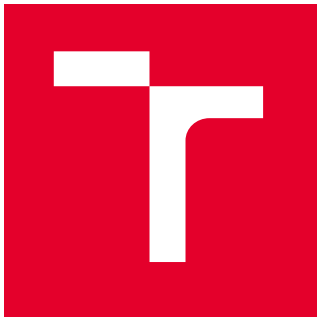


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV TELEKOMUNIKACÍ

DEPARTMENT OF TELECOMMUNICATIONS

MODERNÍ KRYPTOGRAFICKÉ PROTOKOLY S OCHRANOU SOUKROMÍ

MODERN PRIVACY-PRESERVING CRYPTOGRAPHY PROTOCOLS

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE — **Bc. Pavla Hlučková**

AUTHOR

VEDOUCÍ PRÁCE — **doc. Ing. Lukáš Malina, Ph.D.**

SUPERVISOR

BRNO 2022

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

Diplomová práce

magisterský navazující studijní program **Informační bezpečnost**

Ústav telekomunikací

Studentka: Bc. Pavla Hlučková ***ID:*** 203171

Ročník: 2 ***Akademický rok:*** 2021/22

NÁZEV TÉMATU:

Moderní kryptografické protokoly s ochranou soukromí

POKYNY PRO VYPRACOVÁNÍ:

Seznamte se s metodami kryptografie, které zajišťují ochranu soukromí. Zaměřte se především na schémata, jako jsou skupinové podpisy a atributové autentizační schémata postavené na problémech, které odolávají i analýze pomocí kvantového počítače. Analyzujte existující metody a zhodnoťte jejich praktickou použitelnost a porovnejte jejich paměťovou a výpočetní náročnost mezi sebou a s klasickými schématy skupinových podpisů a atributových schémat. Připravte verifikační implementaci vybraných metod a operací ve zvoleném programovatelném jazyce.
Hlavním cílem diplomové práce bude funkční implementace bezpečné kryptografické metody, která poskytuje i ochranu soukromí a reálné zhodnocení výpočetní a paměťové náročnosti na vybrané výpočetní platformě.

DOPORUČENÁ LITERATURA:

[1] DOMINGO-FERRER, J. a BLANCO-JUSTICIA, A., Privacy-Preserving Technologies. The Ethics of Cybersecurity Springer, Cham. 279-297. 2020.

[2] CHEN, Lily, et al. Report on post-quantum cryptography. US Department of Commerce, National Institute of Standards and Technology, 2016.

Termín zadání: 7.2.2022 ***Termín odevzdání:*** 24.5.2022

Vedoucí práce: doc. Ing. Lukáš Malina, Ph.D.

doc. Ing. Jan Hajný, Ph.D.
předseda rady studijního programu

ABSTRAKT

Tato práce se zabývá průnikem dvou moderních a rozrůstajících se odvětví kryptografie, a to technologií umožňujících ochranu soukromí a postkvantové kryptografie. Obecně popisuje vybraná schémata zvyšující ochranu soukromí (PETs) a rodiny postkvantové kryptografie. Podrobněji je práce zaměřena na skupinové podpisy využívající matematických problémů odolných vůči hrozbě kvantových počítačů. Práce dále shrnuje stav techniky a porovnává efektivitu těchto schémat na základě dostupných informací. Hlavní částí práce je implementace skupinového podpisu založeného na hash funkcích a jeho reálné porovnání oproti implementacím skupinových podpisů založeným na mřížkách a teorii kódování, které byly získány od vědeckých týmů aktivních v tomto oboru. Tyto postkvantové skupinové podpisy jsou také porovnány vůči klasickým schématům skupinových podpisů implementovaným pomocí knihovny libgroupsig.

KLÍČOVÁ SLOVA

technologie zvyšující ochranu soukromí, postkvantová kryptografie, skupinové podpisy, atributová autentizace

ABSTRACT

This thesis examines the intersection of two modern and growing branches of cryptography, namely privacy enhancing technologies and post-quantum cryptography. It describes selected privacy enhancing schemes (PETs) and families of post-quantum cryptography. In more detail, it focuses on group signatures based on mathematical problems that are difficult or intractable for both conventional and quantum computers. Furthermore, the thesis surveys the state of the art and compares the efficiency of mentioned schemes based on available data. The main part of this thesis is an implementation of a hash-based group signature and its comparison with lattice-based and code-based group signature implementations which were obtained directly from cryptographers active in this field. The post-quantum group signatures are subsequently compared to classic group signature schemes implemented by using the libgroupsig library.

KEYWORDS

Privacy Enhancing Technologies, post-quantum cryptography, group signatures, attribute authentication

HLUČKOVÁ, Pavla. *Moderní kryptografické protokoly s ochranou soukromí*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2021, 113 s. Diplomová práce. Vedoucí práce: doc. Ing. Lukáš Malina, Ph.D.

Prohlášení autora o původnosti díla

Jméno a příjmení autora:	Bc. Pavla Hlučková
VUT ID autora:	203171
Typ práce:	Diplomová práce
Akademický rok:	2021/2022
Téma závěrečné práce:	Moderní kryptografické protokoly s ochranou soukromí

Prohlašuji, že svou závěrečnou práci jsem vypracovala samostatně pod vedením vedoucí/ho závěrečné práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autorka uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušila autorská práva třetích osob, zejména jsem nezasáhla nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědoma následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno ....

podpis autorky*

*Autor podepisuje pouze v tištěné verzi.

PODĚKOVÁNÍ

Ráda bych poděkovala vedoucímu diplomové práce panu doc. Ing. Lukáši Malinovi, Ph.D. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.
I would like to thank Dr. Frederic Ezerman from Nanyang Technological University in Singapore, as well as Dr. Hyung Tae Lee from Jeonbuk National University in Korea for accomodating my requests and discussing their research with me. Finally, I would like to thank Dr. Gregor Seiler from IBM Research, Switzerland for the help regarding his publication and his insights on lattice-based cryptography.

Obsah

Seznam obrázků

Seznam tabulek

Úvod

Digitální technologie se staly centrálním prvkem většiny okruhů lidské činnosti, jak pracovní, tak volnočasové. Ochrana soukromí uživatelů různých systémů a služeb je tedy velice důležitou součástí tohoto technologického pokroku. Legislativa klade nároky na poskytovatele a správce osobních údajů, proto je na místě zavádění takových autentizačních systémů a digitálních podpisů, které ochraňují identitu uživatelů a usnadňují práci správcům těchto systémů a služeb. Mezi zmíněné systémy s ochranou soukromí patří např. atributová pověření a schémata skupinových podpisů, které jsou také hlavními tématy této diplomové práce.

Klasické systémy s ochranou soukromí, které jsou již v širším povědomí a začínají se prakticky využívat, se ovšem zakládají nejčastěji na problémech, které nejsou odolné vůči hrozbě kvantových počítačů. Technologický pokrok však spěje k vytvoření silného kvantového počítače, který bude schopen všechny matematické problémy založené na faktorizaci a diskrétním logaritmu vypočítat v polynomiálním čase. Proto je nutné brát ohled i na tento aspekt kryptografie a tvořit taková schémata s ochranou soukromí, která budou postavena na kvantově rezistentních problémech.

Cílem této diplomové práce je tedy teoreticky přiblížit schémata s ochranou soukromí a rodiny postkvantové kryptografie, prozkoumat *state-of-the-art* schémata v oblasti, která kloubí ochranu soukromí s matematickými problémy rezistentními vůči kvantové kryptoanalýze, zhodnotit jejich praktickou použitelnost a následně vytvořit implementaci bezpečného postkvantového schématu s ochranou soukromí a zhodnotit její efektivitu.

1 Kryptosystémy s ochranou soukromí

V dnešní době jsou technologie velmi rozšířené a jsou nutnou součástí lidských životů. Narůstá tedy i povědomí o procesech na pozadí užívaných systémů a aplikací. Proto i laická veřejnost přemýšlí o datech, která poskytuje třetím stranám, a žádá o jejich ochranu. Jako odpověď na tyto obavy jsou však k dispozici takové kryptografické systémy, které na první místo kladou ochranu soukromí uživatele.

Principem této ochrany soukromí je minimalizace informací poskytnutých během autentizačního procesu tak, aby nemohly být zneužity či odcizeny. Uživatel se tedy neprokazuje svou plnou identitou, ale pouze určitou vlastností či vlastnostmi (atributy) své osoby. Následně musí dokázat ověřovateli, že takové atributy vlastní. Jiné informace ohledně uživatelovy osoby jsou pro autentizaci nepodstatné.

Tato kapitola se tedy v obecnější rovině věnuje třem typům kryptosystémů s ochranou soukromí, a to **atributovým autentizačním schématům** (kap. 1.1), **skupinovým podpisům** (kap. 1.2) a **homomorfnímu šifrování** (kap. 1.3). Z hlediska této práce jsou zejména podstatné první dva zmíněné typy kryptosystémů.

1.1 Atributová autentizační schémata

S ohledem na bezpečnost, ochranu soukromí a majetku uživatelů je běžné, že autentizace předchází přístupu k informačním zdrojům. Většina tradičních autentizačních metod je však založena na údajích, které jsou přímo vázány k identitě autentizujícího se uživatele, a tudíž jej identifikují. Uživatel tedy získá přístup k prostředkům, ale ztratí svoji anonymitu. V některých případech je tento identifikující typ autentizace nezbytný, v jiných případech však zbytečně zasahuje do soukromí uživatele. Pokud identifikující autentizace příliš zasahuje do práv uživatele, je možné na tomto místě využít atributová autentizační schémata.

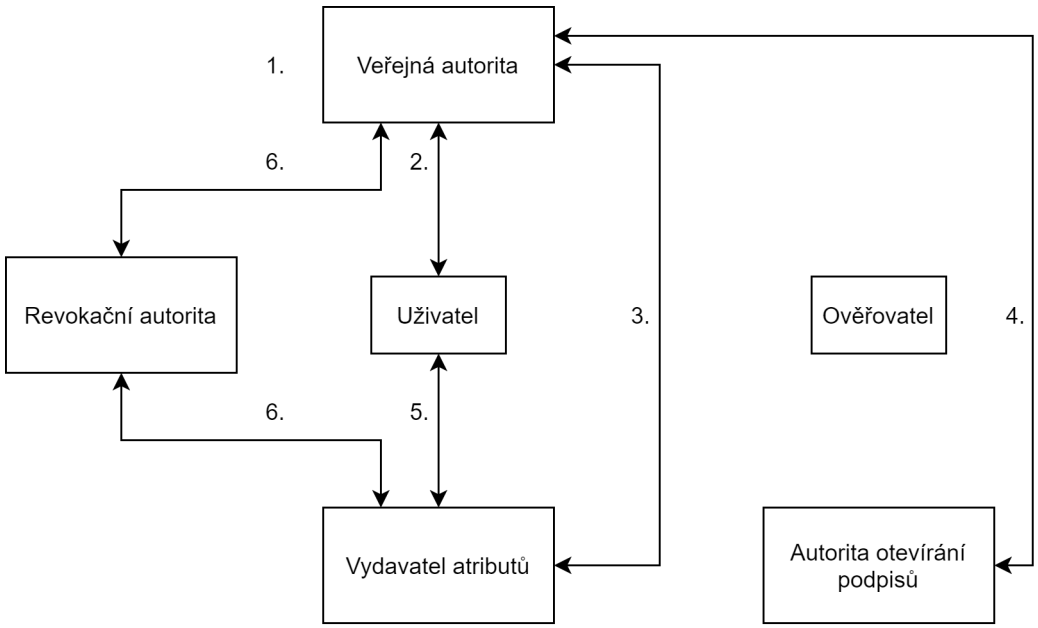
Atributová autentizační schémata využívají takový autentizační přístup, který umožňuje uživateli prokázat svou identitu prostřednictvím vlastnictví jednoho či více atributů. Atributy se týkají vlastností a prostředků uživatele, nicméně k autentizaci je využito pouze několik konkrétních atributů, které jednoznačně neidentifikují uživatelovu osobu.

Typickým příkladem atributově založené autentizace může být přístup k určité službě. Aby uživatel prošel autentizací, musí dokázat, že je vlastníkem atributů, které služba požaduje. Pokud uživatel dané atributy vlastní, pošle službě důkaz o jejich vlastnictví (většinou podpis). Při validním důkazu je uživateli umožněn přístup k dané službě, jinak je přístup zamítnut. Díky tomuto principu je chráněno uživatelovo soukromí a jeho identita je udržena v anonymitě [1].

1.1.1 Princip

Nejdříve je nutné představit entity, které se atributové autentizace účastní. Každá entita má specifickou roli v rámci schématu se svými úkony a oprávněními. Těmito entitami jsou veřejná autorita, uživatel, vydavatel atributů, ověřovatel, revokační autorita a autorita otevírání podpisů.

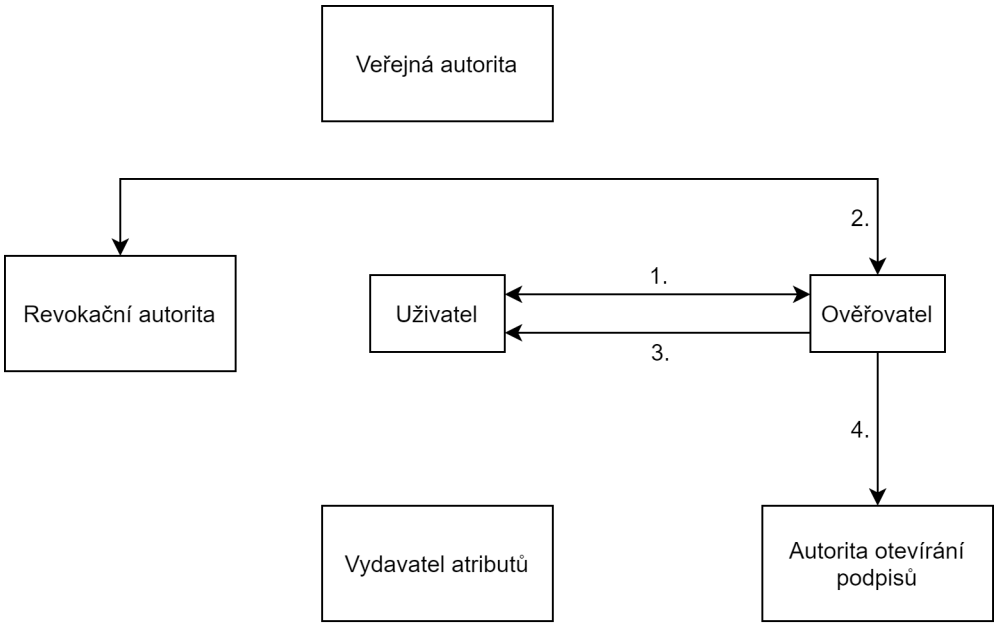
Postup atributové autentizace je rozdělen do dvou fází, počáteční fáze nastavování systému (obr. 1.1) a fáze autentizace (obr. 1.2) [1, 2, 3].



Obr. 1.1: Schéma první fáze atributové autentizace.

První fáze **nastavování systému** probíhá následovně:

1. Veřejná autorita generuje veřejné a soukromé parametry pro celý systém.
2. Uživatel žádá po veřejné autoritě svůj soukromý klíč. Tento klíč může být buď plně generovaný veřejnou autoritou (statická schémata), nebo se na jeho tvorbě může uživatel podílet pomocí protokolu *Join* (dynamická schémata).
3. Vydavatel atributů zjistí od veřejné autority generované parametry a podle nich generuje veřejné a privátní klíče jednotlivých atributů.
4. Autorita otevírání podpisů zjistí od veřejné autority trasovací klíče, které jsou určeny k revokaci anonymity podpisů (podepsaných atributů).
5. Uživatel žádá vydavatele atributů o privátní klíče ke svým atributům.
6. Revokační autorita pravidelně komunikuje s veřejnou autoritou a vydavatelem atributů tak, aby si udržovala databázi s revokačními informacemi.



Obr. 1.2: Schéma druhé fáze atributové autentizace.

Druhá fáze samotné **autentizace**, kdy má uživatel již všechno potřebné k dispozici, se skládá z následujících kroků:

1. Uživatel žádá ověřovatele o přístup. Ověřovatel na žádost odpoví výzvou nebo seznamem atributů, které požaduje. Uživatel podepíše zprávu pomocí svých soukromých atributových klíčů.
2. Ověřovatel se dotáže na informace v databázi revokační autority a pokud uživatelovy klíče nejsou revokovány, ověří validitu podpisu.
3. Ověřovatel zašle odpověď uživateli. V tuto chvíli je uživatel buď úspěšně autentizován, nebo odmítnut.
4. Pokud dojde k problémům a identita autentizujícího se uživatele musí být odhalena, obrátí se ověřovatel na autoritu otevírání podpisů. K odhalení identity a otevření podpisu využije autorita trasovací klíče.

1.1.2 Bezpečnostní požadavky

Z hlediska bezpečnosti jsou požadavky kladené na atributová autentizační schémata s ochranou soukromí následující [4]:

- **bezpečnost** – v souladu s moderními kryptografickými požadavky,
- **anonymita** – identita autentizujícího se uživatele je bezpečně skryta,
- **nesledovatelnost** – vydavatel nemůže vysledovat entity podle vlastnictví vydaných atributů,

- **nespojitelnost relací** – pokud má uživatel otevřeno vícero relací, není možné je spolu spojit,
- **ověření pomocí jednotlivých atributů** – uživatel je ověřen pouze na základě atributů potřebných k vykonání ověření,
- **nepřenositelnost** – uživatelé mezi sebou nejsou schopni atributy sdílet,
- **revokace** – existuje možnost revokace ověření, která nejsou validní.

1.2 Schémata skupinových podpisů

V dnešní době je užívání klasických digitálních podpisů naprosto běžné. Nejčastěji se můžeme setkat s podpisy využívajícími RSA, DSA či ECDSA. Nicméně tyto digitální podpisy nezaručují ochranu soukromí podepisujícího uživatele, jelikož jeho podpis je bezprostředně spojen s jeho identitou. Aby mohlo dojít k ochraně identity podepisujícího, byla sestavena schémata s podstatou ukrytí jeho identity za určitou skupinu, ke které uživatel patří. Uživatel pak může podepsat zprávu anonymně za celou skupinu. Příjemce zprávy ověří validitu podpisu pomocí veřejného klíče dané skupiny. Ví tedy, že obdržel korektní podpis, ale neví, kdo ze skupiny jej vydal. Každá skupina může mít svého manažera (*group manager*), jehož kompetence se mohou lišit v závislosti na konstrukci a entitách schématu. Manažer je nejčastěji vydavatelem parametrů a klíčů, ale v určitých případech může mít např. i možnost „otevřít“ podpis a zjistit, kterému členu skupiny patří.

Příkladem skupinového podpisu může být například situace, kdy skupina představuje firmu. Alice potřebuje získat firmou podepsaný dokument. Z jejího pohledu není podstatné, kdo ze zaměstnanců firmy dokument podepíše, a ani se to nemůže dozvědět. Alice přijme podepsaný dokument a podpis může validovat pomocí veřejného klíče firmy (skupiny). Pokud by došlo k nějakému problému, může kontaktovat Boba, který je manažer firmy (*group manager*). Bob jediný má schopnost podpis zpětně přiřadit ke konkrétnímu zaměstnanci, který jej vydal.

Skupinové podpisy lze chápat jako typ atributového autentizačního schématu s jedním atributem – příslušnost ke skupině.

1.2.1 Historie

První formální schéma skupinového podpisu navrhla dvojice *D. Chaum* a *E. van Heyst*. Schéma bylo publikováno v roce 1991 v rámci konference Eurocrypt [5]. Toto schéma mělo tři zásadní vlastnosti:

- Podepisovat zprávy mohou jen členové skupiny.
- Příjemce podpisu může ověřit, že se jedná o validní podpis dané skupiny. Nemůže však zjistit identitu uživatele, který podpis vytvořil.

- Pokud dojde ke sporu, je možné anonymitu danému podpisu odebrat a zjistit, kterému členu skupiny podpis patří.

Tento článek obsahuje čtyři návrhy schémat skupinových podpisů. Mezi problémy těchto návrhů patří nutnost dopředného stanovení maximálního počtu členů skupiny při zakládání systému a také lineární závislost délky podpisů a veřejného klíče na počtu členů skupiny. Větší skupiny tedy vyžadují delší podpisy a klíče, které kladou větší nároky na paměťový prostor i na šířku pásma v případě odesílání těchto klíčů [2]. Z hlediska této práce je také podstatný fakt, že zmíněná schémata jsou založena na výpočetních problémech faktorizace velkých čísel a diskrétního logaritmu, proto se nejedná o schémata odolná proti analýze pomocí kvantového počítače.

Chaum a van Heyst položili základ skupinovým podpisům, na nějž navázala výzkumná dvojice *J. Camenisch* a *M. Stadler* v roce 1997 [6]. Navrhli schéma umožňující dynamické zvyšování počtu členů skupiny a užívání veřejného klíče a podpisů nezávislých na počtu členů skupiny. Díky nim byly zavedeny termíny *statický* a *dynamický skupinový podpis*.

Statický skupinový podpis má předem stanovený počet členů a všechny dílčí privátní klíče jsou kalkulovány dopředu.

Dynamický skupinový podpis naopak umožňuje dynamicky přijímat členy skupiny, kteří mohou s procesem stanovení privátního klíče interagovat a popř. si určit jeho část.

1.2.2 Princip

Statické skupinové podpisy v sobě zahrnují pět operací. Jedná se o následující:

1. **Zahájení** (*setup*) – autorita generuje svůj veřejný a privátní klíč.
2. **Generování klíčů** (*keygen*) – autorita generuje následující klíče:
   - veřejný klíč skupiny (*general public key*), určen k validaci vytvořených podpisů,
   - trasovací klíč (*tracing key*), určen k revokaci anonymity podpisů,
   - privátní klíče (*private keys*), n privátních klíčů pro n členů skupiny.
3. **Podepsání** (*sign*) – generování podpisu pro danou zprávu za využití privátního klíče podepisujícího uživatele.
4. **Ověření** (*verify*) – kontrola validity podpisu zprávy za využití veřejného klíče skupiny.
5. **Otevření** (*open*) – vysledování původu podpisu za využití trasovacího klíče.

Operace zahájení a generování klíčů jsou někdy v rámci statických skupinových podpisů zahrnovány pod jeden termín *setup*.

Dynamické skupinové podpisy jsou složitější. Z důvodu dynamicky měnícího se počtu členů ve skupině zahrnují více operací než statické skupinové podpisy

a neobsahují operaci zahájení, protože není nutné dopředu generovat všechny privátní klíče pro potenciální členy skupiny:

1. **Generování klíčů** – operace generování tří klíčů:
   - veřejný klíč skupiny,
   - trasovací klíč,
   - vydavatelský klíč (*issuer key*), určen k tvorbě soukromých klíčů pro členy skupiny.
2. **Uživatelské generování klíčů** (*user keygen*) – uživatel, který se chce připojit ke skupině, generuje pár klíčů.
3. **Podepsání**
4. **Ověření**
5. **Otevření**
6. **Připojení** (*join*) – interaktivní protokol, který umožňuje uživateli přidat se ke skupině.
7. **Posouzení** (*judge*) – operace, která napomáhá zaručit odolnost vůči spojenectvím; ověřovateli podpisu umožňuje zjistit, zda bylo otevření podpisu manažerem korektní [7].

1.2.3 Bezpečnostní požadavky

Na schémata skupinových podpisů jsou kladeny bezpečnostní požadavky. Některé z nich byly převzaty od základních digitálních podpisů. Jelikož nejsou tyto požadavky formálně stanoveny, v literatuře se liší. Mezi základní požadavky na schémata skupinových podpisů podle [2, 8, 9] tedy patří:

- **Nepadělatelnost** (*unforgeability*) – skupinový podpis mohou vytvořit pouze členové skupiny; pokud podpis vydá někdo jiný, nedojde ke kladnému ověření.
- **Anonymita** (*anonymity*) – zjištění identity podepisujícího při známém podpisu je výpočetně náročné; schopnost podpis otevřít má pouze důvěryhodná strana, tedy manažer skupiny.
- **Trasovatelnost** (*traceability*) – při sporu je manažer skupiny schopen revokovat anonymitu daného podpisu a přiřadit jej k podepisujícímu.
- **Nespojitelnost** (*unlinkability*) – zjištění, zda byly dva různé podpisy vytvořeny stejným podepisujícím, je výpočetně náročné.
- **Vyvinění** (*exculpability*) – žádný člen skupiny nemůže vytvořit validní podpis za nezúčastněného člena skupiny, ani při spolupráci vícero členů včetně manažera.
- **Odolnost vůči spojenectvím** (*coalition–resistance*) – pokud členové skupiny zformují spojenectví, nemají schopnost společnými silami vytvořit podpis, který nebude možné otevřít nebo trasovat ke konkrétní osobě.

Často jsou uváděny pouze dva požadavky – plná anonymita a plná trasovatelnost (*full anonymity, full traceability*). Tyto dvě silnější verze požadavků v sobě již zahrnují ostatní požadavky, a tedy pokud se podaří tyto dvě vlastnosti prokázat, automaticky to implikuje i platnost ostatních požadavků [10].

1.3 Homomorfní šifrování

Homomorfní šifrování umožňuje provádět výpočetní operace nad šifrovanými daty bez nutnosti data dešifrovat. Díky tomu je zaručena důvěrnost dat i pokud dochází k jejich zpracování a ukládání u nedůvěryhodné třetí strany.

Lze jej definovat jako formu šifrování, kde je specifická algebraická operace prováděná nad otevřenými daty ekvivalentní k jiné algebraické operaci prováděné nad šifrovanými daty [11].

Existují tři typy homomorfního šifrování [12]:

- **Částečně homomorfní šifrování** (*partially homomorphic encryption*)
  - umožňuje provedení jedné operace nad šifrovanými daty (sčítání, násobení), ale ne obou.
- **Poněkud homomorfní šifrování** (*somewhat homomorphic encryption*)
  - umožňuje provedení více operací nad šifrovanými daty, ale jejich počet je limitovaný.
- **Plně homomorfní šifrování** (*fully homomorphic encryption*)
  - umožňuje výpočet libovolné funkce za využití sčítání i násobení.

V současnosti se nejvíce využívají částečně a poněkud homomorfní šifrovací schémata, zejména díky jejich efektivitě.

1.3.1 Historie

Ideu homomorfního šifrovacího schématu jako první začal rozvíjet tým *R. Rivest, L. Adleman* a *M. Dertouzos* v roce 1978[1]. Ve vědeckém článku zmíněná trojice navrhla čtyři aditivní schémata pod názvem „*Privacy Homomorphism*“, nicméně tyto návrhy byly prolomeny o devět let později dvojicí *E. Brickell* a *Y. Yacobi*. Vývoj tohoto odvětví kryptografie stagnoval až do roku 2009, kdy *C. Gentry* ve své disertační práci teoreticky navrhl bezpečné a efektivní schéma plně homomorfního šifrování. Na tomto základu staví novodobý výzkum schémat homomorfního šifrování, který se primárně zabývá tvorbou bezpečného a efektivního plně homomorfního kryptosystému [13].

[1] Rivest a Adleman se roku 1977 proslavili spoluprácí nad návrhem schématu kryptosystému RSA, běžně užívaného i v dnešní době.

1.3.2 Princip

Homomorfní kryptosystém je kryptosystém, který má přidanou homomorfní vlastnost takovou, že existuje výpočetně efektivní algoritmus pro určení součtu či součinu dvou zpráv při známém veřejném klíči a šifrovém textu těchto zpráv, nikoli jejich otevřeném textu. Pokud je prostor zprávy aditivní grupa, jedná se o aditivní homomorfní schéma, v opačném případě jde o multiplikativní homomorfní schéma.

Homomorfní kryptosystém obsahuje čtyři operace:

1. **Generování klíčů** – operace generování páru klíčů k_e a k_d.
2. **Šifrování** – šifrování zpráv za využití šifrovacího klíče k_e.
3. **Homomorfní vlastnost** (*homomorphic property*) – efektivní algoritmus pro výpočet součtu či součinu dvou zpráv.
4. **Dešifrování** – deterministické dešifrování za využití dešifrovacího klíče k_d.

Příkladem jednoduchého deterministického multiplikativního homomorfního kryptosystému je známé RSA, příkladem pravděpodobnostního aditivního homomorfního kryptosystému je schéma Goldwasser–Micali. V případě pravděpodobnostních schémat je do operace šifrování přidán parametr náhodného čísla. Efektivní algoritmus homomorfní vlastnosti by se měl také z hlediska bezpečnosti transformovat na pravděpodobnostní, k tomu se využívají zaslepovací algoritmy. Operace dešifrování zůstává deterministická [13].

1.3.3 Využití

Homomorfní šifrování, ačkoliv je jeho prvotní myšlenka přes čtyřicet let stará, se významně začalo rozvíjet až v posledních letech. Jeho popularita roste zejména díky možnostem jeho využití, příkladem je [13, 14, 15]:

- **Zvýšení bezpečnosti dat uložených na cloudu** – možnost uložení a zálohy dat v šifrované podobě a provádění vyhledávání nad šifrovanými daty. Třetí strana, u které jsou data uložena, je nemůže dešifrovat, a tudíž je pro ni jejich obsah skryt.
- **Šifrování dotazů a odpovědí do internetových vyhledávačů** – možnost skrytí obsahu vyhledávání, zabránění profilování, obrana před cílenou reklamou a ochrana soukromí. Vyhledávač odpoví zasláním šifrovaných dat, která může dešifrovat pouze uživatel pokládající daný dotaz.
- **Zprostředkování analýzy dat v regulovaných/citlivých odvětvích** – možnost sdílení a analýzy šifrovaných citlivých dat bez zásahu do soukromí subjektů údajů. Mezi okruhy využití patří personalizovaná reklama (reklamní systém není schopen poznat, jaké produkty sám doporučuje), analýza zdravotních údajů pacientů, dolování dat, finančnictví, forenzní rozpoznávání obrazu v kriminalistice aj.

- **Vylepšení a bezpečnost volebních protokolů** – např. návrh schématu pro elektronické volby bez nutnosti využití důvěryhodné třetí strany, která je jediným bodem selhání (*single point of failure*) tohoto systému, viz [16].
- **Využití k tvorbě kryptografických konstrukcí** – např. tvorba neinteraktivních protokolů s nulovou znalostí, delegaci výpočtů v cloudovém prostředí, digitálních podpisů, či tzv. *multi-party computation*, kdy se na výpočtu nad sdílenými daty podílí vícero stran.

2 Postkvantová kryptografická schémata

V dnešní době se je v rámci technologií hojně využívána asymetrická kryptografie. Její bezpečnost je z velké části založena na výpočetních problémech, které jsou při současném stavu techniky nespočitatelné – faktorizaci čísel (RSA) a diskrétním logaritmu (DSA, ECDSA, DH, ECDH, ElGamal). Vývoj kvantové výpočetní techniky však ohrožuje bezpečnost těchto velice rozšířených a známých algoritmů. Již v roce 1994 byl publikován tzv. Shorův algoritmus popisující faktorizaci prvočísel v reálném čase na kvantovém počítači a v roce 1996 tzv. Groverův algoritmus, který popisuje možnost útoků na symetrickou kryptografii (AES, SHA-256).

Postkvantová kryptografie je rozvíjející se vědní obor, který se zabývá tvorbou kryptografických schémat odolných vůči výpočetnímu výkonu kvantových počítačů. Předpokládá se, že existence takového počítače umožní výpočet výše zmíněných problémů v polynomiálním čase, a tudíž budou všechna schémata na nich postavená prolomena. Tento nárůst možného výpočetního výkonu se dotkne i symetrických šifer a tradičních hashovacích funkcí, u kterých bude nutné prodloužit délku tajných klíčů resp. délku výstupu, aby byla udržena jejich bezpečnost v přijatelné míře.

Je podstatné rozlišovat mezi termíny kvantová a postkvantová kryptografie. Zatímco kvantová kryptografie se zabývá algoritmy využívajícími principy kvantové fyziky (kvantové provázání, superpozice, tunelování) proveditelnými na kvantovém počítači, postkvantová kryptografie řeší algoritmy, které je možné využít na dnešních počítačích a které jsou odolné vůči standardní i kvantové kryptoanalýze.

I když není zřejmé, zda se takto výkonný kvantový počítač podaří sestavit, *US National Institute of Standards and Technology* (NIST) v roce 2017 započal proces standardizací asymetrických schémat odolných vůči kryptoanalýze kvantovým počítačem. V rámci tohoto projektu jsou v několika kolech od vědců a výzkumných skupin přijímány návrhy s nově vytvořenými postkvantovými schématy. Schémata jsou hodnocena a vhodní vybraní kandidáti na náhradu současných asymetrických kryptosystémů se následně stanou standardem.

Postkvantová schémata se dělí do několika rodin podle matematické podstaty, na které jsou tato schémata založena. Mezi rodiny postkvantové kryptografie patří schémata založená na **hashovacích funkcích** (kap. 2.1), schémata založená na **teorii kódování** (kap. 2.2), schémata založená na **mřížkách** (kap. 2.3), schémata založená na **polynomiálních rovnicích** (kap. 2.4) a v neposlední řadě schémata založená na **isogenii supersingulárních eliptických křivek** (kap. 2.5) [17].

2.1 Schémata založená na hashovacích funkcích

Hashovací funkce jsou takové matematické funkce, které jednosměrně mapují libovolně velký vstup m na výstup fixní délky h. Mezi bezpečnostní požadavky, které jsou na hashovací funkce kladeny, paří:

- **jednocestnost** – vypočítat $h(m)$ je jednoduché, ale pokud známe h, je výpočetně náročné zjistit m (není to však nemožné),
- **bezkoliznost** – je výpočetně náročné najít různé zprávy m_1, m_2 tak, aby měly stejný výsledný hash $h(m_1) = h(m_2)$,
- **bezpečnost** – odolnost vůči útokům, které se snaží nalézt kolize, ať už pomocí bruteforce, či kryptoanalýzou (útok na vnitřní strukturu hashovací funkce, zejména na tzv. kompresní funkci).

Mezi výhody schémat založených na hashovacích funkcích (*hash-based cryptography*) patří fakt, že nemají žádné přídavné bezpečnostní předpoklady kromě základního požadavku bezkoliznosti. Některé konstrukce pak zaručují bezpečnost i v případě, že je hashovací funkce pouze odolná vůči nalezení druhého vzoru (*second preimage resistance*), a tudíž nevyžadují hashovací funkci, která zaručuje odolnost vůči kolizím. Nevýhodou jsou pak dlouhé délky podpisů a absence výměny klíčů. Některá schémata jsou jednorázová, a tedy jejich pár klíčů, jak už z názvu vyplývá, nelze využít více než jednou [18].

V době psaní této práce mají schémata založená na hashích v rámci třetího kola standardizace postkvantových schémat NIST pouze jednoho kandidáta, který však není mezi finalisty. Mezi alternativními kandidáty se umístilo schéma pod názvem *SPHINCS+*, postkvantové podpisové schéma založené na hashích. Mezi další schémata založená na hashích patří např. **Lamportův jednorázový podpis** nebo **Winternitzův jednorázový podpis**, které jsou popsány níže.

2.1.1 Lamportův jednorázový podpis

Lamportův jednorázový podpis (*Lamport OTS*) je schéma postkvantového digitálního podpisu, které umožňuje bezpečné podepsání jedné zprávy. Algoritmus tohoto podpisu byl sestaven *L. Lamportem* roku 1979.

Princip při **podepisování** jednoho bitu zprávy je následující:

1. Vygenerování dvou náhodných řetězců $S_0, S_1 : |S_0| = |S_1| = 256\,\text{b}$. Tyto řetězce tvoří privátní klíč.
2. Výpočet hashů obou řetězců $H_0, H_1 : H_0 = h(S_0), H_1 = h(S_1)$. Tyto hashe tvoří veřejný klíč.
3. Zveřejnění veřejného klíče – hashů H_0, H_1.
4. Při podpisu bitu 0 dojde ke zveřejnění $s = S_0$.

5. Pro ověření platného podpisu bitu 0 stačí výpočet $h(s)$ a porovnání se zveřejněným klíčem H_0. Pokud je výsledek stejný, je podpis validní. Pokud by útočník chtěl prohlásit bit 1 jako validní, musel by zjistit S_1 nebo invertovat hash H_1.

Při podepisování n bitů je nutné generovat $S_{0,1} - S_{0,n}$ a $S_{1,1} - S_{1,n}$ jako privátní klíč, $H_{0,1} - H_{0,n}$ a $H_{1,1} - H_{1,n}$ jako veřejný klíč. Zpráva je zhashována na n bitů a podepisována postupně po jednotlivých bitech principem popsaným výše. Veřejný i soukromý klíč mají kvůli tomuto postupu velikost $2n^2$ bitů.

Lamportův podpis se často kombinuje s Merkleho stromem. Merkleho strom umožňuje generování velkého množství párů klíčů pro Lamportův podpis. Klíče se zahashují tímto stromem a při podpisu jsou zveřejněny jen ty uzly (hashe), které jsou nutné k ověření podpisu – tedy k výpočtu vrcholu stromu [19].

2.1.2 Winternitzův jednorázový podpis

Autor **Winternitzova jednorázového podpisu** (*W-OTS*), *R. Winternitz*, tento podpis publikoval pár měsíců po zveřejnění Lamportova OTS v roce 1979. Postup vytvoření a ověření tohoto podpisu při využití hashovací funkce SHA-256 zahrnuje:

1. Vygenerování třiceti dvou 256 bitových náhodných čísel $S_0 - S_{31}$. Těchto třicet dva hodnot tvoří privátní klíč.
2. Zhashování každé z 32 hodnot přesně 256 krát (př. $P_i = h^{256}(S_i), P_0 - P_{31}$). Tyto výsledné hodnoty tvoří veřejný klíč.
3. Zhashování zprávy pomocí algoritmu SHA-256 vytvoří výstup o délce 256 b, které lze rozdělit do třiceti dvou 8 bitových hodnot $N_0 - N_{31}$.
4. Při podpisu se vezme hodnota N_i a dojde k hashování každé z 32 částí soukromého klíče pomocí vzorce $h^{256-N_i}(S_i)$.
5. Pro ověření podpisu ověřovatel spočítá hash zprávy (hodnoty N_i), následně se daná hodnota podpisu $h^{256-N_i}(S_i)$ zhashuje N_i krát a porovná s hodnotou veřejného klíče $P_i : h^{N_i}(h^{256-N_i}(S_i)) \stackrel{?}{=} P_i$. Při rovnosti je podpis validní.

Winternitzův podpis se také velmi často využívá spolu s Merkleho stromem [20].

2.2 Schémata založená na teorii kódování

Schémata založená na **teorii kódování** (*code-based cryptography*) se začala rozvíjet díky vědci *R. J. McEliece*. V roce 1978 publikoval svou první implementaci využití binárních Goppa kódů v rámci asymetrického kryptografického schématu. Goppa kódy mají několik výhod – je jednoduché je vygenerovat, ale je těžké najít generující matici k existujícímu kódu, a zároveň mají rychlý dekódovací algoritmus řešitelný v polynomiálním čase.

Na výzkum *McElieceho* navázal v roce 1986 *H. Niederreiter*, který navrhl kryptosystém využívající kontrolní matici generalizovaných Reed–Solomonových kódů jako veřejný klíč a umožňoval i vytvoření digitálního podpisu. Reed–Solomonovy kódy byly později nahrazovány dalšími typy kódů (Bose–Chaudhuri–Hocquenghem kódy, Reed–Mullerovy kódy), ale tyto varianty byly zavrženy pro nedostatek důkazů bezpečnosti těchto schémat. Proto se Niederreiterovo schéma v dnešní době používá také s Goppa kódy, u kterých nejsou pochybnosti o důkazech jejich bezpečnosti [21].

Mezi výhody schémat založených na teorii kódování patří fakt, že se nejedná o nové kryptosystémy, a proto jsou dobře známé a prozkoumané. Umožňují provádět jednoduché a rychlé operace. Nevýhodou je však velikost potřebných parametrů (např. veřejného klíče) [18].

2.2.1 Kryptosystém McEliece

Bezpečnost **kryptosystému McEliece** je založena na problému náročnosti dekódování náhodného lineárního Goppa kódu. I když se jedná o poměrně starý kryptosystém, jeho potenciál byl zastíněn schématy založenými na problémech teorie čísel až do doby, kdy byl publikován Shorův algoritmus (viz úvod kap. 2) a byly zjištěny hrozby kvantových počítačů. Od té doby se zájem o tento kryptosystém zvyšuje a je předmětem mnoha studií.

Kryptosystém McEliece je dodnes považován za neprolomený, a dokonce je kandidátem na postkvantovou standardizaci NIST probíhající v roce 2022. V čase psaní této práce je mezi finalisty šifrovacích schémat třetího kola standardizace. Jako jediný je mezi finalisty založený na teorii kódování [17, 21].

Princip fungování kryptosystému McEliece, kde C je lineární $[n; k; t]$ kód délky n, dimenze k, schopný opravit t chyb, pro který existuje rychlý dekódovací algoritmus D_C (binární Goppa kód), je popsán níže [21, 22].

Generování páru klíčů probíhá následujícím způsobem:

1. S je matice o rozměrech $k \times k$, která je náhodnou binární regulární maticí[1],
2. G je matice o rozměrech $k \times n$, která je generující maticí kódu C,
3. P je matice o rozměrech $n \times n$, která je náhodnou binární permutační maticí,
4. G' je matice o rozměrech $k \times n$ taková, že pro ni platí $G' = S \cdot G \cdot P$.
5. Privátní klíč je tvořen trojicí (S, G, P), veřejný klíč je tvořen dvojicí (G', t).

Šifrování a **dešifrování** probíhá následovně:

1. Zpráva m v binární podobě o délce k je převedena do šifrového textu pomocí rovnice $c = m \cdot G' + z$, kde z je chybový vektor (náhodný n bitový vektor s Hammingovou váhou t).
2. Zašifrovaná zpráva c se odešle příjemci.

[1] K regulární matici existuje matice inverzní.

3. Příjemce vypočítá P^{-1} pomocí svého privátního klíče.
4. Po násobení $c' = c \cdot P^{-1}$ příjemce získá výsledek $c' = m \cdot S \cdot G + z \cdot P^{-1}$.
5. Protože podstatou dešifrování je existence rychlého dekódovacího algoritmu ke kódu C, je možné podle něj opravit chyby t a získat výsledek $m \cdot S$. K dekódování Goppa kódů se nejčastěji využívá *Pattersonův algoritmus*.
6. Příjemce ze svého privátního klíče vypočítá inverzní matici S^{-1} a výslednou dešifrovanou zprávu získá pomocí rovnice $m = m \cdot S \cdot S^{-1}$.

Na kryptosystém McEliece byly tvořeny různé matematické útoky, nicméně i přesto je považován za neprolomený, jelikož z praktického hlediska je nemožné jej při dostatečné délce kódu prolomit. Aby byl kryptosystém bezpečný, vědci v [23] doporučují kód C $[2048, 1751, 27]$. V tomto případě je veřejný klíč velice dlouhý, dosahuje délky $k(n-k) = 520047$ bitů.

2.3 Schémata založená na mřížkách

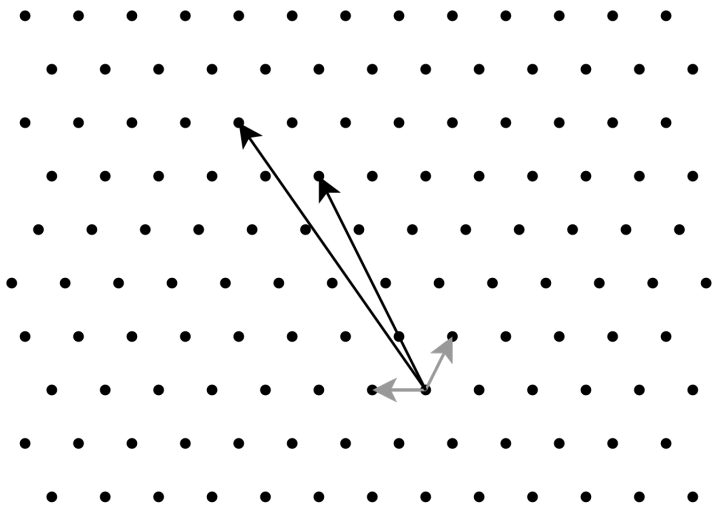
Možnost využití **mřížek** v kryptografii (*lattice-based cryptography*) nebyla známá až do roku 1992, kdy vědec a matematik *M. Ajtai* definoval problém nejkratšího vektoru (SVP) a nastartoval tím výzkum týkající se využití matematických problémů formulovaných na mřížkách ke konstrukci kryptografických schémat. Tato schémata mají zároveň do budoucna velký potenciál, co se týká odolnosti vůči kvantovým počítačům. Kryptografické konstrukce založené na mřížkách mají totiž silné důkazy bezpečnosti založené na výpočetní náročnosti a rychlé, jednoduché operace, nicméně délka parametrů je velká, stejně jako v případě ostatních postkvantových schémat [18, 24].

Mřížka, jako matematická konstrukce, je množina bodů v n–dimenzionálním prostoru, která má periodickou strukturu (obr. 2.1). Je definována množinou lineárně nezávislých vektorů $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_n$ prostoru $\mathbb{R}^n$. Mřížka generovaná danou množinou vektorů je formálně zapsána jako 2.1, kde vektory $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_n$ jsou bázemi této mřížky. Báze mohou být organizovány do matice $\mathbf{B} = [\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_n] \in \mathbb{R}^{n \times n}$, přičemž jednotlivé bázové vektory tvoří sloupce této matice [24].

$$\mathcal{L}(\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_n) = \left\{ \sum_{i=1}^{n} x_i \mathbf{b}_i : x_i \in \mathbb{Z} \text{ pro } 1 \leq i \leq n \right\} \tag{2.1}$$

Nejznámější matematické problémy, na kterých jsou mřížkové kryptosystémy založeny, jsou problémy **nalezení nejkratšího vektoru** (SVP) a **nalezení nejbližšího vektoru** (CVP). SVP využívá složitost nalezení nejkratšího nenulového vektoru v $\mathcal{L}(\mathbf{B})$ při znalosti $\mathbf{B}$. CVP naopak využívá složitost nalezení vektoru v mřížce $\mathbf{v} \in \mathcal{L}(\mathbf{B})$ takového, že má nejmenší vzdálenost k danému vektoru $\mathbf{t} \in \mathbb{R}^n$, při znalosti $\mathbf{t}$ a $\mathbf{B}$. Vektor $\mathbf{t}$ nemusí ležet na mřížce [22, 24].

Ve třetím kole standardizace postkvantových algoritmů organizací NIST se mezi finalisty nachází tři šifrovací a dvě podpisová schémata založená na mřížkách. Schémata založená na mřížkách jsou tedy nejčetnější skupinou v rámci tohoto kola, což svědčí o jejich praktické využitelnosti a popularitě. Finalisty jsou šifrovací schémata *Crystals–Kyber*, *NTRU* a *Saber* a dále podpisová schémata *Crystals–Dilithium* a *Falcon* [17].



Obr. 2.1: Příklad dvou bází $n = 2$ dimenzionální mřížky.

2.3.1 Schéma NTRU

Mřížkové schéma **NTRU** (*N-th degree Truncated polynomial Ring Units*) bylo publikováno roku 1998 v [25] trojicí vědců *J. Hoffstein, J. Pipher* a *J. H. Silverman*. V rámci této publikace se jednalo pouze o šifrovací schéma, roku 2003 však tato vědecká skupina publikovala i NTRU schéma pro digitální podpis (viz [26]). Jelikož se podpisové schéma NTRUSign neumístilo mezi kandidáty na standardizaci NIST, bude tato podkapitola věnována pouze první, šifrovací části NTRU schématu.

Schéma NTRU bylo prvně prezentováno jako schéma založené na okruhu konvolučních polynomů ($\mathcal{R} = \mathbb{Z}[x]/(x^n - 1)$). NTRU lze však přepsat do formy založené na teorii mřížek, jelikož konvoluční násobení polynomů je možné vyjádřit pomocí cyklické matice. Mřížka využívaná tímto kryptosystémem je potom speciální (*konvoluční modulární*) [27, 28].

Schéma využívá třech hlavních celočíselných parametrů n, p, q. Pro čísla p, q platí, že nemusí být prvočísla, ale jsou nesoudělná ($\gcd(p, q) = 1$) a q je mnohem větší než p ($q \gg p$). Zároveň se využívá operace $*$, která je v rámci okruhu $\mathcal{R} = \mathbb{Z}[x]/(x^n - 1)$ ekvivalentní klasickému násobení [29]. Postup fungování schématu NTRU je rozepsán v krocích níže [28, 30].

Generování klíčů probíhá následujícím způsobem:

1. Generování dvou náhodných polynomů f, g nejvyššího stupně $n - 1$ s malými koeficienty. K polynomu f existují inverzní polynomy F_q a F_p takové, že $F_q * f \equiv 1 \mod q$ a $F_p * f \equiv 1 \mod p$. K výpočtu inverzních polynomů se využije rozšířený Euklidův algoritmus.
2. Výpočet polynomu h pomocí rovnice $h \equiv F_q * g \mod q$.
3. Veřejný klíč tvoří polynom h, soukromý klíč tvoří dvojice (f, F_q). Polynom F_q by nemusel být součástí privátního klíče, je však vhodné jej nepřepočítávat při každém dešifrování, proto se uchovává.

Operace **šifrování** zprávy probíhá následovně:

1. Zpráva m bude šifrována pomocí rovnice $e \equiv p \cdot s * h + m \mod q$, kde h je veřejný klíč, p je číslo a s je náhodně vygenerovaný polynom.
2. Zašifrovaná zpráva e je odeslána příjemci.

Dešifrování probíhá v následujících krocích:

1. Výpočet polynomu a za využití soukromého klíče f pomocí rovnice $a \equiv f * e \mod q$. Koeficienty polynomu a budou z rozmezí intervalu od $-q/2$ do $q/2$.
2. Originální zpráva bude získána po výpočtu $m \equiv F_q * a \mod p$. Polynom a lze totiž vyjádřit jako $a \equiv p \cdot s * g + f * m \mod q$. Po redukci modulo p je získán polynom $f * m \mod p$, a tedy díky inverzi F_p k f lze zprávu m získat jako výsledek rovnice $m \equiv F_p * f * m \mod p$.

Bezpečnost tohoto schématu je založena na obou zmíněných mřížkových problémech – CVP i SVP, protože odhad privátního klíče f z veřejného klíče h je ekvivalentní k hledání nejkratšího vektoru mřížky (SVP), zatímco odhad zprávy m z veřejného klíče h a šifrového textu e je ekvivalentní k hledání vektoru mřížky nejbližšího k $[0, \mathrm{e}]$ (CVP) [31].

2.4 Schémata založená na polynomiálních rovnicích

Kryptografická schémata založená na **polynomiálních rovnicích** (*multivariate cryptography*) se začala objevovat už během 80. let 20. století zejména díky vědcům *T. Matsumoto* a *H. Imai*. I když byly jejich původní návrhy polynomiálních kryptosystémů prolomeny, položily základ této rodiny postkvantové kryptografie.

Polynomiální schémata využívají nejčastěji kvadratické polynomy více proměnných na konečném tělesu. Jejich bezpečnost je založena na výpočetním problému

řešení nelineárních rovnic na tomto tělesu, který spadá do třídy složitosti NP-těžký.

Jednosměrná funkce se zadními vrátky $\mathcal{P}$ (*trapdoor one-way function*), která se u těchto kryptografických schémat využívá, je definována jako množina m polynomů malého stupně d s n proměnnými na konečném tělesu F. Matematický zápis této funkce je zobrazen v 2.2. Aby bylo možné dešifrovat zprávu, uživatel musí pro známé $\mathbf{z} = (z_1, \ldots, z_m)$ najít řešení $\mathbf{w} = (w_1, \ldots, w_m)$ takové, aby splňovalo $\mathcal{P}$. $\mathbf{z}$ je hash zprávy v případě digitálního podpisu resp. šifrový text v případě šifrovacího schématu, $\mathbf{w}$ je podpis zprávy resp. otevřený text.

$$(\mathcal{P}) \begin{cases} p_1(w_1, \ldots, w_n) = z_1 \\ \ldots \\ p_m(w_1, \ldots, w_n) = z_m \end{cases} \tag{2.2}$$

Jednosměrné funkce se zadními vrátky jsou jednoduché na výpočet jedním směrem, opačným směrem je však jejich výpočet složitý bez znalosti speciální tajné informace. Je tedy jednoduché ověřit, že dvojice $(\mathbf{w}, \mathbf{z})$ splňuje parametry $\mathcal{P}$. Také je jednoduché vypočítat $\mathbf{z} = \mathcal{P}(\mathbf{w})$ pro dané $\mathbf{w}$. Nalezení řešení $\mathbf{w}$ by však mělo být výpočetně složité pro většinu $\mathbf{z}$ bez znalosti tajného klíče, ale jednoduché se znalostí tohoto klíče.

Konkrétněji jsou tato polynomiální schémata založená na jednoduše řešitelném systému $\mathcal{Q}(\mathbf{x}) = \mathbf{y}$. Následně jsou provedeny dvě tajné náhodné lineární transformace $S : \mathbf{w} \to \mathbf{x}$ a $T : \mathbf{y} \to \mathbf{z}$. Pomocí těchto transformací a $\mathcal{Q}$ je složen nový systém $\mathcal{P}$, který ve schématu tvoří veřejný klíč. Soukromý klíč je tedy tvořen $\mathcal{Q}$ a transformacemi S, T [32].

Mezi výhody schémat založených na polynomech patří rychlost a jednoduchost operací, které provádějí. Nevýhodou je potom velikost parametrů kryptosystémů [18].

V rámci třetího kola standardizace postkvantových schémat NIST se mezi finalisty nachází jedno podpisové schéma založené na polynomiálních rovnicích. Jedná se o schéma *Rainbow*, které je založené na konstrukci typu „*Oil-and-Vinegar*“ [17].

2.5 Schémata založená na isogenii supersingulárních eliptických křivek

Kryptografická schémata založená na homomorfismu mezi supersingulárními eliptickými křivkami se začala rozvíjet až po roce 2011, kdy vědci *D. Jao* a *L. De Feo* v publikaci [33] navrhli nový moderní protokol výměny klíčů založený na isogenii supersingulárních eliptických křivek *SIDH*, jako alternativu k Diffie–Hellman protokolu. Schéma Diffie–Hellman je založené na problému diskrétního logaritmu, a tedy

i jeho klasická varianta na eliptických křivkách je stavěna na stejném problému. Protože diskrétní logaritmus bude nejpravděpodobněji prolomen kvantovým počítačem, zmínění vědci využili podobnou ideu, avšak založenou na problému nalezení j-invarianty a jádra (*kernel*) isogenie. Tento problém je rezistentní vůči kvantové kryptoanalýze. Byl to inovativní krok, jelikož do té doby byla pouze publikována schémata, která využívala isogenii běžných eliptických křivek [34].

Isogenie je druh skupinového homomorfismu. Dvě eliptické křivky $E(\mathbb{F}_q)$ a $E'(\mathbb{F}_q)$ jsou isogenní nad polem $\mathbb{F}_q$ právě tehdy, když existuje morfismus $\varphi : E \to E'$ s koeficienty v $\mathbb{F}_q$ takový, že mapuje neutrální prvek (identitu) křivky E na neutrální prvek křivky E'. Jinými slovy, každá eliptická křivka má jednu j-invariantu. Pokud mají dvě křivky stejnou j-invariantu, jsou k sobě isogenní. Supersingulární eliptické křivky jsou pak speciální křivky na konečném tělesu, jejichž řád (počet bodů) splňuje rovnici $\#E(F_q) \equiv 1 \mod p$ pro $q = p^k$ [35, 36].

Mezi výhody těchto schémat patří fakt, že se zakládají na klasických eliptických křivkách, které jsou již dobře známé a prostudované. Zároveň velikosti parametrů jsou menší oproti ostatním rodinám postkvantové kryptografie. Mezi nevýhody patří nižší rychlost operací a také to, že supersingulární eliptické křivky se začaly využívat v kryptografii poměrně nedávno, a tudíž nejsou tak rozšířené a v obecném povědomí [18].

Ve třetím kole standardizace NIST nemá tato rodina postkvantové kryptografie žádné zastoupení mezi finalisty. Mezi alternativními kandidáty se však umístilo jedno schéma založené na supersingulárních eliptických křivkách, a to šifrovací schéma *SIKE* [17].

3 Trendy v oblasti postkvantové kryptografie s ochranou soukromí

Aby bylo možné stanovit, které rodiny postkvantové kryptografie jsou nejrozšířenější v kombinaci s kryptografickými schématy s ochranou soukromí, je nutné vyhledat publikace v této oblasti a určit nejčetnější zástupce postkvantových rodin. K provedení této statistiky byla vybrána platforma Google Scholar. Mezi relevantní publikace byly započteny pouze ty, které byly publikovány od roku 2015 do konce roku 2021. Zároveň byly hledány pouze články v anglickém jazyce.

Google Scholar umožňuje dva módy hledání podle umístění vyhledávaných slov, a to jejich umístění v titulu publikace a umístění v rámci celé publikace. Aby došlo k vyfiltrování článků, které se o hledaných tématech pouze okrajově zmiňují, do statistik jsou započítávány pouze ty, které mají zmíněné termíny ve svém názvu.

Klíčová slova, pod kterými byly publikace vyhledávány, jsou uvedena v příloze A spolu s tabulkami, které zaznamenávají dílčí hodnoty. Byly zkoumány pojmenovávací konvence publikací daného tématu a vyhledávací dotazy byly pro každé téma manuálně sestavovány tak, aby pokryly co nejvíce článků z daného tématu a zároveň aby byly navzájem disjunktní, a tedy nedocházelo k započítání stejné publikace vícekrát napříč dotazy.

Následující podkapitoly se věnují vyhledávání vědeckých článků o atributově založených schématech a skupinových podpisech (kap. 3.1), článků věnujících se rodinám postkvantové kryptografie (kap. 3.2) a jejich kombinaci (kap. 3.3). Všechny relevantní informace jsou shrnuty v kap. 3.4.

3.1 Atributová schémata a skupinové podpisy

Tématům atributově založených schémat a skupinových podpisů bylo od začátku roku 2015 do listopadu 2021 věnováno celkově 1978 vědeckých článků dohledatelných přes Google Scholar.

Počet článků věnovaných atributově založeným schématům byl k 1. listopadu 2021 celkově 1656. K tomuto datu bylo také zveřejněno celkem 322 vědeckých článků týkajících se skupinových podpisů.

Za posledních 6 let bylo tedy publikováno o 68 % více článků věnovaných atributově založeným schématům než skupinový podpisům. Na obr. 3.1 (a) je tento poměr zobrazen. Zároveň na obr. 3.1 (c) je zobrazen graf počtu publikací jednotlivých témat v daných letech. Z grafu vyplývá, že nejvíce publikací týkajících se atributových schémat bylo zveřejněno v roce 2016 a dále tento počet klesá. Pro skupinové podpisy

byly nejlepšími lety 2018 a 2019. Publikace o skupinových podpisech sledují konstantnější trend, který se drží v podobném průměru počtu publikací ročně, oproti atributovým schématům, které v posledních letech začaly strmě klesat.

V rámci atributově založených schémat byla vyhledávána i nejčastější přívlastky, které se vážou k těmto schématům. Mezi ně patří šifrovací (*encryption*), podpisové (*signature*), pověření (*credentials*) a autentizační (*authentication*) typy atributových schémat. Počet publikací, které v titulku obsahovaly tato slova spolu s danými klíčovými slovy, je zobrazen v grafu na obr. 3.1 (b). Z grafu vyplývá, že nejčastější kombinací je atributově založené šifrovací schéma, které bylo od roku 2015 publikováno 1345 krát. Publikací věnovaných atributovým podpisům, atributové autentizaci obecně a atributovým pověřením je výrazně méně. Šifrovací schémata totiž tvořila celkem 81 % ze všech článků o atributových schématech, což svědčí o jejich popularitě. Na obr. 3.1 (d) je zobrazen graf počtu článků o daných tématech v jednotlivých letech. Počty článků zmíněných typů atributových schémat se v průběhu doby sledování výrazněji nemění s výjimkou šifrovacích schémat, jejichž počet byl nejvyšší v roce 2017 a od té doby klesá.

Zdrojová data k této podkapitole jsou spolu s dílčími výsledky statistiky a vyhledávacími dotazy zobrazena v tabulce v příloze A.1.

3.2 Postkvantové kryptografické rodiny

V této části byly vyhledávány kryptosystémy umožňující šifrování, podpis či obojí, které jsou založeny na problémech rezistentních vůči kvantové kryptoanalýze. Byly započítány pouze ty publikace, které byly zveřejněny v období mezi rokem 2015 a 1. listopadem 2021. Mezi vyhledávané rodiny jsou zařazeny ty, které jsou popsány ve 2. kapitole této práce, a tedy rodiny založené na mřížkách, na hashích, na teorii kódování, na polynomiálních rovnicích a na isogenii supersingulárních křivek.

Celkově bylo pod danými vyhledávacími dotazy nalezeno 708 vědeckých publikací a článků, které se týkají zmíněných pěti rodin postkvantové kryptografie. Nejvíce publikací bylo věnováno schématům založeným na mřížkách – celkově 409. Následovala schémata založené na teorii kódování, hashovacích funkcích, polynomiálních rovnicích a isogenii eliptických křivek s počtem publikací 155, resp. 94, 35 a 15. Vizualizace těchto dat je zobrazena v grafu na obr. 3.2 (a).

Zároveň byla vytvořena statistika nejvíce zastoupených postkvantových kryptografických rodin v rámci třetího kola standardizace postkvantových schémat NIST. Do třetího kola se dostalo celkem 15 schémat, z nichž se do statistiky započítávalo pouze 14. Jedno z kandidátních schémat je totiž založené na symetrických šifrách, proto není započítáno. Tato statistika je porovnána se souborem vyhledaných publikací o 708 nalezených článcích. Procentuální zastoupení jednotlivých rodin v rámci

vyhledaných publikací a v rámci kandidátů je podobné až na výjimku hashovacích funkcí a polynomiálních rovnic. Rodina schémat založených na hashích tvoří ve vyhledaných publikacích celkem 13 %, kdežto v množině kandidátů se jedná pouze o 7 %. V rámci standardizace má tedy tato rodina nižší oblibu. Opačným příkladem jsou pak schémata založená na polynomiálních rovnicích, která v rámci vyhledaných publikací zastoupila 5 %. U standardizace se však jedná o početnější skupinu se 14 %. Lze tedy tvrdit, že schémata založená na polynomiálních rovnicích mají vyšší relativní kvalitu a efektivitu, jelikož mají vícero kandidátů při nižším počtu celkových publikací. Stejně tak lze předpokládat, že hashovací funkce neposkytují tak kvalitní postkvantové řešení, proto nejsou v rámci kandidátů na standardizaci tak zastoupeny. Porovnání je znázorněné v tab. 3.1.

Různé typy postkvantových rodin a počet jejich publikací lze také sledovat během let 2015 až 2021 na grafu na obr. 3.2 (b). Z grafu je zřejmé, že za posledních 6 let strmě vzrostl zájem o schémata založená na mřížkách. Od roku 2016 také vzrostl zájem o schémata založená na teorii kódování. Schémata s hashovacími funkcemi se drží v podobném každoročním počtu publikací a počty publikací se schématy využívajícími polynomiální rovnice jdou nepatrně vzhůru. Stejně tak se zvyšují počty publikací dedikovaných isogenii supersingulárních eliptických křivek, které v roce 2015 byly nulové a od té doby jdou v řádu jednotek víceméně vzhůru.

Zdrojová data k této podkapitole jsou spolu s dílčími výsledky, vyhledávacími dotazy a statistikou standardizace NIST zobrazeny v tabulkách v příloze A.2.

3.3 Postkvantová atributová schémata a skupinové podpisy

Podle statistik z předchozích dvou podkapitol bylo odhadováno, že by se mezi nejpopulárnější postkvantová schémata s ochranou soukromí měla zařadit zejména šifrovací atributová schémata založená na mřížkách a teorii kódování. Následovaly by skupinové podpisy založené na mřížkách popř. na polynomiálních rovnicích či hashích a poté také atributová podpisová schémata založená na těchto stejných postkvantových rodinách.

Výsledky tohoto průzkumu byly však v nějakých ohledech překvapivé – např. skupinové podpisy založené na postkvantových problémech mají celkově od roku 2015 dodnes více publikací než postkvantová atributová schémata. Zároveň skupinové podpisy mají minimálně jeden návrh schématu v rámci každé představené postkvantové rodiny, kdežto atributová schémata nemají žádného zástupce při využití problémů založených na polynomiálních rovnicích či isogenii supersingulárních eliptických křivek.

Z celkového množství 1656 publikovaných atributových schémat (viz kap. 3.1) bylo celkem 43 založeno na postkvantových problémech, což činí pouze 2,5 %. Z 322 schémat skupinových podpisů bylo 49 postkvantových, tudíž až 15 %. Z tohoto výsledku tedy vyplývá, že populárnější formou jsou postkvantové skupinové podpisy. Celkově ze všech vyhledaných článků skupinových podpisů a atributových schémat bylo pouze necelých 5 % založeno na problémech odolných kvantové kryptoanalýze.

Podrobněji bylo tedy dohledáno za období 2015 až 1. listopad 2021 92 publikovaných článků o postkvantových atributových schématech a skupinových podpisech. Postkvantová atributová schémata tvořila 47 % z těchto článků, postkvantové skupinové podpisy 53 %. V rámci těchto dvou typů schémat s ochranou soukromí dominovaly postkvantové problémy založené na mřížkách. Méně častými zástupci pak byly rodiny založené na teorii kódování a na hashích. Hashe byly populárnější u atributových schémat a teorie kódování naopak u skupinových podpisů. Zajímavým trendem pak mohou do budoucna být skupinové podpisy založené na polynomiálních rovnicích a isogenii eliptických křivek – v letošním roce se totiž objevily první publikace u obou těchto témat. Informace uvedeny v tomto odstavci jsou shrnuty v tab. 3.2. Grafické znázornění počtů publikací postkvantových atributových schémat, resp. skupinových podpisů je na obr. 3.3, resp. na obr. 3.4.

Byla vyhledávána i konkrétní přídavná jména spojená s typy atributových schémat. Jednalo se o šifrovací, podpisové, pověřovací a obecně autentizační typy atributových schémat, stejně jako v kap. 3.1. Postkvantová atributová šifrovací schémata byla nejčastěji založená na mřížkách, dále na hashích a na teorii kódování. Postkvantové atributové podpisy byly nejčastěji také založeny na mřížkách a na teorii kódování. Ostatní rodiny u tohoto typu atributového schématu neměly zastoupení. Vědecké publikace s obecným titulkem obsahujícím „atributové autentizační schéma“ byly založeny pouze na hash funkcích. Informace z tohoto odstavce jsou shrnuty v tab. 3.3. Obr. 3.5 obsahuje grafické znázornění počtů článků, které se věnují kombinaci jednotlivých typů atributových schémat během sledovaných let 2015 až 2021.

Zdrojová data k této podkapitole jsou spolu s dílčími výsledky a vyhledávacími dotazy zobrazena v tabulkách v příloze A.3.

Tab. 3.1: Procentuální složení publikací a kandidátů NIST podle PQ rodin.

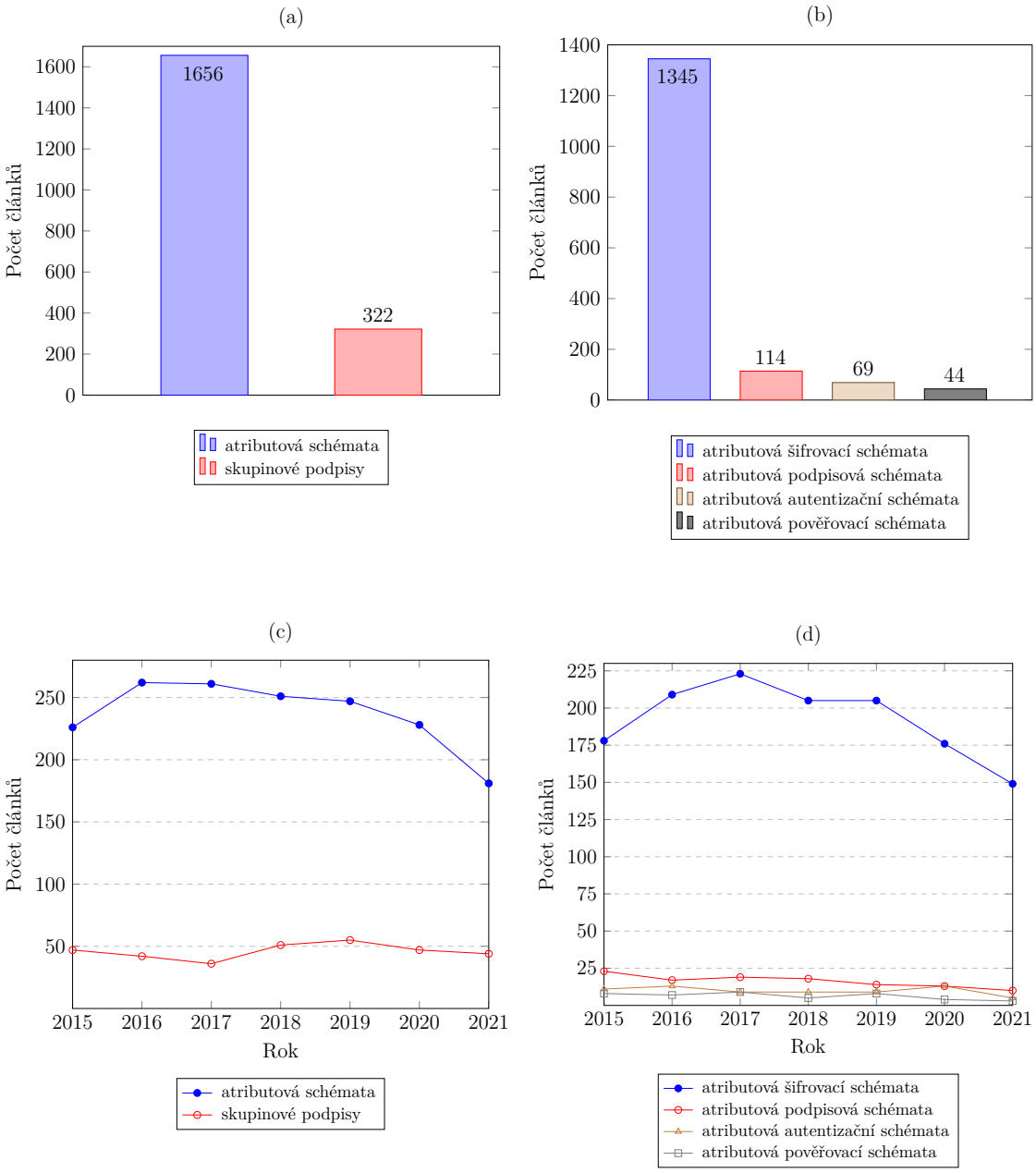
PQ rodina	**PQ články** [%]	**PQ NIST kandidáti** [%]
mřížky	58	51
teorie kódování	22	21
hashovací funkce	13	7
polynomiální rovnice	5	14
isogenie eliptických křivek	2	7

Tab. 3.2: Procentuální složení PQ rodin mezi PQ atr. schémata a sk. podpisy.

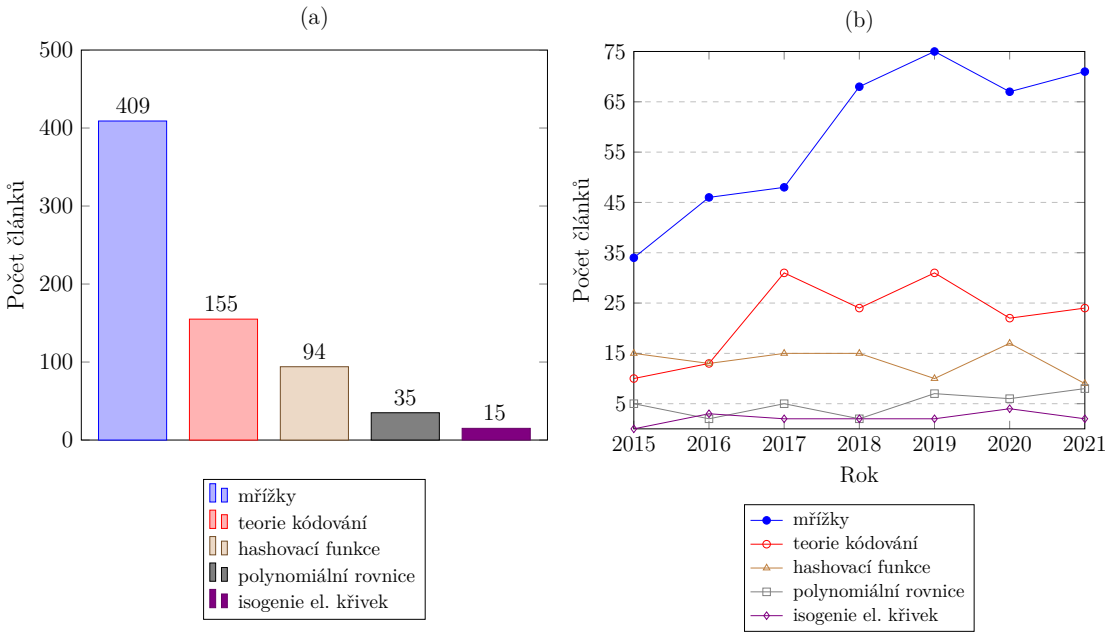
PQ rodina	**PQ atr. schémata** [%]	**PQ sk. podpisy** [%]
mřížky	81	82
teorie kódování	7	10
hashovací funkce	12	4
polynomiální rovnice	0	2
isogenie eliptických křivek	0	2

Tab. 3.3: Procentuální složení PQ rodin mezi typy PQ atr. schémat.

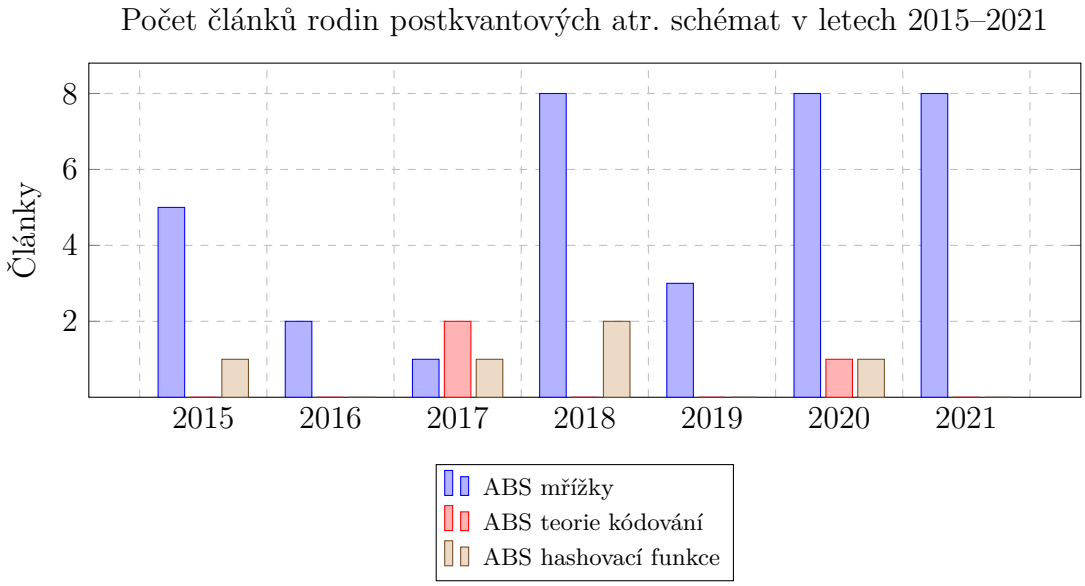
PQ rodina	**ABE** [%]	**ABS** [%]	**ABA** [%]
mřížky	81	89	0
teorie kódování	5	11	0
hashovací funkce	14	0	100
polynomiální rovnice	0	0	0
isogenie eliptických křivek	0	0	0



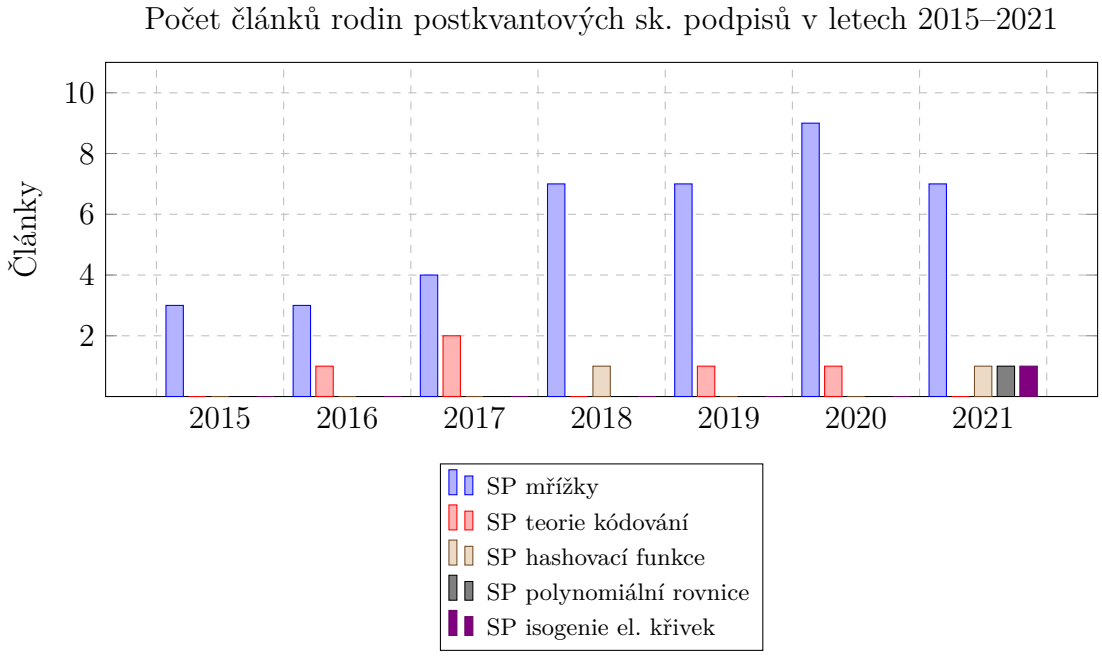
Obr. 3.1: Publikace na témata atr. schémat a sk. podpisů na Google Scholar.



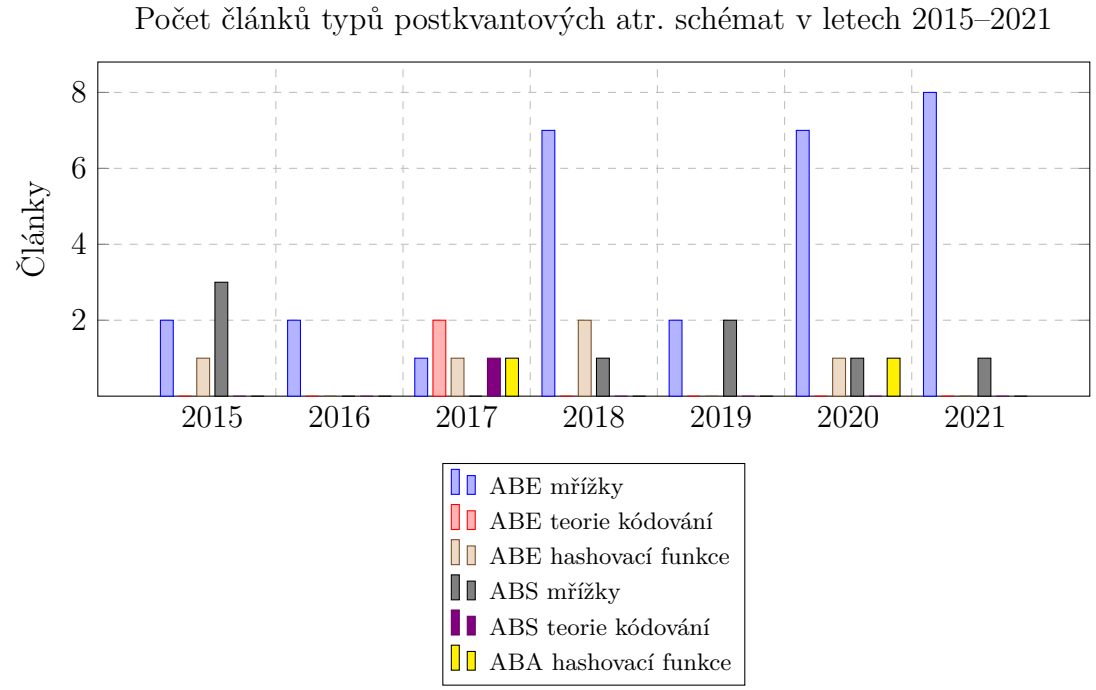
Obr. 3.2: Publikace na témata rodin PQ kryptografie na Google Scholar.



Obr. 3.3: Graf počtu článků rodin postkvantových atr. schémat v daných letech.



Obr. 3.4: Graf počtu článků rodin postkvantových sk. podpisů v daných letech.



Obr. 3.5: Graf počtu článků typů postkvantových atr. schémat v daných letech.

3.4 Shrnutí

V této kapitole byl popsán průzkum publikovaných článků na platformě Google Scholar. Vyhledávací dotazy byly ručně sestavovány po průzkumu jmenných konvencí publikací v rámci každého tématu tak, aby vyloučily překryv výsledků mezi dotazy. Byly vyhledávány pouze anglické články publikované mezi lety 2015 až 2021. Aby byl článek započítán do statistik, musel obsahovat vyhledávaná slova ve svém názvu. Celkově bylo provedeno 294 dotazů a vyhledáno 4397 relevantních publikací.

Z kap. 3.1 vyplývá, že nejužívanějším schématem s ochranou soukromí jsou atributově založená schémata. Bylo jich publikováno o 67 % více než skupinových podpisů. Nejpopulárnějším typem atributového schématu je atributové šifrovací schéma, které tvořilo celkově 81 % z nalezených článků o atributových schématech.

V kap. 3.2 bylo ukázáno, že nejrozšířenější a nejužívanější rodinou postkvantové kryptografie jsou mřížky. V rámci nalezených publikací se jedná o 58% většinu. Následuje teorie kódování, hashovací funkce, dále minimum polynomiálních rovnic a ještě méně isogenií supersingulárních eliptických křivek. Tento soubor výsledků byl porovnán se statistikou, která byla vytvořena z množiny kandidátů na postkvantovou standardizaci NIST. Porování ukázalo poměrně podobné výsledky. Jedinou výjimkou byla snížená popularita hashovacích funkcí a naopak zvýšená popularita polynomiálních rovnic v rámci standardizace.

Kapitola 3.3 kombinuje vyhledávání postkvantových rodin a schémat s ochranou soukromí. Z výsledků statistiky je zřejmé, že populárnějším postkvantovým schématem s ochranou soukromí jsou skupinové podpisy. Ze všech vyhledaných atributových schémat bylo pouze 2,5 % založeno na postkvantových problémech, naopak ze všech skupinových podpisů bylo až 15 % postkvantových. Nejčastější rodinou v rámci postkvantových atributových schémat byly mřížky s 81 %. Následovaly hashovací funkce. Pro skupinové podpisy byly na prvním místě též mřížky s 82 %. Na druhém místě se však umístila teorie kódování. Nejčetnějším typem atributových schémat bylo atributové šifrovací schéma, jak již bylo zmíněno. Tato šifrovací schémata jsou stavěna na postkvantových problémech mřížek (z 81 %) a hashovacích funkcí. Články obecných postkvantových atributových autentizačních schémat se všechny stavěly na hash funkcích.

S ohledem na zjištěná fakta a informace podané v této kapitole lze tvrdit, že nejrozšířenějším a nejvíce studovaným postkvantovým schématem s ochranou soukromí jsou skupinové podpisy stavěné na teorii mřížek. Následují atributová šifrovací schémata stavěná na problémech mřížek a atributová podpisová schémata na těchto problémech. Pro podpisová schémata obecně na druhém místě následují problémy kódování a pro šifrovací schémata problémy hashovacích funkcí.

4 Současná postkvantová schémata s ochranou soukromí

Tato kapitola je věnována vývoji postkvantových schémat – zejména skupinových podpisů – s ochranou soukromí v posledních letech a současnému stavu techniky. Dále jsou nastíněny směry, kterými se v současné době ubírá výzkum v těchto oblastech kryptografie s ochranou soukromí. V poslední podkapitole 4.5 jsou pak volně porovnány informace získané z publikací postkvantových skupinových podpisů spolu s naměřenými parametry klasických schémat, která byla implementována pomocí knihovny `libgroupsig`.

4.1 Schémata s ochranou soukromí na mřížkách

4.1.1 Skupinové podpisy na mřížkách

První koncept skupinových podpisů na mřížkách byl publikován v [37] (**Gordon *et al.*, 2010**). Jednalo se o statický skupinový podpis (neumožňuje připojení či odebrání členů skupiny), jehož bezpečnost byla prokázána v modelu náhodného orákula (ROM). Velikost tohoto podpisu byla však lineárně závislá na počtu členů skupiny, a tudíž podpis nebyl efektivní pro velké skupiny.

Následně v publikaci [38] (**Camenisch *et al.*, 2012**) došlo ke generalizaci a rozšíření předchozího schématu na atributové tokeny. Toto schéma nabízí lepší anonymitu (CCA oproti CPA)[1], avšak neřeší problém linearity velikosti podpisu k počtu členů skupiny. Schéma je též uváděno jako bezpečné v ROM.

Publikace [39] (**Laguillaumie *et al.*, 2013**) vychází ze schématu [37] a upravuje velikost podpisu a veřejného klíče tak, aby byla logaritmicky závislá na počtu členů skupiny. Jedná se také o statický skupinový podpis. Jeho bezpečnost je založena na problémech SIS (nalezení krátkého nenulového vektoru v dané náhodné mřížce) a LWE (nalezení vektoru v dané náhodné mřížce tak, aby byl blízko daného bodu) v ROM [40].

V publikaci [41] (**Langlois *et al.*, 2014**) se objevují první dynamické prvky skupinových podpisů. Navržené schéma umožňuje lokální revokaci ověřovatelem (*verifier-local revocation* – VLR), tzn. že revokační informace nemusí být zasílány všem členům skupiny, pouze ověřovateli podpisů. Schéma nabízí podpisy o délce logaritmicky závislé na velikosti skupiny. Jeho bezpečnost je také zaručena v ROM.

[1] CCA značí odolnost vůči útoku vybraným šifrovým textem, zatímco CPA značí odolnost proti útoku vybraným otevřeným textem. Kontext se vztahuje k šifrovacímu schématu, které se užívá v rámci skupinových podpisů k ukrytí identity podepisujícího takovým způsobem, aby bylo umožněno trasování jeho podpisu a zároveň zajištěna odpovídající anonymita.

Zajímavostí tohoto schématu je, že nevyužívá šifrovacího schématu k zajištění sledovatelnosti podpisu. Anonymita tohoto schématu nesplňuje CCA ani CPA, ale tzv. „*selfless*“ anonymitu definovanou v [42]. Přestože schéma podporuje revokaci, neumožňuje přijímání nových členů do skupiny.

Schéma publikované v [43] (**Nguyen *et al.*, 2015**) dosahuje menších velikostí veřejného klíče skupiny a vytvořených podpisů, nicméně obě tyto velikosti jsou stále logaritmicky závislé na počtu členů skupiny. Bezpečnost tohoto schématu je také postavena na SIS a LWE v ROM.

Publikace [44] (**Ling *et al.*, 2015**) navrhuje určité úpravy a zlepšení v efektivitě schémat do té doby publikovaných. Přináší zmenšení velikostí podpisů a veřejného klíče, nicméně se stále jedná o logaritmickou závislost na velikosti skupiny, a zaručuje CCA anonymitu. Schéma je bezpečné v ROM.

Schéma v publikaci [45] (**Libert *et al.*, 2016**) využívá tzv. akumulátor založený na Merkleho stromech. Akumulátor umožňuje zhashování množiny vstupů na krátký řetězec konstantní délky, přičemž zachovává možnost efektivně dokázat, že určitý konkrétní vstup byl zahrnut do tohoto hashe. Výsledkem této publikace je návrh schématu skupinového podpisu, které v konstrukci neobsahuje žádnou funkci se zadními vrátky typu GPV, což u předchozích schémat bylo standardní. Namísto této funkce je využito rozšíření Sternova protokolu. Schéma tak podle slov autorů umožňuje efektivnější volbu parametrů. Schéma je též bezpečné v ROM.

Další publikace [46] (**Libert *et al.*, 2016**, 2) přináší nové podpisové schéma na mřížkách, které využívá efektivních protokolů a důkazů s nulovou znalostí. Toto podpisové schéma se dle autorů dá s výhodou využít pro skupinové podpisy a anonymní pověření (*anonymous credentials*). Autoři následně dané schéma aplikují v rámci nového schématu skupinového podpisu. Tento skupinový podpis nabízí efektivní protokol *Join*, díky kterému lze přijímat nové členy skupiny. Protokol podporuje současné přijímání vícero členů v jednom časovém okamžiku a skládá se pouze ze dvou zpráv. Revokace členů však není možná. Schéma je bezpečné v ROM.

Schéma [47] (**Libert *et al.*, 2016**, 3) představuje první schéma skupinových podpisů na mřížkách, které omezuje pravomoce autority otevírání podpisů. Schéma zavádí tzv. „*message-dependent opening*“ (MDO), kdy autorita otevírání podpisů potřebuje k otevření kromě svého soukromého klíče i token od jiné autority (*admitter*). Žádná autorita tedy sama nemůže otevřít jakýkoliv podpis, ale otevření musí být povoleno i druhou autoritou – musí spolupracovat. Schéma je statické, ale autoři tvrdí, že se jejich konstrukce dají aplikovat na existující dynamická schémata. Dále je schéma bezpečné v ROM.

Schéma v publikaci [48] (**Ling *et al.*, 2017**) je prvním plně dynamickým skupinovým podpisem, umožňuje tedy členům skupiny vstupovat a odcházet. Jako základ autoři využili schéma [45], které bylo do té doby nejefektivnějším návrhem

skupinového podpisu na mřížkách. Schéma také produkuje kratší podpisy než [45] a je bezpečné v ROM.

Publikace [49] (**Ling *et al.*, 2018**) přináší částečně dynamické schéma, které tvoří podpisy konstantní délky. Umožňuje dynamické přijímání členů skupiny. Všechna dosavadní schémata mají parametry závislé na (maximálním) počtu členů skupiny N, který bývá definován i u do té doby publikovaných částečně nebo plně dynamických schémat. Zmíněné schéma však tento parametr úplně vypouští a tedy N není součástí *setup* fáze tohoto skupinového podpisu. Velikosti podpisů a parametrů tedy závisí pouze na vybraném bezpečnostním parametru λ. Schéma je založeno na problémech Ring SIS a Ring LWE (využívá ideální mřížky) a jeho bezpečnost je ověřena v ROM.

Schéma v publikaci [50] (**del Pino *et al.*, 2018**) přináší výstupy, které jsou dle autorů až o řád menší než u do té doby publikovaných schémat. Toto schéma bylo autory implementováno, aby byly jejich výstupy podloženy i výsledky a reálnými časy výpočtů. Jedná se o všeobecně první experimentální implementaci skupinového podpisu na mřížkách. Tato implementace dokázala na standardním laptopu provést každou operaci za dobu menší než půl sekundy u CPA anonymní verze tohoto schématu s maximální velikostí skupiny 2^{80}. Schéma je postavené na novém systému důkazů s nulovou znalostí navrženém autory. Efektivita tohoto schématu je dosažená díky využití podpisu na mřížkách se selektivní bezpečností – podpisy ve standardním modelu nejsou vhodné ke tvorbě prakticky využitelných skupinových podpisů kvůli své složitosti. Jedná se o statické schéma bezpečné v ROM.

V publikaci [51] (**Ling *et al.*, 2019**) autoři vylepšili své schéma z [48] o funkcionalitu popiratelnosti (*deniability*), která umožňuje trasovací autoritě produkovat důkazy, že určitý člen skupiny není autorem konkrétního podpisu. Součástí tohoto schématu je i nový důkaz nulové znalosti, který umožňuje dokázat, že se konkrétní šifrový text nedešifruje na konkrétní zprávu.

Schéma [52] (**Ling *et al.*, 2019**, 2) přináší první schéma skupinového podpisu na mřížkách s dopřednou bezpečností. To znamená, že útočníci nejsou schopni uměle konstruovat podpisy, které se vážou k minulým časovým periodám, pokud bude v současné době vyzrazen soukromý klíč některého uživatele. Schéma je bezpečné v ROM a týká se statických skupin.

Publikace [53] (**Xie *et al.*, 2019**) popisuje schéma, které je využitelné pro ochranu soukromí v IoT (*Internet-of-Things*), avšak tvrzení o jeho efektivitě není podloženo implementačními výsledky. Autoři sestavili takové schéma skupinového podpisu, které je plně dynamické. Toto schéma je zároveň prvním schématem postkvantového skupinového podpisu, které zaručuje vyvinění[2]. Velikosti parametrů

[2]Členové skupiny nemohou vytvořit podpis za dalšího nezúčastněného člena skupiny.

veřejného klíče a podpisu jsou však lineárně závislé na počtu členů skupiny. Schéma je bezpečné v ROM.

Schéma publikované v [54] (**Luo *et al.*, 2020**) je druhým schématem, které produkuje podpisy konstantní délky. Využívá jiné základní podpisové schéma (bez využití funkce se zadními vrátky) oproti schématu [49]. Je bezpečné v ROM.

Publikované schéma [55] (**Lyubashevsky *et al.*, 2021**) navazuje na předchozí práci tohoto výzkumného týmu v [50]. Velikosti parametrů tohoto schématu nejsou závislé na velikosti skupiny. Skupině se podařilo až třikrát zmenšit velikost vytvořeného podpisu (na ± 200 KB). Schéma je odpovědí na poměrně velký pokrok v efektivitě a konstrukcích důkazů nulové znalosti. Samotná publikace se pozastavuje nad tím, že i přes tento pokrok u zmíněných důkazů nebyl dosažen žádný větší pokrok v rámci skupinových podpisů, které tyto důkazy využívají. Schéma je statické, bezpečné v ROM. Na rozdíl od schématu, na kterém se zakládá, nebylo implementováno a není podloženo praktickými výsledky.

Schéma publikované v [56] (**Sun *et al.*, 2021**) je plně dynamické schéma skupinového podpisu, které podporuje MDO, tudíž omezuje pravomoci autority otevírání podpisů. Kombinuje schéma z [48] spolu s modelem dvojitého šifrování.

Publikace [57] (**Zhang *et al.*, 2021**) přináší schéma, které podporuje VLR, jakožto nejvíce flexibilní mechanismus revokace. Dále upravuje velikosti klíčů užitých ve schématu tak, aby byly vhodné pro velké skupiny uživatelů. Schéma [41], ze kterého tato publikace vychází, má totiž klíče, jejichž velikost je logaritmická k počtu členů skupiny. Tohoto je dosaženo díky efektivnějšímu skrývání identity podepisujícího a nového protokolu nové znalosti. Schéma je plně dynamické a bezpečné v ROM.

Shrnutí výše zmíněných schémat je zobrazeno v tab. 4.1. Ve sloupci Typ značí S statický podpis, D dynamický (popř. ČD pro částečně dynamický). J znamená, že skupinový podpis umožňuje přijímat nové členy skupiny, R zastupuje revokaci členství. G_{PK} je veřejný klíč skupiny a G_{SK} je soukromý klíč členů skupiny. Porovnání je založeno na bezpečnostním parametru λ a (maximálním) počtu členů skupiny $N = 2^{\ell}$. V případě [52] se uvažuje parametr T jako počet časových period týkajících se dopředné bezpečnosti. Schémata [50] a [55] neobsahují zhodnocení na základě bezpečnostního parametru λ, proto je lze porovnávat jen proti sobě.

Tab. 4.1: Porovnání postkvantových SP založených na mřížkách.

Schéma	**Typ**	**J**	**R**	**Velikost podpisu**	**Velikost G_{PK}**	**Velikost G_{SK}**	**Poznámka**
Gordon *et al.*, [37] 2010	S	✗	✗	$\widetilde{\mathcal{O}}(\lambda^2 \cdot N)$	$\widetilde{\mathcal{O}}(\lambda^2 \cdot N)$	$\widetilde{\mathcal{O}}(\lambda^2)$	–
Camenisch *et al.*, [38] 2012	S	✗	✗	$\widetilde{\mathcal{O}}(\lambda^2 \cdot N)$	$\widetilde{\mathcal{O}}(\lambda^2)$	$\widetilde{\mathcal{O}}(\lambda^2)$	–
Laguillaumie *et al.*, [39] 2013	S	✗	✗	$\widetilde{\mathcal{O}}(\lambda \cdot \ell)$	$\mathcal{O}(\lambda^2 \cdot \ell)$	$\widetilde{\mathcal{O}}(\lambda^2)$	–
Langlois *et al.*, [41] 2014	ČD	✗	✓	$\widetilde{\mathcal{O}}(\lambda \cdot \ell)$	$\widetilde{\mathcal{O}}(\lambda^2 \cdot \ell)$	$\widetilde{\mathcal{O}}(\lambda \cdot \ell)$	VLR
Nguyen *et al.*, [43] 2015	S	✗	✗	$\widetilde{\mathcal{O}}(\lambda + \ell^2)$	$\widetilde{\mathcal{O}}(\lambda^2 \cdot \ell^2)$	$\widetilde{\mathcal{O}}(\lambda^2)$	–
Ling *et al.*, [44] 2015	S	✗	✗	$\widetilde{\mathcal{O}}(\lambda \cdot \ell)$	$\widetilde{\mathcal{O}}(\lambda^2 \cdot \ell)$	$\widetilde{\mathcal{O}}(\lambda)$	–
Libert *et al.*, [45] 2016	S	✗	✗	$\widetilde{\mathcal{O}}(\lambda \cdot \ell)$	$\widetilde{\mathcal{O}}(\lambda^2 + \lambda \cdot \ell)$	$\widetilde{\mathcal{O}}(\lambda \cdot \ell)$	bez funkce se zadními vrátky
Libert *et al.*, [46] 2016	ČD	✓	✗	$\widetilde{\mathcal{O}}(\lambda \cdot \ell)$	$\widetilde{\mathcal{O}}(\lambda^2 \cdot \ell)$	$\widetilde{\mathcal{O}}(\lambda)$	přijímání více členů najednou
Libert *et al.*, [47] 2016	S	✗	✗	$\widetilde{\mathcal{O}}(\lambda \cdot \ell)$	$\widetilde{\mathcal{O}}(\lambda^2 \cdot \ell)$	$\widetilde{\mathcal{O}}(\lambda)$	MDO
Ling *et al.*, [48] 2017	D	✓	✓	$\widetilde{\mathcal{O}}(\lambda \cdot \ell)$	$\widetilde{\mathcal{O}}(\lambda^2 + \lambda \cdot \ell)$	$\widetilde{\mathcal{O}}(\lambda) + \ell$	bez funkce se zadními vrátky
Ling *et al.*, [49] 2018	ČD	✓	✗	$\widetilde{\mathcal{O}}(\lambda)$	$\widetilde{\mathcal{O}}(\lambda)$	$\widetilde{\mathcal{O}}(\lambda)$	parametry nezávisí na N
Ling *et al.*, [51] 2019	D	✓	✓	$\widetilde{\mathcal{O}}(\lambda \cdot \ell)$	$\widetilde{\mathcal{O}}(\lambda^2 + \lambda \cdot \ell)$	$\widetilde{\mathcal{O}}(\lambda) + \ell$	bez funkce se zadními vrátky, popiratelnost
Ling *et al.*, [52] 2019	S	✗	✗	$\widetilde{\mathcal{O}}(\lambda(\log N + \log T))$	$\widetilde{\mathcal{O}}(\lambda^2(\log N + \log T))$	$\widetilde{\mathcal{O}}(\lambda^2(\log N + \log T)^2 \log T)$	dopředná bezpečnost
Xie *et al.*, [53] 2019	D	✓	✓	$\mathcal{O}(N)$	$\mathcal{O}(N)$	–	vyvinění
Luo *et al.*, [54] 2020	S	✗	✗	$\mathcal{O}(\lambda \cdot \log^3 \lambda)$	$\mathcal{O}(\lambda \cdot \log^2 \lambda)$	$\mathcal{O}(\lambda \cdot \log^2 \lambda)$	bez funkce se zadními vrátky
Sun *et al.*, [56] 2021	D	✓	✓	$\mathcal{O}(\ell\lambda^2)$	$\mathcal{O}((\lambda \log \lambda)^2)$	$\widetilde{\mathcal{O}}(\lambda) + \ell$	MDO
Zhang *et al.*, [57] 2021	D	✓	✓	$\widetilde{\mathcal{O}}(n^2)$	$\widetilde{\mathcal{O}}(n)$	$\ell \cdot \widetilde{\mathcal{O}}(n)$	VLR, $N = 2^\ell =$ $= \text{poly}(n)$
del Pino *et al.*, [50] 2018	S	✗	✗	581 KB	123 KB	146 KB	$N = 1$ až 2^{80}
Lyubashevsky *et al.*, [55] 2021	S	✗	✗	203 KB	96 KB	6 KB	přímé porovnání s [50]

Současným problémem výzkumu a vývoje skupinových podpisů na mřížkách je dle [56] fakt, že se obecně většina schémat zakládá na vzoru *encrypt-then-prove* a spoléhá se při tom na protokoly nulové znalosti během dokazování, což limituje efektivitu a bezpečnost schématu. Existují tedy dva směry, kterými se ubírá výzkum právě v této oblasti, přičemž oba směřují k eliminaci úzkého hrdla, které důkazy nulové znalosti přináší.

Prvním přístupem je úplné **odstranění důkazů nulové znalosti** a protokolů nulové znalosti z konstrukcí skupinových podpisů. Příkladem může být publikace [58], kde je k tvorbě skupinového podpisu využita kombinace šifrovacího schématu s přidanými vlastnostmi a speciálního atributově založeného podpisu, což umožňuje vyhnutí se nutnosti využití důkazů nulové znalosti. Dalším příkladem je [59], kde je také využit indexovaný ABS pomocí tzv. struktury bonsajového stromu[3], která umožňuje *hash-and-sign* přístup, a tedy i konstrukce ve standardním modelu.

Druhým přístupem je **zlepšení efektivity důkazů nulové znalosti** a aplikace těchto konstrukcí pod ROM. Příkladem tohoto výzkumného směru je publikace [56].

V současnosti jsou tato postkvantová schémata předmětem poměrně aktivního výzkumu, kde ještě není určen nejlepší a nejefektivnější směr, který by se považoval za optimální, a ani není stanovena postkvantová rodina, která by byla pro skupinové podpisy nejvhodnější. Momentálně se tedy výzkum pohybuje zejména v teoretických úrovních, kdy se tato schémata neimplementují, protože to není v současnosti podstatné (ze všech publikací zmíněných výše bylo implementováno pouze jedno schéma). Ideálně je nejprve nutné sestavit efektivní teoretické schéma tak, aby bylo bezpečné nejen v ROM, ale ve standardním modelu. Všechna výše zmíněná schémata jsou pouze bezpečná v ROM, vyjma [59], které předpokládá bezpečnost ve standardním modelu. Tento návrh je však čistě obecný a nemá popsané velikosti parametrů systému ani v teoretické rovině. Současné bezpečnostní předpoklady s bezpečností ve standardním modelu totiž dělají tato schémata poměrně pomalými a neefektivními, proto nejsou vhodná k implementaci a praktickému využití. Zmíněný problém se momentálně týká i dalších rodin postkvantové kryptografie.

4.1.2 Atributově založené podpisy na mřížkách

První kvantově rezistentní atributový podpis byl navržen v publikaci [61] (**Wang *et al.*, 2014**). Schéma je založeno na problému SIS a je bezpečné ve standardním modelu. Schéma má ale vysoké výpočetní a paměťové nároky, proto je neefektivní.

[3]Struktura řeší některé otevřené problémy v rámci kryptografie založené na mřížkách – umožňuje konstrukci bezestavového podpisového schématu ve standardním modelu, protože odstraňuje potřebu náhodných orákulí, a dále umožňuje konstrukci schémat hierarchického šifrování založeného na identitě (HIBE) bez nutnosti využití bilieárního párování. Poprvé navrženo v [60].

V publikaci [62] (**Mao *et al.*, 2014**) je navrženo schéma atributového podpisu na mřížkách, který využívá techniky bonsajového stromu. Schéma bylo vyvíjeno současně jako [61], proto je v něm též tvrzeno, že se jedná o první kvantově rezistentní atributový podpis. Je založeno na problému SIS.

Publikace [63] (**Wang *et al.*, 2015**) přináší novou konstrukci atributově založeného podpisu. Je efektivnější než do té doby publikovaná schémata. Schéma také přináší lepší ochranu soukromí, protože podpis nevyzrazuje nic o atributech a identitě podepisujícího. Bezpečnost schématu je založena na problému SIS v ROM.

Schéma v publikaci [64] (**Jia *et al.*, 2016**) zaručuje bezpečnost v ROM, nepadělatelnost a perfektní ochranu soukromí. V porovnání s existujícími schématy má kratší délku veřejného klíče a výsledného podpisu. Tato publikace navrhuje převedení daného schématu i na NTRU mřížky, které dosahují ještě lepších výpočetních výsledků. Schéma je založeno na problémech SIS a Ring-SIS.

Schéma publikované v [65] (**Bansarkhani *et al.*, 2016**) přichází s novou konstrukcí ABS schématu, které umožňuje anonymní generování podpisu, pokud je dostatečný počet atributů pokrytý validním pověřením – prahové ABS (*threshold*). Z tohoto schématu následně sestavují další schéma ABS pro expresivní politiku tvořenou operátory AND a OR. Konstruuje taky nový systém na agregaci pověření a výsledné velikosti podpisů jsou lineárně závislé na počtu atributů, podobně jako v té době moderní klasická schémata atributových podpisů. Bezpečnost schématu je dokázána v ROM.

V publikaci [66] (**Kaafarani *et al.*, 2018**) se autoři snaží navrhnout schéma, které by podporovalo širokou třídu neohraničených okruhů (*unbounded circuits*[4]) jako politiky, jak je tomu u klasických ABS schémat založených na bilineárních mapách. Jejich konstrukce je založena na primitivech závazkového schématu, schématu digitálního podpisu a sigma protokolu. Schéma je bezpečné v ROM.

Publikace [67] (**Zhang *et al.*, 2020**) se zaměřuje na generalizaci postkvantových atributových podpisů takovým způsobem, aby podporovaly i konjunkce, disjunkce, prahové predikáty, vyhodnocování polynomů a konjunktivní/disjunktivní normální formy. Bezpečnost je založena na problému SIS a schéma je bezpečné v ROM.

Podle [67] je momentálně výzkum v oblasti postkvantových atributových schémat zaměřen právě na generalizaci. Většina dosavadních schémat totiž podporuje jenom AND, OR, prahové predikáty a neohraničené obvody. Proto se tato publikace zabývá jedním z výzkumných směrů v této oblasti a zaměřuje se na to, aby jejich navržené schéma podporovalo vícero tříd z podpisových politik. Problémem je fakt, že pokud schéma bude více generalizované, bude méně efektivní.

[4]Protože český překlad této konstrukce není dohledatelný, bude pro účel této práce využíván doslovný překlad „neohraničené okruhy“.

Většina těchto zmíněných schémat je čistě teoretická, neobsahuje pseudokód ani konkrétní doporučené množiny parametrů. Publikace také často neobsahují teoretické zhodnocení výpočetní či paměťové náročnosti.

4.1.3 Atributová pověření na mřížkách

Co se týká konkrétních schémat atributových pověření na mřížkách, bylo dohledáno pouze jedno novější schéma [68] (**Li *et al.*, 2021**). Autoři se zaměřují na zaručení plné anonymity a zároveň udržení možnosti veřejné vysledovatelnosti. K řešení tohoto problému vytvořili nové primitivum „vysledovatelná atributově založená anonymní autentizace“ (*traceable attribute-based anonymous authentication* – TABAA). Toto primitivum zaručuje plnou anonymitu, znovu-využitelné pověření, řízení přístupu a veřejnou vysledovatelnost. Tato publikace přímo nenavrhuje postkvantové schéma, ale popisuje, jak by se dalo jejich navržené klasické schéma upravit tak, aby splňovalo postkvantové nároky a postkvantovou bezpečnost.

4.2 Skupinové podpisy na hash funkcích

Hashovací funkce a digitální podpisy na nich zkonstruované jsou založeny na minimálních předpokladech, které jsou velmi dobře prostudované. Zároveň se hashovací funkce využívají téměř ve všech konstrukcích digitálních podpisů, zejména těch, které doplňují zprávu na určitou délku. Mají také poměrně dobrý výkon a praktickou využitelnost oproti schématům, které se potýkají s náročnými výpočty. Je třeba pouze vypočítat hash, což je operace náročností podobnější spíše symetrické kryptografii než kryptografii s veřejným klíčem. Tato jednoduchost, která na jednu stranu vede k efektivitě, může ale na druhou stranu vést k limitacím, co se týká sestavení náročnějších schémat a konstrukcí jako např. skupinových podpisů.

V publikaci [69] (**El Bansarkhani *et al.*, 2018**) je představen úplně první skupinový podpis založený na hash funkcích, zvaný *G-Merkle*. Jedná se o stavový skupinový podpis (*stateful*), což znamená, že každý podpis zapříčiňuje změnu soukromého klíče a je nutné udržovat informace o tom, které jednorázové klíče již byly využity, aby nedošlo k jejich opětovnému použití. Schéma je založené na standardních předpokladech a autoři zaručují bezpečnost ve standardním modelu. Schéma nepotřebuje žádné náročné protokoly nulové znalosti jako např. schémata na mřížkách. Je postaveno pouze na Merklově podpisovém schématu v kombinaci s bezpečnou blokovou šifrou a pseudonáhodnou funkcí. Schéma je statické. Jeho výkon je v publikaci podložen experimentální implementací měřenou v cyklech.

Schéma v publikaci [70] (**Shafieinejad *et al.*, 2019**) představuje konstrukci jednorázového skupinového podpisu založeného na hashích a staví na něm trasovatelné

postkvantové schéma skupinových podpisů, které se dá využít vícekrát. Tato konstrukce využívá tři vrstvy. První vrstva klíčového managementu využívá transverzální design a druhá vrstva generuje veřejný klíč skupiny a propojuje uživatele do skupiny pomocí tzv. *hash pools*. Třetí vrstva je postkvantové podpisové schéma založené na hashích. Toto jednorázové schéma je rozvinuto pomocí Merkleho stromů na vícenásobné schéma. Schéma také odděluje autoritu otevírání podpisů a vydavatele klíčů. Při otevírání podpisů musí spolupracovat vícero členů s těmito pravomocemi. Schéma je statické a bezpečnost schématu záleží na využitých primitivech a typech hash funkcí.

Schéma publikované v [71] (**Buser *et al.*, 2019**) přináší skupinový podpis pod názvem *DGM – Dynamic and Revocable Group Merkle Signature*. Publikace si bere [69] jako základ a tvoří plně dynamické schéma, které podporuje přijímání nových členů skupiny a revokaci zajišťuje pomocí nového typu šifrování (tzv. *symmetric puncturable encryption*), které je však výpočetně náročné. K ověřování podpisu je nutná kooperace mezi manažerem skupiny a ověřovateli. Schéma také snižuje závislost velikosti podpisů na počtu členů ve skupině a další limitace schématu *G-Merkle*.

Publikace [72] (**Yehia *et al.*, 2021**) navrhuje nové schéma pod názvem GM^{MT}, které staví na *G-Merkle* a využívá přístup vícero stromů, který byl považován v publikaci [69] jako nepoužitelný pro tento případ. Konstrukce pomocí vícero stromů umožňuje udržovat stejný veřejný klíč skupiny a zároveň umožňuje skupině růst až do dimenze, která podporuje celkově 2^{64} podpisů pod stejným veřejným klíčem. Revokace využitá v tomto schématu má logaritmickou výpočetní složitost. Manažer skupiny musí udržovat v paměti informace o velikosti lineárně závislé na počtu členů ve skupině. U *DGM* je paměťová náročnost pro manažera skupiny lineárně závislá k počtu celkových podpisů, které systém podporuje. Schéma k tomuto tvrzení uvádí i porovnatelné hodnoty – pro systém podporující 2^{64} podpisů pro 2^{15} členů s 256 bitovou bezpečností potřebuje manažer skupiny u GM^{MT} celkem paměť o velikosti 1 MB, zatímco u *DGM* to je $10^{8,7}$ TB, což je absolutně rozhodující rozdíl. Schéma je dynamické.

4.3 Skupinové podpisy na teorii kódování

Jediné schéma tohoto typu postkvantové rodiny, které je známější a zmiňované v relevantních publikacích, je schéma **Ezerman *et al.*,** původně z roku **2015** [73]. Schéma bylo následně upraveno, rozšířeno a znovu publikováno v roce **2020** [74]. Publikace přináší skupinový podpis, jehož bezpečnost je založena na těžkých problémech z teorie kódování – zejména *McEliece* problém, *Learning Parity with Noise* problém a varianta problému syndromového dekódování. Schéma nedisponuje krátkými parametry, velikosti jeho veřejného klíče a výsledných podpisů jsou lineárně

závislé na počtu členů skupiny N. Autoři však tvrdí, že pokud je schéma využito s praktickými parametry, chová se výrazně efektivněji než schéma v [43], což bylo jedno z nejefektivnějších schémat ve smyslu asymptotické složitosti. V porovnání s tímto schématem má pro průměrnou skupinu 2^8 asi 2300 krát, resp. 540 krát menší veřejný klíč, resp. výsledný podpis. Tato výhoda se zmenšuje s rostoucí velikostí skupiny, nicméně schéma zůstává efektivní i při velikosti skupiny 2^{24}, což je ekvivalentní populaci Nizozemí. První publikace přináší CPA anonymní verzi schématu, pozdější publikace přináší silnější CCA anonymitu za cenu menší efektivity. Schéma je podloženo implementačními výsledky. Je statické a bezpečné v ROM.

4.4 Skupinové podpisy na polynomiálních rovnicích

Základem bezpečnosti problémů založených na řešení polynomiálních rovnic (MPKC) je fakt, že vyřešit soustavu rovnic nad konečným polem je NP-těžký problém. Na tomto základu staví první schéma skupinového podpisu založeného na polynomiálních rovnicích v publikaci [75] (**Yang *et al.*, 2011**). Schéma je dedikované problému elektronických voleb a je specificky tvořeno pro využití v této oblasti. Přináší totiž dvě specifické vlastnosti – speciální nespojitelnost a speciální trasovatelnost. Existence skupinového podpisu je dělena na časové periody. Speciální nespojitelnost znamená, že jsou dané podpisy spojitelné pouze v dané časové periodě a mimo tuto periodu jsou nespojitelné. Sčítací autorita tedy může zjistit, zda nebyly některé hlasy duplikované předtím, než je otevře. Speciální trasovatelnost je specificky vytvořena pro sčítací a dozorový orgán elektronických voleb – tyto dva orgány musí spolupracovat, aby mohly otevřít podpis a zjistit identitu podepisujícího či voliče. Schéma je efektivní pro velké skupiny, protože velikosti podpisů a efektivita výpočtů nejsou ovlivněny počtem členů ve skupině. Operace podepisování, verifikace a otevření jsou konstantní, stejně jako délka podpisu. Délka klíče manažera je lineárně závislá na počtu členů ve skupině, avšak délka soukromého klíče členů skupiny je vždy konstantní. Schéma je statické.

Novou publikací v této oblasti je publikace [76] (**Kundu *et al.*, 2021**). Tato publikace užívá identifikační protokol a podpisové schéma založené na polynomiálních rovnicích (*Rainbow*) jako základní stavební kameny. Velikosti podpisů ani veřejný klíč tohoto schématu nezávisí na velikosti skupiny, pouze na velikosti bezpečnostního parametru. Schéma řeší problémy jediného předchozího MPKC schématu [75]. Schéma je částečně dynamické, umožňuje přijímat nové členy skupiny pomocí operace *Join*.

4.5 Porovnání klasických a postkvantových schémat

Cílem této kapitoly je popsat srovnání skupinových podpisů založených na postkvantových rodinách a klasických skupinových podpisů, které jsou založeny na teorii čísel. Přímé srovnání nelze provést zejména z důvodu nedostatku implementací, knihoven, chybějícím informacím v publikacích, popř. rozdílnému využitému hardwaru. Volné srovnání napříč schématy je také velmi složité, jelikož skupinové podpisy nabízí určitou flexibilitu ve svých konstrukcích, a tak lze provádět kompromis mezi paměťovou a časovou náročností. Navrhovaná schémata se často velmi liší ve svých primitivech, konstrukcích, podpůrných schématech, bezpečnostních předpokladech (typ anonymity a dalších požadavků), bezpečnostním modelu, uváděných bezpečnostních parametrech aj.

4.5.1 Současné problémy výzkumu a použití SP

Práce [77] z prosince 2019 shrnuje současný stav výzkumu a využití v oblasti skupinových podpisů a podtrhuje současné problémy. Tvrdí, že nejpodstatnější je pochopení a identifikování konkrétních oblastí použití. Tyto oblasti budou základem k určení modelů hrozeb, požadavků na efektivitu a dalších funkcí. Momentálně jsou totiž oblasti výzkumu a využití skupinových podpisů uvízlé v pomyslné smyčce, kdy na sobě vzájemně závisí. Aby mohly být vytvořeny praktické a efektivní skupinové podpisy, musí být dobře stanovené požadavky na oblast, ve které budou tato schémata použita. Aby bylo možné tyto oblasti použití vůbec prozkoumat, musí existovat implementace skupinových schémat, které by bezpečnostní inženýři mohli jednoduše využít ve svých projektech. Bohužel momentálně existuje velmi málo implementací skupinových podpisů. Nová schémata skupinových podpisů jsou publikována poměrně často, ale bez konkrétního cíle, a oblasti využití nejsou prozkoumávány právě kvůli tomu, že tato schémata nejsou implementována a pravidelně se publikují nová. O ukončení této smyčky se pokusila knihovna `libgroupsig`[5]. Sami autoři ve své publikaci [78] zmiňují nedostatek implementací skupinových podpisů. Sice existuje malé množství implementací, ale jsou buď neudržované nebo implementují pouze jedno konkrétní schéma jako *proof-of-concept*, které se nehodí k obecnějšímu využití. Proto se autoři rozhodli implementovat tři hlavní schémata skupinových podpisů z let 2004–2006 ([79, 81, 80]) v jedné knihovně pod společným API. Publikace [77] tvrdí, že tato snaha o unifikaci implementací skupinových podpisů do jedné knihovny nezískala zájem vědecké veřejnosti. Tuto domněnku lze považovat za pravdivou, protože dle Github stránky[6] této knihovny je využívána pouze v jednom projektu[7], který se

[5] `https://github.com/IBM/libgroupsig`

[6] `https://github.com/IBM/libgroupsig/wiki/Who-Is-Using-libgroupsig`

[7] `https://www.ict4cart.eu/`

zaměřuje na automatizované řízení. Od té doby byla knihovna rozšířena o čtyři další, modernější schémata z let 2016–2021 ([82, 83, 84, 85]). Základní knihovna je psána v jazyce C, ale autoři poskytují i wrappery na další programovací jazyky (Python, NodeJS, Java). Tyto wrappery však nepodporují nejnovější přidaná schémata. Je tedy možné, že se `libgroupsig` bude do budoucna objevovat v různých projektech, nicméně tato snaha je stále pouze v rámci schémat, která jsou založena na teorii čísel, a tedy nejsou kvantově rezistentní. Pro postkvantová schémata je tento problém s nedostatkem implementací ještě markantnější a žádné hnutí k unifikaci a obecně implementování více postkvantových skupinových podpisů momentálně neexistuje.

4.5.2 Referenční implementace klasických schémat SP

Pro představu byla pomocí knihovny `libgroupsig` a jejího Python wrapperu `pygroupsig` naprogramována schémata klasických skupinových podpisů [81, 83, 84, 85] a byla změřena jejich výpočetní náročnost a paměťové nároky (velikost veřejného klíče skupiny, soukromého klíče uživatelů a podpisu). Do skupin bylo přidáno 100 uživatelů a byly podepisovány zprávy o velikosti 11 B a 5 MB. Shrnutí výsledků je níže v tab. 4.2. Konkrétní výsledky a doplňující informace jsou k dispozici v příloze B na obr. B.1.

Tab. 4.2: Přibližné parametry klasických skupinových podpisů v `libgroupsig`.

Podpis [B]	**G_{PK} [B]**	**G_{SK} [B]**	**Setup [ms]**	**Join [ms]**	**Sign [ms]**	**Ver [ms]**	**Open [ms]**
stovky	stovky	stovky	jednotky	stovky	jednotky	jednotky	desítky

Změřené implementace jsou sice všechny dynamické, ale budou považovány za základ k porovnání praktické využitelnosti postkvantových schémat, jelikož cílem `libgroupsig` je poskytnout skupinové podpisy přichystané k použití v reálných projektech.

Z publikací postkvantových schémat skupinových podpisů zmíněných v této kapitole byly získány dostupné informace o výpočetní náročnosti operací a velikosti klíčů a podpisů.

4.5.3 Zhodnocení efektivity SP založených na mřížkách

Publikace [43] dosahuje odhadem pro bezpečnostní parametr $\lambda = 2^8$ a počet členů $N = 2^{10}$ G_{PK} o velikosti 2 GB, G_{SK} v řádu desítek GB a podpis v řádu stovek MB. Publikace [44] dosahuje pro stejné parametry G_{PK} v řádu desítek MB, G_{SK} v řádu desítek KB a podpis o velikosti 1 GB. Tyto informace jsou převzaty z publikace [45], která tato schémata porovnává se svým navrženým skupinovým podpisem. Ten přináší parametry G_{PK} v řádu jednotek MB, G_{SK} v řádu jednotek KB a podpis v řádu

desítek MB. Časová náročnost není v publikacích popsaná, lze však předpokládat, že při takové velikosti parametrů se bude jednat o řádově delší časy než klasická schémata z knihovny `libgroupsig`. Počet členů skupiny se v tomto případě liší, nicméně implementace z `libgroupsig` jsou efektivnější i v případě větších skupin, jelikož mají konstantní velikosti parametrů. Lze tedy tvrdit, že publikace [43, 45] nejsou prakticky využitelné. Velikosti parametrů publikace [45] jsou nižší, ale stále poměrně velké, a problémem může být trvání operace generování klíčů, jelikož se jedná o statické schéma a je nutné vygenerovat všech N soukromých klíčů členů skupiny. Proto se dá toto schéma také považovat za špatně prakticky využitelné.

Velikosti výše uvedených parametrů byly určeny odhadem v publikaci [45], avšak publikace [50] přináší první postkvantové schéma skupinového podpisu, které je podloženo implementačními výsledky. Proto se lze podívat na konkrétní parametry a časovou náročnost operací. Pro maximální skupinu o velikosti $N = 2^{80}$ jsou velikosti G_{PK}, G_{SK} i podpisu v řádu stovek KB. Všechny operace trvají v řádu stovek milisekund. Parametry tohoto schématu vypadají zajímavě, avšak schéma dosahuje těchto parametrů jen díky slabší bezpečnosti (selektivní bezpečnost u využitého podpisového schématu).

Konkrétní parametry jsou uvedeny i v publikaci [55], která pochází od stejných autorů jako [50]. Toto schéma nebylo implementováno a prezentuje pouze novou a vylepšenou konstrukci, která díky menším optimalizacím zkracuje délku G_{PK} na desítky KB, G_{SK} na jednotky KB a délku podpisu asi 3 krát, stále však v řádu stovek KB. Délky parametrů a časová náročnost operací však nezávisí na velikosti skupiny. Z hlediska zaručené bezpečnosti, nedynamičnosti a velikosti parametrů však nemusí být vhodné k praktickému využití.

Z dostupných informací a faktů popsaných v této podkapitole je tedy zřejmé, že současná existující schémata skupinových podpisů založená na mřížkách zatím nejsou dostatečně efektivní, aby bylo výhodné je implementovat a využívat. Proto v této oblasti probíhá aktivní výzkum. Je možné že standardizace postkvantových podpisů NIST pomůže konstrukcím složeným z těchto primitiv pohnout se tím správným směrem. Je také nutné vyčkat, než výzkumné týmy věnující se mřížkám začnou ve svých konstrukcích skupinových podpisů využívat nové důkazy nulové znalosti, ve kterých byl v posledních letech uskutečněn velký pokrok [86, 87, 88, 89, 90, 91, 92].

4.5.4 Zhodnocení efektivity SP založených na hash funkcích

Určení paměťových nároků na hashově založené skupinové podpisy je poměrně náročné – je nutné uvažovat počet členů skupiny N a maximální počet podpisů, které může jeden člen vytvořit B. Veřejný klíč skupiny a soukromý klíč uživatele také závisí na užité hashovací funkci a jednorázovém podpisovém schématu (nejčastěji

Winternitzův jednorázový podpis – kap. 2.1.2). Je nutné také brát v potaz fakt, že při podepisování je vedle podpisu nutné zveřejnit i autentizační cestu stromem, která je základem pro správné ověření tohoto podpisu. Protože tyto parametry a dílčí schémata lze zvolit různými způsoby a v konstrukcích klasických schémat skupinových podpisů nic ekvivalentního neexistuje, budou paměťové nároky porovnávány jen u schémat, kde jsou tyto parametry výslovně uvedeny.

V publikaci [65] jsou výpočetní nároky uvedeny v procesorových cyklech. Tyto cykly byly převedeny na milisekundy za účelem lepšího porovnání. Navržené schéma bylo autory implementováno za využití rozšířeného Merkleho podpisového schématu (XMSS) a zkráceného jednorázového Winternitzova podpisu (W-OTS+). Pro $N = 64$ a $B = 256$ fáze nastavování a počítání stromu trvá 11 sekund. Nejnáročnější částí operace je však operace sestavování stromu, která probíhá pouze na straně manažera skupiny. Podpis a verifikace zabere jednotky milisekund. Otevírání podpisu setiny milisekundy. Implementace však předpokládá, že strom je veřejně a bezpečně dostupný, tudíž výpočty autentizační cesty nejsou do implementace započítány. Jedná se však o celkem náročnou operaci, proto může být efektivita tohoto podpisu zkreslená.

Publikace [71] konstruuje ze základu [65] dynamické schéma umožňující revokaci a příjem nových členů. Velikost podpisu na rozdíl od [65] závisí pouze na bezpečnostním parametru, tudíž je konstantní. Veřejný klíč kromě bezpečnostního parametru závisí na počtu revokovaných členů. Podpisy mají velikost v řádu jednotek KB. Údaje týkající se časové náročnosti všech operací nejsou v publikaci dostupné.

Nejnovějším schématem v této oblasti je [72] z podzimu roku 2021. Jedná se o dynamické schéma. Velikost podpisu závisí na bezpečnostním parametru, který udává velikosti dalších parametrů. V nejobecnějších případech při maximálním počtu podpisů pod jedním klíčem 2^{64} a počtu členů skupiny až 2^{16} má výsledný podpis délku v jednotkách KB. Paměťové nároky na člena skupiny jsou navýšeny kromě nutnosti uložení autentizačních cest a klíčů v listech stromu ještě o další parametry, aby bylo možné dynamicky generovat další podstromy. Konkrétně se jedná o stovky KB při 256 bitové bezpečnosti. Paměťové nároky na manažera skupiny jsou oproti [71] výrazně sníženy díky jinému revokačnímu mechanismu. Časová náročnost pro zakládajících $N = 2^6$ členů skupiny s $B = 2^{10}$ podpisy (možnost rozšíření obou parametrů) a 256 bitovou bezpečností je asi 568 sekund pro vytvoření stromu a vygenerování všech potřebných parametrů. Z toho nejnáročnější operací je generování veřejného klíče manažerem skupiny (566 s). Podpis a verifikace zaberou jednotky milisekund. Otevření podpisu trvá setiny milisekundy. Tyto operace jsou porovnatelné, ne-li rychlejší než u klasických schémat.

Skupinové podpisy založené na hash funkcích by mohly být prakticky využitelné pro konkrétní oblasti ochrany soukromí. Většina výpočetních nároků leží na

straně manažera skupiny, který staví strom. Proto pokud manažer skupiny stojí např. v pozici cloudu s velkým výpočetním výkonem, náročnost operace není tolik podstatná. Ostatní operace jsou dle zmíněných publikací a porovnání s klasickými schématy efektivní. Paměťové nároky jsou poměrně dobré v porovnání s ostatními postkvantovými schématy, pokud bereme ohled na dynamičnost schématu.

Hashovací funkce jsou dobře prostudované a známé a struktura Merkleho stromů také. Využívají jednoduchá primitiva a nabízí určitou flexibilitu. Některá tato schémata jsou také deklarovaná za bezpečná ve standardním modelu. Výhodou může být i velikost veřejného klíče, jakožto hashe kořenu stromu. Proto by již v dnešní době a v současném stavu techniky mohly nalézt praktické uplatnění.

Schémata založená na hashích se také mezi sebou tolik neliší v konstrukci jako např. mřížková schémata. Proto je výzkum v této oblasti poměrně přímočarý a týká se např. vylepšení efektivity vypočítání autentizační cesty stromem, způsobu ukládání revokačních informací a jejich zahrnutí v rámci verifikační operace aj. Mřížková schémata se mezi sebou liší prakticky od základu – jsou například využity různé podpisy a různé důkazy nulové znalosti, oba založené na rozdílných problémech napříč schématy.

4.5.5 Zhodnocení efektivity SP založených na teorii kódování

Schéma v [73] je statické schéma podložené implementací. V CPA anonymní verzi tohoto skupinového podpisu je pro skupinu o $N = 2^8$ členech průměrná velikost podpisu okolo stovky KB a veřejný klíč dosahuje velikosti okolo poloviny MB. Podepisování a verifikace se pohybují v jednotkách milisekund a operace otevírání v desítkách milisekund, což je porovnatelné s klasickými schématy z [78]. Časově nejnáročnější je operace generování klíčů, která trvá 14 sekund pro skupiny do 2^{12} členů. S rostoucí velikostí skupiny se zvětšují veřejné klíče a výsledné podpisy, zvyšují se i časy jednotlivých operací. Operace otevírání podpisů zůstává stejně efektivní. CCA anonymní verze schématu zvětšuje všechny parametry, zejména téměř zdvojnásobuje délku veřejného klíče a tím pádem i dvojnásobně prodlužuje operaci generování klíčů.

Při porovnání tohoto schématu se současnými klasickými schématy je nevýhodou zřejmě fakt, že je statické a téměř všechny jeho parametry závisí na velikosti skupiny. Pokud ale skupina není extrémně velká (do 2^{12}), nabízené parametry a efektivita mohou být přijatelné pro určité oblasti reálného využití.

4.5.6 Zhodnocení efektivity SP založených na polynomiálních rovnicích

Schéma publikované v [76] je také podloženo implementačními výsledky. Pro 80 bitovou bezpečnost a $N = 100$ členů skupiny toto schéma přináší poměrně malé velikosti parametrů – G_{PK} v řádu desítek KB, podpis také a G_{SK} dokonce v řádu desítek B. Tyto poměrně krátké velikosti klíčů jsou však vykoupeny značnou časovou náročností. Generování klíčů dle implementace autorů zabere necelou sekundu, podepisování krátké zprávy až 100 sekund a její verifikace 35 sekund. Z důvodu náročnosti těchto operací se schéma nejeví jako prakticky využitelné. Je však zajímavé, že např. oproti rodině hash funkcí jsou výpočetní nároky na schéma obrácené – operace generování klíčů je nejkratší.

4.5.7 Shrnutí

V kap. 4.5 byly volně porovnány novější publikace a schémata skupinových podpisů, které obsahovaly alespoň některé reálné parametry, které bylo možné zhodnotit. Z celkového počtu asi 40 schémat, které byly v kap. 4 zmíněny a popsány, mělo pouze 6 schémat implementačně podloženou efektivitu. Většina schémat byla čistě teoretická a neobsahovala ani pseudokód, ani návrhy parametrů a odhadované délky podpisů či operací, ale velmi často bylo autory přesto konstatováno, že se jedná o velice efektivní schémata.

Z omezených dostupných informací byly extrahovány ty, které byly podobné podmínkám naměřených implementací klasických schémat (podobná velikost skupiny, délka zprávy, hardware, bezpečnost). Pokud publikace obsahovaly časové parametry, tyto parametry byly nejčastěji měřeny na moderních počítačových procesorech, stejně tak jako schémata implementovaná pomocí knihovny `libgroupsig`.

Na základě získaných informací a faktů popsaných v této kapitole lze tvrdit, že klasická schémata skupinových podpisů nelze bezproblémově nahradit postkvantovými při nynějším *state of the art*. Velikosti parametrů postkvantových schémat se stále pohybují v úplně jiných dimenzích než klasické parametry. Efektivita a rychlost operací je však u některých schémat porovnatelná. Lze uvažovat o aplikaci postkvantových skupinových podpisů v oblastech, kde paměťová náročnost není tak kritická. Např. skupinové podpisy založené na hash funkcích by mohly najít uplatnění už v současné době. Je však nutné podrobně znát požadavky na konkrétní oblasti použití, aby byla aplikace těchto schémat do reálných systémů výhodná a bezproblémová. Výhodou rodiny hash funkcí nad všemi ostatními postkvantovými rodinami je zejména fakt, že se zakládají na známých a dobře prostudovaných primitivech, tudíž již existuje podpora a efektivní implementace těchto primitiv v knihovnách.

Výhodou je i často deklarovaná bezpečnost některých schémat ve standardním modelu, což jiné rodiny postkvantové kryptografie při udržení efektivity nemohou nabídnout. Do budoucna bude jistě očekávání směrováno zejména k rodině mřížek, kde v současné době probíhá asi nejvýznamnější a nejrychlejší pokrok.

5 Implementace postkvantových schémat

Díky projektu postkvantové standardizace NIST existuje poměrně velké množství postkvantových šifrovacích schémat a podpisů. Aby schéma mohlo být zvažováno ke standardizaci, musí vědecká skupina doložit dvě implementace v programovacím jazyce C – referenční a optimalizovanou implementaci. NIST požaduje, aby tyto návrhy byly co nejméně závislé na licencích třetích stran, proto je většina publikovaných kryptosystémů licencována jako volné dílo (*public domain*). Toto umožňuje bezproblémové užívání zmíněných kódů a návrhů, jejich derivací a úprav v rámci dalších vědeckých a soukromých projektů.

Kromě jednotlivých implementací autorů těchto schémat existuje i poměrně velké množství knihoven. Ty sdružují implementace jednotlivých kandidátních schémat v různých jazycích a zjednodušují práci s nimi.

Mezi nejobsáhlejší udržované a dostupné knihovny standardních postkvantových schémat patří zejména knihovny `PQClean`[1], `nistpqc`[2], `pqcrypto`[3] a `liboqs`[4]. Výpis postkvantových schémat obsažených v těchto knihovnách je zobrazen v tab. 5.1.

Tab. 5.1: Dostupné knihovny schémat postkvantové kryptografie.

PQ schéma	**typ**	`PQClean`	`nistpqc`	`pqcrypto`	`liboqs`
prog. jazyk	–	C	C	Py	C, ...
McEliece	Enc	✓	✗	✓	✓
Kyber	Enc	✓	✓	✓	✓
NTRU	Enc	✓	✓	✓	✓
Saber	Enc	✓	✓	✓	✓
FrodoKEM	Enc	✓	✓	✓	✓
HQC	Enc	✓	✗	✓	✓
SIKE	Enc	✗	✓	✗	✓
NTRU Prime	Enc	✗	✓	✗	✓
BIKE	Enc	✗	✗	✗	✓
Dilithium	Sig	✓	✗	✓	✓
Falcon	Sig	✓	✗	✓	✓
Rainbow	Sig	✓	✗	✓	✓
SPHINCS+	Sig	✓	✗	✓	✓
Picnic	Sig	✗	✗	✗	✓

[1] `https://github.com/PQClean/PQClean`
[2] `https://github.com/post-quantum/nistpqc`
[3] `https://github.com/kpdemetriou/pqcrypto`
[4] `https://github.com/open-quantum-safe/liboqs`

Knihovna `PQClean` implementuje všechna schémata finalistů a alternativních kandidátů třetího kola standardizace v programovacím jazyce C. Cílem této knihovny je implementace samostatných schémat, která jsou kvalitně naprogramovaná a otestovaná, dají se integrovat do dalších knihoven jako `liboqs`, vyšších protokolů, využít k benchmarkingu, evaluaci bezpečnosti či formální verifikaci.

Knihovna s názvem `nistpqc` implementuje pouze schémata šifrovací/KEM části standardizace v jazyce C, avšak každé schéma je implementováno v různých variantách. Je udržována a v průběhu kol byli odstraňováni neúspěšní kandidáti.

Knihovna `pqcrypto` je nástavba knihovny `PQClean`, která umožňuje jednoduchou a komprehenzivní práci s obsaženými schématy v programovacím jazyce Python. Obsahuje proto stejná schémata jako knihovna tvořící její jádro.

Knihovna s názvem `libqos` patří pod projekt *Open Quantum Safe*. Tento aktivní open source projekt si klade za cíl podporovat vývoj a prototypování kvantově rezistentních aplikací. Má dva hlavní výzkumné směry – postkvantovou knihovnu a prototypy integrací protokolů a aplikací. Základní knihovna je psána v jazyce C, jsou však dostupné *wrappery* na další programovací jazyky (C++, Go, Java, .NET, Python, Rust). Některá schémata přebírá rovnou z implementací podaných ke standardizaci, jiná z knihovny `PQClean`. Dále umožňuje integrace s OpenSSL a OpenSSH. Implementuje mj. hybridní výměnu klíčů a postkvantovou asymetrickou autentizaci v TLS 1.3, postkvantové algoritmy v generování certifikátů X.509, postkvantovou a hybridní výměnu klíče v SSH.

Dalším slibným projektem je *PALISADE*[5]. Do tohoto projektu přispívá mnoho univerzit, institutů a společností. Narozdíl od předchozích knihoven se jedná o knihovnu, která implementuje stavební kameny kryptografie založené na mřížkách a *state-of-the-art* plně homomorfní šifrovací schémata využívající mřížky, jako je:

- *Brakerski/Fan–Vercauteren* (BFV) schéma,
- *Brakerski–Gentry–Vaikuntanathan* (BGV) schéma,
- *Cheon–Kim–Kim–Song* (CKKS) schéma,
- *Ducas–Micciancio* (FHEW) schéma,
- *Chillotti–Gama–Georgieva–Izabachene* (TFHE) schéma.

Knihovna také obsahuje rozšíření k *multiparty* výpočtům s těmito schématy. V dřívějších verzích knihovna podporovala i digitální podpis založený na mřížkách, schémata šifrování založená na identitě a atributově založená šifrovací schémata. Tyto podknihovny byly odsunuty do vlastních repozitářů a postrádají rozšířenější dokumentaci. Na tomto projektu je aktivně pracováno.

Jedinou známější knihovnou ke tvorbě skupinových podpisů je `libgroupsig`. Tato knihovna byla blíže popsána v kap. 4.5, kde byla také prakticky využita.

[5] `https://gitlab.com/palisade/palisade-release`

Momentálně však neexistuje žádná knihovna, která by podporovala implementace postkvantových schémat skupinových podpisů či atributových podpisů.

5.1 Postkvantové podpisy s knihovnou pqcrypto

Jako součást průzkumu postkvantových schémat a knihoven byly implementovány vybrané varianty postkvantových digitálních podpisů, které jsou finalisty či alternativními kandidáty třetího kola standardizace. Všechna vybraná schémata mají „*NIST security level*“ hodnoty 5. To značí, že prolomení daných schémat je alespoň tak výpočetně náročné, jako je prolomení blokové šifry s 256 bitovým klíčem (např. AES-256) skrze útok hrubou silou zvaný *exhaustive key search* [93].

K implementaci vybraných podpisů byla využita knihovna `pqcrypto` v programovacím jazyce Python. Tato knihovna je uživatelsky přívětivá a umožňuje jednoduchou práci s podporovanými schématy. Následně byl změřen čas provádění úkonů v rámci těchto schémat (generování klíčů, podepisování, verifikace) za využití vestavěného modulu `time` a celkové paměťové nároky s využitím knihovny `guppy3`. Podepisovaná zpráva měla velikost 11 B. Podrobnější informace k měření a tabulku s hlavními výsledky lze nalézt v příloze C a na obr. C.1. Na obr. C.2 je přiložena i celková tabulka všech implementovaných podpisů o jiných bezpečnostních úrovních a při využití jiné délky podepisované zprávy (1024 B řetězec).

Konkrétně byly implementovány podpisy ze tří rodin postkvantové kryptografie, a to *Dilithium*, *Falcon* (mřížky), *Rainbow* (polynomiální rovnice) a *SPHINCS+* (hashovací funkce). Přesněji se v tab. C.1 jedná o následující varianty:

- *Dilithium 5*
- *Falcon 1024*
- *Rainbow V:*
  - *Classic*,
  - *Cyclic* – menší veřejný klíč,
  - *Cyclic Compressed* – menší veřejný i soukromý klíč,
- *SPHINCS+:*
  - *s/f* – délka podpisu,
  - *robust/simple* – rychlost výpočtu, simple využívá ROM,
  - *Haraka 256, SHA-256, SHAKE256* – využívaná hashovací funkce.

Varianta *SPHINCS+* s hashovací funkcí *Haraka* může dle [94] dosahovat pouze úrovně bezpečnosti 2 z důvodu *meet-in-the-middle* útoku a výpočtu kolizí, nicméně v uvádění této hodnoty se zdroje liší – např. [95] uvádí úroveň 5, proto je pro toto měření její 256 bitová verze započítána mezi úrovně 5.

Z tab. C.1 vyplývá, že postkvantový podpis *SPHINCS+* založený na hash funkcích disponuje nejkratším veřejným klíčem (64 B) a krátkým soukromým klíčem (128 B), zatímco vytvořené podpisy jsou nejdelší z testovaných schémat. Nejdelší operací je u tohoto schématu podepisování, zejména u variant *s*. Tato varianta tedy nabízí kratší výsledný podpis (29,7 KB oproti 49,8 KB u varianty *f*), ale jeho vytvoření trvá řádově delší dobu (např. podepisování za využití *Haraka 256s robust* $\approx 4{,}6$ s oproti *Haraka 256f robust* $\approx 0{,}5$ s). Nároky na paměť těchto schémat byly podobné a pohybovaly se spíše na nižším okraji spektra u porovnávaných schémat.

Schéma *Rainbow* využívá dlouhé veřejné klíče. Cyklický mód velikost tohoto klíče snižuje, nicméně i tak se jedná o nejdelší veřejné klíče z testovaných schémat. Soukromé klíče klasického a cyklického módu jsou také největší mezi porovnávanými schématy. Cyklický kompresní mód *Rainbow* však dokáže velikost soukromého klíče snížit z 1,49 MB na 64 B. Předností všech typů schémat *Rainbow* jsou pak krátké výsledné podpisy o velikosti 212 B. Operace generování klíčů a ověřování jsou časově nejnáročnější ze všech porovnávaných schémat. U cyklického kompresního módu je časově náročná i operace podepisování, proto celkové provedení tohoto schématu trvalo nejdéle ze všech změřených, a to 7,6 s. Paměťové nároky těchto schémat byly nejvyšší, zejména pro klasický mód, který pracuje s velkou délkou klíčů.

Schémata *Dilithium5* a *Falcon 1024* založená na mřížkách se vyznačují nejrychlejšími operacemi jak generování klíčů a podepisování, tak i verifikace. Délky klíčů nejsou tak velké jako u *Rainbow* a výsledný podpis není tak dlouhý jako u *SPHINCS+*. Při porovnávání těchto dvou schémat oproti sobě je zřejmé, že *Falcon 1024* nabízí kratší klíče a výsledný podpis – tím pádem i nižší nároky na paměť, avšak čas provedení tohoto schématu je delší jako u *Dilithia5*.

Celkově je z výsledků implementací a měření zřejmé, že každé schéma má určité výhody, které jsou vykoupeny jinými nevýhodami. Nejlepší cestou se tedy zdají schémata založená na mřížkách, která jsou nejrychlejší, paměťově nejméně náročná a klíče ani podpisy nemají extrémní velikosti.

5.2 Postkvantová schémata s ochranou soukromí

Jak již bylo v předchozích kapitolách řečeno, současný stav výzkumu postkvantových schémat s ochranou soukromí je technicky teprve v začátcích. Organizace NIST vypsala postkvantovou soutěž o standardizaci digitálních podpisů a šifrovacích schémat na začátku roku 2017, přičemž výsledky jsou očekávány v roce 2022. Těmto základním typům schémat je tudíž věnovaná největší pozornost postkvantových výzkumníků.

Kryptografie s ochranou soukromí získává větší trakci také až v posledních letech, ačkoliv je koncept PETs známý už několik desítek let. Např. organizace NIST

pořádá PET konference od roku 2019. Dalším příkladem jsou Evropský sbor pro ochranu osobních údajů a Agentura Evropské unie pro kybernetickou bezpečnost (ENISA), kteří společně publikovali v roce 2019 dokument, jenž má pomoci implementovat využití pseudonymů, důkazů znalosti a *secure multiparty computation* v rámci zdravotnictví a kybernetické bezpečnosti [96].

Výzkum průniku těchto dvou nově populárních okruhů kryptografie tudíž zatím není tak aktivní. Do dnešní doby existují maximálně desítky publikovaných schémat, co se týká postkvantových skupinových podpisů a atributových podpisů, popř. atributových autentizačních schémat, jak ostatně průzkum v kap. 3 naznačuje. Většina těchto schémat nebyla prakticky implementována a žádné knihovny podporující vybrané typy postkvantových PET schémat v době psaní této práce neexistují.

Proto, aby mohla být teorie sepsaná v této práci podložena i praktickými výsledky, byli emailem oslovováni výzkumníci ohledně implementací jejich publikovaných postkvantových skupinových a atributových podpisů. Celkem bylo o implementace požádáno 21 výzkumných týmů. Odpovědi se dostalo od šesti z nich. Byly získány dvě implementace postkvantových skupinových podpisů – jedna založená na mřížkách [50] a druhá na teorii kódování [73]. Ostatní čtyři odpovědi byly negativní, dvě z nich odkazovaly na jiné implementace postkvantových schémat, jako jsou kruhové podpisy. Následně bylo implementováno schéma hashového skupinového podpisu *G-Merkle* podle konstrukce v publikaci [69].

Implementace získané od výzkumníků byly zprovozněny, prostudovány a v rámci možností upraveny, aby je bylo možné změřit a porovnat jak mezi sebou, tak oproti implementovanému SP založenému na hash funkcích.

5.2.1 Implementace SP založeného na hash funkcích

Schéma skupinového podpisu bylo implementováno na základě navržené konstrukce všeobecně prvního a stavového skupinového podpisu založeného na hashích, publikovaného roku 2018 autory R. El Bansarkhani a R. Misoczki pod názvem *G-Merkle: A Hash-Based Group Signature Scheme From Standard Assumptions* v [69]. Skupinový podpis vznikl na základě výzkumné spolupráce mezi Technickou univerzitou Darmstadt v Německu a Intel Corporation v USA.

Schéma využívá konstrukce, kdy jeden společný Merkleho strom sdílí všichni členové skupiny a kořen tohoto stromu představuje veřejný klíč skupiny G_{PK}. Každý z N členů skupiny má k dispozici celkem B podpisů. Merkleho strom má potom $N \cdot B$ listů, které představují hashe jednotlivých jednorázových veřejných klíčů členů skupiny. Výška stromu h je tedy dána počtem listů $N \cdot B = 2^h$. K hashům veřejných klíčů je přidán šifrovaný identifikátor, který zajišťuje trasovatelnost podpisu, pokud je nutné jej otevřít. Schéma je složeno pouze z jednoho stromu, tudíž je statické

a nepodporuje revokaci členství ve skupině. V publikaci autoři tvrdí, že nelze využít přístup s vícero stromy[6], který by vedl na větší možný počet podpisů pod jedním veřejným klíčem a umožňoval by dynamičnost, neuvádí však důvody, proč by tomu tak mělo být. Tím je tedy limitována maximální velikost stromu na 2^{20} listů. Na konci roku 2021 byl však publikován skupinový podpis [72], který bere *G-Merkle* jako základ a převádí ho na dynamický skupinový podpis s možností revokace právě díky vícestromové konstrukci, která byla dříve považována za nepoužitelnou. Maximální počet podpisů je díky této konstrukční úpravě v novém schématu rozšířen až na 2^{64} při zachování stejné úrovně bezpečnosti.

Entity schématu

Schéma má celkem tři entity, a to:

- **člen skupiny** (*group member*, U) – pro větší bezpečnost generuje své vlastní jednorázové páry klíčů, pamatuje si stav, podepisuje zprávu jednorázovým soukromým klíčem, předá své vygenerované veřejné klíče manažerovi,
- **manažer skupiny** (*group manager*, GM) – přiřadí veřejným klíčům identifikátory na základě identity jejich vlastníka – člena skupiny, identifikátory zašifruje svým soukromým klíčem manažera skupiny GM_{SK}, vytvoří listy pomocí hashů veřejných klíčů a šifrovaných identifikátorů, listy permutuje na základě šifry indexu, staví Merkleho strom na permutovaných listech,
- **ověřovatel** (*verifier*, V) – na základě podpisu, autentizační cesty stromem a veřejného klíče skupiny dokáže verifikovat skupinový podpis, nedokáže ale určit identitu podepisujícího.

Operace schématu

Schéma tohoto skupinového podpisu funguje na základě čtveřice polynomiálních algoritmů $\mathcal{GS} = (\mathsf{G.KGen}, \mathsf{G.Sign}, \mathsf{G.Verify}, \mathsf{G.Open})$:

- $\mathsf{G.KGen}(1^k, 1^N)$ – zahrnuje generování B jednorázových párů klíčů N členy skupiny, generování tajného (trasovacího) klíče manažera $GM_{SK} \in \{1,0\}^k$, inicializace blokové šifry $(E_{GM_{SK}}(\cdot), D_{GM_{SK}}(\cdot))$ a generování veřejného klíče skupiny G_{PK} sestavením Merkleho stromu pomocí operací:
  - generování množiny $S = \{(\mathsf{list}_1, (E_{GM_{SK}}(1)), \ldots, (\mathsf{list}_{2^h}, (E_{GM_{SK}}(2^h))\}$, kde list_1 značí hash prvního jednorázového veřejného klíče prvního člena skupiny a $E_{GM_{SK}}(1)$ šifrování indexu 1 s výplní na 256 bit délku,

[6] Tzv. *multi-tree approach*, kde se stromy řetězí za sebou přes několik úrovní. List stromu na vyšší úrovni je využit k podepsání kořene stromu na nižší úrovni. Do autentizační cesty stromem je poté zahrnut i podpis na kořenu daného stromu. Tyto vícestromové konstrukce a podpisy vytváří manažer skupiny.

  - indexy jsou přiřazeny členům skupiny následovně – první člen má listy (list_1,... list_B), druhý člen listy (list_{B+1},... list_{2B}) atd.,
  - operace Shuffle, která seřadí dvojice listů a šifrovaných indexů vzestupně podle hodnoty šifrovaného indexu, toto pořadí značí pozici listu v rámci Merkleho stromu (operace je nutná k zaručení nespojitelnosti podpisů mezi sebou, které by jinak mohlo být umožněno na základě korelace autentizačních cest různých podpisů),
  - sestavení první vrstvy stromu nad dvojicemi listů a šifrovaných indexů tak, že uzel nad listy i a j je $h_{i,j} = \mathsf{H}(\mathsf{list}_i, E_{GM_{SK}}(i) \,||\, \mathsf{list}_j, E_{GM_{SK}}(j))$, šifrované indexy jsou pak součástí autentizační cesty stromem,
  - vyšší vrstvy stromu jsou sestaveny beze změn až po kořen stromu, jehož hash představuje G_{PK},
  - manažer sdělí členu skupiny i množinu permutovaných šifrovaných indexů j: $S_i = \{j_{(i-1)B+1}, \ldots, j_{i \cdot B}\}$,
- G.Sign(G_{SK}, m) – členové skupiny udržují počítadlo t a seznam dvojic reprezentujících $\mathsf{stav} = \{((i-1)B+1, E_{GM_{SK}}((i-1)B+1), \ldots, (i \cdot B, E_{GM_{SK}}(i \cdot B))\}$, při podpisu vyvolá podepisující počítadlo t a zjistí $\mathsf{stav}[t]$, nastaví počítadlo na další hodnotu (stav je použit k interní identifikaci jednorázového páru klíčů, který má být využit pro jednorázový podpis na zprávě m), vytvoří jednorázový Winternitzův podpis odpovídajícím soukromým klíčem G_{SK}, do podpisu σ je přidána autentizační cesta stromem,
- G.Verify(σ, m, G_{PK}) – ověření jednorázového podpisu a následné vypočtení G_{PK} pomocí autentizační cesty, pokud se vypočtený G_{PK} a G_{PK} na vstupu verifikace shodují, podpis je validní,
- G.Open(GM_{SK}, σ, m) – otevření podpisu (revokace anonymity podepisujícího), součástí podpisu je i šifrovaný index, pomocí dešifrování manažerem skupiny je získán index podpisu, který je následně přiřazen odpovídajícímu členu skupiny.

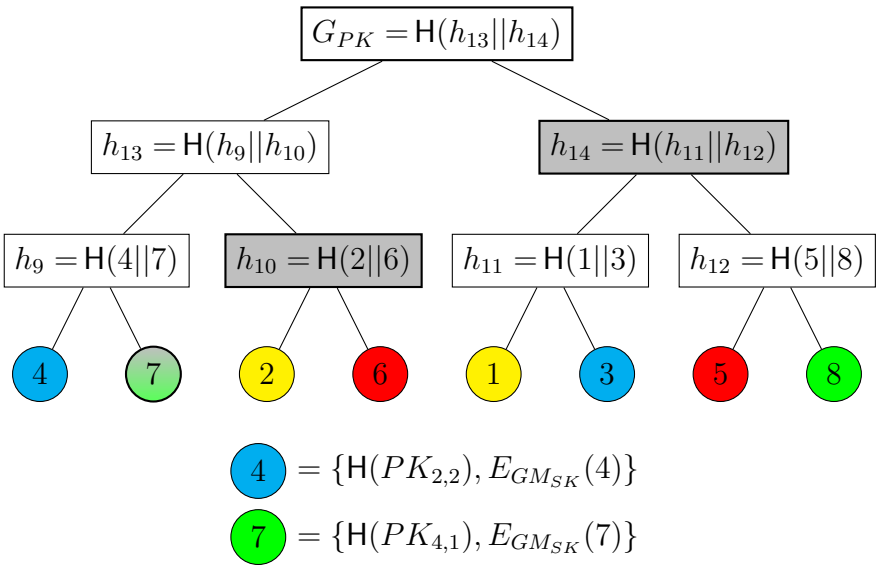
Zjednodušená konstrukce schématu

Na obr. 5.1 je znázorněn příklad *G-Merkle* stromu pro $N = 4$ členů po $B = 2$ podpisech. Listy stromu jsou označeny indexy, které jim přiřadil manažer skupiny. Pro člena 1 jsou tedy přiřazeny indexy – $\{1, 2\}$ a odpovídající žluté listy. Pro člena 2 indexy $\{3, 4\}$ a odpovídající modré listy atd. Listy patřící stejnému členu skupiny jsou rozlišené barevně.

Jak z obrázků vyplývá, listy byly promíchány operací Shuffle. Každý list pak obsahuje odpovídající hash jednorázového veřejného klíče $PK_{\{N\},\{B\}}$ a index šifrovaný pomocí GM_{SK} manažera skupiny. První vrstva stromu je následně vytvořena konkatenací potomků uzlů a jejich zhashováním. Vyšší vrstvy stromu jsou sestaveny

tradičním způsobem, až po kořen stromu G_{PK}. Při podpisu je přenastaven vnitřní stav člena skupiny tak, aby bylo zajištěno, že jeden list nebude využit k podpisu vícekrát.

V obrázku je znázorněna šedou barvou i autentizační cesta stromem pro list 4. Při podpisu člena 2 listem 4 tedy dojde k přiložení autentizační cesty, která obsahuje uzly $\{7, h_{10}, h_{14}\}$.



Obr. 5.1: Příklad *G-Merkle* stromu pro $N = 4$ členů po $B = 2$ podpisech.

Využitá primitiva

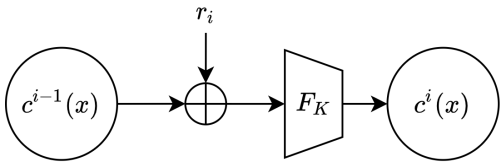
Ke konstrukci schématu byla využita primitiva blokové šifry AES-256, hashovací funkce SHA3-256 (Keccak), standardního Merkleho stromu a dvou typů jednorázového Winternitzova podpisu. Každý typ je využit svou vlastní implementací skupinového podpisu *G-Merkle*, protože využité OTS schéma ovlivňuje jak velikosti klíčů, tak velikost skupinového podpisu a rychlost jeho operací.

Prvním typem je klasický Winternitzův podpis (W-OTS). Jedná se tedy o takový podpis, který byl popsán v kap. 2.1.2.

Druhým typem je Winternitzův podpis Plus (W-OTS+) takový, který je popsán v RFC 8391 [97]. Konstrukce W-OTS+ se liší v tom, že využívá klíčovanou hashovací funkci F, dále využívá pseudonáhodnou funkci PRF, která na vstupu bere n bytový klíč a 32 bytový index a na výstupu generuje pseudonáhodné výstupy délky n. Řetězící funkce následně počítá iterace F na n bytových výstupech z PRF. V každé iteraci řetězce je pomocí PRF generován klíč pro F a bitová maska, která je operací XOR přičtena k výsledku předchozí iterace. Tím je v každé iteraci přidána do

řetězce určitá náhodnost. Mezivýsledek je následně zpracován funkcí F. Řetězení funkcí je znázorněno na obr. 5.2, kde x jsou vstupy derivované ze zprávy, r jsou randomizované výstupy PRF a c jsou hodnoty v daném místě řetězce a představují součásti veřejného klíče, stejně jako hodnoty r a klíče z PRF.

W-OTS+ pak snižuje počet hashování, čímž urychluje operace, zvyšuje tím však velikosti klíčů a podpisu kvůli zahrnutí znáhodněných prvků nutných k verifikaci. Hlavním přínosem W-OTS+ je však snížení nároků na bezpečnost. Přidáním náhodných prvků umožňuje zaručit silnou nepadělatelnost vůči útokům vybranými zprávami (CMA – *chosen message attack*) při využití hashovací funkce, která je odolná vůči útoku nalezení druhého vzoru (*second-preimage resistance*). W-OTS k udržení stejné úrovně bezpečnosti potřebuje hashovací funkci, která je odolná vůči kolizím [97, 98].



Obr. 5.2: Řetězení funkcí W-OTS+.

Proof-of-concept implementace schématu SP G-Merkle

Pro implementaci schématu byl zvolen programovací jazyk Python ve verzi 3.9.11 a byly využity tři externí balíčky:

- `pycryptodome` – využití AES-256 k šifrování a dešifrování indexů,
- `pysha3` – využití SHA3-256 pro veškeré operace vyžadující hashování,
- `merkletools` – sestavení Merkleho stromu a získání autentizační cesty.

Proof-of-concept implementace a demonstrace funkcionality skupinového podpisu *G-Merkle* se sestává z šesti Python souborů:

- `entities.py`,
- `gmerkle.py`,
- `utils.py`,
- `test_scenarios.py`,
- `gmerkle_group_signature.py`,
- `gmerkle_gss_interactive.py`.

Samotná implementace SP využívá tři knihovny naprogramované pro účely této práce ze souborů `entities.py`, `gmerkle.py` a pomocnou knihovnu `utils.py`. V první knihovně `entities.py` jsou obsaženy entity GM a U ve formě tříd `group_manager`

a `group_member`. Knihovna `gmerkle.py` pak obsahuje třídu `g_merkle_GSS` pro instanciaci schématu skupinového podpisu *G-Merkle*. Podpůrné funkce pro SP jako např. hashování, práce s listy, generování klíčů, podepisování a verifikace pomocí OTS schématu jsou naprogramovány v rámci knihovny `utils.py`. K plné práci se skupinovým podpisem tedy stačí tyto tři soubory.

Ukázka jednoduchého použití naprogramované knihovny schématu skupinového podpisu *G-Merkle* je zobrazena na výpisu 5.1. Schéma je inicializované s $N = 10$ a $B = 10$ a člen 5 podepisuje zprávu „Hello world“. Vytvořený podpis je poté ověřen za využití parametrů podpisu, zprávy a veřejného klíče G_{PK} schématu. Následně je podpis i otevřen.

Výpis 5.1: Příklad práce s knihovnou pro SP *G-Merkle*.

```
1  from gmerkle import g_merkle_GSS as GSS
2
3  members = 10              # počet členů
4  signatures = 10           # počet podpisů na člena skupiny
5
6  # instanciace třídy schématu
7  gmerkle_gs = GSS(members, signatures)
8  gpk = gmerkle_gs.gpk      # GPK - Merkleho kořen
9
10 idx = 5                   # id podepisujícího z rozmezí 1 až members
11
12 message = "Hello world" # zpráva k podepsání
13
14 signature = gmerkle_gs.sign(idx, message)
15 verification = gmerkle_gs.verify(signature, message, gpk)
16 opening = gmerkle_gs.open(signature, message)
17
18 print(verification)      # Výstup:  True
19 print(opening)           # Výstup:  Signature was created by user 5.
```

Zbylé tři soubory – `gmerkle_group_signature.py`, `test_scenarios.py`, `gmerkle_gss_interactive.py` – jsou předpřipravené main soubory pro testování implementace skupinového podpisu.

První příklad testování SP využívá jako hlavní spustitelný soubor `gmerkle_group_signature.py`. Soubor využívá knihovny testovacích scénářů, které jsou připraveny v souboru `test_scenarios.py`. Účelem souborů je jednoduchá kontrola korektnosti schématu. Scénáře je možné spustit s předdefinovanými parametry $N = 10$, $B = 5$ z příkazové řádky pomocí argumentu `-s`, `--scenario` spolu s číslem identifikujícím zvolený scénář. Při spuštění bez argumentu dojde k vypsání identifikátorů scénářů a krátkých popisů, po nichž je uživatel dotázán na číslo scénáře, který si přeje spustit.

Předpřipravené scénáře jsou následující:

1. Člen skupiny podepíše zprávu, zpráva je ověřena a otevřena.
2. Pokus o verifikaci a otevření padělaného podpisu.
3. ID, které není součástí skupiny, je vyvoláno při podepisování.
4. Člen skupiny podepíše zprávu, ale verifikace a otevření jsou spuštěny nad rozdílnou zprávou.
5. Člen skupiny podepíše zprávu, následuje pokus o verifikaci podpisu na špatném veřejném klíči. Poté je podpis otevřen.
6. Člen skupiny podepisuje zprávy, dokud nepřijde o všechny jednorázové klíče.

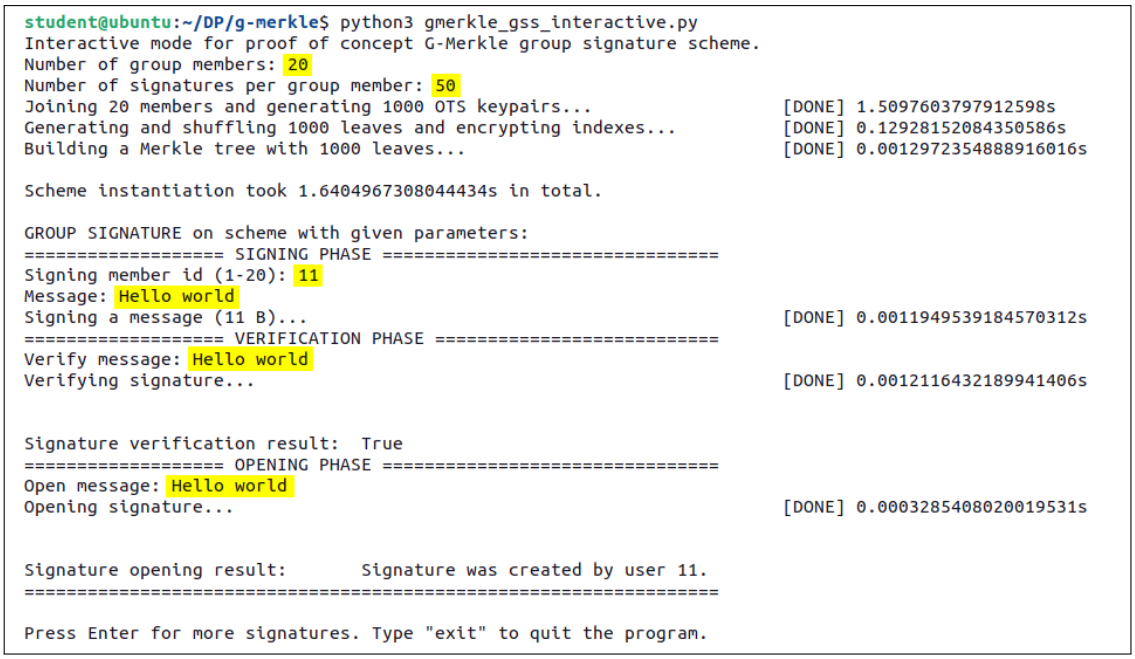
Na obr. 5.3 je zobrazen výstup programu v terminálu při spuštění s parametrem `-s 6`, který značí spuštění šestého scénáře.

```
student@ubuntu:~/DP/g-merkle$ python3 gmerkle_group_signature.py -s 6
Joining 10 members and generating 50 OTS keypairs...                    [DONE] 0.07211494445800781s
Generating and shuffling 50 leaves and encrypting indexes...            [DONE] 0.007562160491943359s
Building a Merkle tree with 50 leaves...                                [DONE] 9.298324584960938e-05s

Scheme instantiation took 0.07988834381103516s in total.

Scheme was instantiated by default with 10 members and 5 signatures per member.
Default message length is 11 B and default signing member id is 8.

Scenarios:
1   A member of the group signs a message and signature verified and opened.
2   A wrong signature is verified and opened.
3   An id that does not belong to any member in the group is invoked on sign message.
4   A member of the group signs a message, but runs verify and open over wrong message.
5   A member of the group signs a message, but verify is ran over wrong public key.
6   A member of the group tries to sign too many times - runs out of keys.
7   Exit program

A member of the group tries to sign too many times - runs out of keys.

Signing a message (11 B)...                                             [DONE] 0.0007367134094238281s
Signing a message (11 B)...                                             [DONE] 0.0007331371307373047s
Signing a message (11 B)...                                             [DONE] 0.0007200241088867188s
Signing a message (11 B)...                                             [DONE] 0.0007076263427734375s
Signing a message (11 B)...                                             [DONE] 0.0007040500640869141s
Signing a message (11 B)... Member with id 8 cannot sign any more messages.
```

Obr. 5.3: Ukázka spuštění `gmerkle_group_signature.py` s parametry `-s 6`.

Druhý příklad testování implementace skupinového podpisu lze spustit pomocí hlavního souboru `gmerkle_gss_interactive.py`. Soubor umožňuje interakci uživatele se schématem. Lze nastavit počet členů N i počet podpisů na člena skupiny B. V tomto případě jsou uměle omezeny rozsahy zadaných parametrů takovým způsobem, aby nebyly příliš velké pro testování ($0 < N \leq 200$ a $0 < B \leq 1024$). Uživatel tedy nejprve zadá parametry skupiny N a B. Následně je založena instance schématu a je na ní možné provádět operace do té doby, než je program ukončen. Uživatel může vybírat ID podepisujícího člena, zprávu k podepsání, zprávu k verifikaci a otevření.

Ukázka je zobrazena na obr. 5.4. Vstupy uživatele jsou zvýrazněny žlutou barvou. Uživatel založil skupinu o 20 členech, kde má každý člen k dispozici až 50 podpisů. Následně vybral člena skupiny s ID 11 k podepsání zprávy „Hello world“. Verifikaci i otevření zprávy spustil nad stejnou zprávou.



Obr. 5.4: Ukázka spuštění `gmerkle_gss_interactive.py`.

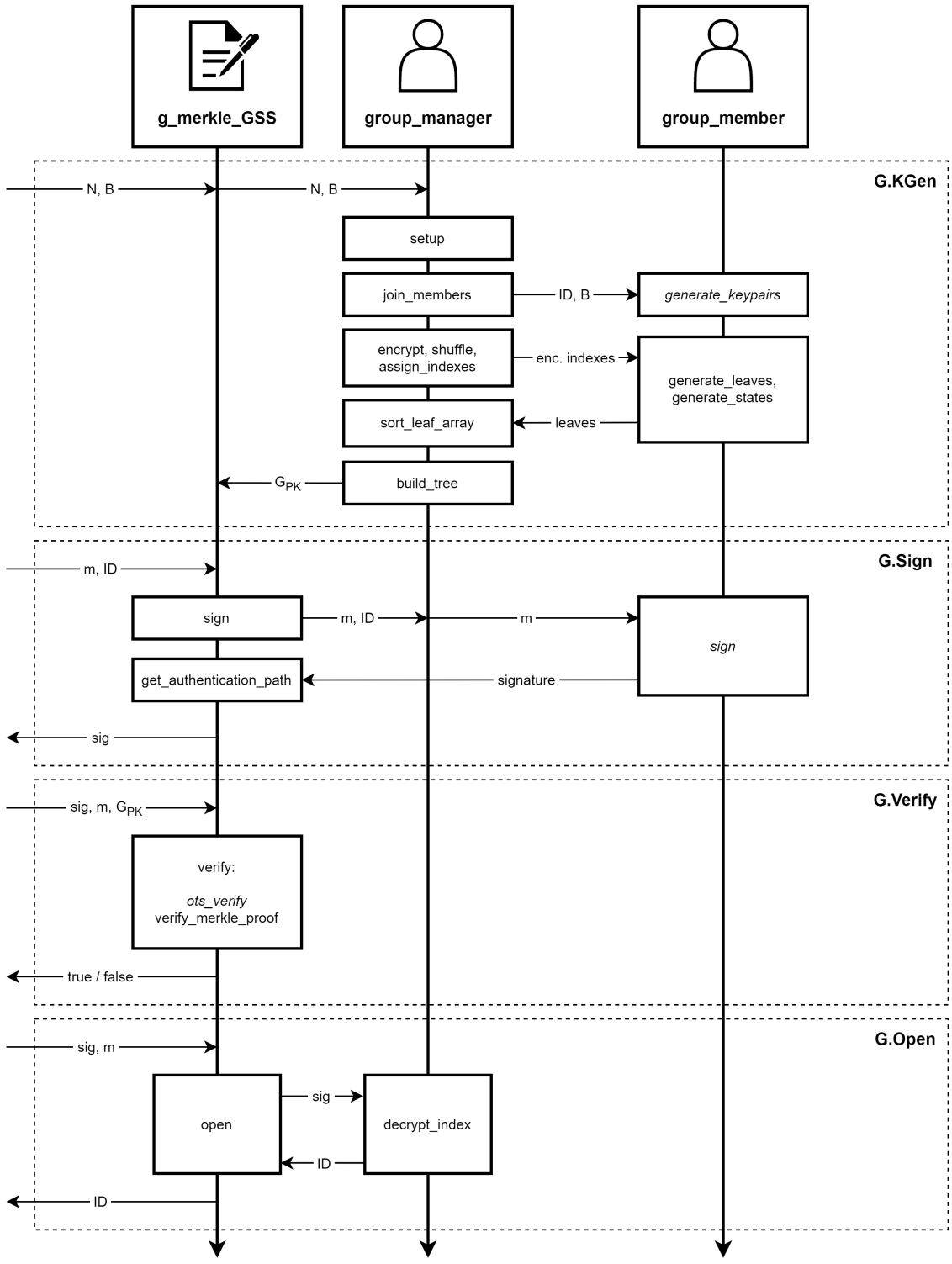
Instrukce k používání demonstračních ukázek schématu a knihovny a bližší informace ke zprovoznění schématu a instalaci potřebných balíčků jsou také dostupné v souboru `README.md` v repozitáři implementací.

Interní kooperace implementovaných tříd

Na obr. 5.5 je znázorněn průběh operací G.KGen, G.Sign, G.Verify a G.Open. Šipky na levé straně schématu označují vstupy a výstupy těchto operací, v závislosti na směru. Důležité je podotknout, že metody psané kurzívou, tedy *generate_keypairs*, *sign* a *ots_verify*, využívají knihovnu `utils.py`, která se liší pro implementaci skupinového podpisu s jednorázovým podpisem W-OTS a s W-OTS+.

V rámci metody `get_authentication_path` je k podpisu přidána autentizační cesta, která je potřebná k ověření.

Osoba ověřovatele nemá vlastní dedikovanou třídu, namísto toho jsou verifikační metody implementovány v rámci třídy `g_merkle_GSS`, nicméně verifikaci může provést prakticky jakákoliv entita. Je nutné pouze ověřit jednorázový podpis na zprávě m a porovnat Merkleho kořen oproti přiloženému veřejnému klíči G_{PK} pomocí výpočtu autentizační cesty.



Obr. 5.5: Průběh operací schématu *G-Merkle*.

Efektivita skupinového podpisu

Implementace skupinového podpisu *G-Merkle* za využití jednorázového podpisového schématu W-OTS bude označována zkráceně jen jako „*G-Merkle*“. Implementace *G-Merkle* s W-OTS+ bude označována jako „*G-Merkle+*“.

Oběma zmíněným implementacím byly pro účely porovnání efektivity změřeny paměťové a výpočetní nároky[7]. Paměťové nároky byly určeny jako reálné velikosti zabrané paměti pro objekty s veřejným klíčem G_{PK}, soukromým klíčem člena skupiny G_{SK} a objektu s podpisem. Velikosti jsou tedy na rozdíl od teoretických zvýšeny o programový overhead. Časové nároky schématu byly měřeny v (mili)sekundách na základě doby provádění vybraných operací. Operace byly prováděny se zprávou o velikosti $|m| = 11\,\mathrm{B}$. Pro účely testování byla vytvořena skupina členů, kde každý člen měl maximálně jeden podpis.

Na grafech v této podkapitole je znázorněna závislost velikostí parametrů a rychlosti operací *G-Merkle* a *G-Merkle+* na počtu jednorázových párů klíčů. Je nutné brát v úvahu fakt, že čím větší počet členů schéma podporuje, tím menší počet podpisů je možné přiřadit každému členu při udržení stejné efektivity a naopak. Osy x těchto grafů jsou logaritmické.

Velikost G_{PK} je stejná pro *G-Merkle* i *G-Merkle+*, jedná se o výstup zvolené hash funkce pro vytvoření Merkleho stromu, v tomto případě SHA3-256. Tento hash zabírá v paměti 120 B pro obě schémata. Implementace je však modulární a bylo by možné bez potíží implementovat využití jiných hash funkcí.

Velikost G_{SK} je 3640 B pro W-OTS a 7560 B pro W-OTS+. Jedná se tedy o více než dvojnásobné zvětšení tohoto parametru při změně využitého schématu jednorázového podpisu. Je nutné uvažovat paměťovou náročnost udržování B párů klíčů pro každého člena skupiny.

Velikost podpisu je při menší skupině $N = 4$ až o 80 % větší u W-OTS+, nicméně tento rozdíl klesá až na 40 % při velikosti skupiny o $N = 200$ tis. členech po jednom podpisu. Je tedy zřejmé, že i když je výsledný podpis menší pro W-OTS, roste rychleji v závislosti na počtu jednorázových párů klíčů v porovnání s W-OTS+ podpisem. Maximální naměřené velikosti podpisů tedy byly při 200 tisících OTS párech klíčů 11 KB pro W-OTS a 15,4 KB pro W-OTS+. Velikost zprávy by na velikosti podpisů neměla mít vliv, jelikož se jedná o *hash-and-sign* přístup. Velikosti těchto parametrů jsou zobrazeny v grafu na obr. 5.6.

Grafy na obr. 5.7, 5.8, 5.9 potom znázorňují časové náročnosti operací G.KGen, G.Sign, G.Verify a G.Open postupně.

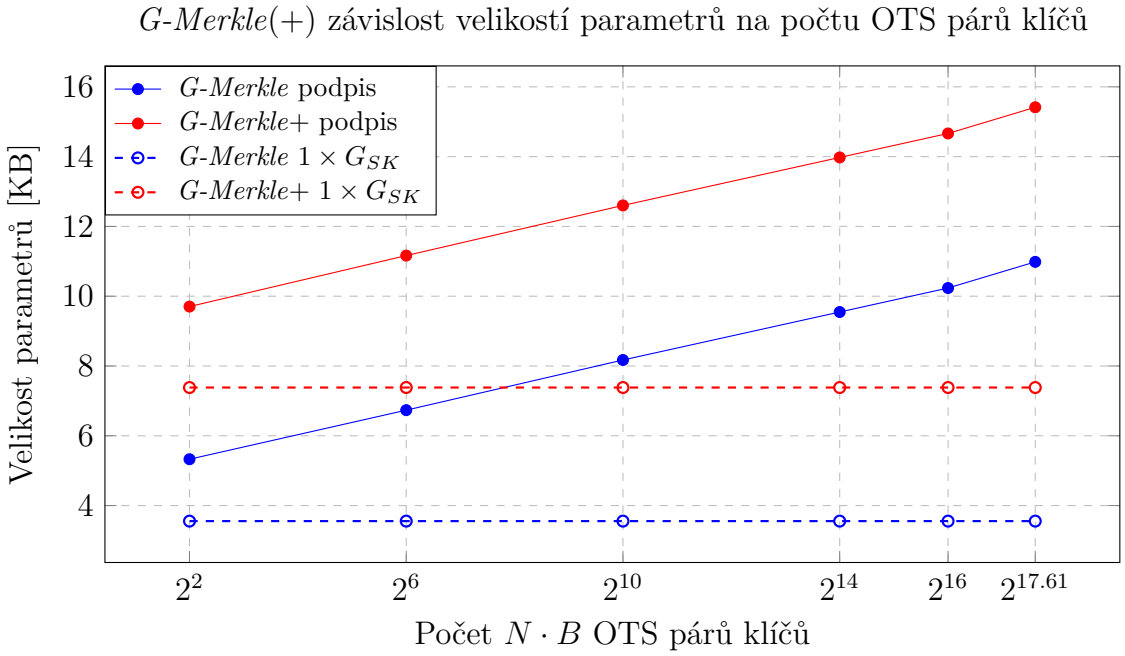
[7]Údaje byly měřeny na virtuálním stroji s OS Linux Ubuntu 21.10. Stroji byly přiděleny dva procesory po jednom jádru @ 3,8 GHz a 8 GB RAM.

Z grafu 5.7 vyplývá, že generování klíčů pro *G-Merkle* trvá delší dobu než pro *G-Merkle+*. Časový rozdíl se začíná více projevovat až při počtu 2^{10} OTS párů klíčů. Doba výpočtu klíčů pro *W-OTS* je tedy významně vyšší. Operace G.KGen v sobě zahrnuje kromě generování jednorázových klíčů ještě šifrování a Shuffle indexů listů a výpočet celého Merkleho stromu. Generování OTS klíčů zabere průměrně 98 % z celkového času G.KGen u *G-Merkle*, 91 % u *G-Merkle+*. Operace šifrování, resp. promíchání indexů pak necelé 2 %, resp. 9 %. Fáze výpočtu stromu pak zabírá pouze dvě setiny, resp. dvě desetiny procenta z celkového času. Je nutné podotknout, že operace generování OTS klíčů není paralelizovaná, a proto neodpovídá potenciálnímu reálnému využití, kdy by členové skupiny generovali klíče zároveň. Klíče by také mohly být členy předgenerovány a v rámci operace by manažerovi byly pouze předány veřejné klíče těchto párů. Takové úpravy by výrazně snížily délku operace při reálném využití.

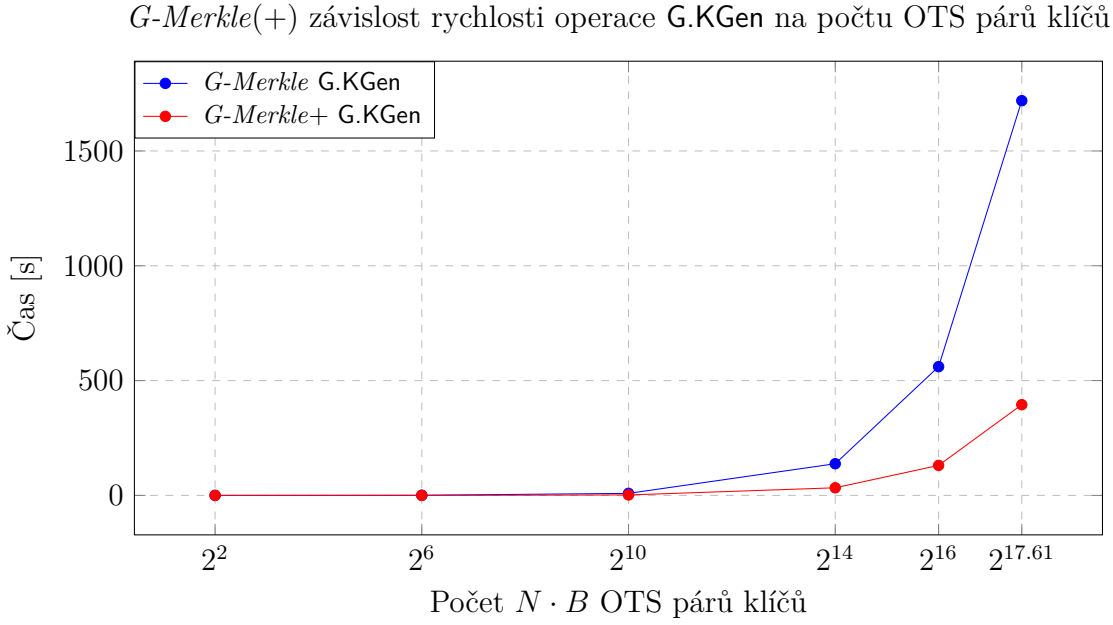
Na obr. 5.8 je zobrazena rychlost operace G.Sign. S vyšším počtem listů stromu se operace zpomaluje na základě přidávání autentizační cesty k podpisu. Je to zapříčiněno programovou režií, kdy instance schématu hledá pozici listu využitého k podpisu v rámci stromu, aby mohla být zjištěna autentizační cesta stromem. Operace se tedy pro *G-Merkle+* pohybuje v rozmezí do 10 ms pro všechny měřené velikosti skupiny, nejčastěji však okolo 1,2 ms. Pro *G-Merkle* je operace podepisování pomalejší kvůli W-OTS, operace trvá až 70 ms pro maximální měřenou velikost skupiny, nejčastěji okolo 4,5 ms. Vyšší velikost zprávy by v tomto případě zapříčinila delší dobu podepisování, zejména kvůli hashování většího objemu dat. Zbytek operace by následně probíhal stejně rychle.

V grafu na obr. 5.9 jsou porovnávány operace G.Verify a G.Open. Operace verifikace trvá průměrně okolo 5 ms pro *G-Merkle* a 1,2 ms pro *G-Merkle+*. Rozdíl je tvořen oveřováním jednorázového podpisu, které je u W-OTS pomalejší. Ověřování autentizační cesty trvá stejnou dobu. Operace otevírání podpisu je pouze dešifrování pomocí symetrické šifry AES-256. Čas trvání operace je uměle prodloužen, protože manažer skupiny před samotným dešifrováním přiloženého indexu kontroluje, zda je vůbec využitý list přítomný v jím spravovaném schématu. Otevření tedy trvá stejnou dobu pro oba dva typy *G-Merkle* skupinového podpisu, nejčastěji 0,3 ms. S větším počtem listů stromu se pak začne projevovat zmíněné prohledávání seznamu listů na rychlosti operace.

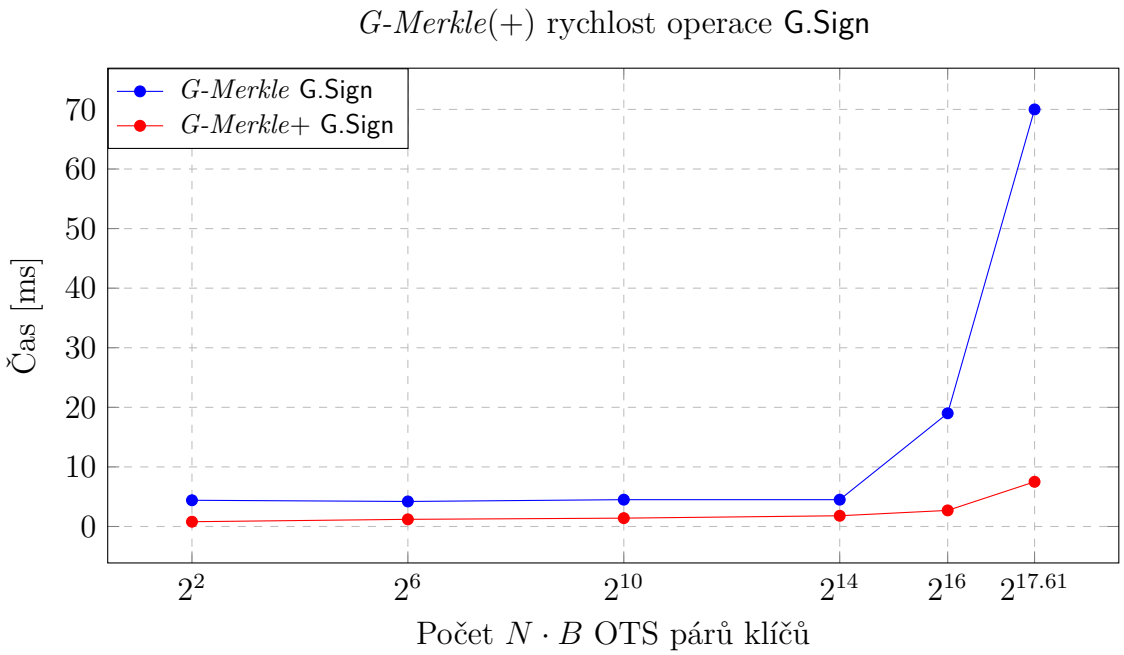
Z této podkapitoly a naměřených dat tedy vyplývá, že *G-Merkle* je vhodnější využít tam, kde je nutné uspořit paměťovou náročnost na úkor výpočetních nároků. Bohužel je nutné počítat s uložením B párů klíčů, popř. pouze soukromých klíčů na straně klienta, pokud nejsou klíče generovány pomocí nějaké pseudonáhodné funkce s určeným počátečním náhodným číslem (*seed*). *G-Merkle+* naopak umožňuje rychlejší operace pro případ, kdy není paměťová náročnost tolik podstatná.



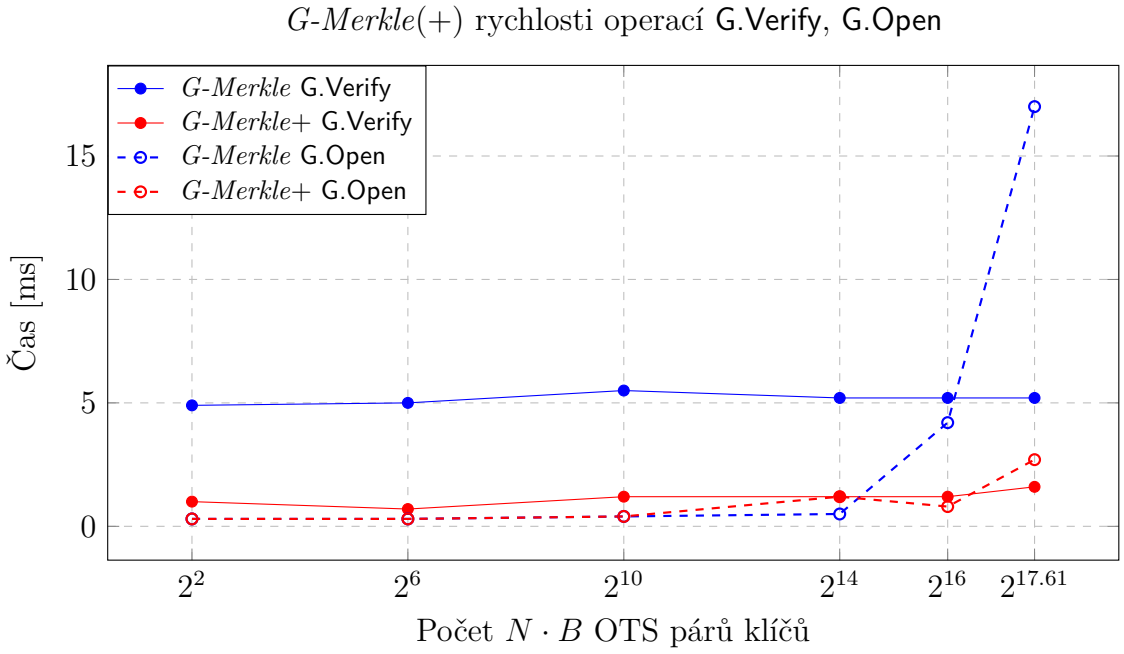
Obr. 5.6: Graf závislosti velikosti podpisu, G_{SK} *G-Merkle*(+) na počtu OTS.



Obr. 5.7: Graf závislosti rychlosti operace G.KGen na počtu OTS párů klíčů.



Obr. 5.8: Graf závislosti rychlosti operace G.Sign na počtu OTS párů klíčů.



Obr. 5.9: Graf závislosti rychlosti operací G.Verify a G.Open na OTS párech klíčů.

5.2.2 Implementace SP založeného na teorii kódování

Implementace statického skupinového podpisu založeného na teorii kódování z publikace [73] byla získána emailem na základě komunikace s výzkumnými pracovníky *dr. Fredericem Ezermanem* (Technická univerzita Nanyang, Singapur) a *dr. Hyung Tae Leem* (Národní univerzita Jeonbuk, Korea). Spolu s dalšími kolegy tato dvojice přináší úplně první implementaci napříč skupinovými podpisy ze všech rodin postkvantové kryptografie. Publikace je z roku 2015.

Prerekvizity, spuštění, testování a měření

Autoři implementaci vytvořili v programovacím jazyce C++. Implementace byla zprovozněna na VM Linux Ubuntu 21.10[8] po instalaci tří externích knihoven pro přesnější a rychlejší výpočty s čísly `GMP`, `NTL` a `MPC`. Ke kompilaci byl využit kompilátor `g++`.

Schéma lze po instalaci prerekvizit zkompilovat např. do souboru `test` a spustit pomocí následujících příkazů, kde prvním parametrem je ID člena skupiny, který bude podepisovat zprávu. Následně bude anonymita tohoto podpisu revokována a ID je odhaleno. Druhým parametrem je náhodný prvek *seed* využitý v rámci schématu.

```
g++ code_GS.cpp -o test -lntl -lgmp
./test 38 10
```

Schéma umožňuje nastavit velikost skupiny a definovat podepisovanou zprávu, v tomto ohledu je tedy modulární. Problémem je pak fakt, že se parametry velikosti skupiny využívají při dimenzování velikostí proměnných při kompilaci souboru, proto je nutné dané parametry měnit v kódu před každou kompilací. Parametry velikosti skupiny se definují v rámci konstant na začátku souboru `code_GSS.cpp`, přesněji zejména v konstantách `log_NGU`, `NGU` a `k_1`. Nejprve je nutné definovat `log_NGU` jako kladné celé číslo. Ostatní parametry se vypočítají následovně:

- `log_NGU` $= x$, kde $x \in \mathbb{N} \wedge x < 31$,
- `NGU` $= 2^x$,
- `k_1` $=$ `k_E` $-$ `log_NGU`,

přičemž `k_E` je parametr definovaný pro využitý McEliece kryptosystém (kap. 2.2.1). Kryptosystém s $[n, k, 2t + 1]$ binárním Goppa kódem je nastavený parametry o velikostech $(n, k, t) = (2^{11}, 1696, 32)$, kde `k_E` odpovídá k. Z parametru `NGU` tedy zřejmé, že velikost skupiny lze pouze nastavit v mocninách dvou. Podepisovaná zpráva lze zvolit v neomezené délce v souboru `msg.txt`.

[8] Stroji byly přiděleny dva procesory po jednom jádru @ 3,8 GHz a 8 GB RAM.

Více informací k instalaci prerekvizit s konkrétními příkazy a další příklady parametrů jsou uvedeny v souboru `README.md` v repozitáři tohoto skupinového podpisu.

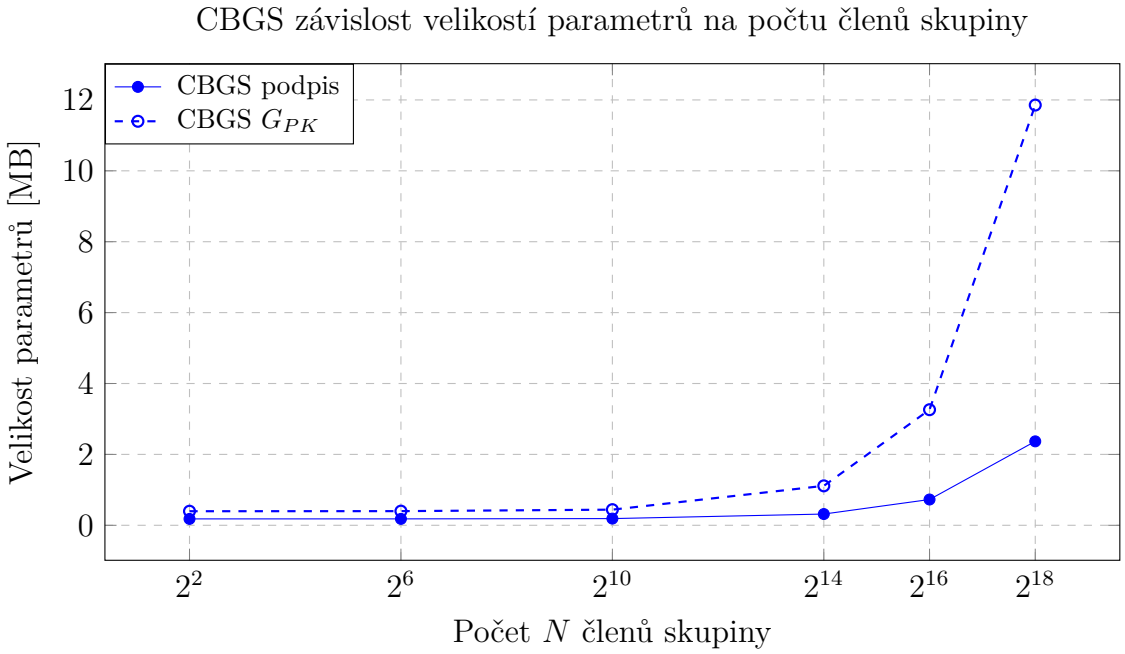
Efektivita skupinového podpisu

Implementaci byly pro účely porovnání efektivity změřeny paměťové a výpočetní nároky. Nároky schématu byly měřeny na zprávě o velikosti $|m| = 11\,\mathrm{B}$ stejným způsobem jako v předchozím případě. Tato implementace bude dále označována jako „CBGS“ (*code-based group signature*).

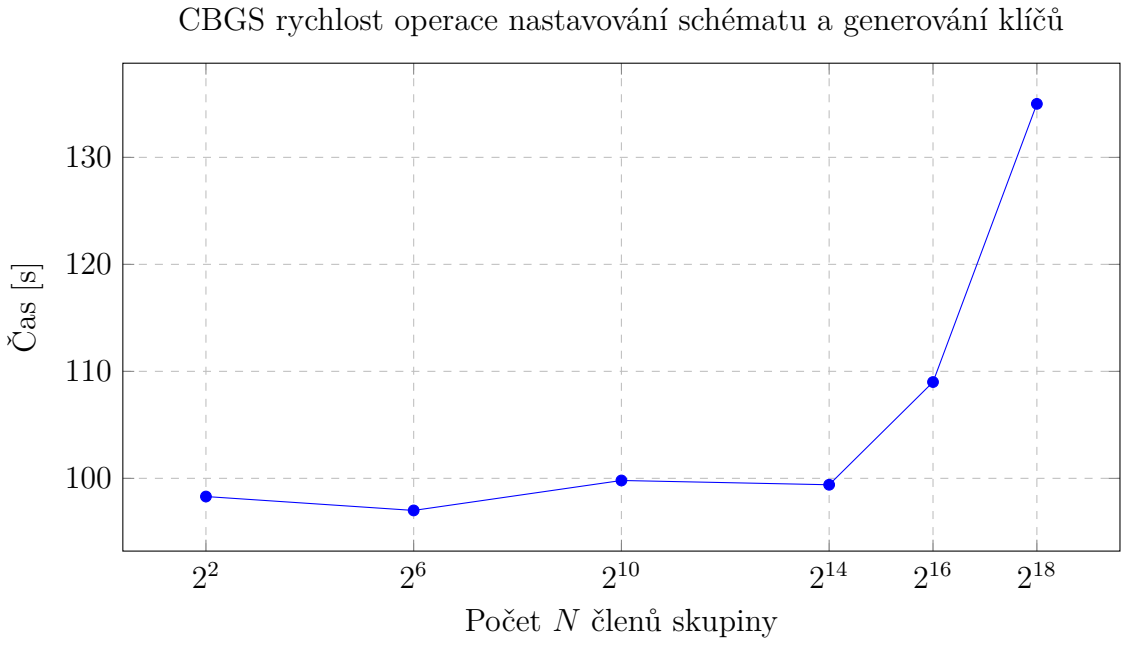
V grafu na obr. 5.10 je zobrazena závislost velikostí podpisu a veřejného klíče skupiny G_{PK} na počtu členů skupiny. Velikosti těchto dvou parametrů jsou lineárně závislé na velikosti skupiny, tento fakt se však projevuje až při velikosti skupiny vyšší než 2^{10}. V podpisu a veřejném klíči jsou totiž zahrnuty dvě velké matice o konstantních rozměrech napříč měřeními různých instancí schématu. Jedná se totiž o parametry stanovené pro využitý McEliece kryptosystém a problém dekódování syndromů. Velikost těchto konstant je daná bezpečnostním parametrem λ pro 80 bitovou bezpečnost. Publikace neuvádí úpravu parametrů pro zaručení vyšší bezpečnosti.

Na obr. 5.11 je znázorněna časová náročnost operace nastavování schématu a generování klíčů. Tato operace se pohybuje těsně pod hranicí 100 s pro velikosti skupiny do 2^{14}, pro větší skupiny se začíná výrazněji prodlužovat. Nejedná se však o extrémní nárůst. Rozdíl zhruba 245 tis. členů tvoří v rámci této fáze časový nárůst výpočetní doby pouze o 35 %.

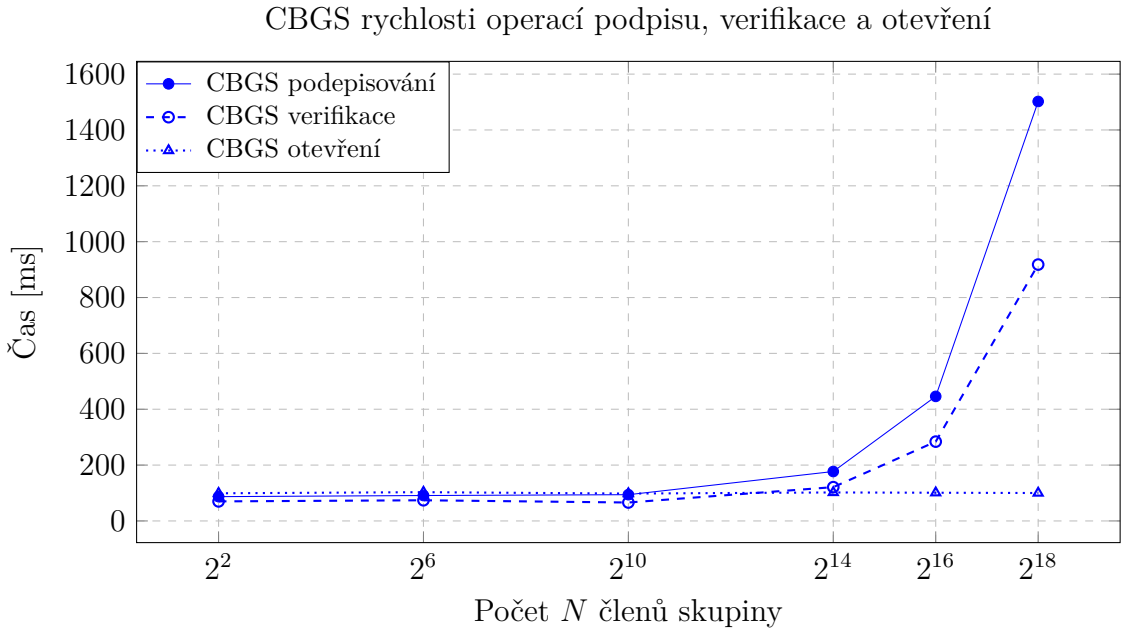
V grafu na obr. 5.12 jsou porovnány doby výpočtu operací podepisování, verifikace a otevření podpisu v závislosti na velikosti skupiny. Z grafu vyplývá, že operace otevření podpisu není závislá na velikosti skupiny. Operace trvá průměrně okolo 100 ms. Operací, jejíž náročnost roste nejvíce v závislosti na velikosti skupiny, je operace podepisování. Do velikosti skupiny 2^{14} se tato operace však pohybuje pod hranicí 200 ms. Výpočetní čas verifikace též od této velikosti skupiny značně narůstá.



Obr. 5.10: Graf závislosti velikosti podpisu, G_{PK} CBGS na počtu členů skupiny.



Obr. 5.11: Graf závislosti rychlosti operace Setup na počtu členů skupiny.



Obr. 5.12: Graf závislosti rychlostí podpisu, verifikace a otevření na velikosti skupiny.

5.2.3 Implementace SP založeného na mřížkách

Implementace statického skupinového podpisu založeného na mřížkách z publikace [50] byla získána emailem na základě komunikace s *dr. Gregorem Seilerem* (IBM Research, Spolková vysoká technická škola v Curychu, Švýcarsko). Vědecká skupina, která toto schéma zkonstruovala, dosáhla až o řád kratších parametrů než v té době nejefektivnější mřížkové schéma (2018). Autoři se zaměřili na návrh a využití nových důkazů znalosti a jejich použití spolu s podpisy, jejichž anonymita je založena na selektivní bezpečnosti. Schéma je podpořeno první experimentální implementací mřížkového skupinového podpisu.

Prerekvizity, spuštění, testování a měření

Implementace byla autory vytvořena v programovacím jazyce C. Byla zprovozněna na stejném virtuálním stroji jako CBGS za pomoci tří externích knihoven pro rychlejší a přesnější výpočty `GMP`, `MPC` a `FLINT`. Ke kompilaci programu byl využit kompilátor `gcc`.

Schéma je možné po instalaci prerekvizit zkompilovat a spustit pomocí následujících příkazů:

```
make
./test_sign
```

Implementace získaná od autorů tohoto schématu se však po prostudování jeví pouze jako *proof of concept*, který byl vytvořen ke kontrole funkcionality navrženého důkazu nulové znalosti a omezeného naměření rychlosti operací. Implementace schématu tedy není modulární a obsahuje natvrdo kódované parametry a operace. Umožňuje tedy vygenerovat veřejný klíč skupiny G_{PK}, jeden klíč pro člena skupiny G_{SK}, s tímto klíčem podepsat zprávu o stanovené velikosti 32 B a výsledný podpis ověřit. Schéma nemá implementovanou operaci otevření podpisu, ani neumožňuje generovat klíče pro další členy skupiny.

Více informací k instalaci prerekvizit s konkrétními příkazy jsou uvedeny v souboru `README.md` v repozitáři tohoto skupinového podpisu.

Efektivita skupinového podpisu

Na základě zjištění popsaných výše bylo možné danou implementaci (dále „LBGS“ *lattice-based group signature*) naměřit pouze s jedinou množinou parametrů. Výsledky měření jsou zapsány v tab. 5.2.

V rámci fáze Setup jsou tedy generovány klíče G_{PK} a G_{SK} pouze pro jednoho člena. Parametrem specifickým pro tuto implementaci je počet odmítnutí vytvořené odpovědi v rámci protokolu nulové znalosti (Rej). Na výstupu se totiž kontroluje, aby daná funkce využitá v protokolu nulové znalosti měla normální rozdělení pravděpodobnosti. Pokud výstup kontrolou neprojde, je zamítnut. Zamítnutí pak znamená, že z konkrétního výstupu unikají informace, a tudíž nemůže být použit k vytvoření podpisu. Tvorba důkazu nulové znalosti se následně opakuje do té doby, než je vytvořen důkaz bez úniku informací. V rámci tohoto měření docházelo průměrně k zamítnutí 20,5 důkazů, než bylo možné podpis vytvořit.

Publikace se nezmiňuje, jakým způsobem může velikost skupiny ovlivňovat výsledné parametry schématu. V porovnávacích grafech bude tento podpis zobrazován s konstantními hodnotami, které nemusí odpovídat reálnému škálování tohoto schématu.

Tab. 5.2: Velikosti parametrů a časy operací LBGS.

N	**Zpráva [B]**	**Podpis [B]**	**G_{PK} [B]**	**G_{SK} [B]**	**Setup [ms]**	**Sign [ms]**	**Ver [ms]**	**Rej [-]**
1	32	638972	475136	393216	301	211	6	20,5

5.2.4 Porovnání implementací

Tato podkapitola je věnována porovnání konkrétních naměřených výsledků popsaných implementací proti sobě a zároveň i proti klasickým skupinovým podpisům implementovaným pomocí knihovny `libgroupsig` (BB04 [81], KLAP20 [83], GL19 [84], PS16 [85]).

Všechna schémata byla měřena na stejné výpočetní platformě – virtuálním stroji Linux Ubuntu 21.10 s přidělenými dvěma procesory po jednom jádru @ 3,8 GHz a 8 GB RAM. Byla podepisována zpráva o délce 11 B.

Maximální velikost měřené skupiny byla pro schémata *G-Merkle*(+) $2^{17,6}$ (což odpovídá zhruba 200 tis. členům pro $B = 1$). Větší počet jednorázových párů klíčů nebylo možné na přiřazeném hardware uložit z důvodu omezené velikosti RAM. Pro ostatní schémata byla maximální měřená hodnota 2^{18}.

Velikosti parametrů

Dle velikosti parametrů G_{PK}, G_{SK} a podpisu je z měření zřejmé, že dané příklady implementací postkvantových skupinových podpisů nemohou konkurovat klasickým variantám skupinových podpisů z pohledu paměťových nároků. Pro klasické SP zůstávají všechny velikosti těchto parametrů konstantní a nezávislé na počtu členů skupiny. Řádově se pohybují ve stovkách bajtů pro všechny parametry.

Pro postkvantové implementace zůstává konstantní velikosti zejména soukromý klíč člena skupiny G_{SK}. Pro CBGS je nejmenší, a to pouze 72 B. Pro *G-Merkle*(+) má jeden soukromý klíč velikost až v jednotkách KB. Je však nutné si uvědomit, že se jedná o jednorázové klíče generované pro celou dobu existence schématu, a tudíž je nutné je uložit do paměti. Při generování je také nutné jednorázově předat veřejné klíče manažerovi skupiny. Paměťové nároky na jednoho člena by se tedy bez uplatnění nějakých dalších mechanismů na lepší generování OTS párů klíčů bez nutnosti jejich uložení mohly pohybovat okolo velikosti dané rovnicí $B \cdot (|G_{SK}| + |PK_{OTS}|)$. Při velkém B jsou pak nároky na paměť člena skupiny extrémní. Tyto nároky jsou ještě dále navýšeny o šifrované indexy k zajištění trasovatelnosti, požadavek udržování stavu a režii s ním spojenou. Zmíněný problém částečně řeší schéma [72] za pomoci dynamičnosti podpisu, kde není nutné při inicializaci schématu generovat všechny klíče, ale pouze jejich část.

Nejmenší velikost veřejného klíče G_{PK} však umožňuje SP založený na hashích, jelikož využívá struktury Merkleho stromu a z toho plynoucí G_{PK} o velikosti paměti zabrané jediným výstupem zvolené hash funkce. Prakticky se jednalo o funkci SHA3-256, jejíž výstup dle měření zabíral v paměti 120 B. CBGS naopak přináší jediný měřený veřejný klíč, jehož velikost roste v závislosti na velikosti skupiny (až na 11,8 MB při velikosti skupiny 2^{18}).

Velikosti podpisů rostly pro *G-Merkle*(+) i CBGS. Pro hashové SP však jen nepatrně v závislosti na prodlužování autentizační cesty stromem, která je přidávána k výslednému podpisu. Velikost podpisu pro největší skupinu dosáhla 11 KB, u CBGS se jednalo až o 2,3 MB.

Parametry zmiňované v této kapitole jsou vypsány v tab. 5.3.

Tab. 5.3: Velikosti parametrů SP.

Schéma	G_{PK}	G_{SK}	Podpis
G-Merkle	120 B	3,5 KB	5 KB – 11 KB
G-Merkle+	120 B	7,4 KB	9,7 KB – 15,4 KB
CBGS	406 KB – 11,8 MB	72 B	185 KB – 2,3 MB
LBGS	464 KB	384 KB	624 KB
BB04	2,8 KB	952 B	552 B
GL19	880 B	536 B	984 B
KLAP20	792 B	312 B	360 B
PS16	528 B	248 B	288 B

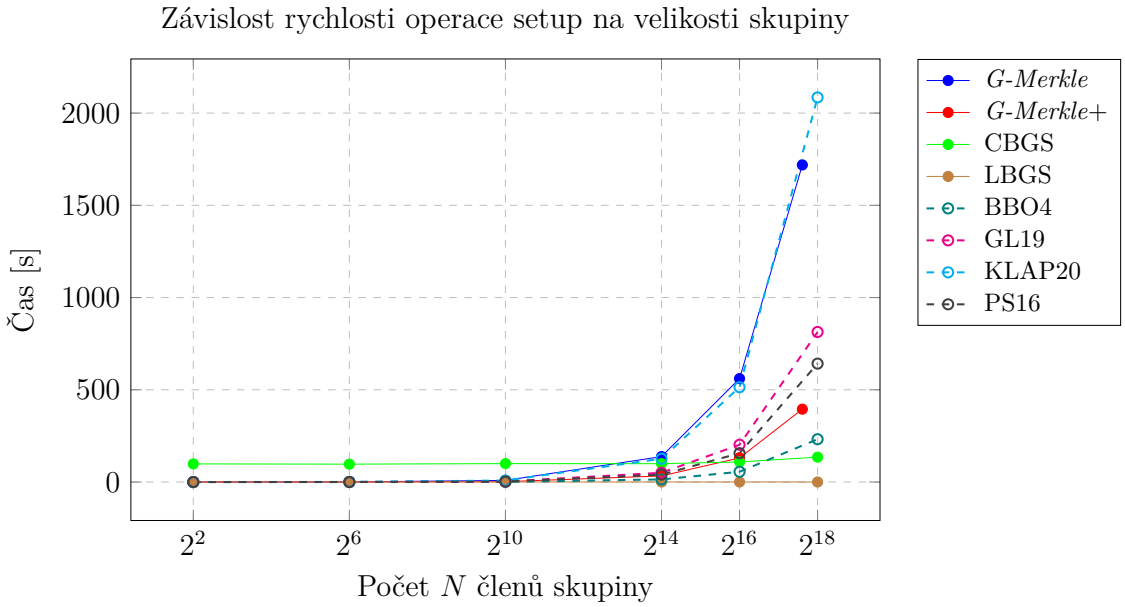
Fáze nastavování (setup) a generování klíčů

Na grafech na obr. 5.13 a 5.14 jsou znázorněny porovnání rychlosti operace setup v závislosti na velikosti skupiny. Plnou čárou jsou zobrazeny postkvantové skupinové podpisy, přerušovaně pak ty klasické.

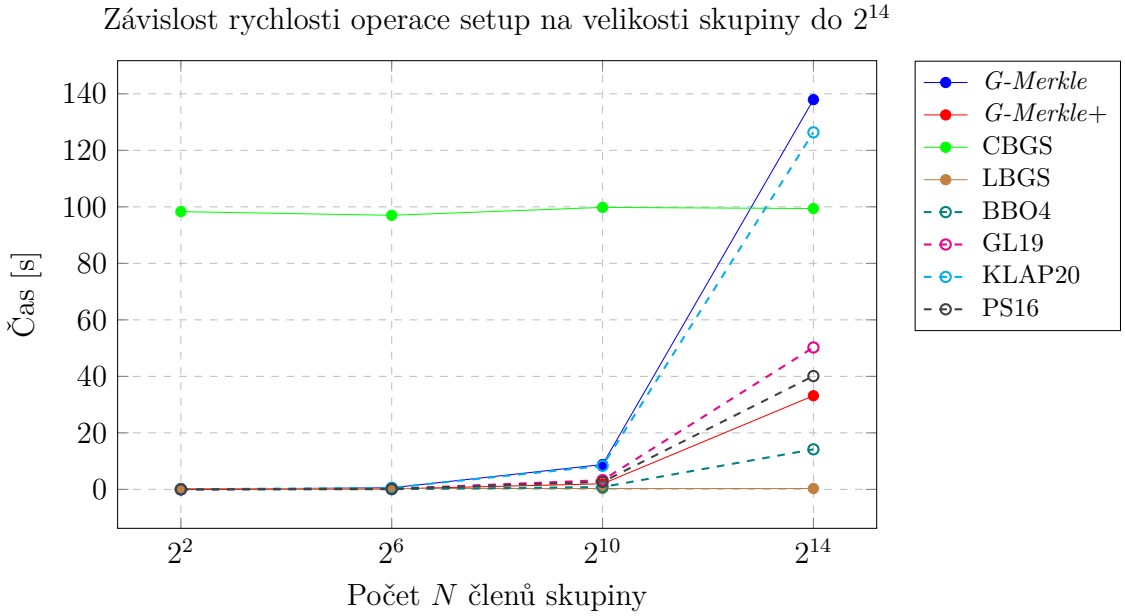
Z grafu je na první pohled zřejmé, že pro CBGS je operace nastavování a generování klíčů téměř konstantní (roste pouze nepatrně v porovnání s ostatními). Zhruba do velikosti skupiny 2^{16} je značně neefektivní, ale při větších skupinách je bezkonkurenčně nejrychlejší.

Skupinovým podpisem s nejefektivnější operací nastavování a přijímání členů je klasické schéma BB04. Až do velikosti skupiny 2^{16} umožňuje výkon této operace v době pod jednu minutu. Za ním následuje schéma *G-Merkle+*, které se pro N rovno počtu OTS párů klíčů pohybuje při stejné velikosti skupiny na času 130 s. Toto schéma hashového SP se tedy při daném nastavení počtu OTS efektivněji škáluje v závislosti na N než další tři měřené klasické podpisy. Zajímavý je pak fakt, že *G-Merkle* s W-OTS je podobně efektivní jako klasický SP KLAP20.

V případě tohoto měření neprobíhá současné přijímání vícero členů do skupiny (schémata z `libgroupsig`) ani současné generování jednorázových klíčů všech členů skupiny (*G-Merkle*(+)). Při reálném využití by tyto funkce mohly být paralelně vykonávány pro všechny členy skupiny, a tudíž by došlo k řádově kratším časovým nárokům operací generování klíčů/přijímání členů skupiny.



Obr. 5.13: Graf závislosti rychlosti operace setup na počtu členů skupiny.

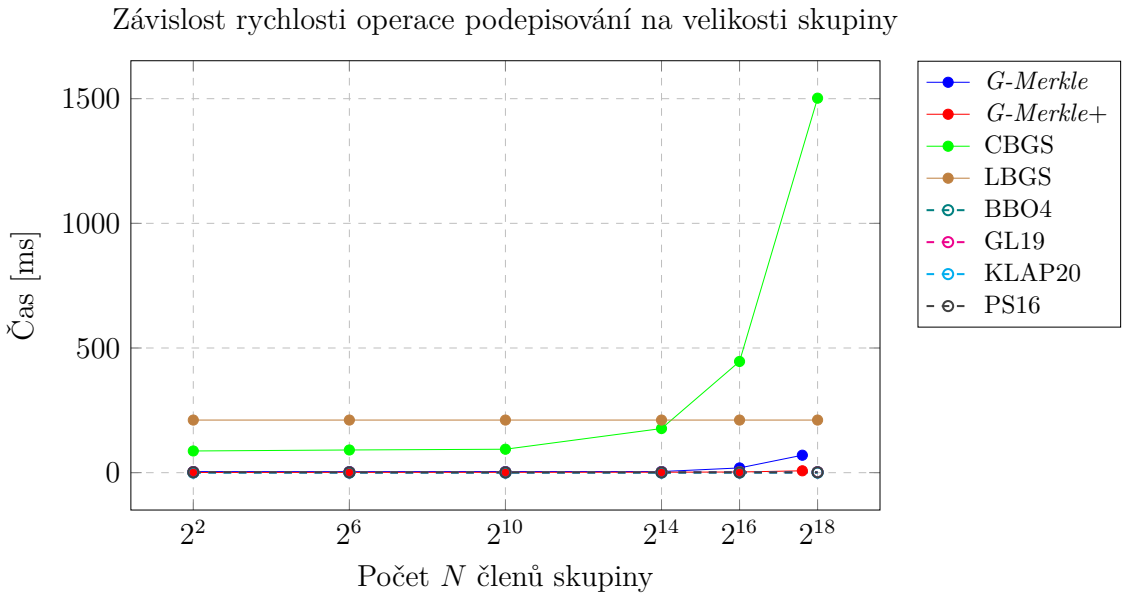


Obr. 5.14: Graf závislosti rychlosti operace setup na počtu členů skupiny do 2^{14}.

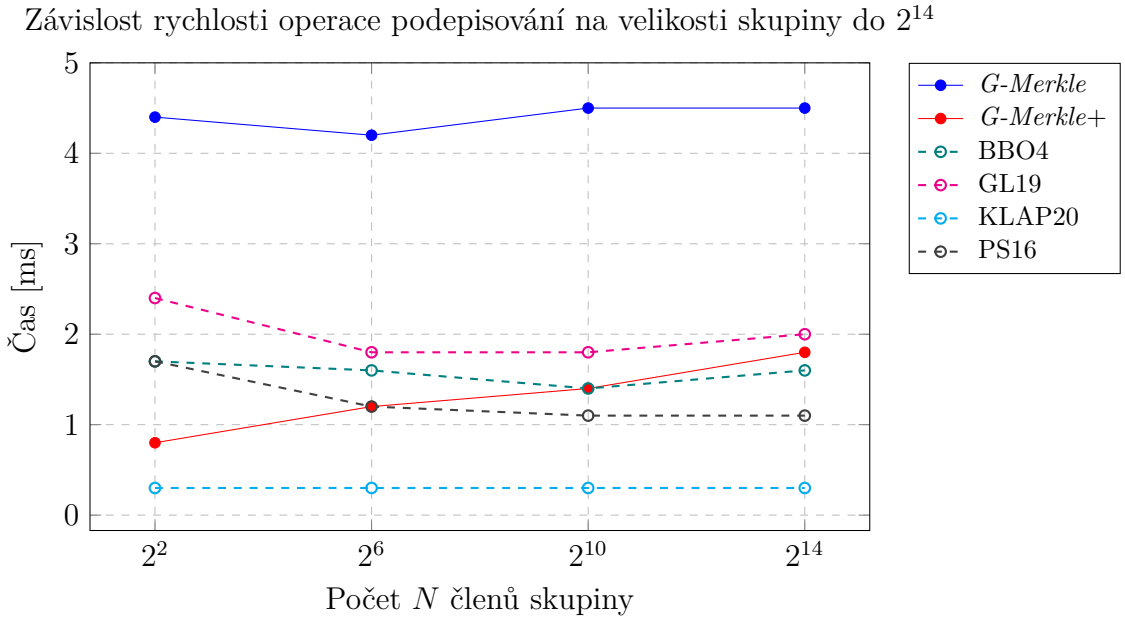
Fáze podepisování

V grafech na obr. 5.15 a 5.16 lze vidět porovnání rychlosti operace podepisování napříč měřenými schématy. Je zřejmé, že nejméně efektivní je LBGS, kde i skupina o jednom členu provádí nejdelší operaci podepisování ze všech měřených schémat. Je to zřejmě zapříčiněné využitým protokolem nulové znalosti, jak bylo popsáno

v podkapitole 5.2.3. Následuje implementace podepisování v CBGS, která je lineárně závislá na počtu členů skupiny. Na grafu 5.16 jsou pak blíže vidět rozdíly rychlostí podepisování u klasických schémat a *G-Merkle*(+). Pro klasické skupinové podpisy je tato operace téměř konstantní, pro *G-Merkle*(+) nepatrně narůstá. Lze však tvrdit, že je *G-Merkle+* svou efektivitou podepisování porovnatelný k efektivitě klasických skupinových podpisů až do velikosti skupiny 2^{14} při $B = 1$.



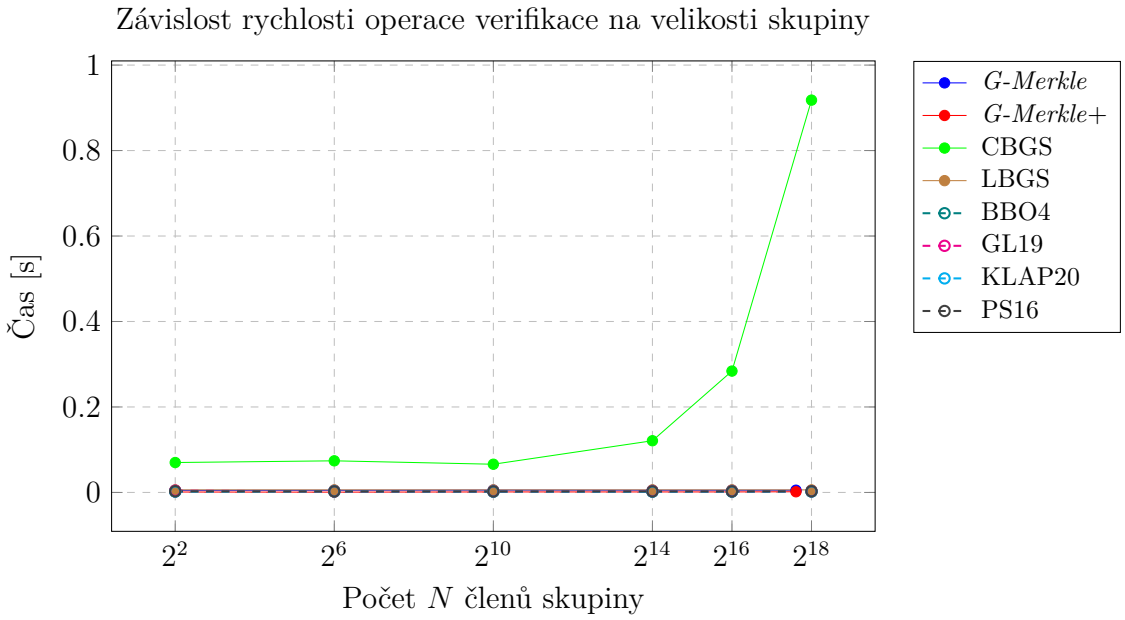
Obr. 5.15: Graf závislosti rychlosti operace podpisu na počtu členů skupiny.



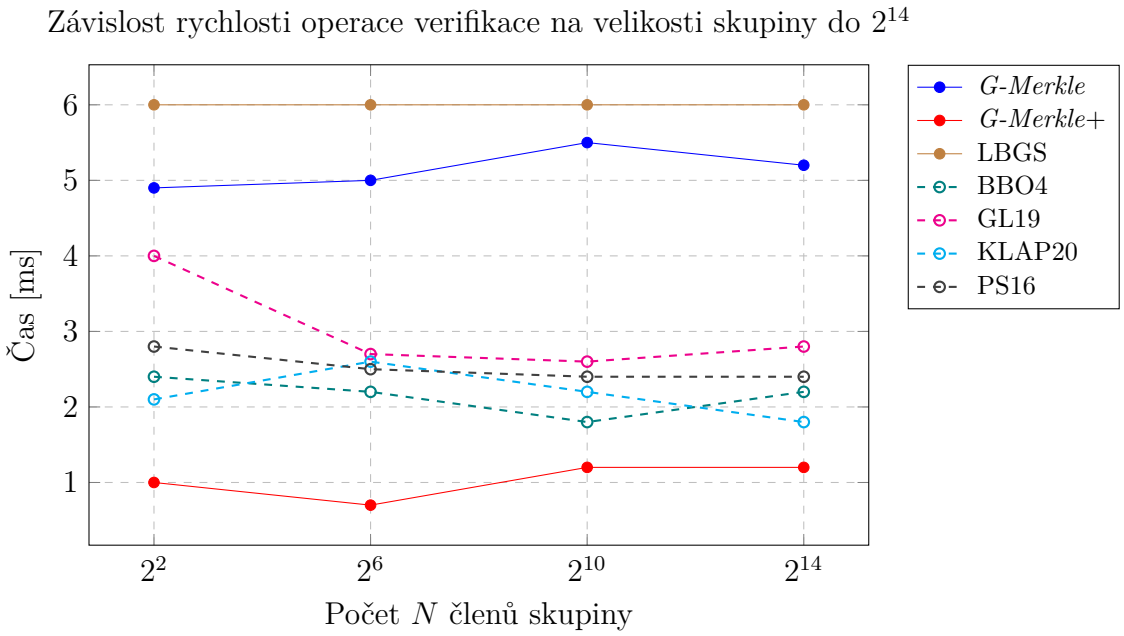
Obr. 5.16: Graf závislosti rychlosti operace podpisu na počtu členů skupiny do 2^{14}.

Fáze verifikace podpisu

Na obr. 5.17 a 5.18 jsou zobrazeny grafy porovnávající operaci verifikace měřených schémat. Z prvního grafu je zřejmé, že nejpomalejší je verifikace v rámci implementace CBGS. Časová náročnost operace je totiž lineárně závislá na počtu členů skupiny. Verifikace ostatních měřených schémat je násobně kratší. Tyto průběhy jsou blíže znázorněny na grafu 5.18. Pro klasické SP se operace nejčastěji pohybuje mezi 2 a 3 ms. Je zajímavé, že je verifikace nejrychlejší pro postkvantový skupinový podpis *G-Merkle+* s W-OTS+ navzdory tomu, že se skládá ze dvou podoperací – ověření OTS a výpočtu autentizační cesty stromem, která v sobě zahrnuje počet nutných hashování v závislosti na výšce stromu. Verifikace se v rámci této implementace pohybuje okolo 1 ms pro všechny měřené velikosti skupiny. Varianta *G-Merkle* s WOTS se pak kvůli pomalejšímu ověřování OTS pohybuje okolo hranice 5 ms. Z pohledu verifikace je tedy *G-Merkle+* nejefektivnější z měřených schémat SP pro $B = 1$.



Obr. 5.17: Graf závislosti rychlosti verifikace podpisu na počtu členů skupiny.

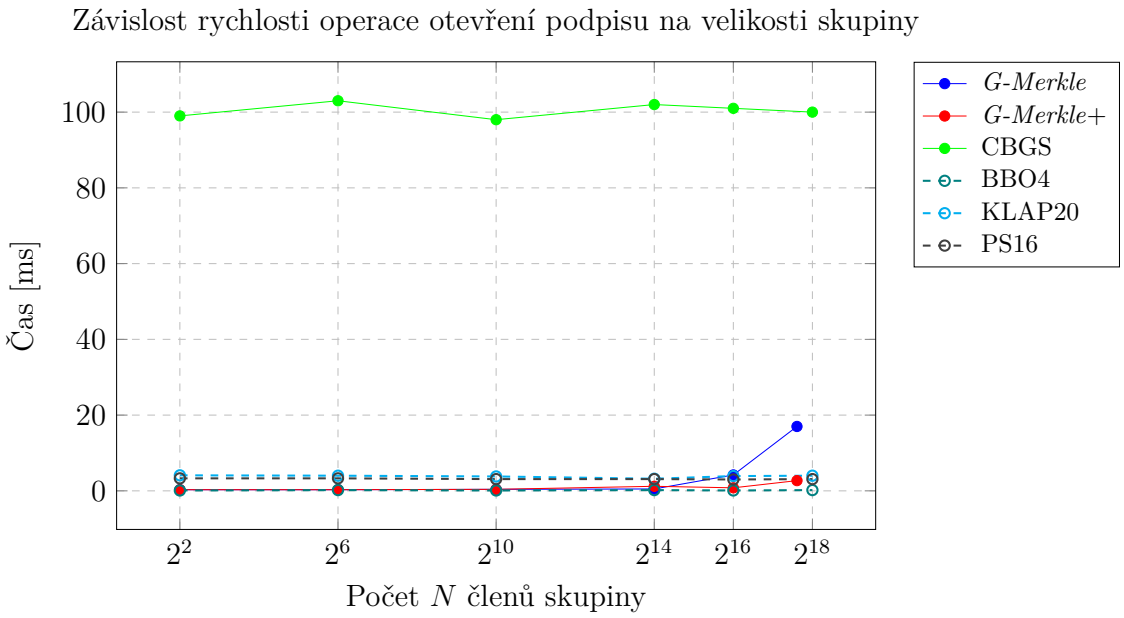


Obr. 5.18: Graf závislosti rychlosti verifikace podpisu na počtu členů skupiny do 2^{14}.

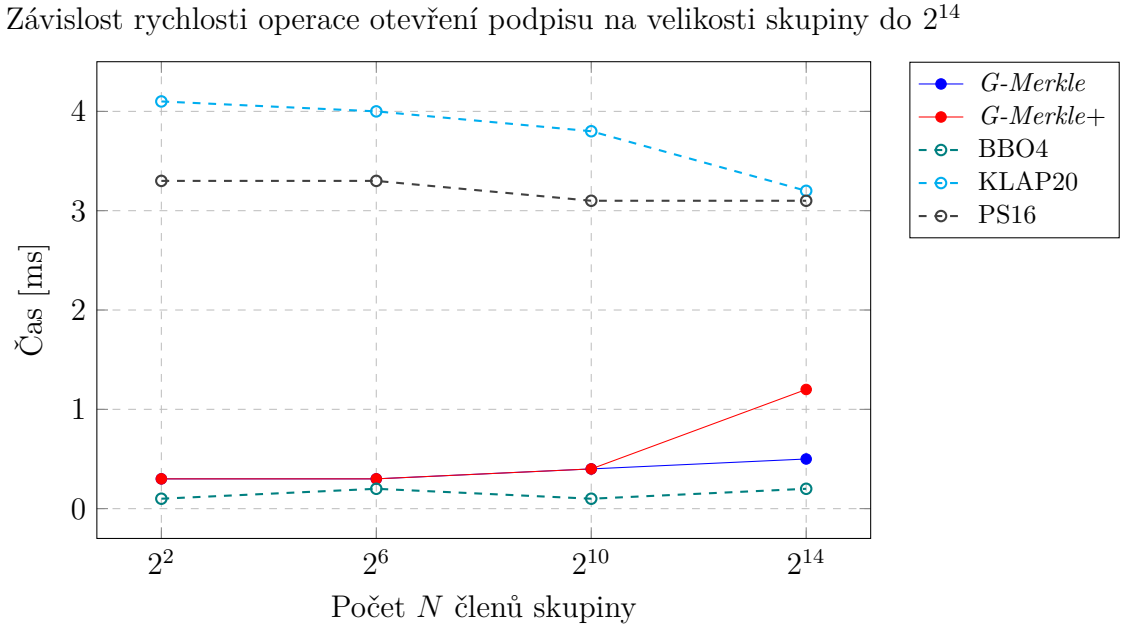
Fáze otevírání podpisu

V grafech na obr. 5.19 a 5.20 je znázorněno porovnání rychlosti průběhu operace otevírání podpisů. Implementace LBGS a GL19 neumožňují otevření podpisů, proto je nebylo možné změřit a nejsou zahrnuty v grafech.

Nejméně efektivní je operace otevírání podpisu v rámci implementace CBGS, která se průměrně pohybuje okolo 100 ms. Naopak nejrychlejší otevírání podpisů podporuje schéma BB04, průměrně 0,15 ms. Pro SP *G-Merkle*(+) je operace otevírání také velice rychlá v porovnání s ostatními, ačkoliv je implementačně zpomalena při vyšších počtech OTS párů klíčů z důvodu prohledávání prostoru listů, jak bylo popsáno v kap. 5.2.1. Otevření podpisu, tedy revokace anonymity v rámci KLAP20 a PS16 probíhá asi 3 – 4 násobně déle.



Obr. 5.19: Graf závislosti rychlosti otevírání podpisu na počtu členů skupiny.



Obr. 5.20: Graf závislosti rychlosti otevírání podpisu na počtu členů skupiny do 2^{14}.

5.2.5 Shrnutí

V této kapitole byla porovnána schémata klasických skupinových podpisů pomocí knihovny `pygroupsig` (Python wrapper `libgroupsig`), implementace schématu SP *G-Merkle*(+) a dvě externě získané implementace – SP založený na teorii kódování v prog. jazyce C++ a SP založený na mřížkách v prog. jazyce C. Na základě naměřených dat implementací postkvantových i klasických skupinových podpisů byly orientačně srovnány paměťové a výpočetní nároky těchto schémat.

Z paměťového hlediska jsou velikosti parametrů postkvantových skupinových podpisů nejčastěji neporovnatelně vyšší než u klasických skupinových podpisů. Nejmenší parametry z postkvantových SP má schéma *G-Merkle*, za kterým následuje SP *G-Merkle+*. Menší velikost G_{PK} a výsledného podpisu je však vykoupena nevýhodou, která vyplývá z podstaty vícenásobných podpisů založených na hashích, a to nutnost využití jednorázových párů klíčů. Překážkou využívání tohoto postkvantového skupinového podpisu tedy můžou být zejména paměťové nároky kladené na členy skupiny. Pokud by to bezpečnost povolovala a bylo by tento problém možné vyřešit např. postupným generováním soukromých klíčů pomocí tajného *seed*u a postupným předáváním veřejných klíčů manažerovi skupiny, mohly by být tyto výpočetní nároky sníženy. CBGS i LBGS mají i více než tisícinásobně vyšší velikosti některých parametrů. Je tedy zřejmé, že pokud je cílem implementovat skupinové podpisy na paměťově omezená zařízení, bude to poměrně velký problém při současném stavu techniky.

Z časového hlediska jsou rychlosti operací skupinových podpisů *G-Merkle*(+) porovnatelné s naměřenými hodnotami klasických schémat skupinových podpisů, ne-li rychlejší (např. operace verifikace *G-Merkle+*). Implementace *G-Merkle* je nepatrně pomalejší v závislosti na využití W-OTS při operacích podepisování a verifikace. CBGS je i při malých velikostech skupiny násobně pomalejší než ostatní měřené skupinové podpisy, z čehož vyplývá, že je jak paměťově, tak i časově náročnější než ostatní implementace. Téměř všechny jeho parametry jsou také lineárně závislé na velikosti skupiny. LBGS sice nebylo možné plně změřit z důvodu stylu, jakým bylo schéma implementováno, nicméně naměřené hodnoty jsou i pro skupinu o jednom členu vysoké a schéma není zajímavé ani výpočetním časem jednotlivých operací.

Z informací zjištěných v rámci této kapitoly tedy lze odvodit, že pokud je hlavním cílem schématu umožňovat svižné operace, jsou postkvantové skupinové podpisy založené na hashovacích funkcích využitelné v praxi. Jedinou překážkou využití těchto podpisů při současném stavu techniky jsou patrně velikosti jednorázových párů klíčů, které je nutné uložit na straně člena skupiny. Řešení tohoto problému tedy zůstává otevřenou otázkou pro navazující publikace.

Data využitá k tvorbě grafů efektivity implementací jsou k dispozici v příloze D.

Závěr

Tato diplomová práce se zabývala moderními technologiemi podporujícími ochranu soukromí (PETs), které jsou také založeny na problémech odolných vůči kvantové kryptoanalýze – zejména postkvantové skupinové podpisy a atributová schémata.

Mezi cíle práce patřilo seznámení se se zmíněnými okruhy kryptografie a následně analýza existujících metod a současného stavu techniky v rámci průniku těchto dvou problematik. Součástí analýzy je i zhodnocení paměťové a výpočetní náročnosti postkvantových a klasických PETs schémat. Cílem praktické části práce bylo vytvoření funkční implementace kvantově bezpečné kryptografické metody, která poskytuje ochranu soukromí a reálné zhodnocení jejích výpočetních a paměťových nároků.

Teoretická část práce se v kap. 1 zabývá kryptosystémy s ochranou soukromí a stručně představuje atributová autentizační schémata, skupinové podpisy a technologii homomorfního šifrování.

Rodiny postkvantové kryptografie jsou následně popsány v kap. 2. Konkrétně se jedná o kryptografii založenou na hash funkcích spolu s popisem schémat jednorázových podpisů Lamport a Winternitz (kap. 2.1), kryptografii založenou na teorii kódování spolu s popisem kryptosystému McEliece (kap. 2.2), kryptografii založenou na mřížkách se schématem NTRU (kap. 2.3), kryptografii založenou na polynomiálních rovnicích (kap. 2.4) a kryptografii využívající isogenii supersingulárních eliptických křivek (kap. 2.5).

Další směr práce byl následně určen na základě průzkumu vědeckých publikací dedikovaných PETs, postkvantové kryptografii a jejich kombinaci v kap. 3. Z výsledků bylo zjištěno, že nejčastěji publikovaným typem postkvantových schémat s ochranou soukromí jsou skupinové podpisy, proto jsou zbylé kapitoly práce zaměřeny zejména na tento typ postkvantových PETs.

Analýza současného stavu techniky byla provedena v kap. 4 a vycházela z nalezených publikací. V kap. 4.1 byly popsány nejrelevantnější a nejvíce citovaná publikovaná schémata skupinových podpisů (kap. 4.1.1), atributových podpisů (kap. 4.1.2) a atributových pověření (kap. 4.1.3) založených na problémech mřížek. Následují stručné popisy publikací skupinových podpisů založených na hash funkcích (kap. 4.2), teorii kódování (kap. 4.3) a polynomiálních rovnicích (kap. 4.4). Jelikož se většina schémat napříč rodinami postkvantové kryptografie velmi liší ve své konstrukci a publikace často neobsahovaly ani orientační informace ohledně paměťové a výpočetní náročnosti, bylo v kap. 4.5 provedeno volné porovnání a zhodnocení efektivity na základě dostupných omezených informací. Postkvantová schémata byla porovnávána oproti referenční implementaci klasických skupinových podpisů, která byla vytvořena pomocí knihovny `libgroupsig` výhradně za tímto účelem. V kap. 4.5.1 jsou také nastíněny současné problémy výzkumu a aplikací skupinových podpisů vyplý-

vající z nedostatku praktických implementací a knihoven.

Poslední kapitola práce (kap. 5) se věnuje implementacím postkvantových schémat. Nejprve došlo k průzkumu dostupných knihoven, projektů a implementací. Bylo zjištěno, že existují pouze knihovny a implementace podporující schémata, která se účastnila soutěže postkvantové standardizace NIST. Jedna z nalezených knihoven (`pqcrypto`) byla využita k implementaci a porovnání postkvantových podpisů v kap. 5.1. Z důvodu nedostatku implementací a knihoven v rámci okruhu postkvantové kryptografie s ochranou soukromí byly oslovovány vědecké týmy, jejichž publikovaná schémata byla popisována v rámci kap. 4. Celkem bylo osloveno 21 vědeckých týmů a byly získány dvě implementace postkvantových skupinových podpisů. Jednalo se zástupce z rodiny mřížek (kap. 5.2.3) a teorie kódování (kap. 5.2.2). Proto bylo zvoleno třetí schéma postkvantového skupinového podpisu z rodiny hash funkcí a toto schéma bylo podrobně popsáno, naimplementováno a změřeno (kap. 5.2.1). Implementace skupinového podpisu byla vytvořena v programovacím jazyce Python a byla koncipována modulárně formou knihovny, umožňuje tedy jednoduché a intuitivní používání. Byly také přiloženy dva soubory k demonstraci využití naprogramovaného schématu; soubor s předpřipravenými scénáři testování skupinového podpisu a interaktivní verze skupinového podpisu. Následně byly naměřeny i získané implementace vědeckých skupin a tyto hodnoty byly porovnány oproti sobě, i oproti implementacím klasických skupinových podpisů vytvořených na základě knihovny `libgroupsig`. Všechny implementace byly měřeny na stejné výpočetní platformě – virtuálním stroji Linux Ubuntu 21.10 s dvěma procesory po jednom jádru @ 3,8 GHz a 8 GB RAM. Výsledky porovnání jsou graficky zobrazeny a slovně popsány v rámci kap. 5.2.4.

Výstupem praktické části diplomové práce je tedy zejména implementace postkvantového skupinového podpisu založeného na hash funkcích. Implementace je podložena měřením reálné paměťové a výpočetní náročnosti, které bylo provedeno za účelem porovnání i u získaných implementací skupinových podpisů, které jsou založeny na problémech mřížek a teorie kódování. Byly využity i implementace klasických skupinových podpisů založených na teorii čísel, aby bylo možné porovnat efektivitu postkvantových skupinových podpisů v kontextu prekvantového a postkvantového vztahu.

Cíle diplomové práce byly dosaženy. V rámci budoucího výzkumu a vývoje v této oblasti je očekáváno zejména zaměření se na tvorbu efektivnějších důkazů nulové znalosti a jejich následná aplikace v konstrukcích mřížkových schémat. V rámci rodiny hash funkcí je naopak očekávána tvorba lepších konstrukcí schémat, která snižuje bezpečnostní požadavky kladenené na využité hash funkce, popřípadě i vývoj efektivnějších schémat jednorázových podpisů.

Literatura

[1] YANG, H., OLESHCHUK, V. *Attribute-Based Authentication Schemes: A Survey.* International Journal of Computing, vol. 14, s. 86–96. 2015.

[2] KHADER, D. *Attribute based authentication schemes.* Bath, 2009. Disertační práce. University of Bath, Department of Computer Science. Vedoucí práce Russell Bradford. Dostupné z URL: <`https://researchportal.bath.ac.uk/en/studentTheses/attribute-based-authentication-schemes`>.

[3] DZURENDA, P., HAJNÝ, J., MALINA, L., RICCI, S. Anonymous Credentials with Practical Revocation using Elliptic Curves. *Proceedings of the 14th International Joint Conference on e-Business and Telecommunications – SECRYPT.* ICETE 2017, vol. 4, s. 534–539. Scitepress, 2017. ISBN 978-989-758-259-2.

[4] HAJNÝ, J. *Authentication protocols and privacy protection.* Brno, 2012. Dostupné z URL: <`https://dspace.vutbr.cz/bitstream/handle/11012/16096/thesis-1.pdf`>.

[5] CHAUM, D., VAN HEYST, E. Group Signatures. *Davies D.W. (Ed.): Advances in Cryptology – EUROCRYPT '91.* Lecture Notes in Computer Science, vol. 547, s. 257–265. Springer, Berlin, Heidelberg, 1991. ISBN 978-3-540-54620-7.

[6] CAMENISCH, J., STADLER, M. Efficient group signature shcemes for large groups. *Kaliski B.S. (Ed.): Advances in Cryptology – CRYPTO '97.* Lecture Notes in Computer Science, vol. 1294, s. 410–424. Springer, Berlin, Heidelberg, 1997. ISBN 978-3-540-63384-6.

[7] FRANKLIN, M., ZHANG, H. Unique Group Signatures. *Foresti S., Yung M., Martinelli F. (Eds.): Computer Security – ESORICS 2012.* Lecture Notes in Computer Science, vol. 7459, s. 643–660. Springer, Berlin, Heidelberg, 2012. ISBN 978-3-642-33166-4.

[8] BELLARE, M., MICCIANCIO, D., WARINSCHI, B. Foundations of Group Signatures: Formal Definitions, Simplified Requirements, and a Construction Based on General Assumptions. *Biham E. (Ed.): Advances in Cryptology – EUROCRYPT '03.* Lecture Notes in Computer Science, vol. 2656, s. 614–629. Springer, Berlin, Heidelberg, 2003. ISBN 978-3-540-14039-9.

[9] SHI, Y., ZHAO, Q., FAN, H., LIU, Q. *Secure Obfuscation for Encrypted Group Signatures.* PLoS ONE 10(7): e0131550, 2015.

[10] PERERA, M. N. S., KOSHIBA, T. Achieving Full Security for Lattice-Based Group Signatures with Verifier-Local Revocation. *Naccache, D. et al. (Eds.): ICICS 2018*. Lecture Notes in Computer Science, vol. 11149, s. 287–302. Springer, Cham, 2018. ISBN 978-3-030-01950-1.

[11] OGBURN, M., TURNER, C., DAHAL, P. *Homomorphic Encryption.* Procedia Computer Science, vol. 50, s. 502–509. 2013. ISSN 1877-0509.

[12] *Introduction*. Homomorphic Encryption Standardization. [online], [cit. 2021-10-12]. Dostupné z URL: <`https://homomorphicencryption.org/introduction/`>.

[13] ARAMPATZIS, A. *Homomorphic Encryption: What Is It and How Is It Used*. Venafi. [online]. 2020, poslední aktualizace 22. 1. 2020, [cit. 2021-10-12]. Dostupné z URL: <`https://www.venafi.com/blog/homomorphic-encryption-what-it-and-how-it-used`>.

[14] GENTRY, C. *Fully homomorphic encryption using ideal lattices.* Stanford, 2009. Disertační práce. Stanford University, Department of Computer Science. Vedoucí práce Dan Boneh. Dostupné z URL: <`https://crypto.stanford.edu/craig/craig-thesis.pdf`>.

[15] ARMKNECHT, F., BOYD, C., CARR, CH., GJØSTEEN, K., JÄSCHKE, A., REUTER, CH., STRAND, M. *A Guide to Fully Homomorphic Encryption.* IACR Cryptology ePrint Archive. 2015.

[16] AZOUGAGHE, A., HEDABOU, M., BELKASMI, M. *An electronic voting system based on homomorphic encryption and prime numbers.* 11th International Conference on Information Assurance and Security (IAS), s. 140–145. IEEE, Marrakech, Morocco, 2015. ISBN 978-1-4673-8715-6.

[17] BEULLENS, W., D'ANVERS, J. P., HÜLSING, A., LANGE, T., PANNY, L., DE SAINT GUILHEM, C., SMART, N. *Post-Quantum Cryptography Current State and Quantum Mitigation.* European Union Agency for Cybersecurity (ENISA), 2021. ISBN 978-92-9204-468-8.

[18] CORRIGAN-GIBBS, H., KIM, S., WU, D. *Lecture 12: Post-Quantum Cryptography and Hash-based Signatures*. CS 355 – Topics in Cryptography. Stanford University. [online]. 2021, poslední aktualizace 9. 5. 2018, [cit. 2021-10-18]. Dostupné z URL: <`https://crypto.stanford.edu/cs355/18sp/lec12.pdf`>.

[19] ZEMAN, V. *Hašovací funkce.* Prezentace předmětu Kryptografie 2020/2021. Informační bezpečnost, Fakulta elektrotechniky a komunikačních technologií, Vysoké učení technické v Brně. 2020.

[20] BUCHANAN, W. *Quantum Robust: Winternitz one time signature scheme (W-OTS).* A security site. [online]. 2021, [cit. 2021-10-18]. Dostupné z URL: <`https://asecuritysite.com/encryption/wint`>.

[21] HARSHDEEP, S. *Code based Cryptography: Classic McEliece.* Computing Research Repository. [online]. 2019, poslední aktualizace 29. 5. 2020, [cit. 2021-10-18]. Dostupné z URL: <`https://arxiv.org/pdf/1907.12754.pdf`>.

[22] ZEMAN, V. *Postkvantové kryptografické systémy.* Prezentace předmětu Kryptografie 2020/2021. Informační bezpečnost, Fakulta elektrotechniky a komunikačních technologií, Vysoké učení technické v Brně. 2020.

[23] BERNSTEIN, D. J., LANGE, T., PETERS, C. Attacking and defending the McEliece cryptosystem. *Buchmann, J., Ding, J. (Eds.): Post-Quantum Cryptography – PQCrypto 2008.* Lecture Notes in Computer Science, vol. 5299, s. 31–46. Springer, Berlin, Heidelberg, 2008. ISBN 978-3-540-88402-6.

[24] MICCIANCIO, D., REGEV, O. Lattice-based Cryptography. *Bernstein, D. J., Buchmann J., Dahmen, E. (Eds.): Post-Quantum Cryptography*, s. 147–191. Springer, Berlin, Heidelberg, 2009. ISBN 978-3-540-88701-0.

[25] HOFFSTEIN, J., PIPHER, J., SILVERMAN, J. H. NTRU: A ring-based public key cryptosystem. *Buhler J.P. (Eds.): Algorithmic Number Theory.* ANTS 1998. Lecture Notes in Computer Science, vol. 1423, s. 267–288. Springer, Berlin, Heidelberg, 1998. ISBN 978-3-540-64657-0.

[26] HOFFSTEIN, J., HOWGRAVE-GRAHAM, N., PIPHER, J., SILVERMAN, J. H., WHYTE, W. NTRUSign: Digital Signatures Using the NTRU Lattice. *Joye M. (Eds.): Topics in Cryptology – CT-RSA 2003.* Lecture Notes in Computer Science, vol. 2612, s. 122–140. Springer, Berlin, Heidelberg, 2003. ISBN 978-3-540-00847-7.

[27] STEHLÉ, D., STEINFELD, R. Making NTRUEncrypt and NTRUSign as Secure as Standard Worst-Case Problems over Ideal Lattices. *Paterson K.G. (Eds.): Advances in Cryptology – EUROCRYPT 2011.* Lecture Notes in Computer Science, vol. 6632, s. 27–47. Springer, Berlin, Heidelberg, 2011. ISBN 978-3-642-20464-7.

[28] DIVIŠOVÁ, J. *Kryptografie založená na mřížkách.* Praha: Univerzita Karlova v Praze, Matematicko–fyzikální fakulta, Katedra algebry, 2010. 59 s. Diplomová práce. Vedoucí práce: RNDr. David Stanovský, Ph.D.

[29] POLÁKOVÁ, K. *Kryptosystém NTRU a jeho varianty.* Praha: Univerzita Karlova v Praze, Matematicko–fyzikální fakulta, Katedra algebry, 2015. 58 s. Diplomová práce. Vedoucí práce: Mgr. Pavel Příhoda, Ph.D.

[30] HOFFSTEIN, J., HOWGRAVE-GRAHAM, N., PIPHER, J., SILVERMAN, J. H., WHYTE, W. Practical Lattice-Based Cryptography: NTRUEncrypt and NTRUSign. *Nguyen P., Vallée B. (Eds.): The LLL Algorithm.* Information Security and Cryptography, s. 349–390. Springer, Berlin, Heidelberg, 2009. ISBN 978-3-642-02294-4.

[31] SILVERMAN, J. H. *NTRU and Lattice-Based Crypto: Past, Present, and Future.* The Mathematics of Post-Quantum Cryptography DIMACS Center, Rutgers University, 12.–16. leden 2015. Dostupné z URL: <`http://archive.dimacs.rutgers.edu/Workshops/Post-Quantum/Slides/Silverman.pdf`>.

[32] GOUBIN, L., PATARIN, J., YANG, B. Y. Multivariate Cryptography. *Tillborg, H., Jajodia, S. (Eds.): Encyclopedia of Cryptography and Security*, s. 824–828. Springer, Berlin, Heidelberg, 2011. ISBN 978-1-4419-5905-8.

[33] JAO, D. DE FEO, L. Towards Quantum-resistant Cryptosystems from Supersingular Elliptic Curve Isogenies. *Yang BY. (Eds.): Post-Quantum Cryptography – PQCrypto 2011.* Lecture Notes in Computer Science, vol. 7071, s. 19–39. Springer, Berlin, Heidelberg, 2011. ISBN 978-3-642-25404-8.

[34] COSTELLO, C. Supersingular Isogeny Key Exchange for Beginners. *Paterson K., Stebila D. (Eds.): Selected Areas in Cryptography – SAC 2019.* Lecture Notes in Computer Science, vol. 11959, s. 21–50. Springer, Berlin, Heidelberg, 2019. ISBN 978-3-030-38470-8.

[35] COOK, J. D. *What is an isogeny?* John D. Cook Consulting. [online]. 2019, poslední aktualizace 21. 4. 2019, [cit. 2021-10-28]. Dostupné z URL: <`https://www.johndcook.com/blog/2019/04/21/what-is-an-isogeny/`>.

[36] STRATIL, P., HASEGAWA, S., SHIZUYA, H. *Supersingular Isogeny-based Cryptography: A Survey.* Interdisciplinary Information Sciences, vol. 27, s. 1–23. 2021. ISSN 1340-9050.

[37] GORDON, S. D., KATZ, J., VAIKUNTANATHAN, V. *A group signature scheme from lattice assumptions.* Proceedings of Conference ASIACRYPT 2010, s. 395–412. Springer, Singapore. 2010

[38] CAMENISCH, J., NEVEN, G., RÜCKERT, M. *Fully Anonymous Attribute Tokens from Lattices.* IACR Cryptology ePrint Archive. 2012.

[39] LAGUILLAUMIE, F., LANGLOIS, A., LIBERT, B., STEHLÉ, D. *Lattice-Based Group Signatures with Logarithmic Signature Size*. IACR Cryptology ePrint Archive. 2013.

[40] MICCIANCIO, D. *Random Lattices and Lattice-Based Cryptography*. CSE 206A: Lattice Algorithms and Applications, UCSD CSE. 2019. Dostupné z URL: <`https://cseweb.ucsd.edu/classes/fa19/cse206A-a/Lec4-Random.pdf`>.

[41] LANGLOIS, A., LING, S., NGUYEN, K., WANG, H. *Lattice-based Group Signature Scheme with Verifier-local Revocation*. IACR Cryptology ePrint Archive. 2014.

[42] BONEH, D., SHACHAM, H. *Group Signatures with Verifier-Local Revocation*. CCS '04: Proceedings of the 11th ACM Conference on Computer and Communications Security, s. 168–177. Association for Computing Machinery, Washington DC, USA. 2004. ISBN: 1581139616

[43] NGUYEN, P. Q., ZHANG, J., ZHANG, Z. *Simpler Efficient Group Signatures from Lattices*. IACR Cryptology ePrint Archive. 2015.

[44] LING, S., NGUYEN, K., WANG, H. *Group Signatures from Lattices: Simpler, Tighter, Shorter, Ring-based*. IACR Cryptology ePrint Archive. 2015.

[45] LIBERT, B., LING, S., NGUYEN, K., WANG, H. *Zero-Knowledge Arguments for Lattice-Based Accumulators: Logarithmic-Size Ring Signatures and Group Signatures without Trapdoors*. IACR Cryptology ePrint Archive. 2016.

[46] LIBERT, B., LING, S., MOUHARTEM, F., NGUYEN, K., WANG, H. *Signature Schemes with Efficient Protocols and Dynamic Group Signatures from Lattice Assumptions*. IACR Cryptology ePrint Archive. 2016.

[47] LIBERT, B., MOUHARTEM, F., NGUYEN, K. *A Lattice-Based Group Signature Scheme with Message-Dependent Opening*. ACNS 2016: 14th International Conference on Applied Cryptography and Network Security. Guildford, United Kingdom. 2016.

[48] LING, S., NGUYEN, K., WANG, H., XU, Y. *Lattice-Based Group Signatures: Achieving Full Dynamicity with Ease*. IACR Cryptology ePrint Archive. 2017.

[49] LING, S., NGUYEN, K., WANG, H., XU, Y. *Constant-size Group Signatures from Lattices*. IACR Cryptology ePrint Archive. 2018.

[50] DEL PINO, R., LYUBASHEVSKY, V., SEILER, G. *Lattice-Based Group Signatures and Zero-Knowledge Proofs of Automorphism Stability*. IACR Cryptology ePrint Archive. 2018.

[51] LING, S., NGUYEN, K., WANG, H., XU, Y. *Lattice-based group signatures: Achieving full dynamicity (and deniability) with ease*. Theoretical Computer Science, vol. 783, s. 71–94. 2019. ISSN 0304-3975.

[52] LING, S., NGUYEN, K., WANG, H., XU, Y. *Forward-Secure Group Signatures from Lattices*. [online]. 2019, poslední aktualizace 24. 1. 2019, [cit. 2022-03-09]. Dostupné z URL: <`https://arxiv.org/pdf/1801.08323.pdf`>.

[53] XIE, R., HE, C., XU, C., GAO, C. *Lattice-based dynamic group signature for anonymous authentication in IoT*. Annals of Telecommunications, vol. 74, s. 531–542. 2019.

[54] LUO, Q., JIANG, C. *A New Constant-Size Group Signature Scheme From Lattices*. IEEE Access, vol. 8, s. 10198–10207. 2020.

[55] LYUBASHEVSKY, V., NGUYEN, N. K., PLANCON, M., SEILER, G. *Shorter Lattice-Based Group Signatures via „Almost Free“ Encryption and Other Optimizations*. IACR Cryptology ePrint Archive. 2021

[56] SUN, Y., LIU, Y. *An efficient fully dynamic group signature with message dependent opening from lattice*. Cybersecurity 4, 15. Springer Open. 2021.

[57] ZHANG, Y., LIU, X., HU, Y., JIA, H., ZHANG, Q. *An Improved Group Signature Scheme with VLR over Lattices*. Security and Communication Networks, vol. 2021. Hindawi. 2021.

[58] KATSUMATA, S., YAMADA, S. *Group Signatures without NIZK: From Lattices in the Standard Model*. IACR Cryptology ePrint Archive. 2019.

[59] SUN, Y., LIU, Y. A Lattice-Based Fully Dynamic Group Signature Scheme Without NIZK. *Wu Y. (Ed.): Information Security and Cryptology - Inscrypt 2020*. Lecture Notes in Computer Science, vol. 12612, s. 359–367. Springer. 2021.

[60] CASH, D., HOFHEINZ, D., KILTZ, E., PEIKERT, C. *Bonsai Trees, or How to Delegate a Lattice Basis*. IACR Cryptology ePrint Archive. 2010.

[61] WANG, Q., CHEN, S. *Attribute-based signature for threshold predicates from lattices*. Security and Communication Networks, vol. 8, s. 811–821. 2014.

[62] MAO, X. P., CHEN, K. F., LONG, Y., WANG, L. L. *Attribute-based signature on lattices*. Journal of Shanghai Jiaotong University (Science), vol. 19, s. 406–411. 2014.

[63] WANG, Q., CHEN, S., GE, A. A New Lattice-Based Threshold Attribute-Based Signature Scheme. *Lopez J., Wu, Y. (Eds.): ISPEC 2015*. Lecture Notes in Computer Science, vol. 9065, s 406–420. Springer. 2015. ISBN 978-3-319-17533-1.

[64] JIA, X., HU, Y., JUNTAO, G, WEN, G., XUELIAN, L. *Attribute-based signatures on lattices*. The Journal of China Universities of Posts and Telecommunications, vol. 23, s. 83–90. 2016.

[65] EL BANSARKHANI, R., EL KAAFARANI, A. *Post-Quantum Attribute-Based Signatures from Lattice Assumptions*. IACR Cryptology ePrint Archive. 2016.

[66] EL KAAFARANI, A., KATSUMATA, S. *Attribute-based Signatures for Unbounded Circuits in the ROM and Efficient Instantiations from Lattices*. IACR Cryptology ePrint Archive. 2018

[67] ZHANG, Y., LIU, X., HU, Y, ZHANG, Q., JIA, H. *Attribute-Based Signatures for Inner-Product Predicate from Lattices*. CSS 2019: Proceedings of Cyberspace Safety and Security, 11th International Symposium, s. 173–185. Guangzhou, China. 2020.

[68] LI, P., LAI, J., WU, Y. *Publicly Traceable Attribute-Based Anonymous Authentication and Its Application to Voting*. Security and Communication Networks, vol. 2021. Hindawi. 2021.

[69] EL BANSARKHANI, R., MISOCZKI, R. G-Merkle: A Hash-Based Group Signature Scheme from Standard Assumptions. *Lange, T., Steinwandt R. (Eds.): Post-Quantum Cryptography 2018.* Lecture Notes in Computer Science, vol. 10786, s. 441—463. Springer, Cham, 2021. ISBN 978-3-319-79063-3.

[70] SHAFIEINEJAD, M., ESFAHANI, N. N. *A Scalable Post-quantum Hash-Based Group Signature*. IACR Cryptology ePrint Archive. 2019.

[71] BUSER, M., LIU, J. K., STEINFELD, R., SAKZAD, A., SUN, S. F. DGM: A Dynamic and Revocable Group Merkle Signature. *Sako, K. et al. (Eds.): Computer Security – ESORICS 2019*. Lecture Notes in Computer Science, vol. 11735, s. 194–214. Springer, Cham, 2019. ISBN 978-3-030-29959-0.

[72] YEHIA, M., ALTAWY, R., GULLIVER, T. A. GMMT: A Revocable Group Merkle Multi-tree Signature Scheme. *Conti, M., Stevens, M., Krenn, S. (Eds.): Cryptology and Network Security – CANS 2021.* Lecture Notes in Computer Science, vol. 13099, s. 136–157. Springer, Cham, 2021. ISBN 978-3-030-92548-2.

[73] EZERMAN, M. F., LEE, H. T., LING, S., NGUYEN, K., WANG, H. *A Provably Secure Group Signature Scheme from Code-Based Assumptions.* IACR Cryptology ePrint Archive. 2015.

[74] EZERMAN, M. F., LEE, H. T., LING, S., NGUYEN, K., WANG, H. *Provably Secure Group Signature Scheme from Code-Based Assumptions.* IEEE Transactions on Information Theory, vol. 66, s. 5754–5773. IEEE, 2020. ISSN 1557-9654.

[75] YANG, G., TANG, S., YANG, L. *A Novel Group Signature Scheme Based on MPKC.* IACR Cryptology ePrint Archive. 2011.

[76] KUNDU, N., DEBNATH, S. K., MISHRA, D. *A secure and efficient group signature scheme based on multivariate public key cryptography.* Journal of Information Security and Applications, vol. 58. 2021. ISSN 2214-2126.

[77] MENDRICK, B. L. *Practically Efficient Group Signature Scheme.* Pittsburgh, PA. 2019. Diplomová práce. Carnegie Mellon University, Department of Computer Science. Vedoucí práce Takanori Isobe. Dostupné z URL: <`https://s3-eu-west-1.amazonaws.com/pstorage-cmu-348901238291901/20277681/Mendrick_cmu_00410_10486.pdf`>.

[78] DIAZ, J., ARROYO, D., RODRIGUEZ, F. B. *`libgroupsig`: An extensible C library for group signatures.* IACR Cryptology ePrint Archive. 2015.

[79] KIAYIAS, A., TSIOUNIS, Y., YUNG, M. *Traceable signatures.* Eurocrypt 2004, s. 571–589. 2004.

[80] CHOI, S. G., PARK, K., YUNG, M. *Short traceable signatures based on bilinear pairings.* IWSEC 2006, s. 88–103. 2006.

[81] BONEH, D., BOYEN, X., SHACHAM, H. *Short group signatures.* CRYPTO 2004, s. 41–55. 2004.

[82] DIAZ, J., LEHMANN, A. *Group Signatures with User-Controlled and Sequential Linkability.* IACR Cryptology ePrint Archive. 2021.

[83] KIM, H., LEE, Y., ABDALLA, M., PARK J. H. *Practical Dynamic Group Signature with Efficient Concurrent Joins and Batch Verifications.* IACR Cryptology ePrint Archive. 2020.

[84] GARMS, L., LEHMANN, A. *Group Signatures with Selective Linkability.* IACR Cryptology ePrint Archive. 2019.

[85] POINTCHEVAL, D., SANDERS, O. *Short Randomizable Signatures.* IACR Cryptology ePrint Archive. 2015.

[86] YANG, R., AU, M. H., ZHANG, Z., XU, Q., YU, Z., WHYTE, W. *Efficient latticebased zero-knowledge arguments with standard soundness: Construction and applications.* CRYPTO 2019. Lecture Notes in Computer Science, vol. 11692, s. 147–175. Springer. 2019.

[87] BOOTLE, J., LYUBASHEVSKY, V., SEILER, G. *Algebraic techniques for short (er) exact lattice-based zero-knowledge proofs.* CRYPTO 2019. Lecture Notes in Computer Science, vol. 11692, s. 176–202. Springer. 2019.

[88] ESGIN, M. F., STEINFELD, R., LIU, J. K., LIU, D. *Lattice-based zero-knowledge proofs: New techniques for shorter and faster constructions and applications.* CRYPTO 2019. Lecture Notes in Computer Science, vol. 11692, s. 115–146. Springer. 2019.

[89] ESGIN, M. F., STEINFELD, R., SAKZAD, A., LIU, J. K., LIU, D. *Short lattice-based one-out-of-many proofs and applications to ring signatures.* ACNS 2019. Lecture Notes in Computer Science, vol. 11464, s. 67–88. Springer. 2019.

[90] ATTEMA, T., LYUBASHEVSKY, B., SEILER, G. *Practical product proofs for lattice commitments.* CRYPTO 2020. Lecture Notes in Computer Science, vol. 12171, s. 470–499. Springer. 2020.

[91] ESGIN, M. F., NGYUEN, N. K., SEILER, G. *Practical exact proofs from lattices: New techniques to exploit fully-splitting rings.* ASIACRYPT 2020, s. 259–288. 2020.

[92] LYUBASHEVSKY, V., NGUYEN, N. K., SEILER, G. *Practical lattice-based zero-knowledge proofs for integer relations.* CCS 2020, s 1051–1070. Association for Computing Machinery, New York, NY, USA. 2020.

[93] MOODY, D. *Let's Get Ready to Rumble – The NIST PQC „Competition“.* Prezentace NIST Computer Security Resource Center. [online]. 2018. Dostupné z URL: <`https://csrc.nist.gov/CSRC/media/Presentations/Let-s-Get-Ready-to-Rumble-The-NIST-PQC-Competiti/images-media/PQCrypto-April2018_Moody.pdf`>.

[94] *SPHINCS+*. Open Quantum Safe. [online]. 2021, [cit. 2021-10-28]. Dostupné z URL: <https://openquantumsafe.org/liboqs/algorithms/sig/sphincs>.

[95] BERNSTEIN, D. J., HÜLSING, A., KÖLBL, S., NIEDERHAGEN, R., RIJNEVELD, J., SCHWABE, P. *The SPHINCS+ Signature Framework*. Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security (CCS '19), s. 2129–2146. Association for Computing Machinery, New York, NY, USA. 2019. ISBN 978-1-4503-6747-9.

[96] *Data Pseudonymisation: Advanced Techniques & Use Cases.* European Union Agency for Cybersecurity (ENISA), 2021. ISBN 978-92-9204-465-7.

[97] HUELSING, A., BUTIN, D., GAZDAG, S., RIJNEVELD, J., MOHAISEN, A. *XMSS: eXtended Merkle Signature Scheme.* Internet Research Task Force (IRTF), 2018. ISSN 2070-1721. Dostupné z URL: <https://www.rfc-editor.org/rfc/rfc8391>.

[98] MATIER, J., WATERLAND, P. *Quantum Resistant Ledger (QRL)*. [online]. 2016, [cit. 2022-04-22]. Dostupné z URL: <https://github.com/theQRL/Whitepaper/blob/master/QRL_whitepaper.pdf>.

Seznam symbolů a zkratek

ABA Attribute-based Authentication

ABC Attribute-based Credentials

ABE Attribute-based Encryption

ABS Attribute-based Scheme či Attribute-based Signature, dle kontextu

AES Advanced Encryption Standard

API Application Programming Interface

BCH Bose–Chaudhuri–Hocquenghem kód

CCA Chosen Ciphertext Attack

CPA Chosen Plaintext Attack

CVP Closest Vector Problem

DH Diffie–Hellman Key Exchange

DSA Digital Signature Algorithm

ECDH Elliptic Curve Diffie–Hellman Key Exchange

ECDSA Elliptic Curve Digital Signature Algorithm

Enc Encryption – šifrování

GPV Gentry–Peikert–Vaikuntanathan schéma

GS Group Signature

HIBE Hierarchical Identity-based Encryption

KEM Key Encapsulation Mechanism

LWE Learning With Errors

MPKC Multivariate Public Key Cryptosystem

NIST US National Institute of Standards and Technology

NTRU N-th degree Truncated polynomial Ring Units

OTS One Time Signature

PET	Privacy Enhancing Technology
PoC	Proof of Concept
PQ	Post Quantum
RLWE	Ring Learning With Errors
ROM	Random Oracle Model
RSA	Rivest–Shamir–Adleman kryptosystém
SHA	Secure Hash Algorithm
Sig	Signature – podpis
SIS	Short Integer Solution
SP	Skupinový podpis
SVP	Shortest Vector Problem
TABAA	Traceable Attribute-based Anonymous Authentication

Seznam příloh

A Vyhledávací dotazy a dílčí výsledky

Tato příloha obsahuje výsledky a výpočty, ze kterých byly sestavovány grafy a podle kterých byl psán text pro kapitolu č. 3 a dílčí podkapitoly 3.1 (v A.1), 3.2 (v A.2) a 3.3 (v A.3).

A.1 Atributová schémata a skupinové podpisy

topic	search query	2015	t15	2016	t16	2017	t17	2018	t18	2019	t19	2020	t20	2021	t21	qt	topic total	total
attribute-based schemes	allintitle: encryption OR signature OR authentication OR credential OR credentials OR signcryption "attribute based" -scheme -schemes	172	226	192	262	201	261	159	251	174	247	151	228	114	181	1163	1656	1978
	allintitle: "attribute based" scheme OR schemes	54		70		60		92		73		77		67		493		
group signatures	allintitle: "group signature" OR "group signatures"	47	47	42	42	36	36	51	51	55	55	47	47	44	44	322	322	
ABE	allintitle: encryption OR signcryption "attribute based" -scheme -schemes	148	178	165	209	183	223	144	205	158	205	132	176	104	149	1034	1345	1572
	allintitle: "attribute based" encryption scheme OR schemes	30		44		40		61		47		44		45		311		
ABS	allintitle: signature OR signcryption "attribute based" -scheme -schemes	14	23	16	17	16	19	13	18	11	14	10	13	7	10	87	114	
	allintitle: "attribute based" signature scheme OR schemes	9		1		3		5		3		3		3		27		
ABA	allintitle: authentication "attribute based" -scheme -schemes	7	11	11	13	8	9	7	9	7	9	10	13	4	5	54	69	
	allintitle: "attribute based" authentication scheme OR schemes	4		2		1		2		2		3		1		15		
AB credentials	allintitle: credential OR credentials "attribute based" -scheme -schemes	8	8	7	7	7	9	5	5	5	8	4	4	3	3	39	44	
	allintitle: "attribute based" credential OR credentials scheme OR schemes	0		0		2		0		3		0		0		5		

Obr. A.1: Tabulka počtů publikací atributových schémat a skupinových podpisů.

A.2 Postkvantové kryptografické rodiny

topic	search query	2015	t15	2016	t16	2017	t17	2018	t18	2019	t19	2020	t20	2021	t21	qt	topic total	total
lattice-based cryptography	allintitle: encryption OR signature OR signcryption "lattice based" -scheme -schemes	9		12		17		15		41		20		28		142	409	708
	allintitle: "lattice based" scheme OR schemes	15	34	20	46	21	48	35	68	22	75	35	67	28	71	176		
	allintitle: lattices based encryption OR signature OR signcryption	10		14		10		18		12		12		15		91		
hash-based cryptography	allintitle: encryption OR signature OR signcryption "hash based" -scheme -schemes	2	15	4	13	7	15	9	15	2	10	8	17	6	9	38	94	
	allintitle: "hash based" scheme OR schemes	13		9		8		6		8		9		3		56		
code-based cryptography	allintitle: encryption OR signature OR signcryption "code based"	6	10	10	13	22	31	13	24	23	31	15	22	20	24	109	155	
	allintitle: codes based encryption OR signature OR signcryption	4		3		9		11		8		7		4		46		
multivariate-based cryptography	allintitle: multivariate based encryption OR signature OR signcryption	5	5	2	2	5	5	2	2	7	7	6	6	8	8	35	35	
isogeny-based cryptography	allintitle: encryption OR signature OR signcryption "isogeny based" -scheme -schemes	0		0		0		0		1		1		1		3	15	
	allintitle: "isogeny based" scheme OR schemes	0	0	2	3	1	2	2	2	1	2	2	4	0	2	8		
	allintitle: isogenies based encryption OR signature OR signcryption	0		1		1		0		0		1		1		4		

Obr. A.2: Tabulka počtů publikací PQ kryptografických rodin.

NIST Round 3 finalists and candidates total type percentage			
Type	**Name**	**Count**	**Percentage**
Code-based	Classic McEliece	3	21%
	BIKE		
	HQC		
Lattice-based	Crystals-Kyber	7	50%
	NTRU		
	Saber		
	Frodo-KEM		
	NTRU-Prime		
	Crystals-Dillithium		
	Falcon		
Isogeny-based	SIKE	1	7%
Multivariate-based	Rainbow	2	14%
	GeMSS		
Hash-based	SPHINCS+	1	7%
Total		14	100%

NIST Round 3 finalists and candidates encryption and signature type percentage				
Function	**Type**	**Name**	**Count**	**Percentage**
Encryption	**Code-based**	Classic McEliece	3	33%
		BIKE		
		HQC		
	Lattice-based	Crystals-Kyber	5	56%
		NTRU		
		Saber		
		Frodo-KEM		
		NTRU-Prime		
	Isogeny-based	SIKE	1	11%
Total Encryption			9	100%
Function	**Type**	**Name**	**Count**	**Percentage**
Signature	**Lattice-based**	Crystals-Dillithium	2	40%
		Falcon		
	Multivariate-based	Rainbow	2	40%
		GeMSS		
	Hash-based	SPHINCS+	1	20%
	Symmetric crypto	Picnic	1	vynecháno ze statistiky
Total Signature			5	100%

Obr. A.3: Procentuální složení PQ rodin kandidátů 3. kola na standardizaci NIST.

A.3 Postkvantová atributová schémata a skupinové podpisy

	topic	search query	2015	t15	2016	t16	2017	t17	2018	t18	2019	t19	2020	t20	2021	t21	qt	topic total	total
ABS	lattice-based	allintitle: encryption OR signature OR authentication OR credential OR credentials OR signcryption "attribute based" - scheme -schemes lattice OR lattices	2	5	1	2	1	1	2	8	2	3	5	8	6	8	19	35	43
		allintitle: "attribute based" scheme OR schemes lattice OR lattices	3		1		0		6		1		3		2		16		
	code-based	allintitle: encryption OR signature OR authentication OR credential OR credentials OR signcryption "attribute based" - scheme -schemes code OR codes	0	0	0	0	0	2	0	0	0	0	0	1	0	0	0	3	
		allintitle: "attribute based" scheme OR schemes code OR codes	0		0		2		0		0		1		0		3		
	hash-based	allintitle: encryption OR signature OR authentication OR credential OR credentials OR signcryption "attribute based" - scheme -schemes hash OR hashes	1	1	0	0	1	1	2	2	0	0	1	1	0	0	5	5	
		allintitle: "attribute based" scheme OR schemes hash OR hashes	0		0		0		0		0		0		0		0		
	multivariate-based	allintitle: encryption OR signature OR authentication OR credential OR credentials OR signcryption "attribute based" - scheme -schemes multivariate	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		allintitle: "attribute based" scheme OR schemes multivariate	0		0		0		0		0		0		0		0		
	isogeny-based	allintitle: encryption OR signature OR authentication OR credential OR credentials OR signcryption "attribute based" - scheme -schemes isogeny OR isogenies	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		allintitle: "attribute based" scheme OR schemes isogeny OR isogenies	0		0		0		0		0		0		0		0		
GS	lattice-based	allintitle: lattices OR lattice based "group signature" OR "group signatures"	3	3	3	3	4	4	7	7	7	7	9	9	7	7	40	40	49
	code-based	allintitle: code OR codes based "group signature" OR "group signatures"	0	0	1	1	2	2	0	0	1	1	1	1	0	0	5	5	
	hash-based	allintitle: hash OR hashes based "group signature" OR "group signatures"	0	0	0	0	0	0	1	1	0	0	0	0	1	1	2	2	
	multivariate-based	allintitle: multivariate based "group signature" OR "group signatures"	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	
	isogeny-based	allintitle: isogeny OR isogenies based "group signature" OR "group signatures"	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	

Obr. A.4: Tabulka počtů publikací PQ atributových schémat a skupinových podpisů.

topic		search query	2015	2016	2017	2018	2019	2020	2021	qt	total
ABE	lattice-based	allintitle: lattices OR lattice based encryption OR signcryption attribute OR attributes	2	2	1	7	2	7	8	29	36
	code-based	allintitle: code OR codes based encryption OR signcryption attribute OR attributes	0	0	2	0	0	0	0	2	
	multivariate-based	allintitle: multivariate based encryption OR signcryption attribute OR attributes	0	0	0	0	0	0	0	0	
	hash-based	allintitle: hash OR hashes based encryption OR signcryption attribute OR attributes	1	0	1	2	0	1	0	5	
	isogeny-based	allintitle: isogeny OR isogenies based encryption OR signcryption attribute OR attributes	0	0	0	0	0	0	0	0	
ABS	lattice-based	allintitle: lattices OR lattice based signature OR signcryption attribute OR attributes	3	0	0	1	2	1	1	8	9
	code-based	allintitle: code OR codes based signature OR signcryption attribute OR attributes	0	0	1	0	0	0	0	1	
	multivariate-based	allintitle: multivariate based signature OR signcryption attribute OR attributes	0	0	0	0	0	0	0	0	
	hash-based	allintitle: hash OR hashes based signature OR signcryption attribute OR attributes	0	0	0	0	0	0	0	0	
	isogeny-based	allintitle: isogeny OR isogenies based signature OR signcryption attribute OR attributes	0	0	0	0	0	0	0	0	
ABA	lattice-based	allintitle: lattices OR lattice based authentication attribute OR attributes	0	0	0	0	0	0	0	0	2
	code-based	allintitle: code OR codes based authentication attribute OR attributes	0	0	1	0	0	1	0	2	
	multivariate-based	allintitle: multivariate based authentication attribute OR attributes	0	0	0	0	0	0	0	0	
	hash-based	allintitle: hash OR hashes based authentication attribute OR attributes	0	0	0	0	0	0	0	0	
	isogeny-based	allintitle: isogeny OR isogenies based authentication attribute OR attributes	0	0	0	0	0	0	0	0	
ABC	lattice-based	allintitle: lattices OR lattice based credential OR credentials attribute OR attributes	0	0	0	0	0	0	0	0	0
	code-based	allintitle: code OR codes based credential OR credentials attribute OR attributes	0	0	0	0	0	0	0	0	
	multivariate-based	allintitle: multivariate based credential OR credentials attribute OR attributes	0	0	0	0	0	0	0	0	
	hash-based	allintitle: hash OR hashes based credential OR credentials attribute OR attributes	0	0	0	0	0	0	0	0	
	isogeny-based	allintitle: isogeny OR isogenies based credential OR credentials attribute OR attributes	0	0	0	0	0	0	0	0	

Obr. A.5: Tabulka počtů publikací typů PQ atributových schémat.

B Porovnání skupinových podpisů

V této příloze jsou v tab. B.1 porovnány skupinové podpisy implementované v knihovně `pygroupsig`, verzi `libgroupsig` pro Python[1].

Schéma	G_{PK} [B]	G_{SK} [B]	Podpis [B]	Zpráva [B]	Setup [ms]	Join (100x) [ms]	Sign [ms]	Verify [ms]	Open [ms]	Total [ms]	Pokusy
GL19 [84]	824	484	932	11	1,8066168	322,1351504	1,8789530	2,7343392	nepodporuje	328,5550594	20
GL19 [84]	824	484	932	5242880	1,8586278	327,7755737	14,6680951	13,6742711	nepodporuje	357,9765677	
BB04 [81]	2868	896	500	11	2,9502988	85,9024167	1,5287876	2,0765185	0,1669526	92,6249743	
BB04 [81]	2868	896	500	5242880	3,3518672	93,9691544	12,1971011	12,4437094	0,1791000	122,1409321	
KLAP20 [83]	740	260	308	11	2,3969889	795,1359272	0,3184080	2,1061659	19,6067810	819,5642710	
KLAP20 [83]	740	260	308	5242880	2,2546649	785,3845596	5,2985191	6,8240881	20,5405951	820,3024268	
PS16 [85]	472	196	236	11	1,0362029	241,0041690	1,0192633	2,2197247	64,0310764	309,3104362	
PS16 [85]	472	196	236	5242880	0,9172916	241,6522145	10,6058002	12,0367646	64,9032474	330,1153183	

Obr. B.1: Porovnání skupinových podpisů z knihovny `pygroupsig`.

[1]Údaje byly měřeny na virtuálním stroji s OS Linux Ubuntu 21.10. Stroji byly přiděleny dva procesory po jednom jádru @ 3,8 GHz a 8 GB RAM.

C Porovnání postkvantových podpisů

Tato příloha obsahuje dodatečná data ke kapitole č. 5.1. Jedná se o porovnání času vykonávání jednotlivých funkcí – generování klíčů, podepisování a verifikace, a paměťových nároků postkvantových podpisů, které jsou finalisty na standardizaci NIST třetího kola. Podpisy jsou implementovány pomocí knihovny `pqcrypto`. Tabulka na obr. C.1 obsahuje jen schémata s 5. úrovní bezpečnosti NIST a zelené hodnoty značí nejmenší hodnoty ve sloupci, červené nejvyšší hodnoty ve sloupci. Tabulku s vícero implementacemi a delší podepisovanou zprávou lze nalézt na obr. C.2[1].

Message	"Hello world"										
PQ Family	Signature	NIST security level	Pk [B]	Sk [B]	Sig [B]	Time [s]					Memory [B]
						Tries	KeyGen	Signature	Verification	Total Time	
Lattice	Dilithium5	5	2595	4864	4595	100	0.000186014	0.000554419	0.000176668	0.000917101	5491880
	Falcon 1024	5	1793	2305	1330	100	0.058311956	0.012664380	0.000142622	0.071118958	5488277
Multivariate	Rainbow V Classic	5	1930600	1408736	212	10	4.531555557	0.031163311	0.032991457	4.595710325	8477260
	Rainbow V Cyclic	5	536136	1408736	212	10	5.425222158	0.032203031	0.084334254	5.541759443	7263652
	Rainbow V Cyclic Compressed	5	536136	64	212	10	5.114265966	2.465212512	0.079362941	7.658841419	6036554
Hash	SPHINCS Haraka 256f robust	5	64	128	49856	20	0.021504641	0.560713482	0.013334131	0.595552254	5532360
	SPHINCS Haraka 256f simple	5	64	128	49856	20	0.011308658	0.310520339	0.006735945	0.328564942	5532360
	SPHINCS Haraka 256s robust	5	64	128	29792	20	0.322689939	4.605864048	0.006606472	4.935160458	5512936
	SPHINCS Haraka 256s simple	5	64	128	29792	20	0.175950086	2.645859945	0.003522515	2.825332546	5512936
	SPHINCS SHA256 256f robust	5	64	128	49856	20	0.020466328	0.473835015	0.012298393	0.506599736	5532360
	SPHINCS SHA256 256f simple	5	64	128	49856	20	0.007041299	0.168175447	0.003781831	0.178998578	5532360
	SPHINCS SHA256 256s robust	5	64	128	29792	20	0.316293001	3.842674172	0.005820644	4.164787817	5512934
	SPHINCS SHA256 256s simple	5	64	128	29792	20	0.109636545	1.462456667	0.001956403	1.574049616	5512934
	SPHINCS SHAKE256 256f robust	5	64	128	49856	20	0.021582103	0.466140366	0.011355543	0.499078012	5532374
	SPHINCS SHAKE256 256f simple	5	64	128	49856	20	0.011376584	0.258361351	0.006083977	0.275821912	5532374
	SPHINCS SHAKE256 256s robust	5	64	128	29792	20	0.353453255	3.956046021	0.005682123	4.315181398	5512948
	SPHINCS SHAKE256 256s simple	5	64	128	29792	20	0.184832633	2.217500746	0.00297395	2.405307329	5512948

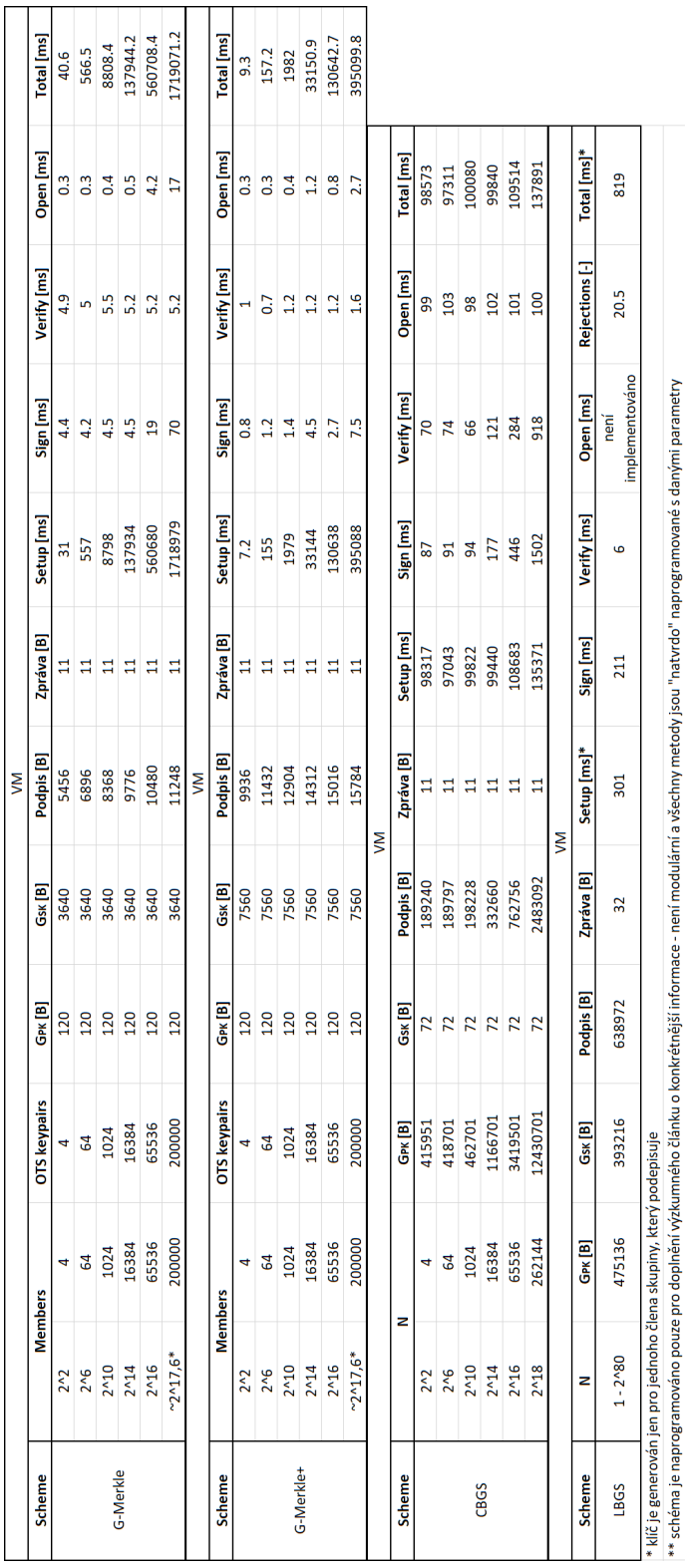
Obr. C.1: Porovnání postkvantových podpisů úrovně 5.

[1] Údaje byly měřeny na virtuálním stroji s OS Kali Linux v2021.2. Stroji byly přiděleny dva procesory po dvou jádrech @ 3 GHz a 2 GB RAM. Důvodem nesouhlasných parametrů v porovnání s předchozím měřením je výměna hardwaru domácí stanice.

Message	"Hello world"										
PQ Family	Signature	NIST Security	Pk [B]	Sk [B]	Sig [B]	Time [s]					Memory [B]
						Tries	KeyGen	Signature	Verification	Total Time	
Lattice	Dilithium5	5	2595	4864	4595	100	0.00018601	0.00055442	0.00017667	0.0009171	5491880
	Dilithium3	3	1952	4000	3293	100	0.0001571	0.0007186	0.00015024	0.00102595	5490575
	Dilithium2	2	1312	2528	2420	100	0.000099342	0.00049888	0.00010211	0.00070033	5488926
	Falcon 1024	5	1793	2305	1330	100	0.05831196	0.012664380	0.00014262	0.07111896	5488277
	Falcon 512	1	897	1281	690	100	0.021054566	0.00573973	0.000075903	0.0268702	5485727
Multivariate	Rainbow I Classic	1	161600	103648	66	100	0.1477602	0.00280623	0.00277983	0.15334626	5786400
	Rainbow I Cyclic	1	60192	103648	66	100	0.15009468	0.002426140	0.00645141	0.15897223	5695442
	Rainbow I Cyclic Compressed	1	60192	64	66	100	0.14723119	0.065483	0.00637694	0.21909113	5602578
	Rainbow III Classic	3	882080	626048	164	20	1.48004369	0.01405118	0.0148795	1.50897436	6766542
	Rainbow III Cyclic	3	264608	626048	164	20	1.66777765	0.01408297	0.03599051	1.71785113	6262739
	Rainbow III Cyclic Compressed	3	264608	64	164	20	1.68857524	0.81174647	0.03692249	2.53724984	5751402
	Rainbow V Classic	5	1930600	1408736	212	10	4.53155556	0.03116331	0.03299146	4.59571033	8477260
	Rainbow V Cyclic	5	536136	1408736	212	10	5.42522216	0.03220303	0.08433425	5.54175944	7263652
	Rainbow V Cyclic Compressed	5	536136	64	212	10	5.11426597	2.46521251	0.07936294	7.65884142	6036554
Hash	SPHINCS Haraka 256f robust	5	64	128	49856	20	0.02150464	0.56071348	0.01333413	0.59555225	5532360
	SPHINCS Haraka 256f simple	5	64	128	49856	20	0.01130866	0.31052034	0.00673594	0.32856494	5532360
	SPHINCS Haraka 256s robust	5	64	128	29792	20	0.32268994	4.60586405	0.00660647	4.93516046	5512936
	SPHINCS Haraka 256s simple	5	64	128	29792	20	0.17595009	2.64585994	0.00352252	2.82533255	5512936
	SPHINCS SHA256 256f robust	5	64	128	49856	20	0.02046633	0.47383502	0.01229839	0.50659974	5532360
	SPHINCS SHA256 256f simple	5	64	128	49856	20	0.0070413	0.16817545	0.00378183	0.17899858	5532360
	SPHINCS SHA256 256s robust	5	64	128	29792	20	0.316293	3.84267417	0.00582064	4.16478782	5512934
	SPHINCS SHA256 256s simple	5	64	128	29792	20	0.10963655	1.46245667	0.0019564	1.57404962	5512934
	SPHINCS SHAKE256 256f robust	5	64	128	49856	20	0.0215821	0.46614037	0.01135554	0.49907801	5532374
	SPHINCS SHAKE256 256f simple	5	64	128	49856	20	0.01137658	0.25836135	0.00608398	0.27582191	5532374
	SPHINCS SHAKE256 256s robust	5	64	128	29792	20	0.35345325	3.95604602	0.00568212	4.3151814	5512948
	SPHINCS SHAKE256 256s simple	5	64	128	29792	20	0.18483263	2.21750075	0.00297395	2.40530733	5512948
Message	string 1024 B										
PQ Family	Signature	NIST security	Pk [B]	Sk [B]	Sig [B]	Time [s]					Memory [B]
						Tries	KeyGen	Signature	Verification	Total Time	
Lattice	Dilithium4	5	2595	4864	4595	100	0.00018372	0.00062821	0.00019537	0.0010073	5492867
	Dilithium3	3	1952	4000	3293	100	0.00015713	0.00076533	0.00015844	0.0010809	5491562
	Dilithium2	2	1312	2528	2420	100	0.000096903	0.00047515	0.0001101	0.00068216	5490013
	Falcon 1024	5	1793	2305	1330	100	0.054918060	0.01163184	0.000133290	0.066683190	5489359
	Falcon 512	1	897	1281	690	100	0.02040223	0.00552358	0.000077200	0.02600301	5486818
Multivariate	Rainbow Ia Classic	1	161600	103648	66	100	0.14005791	0.00252876	0.0024021	0.14498877	5787489
	Rainbow Ia Cyclic	1	60192	103648	66	100	0.157598360	0.00250409	0.00694078	0.16704323	5696633
	Rainbow Ia Cyclic Compressed	1	60192	64	66	100	0.16466669	0.07225659	0.0071283	0.24405158	5603665
	Rainbow IIIc Classic	3	882080	626048	164	20	1.50648712	0.01522551	0.01606838	1.537781000	6767733
	Rainbow IIIc Cyclic	3	264608	626048	164	20	1.68322647	0.01433896	0.03654075	1.73410617	6263830
	Rainbow IIIc Cyclic Compressed	3	264608	64	164	20	1.67900032	0.81021131	0.03556955	2.52478118	5752389
	Rainbow Vc Classic	5	1930600	1408736	212	10	4.45319884	0.03084648	0.0326205	4.51666582	8478349
	Rainbow Vc Cyclic	5	536136	1408736	212	10	5.1810426	0.03148723	0.08199255	5.29452238	7264639
	Rainbow Vc Cyclic Compressed	5	536136	64	212	10	5.0594075	2.4243784	0.07953942	7.56332531	6037745
Hash	SPHINCS Haraka 256f robust	5	64	128	49856	20	0.02105007	0.54753728	0.0128093	0.58139665	5533447
	SPHINCS Haraka 256f simple	5	64	128	49856	20	0.01099137	0.29111661	0.00666542	0.3087734	5533447
	SPHINCS Haraka 256s robust	5	64	128	29792	20	0.34120518	4.92582442	0.00721436	5.27424395	5513923
	SPHINCS Haraka 256s simple	5	64	128	29792	20	0.19082583	2.70343124	0.00351623	2.8977733	5514023
	SPHINCS SHA256 256f robust	5	64	128	49856	20	0.01946906	0.44643676	0.01126393	0.47716975	5533347
	SPHINCS SHA256 256f simple	5	64	128	49856	20	0.00700626	0.17035613	0.00387132	0.18123372	5533347
	SPHINCS SHA256 256s robust	5	64	128	29792	20	0.33067036	4.03710527	0.00582818	4.37360381	5514025
	SPHINCS SHA256 256s simple	5	64	128	29792	20	0.11925895	1.50270631	0.00202314	1.6239884	5514025
	SPHINCS SHAKE256 256f robust	5	64	128	49856	20	0.02123739	0.46915811	0.01145612	0.50185162	5533463
	SPHINCS SHAKE256 256f simple	5	64	128	49856	20	0.01233444	0.29670513	0.00666847	0.31570804	5533463
	SPHINCS SHAKE256 256s robust	5	64	128	29792	20	0.39095483	4.30998297	0.00609437	4.70703217	5514037
	SPHINCS SHAKE256 256s simple	5	64	128	29792	20	0.18623953	2.3396629	0.00317613	2.52907856	5514037

Obr. C.2: Porovnání postkvantových podpisů s různou délkou zprávy.

D Parametry a efektivita implementací SP



VM

Scheme	Members	OTS keypairs	G_{PK} [B]	G_{SK} [B]	Podpis [B]	Zpráva [B]	Setup [ms]	Sign [ms]	Verify [ms]	Open [ms]	Total [ms]
G-Merkle	2^2	4	120	3640	5456	11	31	4.4	4.9	0.3	40.6
	2^6	64	120	3640	6896	11	557	4.2	5	0.3	566.5
	2^10	1024	120	3640	8368	11	8798	4.5	5.5	0.4	8808.4
	2^14	16384	120	3640	9776	11	137934	4.5	5.2	0.5	137944.2
	2^16	65536	120	3640	10480	11	560680	19	5.2	4.2	560708.4
	~2^17,6*	200000	120	3640	11248	11	1718979	70	5.2	17	1719071.2

VM

Scheme	Members	OTS keypairs	G_{PK} [B]	G_{SK} [B]	Podpis [B]	Zpráva [B]	Setup [ms]	Sign [ms]	Verify [ms]	Open [ms]	Total [ms]
G-Merkle+	2^2	4	120	7560	9936	11	7.2	0.8	1	0.3	9.3
	2^6	64	120	7560	11432	11	155	1.2	0.7	0.3	157.2
	2^10	1024	120	7560	12904	11	1979	1.4	1.2	0.4	1982
	2^14	16384	120	7560	14312	11	33144	4.5	1.2	1.2	33150.9
	2^16	65536	120	7560	15016	11	130638	2.7	1.2	0.8	130642.7
	~2^17,6*	200000	120	7560	15784	11	395088	7.5	1.6	2.7	395099.8

VM

Scheme	N	G_{PK} [B]	G_{SK} [B]	Podpis [B]	Zpráva [B]	Setup [ms]	Sign [ms]	Verify [ms]	Open [ms]	Total [ms]
CBGS	2^2	4	415951	72	189240	11	98317	87	70	99	98573
	2^6	64	418701	72	189797	11	97043	91	74	103	97311
	2^10	1024	462701	72	198228	11	99822	94	66	98	100080
	2^14	16384	1166701	72	332660	11	99440	177	121	102	99840
	2^16	65536	3419501	72	762756	11	108683	446	284	101	109514
	2^18	262144	12430701	72	2483092	11	135371	1502	918	100	137891

VM

Scheme	N	G_{PK} [B]	G_{SK} [B]	Podpis [B]	Zpráva [B]	Setup [ms]*	Sign [ms]	Verify [ms]	Open [ms]	Rejections [-]	Total [ms]*
LBGS	1 - 2^80	475136	393216	638972	32	301	211	6	není implementováno	20.5	819

* klíč je generován jen pro jednoho člena skupiny, který podepisuje

** schéma je naprogramováno pouze pro doplnění výzkumného článku o konkrétnější informace - není modulární a všechny metody jsou "natvrdo" naprogramované s danými parametry

Obr. D.1: Porovnání postkvantových skupinových podpisů.

VM

Scheme	N	G_{PK} [B]	G_{SK} [B]	Podpis [B]	Zpráva [B]	Setup + Join [ms]	Sign [ms]	Verify [ms]	Open [ms]	Total [ms]
BB04	2^2	4	2920	952	552	11	6.3	1.7	2.4	0.1	10.5
	2^6	64	2920	952	552	11	61.4	1.6	2.2	0.2	65.4
	2^10	1024	2920	952	552	11	838.1	1.4	1.8	0.1	841.4
	2^14	16384	2920	952	552	11	14177.1	1.6	2.2	0.2	14181.0
	2^16	65536	2920	952	552	11	55698.9	1.4	1.9	0.1	55702.2
	2^18	262144	2920	952	552	11	232098.7	1.6	2.2	0.2	232102.7

VM

Scheme	N	G_{PK} [B]	G_{SK} [B]	Podpis [B]	Zpráva [B]	Setup + Join [ms]	Sign [ms]	Verify [ms]	Open [ms]	Total [ms]
GL19	2^2	4	880	536	984	11	15.4	2.4	4.0	0.0	21.9
	2^6	64	880	536	984	11	225.6	1.8	2.7	0.0	230.0
	2^10	1024	880	536	984	11	3260.7	1.8	2.6	0.0	3265.0
	2^14	16384	880	536	984	11	50231.0	2.0	2.8	0.0	50235.8
	2^16	65536	880	536	984	11	202276.7	2.0	2.8	0.0	202281.6
	2^18	262144	880	536	984	11	813462.4	2.0	2.8	0.0	813467.2

VM

Scheme	N	G_{PK} [B]	G_{SK} [B]	Podpis [B]	Zpráva [B]	Setup + Join [ms]	Sign [ms]	Verify [ms]	Open [ms]	Total [ms]
KLAP20	2^2	4	792	312	360	11	35.5	0.3	2.1	4.1	42.0
	2^6	64	792	312	360	11	548.8	0.3	2.6	4.0	555.8
	2^10	1024	792	312	360	11	8330.4	0.3	2.2	3.8	8336.8
	2^14	16384	792	312	360	11	126370.4	0.3	1.8	3.2	126375.7
	2^16	65536	792	312	360	11	513961.7	0.4	2.2	3.9	513968.1
	2^18	262144	792	312	360	11	2084934.9	0.4	2.3	4.0	2084941.5

VM

Scheme	N	G_{PK} [B]	G_{SK} [B]	Podpis [B]	Zpráva [B]	Setup + Join [ms]	Sign [ms]	Verify [ms]	Open [ms]	Total [ms]
PS16	2^2	4	528	248	288	11	16.7	1.7	2.8	3.3	24.5
	2^6	64	528	248	288	11	163.3	1.2	2.5	3.3	170.3
	2^10	1024	528	248	288	11	2653.2	1.1	2.4	3.1	2659.9
	2^14	16384	528	248	288	11	40111.1	1.1	2.4	3.1	40117.7
	2^16	65536	528	248	288	11	156282.0	1.2	2.5	3.0	156288.6
	2^18	262144	528	248	288	11	641875.7	1.3	2.7	3.1	641882.8

Obr. D.2: Porovnání klasických skupinových podpisů.

E Obsah elektronické přílohy

Složky `codebased`, `hashbased` a `latticebased` obsahují odpovídající schémata postkvantových skupinových podpisů. Podrobnější informace jsou popsány v souborech `README.md` uvnitř jednotlivých složek.

Součástí elektronické přílohy jsou i podpůrné implementace klasických skupinových podpisů v rámci složky `classic`. Daný skupinový podpis je vždy implementován ve dvou verzích, krátké a dlouhé. Krátká verze podepisuje zprávu „Hello world“ a dlouhá zprávu o velikosti 5 MB. Reálné velikosti parametrů byly měřeny pomocí Python knihovny `pympler`.

Elektronická příloha práce je dostupná také online[1].

- `/` ..... kořenový adresář elektronické přílohy
  - `classic` ..... implementace klasických skupinových podpisů (`libgroupsig`)
  - `codebased` ..... implementace CBGS
    - `README.md` ..... informace a návod ke zprovoznění implementace
    - `code_GS.cpp` ..... hlavní soubor implementace
  - `hashbased`
    - `gmerkle` ..... implementace *G-Merkle*
      - `g-merkle` ..... soubory .py pro *G-Merkle*
      - `README.md` ..... informace a návod ke zprovoznění implementace
    - `gmerkle+` ..... implementace *G-Merkle+*
      - `g-merkle` ..... soubory .py pro *G-Merkle+*
      - `README.md` ..... informace a návod ke zprovoznění implementace
  - `latticebased` ..... implementace LBGS
    - `README.md` ..... informace a návod ke zprovoznění implementace
    - `test_sign.c` ..... hlavní soubor implementace
  - `README.md`
  - `DP_Hluckova-Pavla_xhluck01.pdf` ..... text diplomové práce

[1] `https://gitfront.io/r/xhluck01/zwQCnG8cMjDa/dp-hluckova/`