**Рівномірна вибірка Гауса та його застосування для схеми підпису Falcon**

**Анотація:**

Вибірка Гауса для цілих чисел є ключовим інструментом у криптографії на основі решітки, але виявилось напрочуд складним завданням виконувати вибірку у типовий, ефективний спосіб з доведеною захищеністю. У цій роботі представлено блоковий фреймворк для генерації дискретних компонентів Гауса з довільним центром та стандартним відхиленням. Даний фреймворк надзвичайно простий, і саме ця простота дозволила зробити його простим у впровадженні, доказово захищеним, портативним, ефективним та доказово стійким проти часових атак. Відбірник, що буде розглядатися, є гарним кандидатом для будь-якого вибіркового аналізу, нещодавно його було впроваджено у схему ЕП Falcon. Другий внесок спрямований на систематизацію виявлення помилок реалізації у відбірниках Гауса. Ми пропонуємо набір статистичних випробувань для дискретних компонентів Гауса, який називається SAGA (Statistically Acceptable GAussian).

**Ключові слова:**

Криптографія, заснована на алгебраїчних решітках, Вибірка Гауса, рівномірна вибірка.

**1. Вступ**

Вибірка Гауса над цілими числами є центральним елементом криптографії на основі решіток, як в теорії, так і на практиці. Також відомо, що вибірка є процесом, який складно виконувати якісно та безпечно. Це доведено численними атаками бічними каналами на пробовідбірник Гауса, що застосовується у BLISS. З цієї причини деякі схеми обмежують або забороняють використання таких відбірників. Однак у деяких ситуаціях використання відбірника Гауса не уникнути. Найбільш яскравим прикладом є відбір проб: його проведення з іншими розподілами є поки не вирішеним питанням, за винятком одиничних випадків (вибірка з широкого класу розподілів), які спричиняють зростання складності до на виході, що, в свою чергу, призводить до зменшення рівня безпеки. Беручи до уваги незліченну кількість застосувань вибіркових процесів (хеш-сигнатури повного домену, шифрування на основі ідентичності (або IBE) , ієрархічне IBE, тощо), важливо прийти до відбірників Гауса над цілими числами, які були б не тільки ефективними, але й доказово захищеними, стійкими до часових атак та загалом простими в реалізації.

**2 Попередні дані**

**2.1 Функція Гауса**

Для з ми називаємо функцією Гауса параметрів і позначаємо через функцію, визначену над як . Зверніть увагу, що коли , це упускається в позначеннях індексу, наприклад . Параметр (відповідно, ) часто називають стандартним відхиленням (відповідно, центром) Гауса. Крім того, для будь-якої злічувальної множини ми грубо позначаємо через суму . Коли скінченна, позначаємо через і називаємо гауссовим розподілом параметрів розподіл над , визначений як Тут також, коли , ми опускаємо це в позначеннях індексу. Для позначення розподілу Бернуллі параметра використовується позначення .

**2.2 Відхилення Рені**

У цьому пункті згадується визначення відхилення Рені, яке буде активно використовуватися у доказах безпеки.

**Визначення 1 (відхилення Рені).** Нехай - два розподіли такі, що . Для визначимо відхилення Рені порядку як

До того ж, визначаємо відхилення Рені порядку як

Відхилення Рені перевіряє криптографічно корисні властивості, які буде перераховано нижче.

**Лема 2.** Для двох розподілів та двох сімейств розподілів відхилення Рені перевіряє наступні властивості:

* **Нерівність обробки даних.** Для будь-якої функції , де розподіл (відповідно, ) отримується шляхом застосування до (відповідно, ).
* **Мультиплікативність.** Припустимо, що і є спільним розподілом пари випадкових величин (). Припустимо також, що **незалежні**. Тоді позначаємо і . Тоді .
* **Збереження ймовірності.** Для будь-якої події та ,

**Лема 3.** Нехай - два розподіли однакової опори . Припустимо, що відносна похибка між і обмежена: так, що над . Тоді для :

У деяких випадках запити до розподілів не є незалежними, отже, не можна застосовувати мультиплікативність безпосередньо в наших доказах. Для обробки цих залежностей вводиться наступне твердження.

**Твердження 4.** Нехай і позначають два розподіли N-множини випадкових величин . Для припустимо, що (відповідно ) є граничним розподілом , і нехай позначає умовний розподіл , враховуючи, = . Нехай . Припустимо, що для всіх існує так, що для всіх i-множин у опорі для , обмежені першими змінними,

Тоді:

Доказ. Результат доводиться у випадку , тоді загальний випадок слідує за індукцією. Маємо:

На цьому доказ закінчується.

**2.3 Параметр згладжування**

Для параметр згладжування решітки Λ є найменшим значенням таким, що де позначає дуальну до Λ решітку. У літературі деякі визначення параметра згладжування масштабують наше визначення коефіцієнтом .

**2.4 Рівномірні (ізохронні) алгоритми**

**Визначення 5.** Нехай - це (імовірнісний або детермінований) алгоритм із набором вхідних змінних , набором вихідних змінних і нехай буде набором чутливих змінних. Ми говоримо, що є абсолютно рівномірним(ізохронним) відносно , якщо час його роботи є незалежним від жодної змінної в . Крім того, стверджується, що статистично рівномірний(ізохронний) щодо , якщо існує розподіл , незалежний від усіх змінних у , такий, що час роботи статистично близький (для чітко визначеного відхилення) до .

Зазначимо, що ми можемо визначити поняття обчислювально рівномірного(ізохронного) алгоритму. Для такого алгоритму обчислювально важко відновити чутливі змінні, навіть враховуючи розподіл часу роботи алгоритму.

**3 Відбірник**

У цьому розділі описано новий пробовідбірник із довільним стандартним відхиленням і центром. Основне припущенням в даному випадку - вважати, що всі стандартні відхилення обмежені і що центр знаходиться в [0, 1]. Іншими словами, позначаючи верхню та нижню межі на стандартному відхиленні як , ми представляємо алгоритм, який відбирає розподіл для будь-якого та .

Алгоритм вибірки називається , він описаний в Алгоритмі 1. Позначимо як алгоритм, який відбирає елемент із фіксованим напівгаусовим розподілом . Перший крок полягає у використанні BaseSampler. Потім отриманий зразок перетворюється на , де - біт, рівномірно вибраний у множині {0, 1}. Позначимо через розподіл . Розподіл є дискретним бімодальним напіврозподілом Гауса для центрів 0 і 1. Більш формально,

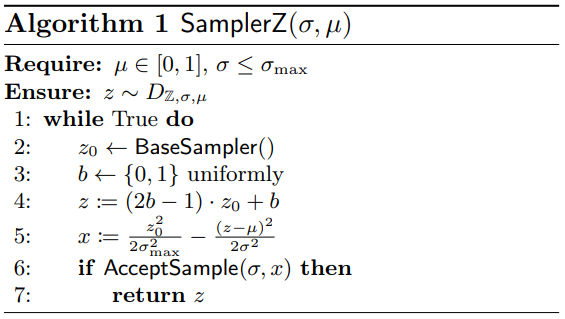


Рис. 3.1 – Алгоритм

Потім, щоб відновити бажаний розподіл для входів (), можна було б застосувати класичну техніку вибірки відкидання, застосовану до схем на основі решітки, і прийняти з імовірністю:

Елемент всередині exp обчислюється на кроці 5. Далі також вводиться алгоритм, позначений як . Він виконує вибірку відкидання (Алгоритм 2): використовуючи алгоритм , який повертає , він повертає вибірку Бернуллі з відповідною ймовірністю. Власне, щодо питань рівномірності(ізохронії), детально описаних у Розділі 6, імовірність останнього прийняття масштабується на коефіцієнт . Оскільки підпорядковується розподілу , після вибірки відкидання кінцевий розподіл () є пропорційним до , який після нормування точно дорівнює . Таким чином, за допомогою цієї конструкції можна вивести наступне твердження.

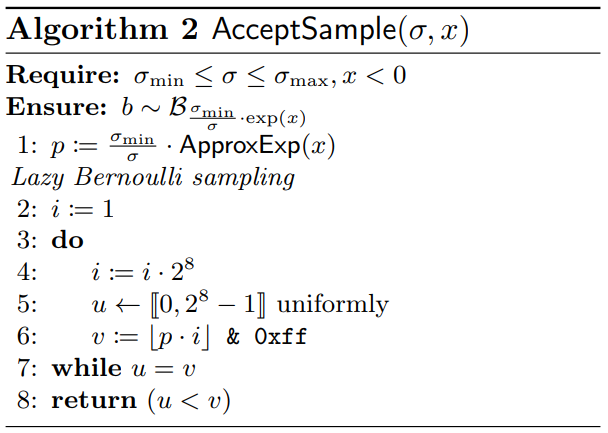


Рис. 3.2 – Алгоритм

**Твердження 6 (Правильність).** Припустимо, що всі рівномірні розподіли є досконалими і що та exp, тоді будова (в алгоритмах 1 та 2) така, що .

Для практичних реалізацій неможливо отримати ідеальні розподіли. Можливо лише отримання та exp. Розділ 6 доводить, що за певних умов на та та за певної кількості запитів вибірки, остаточний розподіл залишається невідмінним від .

**4 Доказ захищеності**

На рисунку 4.1 наведено позначення кількості викликів , а також розглянуті при створенні екземпляру для Falcon значення. Через вибірку відкидання на кроці 6, буде (потенційно нескінченна) кількість ітерацій циклу while. Пізніше в лемі 8 буде показано, що кількість ітерацій відповідає геометричному закону параметра . Позначимо як наближено максимальну кількість ітерацій. За аргументом центральної межі буде лише незначно вищою за очікувану кількість ітерацій. Щоб створити екземпляр значень на прикладі Falcon, беремо. Фактично, для параметрів Falcon.

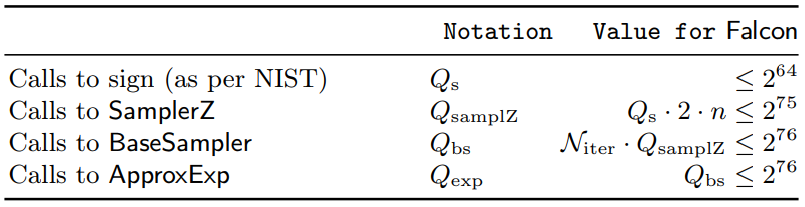


Рис. 4.1 – Кількість викликів

Наступна теорема оцінює безпеку , вона є незалежною від обраних значень кількості викликів.

**Теорема 7 (Безпека SamplerZ).** Розглянемо криптосистему, позначену , яка робить щонайбільше **незалежних** запитів до . Припустимо, що має λ-бітову захищеність від задачі пошуку. Тепер розглянемо копію останньої криптосистеми, позначеної , де єдиною відмінністю є застосування .. Якщо дотримано дві наступні умови:

тоді, буде мати (λ-2)-бітову захищеність від задачі пошуку.

Для отримання конкретних числових значень ми припускаємо, що для вихідної схеми заявлено 256 бітів захищеності, таким чином для реальної реалізації необхідно 254 біта захисту. Тоді для реалізації Falcon числові значення будуть наступними:

**4.1 Створення екземпляру**

Для досягнення умови (1) за допомогою використовується поліноміальне наближення експоненти на . Насправді можна зменшити параметр за модулем таким чином, щоб . Обчислюємо експоненціальні лишки для обчислення . Відзначаючи, що трапляються дуже рідко, тому можна наситити до 63, щоб уникнути переповнення без втрати точності.

Тут використовується інструмент апроксимації поліномів, наданий у GALACTICS. Цей інструмент генерує поліноміальні наближення, які дозволяють проводити обчислення фіксованої точності з обраним розміром коефіцієнтів та ступеня. Як приклад, для 32-розрядних коефіцієнтів і ступеня 10 отримуємо поліном , з:

Для будь-якого , підтверджує , що є достатнім для дотримання умови (1) у реалізації Falcon. Крім того, ми експериментально перевіряємо, чи підтверджує .

**4.2 Створення екземпляру**

Для досягнення умови (2) за допомогою ми спираємось на сукупну таблицю розподілу (CDT). Ми попередньо обчислюємо таблицю сукупної функції розподілу з певною точністю; тоді для отримання вибірки генерується випадкове значення в [0, 1] з такою ж точністю і повертаємо індекс останнього запису в таблиці, який перевищує це значення. У випадковий час вибірку можна зробити досить ефективно за допомогою двійкового пошуку, але реалізація з постійним часом, по суті, не має іншого вибору, як кожен раз читати всю таблицю та проводити кожне порівняння. Цей процес резюмується в алгоритмі 3. Параметри - це відповідно кількість елементів CDT та точність його коефіцієнтів. Нехай . Для визначення параметрів використовується простий скрипт, який враховуючи як вхідні дані:

1. Обчислює найменший зріз при якому відхилення Рені між ідеальним розподілом та його обмеженням до {0,. . . , w} (позначеного як ) підтверджує (;
2. Округлює таблицю щільності ймовірностей (PDT) для з бітами абсолютної точності. Це округлення виконується «розумно» шляхом усічення всіх значень PDT, крім найбільшого:
   * Для значення зрізається:
   * з метою мати розподіл імовірностей, .
3. Виводить і обчислює кінцевий

Взявши, і як вхідні дані, ми отримали .

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Експеримент показує, що для будь-якого , , що задовольняє умові 2 для реалізації Falcon.

**5 Аналіз опору часовим атакам**

У цьому розділі показується, що Алгоритм 1 є стійким проти часових атак. Формально ми доводимо, що він є рівномірним(ізохронним) щодо та вихідного значення (у сенсі визначення 5). Спочатку ми доводимо технічну лему, яка показує, що кількість ітерацій у циклі while Алгоритму 1 (майже) незалежна від

**Лема 8.** Нехай і нехай , є стандартними відхиленнями, такими що . Нехай . Кількість ітерацій циклу while у відповідає геометричному закону параметра

Далі ми показуємо, що Алгоритм 1 є абсолютно рівномірним(ізохронним) відносно та статистично ізохронним (для відхилення Рені) щодо .

Теорема 9. Нехай , нехай , є стандартними відхиленнями, такими що , і нехай константа в (0, 1). Припустимо, що елементарні операції {+, -, ×, /} над цілими числами та числами з плаваючою крапкою є рівномірними(ізохронними). Час роботи Алгоритму 1 слідує за розподілом так, що:

для деякого розподілу , незалежного від його вхідних даних та вихідних - .

Нарешті, ми використовуємо теорему 9, щоб довести, що час роботи не допомагає зловмиснику зламати криптографічну схему. Ми вважаємо, що зловмисник має доступ до деякої функції ), а також до часу роботи : це робиться для того, щоб зафіксувати той факт, що вихідні дані не передаються безпосередньозловмиснику, а обробляється певною функцією раніше. Наприклад, у схемі підпису Falcon зразки обробляються алгоритмами залежно від приватного ключа підписувача. З іншого боку, ми вважаємо, що супротивник має всі необхідні потужності для часових атак, дозволяючи йому дізнатися точний час виконання кожного виклику

**Наслідок 10**. Розглянемо противника , який робить запитів підпису щодо алгоритму підпису, який використовує . Припустимо, що зламує захист, вирішуючи задачу пошуку з імовірністю успіху для деякого . Вивчення часу роботи кожного виклику не збільшує ймовірність успіху більш ніж на постійний коефіцієнт.

**Вплив коефіцієнта масштабування.** Коефіцієнт масштабування Масштабний коефіцієнт ≤ є вирішальним для того, щоб зробити пробовідбірник рівномірним(ізохронним), оскільки він декорелює час роботи від . Однак це також впливає на , оскільки можна легко показати, що пропорційний коефіцієнту масштабування. Тому бажано зробити його якомога меншим. Максимальне значення коефіцієнта масштабування фактично залежить від криптографічної схеми, в якій використовується семплер. Для випадку схеми підпису Falcon . і вплив коефіцієнта масштабування обмежений.

**6 Правильність реалізації**

**6.1 Одновимірні тести**

Ми використовуємо стандартні статистичні інструменти для перевірки роботи гауссового пробовідбірника з правильним середнім значенням, середньоквадратичним відхиленням, перекосом та ексцентричністю, і нарешті перевіряємо, чи проходить він тест на нормальність хі-квадрат. Асиметрія та ексцентричність є дескрипторами нормального розподілу, які відповідно вимірюють симетрію та піковість розподілу. Щоб переглянути повний статистичний аналіз цих тестів, було створено клас , який бере як аргументи ініціалізації очікуване середнє значення (mu), очікуване стандартне відхилення (sigma) та список спостережуваних одновимірних гауссових зразків (data).

**6.2 Багатовимірні тести**

**Багатовимірна нормальність.** Існує ряд статистичних тестів, які оцінюють нормальність багатовимірних розподілів. Ми виявили, що багатовимірні тести на нормальність, які переважно застосовуються в інших областях, страждають від проблем розміру та масштабування. Тобто, великі розміри зразків, які ми збираємося використовувати, і погані енергетичні властивості цих тестів роблять виникнення помилки типу II дуже ймовірними. Були впроваджені тести Мардіа та Хенце-Цирклера і виявлено, що, хоча вони працювали для малих розмірів вибірки, вони розходились, видаючи помилки другого типу для вибірок розміром навіть при малих розмірностях .

Однак тест Доорніка-Хансена мінімізує ці проблеми, використовуючи трансформовані версії перекосу та ексцентричності багатовимірних даних, збільшуючи потужність тесту. Як і при одновимірних тестах, було створено клас , який видає чотири результати; два на основі матриці коваріації, а два на основі нормальності даних.

**Збій у (коваріаційній) матриці.** Другий багатовимірний тест задає наступне питання: як хтось міг правильно реалізувати базовий семплер, але згодом не використовувати його належним чином? Універсальної відповіді на це немає, і, як правило, доводиться покладатися на контекст, досвід та здоровий глузд, щоб встановити найбільш вірогідний спосіб, як це могло статися.

Наприклад, у Falcon одновимірні вибірки лінійно поєднуються відповідно до значень вузлів збалансованого двійкового дерева, обчислених при генерації ключів (дерево Falcon). Якщо в процедурі обчислення дерева (під час генерації ключів) або при комбінуванні зразків (під час підписання) допущено помилку в реалізації, це фактично призводить до неправильних або опущених вузлів дерева Falcon. Такі помилки значним чином впливають на емпіричну коваріаційну матрицю підписів Falcon: вони роблять їх схожими на блокові матриці Тепліца (рис. 1b) замість (масштабованих) аутентичних матриць у номінальному випадку (рис. 1a).

Ми розробили тест, який відрізняє блокові матриці коваріації Тепліца від тих, що очікуються від сферичниого розподілу Гауса. Ключова ідея досить проста: при додаванні коефіцієнтів над (блочною-)піддіагоналлю емпіричної матриці коваріації абсолютне значення суми зростатиме в , якщо емпірична матриця коваріації сходиться до масштабованої аутентичної матриці, і в , якщо це блок Тепліца. Ми використовуємо цю різницю в зростанні для виявлення дефектних Гаусів. Хоча офіційного підтвердження тесту не надано, на практиці він досить добре виявляє Гаусів, викликаних дефектними деревами Falcon.

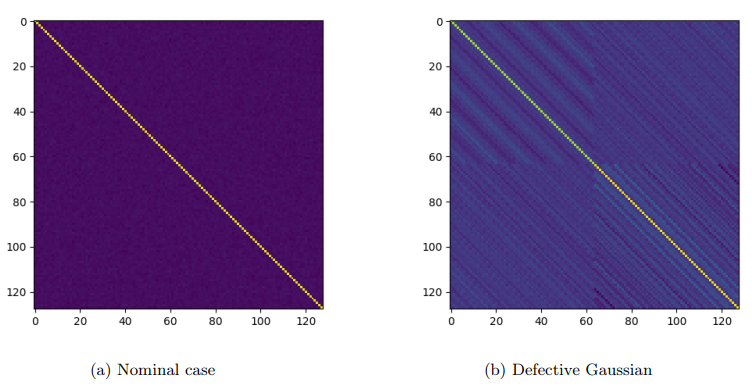


Рис. 6.1 -Емпіричні коваріаційні матриці підписів Falcon. Рисунок 1а відповідає правильній реалізації Falcon. Рисунок 1b відповідає реалізації, де є помилка при побудові дерева Falcon.

**8 Застосування та обмеження**

Тестування цієї реалізації семплера проводиться на одному ядрі процесора Intel Core i7-6500U з тактовою частотою 2,5 ГГц. У таблиці 2 преставлено час роботи нашого рівномірного(ізохронного) пробовідбірника. Зверніть увагу, що для нашого пробовідбірника кількість зразків за секунду в середньому становить, тоді як для COSAC фіксовано.

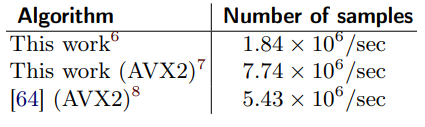


Рис. 6.2 – Кількість зразків за секунду на частоті 2.5ГГц, де як 64 позначено COSAC

У таблиці 3 представлено тривалість роботи рівномірної(ізохронної) реалізації Falcon, яка містить розглянутий відбірник, і порівняння її з другою неізохронною реалізацією, майже ідентичною, за винятком базового відбірника, який є більш швидким лінивим відбірником CDT, і вибіркою відкидання, яка не масштабується константою. Порівняно з неізохронною реалізацією, ізохронна є на 22% повільнішою, але все ще залишається дуже конкурентоспроможною.

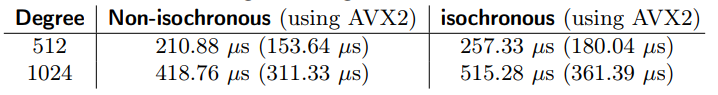


Рис. 6.3 – Час генерації підпису Faclon на частоті 2.5ГГц

**Захист часу кешування.** Запропонований для даної реалізації відбірник також забезпечує захист часу кешування часу.

**Адаптивність до інших схем.** Закономірним є питання, яким чином алгоритми можуть бути адаптовані для інших схем, окрім Falcon. Очевидним вузьким місцем, є розмір CDT, що використовується в , який є лінійним за стандартним відхиленням. Для більших стандартних відхилень, де лінійний пошук стає непрактичним, згортки можна використовувати для зменшення , а отже і часу роботи алгоритму пошуку.

**Переваги та обмеження.** Даний пробовідбірник має коефіцієнт приймання , що робить його особливо придатним, коли і близькі. Зокрема, даний відбірник на сьогоднішній день є найшвидшим ізохронним відбірником для параметрів у Falcon. Однак, чим більший розрив між та , тим менша швидкість прийняття. Крім того, він використовує сукупну таблицю розподілу (CDT), до якої здійснюється ізохронний доступ. Ця таблиця зростає, при зростанні , збільшуючи час роботи та використання пам'яті. Коли великий або далекий від , існують більш швидкі ізохронні пробовідбірники на основі методів згортки.

**Список джерел**

[1] James Howe, Thomas Prest, Thomas Ricosset, Mélissa Ross (2019) Isochronous Gaussian Sampling: From Inception to Implementation With Applications to the Falcon Signature Scheme.

[2] Pierre-Alain Fouque, Jeffrey Hoffstein, Paul Kirchner, Vadim Lyubashevsky, Thomas Pornin, Thomas Prest, Thomas Ricosset, Gregor Seiler, William Whyte, Zhenfei Zhang. Falcon: Fast-Fourier Lattice-based Compact Signatures over NTRU Specifications v1.0 – P.7,23-52. –https://falcon-sign.info/falcon.pdf

[3] J. Ahrens and U. Dieter. Extension of forsythe’s method for random sampling from the normal distribution. Mathematics of computation, 27:927–937, 1973.

[4] L. Ducas, A. Durmus, T. Lepoint, and V. Lyubashevsky. Lattice signatures and bimodal Gaussians. In R. Canetti and J. A. Garay, editors, CRYPTO 2013, Part I, volume 8042 of LNCS, pages 40–56. Springer, Heidelberg, Aug. 2013.

[5] N. C. Dwarakanath and S. D. Galbraith. Sampling from discrete gaussians for lattice-based cryptography on a constrained device. Applicable Algebra in Engineering, Communication and Computing, 25(3):159–180, 2014.

[6] G. E. Forsythe. Von neumann’s comparison method for random sampling from the normal and other distributions. Mathematics of Computation, 26(120):817–826, 1972

[7] A. Hülsing, T. Lange, and K. Smeets. Rounded Gaussians - fast and secure constant-time sampling for lattice-based crypto. In M. Abdalla and R. Dahab, editors, PKC 2018, Part II, volume 10770 of LNCS, pages 728–757. Springer, Heidelberg, Mar. 2018.

[8] Ducas, L., Lyubashevsky, V., and Prest, T. (2014). Efficient identity-based encryption over ntru lattices. In International Conference on the Theory and Application of Cryptology and Information Security, pages 22–41. Springer.

[9] Thomas Prest. Gaussian Sampling in Lattice-Based Cryptography. Theses, École Normale Supérieure, December 2015.

[10] Post-Quantum Cryptography. Round 3 Submissions.