IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 26, 2019, accepted January 28, 2020, date of publication February 11, 2020,
date of current version February 24, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2973325

Secure Fog-Based E-Learning Scheme

ARIJ BEN AMOR[1,3], (Member, IEEE), MOHAMED ABID[2], AND AREF MEDDEB[3]
[1]National School of Engineering of Tunis, University of Tunis El Manar, Tunis 1002, Tunisia
[2]IResCoMath Laboratory, National Engineering School of Gabes, University of Gabes, Gabes 6029, Tunisia
[3]NOCCS Laboratory, National School of Engineering of Sousse, University of Sousse, Sousse 4054, Tunisia

Corresponding author: Arij Ben Amor (arij.benamor@gmail.com)

ABSTRACT Nowadays, several universities and institutions make profit from the information technologies to enhance and develop their educational strategies and attract more learners. Therefore, distance learning (e-learning) and learning-on-the-go are technologies adopted by universities and service providers to afford more flexible education system. In fact, e-learning is gaining popularity worldwide and the number of learners enrolled in on-line courses is growing. This trend is explained mainly by the opportunities provided by Cloud Computing. In the cloud based educational context, the security factor in sharing the educational content is important and poses several security challenges, such as fine-grained access control and security preservation of content learning. Moreover, there is emergence of the new concept of User-Fog-Cloud architecture to bring closer the services to the client. In this paper, a new fog computing e-learning scheme is provided. Specifically, the proposed solution extends learning content from the cloud to the edge of the network. It can improve the efficiency of learning data analysis, reduces the encryption burden in terms of computation cost on user's devices by offloading part of encryption cost to fog servers and provides fine grained access control to learning content by encrypting the course and the exam with different cryptographic techniques like IBBE and CP-ABE . Further, we present a profile matching mechanism that helps teachers to find colleagues within their vicinity in an efficient and secure way. Security analysis shows that our scheme can achieve data confidentiality, fine-grained access control, collusion resistance and unforgeability. Performance evaluations demonstrate the efficiency of our solution, especially in terms of encryption computation costs.

INDEX TERMS E-learning, fog, data sharing, access control, IBBE, CP-ABE.

I. INTRODUCTION

E-learning and smart education represent an emerging and growing areas that allow the rapid integration of smart technologies, smart environment and smart learning and teaching process.

Since the cloud computing is the main paradigm which future of education lays on, teachers and students take profit from its advantages to deliver more effectively learning content within an integrated environment.

Distance e-learning breaks down the barriers of traditional education with classical attendance groups and presents a clear benefit by giving the ability to access educational content anytime and anywhere. Furthermore, the e-learning's enrolled students outnumber the common university's students.

E-learning includes all kinds of educational technology in learning and teaching. E-learning is composed of online education, computer-aided instruction, virtual education, m-learning, virtual learning environments and digital educational collaboration [11].

Since cloud computing is enabling people to control and access data, education ministries can store electronic teaching content for institutions on the cloud servers. This will not only make it possible for students to use online educational content but they will also be able to access data at home and wherever they are. In fact, this will lead to ubiquity of e-learning.

Providing a low latency, extending the data storage capacities and data analyses facilities was the trigger point to design fog computing, which is supported by big software development companies like Cisco Systems, Dell, Intel and Microsoft through *OpenFog Consortium* [1].

In e-learning context, fog computing is not meant to replace the cloud, but rather to complement it and extend its services. As a matter of fact, both fog and cloud share the same resources and same mechanisms and attributes.

The three-tier architecture which includes cloud, fog and edge computing is typical for smart learning environments.

The associate editor coordinating the review of this manuscript and approving it for publication was Jose Saldana.

In this three-tier architecture, learning applications and data sharing can take profit from the advantages of each technology. In fact, learning contents can be moved and be analyzed across this three-tier architecture where the cloud and the fog may help control and manage data transmitted to/from edge users.

Fog based e-learning paradigm concerns teachers as well as students. The main gains for teachers are the flexibility of sharing content, the cooperation and collaboration among teachers and between teachers and students. Otherwise, when students are concerned, fog-based e-learning can provide location aware service and low latency content acquisition due to the geographical distribution and computational power of fog computing.

To protect the on-line educational materials, administrators should prevent potential threats to uploaded data such as unauthorized access by illegible outside users and improper use by the cloud storage server.

For instance, fine-grained data access control is necessary for fog and cloud-based data sharing. According to their attributes, users can be assigned different levels of access privileges. Data owners can make and enforce fine-grained access policies themselves instead of trusting and being dependent on cloud service providers [28].

The fog environment can provide strong social interconnection functions since teachers can communicate with other teachers and students. Specially, they can find colleagues with the same interests and study fields.

In order to protect data confidentiality and availability in e-learning systems, encryption techniques can help to reach these goals.

In this work, a secure and efficient fog based e-learning scheme is introduced. Our contributions are summed up as follows.

We integrate fog computing into e-learning system to reduce the latency of the provided e-learning services. We provide also fine-grained education data sharing through the re-encryption of educational data for suitable students.

We preserve the integrity of data during data sharing. Only authorized users can decrypt the education data ciphertext.

In addition, the sensible data like exams is preserved from rogue fog server without attributes that satisfy the access policy defined by teachers.

We design a combined ABE-IBBE scheme with low encryption cost supported by connected devices.

We categorize the shared information into courses and exams. Courses are encrypted with the identities of students defined by the fog server after verifying the legitimate subscribed students. But, exams are encrypted with the access policy defined by teachers according to knowledge needs, course fulfillment and score reached.

Security analysis shows that our scheme can achieve data confidentiality, fine-grained access control, collusion resistance and unforgeability. Performance evaluations demonstrate the efficiency of our solution, especially in terms of encryption computation costs.

The rest of the paper is organized as follows. We introduce related work in section II. Then, the preliminaries are pre- sented in section III. Next, we introduce a detailed construction of our proposed scheme in section IV. We present the security analysis and performance evaluation in section V and VI. Finally, we end the paper with a conclusion.

II. RELATED WORK

In this section, we present some existing solutions found in the literature. We start by highlighting the cryptographic based schemes in general then we discuss the cloud based schemes in e-learning solutions.

A. CRYPTOGRAPHIC BASED SCHEMES

Currently, there are many techniques utilized to pro- vide data security and access control in cloud based systems, such as Public-Key Encryption (PKE), Identity-Based Encryption (IBE), Identity-Based Broadcast Encryption (IBBE) [2] and Attribute-Based Encryption (ABE) [3]. In an IBBE system, broadcaster can dynamically select a specific group of users, and then encrypts the message thus only the selected users can decrypt it. In an ABE system, secret key and ciphertext are associated with a set of attributes or an access policy to reach fine-grained access control on encrypted data.

In order to support expressive conditions rather than keywords, Xu *et al.* [4] proposed a conditional Identity-Based Broadcast Proxy Re-Encryption PRE scheme in cloud email context, which can transform an IBBE ciphertext into another IBBE ciphertext if the condition keyword is satisfied. In the same context, Liang *et al.* [5] proposed an attribute-based proxy re-encryption scheme in which if the original access policy is satisfied, the disseminator can convert a ciphertext under an access policy to another ciphertext under a new access policy. In the healthcare context, Huang *et al.* [6] deployed an attribute based conditional data re-encryption construction that permits the authorized doctors who satisfy the pre-defined conditions in the ciphertext to convert a ciphertext into a new ciphertext for a specialist using IBE scheme in the cloud platform.

In order to provide data sharing scheme in OSN (Online Social Networks) scheme, Huang *et al.* [7] adopted attribute-based conditional PRE to guarantee that only the authorized data disseminators, whose attributes satisfy access policy, can spread information into their own social space. Indeed, the data owner enforces the access control over the disseminated ciphertexts in a fine-grained manner through the use of re-encryption key based on IBBE.

For e-examinations and e-assessment applications, the authors in [8] presented a cryptographic scheme that fulfills several security properties such as authenticity, secrecy and anonymity. This solution used Public Key Infrastructure (PKI) to authenticate the participating entities, and assumes the existence of an examination center controlled by a supervisor for monitoring the examinees. The scheme of Khlifi and El-Sabagh [34] was dealing with e-assessment issue where authors integrate available databases authentication technologies in conjunction with e-learning environments for controlling unethical behavior during e-assessment process. The Fine-grained access control is granted by the use of challenging questions as knowledge based technique that rely on student profile, course activities, etc.

B. CLOUD BASED E-LEARNING SCHEMES

Nowadays, teaching and learning methods are taking advantages of the emerging technologies that provides efficient interactions between teachers and students [9].

Cloud based solutions are proven to increase the availability of educational services such as the sharing of courses and documents across teachers and between teachers and students, as well as the reused pedagogic tools for collaboration [10].

Many traditional leading universities are integrating cutting-edge technologies like e-learning, distance learning, web based learning, in their instructional methods which attract a growing number of learners. Universities are becoming blended by giving traditional lectures and offering courses to distant learners through the usage of digital platforms [11].

Many leading universities open up access to their courses; the *Massive Open Online Course (MOOC)* gains its popularity in the whole higher education sector [12]. It is an online course targeting on large-scale interactive participation and open access via the Web. Thus, *MOOC*s are becoming an important supplement to the traditional distance education [13]. Despite the fact that *MOOC* can help non subscribed universities' students to get a certificate of e-learning, the enrolled students need to gets marks in the course to get their diploma, which is not the case treated by *MOOC*.

In [14], when using the cloud computing based e-learning environment, authors propose to encrypt and decrypt data stored in the cloud data centers with minimum replication. They encrypt the messages using *Data Encryption Standard (DES)*, and then encode the encrypted messages using Reed Solomon code in the data centers without considering the flexibility of access for the user side.

In the work [15], the authors observed that existing systems designed to support distance e-learning are not sufficient for the needs of learning in a cross-institutional collaborative environment. So, to cope with the challenge of digital information sharing in multi-university e-learning environments, they propose a new rule-based framework to identify and address issues of content sharing in such environments via the use of *Role-Based Access Control (RBAC)* management.

In the last years, higher educational institutions have significantly increased their usage of *Virtual Learning Environments (VLEs)* [16]. The development of full distance learning universities [17], where teaching and learning are completely provided through a *VLE*, offers both video and audio lectures as well as interactive and collaborative instruments such as eportfolios, concept maps, wikis, etc.

In [32], authors suggest a model based on fog computing, to access contents of a *VLE*. This model helps bringing learning contents and applications closer to the necessities of students and instructors. Moreover, virtual laboratories based on cloud computing could benefit from fog computing technologies which provide both delivery of augmented learning course to users entities and remote control of smart devices located in virtual lab [18]. In the latter research, authors discuss fog computing reliability strategies to assure higher levels of confidence in e-learning fog based environment. These e-learning methods involve the usage of cryptographic methods to increase the overall dependability [19]. In [20], authors mentioned that e-learning offers advantages in terms of security when the learning data that concern courses, exams, teachers and students is stored and queried on/from separate servers which can be applicable in a fog computing environment.

Considering the main security-related issues in e-learning, a novel trustworthiness-based methodology is suggested by [21] to increase data security in computer-supported collaborative learning environments.

Moreover, for the smart campus model, authors in [22] have proposed a smart campus architecture that integrates the use of cloud computing and IoT. Their layered architecture aims at ensuring a high level of security as well as high data confidentiality.

Another example that provides services to facilitate students' study procedures and keeps pace with the developments in the deployment of e-learning services is the *King Khalid University (KKU)* [23]. In this university, they made it easier for students to obtain educational services from anywhere and at any time from their teachers and the e-learning Center. Students use *Username* and *Password* to access the cloud based e-learning system.

Despite its beneficial contributions on distance learning environment, the aforementioned literature did not take into account the security matter and the flexibility in accessing the content learning. *MOOC*s, *VLE* and smart campus depend on traditional access techniques such as login/password and *RBAC* whenever a user wants accessing a course stored on a remote cloud based system. These methods cannot fit with the concept of learn-on-the-go which allows learners to attend live classes on-the-go.

To overcome these problems, a secure fog-assisted e-learning system was designed. It can provide fine-grained education data sharing through the re-encryption of educational data for suitable students.

C. COMPARISON OF EXISTING SCHEMES AND MOTIVATION

1) COMPARISON OF EXISTING SCHEMES

The table 1 illustrates the different functionalities deployed in re-encrypted data sharing schemes. We conclude that the secure solutions are not applied to e-learning domain while the ones concerning e-learning dispose of a weak security mechanism. To overcome this lack, we define a new scheme which provides data sharing with strong encryption and re-encryption phases aided by the fog server.

TABLE 1. Comparison of existing schemes.

	Initial data confidentiality	Conditional Re-encryption	Re-encryption entity	Re-encrypted data confidentiality	Application Domain	Profile matching
[4]	IBBE	Keyword	Cloud server	IBBE	Cloud email	No
[5]	ABE	No	Disseminator	ABE	cloud	No
[6]	IBBE	Access-policy	End user (doctor)	IBE	e-health	Yes
[7]	IBBE	Access-policy	Disseminator	IBBE	online social networks	No
[8]	PKE	Full Batch	mix network	PKE (El-Gamal)	e-examinations	No
[14]	DES	No	Data centers	Reed Solomon code	e-learning	No
[15]	RBAC	No	No	No	e-learning	No
[23]	password	No	No	No	e-learning	No
[24]	PKE	No	No	No	Cloud email	Yes
[34]	password	No	No	Behavioral Challenge	e-assessment	No
Our solution	ABE/ PKE	Semi-trusted PKE	Fog node	IBBE	e-learning	Yes

2) MOTIVATION

Learning on the go is becoming more and more common, and with internet available everywhere, it's a great opportunity for cloud server providers and universities to offer a mobile learning and assessment service.

Our solution promote the concept of learn-on-the-go [29], [30] which allow learners to attend live classes on-the-go on their smartphones and tablets and develop their capacity to learn at anytime and from anywhere.

The integration of fog computing into e-learning system can be an extension of a full service offered online and reduce the education data processing cost for users.

Responsive optimization and availability of course content means that students who start at home and decide to finish courses when taking the train to reach their university will continue the course without interruption.

The use of fog servers is then a way to provide course-on-the-go and drive real time online collaboration with learners by sharing learning files (audios, videos) during a live class. It also helps dealing with live teaching (chatboards) and registered courses. Moreover, fog servers provide data access control which help financial management for the paid services.

Also, we aim to reduce the encryption burden in terms of computation cost on the user side when dealing with interactive learning.

Before describing the steps of our solution, we present the preliminaries methods deployed.

III. PRELIMINARIES

This section revisits the preliminaries used to construct the Secure Data Sharing in Fog-based e-learning scheme.

A. BILINEAR PAIRING

Bilinear Pairing: Let G_1 and G_2 be two multiplicative cyclic groups of prime order p and let g be a generator of G_1. The bilinear pairing $e : G_1 \times G_1 \rightarrow G_2$ has the following properties:

- Bilinearity: For all $u, v \in G_1$ and $a, b \in Z_p$, we have $e(u^a, v^b) = e(u, v)^{ab}$.
- Non-degeneracy: The generator g should satisfy $e(g, g) \neq 1$.
- Computability: For any $u, v \in G_1$, $e(u,v)$ can be efficiently computed.

B. IDENTITY-BASED BROADCAST ENCRYPTION (IBBE)

The IBBE can be considered as an extension of the IBE, which allows one to encrypt a message once for many receivers. The definition of IBBE is as follows [2].

- Setup (K, N): The setup algorithm takes as input a security parameter K and the maximal size N of a set of receivers for an IBBE encryption, and outputs a pair of public/master secret key (PK, MK).
- Extract (MK, ID): Given the master secret key MK and a user's identity ID as input, the extract algorithm outputs a secret key SK_{ID} for user.
- Enc(PK, M, U): Given as input the public key PK, a message M and a set U of receivers' identities, the encryption algorithm outputs a ciphertext CT for U.
- Dec(PK, CT, ID, SK_{ID}, U): The decryption algorithm takes as input the public key PK, a ciphertext CT, an identity ID and the secret key SK_{ID} and the set U of receivers, the algorithm outputs the message M if $ID \in U$.

C. ACCESS TREE

Let T be a tree representing an access policy. Each non-leaf node x of tree represents a threshold gate and and each leaf node describes an attribute. Let $numx$ denote the number of children of node x, and kx is the threshold value, then $1 \leq kx \leq numx$. The threshold value is an AND gate if $kx = numx$, and an OR gate if $kx = 1$.

For each leaf node x of tree, we have the threshold value $kx = 1$, and denote $attrx$ as its associated attribute.

For the description of the access tree structure, the following functions are defined: $parent_x$ represents a parent of the node x in the tree, $attrx$ is the attribute of the leaf node x, and $index_x$ returns a uniquely assigned number associated with node x.

Let T be a tree with a root node R, and let T_x be a subtree rooted at the node x in the access tree. If a set of the attributes S satisfies Tx, we denote it as $Tx(S) = 1$. The value of $Tx(S)$ is computed recursively as follows:

- If x is a non-leaf node, we evaluate $Tn(S)$ for all children n of node x, and returns $Tx(S) = 1$if and only if at least kx children return 1.
- If x is a leaf node, then $Tx(S) = 1$ if and only if $attrx \in S$.

D. CIPHERTEXT-POLICY ATTRIBUTE-BASED ENCRYPTION (CP-ABE)

A ciphertext-policy attribute-based encryption *CP-ABE* system for access policy T consists of the following four algorithms [3].

- Setup (K): Given the security parameter K as input, the setup algorithm outputs a public key *PK* and a master secret key *MK*.
- KeyGen (*PK, MK, S*): Given the public key *PK*, the master secret key MK, a set S of attributes as input, the key generation algorithm outputs a secret key *SK*.
- Enc (*PK, M, T*): Given the public key *PK*, a message *M* and an access policy *T* as input, the encryption algorithm outputs a ciphertext *CT*.
- Dec (*PK, SK, CT*): Given the public key *PK*, an attribute key *SK*, a ciphertext *CT* with an access policy *T* as input. If $S \in T$, the decryption algorithm outputs the message M.

E. PRIVATE EQUALITY TEST

A private equality test scheme [24] is a collection of the following six algorithms:

- Setup (K) takes as input a security parameter K, chooses bilinear parameters and two hash functions H_1, H_2, and outputs the system parameters *params* and the public key *PK*.
- KeyGen (*params*): It takes the public parameter params as input, and outputs secret key (*SK*).
- Encrypt (*M, PK*): It takes a message *M* and public key *PK* as input and outputs a ciphertext *CT*.
- Decrypt (*CT, SK*): It takes a ciphertext *CT* and secret key *SK* as input and outputs a plaintext *M*.
- Authorization Type-2 (*params, CT, SK*) takes as input system parameters, the data owner's ciphertext CT and its secret key SK. It outputs a trapdoor *td*.
- Test-2 (*ct1, td1, ct2, td2*) takes as input system parameters, two ciphertexts, and two trapdoors. It outputs 1 if *ct1* and *ct2* contain the same message and zero otherwise.

IV. PROPOSED SCHEME

In this section, we present the system model and define the security model. In our scheme; the shared e-learning data is categorized into two types: data type1 (M1) which represents the course materials including lecture slides, video lectures, journal articles, documents and podcasts and data type2 (M2) representing the exam materials including tests, quizzes, assessments and exams. The description of the system's symbols is given in table 2.

TABLE 2. Description of system symbols.

Symbols	Description
$\boldsymbol{K, MK, PK}$	The master secret key and system public key
$\boldsymbol{IDfog, Lfog, PK_{fog}, SK_{fog}}$	The identity, location, public and private key of FS
$\boldsymbol{ID_t, sk_t, Lt, td_{IDt}, SLP}$	The identity, secret key, location, trapdoor and social learning profile of teacher
$\boldsymbol{ID_{st}, sk1, sk2}$	The identity and two secret keys of student
$\boldsymbol{M1}$	Data type1 corresponding to Course Material
$\boldsymbol{M2}$	Data type2 corresponding to Exam Material
$\boldsymbol{U}$	The set of students' identities
$\boldsymbol{GIDst}$	The symmetric key between RS and students
$\boldsymbol{CT1}$	The ciphertext related to data type 1
$\boldsymbol{CT2}$	The ciphertext related to data type 2
$\boldsymbol{T}$	The access tree of *CT2*
$\boldsymbol{S}$	The set of attributes for T
$\boldsymbol{a, k_2, v, v_1, v_2}$	Random numbers generated by the teacher
$\boldsymbol{X}$	node from the access tree T
$\boldsymbol{N}$	Child of node x
$\boldsymbol{Y1}$	The set of leaf nodes in T
$\boldsymbol{k_{verif}}$	verification key generated by the cloud
$\boldsymbol{k_{sign}}$	signing key of FS
$\boldsymbol{Sign}$	Operation of signing run by FS using k_{sign}
$\boldsymbol{Verify}$	Operation of verifying of signature run by students using k_{verif}
$\boldsymbol{N_a}$	Number of attributes
$\boldsymbol{N_{st}}$	Number of students
$\boldsymbol{n_1}$	A nonce generated by student when requesting data

A. SYSTEM MODEL

In Fig. 1, we illustrate the new secure fog based e-learning architecture along with the four entities: cloud servers, fog servers, teacher and students.

- ***Registration Server (RS)***: the cloud server is composed of an intelligent agent named the Registration Server RS which is a fully trusted entity. RS is in charge of generating system parameters, key pair for fog servers as well as private and attribute keys for each student.
- ***Fog Servers (FS)***: FSs are geo-distributed servers deployed at the local areas of users that offer a variety of services including reducing latency, real-time applications and confidentiality preservation.

Once receiving data, the fog server decrypts *M1* with the secret shared key and analyses its contents. Then FS

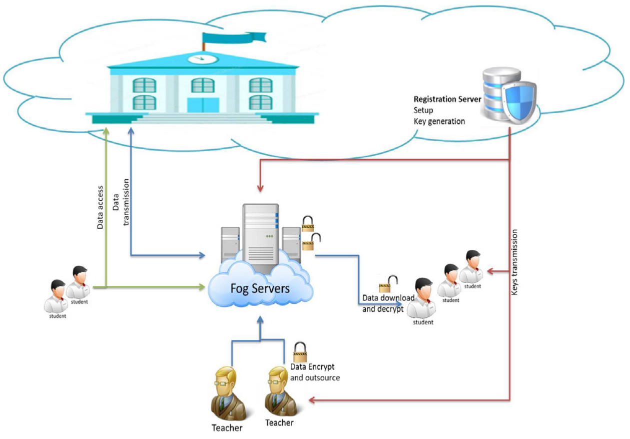

FIGURE 1. Secure fog based e-learning architecture.

re-encrypt the shared *M1* using *IBBE* algorithm. Only legitimate students having registered identities (ex e-mail @) can access this data and profit from the outsourced course.

For example, if information analyzed by the fog server concern java course, the list of students' identities contains the registered computer sciences college's students defined by the fog server combined with attributes defined by teacher such as java certificate obtained, more than 10 degree in programming module, less than 2 times passing exam. The fog server then stores the re-encrypted *M1* and *M2* in the cloud and maintains local storage for data pre-processing.

In addition, the FS could execute the profile matching algorithm of two users without knowing the specific related information. The FS then sends the matching result to the corresponding users aiming to establish social relationship.

- ***Teacher***: he/she is the generator of data and has the right to access, modify and delete previous version of data files.

Teacher aims to share educational data with authorized students. He/she encrypts learning files (courses, exams, etc) with the attributes of a specific group of student (classroom) and uploads all files to third party servers.

Teacher categorizes his/her shared data into two types: *M1* which contains the course with related information (title, chapters, modules …) and *M2* that contains the exam (or test evaluation) related to the course *M1*.

M1 is encrypted using a shared key with the fog server and *M2* is encrypted using ABE with coarse attributes defined by the teacher.

Simultaneously, teachers with the same social learning profile can generate trapdoors and form social relationships according to their specialties.

- ***Students:*** they are the ciphertexts' receivers and are able to decrypt the data according to their access rights: they can decrypt *M1* if they are the intended receiver defined by fog server and they can access *M2* if their attributes satisfy the access policy defined by the teacher.

B. SECURITY MODEL

In our scheme, the RS as a registration authority can be totally trusted strictly issuing secret keys to each student according to its attributes.

The fog servers are assumed to be honest but their system's security threats are different and depend on the type of data: the fog servers provide pre-processing operation for *M1* (course) in order to define the access list of authorized students needed for the encryption of *M1*. But they are curious about the content of *M2*(exams).

We assume that some malicious students will try to obtain unauthorized e-learning data especially accessing the exam by colluding with other students.

Based on the above security assumption for each entity, the e-learning sharing scheme in this article should be designed to possess the following security features.

1) DATA CONFIDENTIALITY

The unauthorized students that are not the legitimate receivers defined by the teacher should be prevented from accessing the course and the exam. The student should not be able to access the re-encrypted course if he/she is not registered as a member of authorized students' list defined by the fog server.

2) FINE-GRAINED ACCESS CONTROL AND SECURITY PRESERVATION OF CONTENT LEARNING

We should provide fine grained access control for learning data generated by teachers. We consider two security course levels:

- Security Course Level I: In this level, students can access only to the courses unless their identities are included in the list defined by the fog server.
- Security Course Level II: In this level, students can access to the courses and the exams unless they satisfy these two conditions: their identities are included in the list defined by the fog server and their attributes satisfy the access policies defined by teachers to acquire the related exam.

Since the fog server can preprocess the course's content, we aim to guarantee Security Course Level I on the fog server, and Security Course Level II on students as end users.

3) COLLUSION RESISTANCE

If all of the students' attributes in the set cannot satisfy the access policy in the ciphertexts alone, the access of ciphertext should not successful.

C. SYSTEM DEFINITION

Based on the system model, our scheme consists of the following algorithms.

1) SYSTEM SETUP

Algorithm 1 describes the system setup and is executed by the registration server. It takes the security parameter $K \in Z_p$

Algorithm 1 Setup (K)

```
Choose two multiplicative groups $G_1$ and $G_2$ of prime order $p$;
e: $G_1 \times G_1 \rightarrow G_2$ bilinear map
Choose three random exponents $\gamma, \lambda, \beta \in Zp$ such that $\gamma \neq \beta \neq \lambda \neq 0$; g, h$\in G_1$
N: max number of receivers (students)
Select cryptographic hash functions
        $H_1$:{0,1}* $\rightarrow Zp^*$;
        $H_2$:{0,1}* $\rightarrow G_1$;
        $H_3$: $G_2 \rightarrow G_1$;
The public key is published as:
$PK = ( H_1; H_2; H_3; g^{\gamma}; e(g, h); h^{\beta}; h^{\lambda}; h, h^{\gamma 0}, \ldots, h^{\gamma N})$
The master key is $MK = (g, \beta, \gamma)$
```

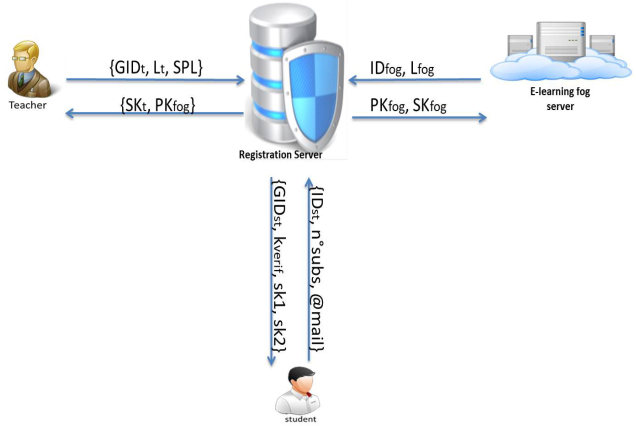

FIGURE 2. Registration and key transmission via secure channel.

and a value l for the identity length as an input, publishes the public parameters PK to all involved entities, and holds the master key MK.

2) REGISTRATION AND KEY GENERATION (ID_{fog}, ID_t, ID_{st}, L_{fog}, L_t) →PK_{fog}, SK_{fog}, sk1, sk2, GID_{st}

The registration of involved entities (fog server, teachers, and students) is also performed by the *Registration Server RS*. The fog server sends its location and identity in order to get two pair of asymmetric keys (PK_{fog}, SK_{fog}) and (k_{sign}, k_{verif}). RS runs $\sigma(1^k)$ [2] to obtain signing key k_{sign} and verification key k_{verif}. Similarly, in order to register, teachers send securely their identities, their *Social Learning Profile (SLP)* and geographical location to the registration server. According to that location, the *RS* finds out the nearest fog servers and sends their public keys (PK_{fog}) to teachers for coming encryption. The registration server sends also a secret key $sk_t = g^{1/(\lambda+H1(IDt))}$ for each registered teacher (see Fig. 2).

Algorithm 2. The *Registration Server* uses the public parameters *PK*, the master key *MK*, and an access policy *T* defined by the teacher to generate a secret key pair *(sk1, sk2)* and a symmetric pre-shared key GIDst for further students' requests. Fig. 2 illustrates the registration of involved entities.

Algorithm 2 Key Generation (*MK, PK, T*) Key Pair (*sk1*, *sk2*)

```
Choose $\gamma, \lambda \in Z_p{}^*$ and compute $sk1 = g^{1/\gamma + H1(IDst)}$;
select random $r_1$ where $r_1 \in Z_p$;
    for each $i \in S$ do
Choose $r_{1i} \in Z_p$ and compute
$D_{1i} = g^{r1} \cdot H(j)^{r1i}; D'_{1i} = h^{r1i}$
    end for
The secret key $sk2$ belonging to $S$ is computed as:
        $sk2 = (D_1 = g^{(1+r1)/\beta}; \forall i \in S : D_{1i}; D'_{1i})$
Broadcast $k_{verif}$ to others to verify the legacy of $FS$
```

Algorithm 3 Encrypt (*M1, M2, PK, T*)

```
Data owners first choose a random $a$, $k_2 \in Z_p$, a random encryption exponent $v \in Z_p$, and two numbers $v_1$ and $v_2 \in Z_p$, where $v = v_1 + v_2$.
Let $AS$ be the access structure represented by $T$ rooted at node $R1$;
Start from the root $R1$ and set $q_{R1}(0) = v_2$;
For each node x in T choose a polynomial degree $q_x$ and set the degree to $d_x = k_x$-1;
    for other nodes $x$ in $T$ do
    Set $q_x(0) = q_{parent(x)}(index(x))$
    Select $d_x$ randomly to define the polynomial $q_x$
    end for
The ciphertext is constructed as follows:
$CT1 = Enc_{PKfog}(M1, v_1)$
$CT2 = \{C0 = Enc_{k2}(M2)$;
    $C1 = h^{\beta v2}; C2 = k_2 \cdot e(g,h)^v$;
    $\forall\, y \in Y1: C_{2y} = h^{qy(0)}$;
    $C'_{2y} = H_1(attr(y))^{qy(0)}\}$
$C'1 = (h^{\lambda} \cdot h^{H1(IDt)})^a$;
$C'2 = a \cdot H_2(SLP) \oplus H_3(e(g,h)^a)$;
The ciphertext is CT = (CT1, $C'1$, $C'2$, CT2)
```

3) ENCRYPT(M1, M2, PK, T)

Algorithm 3 provides the details of the encrypted shared information. It is executed by the data owner (teacher) that takes as inputs the public parameters PK and the access tree structure T then outputs the ciphertext CT.

Teachers categorize their shared information into course and exam data files, and encrypt them with different encryption algorithms. Afterwards, they send the encrypted ciphertext to the fog server.

4) DEC-REENCRYPT(CT, SK_{fog}, PK)

The Dec-ReEncrypt algorithm is run by fog server which decrypts the cipher text *CT1* received from the teacher, pre-process message *M1* with the information stored in its database to obtain convenient students list that match with *M1* (course). Then it reencrypts the ciphertext using the *IBBE* algorithm. We apply the results of Cannetti [25] to construct the *IBBE* algorithm where the fog server uses a strong one-time signature scheme $Sig = (\sigma, Sign, Verify)$. The algorithm takes as input the ciphertext *CT*, its private key SK_{fog} and public key *PK*.

Algorithm 4 Dec-ReEncrypt(*CT, SKfog, PK*)

```
1: decrypts CT1 with its secret key $SK_{fog}$ to obtain M1 and $v_1$.
2: pre-processes M1 to obtain the list of students' identities.
3: stored M1
4: picks a random $k_1 \in \mathrm{Z_p}$,
5: The ciphertext CT1' is constructed as follows:
Re − encrypt $CT1$'
 $CT1' = \{C3 = \mathrm{Enck1}(M1);$
     $C4 = g^{-\gamma v_1}_{;}$
     $C5 = h^{v_1 H_1(k_{verif}) \prod_{ID_{st_{i \in U}}} (\gamma + H_1(ID_{st_i}))}_{;}$
     $C6 = k_1 \cdot e(g, h)^{v_1}; \}$
$CT1'' = (CT1', Sign(CT1', k_{sign}))$
6: the ciphertext is CT'=(CT2; CT1'={C3; C4; C5; C6}; CT1'')
```

5) DECRYPTION (CT′, SK1, SK2, PK, S)

The decryption phase describes the procedure to obtain the shared data *M1*, (*M1, M2*) or ⊥.

a: DECRYPT1 (CT′1, CT″1, SK1, PK, U)

For the ciphertext *CT'1*, the student with identity ID_{st} within the list of $ID_{st}s$ defined by the fog server, runs Decrypt1 (algorithm 5) first to guess k_1 and then retrieves *M1* (course).

b: DECRYPT2 (CT2, SK2, PK, S)

The algorithm 6 is executed by each user (student), which takes as inputs the public parameters *PK*, the secret key pair (*sk1, sk2*) and the ciphertext *CT2*; if their attributes satisfy T, they can guess k_2 and retrieve *M2*.

Note, the Lagrange's coefficients $\Delta_{i,S}$ for $i \in Z_p$ form a set of elements in $\mathrm{Z_p}$ defined as

$$\Delta_{i,S} = \prod_{j \in S, j \neq i} \frac{x-j}{i-j}$$

Note that Algorithm 6 employs a recursive function DecryptNode(), detailed in [3].

DecryptNode(*CT2, SK2, x*) takes the ciphertext CT2, the secret key sk2 which is associated with a set of attributes S, and a node x from the access tree T as input.

6) FULL COURSE ACQUISITION

For students who want to benefit from the full course, fullfill the credit modules learning and pass the learning level, ciphertext *CT2* must be recovered in order to take related exam.

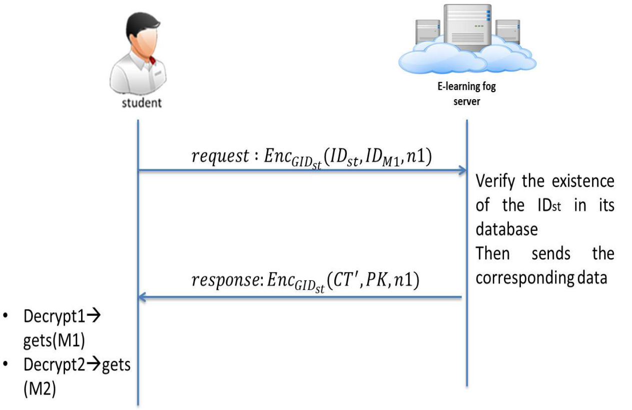
student
E-learning fog server
$request : Enc_{GID_{st}}(ID_{st}, ID_{M1}, n1)$
Verify the existence of the IDst in its database
Then sends the corresponding data
$response: Enc_{GID_{st}}(CT', PK, n1)$
Decrypt1→ gets(M1)
Decrypt2→gets (M2)

FIGURE 3. Full course acquisition.

As presented in Fig. 3, the student must communicate with the fog server by sending a secure request including its identity and the module's identity. FS responds by sending the corresponding ciphertext and PK.

To get the exam, students need to execute both Decrypt1 and Decrypt2 to extract A_1 and A_2.

The basic data flow of our scheme is shown in Fig. 4. The registration server runs *Setup* algorithm to generate system public key and master key. In the *registration* and *key generation* phase, it generates keys for the involved entities (fog server, teacher and student) in the system. At first, teacher, as a data owner, categorizes the shared data into course (*M1*) and exam (*M2*) and runs *Encrypt* algorithm to encrypt *M1* with PKE and *M2* with an access policy *T* for a set of receivers. Then the teacher will outsource encrypted data to the FS. Once receiving the ciphertext from the teacher, the latter runs the *Dec-ReEncrypt* algorithm: it decrypts *M1*, pre-processes the content and defines the access list of authorized students' identities. Then it re-encrypts *M1* for the chosen users having the ability to access the data. The student would send a request of accessing the ciphertext to the FS. After receiving the request, the FS verifies the existence of the student's ID in its database, then, sends the corresponding ciphertexts to the student. If it is the intended receiver, it could be able to run *Decrypt1* algorithm to decrypt the ciphertext and extract *M1* (course). Furthermore, if its attributes satisfy the access policy in the ciphertext corresponding to M2, the student can run *Decrypt2* algorithm and extract *M2* (exam).

D. PROFILE MATCHING

In this subsection, two teachers want to share data between then. The FS determines whether the profile ciphertexts of the two teachers contains the same data according to the authorization type2.

Algorithm 5 Decrypt1 (*CT'1, CT"1, SK1, PK, ID_{st}*)

```
Verify that $\sigma$ is a valid signature of CT1' under the key $k_{verif}$.
1: Verify (CT1", $k_{verif}$ )
2: If it is invalid, output $\perp$.
  else if $ID_{st} \in U$ then student computes
      $A_1 = (e(C_4, h^{\Delta_\gamma(ID_{st},U)}) \cdot e(sk_1, C_5))^{\frac{1}{\prod_{ID_i \in U \wedge ID_i \neq ID} H_1(ID_{sti})}}$
      $= (e(g^{-\gamma v_1}, h^{\Delta_\gamma(ID_{st},U)}) \cdot e(g^{\frac{1}{\gamma+H_1(ID_{st_i})}}, h^{v_1 H_1(k_{verif}) \prod_{ID_{st_i} \in U}(\gamma+H_1(ID_{st_i})}))^{\frac{1}{H_1(k_{verif}) \prod_{ID_{st_i} \in U \wedge ID_{st_i} \neq ID} H_1(ID_{st_i})}} = e(g,h)^{v_1}$
Where $\Delta_\gamma(ID_{st}, U) = \frac{H_1(k_{verif})}{\gamma} \cdot \left(\prod_{ID_i \in U \wedge ID_i \neq ID}(\gamma + H_1(ID_{sti})) - \prod_{ID_i \in U \wedge ID_i \neq ID} H_1(ID_{sti})\right)$
3: Student can compute $k_1$
      $k_1 = \frac{C_6}{A_1} = \frac{k_1 \cdot e(g,h)^{v_1}}{e(g,h)^{v_1}}$
4: Finally student recovers M1 with $k_1$ using symmetric decryption algorithm.
      $M_1 = dec_{k1}(C_3)$
```

Algorithm 6 Decrypt2 (*CT2, SK2, x, S*)

```
1: Compute DecryptNode(CT2, SK2, x)
2: if x is a leaf node then
$z =$   attr$_x$
      if $z \in S$ then
          $\text{DecryptNode}(CT2, SK2, x) = \frac{e(D_{1z}, C_{1z})}{(D'_{1z}, C'_{1z})} = \frac{e(g^{r1} \cdot h(z)^{r_1}, h^{q_y}(0))}{e(h^{r_{1z}, H(z)^{q_z(0)}})} = e(g,h)^{r_1 q_z(0)}$
      else DecryptNode(CT2, SK2, x) = $\perp$
      end if
    else for all node n that are children of x do
    DecryptNode(CT2, SK2, n)
    end if
3:If the access tree $T$ is satisfied by $S$ then
  we set the result of the entire evaluation for the access tree $T$ as
      $Z = e(g,h)^{r1v2}$
  The student can then compute
      $A_2 = \frac{e(C_1, D_1)}{Z} = \frac{e(h^{\beta v_2}, g^{\frac{(r_1+1)}{\beta}})}{e(g,h)^{r_1 v_2}} = \frac{e(g,h)^{v_2(r_1+1)}}{e(g,h)^{r_1 v_2}} = e(g,h)^{v_2}$
4: Recovering CT2 needs the computation of key $k_2$.
      $k_2 = \frac{C_2}{A_1 \cdot A_2} = \frac{k_2 \cdot e(g,h)^v}{e(g,h)^{v_1} \cdot e(g,h)^{v_2}}$ where $v = v_1 + v_2$
5: Finally, M2 (exam) can be retrieved from C0 using a simple symmetric decryption with k2.
      $M_2 = dec_{k_2}(C_0)$
```

1) TRAPDOOR

In order to match the profiles, the teacher t$_i$ generates a trapdoor according to his/her authorization type and with his/her secret key as follows.

$$td_{IDti} = sk_t = g^{1/(\lambda+H1(IDti))}$$

2) TEST

The *FS* runs *Test-2* algorithm with $td_{IDt1} = g^{1/(\lambda+H1(IDt1))}$ and $td_{IDt2} = g^{1/(\lambda+H1(IDt2))}$

The algorithm computes E_1, X_1, E_2, X_2

$$E1 = e(td_{IDt1}, C'1) = e(g^{1/(\lambda+H1(IDt1))}, h^{a1(\lambda+H1(IDt1))})$$
$$= e(g,h)^{a1}$$

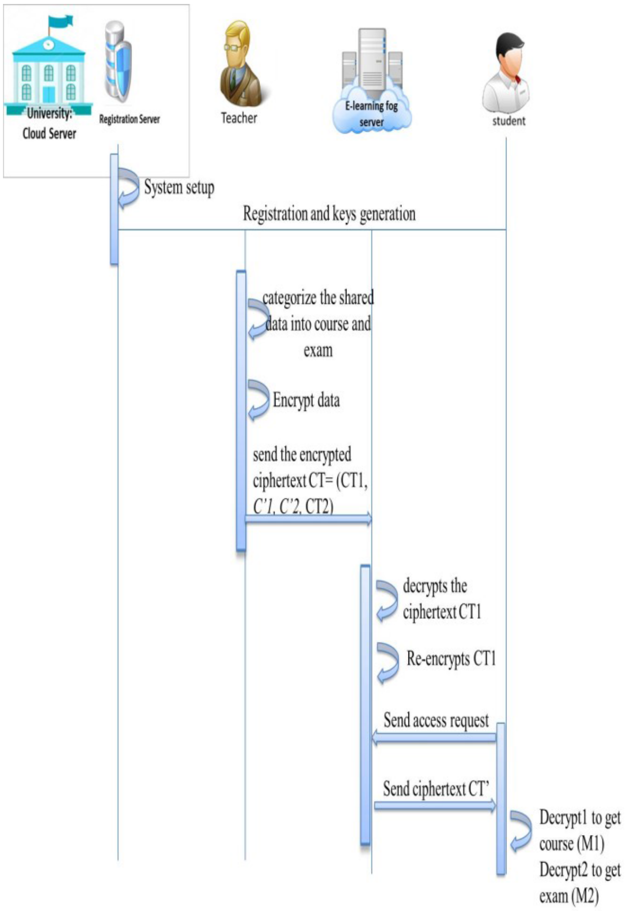
University: Cloud Server
Registration Server
Teacher
E-learning fog server
student
System setup
Registration and keys generation
categorize the shared data into course and exam
Encrypt data
send the encrypted ciphertext CT= (CT1, C'1, C'2, CT2)
decrypts the ciphertext CT1
Re-encrypts CT1
Send access request
Send ciphertext CT'
Decrypt1 to get course (M1)
Decrypt2 to get exam (M2)

FIGURE 4. Data flow of the Fog-based e-learning scheme.

$$
\begin{aligned}
X1 &= C'2 \oplus H(E1) = a_1 \cdot H(SLP_1)\\
E2 &= e(td_{IDt2}, C'1) = e(g^{1/(\lambda+H1(IDt2))}, h^{a2(\lambda+H1(IDt2))})\\
&= e(g, h)^{a2}\\
X2 &= C'2 \oplus H(E2) = a_2 \cdot H(SLP_2)
\end{aligned}
$$

and verifies

$$(E1)^{X2} = (E2)^{X1}$$

If the above equation holds, then FS outputs 1.

V. SECURITY ANALYSIS

The consistency of our proposed scheme is guaranteed by the following Theorem 1, and its security is defined in Theorems 2 and 3.

Theorem 1: For any ciphertext issued from M1 and M2 and for any secret key pair, if the student is the intended receiver, the decryption algorithm can output the plaintext.

Proof: When $ID_{st} \in U$, student can get A1 through the algorithm 5 and thus, recover k1 to decrypt the ciphertext related to *M1*(course). Also, if the student's set of attributes S satisfies T, the intended student can compute A2 through the algorithm 6 and recover k2 to decrypt the ciphertext related to *M2* (exam) successfully.

Theorem 2: The e-learning data in our scheme is encrypted with CP-ABE and IBBE techniques, which are secure against chosen plaintext attack (CPA) since the decisional bilinear Diffie–Hellman (DBDH) assumption holds [4], [33]. Our proposed scheme is CPA secure in the random oracle model with the game among adversary $\mathcal{A}$ and challenger $\mathcal{C}$.

Proof: The adversary $\mathcal{A}$ chooses a challenge access policy T^* and a set U^* of challenge identities to attack.

The challenger $\mathcal{C}$ randomly runs the Setup algorithm to generate a system public key PK and a master secret key MK, and models the hash functions $\mathcal{H}1$, $\mathcal{H}2$ and $\mathcal{H}3$ as three random oracles.

The adversary $\mathcal{A}$ can issue hash query, key generation query and decryption query to challenger $\mathcal{C}$.

In the challenge phase, the adversary sends two pairs of challenge messages $\{m0, m1\}$ for *M1* and $\{m'0, m'1\}$ for *M2* to challenger $\mathcal{C}$. Then, the challenger $\mathcal{C}$ runs encryption algorithm to generate the challenge ciphertext *CT1'** and *CT2**, where b is chosen randomly in $\{0, 1\}$. Finally, the challenger $\mathcal{C}$ sends the two challenge ciphertexts *CT1'** and *CT2** to adversary $\mathcal{A}$.

In the guess phase, the adversary $\mathcal{A}$ outputs a guess $b' \in \{0, 1\}$. With the proof in [4] and the proven security of ABE [33], we can find that the adversary $\mathcal{A}$ can successfully breaks our scheme only if it can break the CPA security of the ABE and IBBE scheme, or can solve the DBDH problem in the decryption query.

Theorem 3: Our scheme is collusion-resistant against colluding students based on the security of ABE.

Proof: Since CP-ABE scheme has been proved semantically secure against chosen-plaintext adversaries [33], this security protection of the shared *M1* ensures that only authorized users whose attribute sets satisfy the access policy can succeed the decryption of *M1*. This means that unauthorized users are not able to access the plaintext even if they have obtained the encrypted data. In fact, unauthorized users cannot earn additional benefits by colluding with fog servers or other unauthorized students which make the scheme resistant against collusion attacks. In particular, preventing collusion between multiple unauthorized students, which can recover $e(g,h)^{r1v2}$ with their secret keys, is achieved by the random blindness in each user's secret key and the unique random number r_j for each attribute in the access policy.

Theorem 4: The proposed secure data sharing scheme in e-learning system achieves data confidentiality, fine-grained data control and unforgeability.

Proof: We give proof for each security goal as follow:

a: DATA CONFIDENTIALITY

The shared data in our scheme is categorized into *M1* (course) and *M2* (exam) which are encrypted with different access policies defined by teachers and the fog server.

M1 and v_1 are encrypted with the symmetric session key of the fog server SK_{fog} where v_1 is the secret element to be used by the fog server in the definition of its access policy. In the re-encryption phase, the course (*M1*) is first encrypted with a random symmetric key *k1* which will be encrypted with a set of students' identities based on *IBBE*. Since the symmetric encryption and *IBBE* scheme are secure [4], the data confidentiality of *M1* can be guaranteed against users whose identities are not in the set of students' identities defined by the fog server.

M2 is encrypted with a random symmetric key, and then this key is protected by *CP-ABE* which is secure against chosen plaintext attacks [26].

In addition, the confidentiality of *M2* which is the most sensible data against cheating or stealing can be guaranteed against semi-trusted cloud and fog server. Indeed, only authorized entities with valid attributes that satisfied access policies defined by the teachers can access the data. The cloud server can neither computes *M2* without attributes that satisfy the access policy T, nor computes *M1* without valid identity in the set defined by the fog. The fog server cannot computes *M2* without attributes that satisfy T.

b: FINE-GRAINED ACCESS CONTROL

Our scheme provides fine-grained access control which allows flexibility by specifying two different access policies for students, one for the course and one for the exam. The access policy of encrypted *M2* is defined in access tree which supports operations including both AND and OR gate and can represent any desired condition set. Students can decrypt the ciphertext only if the fine-grained conditions are satisfied. Our scheme provides two levels of fine grained access control.

Level I is guaranteed on the course materials *M1*. If students' identities are included in the authorized set of users' identities defined by the fog server, students can decrypt the ciphertext and obtain *M1*.

Level II is guaranteed on the exam materials *M2*. If students' attributes satisfy both identities set defined by the fog server and access policy T defined by teachers, students can decrypt the ciphertext *M1* and obtain the course *M'1*, as well as they can decrypt the ciphertext and obtain the exam *M2*.

c: UNFORGEABILITY

An adversary who wants to create a valid signature of a legal FS must possess the FS's signing key sent securely by RS which is not the case.

If the adversary modifies the ciphertext CT'1, the receiver can verify that the ciphertext is illegal using Algorithm 5.

VI. PERFORMANCE EVALUATIONS

A. PERFORMANCE EFFICIENCY

Here we analyze the performance efficiency of data encryption and data decryption in our scheme. Let T_{EXP}, T_{PAIR}, $T_{Asymenc}$, $T_{Asymdec}$, N_a and N_{st} denote the computation cost of exponentiation operation in G_1 and G_2, the computation cost of pairing operation, the computation cost of asymmetric encryption, the computation cost of asymmetric decryption, the number of attributes in access policy, the number of users, respectively. For simplicity, we ignore the multiplication, hash, symmetric encryption and decryption operations. In table 3, we discuss the comparison during the data encryption and decryption phases.

TABLE 3. Computation efficiency.

	ENCRYPTION		*DECRYPTION*	
	DATA OWNER	RE-ENCRYPTION ENTITY	DECRYPT1	DECRYPT2
[4]	$(N_{ST}+6)T_{EXP}$	$2T_{PAIR}+(N_{ST}+2)T_{EXP}$	$2T_{PAIR}+(N_{ST}+2)T_{EXP}$	$3T_{PAIR}+(N_{ST}+2)T_{EXP}$
[5]	$(2N_A+6)T_{EXP}$	$(2N_A+3)T_{PAIR}+(3N_A+5)T_{EXP}$	$(2N_A+2)T_{PAIR}+2N_AT_{EXP}$	$(4N_A+4)TPAIR+4N_AT_{EXP}$
[6]	$(2N_A+2N_{ST}+7)T_{EXP}$	$(2N_A+3)T_{PAIR}+(N_{ST}+2)T_{EXP}$	$2T_{PAIR}+(N_{ST}+2)T_{EXP}$	T_{PAIR}
[7]	$(2N_A+2N_{ST}+4)T_{EXP}$	$(2N_A+3)T_{PAIR}+(N_{ST}+2)T_{EXP}$	$2T_{PAIR}+(N_{ST}+2)T_{EXP}$	$3T_{PAIR}+(N_{ST}+2)T_{EXP}+H_3$
OURS	$T_{ASYMENC}+(2N_A+2)T_{EXP}$	$T_{ASYMDEC}+(N_{ST}+2)T_{EXP}$	$(N_{ST}+2)T_{EXP}$	$(2N_A+1)T_{PAIR}$

In order to evaluate performance of our scheme, we first compare it to Xu *et al.* [4] and Liang *et al.* [5] which depend on a single factor; scheme in [4] depends on number of users since it used the IBBE technique and Liang et al. scheme [5] depends on access attributes since it is based on CP-ABE technique. Then, we compare our scheme with the schemes of Huang *et al.* [6] and [7] where both IBBE and ABE techniques are integrated.

In the data encryption phase, the computation cost in our scheme grows linearly with N_a in the data owner side and with N_{st} in the re-encryption entity side. The computation cost of the data encryption in the scheme of Xu *et al.* [4] is lower than ours for the both sides and the same case in the data owner side for the Liang *et al.* [5] scheme. Unfortunately, the former scheme can only support coarse-grained condition for one type of data and the latter cannot address multiple receivers. In addition, compared with schemes [6] and [7], the encryption cost of our scheme grows linearly with one factor at the slowest pace since the $T_{Asymenc}$, $T_{Asymdec}$, are constant. For [6] and [7], the computation cost is growing faster with the two factors N_a and N_{st}.

During the decryption phase, the computation cost of our scheme grows slower than schemes in [4], [5] and [7] and is higher than scheme of Huang *et al.* [6] in the phase of Decrypt 2 since our scheme provides two level of fine-grained access control.

B. FUNCTIONALITY COMPARISONS

In table 4, we compare our scheme with several re-encrypted data sharing schemes in different domains and three recent e-learning based schemes in terms of data confidentiality, fine-grained access control, collusion resistance and profile matching.

For the schemes Liang *et al.* [5], Huang *et al.* [6], Huang *et al.* [7] and ours, the data confidentiality, the fine-grained access control and the collusion resistance are

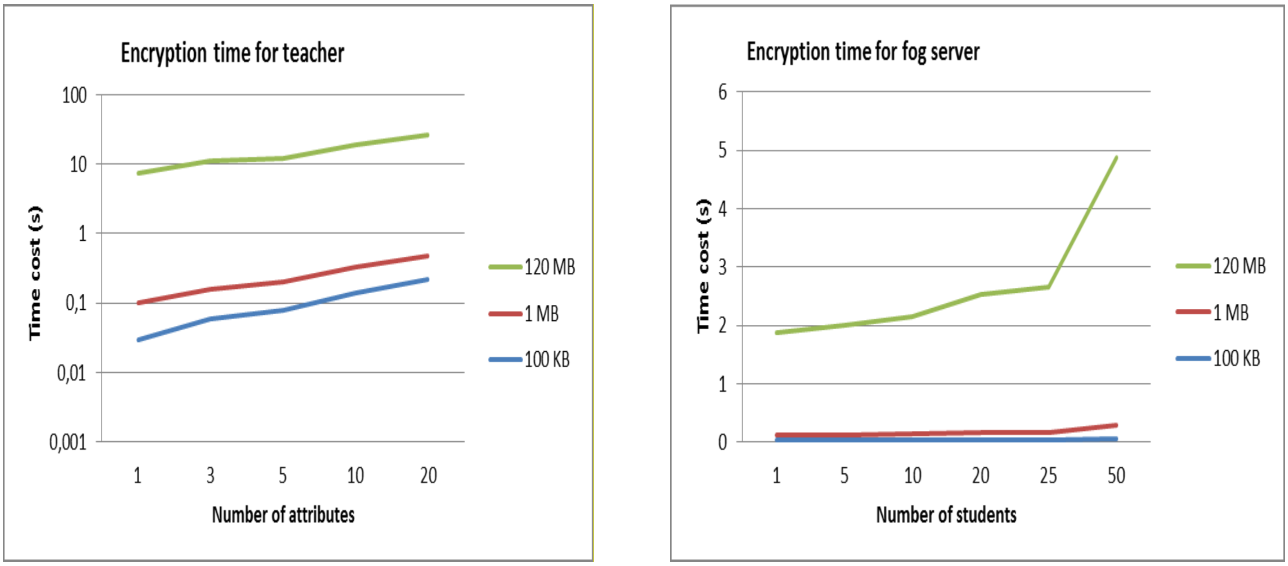

FIGURE 5. **Computation cost of encryption phase.**

TABLE 4. **Functionality comparisons.**

	DATA CONFIDENTIALITY	*FINE-GRAINED ACCESS CONTROL*	*COLLUSION-RESISTANT*	*PROFILE MATCHING*	*APPLICATION DOMAIN*
[4]	Yes	No	No	No	Cloud email
[5]	Yes	Yes	Yes	No	cloud
[6]	Yes	Yes	Yes	Yes	e-health
[7]	Yes	Yes	Yes	No	online social networks
[24]	Yes	No	No	Yes	Cloud email
[23]	Yes	No	No	No	e-learning
[14]	Yes	No	No	No	e-learning
[34]	Yes	Yes	No	No	e-assessment
Our solution	Yes	Yes	Yes	Yes	e-learning

ensured since all of them adopt the IBBE and/or ABE techniques for the initial and re-encrypted data. In addition, our scheme is advanced in data sharing security as data owners could specify which type of encrypted data to be decrypted by users by enforcing fine-grained access policy conditions; while in Xu *et al.* [4], Jose *et al.* [14] and El-Sofany *et al.* [23], data owner can only enforce a simple condition, such as keyword, password, etc.

Khlifi and El-Sabagh [34] adopts behavioral access control policy to realize data access for authorized user; however data collusion is not protected in this model. For the e-learning based schemes Jose *et al.* [14], El-Sofany *et al.* [23], Khlifi and El-Sabagh [34], the security of data sharing is weak and can't neither resist against collusion attack nor support profile matching to allow collaboration among teachers.

Further, Ma *et al.* [24], Huang *et al.* [6] and our scheme all support profile matching. Huang *et al.* [6] achieves matching test on the initial ciphertext of the shared data while in our case the matching concern the social profile of teachers.

C. EXPERIMENTAL ANALYSIS

The experiments are conducted on an Ubuntu 16.04 system with an Intel Core i5 CPU @ 2.30 GHz and 8GB memory, and implemented using cpabe toolkit [3] based on Pairing-Based Cryptography library (*PBC*) [27] which is an open source library that performs the core mathematical functions of pairing-based cryptosystems and AES algorithm. The results are evaluated at encryption and decryption phases.

We assume that the size of both *M1* and *M2* is varying and go from 100 KB to 120 MB for the encryption and decryption process. The size of data depends on type of the course and/or exam which can be video, audio, pdf files. We evaluate the impact of the two factors N_a andN_{st} on the computation cost since the encryption computation time is mainly related to N_a for the teacher side and to N_{st} for the fog server side.

In the data encryption phase, the computation operations are executed by both the teacher to encrypt *M2* depending on the students' attributes and the fog server to encrypt *M1* based on students' identities.

Fig. 5 shows the time cost in both Algorithm 4 and Algorithm 5 which perform the encryption phase according to different sizes of data files. The algorithms run in linear time with N_a and N_{st}. The computation cost in teacher side for *M2* with the size of 100 KB under an access policy with 10 attributes is about 162 ms, while the computation cost in fog server side for *M1* with 10 students takes about nearly 56 ms for course of size 100 KB, which is realistic and meet

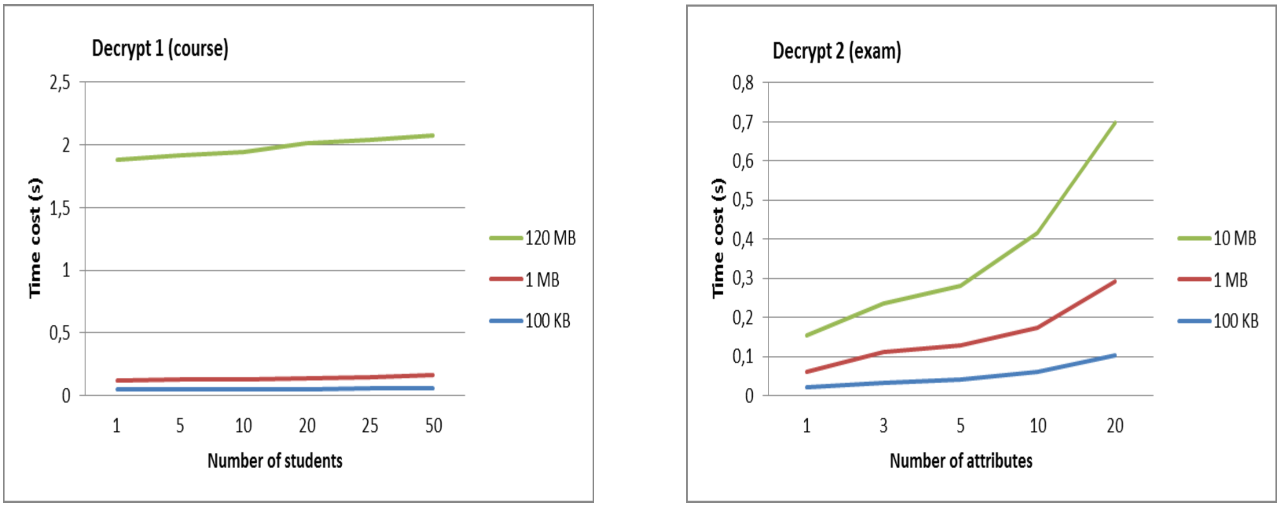

FIGURE 6. Computation cost of decryption phase.

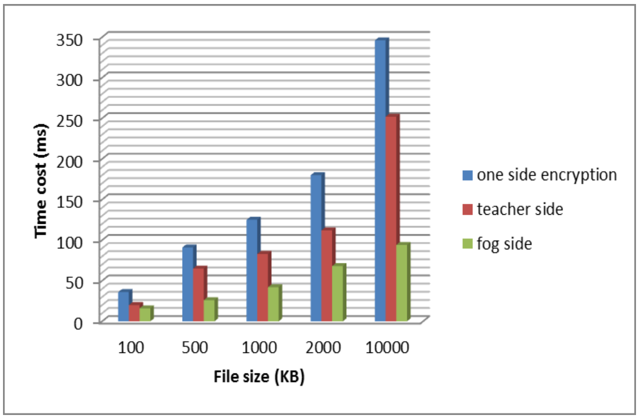

FIGURE 7. Computation cost of encryption phase with different file size.

the regular requirement of data access control in e-learning schemes.

In the decryption phase, a student can decrypt two kinds of encrypted data with the algorithms decrypt1 and decrypt2.

Fig. 6 describes the computation time on student by performing Decrypt1 algorithm to decrypt *M1* with its identity and decrypt2 to decrypt *M2* with its attributes. When decrypting *M1*, the student performs one more pairing operation for the verification process. We can see that the decryption cost has a linear growing with the number of students and their attributes. It is also growing with the increase size of the data.

We consider the impact of data size on execution time of the encryption phase since the size of courses and exams is relatively variable and different from one upload to another. The results in Fig. 6 show that the computation cost increases with the size of data. Moreover, our scheme shows efficiency when sharing the encryption process between teacher and fog server instead of only side encryption which reduces teacher's side computation cost.

In our scheme, we offload the encryption of *M1* from teachers to fog server, which allow fog servers sharing a significant computation cost with teachers and providing location aware service to both students and teachers.

TABLE 5. Execution time for profile matching.

schemes	trapdoor	Execution time (ms)	test	Execution time (ms)
Ma et al [24]	=sk	0,021	2Tpair+2Th+2Texp	75,76
Huang et al [6]	=sk	0,016	4Tpair+2Th	62,01
Our scheme	=sk	0,017	2Tpair+2Th+2Texp	54,06

Next, we compare our scheme with several profile matching schemes to evaluate the trapdoor generation and test algorithms. In Table 5, the evaluation shows that the computation cost in the generation of the trapdoor is the same for all schemes which is equal to time needed for the generation of sk which depends on the cryptographic techniques used in different schemes.

For the test operation, our scheme is lower than the other three schemes.

VII. CONCLUSION

E-learning breaks down the barriers of traditional education and presents advantages compared with classical attending groups by offering the ability to access learning content anytime and anywhere. Learning content can be moved and analyzed across the three-tier architecture cloud-fog-edge where the fog may help control and manage transmitted data to/from edge users.

To guarantee data confidentiality and fine-grained access, we have proposed a secure data sharing and profile matching fog-assisted scheme for e-learning data system. We first categorize the teacher's shared data into course and exam, and encrypt them with different encryption algorithms. Second, the course is re-encrypted by the fog server with *IBBE* cryptographic technique. Further, we provide a profile matching mechanism, which can achieve flexible authorization on encrypted *Social Learning Profiles SLP* s and help teachers to find colleagues in an efficient and privacy-preserving way. In addition, the proposed secure fog based e-learning scheme achieves data confidentiality, fine-grained data control and unforgeability.

Security analysis shows that our solution can achieve fine-grained access control on confident shared data. Performance evaluation demonstrates that our proposed scheme can be applied into e-learning system with efficient encryption burden in terms of computation cost. In fact, results show that the computation cost of encryption phase on teacher side is reduced.

REFERENCES

[1] OpenFog Consortium. (2017). *OpenFog Reference Architecture for Fog Computing*. [Online]. Available: https://www.openfogconsortium.org/wp

[2] C. Delerablée, "Identity-based broadcast encryption with constant size ciphertexts and private keys," in *Proc. 13th Int. Conf. Theory Appl. Cryptol. Inf. Secur.*, Kuching, Malaysia, Berlin, Germany: Springer, 2007, pp. 200–215.

[3] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *Proc. IEEE Symp. Secur. Privacy*, Berkeley, CA, USA, May 2007, pp. 321–334.

[4] P. Xu, T. Jiao, Q. Wu, W. Wang, and H. Jin, "Conditional identity-based broadcast proxy re-encryption and its application to cloud email," *IEEE Trans. Comput.*, vol. 65, no. 1, pp. 66–79, Jan. 2016.

[5] K. Liang, M. H. Au, J. K. Liu, W. Susilo, D. S. Wong, G. Yang, Y. Yu, and A. Yang, "A secure and efficient ciphertext-policy attribute-based proxy re-encryption for cloud data sharing," *Future Gener. Comput. Syst.*, vol. 52, pp. 95–108, Nov. 2015.

[6] Q. Huang, W. Yue, Y. He, and Y. Yang, "Secure identity-based data sharing and profile matching for mobile healthcare social networks in cloud computing," *IEEE Access*, vol. 6, pp. 36584–36594, 2018.

[7] Q. Huang, Y. Yang, and J. Fu, "PRECISE: Identity-based private data sharing with conditional proxy re-encryption in online social networks," *Future Gener. Comput. Syst.*, vol. 86, pp. 1523–1533, Sep. 2018.

[8] A. Huszti and A. Petho, "A secure electronic exam system," *Publicationes Mathematicae Debrecen*, vol. 77, nos. 3–4, pp. 299–312, 2010.

[9] M. J. Kintu, C. Zhu, and E. Kagambe, "Blended learning effectiveness: The relationship between student characteristics, design features and outcomes," *Int. J. Educ. Technol. Higher Educ.*, vol. 14, no. 1, p. 7, 2017.

[10] V. L. Uskov, J. P. Bakken, A. Penumatsa, C. Heinemann, and R. Rachakonda, "Smart pedagogy for smart Universities," in *Smart Education and e-Learning*, V. L. Uskov, R. J. Howlett, and L. C. Jain, Eds. Springer, 2017, Jun. 2017, pp. 3–16, doi: 10.1007/978-3-319-59451-4.

[11] S. Farid, R. Ahmad, M. Alam, A. Akbar, and V. Chang, "A sustainable quality assessment model for the information delivery in E-learning systems," *Inf. Discovery Del.*, vol. 46, no. 1, pp. 1–25, Feb. 2018.

[12] C. King, A. Robinson, and J. Vickers, "Targeted MOOC captivates students," *Nature*, vol. 505, no. 7481, p. 26, Jan. 2014.

[13] T. R. Liyanagunawardena, A. A. Adams, and S. A. Williams, "MOOCs: A systematic study of the published literature 2008-2012," *Int. Rev. Res. Open Distance Learn.*, vol. 14, no. 3, p. 202, Nov. 2016.

[14] G. S. S. Jose and C. S. Christopher, "Secure cloud data storage approach in E-learning systems," *Cluster Comput.*, vol. 22, no. S5, pp. 12857–12862, Sep. 2019.

[15] H. Wang and Q. Li, "Secure and efficient information sharing in multi-university E-Learning environments," in *Advances in Web Based Learning* (Lecture Notes in Computer Science), vol. 4823, H. Leung, F. Li, R. Lau, and Q. Li, Eds. Berlin, Germany: Springer, 2008, pp. 542–553, doi: 10.1007/978-3-540-78139-4_48.

[16] W. Bhuasiri, O. Xaymoungkhoun, H. Zo, J. J. Rho, and A. P. Ciganek, "Critical success factors for E-learning in developing countries: A comparative analysis between ICT experts and faculty," *Comput. Educ.*, vol. 58, no. 2, pp. 843–855, Feb. 2012.

[17] C.-H. Tu, L. Sujo-Montes, C.-J. Yen, J.-Y. Chan, and M. Blocher, "The integration of personal learning environments & open network learning environments," *TechTrends*, vol. 56, no. 3, pp. 13–19, May 2012.

[18] G. Albeanu and F. Popentiu-Vladicescu, "A reliable E-learning architecture based on fog-computing and smart devices," in *Proc. 10th Int. Sci. Conf. eLearning Softw. Educ.*, Bucharest, Romania, Apr. 2014, pp. 24–25.

[19] S. J. Stolfo, M. B. Salem, and A. D. Keromytis, "Fog computing: Mitigating insider data theft attacks in the cloud," in *Proc. IEEE Symp. Secur. Privacy Workshops*, May 2012, pp. 125–128.

[20] A. Velicanu, I. D. V. Lungu, and C. Nisioiu, "Cloud E-learning," in *Proc. Conf. eLearn. Softw. Educ. (eLSE)*, 2013, pp. 380–385.

[21] J. Miguel, S. Caballe, and F. Xhafa, "A knowledge management process to enhance trustworthiness-based security in on-line learning teams," in *Proc. Int. Conf. Intell. Netw. Collaborative Syst.*, Sep. 2015, pp. 272–279.

[22] D. E. Popescu, M. F. Prada, A. Dodescu, D. J. Hemanth, and C. Bungau, "A secure confident cloud computing architecture solution for a smart campus," in *Proc. 7th Int. Conf. Comput. Commun. Control (ICCCC)*, May 2018, pp. 240–245.

[23] H. F. El-Sofany, S. A. El-Seoud, R. T. H. Farouk, and S. Eptember, "Studying the effect of using E-learning through secure cloud computing systems," in *Proc. Int. Conf. Interact. Collaborative Learn.* Cham, Switzerland: Springer, 2018, pp. 54–63.

[24] S. Ma, Q. Huang, M. Zhang, and B. Yang, "Efficient public key encryption with equality test supporting flexible authorization," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 3, pp. 458–470, Mar. 2015.

[25] R. Canetti, S. Halevi, and J. Katz, "Chosen-ciphertext security from identity-based encryption," in *Proc. Int. Conf. Theory Appl. Cryptograph. Techn.* Berlin, Germany: Springer, May 2004, pp. 207–222.

[26] Y. Yang, H. Zhu, H. Lu, J. Weng, Y. Zhang, and K.-K. R. Choo, "Cloud based data sharing with fine-grained proxy re-encryption," *Pervasive Mobile Comput.*, vol. 28, pp. 122–134, Jun. 2015.

[27] B. Lynn. (2006). *The Pairing-Based Cryptography Library*. [Online]. Available: http://crypto.stanford.edu/pbc/

[28] K. Xue, J. Hong, Y. Ma, D. S. L. Wei, P. Hong, and N. Yu, "Fog-aided verifiable privacy preserving access control for latency-sensitive data sharing in vehicular cloud computing," *IEEE Netw.*, vol. 32, no. 3, pp. 7–13, May 2018.

[29] F. Trede, L. Markauskaite, C. McEwen, and S. Macfarlane, "Creating learning opportunities on-the-go," in *Education for Practice in a Hybrid Space*. Singapore: Springer, 2019, pp. 141–155.

[30] C. Steel, R. Springett, and M. Kirk, "Fitting learning into life: Language students' perspectives on benefits of using mobile apps," in *Proc. Conf. ASCILITE (Clil)*, 2012, pp. 875–880.

[31] S. Zhou, Y. Sun, Z. Jiang, and Z. Niu, "Exploiting moving intelligence: Delay-optimized computation offloading in vehicular fog networks," *IEEE Commun. Mag.*, vol. 57, no. 5, pp. 49–55, May 2019.

[32] R. Pecori, "Augmenting quality of experience in distance learning using fog computing," *IEEE Internet Comput.*, vol. 23, no. 5, pp. 49–58, Sep. 2019.

[33] L. Cheung and C. Newport, "Provably secure ciphertext policy ABE," in *Proc. 14th ACM Conf. Comput. Commun. Secur. (CCS)*, 2007, pp. 456–465.

[34] Y. Khlifi and H. A. El-Sabagh, "A novel authentication scheme for e-assessments based on student behavior over E-learning platform," *Int. J. Emerg. Technol. Learn.*, vol. 12, no. 4, p. 62, Apr. 2017.