PREVOD DRUGOG IZDANJA

Algoritmi kriptografije

Novi algoritmi nultog znanja, homomorfnog šifrovanja i kvantne kriptografije

Bookauthority
NAJBOLJA KNJIGA SVIH VREMENA O ALGORITMIMA KRIPTOGRAFIJE
WINNER

MASIMO BERTAĆINI

kompjuter biblioteka

<packt>

Algoritmi kriptografije

Novi algoritmi nultog znanja, homomorfnog šifrovanja i kvantne kriptografije

Masimo Bertaćini

kompjuter biblioteka

Packt›

Izdavač:

kompjuter biblioteka

Obalskih radnika 4a
Beograd, Srbija
Tel: 011/2520272
e-pošta: kombib@gmail.com
veb-sajt: www.kombib.rs

Za izdavača:
Mihailo J. Šolajić, direktor

Autor:
Masimo Bertaćini

Prevod: Nemanja Lukić
Recezent: Miroslav Ristić
Slog: Zvonko Aleksić

Znak Kompjuter biblioteke:
Miloš Milosavljević

Štampa: „Pekograf“, Zemun
Tiraž: 500
Godina izdanja: 2024.

Broj knjige: 583
Izdanje: Prvo
ISBN: 978-86-7310-606-9

Naslov originala:

Cryptography Algorithms
Second Edition
ISBN 978-1-83508-003-0

Packt Publishing Ltd.
Birmingham, UK, packt.com

Algoritmi kriptografije

Autorizovani prevod sa engleskog jezika.

SARADNICI

O AUTORU

Masimo Bertaćini, doktor nauka, istraživač, glavni naučnik, direktor i suosnivač kompanije Cryptolab Inc. Vlasnik je više patenata u oblasti kriptografije, kvantne kriptografije i veštačke inteligencije. Karijeru je počeo kao profesor matematike i statistike, nakon čega je osnovao Cryptolab Inc., startap kompaniju u sferi kriptografskih rešenja za računarsku bezbednost. Sa svojim timom inženjera dizajnirao je i implementirao prvi pretraživač koji radi sa šifrovanim podacima.

Dobitnik je nekoliko međunarodnih nagrada i priznanja, kao što su nagrada Silicijumske doline za izume, Pečat izvrsnosti od strane EU i nagrada za Dobavljača godine, u oblasti bezbednosnih rešenja, u Sjedinjenim Američkim Državama, 2023. godine. Trenutno predaje kriptografiju na kursu za računarsku bezbednost, kao honorarni profesor. Objavio je mnogo radova iz oblasti kriptografije i blokčejn tehnologije.

Prvo izdanje knjige *Algoritmi kriptografije* je 40 nedelja bilo na desetom mestu najprodavanijih knjiga u svojoj kategoriji na veb sajtu Amazon, a prema sajtu BookAuthority to je najbolja knjiga u 2023. godini u oblasti homomorfnog i kvantnog šifrovanja.

Posebno zahvaljujem Marku Masariju i Danijeleu Sartiniju, a od srca posebno hvala Leonu Sjuu i ostalim recenzentima.

Takođe, za podršku tokom moje profesionalne karijere, želim da se zahvalim Polu Hegeru, Mateu Sarici, Davidu Karmečiju, Majkolu Lombardiju, Džeku Voloseviču, Pjerđorđu Montanariju, Gaudenciju Garaviniju, Jiđingu Guu i Sari Raučverger.

Na kraju, želim da zahvalim svim članovima Packt tima koji su mi pomogli tokom pisanja ove knjige.

O RECENZENTIMA

David Tilemans ima više od 20 godina iskustva u oblasti bezbednosti i bezbednog razvoja. Tokom 10 godina se bavio kriptografijom i pametnim karticama tokom razvoja proizvoda za infrastrukturu javnih ključeva. Kasnije je zanimanje inženjera kriptografije zamenio zanimanjem inženjera bezbednosti aplikacija. U toj ulozi je implementirao bezbedan razvojni proces i ulogu kriptografa zamenio ulogom etičkog hakera i savetnika za bezbednosne razvojne procese.

Zahvaljujući tim iskustvima postao je stručnjak za kombinaciju kriptografije, etičkog hakovanja i bezbednih razvojnih praksi koje se tiču kriptografije. Danas radi kao nezavisni savetnik u finansijskom i državnom sektoru, kao arhitekta PKI sistema, programer bezbednosti i savetnik za bezbednost poslovanja.

Dr Pol Duplis je istraživač bezbednosti, na čelu istraživačkog programa za bezbednost, privatnost i sigurnost u okviru Odeljenja za korporativna istraživanja kompanije Robert Bosch GmbH, najvećeg svetskog dobavljača za automobilsku industriju i proizvođača industrijskih, stambenih i proizvoda široke potrošnje. Pol se od 2007. godine bavi primenjenim istraživanjima u raznim oblastima informacione bezbednosti. Trenutno, njegova istraživanja obuhvataju automatizaciju bezbednosti, bezbednost softvera, bezbednost mreža, otkrivanje upada i sisteme za otkrivanje mamaca, aplikacije veštačke inteligencije za bezbednost i bezbednost veštačke inteligencije, inženjering privatnosti i tehnologije očuvanja privatnosti. Pol ima doktorat iz računarskih nauka Univerziteta u Tibingenu.

Leon Su je softverski inženjer u kompaniji TVU Networks, lideru u oblasti inovativne tehnologije za snabdevanje medijima. U svom radu fokusiran je na optimizaciju algoritama za prenos podataka preko mreže i razvijanje aplikacija za detektovanje i prepoznavanje objekata u realnom vremenu.

Ima master diplomu za primenjenu fiziku Univerziteta Stanford . Strastveni je entuzijasta za kvantno računarstvo i učesnik događaja kao što su IBM Quantum Challenge i QHack.

Čast mi je što sam recenzent ove knjige i što sam svojim tehničkim znanjem doprineo uspehu ove publikacije.

Miroslav Ristić je redovni profesor na Prirodno-matematičkom fakultetu Univerziteta u Nišu, sa preko 25 godina iskustva u razvoju statističkog softvera. Posebno se ističe njegov rad na razvoju grafičkog korisničkog interfejsa R Commander za programski jezik R. Dugi niz godina recenzirao je značajan broj knjiga za izdavačku kuću Springer i časopis Journal of Applied Statistics. Od 2023. godine aktivno recenzira najaktuelnija izdanja izdavačke kuće "Kompjuter biblioteka". Nakon prevođenja, svako izdanje prolazi kroz njegovo stručno vrednovanje i recenziju prevoda, sa ciljem da se osigura da prevodi budu ne samo jasni, precizni i prilagođeni čitaocima, već i da održe visok kvalitet i stručnu relevantnost knjiga.

O BETA ČITAOCIMA

Sledeći spisak predstavlja čitaoce iz našeg beta programa koji su svojim povratnim informacijama ljubazno usmeravali razvoj ovog izdanja. Za dragocenu pomoć pri recenziji ovog izdanja, zahvaljujemo sledećim pojedincima:

Tomas Moris – AgileDinosaur

Anđali Latija – Archangel

Arjan Pandi – oumuamua

Predgovor

U ovom dobu visoke povezanosti, računarstva u oblaku, napada softverom za iznuđivanje i hakera, digitalna imovina menja naš način života. Stoga kriptografija i računarska bezbednost imaju suštinski značaj. Promene načina obrade i skladištenja podataka zahtevaju odgovarajući napredak na polju kriptografskih algoritama za nastavak večne bitke protiv informatičke piraterije.

Počevši od osnova simetričnih i asimetričnih algoritama, opisao sam savremene tehnike autentifikacije, prenosa i pretrage šifrovanih podataka, radi zaštite od špijuna i hakera. Upoznaćete algoritme koji koriste protokole sa nultim znanjem, eliptičkim krivama, homomorfnom pretragom i kvantnom kriptografijom.

Mogućnosti kvantne kriptografije neprestano rastu i narušavaju mere za koje se ranije smatralo da su bezbedne. Inovacije u kvantnoj kriptografiji su fokus kriptografskih istraživanja, pa sam u ovom, drugom, izdanju nastojao da pružim uvod u kvantnu kriptografiju i rani pregled kvantne pretrage, sa fokusom na Groverov algoritam.

Ovu knjigu vidim kao alat za studente i profesionalce koji žele da se usredsrede na sledeću generaciju kriptografskih algoritama. Želim da vam pomognem da budete svesni modernih dostignuća u kriptografiji, ali moj primarni cilj je da naučite matematičku logiku ovih algoritama, kako biste razumeli osnove. Kako budete napredovali kroz knjigu, nadam se da ćete postupno otkrivati oblasti praktične primene koje vas najviše zanimaju.

Za koga je ova knjiga

Ova knjiga je namenjena studentima, IT profesionalcima, entuzijastima računarske bezbednosti i svakome ko želi da razvije veštine koje se odnose na savremenu kriptografiju i izgradi uspešnu karijeru na polju računarske bezbednosti.

Koje su teme obuhvaćene ovom knjigom

Prvi deo: Kratka istorija i pregled kriptografije

Poglavlje 1: Detaljno o kriptografiji predstavlja kriptografiju, objašnjava zašto je potrebna i zašto je toliko važna u IT sektoru. Ovo poglavlje, takođe, pruža sveobuhvatan pregled glavnih algoritama u istoriji kriptografije.

Drugi deo: Klasična kriptografija (simetrično i asimetrično šifrovanje)

Poglavlje 2: Algoritmi simetričnog šifrovanja, je analiza simetričnog šifrovanja. Fokusiraćemo se na algoritme kao što su DES, AES i Bulova logika, koji se naširoko koriste za implementaciju računarskih sistema. Na kraju su prikazani napadi na ove algoritme.

Poglavlje 3: Algoritmi asimetričnog šifrovanja, je analiza klasičnih algoritama asimetričnog šifrovanja, kao što su RSA i Difi–Helman, kao i glavnih algoritama za šifrovanje privatnim/javnim ključem.

Poglavlje 4: Heš funkcije i digitalni potpisi u fokus stavlja heš funkcije, kao što je SHA-1, i digitalne potpise, koji su jedan od stubova moderne kriptografije. Razmotrićemo najvažnije i najpoznatije potpise, a kao poseban slučaj anonimnih potpisa, slepe potpise.

Treći deo: Novi kriptografski algoritmi i protokoli

Poglavlje 5: Protokoli nultog znanja obrađuje protokole sa nultim znanjem, koji su jedan od novih, fundamentalnih šifrovanih protokola blokčejn tehnologije. Oni su veoma korisni za autentifikaciju ljudi i mašina bez otkrivanja osetljivih podataka u nebezbednom kanalu komunikacije. Novi protokoli blokčejn tehnologije, kao što je zk-SNARK, zasnovani su na ovim algoritmima. Na kraju ćemo predstaviti Z/K13, novi protokol u oblasti nultog znanja koji sam ja osmislio.

Poglavlje 6: Nove inovacije u kriptografiji i logički napadi predstavlja tri algoritma koja sam izumeo. MB09 je zasnovan na Fermaovoj poslednjoj teoremi. MB11 bi mogao biti alternativa RSA algoritmu. Predstavljeni su i digitalni potpisi povezani sa ovim algoritmima. Takođe je predstavljen MBXX, novi protokol koji se može koristiti za postizanje konsenzusa.

Poglavlje 7: Eliptičke krive obrađuje novu granicu decentralizovanih finansija, eliptičke krive. Satoši Nakamoto je usvojio poseban tip eliptičke krive za implementaciju prenosa digitalne valute Bitcoin, poznate kao SECP256K1. Pogledaćemo kako funkcioniše i koje su glavne karakteristike ovog vrlo robusnog načina šifrovanja.

Poglavlje 8: Homomorfno šifrovanje i kripto pretraživač, predstavlja kripto pretraživač, koji je aplikacija homomorfnog šifrovanja. To je pretraživač sposoban da pretražuje šifrovan sadržaj. Videćemo kako je implementiran, istoriju tog poduhvata i moguće primene tog revolucionarnog pretraživača na poljima bezbednosti i privatnosti podataka.

Četvrti deo: Kvantna kriptografija

Poglavlje 9: Kvantna kriptografija prikazuje kako će s pojavom kvantnog računarstva većina algoritama koje smo do sada istraživali biti pod ozbiljnom pretnjom napada putem grube sile. Jedno od mogućih rešenja je kvantna kriptografija. To je jedna od najuzbudljivijih i najfantastičnijih vrsta šifrovanja koju je ljudski um izmislio. Tek smo na početku kvantne kriptografije, ali će ubrzo postati naširoko prihvaćena.

Poglavlje 10: Kvantni algoritmi za pretraživanje i kvantno računarstvo predstavlja Groverov algoritam, primer kvantne pretrage i napada na klasično simetrično šifrovanje. Učeći logiku ovog algoritma, videćemo kako se neki elementi kvantnog računarstva primenjuju u kriptografiji, posebno za probleme povezane sa slučajnom pretragom i napadima grubom silom.

Kako izvući maksimum iz ove knjige

Ova knjiga sistematski obrađuje matematička pitanja vezana za algoritme. Međutim, neophodno je znanje univerzitetskog nivoa iz matematike, algebre, osnovnih operatora, modularne matematike i teorije konačnih polja. Takođe je korisno poznavanje eliptičkih krivih i kvantnog računarstva, posebno matrica i crtanja krivih, kako biste izvukli maksimum iz ove knjige.

Preuzimanje slika u boji

Takođe, pružamo PDF datoteku koja sadrži slike ekrana/dijagrame korišćene u ovoj knjizi, u boji. Možete je preuzeti ovde: `https://packt.link/gbp/9781835080030`.

Korišćene konvencije

U ovoj knjizi smo koristili određene tekstualne konvencije.

Kurziv smo koristili za označavanje matematičkih jednačina i algebarskih karaktera u tekstu, kao i za naglašavanje. Na primer: „Prvo, izračunavamo $2^4 = 16$.“

Matematičke jednačine postavljene su na sledeći način:

$$a^n + b^n = z^n$$

Podebljan tekst: Označava novi pojam, važnu reč ili reči koje se pojavljuju na ekranu. Primer: „**Šifra** je sistem bilo koje vrste koji može da transformiše otvoreni tekst (poruku) u nečitljiv tekst (šifrovan tekst ili kriptogram).“

> **Upozorenja i važne napomene**
> ili
> **Trikovi i saveti**
> Prikazani su ovako.

Postanite član Kompjuter biblioteke

Kupovinom jedne naše knjige stekli ste pravo da postanete član Kompjuter biblioteke. Kao član možete da kupujete knjige u pretplati sa 40% popusta i učestvujete u akcijama kada ostvarujete popuste na sva naša izdanja. Potrebno je samo da se prijavite preko formulara na našem sajtu.
Link za prijavu: kombib.rs/kblista.php

Skenirajte QR kod
registrujte knjigu
i osvojite nagradu

DEO 1

Kratka istorija i pregled kriptografije

Ovaj deo predstavlja uvodni deo i pruža osnovne definicije, informacije i istorijat kriptografije i njenih algoritama.

Ovaj deo sadrži sledeće poglavlje:

- *Poglavlje 1: Detaljno o kriptografiji*

1

Detaljno o kriptografiji

Dobrodošli u svet kriptografije. U ovoj knjizi otkrićete tajne ove izuzetne nauke, koja može biti veoma važna za vašu karijeru, kao i za vaše opšte znanje. Na kraju ove knjige bićete u mogućnosti da razumete najvažnije algoritme koji čine kriptografiju izuzetnom i otkrićete neke od novih algoritama koje sam ja osmislio i implementirao tokom svoje karijere. Nadam se da će vaše putovanje uz ovu knjigu biti prijatno i da ćete ostvariti svoje akademske i profesionalne ciljeve.

Ovo poglavlje je uvod u kriptografiju i objašnjava zašto je potrebna i zašto je toliko važna za informacione tehnologije. Takođe, pruža sveobuhvatan istorijski pregled glavnih algoritama, od Cezarove šifre do Vernamove šifre i drugih, manje poznatih algoritama, kao što su Bilove šifre. Zatim će algoritmi kao što su **Rivest-Šamir-Adleman** (**RSA**), Difi–Helman, **Napredni standard za šifrovanje** (**AES**), protokoli sa nultim znanjem, eliptičke krive, homomorfno šifrovanje, kvantna kriptografija i drugi poznati algoritmi biti detaljno opisani u narednim delovima ove knjige. Sve u svemu, ovo poglavlje će vam pomoći da razumete kriptografiju i temelje očuvanja bezbednosti.

U ovom poglavlju obrađujemo sledeće teme:

- Kratak uvod u kriptografiju
- Osnovne definicije i glavne matematičke oznake korišćene u knjizi
- Binarna konverzija i **Američki standardni kod za razmenu informacija** (**ASCII**)
- Fermaova poslednja teorema, prosti brojevi i modularna matematika

- Istorija glavnih algoritama kriptografije i objašnjenje nekih od njih (Rozeta, Cezar, ROT13, Bil i Vernam)
- Napomene o bezbednosti (semantika, dokazivost, **jednokratni blok** (**OTP**) itd.)

Uvod u kriptografiju

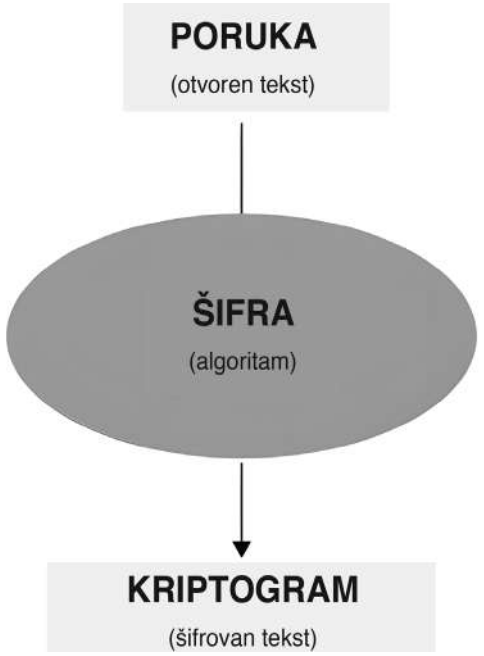
Jedna od najvažnijih stvari u kriptografiji je razumevanje definicija i oznaka. Nikada nisam bio ljubitelj definicija i oznaka, pre svega zato što sam jedini koji koristi oznake koje sam sam izmislio. Ali, shvatam da je veoma važno, naročito kada govorimo o nečemu što je povezano sa matematikom, da se međusobno usaglasimo. Stoga ću u ovom delu predstaviti osnovne informacije i izvore koji se odnose na kriptografiju.

Počećemo sa definicijom algoritma.

U matematici i računarskim naukama, **algoritam** je konačan niz jasno definisanih instrukcija koje se mogu implementirati na računaru.

Evo jednog važnog pitanja: šta je šifra?

Šifra je sistem bilo kog tipa koji je sposoban da pretvori otvoren tekst (poruku) u nerazumljiv tekst (kriptogram ili šifrovan tekst):



Slika 1.1: Proces šifrovanja

Da bi šifra imala svrhu, moramo postaviti dve operacije: šifrovanje i dešifrovanje. Jednostavnije rečeno, moramo čuvati poruku u tajnosti i bezbedno određeni vremenski period.

Definišemo *M* kao skup svih poruka i *C* kao skup svih kriptograma.

Šifrovanje je operacija koja generičku poruku, *m*, pretvara u kriptogram, *c*, primenom funkcije, *E*:

$$m - - - - > f(E) - - - - > c$$

Dešifrovanje je operacija koja vraća poruku u otvorenom tekstu, *m*, iz kriptograma, *c*, primenom funkcije, *D*:

$$c - - - - > f(D) - - - - > m$$

Matematički, $D(E(m)) = m$.

To znači da su funkcije *E* i *D* inverzne jedna drugoj, a funkcija *E* mora biti injektivna. **Injektivna** znači da različite vrednosti iz skupa *M* moraju odgovarati različitim vrednostima iz skupa *C*.

Napominjem da nije važno da li koristim velika ili mala slova, kao što su *(M)* ili *(m)*; to trenutno nije od značaja. Za sada sam koristio oble zagrade bez diskriminacije, ali kasnije ću koristiti uglaste zagrade kako bih razlikovao tajne elemente funkcije od poznatih, za koje ću koristiti uglaste zagrade. Tako da će tajna poruka *M* biti napisana kao *[M]*, kao i svaki drugi tajni parametar. Ovde je cilj da pokažemo kako algoritmi funkcionišu; njihovu implementaciju ćemo prepustiti inženjerima.

Još jedan važan pojam, koji je ključan za šifrovanje i dešifrovanje, jeste ključ. Da bi se poruka šifrovala i dešifrovala, potrebno je postaviti ključ. U kriptografiji, ključ je parametar koji određuje funkcionalni izlaz algoritma kriptografije ili šifre. Bez ključa, algoritam ne bi proizveo korisne rezultate.

Definišemo *K* kao skup svih ključeva korišćenih za šifrovanje i dešifrovanje poruka iz skupa *M*, a *k* kao pojedinačni ključ za šifrovanje ili dešifrovanje, poznat i kao ključ sesije. Međutim, ova dva načina definisanja ključa (skup ključeva je *K*, a pojedinačni ključ je *k*) uvek će se koristiti uz specifikaciju o kojoj vrsti ključa je reč (privatni ili javni).

Sada kada znamo glavne koncepte kriptografske notacije, vreme je da objasnimo razliku između privatnih i javnih ključeva:

- U kriptografiji, privatni ili tajni ključ (*Kpr*), označen kao *[K]* ili *[k]*, je parametar za šifrovanje/dešifrovanje koji zna samo jedna, obe, ili više strana, radi razmene tajnih poruka.
- U kriptografiji, javni ključ (*Kpu*) ili (*K*) je ključ za šifrovanje poznat svima koji žele da pošalju tajnu poruku ili autentifikuju korisnika.

Dakle, koja je glavna razlika između privatnih i javnih ključeva?

Razlika je u tome što se privatni ključ koristi i za šifrovanje i/ili dešifrovanje poruke, dok se javni ključ koristi samo za šifrovanje poruke i proveru identiteta (digitalni potpisi) ljudi i računara. To je suštinska i vrlo važna razlika, jer određuje razliku između simetričnog i asimetričnog šifrovanja.

Hajde da damo generičku definiciju ova dva metoda šifrovanja:

- **Simetrično šifrovanje** koristi samo jedan zajednički ključ za šifrovanje i dešifrovanje poruke.
- **Asimetrično šifrovanje** primenjuje više parametara da bi se generisao javni ključ (za šifrovanje poruke) i samo jedan privatni ključ za dešifrovanje poruke.

Kao što ćemo kasnije videti, privatni ključevi se koriste u simetričnom šifrovanju za šifrovanje i dešifrovanje poruke istim ključem, dok se u asimetričnom šifrovanju privatni ključ generalno koristi za dešifrovanje. S druge strane, javni ključevi se koriste isključivo u asimetričnom šifrovanju za šifrovanje poruke i za obavljanje digitalnih potpisa. Kasnije ćete videti funkciju ovih tipova ključeva, ali za sada imajte na umu da se privatni ključ koristi i za simetrično i asimetrično šifrovanje, dok se javni ključ koristi samo za asimetrično šifrovanje. Napominjem da mi nije namera da raspravljam o akademskim definicijama i oznakama, stoga pokušajte da shvatite svrhu i upotrebu svakog elementa.

Jedan od glavnih problema u kriptografiji je prenos ključa, odnosno razmena ključeva. Ovaj problem je izazvao velike rasprave unutar zajednice matematičara i kriptografa jer je bilo veoma teško odrediti kako preneti ključ, a da se izbegne fizička razmena.

Na primer, ako su Alisa i Bob želeli da razmene ključ (pre pojave asimetričnog šifrovanja), jedini pouzdan način bio je da se fizički sretnu na jednom mestu. Ova situacija je izazvala mnogo problema prilikom masovnog usvajanja telekomunikacionih sistema i interneta. Prvi problem je bio to što internet komunikacija zavisi od razmene podataka preko nebezbednih kanala. Kao što možete lako razumeti, ako Alisa komunicira s Bobom preko nebezbednog javnog komunikacionog kanala, postoji realna mogućnost da privatni ključ bude kompromitovan, što je izuzetno opasno za bezbednost i privatnost komunikacija.

Zbog toga se postavlja pitanje: *ako koristimo simetričnu šifru za zaštitu naših tajnih informacija, kako možemo bezbedno razmeniti tajni ključ?*

Jednostavan odgovor je sledeći: moramo obezbediti bezbedan *kanal* komunikacije za razmenu ključa.

Neko bi mogao da pita: *kako obezbeđujemo bezbedan kanal?*

Odgovor, ili tačnije više odgovora, naći ćemo kasnije, u ovoj knjizi. Čak i u složenim vojnim primenama, poput legendarne *crvene linije* između predsednika Sjedinjenih Američkih Država i Sovjetskog Saveza tokom Hladnog rata, korišćeni su simetrični ključevi za komunikaciju. Danas je uobičajeno koristiti asimetrično šifrovanje za razmenu ključa. Kada je ključ razmenjen, sledeća sesija komunikacije kombinuje se sa simetričnim šifrovanjem za šifrovanje prenetih poruka.

Iz mnogo razloga, asimetrično šifrovanje je dobar način za razmenu ključa i korisno je za autentifikaciju i digitalne potpise. Računarski gledano, simetrično šifrovanje je bolje jer može raditi sa ključevima manje dužine (u bitovima), što štedi dosta propusnog opsega i vremena. Generalno, njegovi algoritmi efikasno obezbeđuju bezbednost pomoću ključeva od 256-512 bita, u poređenju sa 4.000 ili više bitova kod asimetričnog RSA šifrovanja, na primer. Kasnije ću detaljno objasniti zašto i kako je to moguće, tokom analize algoritama u asimetričnom/simetričnom šifrovanju.

U ovoj knjizi analiziraću mnoge vrste tehnika kriptografije ali, u suštini, sve algoritme možemo podeliti u dve velike grupe: algoritme simetričnog i asimetričnog šifrovanja.

Potrebno je još nekoliko definicija da bismo razumeli kriptografiju:

- **Otvoren tekst**: U kriptografiji, to označava nešifrovan tekst ili sve što može biti javno izloženo. Na primer, (*vidimo se sutra u 10h)* je otvoren tekst.
- **Šifrovan tekst**: U kriptografiji, to označava konačan oblik teksta nakon što je izvršena procedura šifrovanja. Na primer, *vidimo se sutra u 10h* može postati *[x549559*ehebibcm3494]* u šifrovanom tekstu.
- Kao što sam već pomenuo, koristim različite zagrade za identifikaciju otvorenog teksta i šifrovanog teksta. Konkretno, ove zagrade (…) identifikuju otvoren tekst, dok uglaste zagrade […] identifikuju šifrovani tekst. Dakle, ovo je pomenuta tajna poruka: *[x549559*ehebibcm3494]*.

Binarni brojevi, ASCII kod i oznake

Kada obezbeđujemo podatke pomoću računara, uobičajeno se koriste podaci kao niske nula i jedinica, koje nazivamo bitovima. Dakle, brojevi se mogu pretvoriti u bitove (sistem osnove 2) umesto u sistem osnove 10, kao što je naš brojevni sistem. Pogledajmo kako mehanizam konverzije funkcioniše. Na primer, broj 123 može se zapisati u sistemu osnove 10 kao:

$$1 * 10^2 + 2 * 10^1 + 3 * 10^0$$

Isto tako, možemo konvertovati broj iz sistema osnove 10 u sistem osnove 2. U ovom slučaju, koristićemo primer broja 29:

Broj 29 konvertovan u binarni sistem

Korak	Operacija	Rezultat	Ostatak	Konverzija (osnova 2)
Korak 1:	29 /2	14	1	$(11101)_2$
Korak 2:	14 /2	7	0	
Korak 3:	7 /2	3	1	
Korak 4:	3/2	1	1	
Korak 5:	1 /2	0	1	

Slika 1.2: Konverzija broja 29 u sistem osnove 2 (bitovi).

Ostatak pri deljenju je vrlo popularan u kriptografiji, jer se **modularna matematika** zasniva na konceptu ostataka. O tome ćemo detaljno govoriti u sledećem odeljku, kada budem objasnio proste brojeve i modularnu matematiku.

Da bi se slova pretvorila u binarni sistem koji koriste računari, Američka asocijacija za standarde je 1960. godine osmislila ASCII kod.

Prema zvaničnom veb sajtu ASCII asocijacije, definicija je sledeća:

> *„ASCII označava Američki standardni kod za razmenu informacija. To je 7-bitni kod karaktera, gde svaki pojedinačni bit predstavlja jedinstven karakter."*

Sledeći primer prikazuje ASCII tabelu sa prvih 10 karaktera:

DEC	OCT	HEX	BIN	Simbol	HTML	Opis
0	000	00	00000000	NUL	&#000;	Nul karakter
1	001	01	00000001	SOH	&#001;	Početak zaglavlja
2	002	02	00000010	STX	&#002;	Početak teksta
3	003	03	00000011	ETX	&#003;	Kraj teksta
4	004	04	00000100	EOT	&#004;	Kraj prenosa
5	005	05	00000101	ENQ	&#005;	Zahtev
6	006	06	00000110	ACK	&#006;	Potvrda prijema
7	007	07	00000111	BEL	&#007;	Zvono
8	010	08	00001000	BS	&#008;	Jedno mesto nazad
9	011	09	00001001	HT		Horizontalni tabulator
10	012	0A	00001010	LF
Novi red

Slika 1.3: Prvih 10 karaktera i simbola predstavljenih u ASCII kodu.

U *poglavlju 4*, *Heš funkcije i digitalni potpisi*, naučićemo heksadecimalan i oktalni sistem. Ključni detalj u ovom trenutku je da posmatramo binarni sistem.

Primetićete da ću u svojim implementacijama, izrađenim pomoću istraživačkog softvera **Wolfram Mathematica**, često koristiti karakter 88 kao X za označavanje broja poruke koju treba šifrovati. U ASCII kodu, broj *88* odgovara simbolu *X*, kao što možete videti u sledećem primeru:

88 130 58 01011000 X X : *Veliko X*

U delu *Dodatak* na kraju knjige možete pronaći svu notaciju korišćenu u ovoj knjizi, kako za algoritme, tako i za njihovu implementaciju pomoću Mathematica koda.

Fermaova poslednja teorema, prosti brojevi i modularna matematika

Kada govorimo o kriptografiji, uvek moramo imati na umu da je ta tema, u suštini, povezana sa matematikom i logikom. Pre nego što počnem da objašnjavam **Fermaovu poslednju teoremu**, želim da uvedem nekoliko osnovnih oznaka koje ćemo koristiti u celoj knjizi, da izbegnemo zabune i da bolje razumete temu. Važno je znati da su neki simboli, poput =, ≡ (ekvivalentno) i := (ovaj poslednji možete naći u Mathematica kodu za računanje =), samo način da vam se kaže da dva elementa odgovaraju jedan drugom u jednakim merama; nije bitno da li je to u konačnom polju (ne brinite, upoznaćete ovu terminologiju), u računarskim naukama, ili u običnoj algebri. Matematičari bi se, možda, zgrozili nad ovim, ali verujem u vašu inteligenciju i da ćete se usredsrediti na suštinu, a ne na uniformnost.

Još jedan simbol, ≈ (približno), može se koristiti za označavanje sličnih približnih elemenata. Takođe, kada bude potrebno, naići ćete na simbol ^ (eksponent), klasičan način da se izrazi stepenovanje: a^x (a na x).

Simbol ≠, kao što se, verovatno, sećate iz srednje škole, znači **nije jednako** ili **različito**, što se u modularnoj matematici prikazuje kao ≢, što znači nije ekvivalentno.

Međutim, uvek ćete imati objašnjenja jednačina, pa ako niste upoznati sa matematičkom i logičkom notacijom, moćićete da se oslonite na opise. U svakom slučaju, objasniću svaku novu notaciju na koju naiđemo.

Na kraju, kao što već znate, verovatno, prost broj je ceo broj koji je deljiv samo sa sobom i sa 1, na primer, 2, 3, 5, 7...23...67...p.

Prosti brojevi su temelj matematike jer svi ostali, složeni brojevi, potiču od njih. Videćete da korišćenje složenog broja umesto prostog broja u kriptografiji može dovesti do ozbiljnog bezbednosnog propusta, ili do napada (vidi poglavlje 5, *Protokoli nultog znanja*).

Sada, hajde da vidimo šta je Fermaova poslednja teorema, gde se primenjuje i zašto je korisna za nas.

Fermaova poslednja teorema je jedna od najpoznatijih i najlepših teorema klasične matematike, strogo povezana sa prostim brojevima, koji su osnova kriptografije. Prema Vikipediji:

„U teoriji brojeva, Fermaova poslednja teorema (ponekad se naziva Fermaova hipoteza, naročito u starijim tekstovima) kaže da ne postoje tri prirodna broja a, b i c koja zadovoljavaju jednačinu $a^n + b^n = c^n$ za bilo koju vrednost n veću od 2. Slučajevi n = 1 i n = 2 su poznati od davnina i imaju beskonačno mnogo rešenja."

Drugim rečima, ona nam kaže da za bilo koji izložilac stepena n≥3, ne postoji nijedan prirodan broj a, b ili c koji zadovoljava ovu jednakost:

$$a^n + b^n = c^n$$

Zašto je ova teorema toliko važna za nas? To je nešto što ćete početi da cenite tokom čitanja ove knjige, kada bude više algoritama. U suštini, Fermaova poslednja teorema je strogo povezana sa prostim brojevima. U stvari, s obzirom na osobine prostih brojeva, da se dokaže Fermaova poslednja teorema, dovoljno je prikazati sledeće :

$$a^p + b^p \neq c^p$$

Ovde, *p* predstavlja bilo koji prost broj veći od 2.

Sam Ferma je u jednom radu naveo da ima „prelepu demonstraciju" teoreme koja, međutim, *nije mogla da stane na marginu njegovih beležaka*, ali ta demonstracija nikada nije pronađena.

Vajlsov dokaz je sa više od 200 stranica smanjen na oko 130, u poslednjoj verziji, i teško je razumljiv. Dokaz se oslanja na eliptičke krive: ove krive poprimaju poseban oblik kada se predstave u modularnoj formi. Vajls je došao do zaključka, posle 7 godina rada, i svoj dokaz izložio na kongresu matematičara 1994. godine. Deo logike Vajlsovog dokaza otkrićete u poglavlju 7, *Eliptičke krive*. Trenutno pretpostavljamo da je Vajls, da bi dokazao Fermaovu poslednju teoremu, morao da se osloni na Tanijama-Šimura hipotezu, koja tvrdi da su *eliptičke krive nad poljem racionalnih brojeva povezane sa modularnim formama*. Ponovo, ne brinite ako vam sve ovo deluje previše komplikovano; vremenom će postajati jasnije, kako budemo napredovali.

Detaljno ćemo analizirati Fermaovu poslednju teoremu u poglavlju 6, *Inovacije u kriptografiji i logički napadi*, gde uvodim MB09 algoritam zasnovan na Fermaovoj poslednjoj teoremi, uz druge inovativne algoritme u sistemima javnog/privatnog ključa. Takođe, analiziraćemo primenu eliptičkih krivih u kriptografiji, u *poglavlju 7*, *Eliptičke krive*.

Ferma je bio opsednut prostim brojevima, baš kao i mnogi drugi matematičari; tokom celog života je istraživao proste brojeve i njihova svojstva. Pokušao je da pronađe opštu formulu za predstavljanje svih prostih brojeva, ali, nažalost, Ferma, kao i mnogi drugi matematičari, uspeo je da konstruiše formulu samo za neke od njih. Sledeća formula je Fermaova formula za proste brojeve gde je *n* neki prirodan broj:

$$2^{2n} + 1$$

Ako *n* zamenimo celim brojevima, dobićemo neke proste brojeve:

- n = 1, p = 5
- n = 2, p = 17
- n = 3, p = 65 (nije prost broj)
- n = 4, p = 257

Verovatno poznatija, ali veoma slična, jeste Mersenova formula za proste brojeve, gde je opet *n* neki prirodan broj:

$$2^n - 1$$

Ovo daje sledeće rezultate:

- n = 1, p = 1
- n = 2, p = 3
- n = 3, p = 7
- n = 4, p = 15 (nije prost broj)
- n = 5, p = 31

Uprkos nebrojenim pokušajima da se pronađe formula koja bi isključivo predstavljala sve proste brojeve, niko do sada nije uspeo u ovom poduhvatu.

Great Internet Mersenne Prime Search (**GIMPS**) je istraživački projekat usmeren na pronalaženje najnovijih i najvećih prostih brojeva pomoću Mersenove formule.

Ako istražite GIMPS veb sajt, možete otkriti sledeće:

- *Svi izložioci stepena ispod 53.423.543 su testirani i provereni.*
- *Svi izložioci stepena ispod 92.111.363 su testirani barem jednom.*
- *51. Mersenov prost broj je pronađen!*
- *21. decembra 2018. godine — **Projekat GIMPS** je došao do najvećeg poznatog prostog broja, $2^{82.589.933}$ - 1, koji ima 24.862.048 cifara. Računar na kome je radio Patrik Laroš iz Okala u državi Floridi, došao je do pomenutog broja 7. decembra 2018. Novi prost broj, poznat i kao M82589933, dobija se tako što se pomnože 82.589.933 dvojki i oduzme 1. Više od milion i po cifara je duži od prethodnog najvećeg poznatog prostog broja.*

Osim toga, GIMPS je verovatno prvi decentralizovan primer kako se može podeliti snaga procesora i računarske moći radi postizanja zajedničkog cilja. Ali zašto je interesovanje za pronalaženje velikih prostih brojeva toliko veliko?

Postoje najmanje tri odgovora na ovo pitanje: strast prema samom istraživanju, novčane nagrade brojnih nagradnih fondova za otkrivanje velikih prostih brojeva i, na kraju, zato što su prosti brojevi od suštinskog značaja za kriptografiju, kao što je kiseonik važan za ljude. To je, takođe, jedan od razloga da se dodeli nagrada za otkrivanje velikih prostih brojeva.

Shvatićete da većina algoritama nove generacije radi sa prostim brojevima. Ali, kako otkriti da li je neki broj prost?

U matematici postoji značajna razlika u računskoj složenosti između operacija množenja i deljenja. Deljenje je računski znatno zahtevnije od množenja. To znači, na primer, da ako računam 2^x, gde je x ogroman broj, lakše je izvesti operaciju stepenovanja, a izuzetno je teško naći delioce datog broja.

Zbog toga su matematičari, kao Ferma, mukotrpno tražili algoritam koji bi ovu računsku operaciju učinio jednostavnom.

U oblasti prostih brojeva, Ferma je tvorac još jedne veoma interesantne teoreme, poznate kao **Fermaova mala teorema**. Pre nego što objasnim ovu teoremu, treba da znamo šta je modularna aritmetika i kako se koristi.

Najjednostavniji način da se nauči modularna aritmetika je da se zamisli sat. Kada kažemo: „Vidimo se u 1 posle podne," zapravo računamo da je 1 prvi sat posle 12 (skazaljka sata završava svoj kružni *put*).

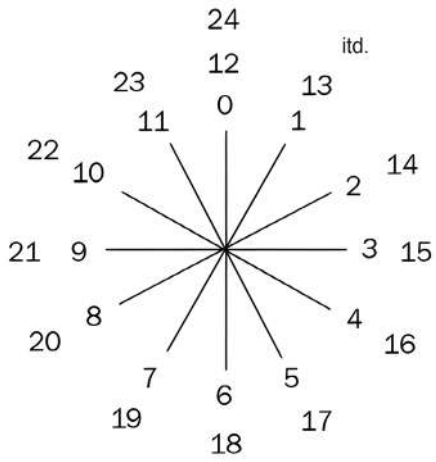
Tako možemo reći da nesvesno računamo po modulu 12, gde se celi brojevi *kreću iz početka* kada dosegnu određenu vrednost (u ovom slučaju, 12), koju nazivamo modul.

Tehnički, rezultat računanja sa modulom predstavlja ostatak deljenja između broja i modula.

Na primer, na našem satu imamo sledeće:

$$13 \equiv 1 \ (mod\ 12)$$

To znači da je 13 *kongruentno* sa 1 po modulu 12. *Kongruentno* znači kao jednako. Drugim rečima, možemo reći da je ostatak deljenja 13 sa 12 jednak 1:



Slika 1.4: Primer modularne aritmetike sa satom

Po Fermaovoj maloj teoremi, *ako je (p) prost broj, tada za bilo koji ceo broj (a) podignut na prost broj (p), kao rezultat sledeće jednačine dobijamo (a)*:

$$a^p \equiv a \ (mod\ p)$$

Na primer, ako je $a = 2$ i $p = 3$, tada je $2^3 = 2\ (mod\ 3)$. Drugim rečima, kada delimo 8 sa 3, ostatak je 2.

Fermaova mala teorema čini osnovu Fermaovog testa prostosti i predstavlja jedan od osnovnih delova elementarne teorije brojeva.

Fermaova mala teorema tvrdi da je broj *p verovatno prost* u sledećem slučaju:

$$a^p \equiv a \ (mod\ p)$$

Sada, nakon što smo osvežili svoje znanje o operacijama konverzije bita, videli kako izgleda ASCII kod i proučili osnovnu matematičku i logičku notaciju, možemo krenuti na putovanje u svet kriptografije.

Kratka istorija i sveobuhvatan pregled kriptografskih algoritama

Verovatno niko sa sigurnošću ne zna koji je kriptogram prvi izmišljen. Kriptografija se koristi već veoma dugo – otprilike 4.000 godina – i tokom vremena značajno je menjala svoje paradigme. Prvo je bila oblik skrivenog jezika, zatim je bila zasnovana na mehaničkoj transpoziciji slova, a na kraju se matematika i logika koriste za rešavanje složenih problema. Šta nas čeka u budućnosti? Verovatno će biti izmišljene nove metode za skrivanje tajni: kvantna kriptografija, na primer, već je u fazi ispitivanja i uskoro će biti dostupna. Objašnjavam nove algoritme i metode u ovoj knjizi, ali dozvolite mi da u ovom odeljku prikažem neke interesantne šifre iz *klasičnog* perioda. Uprkos računskoj snazi koju danas posedujemo, neki od ovih algoritama još uvek nisu razbijeni.

Kamen iz Rozete

Jedan od prvih izuzetnih primera kriptografije bili su hijeroglifi. Kriptografija znači „skrivanje reči" i dolazi od spajanja dve grčke reči: κρυπτός (kriptos) i γράφω (grafo). Među brojnim definicijama ovog pojma nalazimo sledeću: *pretvaranje običnog, otvorenog teksta u nerazumljiv tekst i obrnuto*. Zato hijeroglife možemo obuhvatiti ovom definicijom, jer smo njihovo skriveno značenje uspeli da *prevedemo* u razumljiv tekst tek nakon što je otkriven *Kamen iz Rozete*.

Kao što se verovatno sećate iz osnovne škole, na Kamenu iz Rozete je tekst ispisan na tri jezika: staroegipatskom (u hijeroglifima), demotskom i starogrčkom.

Slika 1.5: Kamen iz Rozete sa tri otkrivena jezika

Kamen iz Rozete je bilo moguće dešifrovati jer je starogrčki u to vreme bio dobro poznat.

Hijeroglifi su predstavljali način komunikacije među ljudima drevnog Egipta (i nekih zemalja iz okolnog područja). Žan-Fransoa Šampolion je priznat kao čovek koji je dešifrovao Kamen iz Rozete, počevši 1822. godine. Međutim, polimat Tomas Jang je od strane egiptologa akreditovan kao prvi čovek koji je objavio delimično tačan prevod teksta sa Kamena iz Rozete. Tomasa Janga ćemo sresti ponovo, u *poglavlju 9*, *Kvantna kriptografija*. Jang je prvi otkrio efekte dualizma talasa i čestica u vezi sa fotonima, što je veoma važno za kvantnu mehaniku.

Sličan problem sa dešifrovanjem nepoznatog jezika mogao bi se javiti u budućnosti, ako i kada stupimo u kontakt sa vanzemaljskom civilizacijom. Projekat pod nazivom **Institut za traženje vanzemaljske inteligencije** (**SETI**) (*https://www.seti.org/*) fokusiran je na sledeće:

> *„Od mikroba do vanzemaljske inteligencije, SETI institut je jedina organizacija u Americi koja je potpuno posvećena potrazi za životom u svemiru."*

Možda ćemo, ako jednog dana stupimo u kontakt sa vanzemaljskim bićima, na kraju uspeti da razumemo njihov jezik. Možete zamisliti da su hijeroglifi (u to vreme) delovali jednako nedodirljivo kao vanzemaljski jezik za nekoga ko nikada nije sreo taj oblik komunikacije.

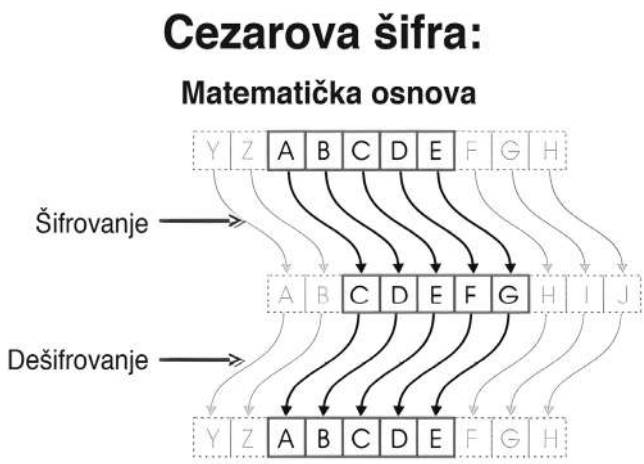
Cezarova šifra

U nastavku našeg putovanja kroz istoriju, saznajemo da se tokom Rimskog carstva kriptografija koristila za prenos poruka, od generala, ka komandirima i vojnicima. To je čuvena **Cezarova šifra**. Zašto je ovaj metod šifrovanja toliko značajan za istoriju kriptografije?

Ne zbog toga što ga je koristio Julije Cezar, jedan od najhrabrijih rimskih državnika i generala, već zato što je to, verovatno, prvi metod koji je implementirao matematiku.

Ova šifra je opšte poznata kao pokretna šifra. Tehnika pomeranja je veoma jednostavna: samo pomerite svako slovo koje želite da šifrujete za određeni broj mesta u alfabetu, tako da se na kraju svako slovo zameni nekim drugim. Na primer, ako pomerim za tri mesta, tada će A postati D, E postaje H i tako dalje.

Na primer, u ovom slučaju, pomeranjem svakog slova za tri mesta, implicitno smo kreirali tajni kriptografski ključ *[K=3]*:



Slika 1.6: Transpozicija slova u Cezarovoj šifri tokom procesa šifrovanja i dešifrovanja

Očigledno je da je Cezarova šifra metod simetričnog šifrovanja sa ključem. U ovom slučaju, algoritam funkcioniše na sledeći način:

- Koristi ključ: (+3)
- Poruka: ZDRAVO
- Za šifrovanje: Pomerite svako slovo za 3 mesta unapred.
- Za dešifrovanje: Pomerite svako slovo nazad za 3 mesta.

Na sledećoj slici je prikazan proces šifrovanja i dešifrovanja Cezarovom šifrom sa *ključem +3*; kao što možete primetiti, reč *ZDRAVO* se nakon šifrovanja pretvara u reč *CGUDYR*, a potom se vraća na *ZDRAVO*, nakon dešifrovanja.

Slika 1.7: Šifrovanje i dešifrovanje pomoću Cezarove šifre

Kao što možete zamisliti, Cezarovu šifru će vrlo lako razbiti običan računar, ukoliko je ključ fiksiran kao u prethodnom primeru. Šema je veoma jednostavna, što samo po sebi nije problem za kriptografski algoritam. Međutim, glavni problem je u ekstremnoj linearnosti same matematike. Lako ćemo razbiti kod metodom *grube sile*, tj. metoda koji isprobava sve moguće kombinacije da bi pronašao ključ nakon što pretpostavi korišćeni algoritam (u ovom slučaju, pomeranje). Treba da proverimo najviše 25 kombinacija — sva slova engleskog alfabeta (26) minus jedno (koje je isto kao u originalnoj poruci u čitljivom obliku). To je ništa, u poređenju sa milijardama i milijardama pokušaja koje računar izvrši kako bi probio moderni kriptografski algoritam.

Međutim, postoji složenija verzija ovog algoritma koja značajno uvećava efikasnost šifrovanja.

Ako promenim ključ za svako slovo i upotrebim taj ključ za zamenu slova radi generisanja šifrovanog teksta, dobijamo novu vrstu šifre, koja je poznata kao *transpoziciona šifra*.

Pogledajmo šta se događa kada šifrujemo reč *HELLO* sledeći ovaj metod:

1. Napišite alfabet.
2. Izaberite šifru (poznatu kao ključna fraza) poput *[JULIUSCAESAR]* i ponovite je, postavljajući svako slovo alfabeta u vezi sa karakterom iz ključne fraze u drugom redu, kao što je prikazano na slici ispod.
3. Nakon što definišemo poruku za šifrovanje, za svaki karakter poruke (u prvom redu) izaberite odgovarajući karakter ključne fraze (u drugom redu).
4. Prikupite izabrane odgovarajuće karaktere iz drugog reda da biste stvorili šifrovani tekst.

Nije vam baš jasno? Ne brinite, sledeći primer će sve razjasniti.

Važna napomena

Sledeći primer pokazuje jedino kako funkcioniše šifrovanje pomoću ključne fraze. Postoje, naravno, mnogi načini za dešifrovanje, ali ovde razmatramo samo šifrovanje.

Hajde da šifrujemo reč *HELLO* pomoću ključne fraze *[JULIUSCAESARJULIUS...]:*

Slika 1.8: Šifrovanje reči HELLO pomoću ključne fraze je teže za napad

Dakle, šifrujući otvoren tekst *HELLO* pomoću alfabeta i ključa (ili bolje rečeno ključne fraze), *JULIUSCAESAR*, ponavljane bez razmaka, dobijamo odgovarajući šifrovan tekst: *AURRL*.

Tako, H postaje A, E postaje U, L postaje R (dva puta), a O postaje L.

U primeru od ranije bilo je potrebno proveriti samo 25 kombinacija da se pronađe ključ Cezarove šifre; ovde je situacija malo drugačija i postoji čak (26!) mogućnosti da se otkrije ključ. To znači da treba pomnožiti 1×2×3…×26, što daje broj 403,2 91,461,126,605,635,584,000,000.

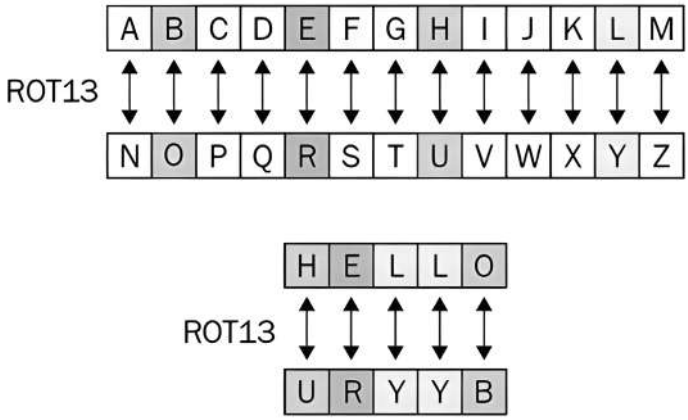
Još jedna prednost izgradnje ovih kriptograma je to što je lako zapamtiti ključnu reč ili frazu, što olakšava rad sa šifrovanim tekstom. Ali, hajde da pogledamo šifru koja koristi sličnu tehniku i koja se primenjuje u komercijalnim kontekstima.

ROT13

Savremen primer algoritma koji se koristi na internetu je **ROT13**, gde ROT znači *rotacija*. ROT13 je jednostavna šifra, izvedena iz Cezarove šifre, sa pomerajem od (+13), *koja koristi istu ROT13 funkciju za šifrovanje i dešifrovanje*. Iako je lako rešiva, ROT13 ima zanimljiv efekat: bez obzira da li se pomeranje vrši ulevo ili udesno, rezultat je isti.

Kao i u prethodnim primerima, u ROT13 biramo slova koja odgovaraju unapred definisanom ključu. Razlika je u tome što se, umesto ključne fraze, koristi prvih 13 slova engleskog alfabeta kao *generator ključa*. U šifrovanju, ROT13 primenjuje samo slova engleskog alfabeta, dok brojevi, simboli i drugi karakteri ostaju nepromenjeni. ROT13 funkcija šifruje otvoren tekst ključem zasnovanim na prvoj polovini abecede, dok se u drugoj polovini abecede vrši obrnuta zamena.

Pogledajte sledeći primer kako biste razumeli šemu šifrovanja:



Slika 1.9: ROT13 šema šifrovanja

Kao što možete videti u prethodnom dijagramu, *H* postaje *U, E* postaje *R, L* postaje *Y* (dva puta), a *O* postaje *B:*

HELLO=URYYB

ROT13 koristi prvih 13 slova alfabeta, do M, koje prelazi u Z, nakon čega se sekvenca vraća na početak: N postaje A, O postaje B i tako dalje, sve do Z, koje postaje M.

ROT13 je u ranom periodu interneta, tokom 1980-ih godina, koristila grupa **net. jokes** da sakrije potencijalno uvredljive viceve ili komentare.

Iako nije osmišljen za visoki stepen tajnosti, ROT13 se i dalje koristi za skrivanje imejl adresa od manje sofisticiranih spam-botova. Takođe, koristi se i za zaobilaženje spam filtera maskiranjem sadržaja imejla, iako je to vrlo slab način zaštite.

ROT13 je svojevremeno koristio **Netscape Communicator** – organizacija koja je kasnije osnovala Mozilla fondaciju (https://www.mozilla.org) – za skladištenje lozinki imejl adresa. Pored toga, Windows XP koristi ROT13 za skrivanje nekih ključeva registra, što pokazuje da čak i velike kompanije mogu imati nedostatak u bezbednosti i privatnosti u komunikaciji.

Bilove šifre

A u istoriji kriptografije dolazimo do izuzetnog primera šifre koja do danas nije dešifrovana, uprkos ogromnoj snazi savremenih računara. Često se kriptografija koristi za skrivene vrednosti ili fascinantno blago, kao što je slučaj sa intrigantnom pričom koja stoji iza **Bilovih** šifri.

Kako bismo bolje razumeli metod šifrovanja koji se koristi za ove šifre, mislim da je zanimljivo znati priču (ili legendu) o Bilu i njegovom blagu.

Priča o Bilovim šiframa uključuje zakopano blago vredno preko 20 miliona dolara, misteriozne dokumente ispunjene šifrovanim brojevima, kauboje Divljeg zapada i vlasnika hotela koji je posvetio život pokušajima da dešifruje te dokumenate. Cela priča se nalazi u pamfletu objavljenom 1885. godine.

Priča (koja se može pronaći u celosti na: http://www.unmuseum.org/bealepap.htm) počinje u januaru 1820. u Linčburgu, u Virdžiniji, u hotelu Vašington, gde se prijavio čovek po imenu Tomas J. Bil. Vlasnik hotela, Robert Moris, i Bil su postali prijatelji, a zbog poverenja koje je imao u gospodina Morisa, Bil mu je poverio kutiju sa tri misteriozna dokumenta popunjena brojevima.

Nakon bezbrojnih poteškoća i mnogo godina truda, samo drugi od tri šifrovana dokumenta je dešifrovan.

Kako zapravo izgledaju Bilove šifre?

Sastoje se od tri stranice koje sadrže isključivo brojeve u, naizgled, slučajnim redosledima.

Prvi dokument izgleda ovako:

71, 194, 38, 1701, 89, 76, 11, 83, 1629, 48, 94, 63, 132, 16, 111, 95, 84, 341, 975, 14, 40, 64, 27, 81, 139, 213, 63, 90, 1120, 8, 15, 3, 126, 2018, 40, 74, 758, 485, 604, 230, 436, 664, 582, 150, 251, 284, 308, 231, 124, 211, 486, 225, 401, 370, 11, 101, 305, 139, 189, 17, 33, 88, 208, 193, 145, 1, 94, 73, 416, 918, 263, 28, 500, 538, 356, 117, 136, 219, 27, 176, 130, 10, 460, 25, 485, 18, 436, 65, 84, 200, 283, 118, 320, 138, 36, 416, 280, 15, 71, 224, 961, 44, 16, 401, 39, 88, 61, 304, 12, 21, 24, 283, 134, 92, 63, 246, 486, 682, 7, 219, 184, 360, 780, 18, 64, 463, 474, 131, 160, 79, 73, 440, 95, 18, 64, 581, 34, 69, 128, 367, 460, 17, 81, 12, 103, 820, 62, 116, 97, 103, 862, 70, 60, 1317, 471, 540, 208, 121, 890, 346, 36, 150, 59, 568, 614, 13, 120, 63, 219, 812, 2160, 1780, 99, 35, 18, 21, 136, 872, 15, 28, 170, 88, 4, 30, 44, 112, 18, 147, 436, 195, 320, 37, 122, 113, 6, 140, 8, 120, 305, 42, 58, 461, 44, 106, 301, 13, 408, 680, 93, 86, 116, 530, 82, 568, 9, 102, 38, 416, 89, 71, 216, 728, 965, 818, 2, 38, 121, 195, 14, 326, 148, 234, 18, 55, 131, 234, 361, 824, 5, 81, 623, 48, 961, 19, 26, 33, 10, 1101, 365, 92, 88, 181, 275, 346, 201, 206, 86, 36, 219, 324, 829, 840, 64, 326, 19, 48, 122, 85, 216, 284, 919, 861, 326, 985, 233, 64, 68, 232, 431, 960, 50, 29, 81, 216, 321, 603, 14, 612, 81, 360, 36, 51, 62, 194, 78, 60, 200, 314, 676, 112, 4, 28, 18, 61, 136, 247, 819, 921, 1060, 464, 895, 10, 6, 66, 119, 38, 41, 49, 602, 423, 962, 302, 294, 875, 78, 14, 23, 111, 109, 62, 31, 501, 823, 216, 280, 34, 24, 150, 1000, 162, 286, 19, 21, 17, 340, 19, 242, 31, 86, 234, 140, 607, 115, 33, 191, 67, 104, 86, 52, 88, 16, 80, 121, 67, 95, 122, 216, 548, 96, 11, 201, 77, 364, 218, 65, 667, 890, 236, 154, 211, 10, 98, 34, 119, 56, 216, 119, 71, 218, 1164, 1496, 1817, 51, 39, 210, 36, 3, 19, 540, 232, 22, 141, 617, 84, 290, 80, 46, 207, 411, 150, 29, 38, 46, 172, 85, 194, 39, 261, 543, 897, 624, 18, 212, 416, 127, 931, 19, 4, 63, 96, 12, 101, 418, 16, 140, 230, 460, 538, 19, 27, 88, 612, 1431, 90, 716, 275, 74, 83, 11, 426, 89, 72, 84, 1300, 1706, 814, 221, 132, 40, 102, 34, 868, 975, 1101, 84, 16, 79, 23, 16, 81, 122, 324, 403, 912, 227, 936, 447, 55, 86, 34, 43, 212, 107, 96, 314, 264, 1065, 323, 428, 601, 203, 124, 95, 216, 814, 2906, 654, 820, 2, 301, 112, 176, 213, 71, 87, 96, 202, 35, 10, 2, 41, 17, 84, 221, 736, 820, 214, 11, 60, 760

Drugi dokument (koji je dešifrovan) izgleda ovako:

115, 73, 24, 807, 37, 52, 49, 17, 31, 62, 647, 22, 7, 15, 140, 47, 29, 107, 79, 84, 56, 239, 10, 26, 811, 5, 196, 308, 85, 52, 160, 136, 59, 211, 36, 9, 46, 316, 554, 122, 106, 95, 53, 58, 2, 42, 7, 35, 122, 53, 31, 82, 77, 250, 196, 56, 96, 118, 71, 140, 287, 28, 353, 37, 1005, 65, 147, 807, 24, 3, 8, 12, 47, 43, 59, 807, 45, 316, 101, 41, 78, 154, 1005, 122, 138, 191, 16, 77, 49, 102, 57, 72, 34, 73, 85, 35, 371, 59, 196, 81, 92, 191, 106, 273, 60, 394, 620, 270, 220, 106, 388, 287, 63, 3, 6, 191, 122, 43, 234, 400, 106, 290, 314, 47, 48, 81, 96, 26, 115, 92, 158, 191, 110, 77, 85, 197, 46, 10, 113, 140, 353, 48, 120, 106, 2, 607, 61, 420, 811, 29, 125, 14, 20, 37, 105, 28, 248, 16, 159, 7, 35, 19, 301, 125, 110, 486, 287, 98, 117, 511, 62, 51, 220, 37, 113, 140, 807, 138, 540, 8, 44, 287, 388, 117, 18, 79, 344, 34, 20, 59, 511, 548, 107, 603, 220, 7, 66, 154, 41, 20, 50, 6, 575, 122, 154, 248, 110, 61, 52, 33, 30, 5, 38, 8, 14, 84, 57, 540, 217, 115, 71, 29, 84, 63, 43, 131, 29, 138, 47, 73, 239, 540, 52, 53, 79, 118, 51, 44, 63, 196, 12, 239, 112, 3, 49, 79, 353, 105, 56, 371, 557, 211, 505, 125, 360, 133, 143, 101, 15, 284, 540, 252, 14, 205, 140, 344, 26, 811, 138, 115, 48, 73, 34, 205, 316, 607, 63, 220, 7, 52, 150, 44, 52, 16, 40, 37, 158, 807, 37, 121, 12, 95, 10, 15, 35, 12, 131, 62, 115, 102, 807, 49, 53, 135, 138, 30, 31, 62, 67, 41, 85, 63, 10, 106, 807, 138, 8, 113, 20, 32, 33, 37, 353, 287, 140, 47, 85, 50, 37, 49, 47, 64, 6, 7, 71, 33, 4, 43, 47, 63, 1, 27, 600, 208, 230, 15, 191, 246, 85, 94, 511, 2, 270, 20, 39, 7, 33, 44, 22, 40, 7, 10, 3, 811, 106, 44, 486, 230, 353, 211, 200, 31, 10, 38, 140, 297, 61, 603, 320, 302, 666, 287, 2, 44, 33, 32, 511, 548, 10, 6, 250, 557, 246, 53, 37, 52, 83, 47, 320, 38, 33, 807, 7, 44, 30, 31, 250, 10, 15, 35, 106, 160, 113, 31, 102, 406, 230, 540, 320, 29, 66, 33, 101, 807, 138, 301, 316, 353, 320, 220, 37, 52, 28, 540, 320, 33, 8, 48, 107, 50, 811, 7, 2, 113, 73, 16, 125, 11, 110, 67, 102, 807, 33, 59, 81, 158, 38, 43, 581, 138, 19, 85, 400, 38, 43, 77, 14, 27, 8, 47, 138, 63, 140, 44, 35, 22, 177, 106, 250, 314, 217, 2, 10, 7, 1005, 4, 20, 25, 44, 48, 7, 26, 46, 110, 230, 807, 191, 34, 112, 147, 44, 110, 121, 125, 96, 41, 51, 50, 140, 56, 47, 152, 540, 63, 807, 28, 42, 250, 138, 582, 98, 643, 32, 107, 140, 112, 26, 85, 138, 540, 53, 20, 125, 371, 38, 36, 10, 52, 118, 136, 102, 420, 150, 112, 71, 14, 20, 7, 24, 18, 12, 807, 37, 67, 110, 62, 33, 21, 95, 220, 511, 102, 811, 30, 83, 84, 305, 620, 15, 2, 10, 8, 220, 106, 353, 105, 106, 60, 275, 72, 8, 50, 205, 185, 112, 125, 540, 65, 106, 807, 138, 96, 110, 16, 73, 33, 807, 150, 409, 400, 50, 154, 285, 96, 106, 316, 270, 205, 101, 811, 400, 8, 44, 37, 52, 40, 241, 34, 205, 38, 16, 46, 47, 85, 24, 44, 15, 64, 73, 138, 807, 85, 78, 110, 33, 420, 505, 53, 37, 38, 22, 31, 10, 110, 106, 101, 140, 15, 38, 3, 5, 44, 7, 98, 287, 135, 150, 96, 33, 84, 125, 807, 191, 96, 511, 118, 40, 370, 643, 466, 106, 41, 107, 603, 220, 275, 30, 150, 105, 49, 53, 287, 250, 208, 134, 7, 53, 12, 47, 85, 63, 138, 110, 21, 112, 140, 485, 486, 505, 14, 73, 84, 575, 1005, 150, 200, 16, 42, 5, 4, 25, 42, 8, 16, 811, 125, 160, 32, 205, 603, 807, 81, 96, 405, 41, 600, 136, 14, 20, 28, 26, 353, 302, 246, 8, 131, 160, 140, 84, 440, 42, 16, 811, 40, 67, 101, 102, 194, 138, 205, 51, 63, 241, 540, 122, 8, 10, 63, 140, 47, 48, 140, 288

Treći dokument izgleda ovako:

317, 8, 92, 73, 112, 89, 67, 318, 28, 96,107, 41, 631, 78, 146, 397, 118, 98, 114, 246, 348, 116, 74, 88, 12, 65, 32, 14, 81, 19, 76, 121, 216, 85, 33, 66, 15, 108, 68, 77, 43, 24, 122, 96, 117, 36, 211, 301, 15, 44, 11, 46, 89, 18, 136, 68, 317, 28, 90, 82, 304, 71, 43, 221, 198, 176, 310, 319, 81, 99, 264, 380, 56, 37, 319, 2, 44, 53, 28, 44, 75, 98, 102, 37, 85, 107, 117, 64, 88, 136, 48, 151, 99, 175, 89, 315, 326, 78, 96, 214, 218, 311, 43, 89, 51, 90, 75, 128, 96, 33, 28, 103, 84, 65, 26, 41, 246, 84, 270, 98, 116, 32, 59, 74, 66, 69, 240, 15, 8, 121, 20, 77, 89, 31, 11, 106, 81, 191, 224, 328, 18, 75, 52, 82, 117, 201, 39, 23, 217, 27, 21, 84, 35, 54, 109, 128, 49, 77, 88, 1, 81, 217, 64, 55, 83, 116, 251, 269, 311, 96, 54, 32, 120, 18, 132, 102, 219, 211, 84, 150, 219, 275, 312, 64, 10, 106, 87, 75, 47, 21, 29, 37, 81, 44, 18, 126, 115, 132, 160, 181, 203, 76, 81, 299, 314, 337, 351, 96, 11, 28, 97, 318, 238, 106, 24, 93, 3, 19, 17, 26, 60, 73, 88, 14, 126, 138, 234, 286, 297, 321, 365, 264, 19, 22, 84, 56, 107, 98, 123, 111, 214, 136, 7, 33, 45, 40, 13, 28, 46, 42, 107, 196, 227, 344, 198, 203, 247, 116, 19, 8, 212, 230, 31, 6, 328, 65, 48, 52, 59, 41, 122, 33, 117, 11, 18, 25, 71, 36, 45, 83, 76, 89, 92, 31, 65, 70, 83, 96, 27, 33, 44, 50, 61, 24, 112, 136, 149, 176, 180, 194, 143, 171, 205, 296, 87, 12, 44, 51, 89, 98, 34, 41, 208, 173, 66, 9, 35, 16, 95, 8, 113, 175, 90, 56, 203, 19, 177, 183, 206, 157, 200, 218, 260, 291, 305, 618, 951, 320, 18, 124, 78, 65, 19, 32, 124, 48, 53, 57, 84, 96, 207, 244, 66, 82, 119, 71, 11, 86, 77, 213, 54, 82, 316, 245, 303, 86, 97, 106, 212, 18, 37, 15, 81, 89, 16, 7, 81, 39, 96, 14, 43, 216, 118, 29, 55, 109, 136, 172, 213, 64, 8, 227, 304, 611, 221, 364, 819, 375, 128, 296, 1, 18, 53, 76, 10, 15, 23, 19, 71, 84, 120, 134, 66, 73, 89, 96, 230, 48, 77, 26, 101, 127, 936, 218, 439, 178, 171, 61, 226, 313, 215, 102, 18, 167, 262, 114, 218, 66, 59, 48, 27, 19, 13, 82, 48, 162, 119, 34, 127, 139, 34, 128, 129, 74, 63, 120, 11, 54, 61, 73, 92, 180, 66, 75, 101, 124, 265, 89, 96, 126, 274, 896, 917, 434, 461, 235, 890, 312, 413, 328, 381, 96, 105, 217, 66, 118, 22, 77, 64, 42, 12, 7, 55, 24, 83, 67, 97, 109, 121, 135, 181, 203, 219, 228, 256, 21, 34, 77, 319, 374, 382, 675, 684, 717, 864, 203, 4, 18, 92, 16, 63, 82, 22, 46, 55, 69, 74, 112, 134, 186, 175, 119, 213, 416, 312, 343, 264, 119, 186, 218, 343, 417, 845, 951, 124, 209, 49, 617, 856, 924, 936, 72, 19, 28, 11, 35, 42, 40, 66, 85, 94, 112, 65, 82, 115, 119, 236, 244, 186, 172, 112, 85, 6, 56, 38, 44, 85, 72, 32, 47, 63, 96, 124, 217, 314, 319, 221, 644, 817, 821, 934, 922, 416, 975, 10, 22, 18, 46, 137, 181, 101, 39, 86, 103, 116, 138, 164, 212, 218, 296, 815, 380, 412, 460, 495, 675, 820, 952

Druga šifra je uspešno dešifrovana oko 1885. godine. U nastavku su objašnjene glavne karakteristike ove vrste šifre.

Pošto brojevi u šifri daleko premašuju broj slova u alfabetu, možemo pretpostaviti da nije reč o šifri zamene niti transpozicije. Takođe, možemo pretpostaviti da svaki broj predstavlja jedno slovo, pri čemu se to slovo dobija iz reči u nekom spoljašnjem tekstu. Šifru koja prati ovaj kriterijum nazivamo **šifra knjige**: u slučaju šifre knjige, knjiga ili bilo koji drugi tekst može se koristiti kao ključ. Efektivni ključ, u ovom slučaju, je metod izvlačenja slova iz teksta.

Tim sistemom, druga šifra je dešifrovana zahvaljujući Deklaraciji nezavisnosti Sjedinjenih Američkih Država. Svakoj reči referentnog teksta (Deklaracije nezavisnosti) dodeljen je broj, a uzimanjem prvog slova svake izabrane reči, u skladu sa redosledom brojeva u šifri, može se izvesti dešifrovan tekst. Ono što je posebno domišljato kod ove šifre jeste činjenica da je tekst ključa (Deklaracija nezavisnosti) javno dostupan, ali je način šifrovanja bio poznat samo onome kome je poruka namenjena. Tek kada osoba poseduje ključ (spisak brojeva) i tekst ključa, može lako dešifrovati poruku.

Evo kako izgleda proces dešifrovanja druge šifre:

1. Dodeliti svakoj reči teksta broj, od prve do poslednje.
2. Izvući prvo slovo svake reči, pomoću brojeva sadržanih u šifri.
3. Pročitati otvoren tekst.

U nastavku je prikazan prvi deo Deklaracije nezavisnosti (do 115. reči), gde je svakoj reči dodeljen odgovarajući broj:

When(1) in(2) the(3) course(4) of(5) human(6) events(7) it(8) becomes(9) necessary(10) for(11) one(12) people(13) to(14) dissolve(15) the(16) political(17) bands(18) which(19) have(20) connected(21) them(22) with(23) another(24) and(25) to(26) assume(27) among(28) the(29) powers(30) of(31) the(32) earth(33) the(34) separate(35) and(36) equal(37) station(38) to(39) which(40) the(41) laws(42) of(43) nature(44) and(45) of(46) nature's(47) god(48) entitle(49) them(50) a(51) decent(52) respect(53) to(54) the(55) opinions(56) of(57) mankind(58) requires(59) that(60) they(61) should(62) declare(63) the(64) causes(65) which(66) impel(67) them(68) to(69) the(70) separation(71) we(72) hold(73) these(74) truths(75) to(76) be(77) self(78) evident(79) that(80) all(81) men(82) are(83) created(84) equal(85) that(86) they(87) are(88) endowed(89) by(90) their(91) creator(92) with(93) certain(94) unalienable(95) rights(96) that(97) among(98) these(99) are(100) life(101) liberty(102) and(103) the(104) pursuit(105) of(106) happiness(107) that(108) to(109) secure(110) these(111) rights(112) governments(113) are(114) instituted(115) ...

Sledeći brojevi predstavljaju prve redove druge šifre; kao što možete primetiti, podebljane reči (sa odgovarajućim brojevima) odgovaraju brojevima koje nalazimo u šifrovanom tekstu:

115, 73, 24, 807, 37, 52, 49, 17, 31, 62, 647, 22, 7, 15, 140, 47, 29, 107, 79, 84, 56, 239, 10, 26, 811, 5, 196, 308, 85, 52, 160, 136, 59, 211, 36, 9, 46, 316, 554, 122, 106, 95, 53, 58, 2, 42, 7, 35...

U nastavku je rezultat dešifrovanja pomoću šifre u kombinaciji sa *tekstom ključa* (Deklaracija nezavisnosti Sjedinjenih Američkih Država), pri čemu se uzima prvo slovo svake odgovarajuće reči (tj. iz *otvorenog teksta*). Na primer:

- *115 = instituted = I*
- *73 = hold = h*
- *24 = another = a*
- *807 (nedostaje) = v*
- *37 = equal = e*
- *52 = decent = d*
- *49 = entitle = e*

Nisam priložio celu Deklaraciju nezavisnosti Sjedinjenih Američkih Država; ovo je samo prvih 115 reči. Ako želite, možete posetiti ovu stranicu *http://www.unmuseum.org/bealepap.htm* i vežbati rekonstrukciju na celom otvorenom tekstu.

Evo rekonstrukcije prve rečenice (sa nekoliko izostalih slova):

I have deposited in the county of Bedford...

Ako nastavimo i uporedimo brojeve sa odgovarajućim brojevima početnih slova u Deklaraciji nezavisnosti, dešifrovanje glasi ovako:

I have deposited in the county of Bedford, about four miles from Buford's, in an excavation or vault, six feet below the surface of the ground, the following articles, belonging jointly to the parties whose names are given in number '3,' herewith:

The first deposit consisted of one thousand and fourteen pounds of gold, and three thousand eight hundred and twelve pounds of silver, deposited November, 1819. The second was made December, 1821, and consisted of nineteen hundred and seven pounds of gold, and twelve hundred and eighty-eight pounds of silver; also jewels, obtained in St. Louis in exchange for silver to save transportation, and valued at $13,000.

The above is securely packed in iron pots, with iron covers. The vault is roughly lined with stone, and the vessels rest on solid stone, and are covered with others. Paper number "1" describes the exact locality of the vault so that no difficulty will be had in finding it.

Mnogi drugi kriptografi i kriptolozi su bezuspešno pokušavali da dešifruju prvu i treću Bilovu šifru. Neki, poput lovca na blago *Mela Fišera*, koji je otkrio stotine miliona dolara vredne dragocenosti ispod mora, otišli su u Bedford da pretraže područje kako bi pronašli blago, ali bez uspeha.

Možda je Bilova priča samo legenda. Ili je možda istinita, ali niko nikada neće saznati gde je blago, jer niko neće uspeti da dešifruje prvu šifru. Ili, blago nikada neće biti otkriveno jer ga je neko već pronašao.

Šta god bilo, ono što je zaista interesantno u ovoj priči jeste primena tako snažne šifre bez pomoći ikakvih računara ili elektronskih uređaja; šifra je napravljena samo uz pomoć uma, olovke i papira.

Paradoksalno, broj pokušaja potreban za razbijanje šifre ide od 1 do beskonačnosti, pod pretpostavkom da napadač koristi grubu silu i istražuje sve tekstove napisane u svetu tog trenutka. Uz to, šta se dešava ako tekst ključa nije javan, već ga je sam pošiljalac napisao i zadržao u tajnosti? U ovom slučaju, ako kriptolog nema ključ (tj. ne poseduje tekst ključa), verovatnoća da dešifruje šifru je *nula*.

Bilove šifre su interesantne i jer bi ta vrsta algoritma mogla u budućnosti da dobije nove primene u modernoj kriptografiji. Neke od tih primena mogle bi biti povezane sa metodima istraživanja šifrovanih podataka u računarstvu u oblaku.

Vernamova šifra

Vernamova šifra ima najviši stepen sigurnosti, jer je teoretski potpuno bezbedna. Pošto koristi zaista slučajan ključ iste dužine kao i originalna poruka, to je **savršena šifra**. Na osnovu Šenonovog principa entropije informacije, pitanje bezbednosti svodi se na entropiju i slučajnost koja određuje jednaku verovatnoću za svaki bit sadržan u šifrovanom tekstu. Ovaj algoritam ćemo ponovo razmotriti u *poglavlju 9*, *Kvantna kriptografija*, gde ćemo govoriti o kvantnoj raspodeli ključa i povezanom metodu šifrovanja otvorenog teksta nakon određivanja kvantnog ključa. Još jedna interesantna primena je Hyper Crypto Satellite, koji koristi ovaj algoritam za šifrovanje otvorenog teksta generisanog slučajnim ključem, prenetim satelitskom radiokomunikacijom, kao beskonačnim nizom bitova.

Za sada, pogledajmo glavne karakteristike ovog algoritma.

Suštinski element algoritma je upotreba ključa samo jednom po sesiji. Drugi zahtev je da ključ mora biti iste dužine kao i poruka. Ove osobine čine algoritam otpornim na napade na šifrovan tekst; čak i u retkom slučaju da ključ bude ukraden, promenio bi se u narednom prenosu. Ključ iste dužine kao i poruka izbegava problem kratkih poruka, što ćemo kasnije objasniti. Na kraju, ključ mora biti potpuno slučajan.

Metod je vrlo jednostavan: dodavanjem ključa poruci (po modulu 2) bit po bit, dobija se šifrovan tekst. Ovaj metod, nazvan **isključiva disjunkcija** (**XOR**), sretaćemo mnogo puta u knjizi, posebno kada budemo govorili o simetričnom šifrovanju u *poglavlju 2*, *Algoritmi simetričnog šifrovanja*. Zapamtite samo da ključ mora biti iste dužine kao i poruka.

Primer s brojevima je sledeći:

- 00101001 (otvoren tekst)
- 10101100 (ključ): Sabiranje svakog bita (po modulu 2)
- 10000101 (šifrovani tekst)

Metod Vernamove šifre prati četiri koraka:

1. Transformisati otvoren tekst u niz bitova pomoću ASCII koda.
2. Generisati slučajan ključ iste dužine kao otvoren tekst.
3. Šifrovati poruku sabiranjem bit po bit (po modulu 2) otvorenog teksta i ključa, čime se dobija šifrovan tekst.
4. Dešifrovati poruku obrnutim postupkom sabiranja šifrovanog teksta ključa ponovo se dobija otvoren tekst.

Za primer sa brojevima i slovima, vratićemo se reči HELLO. Pretpostavimo da svako slovo odgovara određenom broju, počev od 0 = A, 1 = B, 2 = C, 3 = D, 4 = E ... i tako dalje do 25 = Z.

Slučajan ključ je [DGHBC].

Šifrovanje daje sledeću transpoziciju:

Otvoren tekst		H E L L O
		7 4 11 11 14
Ključ	=	D G H B C
	+	3 6 7 1 2
	=	10 10 18 12 16
Šifrovan tekst		K K S M Q

Slika 1.10: Šema šifrovanja u Vernamovom algoritmu

Dakle, nakon transpozicije slova, šifrovanje za [HELLO] je (KKSMQ).

Možete sami uraditi vežbu dešifrovanja (KKSMQ) šifrovanog teksta pomoću Vernamove šifre obrnutim postupkom: primenjujući funkciju **f[-K]** na šifrovani tekst vraćate otvoreni tekst [HELLO].

Jedan od napada na koji mnogi algoritmi mogu biti podložni je poznat kao **napad samo na šifrovani tekst**. Ovaj napad je uspešan ako napadač može da izvuče otvoreni tekst ili, još bolje, ključ, koristeći šifrovani tekst ili njegove delove. Najčešće tehnike su analiza učestanosti i analiza saobraćaja. Korišćenjem analize podataka i učestanosti, nameravam da proučim upotrebu slova ili grupa slova u šifrovanom tekstu. Na primer, slovo e je jedno od najčešće korišćenih u engleskom jeziku, tako da možemo planirati napad zasnovan na učestanostima ovog slova. Drugim rečima, pretpostavimo da ako analiziramo saobraćaj šifrovanih podataka i naiđemo na čestu pojavu nekog karaktera u šifrovanom tekstu, moglo bi se pretpostaviti da se radi o slovu e.

Ovaj algoritam nije podložan napadima koji se oslanjaju samo na šifrovani tekst. Štaviše, ako je poznat deo ključa, biće moguće dešifrovati samo deo koji odgovara odgovarajućim bitovima. Ostatak šifrovanog teksta će biti teško dešifrovati ako je dovoljno dug. Međutim, uslovi u vezi sa implementacijom ovog algoritma su veoma restriktivni kako bi se postigla apsolutna nepovredivost. Prvo, generisanje ključa mora biti potpuno slučajno. Drugo, ključ i poruka moraju biti iste dužine, a treće, uvek postoji problem prenosa ključa.

Ovaj poslednji problem utiče na sve simetrične algoritme i suštinski je problem koji je naveo kriptografe da izmisle asimetrično šifrovanje za razmenu ključeva između Alise i Boba (što ćemo videti u narednom poglavlju).

Drugi problem se tiče dužine ključa: ako je poruka prekratka – na primer, reč "deset", koja označava vreme vojnog napada – napadač bi mogao da se osloni na svoj instinkt ili sreću. Nije važno da li postoji slučajan ključ za kratku poruku. Poruka se može intuitivno dešifrovati ako napadač zna temu prenosa. S druge strane, ako je poruka vrlo duga, primorani smo da koristimo veoma dug ključ. U ovom slučaju, proizvodnja ključa će biti veoma skupa, kao i njegov prenos. Pored toga, uzimajući u obzir da se ključ mora menjati za svaku novi prenos, troškovi implementacije ove šifre u komercijalne svrhe su veoma visoki.

Zato su se, generalno, za vojne svrhe tokom Drugog svetskog rata i posle koristile *monokorisne niske* poput ove. Kao što sam ranije rekao, ovo je bio legendarni algoritam koji je korišćen za *crvenu liniju* između Vašingtona i Moskve kako bi se šifrovale komunikacije između predsednika Sjedinjenih Američkih Država i Sovjetskog Saveza tokom Hladnog rata.

Na kraju, analiziraćemo implementaciju ovog algoritma. Moglo bi biti teško pronaći način za generisanje i prenos slučajnog ključa, čak i ako je bezbednost metode vrlo visoka. U poslednjem delu ove knjige predstaviću novu metodu za prenos i implementaciju ključeva koristeći Vernamovu šifru u kombinaciji sa drugim algoritmima i metodama. Ovaj novi **sistem jednokratne šifre**, nazvan *Hyper Crypto Satellite*, mogao bi se koristiti za autentifikaciju i šifrovanje poruka.

Takođe ću vam pokazati moguće ranjivosti sistema i kako generisati veoma slučajan ključ. Ova metoda je bila kandidat za Vernamovu šifru na **Međunarodnoj konferenciji o svemiru**, ali sam tada odlučio da je ne predstavim javnosti.

Napomene o bezbednosti i računski aspekti

Svi algoritmi koje smo do sada proučili u ovom poglavlju spadaju u simetrične. Međutim, osnovni problem ostaje nerešen: prenos ključa. Kao što je već pomenuto, ovaj problem će biti prevaziđen asimetričnom kriptografijom, koja će biti obrađena u narednom poglavlju. U ovom delu ćemo analizirati računarski problem koji je uopšteno povezan sa bezbednošću kriptografskih algoritama. Kasnije ćemo se fokusirati na bezbednost svakog algoritma koji budemo analizirali.

Da napravimo poređenje, mogli bismo reći da je u kriptografiji *lanac slab koliko i njegova najslabija karika*. To je slično problemu kada se koristi veoma jak kriptografski algoritam za zaštitu podataka, ali se šifra za pristup ostavlja na ekranu računara. Drugim rečima, kriptografski algoritam mora biti dovoljno bezbedan i u matematičkom smislu. Na primer, problemi faktorizacije i diskretnog logaritma trenutno imaju slične računarske karakteristike; međutim, ukoliko bi jedan od ovih problema bio rešen, algoritam koji se oslanja na oba ne bi bio koristan.

Hajde da dodatno analiziramo neke opšteprihvaćene principe u kriptografiji. Prva izjava glasi: *kriptografija mora biti otvorenog koda*.

Pod pojmom *otvoreni kod* misli se na algoritam, ali ne i na ključ. Drugim rečima, oslanjamo se na **Kerkhofsov princip** koji glasi:

> *Kriptosistem bi trebalo da bude bezbedan čak i ako su svi njegovi detalji, osim ključa, javno dostupni.*
>
> *Ovaj princip ne važi samo za kodove i šifre već i za sisteme bezbednosti, uopšte: svaka tajna može biti potencijalna tačka slabosti. Tajnost, drugim rečima, uzrokuje krhkost – i može učiniti sistem sklonim katastrofalnom kolapsu. S druge strane, otvorenost obezbeđuje otpornost.*
>
> *– Brus Šnajer*

U praksi, algoritam koji stoji u osnovi šifre mora biti poznat. Nije korisno, a takođe može biti i opasno, oslanjati se na tajnost algoritma za razmenu tajnih poruka. Razlog je jednostavan: ako algoritam treba da koristi otvorena zajednica (kao što je internet), nemoguće ga je držati u tajnosti.

Druga izjava glasi: *bezbednost algoritma zavisi u velikoj meri od matematičkog problema na kom se zasniva*.

Na primer, RSA, jedan od najpoznatijih i najšire korišćenih algoritama u istoriji kriptografije, oslanja se na matematički problem faktorizacije.

Faktorizacija podrazumeva razlaganje broja na njegove delioce. Na primer:

$$21 = 3 * 7$$

Veoma je lako pronaći delioce broja 21, koji su 3 i 7, i za male brojeve, ali je poznato da povećanjem broja cifara problem faktorizacije eksponencijalno raste.

Detaljnije ćemo analizirati asimetrične algoritme, kao što je RSA, a posebno u *poglavlju 3*, *Algoritmi asimetričnog šifrovanja*, gde ću objasniti asimetrično šifrovanje. Ali ovde je dovoljno objasniti zašto se RSA koristi za zaštitu finansijskih tajni, obaveštajnih tajni i drugih osetljivih informacija.

Razlog za to leži u činjenici da je matematički problem koji stoji iza RSA (faktorizacija) i dalje teško rešiv za računare današnje generacije. Ipak, u ovom uvodu nećemo se dublje baviti RSA algoritmom, pa ću samo reći da RSA ne zavisi samo od faktorizacije kao potencijalne slabe tačke, već i od drugog problema slične težine, a to je problem **diskretnog logaritma**. Kasnije ćemo analizirati oba ova kompleksna računarska problema. Za sada, pretpostavićemo (iako netačno, kao što to čini većina kriptografskih tekstova) da se bezbednost RSA algoritma oslanja na faktorizaciju. U *poglavlju 6*, *Inovacije u kriptografiji i logički napadi*, pokazaću napad na RSA algoritam koji se oslanja na problem drugačiji od faktorizacije. To je ekvivalent *slabe karike u lancu* koja, ako popusti, narušava bezbednost algoritma.

Da vidimo šta se dešava kada pokušamo da razbijemo RSA oslanjajući se samo na problem faktorizacije, pomoću grube sile. Na primer, samo za ilustraciju računarske snage potrebne da se faktoriše RSA broj od 250 cifara, razlaganje velikog poluprostog broja je izrazito teško ako broj ima stotine ili hiljade cifara.

Da bismo ilustrovali, RSA-250 je 829-bitni broj koji se sastoji od 250 decimalnih cifara, a današnji računari veoma teško mogu da ga razbiju.

Ovaj broj je faktorizovan u februaru 2020. godine, a ceo proces je trajao približno 2.700 godina jezgra na procesoru **Intel Xeon Gold 6130** pri brzini 2,1 GHz. Kao i kod mnogih rekorda u faktorizaciji, ovo je ostvareno pomoću mreže računara i optimizovanog algoritma za ubrzanje izračunavanja.

Treća važna izjava je sledeća: *praktična bezbednost je uvek manje bezbedna od teorijske bezbednosti*.

Na primer, kada analiziramo Vernamovu šifru, lako možemo shvatiti koliko je u praksi teško implementirati ovaj algoritam. Vernamov algoritam je, dakle, teoretski neosvojiv, ali ne i u praktičnom smislu. Zaključak ove tvrdnje glasi: implementiranje algoritma podrazumeva sprovođenje njegove teorijske šeme i dodavanje velike složenosti. *Složenost je neprijatelj bezbednosti*: što je sistem složeniji, to je više mogućih tačaka napada.

Dalje razmatranje odnosi se na nivo bezbednosti algoritma. Ovaj koncept može se bolje razumeti kroz Šenonovu teoriju i koncept *savršene tajnosti*. Definicija koju je 1949. godine formulisao Klod Šenon bazira se na statistici i verovatnoćama. Za najviši nivo bezbednosti, Šenon je teorijski tvrdio da šifrovan tekst održava savršenu tajnost ako je stepen informacija o sadržaju poruke isti za napadača pre i posle pregleda šifrovanog teksta, čak i sa neograničenim resursima. To znači da poruka napadaču ne otkriva nikakvu informaciju o sadržaju poruke.

Da bismo bolje razumeli ovaj koncept, zamislimo različite nivoe bezbednosti, gde svaki nivo pruža određenu bezbednost, ali opadajućim stepenom. Drugim rečima, najviši nivo je najjači, dok su niži slabiji; međutim, postoji srednja zona sa neodređenim nivoom bezbednosti koja zavisi od tehnološkog nivoa napadača.

Nije bitno koliko nivoa se smatra bezbednim, a koliko ne. Suština je u tome da je važno razumeti šta je trenutno bezbedno i šta nije, ali i šta se u određenom trenutku može smatrati bezbednim. Sa tim na umu, evo razlike između kriptosistema sa savršenom tajnošću i onog koji je bezbedan:

- Kriptosistem ima *savršenu tajnost* ako zadovoljava najmanje dva uslova:
- Ne može se razbiti čak ni neograničenom računarskom snagom.

- Analizom kriptograma *[c]* nije moguće dobiti nikakve informacije o poruci, *[m]*, i ključu, *[k]* (npr., Vernamov sistem je teoretski savršeno bezbedan, ali samo pod određenim uslovima).
- Kriptogram je *bezbedan* čak i ako teoretski napadač može razbiti kriptosistem (npr., da poseduje kvantne računare i algoritam faktorizacije koji se dobro izvršava), ali se smatra da je osnovni matematički problem u tom trenutku veoma teško rešiti. Pod određenim uslovima, šifre kao što su RSA, Difi-Helman i ElGamal mogu se koristiti, jer se na osnovu empirijskih dokaza faktorizacija i diskretni logaritmi i dalje smatraju teškim problemima za rešavanje.

Koncept bezbednosti je, dakle, dinamičan i nejasan. Ono što je danas bezbedno, sutra možda neće biti. Šta će se desiti sa RSA i celokupnom klasičnom kriptografijom ukoliko kvantni računari postanu dovoljno snažni ili se razvije algoritam koji je sposoban da razbije problem faktorizacije? Ova pitanja će biti razmatrana u *poglavlju 9*, *Kvantna kriptografija*. Za sada možemo reći da će klasični kriptografski algoritmi postati ranjivi na disruptivnu računarsku moć kvantnih računara, ali ne znamo kada će se to dogoditi.

Pod određenim uslovima, videćemo da *kvantna razmena ključa* može biti smatrana *sistemom sa savršenom tajnošću*, ali to nije uvek slučaj i trenutno nije u širokoj upotrebi. Neki OTP sistemi mogli bi se smatrati visoko bezbednim (možda sa polusavršenom tajnošću), ali sve zavisi od praktične primene. Konačno, zapamtimo važnu lekciju: slaba karika u lancu.

U zaključku možemo primetiti sledeće:

- Kriptografija mora biti otvorenog koda (algoritmi moraju biti poznati), osim za ključ.
- Bezbednost algoritma u velikoj meri zavisi od matematičkog problema na kom se zasniva.
- Složenost je neprijatelj bezbednosti.
- Bezbednost je dinamičan koncept: savršena bezbednost je samo teorijska pretpostavka.

Rezime

U ovom poglavlju obradili smo osnovne pojmove kriptografije; osvežili smo znanje o binarnom sistemu i ASCII kodu, a takođe smo istražili proste brojeve, Fermaove jednačine i modularnu matematiku. Zatim smo dali pregled klasičnih kriptografskih algoritama, poput Cezarovog, Bilovog i Vernamovog.

Na kraju, u poslednjem delu, analizirali smo bezbednost iz filozofske i tehničke perspektive, i odredili nivoe bezbednosti u kriptografiji u odnosu na stepen složenosti.

U sledećem poglavlju istražićemo simetrično šifrovanje, gde ćemo detaljno obraditi algoritme kao što su **Standard za šifrovanje podataka** (**DES**) i porodica **AES** algoritama, kao i neke od problema koji su pomenuti u ovom poglavlju.

Pridružite se našoj zajednici na Discord platformi!

Čitajte ovu knjigu zajedno sa drugim korisnicima, postavljajte pitanja, pomažite u rešavanju problema i još mnogo toga.

Da se pridružite zajednici, skenirajte QR kod ili posetite link:

https://packt.link/SecNet