the protocol which was presented by Whitfield Diffie and Martin Hellman in 1976. As Diffie-Hellman protocol, Elgamal encryption is based on the complexity of the problem of discrete logarithm. Elgamal algorithm has achieved significant influence on the development of cryptography. It served as a model for the development of many other algorithms, including the Pretty Good Privacy cryptosystem.

Sadržaj

Glava 1

Uvod

Kriptografija se bavi izučavanjem i konstrukcijom metoda sigurne razmjene poruka. Slobodnije rečeno, to je vještina pisanja i čitanja skrivenih poruka. Koriste se standardni jezički simboli kojima je izmjenjeno značenje, tako da se poruka ne može lako, ili ne može uopšte, pročitati bez odgovarajućeg ključa. Donedavno, usluge kriptografa i kriptoanalitičara naručivala je uglavnom država. Sa razvojem interneta i sve veće upotrebe informacionih i telekomunikacionih tehnologija, krug korisnika takvih usluga se proširio. Postalo je jasno da se veća pažnja mora posvetiti sigurnosti informacionih sistema, zaštiti podataka od neovlašćenog pristupa, modifikacija ili drugih zloupotreba, i da je neophodno stvoriti određene mehanizme koji će računarsku mrežu učiniti sigurnom.

U ovom radu su definisani osnovni pojmovi i terminologija u kriptografiji čime je napravljena dobra osnova da čitalac koji nema znanje iz oblasti kriptografije, može razumjeti obrađene teme. Elgamal algoritam je asimetrični kriptografski algoritam utemeljen na problemu diskretnog logaritma i u suštini predstavlja nadogradnju Difi-Hellman protokola za uspostavljanje tajnog ključa.

Kako bi pojednostavili izražavanje i lakše formulisali različite scenarije koji se javljaju u radu, uvešćemo jedan broj apstraktnih subjekata ili *likova* sa određenom ulogom u procesu razmjene poruka. Glavni likovi u kriptografskom scenariju su **Alisa,**

Bob i Oskar. Alisa i Bob razmjenjuju tajne poruke. Podrazumjevaćemo da Alisa šalje poruku, Bob je prima, dok je Oskar analitičar koji posmatra razmjenu poruka i nastoji da ovlada makar dijelom njihovog sadržaja.

Glava 2

Osnovni pojmovi i terminologija u kriptografiji

Kriptografija je naučna disciplina koja se bavi proučavanjem i konstrukcijom metoda, uglavnom matematičkih, za zaštitu i očuvanje tajnosti podataka.

U Engleskoj literaturi termini *plaintext* ili *cleartext* odnose se na **izvornu poruku** ili **izvorni tekst** koji je moguće pročitati i razumjeti bez primjene bilo kakvih posebnih metoda. Ukoliko Alisa šalje Bobu određen tekst, onda se to naziva **poruka**. Primjenom nekog algoritma kojim se izvorni tekst transformiše u neki oblik koji je nečitljiv za trećeg subjekta u klasičnom kriptografskom scenariju, Oskara, dobija se **šifrovan** ili **kriptovan tekst** (eng. *ciphertext* ili *cipher*).

Postupak pomoću koga se izvorni tekst transformiše u šifrovan tekst se naziva **enkripcija** ili **šifrovanje** (eng. *encryption*). Enkripcija izvornog teksta se obavlja pomoću određene procedure za enkripciju, odnosno kriptografskog algoritma.

Postupak koji omogućava da se od šifrovanog teksta dobije originalni, izvorni tekst naziva se **dekripcija** ili **dešifrovanje** (eng.*decryption*). Dekripcija, dakle, predstavlja suprotni postupak od enkripcije, a to je transformacija šifrovanih podataka u početni, smisleni oblik.

Da bi se realizovala enkripcija i dekripcija potrebno je iskoristiti jedan poseban podatak koji se obično čuva u tajnosti i zove se **ključ**.

Svaki kriptografski algoritam kao ulazne podatke ima izvorni tekst i ključ, a kao izlaz daje kriptovani tekst.

Kriptosistem čine kriptografski algoritam, skup svih mogućih, izvornih i kriptovanih poruka, skup ili *domen* ključa.

2.1 Kriptografija sa javnim ključem

Algoritmi za enkripciju se mogu podjeliti u dvije grupe:

1. **Tajni algoritmi:** bezbjednost se zasniva na tajnosti algoritma.

2. **Algoritmi zasnovani na ključu:** bezbjednost se zasniva na tajnosti ključeva, a ne na detaljima algoritma koji se može publikovati i analizirati. Ovdje je algoritam poznat javnosti (Kerkohov princip), a ključ se čuva u tajnosti.

Danas se najvise koriste algoritmi za enkripciju zasnovani na ključu, a mogu se klasifikovati u tri grupe:

1. **Simetrični**, kod kojih Alisa i Bob koriste jedan ključ;

2. **Asimetrični**, kod kojih Alisa i Bob koriste dva ključa;

3. **Hibridni**, kombinacija prethodna dva.

2.2 Simetrični kriptosistemi

Osnovna osobina simetričnih kriptosistema je da se za enkripciju i dekripciju poruka koristi **isti ključ**. Jedan od osnovnih problema ovakvih kriptosistema je kako uspostaviti zajednički ključ između Alise i Boba. Dakle, potreban je neki oblik *sigurnog* komunikacionog kanala preko koga bi u nekoj, početnoj fazi, dva subjekta koja

komuniciraju uspostavila zajednički ključ. Najpoznatiji algoritmi simetričnih kriptosistema koji se danas koriste su: DES, 3DES, DES-CBC, IDEA, RC5, RC6, AES i drugi. [3]

2.3 Asimetrični kriptosistemi

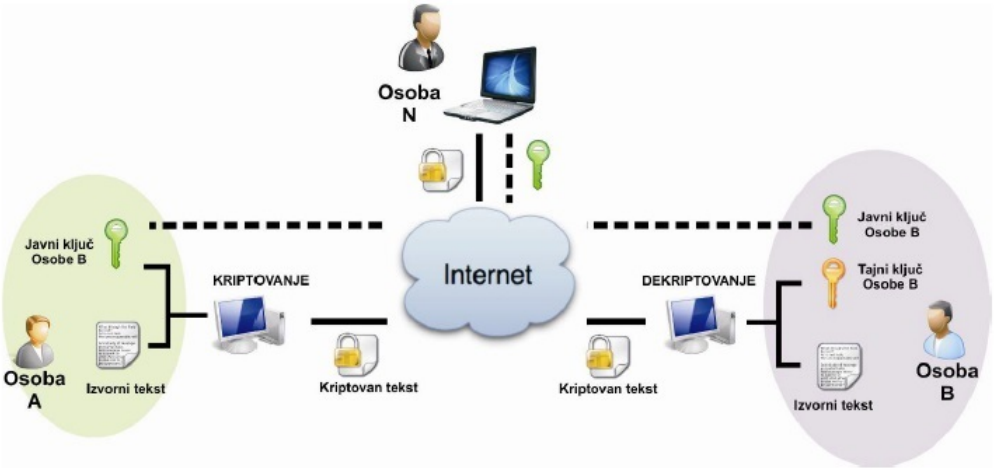
Može se reći da su *Whitefield Diffie* i *Martin Hellman* dali ključan doprinos u razvoju asimetrične kriptografije, kada su 1976. godine u svom radu *"New Direction in Cryptography"* opisali ideju kriptografije koja se temelji na dva ključa, tajnom i javnom ključu. Ova dvojica naučnika su potom napravili konkretan algoritam sigurne razmjene ključeva, a 1977. godine Rivest, Šamir i Aldeman su realizovali i petentirali čuveni RSA algoritam. U literaturi pojam asimetrične enkripcije se poistovjećuje sa terminom *asymmetric-key* ili *public-key* enkripcijom.

Suština je da informacije koje su enkriptovane javnim ključem mogu se dekriptovati samo tajnim ključem, odnosno to može uraditi samo osoba koja je vlasnik tajnog asimetričnog ključa. Oba ključa moraju biti povezana pomoću neke *jednosmjerne* funkcije.

Algoritmi asimetričnih kriptosistema zasnivaju se na određenim svojstvima brojeva. Pri enkripciji se izvorni tekst tretira kao niz prirodnih brojeva koji se odabranom funkcijom enkripcije i ključem $\mathbf{K_e}$ preračunavaju u šifrovani. Funkcija mora biti takva da se iz šifrovanog teksta ne može odrediti izvorni tekst, čak i ako je poznat ključ za enkripciju. Međutim, ukoliko se zna ključ dekripcije $\mathbf{K_d}$ moguće je lako računanje izvornog teksta. Stoga, asimetrična enkripcija predstavlja kompleksan vid zaštite podataka.

2.3.1 Postupak asimetrične enkripcije

Alisa enkriptuje poruku, radi slanja Bobu, upotrebom Bobovog javnog ključa koji je svima dostupan (čak i Oskaru). Alisa je javni ključ mogla dobiti putem imejla, preuzeti ga sa veb-sajta i slično. Poznavanje javnog ključa ne pomaže Oskaru u otkrivanju sadržaja poruke. Kao što smo naglasili, poruku može dekriptovati samo Bob korišćenjem svog tajnog ključa. Na slici je prikazan tok asimetrične enkripcije [1]:



Slika 2.1: Postupak asimetričnog kriptovanja

Osnovni nedostatak asimetrične enkripcije je sporost, što naročito dolazi do izražaja u slučaju razmjene velikih količina podataka. Takođe, ostaje otvoreno pitanje autentičnosti učesnika u protokolu, odnosno same poruke. Drugim riječima, kako Bob može biti siguran da je poruku koju je primio, uistinu poslala Alisa.

Najpoznatiji asimetrični algoritmi enkripcije su: RSA, Elgamal, Cramer Shoup, NTRUEncrypt...

Glava 3

Problem diskretnog logaritma (DLP)

Definicija 1. ***Logaritam realnog broja*** $a > 0$, *za zadatu bazu* $0 < b \neq 1$, *je realan broj* x *kojim treba stepenovati bazu logaritma da bi dobili broj* a, *tj.*

$$\log_b a = x \Leftrightarrow b^x = a$$

Posmatrajmo sad $\log_b a = x$, ali takav da su $a, b, x \in \mathbb{Z}$, $0 < b \neq 1, a > 0$. Ovaj logaritam je definisan na skupu cijelih brojeva $\mathbb{Z}$ i on može, ali i ne mora da postoji. Tako, na primjer $\log_2 8 = 3$, jer je $2^3 = 8$, međutim $\log_2 7$ nema riješenja, jer ne postoji cijeli broj kojim bi stepenovali 2 i dobili vrijednost 7. Neka su data dva prirodna broja b i n takvi da je $b < n$. Uzmemo li proizvoljan prirodan broj $a < n$, cilj nam je da nađemo broj x za koji važi:

$$b^x \equiv a \pmod{n}$$

Tako, za $b = 2$, $n = 25$ i $a = 7$, imamo $2 \equiv 7 \mod 25$, pa je traženi broj jednak 5.

Definicija 2. ***Problem diskretnog logaritma****: Neka je* g *primitivni korijen (generator) multipilikativne grupe konačnog polja* $\mathbb{Z}_p$ *(ostaci po modulu prostog broja* p*) i neka je* $h \in \mathbb{Z}_p$ *takav da* $h \neq 0$. *Problem diskretnog logaritma (DLP) je problem pronalaženja stepena* x *tako da:*

$$g^x \equiv h \pmod{p}$$

Taj broj x se naziva diskretni logaritam od h po osnovi g i označava sa $\log_g h$. [4]

Primjer: Za prost broj $p = 56509$, lako se provjerava da je $g = 2$ primitivni korijen $\mathbb{Z}_p$. Neka je $h = 38679$. Jedan prost i očigledan način da riješimo dati DLP

$$g^x \equiv h \pmod{p}$$

je da računamo sve stepene redom:

$$2^2, 2^3, 2^4, 2^5, 2^6, 2^7, \ldots \mod 56509$$

dok ne nađemo onaj koji daje 38679. Ovo je metod direktnog, *iscrpljujućeg pretraživanja*. Očigledno, ovo je vrlo nepraktičan metod, naročito ako se radi o velikim brojevima koji učestvuju u problemu DLP. Koristeći računar, nalazimo da je $\log_p(h) = 11235$.

Algoritmi za riješavanje DLP

Algoritmi za riješavanje problema diskretnog logaritma (DLP) mogu se svrstati u jednu od tri kategorije:

1. Algoritmi koji rade u proizvoljnim grupama, odnosno oni koji ne koriste nijedno specifično svojstvo grupe. Na primjer, to su: iscrpljujuće pretraživanje, *Šanksov (Shanks) algoritam*, *Polard ρ (Pollard-rho)*...

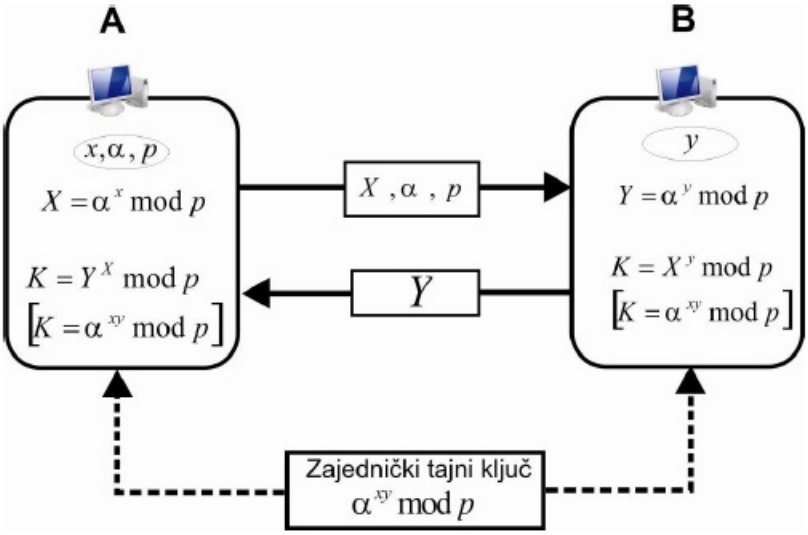
2. Algoritmi koji dobro rade u grupama *glatkog reda*, odosno grupama čiji red nema velike proste faktore. Na primjer, to je *Silver-Polig-Helman (Silver-Pohlig-Hellmanov)* algoritam.

3. Algoritmi koji se zasnivaju na metodama predstavljanja elemenata grupe kao proizvoda elemenata iz relativno malih skupova, takozvanih f*aktorskih baza*. Klasični predstavnici ove kategorije su algoritmi koji su varijacije *indeks kalkulus (Index calculus)* metode. [5]

3.1 Difi-Helman (DH) protokol

Kao što je već istaknuto, Difi i Helman su u svom znamenitom radu "New Directions in Cryptography", predložili algoritam za uspostavljanje javnog ključa između Alise i Boba. Pod pretpostavkom da Oskar ne može da riješi DLP, protokol zadovoljava osnovne sigurnosne kriterijume. Predstavljamo školsku verziju ovog protokola koja ne obuhvata neke važne sigurnosne aspekte neophodne za praktičnu upotrebu.

Na početku izvršavanja protokola, Alisa i Bob uspostavljaju javno prost broj p, kao i α, primitivni korijen polja $\mathbb{Z}_p$.



Slika 3.1: Difi Helman protokol

Metodom slučajnog izbora broja Alisa određuje x iz skupa brojeva $\{1, 2, \ldots, p-1\}$.

Potom, Alisa, preko nezaštićenog kanala, šalje Bobu :

$$X = \alpha^x \pmod{p}.$$

Kada primi poruku i sazna X, α i p, Bob takođe na osnovu slučajnog izbora određuje $y \in \{1, 2, \ldots, p-1\}$, a potom računa broj Y pomoću formule:

$$Y = \alpha^y \pmod{p}.$$

Ovako dobijeni broj šalje Alisi. Istovremeno na osnovu podataka koje je dobio od Alise i broja y (koga je sam generisao) računa vrijednost ključa K po formuli:

$$K = Y^x \pmod{p}.$$

Kako je $X^y = \alpha^{xy} = Y^x \pmod{p}$ to je zajednički tajni ključ $K = \alpha^{xy} \pmod{p}$. Na ovaj način Alisa i Bob dolaze do zajedničkog ključa, koga nadalje koriste za enkripciju odnosno dekripciju međusobnih poruka u nekom simetričnom kripto sistemu koji odaberu.

Ukoliko Oskar želi da dođe u posjed zajedničkog tajnog ključa, znajući X, α i p, prinuđen je da rješava diskretni logaritam. Dakle, sigurnost DH protokola uspostavljanja zajedničkog ključa je zasnovana na težini rješavanja DLP. Naročito, u slučaju velikih prostih brojeva [3], problem postaje dovoljno težak i za savremena računska sredstva i algoritme.

Na osnovu ovog protokola, po ugledu na DLP možemo definisati i poseban oblik problema, tako zvani Difi-Helman problem (DHP). Naime, DHP se sastoji u problemu nalaženja $\alpha^{st} \pmod{p}$ ako su dati α^s i $\alpha^t \pmod{p}$.

Glava 4

Elgamal algoritam

U ovom poglavlju izlagaćemo osnovnu, školsku verziju Elgamal kriptografskog sistema. Kao što je ranije naglašeno, Elgamal enkripcija se u izvjesnom smislu može posmatrati kao ekstenzija DH protokola. Elgamal algoritam ćemo posmatrati u multiplikativnoj grupi $\mathbb{Z}_p^*$, gdje je p prost broj, iako se može analogno primjeniti i u drugim cikličnim grupama. Na primjer, u multiplikativnoj grupi Galoa polja $GF(p^m)$.

Svakako, suština Elgamal algoritma, odnosno njegovo razumijevanje je olakšano nakon upoznavanja sa DH protokolom.

4.1 Opis Elgamal algoritma

Razlikujemo tri faze u Elgamal algoritmu.

Formiranje javnog i tajnog ključa: Bob definiše prost broj p, primitivni korijen α multiplikativne grupe $\mathbb{Z}_p^*$ kao javni ključ. Potom bira na slučajan način broj d iz $(2, ..., p-1)$ koji čuva kao tajni, privatan ključ. Zatim, Bob računa vrijednost $\alpha^d \pmod p$ i to dodaje prethodnom skupu javnih podataka u okviru javnog ključa. Dakle, u Elgamal algoritmu, sa Bobove strane, imamo:

Bobov tajni ključ: d.

Bobov javni ključ: p, α, $\beta = \alpha^d \pmod p$.

Enkripcija:

Alisa bira slučajan broj $i \in (2, \ldots, p-2)$ i računa dvije vrijednosti, takozvani efemeralni ključ K_e i tajni ključ K_m:

$$K_e = \alpha^i \pmod{p}$$

$$K_m = \beta^i \pmod{p}$$

Zatim, Alisa maskira, odnosno enkriptuje poruku $x \in \mathbb{Z}_p^*$ na sledeći način:

$$y \equiv x \cdot K_m \pmod{p}$$

Na kraju, Alisa šalje Bobu šifrovan tekst u obliku uređenog para (K_e, y), koji se sastoji iz dva dijela: efemeralnog ključa i enkriptovane poruke.

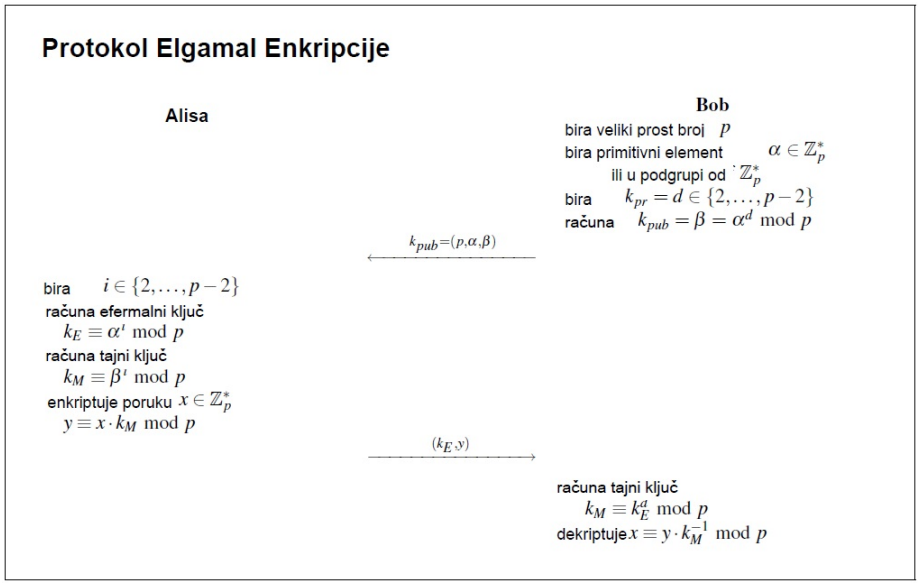
Dekripcija: Koristeći privatni ključ, Bob računa:

$$K_m \equiv K_e^d \pmod{p}$$

Dakle, preostalo je da Bob nađe inverz od K_m po modulu p i pomnoži sa y, drugom komponentom uređenog para koji je dobio. Znači, Bobova dekripcije se sastoji u sledećem:

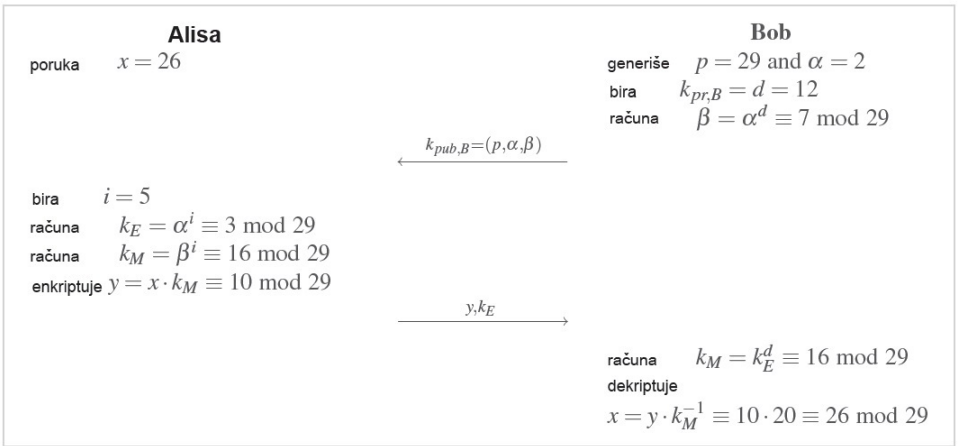
$$x \equiv y \cdot K_m^{-1} \pmod{p}$$

Očigledno, **faza enkripcije** i **faza dekripcije** su pravolinijski. Nešto manje jasno je kako Bob generiše primitivni korijen α po modulu prostog broja p. Naime, ne postoji polinomijalni algoritam za generisanje primitivnih korijena po modulu datog prostog broja. Zapravo, ne postoji ni efektivan algoritam koji provjerava da li je dati broj $h \in \mathbb{Z}_p^*$ primitivan korijen multiplikativne grupe. Iako ne postoji efektivan algoritam za generisanje Elgamalovog javnog ključa, u praksi se to prevazilazi na različite načine. [7]



Slika 4.1: Protokol Elgamal enkripcije

Bitno je primjetiti da Elgamal ima i dijelom vjerovatnosni karakter. Enkripcija dvije identične poruke x_1 i x_2, gdje su x_1, $x_2 \in Z_p^*$ koristeći isti javni ključ (sa veoma velikom vjerovatnoćom) su dva različita šifrovana teksta $y_1 \neq y_2$. To je zato što se Alisin izbor eksponenta, i za efemeralni ključ bira slučajno, $i \in \{2, 3, \ldots, p-2\}$ za svaku pojedinačnu enkripciju.



Slika 4.2: Primjer sa malim brojevima

Jedan od ozbiljnih nedostataka Elgamal algoritma je u tome što je veličina enkriptovane poruke dva puta veća od osnovne, što značajno utiče na brzinu i efikasnost ovog algoritma.

4.2 Računarski aspekt

Generisanje ključa: Kako se sigurnost Elgamala bazira na problemu diskretnog logaritma, p mora da ima dužinu najmanje 1024 bita. Tajni ključ se generiše pomoću generatora slučajnih brojeva. Javni ključ se testira zahtjevnim stepenovanjem za koji se koristi algoritam *brzog stepenovanja*.

Enkripcija: U okviru postupka enkripcije, potrebno je izvršiti jedno stepenovanje i jedno množenje po modulu kako bi se izračunao efemeralni ključ i tajni ključ, kao i za samu enkripciju poruke. Svi pomenuti operanti imaju dužinu $\log_2 p$ bita. Bitno je primjetiti da su stepenovanja nezavisna od šifrovanog teksta. Otud, u nekim primjenama one se mogu izvršiti u razdobljima niske računarske opterećenosti, uskladištiti i iskoristiti kada su potrebni za stvarnu enkripciju. Ovo može biti velika prednost u praksi.

Dekripcija: Glavni korak u dekripciji je prvo stepenovanje $K_m = K_e^d \pmod p$ koristeći algoritam *brzog stepenovanja i jedno množenje.*, zatim slijedi inverzija od K_m koja se radi pomoću proširenog Euklidovog algoritma. Međutim postoji prečica koristeći Malu Fermaovu teoremu koja kombinuje dva koraka u jedan. Na osnovu ove

teoreme slijedi:

$$\begin{aligned} K_m^{-1} &\equiv (K_e^d)^{-1} \pmod{p} \\ &\equiv (K_e^d)^{-1} K_e^{p-1} \pmod{p} \\ &\equiv K_e^{p-d-1} \pmod{p} \end{aligned}$$

Poslednja ekvivalencija omogućava nam da izračunamo inverz tajnog ključa pomoću jednog stepenovanja sa $p - d - 1$. Nakon toga ide jedno množenje po modulu kako bi izračunali $x \equiv y \cdot K_m^{-1} \pmod{p}$. Dakle dekripcija zahtjeva jedno izvršavanje algoritma *brzog stepenovanja i jedno množenje*, nakon čega slijedi jedno množenje po modulu kako bi dobili izvornu poruku.

4.3 Sigurnost

Kako bi procijenili sigurnost Elgamal algoritma bitno je napraviti razliku između **pasivnih** napada (prisluškivanje) i **aktivnih** napada koji omogućavaju Oskaru da eventualno generiše nove i mijenja postojeće poruke.

4.3.1 Pasivni napad

Sigurnost Elgamal algoritma u odnosu na pasivan napad, dobijanja poruke x na osnovu javno dostupnih informacija p, α, $\beta = \alpha^d$, $K_e = \alpha^i$ i $y = x \cdot \beta^i$ bazira se na kompleksnosti DHP problema. U slučaju da Oskar zna da riješi DHP, onda će iz K_e i β lako dobiti K_m nakon čega mu je otvoren put do dekripcije.

Trenutno, ne postoji drugi metod za riješavanje DHP problema sem direktnog računanja diskretnih logoritama. Ako pretpostavimo da Oskar ima moć da riješi

DLP, onda je put do dekripcije bilo koje poruke vrlo jednostavan.

Da nađe x, tada bi otkrio Bobov tajni ključ d:

$$d = \log_\alpha \beta \pmod{p},$$

Oskar može da dekriptuje šifrovan tekst radeći isto što radi i Bob:

$$x \equiv y \cdot (K_e^d)^{-1} \pmod{p}$$

Kao alternativu, umjesto računanja Bobovog tajnog stepena d, Oskar može pokušati da otkrije Alisin slučajni stepen i, u efemeralnom ključu:

$$i = \log_\alpha k \pmod{p}$$

U ovom slučaju, Oskar će dešifrovati poruku kao:

$$x \equiv y(\beta^i)^{-1} \pmod{p}$$

Kao i kod DH protokola onaj koji postavlja javne parametre sistema mora biti obazriv da ne upadne u zamku *napada male podgrupe*. Naime, potrebno je obezbijediti da element α bude generator velike grupe, jer samo tako DLP ima potrebnu snagu.

4.3.2 Aktivni napad

Kao i kod bilo kog asimetričnog algoritma moramo pretpostaviti da je javni ključ autentičan, odnosno da pripada upravo subjektu sa kojim želimo da razmjenjujemo poruke. To znači da enkripciona strana, u našem slučaju Alisa, je uvjerena da javni ključ koji je preuzela sa veb sajta zaista pripada Bobu. Ako Oskar uspije da ubijedi Alisu da je njegov ključ u stvari Bobov, može lako da napadne algoritam. To je opšti problem *autentifikacije* koji, kao što smo rekli, nije karaktističan samo za Elgamal već ima opšti kriptografski značaj.

Još jedna potencijalna slabost Elgamal enkripcije koju Oskar može iskoristiti je ukoliko Alisa koristi više puta isti efemeralni ključ.

Pretpostavimo da Alisa koristi isti eksponent i za enkripciju dvije uzastopne poruke x_1 i x_2. U tom slučaju dva tajna ključa bi bila ista, odnosno $K_m = \beta^i$, kao i dva efemeralna ključa. Alisa šalje dva šifrovana teksta (K_e, y_1) i (K_e, y_2) preko neosiguranog kanala. Ako Oskar dođe u posjed dekripcije jedne od poslatih poruka, na primjer poruke x_1 koja odgovara y_1, onda može i da izračuna tajni ključ kao $K_m \equiv y_1 \cdot x_1^{-1} \pmod{p}$. Jasno, pomoću ovog može da dekriptuje x_2 koji odgovara y_2. Kao posledica ovog napada očigledna je potreba za stalnom promjenom efemeralnog ključa.

4.4 Prednosti i nedostaci Elgamal algoritma

Ukratko, moglo bi se reći da Elgamal dijeli prednosti ostalih asimetričnih kripto sistema ali i njihove nedostatke.

Na primjer, Elgamal algoritam, kao i drugi asimetrični kriptosistemi, može biti efikasan u pogledu smanjenja ukupnog broja potrebnih ključeva u nekom višeentitetskom protokolu. Na primjer, u sistemu u kome komunicira milion korisnika, ako se koristi asimetrični kripto sistem, potrebno je samo 2 miliona ključeva, dok bi u slučaju korišćenja simetričnog kripto sistema bilo potrebno bar 500 milijardi ključeva.

Najveći nedostatak Elgamal algoritma je kompleksnost različitih izračunavanja tokom algoritma, potom korišćenje ogromnih ključeva tokom rada, što zahtjeva mnogo vremena i računarskih kapaciteta. Posebno, nepostojanje efikasnog algoritma za traženje generatora multiplikativne grupe $\mathbb{Z}_p$, po modulu prostog broja p, predstavlja ozbiljan problem. Takođe, enkripcija Elgamal algoritmom množi sa faktorom 2 veličinu izvornih poruka, što rad sa velikim izvornim podacima čini vrlo zahtjevnim. Može se reći da je algoritam efikasan samo u radu sa kratkim porukama.

Kao što smo naglasili, nedostatak ovog kripto algoritma je i taj što se komunikacija između dvije strane, mora u određenom smislu autentifikovati. Ovo je, kao što smo naveli, opšti problem u asimetričnoj enkripciji, a ne isključiva specifičnost Elgamala.

Glava 5

Elgamal šema digitalnog potpisa

Iako nosi isto ime kao originalni kripto sistem, šema Elgamal digitalnog potpisa se prilično razlikuje od Elgamal algoritma enkripcije.

5.1 Školski primjer Elgamal digitalnog potpisa

Generisanje ključeva: Počinjemo odabirom velikog prostog broja p i konstruisanjem problema diskretnog logaritma. Biramo i primitivni elemenat $\alpha \in \mathbb{Z}_p^*$ kao i slučajan broj d iz skupa $\{2, 3, \ldots, p-2\}$. Zatim računamo $\beta = \alpha^d \pmod{p}$. Time smo formirali javni ključ $k_{pub} = (p, \alpha, \beta)$ i privatni kluč $k_{pr} = d$.

Potpis i verifikacija: Korišćenjem privatnog ključa i parametara javnog kluča računa se potpis za poruku x:

$$sig_{k_{pr}}(x, K_e) = (r, s)$$

Primjetimo da se potpis sastoji od dva cijela broja r i s. Potpisivanje se sastoji iz dva glavna koraka:

1. Slučajnog odabira vrijednosti K_e, koja u stvari predstavlja privatni efemeralni ključ

2. Računanja stvarnog potpisa za poruku x

Generisanje Elgamal potpisa

1. Izaberi slučajan efemeralni ključ $K_e \in \{0, 1, 2, \ldots, p-2\}$ za koje važi da je $nzd(K_e, p-1) = 1$

2. Izračunaj parametre potpisa:

$$r \equiv \alpha^{K_e} \pmod{p}$$

$$s \equiv (x - d \cdot r) K_e^{-1} \pmod{p} - 1$$

Na prijemnoj strani , potpis je potrebno verifikovati korišćenjem javnog ključa, potpisa i poruke.

Verifikacija Elgamal potpisa

1. Izračunaj vrijednost:

$$t \equiv \beta^r \cdot r^s \pmod{p}$$

2. Verifikacija slijedi iz:

$$t \equiv \alpha^x \pmod{p} \longrightarrow \text{potpis validan}$$

$$t \not\equiv \alpha^x \pmod{p} \longrightarrow \text{potpis nevalidan}$$

Ukratko, onaj koji verifikuje poruku prihvata potpis (r, s) samo ako je relacija $\beta^r \cdot r^s \equiv \alpha^x \pmod{p}$ zadovoljena. U protivnom, verifikacija nije uspjela. Dokazaćemo korektnost Elgamal šeme potpisivanja.

Dokaz: Specijalno, pokazaćemo da proces verifikacije vraća - potpis je validan, ako onaj koji verifikuje poruku koristi ispravni javni ključ i ispravnu poruku, i ako su parametri potpisa (r, s) odabrani kao što je navedeno. Počinjemo sa jednačinom verifikacije:

$$\begin{aligned} \beta^r \cdot r^s &\equiv (\alpha^d)^r (\alpha^{K_e})^s \pmod{p} \\ &\equiv \alpha^{dr+K_e^s} \pmod{p} \end{aligned}$$

Potpis treba smatrati validnim ako je ovaj izraz jednak α^x:

$$\alpha^x \equiv \alpha^{dr+K_e^s} \pmod{p}$$

Posljednja relacija važi ako su eksponenti na obje strane izraza jednaki u modulu $p-1$ aritmetici:

$$x \equiv dr + K_e^s \pmod{p-1}$$

Odakle dobijamo formiranje parametara potpisa kako slijedi:

$$s \equiv (x - d \cdot r) \cdot K_e^{-1} \pmod{p-1}$$

Uslov da je $nzd(K_e, p-1) = 1$ potreban je zato što je potrebno računanje inverznog elementa efemeralnog ključa po modulu $p-1$ kada računamo x.

5.2 Računarski aspekt

Važno je primjetiti da je postupak generisanja ključeva identičan kao i kod Elgamal enkripcije pa zbog toga p mora da zadovoljava odredjene uslove. Specijalno, mora biti dužine najmanje 1024 bita. Prost broj se može generisati korišćenjem algoritma

za traženje prostih brojeva. Javni ključ se generiše slučajno, a za stepenovanje javnog ključa može se koristiti algoritam brzog stepenovanja.

Potpis se sastoji od para (r, s). Oboje imaju dužinu u bitima kao i broj p, tako da je ukupna dužina paketa $(x, (r, s))$ trostruka dužina poruke x. Računanje broja r zahtjeva podizanje broja α na neki stepen po modulu p, što se može postići korišćenjem algoritma brzog stepenovanja. Glavna operacija prilikom računanja broja s je traženje inverza od K_e. Ovo se može postići korišćenjem proširenog Euklidovog algoritma. Ubrzanje algoritma je moguće uvođenjem preprocesiranja. Potpisnik može unaprijed da generiše efemeralne ključeve K_e i r vrijednosti, i da sačuva ove podatke. Kada treba potpisati poruku, oni se mogu preuzeti i koristiti za računanje vrijednosti s. Onaj koji prima poruku i treba da je verifikuje izvršava dvije operacije stepenovanja korišćenjem algoritma brzog stepenovanja i jedno množenje.

5.3 Sigurnost

Prvo, moramo biti sigurni da onaj koji verifikuje poruku ima ispravan javni ključ.

5.3.1 Računanje diskretnih logaritama

Sigurnost šeme potpisivanja počiva na težini problema diskretnog logaritma (DLP). Ako je Oskar sposoban da računa diskretne logaritme, on može izračunati privatni ključ d iz β kao i efemeralni ključ K_e iz r. Sa ovim informacijama, on može potpisati proizvoljne poruke u tuđe ime. Otuda Elgamal parametri moraju biti izabrani da je problem diskretnog logaritma težak. Da bi problem određivanja diskretnog logaritma bio težak broj p mora bti najmanje 1024 bita.

5.3.2 Ponovno korišćenje efemeralnog ključa

Ako potpisnik poruke više puta koristi isti efemeralni ključ K_e, napadač lako može izračunati privatni ključ d. Ovo vodi do potpunog razbijanja sistema. Evo kako napad funkcioniše.

Oskar posmatra dva digitalna potpisa i poruke oblika $(x, (r, s))$. Ako dvije poruke x_1 i x_2 imaju isti efemeralni ključ K_e, Oskar lako može to da primjeti, jer su dvije r vrijednosti iste pošto se računaju na slijedeći način:

$$r = r_1 = r_2 \equiv \alpha^{K_e} \pmod{p}$$

S vrijednosti su različite, pa Oskar u stvari dobija dvije jednačine:

$$s_1 \equiv (x_1 - d \cdot r)K_e^{-1} \pmod{p-1}$$

$$s_2 \equiv (x_2 - d \cdot r)K_e^{-1} \pmod{p-1}$$

Ovo je sistem jednačina sa dvije nepoznate:

1. d - što je i Bobov privatni ključ
2. K_e - efemeralni ključ

Ako obje jednačine pomnožimo sa K_e ovo postaje linearni sistem jednačina koji se lako riješava. Oskar jednostavno oduzima drugu jednačinu od prve, i dobija:

$$s_1 - s_2 \equiv (x_1 - x_2)K_e^{-1} \pmod{p-1}$$

Iz prethodne jednačine možemo dobiti efemeralni ključ po formuli:

$$K_e \equiv \frac{x_1 - x_2}{s_1 - s_2} \pmod{p-1}$$

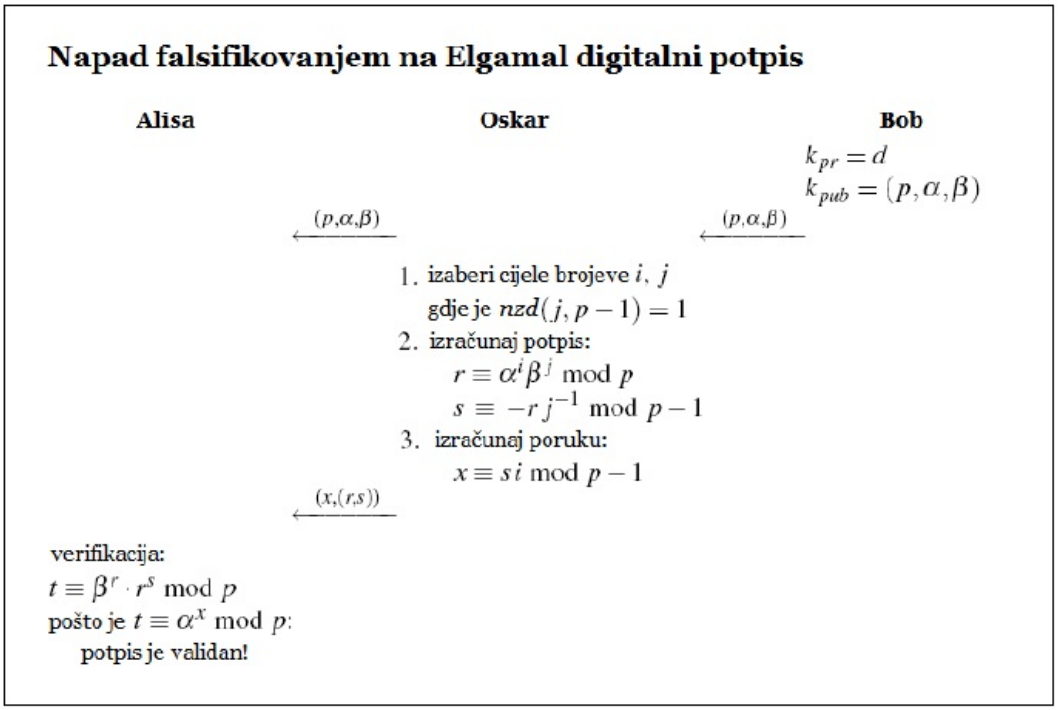
Ako je $nzd(s_1 - s_2, p-1) \neq 1$ jednačina ima višestruko riješenje za K_e, i Oskar mora da provjeri koje je ispravno. U svakom slučaju, korišćenjem K_e, Oskar takođe može izračunati privatni ključ d iz jedne od jednačina sistema, na primjer po formuli:

$$d \equiv \frac{x_1 - s_1 K_e}{r} \pmod{p-1}$$

Sa poznavanjem privatnog ključa d i parametara javnog ključa, Oskar sada slobodno može potpisati bilo koji dokument u Bobovo ime. U cilju izbjegavanja napada, za svako potpisivanje bilo bi poželjno birati efemeralne ključeve koji se slučajno generišu.

5.3.3 Napad falsifikovanjem

Moguće je da napadač generiše validan potpis za slučajnu poruku x. Napadač Oskar glumi Boba, tj. Oskar tvrdi pred Alisom da je on u stvari Bob. Napad se odigrava na sljedeći način [8]:



Slika 5.1: Napad falsifikovanjem na Elgamal digitalni potpis

Verifikacija vodi do validnog potpisa zato što važi:

$$
\begin{aligned}
t &\equiv \beta^r \cdot r^s \pmod{p} \\
&\equiv \alpha^{dr} \cdot r^s \pmod{p} \\
&\equiv \alpha^{dr} \cdot \alpha^{(i+dj)s} \pmod{p} \\
&\equiv \alpha^{dr} \cdot \alpha^{(i+dj)(-rj^{-1})} \pmod{p} \\
&\equiv \alpha^{dr-dr} \cdot \alpha^{-rij^{-1}} \pmod{p} \\
&\equiv \alpha^{si} \pmod{p}
\end{aligned}
$$

Pošto je poruka konstruisana kao $x \equiv si \pmod{p}$, posljednji izraz je jednak:

$$\alpha^{si} \equiv \alpha^x \pmod{p}$$

što je i upravo uslov koji Alisa provjerava za prihvatanje potpisa kao validnog.

Napadač u koraku 3. računa poruku x, čije značenje ne može da kontroliše. Zbog toga, Oskar može praviti validne potpise samo za pseudoslučajne poruke.

Napad nije moguće izvesti ako se poruka hešira, što je u praksi veoma čest slučaj. Umjesto da se poruka koristi za računanje potpisa, prije potpisivanja nad porukom se primjenjuje heš funkcija, odnosno jednačina potpisivanja postaje:

$$s \equiv (h(x) - d \cdot r)K_e^{-1} \pmod{p-1}$$

Glava 6

Zaključak

Elgamalov algoritam je asimetrični kripto sistem koji se u ogromnoj mjeri oslanja na ideju Difi Helman protokola, a sigurnost zasniva na kompleksnosti DLP i DHP problema. U principu, moglo bi se reći da Elgamal kao kripto sistem dijeli prednosti ostalih asimetričnih kripto sistema ali i njihove nedostatke.

Danas se Elgamal algoritam koristi u mnogim kriptografskim proizvodima, kao standard za digitalni potpis u softveru otvorenog koda GnuPG, u okviru PGP algoritma za zaštitu podataka.

Što se tiče efikasnosti enkripcije, Elgamal algoritam posjeduje karakterističan odnos dužine izvorne poruke prema dužini enkriptovane poruke 1:2, što ga čini prilično neefikasnim. Sa druge strane, izvorna poruka se može transformisati u više oblika enkriptovanog teksta, zavisno od odabira efemeralnog ključa, dajući mu vjerovatnosnu dimenziju, što mu značajno uvećava upotrebnu vrijednost u pojedinim aplikacijama.

Elgamal je, od svoje pojave 1985. godine, ostvario značajnu, može se reći istorijsku ulogu u kriptografiju, stvarajući okvir za razvoj mnogih ideja i konkretnih kripto sistema koji su i danas u širokoj upotrebi.

Bibliografija

[1] Z.Ž.Avramović, Dražen Petrović, Kriptografija, simetrični i asimetrični algoritmi, Banja Luka, maj 2008

[2] Man Young Rhee, Internet Security Cryptographic principles, algorithms and protocols, Wiltshire, 2003

[3] William Stalings, Cryptography and Network Security Principles and Practices, London, nov 2005

[4] A.J. McCurley, The Discrite Logarithm Problem, Proceedings in Applied Cryptography, Providence, 1990

[5] A. Menezes, P. Van Oorschot, S. Vanstone, Handbook of Applied Cryptography, Boca Raton, 1996

[6] Christof Paar Jan Pelzl, Understanding Cryptography, Springer-Verlag Berlin Heidelberg, 2010

[7] Slobodan Vujošević, Teorija brojeva i kriptografija

[8] Darinka Vučinić, Digitalni potpis-specijalistički rad, Podgorica, 2010