Mobile Agent for Secure Web-Service: A Survey

Debashis Roy, Katayoon Moazzami, Rachita Singh
School of Computer Science, University of Windsor, Canada
{roy17, moazzam, singh12s}@uwindsor.ca

Abstract

Mobile agent technology and web services compensate each other and play very important roles in e-service applications. Web service provides an open standard for distributed services. It is used in Internet and wireless mobile devices. On the other hand a mobile agent is a composition of computer software and data which migrates from one host to another. The security issues of integrating mobile agent and web service technology has been actively investigated in recent years. In this paper, we present a survey of three papers which deals with securing mobile agent and web service integration.

1 Introduction

"The term Web services describes a standardized way of integrating Web-based applications using the XML, SOAP, WSDL and UDDI open standards over an Internet protocol backbone. XML is used to tag the data, SOAP is used to transfer the data, WSDL is used for describing the services available and UDDI is used for listing what services are available" [5]. This technology was originally developed for B2B communication to allow organizations to communicate data without intimate knowledge of each other's IT systems. But in recent times this technology is widely used in B2C purpose. The increasing complexity of web service composition and use of wireless technology and mobile devices such as mobile phone, PDA etc requires mobile agents to increase efficiency and ease of web service integration, but all these introduce much more security and privacy issues.

In the last couple of years many researchers have done considerable amount of work to enforce security in web service integration. But this survey is based on three papers which specifically deal with securing web-service and mobile agent integration. Among these three papers that have been discussed in this survey, the first one introduces new delegation model for securing agent-based web service integration. This model uses the traditional public key cryptography system to encrypt/decrypt every communication. The other two methods were proposed by the same authors. Both of the papers introduce new encryption schemes to simplify the key management among the users, agents and web-service providers. One of these papers uses Bilinear Pairings and Bilinear Diffie-Hellman (BDH) and the other uses Boneh-Franklin ID-based public key scheme.

2 Why Mobile Agent?

"A mobile agent is a composition of computer software and data which is able to migrate (move) from one computer to another autonomously and continue its execution on the destination computer" [6]. In other words a mobile agent is a mobile executable object and it can be dispatched by its owner so that it can migrate in the network independently and complete the tasks specified by its user. After finishing the task the agent can migrate back to the owner with collected data. During this period it does not need continuous network connection as the conventional RPC (Remote Procedure Call). Therefore it is applicable to devices with limited bandwidth and resources and long-term transactions without continuous interaction.

3 Agent-based Delegation Model

Hwang et. al. [1] proposed an agent-based delegation model for securing web service in ubiquitous computing environments. Instead of directly communicating with the web services, all the communications between the user and the service provider are conducted through different agents. A user gives his/her credentials to his/her agents, the agents transfer user's credentials to web service providers, which then perform web services on their behalf. In order to increase efficiency the agents delegate their various rights to other agents.

According to the authors they have extended the AttributeStatement supported by SAML 1.1/2.0 [7] specification to transfer the delegation information among the user and the agents. They have introduced authentication/delegation authorities in their delegation model to verify a trust relationship and to remove the step of verifying an ordered delegation chain.

3.1 Overview of Delegation Model

Figure 1 presents the delegation model proposed in [1] to support web services in ubiquitous computing environments. The model assumes that a user can have two different kinds of roles. One role is as the consumer of the Web-Service Management Server (WSMS), and the other role is the consumer of Web-Service. The WSMS

grants roles to the user to access the web services. The least role granted by a web service management server is shared with web service providers based on user's retrieval conditions. The user and the agents delegate their rights to other agents with the help of the delegation authority. The authentications of the agents are done by the authentication authority. Following are the main components of this delegation model:

Web-Service Management Server (WSMS): this is the mediator between the users and the web service providers. WSMS system is based on the XACML model [4]. It manages the web services and the policies registers by the web service providers and assigns appropriate roles to the user.

Principal (P): a user who delegates his/her rights to agents to perform web services.

Principal Agent (PA): a software that is controlled by P and communicates with other agents on behalf of P.

Carrier Agent (CA) & Service Agent (SA): PA delegates its rights to CA and CA in turn communicates with other agents. Whereas SA verifies the validity of delegation assertion and processes P's service request.

Authentication Authority (AA): authenticates Ps or agents.

Delegation Authority (DA): issues delegation assertions to authenticated agents.

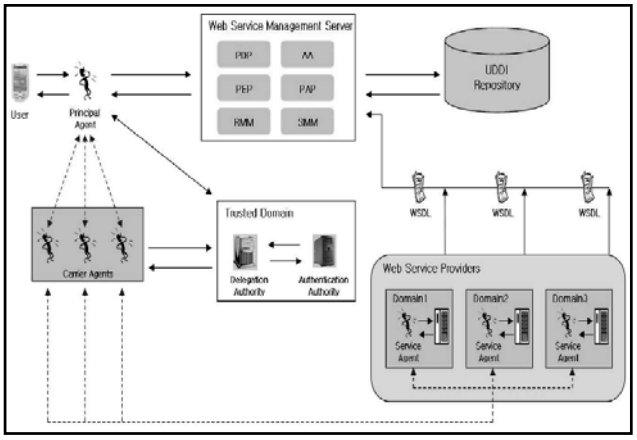

Figure 1: Agent-based delegation model

3.2 Delegation Assertion

Delegation assertion designed is based on the SAML specification. Delegation assertion indicates whether P or agent is capable of delegating their rights or not. To support integrity of the delegation assertion it is digitally signed by the delegation authority DA. For privacy protection P's information is encrypted with AA's public key so that it can be only decrypted by AA. The delegation assertion also contains additional information such as service provider's URL, inputs to the WSDL, least role, recipient agent PA etc. All these information are encrypted with service provider's public key in order to provide confidentiality. Before serving any request the service provider verifies the validity of the delegation assertion.

3.3 Delegation Interaction

Figure 2 presents the sequence diagram of the delegation interaction operations of the model proposed by Hwang et. al.

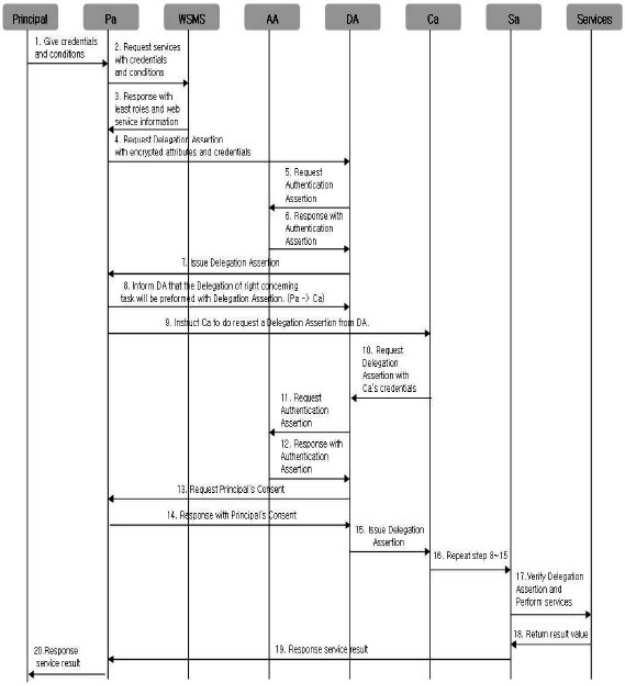

Figure 2: Interaction among components

1. P gives credentials to its agent PA.
2. PA requests web service information to WSMS with P's credentials.
3. WSMS retrieves web service information and returns it with least role as the consumer of web service to PA.
4. PA requests delegation assertion to DA and sends both credentials encrypted by the private key of P and attributes encrypted by the public key of web service.
5. DA requests authentication assertion to AA.
6. AA issues authentication assertion only after validating the credentials from DA.
7. Now DA issues delegation assertion to PA signed by private key of DA.
8. PA informs DA that the some or all of PA's rights related to its task will be delegated to CA.
9. PA gives its delegation assertion to CA and instructs to request for new delegation assertion for CA to DA.
10. CA request delegation assertion to DA with the delegation assertion and credentials of PA.

11. DA checks whether consent element in delegation assertion of PA is true or not and then requests authentication assertion to AA.
12. AA confirms the validity of CA's credentials, and issues authentication assertion for CA.
13. To set consent element of delegation assertion for CA, DA requests consent of PA.
14. PA responses to DA whether it consents to the delegation of its rights or not.
15. DA issues delegation assertion to CA.
16. CA can repeat from step 8 to step 15 if it needs mode delegation. Finally, CA requests service from SA, the agent of web service provider.
17. SA verifies the validity of CA's delegation assertion using DA's public key and requests web services to web service provider.
18. Web service provider decrypts encrypted attributes using its own private key, processes the request and responses the result to SA.
19. SA transfers the result of web services to PA.
20. PA responses the result of web services to P.

3.4 Discussion

According to the authors in the agent-based delegation model can agents can delegate their rights to other agents without any privacy disclosure. Also the principal agent PA has the complete control over all delegation operations. No agent can delegate its rights to other agent without PA's approval and also the authentication authority makes sure that only legitimate agents can obtain rights from other user. The communication between any two components of this model is encrypted with public key cryptosystem. Hence this model requires a considerable amount of time and resource for encryption and decryption. This is the only drawback of this model. As a future work the researchers wants to do further work to make this model suitable for mobile devices with limited memory capacity.

4 Bilinear Diffie-Hellman Public Key System

The paper "Mobile agent web service integration security architecture" discusses a new scheme for solving the security issues that arise along with the integration of mobile agents and the web services. This scheme verifies the mobile agent's identity without using a user-password pair. The mobile agent authentication consists of verifying the identity of the agent owner, agent and the server host. The role-based access control (RBAC) is used for mapping the users and the services; doing so several different services can be assigned to the user's privileges.

Since the mobile agent should not carry any a password or any key the scheme proposed in this paper adopts the ID-based authentication instead of the certification authority (CA), doing so only one key is required for encrypting a service that is available to a group of users.

This scheme is based on the computational Diffe-Hellman and the Bilinear Diffe-Hellman assumption. The assumption about the whole architecture of the system is that the web service provider consists of different web services to which users can be assigned and each of these users has a mobile agent. The users that are assigned to a specific web service resource form a group, in this case the web service provider acts as a key distribution centre (KDC) that distributes the keys allocated to each group of users. The users are free to join any web service resource and leave any one.

This scheme consists of different steps, these steps are: system setup, subscription, signature scheme, authentication scheme, encryption, decryption and re-keying.

4.1 System setup:

In the system setup step the WSP sets the system parameters, these parameters are: a prime number $p=2q+1$ where q is also a prime number, an additive group G_1 and a multiplicative group G_2 (these are the groups that the definitions that lead to BDH assumption are based on) these two groups have the order of p, a master key $s \in Z_q^*$ and a number P that belongs to the additive group G_1.

Also two hash functions $H_1:\{0,1\}^* \rightarrow G_1, H_2:\{0,1\}^* \rightarrow G_2$ are selected and the public key $P_{pub}=sP$ is calculated according to Bonneh's ID-based encryption algorithm. The private key for each user is computed using the ID of each user, the private key would be of the form

$s_{ID}=sQ_{ID}(Q_{ID}=H_1(ID))$.

4.2 Subscription:

Assuming that there are n users using l web service resources a $n \times l$ matrix is set where the ij th element is 1 if user i is a part of user group j and 0 otherwise.

The system setup and subscription are a part of system description and authorization phase.

Signature scheme:

The signature is a triple (R_i, S_i, m) where m is the message, $R_i=rQ_i$ and $S_i=(H_2(m,R_i)+r)s_{IDi}$.

4.3 Authentication:

The signed message can be authenticated using the public key and the user ID; this is done by checking if the following holds:

$e(S_i,P)=e(H_2(M_r,R_i)H_1(ID_i)+R_i,P_{pub})$

e is a computable bilinear map $e: G_1 \times G_1 \rightarrow G_2$ and for some $a,b \in Z_q$ and $P,Q \in G_1$, $e(aP,bQ)=e(P.Q)^{ab}$.

4.4 Encryption:

If we consider the *k*th service provider and let t_k be the number of users using this service resource and Q_t be the user's public key and M_k to be a session key or a message for this group of users, the following are calculated:

$Q_{v_{1k}} = \sum_{ik=1}^{t_k} Q_{ik}$ and $1(t_k - 1) \times t_k$ matrices denoted by a_{ik} are set up thus $Q_{v_{ik}} = (Q_{1k},...,Q_{tk}) \times a_{ik}$.

The ciphertext (U_{ik}, V_k) where $1 \le ik \le t_k$ can be obtained by:

$U_{1k}=r_kP, U_{ik}=rQ_{Vik}(2 \le ik \le t_k), V_k=M_k \oplus H_2(e(P_{pub}, r_kQ_{V1k}))$

where $r_k \in Z_q^*$ is a random number.

4.5 Decryption:

In this step the inverse of a matrix A which is equal to $\begin{pmatrix} a_{1k} \\ a_{2k} \\ \\ a_{tk} \end{pmatrix}$ where $a_{1k}=(0,...,1,...0)$ the i th is 1 for the user with ID_i. Using this inverse and the matrix $\begin{pmatrix} Q_{ik} \\ Q_{V_{2k}} \\ Q_{V_{tk}} \end{pmatrix}$ the $Q_{V_{1k}}$ and M_k (session key or a message for this group of users) would be calculated the following formula: $M_k = v_k \oplus H_2(e(P_{pub}, r_kQ_{v_{1k}}))$

4.6 Re-keying for member changes:

Re-keying of the group session key is done by the WSP when users change their group, leave a group or join a new one; this is done by changing the group registration matrix S (adding a new row when a member joins and removing a row when a member leaves) and recalculating the values of U,V.

4.7 Re-keying for web service changes:

In this case Re-keying is done by adding a new column in the matrix S and recalculating all the parameters calculated in previous steps.

4.8 Discussion

This scheme introduces a new authentication system that unlike the existing security schemes does not use Certification Authorities (CA), it also does not use the username/password pair. This method simplifies the key management, but on the other hand all the users must have their private key pair based on PKI, the server must verify manage and search all user's public keys and use different keys for different user's and calculate the username/password token.

5 Boneh-Franklin ID-based Public Key Scheme

Zhang et. al. [3] proposed a new mobile agent and web services security scheme. The security scheme employs an Identity-based public key system and provides a new authentication protocol without using the username/password pair, which is infeasible for mobile agents, and gives an alternative method for security mechanism without using the Certification Authorities (CA). It uses only one key for each service and one key for each user permanently. The authors have used ***Boneh-Franklin ID based public key*** scheme and presented a new security scheme for web-services.

5.1 System Description:

Authors assume that that the Web Service provider (WSP) acts as a Key Distribution Centre (KDC) and these have secure channels to distribute keys to the users.

Here l denotes the cardinality of the web service resources denoted as $r_1, r_2, ..., r_l$. All the users who subscribe to or are assigned to the same web service resource form a group (G) denoted as $G[1], G[2], ... G[l]$. These groups are $G[1]$ for accessing r_1, $G[2]$ for accessing r_2 and so on. The integration of mobile agent and web services can have three types of composition: parallel, sequential and combination.

5.2 New Scheme Setup:

The scheme proposed in [3] is based on the ID based encryption algorithm. The web service provider (WSP) selects system parameters p, q, G_1, G_2 and computes the system public key $P_{pub}=sP$ which are then sent to all members who have registered with WSP. WSP also selects two strong public one way functions $H_1: \{0,1\}^* \rightarrow G_1$ and $H_2: G_1 \rightarrow \{0,1\}^*$. This scheme requires the user to register with the WSP and become a member. The user has to provide his/her identity whenever he/she joins the group. The Web Service provider is used to authorize a user by sending him/her a private key $S_{ID}=sQ_{ID}$ where the $Q_{ID}=H_1(ID)$.

For n users and l services the web service provider maintains a $n \times l$ matrix S as follows:

$$S = \begin{pmatrix} S_{11} & \cdots & S_{1l} \\ \vdots & \ddots & \vdots \\ S_{n1} & \cdots & S_{nl} \end{pmatrix}$$

Where $S_{mk}=1$, $(1 \le k \le l \ \& \ 1 \le m \le n)$ if user u_m is a member of web service $G[k]$.

5.3 Authentication Scheme:

Each user has a unique identification ID_i. If we have a message M_r, the user U_i signs a message by selecting a random number $r \in \mathbb{Z}_q$, a generator $P \in G_1$ and a public

hash function $H_2:\{0,1\}^* \rightarrow \mathbb{Z}_q$ and computes $R_i \leftarrow rQ_i$ and $S_i \leftarrow (H_2(m, R_i) + r)sID_i$. The signature is the triple (R_i, S_i, M_r). WSP can verify the signature using the public key and the senders ID_i.
$e(S_i, P) = e(H_2(M_r, R_i)H_1(ID_i) + R_i, P_{pub})$.

5.4 Secure Web Service Scheme

The second part of the scheme describes about web service data encryption. This method assumes one encryption key for each service. The data can be encrypted in two different ways depending on its size. If the data set is not large then it is directly encrypted with the service encryption key. On the other hand for large data set, the data is first encrypted with a session key and then the session key is encrypted with the service encryption key. Once the mobile agent receives the encrypted data, it first decrypts the session key using its private key and then uses the session key to decrypt the data.

5.5 Re-keying for member changes and service changes:

When a member changes from one group $G[k]$ to another group $G[k']$, the WSP should re-key the corresponding web service group session key. There can be three cases which includes re-keying, whenever new member joins or existing member leaves or switches from one group to another. When a member switches from one web service group to another, a member U_m unsubscribes from a group $G[k]$ and subscribes to another group say $G[k']$ where $k \neq k'$, then the WSP updates the group registration matrix S.

Here the web service provider recomputed the polynomial function $f^k(x)$ to revoke the member U_m from $G[k]$, and then recomputes another polynomial function $f^{k'}(x)$ to add the member U_m to the data group $G[k']$. The function $f^k(x)$ is given as:

$$f^k(x) = \prod_{i'=1}^{n} \frac{x - s_{i'k}x_{i'}}{x^{(n-m)}} \bmod p = \prod_{i=1}^{m} (x - x_i) \bmod p$$

Where $m = \sum_{i=1}^{n} S_{ik}, (1 \leq i \leq n)(1 \leq k \leq l)$; k denotes the Web services group $G[k]$, i.e., all the members who subscribe to or are assigned to the Web service resource k.

5.6 Discussion

The authors have proposed a new ID-based public key management scheme for securing the integration of mobile agents and web services. According to the authors with this scheme, a mobile agent can be employed to autonomously search web services on behalf of the consumer, also the mobile agent owner can be verified before providing service to the user. The security scheme provides a new authentication protocol without using the username/password pair and gives an alternate method to current security mechanisms without using Certification Authorities (CA). Although this method simplifies the key management, the authors have also mentioned about some drawbacks of their scheme, such as, all the users must have his/her private key pair based on PKI, the service server must verify and manage all user's public keys, it has to search the user's public keys and use different keys for different user's and the username/password tokens are required to do the authentication.

6 Conclusion

The goal of this report is to survey some existing methods which try to take care of the security issues of mobile agent and web service integration process. In recent years there has been a good deal of works to incorporate mobile agents in web service integration. But for the sake of security and confidentiality all data that are being communicated among the users, agents or service providers have to be encrypted. We can use three different types of encryption: Symmetric encryption, asymmetric encryption and hybrid encryption. But symmetric key algorithms such as, DES (Data Encryption Standard) or AES (Advanced Encryption Standard) only use one key for encryption/decryption and for the sake of security the mobile agent cannot be allowed to take the one-and-only secret key with it. Hence symmetric key algorithms cannot be used in the mobile agent and web service system. On the other hand the asymmetric encryption and the hybrid encryption are based on the public key infrastructure (PKI). The PKI consists of a Certification Authority (CA) which issues certificates to different parties, it also helps to retrieve and revoke certificates. In PKI, firstly key pairs are generated for the users and web service providers and the public keys are registered with the registration authority. The certificate authority, which is a trusted third party, issues digital certificate with the public key. The XML Key Management Specification (XKMS) is used to provide the PKI service on the web. The agent based delegation model proposed in [1] uses the traditional public key system. However, [2] points out some drawbacks of the above architecture. First, all the users must have his/her public/private key pair based on PKI and the server has to manage and verify all the public keys. In addition to that the service server has to search the user's public key and use different keys to encrypt different messages for different users. Also the authors proposed two different methods [2, 3] for mobile agent and web service security, which do not need the CA based public key infrastructure. They introduced an ID based authentication scheme and proposed some techniques using two different types of encryption scheme to simplify the key management process.

7 References

[1] H. S. Hwang, H. J. Ko, K. I. Kim, U. M. Kim, D. S. Park, "Agent-Based Delegation Model for the Secure Web Service in Ubiquitous Computing Environments", *International Conference on Hybrid Information Technology, ICHIT '06*, Volume 1, pp.51-57, Nov. 2006.

[2] J. Zhang, Y. Wang, V. Varadharajan, "Mobile Agent and Web Service Integration Security Architecture", *IEEE International Conference on Service-Oriented Computing and Applications, SOCA '07*, pp.172-179, June 2007.

[3] J. Zhang, Y. Wang, V. Varadharajan, "A New Security Scheme for Integration of Mobile Agents and Web Services", Second International Conference on Internet and Web Applications and Services (ICIW'07), pp.43-48, May 2007.

[4] C. A. Ardagna, E. Damiani, S. De Capitani di Vimercati, P.Samarati, XML-based Access Control Language, 2004.

[5] Web services, http://www.webopedia.com/TERM/W/Web_services.html.

[6] Mobile Agent, http://en.wikipedia.org/wiki/Mobile_agent.

[7] SAML, http://en.wikipedia.org/wiki/SAML