SECURITY STRENGTH AND PERFORMANCE COMPARISON OF W-OTS AND W-OTS+ ONE-TIME DIGITAL SIGNATURES

Nguyen Van Nghi*, Pham Thi Hien, Dinh Van Hung, Le Thi Bich Hang
Academy of Cryptography Techniques

ARTICLE INFO

Received: 10/8/2023
Revised: 17/10/2023
Published: 17/10/2023

KEYWORDS

One Time Signature
Hash Function
W-OTS
W-OTS+
Security Strength
Performance

ABSTRACT

With the rapid development of quantum computers, the Shor algorithm can break all popular digital signature schemes such as RSA, ECDSA, and EdDSA in polynomial time. Therefore, post-quantum digital signature schemes are being developed to resist this attack. Digital signature schema based on the cryptographic hash function is one of the post-quantum digital signature scheme that are interested in quantity and have many published works. Two one-time digital signature schemes, W-OTS and W-OTS+, are devised and serve as the foundation for the subsequent development of more advanced post-quantum digital signature schemes, such as XMSS or SPHINCS+. In this paper, we analyze and compare the security and experimental installation performance of two digital signature schemes, W-OTS and W-OTS+, using a research method synthesized from published documents and experiments implemented with Python programming language. We provide our own commentary in addition to citations from other works on the security of these two schemes. The conclusions about the security of these two schemes in the article are cited from other works and also concluded that our own contributions.

SO SÁNH ĐỘ AN TOÀN VÀ HIỆU NĂNG CỦA LƯỢC ĐỒ CHỮ KÝ SỐ MỘT LẦN W-OTS VÀ W-OTS+

Nguyễn Văn Nghị*, Phạm Thị Hiên, Đinh Văn Hùng, Lê Thị Bích Hằng
Học viện Kỹ thuật mật mã

THÔNG TIN BÀI BÁO

Ngày nhận bài: 10/8/2023
Ngày hoàn thiện: 17/10/2023
Ngày đăng: 17/10/2023

TỪ KHÓA

Lược đồ ký số một lần
Hàm băm mật mã
WOTS
WOTS+
Độ an toàn
Hiệu năng

TÓM TẮT

Với sự phát triển nhanh chóng của máy tính lượng tử thì thuật toán Shor có thể phá vỡ hết các lược đồ chữ ký số phổ biến hiện nay như RSA, ECDSA, EdDSA trong thời gian đa thức. Chính vì vậy, các lược đồ chữ ký số hậu lượng tử được phát triển nhằm kháng lại tấn công này. Lược đồ chữ ký số dựa trên hàm băm mật mã là một trong các lược đồ chữ ký số hậu lượng tử đang được quan tâm và có nhiều công trình công bố. Trong đó hai lược đồ chữ ký số một lần W-OTS và W-OTS+ được phát triển và là nền tảng để xây dựng các lược đồ chữ ký số hậu lượng tử tân tiến hơn sau này như XMSS hay SPHINCS+. Mục tiêu của bài báo này đi phân tích, so sánh độ an toàn và hiệu năng cài đặt thực nghiệm của hai lược đồ chữ ký số W-OTS và W-OTS+ dựa trên phương pháp nghiên cứu là tổng hợp từ các tài liệu công bố và cài đặt thực nghiệm bằng ngôn ngữ lập trình Python. Các kết luận về sự an toàn của hai lược đồ này trong bài báo được trích dẫn từ các công trình khác và cũng có kết luận là đóng góp của riêng chúng tôi.

DOI: https://doi.org/10.34238/tnu-jst.8519

* Corresponding author. *Email: nghinv@actvn.edu.vn*

1. Giới thiệu

Chữ ký số hậu lượng tử được thiết kế để đảm bảo sự an toàn trước tấn công của máy tính lượng tử. Thuật toán của Shor chạy trên máy tính lượng tử đã được công bố và chứng minh rằng có thể giải các bài toán phân tích số nguyên và logarit rời rạc trên trường hữu hạn và trên đường cong elliptic trong thời gian đa thức [1]. Điều này có nghĩa là sự an toàn của nhiều lược đồ chữ ký hiện nay như RSA [2], ECDSA [3], EdDSA [4], [5], sẽ bị xâm phạm khi các máy tính lượng tử thực tế được xây dựng. Do đó, việc phát triển mật mã hậu lượng tử nói chung và chữ ký số hậu lượng tử nói riêng để chống lại sự tấn công của máy tính lượng tử, ngày càng trở nên quan trọng để đảm bảo sự an toàn cho các quá trình truyền thông.

Lược đồ chữ ký số hậu lượng tử được thiết kế để đảm bảo an toàn ngay cả các tấn công được thực hiện bằng máy tính lượng tử bao gồm cả thuật toán Shor. Các bài toán được cho là khó giải đối với cả máy tính lượng tử và máy tính truyền thống, chẳng hạn như bài toán tìm các vector ngắn trong lưới hoặc bài toán dựa trên đa thức đa biến. Một số loại lược đồ chữ ký hậu lượng tử, dựa trên cách tiếp cận toán học như trên có thể kể đến: Chữ ký dựa trên hàm băm (Hash based signatures), chữ ký dựa trên lưới (Lattice based signatures), chữ ký dựa trên mã (Code based signatures), chữ ký dựa trên đa thức nhiều biến (Multivariate polynomial based signatures), chữ ký dựa trên đường cong kỳ dị (Isogeny based signatures) [1], [6], [7].

Các nghiên cứu hiện tại cho thấy rằng chưa có thuật toán nào tối ưu trên máy tính lượng tử có thể phá vỡ được lược đồ chữ ký số dựa trên hàm băm mật mã trong thời gian đa thức. Một lược đồ chữ ký dựa trên hàm băm hoặc lược đồ chữ ký số Merkle (MSS) bao gồm sự kết hợp của lược đồ chữ ký số một lần để ký dữ liệu và lược đồ xác thực cây Merkle [1], [6], [7].

Lược đồ chữ ký số một lần (One Time Signature - OTS) là một lược đồ chữ ký số với đặc trưng là mỗi cặp khóa chỉ có thể sử dụng một lần duy nhất để ký cho một thông điệp [1], [6]-[8]. Ưu điểm chính của lược đồ chữ ký số một lần là chúng có quá trình ký và kiểm tra chữ ký số rất nhanh so với các lược đồ chữ ký số khác như RSA, ECDSA, EdDSA. Tuy nhiên, lược đồ chữ ký số một lần cũng có một số hạn chế nhất định như: giới hạn về số chữ ký có để được ký, kích thước chữ ký và kích thước của các khóa khá lớn. Một số lược đồ OTS như: Lamport-Diffie OTS, Merkle OTS, the Winternitz OTS, the Bleichenbacher-Maurer OTS, BiBa OTS và HORS [6]. Lược đồ Winternitz OTS (W-OTS) được chứng minh là phù hợp nhất để kết hợp nó với lược đồ xác thực cây Merkle, vì kích thước xác minh nhỏ và có sự linh hoạt giữa kích thước chữ ký và thời gian tạo chữ ký [6].

Lược đồ chữ ký số một lần Lamport (Lamport-OTS) được thiết kế vào năm 1979 bởi Leslie Lamport, là lược đồ chữ ký số đầu tiên dựa trên hàm băm [6], [7]. Quá trình sinh khóa và quá trình ký của lược đồ chữ ký số Lamport-OTS được xem là rất hiệu quả, tuy nhiên kích thước chữ ký lại khá lớn. Lược đồ chữ ký số một lần Winternitz (W-OTS) có kích thước chữ ký ngắn hơn đáng kể so với kích thước chữ ký trong Lamport-OTS. Ý tưởng là sử dụng một chuỗi bit trong khóa bí mật để đồng thời ký một số bit của thông điệp. Lược đồ chữ ký số W-OTS+ được Andreas Hulsing đề xuất vào năm 2013. Lược đồ W-OTS+ đạt được mức an toàn cao hơn so với lược đồ chữ ký số W-OTS [9].

Trong bài báo này chúng tôi tiến hành phân tích độ an toàn của hai lược đồ chữ ký số W-OTS và W-OTS+. Các nhận xét về sự an toàn của hai lược đồ này trong bài báo được trích dẫn từ các công trình khác và cũng có nhận xét của riêng chúng tôi. Trong lược đồ chữ ký số W-OTS+ sử dụng các hàm chuỗi thay cho các họ hàm thông thường trong W-OTS [6] – [11], chúng tôi nhận định điều này sẽ làm cho thời gian thực thi của lược đồ W-OTS+ chậm hơn so với W-OTS. Để so sánh thời gian thực thi giữa hai thuật toán này chúng tôi đã thực hiện cài đặt và so sánh hiệu suất thực thi của hai thuật toán này bằng ngôn ngữ lập trình Python. Điều này sẽ là cơ sở để xem xét lựa chọn một trong hai lược đồ ký số này làm nền tảng để ứng dụng phát triển hoặc cải tiến các lược đồ ký số mới dựa trên hàm băm mật mã.

Bài báo có bố cục như sau: Sau phần 1 giới thiệu, bài báo sẽ mô tả chi tiết hai lược đồ chữ ký số W-OTS và W-OTS+ trong phần 2. Phần 3 là các nhận xét và so sánh độ an toàn của hai lược đồ chữ ký số này từ các công trình đã công bố và một số nhận xét riêng của chúng tôi. Và cũng trong mục 3 thì chúng tôi có so sánh hiệu suất thực thi của hai lược đồ này dựa trên việc cài đặt thực nghiệm bằng ngôn ngữ lập trình Python trên cùng một nền tảng phần cứng. Cuối cùng là phần kết luận và các tài liệu tham khảo của bài báo.

2. Vấn đề nghiên cứu

2.1. Lược đồ chữ ký số một lần W-OTS

Theo [8], lược đồ chữ ký số W-OTS sử dụng các tham số sau: tham số an toàn $n \in N$, độ dài thông điệp m, tham số Winternitz $w \in N, w \geq 2$ và

$$t_1 = \left\lceil \frac{n}{w} \right\rceil, t_2 = \left\lfloor \frac{\lfloor log t_1 + 1 + w \rfloor}{w} \right\rfloor, t = t_1 + t_2.$$

Bên cạnh đó, lược đồ W-OTS sử dụng một họ hàm

$$F_n := \{f_k : \{0,1\}^n \to \{0,1\}^n | k \in \{0,1\}^n\}.$$

Với $x \in \{0,1\}^n, i \in N$, định nghĩa $f_k^i(x) = f\left(f_k^{i-1}(x)\right)$.

Các tham số n, m, t_1, t_2, t, w và họ hàm F_n là được công khai và có ý nghĩa như sau:

n là số bit của dữ liệu đầu vào và đầu ra họ hàm băm không nén F_n, m là độ dài theo bit của thông điệp đầu vào cần ký, tham số w (có thể bằng 2, 4, 8, 16) là cơ sở số mà thông điệp đầu vào được trình diễn (ví dụ $w = 2$ thì thông điệp được trình diễn dưới dạng cơ sở số nhị phân), tham số $t = t_1 + t_2$ để xác định chỉ số phần tử trong chuỗi số mà thông điệp đầu vào biểu diễn dưới dạng cơ sở w. Lược đồ ký số W-OTS bao gồm ba quá trình là sinh khóa, ký và kiểm tra chữ ký.

2.1.1. Quá trình sinh khóa

Đầu vào: Các tham số n, t, w.

Đầu ra: Khóa bí mật $\boldsymbol{sk}$ và khóa công khai $\boldsymbol{pk}$.

1) Chọn $sk = (sk_{t-1}, \dots, sk_1, sk_0)$ là t chuỗi ngẫu nhiên, trong đó mỗi chuỗi $sk_i, 0 \leq i \leq t-1$ có độ dài n bit.

2) Chọn ngẫu nhiên đều một khóa $k \leftarrow_{\$} \{0,1\}^n$.

3) Khóa $pk = (pk_{t-1}, \dots, pk_1, pk_0)$, trong đó $pk_i = f_k^{2^w-1}(sk_i), 0 \leq i \leq t-1$.

4) Đưa ra khóa bí mật $\boldsymbol{sk}$ và khóa công khai $(k, \boldsymbol{pk})$.

2.1.2. Quá trình ký

Đầu vào: Các tham số n, m, t_1, t_2, t, w, thông điệp M có độ dài m bit và khóa bí mật sk.

Đầu ra: Chữ ký cho thông điệp M.

1) Ký hiệu $M = (M_{m-1}, \dots, M_0)$ là biểu diễn bit của thông điệp M.

2) Thêm $t_1 w - m$ bit "0" vào phía trước chuỗi bit của thông điệp M để độ dài của chuỗi mở rộng của thông điệp M chia hết cho w. Viết lại

$$M = \left(\underbrace{0 \dots 0}_{t_1 w - m} || M_{m-1} || \dots || M_0 \right) = (b_{t-1} || \dots || b_{t-t_1}), \text{ trong đó mỗi } b_i, (t - t_1 \leq i \leq t-1)$$

có kích thước w bit.

3) Các chuỗi b_i được đồng nhất với các số nguyên trong $\{0,1,\dots,2^w-1\}$.

4) Tính $C = \sum_{i=t-t_1}^{t-1} (2^w - b_i)$. Vì $C \leq t_1 2^w$, nên độ dài biểu diễn nhị phân của C là nhỏ hơn $\lfloor \log(t_1 2^w) \rfloor + 1 = \lfloor \log t_1 \rfloor + w + 1$.

5) Thêm một lượng tối thiểu bit "0" và phía trước biểu diễn nhị phân của C để độ dài của chuỗi mở rộng của C là $t_2 w$ bit. Chuỗi mở rộng được chia thành t_2 khối $b_{t_2-1}, \dots, b_0$ mỗi khối có kích thước w bit. Viết lại $C = (b_{t_2-1} || \dots || b_0)$.

6) Đặt $B = (M||C) = (b_{t-1}|| \dots ||b_{t-t_1}||b_{t_2-1}|| \dots ||b_0)$.

7) Tính $\sigma = (\sigma_{t-1}, \dots, \sigma_0) = \left(f_k^{b_{t-1}}(sk_{t-1}), \dots, f_k^{b_1}(sk_1), f_k^{b_0}(sk_0)\right)$.

8) Đưa ra σ là chữ ký cho thông điệp M.

2.1.3. Quá trình kiểm tra chữ ký

Đầu vào: Các tham số n, t, t_1, t_2, w, thông điệp M, khóa công khai $(k, \boldsymbol{pk})$ và chữ ký σ.

Đầu ra: Chữ ký là hợp lệ hoặc chữ ký là không hợp lệ.

1) Tính chuỗi bit $B = (b_{t-1}|| \dots ||b_{t-t_1}||b_{t_2-1}|| \dots ||b_0$ như trong quá trình ký.

2) Nếu $\left(f_k^{2^w-1-b_{t-1}}(\sigma_{t-1}), \dots, f_k^{2^w-1-b_0}(\sigma_0)\right) = (pk_{t-1}, \dots, pk_0)$ thì chữ ký hợp lệ. Nếu không thì hủy bỏ chữ ký.

2.1.4. Tính đúng đắn của lược đồ chữ ký số W-OTS

Nếu σ là chữ ký được sinh ra từ quá trình ký thì:

$$f_k^{2^w-1-b_i}(\sigma_i) = f_k^{2^w-1-b_i}\left(f_k^{b_i}(sk_i)\right) = f_k^{2^w-1}(sk_i) = pk_i, \forall 0 \le i \le t-1.$$

Quá trình sinh khóa của lược đồ chữ ký số W-OTS yêu cầu $t(2^w - 1)$ phép tính sử dụng hàm f_k. Trong trường hợp xấu nhất, các quá trình ký và kiểm tra chữ ký yêu cầu $t(2^w - 1)$ phép tính sử dụng hàm f_k. Khóa bí mật và chữ ký có kích thước là tn bit. Kích thước của khóa công khai là $(t + 1)n$ bit.

2.2. Lược đồ chữ ký số một lần W-OTS+

Theo [9], lược đồ chữ ký số W-OTS+ sử dụng các tham số gồm: tham số an toàn $n \in N$, độ dài thông điệm m, tham số Winternitz $w \in N, w \ge 2$ và

$$l_1 = \left\lceil \frac{m}{\log(w)} \right\rceil, l_2 = \left\lfloor \frac{\log(l_1(w-1))}{\log(w)} \right\rfloor + 1, l = l_1 + l_2.$$

Bên cạnh đó, lược đồ W-OTS+ sử dụng một họ hàm

$$F_n := \{f_k : \{0,1\}^n \to \{0,1\}^n | k \in \{0,1\}^n\}.$$

Bằng việc sử dụng F_n, ta định nghĩa chuỗi hàm $c_k^i(x, r)$ như sau: chuỗi hàm $c_k^i(x, r)$ có đầu vào là giá trị $x \in \{0,1\}^n$, bộ đếm số lần lặp $i \in N$, khóa $k \in \{0,1\}^n$ và các phần tử ngẫu nhiên $r = (r_1, \dots, r_j) \in \{0,1\}^{n \times j}$ với $j \ge i$. Trong trường hợp $i = 0$ thì $c_k^0(x, r) = x$, với $i > 0$ chuỗi hàm $c_k^i(x, r)$ được định nghĩa đệ quy như sau:

$$c_k^i(x, r) = f_k\left(c_k^{i-1}(x, r) \oplus r_i\right).$$

Ta định nghĩa $r_{a,b} := (r_a, \dots, r_b)$ là tập con của r. Trong trường hợp $a > b$ thì $r_{a,b}$ là chuỗi rỗng. Các tham số n, m, w, l_1, l_2, l, họ hàm F_n và chuỗi hàm $c_k^i(x, r)$ được công khai có ý nghĩa như sau: và có ý nghĩa như sau: n là số bit của dữ liệu đầu vào và đầu ra họ hàm băm không nén F_n, m là độ dài theo bit của thông điệp đầu vào cần ký, tham số w (có thể bằng 2, 4, 8, 16) là cơ sở số mà thông điệp đầu vào được trình diễn (ví dụ $w = 2$ thì thông điệp được trình diễn dưới dạng cơ sở số nhị phân), tham số $l = l_1 + l_2$ để xác định chỉ số phần tử trong chuỗi số mà thông điệp đầu vào biểu diễn dưới dạng cơ sở w. Giống như W-OTS thì lược đồ ký số W-OTS+ bao gồm ba quá trình là sinh khóa, ký số và kiểm tra chữ ký số.

2.2.1. Quá trình sinh khóa

Đầu vào: Các tham số (n, w, l).

Đầu ra: Khóa bí mật sk và khóa công khai pk.

1) Chọn $l + w - 1$ chuỗi ngẫu nhiên n bit. Đặt l chuỗi đầu là khóa bí mật $sk = (sk_1, \dots, sk_l)$ và $w - 1$ chuỗi bit còn lại là $r = (r_1, \dots, r_{w-1})$.

2) Chọn một khóa $k \leftarrow_{\$} \{0,1\}^n$.

3) Tính $pk = (pk_1, \dots, pk_l) = \left(c_k^{w-1}(sk_1, r), \dots, c_k^{w-1}(sk_l, r)\right)$.

4) Đưa ra khóa bí mật là $sk = (sk_1, \dots, sk_l)$ và khóa công khai là (r, k, pk).

2.2.2. Quá trình ký số

Đầu vào: Các tham số n, w, l_1, l_2, l, thông điệp M có độ dài m bit, khóa bí mật sk và chuỗi ngẫu nhiên r.

Đầu ra: Chữ ký cho thông điệp M.

1) Biểu diễn thông điệp M theo cơ số w. Ký hiệu, $M = \left(M_1, \dots, M_{l_1}\right)$, trong đó $M = M_1 w^{l_1-1} + \dots + M_{l_1-1} w + M_{l_1}$ và $M_i \in \{0, \dots, w-1\}$ với $1 \le i \le l_1$.

2) Tính tổng kiểm tra $C = \sum_{i=1}^{l_1}(w - 1 - M_i)$.

3) Biểu diễn C theo cơ số w. Ký hiệu, $C = \left(C_1, \dots, C_{l_2}\right)$, trong đó $C = C_1 w^{l_2-1} + \dots + C_{l_2-1} w + C_{l_2}$ và $C_i \in \{0, \dots, w-1\}$ với $1 \le i \le l_2$ (Vì $C \le l_1(w-1)$ nên độ dài biểu diễn của C theo cơ số w tối đa là $\lfloor log_w(l_1(w-1)) \rfloor + 1 = \left\lfloor \frac{\log(l_1(w-1))}{\log(w)} \right\rfloor + 1 = l_2$).

4) Đặt $B = (b_1, \dots, b_l) = M \parallel C$ là sự ghép nối các biểu diễn theo cơ số *w* của *M* và *C*.

5) Tính $\sigma = (\sigma_1, \dots, \sigma_l) = \left(c_k^{b_1}(sk_1, r), \dots, c_k^{b_l}(sk_l, r)\right)$, trong đó $\sigma_i = c_k^{b_i}(sk_i, r)$ với $1 \le i \le l$.

6) Đưa ra σ là chữ ký của thông điệp M.

Lưu ý rằng giá trị tổng kiểm tra đảm bảo cho b_i, $0 < i \le \ell$ tương ứng với một thông báo, b'_i tương ứng với bất kỳ thông báo nào khác bao gồm ít nhất một $b'_i < b_i$.

2.2.3. Quá trình kiểm tra chữ ký số

Đầu vào: Các tham số n, w, l, l_1, l_2, thông điệp M có kích thước m bit, chữ ký σ và khóa công khai (r, k, pk).

Đầu ra: Chữ ký hợp lệ hoặc không hợp lệ.

1) Tính chuỗi $B = (b_1, \dots, b_l) = M \parallel C$ như trong quá trình ký.

2) Nếu $\left(c_k^{w-1-b_1}(\sigma_1, r_{b_1+1,w-1}), \dots, c_k^{w-1-b_l}(\sigma_l, r_{b_l+1,w-1})\right) = (pk_1, \dots, pk_l)$ thì Chữ ký hợp lệ, ngược lại Chữ ký không hợp lệ.

2.2.4. Tính đúng đắn của lược đồ chữ ký số W-OTS+

Để dễ dàng, ta có thể giả sử $b_i = 1$, khi đó cần chứng minh

$$c_k^{w-2}(\sigma_i, r_{2,w-1}) = c_k^{w-2}(c_k^1(sk_i, r), r_{2,w-1}) = c_k^{w-1}(sk_i, r) \qquad (*)$$

Xét $w = 2$, có

$$\begin{aligned} c_k^{w-2}(\sigma_i, r_{2,w-1}) &= c_k^{2-2}(\sigma_i, r_{2,2-1}) \\ &= c_k^0(c_k^1(sk_i, r), r_{2,1}) \\ &= c_k^0(c_k^1(sk_i, r), \emptyset) \\ &= c_k^1(sk_i, r) \end{aligned}$$

Xét $w = 3$, có

$$\begin{aligned} c_k^{w-2}(\sigma_i, r_{2,w-1}) = c_k^{3-2}(\sigma_i, r_{2,3-1}) &= c_k^1(\sigma_i, r_{2,2}) \\ = f_k(c_k^0(\sigma_i, r_2) \oplus r_2) &= f_k(\sigma_i \oplus r_2) \\ = f_k(c_k^1(sk_i, r) \oplus r_2) &= c_k^2(sk_i, r) \end{aligned}$$

Giả sử (*) đúng với $w = n - 1$, tức là có:

$$c_k^{(n-1)-2}(\sigma_i, r_{2,(n-1)-1}) = c_k^{(n-1)-2}(c_k^1(sk_i, r), r_{2,(n-1)-1}) = c_k^{(n-1)-1}(sk_i, r)$$

Ta sẽ cần chứng minh (*) đúng với $w = n$. Thật vậy, ta có:

$$c_k^{n-2}(\sigma_i, r_{2,n-1}) = f_k\left(c_k^{n-3}(\sigma_i, r_{2,n-1}) \oplus r_{n-1}\right)$$

Do $c_k^{n-3}(\sigma_i, r_{2,n-1}) = c_k^{n-3}(\sigma_i, r_{2,n-2}) = c_k^{n-2}(sk_i, r)$. Suy ra

$$c_k^{n-2}(\sigma_i, r_{2,n-1}) = f_k\left(c_k^{n-3}(\sigma_i, r_{2,n-1}) \oplus r_{n-1}\right)$$
$$= f_k\left(c_k^{n-2}(sk_i, r) \oplus r_{n-1}\right) = c_k^{n-1}(sk_i, r)$$

Tổng quát, ta có thể chứng minh

$$c_k^{w-1-b_1}(\sigma_i, r_{b_1+1,w-1}) = c_k^{w-1-b_1}\left(c_k^{b_1}(sk_i, r), r_{b_1+1,w-1}\right) = c_k^{w-1}(sk_i, r)$$

Do vậy, với $1 \le i \le l$, ta có

$$c_k^{w-1-b_i}(\sigma_i, r_{b_i+1,w-1}) = sk_i$$

Quá trình sinh khóa của lược đồ chữ ký số W-OTS+ yêu cầu $l(w-1)$ phép tính sử dụng hàm f_k. Trong trường hợp xấu nhất, các quá trình ký và kiểm tra chữ ký yêu cầu $l(w-1)$ phép tính sử dụng hàm f_k. Kích thước của khóa bí mật và kích thước chữ ký là ln bit. Khóa công khai có kích thước là $(l+w)n$ bit.

3. Kết quả và bàn luận

Trong phần này, chúng tôi đưa ra các đánh giá về độ an toàn của hai lược đồ ký số một lần W-OTS và W-OTS+. Các đánh giá này được tổng hợp từ các công bố [7] – [11] và cũng có các đánh giá riêng của chúng tôi cho sự an toàn của hai lược ký số này trong trường hợp tham số $w = 2$. Phần cuối của mục này là kết quả so sánh hiệu năng cài đặt thực nghiệm của hai lược đồ W-OTS và W-OTS+ bằng ngôn ngữ lập trình python trên cùng nền tảng phần cứng.

3.1. Một số kết luận về sự an toàn của lược đồ chữ ký W-OTS và W-OTS+

Đánh giá về sự an toàn của lược đồ chữ ký số W-OTS

1) Lược đồ chữ ký số W-OTS là an toàn EU-CMA, tức là không thể giả mạo chữ ký dưới các tấn công thông báo lựa chọn thích nghi, nếu F_n là họ hàm một chiều, không thể phát hiện (Một họ hàm F_n được gọi là không thể phát hiện nếu không có kẻ tấn công nào trong thời gian đa thức có thể phân biệt được một giá trị được chọn ngẫu nhiên đầu trong miền giá trị của F_n với một giá trị được tính qua họ hàm F_n với xác suất đáng kể) và kháng va chạm [7], [8].

2) Do tấn công ngày sinh luôn có thể tìm được va chạm trong $2^{n/2}$ bước nên lược đồ chữ ký số W-OTS chỉ đạt được mức an toàn $n/2$ bit [7], [8].

3) Trong lược đồ chữ ký số W-OTS mỗi khóa bí mật chỉ được sử dụng một lần. Việc sử dụng lại khóa bí mật có thể giúp kẻ tấn công đưa ra được chữ ký giả mạo hợp lệ. Ngoài ra, cần phải chọn tham số an toàn n thích hợp [7], [8].

- Giả sử, với $w = 2$ người ký sử dụng khóa bí mật $sk = (sk_3, \dots sk_0)$ để ký hai thông điệp có biểu diễn nhị phân: $M_1 = (1,0,0)$ và $M_2 = (1,1,1)$. Khi đó ta có hai chữ ký tương ứng là $\sigma_1 = \left(f(sk_3), sk_2, f(sk_1), f^3(sk_0)\right)$ và $\sigma_2 = (f(sk_3), f^3(sk_2), f(sk_1), sk_0)$ được sinh ra từ cùng một khóa bí mật. Việc biết hai chữ ký này giúp kẻ tấn công có thể đưa ra chữ ký hợp lệ cho thông điệp $M_3 = (1,1,0)$ mà không cần biết chính xác khóa bí mật. Thật vậy, chữ ký hợp lệ của thông điệp M_3 sẽ có dạng $\sigma_3 = \left(f(sk_3), f^2(sk_2), f(sk_1), f(sk_0)\right)$. Khi đó, kẻ tấn công sẽ tính được $f(sk_3)$ do biết $f(sk_3)$ trong thành phần của chữ ký σ_1, tính được $f^2(sk_2)$ do kẻ tấn công biết được giá trị sk_2 trong chữ ký σ_1, tính được $f(sk_1)$ do biết $f(sk_1)$ cũng là thành phần trong chữ ký σ_1 và tính được $f(sk_0)$ do kẻ tấn công biết được giá trị sk_0 trong σ_2.

4) Các tham số w làm cho W-OTS rất linh hoạt. Nó cho phép cân bằng giữa độ dài của một chữ ký và cặp khóa. Nếu w được tăng lên, nhiều bit của giá trị băm không được xử lý đồng thời và độ dài chữ ký giảm. Tuy nhiên, có nhiều đánh giá của hàm một chiều f được yêu cầu trong quá trình sinh khóa và tạo chữ ký. Giảm w có tác dụng ngược lại. Việc sử dụng $w = 2$ đòi hỏi số lượng ít nhất đánh giá của hàm một chiều f cho mỗi bit. Đây là đánh giá được đưa ra bởi chúng tôi khi áp dụng $w = 2$ cho lược đồ W-OTS.

Nhận xét về sự an toàn của lược đồ chữ ký số W-OTS+

1) Lược đồ W-OTS+ được chứng minh là an toàn EU-CMA nếu họ hàm F_n là họ hàm một chiều, không thể phát hiện và kháng tiền ảnh thứ hai [9]-[11].

2) Lược đồ chữ ký số W-OTS+ đạt mức an toàn $n - \log(w^2 l + w)$ bit [9].

3) Trong lược đồ chữ ký số W-OTS+ mỗi khóa bí mật chỉ được sử dụng một lần. Việc sử dụng lại khóa bí mật có thể giúp kẻ tấn công đưa ra được chữ ký giả mạo hợp lệ. Nhận định này cũng được chúng tôi đưa ra khi xem xét áp dụng $w = 2, m = 3, r = r_1$ cho lược đồ W-OTS+.

- Trường hợp $w = 2, m = 3, r = r_1$, người ký ký hai thông điệp có biểu diễn nhị phân là $M_1 = (0,0,1)$ và $M_2 = (1,0,0)$ với cùng một khóa bí mật $sk = (sk_1, \dots sk_5)$. Khi đó chữ ký tương ứng $\sigma_1 = (sk_1, sk_2, c_k(sk_3, r_1), c_k(sk_4, r_1), sk_5)$ và $\sigma_2 = (c_k(sk_1, r_1), sk_2, sk_3, c_k(sk_4, r_1), sk_5)$. Việc biết hai chữ ký này giúp kẻ tấn công có thể đưa ra chữ ký hợp lệ cho thông điệp $M_3 = (0,1,0)$ mà không cần biết chính xác khóa bí mật. Thật vậy, chữ ký hợp lệ của thông điệp M_3 sẽ có dạng $\sigma_3 = (sk_1, c_k(sk_2, r_1), sk_3, c_k(sk_4, r_1), sk_5)$, trong đó kẻ tấn công biết và tính được $sk_1, c_k(sk_2, r_1), c_k(sk_4, r_1), sk_5$ từ chữ ký σ_1 và tính được sk_3 từ chữ ký σ_2.

3.2. So sánh độ an toàn của lược đồ chữ ký W-OTS và W-OTS+

Bảng 1 đưa ra so sánh số phép toán sử dụng hàm f trong các quá trình sinh khóa, ký, kiểm tra chữ ký và so sánh kích thước khóa bí mật, khóa công khai, chữ ký trong khi ký một thông điệp có kích thước m bit [8], [9].

Bảng 1. *So sánh giữa W-OTS và W-OTS+ với cùng kích thước đầu vào*

Lược đồ	Số phép tính sử dụng hàm f			Kích thước khóa bí mật	Kích thước khóa công khai	Kích thước chữ ký	Mức an toàn
	Quá trình sinh khóa	Quá trình ký	Quá trình kiểm tra chữ ký				
W-OTS	$t(2^w - 1)$	$t(2^w - 1)$	$t(2^w - 1)$	tn	$(t+1)n$	tn	$n/2$
W-OTS+	$l(w-1)$	$l(w-1)$	$l(w-1)$	ln	$(l+w)n$	ln	$n - \log(w^2 + w)$

Từ Bảng 1 thì thấy rằng:

- Lược đồ chữ ký W-OTS có kích thước chữ ký là tn bit sẽ nhỏ hơn hoặc bằng với kích thước chữ ký của W-OTS+ ln bit. Vì ta luôn có $t = t_1 + t_2$ (với $t_1 = \left\lceil \frac{n}{w} \right\rceil, t_2 = \left\lceil \frac{\lfloor log t_1 + 1 + w \rfloor}{w} \right\rceil$) nhỏ hơn $l = l_1 + l_2$ (với $l_1 = \left\lceil \frac{m}{\log(w)} \right\rceil, l_2 = \left\lfloor \frac{\log(l_1(w-1))}{\log(w)} \right\rfloor + 1$)) trường hợp sử dụng các tham số $w = 4, 8, 16$. Với $w = 2$ thì $t = l$.

- Độ an toàn của lược đồ W-OTS+ là $n - \log(w^2 + w)$ bit cao hơn rất nhiều so với mức $n/2$ bit. Điều này được chứng minh bằng việc so sánh thực tế hai giá trị độ an toàn này tính được khi thay $w = 2, 4, 8, 16$ với mọi giá trị n vào công thức tính.

3.3. So sánh hiệu năng cài đặt thực nghiệm của W-OTS và W-OTS+

Chúng tôi đã tiến hành cài đặt hai thuật toán W-OTS và W-OTS+ bằng ngôn ngữ lập trình Python theo mã nguồn trong [12] có kiểm tra tính đúng đắn mã nguồn cài đặt với các Test Vector đã công bố.

Để so sánh hiệu năng thực thi của W-OTS và W-OTS+ chúng tôi đã tiến hành cài đặt hai thuật toán W-OTS và W-OTS+ có các tham số tương đương nhau trong đó: $w = 16$, hàm băm mật mã: openssl_sha512, và hàm giả ngẫu nhiên được sử dụng cho W-OTS+ là prf: hmac_openssl_sha256. Quá trình thực hiện được sử dụng với đầu vào là các tập tin dữ liệu có dung lượng khác nhau trên cùng một nền tảng phần cứng là Laptop HP Zbook 15 Intel Core I7 4680, 16 Gb RAM chạy Window 11. Thời gian thực thi của các thuật toán được thể hiện như bảng 2 (lấy trung bình của 10 lần thực nghiệm của chúng tôi):

Bảng 2. *Thời gian thực thi của hai thuật toán W-OTS và W-OTS+*

Dung lượng file đầu vào	Thời gian thực thi (s)	
	Thuật toán W-OTS	Thuật toán W-OTS+
1MB	0,008	0,028
10MB	0,037	0,062
100MB	0,324	0,474
1GB	6,353	4,521

Dựa trên kết quả trên, thấy rằng quá trình thực thi của thuật toán W-OTS nhanh hơn so với quá trình thực thi của thuật toán W-OTS+. Quá trình thực thi của cả hai thuật toán này là nhanh, đối với tập tin có dung lượng 1MB thì thời gian thực thi lần lược là 0,008s và 0,028s, đối với tập tin 1GB thì thời gian thực thi cũng ở mức 3,233s và 4,794s.

4. Kết luận

Bài báo đã tiến hành phân tích, so sánh được độ an toàn và hiệu năng thực nghiệm của hai lược đồ chữ ký số một lần W-OTS và W-OTS+. Về độ an toàn thì W-OTS+ đạt mức cao hơn so với W-OTS nhưng về mặt hiệu năng thì lược đồ W-OTS nhanh hơn một chút. Các kết quả này có thể giúp các nhà mật mã học hoặc quản trị viên hệ thống lựa chọn được một lược đồ ký số một lần phù hợp trong hoàn cảnh nghiên cứu hoặc kịch bản ứng dụng.

TÀI LIỆU THAM KHẢO/ REFERENCES

[1] K. Chalkias, J. Brown, M. Hearn, T. Lillehagen, I. Nitto, and T.Schroeter, "Blockchained Post-Quantum Signatures," *IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData),* 2018, pp. 1196 – 1203.

[2] J. S. Coron, "Optimal Security Proofs for PSS and Other Signature Schemes," *International Conference on Cryptology in Euro – EUROCRYPT,* 2002, pp. 272–287.

[3] D. Johnson, A. Menezes, and Vanstone, "The Elliptic Curve Digital Signature Algorithm (ECDSA)," *International Journal of Information Security,* vol. 1, pp. 36–63, 2001.

[4] Y. Romailler and S. Pelissier, "Practical Fault Attack against the Ed25519 and EdDSA Signature Schemes," *Workshop on Fault Diagnosis and Tolerance in Cryptography (FDTC),* 2017, pp. 1-22.

[5] Federal Information Processing Standards Publication, "Digital Signature Standard (DSS)," 2023. [Online]. Available: https://doi.org/10.6028/NIST.FIPS.186-5. [Accessed Jul. 18, 2023].

[6] V. Srivastava, A. Baksi, and S. K. Debnath, "An Overview of Hash Based Signatures," *Cryptology ePrint Archive,* no. 411, pp.?, 2023.

[7] J. Buchmann, E. Dahmen, and M. Szydlo, "Hash-based Digital Signature Schemes," in *Post-Quantum Cryptography,* Springer, Chapter 8964, pp. 35–93, 2009.

[8] P. Lafrance and A. Menezes, "On the security of the WOTS-PRF signature scheme," *Cryptology ePrint Archive,* no. 938, pp.?, 2017.

[9] A Hülsing, "W-OTS+ - Shorter Signatures for Hash-Based Signature Schemes," *6th International Conference on Cryptology in Africa,* Springer, Chapter 10, pp.173-188, 2013.

[10] J. Buchmann, E. Dahmen, S. Ereth, A. Hülsing, and M. Rückert, "On the security of the Winternitz one-time signature scheme," *International Conference on Cryptology in Africa – AFRICACRYPT* 2011, pp. 363–378.

[11] L. P. Perin, G. Zambonin, D. M. B. Martins, R. Custódio, and J. E. Martina, "Tuning the Winternitz hash-based digital signature scheme," *IEEE Symposium on Computers and Communications (ISCC),* 2018, pp. 537-542.

[12] H. Heckmann, "Python implementation of Winternitz one-time-signature schemes," [Online]. Available: https://github.com/sea212/winternitz-one-time-signature. [Accessed Jul. 26, 2023].