IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 11, 2018, accepted June 7, 2018, date of publication June 13, 2018, date of current version July 19, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2846798

Oblivious Transfer Based on NTRUEncrypt

BO MI[1], DARONG HUANG[1], (Member, IEEE), SHAOHUA WAN[2], (Member, IEEE), LIBO MI[3], AND JIANQIU CAO[1]

[1]Institute of Information Science and Engineering, Chongqing Jiaotong University, Chongqing 400074, China
[2]School of Information and Safety Engineering, Zhongnan University of Economics and Law, Wuhan 430073, China
[3]Modern Education Technology Centre, Chongqing University of Arts and Sciences, Yongchuan 402160, China

Corresponding authors: Darong Huang (drhuang@cqjtu.edu.cn) and Shaohua Wan (shaohua.wan@ieee.org)

This work was supported in part by the National Science Foundation of China under Grants 61703063, 61573076, and 61663008, in part by the Chongqing Research Program of Basic Research and Frontier Technology under Grant CSTC2017jcyjAX0411, in part by the Scientific Research Foundation for the Returned Overseas Chinese Scholars under Grant 2015-49, in part by the Program for Excellent Talents of Chongqing Higher School under Grant 2014-18, in part by the Science and Technology Research Project of Chongqing Municipal Education Commission of China under Grants KJ1705139, KJ1600518, KJ1705121, and KJ1605002, in part by the Chongqing Municipal Social Livelihood Science and Technology Innovation Project under Grant CSTC2016shmszx30026, and in part by the Urumqi Science and Technology Plan Project under Grant Y161320008.

ABSTRACT Oblivious transfer (OT) is the most fundamental process in cryptosystems and serves as the basic building block for implementing protocols, such as the secure multi-party computation and the fair electronic contract. However, since most implementations of the Internet of Things are time-sensitive, existing works that are based on traditional public cryptosystems are not efficient or secure under quantum machine attacks. In this paper, we argued that the fastest known 1-out-of-n oblivious transfer (OT_n^1) protocol, which was proposed by Chou, cannot achieve semantic security and is time-consuming due to exponent arithmetic of large parameters. Utilizing NTRUEncrypt and OT extension, we devised a one-round post-quantum secure OT_n^1 protocol that is also proved to be active and adaptively secure under the universal composability framework. Compared with Chou's protocol, the computational overheads of our scheme are approximately 6 and 1.7 times smaller on the sender and receiver sides, in line with the standard security level.

INDEX TERMS Oblivious transfer, NTRUEncrypt, universal composability, random oracle.

I. INTRODUCTION

Despite the proliferation of IoT, security issues remain the primary bottleneck for its extensive application [1]–[3]. Until now, no mature cryptographic processes have been sufficiently competent in balancing the relationship between diverse security requirements and efficient applications [4]–[6]. Therefore, we strive to devise a universally practicable building block for secure IoT implementation in accordance with oblivious transfer.

Oblivious transfer, which was introduced as conjugate coding [7], was named by Rabin [8]. Among the many flavors of OT, such as original oblivious transfer, 1-out-of-2 oblivious transfer (OT_2^1) and k-out-of-n oblivious transfer (OT_n^k), 1-out-of-n oblivious transfer has been extensively studied in the literature due to its incomparability to private information retrieval (PIR). To our surprise, this extraordinarily basic process [9] can perform any cryptographic task. OT_n^1 is also known as all-or-nothing disclosure of secrets (ANDOS), owing to its formal definition, which is stated in the Table 1 below.

TABLE 1. Oblivious transfer paradigm $\mathcal{F}_{OT}$.

Input:
–S input $m_1, m_2, \ldots, m_n \in M$
–R input $\tau \in \{1,2,\ldots,n\}$
Output:
–S output $\perp$(i.e., nothing)
–R output m_τ

Definition 1: A 1-out-of-n oblivious transfer protocol is a two-party interactive function $\mathcal{F}_{OT}$ that is executed by a sender (S) and a receiver (R) and is described as

Herein, M represents the message space that contains the sender's data set, while the exact message that the receiver wishes to learn is indicated by τ. The interactive function $\mathcal{F}_{OT}$ also imposes the following requirements, which shape the completeness of OT_n^1.

a. Correctness: if both R and S follow the protocol, R will learn the correct m_τ after interacting with S;

b. Receiver's privacy: Even if S deviates from the protocol, she cannot obtain any information about R's choice τ;

c. Sender's privacy: Even if R deviates from the protocol, she is not allowed to obtain any message or combination of messages except m_τ.

The security notion that captures the above requirements can be testified as full simulation ability, which was proposed by Camenisch *et al.* [10] in 2007 and hints at the indistinguishability between an ideal scenario and a real scenario with respect to any probabilistic poly-time adversary, as shown below.

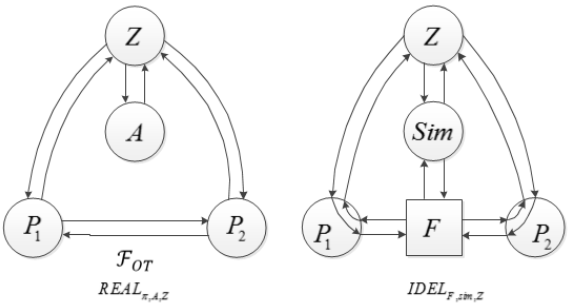

FIGURE 1. Interactions in real and ideal scenarios.

FIGURE 1 indicates that for any communication environment Z, if there exists an ideal attacker *Sim* and a trusted third party F that correspond to the adversary A and OT protocol π, respectively, in the real world, the view of A is limited as follows:

$$REAL_{\pi,A,Z} \equiv IDEL_{F,Sim,Z}, \tag{1}$$

Then, π is proved to be a secure implementation of OT.

Since 1-out-of-n oblivious transfer proved to be a fundamental process in many cryptographic applications, such as hidden access control in cloud, simultaneous contract signing and multi-party computation [11], its efficiency is as important as the security. Although an OT_n^1 protocol can be built from OT_2^1 by invoking the basis n or $log_2 n$ times [12]–[14], it is preferable to implement it directly by means of advanced techniques [15]–[33].

Based on universally composable model, Green and Hohenberger [15] and Jarecki and Liu [16] designed adaptive OT protocols in terms of the q-hidden LRSW assumption and the q-DHI assumption, respectively. Although Kurosawa and Nojima [17] proposed a more explicit scheme, which was not q-based, it was impracticable because of the large communication burden [18]. Soon afterwards, Kurosawa *et al.* [19] revised their scheme by utilizing a verifiable shuffle protocol and reduced the communication cost to $O(1)$. They also summarized their ideas as an adaptive OT construction on any well-known assumption [20].

Since the security model of OT is comprehensively constructed and since its efficiency is closely related to practical scenarios, recent research has intensively focused on efficient instantiations regarding computation and communication costs [21]–[37]. Until now, the most well-known OT protocol that attained UC-security against both active and adaptive corruptions was that of [28]. Since this protocol is achieved by simply tweaking the Diffie-Hellman key-exchange basis over twisted Edwards curve, the scheme reduced the computational cost to $n+6$ exponentiations (3 for the receiver and $n+3$ for the sender) and the communication burden to 2 group elements and $n \times l$-byte ciphertexts for 1-out-of-n OT, where l represents the length of each ciphertext e_i, for $i \in \{1, 2, \ldots, n\}$. To demonstrate the performance improvements of our approach, we describe Chou's protocol in detail as follows [28], and summarize in Table 2.

Despite the simplicity of Chou's protocol, we argue that there are two defects with respect to its security and efficiency:

a. Although the corrupted receiver is incapable of deriving y from S due to the hardness of the Diffie-Hellman problem (DHP), she can forge the message R as $R = \tau S$ and can query the oracle $E_{k_{\tau-1}}(\cdot)$ with $e_{\tau-1}$ in an adaptive chosen ciphertext attack. After acquiring plaintext $m_{\tau-1}$, she is able to simply determine whether $y = y'$ by checking whether $E_{H(y' \cdot S)}(m_{\tau-1}) = e_{\tau-1}$ holds for any $y' \in G$. Furthermore, supposing that the cyclic group G is bilinear, e.g., a super singular elliptic curve, the receiver can trivially distinguish y from y' by comparing $\hat{e}\left(g, y'g'\right)$ with $\hat{e}\left(g', S\right)$, where g' is another generator of cyclic group G and $\hat{e}$ stands for a bilinear mapping. Thus, we claim that Chou's protocol is not secure under the decisional Diffie-Hellman assumption.

b. It is universally known that the security of cryptographic implementations that are based on the Diffie-Hellman problem is limited when parameters are large. That is, the aforementioned protocol is still time-consuming under resource-constrained circumstances due to $O(n)$ exponential operations.

To eliminate the two defects of Chou's protocol, we utilize NTRUEncrypt for OT_n^1 realization. The basic strategy of our scheme is that we exploit the irreversibility of various polynomials and a fail-stop model to avert both active and adaptive corruptions, while the computation and communication burdens are reduced due to convolutional polynomial multiplication and OT extension.

We briefly summarize our contributions as follows:

1. Due to the bilinearity of the cyclic group, the fastest known 1-out-of-n oblivious transfer protocol, which was proposed by Chou et. al., is not semantically secure under the decisional Diffie-Hellman assumption (DDH).
2. For the first time, we introduced a lattice-based cryptographic primitive, namely, NTRUEncrypt, into an $\mathrm{OT}_{\mathrm{n}}^1$ implementation. The devised scheme is proved to be post-quantum secure and resistant to active adaptive attacks under the UC framework.
3. Thanks to the OT extension technology and efficient convolution polynomial multiplication, our protocol requires only 1 round of interaction and the computational overhead of the sender/receiver is

TABLE 2. OT_n^1 protocal in [28].

Sender	Receiver
Setup phase:	
(1) Samples $y \leftarrow G$,computes $S = yg$ and $T = yS$, where g is the generator of cyclic group G	
(2) Sends S to Receiver	(3) Aborts if $S \notin G$
Selection phase:	
	(4) Samples $x \leftarrow G$ and computes $R = \tau S + xg$, wherer $\tau \in \{1,2,\ldots,n\}$ repesents her choice
(6) Aborts if $R \notin G$	(5) Sends R to Sender
Key derivation phase:	
(7) For all $i \in \{1,2,\ldots,n\}$, computes $k_i = H(yR - iT)$, where H is a hash function	(8) Computes $k_R = H(xS)$, where $k_i = k_R$ when $i = \tau$
Transfer phase:	
(9) For all $i \in \{1,2,\ldots,n\}$, computes $e_i \leftarrow E_{k_i}(m_i)$	
(10) Sends $(e_1, e_2, \ldots e_n)$ to Receiver	(11) Computes and output $z = D_{k_R}(m_\tau)$, where E and D represent any symmetric cryptographic algorithms

approximately 6/1.7 times smaller compared to Chou's protocol, which is in line with the standard security level.

The remainder of this paper is organized as follows: In section II, we first present preliminary results regarding NTRUEncrypt and the condition for its correctness. Then, an essential 1-out-of-n oblivious protocol is presented, followed by its complete implementation in section III. Section IV proves the security of our scheme under the universal composability framework, while its performances are evaluated in comparison with Chou's protocol. Finally, we present the conclusions of the paper in section V.

II. PRELIMINARY RESULTS OF NTRUEncrypt

The OT_n^1 protocol that is devised in this paper is structured on the NTRUEncrypt system due to its linearity and resistance to quantum machines. The NTRU encryption algorithm works on a polynomial ring $R = Z[x]/(x^N - 1)$, where the polynomial degree is less than N:

$$\boldsymbol{a}(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_{N-1} x^{N-1}. \tag{2}$$

Since only simple convolutional polynomial multiplications are needed for both encryption and decryption, NTRU outperforms other asymmetric cryptosystems, such as RSA, ElGamal and elliptic curve cryptography. According to the Department of Electrical Engineering, University of Leuven [38], a throughput of up to 200,000 encryptions per second can be reached at a security level of 256 bits by a GTX280 GPU. Compared to a symmetric cipher, this is only approximately 20 times slower than a recent AES implementation [39].

Similar to the prime decomposition problem that is exploited by RSA, the security of NTRUEncrypt relies on the hardness of factoring a reducible polynomial, which is equivalent to lattice reduction. Thus, it is infeasible to usurp the secret key if the parameters are chosen sufficiently securely.

For each system, three integer parameters, namely, (N, p, q), are specified, where p and q are two moduli that truncate the ring R as $R_p = (Z/pZ)[x]/(x^N - 1)$ and $R_q = (Z/qZ)[x]/(x^N - 1)$. It is always assumed that N and p are prime, while q is coprime to both q and N. To prove the correctness of our protocols, a polynomial set $\mathcal{T}(d_1, d_2)$ is specified by two parameters that are defined in advance:

Definition 2: For any positive integers d_1 and d_2,

$$\mathcal{T}(d_1, d_2) = \left\{ \boldsymbol{a}(x) \in R_q : \begin{matrix} \sum_{a_i > 0} a_i = d_1 \\ \sum_{a_i < 0} |a_i| = d_2 \end{matrix}, 0 \le i \le N-1 \right\} \tag{3}$$

In the key generation phase, the receiver randomly chooses two covert polynomials that satisfy

$$\begin{aligned} f(x) &\in \mathcal{T}(d+1, d) \\ g(x) &\in \mathcal{T}(d, d), \end{aligned} \tag{4}$$

and compute

$$\begin{aligned} f_q(x) &= f(x)^{-1} \in R_q \\ f_p(x) &= f(x)^{-1} \in R_p. \end{aligned} \tag{5}$$

If any of the inverses does not exist, she has to go back and try another $f(x)$. After that, she computes and distributes her public key as

$$h(x) = f_q(x) \cdot g(x) \in R_q, \tag{6}$$

while keeping the private key $(f(x), f_p(x))$ secret.

To encrypt a message m, the sender represents it as a binary or ternary string and puts it in the form of a polynomial $m(x)$

whose coefficients belong to $[-p/2, p/2]$. Then, she samples a random polynomial $r(x)$ from $\Im(d, d)$ and computes

$$e(x) = p \cdot h(x) \cdot r(x) + m(x)(mod\ q) \tag{7}$$

as the ciphertext.

After obtaining the ciphertext $e(x)$, the receiver computes

$$a(x) = f(x) \cdot e(x)(mod\ q), \tag{8}$$

and lift its coefficients to the interval $[-q/2, q/2]$ to achieve the plaintext as

$$m(x) = f_p(x) \cdot a(x)(mod\ p). \tag{9}$$

We claim that the correctness of the NTRU encryption algorithm can be guaranteed in the following lemma.

Lemma 1: If the parameters of the NTRU cryptosystem (Enc, Dec) satisfy

$$q > p(4d^2 + 2d + 1), \tag{10}$$

the receiver can accurately recover ciphertext $e(x)$.

Proof: Since

$$a(x) = p \cdot g(x) \cdot r(x) + f(x) \cdot m(x)(mod\ q) \tag{11}$$

by formula (8), the maximal parameter of any $g(x) \cdot r(x)$ can be computed as

$$\begin{aligned} &\max \sum\nolimits_{i+j=k(modN)} g_i r_j \\ &st. \sum\nolimits_{i=0}^{N-1} g_i \le d \\ &\quad \sum\nolimits_{j=0}^{N-1} r_j \le d, \end{aligned} \tag{12}$$

where g_i and r_j are coefficients of the ith and jth orders in $g(x)$ and $r(x)$, respectively. It is concluded that the coefficients of $g(x) \cdot r(x)$ are at most $2d^2$, while the parameters of $f(x) \cdot m(x)$ are no more than $p(2d+1)/2$. Accordingly, the maximal parameter of formula (12) will never exceed $p(2d^2+d+1/2)$. If $p(4d^2 + 2d + 1) < q$, all the parameters of (11) can be lifted to $[-q/2, q/2]$ without losing any information. Thus, by computing

$$\begin{aligned} m(x) &= f_p(x) \cdot a(x)\ (mod\ p) \\ &= f_p(x) \cdot (p \cdot g(x) \cdot r(x) + f(x) \cdot m(x)) \\ &= m(x), \end{aligned} \tag{13}$$

the plaintext can be accurately recovered.

Since the parameters are always small in polynomials $g(x)$, $r(x)$ and $f(x)$, the maximal parameter of formula (12) is generally logarithmically related to dp. For example, if we simply choose the coefficients of these polynomials from $\{-1, 0, 1\}$, the inequality in formula (1) can be relaxed to

$$q > p(6d + 1). \tag{14}$$

III. ONE-OUT-OF-N OBLIVIOUS TRANSFER BASED ON NTRUEncrypt

Due to the capability of withstanding Shor's algorithm-based attacks and the commendable performance, we utilize the NTRU cryptosystem as the main building block to structure a novel OT_n^1 protocol, together with the following primitives:

Symmetric cryptosystem: A non-committing and robust symmetric encryption algorithm (E, D) is used to conceal the messages.

Hash function: A hash function $H{:}Z_q \times Z_q \times \cdots Z_q \to \{0, 1\}^{\omega}$ is used to derive a ω-bit key from a polynomial in truncated ring $R_q = (Z/qZ)[x]/(x^N-1)$ for symmetric encryption.

Our 1-out-of-n oblivious transfer scheme is divided into two parts: first, we present an essential protocol by which the receiver can secretly obtain one of n random keys from the sender; then, the above primitive is combined with OT extension to complete the OT_n^1. The essential protocol is described in Table 3. For clarity, the suffixes (x) of the polynomials are left out, and they are represented as vectors of the coefficients.

During the essential oblivious transfer process, all the operations are executed under modulo q, except for c_S^1 and c_S^2 decryption. Thus, we claim the correctness of the aforementioned protocol in the following lemma:

Lemma 2: If both parties are honest and $q > p\left(4d^3 + 2d + 1\right)$ (or $q > p(8d^2 + 2d + 1)$, when the coefficients of $g(x)$ and $r(x)$ are chosen from $\{-1, 0, 1\}$), the polynomial c_S^{τ} is equivalent to c_R^{τ} modulo q.

Proof: Since $q > p(4d^3 + 2d + 1)$, the receiver can trivially compute r_S^1 and r_S^2 by her private key. Then, she calculates $c_R = p \cdot r_R \cdot r_S \cdot g_S + p \cdot \tau \cdot h_S (mod\ q)$ after recovering r_S and $r_S \cdot g_S$ using the extended Euclidean algorithm. When $i = \tau$, the sender can compute $c_S^{\tau} = c_S - \tau \cdot r_S \cdot f_{Sq} (mod\ q)$, where $c_S = r_R \cdot (r_S)^2 + \tau \cdot r_S \cdot f_{Sq}$. Thus, the corresponding polynomial that she obtains is $c_R^{\tau} = r_R \cdot (r_S)^2\ (mod\ q)$ at the end of the protocol.

However, the parameter d is quadratically or even cubically related to q, which makes the efficiency and security difficult to balance. To ensure the practicability of our protocol, we revisit the sampling process of r_S and g_s to alleviate the parameter requirements, as follows.

Since φ_S is irreversible, the randomness and invariability of r_S are preserved. Since the coefficients of g_S and φ_S are chosen from $\{-1, 0, 1\}$, the maximal parameters of $a_R^1 = p \cdot \varphi_S + f_S \cdot r_S^1$ and $a_R^2 = p \cdot \varphi_S \cdot g_S + f_S \cdot r_S^1$ are $p(2d + 3)/2$ and $p(6d+1)/2$, respectively. Similar to the coefficient lifting requirement in section II, the inequality of Lemma 2 can be eased to

$$q > p(6d + 1). \tag{15}$$

Hitherto, we combine the essential OT_n^1 protocol with hash and symmetric encryption primitives to complete the scheme.

The proof of the correctness of this scheme is straightforward since the symmetric keys $H(c_S^i)$ and $H\left(c_R^{\tau}\right)$ are the same according to Lemma 2.

TABLE 3. Essential protocol π_e.

Sender	Receiver
Key generation phase:	
(1) $(f_S, f_{Sp}, f_{Sq}, g_S) \leftarrow KeyGen(\kappa)$, where $g_S(mod\ q)$ is reversible sk_S: (f_S, f_{Sp}) pk_S: $h_S = f_{Sq} \cdot g_S(mod\ q)$	$(f_R, f_{Rp}, f_R, g_R) \leftarrow KeyGen(\kappa)$, where $g_R(modq)$ is reversible sk_{SR}: (f_R, f_{Rp}) pk_R: $h_R = f_{Rq} \cdot g_R(mod\ q)$
(2) Sends pk_S to Receiver	Sends pk_R to Sender
(3) Aborts if pk_R is irreversible	Aborts if pk_S is irreversible
Oblivious transfer phase:	
(4) Samples $r_S \leftarrow \mathcal{T}(d, d)$, where $r_S(mod\ q)$ is irreversible	
(5) Samples $r_S^1, r_S^2 \leftarrow R_p$ with cofficients between $-p/2$ and $p/2$	
(6) Computes $c_S^1 = p \cdot r_S \cdot g_S \cdot h_R + r_S^1(mod\ q)$, $c_S^2 = p \cdot r_S \cdot h_R + r_S^2(mod\ q)$	
(7) Sends (c_S^1, c_S^2) to Receiver	
	(8) Computes $a_R^1 = f_R \cdot c_S^1$ and $a_R^2 = f_R \cdot c_S^2$
	(9) Aborts if $a_R^1 \notin R_q$ or $a_R^2 \notin R_q$
	(10) Lifts the coefficients of a_R^1 and a_R^2 to $[-q/2, q/2]$ and computes $r_S^1 = f_{Rp} \cdot a_R^1(mod\ p)$, $r_S^2 = f_{Rp} \cdot a_R^2(mod\ p)$
	(11) Computes $r_S \cdot g_S = p^{-1} \cdot (c_S^1 - r_S^1) \cdot h_R^{-1}(mod\ q)$, $r_S = p^{-1} \cdot (c_S^2 - r_S^2) \cdot h_R^{-1}(mod\ q)$
	(12) Samples $r_R \leftarrow \mathcal{T}(d, d)$
	(13) Chooses $\tau \in \{1,2,\ldots,n\}$ and computes $c_R = p \cdot r_R \cdot r_S \cdot g_S + p \cdot \tau \cdot h_S(mod\ q)$
	(14) Sends c_R to Sender
(15) Aborts if $c_R \notin R_q$ or $c_R = p \cdot i \cdot h_S(modq)$ for any i$\in$ {1,2,…,n}	
(16) Computes $c_S = p^{-1} \cdot r_S \cdot g_S^{-1} \cdot c_R(modq)$	
(17) For all $i \in \{1,2,\ldots,n\}$, calculates $c_S^i = c_S - i \cdot r_S \cdot f_{Sq}(mod\ q)$ in parallel	
	(18) Computes $c_R^\tau = r_R \cdot (r_S)^2(mod\ q)$, where $c_R^\tau = c_S^i$ if $i = \tau$

IV. UC SECURITY

We argue that our scheme is secure under the universal composability framework. More precisely, each computationally unbounded adversary is incapable of revealing any information, as stipulated by the OT_n^1 functionality, seen in Tables 4 and 5. The adaptive attacking model is defined as follows:

Adversary model: Assume that $H(\cdot)$, $D_k(\cdot)$, $Dec_{sk}(\cdot)$ as well as $F_r(\cdot)$ are oracles, where $g = F_r(r \cdot g)$, and there exists an authenticated but not confidential channel to avoid identity forgery between the parties. Any corrupted sender or receiver can query the oracle polynomial times, while both malicious sides may preserve all the information about previous repeats as well.

To put aside many of the implementation details and to reach the core of our protocol, two lemmas are presented:

Lemma 3: No computationally unbounded sender on input c_R can guess τ with probability greater than $1/n$.

Proof: Since $c_R = p \cdot r_R \cdot r_S \cdot g_S + p \cdot \tau \cdot h_S(mod\ q)$, even if the sender determined $f_S \cdot c_R$ with the coefficients of $p \cdot f_S \cdot r_R \cdot r_S \cdot g_S$ located in $[0, q-1]$, it is impossible to obtain their exact values modulo q since $c_R = 0(mod\ p)$. By computing $f_S \cdot c_R \cdot p^{-1} \cdot g_S^{-1}(mod\ q)$, polynomial $\tau + r_R \cdot r_S \cdot f_S(mod\ q)$ may be exposed to the sender. Nevertheless, because r_R is randomly sampled from R_q in the random oracle model, she is ignorant of any information about τ as well. In our scheme, the sender should compute:

$$\begin{aligned} c_S^i &= c_S - i \cdot r_S \cdot f_{Sq}(mod\ q) \\ &= r_R \cdot (r_S)^2 + (\tau - i) \cdot r_S \cdot f_{Sq} \end{aligned} \tag{16}$$

for all $i \in \{1, 2, \ldots, n\}$ without knowing r_R. Thus,

$$\begin{aligned} &\Pr[r_R \cdot (r_S)^2 + (\tau - i) \cdot r_S \cdot f_{Sq}] \\ &= \Pr\left[r_R \cdot (r_S)^2 + (\tau - \tau) \cdot r_S \cdot f_{Sq}\right], \end{aligned} \tag{17}$$

TABLE 4. r_S and g_S sampling.

Sender
Key generation phase:
(1) $(f_S, f_{Sp}, f_{Sq}, g_S) \leftarrow KeyGen(\kappa)$, where $g_S(mod\ q)$ is reversible with cofficients that belong to $\{-1,0,1\}$ sk_S: (f_S, f_{Sp}) pk_S: $h_S = f_{Sq} \cdot g_S(mod\ q)$
Oblivious transfer phase:
(4) Samples $\varphi_S \leftarrow \mathcal{T}(d,d)$, where $\varphi_S(mod\ q)$ is irreversible with coefficients that belong to $\{-1,0,1\}$ Computes $r_S = \varphi_S \cdot g_S^{-1}(mod\ q)$

TABLE 5. OT_n^1 scheme based on NTRUEncrypt.

Sender	**Receiver**
Initialization phase:	
(1) Executes the essential protocol with Receiver, obtains all c_S^i for$i \in \{1,2,\ldots,n\}$	Executes the essential protocol with Sender, obtains c_R^τ
Transfer phase:	
(2) Computes $k_i = H(c_S^i)$ according to the cofficients of c_S^i for $i \in \{1,2,\ldots,n\}$ in parallel	Computes $k'_\tau = H(c_R^\tau)$ according to the cofficients of c_R^τ
(3) Computes $c_i = E_{k_i}(m_i)$ for all $i \in \{1,2,\ldots,n\}$ in parallel	
(4) Sends all c_i to Receiver	(5) Computes $m_\tau = D_{k'_\tau}(c_i)$ to obtain message m_τ

where

$$\Pr\left[r_R \cdot (r_S)^2 + (\tau - \tau) \cdot r_S \cdot f_{Sq}\right] = 1/n. \tag{18}$$

Lemma 4: No computationally bounded receiver can produce any polynomial $c_R^i = c_S^i$ with $i \neq \tau$.

Proof: Since $c_S^i = r_R \cdot (r_S)^2 + (\tau - i) \cdot r_S \cdot f_{Sq}\,(mod\ q)$, it is trivial to obtain such a polynomial if the receiver knows f_{Sq}. Although f_{Sq} can be determined by computing $f_{Sq} = h_S \cdot g_S^{-1}$ and she is aware of r_S and $r_S \cdot g_S$, obtaining g_S^{-1} from $r_S \cdot g_S$ is infeasible due to the irreversibility of r_S. In the random oracle model, because $r_S \cdot f_{Sq}$ is irreversible and f_{Sq} is uniformly distributed, we have

$$\Pr\left[c_S^i\right] = \Pr\left[c_S^i = c_R^\tau\right]. \tag{19}$$

Now, we describe how to combine the essential protocol with an appropriate hash function and a symmetric cryptographic scheme to complete the OT_n^1. To demonstrate that a real-world implementation of our scheme is indistinguishable from its simulation, the ideal functionality is first defined as follows:

Definition 3: The ideal functionality $\mathcal{F}_{OT}^{-}(n, 1, l)$ receives an index $\tau \in \{1, 2, \ldots, n\}$ from the receiver R and a vector of l-bit messages $m_1, m_2, \ldots, m_n$ from the sender S but only outputs an l-bit string z to the receiver R.

Then, we encapsulate our setups to model the real functionality $\mathcal{F}_{OT}(n, 1, l)$ and to compare it to the simulation results, as shown in Table 6 and 7. For clarity, we labeled these steps according to Table 3. The messages that are received by corrupted parties (A_S and A_D) or their corresponding simulators (Sim_S and Sim_D) are marked with superscripts $\sim$, while the ceilings $-$ are used to differentiate the parameters from their authentic counterparts.

Tables 6 and 7 indicate that the information that an adversary learned by attacking our protocol is indistinguishable from what can be computed by a simulator that only interacts with the ideal functionality. Therefore, we claim the following:

Theorem 1: Our protocol securely implements the functionality $\mathcal{F}_{OT}^{-}(n, 1, l)$ if the symmetric encryption scheme (E, D) is non-committing and the hash function F perfectly models a random oracle.

Proof: According to Lemma 3 and Table 6,

$$|\Pr[\mathcal{F}_{OT}(n, 1, l) = \tau] - \Pr\left[\mathcal{F}_{OT}^{-}(n, 1, l) = \tau\right]| < \varepsilon(\kappa) \tag{20}$$

Since

$$\Pr\left[\mathcal{F}_{OT}^{-}(n, 1, l) = \tau\right] = 1/n, \tag{21}$$

the receiver's privacy can be achieved.

Due to the non-committing and random oracle properties of (E, D) and F, the distribution of $\overline{k}_i$ or $\overline{m}_i$ is identical between $\mathcal{F}_{OT}(n, 1, l)$ and $\mathcal{F}_{OT}^{-}(n, 1, l)$. By Lemma 4 and Table 7, we have

$$|\Pr[\mathcal{F}_{OT}(n, 1, l) = m_i] - \Pr\left[\mathcal{F}_{OT}^{-}(n, 1, l) = m_i\right]| < \varepsilon(\kappa), \tag{22}$$

and

$$\Pr\left[\mathcal{F}_{OT}^{-}(n, 1, l) = m_i\right] = 1/|M|, \tag{23}$$

TABLE 6. The view of malicious sender versus simulator.

$\mathcal{F}_{OT}(n,1,l)$	$\mathcal{F}_{\overline{OT}}(n,1,l)$
(1) $(pk_S, sk_S), (pk_R, sk_R) \leftarrow KeyGen(\kappa)$	(1) $(pk_S, sk_S), (pk_R, sk_R) \leftarrow KeyGen(\kappa)$
(4) $r_S \leftarrow A_S^{Dec_{sk_R}(\cdot)}(\mathcal{T}(d,d))$	(4) (5) $r_S \leftarrow Sim_S(\mathcal{T}(d,d)), r_S^1, r_S^2 \leftarrow Sim_S(R_p)$
(5) $r_S^1, r_S^2 \leftarrow A_S^{Dec_{sk_R}(\cdot)}(R_p)$	
(6) $c_S^1, c_S^2 \leftarrow A_S(Enc, pk_R, r_S, r_S^1, r_S^2, g_S)$	(6) $c_S^1, c_S^2 \leftarrow \pi_e(Enc, pk_R, r_S, r_S^1, r_S^2, g_S)$
(12) $\overline{r_R} \leftarrow_{Rnd} A_S(\widetilde{c_R}, r_S, g_S, h_S)$	(12) $\overline{r_R} \leftarrow_{Rnd} Sim_R(\widetilde{c_R}, r_S, g_S, h_S)$
VIEW: $\overline{\tau} \leftarrow A_S(c_R, \overline{r_R}, r_S, g_S, h_S)$	VIEW: $\overline{\tau} \leftarrow Sim_S(c_R, \overline{r_R}, r_S, g_S, h_S)$

TABLE 7. The view of malicious receiver versus simulator.

$\mathcal{F}_{OT}(n,1,l)$	$\mathcal{F}_{\overline{OT}}(n,1,l)$
(1) $(pk_S, sk_S), (pk_R, sk_R) \leftarrow KeyGen(\kappa)$	(1) $(pk_S, sk_S), (pk_R, sk_R) \leftarrow KeyGen(\kappa)$
(11) $r_S \cdot g_S, r_S \leftarrow A_R(Enc, sk_R, \widetilde{c_S^1}, \widetilde{c_S^2})$	(11) $r_S \cdot g_S, r_S \leftarrow \pi_e(Enc, sk_R, \widetilde{c_S^1}, \widetilde{c_S^2})$
(1) $\overline{g_S} \leftarrow_{Rnd} A_R^{F_{r,\neq g_S}(\cdot)}(F, r_S \cdot g_S, r_S)$	(1) $\overline{g_S} \leftarrow_{Rnd} Sim_R(F, r_S \cdot g_S, r_S)$
(1) $\overline{f_{Sq}} \leftarrow_{Rnd} A_R(h_S, \overline{g_S})$	(1) $\overline{f_{Sq}} \leftarrow_{Rnd} Sim_R(h_S, \overline{g_S})$
(12) $r_R \leftarrow A_R^{Dec_{sk_S}(\cdot)}(\mathcal{T}(d,d))$	(12) $r_R \leftarrow Sim_R(d,p)$
(13) $c_R \leftarrow A_R(Enc, pk_S, \tau, p, r_R, r_S \cdot g_S)$	(13) $c_R \leftarrow \pi_e(Enc, pk_S, \tau, p, r_R, r_S \cdot g_S)$
(16) (17) $\overline{c_S^i} \leftarrow_{Rnd} A_R(i, r_R, r_S, \overline{f_{Sq}}), i \neq \tau$	(16) (17) $\overline{c_S^i} \leftarrow_{Rnd} Sim_R(i, r_R, r_S, \overline{f_{Sq}}), i \neq \tau$
VIEW: $\overline{k_i} \leftarrow_{Rnd} A_R^{H_{\neq k_i}(\cdot)}(H, \overline{c_S^i}), i \neq \tau,$	VIEW: $\overline{k_i} = Sim_R(H, \overline{c_S^i}), i \neq \tau,$
$\overline{m_i} \leftarrow_{Rnd} A_R^{D_{k_i \neq c_i}(\cdot)}(D, c_i, \overline{k_i}), i \neq \tau$	$\overline{m_i} \leftarrow Sim_R(D, c_i, \overline{k_i}), i \neq \tau$

where $|M|$ stands for the size of the plaintext space. Therefore, our protocol preserves the sender's data ownership as well.

V. PERFORMANCE EVALUATION

In this section, we compare our protocol with the fastest known implementation of OT_n^1 [28]. The parameters of r_S and g_S are sampled according to Table 4. Both protocols require only 1 round of interaction for key exchange, where timing attacks are averted by exploiting the high-level strategy, as in [40]. Since our scheme is based on the NTRU cryptosystem, which uses only simple polynomial multiplication, its operations are very fast compared to other asymmetric cryptographic systems.

To simulate Chou's protocol, the existing Curve25519 implementation [38] is modified to perform scalar multiplications on the twisted Edwards curve [41]. The set of points in cyclic group G are represented as

$$(x, y) \in F_{2^{255}-19} \times F_{2^{255}-19} : -x^2 + y^2 = 1 + dx^2y^2. \quad (24)$$

According to the latest research on NTRUEncrypt [42], the parameters summarized in Table 8, are considered secure:

TABLE 8. NTRU parameters.

Mathematical problem	Shortest-vector problem		
Security level	N	q	p
Moderate Security	401	2048	3
Standard Security	439	2048	3
High Security	593	2048	3
Highest Security	743	2048	3

Thus, we choose the standard and highest security level for comparison.

Without loss of impartiality, the experiments are performed on an Intel Core i3-2330M processor (Sandy Bridge), where each party runs on one core. The computational burden and communication overhead for each essential OT_n^1 is estimated by averaging over 500 tests.

According to Table 9, our protocol outperforms Chou's scheme on both the sender and receiver sides, even when the highest level of security is considered. Although the key generation phase is avoided in Chou's scheme, according to section IV, the security of our protocol would not be compromised if we carried out such a step only once among all iteration. In contrast, since Chou's protocol is not secure under

TABLE 9. Timings for OT_n^1 in μs.

	Our protocol		Chou's protocol
	Standard Security	Highest Security	128-bit security
Key Generation	509	1046	-
Running time of S	83	123	501
Running time of R	136	203	226

the decisional Diffie-Hellman assumption, the sender must resample y and recompute S and T every iteration, despite that they can be regarded as her key pair. To provide post-quantum security as in our scheme, the super singular isogeny Diffie–Hellman key exchange process can be employed by Chou's protocol. However, this process exploits much of the same field arithmetic as existing elliptic curve cryptography approaches and requires computational overhead that is similar to many currently used public key systems [43].

TABLE 10. Communication overhead of OT_n^1 in bits.

	Our protocol		Chou's protocol
	Standard Security	Highest Security	128-bit security
Data size from S	3514	8048	512
Data size from R	1757	4024	512

Table 10 compares the communication burden between our proposed scheme and Chou's protocol. Although the communication load of our scheme is higher than that of Chou's protocol due to the many polynomial coefficients, it is independent of the number of messages and is $O(1)$. Thus, the traffic volume in essential OT_n^1 is acceptable compared with the data size of the ciphertexts.

VI. CONCLUSIONS

Aiming at active and adaptive corruption resistance, a novel 1-out-of-n oblivious transfer protocol that is based on NTRUEncrypt is presented. We note that Chou's protocol is defective in both security and efficiency aspects, although it is the fastest known implementation of OT_n^1. Utilizing OT extension and polynomial multiplication with optimized coefficients, our proposed scheme achieves low computational overhead. In addition, our security analysis demonstrates that the proposed scheme preserves the privacy of both the sender and receiver sides against adaptive chosen ciphertext attacks (CCA2) under the UC framework.

REFERENCES

[1] C. J. D'Orazio, K.-K. R. Choo, and L. T. Yang, "Data exfiltration From Internet of Things devices: iOS devices as case studies," *IEEE Internet Things J.*, vol. 4, no. 2, pp. 524–535, Apr. 2017.

[2] C. J. D'Orazio and K.-K. R. Choo, "Circumventing iOS security mechanisms for APT forensic investigations: A security taxonomy for cloud apps," *Future Gener. Comput. Syst.*, vol. 79, no. 1, pp. 247–261, 2018.

[3] E. Bertino, K.-K. R. Choo, D. Georgakopolous, and S. Nepal, "Internet of Things (IoT): Smart and secure service delivery," *ACM Trans. Internet Technol.*, vol. 16, no. 4, 2016, Art. no. 22.

[4] Z. Liu, K.-K. R. Choo, and J. Grossschadl, "Securing edge devices in the post-quantum Internet of Things using lattice-based cryptography," *IEEE Commun. Mag.*, vol. 56, no. 2, pp. 158–162, Feb. 2018.

[5] L. Wu, B. Chen, K.-K. R. Choo, and D. He, "Efficient and secure searchable encryption protocol for cloud-based Internet of Things," *J. Parallel Distrib. Comput.*, vol. 111, pp. 152–161, Jan. 2018.

[6] M. Ma, D. He, N. Kumar, K.-K. R. Choo, and J. Chen, "Certificateless searchable public key encryption scheme for industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 14, no. 2, pp. 759–767, Feb. 2018.

[7] S. Wiesner, "Conjugate coding," *SIGACT News*, vol. 15, no. 1, pp. 78–88, Jan. 1983.

[8] M. O. Rabin, "How to exchange secrets with oblivious transfer," IACR Cryptol. ePrint Arch., Tech. Rep., 2005, p. 187.

[9] J. Kilian, "Founding crytpography on oblivious transfer," in *Proc. ACM Symp. Theory Comput.*, Chicago, IL, USA, 1988, pp. 20–31.

[10] J. Camenisch and G. Neven, "Simulatable adaptive oblivious transfer," in *Proc. Annu. Int. Conf. Theory Appl. Cryptograph. Techn.* Berlin, Germany: Springer, 2007, pp. 573–590.

[11] W. Biesmans, J. Balasch, A. Rial, B. Preneel, and I. Verbauwhede, "Private mobile pay-TV from priced oblivious transfer," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 2, pp. 280–291, Feb. 2018.

[12] G. Brassard, C. Crepeau, and M. Santha, "Oblivious transfers and intersecting codes," *IEEE Trans. Inf. Theory*, vol. 42, no. 6, pp. 1769–1780, Nov. 1996.

[13] M. Naor and B. Pinkas, "Oblivious transfer and polynomial evaluation," in *Proc. ACM Symp. Theory Comput.* Atlanta, GA, USA: DBLP, 1999, pp. 245–254.

[14] M. Naor and B. Pinkas, "Efficient oblivious transfer protocols," in *Proc. 12th ACM-SIAM Symp. Discrete Algorithms*, 2001, pp. 448–457.

[15] M. Green and S. Hohenberger, "Universally composable adaptive oblivious transfer," in *Proc. Int. Conf. Adv. Cryptol.* New York, NY, USA: Springer-Verlag, 2008, pp. 179–197.

[16] S. Jarecki and X. Liu, "Efficient oblivious pseudorandom function with applications to adaptive ot and secure computation of set intersection," in *Theory of Cryptography*. Berlin, Germany: Springer, 2009, pp. 577–594.

[17] K. Kurosawa and R. Nojima, "Simple adaptive oblivious transfer without random oracle," in *Proc. Int. Conf. Theory Appl. Cryptol. Inf. Secur.* Berlin, Germany: Springer, 2009, pp. 334–346.

[18] M. Green and S. Hohenberger, "Practical adaptive oblivious transfer from simple assumptions," in *Proc. Conf. Theory Cryptogr.* New York, NY, USA: Springer-Verlag, 2011, pp. 347–363.

[19] K. Kurosawa, R. Nojima, and L. T. Phong, "Efficiency-improved fully simulatable adaptive OT under the DDH assumption," in *Proc. Int. Conf. Secur. Cryptogr. Netw.* New York, NY, USA: Springer-Verlag, 2010, pp. 172–181.

[20] K. Kurosawa, R. Nojima, and L. T. Phong, "Generic fully simulatable adaptive oblivious transfer," in *Applied Cryptography and Network Security*. Berlin, Germany: Springer, 2011, pp. 274–291.

[21] J. B. Nielsen, P. S. Nordholt, C. Orlandi, and S. S. Burra, "A new approach to practical active-secure two-party computation," in *Proc. Cryptol. Conf. Adv. Cryptol. (CRYPTO)*. New York, NY, USA: Springer-Verlag, 2012, pp. 681–700.

[22] J. Wang, Y. Yuan, and S. Zhao, "Fractional factorial split-plot designs with two- and four-level factors containing clear effects," *Commun. Statist.-Theory Methods*, vol. 44, no. 4, pp. 671–682, 2015.

[23] S.-L. Zhao and Q. Sun, "On constructing general minimum lower order confounding two-level block designs," *Commun. Statist.-Theory Methods*, vol. 46, no. 3, pp. 1261–1274, 2017.

[24] C. Yin, Y. Shen, and Y. Wen, "Exit problems for jump processes with applications to dividend problems," *J. Comput. Appl. Math.*, vol. 245, pp. 30–52, Jun. 2013.

[25] A. Xu, "Notes on stability of gorenstein categories," *J. Algebra Appl.*, vol. 12, no. 1, p. 1250209, 2013.

[26] W. Sun, Y. Wang, and R. Yang, "L_2 disturbance attenuation for a class of time-delay Hamiltonian systems," *J. Syst. Sci. Complex.*, vol. 24, p. 672, Aug. 2011.

[27] W. Sun and L. Peng, "Observer-based robust adaptive control for uncertain stochastic Hamiltonian systems with state and input delays," *Nonlinear Anal., Model. Control*, vol. 19, no. 4, pp. 626–645, 2014.

[28] T. Chou and C. Orlandi, "The simplest protocol for oblivious transfer," in *Proc. Int. Conf. Cryptol. Inf. Secur. Latin Amer.* Cham, Switzerland: Springer, 2015.

[29] M. Keller, E. Orsini, and P. Scholl, "MASCOT: Faster malicious arithmetic secure computation with oblivious transfer," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2016, pp. 830–842.

[30] Y. Lindell, "Fast cut-and-choose-based protocols for malicious and covert adversaries," *J. Cryptol.*, vol. 29, no. 2, pp. 456–490, 2016.

[31] Y. Sun *et al.*, "Accelerating oblivious transfer with batch multi-exponentiation," in *Proc. Australas. Conf. Inf. Secur. Privacy*. Cham, Switzerland: Springer, 2016, pp. 310–326.
[32] H. Jannati and B. Bahrak, "An oblivious transfer protocol based on elgamal encryption for preserving location privacy," *Wireless Pers. Commun.*, vol. 97, no. 2, pp. 3113–3123, 2017.
[33] G. Asharov, Y. Lindell, T. Schneider, and M. Zohner, "More efficient oblivious transfer extensions," *J. Cryptol.*, vol. 30, no. 3, pp. 805–858, 2017.
[34] S. Wan, Y. Zhang, and J. Chen, "On the construction of data aggregation tree with maximizing lifetime in large-scale wireless sensor networks," *IEEE Sensors J.*, vol. 16, no. 20, pp. 7433–7440, Oct. 2016.
[35] C. Wang, H. Lin, and H. Jiang, "CANS: Towards congestion-adaptive and small stretch emergency navigation with wireless sensor networks," *IEEE Trans. Mobile Comput.*, vol. 15, no. 5, pp. 1077–1089, May 2016.
[36] S. Wan, "Energy-efficient adaptive routing and context-aware lifetime maximization in wireless sensor networks," *Int. J. Distrib. Sensor Netw.*, vol. 10, no. 11, p. 321964, 2014.
[37] S. Wan and Y. Zhang, "Coverage hole bypassing in wireless sensor networks," *Comput. J.*, vol. 60, no. 10, pp. 1536–1544, 2017.
[38] *NTRU: Quantum-Resistant High Performance Cryptography*. Accessed: 2015. [Online]. Avaiable: https://tbuktu.github.io/ntru/
[39] J. Hermans, F. Vercauteren, and B. Preneel, "Speed Records for NTRU," in *Proc. Topics Cryptol. (CT-RSA)* Berlin, Germany: Springer, 2010, pp. 73–88.
[40] D. J. Bernstein, N. Duif, T. Lange, P. Schwabe, and B.-Y. Yang, "High-speed high-security signatures," in *Proc. Int. Workshop Cryptograph. Hardware Embedded Syst.* Berlin, Germany: Springer, 2011, pp. 124–142.
[41] D. J. Bernstein, P. Birkner, M. Joye, T. Lange, and C. Peters, "Twisted Edwards curves," in *Proc. Cryptol. Africa 1st Int. Conf. Progress Cryptol.* Nwe York, NY, USA: Springer-Verlag, 2008, pp. 389–405.
[42] J. Hoffstein, J. Pipher, J. M. Schanck, J. H. Silverman, W. Whyte, and Z. Zhang, "Choosing parameters for NTRUEncrypt," in *Proc. Cryptographers' Track RSA Conf.* Cham, Switzerland: Springer, 2017, pp. 3–18.
[43] D. Jao and L. De Feo, "Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies," in *Proc. Int. Conf. Post-Quantum Cryptogr.* New York, NY, USA: Springer-Verlag, 2011, pp. 19–34.

BO MI was born in 1982. He received the Ph.D. degree in computer system architecture from Chongqing University, China, in 2009. Since 2011, he has been an Associate Professor with the College of Information Science and Engineering, Chongqing Jiaotong University, China. His current research interests include intelligent transportation, vehicular ad hoc networks, and cryptography.

DARONG HUANG was born in 1978. He received the B.S. degree in applied mathematics from the Hubei National Institute, Hubei, China, in 2000, the M.S. degree in applied mathematics from Liaoning University, Liaoning, China, in 2003, and the Ph.D. degree in control theory and control engineering from Chongqing University, Chongqing, China, in 2006. Since 2011, he has been a Professor with the College of Information Science and Engineering, Chongqing Jiaotong University, Chongqing. His research interests include fault diagnosis and fault-tolerant control of dynamical systems, analysis and design of complex systems, big data analytics of transport systems, and reliability engineering.

SHAOHUA WAN received the Ph.D. degree from the School of Computer, Wuhan University, in 2010. In 2015, he was a Post-Doctoral Researcher with the State Key Laboratory of Digital Manufacturing Equipment and Technology, Huazhong University of Science and Technology. From 2016 to 2017, he was a Visiting Scholar with the Department of Electrical and Computer Engineering, Technical University of Munich. He is currently an Associate Professor and a Master Advisor with the School of Information and Safety Engineering, Zhongnan University of Economics and Law. His main research interests include massive data computing for Internet of Things and edge computing.

LIBO MI was born in 1974. He received the M.S. degree in education from Southwest University. He has been engaged in the information planning and technology of the university, the software development of campus digitalization, and the research and practice of higher education information. He has published approximately 10 academic papers in computer science, theory horizon, press circles, and other academic publications. In addition, he was the leading researcher of a provincial scientific project and participated in the research of several scientific projects. He is currently the Senior Experimentalist with the Chongqing University of Arts and Sciences and is mainly engaged in the planning and construction of the campus network, the security of the computer network, the development of the software, and the research and teaching of educational technology.

JIANQIU CAO was born in 1967. He received the master's degree in computer science and technology from Southwest Jiaotong University, China, in 2005. Since 2011, he has been a Professor with the College of Information Science and Engineering, Chongqing Jiaotong University, China. His current research interests include intelligent transportation and image processing.