Bitcoin Security with Post Quantum Cryptography

Meryem Cherkaoui Semmouni[1(✉)], Abderrahmane Nitaj[2], and Mostafa Belkasmi[1]

[1] SIME, Mohammed V University, ENSIAS Rabat, Rabat, Morocco
{meryem.semmouni,m.belkasmi}@um5s.net.ma
[2] University of Caen Normandie, Caen, France
abderrahmane.nitaj@unicaen.fr

Abstract. In a future quantum world with a large quantum computer, the security of the digital signatures used for Bitcoin transactions will be broken by Shor's algorithm. Bitcoin has to switch to post-quantum cryptography. In this paper, we show that the post quantum signatures based on LWE and ring LWE are the most promising to use in the presence of large quantum computers running Shor's algorithm.

Keywords: Bitcoin · Elliptic curve · Lattice · Learning with error

1 Introduction

The influence of new technologies on the economy and individuals has given birth to a new interpretation of money that makes life more easier. This new interpretation aims to emigrate from cash to an electronic money recorded in electronic devices. The use of electronic money is encouraged in several countries. It also has a lot of benefits, so transactions have become easy, cheap, more reliable and can be done anywhere and at anytime. The increase of frauds and the different attacks launched by the hackers, gave birth to the privacy and authentication problem for this kind of electronic system with this kind of danger. In this context cryptography offers multiple solutions to overcome these sensitive data protection issues in e-commerce.

An important application of cryptography is to secure Bitcoin. Bitcoin is a peer-to-peer network without any central authority such as banks or governments. It was presented in 2008 by Satoshi Nakomoto [17]. To authorize payments or transfers, Bitcoin uses the Elliptic Curve Digital Signature Algorithm (ECDSA) [12] with the hash function SHA-256 [13], and the Koblitz curve secp256k1 with the equation:

$$secp256k1: \ y^2 = x^3 + 7 \pmod{p_1}, \ p_1 = 2^{256} - 2^{32} - 2^9 - 2^8 - 2^7 - 2^6 - 2^4 - 1.$$

The curve secp256k1 was proposed in 2000 by the Standards for Efficient Cryptography Group of Certicom in the standards for efficient cryptography SEC2

M. F. Atig and A. A. Schwarzmann (Eds.): NETYS 2019, LNCS 11704, pp. 281–288, 2019.
https://doi.org/10.1007/978-3-030-31277-0_19

and is used in Bitcoin since 2009. The Koblitz curve has many advantages when used in industrial applications, especially efficiency, security and shortness of the key, but the main problem is its weakness in front of quantum attacks.

In this paper, we study the possibility of using the digital signature TESLA# [8] (pronounced "Tesla Sharp") for Bitcoin system. TESLA# has many advantages.

- TESLA# is based on the Ring Learning with Errors (R-LWE) assumption which makes it a prominent candidate for a post-quantum digital signature.
- TESLA# improves all its predecessors such as Ring-Tesla [3] and Tesla.
- TESLA# has a fast key generation, signing and verification.
- TESLA# has highly secure parameters at the level of both pre-quantum and post-quantum cryptography.
- TESLA# has a secure implementation against timing and cache attacks.

We show that TESLA# is an efficient signature scheme in the context of Bitcoin which avoids quantum attacks. We recommend to use it in the future: it is better to use Momentum proof of work which is better than the standard proof of work [2] used todays on Bitcoin Blockchain transaction.

The rest of this paper is organized as follows. In Sect. 2, we recall some facts on Bitcoin and secp256k1. In Sect. 3, we introduce lattices and describe the digital signature scheme TESLA#. In Sect. 4 we study its security. In Sect. 5 we describe the use of TESLA# in Bitcoin. We conclude the paper in Sect. 6.

2 Description of the Cryptography Used in Bitcoin

Bitcoin is a peer-to-peer decentralized digital currency based on asymmetric cryptography. It was first proposed by Satoshi Nakamoto [17] in 2008 and exploited since 2009. It is a proof of work based on cryptocurrency which makes miners able to mining on Bitcoin, and users to transfer directly without the use of an intermediary such as a bank or a government, using just their addresses. Bitcoin is implemented using the Elliptic Curve Digital Signature Algorithm (ECDSA) to verify ownership transactions on the network, with the Koblitz curve Secp256k1. This curve Secp256k1 is define over a finite field $\mathcal{F}_p$ as follows:

- The prime number is $p = 2^{256} - 2^{32} - 2^9 - 2^8 - 2^7 - 2^6 - 2^4 - 1$.
- The equation curve is $y^2 \equiv x^3 + 7 \pmod{p}$.
- The maximum length of the keys is $\lceil \log_2(p) \rceil = 256$.

The hard problem upon which the security is based is the Elliptic Curve Logarithm Problem ECDLP. Unfortunately, the problem can be solved by a quantum computer running Shor's algorithm [19]. For 256 bits, Shor's algorithm needs only 3848 seconds to solve ECDLP.

3 The Digital Signature Scheme TESLA#

In this section, we show how to avoid quantum attacks on Bitcoin with ECDSA by using the lattice-based signature TESLA# [8].

3.1 Lattice

The arithmetic used in TESLA# is based on lattices.

Definition 1. *Let $B = \{b_1, \ldots, b_n\}$, $b_i \in \mathbb{R}^m$ be a set of n linearly independent vectors of m coordinates with $n \leq m$. The lattice $\mathcal{L}$ associated to B is the discrete additive subgroup of $\mathbb{R}^m$ containing all integer linear combinations of the vectors of B:*

$$\mathcal{L}(B) = \left\{ \sum_{i=1}^{n} x_i b_i \quad | \quad x_i \in \mathbb{Z} \right\}.$$

The integer n is the dimension of the lattice and m is the rank. When $m = n$, the lattice is called full-rank. The basis B can be represented as a matrix $B = [b_1, \ldots, b_n]$. The determinant of the lattice is defined by $\det(\mathcal{L}) = \sqrt{B^T \cdot B}$ where B is considered here as the matrix of the vectors $b_1, \ldots, b_n$. In the theory of lattices, several problems are considered hard and are resistant to quantum computers. Lattice-based cryptography is based on the hardness of some lattice problems such as SVP, CVP, and LWE. We list below the main hard problems.

1. **The Shortest Vector Problem (SVP):** Given a lattice basis B, find the shortest nonzero vector in $\mathcal{L}(B)$.
2. **The Closest Vector Problem (CVP):** Given a lattice basis B and a target vector v_0 not in the lattice $\mathcal{L}(B)$, find $v \in \mathcal{L}(B)$, the closest vector to v_0.
3. **Learning With Errors Problem (LWE):** Let A be a $n \times n$ matrix which is uniformly distributed in $\mathbb{Z}/q\mathbb{Z}$. Let s and e be two unknown vectors. The LWE problem is to find s and e using A and $As+e$ with the shortest non-zero vector for the Euclidean norm.
4. **Ring-Learning With Errors Problem (Ring-LWE):** Ring-LWE problem is similar than the LWE problem where the unknown parameters s and e are vectors from a the ring of polynomials $\mathcal{R}_q = \mathbb{Z}_q[x]/(x^n + 1)$.

3.2 The Digital Signature Scheme TESLA#

Tesla# [8] is a candidate for post-quantum digital signatures. It is provably secure with a security reduction to the Ring Learning With Errors (Ring-LWE) problem. The digital signature scheme TESLA# is composed by three algorithms: the key generation algorithm, the signing algorithm, and the verification algorithm.

4 The Security of TESLA#

The Ring-LWE problem is a hard assumption, that was introduced in [16] together with a (quantum) worst case to average-case reduction to certain problems over ideal lattices.

The security of TESLA# stems from the hardness of the Ring Learning with Errors (Ring-LWE) problem. The Ring-LWE problem can be seen as an instantiation of the LWE problem. In this section we present the main attacks against Tesla# signature and countermeasures to avoid these attacks.

4.1 The Decoding Attack

An LWE instance $(A, As + e)$ is seen as an instance of the bounded distance decoding problem (BDDP). The most basic way of solving a BDD instance is using Babai's Nearest Plane algorithm [6]. This method can be described as follows: First Suppose that there is multiple samples $(A, As + e)$ of an LWE instance parameterized by n, α and q. Second, perform lattice reduction basis on the lattice $\mathcal{L}(A^T)$ to obtain a new basis B, where A^T is the transpose of A. Babai's Nearest Plane algorithm works by recursively computing the closest vector on the sublattice spanned by subsets of the the reduced basis. This attack is based on reducing the lattice by lattice reduction techniques such as LLL [15].

The probability to recover the vector s by Babai's Nearest Plane algorithm is approximated by

$$\prod_{i=1}^{m} erf\left(\frac{\|b_i^*\|\sqrt{\pi}}{2\alpha q}\right),$$

where $\{b_1^*, ..., b_m^*\}$ is the Gram-Schmidt orthogonal basis. To ovoid the attack by Baba's technique, the probability must be small, that is $\prod_{i=1}^{m} erf\left(\frac{\|b_i^*\|\sqrt{\pi}}{2\alpha q}\right) < \epsilon$ for a small parameter ϵ .

On the other hand, the complexity of the LLL algorithm is $\mathcal{O}\left(e^{n^3 C} \log M\right)$ where $C > (2/\sqrt{3})^{1/6}$ and M is maximum length of the basis vectors $\{b_1, ..., b_n\}$, that is $M = \max_{i=1}^{n} \|b_i\|$. As a consequence, to ovoid the LLL algorithm attack, the dimension n of the lattice should be large.

4.2 Lattice Reduction

Lattice reduction is to find short vectors in the scaled dual lattice. We construct this lattice from a given $A \in \mathbb{Z}_q^{m \times n}$ by computing a basis for the nullspace of A^Tover $\mathbb{Z}_q$, lift to $\mathbb{Z}$ and extend by $qI \in \mathbb{Z}_q^{m \times m}$ to make it q-ary and compute a basis for $\mathcal{L}$, we obtain at the end the scaled dual lattice $\mathcal{L} = \{x \in \mathbb{Z}_q^m | xA \equiv 0 \mod q\}$. Lattice reduction will return the shortest non-zero vector b_0 which by definition is a short vector in $\mathcal{L}$, so that $b_0 A \equiv 0 \mod q$ which is exactly solving the Short Integer Solutions problem.

So $\langle b_0, As + e\rangle = \langle b_0, e\rangle$, which follows a Gaussian distribution and it often returns small samples for both b_0 and e.

Given an LWE instance characterised by n, α, q and a vector b_0 of length $\|b_0\|$ in the scaled dual lattice $\mathcal{L}^T = \{x \in \mathbb{Z}_q^m | xA \equiv 0 \mod q\}$, the advantage of distinguishing $\langle b_0, e\rangle$ from random is close to $e^{-\pi(\|b_0\|\alpha)^2}$. So if $\|b_0\|$ is too large then the (Gaussian) distribution of $\langle b_0, e\rangle$ will be too flat to distinguish from random. To avoid this attacks $\|b_0\|\alpha$ must be large enough.

4.3 Non-lattice Attacks

There are two non-lattice approaches to solve LWE, namely the attack based on the algorithm by Blum, Kalai, and Wassermann (BKW) [7] and the algorithm

by Arora and Ge [5]. Both algorithms require a large number of LWE samples to be applied efficiently.

BKW solves LWE via the SIS strategy, given m samples (A, c) following D_σ^n, we require short vectors u_i in the scaled dual lattice of the lattice generated by the rows of A. BKW creates these vectors by adding elements from a tables with q^b entries each, where each table is used to find collisions on b components of a (a row of A). The BKW algorithm shows that subexponential algorithms exist for learning parity functions in the presence of noise: the BKW algorithm solves the Learning Parity with Noise problem in time $2^{O(n/logn)}$ [1].

An alternative approach, proposed by Arora and Ge, is used to solve LWE by setting up a system of noise-free non-linear polynomials of which the secret s is a root [5]. Polynomials are constructed from the observation that the error, when falling in the range $[-t, t]$ (for some $t \in \mathbb{Z}$ such that $2t + 1 < q$), is always a root of the polynomial $P(x) = x \prod_{i=1}^{t}(x + i)(x - i)$. Then, we know that the secret s is a root of $P(a \cdot x - c)$ constructed from LWE samples. Arora and Ge, offer an algorithm for solving LWE in time $2^{O(n^{2\xi})}$ where ξ is a constant such that $\alpha q = n^{\xi}$.

Both algorithms require a (very) large number of LWE samples to be applied efficiently. TESLA# inherits the property from Ring-TESLA that Gaussian sampling is only needed for key pair generation. Instances for [3] are given far less LWE samples, so TESLA# also will give less LWE samples. TESLA# is resistant to such attacks.

4.4 Timing Attacks

Tesla# uses an isochronous (Constant time) Gaussian sampler [8] that improve the Gaussian sampler proposed first by Ducas et al. [10]. This improved Gaussian sampler is used to speed up the computation of TESLA#'s key generation and to protect against timing attacks by taking the "same time" of execution regardless of the private data. The design of new algorithm consists to sample according to the Bernoulli distribution $B_{e^{-t/2\sigma^2}}$ with t is an l-bit integer.

4.5 Parameters Recommendation

For hardness guarantees [9], the ring R_q must be instantiated so that $q \equiv 1 \mod 2n$, and the Gaussian parameter $\sigma\sqrt{2\pi}$ must be greater than or equal to two. The parameters presented in [8] provide 128-bit post-quantum security and 256-bit classical security for TESLA#.

5 Using TESLA# for Bitcoin

In this section, we show how to provide more security for Bitcoin systems in the presence of quantum computers.

5.1 Hash Function

In the Bitcoin, the Koblitz curve secp256k1 is combined with the hash function SHA-256 in the ECDSA signature process while TESLA# uses BLAKE2 [4] and the more recent and more secure hash function SHA-3 [11].

5.2 Authentication Process

An efficient quantum algorithm to solve ECDLP problem was given by Shor. Since Bitcoin signature scheme is ECDSA based on ECDLP problem, these attack will impact Bitcoin authentication system security. The bitcoin signature used for authentication is generated by signing the hash of the transaction and the public key belongs to the payer. Both the signature and public key prove the transaction is created by the owner of the bitcoin address.

In Bitcoin system, authentication with cryptographic digital signature is used to secure and authorize payments or transfers. In this paper we demonstrate that TESLA# is a secure signature against the quantum attacks and gives a fast signing and verifying signing, also private key will not be revelated from the public key so the address and transactions will become secure. TESLA# is an efficient signature in the context of Bitcoin to avoid quantum attacks, it could be used to replace the ECDSA digital signature based on Elliptic curve Discret Logarithm Problem which is breakable by a Shor's algorithm.

5.3 Bitcoin Mining

The security of Bitcoin is based on mining with Proof Of Work, in this phase the most important parameter is the hash function. For the present architecture of Bitcoin, the hash function is SHA-256.

Thanks to Grover's quantum search algorithm [14], it is now possible to perform the Bitcoin proof of work using a quadratical fewer hashes needed in standard proof of work using SHA-256, so the use of another type of hash fuction is recommended. To enhance the secuity of mining, it is better to use Momentum proof of work which is better than the standard proof of work [2] which is used for Bitcoin transaction.

6 Conclusion

We have studied and compared the digital signature ECDSA based on the Koblitz elliptic curve secp256k1 and the digital signature TESLA# based on lattices and the Learning with error problem for use in Bitcoin. Our analysis shows that the signature TESLA# is more secure than ECDSA, especially for Shor's quantum attack on the elliptic discrete lograrithm problem. We conclude that TESLA# is more suitable and secure for use in the Bitcoin, especially for long term applications.

References

1. Albrecht, M.R., Player, R., Scott, S.: On the concrete hardness of learning with errors. J. Math. Cryptol. **9**(3), 169–203 (2015)
2. Aggarwal, D., Brennen, G.K., Lee, T., Santha, M., Tomamichel, M.: Quantum attacks on Bitcoin, and how to protect against them. arXiv preprint arXiv:1710.10377 (2017)
3. Akleylek, S., Bindel, N., Buchmann, J., Krämer, J., Marson, G.A.: An efficient lattice-based signature scheme with provably secure instantiation. In: Pointcheval, D., Nitaj, A., Rachidi, T. (eds.) AFRICACRYPT 2016. LNCS, vol. 9646, pp. 44–60. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-31517-1_3
4. Aumasson, J.-P., Neves, S., Wilcox-O'Hearn, Z., Winnerlein, C.: BLAKE2: simpler, smaller, fast as MD5. In: Jacobson, M., Locasto, M., Mohassel, P., Safavi-Naini, R. (eds.) ACNS 2013. LNCS, vol. 7954, pp. 119–135. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-38980-1_8
5. Arora, S., Ge, R.: New algorithms for learning in presence of errors. In: Aceto, L., Henzinger, M., Sgall, J. (eds.) ICALP 2011. LNCS, vol. 6755, pp. 403–415. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-22006-7_34
6. Babai, L.: A las vegas-NC algorithm for isomorphism of graphs with bounded multiplicity of eigenvalues. In: 27th FOCS, pp. 303–312. IEEE Computer Society Press, Toronto, 27–29 October 1986
7. Blum, A., Kalai, A., Wasserman, H.: Noise-tolerant learning, the parity problem, and the statistical query model. In: 32nd ACM STOC, pp. 435–440. ACM Press, Portland, 21–23 May 2000
8. Barreto, P.S., Longa, P., Naehrig, M., Ricardini, J.E., Zanon, G.: Sharper ring-LWE signatures. Cryptology ePrint Archive, Report 2016/1026 (2016)
9. Chopra, A.: Improved parameters for the ring-TESLA digital signature scheme. IACR Cryptology ePrint Archive 2016, p. 1099 (2016)
10. Ducas, L., Durmus, A., Lepoint, T., Lyubashevsky, V.: Lattice signatures and bimodal Gaussians. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013. LNCS, vol. 8042, pp. 40–56. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-40041-4_3
11. Dworkin, M.J.: SHA-3 standard: permutation-based hash and extendable-output functions. National Institute of Standards and Technology (NIST), Gaithersburg (MD), USA, August 2015
12. FIPS PUB 186–4, Digital Signature Standard (DSS), July 2013. http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.186-4.pdf
13. FIPS PUB 180–4, Secure Hash Standard (SHS). https://nvlpubs.nist.gov/nistpubs/fips/nist.fips.180-4.pdf
14. Grover, L.K.: A fast quantum mechanical algorithm for database search. In: Proceedings of the ACM STOC 1996, pp. 212–219. ACM, May 1996
15. Lenstra, A.K., Lenstra, H.W., Lovasz, L.: Factoring polynomials with rational coefficients. Math. Ann. **261**, 513–534 (1982)
16. Lyubashevsky, V., Peikert, C., Regev, O.: On ideal lattices and learning with errors over rings. In: Gilbert, H. (ed.) EUROCRYPT 2010. LNCS, vol. 6110, pp. 1–23. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13190-5_1

17. Nakamoto, S.: Bitcoin: a peer-to-peer digital cash system, 24 May 2009. https://bitcoin.org/bitcoin.pdf
18. Rivest, R.L., Shamir, A., Adleman, L.: A method for obtaining digital signatures and public-key cryptosystems. Commun. ACM **21**(2), 120–126 (1978)
19. Shor, P.W.: Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM J. Comput. **26**, 1484–1509 (1997)