Circuits and Systems, 2016, 7, 1421-1429
Published Online June 2016 in SciRes. http://www.scirp.org/journal/cs
http://dx.doi.org/10.4236/cs.2016.78124

Anonymous Authentication for Secure Mobile Agent Based Internet Business

Sivaraman Audithan[1], Thanjavur Swaminathan Murunya[2], Pandi Vijayakumar[3]

[1]Department of Electronics and Communication Engineering, P.R. Engineering College, Thanjavur, India
[2]Department of Computer Science and Engineering, P.R. Engineering College, Thanjavur, India
[3]Department of Computer Science and Engineering, University College of Engineering, Tindivanam, Tamil Nadu, India
Email: saudithan@gmail.com, murunya@gmail.com, vijibond2000@gmail.com

Received 25 March 2016; accepted 22 April 2016; published 9 June 2016

Abstract

Nowadays, mobile agents are an effective paradigm for accessing the information in distributed applications, especially in a dynamic network environment such as Internet businesses. In such kind of Internet based applications, access must be secure and authentication takes a vital role to avoid malicious use of the system. This kind of security has been provided by several previously proposed algorithms based on RSA digital signature cryptography. However, the computational time for performing encryption and decryption operations in the past literatures is very high. In this paper, we propose an anonymous authentication scheme which potentially reduces the overall computation time needed for verifying the legitimacy of the users. Comparing with previous anonymous authentication schemes, our proposed scheme provides more security and it is effective in terms of computation cost. The experimental results show that the proposed method authenticates the users with low computational time significantly.

Keywords

Mobile Agents, Authentication, Elliptic Curve Digital Signature, Computational Time, Servers

1. Introduction

Providing secure data transmission from one host to another in an open wireless network environment is a challenging issue. At the same time, the number of users at the various access points should be authenticated to avoid malicious users accessing the information. To overcome these problems, an anonymous authentication protocol should be performed which will prevent the possibility of various attacks. Nowadays, the mobile agent

How to cite this paper: Audithan, S., Murunya, T.S. and Vijayakumar, P. (2016) Anonymous Authentication for Secure Mobile Agent Based Internet Business. *Circuits and Systems*, **7**, 1421-1429. http://dx.doi.org/10.4236/cs.2016.78124

(MA) technology can play a vital role in distributed network and systems management. Broadly speaking, the term agent is employed to present a software entity with a well-defined character, usually working on behalf of a human being or other software component, which may be applied in a diversity of applications. A mobile agent is also called as a software agent which is not bound to work only in the system from which it is constructed. In practice, the use of a privacy preserving anonymous authentication protocol between the communicating parties is very important, so that the authenticated entities can send subsequent messages without repeated authentication steps, even if it is probable to authenticate each message. For accessing the information in a distributed application, mobile agent is used in an open wireless environment [1], [2] which can adapt itself to any kind of network environments autonomously. MA is a software component, which can act as a middleware between the users and the server. MA is also used to develop applications in an open, distributed and heterogeneous environment such as the Internet. Bilinear pairing based on elliptic curve cryptography (ECC) [3] is used to protect the mobile agent from various types of security attacks. ECC provides better security for mobile agents and safeguards the mobile agents from malicious attacks. Therefore, in this paper, we propose a new anonymous authentication scheme with privacy preserving during authentication based on bilinear pairing with less computational complexity.

The main contributions of this paper are summarized as follows.

1) To propose a computationally efficient privacy preserving anonymous authentication protocol that thwarts authentication attacks.
2) To provide integrity to messages during subsequent communication with the CA.

The remainder of this paper is organized as follows: Section 2 provides the features of some of the related works. Section 3 describes the overall system architecture and preliminaries. Section 4 discusses the proposed anonymous authentication framework. Section 5 highlights the security strength of our proposed algorithm. Section 6 analyzes the comparative performances of our proposed algorithm. Section 7 gives concluding remarks and suggestions on some future directions.

2. Literature Survey

There are numerous papers on authentication that are present in the literature [4]-[6]. Dilli Prasad Sharma *et al.* [7] proposed a new mobile agent model with an improved digital signature algorithm to support the execution of mobile code at mobile agent by providing better authentication in the distributed applications. It increases the security mechanisms by using the encrypted password with secret key for user authentication for that it maintains a database that's also migrated along with mobile agents on demand. The authentication server updates its database to defend its consistency. This provides better performance in distributed environment rather than in a centralized control environment. The negative aspect of the system is unreliable since distributed system is vulnerable to various failures and also it is necessary to address various fault tolerant metrics to improve the performance.

Berkovits *et al.* [8] proposed a novel secure architectural system model with enhanced trust relationship between the mobile agents. This model imparts authentication and authorization mechanisms to mobile agents. The proposed state appraisal function algorithm provides better authentication and access control mechanism to mobile agents. It is also able to detect malicious mobile agents to improve the trust relationship by selecting the appropriate privileges using the current state of the mobile agent. It improves the throughput by invoking dynamic execution of mobile code at mobile agent. In this paper, the state appraisal function algorithm, provides better security feature to the mobile agent and protects the server from various attacks. The major disadvantage of the system is very hard to maintain consistency among diverse range of servers located in various network locations.

Tao Feng, Xi Zhao *et al.* [9] proposed Typing Authentication Protection (TPA) strategy for enhancing virtual keyboard security features in mobile devices by coalescing login and post login modules. The login module inflicts the user authentication by accessing user biometric haptic feedback information from the user while accessing the virtual keyboard. The post login module in TAP strategy observes and appraises user's virtual key dynamics behavior by entailing various virtual key settings to constantly authenticate the user. In this scheme, the verification technique leads to overhead in terms of computational cost and storage complexity.

Basel and Radha [10] described indistinguishability under Chosen Plaintext Attacks (IU-CPA) algorithm for mobile devices which provides data privacy and authenticity. It invokes the security mechanisms in mobile devices by authenticating the ciphertext messages in the intended mobile receiver. It gives secure authentication by randomly generating short string messages which are to be added into the plaintext before pertaining suitable

encryption algorithm. The main limitation of this system is high computational cost, the sender and the receiver needs to perform additional computation to verify the authenticity which leads to high overhead.

C. Tang and D. O. Wu [11] proposed a novel authentication framework for low power mobile devices. This paper proposes an efficient authentication scheme, which reduces the computational cost as well as communication cost, as a result it is suitable for low-power mobile devices. It effusively preserves all known attacks allied with mobile networks including denial of service attacks by generating delegation passcode for mobile station authentication which make use of It make use of an elliptic-curve-cryptosystem based trust delegation mechanism. The disadvantage of this framework is the authentication delay will be degraded if the number of mobile node increases.

3. Overall System Architecture and Preliminaries

In this section, we describe our system architecture, work flow of our proposed work and bilinear pairing.

3.1. System Architecture

The overall architecture is shown in **Figure 1**. It consists of three main components, namely, central authority (CA), server and the authorized users (AU).

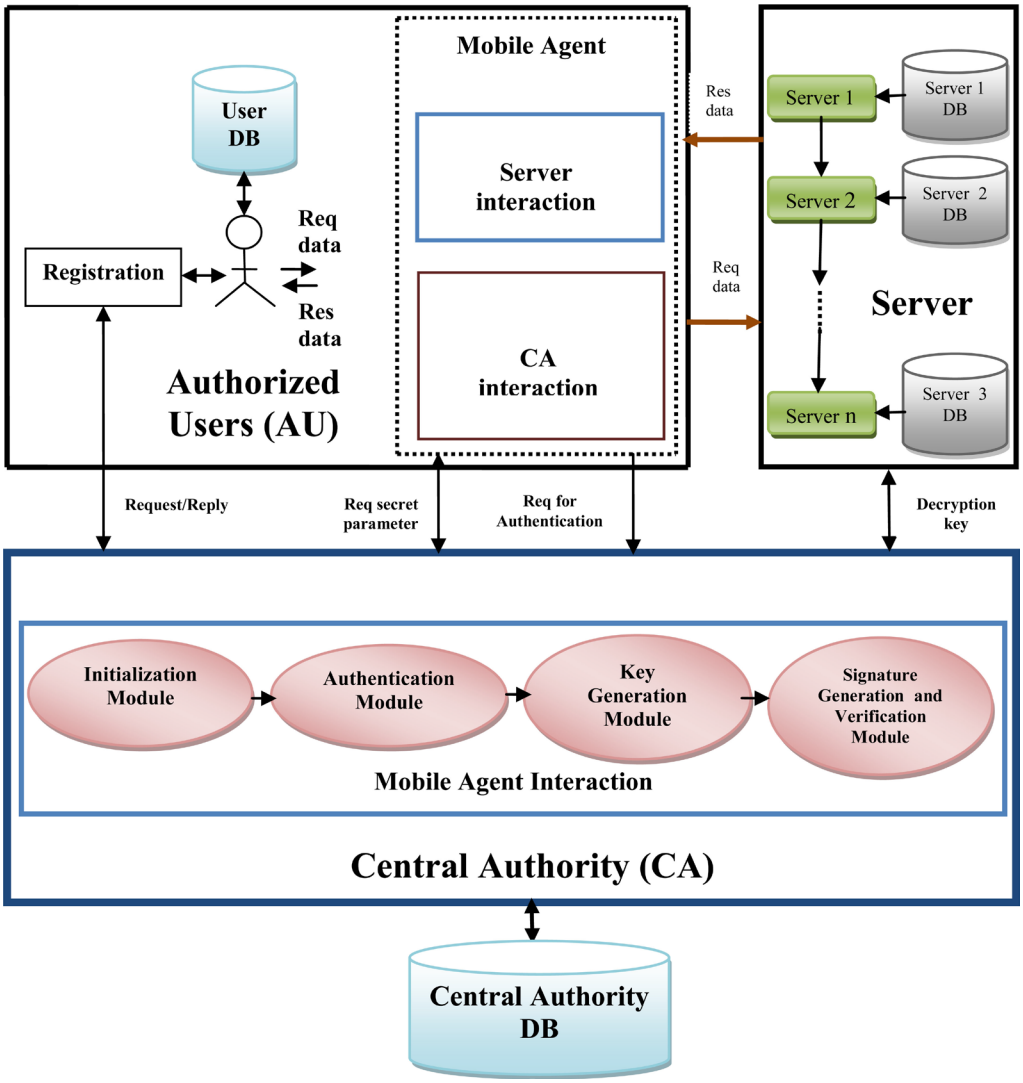

Figure 1. Architecture of the proposed key management scheme.

- Central Authority is an entity used to provide the necessary keys to AU and server and also it is responsible for maintaining all the security related information in a database (DB) called as Central Authority DB. The CA has four modules, namely initialization, key generation module, authentication module and signature verification module. The initialization module setups the system initial parameters and then publishes the necessary public parameters. The key generation module generates a private key for each user and then stores the key values and the public values in its DB. The key generation module also sends the private keys to each user in a secure way. The authentication module performs anonymous authentication to avoid communication with malicious entities. The signature generation and verification module is used to verify a digital signature generated by AUs to preserve the message integrity.
- The server is a component used to maintain the actual data in an encrypted format in the Server database (Server DB).
- Authorized Users are the Internet users who are allowed to access various files located on various servers in a distributed environment after successful authentication by the CA.
- For file access, file search and file decryptions we use a new component called mobile agent in the AU's area. MA is the software program [12] that can roam freely in the Internet environment from local host to other remote hosts in a network and execute tasks assigned by its user. Nowadays, mobile agents are not only used for distributed computation and data search in remote environments, but also used in network management and work flow system. The MA consists of three modules namely data Decryption, key signature check module and key derivation module. The key derivation module is used to derive all the lower level keys from a particular level to which the user has registered. The data decryption module is used to decrypt the data retrieved from various servers located in the distributed environment. The key signature check module is mainly used to check the signature that was created by the CA.

3.2. Work Flow

Initially, each authorized users completes the registration process with CA by sending a Request to CA. CA assigns a private key generated during the "Initialization phase" and sends it to each user by using SSL (Secure Socket Layer). Once a user wants to access the file, it sends request to the server by sending a request "Req Data" through the mobile agent. Then, the server sends the requested file to the MA in the encrypted format. The file request contains the file name and user identity (ID) of the user who is sending the request. The mobile agent submits the ID of the user and the file name to be accessed which was obtained from AU to CA. Then, CA checks whether the user is an authorized user by using the anonymous authentication process performed in the CA side through the authentication module and check the integrity of the request message by verifying the signature of the request message which was generated by the user. If the user is an AU then the CA derives the corresponding private key of the user from central authority DB by using the users ID.

Then, the CA mutually communicates with the server to get the decryption key of the particular file and then sends it to the mobile agent through a secure channel. After receiving the decryption key, the requested file is decrypted using the corresponding decryption key by the mobile agent and thus gives the response to the user by sending a reply "Res Data".

3.3. Bilinear Pairing

The properties of the bilinear operation are defined as follows: Let G_1 and G_2 denote additive cyclic groups, and G_T denote a multiplicative cyclic group of the same prime order p. Let g_1 be a generator of G_1, g_2 be a generator of G_2, and φ be an isomorphism from G_2 to G_1 such that $\varphi(g_2) = g_1 \cdot e : G_1 \times G_2 \to G_T$ is a bilinear map, which satisfies the following.

1) Bilinear: $e(g_1^a, g_2^b) = e(g_1, g_2)^{ab}$ for all $g_1 \in G_1, g_2 \in G_2$ and $a, b \in Z_p^*$.
2) Non degeneracy: $e(g_1, g_2) \neq 1_{G_T}$.
3) Admissible: Map e and isomorphism φ are efficiently computable.

4. Proposed Anonymous Authentication Scheme

The proposed anonymous authentication scheme is efficient in terms of computational overhead. The difference between our previous work [13] and this proposed key management is that the computation load taken by the

CA and user is reduced significantly by minimizing the number of mathematical operations. The proposed framework is briefly explained in the following

Step 1: The CA first chooses two random numbers $u, v \in Z_p^*$ as the masterkeys and computes $U_1 = g_1^u$, and $V_1 = g_1^v$. In addition, the CA chooses a public collision- resistant hash function: $\{0,1\}^* \to Z_p^*$. In the end, the CA publishes the system parameters $params = (p, G_1, G_2, G_T, e, g_1, g_2, U_1, V_1, H)$.

Step 2: When a user u_i with identity ID_i joins the system, the CA first chooses a random number $Pr_i \in Z_q^*$ such that $Pr_i + u \neq 0 \ mod \ q$ and computes $A_i = g_1^{1/Pr_i + u}$. Then, the CA stores (ID_i, A_i^u) in the storage list and returns $ASK_i = (Pr_i, A_i = g_1^{1/Pr_i + u})$ as the authorized anonymous secret key to the user.

Step 3: To access the data, each user sends the file request to the server by sending a request "Req Data" through the mobile agent. The file request contains the file name and user identity (ID) of the user who is sending the request.

$$\text{Req Data} = \{\text{File name} \| ID_i \| H(\text{File name} \| ID_i)\}$$

The mobile agent submits the "Req Data" which was obtained from the user to the CA. Then, the CA checks whether the user is an authorized user by using the following anonymous authentication process.

Step 4: *User Authentication*

The user runs the following steps to generate the anonymous short-life keys used for the authentication process,

1) The user u_i first choose l random numbers $x_1, x_2, \cdots, x_l \in Z_n^*$ as the short-life private keys and computes the corresponding public key $P_j = g_1^{x_j}$ for $j = 1, 2, \cdots, l$.
2) For each short-life public key Y_j, u_i computes the anonymous self-delegated certificate $Cert_j$.

- Randomly choose $r, q \in Z_p^*$ and compute

$$T_1 = g_1^{r+q}, \quad T_2 = g_1^{x_j + q}$$

- Compute $C = H(Y_j \| T_1 \| T_2)$ as well as

$$S_1 = g_1^{2q}, \quad S_2 = g_1^{r-q}, \quad S_3 = g_1^{x_j - q},$$

$$S_4 = A_i \cdot V_1^r, \quad S_5 = U_1^r$$

- Set $Cert_j = \{P_j \| S_1 \| S_2 \| S_3 \| S_4 \| S_5 \| C\}$ as the certificate.

3) Then the user generates a signature $\sigma = g_2^{\frac{1}{x_j + H(\text{File name} \| ID_i)}}$ and for message M and broadcast $msg = (\text{File name} \| ID_i \| \sigma \| P_j \| Cert_j)$.
4) If the certificate $Y_j \| Cert_j$ has not been checked the TA first computes $T_1' = S_1 \times S_2$, $T_2' = S_1 \times S_3$ and check whether $C = H(P_j \| T_1' \| T_2')$.

$$T_1' = S_1 \times S_2 = g_1^{2\beta} \times g_1^{\alpha - \beta} = g_1^{2\beta + \alpha - \beta} = g_1^{\alpha + \beta} = T_1$$

$$T_2' = g_1^{2\beta} \times g_1^{x_j - \beta} = g_1^{2\beta + x_j - \beta} = g_1^{x_j + \beta} = T_2$$

5) Once the certificate Y_j and $Cert_j$ has passed the verification, then the TA checks

$$e\left(g_1^{H(\text{File name} \| ID_i)} \cdot P_j, \sigma\right) = e(g_1, g_2)$$

Once the message File name and ID_i under the certificate $Cert_j = \{P_j \| S_1 \| S_2 \| S_3\}$ has been verified, then the CA uses its master keys (u, v) to compute

$$\frac{(S_4)^u}{(S_5)^v} = \frac{(A_i \cdot V_1^r)^u}{(U_1^r)^v} = \frac{(A_i^u \cdot V_1^{ru})}{(g_1^{ur})^v} = \frac{A_i^u \cdot g_1^{vru}}{g_1^{vru}} = A_i^u$$

Hence, the CA can authenticate the user.

Step 5: Then, the CA mutually communicate with the server to get the decryption key of the particular file and then sends it to the mobile agent through a secure channel. After receiving the decryption key, the requested

file is decrypted using the corresponding decryption key by the mobile agent and thus gives the response to the user by sending a reply "Res Data".

5. Security Analysis

In this section, security analysis of our proposed approach against three types of attacks namely internal collusion attack, date alteration attack and external attack are explained.

5.1. Message Integrity

Generally, the message integrity is achieved by verifying the signature attached with each message. In this scheme, the signature on message Req data is defined as $\sigma = g_2^{\frac{1}{x_j + H(\text{File name} \| ID_i)}}$. In this signature, the temporary short time private key x_j is used and so no other users can forge the signature. However, it is infeasible to perform message modification, because Elliptic Curve Discrete Logarithm Problem (ECDLP) would be difficult to decode. Moreover, there is a periodic change in thex_j value. Therefore, then it is infeasible to forge the signature. Since the vehicle certificates are generated using the vehicle is private key $uprk$ and short-time private key x_j. Hence, no other user can forge the certificate.

5.2. Source Authentication

This scheme can guarantee source authentication. The source authentication is performed using the master keys of the CA. The CA stores $\left(ID_i, A_i^u\right)$ in the storage list. The value A_i^u cannot be generated by anyone except the CA. No one can hack the value of *u* from the CA, because it is considered to be fully trusted and more powerful in security. Therefore, impersonation attack and bogus attack can be avoided due to the nature of source authentication.

5.3. External Attack

The external attackers attempt to find the $A_i = g_1^{1/Pr_i + u}$ value to access the protected data. In order to find the $A_i = g_1^{1/Pr_i + u}$ of the users, external attackers take $Cert_j = \left\{P_j \| S_1 \| S_2 \| S_3 \| S_4 \| S_5 \| C\right\}$. In this certificate, the external attacks need to break $S_4 = A_i \cdot V_1^r$ to find the A_i value. However, it is not feasible for them to derive the A_i value due to ECDLP.

6. Performance Analysis

In this section, we evaluate the performance of the proposed authentication in terms of computational cost. The computational cost is defined as the total time required for the CA to successfully authenticate the user. The computational cost of our authentication scheme is compared with many existing schemes BLS [14], ECPP [15], CAS [16], GSB [17], KPSD [18]. Let T_p is the time required for performing a pairing operation, T_h is the time required for performing a hash operation and the time required for performing one multiplication is T_m. The time needed to perform exponentiation operation in G_1 and G_2 are denoted as T_{ep-2} and T_{ep-2}. The proposed method is simulated on a P4 machine with 2 GB RAM running Cygwin 1.7.35 - 15 [19] with the gcc version 4.9.2 for our implementations.

From **Table 1**, it can be observed that our proposed scheme takes low computational cost among the various existing schemes to perform certificate and signature verification process. Because, our scheme takes only $2T_p, 2T_m$ and T_h for verifying one certificate & signature. Therefore, the proposed scheme can verify maximum numbers of signatures and certificates within 300 ms compared to BLS, ECPP, CAS, GSB and KPSD schemes. It can be seen that T_p and T_h are the most time-consuming operations in the signature verification process. Among the various existing schemes, our scheme use only two pairing operations for verifying one signature and requires only $(1+n)$ pairing operations for verifying *n* signatures. Therefore, our scheme takes less computational cost in comparison with all the existing schemes.

Table 2 shows the computation time measured in milliseconds generically for various functions that are used in various algorithms. When compared with all other functions, modulo operation takes less computation time

and Point Multiplication takes more computation time. Modulo operation takes 2.8 ms for 16 bit key values, 3.1 ms for 32 bit key values and 3.2 ms for 64 bit key values. Point Multiplication takes 14.2 ms for 16 bits, 29.0 ms for 32 bits and 36.4 ms for 64 bits.

For performing the hash operation, exponential operation, multiplication and pairing operation, the pairing-based cryptography (PBC) library [20] is used in this paper. For the aforementioned operations, the Type-A curve defined in the PBC library is used with the default parameters.

The results are analyzed over 50 randomized simulation runs and then the average of the results is considered as final. **Figure 2** clearly shows the authentication cost in ms for the number of the users. It can be seen that when n is large, the proposed authentication scheme is much more efficient than the other existing schemes and affords the lowest authentication cost among the schemes under comparison. It is very clear to understand that our proposed authentication scheme takes only 600 ms for. However, other existing schemes take more than 800 ms for authenticating 100 users.

Table 1. Comparison of computational cost of our authentication scheme with existing schemes.

Method	For single user	For n users
BLS	$4T_p + 2T_h$	$(2n+2)T_p + 2nT_h$
ECPP	$3T_p + 11T_m + T_h$	$3nT_p + (10+n)T_m + nT_h$
CAS	$5T_p + 2T_h$	$(4n+1)T_p + 2nT_h$
GSB	$3T_p + 4T_{ep-1} + 5T_{ep-2} + T_h$	$3nT_p + 4nT_{ep-1} + 5nT_{ep-2} + nT_h$
KPSD	$4T_p + 5T_{ep-1} + 5T_{ep-2} + T_h$	$(3+n)T_p + (4+n)T_{ep-1} + 5nT_{ep-2} + nT_h$
Proposed scheme	$2T_p + 2T_{ep-1} + T_h$	$(1+n)T_p + 2nT_{ep-1} + nT_h$

Table 2. Computation time complexities of various functions.

	16 bits (ms)	32 bits (ms)	64 bits (ms)
Mod	2.8	3.1	3.2
Hash	2.9	3.9	4.6
Point Addition	4.2	4.4	4.7
Multiplication	4.8	5.0	5.4
Inverse	5.3	5.4	6.0
Point Multiplication	14.2	29.0	36.4

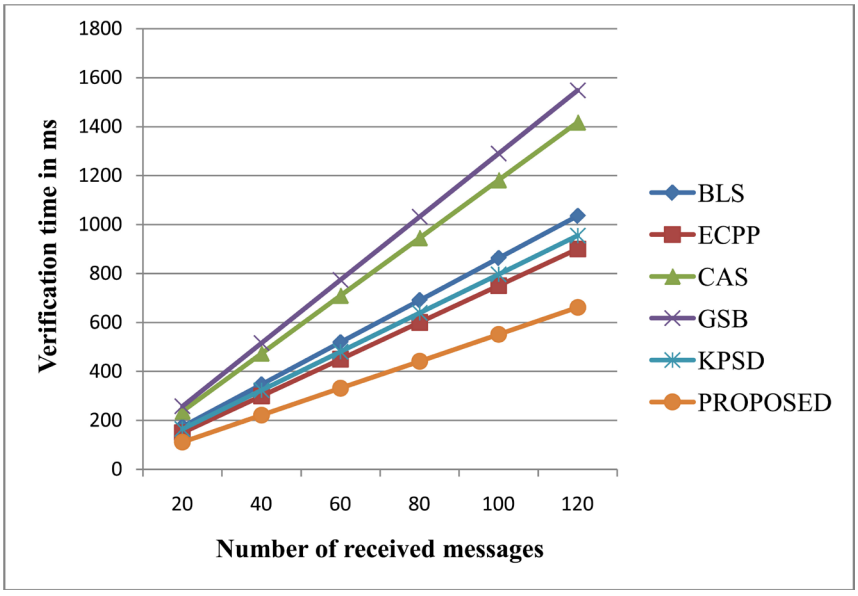

Figure 2. Computational cost of different authentication schemes.

7. Conclusion

The explosive growth of network environments requires adequate and effective security services such as authentication and message integrity for such networks. In this paper, an anonymous authentication scheme is proposed to authenticate the user. If the user is not a legitimate user, then the user cannot access any information from the server. The result shows that the proposed scheme is suitable and can be applied in the Internet environment and it is not easy for an attacker to malicious access. Thus, the main contribution of this proposed scheme is to secure data which are transmitted in Internet business application through the mobile agent. The future extension of this work is to provide confidentiality to the files to be accessed. In addition to that, the mobile agents created in one network environment can change its state to another network and so the authentication process should be performed to verify the mobile agent to improve the overall efficiency and security of the system.

References

[1] Karmouch, A. (1998) Mobile Software Agents for Telecommunications. Guest Editorial. *IEEE Communications Magazine*, **36**, 24-25. http://dx.doi.org/10.1109/MCOM.1998.689627

[2] Lange, D.B. and Oshima, M. (1998) Programming and Deploying Java Mobile Agents with Aglets. Addison-Wesley Press, Massachusetts.

[3] Koblitz, N. (1987) Elliptic Curve Cryptosystems. *Mathematics of Computation*, **48**, 203-209. http://dx.doi.org/10.1090/S0025-5718-1987-0866109-5

[4] Li, M., Poovendran, R. and Berenstein, C. (2002) Design of Secure Multicast Key Management Schemes with Communication Budget Constraint. *Communications Letters*, **6**, 108-110. http://dx.doi.org/10.1109/4234.991148

[5] Poovendran R. and Baras, J.S. (2001) An Information-Theoretic Approach for Design and Analysis of Rooted-Tree-Based Multicast Key Management Schemes. *IEEE Transactions on Information Theory*, **47**, 2824-2834. http://dx.doi.org/10.1109/18.959263

[6] Trappe, W., Song, J., Poovendran, R. and Liu, K.J.R. (2003) Key Management and Distribution for Secure Multimedia Multicast. *IEEE Transactions on Multimedia*, **5**, 544-557. http://dx.doi.org/10.1109/TMM.2003.813279

[7] Sharma, D.P. (2015) Mobile Agent-Based Authentication. A Model for User Authentication in a Distributed System. *International Journal of Computer Applications*, **112**, 975-8887.

[8] Berkovits, S., Guttman, J.D. and Swarup, V. (1998) Authentication for Mobile Agents. In: Vigna, G., Ed., *Mobile Agents and Security*, Springer-Verlag, LNCS 1419, Berlin.

[9] Tang, C. and Wu, D.O. (2013) Continuous Mobile Authentication Using Virtual Key Typing Biometrics. *The* 12*th IEEE International Conference on Trust*, *Security and Privacy in Computing and Communications* (*Trust Com*), Melbourne, 16-18 July 2013, 1547-1552.

[10] Alomair, B. and Poovendran, R. (2010) Efficient Authentication for Mobile and Pervasive Computing. *The12th International Conference on Information and Communications Security*, Spain, 15-17 December 2010, 186-202.

[11] Tang, C. and Wu, D.O. (2008) An Efficient Mobile Authentication Scheme for Wireless Networks. *IEEE Transactions on Wireless Communications*, **7**, 1408-1416. http://dx.doi.org/10.1109/TWC.2008.061080

[12] Roth, V. and Sohi, M. (1998) Access Control and Key Management for Mobile Agents. *Computer Graphics*, **22**, 457-461. http://dx.doi.org/10.1016/S0097-8493(98)00035-1

[13] Vijayakumar, P., Anand, K., Bose, S., Kannan, A., Maheswari, V. and Kowsalya, R. (2012) Hierarchical Key Management Scheme for Securing Mobile Agents with Optimal Computation Time. *Procedia Engineering*, **38**, 1432-1443. http://dx.doi.org/10.1016/j.proeng.2012.06.177

[14] Boneh, D., Gentry, C., Lynn, B. and Shacham, H. (2003) Aggregate and Verifiably Encrypted Signatures from Bilinear Maps. *Advances in Cryptology*, *EUROCRYPT*, **2656**, 416-432. http://dx.doi.org/10.1007/3-540-39200-9_26

[15] Lu, R., Lin, X., Zhu, H., Ho, P. and Shen, X. (2008) ECPP: Efficient Conditional Privacy Preservation Protocol for Secure Vehicular Communications. *INFOCOM* 2008, *IEEE the* 27*th Conference on Computer Communications*, Phoenix, 13-18 April 2008, 1229-1237. http://dx.doi.org/10.1109/infocom.2008.179

[16] Gong, Z., Long, Y., Hong, X. and Chen, K. (2007) Two Certificateless Aggregate Signatures from Bilinear Maps. *The* 8*th ACIS International Conference on Software Engineering*, *Artificial Intelligence*, *Networking*, *and Parallel/Distributed Computing*, **3**, 188-193. http://dx.doi.org/10.1109/snpd.2007.132

[17] Lin, X., Sun, X., Ho, P.-H. and Shen, X. (2007) GSIS: A Secure and Privacy Preserving Protocol for Vehicular Communication. *IEEE Transactions on Vehicular Technology*, **56**, 3442-3456. http://dx.doi.org/10.1109/TVT.2007.906878

[18] Lin, X., Lu, R. and Luan, T.-H. (2012) Pseudonym Changing at Social Spots: An Effective Strategy for Location Pri-

vacy in VANET. *IEEE Transaction on Vehicular Technology*, **61**, 86-96. http://dx.doi.org/10.1109/TVT.2011.2162864

[19] Cygwin: Linux Environment Emulator for Windows. http://www.cygwin.com/

[20] Pairing-Based Cryptography Library. http://crypto.stanford.edu/pbc/