# Lightweight Verification and Fine-grained Access Control in Named Data Networking Based on Schnorr Signature and Hash Functions

Wu, S., Yuan, Y., Kar, P.

University of Nottingham Ningbo China, 199 Taikang East Road, Ningbo,
315100, Zhejiang, China.

First published 2021

# Lightweight Verification and Fine-grained Access Control in Named Data Networking Based on Schnorr Signature and Hash Functions

Shanglun Wu, Yujie Yuan, Pushpendu Kar*
University of Nottingham Ningbo China
199 Taikang East Road, Ningbo 315100, China
Email: {scysw2, smyyy3, Pushpendu.Kar}@nottingham.edu.cn

***Abstract*—Named Data Networking (NDN) is a new kind of architecture for future Internet, which is exactly satisfied with the rapidly increasing mobile requirement and information-depended applications that dominate today's Internet. However, the current verification-data accessed system is not safe enough to prevent data leakage because no strongly method to resist any device or user to access it. We bring up a lightweight verification based on hash functions and a fine-grained access control based on Schnorr Signature to address the issue seamlessly. The proposed scheme is scalable and protect data confidentiality in a NDN network.**

***Index Terms*—NDN; Verification; Schnorr Signature; Hash functions;**

## I. Introduction

The Internet has become one of the most important infrastructures in modern society for promoting the progress in all fields of the global economy and society, in which, the scale of users and types of business are expanding rapidly [1]. The traditional TCP/IP network protocol is designed to meet the needs of physical address based communication. With the rapid expansion of network scale, the disadvantages of poor security and low content distribution efficiency are exposed. In order to better meet the needs of users, cope with the unprecedented challenges brought by the current development of the Internet, necessity of future development, and realize the sustainable development of the network, Named Data Network (NDN) has introduced. In NDN communication, the receiver (data consumer) communicates through two different types of packets: interest packet and data packet as shown in Fig. 1. Its communication is consumer-driven, and data can be transferred in the form of blocks [2]. NDN routing nodes complete forwarding of data through three important data structures, namely, Forwarding Information Base (FIB), Content Store (CS), and Pending Interest Table (PIT) [3].

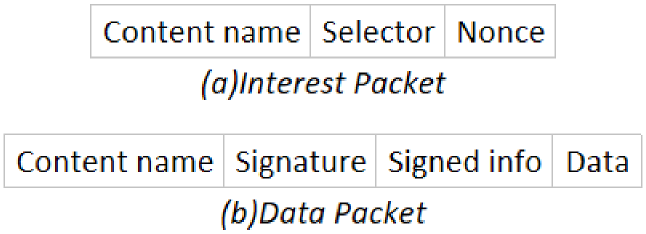

Fig. 1. NDN packets

Unlike the traditional end-to-end TCP/IP network architecture, in NDN, routers directly obtain matching copies of messages from nearby node caches by forwarding the user's interest packet request without having to route to the data source [3]. This kind of non-end-to-end communication mode makes sharing more efficient, but at the same time faces the problem of message security. Users in NDN can quickly obtain matching packets only if the message is reliable, while without message authentication mechanism, the NDN network is vulnerable to attacks such as malicious tampering of messages and man-in-the-middle attack. To solve this security problem each packet needs to be authenticated in the NDN network [4]. Each packet need to be digitally signed and the information obtained from the signature can be used to authenticate the integrity and authenticity of the message. In NDN, because the data packet is generated at the data source, the data packets are needed to be authenticated for many times in the forwarding process [5]. Therefore, when studying the message authentication mechanism, computing efficiency, bandwidth efficiency, and authentication cost need to be comprehensively considered. This research is motivated by these observations.

In this paper, we bring up a lightweight verification and a fine-grained access control technique based on hash functions and Schnorr Signature mechanism, respectively. This technique focuses on the issue of preventing data leakage, which is the purpose of safe verification and data access control.

## II. Background

### A. Named Data Networking (NDN)

Different from the traditional IP network architecture, NDN uses hierarchical name structure rather than IP address for the purpose of packet forwarding and routing. NDN includes two types of packets: interest packet and data packet. In order to request content, consumer sends an interest message with the content name. If in the router the content is cached, then the router sends back the requested content. Or if the router do not have the content, then the content producer sends back the requested content. There are three types of data structures in the NDN network: Forwarding Information Base(FIB), Content Store(CS) and Pending Interest Table(PIT). The FIB reserves the next hop interface to allow the router to reach the producer hamdane2015credential. Data contents are cached by CS. PIT tracks the interests that has not responded and their arrival interface in order that the requested can be returned down the reverse path.

### B. The Identity-based Combined Public Key(ID-CPK)

Combined Public Key (CPK) is an encryption algorithm that generates a large-scale Public Key combination with a small amount of resources. Based on the identification-based digital signature protocol and Key exchange protocol, it satisfies the scale of proof and the straightness of verification. It realizes the assumption of Shamir [6] and opens a new way to solve the scale by combination. In 2007, a two-factor composite public key TF-CPK was formed on the basis of the composite public key, which retains the advantages of the composite public key, enhances the security, realizes the composite digital signature and key exchange mechanism, respectively, and solves the difficulty of defining signature key by individuals in the centralized management mode.

### C. Schnorr Signature Method

In cryptography, a Schnorr Signature, a digital signature, which is generated by Schnorr Signature algorithm described by Claus Schnorr [7]. This kind of digital signature scheme is famous for its simplicity. Its security is on the basis of the intractability of some discrete logarithm problems. It is valid and generates a short signature [8]. It was overdue in February 2008 covered by US patent 4995082.

### D. Hash Functions and Merkle Hash Tree Algorithm

A hash function plays a role of mapping data of any size to a fixed size value. The value returned by a hash function is called a hash value, hash code, digest, or simple hash. These values can be used to index a fixed-size table which is named a hash table. Using a hash function to build an index for a hash table is called hashing or decentralized storage addressing.

The hash function and its relevant hash table are used in data storage and retrieval applications. The data is accessed in a small and almost constant time during each retrieval, and the storage space is only one larger than the total space required by the data or the record itself. Hash is a form of data access that is computationally and storage space efficient. It avoids the non-linear access time of ordered and unordered lists and structured trees. The usual exponents when directly accessing the state space of large keys or variable-length keys storage requirements.

The use of hash functions depends on the statistical characteristics of the interaction between the key and the function: the worst-case behavior is unbearably bad with a very small probability and the average-case behavior can be close to the best-case (minimum collision) [9] .

## III. Problem Description

Unlike the traditional protocol, such as TCP/IP, access of data by their physical addresses, the NDN structure is mainly linked to names of the data instead of data locations to realize better content searching and acquirement. In addition, the security is important aspect of internet protocol need to improve, in other word, NDN has necessity of a better performance in security aspect than traditional architectures. However, there are still some security problems need to be solved, one of them is the data leakage.

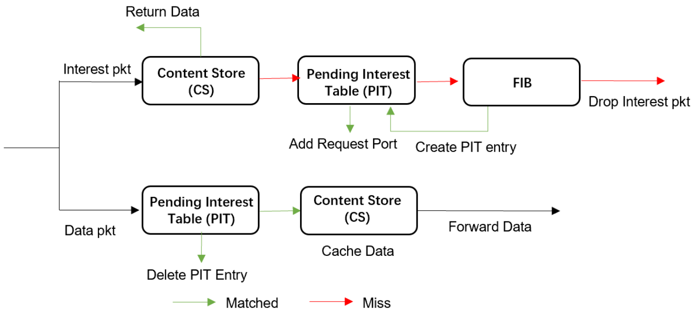

Fig. 2. Process at a NDN node.

Fig. 2 shows the packet transmission process at a NDN node. When data packet transmits in a path, the signature and signed information might be accessed by any user or devices without proper verification and access control.

Previous work on the security aspect of NDN mostly focused on the packet that was transmitted, such as the size of the file, the type of the file, or increased transmission time to reduce the risk, which do not have strong practicability. One of them is the Broadcast Encryption (BE) access control mechanism. BE is used for distribution content symmetric key, but failed to effectively solve the user collusion key problem. In some proxy re-encryption schemes, edge routers are beeping and a random key is generated to re-encrypt the data before it is used. In CPK encryption, random key and re-encrypted data response are used. The user then requests the decryption key to CPK, but re-encryption operation takes up excessive computing overhead on edge router. In this paper, we proposed a lightweight and fined-grained method to protect the security of accessing, which can decrease the related overhead to some extent.

## IV. Proposed Solution

### A. System Model

Fig. 3 describes the system model of the proposed scheme, which combines the signature verification and access control parts. It shows the basic steps of the system. The detailed descriptions are as follows.

The three-way authentication structure is the main model of the whole system, which is depicted in Fig. 4. The signature and verification are contained in the three verification process. The detailed method and related equations are introduced in the following section as mathematical model.

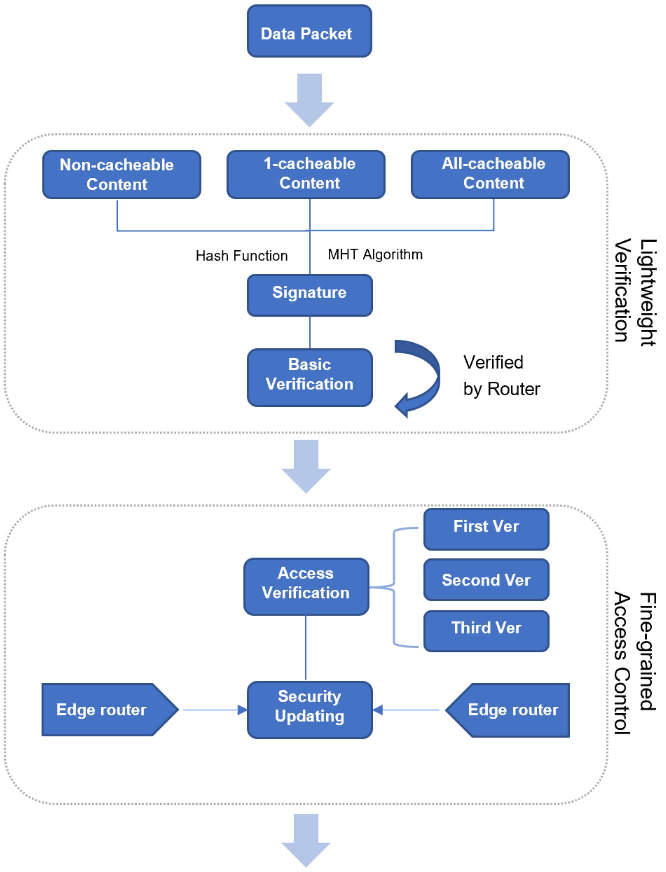

Fig. 3. System Model

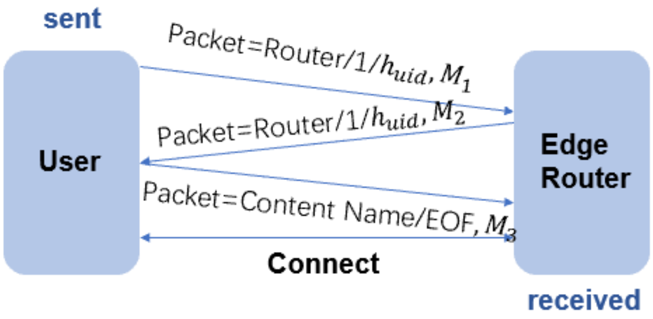
sent
Packet=Router/1/$h_{uid}$, $M_1$
Packet=Router/1/$h_{uid}$, $M_2$
User
Edge Router
Packet=Content Name/EOF, $M_3$
Connect
received

Fig. 4. Identy Verification

### B. Proposed Scheme

To solve the above problem and develop a relatively secure network transmission architecture, this paper presents a Fine-grained Access Control (FGAC) for NDN. FGAC is Combined with identity-based Combined Public Key (CPK) [7] and efficient security Schnorr signature method on the edge to edge routers for three-way anonymous user security for illegal users request filtering. To complete the record fetching and trace FGAC after three-way authentication on legitimate users, the shared secret key and the lightweight one-way hash function [9] is used. To reduce subsequent certification cost and efficient key distribution the FGAC combines with Cryptography Based Identity [10] (IBC) and improved Secret Sharing [6] method.

For the verification and access control part, this technique uses hash function and Merkle Hash Tree (MHT) algorithm [11] to realize the lightweightness, which has been mentioned in both the algorithms,

**Algorithm 1** Signature Generation Algorithm

**Input:** C (Content), Rc(set of routers), C1(1-cacheable), CN(non-cacheable), CA(all-cacheable), Key vector X', X†, X stands for common routers, CR, node for user, i (requesting router);

**Output:** Signature required encryption S

```
1:  X'← x'1, x'2, ···, x'n;
2:  P' ← $h(f(h(x'1))\|f(h(x'2))\|$ ··· f(h(x'n));
3:  if C == CN then
4:      for r=1 to 2l do
5:          yr ← $X\|X$;
6:      end for
7:  else if ((i belongs to Rc) and (C == C1)) then
8:      for r=1 to 2l do
9:          yr ← X† $\|X$;
10:     end for
11: else if (C = CA then
12:     for r=1 to 2l do
13:         yr ← $X'\|X$;
14:     end for
15: end if
16: g ← MHT(P' + C);
17: for i=1 to 2l do
18:     if gi != 0 then
19:         si ← yi;
20:         si ← CPK(si);
21:     else
22:         si ← f(h(yi));
23:     end if
24: end for
25: return  S;
```

1) ***Signing Process***: The Algorithm 1 shows the signature process based on CPK. In order to construct a better signature. The CPK is required to get a fresh token vector(key) X and then uses the Hash Function to produce a fresh token (according first and second step). In non-cachable condition, a vector y1, y2, y3, ···, y2l is produced by CPK through connecting two Xs, where X is the key vector connected with the tokens transmit to the users (step four). This assures that only the produced signatures can be verified by the users. Similarly, in one-cacheble and the content requester condition, the y set is created based on connecting X† and X, where the X† is generated for i like step six. In all-cachable condition, the

CPK is required to produce the bit vector through connecting public tokens X and X'. In step 10, the MHT algorithm was used by CPK to build a bit vector $g$. After that, it produces signature S according to the bit vector $g$. Because each $g$ produces the signatures either for users or routers, we need to connect $g$ to produce the signature for both NDN routers and users (step eleven). From step 14 to 16, the algorithm checks if bits in $g$ are equal to 1 or not, and gives the different to the signature $sj$. Finally, the signature is constructed by connecting all the signature parts. The content signature, the content payload, a token that is used to verify the signatures, a token to update in the NDN router node, and a token to refresh in the user nodes together consist the content packet under deliver after the whole computation process.

**Algorithm 2** Verification Algorithm

```
Input: C(Content), S(Signature), P(the token set), P'(the Content public key), TW(three-way verification function), M (calculated in three-way verification);
Output: True or False
 1: g ← MHT(P' + C);
 2: if S matched routers then
 3:     k ← 0;
 4: else
 5:     k ← 1;
 6: end if
 7: for j=z+1 to z+l do
 8:     if gj = 1 then
 9:         vj ← h(f(sj));
10:     else
11:         vj ← sj;
12:     end if
13: end for
14: V ← $h(Vz+1\|Vz+2\|$... V z+l);
15: return S ← $s1||s2||$···s2l;
16: if m = 1 then
17:     while i=1,i¡4 do
18:         ki = TWi(h(uid),Mi)
19:         if ki=1 then
20:             continue;
21:         else
22:             return False;
23:         end if
24:         i++;
25:     end while
26: else
27:     continue;
28: end if
29: if P is matched with V then
30:     return True;
31: else
32:     return False;
33: end if
```

2) ***Verification Progress*:** The second procedure, signature verification progress algorithm, also proceed in an NDN node. Actually, it has many same steps from the Algorithm 1. At first, in step 1, the vector $g$ is computed with the MHT algorithm, which is according to the token P in the content packet and the received content. Every signature(si) in S is used directly by the node to build a new bit vector V. In one condition, when a router is considered as the NDN node, only the first part of signature in S is needed to get the components in v. In another condition, if the NDN node is a node for users, the progress can immediately start at the second part(step 5) of signatures in S to compute the V. In step 7-13, the NDN node computes all relevant sj according to the bits in $g$, like step 12-18 in Algorithm 1. When all the components of bit are constructed, we come to the three way verification part, use the the three check function in the mathematical model part to check the connection between router and user, then the algorithm connects vj and produces the Hash value of the final signature V. Then, it compares the CPK previously sent to the NDN node of the token with the signature. If V has matched the token in P, it means that the process has successfully verified the node's authorization to access the content. In other words, the content can be transmitted to the user applications or to the Content Store. Otherwise, the content is deleted.

### *C. Mathematical Model*

The data requested by users is divided into three categories: unrestricted shared content, restricted content, and restricted content requiring access verification. When a user requests content, which is classified into 3 categories, the edge router must initiate an authentication request along with an Authentication State Table (AST). The router waits for the confirmation that the user has completed the "Three-way Verification".

(i) *The first verification*: At this stage, the edge router is in the LISTEN state waiting for the user's connection request. In this process, the user needs to hide its certificate to get $W, E_u$ in order to prevent the attacker from intercepting the signed certificate. First, the user needs to get the current timestamp $T_1$ and $h_{uid} = H_2(UID\|T_S)$, then selects two positive integers $u_1$, $r_u$ as random numbers, and finally calculates as follows.

$$R_1 = u_1 \cdot G \tag{1}$$

$$h_T = H_2(T_1) \tag{2}$$

$$R_u = r_u \cdot G \tag{3}$$

$$W = h_T \cdot U + R_u \tag{4}$$

$$\begin{aligned} E^{'}_{u} &= (h_t E_u) \cdot G + R_u, h_1 \\ &= H_2(CPK\|R_1\|T_1), Z_1 \end{aligned} \tag{5}$$

Then, add $M_1$ as calculated by the equation 6

$$M_1 = \{T_1, T_s, R_1, Z_1, W, E^{'}_{u}, CPK\} \tag{6}$$

to the interest packet named /Router/1/$h_{uid}$ and send the interest packet to the edge Router. At this point, the user is in the state of SYN-SENT, which is waiting for a matching connection response.

(ii) *The second verification*: After the edge router receives the interest packet of the connection request, it first verifies whether $T_1$ is in the valid range. If $T_1$ is not within the valid time range, the interest packet is dropped; otherwise, check whether the corresponding CPK or intermediate node request is cached. The edge router then verifies that $T_s$ is less than $T_a$ by the equation 5. If the verification is successful, a positive integer $u_2$ is selected as a random number and then calculate the following equations:

$$R_2 = u_2 \cdot G \tag{7}$$

$$K_u = u_2 \cdot R_1 \tag{8}$$

$$h_2 = H_2(CPK \| R_1 \| T_2) \tag{9}$$

Where $T_2$ is the current timestamp. Finally, the edge router creates a new entry in the AST and gives the $M_2 = \{T_2, R_2, h_2\}$ response to the user. At this point, the edge router is in the SYN-RECEIVED state, waiting for the user's final connection confirmation. If no acknowledgement is received within a certain time frame, the corresponding entry in the AST is deleted.

(iii) *The third verification*: After the user receives $M_2$, verify whether $T_2$ is within the valid range using the equation 10.

$$h_2 = H_2(CPK \| R_1 \| T_2) \tag{10}$$

If the validation is successful, calculate $h_3$ by the equation 11.

$$h_3 = H_2(CPK \| K_u \| T_3) \tag{11}$$

Add $M_3$, calculated by the equation 12, to the interest packet named /ContentName/EOF and send to the edge router.

$$M_3 = \{T_3, h, CPK, h_3\} \tag{12}$$

At this point, the user is in the ESTABLISHED state by completing the authentication connection and starting the request for content "ContentName". When the edge router receives the acknowledgement, it checks to see if $T_3$ is greater than the timestamp in the AST entry and verifies by the equation 11. If the authentication is successful, the connection is successfully established. The edge router transmits the interest packet into the core network, updates the corresponding item in the AST, and start recording the access information at the same time. If the user does not request data for a long time, the edge router releases the connection and clears the corresponding entry in the AST.

After the Three-way verification, if the timestamp meets the requirements, the edge router updates the timestamp of the corresponding item in the AST and transfers the interest packet into the core network.

## V. Discussion

In the paper, we have presented two algorithms by using the CPK, Schnorr signature, and combination of the hash function and MHT algorithm as the verification part. It also divides the access request in three types and gives different signature and verification steps for each part. The proposed scheme fits to the three-way verification in the data access control. By using division and the hash function, it increases the rate of the verification and avoid the high-cost process in traditional method. The FGAC model presents the security protection in the combination of high quality signature and verification, which has higher level of safety. The proposed scheme provides data confidentiality by stopping malicious users to read data passing through the edge routers. When a new node is attached with the existing NDN network, CPK selects an unused polynomial and calculates the United Signature. When CPK updates the secret value $s$ and polynomial, it only needs to update $w$ and $q$ without the need of the authorized user's holdings. Therefore, the Secure Secret Key method improves the scalability. In the first handshake, the sent message has a timestamp, which prevents the attacker in reply attack. Since the request does not contain real identity information, the user request privacy does not disclose through the insecure channel.

To be mentioned, the three-way verification use the similar verification algorithms to check the connection between the user and the edge rounter, which can effectively reduce the risk of the illegal visit. In signature construction, Schnorr Signature method we used is famous for its simplicity, which reduced the complexity of the verification to some extent. In addition, the proposed scheme has divided the requested data into three categories, which is convenient to carry out targeted verification. In this way, the edge router plays an auxiliary verification role in the process, which makes the to realization of light-weightiness of the proposed scheme. That is, three package of verification is not complicated and further lighter than traditional method. Hence, it realizes the light-weight in current level.

Due to the complexity of the verification systems, we further develop the security and privacy based on the CPK signature. So, we focus on the signing algorithm, content encryption, content decryption, and content verification. These four algorithms play as the base of signature security to apply on the NDN system.

## VI. Conclusion

In the paper, we bring up a lightweight verification mechanism for NDN to achieve a complete and exhaustive verification mechanism. We also further discuss the fine-grained access control to assist the main security architecture. In future, we have the plan to focus on the development of a more efficient access control mechanism by combining the specific feature of NDN to improve the CPK authentication.

We still have a lot of work to do in the future for the proposed scheme, such as, proof the concept through simulation and implement it on real network.

## VII. Acknowledgement

The authors would like to extend sincere thanks to University of Nottingham Ningbo China for supporting this research project under Faculty Inspiration Grant (I01190900047).

# References

[1] S. Misra, R. Tourani, F. Natividad, T. Mick, N. E. Majd, and H. Huang, "Accconf: An access control framework for leveraging in-network cached data in the icn-enabled wireless edge," *IEEE Transactions on Dependable and Secure Computing*, vol. 16, no. 1, pp. 5–17, 2017.

[2] D. Wu, Z. Xu, B. Chen, and Y. Zhang, "Towards access control for network coding-based named data networking," in *In Proceedings of the IEEE Global Communications Conference*, Singapore, 2017, pp. 1–6.

[3] B. Hamdane, M. Msahli, A. Serhrouchni, and S. G. El Fatmi, "Data-based access control in named data networking," in *In proceedings of the* $9^{th}$ *IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing*, Austin, TX, USA, 2013, pp. 531–536.

[4] R. S. Da Silva and S. D. Zorzo, "An access control mechanism to ensure privacy in named data networking using attribute-based encryption with immediate revocation of privileges," in *In Proceedings of the* $12^{th}$ *Annual IEEE Consumer Communications and Networking Conference (CCNC)*, Las Vegas, NV, USA, 2015, pp. 128–133.

[5] W. Shang, Q. Ding, A. Marianantoni, J. Burke, and L. Zhang, "Securing building management systems using named data networking," *IEEE Network*, vol. 28, no. 3, pp. 50–56, 2014.

[6] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, 1979.

[7] C.-P. Schnorr, "Efficient signature generation by smart cards," *Journal of cryptology*, vol. 4, no. 3, pp. 161–174, 1991.

[8] B. Hamdane, A. Serhrouchni, and S. G. El Fatmi, "Access control enforcement in named data networking," in *In Proceedings of the* $8^{th}$ *International Conference for Internet Technology and Secured Transactions (ICITST)*, London, UK, 2013, pp. 576–581.

[9] M. Naor and M. Yung, "Universal one-way hash functions and their cryptographic applications," in *In Proceedings of the twenty-first annual ACM symposium on Theory of computing*, Seattle, Washington, USA, 1989, pp. 33–43.

[10] A. Shamir, "Identity-based cryptosystems and signature schemes," in *In Proceedinngs of the Workshop on the theory and application of cryptographic techniques*. Santa Barbara, CA, USA: Springer, 1984, pp. 47–53.

[11] R. C. Merkle, "A digital signature based on a conventional encryption function," in *In Proceedins of the Conference on the Theory and Application of Cryptographic Techniques*. Santa Barbara, CA, USA: Springer, 1987, pp. 369–378.