Dokumentace k druhému projektu do KRY – Implementace a prolomení RSA

Kateřina Fořtová (xforto00)

duben 2022

1 Zadání projektu

Úkolem projektu byla implementace a prolomení RSA – asymetrického algoritmu založeného na problému faktorizace čísel. Program funguje ve čtyřech režimech – generování klíčů, šifrování, dešifrování a faktorizace. Spuštění je pro jednotlivé módy provedeno následovně:

- Generování klíčů: vstup: `./kry -g B`, výstup: `P Q N E D`
- Šifrování: vstup: `./kry -e E N M`, výstup: `C`
- Dešifrování: vstup: `./kry -d D N C`, výstup: `M`
- Prolomení RSA: vstup: `./kry -b N`, výstup: `P`

Níže jsou vysvětleny dané proměnné:

- `B` – požadovaná velikost veřejného modulu v bitech (např. 1024),
- `P` – prvočíslo (při generování náhodné),
- `Q` – jiné prvočíslo (při generování náhodné),
- `N` – veřejný modulus,
- `E` – veřejný exponent (většinou 3),
- `D` – soukromý exponent,
- `M` – otevřená zpráva (číslo, nikoli text),
- `C` – zašifrovaná zpráva (číslo, nikoli text).

2 Implementace

Implementace byla provedena v jazyce C++ s využitím knihovny GMP pro práci s velkými čísly. Vzhledem k užití této knihovny však musel být kladen důraz na uvolnění paměti proměnných, aby nedocházelo k chybám typu memory leak.

Při generování klíčů byl využit algoritmus Miller-Rabin pro generování dvou dostatečně velikých prvočísel. Před samotným během algoritmu je vygenerováno náhodné číslo n o odpovídající velikosti veřejného modulu, kdy je nejvyšší bit nastaven na 1. Pokud je vygenerované číslo sudé a není hodnoty 2, pak nemůže být prvočíslo.

Každý lichý kandidát na prvočíslo n může být rozložen do formy $n-1 = 2^k \cdot u$. Pokud nalezneme náhodně vygenerované číslo a do hodnoty $n-2$, pro které platí, že $a^u \neq 1 mod n$ a $a^{u2^i} \neq n-1 mod n$ pro všechna $i \in \{0, 1, \cdots, k-1\}$, pak a je svědek, že je číslo n složené, jinak je číslo n možné prvočíslo.

Je vygenerováno několik náhodných čísel do hodnoty $n-2$ (tzv. svědci), v implementaci je tato hodnota nastavena na 15. Pokud žádné ze svědků není důkazem toho, že je číslo n složené, pak může být číslo n prohlášeno za pravděpodobné prvočíslo.

Při nalezení prvního vygenerovaného prvočísla p je tato hodnota prvočísla nadále inkrementována dokud není nalezeno i druhé prvočíslo q. Následně je spočítán veřejný modulus n, který je součinem obou vygenerovaných prvočísel a $phi(n) = (p-1) \cdot (q-1)$. Poté je náhodně vygenerován veřejný exponent e v intervalu mezi 1 a $phi(n)$. Pro tohoto kandidáta na veřejný exponent je testováno zda platí, že největší společný dělitel e a $phi(n)$ je 1. Pro nalezení největšího společného dělitele je implementována rekurzivní funkce. Pokud podmínka neplatí, kandidát na veřejný exponent je inkrementován a znovu otestován, dokud není nalezen kandidát splňující danou podmínku.

Soukromý exponent d je vypočítán jako $d = inv(e, phi(n))$ s využitím algoritmu nalezení inverzního prvku (Modular multiplicative inverse). Vstupem tohoto algoritmu jsou čísla a a m, výsledkem je číslo x, pro které platí, že $ax \cong 1(modm)$. Hodnota x pak leží v intervalu $\langle 1, 2, \cdots, m-1 \rangle$.

Bylo využito rozšířeného Euclidova algoritmu, který pro čísla a a b nalezne největšího společného dělitele a také čísla x a y aby platilo následující: $ax + by = gcd(a, b)$. Pro algoritmus nalezení inverzního prvku je $b = m$, následně můžeme v našem případě uvažovat, že $gcd(a, m) = 1$. Výraz lze vyjádřit jako $ax + my \cong 1(modm)$, s úpravou na $ax \cong 1(modm)$.

RSA bylo pro projekt využito pro režim utajení a nikoliv elektronického podpisu. Nechť m je zpráva ve formě celého čísla, e je veřejný exponent a n představuje veřejný modulus. Pak je zašifrovaný text c ve formě celého čísla vypočítán jako: $c = m^e modn$. Dešifrováním soukromým klíčem je z zašifrovaného textu c ve formě celého čísla opět získána zpráva m ve formě celého čísla, je využito soukromého exponentu d: $m = c^d modn$.

Faktorizace spočívá v získání vygenerovaných prvočísel p a q z zadaného veřejného modulu $n = pq$. Zpočátku je provedena metoda triviálního dělení pro prvních 1 000 000 čísel – zkoumá se případná dělitelnost veřejného modulu některým z dělitelů a tedy jeho možného rozložení na dvě čísla. Pokud je tato forma faktorizace neúspěšná, pak se provede efektivnější Fermatova faktorizace.

Fermatova faktorizace byla zvolena z důvodu své jednoduché implementace a uvedení této metody jako příkladu v zadání projektu. Tento přístup je založen na možnosti reprezentace dvou lichých čísel a a b jako rozdílu dvou čtverců: $n = a^2 - b^2 = (a+b)(a-b)$. Čísla a a b jsou vyjádřit jako $a = \frac{1}{2} \cdot (p+q)$ a $b = \frac{1}{2} \cdot (q-p)$. Algoritmus 1 znázorňuje pseudokód této metody. Výsledkem běhu algoritmu jsou dvě prvočísla p a q, pro které platí, že $p = a - b$ a $q = a + b$. Uživateli je pak zobrazeno pouze jedno prvočíslo p.

Algorithm 1 Pseudokód Fermatovy faktorizace

```
a ← ceiling(sqrt(n))
while true do
    b2 ← a · a − n
    b ← sqrt(b2)
    if b · b == b2 then
        break
    else
        a ← a + 1
    end if
end while
```