Лабораторная работа № 4
Атака на подписи GGH/NTRU
05.05.2021

1 Алгоритм подписи GGH

Цель этой лабораторной – атаковать подпись GGH (названа в честь авторов Goldreich, Goldwasser, Halevi) [2], на основе которой было предложено несколько популярных подписей, в том числе запатентованная[1] подпись NTRU [1](сегодня мы знаем, как обойти эту атаку, и построить безопасную подпись, см. лекцию).

Для начала рассмотрим алгоритм подписи. Пару (публичный, секретный) ключи формируют базисы d-размерной решетки (полного ранга). Секретный ключ sk – 'хороший' базис решетки (например, HKZ-редуцированный), открытый ключ pk – 'плохой' базис этой же решетки (например, HNF форма sk).

Сообщения предполагаются выбранными из множества $\mathbb{Z}^d$ (любое отображение $\{0,1\}^\star \to \mathbb{Z}^d$ подойдет).

Процедура генерации подписи берет на вход хороший базис решетки sk и решает задачу нахождения ближайшего вектора (CVP) к m. Например, при достаточно редуцированном базисе для решения аппроксимации CVP запускается алгоритм Бабая. Заметьте, что вектор-ошибки $(m - \texttt{CVP}(sk, m))$ лежит в фундаментальном параллелепипеде базиса sk, $\mathcal{P}(sk)$, и этот вектор ошибки легко вычислить, зная пару (сообщение, подпись).

Процедура верификации, получив на вход тройку (m, σ, pk), проверяет с помощью pk – базиса L, евклидову норму ошибки $\|m - \sigma\|$. Если она мала, а именно меньше чем некая граница η (полагаем η известным), то подпись принимается, если нет, отклоняется.

function KeyGen($L = \mathcal{L}(B)$– решетка, порожденная B)
  return $pk = (\mathrm{HNF}(B), \eta), sk = \mathrm{HKZ}(B)$ ▷ η зависит от базиса Грам-Шмидта базиса sk.
end function

function Sign(m, sk)
  return $\sigma = \texttt{CVP}(sk, m)$
end function

function Verify(m, σ, pk)
  return $\|\sigma - m\| \leq \eta$
end function

В лабораторной работе, предлагается взять $sk = qI_d$ (с тривиальным алгоритмом Бабая) для некого фиксированного q, а в качестве $pk = U \cdot sk$, где U – случайная унимодулярная матрица. Вообще, так генерировать публичный ключ нельзя, но это работает быстро и не влияет на суть лабораторной. С корректно сгенерированным pk атака работает точно также.

Пример реализации подписи можно найти по ссылке `https://crypto-kantiana.com/elena.kirshanova/teaching/lattices_2021/lab4_GGHsign.sage`.

[1] Патенты в крипто–зло!

2 Атака

Проблема этой подписи заключается в том, что достаточное большое число подписей “выдают” форму фундаментального параллелепипеда $\mathcal{P}(sk)$, а вместе с ним и секретного ключа sk. Атака была формализована в 2006 в работе Нгуена-Регева [3] и состоит из следующих шагов:

1. Получение большого числа подписей под одним ключом (в этой лабораторной мы планируем взломать один фиксированный ключ)
2. Аппроксимация матрицы $sk^t sk$. Как это сделать описано в Разделе 4.1 статьи [3]. Из этой аппроксимации получить матрицу L, отображающую элементы $\mathcal{P}(sk)$ в элементы фундаментального параллелепипеда куба. В примере, выбранном в лабораторной, этот куб есть (I_d) (в статье описывается общий случай, когда этот куб может быть “перевернут”, то есть любой ортогональной матрицей). Матрица L получена на шаге 2 Алгоритма 1 в [3].
3. С помощью L отобразить все элементы $\mathcal{P}(sk)$ в элементы $\mathcal{P}(I_d)$, то есть в элементы куба со сторонами длины 1. Здесь допустима погрешность, так как на первом шаге мы получим не точное значение $sk^t sk$, а его аппроксимацию.
4. Аппроксимацией моментов этого куба (см. раздел 4.3 в [3]), получить аппроксимацию I_d. Суть этого шага в следующем: имея выборку и некоего многомерного тела (в нашем случае $\mathcal{P}(I_d)$), мы можем аппроксимировать образующую этого параллелепипеда (в нашем случае I_d), имея достаточное число элементов в выборке. Это своего рода обобщение неравенства Чебышёва (где аппроксимируется первый момент – мат. ожидание) на моменты более высоких порядков.
5. С помощью обратного отображения L^{-1}, отобразить I_d в sk. Это сделано в шаге 5 Алгоритма 1 в [3].

3 Задание к лабораторной

Задача: реализовать алгоритм, описанный в Разделе 4 статьи [3] и получить аппроксимацию sk.

Пояснения к заданию:

- Для предложенной размерности $d = 70$ понадобится не менее 100 000 подписей. Для такой размерности атака должна занимает около часа на 2,8 GHz Intel Core i5. Демонстрировать работоспособность кода можно и на меньших размерностях, однако нужно показать результат работы на больших размерностях.
- Параметры подписи, такие как форма sk, можно изменять (при этом убедитесь, что алгоритм Бабая работает верно).
- Матрица V раздела 4.1 в [3] есть sk в наших обозначениях. Однако обратите внимание, что в разделе 4.4 V уже обозначает другую матрицу (в нашем примере, I_d).
- Вектор ошибки, возвращаемый алгоритмом Бабая в реализации, лежит в фундаментальном параллелепипеде $\mathcal{P}_{1/2}(sk) = \{\sum_{i=1}^{d} x_i v_k \ x_i \in [-1/2, 1/2]\}$, в то время как атака в [3] описывает $\mathcal{P}(sk) = \{\sum_{i=1}^{d} x_i v_k \ x_i \in [-1, 1]\}$. Это, в частности, влияет на фактор 3, описанный в Лемме 1 в [3].
- При реализации Алгоритма 2 из [3] обратите внимание, что только те w, 4-ый момент которых близок к 1/5, ведут к верной аппроксимации I, а следовательно, sk. Подтверждает этот факт Лемма 3 в [3]. Не все w полезны. Параметр δ в этом алгоритме можно выбрать из диапазона $[0.65, 0.8]$.
- Аппроксимация $\mathrm{mom}_{sk}(w)$, упомянутая в разделе 4.3 в [3] реализуется с помощью неравенства Чебышёва [4].

Список литературы

[1] J. Hoffstein, N. A. Howgrave Graham, J. Pipher, J. H. Silverman, and W. Whyte. *NTRUSIGN: Digital signatures using the NTRU lattice.*

[2] O. Goldreich, S. Goldwasser, and S. Halevi. *Public-key cryptosystems from lattice reduction problems.*

[3] Phong Q. Nguyen and Oded Regev. *Learning a Parallelepiped: Cryptanalysis of GGH and NTRU Signatures.* EuroCrypt'06 `https://cims.nyu.edu/~regev/papers/gghattack.pdf`

[4] `https://en.wikipedia.org/wiki/Chebyshev%27s_inequality#Probabilistic_statement`