Univerzita Pavla Jozefa Šafárika v Košiciach

Prírodovedecká fakulta

Kryptovanie s verejným kľúčom

Diplomová práca

Ľubomír Krupa

2008

Prírodovedecká fakulta Univerzity Pavla Jozefa Šafárika v Košiciach
ÚMV - Ústav matematických vied

Zadanie magisterskej záverečnej práce

Meno a priezvisko študenta: Bc. Ľubomír Krupa

Študijný program: EFMm - Ekonomická a finančná matematika (Jednoodborové štúdium, magisterský II. st., denná forma)

Názov:

Kryptovanie s verejným kľúčom

Cieľ:

Oboznámiť sa s matematickými a technologickými princípmi kryptovania s verejným kľúčom. Oboznámiť sa s alternatívnymi metódami šifrovania a princípmi kvantových algoritmov. Preskúmať možnosti využitia zložitejších matematických štruktúr v roli verejného kľúča.

Odporúčaná literatúra:

1. Goldwasser S., Bellare, M.: Lectures Notes on Cryptography, http://www.cs.ucsd.edu/users/mihir/papers/gb.html
2. A. Yu. Kitaev, Classical and Quantum Computation, American Mathematical Society, Graduate Studies in Mathematics 47 (2002), ISBN 0-8218-3229-8
3. Simon Singh:Kniha kódu a šifer, Dokořán 2003, ISBN 80-86569-18-7
4. Zdroje na internete

Vedúci magisterskej záverečnej práce: doc. RNDr. Gabriel Semanišin, PhD.

Oponent magisterskej záverečnej práce:

Dátum zadania magisterskej záverečnej práce: 7.10.2004

Dátum odovzdania magisterskej záverečnej práce:

Dátum potvrdenia:

prof. RNDr. Stanislav Jendroľ, DrSc.
riaditeľ ústavu

Vyhlásenie

Vyhlasujem, že túto prácu som vypracoval samostatne, na základe vedomostí nadobudnutých štúdiom a s pomocou uvedenej literatúry.

_____

Ľubomír Krupa

Poďakovanie

Ďakujem doc. RNDr. Gabrielovi Semanišinovi, PhD. za odborné vedenie a cenné rady poskytnuté pri vypracovaní tejto práce.

Abstrakt

Táto práca sa zaoberá problematikou asymetrickej kryptografie. Popíšeme v súčasnosti používané kryptosystémy. Rozoberieme možnosti faktorizácie čísel. Načrtneme možnosti využitia ďalších matematických štruktúr v kryptosystémoch. Ukážeme využitie kvantovej technológie v kryptografii a kryptoanalýze.

Abstract

This diploma thesis deals with asymmetric cryptography problems. We describe cryptosystems used in the present. We analyze ability of number factorization. We design utilization of another mathematical structure in cryptosystems. We display using of quantum technology in cryptography and cryptanalysis.

Obsah

Zoznam obrázkov

Úvod

Cieľom tejto práce je priniesť ucelený obraz o súčasných kryptosystémoch, možnostiach ich prelomenia a o možných riešeniach týchto rizík.

V prvej kapitole uvedieme čitateľa do problematiky kryptografie, predstavíme princípy dvoch hlavných prístupov v kryptografii, a to symetrickej a asymetrickej kryptografie.

Bezpečnosť súčasných kryptosystémov je založená na zložitosti problému faktorizácie čísla na súčin prvočiniteľov. Možnosťami faktorizácie pomocou rôznych algoritmov sa budeme zaoberať v druhej kapitole. Tu tiež ukážeme prognózy faktorizácie v ďalšom období.

V tretej kapitole sa pozrieme na možnosť použitia iných matematických štruktúr ako prvočísel v kryptosystémoch. Rozoberieme hlavne použitie grúp.

Celkom odlišný prístup v kryptovaní a kryptoanalýze znamená využitie princípov kvantovej mechaniky. Štvrtá kapitola pojednáva o využití vlastností kvantovej mechaniky na bezpečný prenos kľúča a taktiež o efektívnom kvantovom faktorizačnom algoritme.

Protokol na prenos kľúča pomocou kvantových častíc ilustruje program, ktorý sme navrhli. Jeho popis ako aj popis užívateľského prostredia je zachytený v piatej kapitole.

V prílohách porovnáme kryptosystémy RSA a MST_1, teda použitie prvočísel a grúp v kryptosystémoch, a tiež klasický a kvantový prístup pri prenose šifrovacieho kľúča.

1. Úvod do problematiky

Na úvod oboznámime čitateľa so základnými pojmami kryptografie. Uvedieme základné definície potrebné na pochopenie ďalších problémov rozoberaných v tejto práci, delenie kryptosystémov, princíp bezpečnosti jednotlivých kryptosystémov.

1.1. Kryptosystém

Ako prvý definujeme kryptosystém. Následne ukážeme, ako sa delia kryptosystémy.

Definícia:

Kryptosystémom nazývame päticu $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$, kde $\mathcal{P}$ je množina správ, $\mathcal{C}$ je množina šifrovaných správ, $\mathcal{K}$ je množina kľúčov, $\mathcal{E}=\{E_k; k \in \mathcal{K}\}$ je množina funkcií $E_k : \mathcal{P} \to \mathcal{C}$, ktoré sa používajú na šifrovanie a $\mathcal{D}=\{D_k; k \in \mathcal{K}\}$ je množina funkcií $D_k : \mathcal{C} \to \mathcal{P}$, ktoré sa používajú na dešifrovanie a platí: pre každý kľúč $e \in \mathcal{K}$ existuje kľúč $d \in \mathcal{K}$ taký, že pre každé $p \in \mathcal{P} : D_k\left(E_k\left(p\right)\right) = p$.

Na základe tejto definície môžeme rozdeliť kryptosystémy na:

- Symetrický kryptosystém (kryptosystém s tajným kľúčom), ak šifrovací a dešifrovací kľúč sú rovnaké, alebo dešifrovací kľúč sa dá jednoducho vypočítať zo šifrovacieho kľúča.
- Asymetrický kryptosystém (kryptosystém s verejným kľúčom), ak šifrovací a dešifrovací kľúč sú rôzne a je výpočtovo zložité vypočítať dešifrovací kľúč zo šifrovacieho kľúča. V tomto prípade sa šifro-

vací kľúč nazýva verejný kľúč a dešifrovací kľúč sa nazýva tajný kľúč.

1.1.1. Bezpečnosť kryptosystémov

Keďže kryptosystémy slúžia na zabezpečenie komunikácie a výmeny informácií, má zmysel zaoberať sa otázkou aké parametre má spĺňať bezpečná komunikácia.

Bezpečná komunikácia má tri vlastnosti:

1. dôvernosť
2. integrita
3. nepopierateľnosť

Zabezpečenie komunikácie s týmito vlastnosťami sa dá zabezpečiť s rôznou silou, resp. odolnosťou voči prelomeniu. Preto uvedieme niekoľko typov bezpečnosti kryptosystémov.

- výpočtová bezpečnosť – hovoríme, že kryptosystém je bezpečný, ak vypočítanie kľúča je možné, ale nie v potrebnom čase
- dokázateľná bezpečnosť – hovoríme, že kryptosystém je bezpečný, ak sa dá dokázať, že problém jeho prelomenia je ekvivalentný vyriešeniu výpočtovo ťažkého problému (napr. problém diskrétneho logaritmu)
- bezpodmienečná bezpečnosť – hovoríme, že kryptosystém je bezpečný, ak nájdenie kľúča nie je možné ani pri použití nekonečnej výpočtovej sily

Existuje však aj definícia bezpečnosti kryptosystému, ktorá sa opiera o teóriu pravdepodobnosti.

Definícia (perfektná bezpečnosť):

Hovoríme, že kryptosystém $(\mathcal{P}, C, \mathcal{K}, \mathcal{E}, \mathcal{D})$ má perfektnú bezpečnosť ak platí:

$$(\forall x \in \mathcal{P})(\forall y \in C) : P(X = x \mid Y = y) = P(X = x),$$

kde X predstavuje dešifrovanú správu a Y šifrovanú správu.

To znamená, že kryptosystém je perfektne bezpečný, ak pravdepodobnosť toho, že dešifrujeme správu, ak poznáme jej zašifrovanú verziu, je rovnaká ako keď správu náhodne zvolíme. [19]

1.2. Symetrické kryptosystémy

Ako sme už spomenuli, symetrické kryptosystémy používajú na šifrovanie aj dešifrovanie ten istý kľúč. Teda každá dvojica účastníkov komunikácie musí mať svoj vlastný kľúč. To je nevýhodné pri vyššom počte účastníkov.

Symetrické kryptosystémy sa vo všeobecnosti delia na blokové a prúdové.

1.2.1. Blokové kryptosystémy

Blokový kryptosystém rozdelí šifrovanú správu na bloky rovnakej dĺžky, každý blok sa následne zašifruje. Dva dôležité triedy blokových kryptosystémov sú substitučné a transpozičné šifrátory.

1.2.1.1. Substitučné blokové kryptosystémy

Substitučný šifrátor je taký blokový kryptosystém, ktorý nahrádza symbol alebo skupinu symbolov iným symbolom alebo skupinou symbolov. Uvedieme definície dvoch substitučných šifrátorov.

Definícia (monoalfabetická substitučná šifra):

Nech $\mathcal{A}$ je konečná abeceda a $\mathcal{M}$ je množina všetkých reťazcov dĺžky $t \in \mathbb{N}$ nad $\mathcal{A}$. Nech $\mathcal{K}$ je množina všetkých permutácií množiny $\mathcal{A}$. Pre každé $e \in \mathcal{K}$ definujeme šifrovacie pravidlo E_e ako:

$$E_e(m) = \left(e(m_1)\, e(m_2) \ldots e(m_t)\right) = (c_1 c_2 \ldots c_t) = c,$$

kde $m = (m_1 m_2 \ldots m_t) \in \mathcal{M}$, a dešifrovacie pravidlo D_t ako:

$$D_d(c) = \left(d(c_1)\, d(c_2) \ldots d(c_t)\right) = (m_1 m_2 \ldots m_t) = m,$$

kde d je inverzná permutácia $d = e^{-1}$.

E_e sa nazýva monoalfabetická šifra.

Príkladom monoalfabetickej šifry je kryptosystém Cézarov posun.

Definícia (polyalfabetická substitučná šifra):

Polyalfabetická substitučná šifra je bloková šifra s blokom dĺžky $t \in \mathbb{N}$ nad konečnou abecedou $\mathcal{A}$ s nasledujúcimi vlastnosťami:

i. priestor kľúčov $\mathcal{K}$ pozostáva zo všetkých usporiadaných množín t permutácií $(p_1, p_2, \ldots, p_t)$, kde každá permutácia p_i je definovaná nad konečnou abecedou $\mathcal{A}$

ii. šifrovacie pravidlo E_e je definované pre správu $m = (m_1 m_2 \ldots m_t)$ a kľúč $e = (p_1, p_2, \ldots, p_t)$ ako:

$$E_e(m) = \left(p_1(m_1)\, p_2(m_2) \ldots p_t(m_t)\right)$$

iii. dešifrovací kľúč odpovedajúci šifrovaciemu kľúču $e = (p_1, p_2, \ldots, p_t)$ je $d = \left(p_1^{-1}, p_2^{-1}, \ldots, p_t^{-1}\right)$

Príkladom tejto šifry je Vigenèreov kryptosystém. Pre viac príkladov odkazujeme čitateľa na [4].

1.2.1.2. Transpozičné blokové kryptosystémy

Transpozičný šifrátor je taký kryptosystém, ktorý na šifrovanie využíva prepermutovanie symbolov v bloku.

Definícia (jednoduchá transpozičná šifra):

Jednoduchá transpozičná šifra je bloková šifra s blokom dĺžky $t \in \mathbb{N}$ nad konečnou abecedou $\mathcal{A}$ s nasledujúcimi vlastnosťami:

i. priestor kľúčov $\mathcal{K}$ pozostáva zo všetkých permutácií množiny $\{1, 2, \ldots, t\}$

ii. šifrovacie pravidlo E_e je definované pre správu $m = (m_1 m_2 \ldots m_t)$ a kľúč $e \in \mathcal{K}$ ako:

$$E_e(m) = \left(m_{e(1)} m_{e(2)} \ldots m_{e(t)}\right)$$

1.2.2. Prúdové kryptosystémy

U tohto typu kryptosystémov ide vlastne o blokový kryptosystém s blokom dĺžky 1. Navyše šifrovacie pravidlo sa môže meniť pre každý šifrovaný symbol.

Definícia:

Nech $\mathcal{K}$ je množina kľúčov. Postupnosť $e_1 e_2 \ldots e_n \in \mathcal{K}$ budeme nazývať prúd kľúčov[1].

Definícia (prúdový kryptosystém):

Nech $\mathcal{A}$ je konečná abeceda a nech E_e je jednoduchá substitučná šifra s blokom dĺžky 1. Pre každú správu $m_1 m_2 \ldots m_n$ a pre každý prúd kľúčov $e_1 e_2 \ldots e_n \in \mathcal{K}$ definujeme šifrovanú správu $c_1 c_2 \ldots c_n$, kde $c_i = E_{e_i}(m_i)$. Ak d_i je inverziou e_i, potom definujeme dešifrovacie pravidlo ako $D_{d_i}(c_i) = m_i$, kde $c_1 c_2 \ldots c_n$ je šifrovaná správa.

Prúd kľúčov môže byť generovaný náhodne, alebo pomocou algoritmu, ktorý ho generuje z počiatočného prúdu kľúčov, nazývaného zdroj[2]. Takýto algoritmus nazývame generátor prúdu kľúčov.

[1] angl. keystream

[2] angl. seed

1.3. Asymetrické kryptosystémy

V asymetrickej kryptografii sa používajú dva odlišné kľúče: súkromný a jemu odpovedajúci verejný. Súkromný kľúč je známy len jeho vlastníkovi. Používa sa na autentifikáciu posielanej správy a dešifrovanie prijatej správy. Verejný kľúč je všeobecne známy a slúži na šifrovanie posielanej správy a overenie autentifikácie prijatej správy.

Výhodou asymetrickej kryptografie oproti symetrickej je menší počet kľúčov potrebných na komunikáciu medzi n subjektmi. Pri asymetrickom modeli je potrebných $2n$ kľúčov, zatiaľ čo pri symetrickom modeli je to $\binom{n}{2}$ kľúčov. Nevýhodou je jej nízka rýchlosť.

Asymetrický kryptosystém musí spĺňať tri vlastnosti:

- korektnosť – dešifrovanie šifrovanej správy vedie k pôvodnej správe: $(\forall p \in \mathcal{P})(\forall k \in \mathcal{K}) : D_k\left(E_k\left(p\right)\right) = p$
- realizovateľnosť – funkcie E_k a D_k sú algoritmicky efektívne realizovateľné, tzn. s deterministickou, resp. pravdepodobnostnou polynomiálnou časovou zložitosťou
- bezpečnosť – zo znalosti funkcie E_k je prakticky nemožné určiť funkciu $D_k{}^*$ takú, že $D_k{}^*$ je efektívne realizovateľná, a pre nezanedbateľné množstvo $c \in C : D_k^*\left(c\right) = D_k\left(c\right)$

Medzi najznámejšie a najčastejšie používané asymetrické kryptosystémy patrí kryptosystém Diffie-Hellman, algoritmus RSA a ElGamalov kryptosystém.

1.3.1. Princíp asymetrickej kryptografie

Majme dvojicu subjektov Alica a Bob, ktoré medzi sebou komunikujú a subjekt Cyril, ktorý má prístup ku komunikačnému kanálu medzi Alica a Bobom. Alica aj Bob majú svoj súkromný a verejný

kľúč. Predpokladajme, že Alica chce poslať správu Bobovi. Alica teda podpíše správu svojím súkromným kľúčom (AS). Podpísanú správu zašifruje Bobovým verejným kľúčom (BV) a odošle Bobovi. Bob prijatú správu dešifruje najprv svojím súkromným kľúčom (BS), a potom Aliciným verejným kľúčom (AV).

Útočník Cyril má niekoľko možností:

1. môže odchytiť správu, ktorú posiela Alica Bobovi. Tá je však zašifrovaná Bobovým verejným kľúčom a bez znalosti Bobovho súkromného kľúča správu nedešifruje
2. môže Bobovi poslať vlastnú správu. Bez znalosti Alicinho súkromného kľúča správu nepodpíše
3. v prípade, že sú verejné kľúče distribuované po tomto informačnom kanále, môže odchytiť Bobov aj Alicin verejný kľúč a miesto neho odoslať svoj verejný kľúč. Alica aj Bob by potom šifrovali a dešifrovali pomocou Cyrilovho verejného kľúča, teda Cyril by mal úplnú kontrolu nad posielanými správami. Tomu sa dá zabrániť symetrickou kryptografiou.

1.3.2. Kryptosystém RSA

Kryptosystém RSA, pomenovaný po trojici autorov R. L. Rivest, A. Shamir, L. Adleman [17], je jeden z prvých šifrovacích systémov založených na princípe asymetrickej kryptografie. V súčasnosti patrí medzi najpoužívanejšie algoritmy využívané v bezpečnej komunikácii.

Algoritmus je vo všeobecnosti založený na vysokej výpočtovej zložitosti problému faktorizácie veľkých čísel na prvočísla. Používa sa s úpravami dodnes napr. v mobilných telefónoch, bankomatoch, elektronických podpisoch, atď.

1.3.2.1. Popis algoritmu RSA

Ukážeme ako sa generuje verejný a tajný kľúč, a ako sa pomocou nich šifruje a dešifruje v kryptosystéme RSA.

V popise algoritmu budeme používať Eulerovu funkciu, preto tento pojem najprv zadefinujeme.

Definícia: (Eulerova funkcia)

Zobrazenie $\varphi : \mathbb{N} \to \mathbb{N}$, ktoré každému $n \in \mathbb{N}$ priraďuje počet čísel z množiny $\{1, 2, ..., n\}$ nesúdeliteľných s číslom *n*, sa nazýva Eulerova funkcia.

Priamo z definície vyplýva, že $\varphi(1) = 1$, a taktiež pre každé prvočíslo *p* platí, že $\varphi(p) = p - 1$.

Nasledujúca lema hovorí ako sa dá jednoducho spočítať hodnota Eulerovej funkcie pre hodnotu $n = p \cdot q$.

Lema:

Nech $n \in \mathbb{N}$, $n = p \cdot q$, kde *p,q* sú prvočísla. Potom

$$\varphi(n) = \varphi(p \cdot q) = (p-1) \cdot (q-1)$$

Dôkaz:

Uvažujme, že $n = p \cdot q$. Všetky čísla súdeliteľné s *n* sú v tvare $k \cdot p$ alebo $l \cdot q$. Počet čísel menších ako *n* a súdeliteľných s $k \cdot p$ je práve $q - 1$. Rovnako čísel menších ako *n* a súdeliteľných s $l \cdot q$ je práve $p - 1$. Dohromady teda dostávame

$$\varphi(n) = \varphi(p \cdot q) = (p \cdot q - 1) - (q - 1) - (p - 1) =$$
$$= p \cdot q - p - q + 1 = (p-1) \cdot (q-1)$$

QED

Teraz už môžeme prejsť k samotnému popisu algoritmu RSA.

Postup vytvárania verejného a tajného kľúča [21]:

1. náhodne vygenerujeme dve veľké prvočísla, označme ich *p,q*;
   – $p, q \approx 2^{512}$.
2. spočítame číslo *n* a číslo $\varphi(n)$;
   – *n* je súčin prvočísiel *p,q*, $n \approx 2^{1024}$
   – $\varphi(n)$ je Eulerova funkcia, $\varphi(n) = (p-1)\cdot(q-1)$. Preto sa v praxi dá miesto Eulerovej funkcie použiť najmenší spoločný násobok čísel *p*-1,*q*-1.
3. zvolíme číslo *e* náhodne tak, aby platilo:
   $1 < e < \varphi(n)$, $(e, \varphi(n)) = 1$
   – $e \approx 2^{1024}$
   – dvojica (e, n) je verejný kľúč
4. dopočítame *d* tak, aby platilo:
   $1 < d < \varphi(n)$, $e \cdot d \equiv 1 \left(\varphi \left(\mathrm{mod}\, n\right)\right)$. Po úprave dostaneme $d \equiv e^{-1} \left(\varphi \left(\mathrm{mod}\, n\right)\right)$
   – na výpočet sa dá použiť Euklidov algoritmus
   – existencia čísla *d* je zaručená Bautzovou vetou
   – $d \approx 2^{1024}$
   – dvojica (d, n) je tajný kľúč

Keď je dvojica kľúčov (e, d) vygenerovaná, môže účastník komunikácie zverejniť verejný kľúč (e, n). Po tomto kroku môže začať komunikácia.

Predpokladajme, že Alica chce poslať Bobovi správu *M*. Bob vygeneroval a zverejnil svoj verejný kľúč. Alica prevedie správu do číselnej formy (každému písmenu priradí jeho číselnú hodnotu, napr. ASCII kód). Takto upravenú správu potom rozdelí na bloky m_i rovna-

kej dĺžky. Na určenie dĺžky blokov a zarovnania na rovnakú dĺžku sa v praxi používajú pravidlá formátovania. Bloky m_i Alica zašifruje pomocou Bobovho verejného kľúča nasledovne: $c_i \equiv m_i^e \pmod n$. Takto zašifrovanú správu môže Alica odoslať Bobovi.

Bob prijatú správu dešifruje pomocou svojho tajného kľúča: $m_i \equiv c_i^d \pmod n$.

1.3.2.2. Matematické pozadie

Uvedieme niektoré definície, tvrdenia a vety, o ktoré sa kryptosystém RSA opiera, alebo ktoré s týmto kryptosystémom súvisia.

Veta:(Základná veta aritmetiky)

Každé $a \in \mathbb{Z} - \{0, 1, -1\}$ je možné vyjadriť v tvare $a = \pm p_1 \cdot p_2 \cdot \ldots \cdot p_n$, kde $n \in \mathbb{N}$ a $p_1, p_2, \ldots, p_n$ sú prvočísla.

Toto vyjadrenie je jednoznačné, až na poradie činiteľov, tzn. ak $m \in \mathbb{N}$ a $q_1, q_2, \ldots, q_m$ sú také prvočísla, pre ktoré platí $a = \pm q_1 \cdot q_2 \cdot \ldots \cdot q_m$, tak znamienka oboch vyjadrení čísla a sú rovnaké, $m = n$ a ak $p_1 \le p_2 \le \ldots \le p_n$, $q_1 \le q_2 \le \ldots \le q_m$, tak $p_1 = q_1, p_2 = q_2, \ldots, p_n = p_m$.

V dôkaze správnosti algoritmu RSA budeme využívať Eulerovu vetu. Definujme pojmy úplný systém zvyškov modulo *m* a redukovaný systém zvyškov modulo *m*. Tie neskôr využijeme v dôkaze Eulerovej vety.

Definícia:

Nech $m \in \mathbb{N}, m > 1$.

Úplným systémom zvyškov modulo *m* nazývame *m*-prvkovú množinu celých čísel, ktoré sú navzájom nekongruentné modulo *m*.

Redukovaným systémom zvyškov modulo *m* nazývame $\varphi(m)$-prvkovú množinu celých čísel, ktoré sú navzájom nekongruentné modulo *m* a každé z nich je nesúdeliteľné s *m*.

Lema:

Nech $A = \{x_1, x_2, ..., x_{\varphi(m)}\}$ je redukovaný systém zvyškov modulo *m* a nech $a \in \mathbb{Z}, (a, m) = 1$. Potom $B = \{a \cdot x_1, a \cdot x_2, ..., a \cdot x_{\varphi(m)}\}$ je tiež redukovaným zvyškom modulo *m*.

Dôkaz tejto lemy je triviálny, preto ho nebudeme uvádzať a prenecháme ho čitateľovi.

Teraz už môžeme vysloviť a dokázať Eulerovu vetu.

Veta: (Eulerova)

Nech $a \in \mathbb{Z},\ m \in \mathbb{N}, m > 1,\ (a, m) = 1$. Potom

$$a^{\varphi(m)} \equiv 1 \pmod{m}.$$

Dôkaz:

Z množiny $\{1, 2, ..., m-1\}$, ktorá je úplným systémom zvyškov modulo *m*, vynechajme čísla, ktoré nie sú nesúdeliteľné s *m*. Takto vytvorená množina, označme $A = \{x_1, x_2, ..., x_{\varphi(m)}\}$, je redukovaným zvyškom modulo *m*.

Podľa predchádzajúcej lemy je aj množina $B = \{a \cdot x_1, a \cdot x_2, ..., a \cdot x_{\varphi(m)}\}$ redukovaným zvyškom modulo *m*.

Podľa vety o delení so zvyškom platí, že pre každé $b \in B$ existuje $r_b \in \{0, 1, ..., m-1\}$ a $q_b \in \mathbb{Z}$ také, že $b = m \cdot q_b + r_b$.

Potom $D(b,m) = D(m,r_b)$ a $(m,r_b) = (b,m) = 1$. Pretože $\{r_b : b \in B\} \subseteq \{1,2,...,m-1\}$ a pre ľubovoľné $b \in B$ platí $b \equiv r_b \pmod m$, je každý prvok z množiny *B* kongruentný modulo *m* s niektorým prvkom z množiny *A*, pritom rôzne prvky z *A* sú kongruentné modulo *m* s rôznymi prvkami z *B*. Preto súčin všetkých prvkov množiny *B* je kongruentný modulo *m* so súčinom všetkých prvkov množiny *A*:

$$\prod_{i=1}^{\varphi(m)} (a \cdot x_i) \equiv \left(\prod_{i=1}^{\varphi(m)} x_i \right) \pmod m$$

Úpravou dostávame

$$a^{\varphi(m)} \cdot \left(\prod_{i=1}^{\varphi(m)} x_i \right) \equiv \left(\prod_{i=1}^{\varphi(m)} x_i \right) \pmod m$$

Keďže každý prvok x_i množiny *A* je nesúdeliteľný s *m*, môžeme vydeliť oba strany súčinom $\prod_{i=1}^{\varphi(m)} x_i$. Teda dostaneme $a^{\varphi(m)} \equiv 1 \pmod m$.

QED

Špeciálnym prípadom Eulerovej vety je Malá Fermatova veta. V Malej Fermatovej vete je číslo *m* z Eulerovej vety navyše prvočíslom.

Teraz môžeme dokázať správnosť algoritmu RSA.

Veta (Správnosť algoritmu RSA):

Nech *p,q* sú prvočísla, $n = p \cdot q$. Nech *e,d* sú ľubovoľné čísla, ktoré spĺňajú podmienku $e \cdot d \equiv 1 \pmod{\varphi(n)}$. Ďalej označme *v,w* čísla také, že $0 \le v \le n-1$ a nech platí $w \equiv v^e \pmod n$. Potom platí $v \equiv w^d \pmod n$.

Dôkaz:

Umocnením vzťahu $w \equiv v^e \pmod n$ na d dostávame $w^d \equiv v^{e\cdot d} \pmod n$. Z podmienky $e \cdot d \equiv 1 \left(\mathrm{mod}\, \varphi(n)\right)$ vyplýva $e \cdot d = 1 + r \cdot \varphi(n)$. Stačí teda ukázať $v \equiv v^{1+r\cdot\varphi(n)} \pmod n$.

Uvažujme ďalej dva prípady: v je súdeliteľné s n alebo v je nesúdeliteľné s n.

V prípade, že v je nesúdeliteľné s n, tak môžeme písať $v^{1+r\cdot\varphi(n)} \equiv v \cdot v^{r\cdot\varphi(n)} \pmod n$. Takže stačí ukázať, že $v^{r\cdot\varphi(n)} \equiv 1 \pmod n$. Upravením tohto vzťahu a využitím Eulerovej vety dostávame

$$v^{r\cdot\varphi(n)} \equiv \left(v^{\varphi(n)}\right)^r \equiv 1^r \equiv 1 \pmod n$$

V prípade, že v je súdeliteľné s $n = p \cdot q$, tak v musí byť v tvare $v = a \cdot p$ alebo $v = b \cdot q$. Bez ujmy na všeobecnosti možno predpokladať, že $v = a \cdot p$. Potom je ale v nesúdeliteľné s q, lebo q je prvočíslo. Čiže môžeme písať

$$v^{r\cdot\varphi(n)} \equiv \left(v^{\varphi(n)}\right)^r \equiv \left(v^{(p-1)\cdot(q-1)}\right)^r \equiv \left(v^{(q-1)}\right)^{(p-1)\cdot r} \equiv 1 \pmod q$$

Posledná kongruencia vyplýva z Malej Fermatovej vety, lebo $v^{(q-1)} \equiv 1 \pmod q$.

Fakt $v^{r\cdot\varphi(n)} \equiv 1 \pmod q$ môžeme prepísať do tvaru $v^{r\cdot\varphi(n)} = 1 + t \cdot q$. Prenásobme tento vzťah číslom v. Dostávame $v^{1+r\cdot\varphi(n)} = v + v \cdot t \cdot q$, t.j. $v^{1+r\cdot\varphi(n)} = v + a \cdot t \cdot p \cdot q$. Ale vieme $n = p \cdot q$. Teda $v^{1+r\cdot\varphi(n)} = v + (a \cdot t) \cdot n$, čo znamená $v \equiv v^{1+r\cdot\varphi(n)} \pmod n$. [24]

QED

1.4. Hybridné šifrovanie

V praxi sa často využívajú výhody symetrického a asymetrického šifrovania súčasne. Ide o hybridné šifrovanie.

Pri hybridnom šifrovaní sa na šifrovanie správy použije symetrický kryptosystém s náhodne vygenerovaným kľúčom, ktorý sa zašifruje asymetrickým kryptosystémom s verejným kľúčom adresáta. Adresát pri prijatí šifrovanej správy najprv dešifruje súkromným kľúčom kľúč symetrického kryptosystému a ním potom dešifruje samotnú správu.

Výhodou oproti asymetrickému šifrovaniu je rýchlosť. Na druhej strane výhodou oproti čisto symetrickému šifrovaniu je menší počet kľúčov.

2. Faktorizácia čísel

Ako už bolo povedané, bezpečnosť kryptosystému RSA, ale aj iných, je založená na zložitosti problému faktorizácie veľkých čísel. Dá sa ukázať, že ak vieme faktorizovať číslo $n = p \cdot q$, potom vieme spočítať $\varphi(n) = (p-1) \cdot (q-1)$, a teda aj odvodiť tajný exponent *d* z verejného exponentu *e*.

Kryptosystém RSA sa dá prelomiť aj bez faktorizácie čísla *n*. Stačí vyriešiť problém RSAP.

Problém RSAP:

Nech $n = p \cdot q$, e je číslo také, že $(e, \varphi(n)) = 1$ a $c \in \mathbb{Z}_n$. Nájdi $m \in \mathbb{Z}_n$ také, že $m^e \equiv c \pmod{n}$.

Algoritmus na riešenie tohto problému nebol doposiaľ nájdený a neexituje dôkaz, že existuje. Preto sa budeme zaoberať možnosťami faktorizácie čísla *n*.

Definujme problém faktorizácie zložených čísel.

Definícia:

Nech je dané kladné celé číslo *n*. Nájdi prvočíselný rozklad čísla *n*, t.j. $n = p_1^{\alpha_1} \cdot p_2^{\alpha_2} \cdot \ldots \cdot p_k^{\alpha_k}$, kde p_i sú rôzne prvočísla a $\alpha_i \geq 1$ pre každé *i*.

Tento problém sa dá riešiť klasickými aj kvantovými metódami. Kvantovej problematike budeme venovať jednu z nasledujúcich kapitol. V tejto kapitole rozoberieme v súčasnosti používané klasické faktorizačné algoritmy.

Klasické faktorizačné algoritmy môžeme v podstate rozdeliť na všeobecné a špeciálne faktorizačné algoritmy.

Faktorizačné algoritmy v špeciálnom zmysle pracujú efektívnejšie, ak faktorizované číslo má istý tvar. Zložitosť týchto algoritmov je závislá od niektorých vlastností faktorov čísla *n*. Aj preto sú efektívne pri rozkladaní čísel s malými faktormi. Ale v kryptosystéme RSA sa využívajú ako moduly čísla, ktoré nemajú malé faktory. Preto väčší význam pri skúmaní bezpečnosti kryptosystémov majú faktorizačné algoritmy vo všeobecnom zmysle. Tieto algoritmy majú zložitosť závislú jedine od čísla *n*.

Medzi špeciálne algoritmy zaraďujeme Pollardovu rho metódu, Pollardovu $p-1$ metódu, Pollardovu $p+1$ metódu, metódu eliptických kriviek a špeciálne numerické sito. Všetky tieto algoritmy pracujú v čase, ktorý je exponenciálny vzhľadom na dĺžku faktoru *p*. Preto sú pomalé pre väčšinu faktorizácii.

2.1. Pollardov rho algoritmus

Pollardov rho algoritmus sa využíva na hľadanie malých faktorov celých čísel.

Uvažujme *p* prvočíselný faktor celého čísla *n*. V tomto faktorizačnom algoritme hľadáme rovnaké členy postupnosti čísel $x_1, x_2, x_3, \ldots$, definovanej nasledovne: $x_0 = 2, x_{i+1} = f\left(x_i\right) = x_i^{\,2} + 1 \bmod p$ pre $i \geq 0$. Pomocou Floydovho algoritmu na vyhľadávanie cyklov nájdeme x_m a x_{2m} také, že $x_m \equiv x_{2m} \left(\bmod p\right)$. Keďže *p* je neznáme, robíme výpočty modulo *n* a testujeme, či $\left(x_m - x_{2m}, n\right) > 1$. Ak platí aj $\left(x_m - x_{2m}, n\right) < n$, tak $\left(x_m - x_{2m}, n\right)$ je hľadaný netriviálny faktor čísla *n*.

Algoritmus:

VSTUP: zložené celé číslo *n*, ktoré nie je mocninou prvočísla

VÝSTUP: netriviálny faktor d čísla n

1. priraď $a \leftarrow 2, b \leftarrow 2$
2. pre $i = 1, 2, 3, \ldots$ vykonaj:
   2.1. vypočítaj
   $a \leftarrow a^2 + 1 \pmod n$, $b \leftarrow b^2 + 1 \pmod n$, $b \leftarrow b^2 + 1 \pmod n$
   2.2. vypočítaj $d \leftarrow (a - b, n)$
   2.3. ak $1 < d < n$, tak vráť d a skonči
   2.4. ak $d=n$, tak vráť „CHYBA" a skonči

Zložitosť tohto algoritmu je $O\left(n^{1/4}\right)$.

2.2. Pollardov p-1 algoritmus

Hľadáme prvočíselný faktor p čísla n. Potom $p-1$ je párne číslo, teda 2 je deliteľom čísla $p-1$. Táto metóda je efektívna, ak aj ostatné delitele čísla $p-1$ sú malé, napr. zhora ohraničené číslom B. Vtedy hovoríme, že číslo $p-1$ je B-hladké.

Myšlienkou algoritmu je fakt, že ak existuje číslo Q také, že $p-1$ delí Q, tak podľa Fermatovej vety p delí $a^Q - 1$. Pretože p je faktor čísla n, nájdeme ho ako $(a^Q - 1, n)$.

Definujme číslo Q. Keďže $p-1$ má všetky delitele menšie ako B, tak definujme $Q = \prod_{q \le B} q^{M(q)}$, kde $M(q) = \left\lfloor \frac{\ln(n)}{\ln(q)} \right\rfloor$[1]. $M(q)$ je najväčšie také, aby platilo $q^{M(q)} \le n$. Potom Q obsahuje všetky mocniny prvočiniteľov $p-1$. Teda $p-1$ musí deliť Q.

[1] $\lfloor c \rfloor$ označuje dolnú celú časť čísla c

Algoritmus:

VSTUP: zložené celé číslo *n*, ktoré nie je mocninou prvočísla

VÝSTUP: netriviálny faktor *d* čísla *n*

1. zvoľ hranicu *B* (10^5 alebo 10^6)
2. zvoľ náhodne celé číslo *a*, $2 \le a \le n-1$ a spočítaj $d = (a, n)$. Ak $d \ge 2$, tak vráť *d*.
3. pre každé prvočíslo *q*, $q \le B$ vykonaj:

   3.1. spočítaj $l = \left\lfloor \frac{\ln(n)}{\ln(q)} \right\rfloor$

   3.2. spočítaj $a \leftarrow a^{q^l} \pmod{n}$
4. spočítaj $d = (a-1, n)$
5. ak $d = 1$ alebo $d = n$, tak vráť „CHYBA", inak vráť *d*, a skonči

Zložitosť tejto faktorizačnej metódy je $O\left(B \cdot \ln^2(n) + \ln^3(n)\right)$.

2.3. Metóda eliptických kriviek

Nech *E* označuje nedegenerovanú eliptickú krivku. Teda

$$E = \left\{(x, y) : y^2 = x^3 + ax + b,\ a, b \in \mathbb{Z}, \text{ kde } a^3 + 27b^2 \neq 0\right\}.$$

Definujme binárnu operáciu $*$ na množine bodov $(x, y) \in E$ nasledovne:

Ak $P = (x_1, y_1)$ a $Q = (x_2, y_2)$ sú body *E*, tak $P * Q$ je bod(x_3, y_3), kde $x_3 = m^2 - x_1 - x_2$, $y_3 = m \cdot (x_1 - x_3) - y_1$ a *m* je gradient priamky *PQ*, takže

ak $P \neq Q$ a $x_1 \neq x_2$, tak $m = \frac{y_2 - y_1}{x_2 - x_1}$,

ak P=Q, tak $m = \frac{3x_1^2 + a}{2y_1}$,

ak $P \neq Q$ a $x_1 = x_2$, tak $P * Q$ je bod v nekonečne na osi y (označme ho O).

Dá sa ukázať, že množina E, vrátane bodu O a operácia $*$ tvoria grupu. Potom bod O je neutrálny prvok grupy a pre každý konečný bod $A = (x_1, y_1) \in E$ existuje opačný $-A = (x_1, -y_1)$.

Teraz môžeme prejsť k samotnej metóde na faktorizáciu čísla n. Vezmime ľubovoľnú nedegenerovanú eliptickú krivku E a celočíselný bod $P_1 \in E$. Spočítajme členy postupnosti $A(k)$:

$$P_2 = P_1 * P_1 = P_1^2, \quad \mathrm{P}_3 = P_2^3 = P_1^{2\cdot 3}, \quad \mathrm{P}_3 = P_4^5, \ldots$$

Tento proces skončí, ak nevieme spočítať P_k, resp. ak nie je možné spočítať gradient $m \pmod n$ na výpočet $P * Q$. To znamená, že $P_k = O$. Tento prípad však nastane vtedy, ak pre nejaké y_i neexistuje inverzný prvok $\pmod n$. Vtedy ale platí, že $(y_i, n) > 1$. Faktor čísla n potom dostaneme ako (y_i, n).

Keď algoritmus zlyhá, t.j. nájdeným faktorom je samotné n, alebo ak proces trvá príliš dlho, tak začneme znovu s iným bodom krivky E. Môžeme tiež použiť novú eliptickú krivku.

Asymptotická časová zložitosť tejto metódy faktorizácie je $O\left(e^{\sqrt{2\cdot(\ln p)\cdot(\ln\ln p)}}\right)$. Často sa používa na faktorizáciu náhodných čísel, avšak nie je dostatočne rýchla pri faktorizácii modulov používaných v kryptosystéme RSA. [7]

2.4. Numerické sito

V súčasnosti najlepším algoritmom na faktorizáciu je numerické sito (NFS[1]), ktorého časová zložitosť je $O\left(e^{1,9\cdot(\ln n)^{1/3}\cdot(\ln\ln n)^{2/3}}\right)$.

Existujú dve verzie NFS, a to špeciálna (SNFS) a všeobecná (GNFS). Špeciálne numerické sito má menšiu časovú náročnosť, avšak úspešne faktorizuje len čísla v špeciálnom tvare. Všeobecné numerické sito je flexibilnejšie, čo sa týka vstupu, ale jeho časová náročnosť je vyššia ako u SNFS. Aj napriek tomu sa v praxi viac využíva GNFS. Preto sa v tejto práci budeme zaoberať touto všeobecnou metódou numerického sita [5].

GNFS pozostáva zo štyroch častí. Hlavným cieľom metódy je nájsť x a y také, že $x^2 \equiv y^2 \pmod n$. Na to využíva kombinácie čísel a polynómov. Označme preto $\mathbb{Z}[\alpha]$ okruh polynómov s celočíselnými koeficientmi do stupňa $d-1$ vrátane. Čiže ide o polynómy tvaru $a_{d-1}\alpha^{d-1} + a_{d-2}\alpha^{d-2} + \ldots + a_0$.

2.4.1. Fáza prepojenia čísel polynómov

Vyberieme ireducibilný polynóm $f(\alpha)$ stupňa $d > 1$ a pomocou neho vytvoríme okruh $\mathbb{Z}[\alpha]/f(\alpha)$, teda množinu polynómov nad $\mathbb{Z}[\alpha]$ modulo $f(\alpha)$. Polynóm $f(\alpha)$ je nulovým prvkom.

Ďalej uvažujme zvyškovú triedu $\mathbb{Z}_n$, kde n je číslo, ktoré chceme faktorizovať, a číslo m také, že platí $f(m) \equiv 0 \pmod n$. Môžeme definovať homomorfizmus medzi okruhmi $\mathbb{Z}[\alpha]/f(\alpha)$ a $\mathbb{Z}_n$: $\Phi : \mathbb{Z}[\alpha]/f(\alpha) \to \mathbb{Z}_n$, kde $\Phi(q(\alpha)) \equiv q(m) \pmod n$. V ďalšom bude-

[1] Number Field Sieve

me využívať vlastnosť homomorfizmu, že $\Phi(P * Q) = \Phi(P) * \Phi(Q)$, kde *P,Q* sú polynómy nad $\mathbb{Z}[\alpha]/f(\alpha)$.

2.4.2. Fáza presievania

V tejto fáze hľadáme množinu dvojíc (a_i, b_i), kde a_i, b_i sú nesúdeliteľné, v obdĺžniku $\langle -A, A\rangle \times \langle 0, B\rangle$. Navyše požadujeme, aby $(a_i + b_i m)$ a $(a_i + b_i \alpha)$ boli hladké a (a_i, b_i) čo najmenšie. Množinu takýchto dvojíc budeme označovať $\mathbb{D}$.

Keďže čísla $(a_i + b_i m)$ sú hladké, dajú sa zapísať v tvare $\prod_{p_j \in F} p_j^{e_{ij}}$, kde *F* je zvolená množina prvočísel p_j $(p_1 < p_2 < p_3 < \ldots)$. Teda každé z čísel $(a_i + b_i m)$ je vlastne súčin mocnín prvočísel z *F*.

2.4.3. Spracovanie matice

Ide o fázu, v ktorej hľadáme podmnožinu $\mathbb{S} \subseteq \mathbb{D}$. Množina $\mathbb{S}$ je taká množina prvkov z $\mathbb{D}$, pre ktoré platí:

i. Súčin čísel $\prod_{(a,b)\in\mathbb{S}} (a + bm)$ je štvorec v $\mathbb{Z}$

ii. Súčin polynómov $\prod_{(a,b)\in\mathbb{S}} (a + b\alpha)$ je štvorec v $\mathbb{Z}[\alpha]$

Každý činiteľ $(a_i + b_i m)$ vieme zapísať v tvare $\prod_{p_j \in F} p_j^{e_{ij}}$. Potom platí, že $\prod_{(a,b)\in\mathbb{S}} (a + bm) = \prod_{(a,b)\in\mathbb{S}} \prod_{p_j \in F} p_j^{e_j}$. Je to teda znovu súčin mocnín prvočísel z *F*. Ak chceme zabezpečiť, aby takýto súčin bol štvorcom v $\mathbb{Z}$, musíme vyberať čísla $(a_i + b_i m)$ tak, aby každý exponent e_{ij} bol párny. Podobne postupujeme aj pri polynómoch.

Metódu, ktorou sa hľadá množina $\mathbb{S} \subseteq \mathbb{D}$, popíšeme pre čísla. Z každého čísla $(a_i + b_i m) = \prod_{p_j \in F} p_j^{e_{ij}}$ vytvoríme z exponentov e_{ij} prvo-

čísel p_j jeden riadok matice, t.j. $(e_{i1}, e_{i2}, e_{i3}, \ldots)$. V skutočnosti nám postačí do matice zapisovať $e_{ij} \pmod n$, lebo nás zaujíma len parita exponentu. Vznikne binárna matica, v ktorej hľadáme takú podmnožinu riadkov, aby súčet prvkov v každom stĺpci bol párny (teda rovný $0 \pmod 2$). Množina $\mathbb{S}$ potom bude obsahovať čísla zodpovedajúce vybraným riadkom matice.

Tento postup musí byť koordinovaný s výberom polynómov, pretože podmienky i. a ii. musia platiť súčasne.

2.4.4. Výpočet faktorov

Vieme, že existuje číslo $x \in \mathbb{Z}$, ktoré je odmocninou súčinu v i. Podobne existuje polynóm $\beta \in \mathbb{Z}[\alpha]$, ktorý je odmocninou súčinu v ii. Môžeme uvažovať $x := x \pmod n$, aby $x \in \mathbb{Z}_n$, a $\beta := \beta \pmod{f(\alpha)}$, aby $\beta \in \mathbb{Z}[\alpha]/f(\alpha)$. Keďže homomorfizmus Φ prevádza každý člen prvého súčinu na člen druhého súčinu, prevádza tiež súčin z podmienky i. na súčin z podmienky ii. Teda platí rovnosť $\Phi(\beta^2) \equiv x^2 \pmod n$.

Teraz označme *y* obraz polynómu $\beta \in \mathbb{Z}[\alpha]/f(\alpha)$ v $\mathbb{Z}_n$, teda $\Phi(\beta) = y$. Z vlastnosti homomorfizmu vyplýva $\Phi(\beta^2) = \Phi(\beta * \beta) = \Phi(\beta) * \Phi(\beta) = y^2 \pmod n$. Použitím tohto vzťahu a predchádzajúcej rovnosti dostávame $y^2 \equiv x^2 \pmod n$, čo je hľadaný výraz. Faktor čísla *n* sa potom vypočíta ako $(x - y, n)$.

2.4.5. Faktorizačná sila metódy NFS

Spoločnosť RSASI odhadla možnosti faktorizácie metódou GNFS za predpokladu využitia všetkých súčasných možnosti tejto metódy a faktu, že riešenie malo byť známe do jedného roka. Zanedbané boli

časové a pamäťové nároky fázy spracovania matice. Výsledok znázorňuje tabuľka 1.

V súčasnosti je známe vylepšenie metódy NFS. Autorom tohto vylepšenia je prof. Bernstein. Nejedná sa o nový objav, ale o optimalizáciu doteraz známych techník. Bernsteinovo vylepšenie NFS však nepredstavuje výraznejšiu hrozbu pre kryptosystémy ako napr. RSA, lebo cieľom Bernsteinových vzorcov nie je popísať funkčné hodnoty, ale asymptotické správanie sa danej funkcie.

„Označme $f(n)$ dĺžku modulu, ktorý Bernsteinove zlepšenia metódy NSF zvládajú faktorizovať v rovnakom čase, ako predchádzajúce stroje zvládali faktorizovať *n*-bitové moduly metódou NFS bez Bernsteinových zlepšení. Všetko, čo vieme je, že $f(n)$ pre *n* blízke nekonečnu môže byť v ideálnom prípade $3 \cdot n$. Avšak nevieme nič o $f(n)$ pre malé čísla, napr. 512, nevieme, či $f(512)$ nie je dokonca väčšie ako 512.“ [9]

dĺžka čísla, ktoré má byť faktorizované (v bitoch)	**potrebný počet PC Pentium 500 MHz**	**požadovaná veľkosť RAM pamäte**
430	1	minimálna
760	215 000	4 GB
1 020	342 000 000	170 GB
1 620	1 600 000 000 000 000	120 TB

Tabuľka 1. Prognóza možností faktorizácie metódou GNFS [8]

2.5. Prognózy

Na základe doteraz faktorizovaných čísel sa dá závislosť veľkosti faktorizovaného čísla a roku faktorizácie aproximovať rôznymi metó-

dami. Pomocou týchto aproximácií potom môžeme vysloviť prognózu roku pre faktorizáciu konkrétneho čísla.

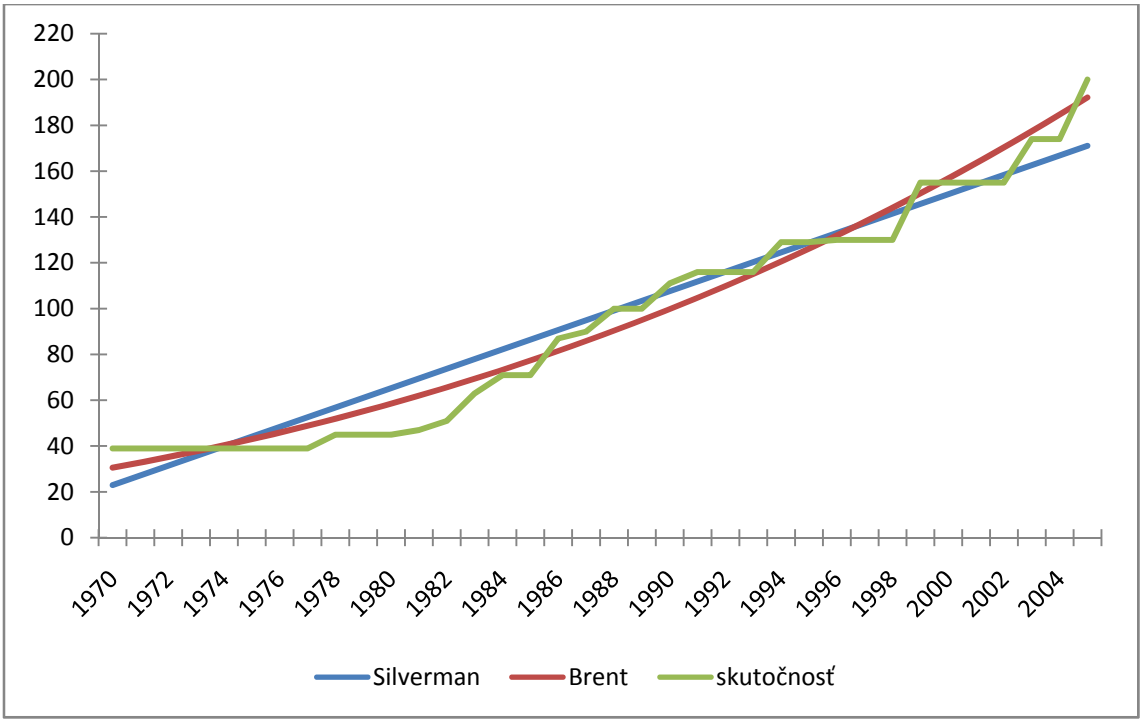
Dve najznámejšie aproximácie sú Silvermanova a Brentova. Silvermanova aproximácia vyjadruje vzťah medzi počtom cifier faktorizovaného čísla *d* a rokom faktorizácie *t* ako

$$d = 4,23 \cdot (t - 1970) + 23 .$$

Brentova aproximácia vyjadruje tento vzťah nasledovne:

$$d = \left(\frac{t - 1928,6}{13,24} \right)^3$$

Ako je vidieť v grafe (obrázok 1), Brentov vzťah lepšie aproximuje skutočnosť.



Obrázok 1. Závislosť veľkosti čísla a roku [8]

Tabuľka 2 ukazuje prognózu faktorizácie súťažných čísel súťaže The RSA Factoring challenge. V poslednom stĺpci je uvedená odmena firmy RSA Laboratories za faktorizovanie príslušného čísla.

Číslo	Počet bitov	Počet cifier	Odhad roku faktorizácie[1]	Odmena (v $)
RSA-640	640	193	2005	20 000
RSA-704	704	212	2007	30 000
RSA-768	768	232	2009	50 000
RSA-896	896	270	2014	75 000
RSA-1024	1024	309	2018	100 000
RSA-1536	1536	463	2031	150 000
RSA-2048	2048	617	2041	200 000

Tabuľka 2. Súťažné čísla súťaže The RSA Factoring challenge, sponzorovanej firmou RSA Laboratories [8]

[1] Odhad Brentovou metódou

3. Využitie iných matematických štruktúr v úlohe verejného kľúča

V predchádzajúcej kapitole sme rozoberali možnosti prelomenia kryptosystémov pomocou faktorizácie. Išlo o kryptosystémy, ktorých súkromný kľúč pozostával z dvoch veľkých prvočísiel a verejný kľúč potom tvoril ich súčin.

V tejto časti sa budeme zaoberať takými matematickými štruktúrami, ako sú grupy, príp. grafy a ich vlastnosťami. Budeme tiež skúmať možnosti aplikácie týchto štruktúr v kryptosystémoch v pozícii dvojice súkromného a verejného kľúča.

Porovnanie kryptosystému RSA a kryptosystému navrhnutého v tejto kapitole uvádzame v prílohe A.

3.1. Grupy a ich využitie pri kryptovaní

Využitím grúp pri konštrukcii kryptosystému sa zaoberal S. Magliveras v práci [12].

3.1.1. Základné pojmy teórie grúp

V úvode popíšeme grupy a ukážeme niektoré ich vlastnosti. Najprv definujme pojem grupa.

Definícia:

Dvojicu $(G,\bullet)$, kde G je množina a $\bullet$ je binárna operácia, budeme nazývať grupa, ak sú splnené tieto vlastnosti:

i. Ak $x, y \in G$, potom aj $x \bullet y \in G$

ii. Ak $x, y, z \in G$, potom $x \bullet (y \bullet z) = (x \bullet y) \bullet z$ (asociatívnosť)

iii. $(\exists e \in G)(\forall x \in G) : e \bullet x = x \bullet e = x$ (neutrálny prvok)

iv. $(\forall x \in G)(\exists u \in G) : u \bullet x = x \bullet u = e$ (inverzný prvok)

Definícia:

Grupu $(G, \bullet)$ budeme nazývať ábelovská (resp. komutatívna), ak platí:

$$\forall x, y \in G : x \bullet y = y \bullet x$$

Definícia:

Grupa $(G, \bullet)$ je konečná, ak je konečný počet jej prvkov. Počet prvkov konečnej grupy budeme nazývať rádom grupy.

Definícia:

Grupu $(G, \bullet)$ nazývame cyklická, ak platí:

$$(\exists \alpha \in G)(\forall b \in G)(\exists i \in \mathbb{N}) : b = \alpha^i .$$

Prvok α nazývame generátor grupy. [15]

Je dobré si uvedomiť, že každá konečná grupa môže byť interpretovaná ako podgrupa symetrickej grupy S_n, a teda ju môžeme považovať za grupu permutácií rádu *n*. [2]

V ďalšom budeme označovať $G^{[\mathbb{Z}]}$ množinu všetkých konečných postupností v *G*. Prvky tejto množiny môžeme chápať ako jednoriadkové matice, ktorých prvky sú prvkami z *G*.

Pod tenzorovým súčinom matíc budeme rozumieť operáciu definovanú nasledovne:

$$[x_1, \ldots, x_m] \otimes [y_1, \ldots, y_n] = [x_1 y_1, \ldots, x_1 y_n, \ldots, x_m y_1, \ldots, x_m y_n]$$

Ak $X = [x_1, \ldots, x_r] \in G^{[\mathbb{Z}]}$, potom veľkosť *r* matice *X* budeme označovať $|X|$ a prvok $\sum_{i=1}^{r} x_i$ budeme označovať $\bar{X}$.

Dá sa ukázať, že platí:

$$\forall X, Y \in G^{[\mathbb{Z}]} : \overline{XY} = \bar{X}\bar{Y} \text{ a } |XY| = |X||Y|$$

Teraz uvažujme postupnosť $\alpha = [A_1, A_2, \ldots, A_s]$, kde $A_i \in G^{[\mathbb{Z}]}$ pre konečnú grupu G, takú, že $\sum_{i=1}^{s} |A_i|$ je polynomiálne ohraničená vzhľadom na rád grupy. Ďalej nech $\bar{A}_1 \cdot \bar{A}_2 \cdots \bar{A}_s = \sum_{g \in G} a_g g, a_g \in \mathbb{Z}$. Potom hovoríme, že α je:

i. pokrytie G, ak $\forall g \in G : a_g > 0$

ii. pseudologaritmický popis G, ak $\prod_{i=1}^{s} |A_i| = |G|$

iii. logaritmický popis G, ak $\forall g \in G : a_g = 1$

iv. $[s, r]$-sieť grupy G, ak

   a. $\alpha = [A_1, \ldots, A_s]$ je pokrytie, kde $|A_i| = r$ pre $1 \le i \le s$

   b. rozdelenie pravdepodobnosti $\left\{ \frac{a_g}{r^s}, g \in G \right\}$ je približne rovnomerné

3.1.2. Logaritmický popis grupy

V nasledujúcej časti sa budeme bližšie zaoberať logaritmickým popisom grupy G.

Je vidieť, že $\alpha = [A_1, \ldots, A_s]$ je logaritmický popis G, ak každý prvok $y \in G$ sa dá jednoznačne zapísať v tvare

$$y = q_1 \cdot q_2 \cdots q_{s-1} \cdot q_s,$$

kde $q_i \in A_i$. Množinu všetkých logaritmických popisov grupy G označíme ako $\Lambda(G)$.

Typický postup konštrukcie logaritmického popisu konečnej grupy permutácií G je takýto: označme e neutrálny prvok v grupe a uvažujme postupnosť podgrúp $G = G_0 > G_1 > \cdots > G_s = e$. Teraz vezmime $\alpha = [A_1, \ldots, A_s]$ také, že každé A_i je ľavý rozklad grupy G_{i-1} podľa podgrupy G_i.

Definícia:

Nech $\alpha = [A_1, \ldots, A_s]$ je logaritmický popis G a nech $|A_i| = r_i$. Potom A_i budeme nazývať bloky popisu α. Vektor veľkostí blokov $(r_1, \ldots, r_s)$ budeme nazývať typ α. Dĺžkou popisu α budeme nazývať číslo $L = \sum_{i=1}^{s} r_i$.

Logaritmický popis $\alpha = [A_1, \ldots, A_s]$ je netriviálny, ak $s \geq 2, r_i \geq 2$ pre $1 \leq i \leq s$. V opačnom prípade je α triviálny.

Nech G je konečná grupa permutácií, $\alpha = [A_1, \ldots, A_s]$ je logaritmický podpis a $(r_1, \ldots, r_s)$ je typ α. Môžeme skonštruovať zobrazenia λ, Θ_α a $\breve{\alpha}$.

$$\lambda : \mathbb{Z}_{r_1} \times \cdots \times \mathbb{Z}_{r_s} \to \mathbb{Z}_{|G|}$$
$$(n_1, \ldots, n_s) \mapsto \sum_{i=1}^{s} \left(n_i \cdot \prod_{j=1}^{i-1} r_j \right),$$
$$\Theta_\alpha : \mathbb{Z}_{r_1} \times \cdots \times \mathbb{Z}_{r_s} \to G$$
$$(n_1, \ldots, n_s) \mapsto \alpha_{1n_1} \cdots \alpha_{sn_s}$$

Teraz už vieme definovať zobrazenie $\breve{\alpha}$ ako zloženie zobrazení Θ_α a λ^{-1}:

$$\breve{\alpha} : \mathbb{Z}_{|G|} \to G$$
$$n \mapsto \left(\Theta_\alpha \lambda^{-1}\right)(n) = \Theta_\alpha\left(\lambda^{-1}(n)\right)$$

Dá sa ukázať, že toto zobrazenie je bijektívne. Bezpečnosť kryptosystému MST_1 spočíva v predpoklade, že nájsť inverzné zobrazenie k $\breve{\alpha}$ je pre protivníka nereálne.

Týmto je motivovaná nasledujúca definícia.

Definícia:

Nech *G* je konečná grupa permutácií rádu *n*. Potom hovoríme, že dva logaritmické popisy grupy *G* α, β sú ekvivalentné, ak $\breve{\alpha} = \breve{\beta}$.

Logaritmický popis grupy *G* α budeme nazývať

i. Polynomiálny[1], ak $\breve{\alpha}^{-1}$ sa dá vypočítať v polynomiálnom čase vzhľadom k *n*

ii. Superpolynomiálny[2], ak $\breve{\alpha}^{-1}$ sa dá vypočítať v čase $O\left(n^2\right)$

iii. Nepolynomiálny[3], ak α nie je polynomiálny [2].

Logaritmický popis, ktorý je odvodený z postupnosti podgrúp spôsobom uvedeným vyššie nazývame presný *l*-transverzálny[4]. Logaritmický popis skonštruovaný analogicky, s použitím pravého rozkladu grupy, potom budeme nazývať presný *p*-transverzálny. Taktiež môžeme definovať presný miešane-transverzálny logaritmický popis, ak každý blok A_i je ľavý alebo pravý rozklad grupy G_{i-1} podľa podgrupy G_i.

[1] Angl. tame

[2] Angl. supertame

[3] angl. wild

[4] angl. exact l-transversal

Množinu všetkých presných *l*-, *p*- a miešane-transverzálnych logaritmických popisov nazveme množina presných transverzálnych logaritmických popisov grupy *G* a budeme ju označovať $\mathcal{E}(G)$.

Podobne môžeme rozdeliť aj logaritmické popisy, ktoré nevznikajú vyššie uvedeným postupom.

Definícia:

Logaritmický popis α konečnej permutačnej grupy *G* nazývame:

- transverzálny, ak je ekvivalentný s logaritmickým popisom toho istého typu z $\mathcal{E}(G)$
- netransverzálny, ak nie je transverzálny
- permutačne transverzálny, ak prepermutovaním blokov dostaneme transverzálny logaritmický popis
- totálne netransverzálny, ak žiadny jeho blok nie je rozklad netriviálnej podgrupy grupy *G*

Potom množinu všetkých transverzálnych, netransverzálnych, permutačne transverzálnych, resp. totálne netransverzálnych logaritmických popisov konečnej permutačnej grupy *G* budeme označovať $\mathcal{T}(G)$, $\mathcal{NT}(G)$, $\mathcal{PT}(G)$, *resp.* $\mathcal{TNT}(G)$.

Uvedieme tvrdenie, ktoré hovorí o bezpečnosti kryptosystému MST_1.

Tvrdenie:

Nech *G* je konečná grupa, α nepolynomiálny logaritmický popis a β polynomiálny logaritmický popis. Potom zobrazenie $\breve{\alpha} \cdot \breve{\beta}^{-1} : \mathbb{Z}_{|G|} \to \mathbb{Z}_{|G|}$ je jednocestná permutácia.

Dôkaz tohto tvrdenia nájde čitateľ v [11].

3.1.3. Konštrukcia potenciálneho kryptosystému MST_1

V nasledujúcom budeme predpokladať, že *G* je známa konečná grupa permutácií a η je nejaký fixný superpolynomiálny logaritmický popis grupy *G*, taktiež známy. Pre ľubovoľný logaritmický popis $\alpha \in \Lambda(G)$ zadefinujeme permutáciu $\hat{\alpha} := \breve{\eta}^{-1}\breve{\alpha} \in S_{|G|}$.

Teraz za verejný kľúč položíme $(\alpha, \beta) \in \Lambda(G) \times \Lambda(G)$, kde α je nepolynomiálny logaritmický popis a β je polynomiálny. Súkromný kľúč tvorí postupnosť $[\theta_1, \ldots, \theta_k] \in \mathcal{T}(G)^k$, pre ktorú platí $\hat{\beta}^{-1}\hat{\alpha} = \hat{\theta}_1 \cdots \hat{\theta}_k$. Komunikácia potom prebieha nasledovne:

Nech $m \in \mathbb{Z}_{|G|}$ je správa, ktorú chce Alica poslať Bobovi. Alica zašifruje správu ako $c = \hat{\beta}^{-1}\hat{\alpha}(m) \in \mathbb{Z}_m$. Bob pomocou súkromného kľúča získa pôvodnú správu ako $m = \hat{\alpha}^{-1}\hat{\beta}(c) = \hat{\theta}_k^{-1} \cdots \hat{\theta}_1^{-1}(c)$.

Algoritmus na generovanie dvojice súkromného a verejného kľúča je zatiaľ neznámy.

3.1.3.1. Bezpečnosť kryptosystému MST_1

Možnosti útočníka na prelomenie kryptosystému sú nájdenie inverzie $\hat{\alpha}$ alebo faktorizácia $\hat{\beta}^{-1}\hat{\alpha}$ na súčin $\hat{\theta}_1 \cdots \hat{\theta}_k$. Nájdenie inverzie $\hat{\alpha}$ znamená nájdenie inverzie $\hat{\alpha} := \breve{\eta}^{-1}\breve{\alpha}$, kde η je superpolynomiálny logaritmický popis grupy *G* (a teda aj polynomiálny) a α je nepolynomiálny popis tejto grupy. Podľa predchádzajúceho tvrdenia je ale $\breve{\eta}^{-1}\breve{\alpha}$ jednocestná permutácia.

Problém faktorizácie $\hat{\beta}^{-1}\hat{\alpha}$ na súčin $\hat{\theta}_1 \cdots \hat{\theta}_k$ je vo všeobecnosti ťažký a platí, že keby sme mali efektívny algoritmus na faktorizáciu $\hat{\beta}^{-1}\hat{\alpha}$, tak by sme boli schopní riešiť problém diskrétneho logaritmu.

3.2. Využitie ďalších matematických štruktúr

Ako sme už spomenuli v kapitole 1.3.2, úspech kryptosystému RSA spočíva v jednoduchosti spočítania $p \cdot q$, kde *p, q* sú prvočísla, a vo veľkej zložitosti opačného procesu, teda rozkladu čísla *n* na činitele *p, q*. Jednoznačnosť tohto rozkladu zabezpečuje Základná veta aritmetiky.

Istou analógiou Základnej vety aritmetiky je Veta o jednoznačnej faktorizácii pre dedičné vlastnosti grafov, navrhnutá P. Mihókom a G. Semanišinom v [14]. V tomto prípade by v úlohe prvočísel vystupovali grafy.

Výhodou takéhoto modelu by bola odolnosť voči potenciálnej hrozbe prelomenia RSA zvyšujúcou sa výpočtovou silou počítačov, prípadne kvantovou technológiou.

4. Kvantová mechanika v kryptovaní

Ak chceme predísť prelomeniu kryptosystému kvantovou technológiou, musíme sa orientovať na problémy, ktorých zložitosť nie je ohrozená potenciálom kvantových počítačov, ako napríklad u faktorizácie. Lepšie ako hľadať matematický problém, ktorý je neriešiteľný pri dnešných technologických možnostiach, je preto nájsť dej, o ktorom vieme, že nemôže nastať. V tejto kapitole sa teda budeme zaoberať takýmito kvantovými dejmi.

Systémy založené na princípoch kvantovej mechaniky využívame ako v kryptosystémoch, tak aj v kryptoanalýze.

4.1. Základné pojmy

V úvode objasníme základné pojmy a princípy kvantovej mechaniky, potrebné v ďalších častiach tejto práce.

Qubit je kvantový bit, bit realizovaný na kvantovej úrovni. Jeho stav budeme označovať $|\psi\rangle$ a je to prvok dvojrozmerného Hilbertovho priestoru $\mathbb{H}_2$. Prvky bázy tohto priestoru budeme označovať $|0\rangle$ a $|1\rangle$, vyjadrujú vlastné stavy qubitu. Pomocou prvkov bázy vieme zapísať stav qubitu ako $|\psi\rangle = \omega_0 |0\rangle + \omega_1 |1\rangle$, kde $\omega_0, \omega_1 \in \mathbb{C}$. Toto skladanie stavov budeme nazývať superpozícia vlastných stavov. Superpozícia stavov však nie je pozorovateľná. Pri akomkoľvek zásahu do qubitu tento skolabuje do jedného z vlastných stavov.

Ak skalárne koeficienty ω_0, ω_1 normujeme, t.j. $|\omega_0|^2 + |\omega_1|^2 = 1$, tak hodnoty $|\omega_i|^2$ zodpovedajú pravdepodobnosti, že qubit skolabuje pri meraní do vlastného stavu *i*.

N-qubitovým kvantovým registrom budeme nazývať usporiadanú *n*-ticu qubitov. Jeho stav môžeme vyjadriť ako superpozíciu vlastných stavov $|\psi\rangle = \sum_{x=0}^{2^{n-1}} \omega_x |x\rangle$, kde $\forall x : |x\rangle$ sú vlastné stavy a $\omega_x \in \mathbb{C}$. Kvantový register v superpozícii môžeme chápať ako exponenciálne paralelizovanú verziu pamäťového registra, ktorý môže dosiahnuť 2^n hodnôt súčasne [22]. Ak na kvantový register aplikujeme kvantovú operáciu, tak ovplyvňujeme 2^n stavov zároveň. Túto vlastnosť nazývame kvantový paralelizmus a rozhoduje o efektivite kvantového počítača.

Qubity v kvantovom registri sa nachádzajú v entaglovaných, čiže prepletených stavoch. To znamená, že jednotlivé qubity sú vzájomne prepojené, a teda meraním jedného qubitu vieme zistiť aj hodnoty ostatných qubitov. Meraním sa však toto prepletenie rozpadne.

Zariadenia, ktoré vykonávajú operácie s qubitmi, budeme nazývať kvantové brány. Každá kvantová brána je popísaná unitárnou štvorcovou maticou. Aplikovať bránu na stav $|\psi\rangle$ znamená vynásobiť stĺpcový vektor stavu s maticou popisujúcou daný operátor, prislúchajúci konkrétnej bráne. Zapisujeme to $|\psi_{i+1}\rangle = U|\psi_i\rangle$, kde *U* je unitárna matica.

4.2. Kvantová mechanika v kryptosystémoch

Jediným perfektne bezpečným kryposystémom je Vernamova šifra. Keďže ide o symetrický kryptosystém, jediným jej nedostatkom je bezpečná distribúcia kľúča. Tento nedostatok je však taký veľký, že sa Vernamova šifra prakticky nepoužíva.

Keďže ide o problém, ktorý sa nedá dostatočne bezpečne vyriešiť klasickým matematickým prístupom, budeme sa zaoberať riešením pomocou kvantovej mechaniky.

Porovnanie klasického a kvantového prístupu pri prenose kľúča uvádzame v prílohe B.

4.2.1. Protokol BB84

Tento protokol bol navrhnutý v roku 1984 Charlesom Bennettom a Gillesom Brassardom v práci [1]. Na kódovanie kvantových stavov používa polarizované fotóny.

Uvažujme prípad, keď Alica chce poslať Bobovi kryptografický kľúč, pričom odchytiť ho môže Eva. Alica a Bob budú používať dve bázy polarizácie fotónov. Jednu bázu tvorí vertikálna a horizontálna polarizácia (ozn. +) a druhú diagonálna a antidiagonálna polarizácia (ozn. ×). Polarizácie jednej bázy sú teda na seba ortogonálne.

Ďalej sa Alica a Bob dohodnú, že polarizácia fotónu pod uhlom 0° a 135° bude reprezentovať bit s hodnotou 0 a polarizácia fotónu pod uhlom 90° a 45° bude reprezentovať bit s hodnotou 1.

	0	1
+	↔	↕
×	⤡	⤢

Tabuľka 3. Kódovanie bitov v jednotlivých bázach [16]

Prenos kľúča znázorňuje tabuľka 4. Alica vygeneruje kľúč ako náhodnú sekvenciu bitov (1. riadok). Taktiež náhodne vygeneruje rovnako dlhú sekvenciu polarizačných báz (2. riadok), pomocou ktorých bude jednotlivé bity kódovať (3. riadok). Takto polarizovaný fotón pošle cez kvantový kanál Bobovi. Bob náhodne, nezávisle na Alici, generuje svoju sekvenciu polarizačných báz (4. riadok). Pomocou týchto báz dekóduje prijaté polarizované fotóny (5. a 6. riadok). Ak Bob zvolí rovnakú bázu ako Alica, tak určí výsledok správne, ak sa však nezhodnú vo voľbe báz, výsledky budú neurčité. Ak Alica pošle Bobovi postačujúce množstvo fotónov, tak Bob zverejní svoju sek-

venciu báz (7. riadok) a Alica následne potvrdí, v ktorých bázach sa zhodli(8. riadok). Iba bity na týchto miestach nameral Bob s určitosťou správne, teda ďalej budú pracovať iba s týmito bitmi (9. riadok). Pre odhalenie útočníka zverejní Bob niekoľko bitov (10. riadok), ktorých správnosť Alica overí (11. riadok). Ak sa zhodujú, tak s pravdepodobnosťou $1-\left(\frac{3}{4}\right)^n$, kde *n* je počet overovaných bitov, nebola ich komunikácia odpočúvaná. Vo výslednom kľúči sa porovnávané bity nepoužijú (12. riadok).

V celom procese môže dôjsť k stratám fotónov rôznymi technickými vplyvmi. Preto v prípade, že Bob nezachytí žiaden fotón, nechá na mieste bitu prázdne miesto.

Aké možnosti ma útočník Eva? Keďže fotóny ako kvantové častice sa nedajú klonovať, jedinou možnosťou Evy, ako získať informácie, je zostrojenie podobných zariadení, aké používajú Alica a Bob.

Eva teda bude odchytávať fotóny odosielané Alicou. Keďže nevie, akú voľbu báz používa Alica, musí svoje bázy voliť náhodne, a teda s pravdepodobnosťou $\frac{1}{2}$ zvolí rovnakú bázu ako Alica a správne dešifruje bit. Ak však zvolí nesprávnu bázu, tak s pravdepodobnosťou $\frac{1}{2}$ uhádne správny bit. Čiže pravdepodobnosť, že Eva zostane neodhalená pri porovnaní jedného bitu je $\frac{3}{4}$, pri porovnaní *n* bitov je to $\left(\frac{3}{4}\right)^n$ a s pravdepodobnosťou $1-\left(\frac{3}{4}\right)^n$ ju porovnanie n bitov odhalí. Pri voľbe $n=32$ je táto pravdepodobnosť 99,9899%.

1.	1	0	0	0	1	1	0	1	1	1	0	1	0	1	1	0	0	0	1	1	0
2.	+	×	+	×	×	+	×	+	+	×	+	+	×	×	×	+	×	+	×	×	+
3.	↕	⤡	↔	⤡	⤢	↕	⤡	↕	↕	⤢	↔	↕	⤡	⤢	⤢	↔	⤡	↔	⤢	⤢	↔
4.	×	+	+	×	+	×	+	×	+	+	+	×	×	×	+	×	+	×	+	×	×
5.	⤢	↔	↔	⤡		⤢	↔	⤢	↕	↔	↔	⤢	⤡	⤢	↕		↕	⤡	↕	⤢	⤢
6.	1	0	0	0		1	0	1	1	0	0	1	0	1	1		1	0	1	1	1
7.	×	+	+	×		×	+	×	+	+	+	×	×	×	+		+	×	+	×	×
8.			✓	✓					✓		✓		✓	✓						✓	
9.			0	0					1		0		0	1						1	
10.				0									0								
11.				✓									✓								
12.			0						1		0			1						1	

Tabuľka 4. Distribúcia kľúča pomocou protokolu BB84

4.3. Kvantová mechanika v kryptoanalýze

Z predchádzajúcich kapitol vieme, že bezpečnosť kryptosystému RSA je založená na veľkej zložitosti riešenia problému faktorizácie čísla na prvočíselné faktory. V nasledujúcej časti ukážeme ako sa dajú využiť vlastnosti kvantovej mechaniky na faktorizáciu čísel.

4.3.1. Kvantová Fourierova transformácia

Kvantová Fourierova transformácia tvorí základ Shorovho faktorizačného algoritmu. Ide v podstate o diskrétnu Fourierovu transformáciu aplikovanú na stavy kvantového počítača.

Táto transformácia je daná maticou, ktorej riadky a stĺpce sú indexované od 0 podľa stavov kvantového počítača. Je to unitárna matica, ktorej prvok na pozícii $[a, c]$ je $\frac{1}{q^{\frac{1}{2}}} e^{\frac{2\pi i a c}{q}}$, kde $0 \leq a < q$ pre ľubovoľné q. Teda kvantová Fourierova transformácia prevádza stav $|a\rangle$ do stavu $\frac{1}{q^{\frac{1}{2}}} \sum_{c=0}^{q-1} |c\rangle e^{\frac{2\pi i a c}{q}}$.

Pri realizácii Shorovho faktorizačného algoritmu budeme potrebovať kvantovú Fourierovu transformáciu pre q v tvare $q = 2^m, m \in \mathbb{Z}$. V nasledujúcom teda ukážeme, ako skonštruovať maticu takejto transformácie. Jedná sa o rýchlu Fourierovu transformáciu.

Položme $q = 2^m, m \in \mathbb{Z}$, číslo a vyjadríme binárne ako $|a_{m-1} a_{m-2} \ldots a_0\rangle$. Budeme používať dva typy kvantových brán, a to R_j a $S_{j,k}$, ktoré pracujú s bitmi na j-tej, resp. j-tej a k-tej pozícii kvantového počítača, kde $j < k$:

$$R_j = \begin{vmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{vmatrix}, \quad S_{j,k} = \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{i\theta_{k-j}} \end{vmatrix}, \text{ kde } \theta_{k-j} = \frac{\pi}{2^{k-j}}. \text{ [20]}$$

Takže na zostavenie kvantovej Fourierovej transformácie budeme aplikovať matice R_j a $S_{j,k}$ nasledovne: bránu R_j aplikujeme v poradí od R_{m-1} po R_0 a bránu $S_{j,k}$ aplikujeme tak, že medzi R_{j+1} a R_j použijeme všetky $S_{j,k}$, pre ktoré $j < k$, teda dostávame

$$R_{m-1}S_{m-2,m-1}R_{m-2}S_{m-3,m-1}S_{m-3,m-2}R_{m-3} \dots R_1 S_{0,m-2} \dots S_{0,2}S_{0,1}R_0 .$$

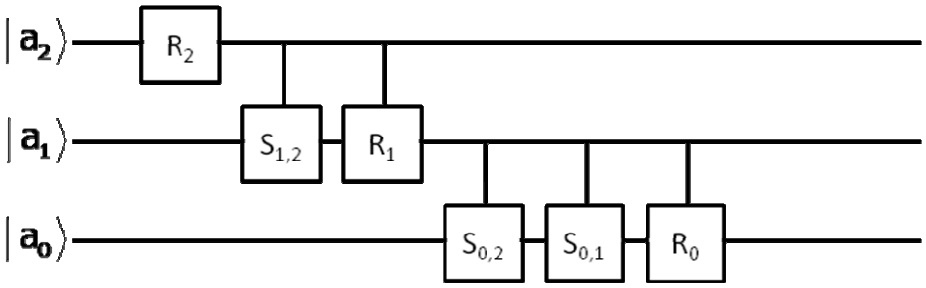
Teda na kvantovú Fourierovu transformáciu pre $q = 2^m, m \in \mathbb{Z}$ budeme potrebovať $\frac{m(m+1)}{2}$ kvantových brán. Keďže táto procedúra vracia bitovo prevrátený register, na dokončenie kvantovej Fourierovej transformácie musíme register spätne invertovať.

Príklad:

Nech $m = 3$, máme teda trojbitový register a použijeme 6 kvantových brán v tomto poradí:

$$R_2 S_{1,2} R_1 S_{0,2} S_{0,1} R_0$$

Obvod tejto kvantovej Fourierovej transformácie môžeme znázorniť takto [3]:



Obrázok 2. Obvod kvantovej Fourierovej transformácie pre m=3

4.3.2. Shorov faktorizačný algoritmus

Problém hľadania prvočíselných faktorov sa dá pretransformovať na problém hľadania periódy funkcie, ktorý sa dá riešiť na kvantovom počítači. [20]

Majme dané číslo $n \in \mathbb{N}$, ktoré chceme faktorizovať. Zostrojíme funkciu $f_{y,n}(a) = y^a \pmod n$, ktorá je periodická a kde y je náhodné číslo také, že $y \in \mathbb{Z}, (y, n) = 1$. Perióda tejto funkcie je rovná rádu prvku y v grupe $\mathbb{Z}_n^*$ a označíme ju r, $r \in \mathbb{N}$. Potom platí

$$y^{a+r} \equiv y^a \pmod n$$

$$y^a \cdot y^r \equiv y^a \pmod n$$

$$y^r \equiv 1 \pmod n$$

$$\left(y^{\frac{r}{2}}\right)^2 - 1 \equiv 0 \pmod n$$

$$\left(y^{\frac{r}{2}} + 1\right)\left(y^{\frac{r}{2}} - 1\right) \equiv 0 \pmod n$$

Ak je r nepárne, musíme zvoliť iné y. Z posledného tvaru je vidieť, že ak $y^{\frac{r}{2}} \not\equiv \pm 1 \pmod n$, tak potom aspoň jedno z čísel $\left(y^{\frac{r}{2}} + 1\right)$, $\left(y^{\frac{r}{2}} - 1\right)$ musí mať spoločného deliteľa s n. Tieto delitele nemusia byť ešte prvočíselné. V rozklade získaných faktorov pokračujeme, až kým nedostaneme prvočísla.

Týmto spôsobom sa dá dopracovať k faktorizácii čísla n s pravdepodobnosťou $1 - \frac{1}{2^{k-1}}$, kde k je počet navzájom rôznych nepárnych prvočíselných faktorov čísla n, pre ľubovoľné y.
V nasledujúcej časti načrtneme dokaz tohto tvrdenia (podľa [6]), teda že $P(r$ je párne a $y^{\frac{r}{2}} \not\equiv \pm 1 \bmod n) \geq 1 - \frac{1}{2^{k-1}}$.

Dôkaz:

Ukážeme, že $P(r$ je nepárne alebo $y^{\frac{r}{2}} \equiv -1 \bmod n) \leq \frac{1}{2^{k-1}}$.

Zapíšme si n v tvare $n = \prod_{i=1}^{k} p_i^{\alpha_i}$. Označme $r_i \in \mathbb{N}$ rád prvku $y \bmod p_i^{\alpha_i}$, t.j. $y^{r_i} \equiv 1 \left(\bmod p_i^{\alpha_i}\right)$. Potom r je najmenší spoločný násobok všetkých r_i.

Platí

$$y^{\frac{r}{2}} \equiv -1 \left(\bmod n\right) \Leftrightarrow y^{\frac{r}{2}} \equiv -1 \left(\bmod p_i^{\alpha_i}\right)$$

Implikácia zľava doprava je zrejmá, v opačnom smere vyplýva z Čínskej vety o zvyškoch.

Uvažujme pre každé i najväčšiu mocninu dvojky t_i, ktorá delí r_i, t.j. $r_i = s_i \cdot 2^{t_i}$. Ak označíme $s = nsn\left(s_1, \ldots, s_k\right)$ a $t = \max\left(t_1, \ldots, t_k\right)$, potom $r = s \cdot 2^t$.

Ak pre každé $i = 1, \ldots, k$ platí $y^{\frac{r}{2}} \equiv -1 \left(\bmod p_i^{\alpha_i}\right)$, tak potom $t_i = t$. Ak by $t_i < t$, tak r_i musí deliť $\frac{r}{2}$. To však nemôže nastať, lebo $y^{\frac{r}{2}} \equiv -1 \left(\bmod p_i^{\alpha_i}\right)$ a zároveň $y^{r_i} \equiv 1 \left(\bmod p_i^{\alpha_i}\right)$.

Tiež je vidieť, že t je nepárne práve vtedy, keď $\forall i : r_i$ je nepárne, a teda $t_1 = \cdots = t_k = 0$.

Teraz môžeme vyjadriť

$$P\left(r \text{ je nepárne alebo } y^{\frac{r}{2}} \equiv -1 \left(\bmod n\right)\right) \leq P\left(t_1 = \cdots = t_k\right)$$

Multiplikatívna grupa modulo p^{α} je cyklická pre ľubovoľné nepárne p^{α}, kde p je prvočíslo rôzne od 2. [10] Takže $\forall i, j : P\left(t_i = j\right) \leq \frac{1}{2}$.

Dostávame

$$P\left(t_1 = \cdots = t_k\right) = \sum_j \prod_{i=1}^{k} P\left(t_i = j\right) =$$
$$= \sum_j P\left(t_1 = j\right) \cdot \ldots \cdot P\left(t_k = j\right) \leq$$
$$\leq \sum_j P\left(t_1 = j\right) \cdot \frac{1}{2} \cdot \ldots \cdot \frac{1}{2} =$$
$$= \sum_j P\left(t_1 = j\right) \cdot \frac{1}{2^{k-1}} = \frac{1}{2^{k-1}}$$

QED

Kryptosystém RSA používa číslo n zložené z dvoch prvočísel, teda úspešnosť prelomenia RSA touto metódou bude 50%.

Takže sme pôvodnú úlohu previedli na problém hľadania najväčšieho spoločného deliteľa, ktorý sa dá efektívne riešiť napríklad pomocou Euklidovho algoritmu v polynomiálnom čase aj na klasickom počítači. Problémom však zostáva hľadanie periódy funkcie, resp. rádu prvku v grupe. Tento problém nie je riešiteľný v polynomiálnom čase na klasickom počítači, avšak efektívne ho vieme riešiť pomocou Shorovho algoritmu na kvantovom počítači.

Majme číslo *g* a číslo *n*, ktoré chceme faktorizovať. Na výpočet rádu prvku *g* v grupe $\mathbb{Z}_n^*$ budeme potrebovať kvantový počítač s dvoma kvantovými registrami. Stav tohto počítača bude popisovať priestor, ktorého bázu tvoria vektory $|a\rangle|b\rangle$, kde *a* je celé číslo zodpovedajúce vlastnému stavu prvého registra a *b* je celé číslo zodpovedajúce vlastnému stavu druhého registra. V prvom kroku nájdeme mocninu dvojky *q* takú, že $n^2 \leq q < 2n^2$. Teraz potrebujeme nastaviť prvý register do vyváženej superpozície stavov reprezentujúcich číslo $a \bmod q$, t.j. dať každý bit v prvom registri do superpozície $\frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right)$, čiže dostaneme počítač do stavu

$$|\psi_1\rangle = \frac{1}{q^{1/2}} \sum_{a=0}^{q-1} |a\rangle |0\rangle$$

Ďalším krokom je aplikovanie funkcie $f(x) = g^x \pmod n$, ktorej periódu hľadáme, na prvý register. Výsledok uložíme do druhého registra, v prvom registri ostáva hodnota *a*, preto je tento proces reverzibilný. Tento krok prevedie náš počítač do stavu

$$|\psi_2\rangle = \frac{1}{q^{1/2}} \sum_{a=0}^{q-1} |a\rangle |g^a \pmod n\rangle$$

Teraz na prvom registri prevedieme kvantovú Fourierovú transformáciu a dostaneme stav

$$|\psi_3\rangle = \frac{1}{q} \sum_{a=0}^{q-1} \sum_{c=0}^{q-1} e^{\frac{2\pi i a c}{q}} |c\rangle |g^a \pmod n\rangle$$

V tomto momente môžeme zmerať stav počítača. Pravdepodobnosť, že sa nachádza v stave $|c, g^k \pmod n\rangle$ pre $0 \le k < r$, dostaneme sumáciou cez všetky možné hodnoty *a*, pre ktoré sa dá stav $|c, g^k \pmod n\rangle$ dosiahnuť. Takže výsledná pravdepodobnosť potom je

$$\left| \frac{1}{q} \sum_{\substack{a: g^a \equiv g^k \\ 0 \le a < q}} e^{\frac{2\pi i a c}{q}} \right|^2$$

Keďže funkcia je periodická, dochádza na koeficientoch v superpozícii v stave $|\psi_3\rangle$ k interferencii. Interferujú koeficienty stavov $|c\rangle |f(a_1)\rangle$ a $|c\rangle |f(a_2)\rangle$, kde $f(x) = g^x \pmod n$ a $r/(a_1 - a_2)$. Keď položíme $x \equiv a \pmod r$, teda $\forall j \in \mathbb{N} : a = jr + x$, tak môžeme písať $|c\rangle |f(a_1)\rangle = |c\rangle |f(a_2)\rangle = |c\rangle |f(x)\rangle$. To znamená, že ide o ten istý stav, ktorý sa nachádza na viacerých miestach výrazu $|\psi_3\rangle$. [18]

Superpozičný koeficient, ktorý prislúcha stavu $|c\rangle |f(x)\rangle$ môžeme zapísať v tvare

$$\omega_c^{\lceil x \rceil} = \frac{1}{q} \sum_{j=0}^{\frac{q}{r}-1} e^{\frac{2\pi i (jr+x)c}{q}} .$$

Pomocou Eulerovej formuly $e^{ix} = \cos x + i \sin x$ sa dá ukázať, že

$$\sum_{j=0}^{\frac{q}{r}-1} e^{2\pi i \frac{jrc}{q}} = \begin{cases} \frac{q}{r}, & \text{ak } q/rc \\ 0, & \text{inak} \end{cases} .$$

Potom koeficient

$$\omega_c^{\lceil x \rceil} = \begin{cases} \frac{1}{r} e^{\frac{2\pi i x c}{q}}, & \text{ak } q/rc \\ 0, & \text{inak} \end{cases} .$$

Stav počítača môžeme zapísať v tvare

$$\left| \psi_3 \right\rangle = \frac{1}{r} \sum_{x=0}^{r-1} \sum_{c: g/rc} e^{\frac{2\pi i x c}{q}} \left| c \right\rangle \left| g^x \pmod n \right\rangle .$$

Meraním prvého registra dostaneme hodnotu c, ktorá spĺňa podmienku q/rc. Teda poznáme c a g, a vieme, že platí $\frac{c}{q} = \frac{b}{r}$, kde $b \in \mathbb{Z}$. Ak $(b, r) = 1$, potom vieme určiť r vykrátením zlomku $\frac{c}{q}$ do základného tvaru.

Ak nájdené r nie je periódou funkcie, postup opakujeme. Po $O(\log r)$ opakovaniach dostaneme vyhovujúce r s pravdepodobnosťou blízkou 1.

Ak však zvolíme q, ktoré nespĺňa podmienku $rc \pmod q \equiv 0$, tak sa dá ukázať, napr. v [6], že ak q spĺňa podmienku $n^2 \le q < 2n^2$, tak po polynomiálnom počte opakovaní dostaneme s vysokou pravdepodobnosťou c, ktoré vyhovuje podmienke $-\frac{r}{2} \le rc \pmod q \le \frac{r}{2}$, čo je ekvivalentné výrazu $|rc - bq| \le \frac{r}{2}$, kde $b \in \mathbb{Z}$. Malou úpravou dostaneme

$$\left|\frac{c}{q}-\frac{b}{r}\right| \leq \frac{1}{2q}.$$

Znovu poznáme *c* a *q*, ak $(b,r)=1$, dostaneme hodnotu r pomocou reťazového zlomku, ktorým budeme aproximovať zlomok $\frac{c}{q}$. [18]

Algoritmus:

VSTUP: zložené celé číslo *n*

VÝSTUP: netriviálny faktor *p* čísla *n*

1. Zvoľ číslo $q \in \mathbb{Z}$ také, že $n^2 \leq q < 2n^2, q = 2^m, m \in \mathbb{Z}$

   Zvoľ číslo $g \in \mathbb{Z}$ také, že $g < n, (g,n) = 1$

   Definuj $f(x) \overset{df}{\equiv} g^x \pmod n$

2. Superpozícia stavov 1. registra $\rightarrow |\psi_1\rangle = \frac{1}{q^{1/2}} \sum_{a=0}^{q-1} |a\rangle |0\rangle$

3. Aplikuj $f(x) \equiv g^x \pmod n$ na 1. register $\rightarrow$

   $|\psi_2\rangle = \frac{1}{q^{1/2}} \sum_{a=0}^{q-1} |a\rangle |g^a \pmod n\rangle$

4. Kvantová Fourierova transformácia na 1. registri $\rightarrow$

   $\rightarrow |\psi_3\rangle = \frac{1}{q} \sum_{a=0}^{q-1} \sum_{c=0}^{q-1} e^{\frac{2\pi i a c}{q}} |c\rangle |g^a \pmod n\rangle$

5. Zmeraj hodnotu prvého registra, výsledok $\rightarrow c$

6. Aproximuj zlomok $\frac{c}{q}$ ako reťazový zlomok $\frac{b}{r}$

7. Ak *r* nie je perióda funkcie, tak pokračuj krokom 2, inak pokračuj ďalším krokom

8. Ak r je nepárne, tak pokračuj krokom 1, inak pokračuj ďalším krokom

9. $y \leftarrow g^{\frac{r}{2}} \pmod n$

10. Ak $y = n - 1$, tak pokračuj krokom 1, inak pokračuj ďalším krokom
11. $p \leftarrow \left(n, \left(y - 1\right)\right)$
12. Vypíš p a skonči

Kroky 2 – 5 prebiehajú na kvantovom počítači, kroky 6 – 12 na klasickom počítači.

Zložitosť algoritmu je $O\left(P\left(\log n\right)\right)$, krokov na klasickom počítači, kde $P\left(x\right)$ je polynóm a $O\left(\left(\log n\right)^2 \left(\log\log n\right)\left(\log\log\log n\right)\right)$ krokov na kvantovom počítači. [18]

5. Návrh programu

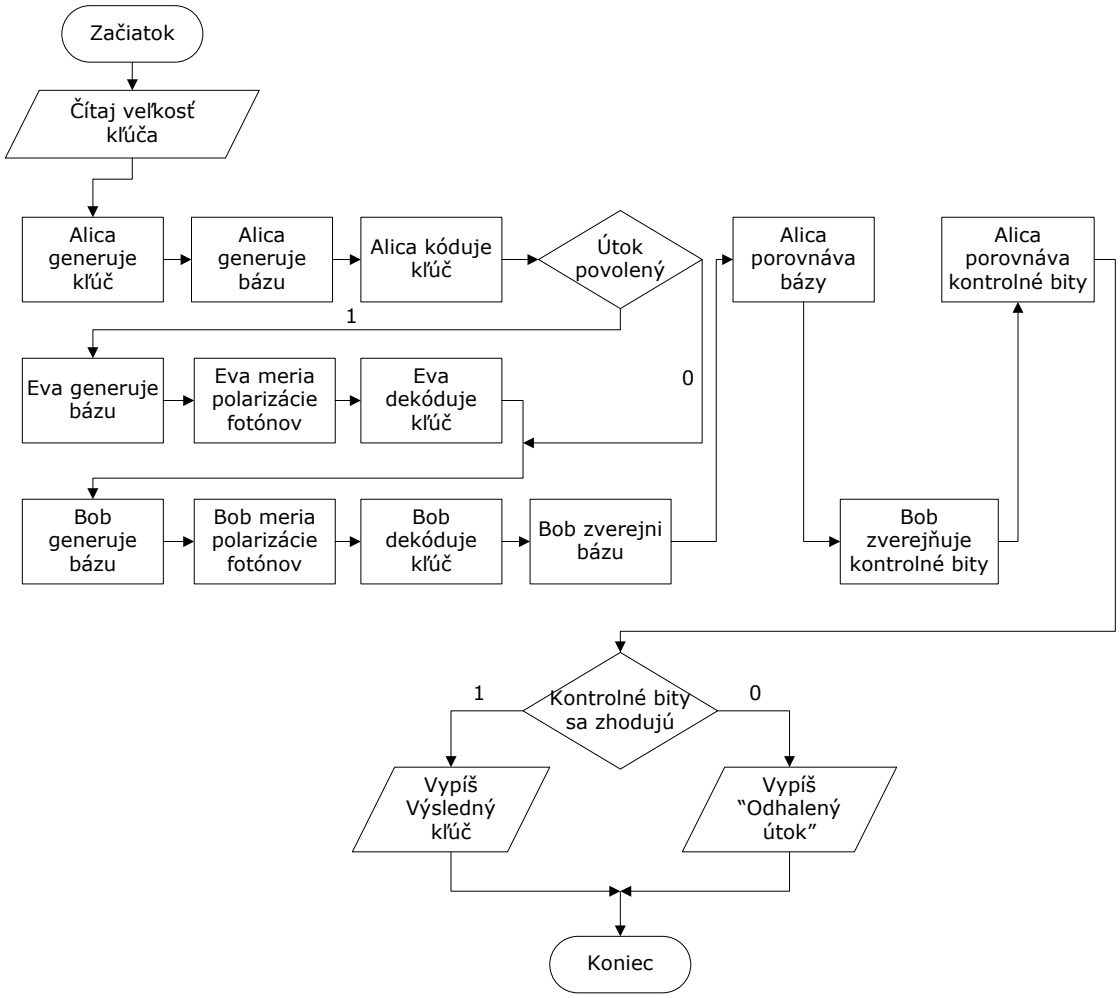
Cieľom programu je názorne vysvetliť princíp bezpečného prenosu kľúča pomocou protokolu BB84, viď s. 44. V jednotlivých krokoch ilustruje postup oboch účastníkov komunikácie, Alice a Boba, s možnosťou zapojiť do aktivity narušiteľa Evu. Pomocou programu sa dá v praxi demonštrovať závislosť šance odhalenia útočníka od dĺžky kľúča.

5.1. Popis programu

Program ilustruje protokol BB84 pomocou tabuľky, v ktorej každý riadok zodpovedá jednému kroku účastníka komunikácie. Činnosť programu je možné demonštrovať buď pomocou jednotlivých krokov, a teda tabuľka sa vypĺňa po riadkoch, alebo vygenerovaním celej tabuľky naraz. Užívateľ má možnosť zvoliť si dĺžku prenášaného kľúča, a tiež či bude simulovať protokol bez útoku alebo s ním.

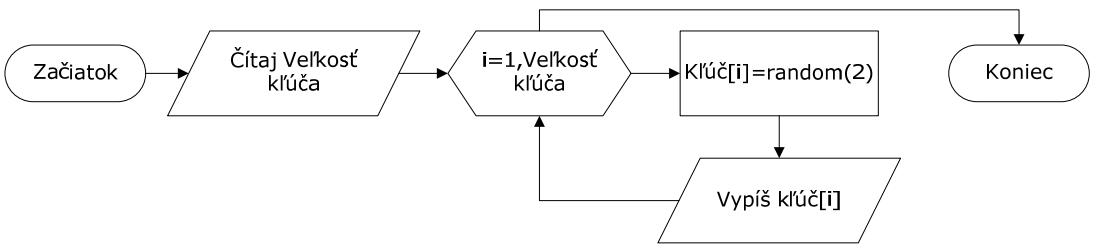
Činnosť programu môžeme znázorniť pomocou vývojového diagramu na obrázku 3. Je vidieť, že vstupom je len dĺžka kľúča a výstup je buď výsledný kľúč alebo oznámenie o odhalení útoku.

Každá procedúra zodpovedá jednému kroku protokolu BB84 a prislúcha jej jedno tlačidlo. Procedúra generovania bázy je rovnaká pre Alicu, Boba aj Evu, procedúra na dekódovanie kľúča je spoločná pre Boba a Evu. Jednotlivé procedúry graficky znázorňujú vývojové diagramy na obrázkoch 4 až 10.

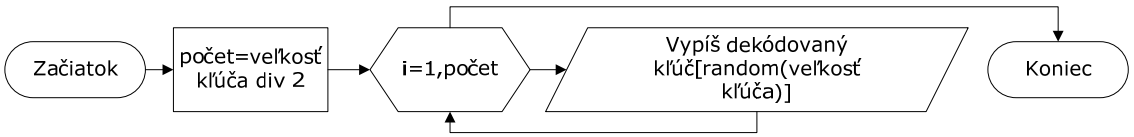


Obrázok 3. Vývojový diagram programu

Nasledujúci obrázok znázorňuje procedúru generovania kľúča. Vstupným údajom je dĺžka kľúča, výstupným sekvencia kľúča v dvojkovej sústave.

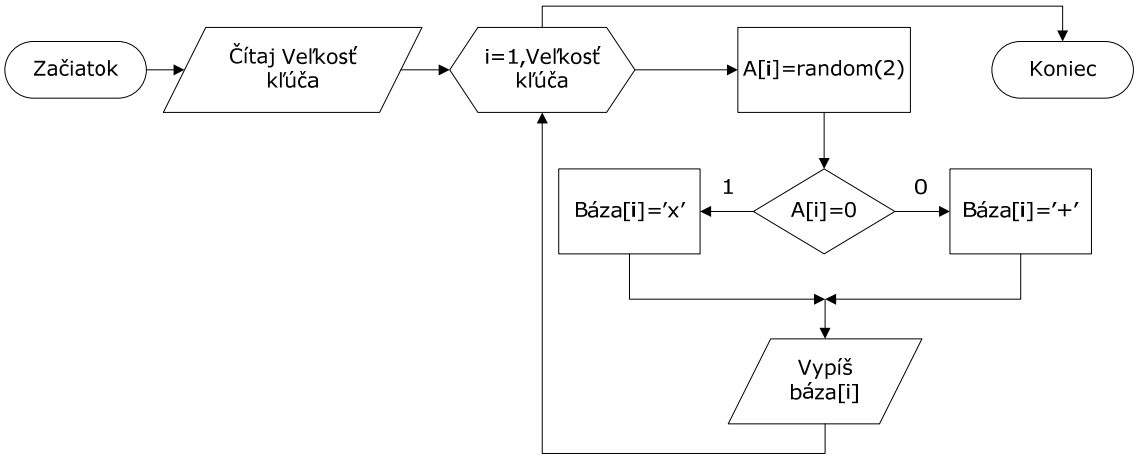


Obrázok 4. Generovanie kľúča



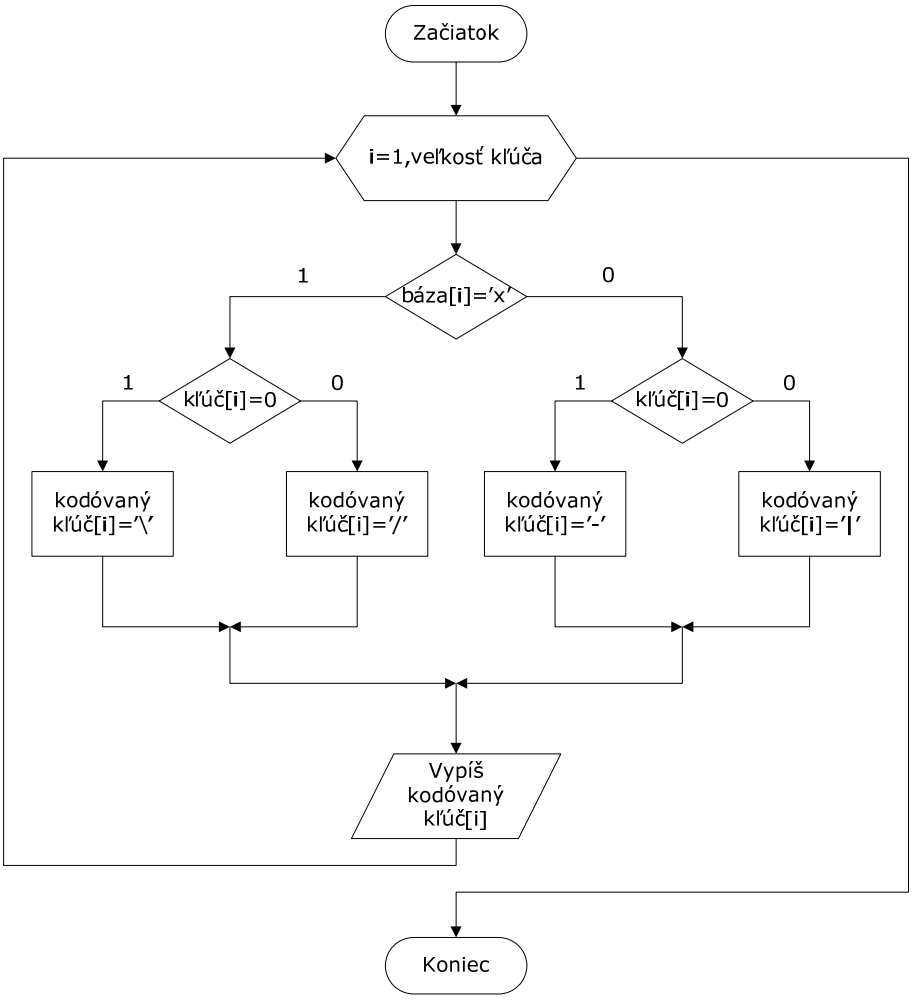
Obrázok 5. Zverejnenie kontrolných bitov

Na obrázku 5 je znázornená procedúra zverejnenia kontrolných bitov. Počet zverejnených bitov závisí od veľkosti kľúča.



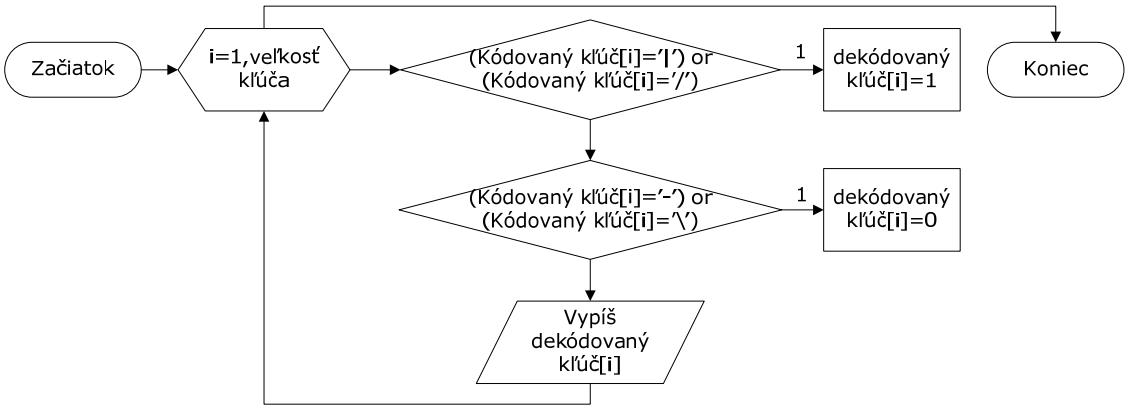
Obrázok 6. Generovanie bázy

Generovanie bázy, znázornené na obrázku 6, znamená náhodné generovanie sekvencie, pozostávajúcej zo symbolov +, ×, ktoré prislúchajú jednotlivým bázam.

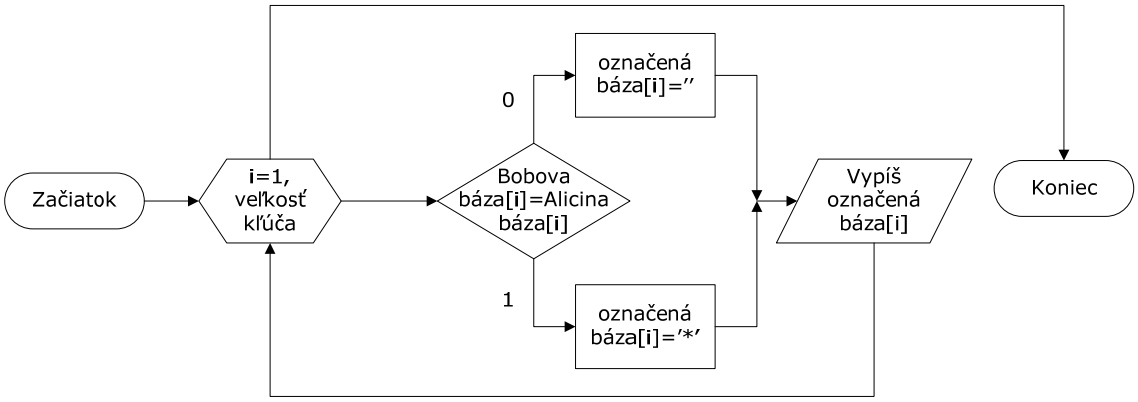


Obrázok 7. Kódovanie kľúča

Obrázok 7 ilustruje proces kódovania kľúča podľa zvolenej bázy, viď tabuľka 3. Spätný proces dekódovania kľúča ilustruje obrázok 8.

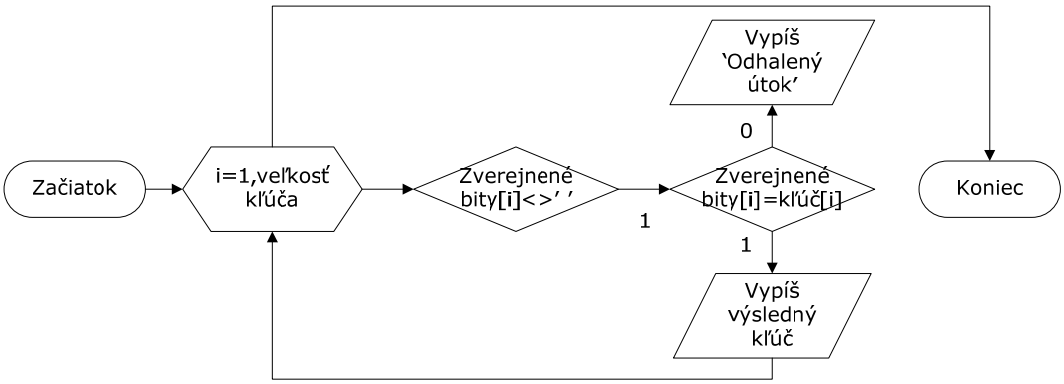


Obrázok 8. Dekódovanie kľúča



Obrázok 9. Porovnanie báz

Pri porovnávaní báz Alica označí zhodné bázy symbolom *, ak sú bázy rôzne, ponechá Alica na danom mieste prázdne políčko.

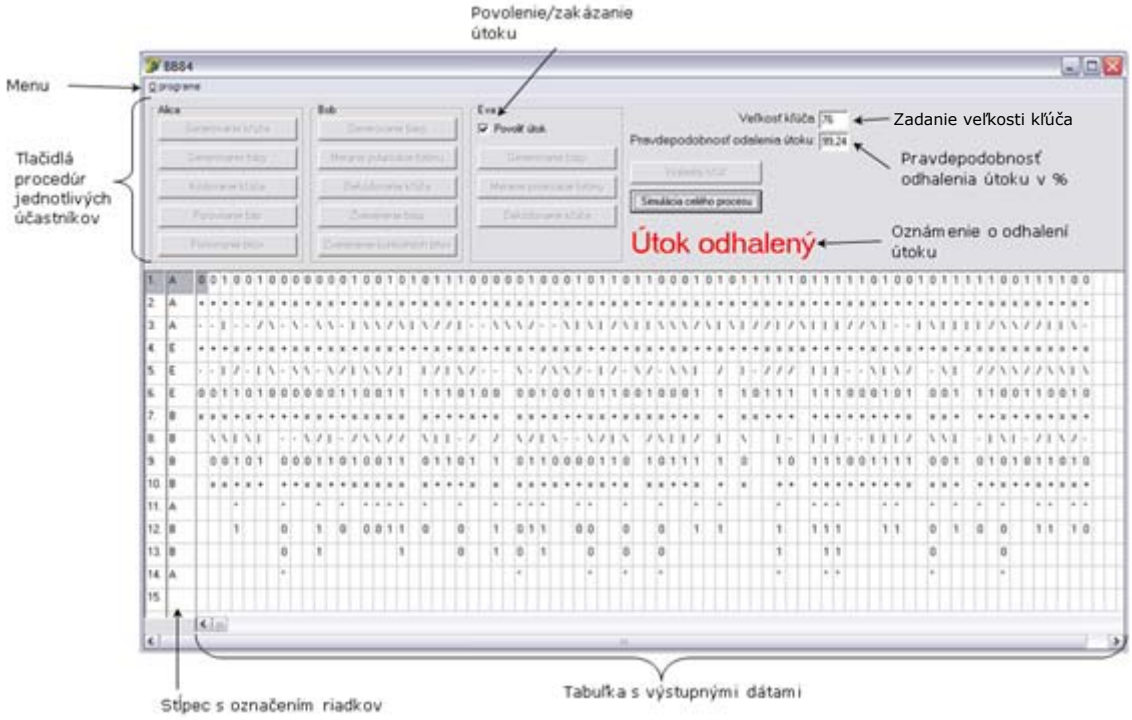


Obrázok 10. Porovnanie kontrolných bitov

Ak pri porovnávaní kontrolných bitov niektorá porovnávaná dvojica nie je zhodná, tak je odhalený útok, čo je oznámené výpisom, v opačnom prípade sa vypíše výsledný kľúč.

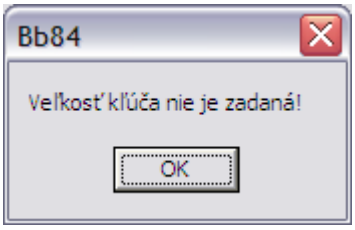
5.2. Užívateľské prostredie

Užívateľské prostredie tvorí okno s tlačidlami procedúr, výstupnou tabuľkou a ďalšími ovládacími prvkami. Užívateľské prostredie je popísané na obrázku 11.



Obrázok 11. Popis užívateľského prostredia

Prvým krokom užívateľa je zadanie veľkosti kľúča v bitoch do editačného riadku „Veľkosť kľúča". Je možné zadať celé číslo v rozsahu 1 až 1000. Ak užívateľ tento krok vynechá, program ho v ďalšom kroku upozorní zobrazením upozornenia (obrázok 12).



Obrázok 12. Upozornenie na nezadanú veľkosť kľúča

V ďalšom môže užívateľ zaškrtnutím checkboxu „Povoliť útok" rozhodnúť o zapojení útočníka do komunikácie. Tlačidlá procedúr útočníka Evy sa zobrazia pri zaškrtnutí checkboxu „Povoliť útok" a pri jeho odškrtnutí sa skryjú (viď obrázky 13 a 14).

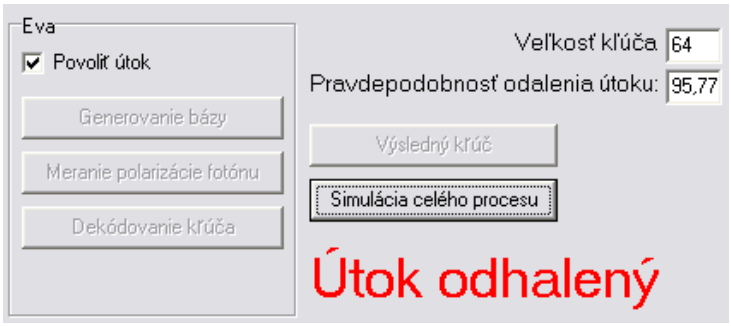
Obrázok 13. Útok nepovolený

Obrázok 14. Útok povolený

V každom kroku tejto simulácie je aktívne len jedno tlačidlo, zodpovedajúce nasledujúcemu kroku. Po stlačení tlačidla sa vyplní príslušný riadok tabuľky, označí sa v druhom stĺpci riadok podľa toho, čí krok je v danom riadku znázornený (A – Alica, B – Bob, E - Eva) a aktivuje sa tlačidlo prislúchajúce nasledujúcemu kroku.

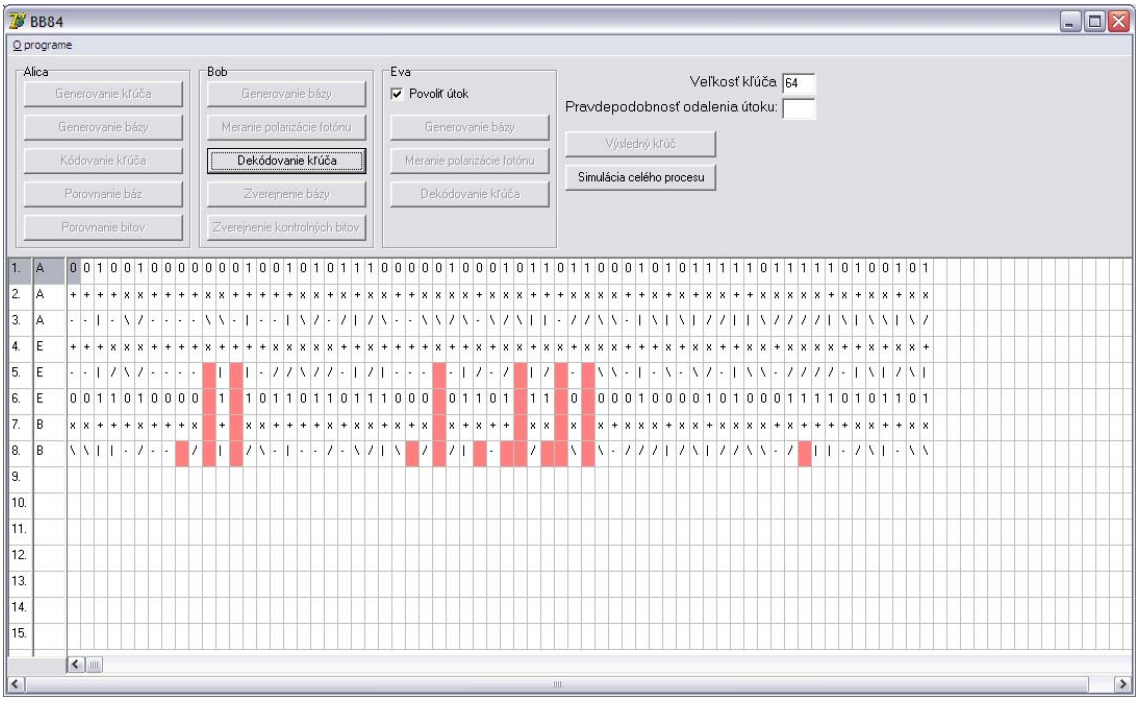
Program tiež počíta s technologickými stratami v prenose polarizovaných fotónov cez kvantový kanál. V mieste, kde Bob, resp. Eva, nenamerajú žiaden fotón, zostane prázdne políčko.

Obrázok 16 ilustruje situáciu po stlačení tlačidla „Meranie polarizácie fotónu" Bobom. Je vidieť, že ôsmy riadok je označený písmenom B a je vyplnený symbolmi /, \, |, —, ktoré reprezentujú jednotlivé polarizácie fotónov. Technologické straty, teda miesta, kde Bob, resp. Eva nenamerali žiadny fotón, sú označené červenou farbou.



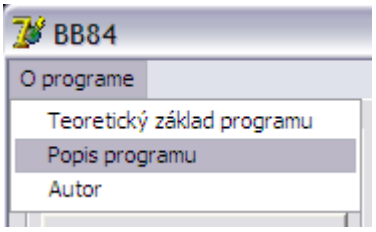
Obrázok 15. Informácia o odhalení útoku

Odhalenie útočníka je oznámené zobrazením nápisu „Útok odhalený" (obrázok 15). Keďže pravdepodobnosť odhalenia útoku závisí na počte zverejnených kontrolných bitov, jej hodnota sa zobrazí až po vykonaní procedúry „Zverejnenie kontrolných bitov".



Obrázok 16. Príklad simulácie protokolu BB84

Užívateľ má možnosť použiť menu programu (obrázok 17) s tromi položkami, ktoré obsahujú informácie o teoretickom pozadí programu, používaní programu a autorovi.



Obrázok 17. Menu programu

Záver

V súčasnosti tvorí komunikácia a výmena informácií neoddeliteľnú časť života. To však v sebe nesie aj riziko odchytenia informácie treťou osobou. Preto je nevyhnutné, aby boli informácie zabezpečené šifrovaním.

Dnes používané kryptosystémy sú do istej miery bezpečné, v princípe sú však prelomiteľné. V práci sme ukázali možné použitie grúp v kryptosystéme. Takýto kryptosystém nie je ohrozený zvyšujúcou sa efektivitou faktorizácie s rastúcou výpočtovou silou počítačov. Ako ďalšou použiteľnou štruktúrou sa ukazujú grafy a ich dedičné vlastnosti.

Rozhodujúci vplyv na budúcnosť kryptosystémov má kvantová technológia. Shorov faktorizačný algoritmus dokáže potenciálne v polynomiálnom čase faktorizovať číslo, a teda predstavuje hrozbu pre súčasné kryptosystémy.

Na druhej strane, použitie kvantovej technológie na prenos kľúča vytvára podmienky pre neprelomiteľnú Vernamovu šifru. Vytvorili sme program, ktorý simuluje prenos kľúča pomocou protokolu BB84, ktorý využíva kvantové častice. Tento program pomáha lepšie pochopiť bezpečný prenos kľúča a pre jeho názornosť je vhodný na použitie v pedagogickom procese.

Literatúra

[1] BENNETT, C. H. - BRASSARD, G.: Quantum cryptography: Public-key distribution and coin tossing, Proceedings of IEEE International Conference on Computers, Systems and Signal Processing, Bangalore, India, December 1984, pp. 175 - 179.

[2] BOHLI, J. - VASCO, M. - MARTÍNEZ, C.: Weak Keys in MST1. Oviedo : Universidad de Oviedo, 2002. 19 s.

[3] BROWN, M. S.: Classical Cryptosystems In A Quantum Setting. Waterloo (Ontario, Canada): University of Waterloo, 2004. 170 s.

[4] CAMERON, P.: Notes on cryptography. London : University of London, 2003. 132 s.

[5] CASE, M.: A Beginner's Guide To The General Number Field Sieve. Oregon: Oregon State University, 2003. 19 s.

[6] EKERT, A. - JOZSA, R.: Quantum Computation and Shor's Factoring Algorithm. In Reviews of Modern Physics, Vol.68, no. 3, p. 733 - 753.

[7] GOLDWASSER, S. - BELLARE, M.: Lecture Notes on Cryptography. Cambridge: s.n., 2001. 283 s.

[8] KLÍMA, V.: Dvě čísla za 200 000 dolarů. In Chip. ISSN 1210-0684, 2001, vol. 11, no. 9, p. 176 – 181

[9] KLÍMA, V.: Kritika článku "Bezpečnost RSA - význačný posun?". In Crypto-World. ISSN 1801-2140, 2002, vol. 4, no. 4, p. 16 -17

[10] KNUTH, D. E.: The Art of Computer Programming. Vol. 2: Seminumerical Algorithms, Second ed., Addison-Wesley, 1981

[11] MAGLIVERAS, S. - STINSON, D. – VAN TRUNG, T.: New approaches to designing public key cryptosystems using one-way functions and trap-doors in finite groups. Lincoln : University of Nebraska, 2004. 15 s.

[12] MAGLIVERAS, S.: Secret And Public-Key Cryptosystems From Group Factorizations, Tatra Mt. Math. Publ. 25 (2002), 1-12 s.
[13] MENEZES, A. - VANOORSCHOT, P. - VANSTONE, S.: Handbook of Applied Cryptography. Boca Raton: CRC Press, 1996. 816 s.
[14] MIHÓK, P. – SEMANIŠIN, G.: Unique Factorization Theorem and Formal Concept Analysis. Košice: s.n., 2006. 9 s.
[15] MILNE, J.: Group Theory. Ann Arbor: s.n., 2001. 121 s.
[16] NOVÁK, V.: Kvantová kryptografia. Bratislava : s.n., 2004. 95 s.
[17] RIVEST, R. L.- SHAMIR, A. - ADLEMAN, L. A.: A method for obtaining digital signatures and public-key cryptosystems; Communications of the ACM, Vol.21, Nr.2, 1978, S.120-126. http://citeseer.ist.psu.edu/rivest78method.html
[18] ROSA, T.: Od bitů ke qubitům. In Chip. ISSN 1210-0684, 2002, vol. 12, no. 5, p. 138 – 141
[19] ROTHE, J.: Some Facets of Complexity Theory and Cryptography: A Five-Lecture Tutorial. ACM Computing Surveys, Vol. 34, No. 4, December 2002, pp. 504–549.
[20] SHOR, P.: Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer, Proceedings of the 35th Annual Symposium on Foundations of Computer Science, Santa Fe, NM, 1994, IEEE Computer Society Press, pp. 124–134.
[21] STANEK, M.: Základy kryptológie. : verzia 0.16, 2004. 130 s.
[22] STAROŇ, G.: Kvantové počítače. analyzer.nigga.sk, [cit. 30.1.2008]. Dostupné na webovskej stránke (world wide web): http://analyzer.nigga.sk/kvantove%20pocitace.pdf
[23] ŠIMKA, M.: Vložené architektúry v kryptografických systémoch. Košice : Technická univerzita v Košiciach, 2004. 68 s.
[24] VYCHODIL, V.: Algoritmus RSA. vychodil.inf.upol.cz, 2002. Dostupné na webovskej stránke (world wide web): http://vychodil.inf.upol.cz/courses/cs1pp/doc/rsa.pdf

Prílohy

Pre lepšie pochopenie problematiky uvádzame porovnania kryptografických prístupov. V prvom prípade ide o porovnanie kryptosystémov RSA a MST_1, v druhom prípade je to porovnanie distribúcie kľúča s využitím klasickej cesty (protokol Diffie-Hellman) a s využitím kvantového kanála (protokol BB84).

Kvôli názornosti a prehľadnosti sme zvolili formu tabuliek.

K práci tiež prikladáme CD nosič, ktorého obsah uvádzame v prílohe C.

A. Porovnanie kryptosystémov RSA a MST_1

	RSA	MST_1
Verejný kľúč	Dvojica (e, n), kde n je súčin dvoch veľkých prvočísel p, q, a e je náhodne zvolené číslo také, že platí: $1 < e < \varphi(n)$, $(e, \varphi(n)) = 1$	Dvojica logaritmických popisov konečnej grupy permutácií G (α, β), kde α je nepolynomiálny logaritmický popis a β je polynomiálny
Privátny kľúč	Dvojica (d, n), kde n je súčin $p \cdot q$ a d je také číslo, že $1 < d < \varphi(n)$ a platí $d \equiv e^{-1} \pmod{\varphi(n)}$	Postupnosť $[\theta_1, \ldots, \theta_k] \in \mathcal{T}(G)^k$, kde $\mathcal{T}(G)$ je množina transverzálnych logaritmických popisov grupy G, pre ktorú platí $\hat{\beta}^{-1}\hat{\alpha} = \hat{\theta}_1 \cdots \hat{\theta}_k$
Šifrovanie	$m \rightarrow m_i$, m_i – bloky rovnakej dĺžky $m_i \rightarrow c_i \equiv m_i^e \pmod{n}$ $c_i \rightarrow$ Bob	$m \rightarrow c = \hat{\beta}^{-1}\hat{\alpha}(m) \in \mathbb{Z}_m$ $c \rightarrow$ Bob
Dešifrovanie	$c_i \rightarrow m_i \equiv c_i^d \pmod{n}$ $m_i \rightarrow m$	$c \rightarrow m = \hat{\alpha}^{-1}\hat{\beta}(c) = \hat{\theta}_k^{-1} \cdots \hat{\theta}_1^{-1}(c)$

	RSA	MST_1
Bezpečnosť	2 možnosti prelomenia RSA: A. faktorizácia čísla *n* na súčin $p \cdot q$ B. vyriešenie problému RSAP ad A.: dá sa predísť voľbou dostatočne veľkého *n*. V súčasnosti stačí zvoliť *n* o veľkosti 1024 bitov, keďže maximálne doteraz (9.5.2005) faktorizované číslo má veľkosť 663 bitov. ad B.: algoritmus na riešenie RSAP nebol doposiaľ nájdený a ani neexituje dôkaz, že existuje taký algoritmus	2 možnosti prelomenia MST_1: A. nájdenie inverzie $\hat{\alpha}$ B. faktorizácia $\hat{\beta}^{-1}\hat{\alpha}$ na súčin $\hat{\theta}_1 \cdots \hat{\theta}_k$ ad A.: $\hat{\alpha} := \breve{\eta}^{-1}\breve{\alpha}$ je jednocestná permutácia ad B.: problém faktorizácie $\hat{\beta}^{-1}\hat{\alpha}$ je ekvivalentný problému diskrétneho logaritmu
Praktické obmedzenia	prvočísla *p, q* by mali byť aspoň 512 bitové *n* by malo mať 1024 bitov *e* by malo mať 1024 bitov	Nie sú známe žiadne obmedzenia, ale je možné, že z použitia v praxi vyplynú nejaké praktické obmedzenia
Výhody	Jednoduchosť generovania dvojice kľúčov, ak je známe *p, q*	Rastúca výpočtová sila nepredstavuje bezpečnostné riziko

	RSA	MST_1
Nevýhody	S rastúcou výpočtovou silou počítačov rastú aj nároky na veľkosť *p, q*, hrozbu tiež predstavuje rozvoj kvantovej technológie	Doposiaľ neexistujú algoritmy na generovanie verejného a privátneho kľúča

B. Porovnanie distribúcie kľúča klasickou a kvantovou cestou

	Klasický prístup (protokol Diffie-Hellman)	Kvantový prístup (protokol BB84)
Vstupné podmienky	Verejná dvojica (p, g), p je prvočíslo a $g \in \mathbb{Z}_p^*$, kde g je generátor grupy $\mathbb{Z}_p^*$	Alica: zdroj fotónov a polarizátor Bob: analyzátor a detektor fotónov verejný kanál Alica$\leftrightarrow$Bob na prenos fotónov

	Klasický prístup (protokol Diffie-Hellman)	**Kvantový prístup (protokol BB84)**
Protokol	Alica $\rightarrow$ $x \in \mathbb{Z}_{p-1}$ náhodne $x \overset{A}{\rightarrow} X \equiv g^x \pmod{p}$ $X \overset{A}{\rightarrow}$ Bob Bob $\rightarrow$ $y \in \mathbb{Z}_{p-1}$ náhodne $y \overset{B}{\rightarrow} Y \equiv g^y \pmod{p}$ $Y \overset{B}{\rightarrow}$ Alica Platí: $X^y = \left(g^x\right)^y = g^{xy} = \left(g^y\right)^x = Y^x \overset{df}{=} K$ K je spoločný kľúč [23]	Alica $\rightarrow$ $(bit, báza)$ náhodne, kde $bit \in \{0,1\}$ a $báza \in \{+,\times\}$ $(bit, báza) \overset{A}{\rightarrow}$ polarizovaný fotón $\in \{\leftrightarrow, \updownarrow, \nearrow, \nwarrow\}$ polarizovaný fotón $\overset{A}{\rightarrow}$ Bob Bob $\rightarrow$ $báza \in \{+,\times\}$ náhodne $(polarizovaný$ fotón, $báza) \overset{B}{\rightarrow} bit \in \{0,1\}$ sekvencia báz $\overset{B}{\rightarrow}$ Alica pozície rovnako zvolených báz $\overset{A}{\rightarrow}$ Bob (iba bity na týchto pozíciách môžu patriť do kľúča) časť týchto bitov $\overset{B}{\rightarrow}$ Alica ak sa bity Alice a Boba zhodujú, kľúč tvorí množina bytov na pozíciách zhodných báz okrem zverejnených bitov

	Klasický prístup (protokol Diffie-Hellman)	**Kvantový prístup (protokol BB84)**
Bezpeč-nosť	Útočník Eva: - pozná *p, g*, vie odchytiť *X, Y* - nepozná *x, y* - potrebuje vypočítať *K* 2 možnosti: A. nájsť *x*, ak pozná *X*, alebo *y*, ak pozná *Y* B. riešiť D-H problém ad A.: ide o problém diskrétneho logaritmu v grupe $\mathbb{Z}_p^*$ ad B.: <u>*D-H problém:*</u> Nech $x, y \in \mathbb{Z}_{p-1}$ sú náhodné. Pri danom g^x a g^y vypočítaj g^{xy}. [13] V súčasnosti je najlepší algoritmus na riešenie D-H problému riešenie problému diskrétneho logaritmu. Je zrejmé, že ak vieme vyriešiť problém diskrétneho logaritmu, tak vieme riešiť aj D-H problém. Platnosť opačného tvrdenia zostáva dodnes otvorená.	Útočník Eva: - má plný prístup ku komunikačnému kanálu medzi Alicou a Bobom - potrebuje zistiť kľúč Jedinou možnosťou je vystupovať voči Bobovi ako Alica a opačne. V tomto prípade ju však odhalí porovnanie báz a následné porovnanie bitov pri rovnakých bázach, keďže nevie aké bázy používa Alica. Pravdepodobnosť odhalenia je $1-\left(\frac{3}{4}\right)^n$, kde *n* je počet porovnaných bitov.

C. CD nosič

Obsah priloženého CD nosiča:

- program na simuláciu protokolu BB84
- užívateľská príručka k programu
- elektronická verzia diplomovej práce

Register