Dokumentace k projektu č. 2 - Implementace a prolomení RSA

Autor: Tomáš Mlčoch (xmlcoc06@stud.fit.vutbr.cz)

Tato dokumentace popisuje implementaci metod použitých v projektu implementujícím generování RSA klíčů, šifrování, dešifrování a prolamování tohoto šifrování do předmětu KRY (Kryptografie).

Implementace RSA

V tomto ohledu byl implementován klasický RSA algoritmus:

- Generování dvou různých prvočísel p a q
- n = p * q
- phi(n) = (p - 1) * (q - 1)
- e < n takové, že gcd(e, phi(n)) = 1
- d = e^{-1} mod phi(n)

Pro testování prvočíselnosti byl implementován algoritmus Miller-Rabin, před který byl ještě přidán jeden Fermatův test, který významně zrychlil funkci testování prvočíselnosti.

Při implementaci RSA byla použita knihovna GPM, přičemž dle zadání část významných algoritmů byla implementována ručně. Jde o algoritmus pro nalezení největšího společného dělitele, algoritmus pro výpočet multiplikativní inverze, modulárního umocňování, zmíněný algoritmus Miller-Rabin a algoritmus generování pravděpodobných prvočísel.

Prolamování RSA (Faktorizace veřejného modulusu)

Pro faktorizaci veřejného modulu byla použita knihovna **msieve**. Jde o knihovnu v jazyce C, implementující množinu algoritmů pro faktorizování velkých celočíselných hodnot. Konkrétně implementuje algoritmy SIQS (Self-initializing quadratic sieve) a GNFS (General number field sieve).

Obě tyto metody jsou založeny na nalezení kongruence čtverců. Jde o kladná čísla x a y taková, že $x^2 \equiv y^2 \pmod n$, kde $x \neq \pm y$ a obě čísla jsou menší než n. Pak zřejmě platí rovnost $x^2 - y^2 = kn$ pro libovolné celé k. Dostáváme $(x + y)(x - y) = kn$. Vypočítáme-li $gcd(x - y, n)$, respektive $gcd(x + y, n)$, potom dostaneme s pravděpodobností 50% dělitele čísla n [1].

Při náhodné volbě čísela x by byl výpočet čísla y tak, aby byla splněna kongruence čtverců, neefektivní. Proto jsou hledány kongruence ve speciálním tvaru, kde se využívají prvočísla. [1]

Metoda kvadratického síta (SIQS)

Založena na kvadratickém polynomu: $Q(a) = (x+d)^2 - N$, kde d = sqrt(N) a N je faktorované číslo. Z čehož plyne $Q(a) = (x + d)^2 \pmod N$ pro libovolné celé a. Navíc dělí-li celé číslo n Q(a), potom n | Q(a + kn) pro všechna celá k. Tato vlastnost nám pak umožní efektivní prosívání. Nechť [−A, A] je interval, na kterém budeme prosívat, a B je horní hranice faktorové báze. Potom pro všechna prvočísla z báze faktorizace a všechna a z definovaného intervalu určíme, zda p dělí Q(a). Pokud ano, označíme takto a i všechny jeho k-násobky v intervalu. Jakmile skončíme s prosíváním, je nutné určit taková a, která jsou B-hladká, resp. B-hladká s výjimkou jednoho dělitele. [1]

Metoda síta v číselném tělese (GNFS)

Princip metody je následující: mějme číslo n, které chceme faktorizovat, polynom $f(x) \in Z[x]$, jeho kořen α a celé číslo m splňující $f(m) \equiv 0 \pmod n$. Snažíme se najít celá čísla a, b, pro než (a + bm) je druhou mocninou v Z a (a + bα) tvoří druhou mocninu v Z[α]. Je-li $a+bm = x^2$ a $(a+b\alpha) = y^2$, pro $x \in Z$, $y \in Z[\alpha]$ jsme schopni rozložit číslo n. s využitím zobrazení φ, platí $\varphi(y^2) = \varphi(y^2) =$

$\varphi(a+b\alpha) = (a + bm) + kn = x^2 + kn$ pro nějaké celé k. [1]

Literatura

[1] KLEINOVÁ, Hana. Algoritmus RSA a problém faktorizace v praxi [online]. 2008 [cit. 2013-04-13]. Diplomová práce. Masarykova univerzita, Fakulta informatiky. Vedoucí práce Václav Matyáš. Dostupné z: <http://is.muni.cz/th/73040/fi_m/>.