International Journal of Security and Its Applications
Vol. 10, No. 1 (2016), pp.265-276
http://dx.doi.org/10.14257/ijsia.2016.10.1.25

Non-interactive Security Framework for Mobile Device based Academic Monitoring System

B. Muthusenthil[1], C. Vijayakumaran[2], Hyunsung Kim[3]

[1,2]*Dept. of Computer Science and Engineering,*
Valliammai Engineering College
Chennai, 603203, India
bmssen@gmail.com, c_vijayakumaran@yahoo.com
[3](*Corresponding Author*)*Dept. of Cyber Security, Kyungil University*
Kyungsan, Kyungbuk 712-701, Korea
kim@kiu.ac.kr

Abstract

Cloud computing is an emerging computing paradigm in which resources of the computing infrastructure are provided as services over the web. Mobile device based academic monitoring system (AMS) are inherently open systems and thereby vulnerable to various attacks. This paper proposes a non-interactive security framework for mobile device based AMS, which is based on a security and privacy model with the tree permission hierarchy on Bilinear pairing. It has advantageous especially in the communication cost, which required non-interactive communications to establish session key right after the proper authentication. The proposed framework supports privacy based on anonymity, untraceability, and security of confidentiality, integrity, nonrepudiation, key management, and authentication and so on. Thereby, the framework could be used as a basic security building block for AMS over cloud services.

Keywords*: Academic Monitoring System, Security, Privacy, Hierarchical Access Control, Security Framework*

1. Introduction

Cloud computing could be a promising computing paradigm that recently has drawn intensive attention from each academia and industry. As compared to building their own infrastructures, users are ready to save their investments considerably by migrating their businesses into the cloud. With the increasing development of cloud computing technologies, it's not arduous to imagine that in the near future more and more businesses will be moved into the cloud.

The quick advancement and rapid growth of technology in the 21st century changes the living environment and provides several ways to gather information about any entity in the world. At present, numerous schools and colleges have done a lot of helpful investigation on the development of system presenting useful framework, and made a few accomplishments towards the mobile based student academic monitoring system (AMS). At the same time, Learners requirements for access to information and services are growing, the development of mobile learning and ubiquitous learning further put forward higher data and service requirements to the network. Along with the computer hardware and software upgrading and the rapid development of a variety of technology, which requires colleges and universities to develop and integrate the student personal information with their network equipment, which has led to the development of mobile device based AMS.

Data confidentiality and data security are the most security/privacy issues that

ISSN: 1738-9976 IJSIA

would raise great concerns from users when they store sensitive information on cloud servers. It originates from the fact that cloud servers are usually operated by commercial cloud providers which are very likely to be outside of the trusted domain of the users.

In the academic cloud environment, digitization of student personal information (SPI) for wireless student AMS has brought several edges and challenges for sensitive data while sharing on cloud servers especially when they are not within the trusted domain. To keep sensitive student data confidential against un-trusted servers, existing solutions provides cryptographic methods by disclosing data decryption keys solely to authorized users. However, these solutions introduce process overhead on the data owner for key distribution and data management and therefore don't scale well. The problem of achieving scalability, fine-graininess, and data confidentiality of access control really still remains unresolved. In this view, we tend to propose trust based student identity management framework as a cloud based utility service. It achieves student anonymity, session key secrecy and resistance against various security attacks, especially including replay attacks.

Student AMSs allow the study of student phenomena and the design of prediction and reaction mechanisms to dangerous situations. In its general form, a monitoring system is composed by a certain number of devices designed to measure different physical quantities, one or more processing nodes, and a communication network. The sensors provide in output analogical signals, which are conditioned and converted into the digital domain. The digital signals are then transmitted to the computing devices, which perform the aggregation of the obtained data to understand the measured phenomenon.

Privacy - It is protection of transmitted student private data from passive attacks. The objective is to ensure that sensitive data of student is not being accessed by or disclosed by any unauthorized person.

Security - It is protection of student sensitive data from vulnerable attacks.

Miranda *et al*. in [1] proposed client based Manager which helps to reduce the risk related to data leakage and loss of sensitive data using obfuscation and deobfuscation techniques. The advantage of these methods is that it preserves privacy of data by customizing the end user service problems. The limitation of this work is that there is requirement of honest co-operation of service provider and also vendors not add extra services for privacy protection. Wang *et al*. in [2] provided anonymity based method for privacy of data in cloud. Advantages of this method are it is simple and flexible and differ from the traditional cryptography technology. It is limited for number of services. Gentry *et al*. in [3] presented encryption method for preserving the privacy of data. This method of encryption enables the computation on the encrypted data which is stored in cloud. Cloud provider is not aware of data processing function as well as result of computation. It is a powerful tool for privacy maintenance but it fails to use practically. Greveler *et al*. in [4] provided the architecture for database storage in cloud which preserves the privacy of users' data by encrypting and assigning secure identities to the each and every request and response and they send as the XML request to database with the maintenance of machine readable access rights. An advantage of it is that easier to handle the encryption schemes and helps to preserve the privacy but limitation is providing machine readable access rights. Zhou *et al*. in [5] proposed a method of access control by considering privacy of data. In case of this method every user is provided with some attributes, which defines their access rights. Singh *et al*. in [6] used the RC5 encryption algorithm to secure storage data. Gampala *et al*. in [7] proposed the data security using the elliptic curve cryptography. In Wang *et al*. of [8], the cryptography fails to secure efficiently storage data so they proposed the

technique of auditing with concept of third party auditing (TPA) uses the Homomorphic authenticator which gives the guarantee about the TPA. Bohli *et al*. in [9] is providing more security and privacy to sensitive data in cloud using multi-cloud environment in which enhanced security is provided by dividing file into multiple chunks and storing those chunks into multiple clouds to protect security and privacy. This paper gives an insight of different threats in cloud computing environment with respect to security and privacy of user's sensitive data in the cloud environment. Researchers have proposed different methods to tackle issues using different approaches which somewhat helps to minimize the problem over the data security and privacy in the cloud. We have discussed about the advantages and limitations of existing methods to completely solve security and privacy issue. These are the open issues to work on.

To solve the security and privacy problems in the previous researches, we propose a non-interactive security framework for mobile device based AMS. For the privacy aspect, it is required to establish a practical security framework which provides sophisticated privacy management and addresses security simultaneously. To achieve the security and privacy goal of the proposed non-interactive framework, we use the identity-based encryption (IBE) and identity-based signature (IBS) as the basic security building blocks proposed by Shamir [10].

Thus, in this paper, the following activities will be taken up:

1. Development of a trust based student identity management framework as a cloud based utility service for Student Personal Information (SPI).
2. To achieve fine-grained, scalability and data confidentiality for data access control in cloud computing.
3. To enable the data owner to delegate most of computation intensive tasks to cloud servers without disclosing data contents or student access privilege information.
4. To propose a non-interactive security model based on IBE and IBS for SPIs on the cloud.

2. Background

This section defines a target AMS system configuration and the role of each entity in the system. Providing privacy and security seems to be a dilemma and has been a unique challenge in wireless networks. A student has to reveal his (or her) identity in order to be verified for authentication purposes. On the other hand, the identity of the student serves as a unique information that an attacker can make use of to filter out a particular student's behaviors, and trace his (or her) whereabouts and activities, which may leak sensitive privacy. The linkability among a student's online transactions may also enable an unauthorized students profiling without the student's consent.

2.1 System Configuration

We consider a basic architecture, depicted in Figure 1, consisting network of disparate devices and multiple access points. The main parties in our system are wireless network node WNN_i, a gateway GW_i and a back-end server TPA_i, which are composed of an e-AMS server SV with the SPI database and attending various access clients. A client can read his (or her) academic and non-academic records *etc*. from the server. This request is performed directly by the department gateway, which in turn forwarded it to the e-AMS server. Authorized person such as TPA_i and academician could access the databases directly to monitor and view the student information.

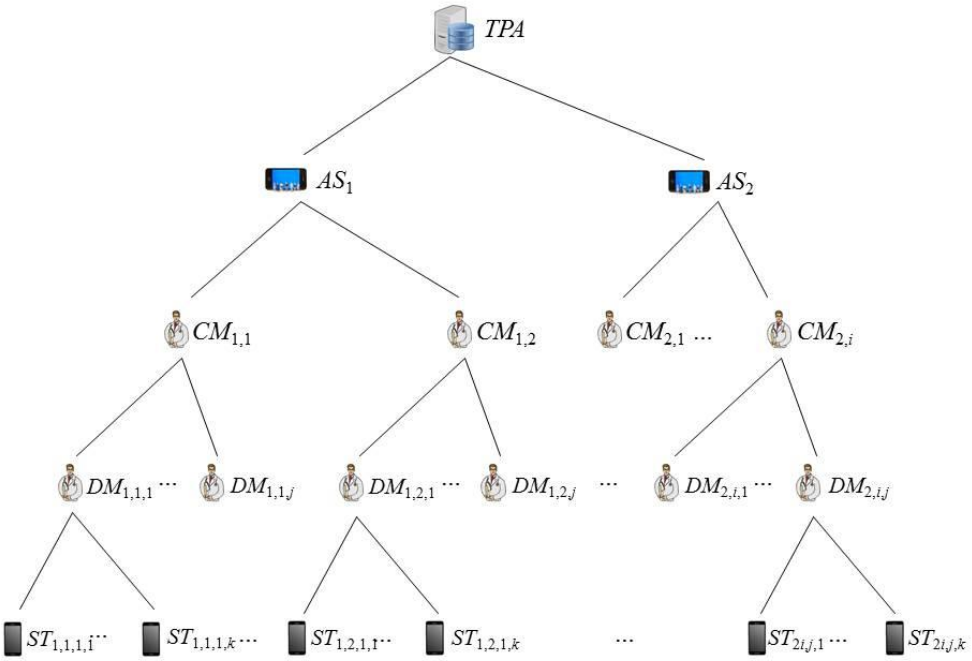

Figure 1. System Entity Hierarchy based on their Role

2.2 Role of System Entities (or Applications over AMS)

Our aim is to establish a security and privacy framework to cope with tampering throughout AMS network, while allowing it to be easily viewed by the legal entities. A simplified AMS usually consists of five layers as shown in Figure 1 and each entity has the following role

ST_i: This is student with mobile device, which are responsible for the receiving and transmitting of the SPI to TPA_i.

DM_i: This is department manager, which has responsibility for the collection of the request with the e-AMS server *SV* or through CM_i.

CM_i: This is college manager, which is responsible for the planning and managing whole college private data on students.

AS_i: This is application server, which uses students database SPI managed by higher authority. The SPI is a subset of the electronic student record maintained by each AS_i and is created and owned by the student ST_i.

TPA: This is trusted party authority, which works as the back-end server, *i.e*. e-AMS server and has the role of private key generation (PKG). It is mainly responsible for key setup, key management, data analysis, data management and data processing.

Note that it is possible to expand the role hierarchy depending on University's request or expanding their services to students. It is inevitable in modern academic system that requires sophisticated and comprehensive monitoring tool which helps in frequently monitoring the progress of a student right from his or her admission to the graduation by both faculties and parents. An up-to-date student's specific information provided by the AMS helps course advisor, faculties and parents in multi-dimensional aspects such as advising sessions, preparing progress report and attendance report and so on.

2.3 Applications of Academic Monitoring System

The main application of this work is to build an open source, cloud-based education data infrastructure in the hopes of addressing a number of problems schools, colleges and universities face:

1) It improves the lack of data interoperability between the various databases and software systems.
2) It acts as a tool for student information systems that store students' education records. That makes for a lot of bureaucratic inefficiencies with teachers and staff manually downloading, uploading, re-entering student information — rosters, grades, and so on — into various applications.
3) It builds a full profile on students and to track and support their progress.
4) It provides to have sharing "confidential student and teacher information with the various parties like placement companies, training vendors, *etc*. The information to be shared will likely include student names, test scores, grades, disciplinary and attendance records, special education.
5) It also concerns about privacy and security in the cloud, as colleges/universities/schools move their data storage and servers from a local to a virtualized environment.
6) It provides better data privacy, security and transparency.

3. Security and Privacy Requirements on AMS

In the presence of various security attacks, AMS should satisfy the following security and privacy requirements. To set the goal of security and privacy requirements, we first define an adversarial model.

3.1 Adversarial Model

Smartphone-based AMSs are inherently open systems and thus vulnerable to adversarial behavior. We first consider external adversaries, *i.e.*, unauthorized identities that try to harm the system operation. Such adversaries can eavesdrop, intercept, and modify the communication of the system entities. They can also launch jamming attacks, but we do not dwell on such attacks; we rely on the cellular operator for their mitigation. We also consider internal adversaries, *i.e.*, user devices or AMS entities, which exhibit malicious behavior. Malicious or comprised mobile devices might submit faulty student reports. After a disruptive action, adversaries might repudiate it. For the infrastructure components, we consider honest-but-curious system entities that correctly execute protocols but try to harm the privacy of users. More than one system entity could collude to harm user privacy.

Furthermore, we consider privacy threats focused on anonymity, SPI data privacy and untraceability even against to the insiders of AMS. Privacy is considered as information relating to an identified or identifiable individual. Privacy threats come in a number of forms, including threats to academic reports, reputation, solitude, autonomy and safety. Intrusion or interruption of an individual's life or activities can threaten the individual's ability to be left alone. Communications may be directed between the initiator and the recipient or additional entities may be involved in packet forwarding, which may interfere with privacy protection goals, as well. Although the additional entities may not generally be considered as attackers, they may all pose privacy threats, because they are able to observe and collect privacy-relevant data. From a privacy perspective, one important type of attacker is a passive attacker, who is an entity that passively observes the entity's communications without the entity's knowledge or authorization. Different kinds of attacks may be feasible at different points in the communications path. A passive

attacker could mount surveillance or identification attacks between two communication participants.

3.2 Security Requirements

Security is one of very important issue on AMS. Like the other network system, we consider the basic security functionalities including confidentiality, integrity, authentication, authorization, accountability, key management and nonrepudiation in the contexts of AMS.

Confidentiality/Integrity (*R*1): The confidentiality and the integrity of the communications between the system entities (*i.e.*, infrastructure and smartphones) should be ensured.
Authentication and authorization (*R*2)*:* Only authorized devices shall be able to submit student reports or retrieve student status updates from AMS.
Accountability (*R*3)*:* User devices should be reliable for actions disrupting the system operation. The system should provide the necessary means for the identification (deanonymization) and the eviction of faulty devices. After their eviction (revocation of their credentials), offending devices should no longer be able to participate in AMS.
Key management (*R*4)*:* Parties performing key management functions are properly authenticated and their authorizations to perform the key management functions for a given key are properly verified by AMS.
Nonrepudiation (*R*5)*:* i) provide the recipient(s) of a message with the proof of the origin of the message. ii) Provides the originator of a message with the proof that the message has been delivered to the originally specified recipient(s).iii) provides the originator of a message with the proof of receipt of the message. It will protect against any attempt by the recipient(s) to falsely deny receiving the message.

3.3 Privacy Requirements

In the presence of adversaries, the system should satisfy the following privacy requirements.

Anonymity (*R*6)*:* Transactions should be performed in a privacy-preserving manner. More specifically, AMS should receive guarantees for the eligibility of the device with respect to AMS service. No information concerning the real identity of the device and, consequently, of the subscriber should leak.
Report unlinkability (*R*7)*:* Ideally, AMS should not be able to link reports originating from the same device. However, inference techniques can (with some probability) link anonymous reports from the same device. To this end, AMS system should render such inference attacks hard.

4. Non-Interactive Security Framework

To achieve anonymity of entities in AMS, non-interactive security framework (NSF) adopts pseudonyms which are issued by PKG to students. This section proposes a new NSF for AMS, which uses IBC. It uses a security model based on the tree permission hierarchy based on the IBC.

4.1 Security Primitive – Bilinear Pairing

To achieve the security and privacy goal of the proposed framework, we use the IBE and the IBS as the basic security building blocks. This section briefly describes them. Similar to the other IBC-based schemes, NSF requires a PKG for the system initialization. Let G_1 be an additive group of prime order q and G_2 be a

multiplicative cyclic group of the same order. In reality, G_1 is a subgroup of points on an elliptic curve over Z_q^*, and G_2 is a subgroup of the multiplicative group of a finite field $Z_{q^k}^*$ for some $k \in Z_q^*$. Let P denote a generator of G_1. Then we can have the two definitions for the building blocks.

[Definition 1] There exists an efficient computable bilinear map ê: $G_1 \times G_1 \rightarrow G_2$, which has the following bilinearity, non-degeneracy and computability properties [10]:

- Bilinearity: Given P and Q in G_1 and $a, b \in Z_q^*$, we have $\hat{e}(a \cdot P, b \cdot Q) = (P, Q)a \cdot b$.
- Non-degeneracy: $\hat{e}(P, P) \neq 1$ in G_2.
- Computability: There exists an efficient algorithm to compute $\hat{e}(P, Q)$ for any $P, Q \in G_1$

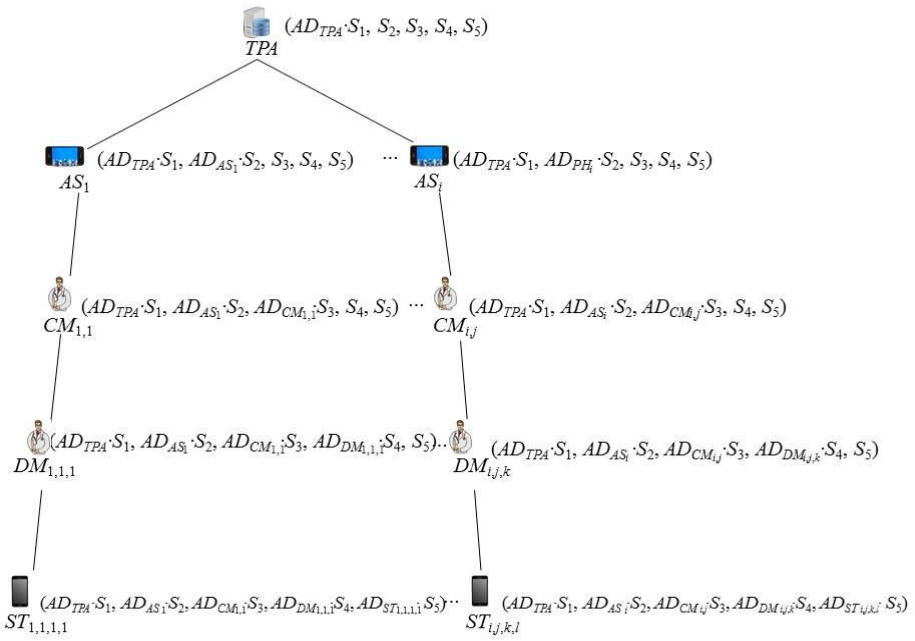

Figure 2. Proposed Security Model based on the Permission Hierarchy Tree

[Definition 2] There are two Diffie Hellman (DH) problems, bilinear DH (BDH) and computational DH (CDH). The BDH problem is to compute $\hat{e}(P, P)_{a \cdot b \cdot c} \in G_2$ given $P \in G_1$ and elements $a \cdot P$, $b \cdot P$, $c \cdot P \in G_1$ for $a, b, c \in Z_q^*$. Computing such a problem is assumed to be hard on $\{G_1, G_2, \hat{e}\}$. The CDH problem is given as $(P, a \cdot P, b \cdot P)$, and computing $a \cdot b \cdot P$ is assumed to be hard [11].

4.2 Security Model - Permission Hierarchy Tree

The permission hierarchy shown in Figure 2 performs a very important role inside of NSF in two ways that classify the capabilities of each entity in the hierarchy and could help one-round key establishment between any two parties in the hierarchy. It is a natural way to implement a hierarchy, so as to reflect the role structure to show the line of authority and responsibility. Conventionally, more privilege is shown toward the top of the tree and less privilege towards the bottom. The role of each participant in the hierarchy is pre-allocated before setup the security model. Each entity in the tree contains the bindings between the amplified identity and the secret key of the entity and the higher level entities in its hierarchy.

It should be noted that in order to provide privacy, our security model uses not the real identity but the dynamic identity dependent of AMS network participants. First of all, *TPA* defines a secure hash function $H(\cdot)$ and a symmetric cryptosystem

$E_K(M)$, which encrypts a message M with a key K based on advanced encryption standard (AES). *TPA* with its identity ID_{TPA} creates a private key set (S_1, S_2, S_3, S_4, S_5) for an AMS and computes an amplified identity $AD_{TPA}=H(ID_{TPA})$ and $AD_{TPA}\cdot S_1$. After that, *TPA* stores the information in its memory. To allow identity revocation, *TPA* could add a random number r_i into AD_i, such that each of the amplified identities is derived as $AD_i = H(ID_i||r_i)$.

When AS_i needs to be registered to *TPA*, an AS_i submits its amplified identity AD_{ASi} to *TPA*, which is computed as $AD_{ASi} = H(ID_{ASi})$. *TPA* computes $AD_{ASi}\cdot S_2$ and sends $\{(AD_{TPA}\cdot S_1, AD_{ASi}\cdot S_2, S_3, S_4, S_5)$ and $(AD_{TPA}, AD_{ASi})\}$ to AS_i via a secure channel, and AS_i keeps the information in private, which is preferably stored in the smartcard of his (or her) identity card or the smart device. The other registration is the similar with AS_i as shown in Figure 2 and thereby, we omit the detailed procedures in this paper.

4.3 Security and Privacy Provisions

When any entity in AMS needs to communicate with an entity in the hierarchy shown in Figure 2, it is possible to establish a secure session based on the permission hierarchy after the proper authentication. As an example we consider a communication between a student $ST_{a,b,c,d}$ and an application server AS_i in AMS. To establish a secure channel, they conduct the following tasks:

Step 1. $ST_{a,b,c,d}$ with its private key set ($AD_{TPA}\cdot S_1$, $AD_{ASa}\cdot S_2$, $AD_{CMa,b}\cdot S_3$, $AD_{DMa,b,c}\cdot S_4$, $AD_{STa,b,c,d}\cdot S_5$) chooses a random number r_1, computes $R_1=r_1\cdot AD_{STa,b,c,d}$ and a fresh session key $SK_1=\hat{e}(AD_{TPA}\cdot S_1, AD_{TPA'})\cdot\hat{e}(AD_{ASa}\cdot S_2, AD_{ASi'})\cdot\hat{e}(AD_{CMa,b}\cdot S_3, AD_{ASi'})\cdot\hat{e}(AD_{DMa,b,c}\cdot S_4, AD_{ASi'})\cdot\hat{e}(AD_{STa,b,c,d}\cdot S_5, AD_{ASi}')r_1$ by using the amplified identity set of the counterpart AS_i, which is ($AD_{TPA'}$, AD_{ASi}'). After that, $ST_{a,b,c,d}$ computes $MAC_1=H(SK_1||R_1)$ and sends $\{R_1, AD_{STa,b,c,d}, MAC_1\}$ to AS_i.

Step 2. When AS_i receives the message from $ST_{a,b,c,d}$, AS_i establishes the session key $SK_1'=\hat{e}(AD_{TPA}\cdot S_1, AD_{TPA'})\cdot\hat{e}(AD_{ASi}\cdot S_2, AD_{ASa'})\cdot\hat{e}(AD_{ASi}, AD_{CMa,b'})S_3\cdot\hat{e}(AD_{ASi}, AD_{DMa,b,c'})S_4\cdot\hat{e}(AD_{ASi}, R_1)S_5$ by using the amplified identity set of the counterpart $ST_{a,b,c,d}$, which is ($AD_{TPA'}$, $AD_{ASa'}$, $AD_{CMa,b'}$, $AD_{DMa,b,c'}$, $AD_{STa,b,c,d'}$). AS_i assures the correctness of the established fresh session key only if the validity check of MAC_1 is successful, by comparing it with AS_i's computation of $H(SK_1'||R_1)$.

The established session key is used to secure the forthcoming messages, which is to provide confidentiality and content privacy based on $E_K(M)$. Furthermore, a message integrity code (MIC) with the input of the message and the established session key could be attached to the cipher text by using cipher block chaining mode of AES, which is to provide integrity and authenticity of the message by the recipient. By using both of the cipher text and the MIC, the communication counterpart could support authentication of participant because only the registered participant could compute the session key correctly.

For the purpose of providing anonymity and non-linkability, each participant could send the session dependent identity DAD_i instead of using the real identity ID_i, which is computed as $DAD_i=R_1\cdot H(ID_i)$ by using the session public key R_1. The communication counterpart j could derive its session key by using its private key ADS_j and the received dynamic session dependent identity. From the computation of the session key, we could know that there is no way that the participant j could know the real identity ID_i of any entity in AMS from the session dependent identity DAD_i. Additionally, to support non-repudiation, each participant could sign on their messages under the IBS.

In summary, the proposed framework could support whole security and privacy properties discussed before based on the security model. Anonymity and non-

linkability are achieved by using the session dependent identity DAD_i. However, participants' real identity can be revealed when disputes arise, and at least some specific forms of additional message exchanges are required but we omit the detailed steps due to the space limit. The proposed framework can be further developed to implement a suite of security and privacy services.

5. Security and Privacy Analysis

This section provides security and privacy analysis of the proposed framework. This section follows the security analysis approaches used in [12-15]. The analyses are focused on verifying the overall security and privacy requirements for the NSF, including passive and active attacks, as follows.

Proposition 1. NSF provides entity anonymity.
Proof: In NSF, the anonymity of the entity is obtained by applying the hash function and is based on the BDH problem. Security model and steps to establish a session key use amplified identities by using the one-way hash function. TPA only gets the real identity of each entity only after some additional communication. There is no way for an attacker to know the real identity, even if the attacker could capture the message $\{R_1, AD_{STa,b,c,d}, MAC_1\}$ during the session run to establish a secure channel.

Proposition 2. NSF cannot reveal the private key set or the generated session key to outsiders.
Proof: The security of the private key set is based on the combinations of the amplified identities and the secret values. This indicates that an attacker has to know both of them to retrieve the private key set. However, there is no way that the attacker could derive the secret values or the amplified identities from the private key set due to the BDH and the CDH problems. For the concern of revealing the session key *SK*, the attacker needs to get power to analyze and get necessary information from the intercepted message $\{R_1, AD_{STa,b,c,d}, MAC_1\}$. However, there is no way that the attacker could know the session key in NSF due to the BDH and the CDH problems.

Proposition 3. NSF provides session key freshness and thereby can prevent from the replay attack.
Proof: The random number r_i used to establish the session key guarantees the freshness of the session key. There is no way that an attacker could get any information to know the session key due to the BDH problem. Furthermore, NSF is strong against the replay attack due to the session key freshness support with MAC_1.

Proposition 4. NSF is secure against passive attack.
Proof: We assume that an attacker is successful if the attacker knows any useful information from the intercepted messages. We show that the probability of success for learning them is negligible due to the difficulty of the BDH and the CDH problems.

- The completeness of NSF is already proven by describing the run of communication sessions in the previous section.
- If the attacker is passive, all the attacker can gather is only the intercepted message $\{R_1, AD_{STa,b,c,d}, MAC_1\}$. However, it is negligible to find the key related information from them due to the difficulty of the BDH and the CDH problems.

Finally, we could say that NSF is secure against passive attack.

Proposition 5. NSF is secure against active attack.

Proof: We could argue that an attack from an attacker is successful if the attacker finds the session key SK_i or knows any of the messages M_1 and M_2. Therefore, we will show that the probability of the success of finding them is negligible due to the difficulty of the BDH and the CDH problems.

The acceptance by all entities means that each MAC_i in the corresponding message is successfully verified. This means that MAC_i is verified successfully by using the correct session key SK_i dependent on the session fresh information. If it is the case that entities accept the messages and they continue the session, the probability that the attacker could modify the messages is negligible. Additionally, the only way for the attacker to find the session key or the private key information is to solve the difficulty of the BDH and the CDH problems.

- Now, we consider the active attacker with the following cases.

(1)There is no way that an attacker could get the private key set related to $\{S_1, S_2, S_3, S_4, S_5\}$ due to the difficulty of the BDH and the CDH problems.

(2)An attacker cannot masquerade as any AMS entity in NSF because the attacker cannot generate a valid message without deriving the correct session key SK_i. Furthermore, the attacker could not compute the proper MAC_i, which is required for the verification of the session for the counter party.

Finally, we could say that NSF is secure against active attack.

6. Conclusion

In this paper, we have proposed a practical non-interactive security framework which provides sophisticated privacy management and addresses security requirements simultaneously for the mobile device based AMS. To achieve the security and privacy goal of the proposed non-interactive framework, we used the identity-based encryption (IBE) and identity-based signature (IBS). To propose a communication-efficient protocol, we proposed a permission hierarchical tree for AMSs with the consideration of the network requirements. In the IBE and IBS, each entity in the AMS is only pseudonymously identified, hence protecting the entity from the negative effects of identity theft, such as fraudulent student data claims by attackers. This allows the academician and his/her stakeholders to establish a secure channel via a session key, which only requires one-round communication and does not require any further interactive communications. The analyses have shown that the proposed non-interactive security framework based on IBE and IBS achieves good security and privacy properties and functionalities.

Acknowledgements

This work was supported by the National Research Foundation of Korea Grant funded by the Korean Government (MEST) (NRF-2010-0021575). This paper is a revised and expanded version of a paper entitled “Security and Privacy Framework for Academic Monitoring System” presented at SECTECH 2015, Jeju, Korea, Nov. 26 2015.

References

[1] M. Miranda and S. Pearson, “A Client-based Privacy Manager for Cloud Computing,” Proc. of the Fourth International ICST Conference on Comm. and Middleware, Article no. 5, **(2009)** .

[2] J. Wang, Y. Zhao, S. Jiang and J. Le, “Providing privacy preserving in cloud computing,” Prof. of International Conference on Test and Measurement, vol. 2, **(2009)**, pp. 213–216.

[3] C. Gentry, “Fully Homomorphic encryption using ideal lattices,” Proc. of the forty-first annual ACM symposium on Theory of computing, **(2009)**, pp. 169–178.

[4] U. Greveler, B. Justus and D. Loehr, “A Privacy Preserving System for Cloud Computing,” Proc. of the 11th IEEE International Conference on Computer and Information Technology, **(2011)**, pp. 648–653.

[5] M. Zhou, Y. Mu, W. Susilo and M. H. Au, "Privacy-Preserved Access Control for Cloud Computing," Prof. of International Joint Conference of IEEE TrustCom 2011/IEEE ICESS 2011/FCST 2011, **(2011)**, pp. 83–90.
[6] J. Singh, B. Kumar and A. Khatri, "Securing the Storage Data using RC5 Algorithm," Internatinal Journal of Advanced Computer Research, vol. 2, no. 4, **(2012)**, pp. 94-98.
[7] V. Gampala, S. Inuganti and S. Muppidi, "Data Security in Cloud Computing using Elliptic Curve Cryptography," International Journal of Soft Computing and Engineering, vol. 2, no. 3, **(2012)**, pp. 138-141.
[8] C. Wang, S. Chow, Q. Wang and K. Ren, "Privacy-Preserving Public Auditing for Secure Cloud Storage," IEEE Transactions on Computers, vol. 62, no. 2, **(2013)**, pp. 362–375.
[9] J. M. Bohli, N. Gruschka, M. Jensen, L. L. Iacono and N. Marnau, "Security and Privacy-Enhancing Multi-cloud Architectures," IEEE Trans. on Dependable and Secure Computing, vol. 10, no. 4, **(2013)**, pp. 212-224.
[10] A. Shamir, "Identity-based cryptosystems and signature schemes," Lecture Notes in Computer Science, vol. 196, **(1984)**, pp. 47-52.
[11] H. Guo, Y. Mu, Z. Li, X. Zhang, "An efficient and non-interactive hierarchical key agreement protocol," Computers & Security, vol. 30, **(2011)**, pp. 28-34.
[12] H. Kim, "Location-based authentication protocol for first cognitive radio networking standard," Journal of Network and Computer Applications, vol. 34, **(2011)**, pp. 1160-1167.
[13] H. Kim, "End-to-end authentication protocols for personal/portable devices over cognitive radio networks," International Journal of Security and Its Applications, vol. 8, no. 4, **(2014)**, pp. 123-138.
[14] S. W. Lee, H. Kim, "Freshness Consideration of Hierarchical Key Agreement Protocol in WSNs," International Journal of Security and Its Applications, vol. 8, no. 1, **(2014)**, pp. 81-91.
[15] H. Kim, "Efficient and non-interactive hierarchical key agreement in WSNs," International Journal of Security and Its Applications, vol. 7, no. 2, **(2014)**, pp. 159-170.

Authors

Balasubramanian Muthusenthil, He received the degree in Electronics & Communication Engg from Madras University, in 1996 and Master's Degree from Satyabama University in 2007. He is a research student of Anna University, Chennai. Currently, he is an Assistant Professor (Senior Grade) at Valliammai Engineering College, Chennai. His interests are in Mobile Ad-hoc Networks, Network Security, and Network Attacks. Privacy preservation, Trust Evaluation and cloud computing.

Chellavelu Vijayakumaran, He received his Bachelor degree in Computer Science & Engineering from Madras University, and Master Degree from SRM University in 1994 and 2005, respectively. He is a research student of AISECT University, Bhopal, India. Currently, he is working as an Assistant Professor at the department of Computer Science and Engineering, Valliammai Engineering College, Chennai. His research interests include Mobile Ad-hoc Networks, Cloud Computing, Pervasive computing and Network Security.

Hyunsung Kim, He is a professor at the Department of Cyber Security, Kyungil University, Korea from 2012. He received the M.S. and Ph.D. degrees in Computer Engineering from Kyungpook National University, Republic of Korea, in 1998 and 2002, respectively. From 2000 to 2002, he worked as a senior researcher at Ditto Technology. He had been an associate professor from 2002 to 2012 with the Department of Computer Engineering, Kyungil University. His research interests include cryptography, VLSI, authentication technologies, network security and ubiquitous computing security.