Hash Based Digital Signature Scheme with Integrated TRNG

Maksim Iavich
Caucasus University
Tbilisi, Georgia
m.iavich@scsa.ge

Avtandil Gagnidze
Georia University
Tbilisi, Georgia
gagnidzeavto@yahoo.com

Giorgi Iashvili
Georia University
Tbilisi, Georgia
g.iashvili@scsa.ge

***Abstract*— The critical analysis of the existing hash based digital signature schemes and their improvements is done in the article. One of the improvements is integrating PRNG to Merkle crypto system. It is shown that integrating PRNG to Merke is not safe, because quantum computers are able to break PRNG-s, that were considered safe against attacks implemented on classical computers.**

TRNG based on qubits behavior is offered in the article and is offered to integrate this TRNG to Merkle. By means of the size of the signature key is reduced.

The principle of the crypto system is not changed, but only TRNG is integrated. TRNG is completely safe; it is based on the state of qubits, which are real random numbers. So the received scheme is secure.

Keywords—TRNG, hash-based, signature schemes, quantum, cryptography

I. INTRODUCTION

Scientists and cyber security experts are actively working on the development of quantum computers.

Google Corporation, Universities Space Research Association and federal agency NASA together with D-WAVE, the manufacturer of quantum processors began to work on creation of quantum processors. D-Wave 2X is a quantum processor that contains 2,048 physical qubits. Usually 1152 qubits are used to perform calculations.

Google is working on releasing the new CPU. Twenty qubit processor is already undergoing tests, and the company promises to have its working forty nine qubit chip ready by the end of this year. Until it began trialing the twenty qubit chip, Google's most powerful quantum chip was the nine qubit effort from 2015. Each additional qubit doubles the data search area, so the calculation speed is significantly

Quantum computers will be able to break most, if not absolutely all conventional cryptosystems, that are widely used in practice.

So, traditional digital signature systems that are used presently in practice are vulnerable to attacks implemented on quantum computers. The security of these systems is based on the problem of factoring large numbers and calculating discrete logarithms. Some cryptosystems for example RSA system with four thousand bit keys are considered useful to protect information from attacks implemented on classical computer, but are absolutely not protected against attacks implemented using quantum computers [1,2].

RSA cryptosystem is used in many products on different platforms and in different areas. To date, this cryptosystem is integrated into many commercial products, the number of which is growing every day. RSA system is also widely used in operating systems from Microsoft, Apple, Sun, and Novell. In hardware performance RSA algorithm is used in secure phones, Ethernet, network cards, smart cards, and is also widely used in the cryptographic hardware. Along with this, the algorithm is a part of the underlying protocols protected Internet communications, including S / MIME, SSL and S / WAN, and is also used in many organizations, for example, government, banks, most corporations, public laboratories and universities.

RSA BSAFE encryption technology is used approximately by 500 million users worldwide. Since in encryption technology is mostly used the RSA algorithm, it can be considered one of the most common public key cryptosystems being developed together with the development of the Internet.

On this basis the RSA destruction will entail easy hacking of most products that can grow into a complete chaos

Active work is being conducted to create RSA alternatives, which are protected from attacks by a quantum computer.

Hash based digital signature schemes are considered as one of the RSA alternatives. These schemes use a cryptographic hash function. The security of these digital signature schemes relies on the collision resistance of the hash functions that they use [3,4]. Lamport–Diffie one-time signature scheme Lamport–Diffie one-time signature scheme is hash based digital signature scheme, and it is considered as alternative for the post-quantum era.

A. Keys generation

The signature key X of this system consists of 2n lines of length n, and the lines are selected randomly.

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

Verification key Y of this system consists of 2n lines of length n, and the lines are selected randomly.

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

To calculate the key we use one way function f:

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

$y_i[j] = f(x_i[j]), 0<=i<=n-1, j=0,1$ (3)

B. Document signature:

Message m of arbitrary size, is transformed into size n by means of the hash function:

$h(m)=hash = (hash_{n-1}, \ldots , hash_0)$ (4)

Function h is a cryptographic hash function:

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

Signature occurs as follows:

$sig= (x_{n-1}[hash_{n-1}], \ldots, x_0[hash_0]) \in \{0,1\}^{n,n}$ (5)

If the i-th bit in the message is equal to 0, i-th string in this signature is assigned to $x_i[0]$. If the i-th bit in the message is equal to 1, i-th string in this signature is assigned to $x_i[1]$. The length of the signature is n^2.

C. Signature verification

For signature verification $sig = (sig_{n-1}, \ldots, sig_0)$, the hash of the message is calculated.

$hash = (hash_{n-1}, \ldots , hash_0)$ and the following equality must be verified:

$(f(sig_{n-1}), \ldots, f(sig_0)) = (y_{n-1}[hash_{n-1}], \ldots, y_0[hash_0])$ (6)

If it is true, then the signature is correct.

D. Winternitz one time signature scheme.

In Lamport one-time signature scheme, key generation and signature generation are quite effective, the size of the signature is very large as it is equal to n^2. Winternitz one-time signature scheme was proposed in order to decrease the signature size. In Winternitz one-time signature scheme several bits of the hashed message are signed simultaneously with one string of the key, so the length of the signature is reduced significantly [5].

E. Keys generation

The signature key X of this system consists of sn lines of length n, and the lines are selected randomly.

Winternitz parameter is selected w> = 2, it must be equal to the number of bits to be signed simultaneously.

$s_1=n/w$ and $s_2=(\log_2 s_1 +1+w)/w$ are calculated (7)

$s= s_1+ s_2$ (8)

$X= (x_{s-1}[0], \ldots, x_0) \in \{0,1\}^{n,s}$ (9)

Verification key is derived as following:

$Y= (y_{s-1}[0], \ldots, y_0) \in \{0,1\}^{n,s}$, where (10)

$y_i=f^{2^{\wedge}w-1}(x_i), 0<=i<=s-1$ (11)

The verification key and the signature are equal to ns bits.

F. Document signature

To sign the document, message must be hashed hash=h(m)

If length of hash is not devisable by w minimum number of zeros are prepended to hash and it is divided into s_1 parts of length w.

$hash=k_{s-1},\ldots, k_{s-s1}$ (12)

the checksum is calculated:

$c=\sum_{i=s-s1}^{s-1}(2^w-k_i)$ (13)

as $c<= s_1 2^w$, the length of its binary representation is

$\log_2 s_1 2^w+1$ (14)

If the length of representation is not devisable by w, the minimum number of zeros must be prepended to this binary representation and must be divided into s_2 parts of length w.

$c=k_{s2-1},\ldots, k_0$ (15)

Finally, the signature of m is calculated:

$sig=(f^{\wedge}k_{s-1}(x_{s-1}), \ldots, f^{\wedge}k_0(x_0))$ (16)

The size of the signature is sn.

G. Signature verification

bit strings $k_{s-1},\ldots, k_0$ are calculated to verify the signature

$sig = (sig_{n-1}, \ldots, sig_0)$ (17)

The following equality is checked:

$(f^{\wedge}(2^w-1-k_{s-1}))(sig_{n-1}), \ldots, (f^{\wedge}(2^w-1-k_0))(sig_0) = y_{n-1}, \ldots y_0$ (18)

If the signature is correct, then $sig_i =f^{\wedge}k_i(x_i)$ (19)

$(f^{\wedge}(2^w-1-k_i))(sig_i)= (f^{\wedge}(2^w-1))(x_i)=y_i ; i=s-1,\ldots, 0$ (20)

TABLE 1. LAMPORT AND WINTERNITZ

	Lamport	Winternitz
Key size	$2n^2$	ns
Using f for key generation	2n	$k(2^w-1)$
Signature length	n^2	ns
Using f for signature generation	Not used	$k(2^w-1)$
Using f for signature verification	n	$k(2^w-1)$

The size of the signature in Winternitz one-time signature scheme is significantly reduced $ns<n^2$, so the length of the key is significantly reduced, because: $ns<n^2<2n^2$

II. MERKLE CRYPTO-SYSTEM

To sign the message using one-time signature scheme, different key pair must be used for every message. This is very problematic, because it is very difficult to exchange big number of keys.

Merkle digital signature scheme is the solution for this problem. The scheme uses a binary tree in order to replace a large number of verification keys with one public key, that is the root of this tree [6,7]. One-time signature scheme and a cryptographic hash function are used in this system:

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

A. Key generation

The length of the tree is chosen as H>=2, using one public key 2H documents can be signed. 2H signature and verification key pairs are generated; Xi, Yi, 0<=i<=2H. Xi is the signature key, Yi is the verification key. h(Yi) are calculated and they are used as the leaves of the tree. Each tree node is a hash value of concatenation of its children.

$a[1,0]=h(a[0,0] \,||\, a[0,1])$ (21)

The root of the binary tree *pub* is the public key of the crypto scheme. 2^H pairs of one-time keys must be calculated in order to generate the public key.

B. Signature generation

Message m of arbitrary size, is transformed into size n by means of the hash function.

h (m) = hash, and is generated a one-time signature using arbitrary one-time key X_{arb}, the document's signature will be the concatenation of: one-time signature, one-time verification key Y_{arb}, index *arb* and all fraternal nodes $auth_i$ in relation to Y_{arb}.

$$Signature= (sig||arb|| Y_{arb}||auth_0,\ldots,auth_{H-1}) \quad (22)$$

C. Signature verification

To verify the signature, the one-time signature of sig is verified using Y_{arb}, if it is true, all the nodes a [i, j] are calculated using "$auth_i$", index *arb* and Y_{arb}. Finally, the root of the tree is compared with public key, if they are equal, then the signature is correct.

III. PRNG INTEGRATION TO MERKLE

2^H pairs of one-time keys must be calculated in order to generate the public key. Storing such a big number of key is problematic in practice.

To save the space, it was suggested to use the PRNG random number generator in Merkle signature scheme [8]. When using PRNG, only the seed of the generator must be stored and it must be used to generate one-time keys. In this case one-time keys must be calculated twice. The must be calculated in key generation stage and in the signature stage. PRNG uses as input the seed of length n and outputs a new seed and a random number of length n.

$PRNG : \{0, 1\}^n \rightarrow : \{0, 1\}^n \times \{0, 1\}^n$

A. Key generation using PRNG

The seed $seed_0$ of length n is chosen randomly, using $seed_i$ is generated ots_i, as following:

$PRNG(seed_i) = (ots_i , seed_{i+1}) \quad 0 \le i < 2^H$ (23)

ots_i is changed each time when PRNG launches. For X_i key calculation only $seed_i$ must be known.

Signature and verification are performed in the same way as in the standard version of Merkle crypto system.

Quantum computers are able to break PRNG-s, that were considered safe against attacks implemented on classical computers. PRNG Blum-Micali turned out to be vulnerable to polynomial quantum time attack. This PRNG is considered safe from attacks implemented on standard computers. This attack uses Grover algorithm along with the quantum discrete logarithm, and can restore the values at the generator output for this attack.

These type of attacks represent a threat of cracking PRNGs, that are used in many real-world crypto systems. Merkle crypto system with built-in PRNG can be vulnerable to attacks implemented on quantum computers. We suggest using a true random number generator based on qubits, TRNG.

To construct this TRNG states and qubits must be considered.

IV. QUANTUM TRNG

Information in a quantum computer is encoded in quantum bits or qubits. Like a bit, the qubit allows two eigenstates (conditionally | 0> and | 1>), but unlike the bit, the qubit admits a superposition of these states, which means that it is more "informative" [9-11].

In a system with one qubit, the quantum state of a qubit is denoted by:

$\alpha|0\rangle+\beta|1\rangle$

where α and β are complex numbers;

$|\alpha|^2+|\beta|^2=1$ (24)

$|0\rangle$ - is the ground state of qubit

$|1\rangle$- is the excited state of qubit

This qubit is in the state $|0\rangle$ with probability α^2 and is in a state $|1\rangle$ with probability β^2.

When a qubit is measured, it gets into one of two states with probability 1.

In the system with two qubits the quantum state of two qubits is denoted by:

$\alpha_{00}|00\rangle+\alpha_{01}|01\rangle+\alpha_{10}|10\rangle+\alpha_{11}|11\rangle$

where α_i are complex numbers;

$\sum|\alpha_i|^2=1$ (25)

These qubits are connected using Bell state [12], in this case the quantum state of these qubits is denoted by:
$1/2^{1/2}|00>+1/2^{1/2}|11>$
When a given qubit is measured,
it will be in a state |0> with probability ½ and likewise in a state |1> also with probability ½.
The second qubit is measured; it will be in the same state as the first qubit when it was measured with probability 1.
When n qubits are measured the true number of size n is got.

A. *Key generation*

The tree height H>=2 is chosen. 2^H documents are signed using one public key. A connection is established between two qubits using Bell state. 2^H pairs of such qubits are token qub_i and b_i ; every qub_i and b_i consists of n qubits. $0<=i<=2^H$; 2^H*n qubits qub_i are measured and 2^H signature keys are got X_i and the verification keys Y_i are calculated. $h(Y_i)$ are calculated and are used as leaves of a tree.

B. *Signature and verification of the message*

Message m of arbitrary size, is transformed into size n by means of the hash function.
$h(m) = hash$ (26)
Arbitrary set, that contains n qubits is measured,
$b_{arb} = qub_{arb}=X_{arb}$ with probability equal to 1.
One-time signature is generated using one-time signature key X_{arb}, the document's signature will be the concatenation of one-time signature, one-time verification key Y_{arb}, index *arb* and all fraternal nodes $auth_i$ in relation to Y_{arb}.
$Signature= (sig||arb|| Y_{arb}||auth_0,\ldots,auth_{H-1})$ (27)
Verification of the message occurs similarly to the standard Merkle system.

C. *Security*

The principle of the crypto system is not changed, but only is integrated TRNG, to reduce the size of the signature key.
TRNG is completely safe; it is based on the state of qubits, which are real random numbers.
So, the received scheme is absolutely secure.

D. *Conclusion*

2^H pairs of one-time keys must be calculated in order to generate the public key in Merkle crypto system. It is not efficient to store these numbers of keys in practice. PRNG can be integrated to Merkle in order to reduce this number. Quantum computers can break PRNGs, that are used in many real-world crypto systems. Merkle crypto system with built-in PRNG can be vulnerable to attacks implemented on quantum computers. TRNG that is offered in the paper can be integrated to Merkle. In this case the system will be safe and the numbers of one-time key will be reduced.

ACKNOWLEDGEMENTS

The Work Was Conducted as a Part of Research Grant of Joint Project of Shota Rustaveli National Science Foundation and Science & Technology Center in Ukraine [№ STCU-2016-08]

REFERENCES

[1] Bernstein D.J. (2009) Introduction to post-quantum cryptography. In: Bernstein D.J., Buchmann J., Dahmen E. (eds) Post-Quantum Cryptography. Springer, Berlin, Heidelberg

[2] Avtandil Gagnidze, Maksim Iavich, Giorgi Iashvili, SOME ASPECTS OF POST-QUANTUM CRYPTOSYSTEMS, Eurasian Journal of Business and Management, 5(1), 2017, 16-20 DOI: 10.15604/ejbm.2017.05.01.002

[3] Dods C., Smart N.P., Stam M. (2005) Hash Based Digital Signature Schemes. In: Smart N.P. (eds) Cryptography and Coding. Cryptography and Coding 2005. Lecture Notes in Computer Science, vol 3796. Springer, Berlin, Heidelberg

[4] Dahmen E., Krauß C. (2009) Short Hash-Based Signatures for Wireless Sensor Networks. In: Garay J.A., Miyaji A., Otsuka A. (eds) Cryptology and Network Security. CANS 2009. Lecture Notes in Computer Science, vol 5888. Springer, Berlin, Heidelberg

[5] A.Gagnidze, M.Iavich, N.Inasaridze, G.Iashvili Analysis of one-time signature schemes // Scientific & practical cyber security journal (SPCSJ) № 1.[Electronic journal]. URL: http://journal.scsa.ge/issues/2017/09/455

[6] Szydlo, M.: Merkle tree traversal in log space and time. In: Cachin, C., Camenisch, J.L. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 541–554. Springer, Heidelberg (2004)

[7] Dods, C., Smart, N., Stam, M.: Hash based digital signature schemes. In: Smart, N.P. (ed.) Cryptography and Coding 2005. LNCS, vol. 3796, pp. 96–115. Springer, Heidelberg (2005)

[8] Buchmann J., García L.C.C., Dahmen E., Döring M., Klintsevich E. (2006) CMSS – An Improved Merkle Signature Scheme. In: Barua R., Lange T. (eds) Progress in Cryptology - INDOCRYPT 2006. INDOCRYPT 2006. Lecture Notes in Computer Science, vol 4329. Springer, Berlin, Heidelberg

[9] Daniel F. V. James, Paul G. Kwiat, William J. Munro, and Andrew G. White Asymptotic Theory of Quantum Statistical Inference. February 2005, 509-538

[10] U. Leonhardt, Measuring the quantum state of light (Cambridge University Press, 1997)

[11] A. G. White, D. F. V. James, W. J. Munro and P. G. Kwiat, "Exploring Hilbert Space: accurate characterization of quantum information," submitted to Science (2001)

[12] Deng, FG., Li, XH., Li, CY. et al., Quantum state sharing of an arbitrary two-qubit state with two-photon entanglements and Bell-state measurements, PHYSICAL JOURNAL D, (2006) 39: 459. https://doi.org/10.1140/epjd/e2006-00124-1

[13] Wozniak, M., Polap, D., Borowik, G. and Napoli, C., "A first attempt to cloud-based user verification in distributed system." in Asia-Pacific Conference on Computer Aided System Engineering (APCASE), 2015, pp. 226-231.