VILNIAUS UNIVERSITETAS
MATEMATIKOS IR INFORMATIKOS FAKULTETAS
PROGRAMŲ SISTEMŲ BAKALAURO STUDIJŲ PROGRAMA

McEliece viešojo rakto kriptografinės sistemos saugumo tyrimas

Study of the security of McEliece public-key cryptosystem

Bakalauro baigiamasis darbas

Atliko: Ingvaras Galinskas (parašas)
Darbo vadovas: asist. dr. Gintaras Skersys (parašas)
Recenzentas: asist. dr. Linas Litvinas (parašas)

Vilnius – 2022

Santrauka

Šio darbo tiriamasis objektas yra McEliece viešojo rakto kriptografinė sistema. Darbe bendrai apžvelgiami kriptografijos aspektai: uždaviniai, saugumo sąvokos ir lygiai, analizuojami galimų atakų tipai. Taip pat darbe nagrinėjama McEliece ir Niederreiter sistemos sandara norint suprasti ir įvertinti McEliece kriptosistemos saugumą, silpnąsias vietas. Sistemos saugumas vertinamas atliekant skirtingas kriptoanalizės atakas bandant pažeisti kriptosistemos saugumą. Darbe analizuojamos atakos: perrinkimo, pranešimo persiuntimo, žinomo dalinio teksto ir gavėjo atsako. Pateikiami įgyvendintų atakų programų vykdymo laikai, atakos lyginamos tarpusavyje, įvertinami jų praktiškumo aspektai, pateikiamos rekomendacijos, kaip galima būtų apsisaugoti nuo šių atakų. Darbe prieinama prie išvados, kad McEliece kriptografinė sistema su standartiniais saugumo parametrais yra atspari nagrinėtoms atakoms, tol kol sistemos naudotojas elgiasi atsargiai ir nesuteikia kriptoanalitikams jokios papildomos informacijos.

Raktiniai žodžiai: McEliece kriptografinė sistema, Niederreiter kriptografinė sistema, kriptografija, klaidas taisantys kodai, Goppa kodai, perrinkimo ataka, pranešimo persiuntimo ataka, žinomo dalinio teksto ataka, gavėjo atsako ataka.

Summary

The research object of this paper is McEliece public key cryptosystem. General cryptography aspects are discussed in this paper: tasks, security concepts and levels; possible attack types are analyzed as well. Additionally, the structure of McEliece and Niederreiter systems is analyzed to understand and to evaluate the security of McEliece cryptosystem, its weak points. The security of the system is evaluated by executing various cryptoanalysis attack trying to break the security of the system. Attacks analyzed in this paper: brute force, message resend, known partial text and receiver response. Execution times of each attack is provided, attacks are compared with each other, practicality aspects are evaluated, and recommendations are provided to suggest the ways of improving protection against the attacks. A conclusion is made that the McEliece cryptosystem with its standard security parameters is resistant to the analyzed attacks if the user of the system is cautious and does not provide the attacker any additional information.

Keywords: McEliece cryptosystem, Niederreiter cryptosystem, cryptography, error correcting codes, Goppa codes, brute force attack, message resend attack, partially known text attack, receiver response attack.

Turinys

Įvadas 5
1. Kriptografijos aspektų apžvalga 7
1.1. Kriptografijos uždaviniai 7
1.2. Saugumo lygiai 8
1.3. Galimi atakų tipai 8
2. McEliece ir Niederreiter kriptografinės sistemos 10
2.1. Klaidas taisantys kodai 10
2.2. McEliece kriptografinė sistema 11
2.2.1. Raktų generavimas 11
2.2.2. Šifravimas 11
2.2.3. Dešifravimas 11
2.3. Niederreiter kriptografinė sistema 12
2.3.1. Raktų generavimas 12
2.3.2. Šifravimas 12
2.3.2. Dešifravimas 12
3. McEliece kriptosistemos sandaros įtaka saugumui 13
3.1. Goppa kodai 13
3.2. S ir P matricos 13
3.3. Klaidos vektorius 14
3.4. Silpnosios vietos 14
4. Atakų analizė 15
4.1. Perrinkimo ataka 15
4.2. Perrinkimo atakos įgyvendinimas 16
4.3. Pranešimo persiuntimo ataka 17
4.4. Pranešimo persiuntimo atakos įgyvendinimas 18
4.5. Žinomo dalinio teksto ataka 19
4.6. Žinomo dalinio teksto atakos įgyvendinimas 20
4.7. Gavėjo atsako ataka 21
4.8. Gavėjo atsako atakos įgyvendinimas 23
5. Atakų palyginimas ir rekomendacijos 25
Rezultatai ir išvados 27
Šaltiniai 28

Įvadas

Informacijos slėpimas žmonijai buvo aktualus net ir tais laikais, kuomet dar nebuvo kompiuterių ar kitų nuotolinio ryšio priemonių. Apie pirmuosius bandymus užšifruoti (paslėpti) informaciją liudija maždaug 3-4 tūkstantmečio pr. Kr. egiptiečių įrašai antkapiuose [Sta02]. Žmonės matė poreikį užšifruoti užrašomą informaciją taip, kad ją suprasti galėtų tik žinantys jų pačių sugalvotas šifravimo taisykles. Visuomet buvo ir tokių, kurie nežinodami dešifravimo taisyklių, vis tiek bandė jas perprasti, jog galėtų perskaityti slaptus pranešimus. Tai skatino žmones kurti vis sudėtingesnius metodus šifravimui. Ši kriptografijos metodų evoliucija lėmė tai, kad šių laikų kriptografijos metodai tapo tokiais sudėtingais, jog jiems dešifruoti nepakaktų ir visų pasaulio kompiuterinių resursų.

Šiais laikais dažniausiai naudojami šifravimo ir dešifravimo metodai remiasi viešojo ir privataus raktų principu: informacijos gavėjas sugeneruoja du raktus – vieną skirtą informacijos užšifravimui, kitą skirtą dešifravimui. Gavėjas raktą skirtą dešifravimui laiko paslaptyje ir su niekuo nesidalina – šis raktas vadinamas privačiu, o raktu skirtu užšifravimui yra pasidalinama su visais norinčiais tam gavėjui perduoti informaciją. Kadangi su viešuoju raktu duomenis galima tik užšifruoti, nėra svarbu kas sužinos šį raktą, nes jis nepagelbės su slaptos informacijos dešifravimu. Kai gavėjui yra persiunčiama užšifruota informacija, jis vienintelis gali dešifruoti ją su savo slaptu raktu, tokiu būdu užtikrinant perduotos informacijos konfidencialumą. Viena iš kriptografinių sistemų, kuri remiasi šiuo viešojo-privataus raktų principu yra McEliece kriptografinė sistema.

McEliece kriptosistema buvo pasirinkta šio darbo tyrimo subjektu todėl, kad ji yra laikoma saugia net ir prieš kvantinius algoritmus [DMR11]. Šiuo metu kvantiniai kompiuteriai dar nėra pakankamai pažangūs, jog galėtų būti efektyviai naudojami prieš aukšto saugumo lygio kriptosistemas, tačiau manoma, jog šių kompiuterių techninės galimybės ateityje didės. Tai suteikia aktualumą šio darbo temai.

Pagrindinis darbo tikslas yra išanalizuoti McEliece kriptografinės sistemos saugumą siekiant surasti kriptosistemos silpnąsias vietas. Šio darbo tikslui pasiekti buvo iškelti šie uždaviniai:

1. Bendrai apžvelgti kriptografijos aspektus: uždavinius, saugumo sąvokas ir lygius, atakų tipus.
2. Išnagrinėti McEliece ir Niederreiter kriptografinių sistemų veikimą.
3. Nustatyti McEliece kriptosistemos sandaros įtaką saugumui: apžvelgti kriptosistemos silpnas vietas, išnagrinėti silpnų raktų bei vietų išvengimo būdus, išnagrinėti McEliece kriptosistemos sandaros atskirų sudedamųjų dalių įtaką bendram sistemos saugumui, apskaičiuojant jų sudarymui galimų kombinacijų skaičių.

4. Atlikti perrinkimo, dalinio teksto, pranešimo persiuntimo, gavėjo atsako atakų prieš McEliece kriptosistemą teorinę analizę ir praktinę analizę. Įvykdyti atakas su mažais kriptosistemos parametrais, surinkti statistinius atakų vykdymo rezultatus, įvertinti, kokie būtų rezultatai su realiais parametrais.

1. Kriptografijos aspektų apžvalga

Kriptografijos mokslas tiria matematinius metodus, kuriais galima saugiai užšifruoti ir dešifruoti informaciją. Šiais laikais informacijos slaptumas dažnai nėra vienintelis kriptografijos uždavinys, todėl kriptografijos sąvoką galima apibrėžti tiksliau – „Kriptografija yra mokslas apie matematinius metodus susijusius su informacijos saugumu: konfidencialumu, duomenų vientisumu, autorizacija, duomenų kilmės autentikavimu.“ [MOV96].

1.1. Kriptografijos uždaviniai

Pagrindinis kriptografijos uždavinys yra užtikrinti informacijos saugumą. Saugumas yra gana abstrakti sąvoka, todėl kalbant apie kriptografiją, saugumo sąvoka dažnai yra išskaidoma į keturias sritis [MOV96]:

- duomenų konfidencialumas (angl. *data confidentiality*)
- duomenų vientisumas (angl. *data integrity*)
- autentiškumas (angl. *authentication*)
- veiksmų neatsisakymas (angl. *non-repudiation*)

Duomenų konfidencialumas leidžia užtikrinti, jog persiunčiama arba saugoma informacija galės būti perskaityta tik įgaliotų asmenų. Dažnu atveju įgaliotas asmuo yra tik tos informacijos savininkas (valdytojas) arba asmuo, kuriam yra persiunčiama informacija. Duomenų konfidencialumas dažnai yra užtikrinamas viešo ir privataus raktų generavimo metu, kuomet asmuo turintis privatų raktą yra įgaliotas skaityti informaciją, kuri yra užkoduojama viešuoju raktu.

Duomenų vientisumas užtikrina, kad persiųsta informacija po išsiuntimo nebuvo modifikuota kieno nors kito negu pačio siuntėjo ir jog gavėjas gavo būtent tai, kas jam buvo išsiųsta. Dažniausiai duomenų vientisumas yra užtikrinamas paskaičiuojant duomenims jų kontrolinę sumą ir perduodant ją gavėjui saugiu kanalu. Tuomet gavėjęs gavęs duomenis gali pats paskaičiuoti jų kontrolinę sumą ir sulyginti su jam persiųsta suma. Kontrolinė suma dažnai yra paremta maišos funkcija, kuri garantuoja jog nežymiai pasikeitus duomenims jų kontrolinė suma pasikeis nenuspėjamai.

Autentiškumas leidžia įsitikinti duomenų gavėjui, jog jam persiųsti duomenys buvo išsiųsti konkretaus asmens, o ne persiųsti prisidengiant kito asmens vardu. Tai dažnai yra užtikrinama viešojo ir privataus raktų principu, tik šį kartą siuntėjas turi privatų raktą, o gavėjas ir visi kiti gali patikrinti pranešimo autentiškumą viešuoju raktu. Taigi gaunasi, kad siuntėjas pranešimą šifruoja du kartus: vieną kartą savo privačiu raktu, norėdamas užtikrinti autentiškumą, o kitą kartą gavėjo

viešuoju raktu, norėdamas užtikrinti duomenų konfidencialumą [Sta02]. Tuomet gavėjas pirma dešifruoja pranešimą savo privačiu raktu, o po to siuntėjo viešuoju raktu. Jei pavyksta sėkmingai dešifruoti duomenis, vadinasi pranešimas yra autentiškas, nes ir gali priklausyti tik tam asmeniui, kuris pasidalino viešuoju raktu.

1.2. Saugumo lygiai

Ne visi šifravimo algoritmai yra vienodai saugūs – vieniems dešifruoti pakaktų vieno kompiuterio skaičiavimo resursų ir mažiau nei vienos sekundės laiko, kitiems algoritmams dešifruoti gali prireikti ir viso pasaulio skaičiavimo resursų bei daugybės metų laiko. Todėl dažnai kalbant apie kriptografinį saugumą, yra bandoma nustatyti kriptosistemos sudėtingumą. Galima išskirti kelis saugumo lygius [Sta02]:

- besąlygiškai saugus (angl. *unconditional security*)
- saugia sudėtingumo teorijos požiūriu (angl. *complexity-theoretic security*)
- įrodomas saugumas (angl. *provable security*)
- skaičiavimų požiūriu saugus (angl. *computational security*)
- euristiškai saugus

Besąlygiškai saugi kriptosistema yra tokia, kurios šifro nepavyktų dešifruoti net ir turint neribotus skaičiavimo resursus.

Saugia sudėtingumo teorijos požiūriu kriptosistema laikoma tokia, kurios šifro nepavyktų dešifruoti taikant polinominio laiko skaičiavimo algoritmus. Pavyzdžiui, kriptosistema, kurios sudėtingumas didėja eksponentiškai priklausomai nuo šifruojamų duomenų dydžio, yra saugi sudėtingumo teorijos požiūriu.

Įrodomo saugumo kriptosistema yra tokia, kurios šifro įveikimas yra prilygintas sudėtingos matematinės problemos išsprendimui. Pavyzdžiui, kriptosistema, kuri remiasi didelių skaičių išskaidymu pirminiais dauginamaisiais, yra įrodomo saugumo.

Skaičiavimų požiūriu saugi kriptosistema yra laikoma tokia, kuriai įveikti naudojant geriausiai žinomas atakas nepakaktų skaičiavimo resursų.

Euristiškai saugi kriptosistema yra tokia, kurios neįveikiamumas yra paremtas dažnai euristiniais argumentais.

1.3. Galimi atakų tipai

Kalbant apie kriptosistemos saugumą taip pat svarbu yra analizuoti galimus būdus pažeisti sistemos saugumą. Tokie bandymai dešifruoti užšifruotą pranešimą neturint privataus rakto yra

vadinami atakomis arba kriptoanalize [Sta02]. Dažniausiai tokių atakų vykdymas yra motyvuojamas noru sužinoti slaptą informaciją, tačiau ne retai kriptoanalizę galima naudoti kaip įrankį kriptosistemos saugumo tyrimui. Jei ataka prieš kriptosistemą nepavyko, dar negalima teigti, jog kriptosistema yra visiškai saugi. Galima išskirti kelis skirtingus atakų tipus, priklausomai nuo to kokia informacija atakos metu yra pasiekiama ir kokie metodai naudojami [Sta02]:

- pavienių šifrų ataka (angl. ciphertext-only attack)
- teksto-šifro porų ataka (angl. known-plaintext attack)
- pasirinktų teksto-šifro porų ataka (angl. chosen-plaintext attack)
- adaptyvia pasirinktų teksto-šifro porų ataka (angl. adaptive-chosen-plaintext attack)
- pasirinktų šifrų ataka (angl. chosen-ciphertext attack)

Pavienių šifrų atakos metu yra žinomi tik užšifruoti pranešimai. Šios atakos tikslas yra surasti arba pradinius pranešimus, kurie buvo užšifruoti, arba šifravimo raktą, kuriuo buvo šifruojama.

Teksto šifro porų atakos metu yra žinomi užšifruoti pranešimai bei juos atitinkantys pradiniai pranešimai. Dažniausiai ši ataka yra vykdoma po pavienių šifrų atakos, kuomet pavyko surasti nustatytą kiekį pradinių pranešimų. Atakos tikslas yra sužinoti dešifravimo algoritmą, kuris bet kokiam šifrui galėtų surasti jį atitinkantį pradinį pranešimą.

Pasirinktų teksto-šifro porų ataka yra atliekama, kuomet šifravimo sistema yra prieinama viešai, tačiau dešifravimo algoritmas nėra žinomas. Šiuo atveju pradinius pranešimus pasirenka pats kriptoanalitikas ir jiems, naudojantis prieinama šifravimo sistema, sugeneruoja atitinkamus užšifruotus pranešimus. Ši ataka dažniausiai yra vykdoma prieš viešo-privataus raktų tipo kriptosistemą, nes šioje sistemoje visiems yra prieinamas viešasis raktas ir bet kas gali pasirinktiems pranešimams sugeneruoti juos atitinkančius šifrus. Atakos tikslas – rasti dešifravimo algoritmą (arba privatų raktą).

Pasirinktų teksto-šifro porų ataką galima patobulinti po nustatyto kiekio bandymų keičiant naudojamus pradinius pranešimus. Tokia patobulinta ataka vadinama adaptyvia pasirinktų teksto-šifro porų ataka. Pavyzdžiui, tarkim pavyko nustatyti, jog pradiniuose pranešimuose esant pasikartojantiems simboliams, šifruose taip pat pasikartoja simboliai, todėl prasminga yra toliau bandyti tokio tipo pranešimus, nes galbūt pavyks atskleisti daugiau informacijos apie šifravimo algoritmą.

Pasirinktų šifrų ataka yra naudojama prieš autentiškumo įrodymui naudojamas kriptosistemas, kuomet yra žinomas dešifravimo raktas, bet nežinomas šifravimo raktas. Atakos tikslas – dešifruojant prieinamus šifruotus pranešimus gauti informacijos apie naudojamą šifravimo algoritmą.

2. McEliece ir Niederreiter kriptografinės sistemos

McEliece kriptosistema buvo sukurta 1978 metais Robert McEliece [Mce78]. Tai buvo pirmoji asimetrinio šifravimo kriptosistema naudojanti atsitiktinių skaičių generavimą šifravimo procese. Niederreiter kriptosistema yra patobulinta McEliece kriptosistemos variacija, sukurta 1986 metais Harald Niederreiter [Nie86]. Tiek McEliece, tiek Niederreiter kriptografinės sistemos veikimas yra pagrįstas klaidas taisančiais kodais, todėl yra svarbu aptarti šiose sistemose naudojamas klaidas taisančių kodų sąvokas ir veikimo principus.

2.1. Klaidas taisantys kodai

Klaidas taisančių kodų teorija (arba kodavimo teorija) nagrinėja praktinius metodus pranešimų kodavimui ir dekodavimui [Ske05]. Įprastai klaidas taisančių kodų paskirtis yra ne slėpti informaciją, bet apsaugoti ją nuo galimų iškraipymų siunčiant nepatikimu kanalu. Koduojant pranešimą prie jo yra pridedama papildoma informacija leidžianti gavėjui patikrinti, ar gautasis pranešimas atitinka siųstąjį, ir ištaisyti dalį klaidų atsiradusių dėl triukšmo. Vienas paprasčiausių būdų koduoti pranešimą galėtų būti kiekvieno pranešimo simbolio pakartojimas $n = 3$ kartus. Pavyzdžiui, pranešimą $m =$ „010“ galima koduoti kodo žodžiu $c =$ „000 111 000“. Jei gavėjui išsiųstas pranešimas buvo iškraipytas ir gautasis užkoduotas pranešimas yra $c' =$ „010 110 000“, tuomet jis, žinodamas kodavimo algoritmą, gali ištaisyti padarytas klaidas ir atkurti pradinį pranešimą m. Šis klaidas taisantis kodas geba ištaisyti $t = \left\lfloor \frac{n-1}{2} \right\rfloor$ klaidų kiekviename n ilgio bloke [Ske05].

Tegul F_q – baigtinis kūnas turintis q elementų. Generuojanti matrica G yra $k \times n$ matrica virš F_q, kurios eilutės sudaro kodo C bazę [Ske05]. Ši matrica generuoja $C[n, k]$ kodą virš F_q. Pavyzdžiui pranešimui $m =$ „010“ generuojanti matrica G galėtų būti:

$$G = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \end{pmatrix}. \quad (1)$$

Tuomet kodo žodį c apskaičiuoti galima dauginant pranešimą m su jo kodą generuojančia matrica G:

$$c = mG = (0\ 1\ 0)\begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \end{pmatrix} = (1\ 0\ 1\ 0). \quad (2)$$

2.2. McEliece kriptografinė sistema

McEliece kriptografinės sistemos veikimas yra pagrįstas trimis algoritmais: viešo ir privataus raktų generavimu, tikimybėmis paremtu šifravimu ir deterministiniu dešifravimu. Pradžioje yra nustatomas saugumo parametrų rinkinys (n, k, t), kur n yra siunčiamo šifro ilgis, k – pradinio pranešimo m ilgis, t – klaidų, kurias galima ištaisyti skaičius. Žinoma, $n > k$.

2.2.1. Raktų generavimas

McEliece kriptosistemos raktų generavimo metu yra sukuriami du raktų rinkiniai, privataus rakto rinkinys R_p ir viešojo rakto rinkinys R_v. Gavėjas pasidalina viešuoju raktu su visais norinčiais jam perduoti pranešimą, o privatų raktą laiko paslaptyje ir naudoja tik gautų pranešimų dešifravimui. Raktų generavimas vykdomas tokiu algoritmu [Til98]:

- Sukonstruojama $k \times n$ matrica G, generuojanti tiesinį Goppa kodą C, kuriame galima ištaisyti t klaidų.
- Sugeneruojama atsitiktinė $k \times k$ dvejetainė neišsigimusi matrica S.
- Sugeneruojama atsitiktinė $n \times n$ perstatos matrica P.
- Apskaičiuojama $G' = SGP$.
- Tuomet viešo rakto rinkinys $R_v = (G', t)$, privataus rakto rinkinys $R_p = (S, P, G)$.

2.2.2. Šifravimas

McEliece kriptosistemos pranešimo šifravimo metu, naudojantis viešuoju raktu R_v, dvejetainio formato pranešimas m yra užšifruojamas į šifrą c. Šifravimo vykdymo algoritmas [Til98]:

- Apskaičiuojamas vektorius $c' = mG'$.
- Sugeneruojamas atsitiktinis dvejetainis n ilgio klaidų vektorius z, turintis t vienetų.
- Apskaičiuojamas šifras $c = c' + z$.

2.2.3. Dešifravimas

Gavus šifrą c, jis yra dešifruojamas į pradinį pranešimą m naudojantis privačiu raktu R_p [Til98]:

- Apskaičiuojamas vektorius $\hat{c} = cP^{-1}$, lygus vektoriui $mSG + z'$, kur $z' = zP^{-1}$.
- Naudojantis Patterson dekodavimo algoritmu, randamas klaidų vektorius z' ir apskaičiuojamas vektorius m', kuris lygus vektoriui mS.
- Apskaičiuojamas pranešimas $m = m'S^{-1}$.

2.3. Niederreiter kriptografinė sistema

Kaip ir McEliece kriptografinėje sistemoje, Niederreiter kriptosistema yra pagrįsta trimis algoritmais: viešo ir privataus raktų generavimu, šifravimu ir dešifravimu. Pradžioje yra nustatomas saugumo parametrų rinkinys (n, k, t), kur n yra pradinio pranešimo m ilgis, $n - k$ yra siunčiamo šifro c ilgis, t – klaidų, kurias galima ištaisyti skaičius.

2.3.1. Raktų generavimas

Niederreiter kriptosistemos raktų generavimo metu yra sukuriami du raktų rinkiniai, privataus rakto rinkinys R_p ir viešojo rakto rinkinys R_v. Raktų generavimas vykdomas tokiu algoritmu [Til98]:

- Sukonstruojama $k \times n$ matrica G, generuojanti tiesinį Goppa kodą C, kuriame galima ištaisyti t klaidų.
- Sugeneruojama $(n - k) \times n$ kontrolinė matrica H, kodui C.
- Sugeneruojama atsitiktinė $(n - k) \times (n - k)$ dvejetainė neišsigimusi matrica S.
- Sugeneruojama atsitiktinė $n \times n$ perstatos matrica P.
- Apskaičiuojama $H' = SHP$.
- Tuomet viešo rakto rinkinys $R_v = (H', t)$, privataus rakto rinkinys $R_p = (S, P, H)$.

2.3.2. Šifravimas

Niederreiter kriptosistemos pranešimo šifravimo metu, naudojantis viešuoju raktu R_v, dvejetainio formato pranešimas m, kurio svoris yra ne daugiau negu t, yra užšifruojamas į šifrą c. Šifravimo vykdymo algoritmas [Til98]:

- Apskaičiuojamas šifras $c = H'm^T$.

2.3.3. Dešifravimas

Gavus šifrą c, jis yra dešifruojamas į pradinį pranešimą m naudojantis privačiu raktu R_p [Til98]:

- Apskaičiuojamas vektorius $\hat{c} = S^{-1}c$, lygus vektoriui HPm^T.
- Naudojantis sindrominiu dekodavimo algoritmu kodui C, randamas vektorius Pm^T.
- Apskaičiuojamas pranešimas $m = (P^{-1}Pm^T)^T$.

3. McEliece kriptosistemos sandaros įtaka saugumui

Kaip galima matyti iš McEliece kriptosistemos raktų generavimo ir šifravimo algoritmų, kriptosistemą saugumą užtikrina šie elementai: Goppa kodai, S ir P matricos, klaidos vektorius. Visi šie elementai, išskyrus klaidos vektorių, yra žinomi tik pranešimo gavėjui, tačiau nėra garantijos, jog šie elementai negali būti pažeisti perrinkimo ataka (angl. *brute-force attack*). Perrinkimo atakos metu yra išbandomi visi galimi nežinomojo kintamojo variantai, kol randamas tenkinantis variantas. Pavyzdžiui, McEliece kriptosistemos atveju galima būtų ieškoti privatų raktą atitinkančių matricų generuojant visas įmanomas šių matricų kombinacijas, kol jos tinkamai dešifruos pranešimą. Žemiau yra pateikiami McEliece kriptosistemos saugumo elementai, jų saugumo analizė ir skirtingų variantų skaičiai, kuriuos kompiuteriui tektų perrinkti atliekant perrinkimo ataką. Laikoma, kad McEliece saugumo parametrų rinkinys $(n, k, t) = (1024, 524, 50)$ [Mce78].

3.1. Goppa kodai

Dvejetainiai Goppa kodai naudojami McEliece kriptosistemoje yra klaidas taisantys kodai ir priklauso bendrai Goppa kodų klasei [Joc02]. Šifravimo metu McEliece kriptosistemoje generuojanti matrica G naudojama Goppa kodo C konstravimui, kuriame galima ištaisyti t klaidų. Dešifravimo metu Goppa kodo žodis $\hat{c}$ yra dekoduojamas naudojantis Patterson dekodavimo algoritmu ištaisant padarytas t klaidų.

Nors įprastinis klaidas taisančių kodų pritaikymas yra informacijos apsaugojimas nuo signalo iškraipymų, tačiau šiuos kodus galima pritaikyti ir kriptografijoje, jei generuojanti matrica G yra laikoma paslaptyje. Literatūroje laikoma, jog Goppa kodai yra saugūs plačiam taikymui praktikoje [Ber73].

3.2. S ir P matricos

McEliece kriptosistemoje matricos S ir P naudojamos šifro sumaišymui atsitiktiniu būdu. Žinant šias matricas ir viešąjį raktą $R_v = (G', t)$ nesunkiai galima būtų sužinoti ir Goppa kodą generuojančią matricą G:

$$G = S^{-1}G'P^{-1}. \tag{3}$$

Apskaičiavus Goppa kodą generuojančia matricą G, yra pasiekiamas pilnas privataus rakto rinkinys $R_p = (S, P, G)$ ir naudojantis aprašytu dešifravimo algoritmu, galima nesunkiai atkurti

pradinį pranešimą m. Tačiau šios matricos yra labai didelės ir jas atspėti perrinkimo atakos būdu yra nepaprastai sudėtinga. Kadangi S yra neišsigimusi $k \times k$ dimensijų matrica, skirtingų būdų ją sugeneruoti yra:

$$\prod_{i=0}^{k-1}(2^k - 2^i) = (2^{524} - 2^0) \times (2^{524} - 2^1) \times \ldots \times (2^{524} - 2^{523}) \approx 1.18 \times 10^{82655}.$$

(4)

Matricai P skirtingų būdų sugeneruoti yra kiek mažiau, nes vienetas matricos eilutėse ir stulpeliuose negali kartotis daugiau negu vieną kartą. Taigi skirtingų būdų sugeneruoti matricą P yra:

$$n! = 1024! \approx 5.42 \times 10^{2639}.$$

(5)

3.3. Klaidos vektorius

Klaidos vektorių z galima laikyti viena silpniausiųjų McEliece kriptosistemos vietų. Žinant viešąjį raktą $R_v = (G', t)$ ir klaidos vektorių z iš šifro c nesunkiai galima atkurti pradinį pranešimą m sprendžiant tiesinių lygčių sistemą:

$$c = mG' + z.$$

(6)

Kadangi klaidos vektorius z yra ilgio n ir sudarytas iš t vienetų, skirtingų būdų sugeneruoti šį vektorių yra:

$$\binom{n}{t} = \frac{n!}{t!\,(n-t)!} = \frac{1024!}{50! \times 974!} \approx 3.19 \times 10^{85}.$$

(7)

3.4. Silpnosios vietos

Kaip matyti silpniausias McEliece kriptosistemos elementas, perrinkimo atakos atžvilgiu, yra klaidos vektorius z, kadangi šio vektoriaus žinojimo užtenka šifro dešifravimui ir todėl, kad visoms jo kombinacijoms perrinkti reikia mažiausiai bandymų, lyginant su kitais kriptosistemos elementais. Tačiau taikyti perrinkimo ataką prieš McEliece kriptosistemą yra nepraktiška, nes net klaidos vektoriaus kombinacijų skaičius kompiuteriams yra per didelis, jog visas kombinacijas būtų įmanoma perrinkti per praktiškai trumpą laiką.

4. Atakų analizė

Atakų praktiniam įgyvendinimui buvo pasirinkta Java programavimo kalba dėl naudingų bibliotekų palaikymo bei patogios matricų ir vektorių objektų reprezentacijos. McEliece kriptosistemos raktų generavimui, šifravimui bei dešifravimui panaudota BouncyCastle biblioteka (kodas pasiekiamas viešai internetiniu adresu: https://github.com/bcgit/bc-java). Tiesinių lygčių sistemos sprendimas įgyvendintas remiantis Gauso metodu pritaikytu dvejetainei logikai. Programos kodas pasiekiamas internetiniu adresu: https://github.com/ingvaras/mceliece-cryptosystem-attack.

Parašytos programos tikslas yra išbandyti kelias skirtingas atakas prieš McEliece kriptosistemą ir palyginti atakų vykdymo laikus su skirtingais saugumo parametrais. Verta paminėti, jog panaudota McEliece kriptosistemos biblioteka vietoje saugumo parametrų rinkinio (n, k, t) naudoja rinkinį (m, t), kur $n = 2^m$, o $k = 2^m - m \times t$. Taigi, keičiant parametro t reikšmę keisis ir parametro k reikšmė, todėl nežymiai skirsis tiesinės lyčių sistemos sprendimo laikas skirtingoms t reikšmėms. Taip pat svarbu paminėti kompiuterio, kuriuo buvo atlikti bandymai, specifikacijas: procesorius: Intel Core i7-9850H @ 2.60 GHz; RAM: 32GB.

Taip pat verta paminėti, kad bandymai, kurie įprastai užtrunka iki 15 minučių, buvo atlikti pakartotinai po 10 kartų ir vykdymo trukmė paskaičiuota imant atakų vykdymo trukmės vidurkį. Taip buvo padaryta todėl, nes McEliece kriptosistemos šifravimo algoritmas priklauso nuo atsitiktinumo (klaidos vektoriaus z generavimas).

4.1. Perrinkimo ataka

Perrinkimo ataka yra dažniausiai naudojama, kuomet nėra žinomi jokie kriptosistemos privatūs parametrai, arba po to, kai pavyko sumažinti kriptosistemos saugumą kitomis atakomis. Perrinkimo atakos metu yra generuojamos kurio nors sistemos nežinomojo elemento kombinacijos, kuriomis paeiliui bandoma pažeisti sistemos saugumą. McEliece kriptosistemos atveju tai galėtų būti S ir P matricos, Goppa kodą C generuojantis polinomas, pradinis pranešimas m arba klaidos vektorius z. Kaip jau buvo parodyta analizuojant šios kriptosistemos sudedamuosius elementus, klaidos vektorius z yra laikomas silpniausia sistemos dalimi dėl mažiausio kiekio reikalingų kombinacijų šiam vektoriui sugeneruoti. Sugeneravus galimą klaidos vektoriaus kombinaciją z', sprendžiama tiesinė lygčių sistema:

$$G'^{T} m' = c - z'. \quad (8)$$

Ši lygčių sistema turės sprendinį tada ir tik tada, jei $z' = z$ [Joc02]. Jei lygčių sistema neturi sprendinio, galima generuoti kitą klaidos vektoriaus kombinaciją ir su ja spręsti lygčių sistemą.

4.2. Perrinkimo atakos įgyvendinimas

Perrinkimo atakos praktinio įgyvendinimo tikslas yra išbandyti kelias skirtingas klaidų, kurias galima ištaisyti, reikšmes t ir palyginti programos vykdymo laikus. Klaidos vektoriaus kombinacijų generavimas įgyvendintas Kenneth H. Rosen aprašytu kombinacijų generavimo metodu [Ros91].

Programa sugeneruoja viešojo rakto rinkinį $R_v = (G', t)$ pagal saugumo parametrų rinkinius $(m, t) = (6,\ t \in \{1, 2, \dots, 7\})$. Tuomet parinktas pranešimas m užšifruojamas į šifrą c McEliece kriptosistemos šifravimo algoritmu. Vėliau generuojamos n ilgio, t svorio dvejetainio vektoriaus z' kombinacijos. Su kiekviena vektoriaus z' kombinacija Gauso metodu sprendžiama tiesinė lygčių sistema (lygtis 8), kol randama lygčių sistema turinti sprendinį. Galiausiai, fiksuojama atakos vykdymo trukmė su tam tikra t reikšme. Atakų vykdymo trukmės pateikiamos lentelėje:

(n, k, t)	Klaidos vektoriaus z galimos kombinacijos	Vykdymo trukmė (s)
$(64, 58, 1)$	6.4×10^1	0.16
$(64, 52, 2)$	2.02×10^3	0.61
$(64, 46, 3)$	4.17×10^4	22.06
$(64, 40, 4)$	6.35×10^5	135.34
$(64, 34, 5)$	7.62×10^6	1245.15
$(64, 28, 6)$	7.50×10^7	3418.46
$(64, 22, 7)$	6.12×10^8	42714.09
$(1024, 1014, 1)$	1.02×10^3	1561.66

Iš gautų rezultatų matyti, jog didėjant klaidų, kurias galima ištaisyti, skaičiui t ilgėja ir programos vykdymo trukmė. Taip pat, naudojantis šia ataka, pavyko sėkmingai dešifruoti pranešimą su saugumo parametrais $(m, t) = (10,\ 1)$ per apytiksliai 0.5 valandos. Nors parametras $m = 10$ atitinka praktikoje naudojamą McEliece kriptosistemos saugumo parametrą, tačiau parametras $t = 1$ yra žymiai mažesnis už standartinį parametrą $t = 50$. Norint įveikti standartinio saugumo lygio McEliece sistemą su perrinkimo ataka tikėtina prireiktų perrinkti iki 3.19×10^{85}

(pagal 7 lygtį) klaidos vektoriaus kombinacijų. Tai, pagal gautus rezultatus, užtruktų iki $0.43h \times \frac{3.19\times10^{85}}{1.02\times10^{3}} = 1.34 \times 10^{82}$ valandų.

Taigi, nors McEliece kriptosistemos užšifruotą pranešimą neturint privataus rakto rinkinio R_p pavyko dešifruoti su saugumo parametrais $(n, k, t) = (64, 22, 7)$ per apytiksliai 12 valandų, tačiau šią ataką sėkmingai atlikti prieš įprastinio saugumo parametrų rinkinį $(n, k, t) = (1024, 524, 50)$ būtų praktiškai neįmanoma. Parašytą programą dar galima būtų optimizuoti išlygiagretinant lygčių sistemos sprendimo veiksmus su matrica taip, jog skaičiavimus optimaliai atliktų vaizdo plokštė.

4.3. Pranešimo persiuntimo ataka

Pranešimo persiuntimo ataka yra taikoma, kuomet kriptoanalitikas turi galimybę perimti kelis to pačio pradinio pranešimo m šifrus $c_1, c_2, \dots, c_i$. Perimti šifrai padeda sumažina klaidos vektoriaus z galimų kombinacijų skaičių. Šiame darbe bus nagrinėjamas atvejis, kai ataką vykdančiajam pavyko perrimti $i = 2$ šifrus c_1, c_2.

Pranešimo persiuntimo atakos idėja yra, kad šifrai c_1, c_2 suteikia informacijos apie klaidos vektorius z_1, z_2:

$$c_1 - c_2 = mG' + z_1 - (mG' + z_2) = z_1 - z_2. \quad (9)$$

Šią informaciją galima pritaikyti nustatinėjant šifrų koordinates, kurios tikėtina nebuvo iškraipytos klaidos vektoriaus. Pagal gautą šifrų skirtumą, sudarinėjamas vektorius L [Ber97]:

$$L = \{\, l \in [1; n] : c_1(l) - c_2(l) = 0 \,\}. \quad (10)$$

Šiame vektoriuje yra šifrų c_1, c_2, koordinatės, kuriose šifrai sutapo. Šifrai koordinatėje sutampa, jei:

- abu šifrai buvo iškraipyti klaidos vektoriaus toje koordinatėje. Šio įvykio tikimybė yra $\left(\frac{t}{n}\right)^2$ [Ber97].
- abu šifrai buvo neiškraipyti klaidos vektoriaus toje koordinatėje. Šio įvykio tikimybė yra $1 - \left(\frac{t}{n}\right)^2$.

Akivaizdu, kad standartinio saugumo McEliece kriptosistemoje, pirmojo įvykio tikimybė bus labai maža (~ 0.0024). Remiantis tuo, galima teigti, kad šiuo atveju vektoriuje L didžioji dalis esančių šifrų koordinačių tikėtina nebuvo iškraipytos klaidos vektoriaus.

Tuomet iš vektoriaus L yra parenkamos k atsitiktinių koordinačių ir suformuojama matrica G'' pagal šias atsitiktines koordinates parenkant k stulpelių iš G'. Gautoji G'' yra $k \times k$ dimensijų matrica. Tokiu pačiu principu, pagal tas pačias k atsitiktinių koordinačių, yra parenkama k elementų iš šifro vektoriaus c_1 ir suformuojamas vektorius c_1', turintis lygiai k elementų. Galiausiai, bandoma išspręsti tiesinę lygčių sistemą:

$$G''^{T} m' = c_1'. \quad (11)$$

Jeigu parinktos k atsitiktinės koordinatės nebuvo iškraipytos klaidos vektoriaus, tuomet išsprendus lygčių sistemą, bus gautas tikrasis pradinis pranešimas m.

Tarkime, taikant pranešimo persiuntimą ataką prieš standartinio saugumo McEliece kriptosistemą, buvo gautas vektorius L su 930 koordinačių, kuriose buvo padaryti tik $t - \frac{n-|L|}{2} = 3$ iškraipymai. Tuomet, tikimybė atkurti pradinį pranešimą m, parenkant 524 atsitiktinių koordinačių iš L, bus [Ber97]:

$$\frac{\binom{927}{524}}{\binom{930}{524}} \approx 0.0828 \quad (12)$$

Vadinasi, vidutiniškai reikės apie 12 spėjimų, norit atkurti pradinį pranešimą.

4.4. Pranešimo persiuntimo atakos įgyvendinimas

Pranešimo persiuntimo atakos praktinio įgyvendinimo tikslas yra išbandyti kelias skirtingas klaidų, kurias galima ištaisyti, reikšmes t ir palyginti atakos vykdymo laikus. Tiesinis lygčių sprendimas įgyvendintas naudojantis Gauso metodu.

Kaip ir perrinkimo atakos atveju programa sugeneruoja viešojo rakto rinkinį $R_v = (G', t)$ pagal saugumo parametrų rinkinius $(m, t) = (6,\ t \in \{1, 2, \dots, 7\})$. Tuomet parinktas pranešimas m užšifruojamas pakartotinai du kartus į šifrus c_1 ir c_2 McEliece kriptosistemos šifravimo algoritmu. Pranešimo persiuntimo atakos teorinėje analizėje aprašytu metodu generuojamas vektorius L ir iš jo atsitiktiniu būdu parenkamos k koordinatės. Su kiekvienu spėjimu sprendžiama tiesinė lygčių sistema (lygtis 11), kol lygčių sistemos sprendinys yra lygus m. Galiausiai, fiksuojama atakos vykdymo trukmė su tam tikra t reikšme. Kadangi įgyvendinta pranešimo persiuntimo ataka su nurodytais saugumo parametrų rinkiniais sėkmingai dešifravo pranešimą per pakankamai trumpą laiką, buvo nuspręsta papildomai atlikti bandymą su $(m, t) = (10,\ 50)$. Atakų vykdymo trukmės pateikiamos lentelėje:

(n, k, t)	Vykdymo trukmė (s)
$(64, 58, 1)$	0.01
$(64, 52, 2)$	0.01
$(64, 46, 3)$	0.01
$(64, 40, 4)$	0.01
$(64, 34, 5)$	0.01
$(64, 28, 6)$	0.01
$(64, 22, 7)$	0.01
$(1024, 524, 50)$	9.24

Iš gautų rezultatų matyti, jog taikant pranešimo persiuntimo ataką, pavyko įveikti standartinio saugumo McEliece kriptosistemą per apytiksliai 9 sekundes. Taigi, turint du to paties pranešimo šifrus, praktiškai nėra sudėtinga atkurti pradinį pranešimą taikant pranešimo persiuntimo ataką.

4.5. Žinomo dalinio teksto ataka

Kaip jau buvo minėta prieš tai, kriptosistemos saugumas priklauso ir nuo to, kokia informacija yra prieinama ją atakuojančiajam kriptoanalitikui. Turima omenyje, jog ši papildoma informacija sumažina reikalingų perrinkimų skaičių bandant dešifruoti pranešimą neturint privataus rakto rinkinio. Viena iš tokio pobūdžio atakų, kurią galima panaudoti prieš McEliece kriptosistemą, yra žinomo dalinio teksto ataka.

Žinomo dalinio teksto ataka yra naudojama, kuomet kriptoanalitikui yra žinoma dalis pradinio pranešimo m ir pilną pranešimą atitinkantis šifras c. Vienas gerai žinomas pavyzdys, kur šio tipo ataka buvo sėkmingai pritaikyta, yra bandymas dešifruoti antrojo pasaulinio karo vokiečių naudotos šifravimo mašinos „Enigma“ šifruotus pranešimus. Manoma, kad kriptoanalitikų komandai vykdžiusiai šią ataką buvo prieinama informacija apie tai, kurie šifrai atitinka orų prognozes, ir buvo žinoma, kad šiose prognozėse visuomet yra žodis „Wetter“ (liet. *oras*), bei šio žodžio pozicija pradiniame pranešime [Lee91]. Kadangi „Enigma“ naudojamas šifravimo algoritmas buvo keičiamas tik kas dieną, dešifravus vieną pranešimą, galima buvo nesunkiai dešifruoti visus tą dieną siųstus pranešimus. Nors McEliece kriptosistema neturi tokio pažeidžiamumo, nes dažniausiai klaidos vektorius yra keičiamas po kiekvieno šifravimo, tačiau šio tipo ataką vis vien galima pritaikyti prieš šią kriptosistemą.

Tarkime, yra žinoma šifras c, kuris atitinka pranešimą m, kurio ilgis yra k. Taip pat yra žinoma dalis pradinio pranešimo m_r, kurio ilgis k_r. Nežinoma pranešimo dalis yra m_l, kurios ilgis k_l. Taigi pranešimas m susideda iš dviejų dalių:

$$m = (m_l || m_r),$$

(13)

ir pranešimo m ilgį galima išreikšti kaip pranešimo dalių ilgių sumą:

$$k = k_l + k_r.$$

(14)

Turint šią informaciją galima supaprastinti dešifravimo uždavinį [KI01]:

$$c = mG' + z = m_l G'_l + m_r G'_r + z$$

$$c' = m_l G'_l + z,$$

(15)

kur G'_l yra pirmosios k_l matricos G' eilučių, o G'_r yra paskutinės k_r matricos G' eilutės.

Pritaikius tokią dešifravimo uždavinio transformaciją, kriptosistemos saugumo parametrų rinkinys dabar yra lygus (n, k_l, t). Akivaizdu, jog $k_l < k$. Nors laikoma, kad kriptosistemos silpniausioji vieta yra klaidos vektorius z, tačiau sumažėjęs saugumo parametras k neturi įtakos šio vektoriaus sugeneravimo kombinacijų skaičiui. Sumažėja tik matricos S ir generuojančios matricos G galimų kombinacijų skaičiai. Jei žinoma pradinio pranešimo dalis buvo pakankamai ilga, kad šifro c' dešifravimas su turimais skaičiavimų resursais būtų įveikiamas per praktiškai trumpą laiką, tuomet galima naudoti perrinkimo ataką arba bandyti dar labiau supaprastinti dešifravimo uždavinį kitokiomis atakomis.

4.6. Žinomo dalinio teksto atakos įgyvendinimas

Įgyvendintoje programoje žinomo dalinio teksto ataka buvo panaudota McEliece kriptosistemos saugumo parametrų supaprastinimui. Po to pranešimas buvo dešifruojamas pasinaudojant perrinkimo ataka. Nors žinomo dalinio teksto ataka nesumažina klaidos vektoriaus galimų kombinacijų skaičiaus, tačiau tiesinių lygčių sprendimas supaprastėja dėl sumažėjusių matricos G' dimensijų.

Kaip ir kitų aprašytų atakų atveju programa sugeneruoja viešojo rakto rinkinį $R_v = (G', t)$ pagal saugumo parametrų rinkinius $(m, t) = (6,\ t \in \{1, 2, \dots, 7\})$. Tuomet parinktas pranešimas m užšifruojamas į šifrą c McEliece kriptosistemos šifravimo algoritmu. Toliau kiekvienas bandymas yra atliekamas po kartą su skirtingomis žinomo pranešimo dalimis: 25%, 50% ir 75% pradinio pranešimo ilgio. Atakos vykdymo metu žinoma pranešima dalis yra naudojama

supaprastinti tiesinių lygčių sprendimo uždavinį pagal 15 lygtį. Šifro klaidos vektoriaus generavimui ir lygčių sprendimui yra naudojami perrinkimo atakos analizėje aprašyti metodai. Atakų vykdymo trukmės pateikiamos lentelėje:

(n, k, t)	Klaidos vektoriaus z galimos kombinacijos	Vykdymo trukmė (s), kai $\frac{k_r}{k} = 0.25$	Vykdymo trukmė (s), kai $\frac{k_r}{k} = 0.5$	Vykdymo trukmė (s), kai $\frac{k_r}{k} = 0.75$
$(64, 58, 1)$	6.4×10^1	0.04	0.02	0.01
$(64, 52, 2)$	2.02×10^3	0.60	0.23	0.06
$(64, 46, 3)$	4.17×10^4	13.08	5.99	1.89
$(64, 40, 4)$	6.35×10^5	56.48	27.45	8.83
$(64, 34, 5)$	7.62×10^6	855.98	423.96	127.88
$(64, 28, 6)$	7.50×10^7	2695.15	1435.90	650.36
$(64, 22, 7)$	6.12×10^8	31255.89	10012.15	3561.56

Iš gautų rezultatų matyti, kad kuo didesnė pradinio pranešimo dalis yra žinoma, tuo greičiau pranešimas yra dešifruojamas. Taikant praktikoje žinomo dalinio teksto ataką, žinant 75% pradinio pranešimo, kartu su perrinkimo ataka prieš standartinio saugumo McEliece kriptosistemą, pranešimo dešifravimas užtruktų kelis ar keliolika kartų trumpiau nei taikant tik perrinkimo ataką.

Norint įvertinti šios atakos pajėgumą prieš standartinio saugumo lygio McEliece kriptosistemą, buvo atliktas papildomas bandymas su parametrais $(n, k, t) = (1024, 1004, 2)$ ir 99.9% žinomo pradinio pranešimo dalimi. Tokią sistemą programai įveikti užtruko 24.12 sekundės ir iki $\binom{1024}{2} \approx 5.24 \times 10^5$ perrinkimų. Pagal tai, galima būtų įvertinti, kad standartinio saugumo McEliece kriptosistemą įveikti žinant 99.9% pranešimo užtruktų iki $24.12 \frac{3.19 \times 10^{85}}{5.24 \times 10^5} \approx 1.47 \times 10^{81}$ sekundžių arba iki 4.08×10^{77} valandų. Nors tai yra apie 10^5 kartų greitesnis vykdymo laikas nei perrinkimo atakos atveju, tačiau ataka vis tiek užtruktų praktiškai per ilgai. Žinoma, nagrinėtu atveju yra paprasčiau atspėti nežinomą pranešimo dalį, jei yra būdas patikrinti, ar atspėtas pranešimas yra lygus pradiniam pranešimui.

4.7. Gavėjo atsako ataka

Gavėjo atsako ataka yra naudojama, kuomet kriptoanalitikas turi galimybę patikrinti, ar tam tikras šifras c' yra dešifruojamas į tą patį pradinį pranešimą m, kaip ir šifras c. Ši ataka susideda iš dviejų žingsnių: šifro c', kuriame buvo padarytos $t + 1$ klaidos, radimo ir klaidos vektoriaus radimo [HGS97].

Pirmuoju žingsniu, skaičiuojami vektoriai c_i' (kai $i \in [1;\ 2t+1]$):

$$c_i' = c + [a_1 \quad a_2 \quad \dots \quad a_i \quad a_{i+1} \quad a_{i+2} \quad \dots \quad a_n],$$

(16)

kur $a_{1\dots i} = 1$, o $a_{i+1\dots n} = 0$. Kiekvienos iteracijos metu tikrinama, ar šifras c_i' dešifruojasi į m. Jei šifrą c_i' pavyksta dešifruoti, vadinasi vektorių sudėties operacijos metu buvo ištaisyta daugiau arba tiek pat klaidų, kiek naujų klaidų buvo privelta, todėl šis šifras turi t arba mažiau klaidų. Jei šifro nepavyksta dešifruoti į m, vadinasi jis turi $t+1$ klaidą. Šis šifras pavadinamas c' ir daugiau iteracijų nebevykdoma.

Antruoju žingsniu, vektoriuje c' paeiliui atliekant vieno bito modifikacijas, tikrinami kurie vektoriai c_i'' (kai $i \in [1;\ n]$) dešifruojasi į m, kai

$$c_i'' = c' + [0 \quad 0 \quad \dots \quad a_i \quad 0 \quad \dots \quad 0],$$

(17)

kur a_i yra i-tasis vektoriaus elementas ir $a_i = 1$. Kiekvienos sudėties operacijos metu yra arba ištaisoma viena klaida (tuomet c_i'' turės lygiai t klaidų ir bus dešifruojamas į m) arba padaroma viena klaida (tuomet c_i'' turės lygiai $t+2$ klaidas ir bus dešifruojamas ne į m). Atlikus šiuos patikrinimus formuojamas klaidos vektorius z':

$$z_i' = \begin{cases} 1, & jei\ c_i''\ de\check{s}ifruojasi\ \text{į}\ m \\ 0, & jei\ c_i''\ de\check{s}ifruojasi\ ne\ \text{į}\ m \end{cases},$$

(18)

kur z_i' yra i-tasis z' elementas.

Galiausiai, turint šifrą c' ir jo klaidos vektorių z', spendžiama tiesinė lygčių sistema (8 lygtis) ir randamas pradinis pranešimas m. Taigi, gavėjo atsako ataka yra pavojinga dėl palyginus nedidelio c_i' ir c_i'' visų galimų kombinacijų skaičiaus $n + 2t + 1$. Net ir naudojant šią ataką prieš standartinio saugumo McEliece kriptosistemą, blogiausiu atveju, užtektų padaryti tik 1075 patikrinimus. Tiesa, kiekvienas patikrinimas, ar šifras dešifruojasi į m, gali užtrukti nemažai papildomo laiko priklausomai nuo komunikacijos kanalo ir gavėjo atsakymo greičio.

Gali atrodyti, kad ši ataka praktikoje nėra pritaikoma, nes jos sėkmė priklauso nuo to, ar pranešimo gavėjas pateiks informaciją kriptoanalitikui apie jam atsiųstus šifrus. Tačiau, gavėjas šią informaciją gali pateikti nesąmoningai, pats to nesuprasdamas. Tarkime, kriptoanalitikas gavo prieigą prie duomenų bazės, kurioje yra saugomi tam tikros sistemos prisijungimo duomenys užšifruoti McEliece šifravimo algoritmu. Tuomet, kriptoanalitikas duomenų bazėje galėtų pakeisti pasirinkto vartotojo slaptažodžio šifrą į bet kokį kitą šifrą. Kai vartotojas bandytų prisijungti prie sistemos, jam pavyktų arba ne, priklausomai nuo to, ar kriptoanalitiko slaptažodžio šifras dešifruojasi į tikrąjį vartotojo slaptažodį. Jei atakuojamam vartotojui pavyko prisijungti prie

sistemos, kriptoanalitikas sužinotų, kad jo šifras dešifruojasi į tą patį slaptažodį kaip ir originalus šifras buvęs duomenų bazėje. Tuomet, kriptoanalitikas, naudodamasis aprašytu gavėjo atsako atakos algoritmu, galėtų išsiaiškinti vartotojo slaptažodį.

4.8. Gavėjo atsako atakos įgyvendinimas

Gavėjo atsako atakos praktinio įgyvendinimo tikslas yra išbandyti kelias skirtingas klaidų, kurias galima ištaisyti, reikšmes t ir palyginti programos vykdymo laikus. Kadangi atakos teorinėje analizėje buvo nustatyta, jog šios atakos įgyvendinimas reikalauja nedidelio kiekio perrinkimų, buvo nuspręsta išbandyti šią ataką prieš standartinio saugumo McEliece kriptosistemą. Verta paminėti, kad parašytoje programoje gavėjo atsakas neturi jokio delsos, t.y. programa pati patikrina, ar šifras dešifruojasi į pradinį pranešimą.

Programa sugeneruoja viešojo rakto rinkinį $R_v = (G', t)$ pagal saugumo parametrų rinkinius $(m, t) = (6,\ t\ \epsilon\ \{1, 2, \dots, 7\})$. Tuomet parinktas pranešimas m užšifruojamas į šifrą c McEliece kriptosistemos šifravimo algoritmu. Vėliau teorinėje analizėje aprašytais metodais generuojami šifrai c_i'. Tuomet tikrinama ar šie šifrai nesidešifruoja į m, kol randamas c'. Radus c', generuojami c_i'' ir, pagal tai, ar jie dešifruojasi į m, formuojamas klaidos vektorius z'. Turint klaidos vektorių išsprendžiama tiesinė lygčių sistema Gauso metodu ir gaunamas pradinis pranešimas m. Galiausiai, fiksuojama atakos vykdymo trukmė su tam tikra t reikšme. Atakų vykdymo trukmės pateikiamos lentelėje:

(n, k, t)	c_i' ir c_i'' visų kombinacijų skaičius	Vykdymo trukmė (s)
$(64, 58, 1)$	6.7×10^1	0.01
$(64, 52, 2)$	6.9×10^1	0.01
$(64, 46, 3)$	7.1×10^1	0.01
$(64, 40, 4)$	7.3×10^1	0.01
$(64, 34, 5)$	7.5×10^1	0.01
$(64, 28, 6)$	7.7×10^1	0.01
$(64, 22, 7)$	7.9×10^1	0.01
$(1024, 524, 50)$	1.08×10^3	4.12

Iš gautų rezultatų matyti, kad gavėjo atsako ataka, lyginant su kitomis tirtomis atakomis, yra labai pavojinga – pranešimą, užšifruotą su standartinio saugumo McEliece kriptosistema, pavyko dešifruoti per apytiksliai 4 sekundes. Verta paminėti, kad realybėje ši ataka nebūtų tokia greita,

nes komunikacija tarp gavėjo ir kriptoanalitiko kiekvieno šifro tikrinimo metu užtruktų papildomai laiko, tačiau vis tiek būtų įvykdoma per praktiškai trumpą laiką.

5. Atakų palyginimas ir rekomendacijos

Atlikus praktinius bandymus su atakomis, iš gautų rezultatų galima matyti, kad vienos atakos yra pavojingesnės už kitas. Žinoma, kiekvienos atakos rezultatas priklauso nuo kriptoanalitikui pasiekiamos informacijos kiekio ir pobūdžio, todėl lyginti atakų rezultatus tarpusavyje nėra prasmės. Tačiau praktiškai tam tikras atakas yra lengviau pritaikyti ir panaudoti nei kitas. Pagal tai buvo sudarytas tirtų atakų prieš McEliece kriptosistemą pavojingumo sąrašas:

1. Perrinkimo ataka. Nors šią ataką galima pritaikyti neturint jokios papildomos informacijos apie šifrą ar pradinį pranešimą, ši ataka yra praktiškai nepanaudojama prieš standartinio saugumo McEliece kriptosistemą dėl atakai reikalingų milžiniškų skaičiavimo ir laiko resursų.
2. Žinomo dalinio teksto ataka. Ši ataka remiasi tokiu pat principu, kaip ir perrinkimo ataka, tik skaičiavimų laikas yra trumpesnis iki kelių šimtų tūkstančių kartų lyginant su perrinkimo ataka, priklausomai nuo žinomo pradinio pranešimo dydžio. To praktikoje vis tiek nepakanka norint pažeisti standartinio saugumo McEliece kriptosistemą. Tačiau ši ataka turi privalumą, kad ją galima taikyti su bet kuria kita ataka norint palengvinti tiesinių lygčių sprendimo uždavinį.
3. Pranešimo persiuntimo ataka. Ištyrus šią ataką, nustatyta, kad žinant tik du šifrus, pranešimo persiuntimo atakos vykdymas užtrunka vos kelias sekundes. Praktiškai ataka yra pritaikoma tik jei kriptoanalitikas turi galimybę perrimti du to paties pradinio pranešimo šifrus.
4. Gavėjo atsako ataka. Nustatyta, kad ši ataka yra tokia pat pavojinga kaip ir pranešimo persiuntimo ataka, nes šios atakos vykdymo laikas tiesiškai priklauso nuo n ir t saugumo parametrų, t.y. atakos vykdymas užtrunka vos kelias sekundes su standartiniais McEliece saugumo parametrais. Praktiškai ataka yra pritaikoma tik radus sistemose, naudojančiose McEliece kriptosistemą, tam tikrų spragų arba pasinaudojant gavėjo neatsargumu.

Pagal nustatytus atakų privalumus ir trūkumus buvo sudarytos rekomendacijos, kaip galima būtų išvengti silpnų McEliece kriptosistemos vietų bei pagerinti sistemos atsparumą prieš šias atakas:

1. Perrinkimo ataka. Nors ši ataka yra nepraktiška, tačiau vis tiek galėtų būti panaudoti prie silpnesnių saugumo parametrų McEliece kriptosistemą. Norint apsisaugoti prieš šią ataką rekomenduojama naudoti standartinio saugumo McEliece kriptosistemą, nes šia sistema užšifruotą pranešimą dešifruoti kriptoanalitikui gali užtrukti apie 10^{82} valandų.

2. Žinomo dalinio teksto ataka. Šiai atakai galima taikyti tokią pat rekomendaciją kaip ir perrinkimo atakai. Papildomai rekomenduojama neatskleisti jokio informacijos apie pradinį pranešimo turinį, net jei atskleista informacijos dalis yra nereikšminga.
3. Pranešimo persiuntimo ataka. Norint apsisaugoti prieš šią ataką rekomenduojama nešifruoti to paties pranešimo daugiau nei vieną kartą. Jeigu nepavyko išsiųsti pranešimo gavėjui dėl kanalo trikdžių, rekomenduojama šifruoti pranešimą kitomis kriptosistemomis.
4. Gavėjo atsako ataka. Kadangi atakos vykdymo laikas turi tiesinę priklausomybę nuo saugumo parametrų verčių, jų didinimas nuo gavėjo atsako atakos apsisaugoti nepagelbės. Vienintelis būdas apsisaugoti nuo šios atakos, neatsakyti siuntėjui apie siunčiamus šifrus, jei tas pats pranešimas yra siunčiamas kelis kartus.

Rezultatai ir išvados

Darbe buvo analizuojama McEliece kriptosistema ir jos saugumas. Pirmiausia, buvo nagrinėjami bendri kriptografijos uždaviniai, saugumo lygiai bei galimi atakų tipai prieš kriptosistemas. Taip pat apibrėžtas viešojo ir privataus raktų veikimo principas kriptosistemose.

Didelis dėmesys buvo skirtas McEliece kriptosistemos sandarai analizuoti. Pirmiausia, buvo aptarti klaidas taisantys kodai, kurie ir yra pritaikyti McEliece kriptosistemoje. Vėliau buvo analizuojama kriptosistemos sandara bei veikimo principas: raktų generavimas, šifravimo bei dešifravimo algoritmai. Tiriant kriptosistemos sandarą, buvo išskirti šie pagrindiniai McEliece kriptosistemos saugumą užtikrinantys sudedamieji elementai: Goppa kodai, S ir P matricos, klaidos vektorius z. Remiantis šifravimo ir dešifravimo algoritmais, buvo nagrinėjama kiekvieno elemento saugumas perrinkimo atakos atžvilgiu. Paaiškėjo, kad klaidos vektorius z yra silpniausioji kriptosistemos vieta. Tačiau reikalingas perrinkimų skaičius visoms klaidos vektoriaus kombinacijoms perrinkti per praktiškai trumpą laiką vis tiek yra per didelis šių laikų kompiuteriams.

Galiausiai, išanalizuotos perrinkimo, pranešimo persiuntimo, žinomo dalinio teksto ir gavėjo atsako atakos ir jų veiksmingumas prieš McEliece kriptosistemą. Naudojantis praktiškai įgyvendintomis atakomis buvo atlikti bandymai dešifruoti McEliece šifravimo algoritmu užšifruotus pranešimus su įvairiais saugumo parametrų rinkiniais. Nustatyta, kad perrinkimo ir žinomo dalinio teksto atakos yra praktiškai nepanaudojamas prieš standartinio saugumo McEliece kriptosistemą. Lyginant su pastarosiomis, pranešimo persiuntimo ataka turėjo geresnių rezultatų – turint vos du to pačio pradinio pranešimo šifrus, pranešimą pavyko dešifruoti per kelias sekundes. Nustatyta, kad gavėjo atsako ataka yra tokia pat pavojinga, kaip ir pranešimo persiuntimo ataka prieš McEliece sistemą, nes atakos vykdymo laikas turi tiesinę priklausomybę nuo McEliece saugumo parametrų verčių. Pagal gautus rezultatas darbe buvo pateiktos rekomendacijos, kaip galima būtų sustiprinti sistemos saugumą prieš konkrečias atakas.

Šaltiniai

[Ber73] Elwyn R. Berlekamp, Goppa Codes, IEEE Transactions on information theory, Vol. IT-19, No. 5, September 1973. Prieiga per internetą: https://web.archive.org/web/20170829142555/http://infosec.seu.edu.cn/space/kangwei/senior_thesis/Goppa.pdf. Žiūrėta 2022-01-12.

[Ber97] Thomas A. Berson. Failure of the McEliece Public-Key Cryptosystem under Message-Resend and Related-Message Attack. Burton S. Kaliski Jr. (Ed.), Advances in Cryptology - CRYPTO '97, Proceedings of 17th Annual International Cryptology Conference, Santa Barbara, California, USA, August 1997, Lecture Notes in Computer Science 1294, p. 213-220, 1997. Prieiga per internetą: http://www.anagram.com/berson/mceliece.pdf. Žiūrėta 2022-5-29.

[DMR11] Dinh, Hang; Moore, Cristopher; Russell, Alexander (2011). Rogaway, Philip (ed.). McEliece and Niederreiter cryptosystems that resist quantum Fourier sampling attacks. Advances in cryptology—CRYPTO 2011. Lecture Notes in Computer Science. 6841. Heidelberg: Springer. pp. 761–779.

[HGS97] Chris Hall, Ian Goldberg, Bruce Schneier. Reaction Attacks Against Several Public-Key Cryptosystems. Prieiga per internetą: http://www.cypherpunks.ca/~iang/pubs/paper-reaction-attacks.pdf. Žiūrėta: 2022-05-19

[Joc02] Ellen Jochemsz. Goppa Codes & the McEliece Cryptosystem. Masters thesis, Vrije Universiteit Amsterdam (2002).

[KI01] Kazukuni Kobara ir Hideki Imai. Semantically Secure McEliece Public-Key cryptosystems-conversions for McEliece PKC. International Workshop on Public Key Cryptography, p.p. 22-23, 2001. Prieiga per internetą: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.92.8099&rep=rep1&type=pdf. Žiūrėta 2022-01-15.

[Lee91] Lee, Loyd E. (1991). World War II: Crucible of the Contemporary World: Commentary and Readings. New York: M. E. Sharpe. pp: 240.

[Mce78] Robert J Mceliece. A Public-Key Cryptosystem Based On Algebraic Coding Theory. JPL DSN Progress Report 42-44, pp. 114-116, April 15, 1978. Prieiga per internetą: http://tmo.jpl.nasa.gov/progress_report2/42-44/44N.PDF. Žiūrėta 2021-12-16.

[MOV96] Alfred J. Menezes, Paul C. van Oorschot and Scott A. Vanstone. Handbook of Applied Cryptography. CRC Press, October 1996, 816 pages. Prieiga per internetą: http://www.cacr.math.uwaterloo.ca/hac/. Žiūrėta 2021-12-10.

[Nie86] H. Niederreiter. "Knapsack-type cryptosystems and algebraic coding theory". Problems of Control and Information Theory. Problemy Upravlenija I Teorii Informacii, 1986, pp: 159–166.

[Ros91] Kenneth H. Rosen, *Discrete Mathematics and Its Applications*, 2nd edition (NY: McGraw-Hill, 1991), pp. 284-286

[Ske05] Gintaras Skersys. Klaidas taisančių kodų teorija. Paskaitų konspektai, 2005. Prieiga per internetą: http://www.mif.vu.lt/katedros/cs/Asmen/KodavimoTeorija.pdf. Žiūrėta 2021-11-03.

[Sta02] V. Stakėnas. Kriptologija. Paskaitų konspektas, 2002.

[Til98] Henk C. A. van Tilborg. Coding theory at work in cryptology and vice versa. In: V.S. Pless and W.C. Huffman (Eds.), Handbook of Coding Theory, ch. 14, Elsevier, 1998, pp. 1195-1227.