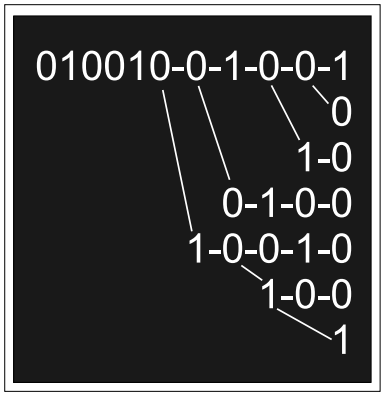
$\pi\mathrm{l}\alpha y\sqrt{\mathbf{mat}\chi}$

Mjesec matematičke svjesnosti – matematika i sigurnost Interneta

Vanja Vagner

Mjesec matematičke svjesnosti **obilježava se svake godine u travnju**, s ciljem da se poveća razumijevanje i zanimanje javnosti za matematiku. Matematika igra važnu ulogu u brojnim aspektima našeg života - kao temelj i podrška gotovo svim granama industrije i znanosti. Stoga



je promicanje njenog razvoja i primjene glavni cilj ove akcije, koja je prije 20 godina pokrenuta u SAD-u. Tijekom travnja diljem Sjedinjenih Država održavaju se brojne manifestacije - od radionica, natjecanja i predavanja, do izložbi i festivala. Svake se godine određuje nova tema, a ovogodišnja je ***Matematika i sigurnost Interneta***.

Razdoblje Interneta donijelo je brojne nove mogućnosti. Od sada možete kupovati, obavljati novčane transakcije i istovremeno komunicirati s tetom iz Zimbabvea, a da ni ne napustite udobnost svoga doma. Naravno, za sve je to potreban visoki stupanj sigurnosti i privatnosti, koji nije lako postići u tako otvorenoj i slobodnoj okolini kao što je Internet. Nesmetan protok informacija jedan je od glavnih problema sigurnosti Interneta, pa je tako i njegovo rješavanje od velike važnosti.

Jedna od najkorisnijih grana matematike u tom području je **kriptografija** (ili *tajno pismo*), koja se posebno bavi problemom razmjene privatnih podataka između dvije strane putem javnog kanala. Kad se te dvije strane poznaju, mogu koristiti unaprijed dogovorenu privatnu šifru. Problem nastaje kada se dvije strane ne poznaju, tj. nikad prije nisu komunicirale ili nisu bile u prilici dogovoriti šifru. Primjerice, kupujete CD preko Interneta od diskografske kuće s kojom niste imali apsolutno nikakvog prethodnog kontakta. Podatke koje šaljete (ime, prezime, adresu, broj kreditne kartice,...) potrebno je zaštititi da bi se transakcija uspješno obavila, no za takav vam je pothvat potrebno ipak nešto malo više od obične šifre.

Šifre i ključevi

Većina šifri određena je **ključem** – pomoćnim podatkom o kojem ovisi postupak *šifriranja*, odnosno *dešifriranja*. Odmah je jasno da nije svaka šifra jednako sigurna i pouzdana. Na primjer, "Cezarova šifra", koju je navodno koristio **Julije Cezar** pri slanju važnih poruka, kao ključ koristi pomicanje pojedinog slova za 3 mjesta udesno (A se zamjenjuje s D, B sa E, itd.). Očito takva šifra nije pretjerano sigurna, budući da je lako pogoditi zamjenu slova i dekodirati poruku ☹ (dekodiranje se vrši pomicanjem za 3 mjesta u lijevo).

U odnosu na Cezarovu šifru, veliko je poboljšanje šifriranje "*množenjem*" pojedinog slova (npr sa 7). Ako slova abecede zamijenimo redom brojevima od 1 do 30, tada je zamjena slova zastupljenog

$\pi \mathbf{l} \alpha y \sqrt[\chi]{\mathbf{mat}}$

brojem k slovo zastupljeno brojem $7k$ mod 30 (ostatkom pri dijeljenju broja $7k$ s brojem 30). Dakle slovo A (1) prelazi u slovo DŽ (7), slovo G (11) u LJ (17), itd. Da bismo dešifrirali poruku pojedino slovo trebamo pomnožiti s 13 i naći ostatak dobivenog broja pri dijeljenju s 30 – slova koja su zastupljena tako dobivenim brojevima otkrivaju početnu poruku.

Dok se kod Cezarove šifre postupak šifriranja lako uoči, kod šifririranja množenjem nije odmah sasvim jasno kojim smo se postupkom koristili, što čini ovaj tip šifriranja uvelike sigurnijim.

Običan tekst		ZNAŠ ODGOVOR NA PETO
Cezarova šifra	→	BPDV SEJSAST PD ŠHZS
Množenje s 7	→	RI(DŽ)Š UH(LJ)ULUG I(DŽ) ČCBU

Da bi tehnika množenja funkcionirala, broj kojim "množimo" mora biti relativno prost s brojem 30 (brojevi su međusobno relativno prosti ako osim 1 nemaju zajedničkih djelitelja), a kako brojeva relativno prostih s 30 i manjih od 30 ima 7 (ne računajući 1, jer množenjem s 1 dobijemo početni tekst, pa ga ne smatramo šifrom), ograničeni smo na samo 7 različitih parova ključeva za šifriranje i dešifriranje, što također olakšava njeno razbijanje. Dakle, uzimanjem većeg modula smanjuje se vjerojatnost razbijanja šifre, tj. otkrivanja ključa za dešifriranje. Poruku zapisanu u ASCII kodu možemo uspješnije šifrirati, budući da u tom slučaju promatramo modul 128, pa je ukupan broj parova ključeva jednak 64 (ukupan broj brojeva relativno prostih sa (i manjih od) 128 je 64). Još je bolji odabir *blok-šifra* koju možemo konstruirati tako da grupiramo pojedina slova poruke u grupe od po 4 člana. Tako će naša gornja rečenica sada izgledati "ZNAŠ ODGO VORN APET O?", pa blok znaš sada zamjenit brojem 29190125 (Z - 29, N - 19, A - 01, Š - 25), te šifrirati ga modulo 30303030. Uočimo da nam ovo grupiranje u blokove sada nudi čak 7 833 600 parova ključeva, što znatno smanjuje opasnost od razbijanja šifre.

Kriptografija javnog ključa i RSA algoritam

Pokazali smo da ako imamo dovoljno snažnu šifru i ako obe strane poznaju pripadne ključeve, međusobno slanje poruka ne predstavlja problem. No postavlja se novo pitanje - kad nađemo dovoljno "dobru" šifru za kodiranje podataka, kako, bez opasnosti od presretanja, prenijeti potrebne ključeve strani s kojom nismo imali nikakvog prethodnog kontakta? Dosad najefikasnije rješenje tog problema je kriptografija javnog ključa. Ideja je konstruirati takav par ključeva, da ključ za dešifriranje bude (gotovo) nemoguće odrediti iz ključa za šifriranje. Ako želimo primati "osjetljive" podatke od različitih strana, a da smo jedino mi u mogućnosti dešifrirati ih, dovoljno je objaviti drugim stranama ključ za šifriranje (javni ključ), a ključ za dešifriranje (tajni ključ) zadržati za sebe. Za sigurno razmjenjivanje poruka dovoljno je da obe strane jedna drugoj pošalju svoje javne ključeve, pomoću kojih mogu jedna drugoj šifrirati poruke. Ako i neka treća osoba sazna njihove javne ključeve i presretne poruku, bez tajnog ključa (koji je poznat samo primatelju) neće biti u mogućnosti dešifrirati ju.

No kako konstruirati takav par ključeva? Dosad najbolje rješenje tog problema pružio je RSA (**Rivest - Shamir - Adleman**) algoritam, otkriven 1977 g. na MIT-u[1]. Pokazali smo da je šifra množenjem pouzdana, ali se ključ za dešifriranje može efikasno izračunati iz ključa za šifriranje (koristeći Euklidov algoritam). Čak i kad su u pitanju jako veliki brojevi, do rješenja se dolazi prilično lako, pa takva šifra nije pogodna za razmjenu podataka na mreži. Sljedeći je logični korak promatranje šifre *potenciranjem*, koja poruku diže na potenciju a, uz određeni modul n (kao na već prije opisan način; A prelazi u A, B u slovo zastupljeno brojem 2^a mod n, itd.). Da bi se proces šifriranja mogao obrnuti (tj. da bi se mogao dobiti ključ za dešifriranje) potrebno je da su brojevi $\varphi(n)$ i a međusobno relativno prosti, pri čemu je $\varphi(n)$ ukupan broj brojeva relativno prostih s n i manjih od n, dan formulom:

$$\varphi(n) = n\left(1-\frac{1}{p_1}\right)\cdot\left(1-\frac{1}{p_2}\right)\cdots\left(1-\frac{1}{p_k}\right),$$

gdje su $p_1,\ p_2,\ \ldots\ p_k$ prosti djelitelji broja n.

[1] Massachusetts Institute of Technology

Da bi šifra bila pouzdana, za modul i eksponent moramo uzeti velike brojeve a i n, jer se inače kod malih brojeva razbijanje šifre jednostavno svodi na isprobavanje svih mogućih parova ključeva. No takvim odabirom velikih brojeva prilično je otežano računanje broja $\varphi(n)$, budući da je rastavljanje velikih brojeva na proste faktore zahtjevan posao. Kako bismo izbjegli taj problem, za modul n uzimamo umnožak dva velika prosta broja, pa je sada $\varphi(n) = (p-1)(q-1)$. Sama generacija ključeva izvršava se na slijedeći način:

1. Odaberemo par nasumičnih, velikih prostih brojeva p i q. Prosti brojevi iznenađujuće su česti, a i imaju brojna svojstva koja ih razlikuju od složenih. Prema tome, testiranja prostih brojeva svode se na utvrđivanje tih svojstava, umjesto na obično rastavljanje na faktore. To uvelike pojednostavljuje postupak nasumičnog odabira prostih brojeva, što osigurava isplativost ovog algoritma.

2. Modul n dobijemo kao umnožak prostih brojeva p i q.

$$n := p \cdot q.$$

3. Odaberemo neparni eksponent a između 3 i $n-1$, takav da je relativno prost sa $p-1$ i $q-1$.

4. Tajni eksponent b dobijemo rješavanjem jednadžbe $a \cdot b \bmod \varphi(n) = 1$, što ne predstavlja problem ukoliko znamo proste faktore p i q.

5. Tako dobiveni eksponenti a (javni ključ koji objavimo svim zainteresiranim stranama) i b (tajni ključ koji zadržimo za sebe) čine par ključeva pogodan za šifriranje podataka.

Kad malo bolje pogledamo, ovako generirani par ključeva nije apsolutno siguran, jer pronalaženje prostih faktora broja n automatski daje eksponent b. No, ovaj algoritam svoju efikasnost opravdava činjenicom da je unatoč stotinama godina istraživanja, za **rastavljanje velikih brojeva na proste faktore** i dalje potrebno **dosta vremena**. Za jako velike brojeve taj se postupak čak ne može ni obaviti u realnom vremenu. Tako bi na primjer za rastavljanje 300-znamenkastog broja na proste faktore, koristeći pri tome najsuvremeniju tehnologiju i najjača računala, bilo potrebno čak **godinu dana**. ☹ Uzimajući za p i q jako velike brojeve (svaki od po 300-tinjak znamenaka ili više), nekome tko ne poznaje brojeve p i q rastavljanje broja n činimo gotovo nemogućim. ☺ Lako računanje ključeva za šifriranje i njezino gotovo nemoguće razbijanje čine ovu šifru veoma uspješnom, pa je stoga zastupljena u gotovo svim današnjim sigurnosnim sustavima.

Naravno, potreba za daljnjim razvojem i unaprijeđenjem već postojećih metoda uvijek postoji. I ovako pouzdana metoda kao što je RSA šifriranje lako bi mogla biti poljuljana otkrićem neke nove efikasne metode rastavljanja brojeva na proste faktore. Očigledna je važnost što ranijeg otkrivanja takve metode ili opovrgavanja njenog postojanja, te pronalaženja novih alternativnih metoda. Naoko sasvim beznačajan problem, efikasno rastavljanje brojeva na proste faktore, sada poprima potpuno novu dimenziju, ukazujući pritom na **rastuću važnost matematike u današnjem životu**.

Literatura

[1] *Andrej Dujella*: Kriptografija (predavanja s istoimenog kolegija), `http://web.math.hr/~duje/kript/kriptografija.htm`

[2] *Andrej Dujella*: Vigerenova šifra, math.e br. 1, HMD, Zagreb 2004. `http://www.math.hr/~mathe/vigenere`

[3] *Azra Tafro:* Kongruencije, *PlayMath* br. 3 (2003.)

[4] STRANICA MJESECA MATEMATIČKE SVJESNOSTI, `http://www.mathaware.org`

Prilozi o šifriranju *PlayMath*-u

[1] *Marko Horvat:* Šifriranje matricama, *PlayMath* br. 3 (2003.)

[2] *Tvrtko Tadić:* Matrice u MAPLE-u, *PlayMath* br. 3 (2003.)