# ЗАШТИТА ПОДАТАКА

## Шифровање јавним кључем и хеш функције

Увод у теорију бројева

# Преглед

- Биће објашњено:
  - прости бројеви
  - Fermat-ова и Euler-ова теорема
  - тестирање простости броја
  - прости корени
  - дискретни логаритми

# Прости бројеви

- прости су они бројеви који немају дељеника осим себе и јединице
  - не могу се записати као производ других бројева
  - 1 је прост, али генерално није од интереса
- нпр. 2,3,5,7 су прости, 4,6,8,9,10 нису
- прости бројеви су у центру пажње у теорији бројева
- листа простих бројева мањих од 200 је:

```
2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59
61 67 71 73 79 83 89 97 101 103 107 109 113 127
131 137 139 149 151 157 163 167 173 179 181 191
193 197 199
```

# Проста факторизација

- факторизовати број $n$ значи написати га као производ других бројева:

$$n=a \times b \times c$$

- факторизација броја је релативно тешка у односу на множење фактора да би се добио број
- Проста факторизација броја $n$ је када се напише као производ простих бројева
  - нпр. $91=7\times13$ ; $3600=2^4\times3^2\times5^2$

# Проста факторизација

- У општем случају може се написати:

$$a = \prod_{p \in P} p^{a_p}$$

при чему је $a_p \geq 0$

- Тада се вредност броја може представити помоћу коефицијената различитих од нула
  - 12: $\{a_2 = 2, a_3 = 1\}$
  - 18: $\{a_2 = 1, a_3 = 2\}$
  - 91: $\{a_7 = 1, a_{13} = 1\}$
- Множење два броја еквивалентно је сабирању кореспондентних експонената
  - $12 * 18 = (2^2 * 3) * (2*3^2) = 216$
  - $a_2 = 2 + 1 = 3$; $a_3 = 1 + 2 = 3$
  - $216 = 2^3 * 3^3 = 8 * 27$

# Узајамно прости бројеви и GCD

- два броја $a$ и $b$ су узајамно прости ако немају заједничког дељеника већег од 1
  - нпр. 8 & 15
- може се одредити највећи заједнички делилац упоређујући просту факторизацију бројева и користећи најмање степене
  - нпр. $300=2^2\times3^1\times5^2$ $18=2^1\times3^2$ дакле $GCD(18,300)=2^1\times3^1\times5^0=6$

# Fermat-ова теорема

- $a^{p-1} \equiv 1 \pmod{p}$
  - где је $p$ прост и $\gcd(a,p)=1$
- корисно у тестирању да ли је број прост и код јавних кључева

# Ојлерова фи функција $\phi(n)$

- када се ради аритметика по модулу n
- **потпуни скуп остатака је**: $0..n-1$
- **редуковани скуп остатака** чине они бројеви који су узајамно прости у односу на n
  - нпр за n=10,
  - потпуни скуп остатака је {0,1,2,3,4,5,6,7,8,9}
  - редуковани скуп остатака је {1,3,7,9}
- број елемената у редукованом скупу остатака се назива **Ојлерова фи функција ф(n)**

# Ојлерова фи функција $\phi(n)$

- да би се израчунало ф(n) потребно је пребројати колико има елемената који ће бити искључени
- потребна је проста факторизација, али
  - за p (p прост број) $\phi(p) = p-1$
  - за pq (p,q прости бројеви) $\phi(pq) = (p-1)(q-1)$
- нпр.
  - $\phi(37) = 36$
  - $\phi(21) = (3-1)\times(7-1) = 2\times6 = 12$

# Ојлерова теорема

- генерализација Fermat-ове теореме
- $a^{\phi(n)} \equiv 1 \pmod{n}$
  - где је $\gcd(a,n)=1$
- нпр.
  - $a=3; n=10;\ \phi(10)=4;$
  - дакле $3^4 = 81 = 1 \bmod 10$
  - $a=2; n=11;\ \phi(11)=10;$
  - дакле $2^{10} = 1024 = 1 \bmod 11$

# Тестирање простости бројева

- за криптографске алгоритме, важно изабрати на случајан начин један или више великих простих бројева
- задатак је одредити да ли је дати велики број прост?
- не постоји једноставан, а уједно ефикасан начин за постизање поменутог

# Miller-Rabin алгоритам

- Служи за тестирање простости великих бројева
- Сваки непаран позитиван цео број $n \geq 3$, може се написати као $n - 1 = 2^k q$, при чему је $k > 0$ и $q$ непарно.

# Miller-Rabin алгоритам

- Ако је *p* прост број и *a* позитиван цео број мањи од *p* тада је $a^2 \bmod p = 1$, ако и само ако је $a \bmod p = 1$ или $a \bmod p = -1 \bmod p = p - 1$.

# Miller-Rabin алгоритам

- Нека је *p* прост број већи од 2. Можемо да напишемо $p - 1 = 2^k q$, при чему је $k>0$ и *q* непарно. Нека је *a* било који цео број у опсегу $1 < a < p - 1$. Тада је испуњен један од следећа два услова:
  - $a^q \equiv 1 \pmod{p}$ или
  - $a^{2^{j-1}q} \equiv -1 \pmod{p}$, где 1 ≤ j ≤ к.

# Miller-Rabin алгоритам

- Ако је $n$ прост број онда је или први елемент у листи остатака ($a^q$, $a^{2q}$, …, $a^{2^{k-1}q}$, $a^{2^k q}$) по модулу $n$ једнак 1 или је неки од наредних једнак $n - 1$, у супротном $n$ није прост број. Са друге стране, уколико је услов испуњен не мора да значи да је $n$ прост број.

# Miller-Rabin алгоритам

- TEST (n)
- Pronaći k>0 i neparno q, tako da je $n - 1 = 2^k q$
- Izabrati nasumično a, $1 < a < n - 1$
- If $a^q \bmod n = 1$ then return “mozda prost”
- For j=1 to k – 1 do
  - If $a^{2^j q} \bmod n = n - 1$ then return “mozda prost”
- Return “nije prost”

# Miller-Rabin алгоритам

- n = 29
  $n - 1 = 2^2 7$; a = 10; $10^7 \bmod 29 = 17$;
  $(10^7)^2 \bmod 29 = 28$; “mozda prost”
- n = 221
  $n - 1 = 2^2 55$; a = 5; $5^{55} \bmod 221 = 112$;
  $(5^{55})^2 \bmod 221 = 168$; “nije prost”
  a = 21; $21^{55} \bmod 221 = 200$;
  $(21^{55})^2 \bmod 221 = 220$; “mozda prost”

# Miller-Rabin алгоритам

- Вероватноћа да за било који непаран број *n* који није прост и насумично изабрано *a* алгоритам врати “mozda prost” износи мање од ¼. За *t* изабраних вредности за *a* вероватноћа да сви прођу са овим одговором је мања од $(¼)^t$.

# Прости корени

- из Eulerове теореме имамо $a^{\Phi(n)} \equiv 1 \pmod{n}$
- размотримо $a^{m} \equiv 1 \pmod{n}$, $GCD(a,n)=1$
  - мора да важи за m= ф(n) али може бити мање
  - када степеновање дође до m, циклус се понавља

$$7^1 \equiv 7 \pmod{19}$$

$$7^2 = 49 = 2 \times 19 + 11 \equiv 11 \pmod{19}$$

$$7^3 = 343 = 18 \times 19 + 1 \equiv 1 \pmod{19}$$

$$7^4 = 2401 = 126 \times 19 + 7 \equiv 7 \pmod{19}$$

$$7^5 = 16807 = 884 \times 19 + 11 \equiv 11 \pmod{19}$$

# Прости корени

## Степени целих бројева по модулу 19

| $a$ | $a^2$ | $a^3$ | $a^4$ | $a^5$ | $a^6$ | $a^7$ | $a^8$ | $a^9$ | $a^{10}$ | $a^{11}$ | $a^{12}$ | $a^{13}$ | $a^{14}$ | $a^{15}$ | $a^{16}$ | $a^{17}$ | $a^{18}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 2 | 4 | 8 | 16 | 13 | 7 | 14 | 9 | 18 | 17 | 15 | 11 | 3 | 6 | 12 | 5 | 10 | 1 |
| 3 | 9 | 8 | 5 | 15 | 7 | 2 | 6 | 18 | 16 | 10 | 11 | 14 | 4 | 12 | 17 | 13 | 1 |
| 4 | 16 | 7 | 9 | 17 | 11 | 6 | 5 | 1 | 4 | 16 | 7 | 9 | 17 | 11 | 6 | 5 | 1 |
| 5 | 6 | 11 | 17 | 9 | 7 | 16 | 4 | 1 | 5 | 6 | 11 | 17 | 9 | 7 | 16 | 4 | 1 |
| 6 | 17 | 7 | 4 | 5 | 11 | 9 | 16 | 1 | 6 | 17 | 7 | 4 | 5 | 11 | 9 | 16 | 1 |
| 7 | 11 | 1 | 7 | 11 | 1 | 7 | 11 | 1 | 7 | 11 | 1 | 7 | 11 | 1 | 7 | 11 | 1 |
| 8 | 7 | 18 | 11 | 12 | 1 | 8 | 7 | 18 | 11 | 12 | 1 | 8 | 7 | 18 | 11 | 12 | 1 |
| 9 | 5 | 7 | 6 | 16 | 11 | 4 | 17 | 1 | 9 | 5 | 7 | 6 | 16 | 11 | 4 | 17 | 1 |
| 10 | 5 | 12 | 6 | 3 | 11 | 15 | 17 | 18 | 9 | 14 | 7 | 13 | 16 | 8 | 4 | 2 | 1 |
| 11 | 7 | 1 | 11 | 7 | 1 | 11 | 7 | 1 | 11 | 7 | 1 | 11 | 7 | 1 | 11 | 7 | 1 |
| 12 | 11 | 18 | 7 | 8 | 1 | 12 | 11 | 18 | 7 | 8 | 1 | 12 | 11 | 18 | 7 | 8 | 1 |
| 13 | 17 | 12 | 4 | 14 | 11 | 10 | 16 | 18 | 6 | 2 | 7 | 15 | 5 | 8 | 9 | 3 | 1 |
| 14 | 6 | 8 | 17 | 10 | 7 | 3 | 4 | 18 | 5 | 13 | 11 | 2 | 9 | 12 | 16 | 15 | 1 |
| 15 | 16 | 12 | 9 | 2 | 11 | 13 | 5 | 18 | 4 | 3 | 7 | 10 | 17 | 8 | 6 | 14 | 1 |
| 16 | 9 | 11 | 5 | 4 | 7 | 17 | 6 | 1 | 16 | 9 | 11 | 5 | 4 | 7 | 17 | 6 | 1 |
| 17 | 4 | 11 | 16 | 6 | 7 | 5 | 9 | 1 | 17 | 4 | 11 | 16 | 6 | 7 | 5 | 9 | 1 |
| 18 | 1 | 18 | 1 | 18 | 1 | 18 | 1 | 18 | 1 | 18 | 1 | 18 | 1 | 18 | 1 | 18 | 1 |

# Прости корени

- ако је најмањи m= ф(n) тада се a назива **прост корен** за n
- Значај простих корена огледа се у томе што уколико је a прост корен за n тада су његови експоненти

$$a,\ a^2,\ \ldots,\ a^{\phi(n)}$$

различити (по модулу n) и узајамно прости са n.
- Конкретно за прост број p, коме је a прост корен, сукцесивни степени од a "генеришу" групу по mod p
- ови су корисни али релативно тешки за проналажење

# Дискретни логаритми

- инверзан проблем експонентизацији је да се пронађе дискретни логаритам броја по модулу p
- то подразумева налажење x где је $a^x = b \bmod p$
- записано као $x=\log_a b \bmod p$ или $x=\mathrm{dlog}_{a,p}(b)$
- ако је a прост корен онда увек постоји решење, у супротном може да не постоји
  - $x = \log_3 4 \bmod 13$ (значи $3^x = 4 \bmod 13$) нема решење
  - $x = \log_2 3 \bmod 13 = 4$ покушавањем суседних степеновања
- док је експонентизација релативно једноставна, налажење дискретних логаритама је тежак проблем