A Novel Time-bound Hierarchical Key Assignment Scheme for Mobile Agent

Chia-Hui Liu, Yu-Fang Chung, Jin-De Jhuo, *Tzer-Shyong Chen, Sheng-De Wang

***Abstract*—A mobile agent is a kind of software with the abilities of autonomy and mobility. Mobile agents can switch themselves among the hosts connected in a heterogeneous networking. Mobile agents can independently finish the assigned tasks and also can communicate with the other agents with the property of less transmission delay. The goal of efficiency and availability in a distributed system can be achieved by the mobile-agent technology. Nowadays mobile agents have been widely used in e-commerce, wireless communication, information security and so forth. However, when a mobile agent performs its task, it can exchange messages with the other agents over the Internet. In such way, mobile agents will encounter some security threats such as data stolen or data falsified. Thus Volker and Mehrdad proposed a tree-structured access control scheme in 1998 to secure the mobile agents by adding the key management technology so as to achieve the goal of security and confidentiality. Two drawbacks, larger amount of duplicated keys and key-derivation computations, make their scheme inefficient. Thus, a hierarchical structure of mobile agents is proposed in this paper. In this structure, users are restricted to access data only in a range of legal time. Also the technique of bilinear pairings over elliptic curves is used to construct a time-bound key management for controlling the access of private keys. From the security analyses, the security and efficiency of mobile agents can be achieved in the proposed scheme.**

***Keywords*—Mobile agent, access control, key management, bilinear pairings, time bound**

I. Introduction

Owing to the popular networking, connected hosts now will share their information with one another, which obviously impels the technological developments of an open distributed system. Imagine a great deal of information transmitted forward and backward in the Internet, which should overload the networking of the distributed system. In order to solve this problem, the mobile-agent applications are developed.

Chia-Hui Liu is a Ph.D student in the Electrical Engineering Department of National Taiwan University.

Yu-Fang Chung is currently an associate professor in the department of Electrical Engineering at Tunghai University.

Jin-De Jhuo is a B.S. student in the Information Management Department of Tunghai University

Tzer-Shyong Chen is currently a professor in the department of Information Management at Tunghai University. (e-mail: arden@thu.edu.tw)

Sheng-De Wang is currently a professor in the department of Electrical Engineering at National Taiwan University.

A mobile agent is a kind of software with the abilities of autonomy and mobility. A mobile agent is sent to the Internet; afterwards, it can control itself and move freely among different hosts [4, 5]. It can transmit messages to the other mobile agents and also interact or distribute resources among the other distributed systems. With the mobile-agent technology, our life becomes more convenient, which makes the further development of mobile agents more valuable.

The illustration of Figure 1 is the simple concept of mobile agents.Basically Internet is an open and distributed environment. When a mobile agent performs its task, it can exchange messages with the other agents over the Internet. In such way, mobile agents will encounter some security threats. Therefore, different structures of mobile agents are designed so as to reduce the threats. The SOMA structure proposed by Corradi et al. [6] emphasizes it is suitable for an open and heterogeneous environment. The Ajanta structure proposed by Karnik and Tripathi [7] uses the characteristics of Java to avoid threats. The tree-based structure proposed by Volker and Mehrdad [1] secures the mobile agents by adding the technology of key management and proposes an access control scheme to achieve the goal of security and confidentiality. However, this structure will dispose the keys of mobile agents and make mobile agents become too large. Apart from those, the key-derivation computations are also enormous such that a large of memory space is wasted. Hence, this method is not efficient.

To overcome the above-mentioned drawbacks, we will propose a hierarchical structure to reduce the consumed space for keys in this paper. Nevertheless, the key-derivation computations would be still large. The characteristic of ID-based in bilinear pairing [2] will be applied to build a hierarchical structure of mobile agents with the technology of key management. This method shows the goal of reducing the amount of keys and decreasing key-derivation computations.

Here one question is taken into considerations. It is the responsibility of the owner of a mobile agent to regularly update the information of his mobile agent. If the mobile agent does not set up a time restriction, the owner cannot control host access time effectively. Therefore, the owner can not ensure whether the data accessed by the host is updated or not. In management, this is inconvenient and unsafe.

Thus, we use the concepts of bilinear and the scheme proposed by Shang and Wangstaff Jr.[8] to structure a time-bound key management scheme. In our scheme, users are restricted to access data only in a range of legal time.

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Then, the security and efficiency of mobile agents can be achieved by the owners.

II. Related Work

A. Mobile agents

Mobile agents [3] are a new technique developed in recent years. Owing to the heavy load of information and service on the Internet, the user always needs to spend a lot of time and cost in collecting information or executing tasks. Therefore, the technique of mobile agents has been developed to substitute the user for completing heavy tasks. Using the technique of mobile agents can help the user to collect, filter and exchange information. So, it is suitable to apply in e-commerce [9].

A mobile agent is a software program with the characteristics of autonomy, mobility, community and cooperation. Because of these characteristics, a mobile agent can itself decide to leave for its corresponding host among different network systems after accepting an assigned task, and also can transmit the completed task backward or execute the next step efficiently. Moreover, a mobile agent has many advantages. It can reduce the network load. It can overcome the network delay. It can be used offline. It can pack the protocols. It can adapt itself to dynamic environments. Finally, it can possess innate heterogeneity.

To sum up, a mobile agent can not only reduce the consumption of network resources, but also share its network load among different hosts. It shows that it can improve the bandwidth insufficiency on an instable network environment. Though a mobile agent has so many advantages, it got some security problems into discussions[10]:

1. Integrity attacks: A malicious host may add, delete or modify the information in mobile agents such as execution codes. This will threaten the information integrity and execution integrity of mobile agents.
2. Denial of Service: A malicious host may refer mobile agents to access resources, delay server time or refuses mobile agents sent to the next host.
3. Confidentiality attacks: A malicious host can monitor or analyze the data held by mobile agents such as execution codes or the status in order to obtain confidential data during the mobile agent is executing tasks. So, the confidentiality of hosts can't be ensured.
4. Authentication risks: A malicious host may pretend to be the visited host or copy the mobile agent so that the mobile agent cannot be authenticated by the other hosts.

B. Volker and Mehrdad's scheme

A mobile agent [3] may be attacked by malicious hosts if it is not protected by a security scheme. When it executes tasks over the Internet, it faces the possibility of data stolen or data falsified. Therefore, Volker and Mehrdad proposed a tree-based structure of mobile agents in 1998 [1]. Their goal is to prevent the data, the execution codes and information of mobile agents from being attacked when agents are roaming over the Internet. Figure 2 is the structure proposed by Volker and Mehrdad. Basically, each node in the tree-based structure represents a file or a folder. To avoid being attacked by the other hosts, a mobile agent uses a well-designed key management scheme to deal with the access control of data in the structure.

The whole tree-based structure is divided into two folders named Static and Mutable [11] respectively. Mobile agents will construct a corresponding folder for a legitimate remote host under the folder of static/sctx/acl (see Figure 3). All the decryption keys used by the corresponding hosts to access confidential data are stored under those folders. Each folder is encrypted by the certain public and private keys of a remote host. Therefore, each host can only use its own private key to decrypt and access its corresponding folder. So, the private key of the confidential data can be protected properly.

Though the scheme proposed by Volker and Mehrdad can protect confidential data, prevent data being stolen, and avoid malicious attacks by other hosts, the scheme did not consider the efficiency of the key management. The scheme has to store keys repeatedly, which would cause mobile agents and the computation of key derivation to overload. An ideal mobile agent should not waste storage spaces. So we try to present an ideal scheme to amend these faults and to improve the tree-based structure in the next section. The computations of key derivation and key generation become more efficient.

C. Time-Bound Hierarchical Key management scheme

In 2004, Chien [12] proposed a time-bound hierarchical key assignment scheme based on a tamper-resistant device and a secure hash function. But this scheme has a security drawback against Yi's three-party collusion attacks [13]. So Shang and S. Wangstaff Jr. proposed another scheme based on elliptic curve cryptography and the hierarchical structure inspired Chien [8]. The purpose is to restrict a user at specific time interval when he accesses the information in class C_i so that this scheme combines the key management with the concept of time-bound. At first, the system is supposed to have already generated many different classes of nodes marked with the policy configurations which are the set of policies applying to a node. Then the system will generate the key K_i for class C_i according to the following steps.

1. The system's owner chooses an elliptic curve E over a finite field IF_q.
2. He chooses a point $Q \in E(IF_q)$ with a large prime order p
3. He chooses 2n integers n_i and g_i such that $n_i g_i$ are all distinct modulo p for $1 \le i \le n$
4. He computes $P_i = n_i Q$ on $E(IF_q)$ and h_i such that $g_i h_i \equiv 1 \pmod{p}$
5. The key $K_i = g_i P_i$ is computed for class C_i

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Suppose two classes in the hierarchy with the relation $C_j \prec C_i$. The system will compute $R_{i,j}=g_iK_j+(-K_i)$ for the relation and then publish $R_{i,j}$ on the authenticated board. The system will encrypt class C_i in a symmetric encryption algorithm and set up a time interval $[t_1, t_2]$ of access control policy acp_i for class C_i to restrict the users to decrypt class C_i only at the time granule t which must be in the time interval $[t_1, t_2]$, that is $t_1 \le t \le t_2$. So the system's owner will choose two random integers a, b and a hash-key message authentication code $H_{K(\cdot)}$ [14]. Note that the authentication code is built by a hash function $H(\cdot)$ and a fixed secret key K. Here K is the system's master key and is only known to the owner. And then the system's owner computes $K_{i,t}$:

$$K_{i,t}=H_k((K_i)_Y \oplus H^t(a) \oplus H^{Z-t}(b) \oplus ID_i)),$$

where $(K_i)_Y$ is the *y-coordinate* of K_i, $H^t(a)$ is the t-mold iteration of $H(\cdot)$ applied to a, ID_i is the identity of the class C_i, and $\oplus$ is the bitwise XOR.

D. Bilinear Pairing

In 1984, Shamir [15] proposed an identity-based cryptosystem, also called ID-based Cryptography. The simple personal information of a user is regarded as the public key of the user in his system. But this system is inefficient. So Boneh and Franklin [2] used the technique of pairings to develop an efficient ID-based encryption (IBE) system. It has had a rapid development until 2001. However, the pairing used in this paper refers to Weil pairing and Tate pairing [16]. Both of them have the same characteristics of bilinear pairings. First, they assumed p to be a prime such that $q \mid p-1$ for some large prime q and let G_1 and G_2 denote two cyclic groups of the same prime order q, where G_1 was a subgroup of the additive group of points on an elliptic curve E over F_p, and G_2 was a subgroup of the multiplicative group of a finite field F^*_{p2}. The bilinear mapping function is $\hat{e}$: $G_1 \times G_1 \to G_2$ which satisfies the following conditions:

1. Bilinear:
   (1). $\hat{e}(P+Q, R) = \hat{e}(P, R) \cdot \hat{e}(Q, R)$
   (2). $\hat{e}(P, Q+R) = \hat{e}(P, Q) \cdot \hat{e}(P, R)$
   (3). $\hat{e}(aP, bQ) = \hat{e}(bP, aQ) = \hat{e}(P, Q)^{ab}$
   for all $P, Q, R \in G_1$ and $a, b \in Z_q^*$.
2. Non-degenerate:
   If P is a generator of group G_1, then $\hat{e}(P, P)$ is a generator of group G_2.
   There exists $\hat{e}(P, P) \neq 1$.
3. Computable:
   There exist efficient algorithms to compute $\hat{e}(P, Q) \in G_2$ in polynomial time for all $P, Q \in G_1$.

III. The Proposed Scheme

We propose a new scheme to amend the drawbacks in Volker and Mehrdad's[1] scheme and construct a key management with time-bound constraints in this paper. We use a hierarchical structure to organize mobile agents and use the technique of bilinear pairings over elliptic curves to construct a time-bound key management for controlling the access of private keys [18-20]. By the characteristics of ID-based systems in bilinear pairings, we can use the identity of server S_i in this structure. We set up the restricted time interval $[t_1, t_2]$ to ensure server S_i can obtain the key DK_m at time granule t, which is denoted as the restricted time interval. So the scheme we proposed not only can amend the drawbacks in Volker and Mehrdad's scheme efficiently but also can ensure server S_i to obtain the key to decrypt the confidential documents only in the legal time interval.

In Figure 4, internal node S_i represents the folders corresponding to the server, where a hierarchical relation exists. If a higher server S_i can access a lower server S_j, then the relation $S_j \le S_i$ exists. Leaf node DK_m represents the decryption key used to encrypt/decrypt confidential files.

A. Initialization

A mobile-agent owner sets up the parameters such as the lifetime Z of the mobile agent, which servers will be visited, which route probable goes, and what is the access control policy of the agent. Then the owner of the mobile agent performs the following steps before commanding the mobile agent to execute tasks on the Internet.

Step1: Select non-repeated random integers $\{DK_1, DK_2, \ldots, DK_m\}$(supposing there are m confidential files) as the decryption key for encrypting or decrypting confidential files, and choose a bilinear map $\hat{e}$: $G_1 \times G_1 \to G_2$ and $P_0 \in G_1$.

Step2: Generate three one-way hash functions:
H_1 : $\{0, 1\}^* \to G_1$
H_2 : $G_2 \to \{0, 1\}^*$
H: the hash function is only used to compute time

Step3: Select $\alpha \in Z_q^*$ randomly, and calculate $P_{pub}=\alpha \times P_0$

Step4: Calculate $Q_{Si}=H_1(ID_{Si})$ and $D_{Si}=\alpha \times (Q_{Si})$ for each internal node (S_i), where D_{Si} is secret, and Q_{Si} is public .

Step5: Choose a random value a and set up the legal time interval $[t_1, t_2]$ and compute T_b, where $T_b=H^{t1}(a)$, and T_e, where $T_e=H^{Z-t2}(a)$. Here t_1 is the number of fold iteration of hash function H applied to a and Z-t_2 is the number of fold iteration of hash function H applied to a. Finally, disclose $DecInf=(T_b, T_e)$.

Step6: Choose a random value r and calculate $U=r \times P_0$, then construct the function given below for each leaf node (DK_m)

$$F_{DK_m}(x) = DK_m \oplus \left\{ \prod_{S_j \le S_i} \left[x \oplus H_2(g_{S_i})^r \right] \right\} \oplus H^{Z-2t}(H^t(a)) \oplus H^{Z-t}(a) \quad (1)$$

where $x = H_2(\hat{e}(D_{S_i}, U))$, $g_{S_i} = \hat{e}(Q_{S_i}, P_{pub})$, t is the time granule that server S_i wants to obtain DK_m

B. Key Derivation Phase

When server S_i wants to obtain the key DK_m at time granule t, server S_i needs to perform the following steps:
Step1: Take $T_b=H^{t1}(a)$ and $T_e=H^{Z-t2}(a)$ into formula (2) and

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

(3) to compute $H^t(a)$and $H^{Z-t}(a)$ at time granule t.

The formula is given as follows:

$$H^t(a)=H^{t-tl}(T_b) \quad (2)$$

$$H^{Z-t}(a)=H^{t2-}t(T_e) \quad (3)$$

If the time granule t is illegal because of not in the time interval $[t_1, t_2]$, then $H^t(a)$ or $H^{Z-t}(a)$ will be unable to compute. So we use this computation to constrain server S_i to obtain DK_m at legal time. This computation is the time-bound constraint in the key management scheme.

Step2: When server S_i takes $H^t(a)$, $H^{Z-t}(a)$, the corresponding secret parameter D_{Si} and public parameter $U=r\times P_0$ into the equations F_{DKm}, it can obtain the key to decrypt the confidential data in the legal time interval.

IV. Analysis of Security

In this section, we will discuss and analyze the attacks our scheme may face and prove our scheme is secure to resist the attacks.

A. External Attacks

External attacks mean that a user who doesn't have any relationship with the system wants to obtain a desired file from it. But because an intruder can only obtain the public parameters of the mobile agent, he/she cannot obtain the secret parameters of it. Even if an intruder wants to obtain the key DK_m by deriving α, r, and a, it is very hard to do because he faces the Elliptic Curve Discrete Logarithm Problem: $P_{pub}=\alpha\times P_0$ where $P_0\in G_1$, $\alpha\in Z_q^*$ and the Bilinear Diffie-Hellman problem (BDHP) : $H_2(g_{Si})^r$ where $g_{Si}^r= \hat{e}(Q_{Si}, P_{pub})^r\in G_2$, $r\in Z_q^*$. In addition, the one-way property of hash function H ensures that it is unable to derive a only knowing the value of $H^t(a)$. So based on this reasoning, it is hard for an intruder to obtain the secret parameters.

B. Reverse Attacks

It is legal to use authorized data in the system for internal servers. So when an internal server wants to obtain the unauthorized data, he/she may try to use more authorized data than an intruder to endanger the system. We suppose that an internal server S_j wants to obtain the key DK_m by using the secret parameter D_{Si} of the higher server S_i. Then server S_j will try to use the computation $D_{Sj} = \alpha\times(Q_{Sj})$ to derive α. Or server S_j will try to use the public parameter Q_{Si} and derived α to obtain D_{Si}. But server S_j will face the Elliptic Curve Discrete Logarithm Problem (ECDLP) when doing the computation $D_{Sj}=\alpha\times(Q_{Sj})$, it is hard to derive α. Therefore it is impossible for lower server S_j to obtain the unauthorized key DK_m by using the secret parameter D_{Si} of higher server S_i. Similarly, it is impossible to derive DK_m by attacking the higher server for another malevolent server.

C. Collusion Attacks

Collusion attacks mean that a lot of internal servers want to derive the unauthorized key so as to obtain desired data by collecting the information they process. However, even if several servers cooperate to collect related public system parameters and their own secret parameters, they will still be unable to derive the secret parameter D_{Si} to obtain the wanted key. Because we calculate the related parameters through the method of one-way hash function and bilinear pairing over elliptic curves, it is infeasible to derive the message from $H(\cdot)$ which applied to the message and compute without solving the BDHP.

D. Equation hacking Attacks

The visited server S_6 can derive the key DK_4 by using the equation $F_{DK4}(x)$ and its secret parameter D_{S6}. If server S_6 wants to derive the secret parameter of server S_1 by using its own secret parameter and public parameter and the equation $F_{DK4}(x)$, he/she may derive it by using the following derivation.

$$F_{DK_4}(x)=DK_4\oplus[x\oplus H_2(g_{S_1}^r)]\times[x\oplus H_2(g_{S_3}^r)]\times[x\oplus H_2(g_{S_6}^r)]\oplus H^{Z-2t}(H^t(a))\oplus H^{Z-t}(a)$$

Then we change the position:

$$\Rightarrow F_{DK_4}(x)\oplus DK_4=[x\oplus H_2(g_{S_1}^r)]\times[x\oplus H_2(g_{S_3}^r)]\times[x\oplus H_2(g_{S_6}^r)]\oplus H^{Z-2t}(H^t(a))\oplus H^{Z-t}(a)$$

$$\Rightarrow \frac{F_{DK_4}(x)\oplus DK_4}{x\oplus H_2(g_{s_6}^r)}\oplus H^{Z-2t}(H^t(a))\oplus H^{Z-t}(a)=[x\oplus H_2(g_{S_1}^r)]\times[x\oplus H_2(g_{S_3}^r)]$$

In this step, because server S_6 can derive the key DK_4 by using the equation $F_{DK4}(x)$, the operation becomes the following form.

$$\Rightarrow \frac{DK_4\oplus DK_4}{x\oplus H_2(g_{s_6}^r)}\oplus H^{Z-2t}(H^t(a))\oplus H^{Z-t}(a)=[x\oplus H_2(g_{S_1}^r)]\times[x\oplus H_2(g_{S_3}^r)]$$

$$\Rightarrow \frac{0}{x\oplus H_2(g_{s_6}^r)}\oplus H^{Z-2t}(H^t(a))\oplus H^{Z-t}(a)=[x\oplus H_2(g_{S_1}^r)]\times[x\oplus H_2(g_{S_3}^r)]$$

If we want to obtain the secret parameter of server S_1, we must to obtain the secret parameter of server S_3 based on above equation. Though server S_6 can persuade server S_3 to cooperate with each other to derive the secret parameter of server S_1, it will remain unobtainable because they will face the Bilinear Diffie-Hellman problem. They can not solve the secret parameter r to obtain the secret parameter of server S_1 in our scheme.

V. Conclusion and Future Work

The popular networking along with the advanced development of wireless communication technology makes mobile agents a tendency nowadays and in the future too. Mobile agents with the properties of autonomy and mobility have become important for researches. Mobile agents can switch themselves among the hosts connected in a heterogeneous networking. Mobile agents can independently finish the assigned tasks and also can communicate with the other agents with the property of less transmission delay. However, when mobile agents perform

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

their tasks, they exchange messages with the other agents over the Internet. In such way, mobile agents will face some security threats such as data stolen or data falsified. Thus Volker and Mehrdad proposed a tree-structured scheme in 1998 to secure mobile agents by adding the key management technology so as to achieve the goal of security and confidentiality. Two drawbacks, larger amount of duplicated keys and key-derivation computations, make their scheme inefficient. Therefore, a hierarchical structure of mobile agents is proposed in this paper. In this structure, users are restricted to access data only in a range of legal time. Also the technique of bilinear pairings over elliptic curves is used to construct a time-bound key management for controlling the access of private keys. From the security analyses, the security and efficiency of mobile agents can be achieved in the proposed scheme.

ACKNOWLEDGMENT

This work was supported partially by National Science Council of Republic of China under Grants NSC 98-2221-E-029 -025.

REFERENCES

[1] R. Volker and J. S. Mehrdad, "Access Control and Key Management for Mobile Agents," *Computers and Graphics*, Vol. 22, No. 4, pp. 457-461, 1998.

[2] D. Boneh and M. Franklin, "Identity-based Encryption from the Weil Pairing," *Advances in Cryptology-Proceedings of CRYPTO 2001*, Springer-Verlag LNCS 2139, pp. 213-229, 2001.

[3] N. M. Karnik and A. R. Tripathi, "Design Issues in Mobile-Agent Programming Systems," *IEEE Concurrency*, pp. 52-61, Jul.-Sep. 1998.

[4] A. Karmouch, "Mobile Software Agents for Telecommunications," *IEEE Communications Magazine*, Vol. 36, No. 7, pp. 24-25, 1998.

[5] D. Chess, B. Grosof, C. Harrison, D. Levine, C. Parris and G. Tsudik, "Itinerant Agents for Mobile Computing," *IEEE Personal Communications*, Vol. 2, No. 5, pp. 34-49, 1995.

[6] A. Corradi, R. Montanari and C. Stefanelli, "Security Issues in Mobile Agent Technology," *Proceedings of the 7th IEEE Workshop on Future Trends of Distributed Computing Systems (FTDCS '99), IEEE Computer Society Press*, pp. 3-81, Cape Town, South Africa, 1999.

[7] N. M. Karnik and A. R. Tripathi, "A Security Architecture for Mobile Agents in Ajanta," *Proceedings of the International Conference on Distributed Computing Systems*, pp. 402-409, Taiwan, 2000.

[8] N. Shang, and S. Wagstaff Jr. "An Efficient Time-Bound Hierarchical Key Management Scheme for Secure Broadcasting," *IEEE Transactions on Dependable and Secure Computing*, Vol. 5, No. 2, Apr.-Jun. 2008.

[9] D. B Lange and M. Oshima, "Seven Good Reasons for Mobile Agents," *Communication of The ACM*, Vol. 42, No. 3, pp. 88-89, Mar. 1999.

[10] I. C. Lin, H. H. Ou and M. S. Hwang. "Efficient Access Control and Key Management Schemes for Mobile Agents," *Computer Standards & Interfaces*, Vol. 26, No. 5, pp. 423-433, 2004.

[11] F. Hohl, "A Model of Attacks Malicious Hosts against Mobile Agents," *Proceedings of the 4th Workshop on Mobile Object Systems: Secure Internet Mobile Computations*, Belgium, pp. 105-120. Jul.21, 1998.

[12] H-Y. Chen, "Efficient Time-Bound Hierarchical Key Assignment Scheme," *IEEE Trans. Knowledge and Data Eng*, Vol. 16, No. 10, pp. 1302-1304, Oct. 2004.

[13] X. Yi, "Security of Chien's Efficient Time-Bound Hierarchical Key Assignment Scheme," *IEEE Trans. Knowledge and Data Eng*., Vol. 17, No. 9, pp. 1298-1299, Sept. 2005.

[14] FIPS Publication 198, *The keyed-Hash Message Authentication Code* (*HMAC*), http://csrc.nist.gov/publications/fips/fips198/fips-198a.pdf, 2008.

[15] A. Shamir. "Identity-Based Cryptosystems and Signature Schemes," *Advances in Cryptology-Proceedings of CRYPTO'84*, *Springer-Verlag LNCS 196*, pp. 47-53, 1985.

[16] A. Joux, "The Weil and Tate Pairings as Building Blocks for Public Key Cryptosystems," *in Proceedings Fifth Algorithmic Number Theory Symposium*, *Springer-Verlag LNCS*, 2002.

[17] F. Bao, R. Deng and H. Zhu, "Variations of Diffie–Hellman Problem," *In Proceedings of ICICS 2003*, *Springer-Verlag LNCS 2836*, pp. 301-312, 2003.

[18] M. S. Hwang, "Extension of CHW Cryptographic Key Assignment Scheme in a Hierarchy," *IEE Proceedings-Computers and Digital Techniques*, Vol. 146, No. 4, pp. 219, 1999.

[19] S. G. Akl and P. D. Taylor, "Cryptographic Solution to a Problem of Access Control in a Hierarchy," *ACM Transactions on Computer Systems*, Vol. 1, No. 3, pp. 239-248, 1983.

[20] M. S. Hwang, "An Asymmetric Cryptographic Scheme for a Totally-ordered Hierarchy," *International Journal of Computer Mathematics*, Vol. 73, pp. 463-468, 2000.

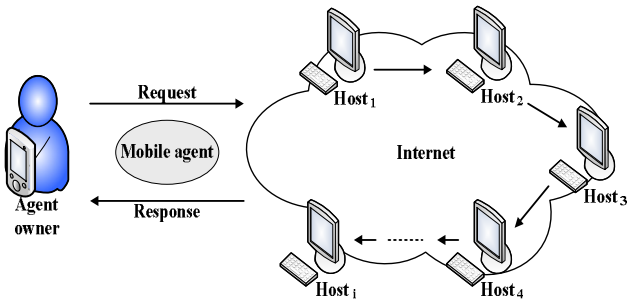

Figure 1: The concept of mobile agents

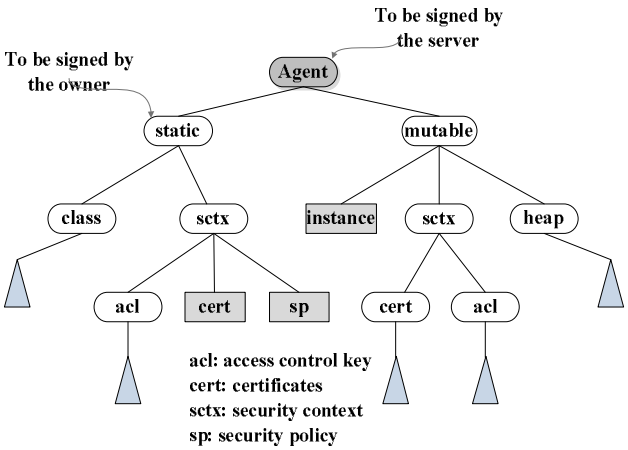

Figure 2: Tree structure of mobile agent

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

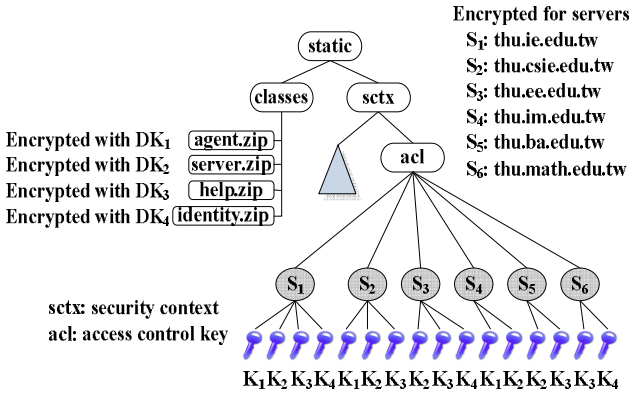

Figure 3: The example of access control scheme

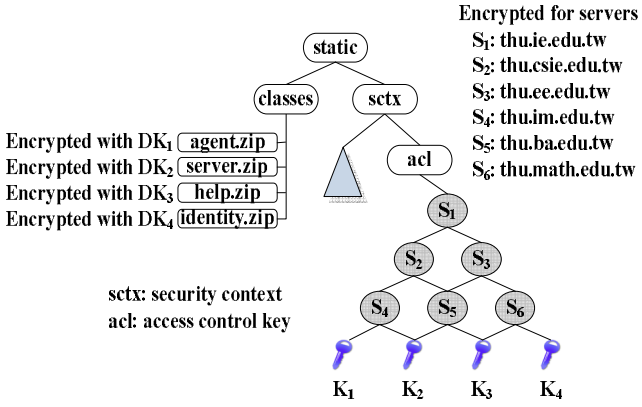

Figure 4: Structure of Mobile Agent based on hierarchy and bilinear pairings

ISBN: 978-988-17012-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)