

Diffie-Hellman algoritam

Seminarski rad u okviru kursa
Tehničko i naučno pisanje
Matematički fakultet

Dunja Milenković mi22056@alas.matf.bg.ac.rs	Jana Vuković mi22124@alas.matf.bg.ac.rs
Lazar Nikolić mi22164@alas.matf.bg.ac.rs	Sofija Janevska mi22035@alas.matf.bg.ac.rs

15. novembar 2022.

Sažetak

Ovde će ići abstrakt

Sadržaj

1	Uvod	2
2	Osnove Diffie-Hellman algoritma	2
3	Algoritam	2
3.1	Koraci algoritma	3
3.2	Problemi oslonci	4
4	Primer Diffie-Hellman algoritma na konkretnim vrednostima	5
4.1	... podnaslov	5
5	Primene na bezbednosnim protokolima	6
5.1	SSL	6
5.2	SSH	6
5.3	IPSec	6
6	Poslednji naslov	6
7	Zaključak	6
	Literatura	6
A	Dodatak	7

1 Uvod

Sve rasprostranjenija upotreba interneta širom sveta donosi sa sobom, kako značajne prednosti u načinu istraživanja, rada, povezivanja i sl. tako i povećan rizik od nebezbedne komunikacije i razmene podataka. Bilo da se radi o individualnim, privatnim razmenama informacija između ljudi ili rada nekih od najvećih državnih institucija, bezbedna i neometana interakcija i komunikacija među korisnicima na internetu je od presudnog značaja. Glavni problem nastaje kada spoljni korisnik koji nije predviđen kao deo veze pokuša da je prekine i dođe do određenih informacija koje su originalno bile namenjene jednom od povezanih korisnika. Upravo rešavanjem ovakvih problema bavi se **kriptografija**.

Kriptografija je oblast koja razvija tehnike koje omogućavaju zaštićenu i efikasnu digitalnu komunikaciju [1]. Bez nje, komunikacija nebezbednim i nepoverljivim kanalima, što uključuje sve vrste mreža, a pogotovo internet, ne bi bila moguća. U ovom radu bavićemo se jednom od tehnika kriptografije koja se naziva **Diffie-Hellman algoritam** tj. **Diffie-Hellman protokol** i predstavimo način rada algoritma, njegova osnovna svojstva, primene, prednosti i mane.

2 Osnove Diffie-Hellman algoritma

Glavna klasifikacija kriptografskih algoritama deli ih na kriptografiju privatnog ključa (eng. secret key cryptography), kriptografiju javnog ključa (eng. public key cryptography) i heš funkcije (eng. hash functions) [1]. Kako je Diffie-Hellman protokol predstavnik asimetrične kriptografije, njemu ćemo posebno izdvojiti i definisati.

Kriptografija javnog ključa ili **asimetrična kriptografija** podrazumeva korišćenje dva ključa – jednog za šifrovanje i drugog za dešifrovanje koda. „Tvorcima“ tj. pokretačima asimetrične kriptografije smatraju se **Whitfield Diffie** (Vitfild Difi) i **Martin Hellman** (Martin Helman), dok je Ralph Merkle (Ralf Merkl) takođe dao svoj doprinos ovoj oblasti radom na distribuciji javnog ključa [2]. Njihova saradnja rezultirala je objavom zajedničkog rada pod nazivom „Novi pravci u kriptografiji“ (eng. "New Directions in Cryptography"), novembra 1976. godine. U njemu su opisane osnove Diffie-Hellman algoritma, koji je upravo otud i dobio naziv. Diffie i Hellman su 2015. godine dobili Tjuringovu nagradu za svoja dostignuća u oblasti kriptografije.

U spomenutom radu, Diffie-Hellman algoritam opisan je kao algoritam koji koristi dva ključa i tako omogućava korisnicima bezbednu komunikaciju bez potrebe za deljenjem privatnog ključa. Takođe, simetrična kriptografija omogućava postojanje digitalnih potpisa tj. autentifikacije pošiljaoca poruke. Osnovni princip na kome funkcioniše jeste korišćenje tzv. jednosmernih matematičkih funkcija (eng. one-way functions), koje se lako izračunavaju u jednom smeru, dok isto ne važi i za njihove inverzne funkcije [1]. Na ovaj način omogućeno je postojanje javnog ključa, bez rizika da ga treći korisnik koji nije predviđen kao deo mreže može zloupotrebiti.

3 Algoritam

Pre početka razmene ključeva, uspostavlja se 2 javno poznata broja: p i g .

- p - bezbedan prost broj i modulo po kom radimo, preporuka je da to bude broj dužine 2048 bita (broj reda 10^{616}) [4]
- g - generator, mora da bude primitivni koren od p

Bezbedan prost broj: p je bezbedno prost ako može da se izrazi kao $p = 2q + 1$ gde je q takođe prost broj. q je Sofija Žermen prost (eng. *Sophie Germain prime*). Ovo je bitno kako bi se izbegao specifičan napad na Diffie-Hellman — Silver-Polig-Hellman algoritam (eng. *Silver-Pohlig-Hellman algorithm*) [5]

Primitivni koren: g je primitivni koren po modulu p ako za svaki celi broj a koji je uzajamno prost sa p postoji celi broj k tako da je

$$g^k \equiv a \pmod{p}$$

Takvo k se naziva indeks ili diskretni logaritam od a sa odnoveom $g \pmod{p}$.

3.1 Koraci algoritma

Recimo da Anastasija i Boban žele da razmene ključeve.

- Pretpostavljamo da su se Anastasija i Boban već dogovorili oko brojeva g i p
- Anastasija nasumično bira neki tajni broj $x \in \mathbb{N}$ i izračuna njegov *najmanji pozitivni ostatak* (objašnjeno kasnije), čime dobija a , ovo je njen privatni ključ
- Zatim kalkuliše njen javni ključ koji je jednak $A = g^a \pmod{p}$
- Boban će uraditi isto, nakon čega poseduje privatni ključ b i javni ključ $B = g^b \pmod{p}$
- Anastasija i Boban sada razmene njihove javne ključeve (privatni ključevi ostaju tajni)
- Anastasija sada poseduje svoje ključeve, kao i Bobanov javni ključ
- Ona će izračunati tajni broj $K = B^a \pmod{p}$. Ako zamenimo B , videćemo da je

$$\begin{aligned} K &= (g^b \pmod{p})^a \pmod{p} = \\ &= (g^b)^a \pmod{p} = \\ &= g^{ab} \pmod{p} \end{aligned}$$

- Boban poseduje svoje ključeve i Anastasijin javni ključ
- On će istim postupkom kao i Anastasija pronaći K

$$\begin{aligned} K &= A^b \pmod{p} = \\ &= (g^a \pmod{p})^b \pmod{p} = \\ &= (g^a)^b \pmod{p} = \\ &= g^{ab} \pmod{p} \end{aligned}$$

Definicija 3.1.1 *Najmanji pozitivni ostatak nekog x po modulu p je najmanji pozitivni celi broj a tako da je*

$$x \equiv a \pmod{p}$$

Trenutni korak	Anastasija zna	Javno poznato	Boban zna
Početak algoritma		g, p	
Oboje računaju svoje ključeve	$a, A = g^a \bmod p$	g, p	$b, B = g^b \bmod p$
Razmenjuju se javni ključevi	a, A, B	g, p, A, B	b, B, A
Oboje izračunaju isto, tajno K	$a, A, B, K = B^a \bmod p = g^{ab} \bmod p$	g, p, A, B	$b, B, A, K = A^b \bmod p = g^{ab} \bmod p$

Tabela 1: ovo je baš očajna tabela

Primetimo da će i Anastasija i Boban izračunati isti broj K (1). K predstavlja ključ za enkripciju ključeva (engl. *Key-Encryption Key (KEK)*). Pomoću tog ključa mogu sinhrono da razmene novi ključ za enkripciju sadržaja (engl. *Content-Encryption Key (CEK)*) i da nastave komunikaciju pomoću njega. [6]

Neko ko je prisluškivao ovoj razmeni zna p, g, g^a i g^b . Da bi od ovih brojeva pronašao K , on mora da izračuna vrednost a ili b , problem koji se zove *komputacioni Diffie-Hellman problem* (engl. *computational Diffie-Hellman problem (CDH)*).

3.2 Problemi oslonci

Neka G bude ciklična grupa reda q , sa generatorom g . Drugačije rečeno, svaki broj x iz G je kongruentan sa nekim g^k , gde je k neki celi broj tako da je $0 \leq k \leq q - 1$.

$$x = g^k$$

Ako znamo g i k , lako možemo izračunati x . Međutim, ako znamo x i g , pronalaženje k je veoma teško, i to je zapravo problem diskretnog logaritma.

Definicija 3.2.1 *Problem diskretnog logaritma u cikličnoj grupi G reda q je pronalaženje broja k , $0 \leq k \leq q - 1$ tako da $x = g^k$, za neko g i x .*

Ako možemo da rešimo problem diskretnog logaritma, možemo da nađemo i tajni broj K iz Diffie-Hellman ako znamo oba javna ključa. Obrnuto nije dokazano: ako možemo da pronađemo tajni broj K , ne mora da znači da možemo rešiti bilo koji problem diskretnog logaritma.

Ne treba da pretpostavimo da je diskretni logaritam jedini način da se razbije Diffie-Hellman. Dovoljna je bilo koja metoda koja može da pronađe g^{ab} iz g^a i g^b , i otud uvođenje sledećeg problema:

Definicija 3.2.2 *Neka $a, b \in \mathbb{Z} \setminus q\mathbb{Z}$ (a i b su u nekoj cikličnoj grupi reda q nek neko proveriti ovo!), i neka $A = g^a$ i $B = g^b$. **Komputacioni Diffie-Hellman problem (CDH)** je problem nalaženja g^{ab} ako znamo A i B u cikličnoj grupi $G = \langle g \rangle$ reda q .*

Na težini ovog problema se oslanja bezbednost Diffie-Hellman šeme. Najefikasnije rešenje za koje znamo je da rešimo problem Diskretnog Logaritma ali još uvek nije dokazana ekvivalencija između njih. [7] Takođe slična je i ElGamal enkripcija o kojoj će biti reči kasnije. Ona se oslanja na sledeći problem:

Definicija 3.2.3 *Neka $a, b, c \in \mathbb{Z} \setminus q\mathbb{Z}$ i $A = g^a$, $B = g^b$. Sa verovatnoćom $\frac{1}{2}$ postavimo $C = g^c$, a inače $C = g^{ab}$. Odlučujući Diffie-Hellman problem (engl. *Decisional Diffie-Hellman problem (DDH)*) je problem odlučivanja da li je $C = g^{ab}$, ako znamo $\langle g \rangle$ i A, B, C .*

Ako znamo da rešimo komputacioni Diffie-Hellman problem, lako je rešiti odlučujući Diffie-Hellman problem. Obrnuto ne važi. [8]

4 Primer Diffie-Hellman algoritma na konkretnim vrednostima

Algoritam opisan u prethodnim poglavljima, primenjen na vrednostima konkretizovanim kroz interakciju dva korisnika izgleda ovako:

- Neka je izabrani prost broj $q = 353$.
- Prost koren za ovu vrednost je $\alpha = 3$.
- Korisnik A bira tajni ključ $X_A = 97$.
- Korisnik B bira tajni ključ $X_B = 233$.
- Korisnik A izračunava javni ključ $Y_A = 3^{97} \bmod 353 = 40$.
- Korisnik B izračunava javni ključ $Y_B = 3^{233} \bmod 353 = 248$.
- Korisnici A i B razmenjuju javne ključeve Y_A i Y_B , nakon čega oba korisnika mogu da izračunaju tajni ključ K .
- Korisnik A tajni ključ izračunava po formuli:

$$K = (Y_B)^{X_A} \bmod 353 = 248^{97} \bmod 353 = 160$$

- Korisnik B tajni ključ izračunava po formuli:

$$K = (Y_A)^{X_B} \bmod 353 = 40^{233} \bmod 353 = 160$$

Po samoj definiciji algoritma možemo slobodno pretpostaviti da napadač na raspolaganju ima samo podatke o vrednosti izabranog prostog broja i njegovog prostog korena, kao i vrednosti javnih ključeva korisnika A i B. Dakle, napadač zna da je:

$$q = 353, \alpha = 3, Y_A = 40 \text{ i } Y_B = 248.$$

S obzirom da su radi jednostavnosti izabrani brojevi u ovom primeru veoma mali, napadač može da pokuša i teorijski uspe da otkrije vrednost tajnog ključa K , primenjujući čistu metodu grube sile ("brute force").

Da bi napadač otkrio vrednost tajnog ključa K , neophodno je da otkrije bar jednu vrednost koja će zadovoljavati jednačinu za izračunavanje tajnog ključa koju koriste korisnici. Koristeći informacije koje su mu poznate, napadač može da rekonstruiše kako izgledaju te jednačine, a potom pokuša da ih reši metodom grube sile:

$$Y_A = 3^a \bmod 353 = 40 \text{ ili } Y_B = 3^b \bmod 353 = 248.$$

Metoda grube sile u ovom slučaju podrazumeva stepenovanje trojke sve dok dobijena vrednost po modulu 353 ne postane 40 ili 248.

Jednačina biva zadovoljena kada je eksponent $a = 97$. Sa ovim saznanjem, napadač izračunava vrednost tajnog ključa K i dešifruje podatke deljene između korisnika A i B.

Zbog ovoga se u Diffie-Hellman algoritmu koriste vrednosti koje su značajno veće. U tom slučaju razbijanje algoritma grubom silom postaje krajnje nepraktično.

4.1 ... podnaslov

Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst.

5 Primene na bezbednosnim protokolima

Diffie-Hellman algoritam ima široku primenu na različite internet protokole, od kojih su najvažniji SSL, SSH i IPSec, koji će biti detaljnije opisani u nastavku.

5.1 SSL

SSL (skraćeno od eng. Secure Socket Layer) je protokol koji omogućava bezbednu onlajn komunikaciju između veb servera i pretraživača, koristeći upravo Diffie-Hellman algoritam [3]. Sastoji se iz dva sloja: donjeg i gornjeg. Donji sloj omogućava privatnu i sigurnu komunikaciju koristeći TCP (skraćeno od eng. Transmission Control Protocol), koji se zasniva samo na simetričnoj kriptografiji. Upravo u gornjem sloju, koji se naziva i protokol rukovanja (eng. handshake protocol), koristi se Diffie-Hellman algoritam. On omogućava autentifikaciju servera klijentu, kao i korišćenje javnog ključa za „dogovaranje“ o načinu šifrovanja i razmene ključeva koju će koristiti za komunikaciju [3].

Jedna od najznačajnijih primena SSL protokola odnosi se na osiguravanje onlajn transakcija kao što je npr. elektronsko bankarstvo.

5.2 SSH

SSH (skraćeno od eng. Secure Shell) je protokol koji se koristi za osiguravanje internet veze između dva računara [3]. Na ovaj način, omogućeno je bezbedno prijavljivanje sa jednog računara na neki drugi, udaljeni računar, vršenje komandi na udaljenom računaru, kao i prenos podataka između njih. Šifrovanje se odvija na sličan način kao u SSL protokolu.

5.3 IPSec

IPSec (skraćeno od eng. Internet Protocol Security) je protokol koji obezbeđuje komunikaciju između dve jedinice povezane IP mrežom [3]. Dok prethodna dva protokola štite samo saobraćaj koji se odvija preko njih tj. njihovih aplikacija, IPSec funkcioniše za sve konekcije ostvarene između bilo koja dva računara u okviru IP mreže [3]. Drugim rečima, IPSec podrazumeva transparentnu komunikaciju, u kojoj ni korisnici ni aplikacija ne moraju znati ništa o šifrovanju veze, što ga dovodi u prednost u odnosu na prethodno navedene protokole.

6 Poslednji naslov

Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst. Ovde pišem tekst.

7 Zaključak

Ovde pišem zaključak. Ovde pišem zaključak. Ovde pišem zaključak. Ovde pišem zaključak. Ovde pišem zaključak. Ovde pišem zaključak. Ovde pišem zaključak. Ovde pišem zaključak. Ovde pišem zaključak. Ovde pišem zaključak.

Literatura

- [1] Gary C. Kessler. *An Overview of Cryptography*, 2015.
- [2] Maryam Ahmed, Baharan Sanjabi, Difo Aldiaz, Amirhossein Rezaei, Habeeb Omotunde. *Diffie-Hellman and Its Application in Security Protocols*, International Journal of Engineering Science and Innovative Technology (IJESIT), 2012.
- [3] David A. Carts. *A Review of the Diffie-Hellman Algorithm and its Use in Secure Internet Protocols*, SANS Institute, 2001.
- [4] Adrian, David; et al. (October 2015). *Imperfect Forward Secrecy: How Diffie-Hellman Fails in Practice*
- [5] Mollin, Richard (2006-09-18). *An Introduction To Cryptography* (2nd ed.). Chapman and Hall/CRC. p. 344
- [6] E. Rescorla (June 1999). *Diffie-Hellman Key Agreement Method*
- [7] Kevin S. McCurley (1990). *The Discrete Logarithm Problem*
- [8] A. Joux, K. Nguyen (2003). *Separating Decision Diffie-Hellman from Computational Diffie-Hellman in Cryptographic Groups*

A Dodatak

Ovde pišem dodatne stvari, ukoliko za time ima potrebe. Ovde pišem dodatne stvari, ukoliko za time ima potrebe. Ovde pišem dodatne stvari, ukoliko za time ima potrebe. Ovde pišem dodatne stvari, ukoliko za time ima potrebe. Ovde pišem dodatne stvari, ukoliko za time ima potrebe.