Scientific and Practical Cyber Security Journal (SPCSJ) 2(3): 92-98 ISSN 2587-4667
Scientific Cyber Security Association (SCSA)

HYBRID POST QUANTUM CRYPTO SYSTEM

ГИБРИДНАЯ ПОСТ КВАНОТОВАЯ КРИПТО СИСТЕМА

Maksim Iavich[1], Gagnidze Avtandil[2], Giorgi Iashvili[1]
[1]Caucasus University [2] Black Sea University

ABSTRACT. Scientists are actively working on the creation of quantum computers. Quantum computers can easily solve the problem of factoring large numbers. Because of this, quantum computers are able to crack the crypto RSA system, which is used in many products.

In the article it is proposed to replace the hash function with a lattice-based hash function in the standard Merkle scheme, and to use a one-way lattice-based function as a one-way function.

Ученые активно работают над созданием квантовых компьютеров. Квантовые компьютеры с легкостью могут решить задачу факторизации больших чисел. Благодаря этому квантовые компьютеры способны взломать крипто систему RSA, которая используются во многих продуктах.

В статье предложено в стандартной схеме Меркле заменить функцию хеширования, функцией хеширования основанной на решетках, а в качестве односторонней функции использовать одностороннюю функцию, основанную на решетках.

Крипто-системы основанные на решетках

Крипто-системы основанные на решетках, являются одной из альтернатив RSA. Данные крипто системы имеют очень надежные доказательства безопасности, основанные на «наихудших случаях» (worst-case hardness) и являются стойкими к атакам квантовых компьютеров. Безопасность крипто-системы основанных на решетках основана на сложности проблем решеток, основной из которых является проблема кратчайшего вектора (SVP).

Односторонние функции основанные на решетках

Аджтай предложил семейство односторонних функций, безопасность которых основана на наихудших случаях приближенного SVP с точностью nt, где t положительная константа.

Позже Голдреих показал, что данная функция является стойкой к коллизиям, что уже позволяет использовать ее в виде хеш функции. Проводится большая работа по уменьшению размера константы, в последних работах константа уже равна 1.

Функция имеет параметры n, m, a и b, которые являются целыми числами. Безопасность функции зависит от выбора n. Для хеширования $m > n\log a/ \log b$.

В качестве ключа выбирается матрица K из $Z^{n\times m}{}_a$.Одноразовая функция f работает следующим образом:
f(x) = Kx mod a. Функция переводит mlog b в nlog a битов. Как мы видим, вся арифметика может быть выполнена очень эффективно без необходимости использования точности целых чисел, которые обычно используют в криптографических функциях.

Крипто-системы основанные на хешировании

Электронной подписи, основанные на хешировании Электронной подписи, основанные на хешировании также являются пост квантовой альтернативой RSA. Данные системы используют криптографическую хеш функцию. Безопасность этих систем зависит от стойкости к коллизиям хеш функций, которые они используютх[1,2].

Одноразовые подписи

Lamport–Diffie one-time signature scheme.

Была предложена схема одноразовой подписи Лэмпорта (Lamport–Diffie one-time signature scheme). Для ключа подписи X в данной системе генерируется 2n случайных строк размера n.

$X= (x_{n-1}[0], x_{n-1}[1], \ldots, x_0[0], x_0[1]) \in \{0,1\}^{n,2n}$

Ключ верификации $Y= (y_{n-1}[0], y_{n-1}[1], \ldots, y_0[0], y_0[1]) \in \{0,1\}^{n,2n}$

Вычисляется следующим обраом:

$y_i[j] = f(x_i[j]), 0<=i<=n-1, j=0,1$

f – это односторонняя функция:

$f: \{0,1\}^n \rightarrow \{0,1\}^n$;

Как мы видим для генерации Y односторонняя функция f используется **2n** раз.

Подпись сообщения

Для подписи сообщения m мы хешируем:

$h(m)=hash = (hash_{n-1}, \ldots , hash_0)$

h- это криптографическая хеш функция:

$h: \{0,1\}^* \rightarrow \{0,1\}^n$

Подпись высчитывается следующим образом:

$sig= (x_{n-1}[hash_{n-1}], \ldots, x_0[hash_0]) \in \{0,1\}^{n,n}$

Длина подписи равна $\mathbf{n^2}$ односторонняя функция f не испульзуется.

Верификация сообщения

Для верификации подписи sig, сообщение хешируется hash = ($hash_{n-1}$, … , $hash_0$) после проверяется следующее равенство:

($f(sig_{n-1})$, …, $f(sig_0)$) = ($y_{n-1}[hash_{n-1}]$, …, $y_0[hash_0]$)

Если равенство верно, то подпись верна.

Для верификации односторонняя функция f используется **n** раз.

Winternitz one-time signature scheme

В схеме Лэмпорта генерация ключа и генерация подписи эффективны, но размер подписи равен n^2. Для его уменьшения используется схема Винтерница (Winternitz one-time signature scheme). В этой схеме одной строчкой ключа одновременно подписываются несколько битов хешированного сообщения.

Параметр Винтерница это количество битов хешированного сообщения, которое будет которые будут подписывать одновременно. Он выберяется w>=2.

Позже вычесляем:

$p_1=n/w$ и $p_2=(\log_2 p_1 +1+w)/w$, $p= p_1+ p_2$

Генерируем случайным образом ключи подписи:

$X= (x_{p-1}[0], …, x_0) \in \{0,1\}^{n,p}$

Вычесляем ключ верификации:

$Y= (y_{p-1}[0], …, y_0) \in \{0,1\}^{n,p}$, где

$y_i=f^{2\wedge w-1}(x_i)$, 0<=i<=p-1

Подпись сообщения

Длина подписи и ключа верификации равна **np** битам, односторонняя функция f используется $p(2^w-1)$ раз.

Для подписи сообщение хешируется hash=h(m) к hash прибавляется минимальное количество нулей, чтобы hash было бы кратно w. Позже делится на p_1 частей длины w.

$hash=k_{p-1},…, k_{p-p1}$

Контрольная сумма:

$c=\sum_{i=p-p1}^{p-1}(2^w-k_i)$

т.к. $c<= p_1 2^w$, длина ее двоичного представления меньше чем $\log_2 p_1 2^w+1$

Прибавляем к этому двоичному представлению минимальное количество нулей, чтобы оно было бы кратно w, и разделяем его на p_2 частей длины w.

$c=k_{p2-1},…, k_0$

подпись сообщения вычесляется следующим образом:

$sig=(f^{k_{p-1}}(x_{p-1}), \ldots, f^{k_0}(x_0))$

В худшем случае односторонняя функция f используется **$p(2^w-1)$** раз. Размер подписи равен pn.

Верификация подписи

Для верификации подписи $sig = (sig_{n-1}, \ldots, sig_0)$ вычисляются битовые строки $k_{p-1},\ldots, k_0$.

После проверяется следующее равенство:

$(f^{(2^w-1-k_{p-1})})(sig_{n-1}), \ldots, (f^{(2^w-1-k_0)})(sig_0) = y_{n-1}, \ldots y_0$

В худшем случае для верификации подписи нужно использовать функцию f **$p(2^w-1)$** раз.

Сравнение схем одноразовых подписей Lamport и Winternitz

	Lamport	Winternitz
Использование f для генерации ключей	2n	$p(2^w-1)$
Использование f для генерации подписи	Не используется	$p(2^w-1)$
Использование f для верификации подписи	n	$p(2^w-1)$

Merkle crypto-system

Одноразовые системы не удобны в использовании, т.к. для подписи каждого сообщения нужен уникальный ключ. Схема подписей Меркле позволяет одним и тем же ключом подписывать множество сообщений. Данная система использует одноразовые подписи и бинарное дерево, корень которого является открытым ключом.

Генерация ключа

Длина дерева должна быть H>=2 и одним открытым ключем можно подписать 2H документов. Генерируются ключи подписи и верификации; Xi, Yi, $0<=i<=2H$. X_i- это ключ подписи, Y_i- ключ верификации. Для получение листов дерево, хешируются ключи подписи с помощью хеш функции $h:\{0,1\}^* \rightarrow \{0,1\}^n$.

Для получения родительского узла, хешируется объединение двух предыдущих.

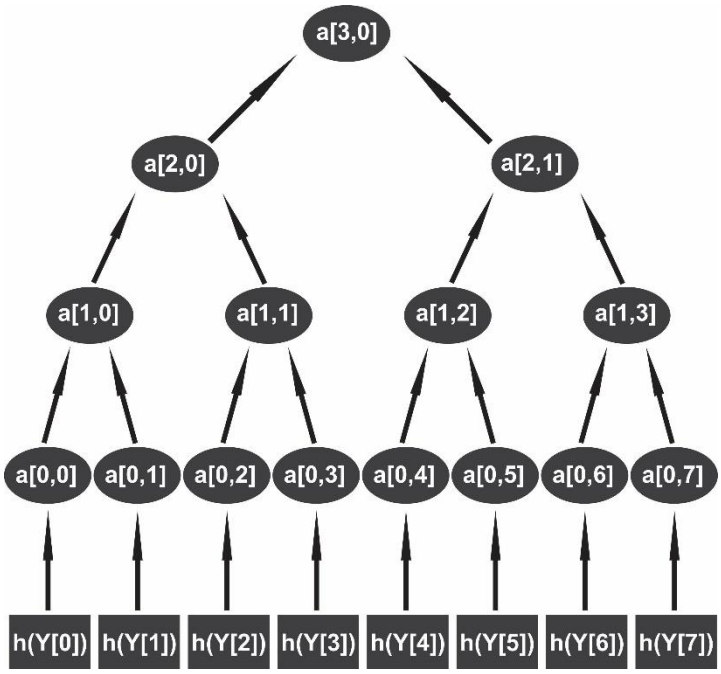

Fig. 1. Merkle the tree with H=3

Fig.1. дерево Меркле с H=3; a[i,j] узлы дерева;

a[1,0]=h(a[0,0] || a[0,1])

Корень дерева то открытый ключ подписи - pub. Для вычесление открытого ключа, нужно сгенерировать $\mathbf{2^H}$ пар открытых ключейand, а функция хеширования используется $\mathbf{2^{H+1}-1}$ раз[3].

Подпись сообщения

Для подписи сообщения любого размера оно хешируется и становится размера n.

h (m) = hash, для подписи сообщения используется произвольный одноразовый ключ X_{any} и подпись является объединением: одноразовой подписи, одноразового ключа верификации, индекса ключа и все братские узлы по отношению к выбранному произвольному ключу с индексом any.

Signature= (sig||any|| Y_{any}||$auth_0$,…,$auth_{H-1}$)

Верификация подписи

Для верификации подписи , проверятся одноразовая подпись с помощью выбранного ключа верификации, если верификация прошла вычисляются все a [i, j] используя "$auth_i$", index "any" и Y_{any}. Если корень дерева совпадет с открытым ключом, то подпись верна.

Как мы видим функция хеширования в Меркле используется **2^{H+1}-1, однострочные функции используются f в случае Винтерница используется** 3p(2^w-1), а в случае Лэмпотра 3n раз. Функии хеширования считаются стойкими к атакам квантового компьютера. Но алгоритм Гровера позволяет достичь квадратичного ускорения в алгоритмах перебора. Что значит, что хеш функции надо усложнять для квантовых компьютеров. Также проводятся исследования по определению стоимости атак на SHA2 и SHA3 семейства хеш функций с помощью алгоритма Гровера [4]

Мы предлагаем заменить функцию хеширования, функцией хеширования основанной на решетках, а в качестве односторонней функции использовать одностороннюю функцию, основанную на решетках.

Если использовать семейство односторонних функций предложенных Аджтаем. В случае хеш функций в качестве ключа выбираем матрицу K из $Z^{n\times m}_a$, которая переводит mlog b в nlog a количество нулей и единиц и вычисляем h(x) = Kx mod a. [5]

В случае односторонней функции в качестве ключа выбираем матрицу K из $Z^{n\times n}_b$, которая переводит mlog b в mlog b количество нулей и единиц и вычисляем f(x) = Kx mod a.

В данном варианте предлагается использовать односторонних функций предложенных Аджтаем, это предложено на уровне начальной идее. Стоит рассмотреть идею использования оптимизированных односторонних функций основанных на решетках.

REFERENCES

1. Avtandil Gagnidze, Maksim Iavich, Giorgi Iashvili// Novel Version of Merkle Cryptosystem// BULLETIN OF THE GEORGIAN NATIONAL ACADEMY OF SCIENCES, vol. 11, no. 4, 2017, p. 28-33

2. Явич М.П., Аракелян А.А. Реализация крипто-системы Merkle и ее анализ // Современные научные исследования и инновации. 2017. № 6 [Электронный ресурс]. URL: http://web.snauka.ru/issues/2017/06/83971

3. C. Liu, R. Ranjan, C. Yang, X. Zhang, L. Wang and J. Chen, "MuR-DPA: Top-Down Levelled Multi-Replica Merkle Hash Tree Based Secure Public Auditing for Dynamic Big Data Storage on Cloud," in IEEE Transactions on Computers, vol. 64, no. 9, pp. 2609-2622, 2015.

doi:10.1109/TC.2014.2375190

4. [Amy M., Di Matteo O., Gheorghiu V., Mosca M., Parent A., Schanck J. (2017) Estimating the Cost of Generic Quantum Pre-image Attacks on SHA-2 and SHA-3. In: Avanzi R., Heys H. (eds) Selected Areas in Cryptography – SAC 2016. SAC 2016. Lecture Notes in Computer Science, vol 10532. Springer, Cham]

5. Güneysu T., Lyubashevsky V., Pöppelmann T. (2012) Practical Lattice-Based Cryptography: A Signature Scheme for Embedded Systems. In: Prouff E., Schaumont P. (eds) Cryptographic Hardware and Embedded Systems – CHES 2012. CHES 2012. Lecture Notes in Computer Science, vol 7428. Springer, Berlin, Heidelberg