Indian J.Sci.Res. 17(2): 45 - 53, 2018 ISSN: 0976-2876 (Print)
ISSN: 2250-0138(Online)

NOVEL STEPS OF AUTHENTICATION TECHNIQUE FOR PRIVACY PRESERVATION IN CLOUD INFRASTRUCTURE

[1]Thade Lakshmi Devi, [2]Dr S Krishna Mohan Rao
[1]Department of Computer Science & Engineering, Mewar University, Chittorgarh, Rajasthan
[2]Gandhi Institute of Technology, Bhubaneswar, Odisha

Abstract - Cloud computing has recently emerged as a technology to allow users to access infrastructure, storage, software and deployment environment based on a pay-for-what-they-use model. Traditional features cannot handle the dynamic and multi-tenant nature of the cloud environment as it has to address various technical, legal, and organizational challenges typical to the cloud systems. A new fine-grainedtwo-factor authentication (2FA) access control system for web-based cloud computing services. Specifically, in our proposed 2FA access control system, an attribute-based access control mechanism is implemented with the necessity of both a user secret key and a lightweight security device. As a user cannot access the system if they do not hold both, the mechanism can enhance the security of the system, especially in those scenarios where many users share the same computer for web-based cloud services. In addition, attribute-based control in the system also enables the cloud server to restrict the access to those users with the same set of attributes while preserving user privacy, i.e., the cloud server only knows that the user fulfils the required predicate, but has no idea on the exact identity of the user. The experimental results has improved storage efficiency and error recovery measures than existing techniques.

Keywords - Fine-grained, two-factor, access control, Web services.

I.Introduction

Cloud computing is one of the widely used emerging technique that others various methods to acquire and manage IT resources on a large-scale [19, 22]. Cloud computing, in turn, provides different types of services such as Infrastructure-as-a-service (IaaS) also sometimes called as hardware as a service (HaaS) [1, 7], Platform-as-a-service (PaaS) and Software-as-a-service (SaaS). Cloud computing planning promotes the resource sharing in a pure plug and provides a model that dramatically simplifies its infrastructure. The major advantage of cloud computing includes ease-of-use in accessing the resources over the Internet. Employing the resources in the cloud provides greater expediency to the user because of its systematic manner. Cloud helps us to make use of the existing technologies such as virtualization, service-orientation and grid computing in large-scale distributed environment [4, 5]. To assure the cloud data integrity and availability, efficient approaches that enable storage correctness assurance on behalf of cloud users have to be premeditated. Hence, cloud operations should also imperatively support the dynamic features that make the system design even more challenging.

As Cloud computing is a new emergent technology de-spite having many beneficial factors, it faces many threats in various ways. It has spread very fast due to its exibility over ease of access as it eliminates the need for extra hard drives and memory space allocation. As the cloud is a distributed system, the data stored in it is widespread in distinct locations, and it is accessed anywhere. The distributed nature of the data creates the requirement for high security over outsourced data as there exists a probability that anyone can exploit the outsourced data. The hackers [1, 2, 16], can also access the outsourced data by hacking any server.

Though the new paradigm of cloud computing provides great advantages, there are meanwhile also concerns about security and privacy especially for web-based cloud services.

As sensitive data may be stored in the cloud for sharing purpose or convenient access; and eligible users may also access the cloud system for various applications and services, user authentication has become a critical component for any cloud system. A user is required to login before using the cloud services or accessing the sensitive data stored in the cloud. There are two problems for the traditional account/password-based system.

First, the traditional account/password-based authentication is not privacy-preserving. However, it is well acknowledged that privacy is an essential feature that must be considered in cloud computing systems. Second, it is common to share a computer among different people. It maybe easy for hackers to install some spyware to learn the login password from the web-browser. A recently proposed access control model called *attribute-based access control* is a good candidate to tackle the first problem. It not only provides anonymous authentication but also further defines access control policies based on different attributes of the requester, environment, or the data object. In an attribute-based access control system,[1] each user has a user secret key issued by the authority. In practice, the user secret key is stored inside the personal

[1]Corresponding Author

computer. When we consider the above mentioned second problem on web-based services, it is common that computers may be shared by many users especially in some large enterprises or organizations. For example, let us consider the following two scenarios:

In a hospital, computers are shared by different staff. Dr.Alice uses the computer in room A when she is on duty in the daytime, while Dr.Bob uses the same computer in the same room when he is on duty at night.

In a university, computers in the undergraduate lab are usually shared by different students.

A more secure way is to use two-factor authentica-tion (2FA). 2FA is very common among web-based e-banking services. In addition to a username/password, the user is also required to have a device to display a one-time password. Some systems may require the user to have a mobile phone while the one-time password will be sent to the mobile phone through SMS during the login process. By using 2FA, users will have more confidence to use shared computers to login for web-based e-banking services. For the same reason, it will be better to have a 2FA system for users in the web-based cloud services in order to increase the security level in the system.

With this device, our protocol provides a 2FA security. First the user secret key (which is usually stored inside the computer) is required. In addition, the security device should be also connected to the computer (e.g. through USB) in order to authenticate the user for accessing the cloud. The user can be granted access only if he has both items. Furthermore, the user cannot use his secret key with another device belonging to others for the access.

Our protocol supports fine-grained attribute-based access which provides a great flexibility for the system to set different access policies according to different scenarios. At the same time, the privacy of the user is also preserved. The cloud system only knows that the user possesses some required attribute, but not the real identity of the user.

To show the practicality of our system, we simulate the prototype of the protocol.

In the next section, we will review some related works that are related to our concept.

II. Related Works

Cloud based Access Control Techniquespresents a data access control scheme called DAC-MAC for the multi authority cloud storage system. It provides a multi-authority CP-ABE scheme with efficient data decryption and user revocation functions. This work further offers an Extensive Data Access Control Scheme (EDAC-MACS) that provides secured user data access even at weaker security assumptions. The security analysis results of this scheme prove that this scheme is collusion resistance but lacks at the property of ne-grained access provision to the individual users of the system. In work done by [25, 10], integration of cryptographic techniques with RBAC techniques was made and it uses role keys for data decryption. Further this work presents a hybrid cloud architecture, where the public cloud contains the basic level details and most sensitive information over the private cloud. This work separates the property of user delegation to active and passive types and establishes effective role management through the use of delegation servers and protocols. The Cipher text-Policy Attribute-Based Encryption was given by; it realizes the complex ac-cess control mechanisms over the encrypted data [23, 14]. Here the attributes expressed solitarily the user credentials and the person who encrypts the data could x the access limit to the users for data decryption. Through the use of this scheme, the data stored could be kept confidential even though it resides on the untrusted server. The ID-based cryptographic scheme [8], makes use of the user attributes such as user id for encryption and decryption process of the outsourced data. The development of ID-based cryptographic scheme provides the secured data storage over the public cloud and improved client authorization for other users to access the data content.

A. Attribute-Based Cryptosystem

Attribute-based encryption (ABE) [20], is the cornerstone of attribute-based cryptosystem. ABE enables fine-grained access control over encrypted data using access policies and associates attributes with private keys and ciphertexts. Within this context, ciphertext-policy ABE (CP-ABE) [6] allows a scalable way of data encryption such that the encryptor defines the access policy that the decryptor (and his/her attributes set) needs to satisfy to decryptthe ciphertext. Thus, different users are allowed to decrypt different pieces of data with respect to the pre-defined policy. This can eliminate the trust on the storage server to prevent unauthorised data access.

Besides dealing with authenticated access on encrypted data in cloud storage service [21], [23], [24], [27]–[29], ABE can also be used for access control to cloud computing service, in a similar way as an encryption scheme can be used for authentication purpose: The cloud server may encrypt a random message using the access policy and ask the user to decrypt. If the user can successfully decrypt the ciphertext (which means the user's attributes set satisfies prescribed policy), then it is allowed to access the cloud computing service.

In addition to ABE, another cryptographic primitive in attribute-based cryptosystem is attribute-based signa-ture (ABS) [30]. An ABS scheme enables a user to sign a message with fine-grained control over identifying information. Specifically, in an ABS scheme, users obtain their attribute private keys from an attribute authority. Then they can later sign messages for any predicate

satisfied by their attributes. A verifier will be convinced of the fact that the signer's attributes satisfy the signing predicate if the signature is valid. At the same time, the identity of signer remains hidden. Thus it can achieve anonymous attribute-based access control efficiently. Recently, Yuen *et al.* proposed an attribute-based access control mechanism which can be regarded as the interactive form of ABS.

B. Access Control With Security Device

Security Mediated Cryptosystem: Mediated cryptographywas first introduced in [8] as a method to allow immediate revocation of public keys. The basic idea of mediated cryp-tography is to use an on-line mediator for every transaction. This on-line mediator is referred to a SEM (SEcurity Mediator) since it provides a control of security capabilities. If the SEM does not cooperate then no transactions with the public key are possible any longer. Recently, an attribute-based version of SEM was proposed in [13].

The notion of SEM cryptography was further modi-fied as security mediated certificate less (SMC) cryptography [14]. In a SMC system, a user has a secret key, public key and an identity. In the signing or decryption algorithm, it requires the secret key and the SEM together. In the signature verification or encryption algorithm, it requires the user public key and the corresponding identity. Since the SEM is controlled by an authority which is used to handle user revocation, the authority refuses to provide any cooperation for any revoked user. Thus revoked users cannot generate signature or decrypt ciphertext.

Note that SMC is different from our concept. The main purpose of SMC is to solve the revocation problem. Thus the SME is controlled by the authority. In other words, the authority needs to be *online* for every signature signing and ciphertext decryption. The user is not anonymous in SMC. While in our system, the security device is controlled by the user. Anonymity is also preserved.*Key-Insulated Cryptosystem:* The paradigm of key-insulated cryptography was introduced in [17].

The general idea of key-insulated security was to store long-term keys in a physically-secure but computationally-limited device. Short-term secret keys are kept by users on a powerful but insecure device where cryptographic computations take place. Short term secrets are then refreshed at discrete time periods via interaction between the user and the base while the public key remains unchanged throughout the lifetime of the system. At the beginning of each time period, the user obtains a partial secret key from the device. By combining this partial secret key with the secret period.While our concept *does* require the security device every time security device. Special care must be taken in the process since normal ABS does not guarantee that the leakage of part of the secret key does not affect the security of the scheme while in two 2FA, the attacker could have compromised one of the factors. Besides, the splitting should be done in such a way that most of the computation load should be with the user's computer since the security device is not supposed to be powerful key for the previous period, the user renews the secret key for the current time period.

The user tries to access the system. Furthermore, there is no y updating required in our system.A naive thinking to achieve our goal is to use a normal ABS and simply split the user secret key into two parts. One part is kept by the user (stored in the computer) while another part is initialized into the

Different from our concept, key-insulated cryptosystem requires all users to update their keys in every time period. The key update process requires the security device. Once the key has been updated, the signing or decryption algorithm does *not* require the device anymore within the same time

We specifically design our system in another manner. We do not split the secret key into two parts. Instead, we introduce some additional unique information stored in the security device. The authentication process requires this piece of information together with the user secret key. It is guaranteed that missing either part cannot let the authentication pass. There is also a linking relationship between the user's device and the secret key so that the user cannot use another user's device for the authentication. The communication overhead is minimal and the computation required in the device is just some lightweight algorithms such as hashing or exponentiation over group G_T .[2]

All the heavy computations such as pairing are done on the compute. Different from our concept, key-insulated cryptosystem requires all users to update their keys in every time period. The key update process requires the security device. Once the key has been updated, the signing or decryption algorithm does *not* require the device anymore within the same time.We specifically design our system in another manner. We do not split the secret key into two parts. Instead, we introduce some additional unique information stored in the security device. The authentication process requires this piece of information together with the user secret key. It is guaranteed that missing either part cannot let the authentication pass. There is also a linking relationship between the user's device and the secret key so that the user cannot use another user's device for the authentication.

Entities

Our system consists of the following entities:

- Trustee: It is responsible for generating all system parameters and initialise the security device.

- Attribute-issuing Authority: It is responsible to generate user secret key for each user according to their attributes.
- User: It is the player that makes authentication with the cloud server. Each user has a secret key issued by the attribute-issuing authority and a security device initialized by the trustee.
- Cloud Service Provider: It provides services to anonymous authorised users. It interacts with the user during the authentication process.

Assumptions

The focus of this paper is on preventing private information leakage at the phase of access authentication. Thus we make some assumptions on system setup and communication channels. We assume each user communicates with the cloud service provider through an anonymous channel [26], or uses IP-hiding technology. We also assume that trustee generates the security parameters according to the algorithmprescribed. Other potential attacks, such as IP hijacking, distributed denial-of-service attack, man-in-the-middle attack, etc., are out of the scope of this paper.

Threat Model

In this paper, we consider the following threats:

1) Authentication: The adversary tries to access the system beyond its privileges. For example, a user with attributes {Student, Physics} may try to access the systemwith policy "Staff" AND "Physics". To do so, he may collude with other users.
2) Access without Security Device: The adversary tries to access the system (within its privileges) without the security device, or using another security device belonging to others.
3) Access without Secret Key: The adversary tries to access the system (within its privileges) without any secret key. It can have its own security device.

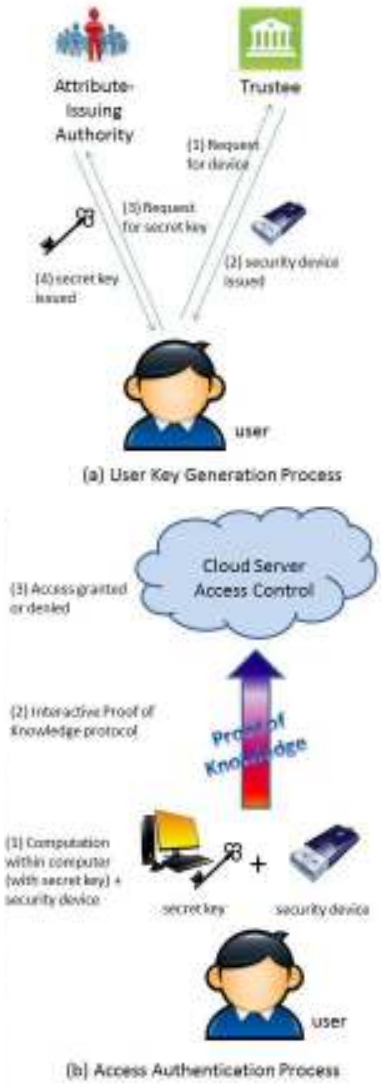

Fig. 1. Overview idea of our system

III. Our Proposed System

A. Specification of the Security Device

We assume the security device employed in our system satisfies the following requirements.

Tamper-resistance.

The content stored inside the security device is not accessible nor modifiable once it is initialized. In addition, it will always follow the algorithm specification.

Capability.

It is capable of evaluation of a hash function. In addition, it can generate random numbers and compute exponentiations of a cyclic group defined over a finite field.

B. Construction

Let A be the desired universe of attributes. For simplicity, we assume A= [1,*n*] for some natural number *n*. We will use a vector $x \in \{0, 1\}^n$ to represent the user's attribute set. Let $x=(x_1, \ldots, x_n) \in \{0, 1\}^n$. If the user is in possession of attribute i ,$x_i = 1$. Otherwise, $x_i = 0$.

System Setup

The system setup process consistsoftwo parts. The first part TSetup is run by a trustee to generate public parameters.

This proof of knowledge of discrete logarithm is straight-forward and is shown in the next subsection. If the proof is correct, the attribute-issuing authority chooses random *authentication:* indeed knows some "knowledge".

If *R* is a binary relation, we let $R(x)= \{y:(x,y) \in R\}$ and the language $L_R= \{x:\exists y$ such that $(x,y) \in R\}$. If $(x,y) \in R$, we call *y* the witness of *x* .

A proof of knowledge is a two-party protocol with the following properties:

Completeness

If $(x,y) \in R$, the honest prover who knows witness *y* for *x* succeeds in convincing the honest verifier of his knowledge.

Soundness

If $(x,y) \in /R$, no cheating prover can convince the honest verifier that $(x,y) \in R$, except with some small probability. It can be captured by the existence of a *knowledge extractor E* to extract the witness *y*: given oracle access to a cheating prover*P*, the probability that *E* outputs *y* must be at least as high as the success probability of *P* in convincing the verifier. We allow the attacker to specify the security device for revocation. If a security device token is revoked, oracle O_i will no longer be

To model the temper-resistant nature of the security device, we model token_i as an oracle O_i with the following behaviour:

Oracle O_i (Internal state : TG, TY tsk

Input $\left(c_R, z_R\right)$ *if* $TG = R_c^{cy+} \wedge TY = TG^y$

Output $s.t.c_R = H\left(tpk \,\text{Re}\hat{}\left(g, h_0\right)^z R \square R \square C\right)$

$\perp$ otherwise

We allow the attacker to specify the security device for revocation. If a security device token_i is revoked, oracle O_i will no longer be available.

We further assume the claim-predicate Υ is chosen by the attacker. An attacker is said to breach the security requirement of authentication, access without security device or access without secret key if it can authenticate successfully for the predicate Υ if for all i such that U_i is controlled by the attacker, $\Upsilon(A_i)= 1$ unless the token_i has been revoked.

The last condition is to capture the situation that the security device is used as a mechanism to revoke a user. A user who is in possession of a security device should not be able to authenticate anymore after it has been revoked.

Regarding the security of our scheme, we have the following lemma.

Lemma *:* If there exists an attacker*F*against our scheme, there exists a simulator *S*, having blackbox access to *F*, that can existentially forge a BBS+ signature or the Schnorr signature under the adaptive chosen message attack or solving the discrete logarithm problem.

Proof*:* In the following we prove Lemma 1 by constructing the simulator *S* under the assumption that attacker

F exists. We utilise the fact that $P K_0$ and $P K_1$ are zero-knowledge proof-of-knowledge protocols. In other words, there exist knowledge extractors E_0 and E_1 that can extract the underlying witnesses of the corresponding protocols.

Common Parameters. Let$p$,G,G_T,$e\hat{}$be a bilinear groupwith$g \in$G being a generator. Let $g\hat{},h,h_0,h_1, \ldots, h_n$ be additional generators of G. We use TG to denote $e\hat{}(g,h_0)$. We further assume full domain hash H: $\{0, 1\}^* \rightarrow Z_p$ which is modelled as a random oracle.

Problem Instances.*S*is given a discrete logarithm problem instance Y^*,h_0. *S* is also given the public key of an instance of BBS+ signature B.PK and the public key of an instance of Schnorr signature S.PK defined over the common parameters. Specifically

B.PK =$w=h^{\gamma}$, S.PK =tpk=TG^{tsk}.

The corresponding signing keys, B.SK=γ and S.SK = tsk, are kept secret from*S*. As an adaptivechosen message attacker, *S* can issue signature queries to the following two oracles:

– $O_{B\,B\,S+}$. On input$(m_0,m_1, \ldots, m_n)$, this oraclereturns(A,e,s) such that

$e\hat{}(A, wh^e)= e\hat{}(hh_0^{x0} h_1^{x1} \cdots h_n^x ng\hat{}^s, h)$.

– $O_{Schnorr}$. On input(m), this oracle returns(c,z)such that

$H(\text{tpk}^c TG^z$ |implies $a=c$ and $b=d$ under the discrete logarithm assumption.

Consider the last two relations from $P K_1$:

Privacy: In every authentication, the informationobtained by the server consists of two parts, namely, $(C,c_R,z_R,C_1, \ldots, C, D_1, \ldots, D)$ and the verifier's viewofPK_1. Note that PK_1 is zero-knowledge and is simulated without having the actual witnesses. Note that this implies we have to employ the zero-knowledge version instead of the honest-verifier zero-knowledge version since the server is playing the role of the verifier. c_R, z_R leaks no information since they are distributed identically for all legitimate users. $C_1, \ldots, C, D_1, \ldots, D$ are information-theoretic secure commitment and again leak no information.

The only componentcontaining information about the user is $C=e\hat{}(g,h_0)y^+R$. This is the verifiable random function with seed y on input R and is computationally indistinguishable from a random value in G_T [18].

E. Efficiency Analysis

We analyse the efficiency of our protocol in two parts. In the first part, we identify the major operations for the authentication protocol in Table IV. The symbols P, $E1$, E T represent in group G and group G_T respectively. The symbol Z p, G, G T represents the size of an element (in bits) inZ_p,Gand$G_T$respectively.

We consider three different platforms, namely, a computer, a smart phone and a smart card.

For the time cost on a smartcard, we use the benchmark result from [40]. The configuration of our test platforms, namely, Computer and Smartphone, are shown in Table III. The time and space cost on the three platforms are listed in Table II. Details of the experiment settings are discussed below.

We use Miracl library version 5.2. The base field is a prime field F_q , where q is a 512-bit prime whose value is:

8*B A*2 *A*5229*B D*9*C*57*C F C*8 *AC E C*76*D F D B F*3*E*

3*E*1952*C*6*B*3193*E C F*5*C*571*F B*502*F C*5*D F*4

10*F*9267*E*9*F*2 *A*605*B B*0*F*76*F*52*A*79*E*8043

*B F*4 *A F*0*E F*2*E*9*F A*78*B*0*F*1*E*2*C D F C*4*E*8549*B*

F. Efficiency Analysis

We analyze the efficiency of our protocol in two parts. In the first part, we identify the major operations for the authentication protocol in Table IV. The symbols P, $E1$, E T representthe time cost (in ms) of a pairing operation, an exponentiation in group G and group G_T respectively. The symbol Z p, G, G T represents the size of an element (in bits) inZ_p,Gand$G_T$respectively.

We consider three different platforms, namely, a computer, a smart phone and a smart card.

For the time cost on a smartcard, we use the benchmark result from [40]. The configuration of our test platforms, namely, Computer and Smartphone, are shown in Table III. The time and space cost on the three platforms are listed in Table II. Details of the experiment settings are discussed below.

We use Miracl library version 5.2. The base field is a prime field F_q , where q is a 512-bit prime whose value is:

8*B A*2 *A*5229*B D*9*C*57*C F C*8 *AC E C*76*D F D B F*3*E*

3*E*1952*C*6*B*3193*E C F*5*C*571*F B*502*F C*5*D F*4

10*F*9267*E*9*F*2 *A*605*B B*0*F*76*F*52*A*79*E*8043

*B F*4 *A F*0*E F*2*E*9*F A*78*B*0*F*1*E*2*C D F C*4*E*8549*B*

The elliptic curve is defined by the equation $y^2=x^3+1 \bmod q$. The group G (as well as G_T) is of order p =800000000000000000000000000000000020001, where

P is a 160-bit prime. The pairing is Tate pairing. Table IV listed the number of operations and communication for an authentication transaction. Recall that n is the size of the attribute universe, and m are the length and width of the span program representing the access policy.

Simulation: Assume the total number of attributes in the system is 100. In other words, the attribute universe A = {1, . . . ,100}. In the following we estimate the efficiency of our system using policy of the following format:

$$\left(\sum_{i=1}^{a}\left(\prod_{j=1}^{b}\mathrm{attr}_{i,j}\right)\right),$$

whereattr$_{i,j}$ maybe re-used in different clauses. In general, this kind of policy can be represented by a span program of length $=a*b$ and width $m=a*(b-1)+1$. The following graphs shows the bandwidth requirement, computational cost at server and user of our system for policy of various size.

Fig. 2 shows the time cost of the server to authenticate a single user. For a relatively simple policy, say, consisting of 2 clauses with 2 attributes per clause for a total of 4 attributes, the time is less than 0.3 seconds. For a policy of 10 clauses with 10 attributes per clause, the time is around 3 seconds. While the asymptotic complexity at the user is similar to that of the server, the time cost for a user is494IEEE Transactions On Information Forensics And Security, Vol. 11, No. 3, March 2016

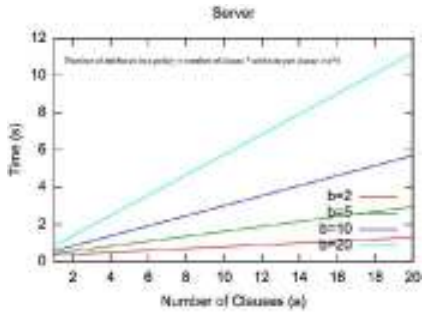

Fig. 2.Running time of the Auth protocol (Server side) (s).

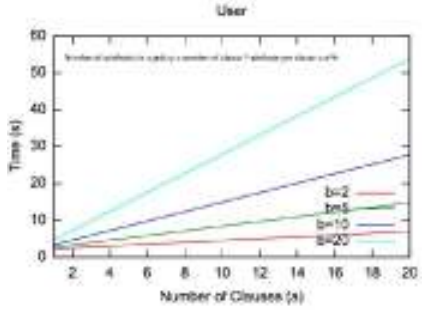

Fig. 3. Running time of the Auth protocol (User side) (s).

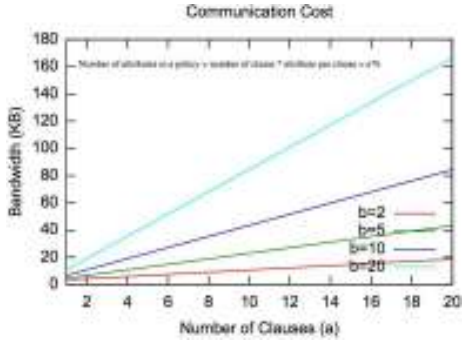

Fig. 4.Communication cost of the Auth protocol (KB).

about five times slower due to the use of a less powerful computing device (a smartphone). One should note that the security device is not the bottleneck as it only accounts for a constant time cost of 0.6 seconds. Please refer to Fig. 3 for the time complexity at the user side. The total authentication time for a policy with 100 attributes, arranged as 10 clauses with10 attributes each, is about 18 seconds. The communication cost of our protocol is depicted in Fig. 4. In particular, for a policy of 100 attributes, the total bandwidth requirement is around 45 KB, which is acceptable for today's network. One could conclude that our protocol is plausible for very simple policy and is still not practical yet for policy of medium size.

Having said that, we would like to remark that the protocol might be optimised. Two possible approaches could be adopted. Firstly, notice that many of the exponentiations are of the form $g^x h^y$ for some fixed bases g and h. This kind of operation is known as multi-base exponentiation and can be computed at about the cost of 110% of a single base exponentiation. It is also worth noting that for fixed base, there are a number of pre-processing techniques available. It is quite likely to reduce the time by half.

IV. Conclusion

In this paper, we have presented a new 2FA (including both user secret key and a lightweight security device) access control system for web-based cloud computing services. Based on the attribute-based access control mechanism, the proposed 2FA access control system has been identified to not only enable the cloud server to restrict the access to those users with the same set of attributes but also preserve user privacy. Detailed security analysis shows that the proposed 2FA access control system achieves the desired security requirements. Through performance evaluation, we demonstrated that the construction is "feasible". We leave as future work to further improve the efficiency while keeping all nice features of the system used by the users in an authentication.

References

[1] M. H. Au and A. Kapadia, "PERM: Practical reputation-based blacklisting without TTPS," in Proc. ACM Conf. Comput. Commun.Secur. (CCS), Raleigh, NC, USA, Oct. 2012, pp. 929–940. Pp. M. H. Au, A. Kapadia, and W. Susilo, "BLACR: TTP-free blacklistable anonymous credentials with reputation," in Proc. 19th NDSS, 2012, 1–17.

[2] M. H. Au, W. Susilo, and Y. Mu, "Constant-size dynamic k-TAA," in Proc. 5th Int. Conf. SCN, 2006, pp. 111–125.

[3] J. Baek, Q. H. Vu, J. K. Liu, X. Huang, and Y. Xiang, "A secure cloud computing based framework for big data information management of smart grid," IEEE Trans. Cloud Comput., vol. 3, no. 2, pp. 233–244, Apr./Jun. 2015.

[4] M. Bellare and O. Goldreich, "On defining proofs of knowledge," in Proc. 12th Annu. Int. CRYPTO, 1992, pp. 390–420.

[5] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in Proc. IEEE Symp.Secur. Privacy, May 2007, Pp. 321–334.

[6] D. Boneh, X. Boyen, and H. Shacham, "Short group signatures," in Advances in Cryptology. Berlin, Germany: Springer-Verlag, 2004, Pp. 41–55.

[7] D. Boneh, X. Ding, and G. Tsudik, "Fine-grained control of security capabilities," ACM Trans. Internet Technol., vol. 4, no. 1, pp. 60–82, 2004.

[8] J. Camenisch, "Group signature schemes and payment systems based on the discrete logarithm problem," Ph.D. dissertation, ETH Zurich, Zürich, Switzerland, 1998.

[9] J. Camenisch, M. Dubovitskaya, and G. Neven, "Oblivious transfer with access control," in Proc. 16th ACM Conf. Comput.Commun.Secur. (CCS), Chicago, IL, USA, Nov. 2009, pp. 131–140.

[10] J. Camenisch and A. Lysyanskaya, "A signature scheme with efficient protocols," in Proc. 3rd Int. Conf. Secur.Commun.Netw. (SCN), Amalfi, Italy, Sep. 2002, pp. 268–289.

[11] J. Camenisch and A. Lysyanskaya, "Signature schemes and anonymous credentials from bilinear maps," in Advances in Cryptology. Berlin, Germany: Springer-Verlag, 2004, pp. 56–72.

[12] Y. Chen, Z. L. Jiang, S. M. Yiu, J. K. Liu, M. H. Au, and X. Wang, "Fully secure ciphertext-policy attribute based encryption with security mediator," in Proc. ICICS, 2014, pp. 274–289.

[13] S. S. M. Chow, C. Boyd, and J. M. G. Nieto, "Security-mediated certificateless cryptography," in Public Key Cryptography (Lecture Notes in Computer Science), vol. 3958. Berlin, Germany: Springer-Verlag, 2006, pp. 508–524.

[14] C.-K. Chu, W.-T.Zhu, J. Han, J.-K. Liu, J. Xu, and J. Zhou, "Security concerns in popular cloud storage services," IEEE Pervasive Comput., vol. 12, no. 4, pp. 50–57, Oct./Dec. 2013.

[15] R. Cramer, I. Damgård, and P. D. MacKenzie, "Efficient zero-knowledge proofs of knowledge without intractability assumptions," in PublicKey Cryptography (Lecture Notes in Computer Science), vol. 1751,H. Imai and Y. Zheng, Eds. Berlin, Germany: Springer-Verlag, 2000, pp. 354–373.

[16] Y. Dodis, J. Katz, S. Xu, and M. Yung, "Key-insulated public key cryptosystems," in Proc. EUROCRYPT, 2002, pp. 65–82.

[17] Y. Dodis and A. Yampolskiy, "A verifiable random function with short proofs and keys," in Public Key Cryptography (Lecture Notes in Computer Science), vol. 3386, S. Vaudenay, Ed. Berlin, Germany: Springer-Verlag, 2005, pp. 416–431.

[18] M. K. Franklin, in Proc. 24th Annu. Int. Cryptol. Conf., Santa Barbara, CA, USA, Aug. 2004.

[19] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryp-tion for fine-grained access control of encrypted data," in Proc. 13thACM Conf. Comput. Commun.Secur., 2006, pp. 89–98.\

[20] J. Han, W. Susilo, Y. Mu, and J. Yan, "Privacy-preserving decentralized key-policy attribute-based encryption," IEEE Trans. Parallel Distrib.Syst., vol. 23, no. 11, pp. 2150–2162, Nov. 2012.

[21] X. Huang et al., "Cost-effective authentic and anonymous data shar-ing with forward security," IEEE Trans. Comput., vol. 64, no. 4, pp. 971–983, Apr. 2015.

[22] J. Hur, "Attribute-based secure data sharing with hidden policies in smart grid," IEEE Trans. Parallel Distrib. Syst., vol. 24, no. 11, pp. 2171–2180, Nov. 2013.

[23] J. Hur, "Improving security and efficiency in attribute-based data shar-ing," IEEE Trans. Knowl. Data Eng., vol. 25, no. 10, pp. 2271–2282, Oct. 2013.

[24] T. Jiang, X. Chen, J. Li, D. S. Wong, J. Ma, and J. Liu, "TIMER: Secure and reliable cloud storage against data re-outsourcing," in Proc. 10th Int.Conf. ISPEC, 2014, pp. 346–358.

[25] K. Liang et al., "A DFA-based functional proxy re-encryption scheme for secure public cloud data sharing," IEEE Trans. Inf. Forensics Security, vol. 9, no. 10, pp. 1667–1680, Oct. 2014.

[26] K. Liang, J. K. Liu, D. S. Wong, and W. Susilo, "An effi-cient cloud-based revocable identity-based proxy re-encryption scheme for public clouds data sharing," in Proc. 19th ESORICS, 2014, pp. 257–272.

[27] K. Liang, W. Susilo, and J. K. Liu, "Privacy-preserving ciphertext multi-sharing control for big data storage," IEEE Trans. Inf. Forensics Security, vol. 10, no. 8, pp. 1578–1589, Aug. 2015.

[28] J. K. Liu, M. H. Au, W. Susilo, K. Liang, R. Lu, and B. Srinivasan, "Secure sharing and searching for real-time video data in mobile cloud," IEEE Netw., vol. 29, no. 2, pp. 46–50, Mar./Apr. 2015.

[29] J. K. Liu, M. H. Au, W. Susilo, and J. Zhou, "Enhancing location privacy for electric vehicles (at the right time)," in Proc. 17th Eur. Symp. Res.Comput. Secur., Pisa, Italy, Sep. 2012, pp. 397–414.