Estudo preliminar da adoção de assinaturas baseadas em hash no blockchain do Bitcoin

Antônio Unias de Lucena[1], Marco Aurélio Amaral Henriques[1]

[1]Faculdade de Engenharia Elétrica e de Computação
Universidade Estadual de Campinas (Unicamp)
Campinas-SP-Brasil CEP 13083-852

{alucena,marco}@dca.fee.unicamp.br

Abstract. *A quantum computer with high processing capacity will break the digital signatures used by the main blockchains. This work brings a preliminary study on the adoption of hash-based signatures in the Bitcoin's blockchain in order to make it resistant to quantum computers. Our study describes which features need to be changed and their impacts. The major impact is on the digital signature size, leading to a bigger transaction and lowering the number of transactions per block. The solutions to such a problem can be to increase the block size, reduce the size of hash-based signatures and/or adopt post-quantum algorithms with smaller digital signatures.*

Resumo. *Um computador quântico com alta capacidade de processamento quebrará as assinaturas digitais utilizadas nos principais blockchains. Este trabalho traz um estudo preliminar sobre a adoção de assinaturas baseadas em hash no blockchain do Bitcoin para torná-lo resistente ao computador quântico. O estudo mostra os pontos a serem modificados e seus impactos. O maior deles refere-se ao tamanho da assinatura digital, que influencia no tamanho de uma transação e diminui o número de transações por bloco. Soluções para este problema incluem o aumento do tamanho do bloco, a diminuição do tamanho das assinaturas baseadas em hash e/ou a adoção de algoritmos pós-quânticos que produzam assinaturas digitais menores.*

1. Introdução

Os principais blockchains da atualidade não são resistentes à criptanálise realizada por um computador quântico de alta capacidade de processamento, sendo possível, desta forma, adulterar uma transação antes desta ser armazenada no blockchain, bem como adulterar os dados já armazenados no mesmo, tornando inseguras aplicações desenvolvidas em blockchains. O motivo disso é que existem algoritmos desenvolvidos para computadores quânticos que são capazes de quebrar os esquemas de assinatura digital ECDSA (Elliptic Curve Digital Signature Algorithm) [Shor 1994] e diminuir pela metade o número de bits do espaço de busca das funções hash utilizadas para amarrarem os blocos de um blockchain [Grover 1996].

Este artigo traz um estudo preliminar do impacto da alteração do esquema de assinatura digital e da função hash utilizados no blockchain para opções resistentes ao computador quântico. Pelo fato de serem mais conhecidas e se basearem apenas em conceitos já bem utilizados, as assinaturas baseadas em hash (HBS - Hash Based Signatures) são o foco deste primeiro estudo.

O artigo está organizado da seguinte forma: a seção 2 explana o conceito de blockchain; a seção 3 detalha por que um computador quântico vai tornar blockchains inseguros; a seção 4 apresenta o conceito e os esquemas HBS abordados no estudo; a seção 5, por sua vez, apresenta as modificações nos esquemas de assinatura e de hash e os impactos das mesmas no desempenho do blockchain. Por fim, o artigo traz algumas conclusões e propostas de trabalhos futuros.

2. O blockchain do Bitcoin

O blockchain é uma base de dados distribuída, descentralizada, resistente a adulterações e na qual é possível apenas acrescentar dados. Ele surgiu com a criptomoeda Bitcoin [Nakamoto 2008] para ser um livro-razão, armazenando todas as transações realizadas pelos usuários da criptomeda, de forma a impedir gasto duplo (uma quantia de dinheiro ser utilizada mais de uma vez). Apesar de já existir um grande número de blockchains, este trabalho é focado apenas no blockchain da criptomoeda Bitcoin.

O blockchain utiliza funções hash e assinatura digital em sua estrutura. Funções hash aumentam a segurança e dificultam a adulteração dos dados armazenados. Assinatura digital, por sua vez, autentica a origem de toda atividade (transação) armazenada em um bloco.

Uma das inovações do blockchain foi armazenar em um bloco o hash do bloco anterior e organizar as transações de um bloco em uma árvore de Merkle [Merkle 1980]. Assim, qualquer modificação em uma transação é percebida por causa da mudança da raiz da árvore de Merkle e qualquer adulteração em um bloco é perceptível devido à discrepância que surge com o hash armazenado no cabeçalho do próximo bloco.

Um bloco do blockchain do Bitcoin é formado por duas partes: cabeçalho e corpo. O cabeçalho armazena informações como a versão do Bitcoin utilizada, dificuldade (valor que vai definir o esforço computacional necessário para criação de novos blocos e moedas – processo de mineração), nonce (número aleatório usado durante a mineração), hash do cabeçalho do bloco anterior e a raiz da árvore de Merkle que representa as transações contidas no corpo do bloco. O corpo, por sua vez, armazena as transações escolhidas pelo minerador dentro de um conjunto que está em fila de espera.

Uma transação é a transferência de um valor em moeda de um endereço para outro na rede do Bitcoin. Uma transação é constituída por duas partes: uma que se refere às informações do endereço de onde provêm as moedas e outra referente ao endereço a que se destinam as moedas utilizadas na transação. A origem da transação utiliza um *script*, conjunto de instruções executados para validar a transação, *scriptSig*, que contém a assinatura que confirma que o usuário possui a chave privada relacionada ao endereço receptor de uma transação anterior através da qual foram recebidas as moedas agora utilizadas. O destino da transação, por sua vez, possui outro *script*, *scriptPubKey*, que contém o endereço do destinatário da transação. Quando se faz uma transação, a chave pública e a assinatura ficam disponíveis e, assim, podem ser usadas em um ataque para encontrar a chave privada.

3. Impacto do computador quântico sobre os esquemas atuais de assinatura digital

O advento de um computador quântico de alta capacidade de processamento resolverá

problemas que hoje demandam muito esforço de computadores clássicos. Um exemplo é o cálculo de logaritmos discretos pelo algoritmo quântico de Shor [Shor 1994], o que inviabilizará algoritmos de assinatura como o ECDSA, utilizado nas assinaturas de transações de Bitcoin.

O algoritmo de Grover [Grover 1996], por sua vez, diminui o espaço de busca para um ataque de força bruta de 2^N para $2^{N/2}$, ou seja, o número de bits de segurança do espaço será reduzido pela metade. Isso vale para os bits de saída de uma função hash. A consequência do algoritmo de Grover é que as funções hash terão que ter seus números de bits dobrados para manter a segurança original.

Para se antecipar ao computador quântico, a comunidade científica vem desenvolvendo algoritmos de criptografia resistentes a ataques destes computadores. Dentre as opções existentes, este trabalho foca inicialmente na de assinaturas baseadas em funções hash, deixando outras alternativas para trabalhos futuros.

4. Assinaturas baseadas em hash

Assinaturas baseadas em hash (HBS – Hash Based Signatures) são assinaturas que fazem uso apenas de funções hash. As HBS foram primeiramente desenvolvidas por Lamport [Lamport 1979], mas não chamaram muito a atenção devido ao fato de poderem ser utilizadas apenas uma vez e por serem relativamente grandes quando comparadas a assinaturas RSA (Rivest–Shamir–Adleman) e ECDSA. Merkle modificou o esquema de assinaturas proposto por Lamport e desenvolveu a assinatura WOTS – Winternitz One-Time Signature [Merkle 1979]. Este esquema tem assinaturas e chaves consideravelmente menores, mas exige mais cálculos de funções hash em sua operação.

Funções hash são funções de mão única que transformam uma mensagem de tamanho arbitrário em um arranjo de bits de tamanho fixo. Elas são atraentes para criptografia por causa das seguintes propriedades: (1) é computacionalmente inviável encontrar a mensagem M a partir de seu hash H(M) (função não inversível); (2) dados uma mensagem M e seu hash H(M), é computacionalmente inviável achar uma mensagem M’ tal que H(M) = H(M’) (resistência à segunda pré-imagem); (3) é computacionalmente inviável encontrar duas mensagens M e M’ quaisquer tais que H(M) = H(M’) (resistência à colisão).

Dentre os esquemas HBS destacam-se o XMSS [Buchmann et al 2011], SPHINCS [Bernstein et al 2015] e SPHINCS$^+$ [Bernstein et al 2017]. Os dois últimos esquemas se destacam por serem *stateless*, isto é, o usuário não necessita de informações sobre a última chave privada utilizada antes de fazer uma nova assinatura.

O XMSS é construído como uma árvore de Merkle em cujas folhas são armazenadas as chaves públicas utilizadas pelo usuário a cada assinatura. Cada folha do XMSS armazena chaves do tipo WOTS$^+$ [Hülsing 2013]. É um esquema dito *stateful*, pois o usuário necessita armazenar dados sobre a última folha utilizada.

O SPHINCS é construído de forma semelhante ao XMSS, porém, em vez de ser formado por uma única árvore, é construído como um conjunto de árvores, tendo várias camadas de árvores em sua estrutura. Em cada uma destas camadas, excetuando a mais baixa, as folhas são as raízes de sub-árvores armazenando chaves WOTS$^+$. Na camada mais baixa, as folhas são raízes de árvores HORST, as quais armazenam chaves HORS (Hash to Obtain Random Subset) [Reyzin & Reyzin 2002]. O esquema de assinatura HORS permite que uma mesma chave seja utilizada em mais de uma assinatura, sendo

um esquema do tipo FTS (few-times signature).

O SPHINCS$^+$, por sua vez, é uma modificação do SPHINCS que utiliza árvores FORS (Forest of Random Subset), em vez de árvores HORS. O SPHINCS$^+$ é uma das propostas de padrão de assinatura pós-quântica submetidas para o NIST [Post-Quantum Cryptography Standardization (2018)]. Existe um outra proposta para NIST, chamada GRAVITY-SPHINCS [Aumasson & Endignoux 2018], que utiliza árvores PORS (PRNG to Obtain a Random Subset) no lugar das árvores HORS.

5. Impactos no blockchain provocados pela alteração do esquema de assinatura digital e da função hash utilizados

5.1. Alterações necessárias e parâmetros adotados

A fim de amenizar o impacto do computador quântico no blockchain do Bitcoin, duas modificações em seu projeto são necessárias: (a) dobrar o número de bits da função hash utilizada para conectar os blocos; (b) substituir os esquemas de assinatura digital por opções resistentes ao computador quântico.

O blockchain armazena em cada um dos seus blocos o hash do bloco anterior. Atualmente é utilizado uma função hash de 256 bits que terá seu número de bits modificado para 512 bits na versão proposta do Bitcoin resistente ao computador quântico, doravante denominada PQ-BTC. O blockchain utiliza outra função hash, RIPEMD-160, para gerar, junto com SHA-256, o endereço público do usuário na rede do blockchain. A função RIPEMD-160 não precisa ter seus números de bits duplicados, pois ela apenas comprime o hash SHA-256 da chave pública antes de armazená-lo no *scriptPubkey*, não sendo alvo de criptanálise.

A alteração do número de bits da função hash que interconecta os blocos causa impacto nos campos *previous tx hash* da transação e também no cabeçalho do bloco, alterando o tamanho dos campos *previous block* e *merkle root*. O cabeçalho do bloco do Bitcoin possui 80 bytes de tamanho, enquanto o cabeçalho do PQ-BTC, terá (64 + 64 + 16 = 144) bytes, onde 64 bytes são relativos ao hash do bloco anterior, outros 64 bytes referem-se à raiz da árvore de Merkle e 16 bytes são destinados aos demais campos do cabeçalho, que permanecerão constantes.

Tabela 1 – Tamanhos típicos de assinatura, chave privada e pública para vários esquemas de assinatura

Tipo de assinatura	Tamanho (bytes)		
	assinatura	**chave privada**	**chave pública**
XMSS	2800[a]	1300	2200
SPHINCS	41000	1088	1056
SPHINCS$^+$-128s [b]	8080	64	32
GRAVITY-SPHINCS	≈ 30000	64	32
ECDSA do Bitcoin	71	32	33[c]

[a] Tamanho de assinatura para árvore com altura h = 20, utilizando SHA-256 e capaz de assinar até 2^{20} mensagens.
[b] Versão do SPHINCS$^+$ com o menor tamanho de assinatura.
[c] Chave pública de 64 bytes, mas armazenada comprimida em 33 bytes.

Apesar de o tamanho das chaves pública e privada e da assinatura do WOTS$^+$ serem menores que para os esquemas pós-quânticos apresentados na Tabela 1, não é

possível utilizá-lo no blockchain, pois, cada vez que o usuário realize uma transação (assinatura) com as moedas de um endereço, parte de sua chave privada seria exposta, tornando-a vulnerável a criptanálise.

Baseando nos valores da Tabela 1, foram escolhidos o XMSS, esquema que fornece menor tamanho de assinatura e o SPHINCS+, opção com menor tamanho de assinatura entre os esquemas stateless, para o estudo do impacto da mudança do esquema de assinatura digital no funcionamento do blockchain.

A Tabela 2 compara os blockchains do Bitcoin com o PQ-BTC para esquema de assinatura utilizando SPHINCS$^+$ ou XMSS quanto aos campos da transação que terão seus valores modificados. Os valores do tamanho da transação foram calculados para seu caso típico mais simples: um endereço de origem (um *scriptSig*) e dois endereços de destino (dois *scriptPubKey*), um para receber a quantia desejada e outro para receber o restante das moedas (troco).

Tabela 2 – Comparativo das transações do Bitcoin e PQ-BTC

	Tamanho (bytes)		
Campo	**BTC (ECDSA)**	**PQ-BTC (SPHINCS$^+$)**	**PQ-BTC (XMSS)**
previous tx hash	32	64	64
scriptSig	106	8115	5004
scriptPubKey	25 (x 2)	25 (x 2)	25 (x 2)
Demais campos	29	30	30
Tamanho da transação	217	8259	5148

A modificação no tamanho da assinatura causa impacto no tamanho do *scripSig*, cujo tamanho no Bitcoin é igual a (1 + 71 + 1 + 33 = 106) bytes, sendo formado pela assinatura, chave pública e por mais dois bytes, responsáveis por informar o tamanho da assinatura e da chave pública armazenadas no *scriptSig*. Assim, o *scriptSig* do PQ-BTC, terá tamanho igual a (2 + 8080 + 1 + 32) = 8115 bytes, utilizando o SPHINCS$^+$, e (2 + 2800 + 2 + 2200) = 5004 bytes, utilizando o XMSS.

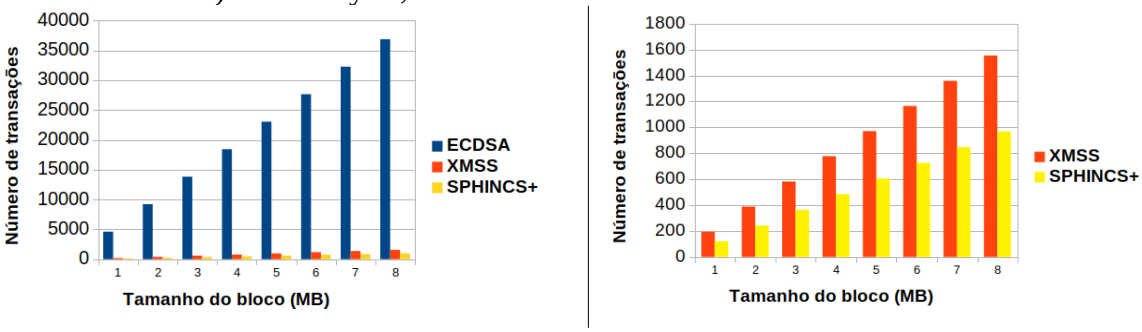

Figura 1 – Capacidade média de armazenamento de transações para diversos tamanhos de bloco para os blockchain do Bitcoin e PQ-BTC

Um novo tamanho de chave pública também influenciará no tamanho do *scriptSig*, já que este armazena a chave pública. O *scriptPubKey*, por sua vez, não será modificado, pois tanto o Bitcoin como o PQ-BTC utilizam a função RIPEMD160 para comprimir o endereço do usuário que é armazenado no *scriptPubKey*. Assim, o *scriptPubKey* de ambos terá tamanho igual a (2 + 1 + 20 + 2 = 25) bytes, onde 4 bytes são relativos às instruções do *script*, 1 byte informa o tamanho da chave pública

armazenada e os 20 bytes restantes são o hash da chave pública do usuário.

5.2. Avaliação do impacto das mudanças no tamanho de transações e de blocos

A Figura 1 mostra a capacidade média de armazenamento de transações para os blockchains do Bitcoin e PQ-BTC para diferentes tamanhos de bloco. Atualmente o Bitcoin tem bloco de tamanho máximo igual a 1MB. Discussões sobre escalabilidade e capacidade de armazenamento de dados fizeram que outras versões do Bitcoin surgissem, como o BitcoinCash [BitcoinCash 2018], que trabalha com blocos de 8MB de tamanho. O fato de o SPHINCS$^+$ possuir assinaturas de maior tamanho faz com que seja necessário um bloco de 38MB para comportar a mesma quantidade de transações por bloco que o Bitcoin, enquanto utilizando-se o XMSS, o bloco necessitaria de tamanho mínimo de 24 MB. É necessário aumentar o bloco do PQ-BTC pelo fato de este comportar menos transações em um bloco normal de 1 MB; caso contrário, os mineradores teriam que cobrar mais por transação para que o processo de mineração continuasse financeiramente atraente.

A Figura 2 traz dois gráficos com a taxa de crescimento das transações do Bitcoin e do PQ-BTC em função do número endereços de entrada e de saída. Para cada novo endereço de entrada em uma transação no Bitcoin, é necessário armazenar um *scriptSig* de 106 bytes, enquanto no PQ-BTC utilizando SPHINCS$^+$, é preciso armazenar um *scriptSig* de 8115 bytes, ou 5004 bytes, quando é feito uso do XMSS. Já para cada novo endereço de saída em uma transação, tanto o Bitcoin como o PQ-BTC, utilizando SPHINCS$^+$ ou XMSS, necessitam armazenar um *scriptPubKey* de 25 bytes.

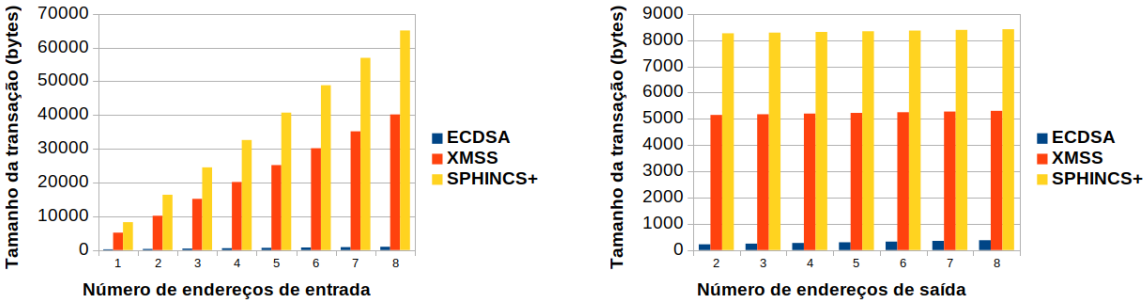

Figura 2 – Taxa de crescimento de uma transação no Bitcoin e no PQ-BTC em função do número de endereços de entrada

Tanto para o SPHINCS$^+$ como para o XMSS, a chave pública é a raiz da árvore que armazena as chaves utilizadas para assinar uma transação. Por causa disso, será custoso ao usuário trocar de endereço a cada transação, uma prática recomendável no Bitcoin, para que o usuário tenha maior anonimato na rede, já que esta troca significa trocar a árvore e todo seu conteúdo.

A troca do esquema de assinatura digital para um esquema pós-quântico vai impactar principalmente no tempo de criação de uma assinatura, conforme descrito na Tabela 3, que apresenta estes valores para o ECDSA utilizado no Bitcoin e o SPHINCS$^+$. Não foram encontradas referências a estes valores para o XMSS.

Tabela 3 – Tempos de verificação e criação de assinatura para o ECDSA e o SPHINCS$^+$

	Verificação (ms)	Criação (ms)
ECDSA[a]	**2,02**	**0,56**
SPHINCS$^+$ [b]	**0,28**	**255**

[a] [Cryptopp 2018]
[b] [Bernstein et al 2017]

Em suma, a adaptação do blockchain do Bitcoin para uma versão pós-quântica terá como gargalo os tamanhos da assinatura e da chave pública utilizadas, pois estes parâmetros influenciam no tamanho do *scriptSig* de uma transação, diminuindo a capacidade de armazenamento de transações em um bloco. Uma forma de lidar com este problema seria criar novos mecanismos de assinatura digital pós-quânticos com menor tamanho de assinatura ou modificar os esquemas de assinatura do SPHINCS$^+$ e do XMSS, através da diminuição da altura da árvore que organiza as chaves, gerando casos especiais destas assinaturas de menor tamanho e voltadas exclusivamente para operação no blockchain. Além disso seria preciso encontrar formas de tratar a questão do XMSS ser *stateful*, característica que complica ou torna inviável seu uso.

6. Conclusão

A segurança e confiabilidade de um blockchain advêm do uso de funções hash e assinatura digital. O surgimento de um computador quântico de alto desempenho pode tornar os blockchains inseguros, inviabilizando a garantia contra alteração dos dados armazenados e inviabilizando novas aplicações construídas sobre blockchains.

A fim de viabilizar a construção de novos blockchains que sejam resistentes a ataques de computadores quânticos, este trabalho apresentou um estudo preliminar sobre os impactos que as alterações nos hashes e nas assinaturas digitais teriam sobre o desempenho do novo blockchain. Dado que as assinaturas pós-quânticas têm tamanhos muito superiores aos utilizados atualmente, haveria uma significativa redução no número de transações por bloco, o que exigiria um aumento nas taxas cobradas pelos mineradores e/ou um aumento no tamanho dos blocos. A assinatura baseada em hash XMSS é a que traz o menor impacto, mas sua característica de ser *stateful* dificulta o uso em múltiplas plataformas, sendo o SPHINCS+ o esquema de assinatura *stateless* que pode ser adotado como alternativa às assinaturas ECDSA atuais, enquanto nova solução não é encontrada.

7. Trabalhos Futuros

Uma primeira frente de trabalho seria na busca de soluções para a adoção da assinatura baseada em hash XMSS, contornando os problemas trazidos pela sua característica *stateful*. Mesmo assim é possível que os custos de adoção de assinaturas HBS sejam muito altos e outros esquemas de assinaturas pós-quânticos precisem ser avaliados. Uma opção que será avaliada em trabalhos futuros é o esquema de assinatura digital NTRU [Hoffstein 1998], que oferece assinaturas menores que as HBS, mas com o mesmo nível de segurança. Esta hipótese está sendo estudada e será alvo de um futuro trabalho comparativo.

References

Aumasson, J. P., & Endignoux, G. (2018, April). Improving stateless hash-based signatures. In *Cryptographers' Track at the RSA Conference* (pp. 219-242). Springer, Cham.

Bernstein, D. J., Dobraunig, C., Eichlseder, M., Fluhrer, S., Gazdag, S., Hülsing, A., Kampanakis, P., Kölbl, S., Lange, T., Lauridsen, M.M., Mendel, F., ... (2017, November). SPHINCS+ - Submission to the NIST post-quantum project

https://sphincs.org/data/sphincs+-specification.pdf (acesso: 04/07/2018).

Bernstein, D. J., Hopwood, D., Hülsing, A., Lange, T., Niederhagen, R., ... & Wilcox-O'Hearn, Z. (2015, April). SPHINCS: practical stateless hash-based signatures. In Annual International Conference on the Theory and Applications of Cryptographic Techniques (pp. 368-397). Springer, Berlin, Heidelberg.

BitcoinCash (2018). https://www.bitcoincash.org/ (acesso: 04/09/2018).

Buchmann, J., Dahmen, E., & Hülsing, A. (2011, November). XMSS-a practical forward secure signature scheme based on minimal security assumptions. In International Workshop on Post-Quantum Cryptography (pp. 117-129). Springer, Berlin, Heidelberg.

Cryptopp (2018). https://www.cryptopp.com/benchmarks.html (acesso 11/09/2018).

Grover, L. K. (1996, July). A fast quantum mechanical algorithm for database search. In Proceedings of the twenty-eighth annual ACM symposium on Theory of computing (pp. 212-219). ACM.

Hoffstein, J., Pipher, J., & Silverman, J. H. (1998, June). NTRU: A ring-based public key cryptosystem. In *International Algorithmic Number Theory Symposium* (pp. 267-288). Springer, Berlin, Heidelberg.

Hülsing, A., (2013, June). W-OTS+–shorter signatures for hash-based signature schemes. In International Conference on Cryptology in Africa (pp. 173-188). Springer, Berlin, Heidelberg.

Lamport, L. (1979). Constructing digital signatures from a one-way function (Vol. 238). Palo Alto: Technical Report CSL-98, SRI International.

Merkle, R. C. (1979). Secrecy, authentication, and public key systems, ph.D. thesis, Electrical Engineering, Stanford.

Merkle, R. C. (1980, April). Protocols for public key cryptosystems. In *Security and Privacy, 1980 IEEE Symposium on* (pp. 122-122). IEEE.

Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system. https://bitcoin.org/bitcoin.pdf (acesso: 01/09/2018).

Post-Quantum Cryptography Standardization (2018). https://csrc.nist.gov/projects/post-quantum-cryptography (acesso: 06/09/2018).

Reyzin, L., & Reyzin, N. (2002, July). Better than BiBa: Short one-time signatures with fast signing and verifying. In Australasian Conference on Information Security and Privacy (pp. 144-153). Springer, Berlin, Heidelberg.

Shor, P. W. (1994, November). Algorithms for quantum computation: Discrete logarithms and factoring. In Foundations of Computer Science, 1994 Proceedings., 35th Annual Symposium on (pp. 124-134). Ieee.