Secure and Efficient Conjunctive Keyword Search Scheme without Secure Channel

Jianhua Wang[1], Zhiyuan Zhao[1*], Lei Sun[1], Zhiqiang Zhu[1]
[1] Zhengzhou Information Science and Technology Institute
Zhengzhou 450001, Henan – P. R. China
[e-mail: zzy_taurus@foxmail.com]
*Corresponding author: Zhiyuan Zhao

Received April 15, 2018; revised November 13, 2018; accepted December 10, 2018; published May 31, 2019

Abstract

Conjunctive keyword search encryption is an important technique for protecting sensitive data that is outsourced to cloud servers. However, the process of searching outsourced data may facilitate the leakage of sensitive data. Thus, an efficient data search approach with high security is critical. To solve this problem, an efficient conjunctive keyword search scheme based on ciphertext-policy attribute-based encryption is proposed for cloud storage environment. This paper proposes an efficient mechanism for removing the secure channel and resisting off-line keyword-guessing attacks. The storage overhead and the computational complexity are regardless of the number of keywords. This scheme is proved adaptively secure based on the decisional bilinear Diffie-Hellman assumption in the standard model. Finally, the results of theoretical analysis and experimental simulation show that the proposed scheme has advantages in security, storage overhead and efficiency, and it is more suitable for practical applications.

Keywords: Ciphertext-policy attribute-based encryption, Conjunctive keyword search, Secure channel, Keyword-guessing attacks, Search policy

This work was supported by the National Key Basic Research Program of China under Grant No.2013CB338000 and the National key Research Program of China under Grant 2016YFB0501900.

http://doi.org/10.3837/tiis.2019.05.026 **ISSN : 1976-7277**

1. Introduction

Cloud storage is a new concept that extends and develops in the concept of cloud computing [1], and it is a new network storage technology [2]. We know that the direct control power of outsourced data is deprived from the data owners (DO). So, the data should be encrypted before stored to the cloud storage. Whatever, it may severely hinder several functionalities that users are accustomed to receiving from the cloud storage. For instance, it is impossible to search the encrypted data. One solution for this problem is the use of searchable encryption schemes.

There are two categories of searchable encryption schemes: symmetric encryption with keyword search and public key encryption with keyword search (PEKS). Song et al. [3] proposed the first symmetric encryption scheme with keyword search. Boneh et al. [4] proposed the first PEKS. For a network with too many users, PEKS is better than symmetric encryption scheme with keyword search.

In order to reduce the search scope and improve the query performance, mulit-keywords search capability is very important. Therefore, Golle et al. [5] proposed a conjunctive field keyword search scheme, which assumed that n keyword fields are associated with each document. Zhang et al. [6] proposed a conjunctive-subset keyword search algorithm that enables users to list keywords in any order. Lai et al. [7] proposed a PEKS based on the key-policy attribute-based encryption scheme, which is very efficient and has a strong expression.

The above schemes require a secure channel to transfer the trapdoor between the data users (DU) and the cloud service provider (CSP). And heavy computational and communication loads, such as a Secure Sockets Layer (SSL) between the DU and the CSP, are typically required to establish a secure channel. Aiming at solving this problem, Baek et al. [8] considered removing the secure channel and proposed a secure channel-free PEKS (SCF-PEKS). In this scheme, the CSP maintains its public key and private key. The DO generates the keyword ciphertexts based on the public key of the CSP. The CSP having the corresponding private key can run the Test algorithm.

Byun et al. [9] indicated that the scheme proposed by Baek [8] may be attacked by off-line keyword guessing attacks. It is because that the keyword space is much smaller than the password space. Then, Rhee et al. [10] constructed a new secure SCF-PEKS scheme against keyword guessing attacks. But the security proof of the scheme is not in the standard model. Yang et al. [11] proposed a scheme that supports the conjunctive keyword search and resists off-line keyword guessing attacks. However, this scheme uses a significantly complex assumption. Liang et al. [12] proposed a searchable attribute-based proxy re-encryption system, which is able to achieve fine-grained access control. At the same time, the encrypted data is also searchable.

Contribution. Based on the above analyses, an efficient conjunctive keyword search scheme without a secure channel is proposed for the cloud storage environment, which is called searchable encryption with conjunctive keyword search (SE-CKS). We propose an efficient mechanism for removing the secure channel and resisting off-line keyword guessing attacks. The DU is connected to the CSP via an unsecure communication channel, such as a GPRS network. The basic concept is for the server to maintain its private and public key pairs. By referencing the access structure [13] of ciphertext policy attribute-based encryption

(CP-ABE) [14], the DO constructs the search policies using the keywords of the data files, generates keyword ciphertexts through the public keys of the server and the receiver, and then uploads the keyword ciphertexts to the CSP. The keyword set L is used to search the data files, and the AA then generates a trapdoor for the DU. The DU can then send the trapdoor to retrieve data associated with the keyword list and send it via a public channel. After receiving the trapdoor, the CSP can test whether the provided ciphertexts match the trapdoor using its private key. Our scheme is proved adaptively secure based on the decisional bilinear Diffie-Hellman (DBDH) assumption in the standard model. The results of theoretical analysis and experimental simulation show that the proposed scheme has advantages in security, storage overhead and efficiency, and it is more suitable for practical applications.

The remainder of this paper is organized as follows: In Section 2, we describe the formal definition and security model. In Section 3, we propose the concrete SE-CKS scheme and analyse the security of the proposed generic construction. In Section 4, we describe the performance comparison. In Section 5, we present the conclusions.

2. Formal Definition and Security Model

2.1 Formal Definition of SE-CKS

The SE-CKS scheme contains six polynomial time algorithms: $GlobalSetup$, $AASetup$, $KeyGen_{CSP}$, $EncIndex$, $Trapdoor$ and $Test$. These algorithms are presented as follows:

$GlobalSetup(1^{\lambda}) \to GP$: The algorithm is executed by the trusted authority center (AC) and it takes the security parameter λ as input. It returns the global parameter GP.

$AASetup(GP,U) \to (PK,MSK)$: It takes GP and keyword set U as inputs. It returns the AA's public key PK and master private key MSK, respectively.

$KeyGen_{CSP}(GP) \to (pk_{CSP}, sk_{CSP})$: It takes GP as input and returns the CSP's public key pk_{CSP} and private key sk_{CSP}, respectively.

$EncIndex(GP, PK, pk_{CSP}, W) \to CT$: It takes GP, PK, pk_{CSP}, the search policies W based on keywords as inputs. It returns the ciphertext CT.

$Trapdoor(GP, pk_{CSP}, MSK, L) \to TD_L$: It takes GP, pk_{CSP}, MSK, the keyword list L as inputs and outputs a trapdoor TD_L.

$Test(GP, sk_{CSP}, CT, TD_L) \to (0,1)$: It takes GP. sk_{CSP}, CT, TD_L as inputs and determines whether L satisfies W. If L satisfies W, it returns 1; Otherwise, it returns 0.

2.2 Security Model of SE-CKS

Definition 1. Consistency [15]. Assume that the adversary $\mathcal{A}$ wants to cause a failure in consistency. Consistency is formally defined as follows:

Setup: The simulator $\mathcal{B}$ executes $GlobalSetup(1^{\lambda})$, $AASetup(GP,U)$, $KeyGen_{CSP}(GP)$.

Phase 1: $\mathcal{A}$ submits a keyword list L and a search policies W based on keywords, where $L \not\models W$. Then, $EncIndex(GP, PK, pk_{CSP}, W)$ and $Trapdoor(GP, pk_{CSP}, MSK, L)$ are executed.

Challenge: $Test(GP, sk_{CSP}, CT, TD_L)$ is executed, where $L \not\models W$.

Guess: If $Test(GP, sk_{CSP}, CT, TD_L) \to 1$, then the adversary $\mathcal{A}$ wins the game.

The advantage of $\mathcal{A}$ is defined as:

$$Adv_{\mathcal{A}}^{\text{cons}}(\lambda) := \Pr[Test(GP, sk_{CSP}, CT, TD_L) \to 1] \quad (1)$$

If the advantage of all polynomial time adversaries is negligible in above game, then the SE-CKS scheme is computationally consistent.

Compared with traditional conjunctive keyword search, our scheme is based on CP-ABE. Therefore, the security model must be redefined. According to the definition of the CP-ABE security model and the characteristics of our scheme, this paper presents a new security game model for our conjunctive keyword search scheme. Note that there are two types of adversaries in this scheme, namely CSP and the outside attacker (including the receiver). Informally, indistinguishability of secure channel free against chosen keyword attack (IND-CF-CKA) guarantees the adversary, which has not obtained the trapdoors for given keywords, cannot distinguish the keywords. When the adversary $\mathcal{A}$ is the CSP, $\mathcal{A}$ can obtain the CSP's private key. When the adversary $\mathcal{A}$ is the outside attacker, the adversary $\mathcal{A}$ cannot obtain the CSP's private key. Therefore, the CSP has a stronger attack capability than the outside attacker. And this paper only proves that the CSP cannot attack the SE-CKS scheme. The definition of IND-CF-CKA is formalized according to a security game between the adversary $\mathcal{A}$ and the simulator $\mathcal{B}$.

Definition 2. IND-CF-CKA. We consider the following game between $\mathcal{A}$ and $\mathcal{B}$.

Setup: The simulator $\mathcal{B}$ executes $GlobalSetup(1^{\lambda})$, $AASetup(GP, U)$, $KeyGen_{CSP}(GP)$ to obtain the global parameter GP, AA's public key PK and master private key MSK, CSP's public key pk_{csp} and private key sk_{CSP}. Then, $\mathcal{B}$ provides (pk_{CSP}, sk_{CSP}) and PK to $\mathcal{A}$.

Phase 1: The adversary $\mathcal{A}$ submits a keyword list L in a trapdoor query $Trapdoor(GP, pk_{CSP}, MSK, L)$, where $L \not\models W_0 \wedge L \not\models W_1$. The simulator $\mathcal{B}$ answers with a trapdoor for the keyword list L. Note that these queries can be adaptively repeated.

Challenge: The simulator $\mathcal{B}$ chooses $w \in \{0,1\}$ and executes $EncIndex(GP, PK, pk_{CSP}, W_w)$. The simulator $\mathcal{B}$ provides the ciphertext CT to $\mathcal{A}$.

Phase 2: It is the same as **Phase 1**. $\mathcal{A}$ sends L' to the simulator $\mathcal{B}$ for a query. The simulator $\mathcal{B}$ answers with a trapdoor for the keyword list. Notice that $L' \not\models W_0 \wedge L' \not\models W_1$.

Guess: $\mathcal{A}$ outputs the guess $w' \in \{0,1\}$. $\mathcal{A}$ wins if $w' = w$.

The advantage of $\mathcal{A}$ is defined as:

$$Adv_{\mathcal{A}}(\lambda) := |\Pr(w' = w) - 1/2| \quad (2)$$

The SE-CKS scheme is said to be IND-CF-CKA secure if the advantage of all polynomial time adversaries is negligible in above game

Definition 3. Off-Line Keyword Guessing Attacks on SE-CKS.

Because a trapdoor is sent without a secure channel, an outside adversary is capable of capturing the trapdoor and performing off-line keyword guessing attacks. The attacker may reveal the encrypted keyword list that is used by the receiver to search for a data. Similarly, an inside adversary (malicious server) can perform the attack to reveal the keyword in the trapdoor and execute the Test algorithm to determine the ciphertext that contains the keyword list. However, the outside adversary is unable to distinguish ciphertexts from encrypting a specific keyword list because the Test phase requires the server's private key.

A SE-CKS scheme that is secure against keyword guessing attacks, where the attacker is the server, cannot be constructed [16]. Therefore, in this work, we do not consider the keyword

guessing attacks of an inside adversary.

3. SE-CKS Scheme

3.1 Search Policy

The access policy [13] is as follows: Assume that the set of attributes in universe $U=\{att_1,att_2,\cdots,att_n\}$ contains n attributes. Each attribute att_i can take two values: 1 and 0. Assume that $L=[L_1,L_2,\cdots,L_n]$ is a set of attributes for a user, which is called the attribute list. AA generates a user's secret key through $L=[L_1,L_2,\cdots,L_n]$. Assume that $W=[W_1,W_2,\cdots,W_n]$ is an access policy for a ciphertext. Formally, the attribute list $L=[L_1,L_2,\cdots,L_n]$ for a user and the access policy $W=[W_1,W_2,\cdots,W_n]$ for a ciphertext are given. For all i $(1\le i\le n)$, if $L_i=W_i$ or $W_i=*$, L satisfies W, which is represented by the notation $L|=W$. Otherwise, L does not satisfy W, which is represented by the notation $L|\neq W$. The wildcard "*" represents "do not care". For instance, we can let $W=[W_1,W_2,\cdots,W_n]=[0,*,1,*,1,0]$, where $n=6$. If a user has $L=[0,1,1,0,1,0]$, he can obtain a secret key associated with $L=[0,1,1,0,1,0]$ and decrypt the ciphertext encrypted with $W=[0,*,1,*,1,0]$. But the user with $L=[0,1,1,0,1,1]$ cannot decrypt the ciphertext encrypted with $W=[0,*,1,*,1,0]$.

Compared with the access policy [13], we let $U=\{kw_1,kw_2,\cdots,kw_n\}$ represent the set of keywords of data file that replace the attributes. W is the search policy based on the keywords. For all i $(1\le i\le n)$, each keyword kw_i can take two or more values. More formally, assume that $S_i=\{v_{i,1},v_{i,2},\cdots,v_{i,n_i}\}$ is the set of all possible values for kw_i, where n_i is the number of the possible values for kw_i, specifically $n_i=|S_i|$. When the encryptor specifies a wildcard * for W_i, this action corresponds to specify $W_i=S_i$. We achieve keyword privacy by hiding the subset W_i for each keyword kw_i that is specified in the search policy of the AND-gate of all keywords.

3.2 Concrete Construction

The six polynomial time algorithms of SE-CKS are as follows:

$GlobalSetup(1^\lambda)\to GP$: The AC generates the tuple $\mathbb{G}=[p,q,N=pq,G,G_T,e]$ with $G\times G\to G_T$, where G and G_T are cyclic groups of order $N=pq$. g_p and g_q are the generators of G_p and G_q, respectively. The global parameter is $GP=[p,r,g_p,g_q,N=pq,G,G_T,e]$.

$AASetup(GP,U)\to(PK,MSK)$: Randomly choose $\alpha\in Z_N^*$, $a',g_2\in G_p$, $R_0,R_1\in G_q$ and compute $g_1=g_p^\alpha$. For each keyword $kw_i\in U$, where $1\le i\le n$, AA chooses random values $\{a_{i,t}\in Z_N^*\}_{1\le t\le n_i}$ and $\{R_{i,t}\in G_q\}_{1\le t\le n_i}$. AA's master private key is $MSK=[\alpha,a',g_2,\{a_{i,t}\}_{1\le i\le n,1\le t\le n_i}]$. AA's public key is $PK=[Y=e(g_1,g_2),A_0=g_p\cdot R_0,A'=a'R_1,\{A_{i,t}=g_p^{a_{i,t}}\cdot R_{i,t}\}_{1\le i\le n,1\le t\le n_i}]$.

$KeyGen_{CSP}(GP)\to(pk_{CSP},sk_{CSP})$: Uniformly and randomly choose $\beta\in Z_N^*$ and compute $B=g_p^\beta$. The CSP's public key is $pk_{CSP}=[B]$. The CSP's private key is $sk_{CSP}=[\beta]$.

$EncIndex(GP,PK,pk_{CSP},W)\to CT$: Choose a search policy based on the keywords $W=[W_1,\cdots,W_n]$. The DO selects random values $s\in Z_N^*$ and $R_0',R_1'\in G_q$, then the ciphertext is

$CT=[C=Y^s, C_0=A_0^s\cdot B^s\cdot R_0', C_1=(A'\prod_{v_{i,t}\in W}A_{i,t})^s\cdot R_1']$.

$Trapdoor(GP, pk_{CSP}, MSK, L)\rightarrow TD_L$: The DU uses $L=[L_1,\cdots,L_n]=[v_{1,t_1},\cdots,v_{n,t_n}]$ to obtain the corresponding secret key for searching, which is regarded as the searching trapdoor. The AA selects a value $r\in Z_N^*$. The searching trapdoor is: $TD_L=[D_0=g_p^r\cdot B^r, D_1=g_2^{\alpha}(a'\cdot g_p^{\sum_{v_{i,t}\in L}a_{i,t}})^r]$.

$Test(GP, sk_{CSP}, CT, TD_L)\rightarrow(0,1)$: The DU sends TD_L to the CSP for implementing the search request. Then, the CSP tests whether $e(D_0,C_1)\cdot C^{\beta+1}=e(D_1,C_0)$ is true. If it is true, then return 1; Otherwise, return 0.

Note that in *Trapdoor* algorithm, this paper assumes $\forall L, L'(L\neq L')$, $\sum_{v_{i,t}\in L}a_{i,t}\neq\sum_{v_{i,t}\in L'}a_{i,t}$. Emura et al. [17] gave the result that this assumption holds with overwhelming probability $P_{assump}(>1-N_0^2/N)$, where $N_0=|S_i|^n$.

Correctness. Let the ciphertext be $CT=[C,C_0,C_1]$, which is associated with the search policy $W=[W_1,W_2,\cdots,W_n]$ based on keywords. The trapdoor is $TD_L=[D_0,D_1]$. This process produces the equation:

$$\begin{aligned}
&\frac{e(D_0,C_1)\cdot C^{\beta+1}}{e(D_1,C_0)}\\
&=\frac{e(g_p^r\cdot B^r,(A'\prod_{v_{i,t}\in W}A_{i,t})^s\cdot R_1')\cdot C^{\beta+1}}{e(g_2^{\alpha}(a'\cdot g_p^{\sum_{v_{i,t}\in L}a_{i,t}})^r, A_0^s\cdot B^s\cdot R_0')}\\
&=\frac{e(g_p^r\cdot g_p^{\beta r},(a'R_1\prod_{v_{i,t}\in W}g_p^{a_{i,t}}\cdot R_{i,t})^s\cdot R_1')\cdot C^{\beta+1}}{e(g_2^{\alpha}(a'\cdot g_p^{\sum_{v_{i,t}\in L}a_{i,t}})^r,(g_p\cdot R_0)^s\cdot g_p^{\beta s}\cdot R_0')}\\
&=\frac{e(g_p^r\cdot g_p^{\beta r},(a')^s)\cdot e(g_p^r\cdot g_p^{\beta r},\prod_{v_{i,t}\in W}g_p^{a_{i,t}s})\cdot e(g_1,g_2)^{s(\beta+1)}}{e((a'\cdot g_p^{\sum_{v_{i,t}\in L}a_{i,t}})^r,(g_p)^s\cdot g_p^{\beta s})\cdot e(g_2^{\alpha},(g_p)^s\cdot g_p^{\beta s})}\\
&=\frac{e(g_p^r\cdot g_p^{\beta r},(a')^s)\cdot e(g_p^r\cdot g_p^{\beta r},\prod_{v_{i,t}\in W}g_p^{a_{i,t}s})}{e((a')^r,(g_p)^s\cdot g_p^{\beta s})\cdot e((g_p^{\sum_{v_{i,t}\in L}a_{i,t}})^r,(g_p)^s\cdot g_p^{\beta s})}=1
\end{aligned}\tag{3}$$

That is $e(D_0,C_1)\cdot C^{\beta+1}=e(D_1,C_0)$.

3.3 Security Analysis

Theorem 1. SE-CKS is computationally consistent.

Proof: Assume that an adversary $\mathcal{A}$ can attack the computational consistency of SE-CKS. Let (W,L) denote the search policy based on keywords and the keyword list for the DU. At the same time, assume that L does not satisfy W. The the consistency game is as follows:

Select random values $s\in Z_N^*$ and $R_0', R_1'\in G_q$. Compute $C=Y^s$, $C_0=A_0^s\cdot B^s\cdot R_0'$, $C_1=(A'\prod_{v_{i,t}\in W}A_{i,t})^s\cdot R_1'$, $D_0=g_p^r\cdot B^r$, $D_1=g_2^{\alpha}(a'\cdot g_p^{\sum_{v_{i,t}\in L}a_{i,t}})^r$, where $r\in Z_N^*$.

If L does not satisfy W and $e(D_0,C_1)\cdot C^{\beta+1}=e(D_1,C_0)$ holds, then $\mathcal{A}$ wins the game. Assume that $L_k\not\models W_k$, then $a_{k,t_1}=z_1$ in L_k and $a_{k,t_2}=z_2$ in W_k.

$$e(D_0, C_1) \cdot C^{\beta+1} = e(D_1, C_0)$$

$$\Leftrightarrow e(g_p^r \cdot g_p^{\beta r}, (a' R_1 \prod_{v_{i,t} \in W} g_p^{a_{i,t}} \cdot R_{i,t})^s \cdot R_1') \cdot e(g_1, g_2)^{s(\beta+1)} = e(g_2^{\alpha} (a' \cdot g_p^{\sum_{v_{i,t} \in L} a_{i,t}})^r, (g_p \cdot R_0)^s \cdot g_p^{\beta s} \cdot R_0')$$

$$\Leftrightarrow e(g_p^r \cdot g_p^{\beta r}, (a')^s) \cdot e(g_p^r \cdot g_p^{\beta r}, \prod_{v_{i,t} \in W} g_p^{a_{i,t} s}) = e((a' \cdot g_p^{\sum_{v_{i,t} \in L} a_{i,t}})^r, (g_p)^s \cdot g_p^{\beta s})$$

$$\Leftrightarrow e(g_p^r \cdot g_p^{\beta r}, (a')^s) \cdot e(g_p^r \cdot g_p^{\beta r}, \prod_{v_{i,t} \in W} g_p^{a_{i,t} s}) = e((a')^r, (g_p)^s \cdot g_p^{\beta s}) \cdot e((g_p^{\sum_{v_{i,t} \in L} a_{i,t}})^r, (g_p)^s \cdot g_p^{\beta s})$$

$$\Leftrightarrow e(g_p^r \cdot g_p^{\beta r}, \prod_{v_{i,t} \in W} g_p^{a_{i,t} s}) = e((g_p^{\sum_{v_{i,t} \in L} a_{i,t}})^r, (g_p)^s \cdot g_p^{\beta s})$$

$$\Leftrightarrow e(g_p, g_p^{\sum_{v_{i,t} \in W} a_{i,t}})^{rs(\beta+1)} = e(g_p^{\sum_{v_{i,t} \in L} a_{i,t}}, g_p)^{rs(\beta+1)}$$

$$\Leftrightarrow \sum_{v_{i,t} \in W} a_{i,t} - \sum_{v_{i,t} \in L} a_{i,t} = 0$$

$$\Leftrightarrow z_1 - z_2 = 0 \tag{4}$$

Because the DU does not know the secret value z_1 and z_2 in Z_N^*. Hence, $Pr[z_1 = z_2] = 1/(N-1)$, where $N-1$ is the total number of all elements in Z_N^*. When $L \not\models W$ and $Test(GP, sk_{CSP}, CT, TD_L) \to 1$, the advantage of the adversary $\mathcal{A}$ winning the above game is:

$$Adv_{\mathcal{A}}^{cons}(\lambda) = Pr[z_1 = z_2] \le 1/(N-1) \tag{5}$$

Theorem 2. If the $(1 - N_0^2/N)(\varepsilon/16(n+1)\theta)$ DBDH assumption holds, then SE-CKS is (t, θ, ε)-IND-CF-CKA secure in the standard model, where $N_0 = |S_i|^n$ is the number of all possible expressed search policies.

Proof: If an (t, θ, ε) adversary $\mathcal{A}$ can break through SE-CKS, then a simulator $\mathcal{B}$ can be constructed to solve the DBDH problem with the advantage no less than $(1 - N_0^2/N)(\varepsilon/16(n+1)\theta)$. The DBDH challenger $\mathcal{C}$ selects $a, b, c \in Z_N^*$, $v \in \{0,1\}$, g_p, g_q, where $\langle g_p \rangle = G_p$, $\langle g_q \rangle = G_q$. If $v = 0$, then $Z = e(g_p, g_p)^{abc}$; if $v = 1$, then $Z = e(g_p, g_p)^z$. The security game based on the tuple $[g_p, g_r, g_p^a, g_p^b, g_p^c, Z]$ is simulated between the adversary $\mathcal{A}$ and the simulator $\mathcal{B}$.

Setup: $\mathcal{B}$ computes $u = 4\theta$ and randomly chooses an value $k \in \{0, \cdots, n\}$. Then $\mathcal{B}$ chooses $\{x_{i,t}\}_{1 \le i \le n, 1 \le t \le n_i}$, where $x_{i,t} \in (0, \cdots, u-1)$, $x' \in (0, \cdots, u-1)$. Additionally, $\mathcal{B}$ chooses $y' \in Z_N^*$ and $\{y_{i,t}\}_{1 \le i \le n, 1 \le t \le n_i}$, where $y_{i,t} \in Z_N^*$.

Define functions $F(L) = (N - uk) + x' + \sum_{v_{i,t} \in L} x_{i,t} (\mathrm{mod}\, N)$ and $J(L) = y' + \sum_{v_{i,t} \in L} y_{i,t} (\mathrm{mod}\, N)$ for the attribute list L. Then, if $x' + \sum_{v_{i,t} \in L} x_{i,t} \equiv 0 (\mathrm{mod}\, u)$, we define a binary function $B(L) = 0$; Otherwise, $B(L) = 1$.

$\mathcal{B}$ assigns $g_1 = g_p^a$, $g_2 = g_p^b$ and chooses $r' \in Z_N^*$, $r_{i,t} \in Z_N^*$, $R_1 \in G_r$. Then it outputs the AA's PK parameters $\{A_{i,t} = g_2^{x_{i,t}} g_p^{y_{i,t}} g_q^{r_{i,t}}\}_{1 \le i \le n, 1 \le t \le n_i}$, $Y = e(g_1, g_2)$, $A_0 = g_p \cdot R_1$, $A' = g_2^{p - uk + x'} g_p^{y'} g_q^{r'}$. Here PK implies that $a_{i,t} = b x_{i,t} + y_{i,t}$. $\mathcal{B}$ selects $\delta \in Z_N^*$ and computes $B = g_p^{\delta}$. Establish the CSP's public key $pk_{CSP} = (B)$ and the CSP's private key $sk_{CSP} = (\delta)$. Finally, $\mathcal{B}$ provides (PK, pk_{CSP}, sk_{CSP}) to $\mathcal{A}$.

Phase 1: $\mathcal{A}$ submits the keyword list $L=[L_1,\cdots,L_n]$ in a trapdoor query. If $B(L)=0$, $\mathcal{B}$ stops the game and outputs a random value v' as the guess of v. Otherwise, $\mathcal{B}$ chooses $r\in Z_N^*$ and computes:

$$D_0=(g_1^{\frac{-1}{F(L)}}g_p^r)^{\delta+1}, D_1=g_1^{\frac{-J(L)}{F(L)}}(a'\cdot g_p^{\sum_{v_{i,t}\in L}a_{i,t}})^r \tag{6}$$

where $a'=g_2^{F(L)-\sum_{v_{i,t}\in L}x_{i,t}}g_p^{y'}$.

Let $\tilde{r}=r-a/F(L)$. Then we have:

$$\begin{aligned} D_1 &= g_1^{\frac{-J(L)}{F(L)}}(a'\cdot g_p^{\sum_{v_{i,t}\in L}a_{i,t}})^r \\ &= g_1^{\frac{-J(L)}{F(L)}}(g_2^{F(L)}g_p^{J(L)})^r \\ &= g_2^a(g_2^{F(L)}g_p^{J(L)})^{\frac{-a}{F(L)}}(g_2^{F(L)}g_p^{J(L)})^r \\ &= g_2^a(g_2^{F(L)}g_p^{J(L)})^{r-\frac{a}{F(L)}} \\ &= g_2^a(a'\cdot g_p^{\sum_{v_{i,t}\in L}a_{i,t}})^{\tilde{r}} \end{aligned} \tag{7}$$

Then verify if $D_0=(g_1^{\frac{-1}{F(L)}}g_p^r)^{\delta+1}=(g_p^{r-\frac{a}{F(L)}})^{\delta+1}=(g_p^{\tilde{r}})^{\delta+1}=g_p^{\tilde{r}}\cdot(g_p^{\delta})^{\tilde{r}}=g_p^{\tilde{r}}\cdot B^{\tilde{r}}$.

Iff $F(L)\neq 0 \bmod N$, $\mathcal{B}$ can complete the above calculation process. And only when $B(L)\neq 0$ ($B(L)\neq 0$ implies $F(L)\neq 0$), the game continues.

Challenge: $\mathcal{A}$ submits two search policies W_0, W_1. $\mathcal{B}$ randomly selects $w\in\{0,1\}$. If $x'+\sum_{v_{i,t}\in W_w}x_{i,t}\neq uk$, $\mathcal{B}$ stops the game and outputs a random value v'; Otherwise, $F(W_w)\equiv 0(\bmod N)$ holds. Next, $\mathcal{B}$ chooses $R_0\in G_r$, $r_c\in Z_N^*$. r_c can be written in the form $r_c=cr''+r_0$ for some unknown r_0, where $c,r''\in Z_N^*$, $r''=r'+\sum_{v_{i,t}\in W_w}r_{i,t}$. We let $g_q^{r_0}=R_1'$, then the ciphertext is $CT=[C=Z, C_0=C^{\beta+1}\cdot R_0, C_1=C^{J(W_w)}g_q^{r_c}]$.

The correctness of the ciphertext is verified as follows:

$$C_0=C^{\beta+1}R_0=g_p^{c(\beta+1)}R_0=g_p^c\cdot g_p^{\beta c}\cdot R_0=g_p^c\cdot B^c\cdot R_0 \tag{8}$$

$$\begin{aligned} C_1 &= C^{J(W_w)}g_r^{r_c}=(g_2^{F(W_w)}g_p^{J(W_w)}g_q^{r''})^c R_1' \\ &= (g_2^{F(W_w)-\sum_{v_{i,t}\in W_w}x_{i,t}}g_p^{y'}g_q^{r'}\prod_{v_{i,t}\in W_w}(g_p^{a_{i,t}}g_q^{r_{i,t}}))^c R_1' \\ &= (a'g_q^{r'}\prod_{v_{i,t}\in W_w}A_{i,t})^c R_1'=(A'\prod_{v_{i,t}\in W_w}A_{i,t})^c R_1' \end{aligned} \tag{9}$$

Phase 2: It is the same as **Phase 1**.

Guess: $\mathcal{A}$ outputs the guess $w'\in\{0,1\}$. If $w'=w$, $\mathcal{B}$ outputs $v'=0$; Otherwise, $\mathcal{B}$ outputs $v'=1$.

Analysis. If $Z=e(g_p,g_p)^{abc}$ then $C=(g_p^a,g_p^b)^c$. The CT is a valid ciphertext index based on W_w. So, $\mathcal{A}$ has the advantage ε to solve the problem. Hence, $Pr[w'=w|e(g_p,g_p)^{abc}]=1/2+\varepsilon$. If $Z=e(g_p,g_p)^z$, the adversary $\mathcal{A}$ cannot distinguish w. Hence, $Pr[w'=w|e(g_p,g_p)^z]=1/2$. We can see that the game will not abort if

$x' + \sum_{v_{i,t} \in L} x_{i,t} \neq 0 (\mathrm{mod}\, u)$ and $x' + \sum_{v_{i,t} \in W_w} x_{i,t} = uk (\mathrm{mod}\, N)$ hold.

Inspired by literature [18], $\mathcal{B}$'s advantage is at least $(1 - N_0^2 / N)(\varepsilon / 16(n+1)\theta)$ in the above game.

Theorem 3. SE-CKS is secure against off-line keyword guessing attacks.

Proof: Suppose the outside adversary $\mathcal{A}$ exists who can intercept and capture the trapdoor TD_L. $\mathcal{A}$ can obtain the parameter GP, AA's public key PK and CSP's public key $pk_{CSP} = (B)$ from the public network.

Aiming at obtaining the encrypted keywords, $\mathcal{A}$ selects the keyword list $\hat{L}$ and initiates the keyword guessing attacks as follows:

$$
\begin{aligned}
& Y \cdot e(D_0, \prod A' \hat{A}_{i,t}) \stackrel{?}{=} e(g_p, D_1) \\
& e(g_1, g_2) e(g_p^r B^r, \prod a' R_1 g_p^{\hat{a}_{i,t}} R_{i,t}) \stackrel{?}{=} e(g_p, g_2^{\alpha}) (a' \cdot g_p^{\sum_{v_{i,t} \in L} a_{i,t}})^r) \\
& e(g_1, g_2) e(g_p^r g_p^{r\beta}, \prod a' g_p^{\hat{a}_{i,t}}) \stackrel{?}{=} e(g_p, g_2^{\alpha}) e(g_p, (a' \cdot g_p^{\sum_{v_{i,t} \in L} a_{i,t}})^r) \\
& e(g_p^{r(\beta+1)}, \prod a' g_p^{\hat{a}_{i,t}}) \stackrel{?}{=} e(g_p, (a' \cdot g_p^{\sum_{v_{i,t} \in L} a_{i,t}})^r) \\
& e(g_p^{r(\beta+1)}, a') e(g_p^{r(\beta+1)}, \prod g_p^{\hat{a}_{i,t}}) \stackrel{?}{=} e(g_p, (a')^r) e(g_p, (g_p^{\sum_{v_{i,t} \in L} a_{i,t}})^r) \\
& e(g_p^{r(\beta+1)}, a') e(g_p^{r(\beta+1)}, g_p^{\sum \hat{a}_{i,t}}) \stackrel{?}{=} e(g_p^r, a') e(g_p^r, g_p^{\sum_{v_{i,t} \in L} a_{i,t}}) \\
& \left(e(g_p^r, a') e(g_p^r, g_p^{\sum \hat{a}_{i,t}}) \right)^{(\beta+1)} \stackrel{?}{=} e(g_p^r, a') e(g_p^r, g_p^{\sum_{v_{i,t} \in L} a_{i,t}})
\end{aligned}
\tag{10}
$$

In these derivations, $\mathcal{A}$ does not know the AA's private key β. So, $\mathcal{A}$ cannot break SE-CKS through initiating keyword guessing attacks.

4. Performance Comparison

This section compares SE-CKS with other schemes in terms of the features, storage overhead and computational efficiency. In the comparison process, let $|p|$ be the size of the element in Z_p, let $|N|$ be the size of the element in Z_N, let $|\lambda|$ be the size of the security parameter λ, let $|S|$ be the number of attributes in an attribute set, let $|g|$ and $|g_T|$ be the size of the element in G and G_T, respectively ($|g_1|$ and $|g_2|$ are denoted by $|g|$). Let n indicate the total number of keywords in the system, let l indicate the row number of $\boldsymbol{M}$ in the scheme [7].

4.1 Features Comparison

Table 1 shows a comparison of the features of certain aspects of the system. The schemes of Zhang and Liang are based on the prime order bilinear group. Lai's scheme and our scheme are based on the composite order bilinear group. Under the same security, the computational efficiency of the composite order group is lower than that of the prime order group. The security of Zhang's scheme and Liang's scheme is based on the strong assumptions of p-DDHI, q-BDHEA, respectively. The security of Lai's scheme is based on the static assumption of the composite order group. Our scheme's security is based in the simple assumption of DBDH.

Zhang's scheme does not provide proof of security. Liang's scheme is selective security in the random oracle model (ROM). Lai's scheme and our scheme are adaptive security in the standard model. Therefore, our scheme is stronger than the other three schemes in terms of security. Liang's scheme does not support the multi-keywords search, which is supported in the other three schemes. In addition, our scheme removes the secure channel and can resist off-line keyword guessing attacks. The establishment of a secure channel requires a lot of computing resources.

Table 1. Features Comparison

Features	Zhang [6]	Lai [7]	Liang [12]	Ours
Bilinear Maps	Prime	Composite	Prime	Composite
Assumption	P-DDHI	Static	Q-BDHE	DBDH
Secure Model	-	Standard	ROM	Standard
Security	-	Adaptive	Selective	Adaptive
Multi-Keywords	Yes	Yes	No	Yes
SCF	No	No	No	Yes
Off-line KGA	No	No	No	Yes

4.2 Storage Overhead

Table 2 shows the comparison of the storage overhead on each entity in the system. To achieve the multi-user search, Liang's scheme and our scheme require the AA to generate a master private key, which generates a trapdoor for each receiver. The main storage overhead on the AA is generated by the master key. In our scheme, the AA must generate a master key for each keyword value. Liang's scheme only generates two elements in Z_p^* and two elements in G_1. All public parameters contribute to the storage overhead on the owner. The storage overhead of all schemes is almost the same, it is $\mathcal{O}(n)|g|+\mathcal{O}(1)|g_T|$. The CSP is required to store the ciphertext. In our scheme, the CSP also needs to store a private key. In Zhang and Lai's schemes, the size of the ciphertext is linearly related to the number of keywords. Since Liang's scheme is based on the KP-ABE, the size of ciphertext is positively related to the size $|S|$ of the attribute set. Our scheme is based on the CP-ABE, but the ciphertext length is a fixed value. Although the CSP requires an additional private key in our scheme, the storage overhead of the CSP is a fixed value $\mathcal{O}(1)|g|+\mathcal{O}(1)|g_T|$. The storage overhead of each receiver is associated with the trapdoor in the four scenarios. As with the situation of the ciphertext, the size of the trapdoor is linearly related to the number of keywords in Zhang and Lai's schemes. The size of trapdoor is positively related to the row number l of $\boldsymbol{M}$ in Liang's scheme. The size of the trapdoor is a fixed value $\mathcal{O}(1)|g|$ in our scheme. So the receiver has a smaller storage burden in our scheme.

Table 2. Comparison of Storage Overhead

Schemes	AA (MSK)	Owner (All PK)	CSP (CT)	Receiver (Trapdoor)
Zhang[6]	-	$\mathcal{O}(n)\|g\|$	$\mathcal{O}(n)\|g\|+\mathcal{O}(1)\|p\|$	$\mathcal{O}(n)\|g\|$
Lai [7]	-	$\mathcal{O}(n)\|g\|+\mathcal{O}(1)\|g_T\|$	$\mathcal{O}(n)\|g\|+\mathcal{O}(1)\|g_T\|$	$\mathcal{O}(l)\|g\|$
Liang[12]	$\mathcal{O}(1)\|g\|+\mathcal{O}(1)\|p\|$	$\mathcal{O}(n)\|g\|+\mathcal{O}(1)\|g_T\|$	$\mathcal{O}(\|S\|)\|g\|+\mathcal{O}(1)\|\lambda\|+\mathcal{O}(1)\|g_T\|$	$\mathcal{O}(l^2)\|g\|$
Ours	$\mathcal{O}(n)\|g\|+\mathcal{O}(1)\|N\|$	$\mathcal{O}(n)\|g\|+\mathcal{O}(1)\|g_T\|$	$\mathcal{O}(1)\|g\|+\mathcal{O}(1)\|g_T\|$	$\mathcal{O}(1)\|g\|$

4.3 Computational Efficiency

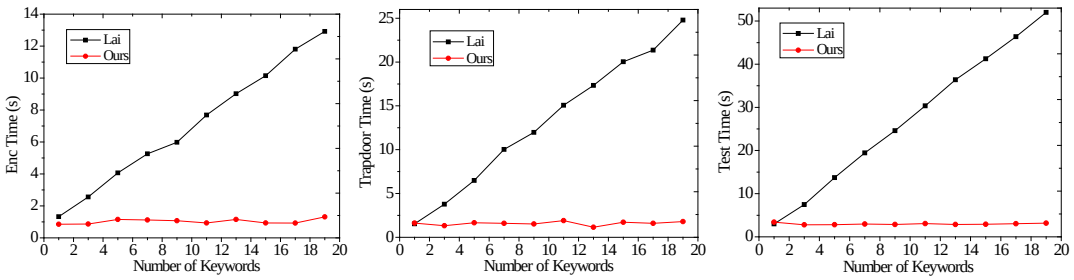
Experiment Setup: We conducted the experiment on a 64-bit Ubuntu 14.04 operating system with an Intel CoreTM i5-6200U (2.3 GHz) processor and 8 G RAM. The experimental code uses the Pairing-based Cryptography Library (PBC-0.5.14) and cpabe-0.11 to implement the schemes. We employ the 160-bit elliptic curve group in the hyper-singular curves $y^2 = x^3 + x$ based on 512-bit finite fields. Specifically, the pairing operation time of the PBC library is approximately 1.27s, and the exponential operation times of G and G_T are approximately 0.33s and 0.18s, respectively.

Table 3 shows a comparison of the execution time of Encryption, Trapdoor and Test. Because the time required for the multiplication operation is significantly smaller than the exponential operation, the multiplication time is omitted in **Table 3**. Assume that the keywords in the access structure are used for the phase of Test. The execution times of Encryption, Trapdoor and Test increase with an increase in the number of keywords in Lai's scheme. The time complexity of the Encryption, Trapdoor and Test in our scheme is a constant order. But Lai et al.'s scheme supports arbitrary monotone boolean predicates based on the linear secret sharing schemes (LSSS). And LSSS is a strong expressive access structure. Our scheme supports AND-gate access structure with multiple values. And the AND-gate access structure is a weak expressive access structure. In other words, Lai et al.'s scheme supports AND, OR and the threshold of keywords. Our scheme only supports AND of the keywords.

Table 3. Comparison of Computational Efficiency

Operation	Time (s)	Lai [7]			Our scheme		
		Encryption	Trapdoor	Test	Encryption	Trapdoor	Test
Exp in G	0.33	$2n+1$	$4l$	0	2	4	0
Exp in G_T	0.18	1	0	l	1	0	1
Bilinear Pairing	1.27	0	0	$2l$	0	0	2
Total Time		$0.66n+0.51$	$1.32l$	$2.72l$	0.84	1.32	2.72

The simulation experiment system is built, and the operation time is tested in the system. In the simulation process, the relationship between the number of keywords and the execution time of Encryption, Trapdoor and Test is tested. We select a 1 MB file and encrypt the file with a different number of keywords. We test the execution time of the Encryption, Trapdoor and Test processes as the number of keywords changes (from 1 to 20). All simulation results are the mean of 30 trials. According to this method, Lai's scheme is simulated. The relationships are shown in **Fig. 1**. The horizontal axis represents the number of keywords in the search, and the vertical axis represents the execution time of Encryption, Trapdoor and Test.



(a) Encryption Time (b) Trapdoor Time (c) Test Time

Fig. 1. Comparison of Time with Different Number of Keywords

Fig. 1(a) plots the executed time of Encryption, which is executed by the data owner. **Fig. 1(b)** shows the execution time of Trapdoor. **Fig. 1(c)** shows the execution time of Test, which is executed by the CSP. In Lai's scheme, the executed times of Encryption, Trapdoor and Test increase with the number of keywords. When the number of keywords changes from 1 to 20, the executed time is approximately linearly. But the executed time of Encryption, Trapdoor and Test is a fixed value in our scheme, which is not related to the number of keywords. It is consistent with the theoretical analysis in **Table 3**.

Through the experimental comparison, our scheme is superior to other schemes in terms of security. Our scheme is proved adaptively secure based on the simple assumption DBDH in the standard model. In terms of storage, the public key length of our scheme and other schemes is similar. But the sizes of ciphertext and trapdoor are smaller than other schemes, they are the fixed value in our scheme. In terms of efficiency, our scheme is constructed based on the composite order group, which is less efficient than the scheme based on the prime order group. But the operations of the Encryption, Trapdoor and Test are constant level, regardless of the number of keywords. When the keywords are more, the advantage of our scheme will be highlighted.

5. Conclusion

In this paper, we propose an efficient conjunctive keyword search scheme without a secure channel for the cloud storage environment, which is called SE-CKS. This scheme implements conjunctive keyword search based on CP-ABE. At the same time, we propose an efficient mechanism for removing the secure channel and resisting off-line keyword guessing attacks. The storage overhead of the CSP and DU are the fixed value and the amount of calculations of Encryption, Trapdoor and Test are constant level, regardless of the number of keywords. This scheme is proved adaptively secure based on the DBDH assumption in the standard model. Finally, the results of theoretical analysis and experimental simulation show that the proposed scheme has advantages in security, storage overhead and efficiency, and it is more suitable for practical application.

References

[1] Q. Yan, R. Yu, Q. Gong and et al, "Software-defined networking (SDN) and distributed denial of service (DDoS) attacks in cloud computing environments: A survey, some research issues, and challenges," *IEEE Communications Surveys and Tutorials*, vol. 18, no. 1, pp. 602-622, 2016. Article (CrossRef Link).

[2] F. Chen, T. Xiang, Y. Yang and et al, "Secure cloud storage meets with secure network coding," *IEEE Transactions on Computers*, vol. 65, no. 6, pp. 1936-1948, 2016. Article (CrossRef Link).

[3] D. X. Song, D. Wagner and A. Perrig, "Practical techniques for searches on encrypted data," in *Proc. of IEEE Symposium on Security and Privacy*, pp. 44-55, May 14-17, 2000. Article (CrossRef Link).

[4] D. Boneh, G. D. Crescenzo, R Ostrovsky and et al, "Public key encryption with keyword search," in *Proc. of International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 506-522, May 2-6, 2004. Article (CrossRef Link).

[5] P. Golle, J. Staddon and B. Waters, "Secure conjunctive keyword search over encrypted data," in *Proc. of the 2th International Conference on Applied Cryptography and Network Security*, pp. 31-45, June 8-11, 2004. Article (CrossRef Link).

[6] B. Zhang and F. Zhang, "An efficient public key encryption with conjunctive-subset keywords search," *Journal of Network and Computer Application*, vol. 34, no. 1, pp. 262-267, 2011. Article (CrossRef Link).
[7] J. Lai, X. Zhou, R. H. Deng and et al, "Expressive search on encrypted data," in *Proc. of the ACM SIGSAC symposium on Information, computer and communications security*, pp. 243-252, May 8-10, 2013. Article (CrossRef Link).
[8] J. Baek, R. Safavinaini and W. Susilo, "Public key encryption with keyword search revisited," in *Proc. of International Conference on Computational Science and Its Applications*, pp. 1249-1259, June 30-July 3, 2008. Article (CrossRef Link).
[9] J. Byun, H. Rhee, H. Park and et al, "Off-line keyword guessing attacks on recent keyword search schemes over encrypted data," in *Proc. of Workshop on Secure Data Management*, pp. 75-83, September 10-11, 2006. Article (CrossRef Link).
[10] H. Rhee, W. Susilo and H. Kim, "Secure searchable public key encryption scheme against keyword guessing attacks," *IEICE Electron. Express*, vol. 6, no. 5, pp. 237-243, 2009. Article (CrossRef Link).
[11] Y. Yang and M. Ma, "Conjunctive keyword search with designated tester and timing enabled proxy re-encryption function for e-health clouds," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 4, pp. 746-759, 2016. Article (CrossRef Link).
[12] K. Liang and W. Susilo, "Searchable attribute-based mechanism with efficient data sharing for secure cloud storage," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 9, pp. 1981-1992, 2015. Article (CrossRef Link).
[13] L. Cheung and C. Newport, "Provably secure ciphertext policy ABE," in *Proc. of the 14th ACM conference on Computer and communications security*, pp. 456-465, October 29-November 02, 2007. Article (CrossRef Link).
[14] J. Bethencourt, A. Sahai and B. Waters, "Ciphertext-policy attribute-based encryption," in *Proc. of IEEE symposium on security and privacy*, pp. 321-334, May 20-23, 2007. Article (CrossRef Link).
[15] M. Abdalla, M. Bellare, D. Catalano and et al, "Searchable encryption revisited: Consistency properties, relation to anonymous IBE, and extensions," in *Proc. of the 25th Annual International Cryptology Conference*, pp. 205-222, August 14-18, 2005. Article (CrossRef Link).
[16] L. Fang, W. Susilo, C. Ge and et al, "Public key encryption with keyword search secure against keyword guessing attacks without random oracle," *Information Sciences*, vol. 238, no. 7, pp. 221-241, 2013. Article (CrossRef Link).
[17] K. Emura, A. Miyaji, A. Nomura and et al, "A ciphertext-policy attribute-based encryption scheme with constant ciphertext length," *International Journal of Applied Cryptography*, vol. 2, no. 1, pp. 46-59, 2010. Article (CrossRef Link).
[18] B. Waters, "Efficient identity-based encryption without random oracles," in *Proc. of the 24th International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 114-127, May 22-26, 2005. Article (CrossRef Link).

Jianhua Wang received a Ph.D. degree from Zhengzhou Information Science and Technology Institute, China in 2008 and became a Full Professor in 2006. His research interests primarily focus on information security. He has authored and co-authored more than 100 journal and conference papers.

Zhiyuan Zhao received an M.S. degree from Zhengzhou Information Science and Technology Institute, Zhengzhou, China in 2015. He is currently pursuing a Ph.D. degree at Zhengzhou Information Science and Technology Institute, China. His research interests include cryptograph theory, especially attribute-based encryption.

Lei Sun received the Ph.D. degree from the Department of Computer and Communication Engineering, Wuhan University in 2003. He became a Full Professor in 2013. His research interests include information security, cloud computing and computer network.

Zhiqiang Zhu received the Ph.D. degree from the Department of Information Security, Wuhan University in 2011 and became the Full Professor in 2011. His research interests lie in information management, cloud computing and access control.