A New Security Scheme for Integration of Mobile Agents and Web Services

Junqi Zhang, Yan Wang and Vijay Varadharajan
Department of Computing, Macquarie University
Sydney, Australia
{Janson, yanwang, vijay}@ics.mq.edu.au

Abstract

Web services specification provides an open standard for the distributed service oriented architecture. It is widely used in Internet and pervasive networks supporting wireless mobile devices. A mobile agent is a composition of computer software and data which is able to migrate from one host to another autonomously and continue its execution on the destination host. Mobile agent technology can reduce the bandwidth requirement and tolerate the network faults - able to operate without an active connection between clients and server. Hence, the applications of the combination of mobile agents and web service have been widely investigated in recent years. However, the security issue is still of a major concern. In this paper, we propose a novel agent-based web service security scheme. This scheme provides a new authentication protocol without using the username/password pair, which is infeasible for mobile agent, and gives an alternative method to current security mechanism without using Certification Authorities (CA) based public key infrastructure. With this scheme, we can simplify the key management and reduce the computation particularly for group-oriented web services.

1. Introduction

Mobile communication networks provide us flexibility for working at any time, nearly anywhere and in many forms. More and more flexible-working users heavily depend on it to connect to information sources and network services. However, current Internet and pervasive networks using wireless mobile devices have several shortcomings. First of all, the wireless Internet reliability depends on where the users are as well as the strength of the reception signal, it may be unreliable. Moreover, the wireless connections are usually of low bandwidth and high latency. This may lead to time consuming and cost user more money. Web services and mobile agents technologies can compensate each other and overcome these weaknesses. Web services are reusable web-applications that are means to share services and information between other applications [15]. The W3C defines a Web service as a software system designed to support interoperable machine-to-machine interaction over networks. It refers to those services that use SOAP-formatted XML envelopes and have their interfaces described by WSDL. Web services use XML and the HTTP network protocol. This makes the Web services nearly unlimitedly potential. Any program can be mapped to Web services. XML can accommodate any data type and structure. Additionally, Web services technologies bridge any operating system, hardware platform, or programming language and create an open era for distributed computing [11].

Mobile agents are software programs that move between heterogeneous networks taking their state with them to perform tasks on behalf of their owners and customers in an autonomous fashion. A mobile agent can reduce latency and improve the use of bandwidth because they execute the task locally on the remote host's platform. Moreover, agents can handle the intermittently connected networks. Hence, the applications of the combination of mobile agents and Web services have been widely investigated in recent years [18, 14, 7, 2, 5].

On the other hand, the mobile agent and Web services security is still of a primary concern for applications. Currently, Web services and mobile agent security is mostly based on Certification Authorities (CA) based public key infrastructure. Consequently, a trusted third party is required in advance. Users must have their key pairs and the service owner must manage all the users' public key and encrypted data using different keys. In some applications, this may be not necessary such as services just for particular groups.

In this paper, we present a new mobile agent and Web services security scheme. This security scheme employs a novel 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 to current security mechanism without using the Certification Authorities (CA) based public key infrastructure. It simplifies the

Second International Conference on
Internet and Web Applications and Services (ICIW'07)
0-7695-2844-9/07 $20.00 © 2007 IEEE

key management and reduces the computation cost particularly for group-oriented Web services.

The rest of this paper is organized as follows. In Section 2, we introduce related notations and works. Section 3 presents our new scheme for the integration of Web services and mobile agents. Finally, some concluding remarks are provided in Section 4.

2. Web Services, Mobile Agent and Security Issues

Web services specification provides an open standard for distributed service oriented architecture. Web services standards employ several XML-based technologies to transport and transform data into and out of programs and database [11] such as Web Services Description Language (WSDL), Simple Object Access Protocol (SOAP), Universal Description, Discover, and Integration (UDDI). These basic standards are used together for Web Services. Figure 1 shows Web services procedure. When a business Web service is produced, it is put in a UDDI for advertising. The service provider's WSDL file describes the methods and required parameters for the Web service. Messages are exchanged through the SOAP protocol, which exchanges information using GET/POST over HTTP.

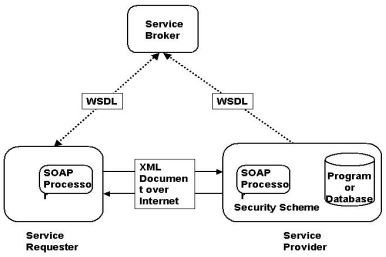

Figure 1. Web Service

A mobile agent is a composition of computer software and data, which is able to migrate from one host to another autonomously and continue its execution on the destination host [3]. A mobile agent has five attributes: state, implementation, interface, identifier, and principals (see Figure 2). A mobile agent migrates to the services host to enable fast local interaction with the service such as analysis and search method and then only brings back the necessary result. Furthermore, a mobile agent user has to connect to the service server only when launching agents and receiving agents, and can select the more stable and faster response hosts to interact with. Apart from that, mobile agent systems also support cloning and complement migration. If an agent accesses multiple services in local hosts, dispatching a clone with necessary data can reduce the migration cost [16, 17]. Thus mobile agent technology can reduce the bandwidth requirement and tolerate the network faults - able to operate without an active connection between clients and servers. Mobile agents can encapsulate protocols, execute asynchronously and autonomously, and adapt dynamically. They are naturally heterogeneous, robust and fault-tolerant [3].

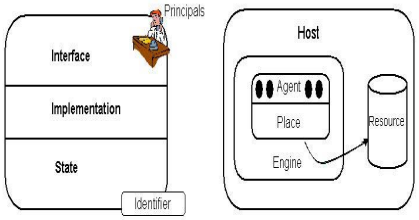

Figure 2. Mobile Agent

In addition, the Internet has been extended from workstations, desktops and notebooks to PDAs, mobile phones and other small mobile devices. Web services have been widely used in Internet and pervasive networks using wireless mobile devices such as notebooks and PDAs. On the other hand, the current Web server centric approach can only be used online or when a service provider offers a certain service. Moreover, it must be connected to appropriate server. The integration of mobile agents and Web service technologies can overcome these shortcomings. Hence, the applications combining mobile agents and Web services technology have drawn much attention in recent years [18, 14, 7, 2, 5].

Dominic Cooney et al. proposed a model for implementing Web services with mobile agents [2]. Fuyuki Ishikawa et al. implemented a mobile agent system for Web service integration [6] and proposed a general framework for "Mobile Web Services" in [7]. Riccardo Pascotto presented the ACTS AMASE project which described a mobile agent approach for users' access to network-based services [13]. Jan Peters introduced an integration architecture of mobile agents and Web services [14]. Wassam Zahreddine et al. presented an agent-based approach for composite mobile Web services over mobile devices [18].

Some combination schemes of agents and Web services are also presented by other studies [5, 4, 10, 9, 12].

However, the security of mobile agents and Web services is still a major concern. In practice, both Web services and

mobile agent security are based on the public key infrastructure (PKI). It consists of the certificates of different parties issued by Certification Authorities (CA), a repository for retrieving certificates, a method of revoking certificates, and a method of evaluating a chain of certificates from public keys that are known and trusted in advance to the target name [8].

On the other hand, such an existing security scheme has its drawbacks. Namely, all the users must have his/her public/private key pair based on PKI, and the service server must verify and manage all users' public keys. In addition, the service server has to search the user's public keys and use different keys for different users to encrypt messages whenever they send a message to a user. Furthermore, the username/password tokens are required in order to do the authentication. However, in the mobile agent system, the mobile agents could not take any password or private key with them for security reasons. This has been a big challenge for mobile agent security research community. In the following section, we propose a new mobile agent and Web services security scheme. In this security scheme, we do not need Certification Authority (CA) based public key infrastructure. We adopt the ID-based encryption scheme so the mobile agent is not required to carry a password for authentication. Furthermore, the server only needs one key to encrypt one service that is available to a group of users.

3. A New Security Scheme for Integration of Agent and Web Services

In this section, we first briefly introduce the integration system of mobile agent and Web services. Then we present the Boneh-Franklin ID-based public key scheme. Then we present new scheme system setup, authentication scheme and the secure Web service scheme.

3.1. System Description

Assume there is a Web services provider who provides a set of Web services and there are a number of users who have mobile agent enabled devices. Users can subscribe to or are assigned to any of the Web services and form several groups. Each corresponds to the users who subscribe to a specific service resource. Assume the Web service provider (WSP) can act as a Key Distribution Centre (KDC) and have the secure channels to distribute keys to the users. Let l denote the cardinality of the Web service resources denoted as $r_1, r_2, \cdots, r_l$. All the users who subscribe to or are assigned to the same Web service resource form a group (G) denoted as $G[1], G[2], \cdots, G[l]$. For example, suppose that WSP provides several related Web services resource i.e. (r_1, r_2 and r_3). The corresponding mobile agent enabled user groups are $G[1]$ for accessing r_1; $G[2]$ for accessing r_2, and $G[3]$ for accessing r_3. Users can subscribe to or be assigned one to all of the Web Services resources. Based on our protocol, each client may join, leave a group or switch from one group to another dynamically. The integration of mobile agent and Web services or Mobile Web service can have three types of composition: parallel, sequential, and combination.

3.2. Boneh-Franklin ID-based Public Key Scheme

We introduce Boneh-Franklin ID-based public key scheme [1] on which our new scheme is based. ID-based public key scheme includes a trusted Key Distributor Center (KDC) and users. The basic operations consist of set up, private key extraction, encryption and decryption. KDC selects a large prime $p = 2q + 1$ where q is also a prime and then runs parameter generator to generate an additive group $\mathbb{G}_1$ and a multiplicative group $\mathbb{G}_2$ (both have order p) and a bilinear pairing $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_2$. It chooses an arbitrary generator $P \in \mathbb{G}_1$ and defines two one-way hash functions: $H_1\{0,1\}^* \rightarrow \mathbb{G}_1$, $H_2 : \mathbb{G}_2 \rightarrow \{0,1\}$.

- **Set UP**: KDC chooses a random number $s \in Z_q^*$ and sets public key $P_{pub} = sP$, then the KDC publishes system parameters $params = \{\mathbb{G}_1, \mathbb{G}_2, q, P, P_{pub}, H_1, H_2\}$, and keeps key s as master key
- **Extraction**: a user submits his identity information ID to KDC. KDC computes the user's public key as $Q_{ID} = H_1(ID)$, and returns his private key $S_{ID} = sQ_{ID}$ using secure channel.
- **Encryption**: Let m denote the message to be encrypted. The KDC computes $U = rP$ where $r \in_R Z_Q^*$, and

$$V = m \oplus H_2(\hat{e}(P_{pub}, rQ_{ID}) \quad (1)$$

Then KDC sends the ciphertext (U,V) to users.

- **Decryption:** User can perform Decryption by computing

$$V \oplus H_2(\hat{e}(U, S_{ID})) = V \oplus H_2(\hat{e}(rP, sQ_{ID}))$$

$$= V \oplus H_2(\hat{e}(sP, rQ_{ID}))$$

$$= V \oplus H_2(\hat{e}(P_{puc}, rQ_{ID})) = m \quad (2)$$

3.3. New Scheme System Setup

Based on the ID-based encryption algorithm introduced in previous section, the Web services provider (WSP) select

system parameters p, q, and two groups $\mathbb{G}_1, \mathbb{G}_2$ 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 $H_1 : \{0,1\}^* \rightarrow \mathbb{G}_1$ and $H_2 : \mathbb{G}_1 \rightarrow \{0,1\}^*$.

Any user who wants to subscribe to or is assigned any Web service has to register with the WSP and become a member. We assume that there is a secure channel between each user and the WSP. The user applies to join the group and provides his/her ID . The WSP authorizes a privileged user by sending him/her a private key $S_{ID} = sQ_{ID}$, where $Q_{ID} = H_1(ID)$. After registration, users become members and they can subscribe to or are assigned to any Web services.

Assume members subscribe to or be assigned to some Web service resources $r_1, r_2, \cdots, r_l$ and then become a member of the Web services. Consequently, the Web service provider can manage a $n * l$ matrix S as follows.

$$\mathcal{S} = \begin{pmatrix} S_{11} & S_{12} & \cdots & S_{1k} & \cdots & S_{1l} \\ S_{21} & S_{22} & \cdots & S_{2k} & \cdots & S_{2l} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ S_{m1} & S_{m2} & \cdots & S_{mk} & \cdots & S_{ml} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ S_{n1} & S_{n2} & \cdots & S_{nk} & \cdots & S_{nl} \end{pmatrix}$$

Where $S_{mk} = 1$ if the user u_m is a member of Web service $G[k]$, i.e. u_m is in group $G(k)$. $S_{mk} = 0$ if the user m is not a member of Web service $G[k]$, i.e. u_m is not in group $G(k)$. n is the number of the users; l is the number of the services; k denotes the Web service $(1 \leq k \leq l)$; and m the user $(1 \leq m \leq n)$.

3.4. Mobile Agent and Web Services Authentication Scheme

Our new security scheme for the integration system of mobile agent and Web services consist of mobile agent user authentication and Web service data encryption algorithms. In this section, we present an authentication scheme. As we mentioned in previous section, each user has a unique identification ID_i. No matter which user launched the mobile agent to the Web service system, the WSP can verifies whether the mobile agent is really from the legitimate user or not.

3.4.1. Signature Scheme To sign a message $m \in \{0,1\}^*$, user U_i selects a random number $r \in \mathbb{Z}_q$, a generator $P \in \mathbb{G}_1$, and a public hash function $H_2 : \{0,1\}^* \rightarrow \mathbb{Z}_q$ and computes $R_i \leftarrow rQ_i$ and

$$S_i \leftarrow (H_2(m, R_i) + r)s_{ID_i}. \quad (3)$$

The signature is now a triple (R_i, S_i, m).

To verify the signature, the verification is conducted as follows:

$$\hat{e}(S_i, P) \stackrel{?}{=} \hat{e}(H_2(m, R_i)H_1(ID_i) + R_i, P_{pub}). \quad (4)$$

3.4.2. Authentication Scheme The scheme works as follows. The WSP sets up the system by following the algorithm presented in the preceding section and then distributes each user's secret key via secure channels separately. At this stage, each legitimate user has a public key P_{pub}, secret signing key s_{ID_i}, and a unique identification ID_i.

Before launching the mobile agent, the user sign the message M_r using his/her secret signing key s_{ID_i} and generate a signature triple (R_i, S_i, M_r), where $R_i \leftarrow rQ_i$, $S_i \leftarrow (H_2(M_r, R_i) + r)s_{ID_i}$, and M_r is the message.

When the WSP receives the signing register message, it can verify the signature using the public key P_{PUB} and the sender's ID_i.

$$\hat{e}(S_i, P) \stackrel{?}{=} \hat{e}(H_2(M_r, R_i)H_1(ID_i) + R_i, P_{PUB}). \quad (5)$$

3.5. Secure Web Services Scheme

In this section, we present the second part of our scheme, i.e. the Web service data encryption. After the Web service system verified that the mobile agent owner is legitimate, the mobile agent can move in the Web Server and acquire all the service data. In order to keep the confidentiality, all the data with the mobile agent is required to be encrypted. In this scheme, there is one encryption key for each service. We can have two methods to encrypt the data before it is transferred to the mobile agent. If the data set is not large, the data can be encrypted directly with the service encryption key, and only the legitimate user can decrypt it. If the data set is large, we can use the session key for this service to encrypt the data, then encrypt the session key with the service encryption key. After the mobile agent user receives the encrypted data, she/he can decrypt the session key first using her/his private key, and then decrypt the data with the session key. We present this scheme first and then discuss rekeying scheme for member changes and the rekeying for service changes.

3.5.1. Secure scheme This scheme consists of the following three steps: Encryption Setup, Encryption process and Decryption process.

Encryption Setup

In order to provide the l Web services to corresponding members, Web service provider needs to set up the following parameters.

- Select a random number $r \in \mathbb{Z}$.
- Compute $R = rP$
- Compute $x_i = \hat{e}(rQ_{ID_i}, P_{pub})$.
  where $\hat{e}$ is the Weil pairing mapping.

- Compute the following polynomial function
  $f^k(x) = \prod_{i'=1}^{n}(x - s_{i'k}x_{i'})/(x^{n-m}) mod p = \prod_{i=1}^{m}(x - x_i) \bmod p$
  where $m = \sum_{i=1}^{n} S_{ik}$, $(1 \le i \le n)(1 \le k \le l)$; 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.

We have the following equation:

$$\prod_{i=1}^{m}(x - x_i) = \sum_{i=0}^{m} a_{ik}x^i \bmod p$$

Therefore we obtain the followings:

$$a_{0k} = \prod_{j=1}^{m}(-x_j) \tag{6}$$

$$a_{1k} = \sum_{i=1}^{m}\prod_{j=1, j\neq i}^{m}(-x_j) \tag{7}$$

$$\cdots$$

$$a_{m-2,k} = \sum_{i=1}^{m}\sum_{j=i+1}^{m}(-x_i)(-x_j) \tag{8}$$

$$a_{m-1,k} = \sum_{j=1}^{m}(-x_j) \tag{9}$$

$$a_{mk} = 1 \tag{10}$$

$\{a_{ik}\}$ satisfy $\sum_{i=0}^{m} a_{ik}x_{jk} = 0 \bmod p$, $j = 1, \cdots, m$.

We can use the set $\{a_{ik}\}$ to construct the corresponding exponential functions.

$$\{a_{0k}P, a_{1k}P, a_{2k}P, \cdots, a_{nk}P\}$$

$$\equiv \{P_{0k}, P_{1k}, P_{2k}, \cdots, P_{mk}\} \tag{11}$$

Encryption

Let $M^k \in \{0,1\}^*$ be the Web service group $G[k]$ secret key or session key, then we can encrypt it as follows.

- Select a random number $R_k \in \mathbb{Z}$ and a random number $D_k \in \mathbb{G}_1$.
- Compute $(m+2)$ tuple

$$T_k \leftarrow (R, M^k \oplus H_2(D_k), D_k$$

$$+R_kP_{0k}, R_kP_{1k}, \cdots, R_kP_{mk})$$

$$= (R, C_k, C_{0k}, C_{1k}, \cdots, C_{mk}) \tag{12}$$

- Broadcast T_k to users $G[k]$

Decryption

When the members in group $G[k]$ receive the T_k, they can decrypt the correspondent session key by using his/her private key as follows.

$$\hat{e}(S_{ID_i}, R) = \hat{e}(sQ_{ID_i}, rP) = \hat{e}(rQ_{ID_i}, P_{pub}) = x_i \tag{13}$$

$$C_{0k} + \sum_{j=1}^{m} x_i^j C_{jk} = D_k + R_k(a_{0k}, +a_{1k}x_i + \cdots, +a_{mk}x_i^m)P$$

$$= D_k \tag{14}$$

$$C_k \oplus H_2(D_k) = M^k \tag{15}$$

If a member does not subscribe to data stream group $G[k]$, she/he will not be able to decrypt the session key. Hence s/he can not get the session key. This is because for any member not belonging to $G[k]$, $\sum_{i=0}^{m} a_{ik}x_j^i \neq 0 \bmod p$.

3.5.2. Re-keying for Member Changes and Service Changes .

Member Changes (Members Join, Leave and Switch Web Service Groups)

In order to maintain the forward secrecy and backward secrecy, when a member changes to another Web services group $G[k]$, the Web service provider needs to re-key the corresponding Web service group session key. This includes cases where new members join, old members leave, and an existing member switches from one or several Web service groups to another one or several Web service groups.

Here we consider the case where one member switches from one Web service group to another. The join and leave operations are similar to the group switch operation. Suppose that a member u_m unsubscribes from the web service group $G[k]$ and subscribes to Web service group $G[k^{'}](k \neq k')$, then the Web service provider needs to update the Web service group registration matrix S as shown below.

$$\mathcal{S} = \begin{pmatrix} S_{11} & S_{12} & \cdots & S_{1k} & \cdots & S_{1k'} & \cdots & S_{1l} \\ S_{21} & S_{22} & \cdots & S_{2k} & \cdots & S_{2k'} & \cdots & S_{2l} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ S_{m1} & S_{m2} & \cdots & S_{mk} & \cdots & S_{mk'} & \cdots & S_{ml} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ S_{n1} & S_{n2} & \cdots & S_{nk} & \cdots & S_{nk'} & \cdots & S_{nl} \end{pmatrix}$$

Here the S_{mk} is set to 0, and $s_{mk'}$ is set to 1.

Then the Web service provider needs to recompute $f^k(x)$ to revoke the member u_m from Web service group $G[k]$, and then recompute $f^{k'}(x)$ to add the member u_m to data group $G[k']$. Consequently, the Web service provider can get the new set $\{a_{ik}\}$ and $\{a_{ik'}\}$ and then encrypt those two Web service session keys by computing the two $(m+2)$ tuples T and T'.

$$T^k \leftarrow (R_k, M_k \oplus H_2(D_k), D_k + R_k P_{0k}, R_k P_{1k}, \cdots, R_k P_{mk})$$

$$T^{k'} \leftarrow (R_{k'}, M_{k'} \oplus H_2(D_{k'}), D_{k'} +$$

$$R_{k'} P_{0k'}, R_{k'} P_{1k'}, \cdots, R_{k'} P_{mk'}) \quad (16)$$

The corresponding Web service group members need to decrypt the session key by using their ID-based private keys.

When a new member joins the group, the Web service provider only needs a new row in the registration matrix, then recomputes the related T_k which is similar to the member's switch into the new Web service group.

When an existing old member leaves the group, the Web service provider needs to remove one line in the registration matrix or just set all values to 0, and then recompute the related T_k which is similar to the member's switch away from one Web service group.

Web Service Changes (Web Services Provider adds or removes Web service)

When a Web service provider wants to add a new Web service, s/he only needs to add a new column in the registration matrix, then recompute the related parameters.

When the Web service provider wants to remove a Web service, the process is much easier; s/he only removes the corresponding column in the registration matrix.

4. Conclusion

In this paper, we have proposed a new security scheme for the integration of mobile agents and Web services. In this security scheme, a mobile agent can be employed to autonomously search Web services on behalf of the customers. Additionally, Web services security is based on an ID-based public key management algorithm. With this mobile agent authentication scheme, the mobile agent owner can be verified before providing service. This scheme provides an alternative to the current literature without using Certification Authorities (CA) based-public key infrastructure. In this scheme only one key for each service and one key for each user permanently, it simplified the key management.

References

[1] D. Boneh and M. Franklin. Identity-based encryption from the weil pairing. *Advances in Cryptology-Crypto 2001, LNCS*, 2139:213–229, 2001. Springer-Verlag.

[2] D. Cooney and P. Roe. Mobile agents make for flexible web services. In *Proceedings of The Ninth Australian World Wide Web Conference*. Quensland, Australian, July, 2003.

[3] L. Danny B and M. Oshima. *Programming and developing Java Mobile Agents with Aglets*. Addison-Wesley, Massachusetts 01867, 2003.

[4] N. Gibbins, S. Harris, and N. Shadbolt. Agent-based semantic web services. In *WWW2003*, pages 710–717. ACM 1-58113-680-3/03/0005, May 2003.

[5] D. Greenwood and M. Calisti. Engineeringweb service - agent integration. In *IEEE International Conference on Systems, Man and Cybernetics (SMC 2004)*. The Hague, The Netherlands, October 2004.

[6] F. Ishikawa, N. Yoshioka, Y. Tahara, and S. Honiden. Mobile agent system for web services integration in pervasive networks. In *International Workshop on Ubiquitous Computing (IWUC 2004)*. New York, USA, 2004.

[7] F. Ishikawa, N. Yoshioka, Y. Tahara, and S. Honiden. Toward synthesis of web services and mobile agents. In *Proceedings of the AAMAS2004 Workshop on Web Services and Agent-based Engineering (WSABE)*. New York, USA, July, 2004.

[8] C. Kaufman, R. Perlman, and M. Speciner. *Network Security Private Communication in a Public World*. Prentice hall PTR, Upper Saddle River, NJ 07458, 2002.

[9] R. Liu, F. Chen, H. Yang, W. C. Chu, and Y.-B. Lai. Agent-based web services evolution for pervasive computing. In *Proceedings of the 11th Asia-Pacific Software Engineering Conference (APSEC04)*, 2004.

[10] Z. Maamar, F. Akhter, and M. Lahkim. An agent-based approach to specify a web service-oriented environment. In *Proceedings of the Twelfth IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE03)*, 2003.

[11] E. Newcomer. *Understanding Web Services XML, WSDL, SOAP, and UDDI*. Addison-Wesley, 201 W. 103rd Street Indianapolis, IN 46290, 2002.

[12] A. Padovitz, S. Krishnaswamy, and S. W. Loke. Towards efficient selection of web services. In *Proceedings of Workshop on Web Services and Agent-based Engineering AAMAS'2003*, 2003.

[13] R. Pascotto, B. Schiemann, and E. Kovacs. Giving mobile users access to net-based services a mobile agent approach. In *Flexsible Working*, pages 164–172. IOS Press The Netherlands, 2004.

[14] J. Peters. Integration of mobile agents and web services. In *The First European Young Researchers Workshop on Service Oriented Computing (YR-SOC 2005)*, April 2005.

[15] S. Tsur, S. Abiteboul, R. Agrawal, U. Dayal, J. Klein, and G. Weikum. Are web services the next revolution in e-commerce? VLDB2001, pp. 614-617.

[16] Y. Wang. Dispatching multiple mobile agents in parallel for visiting e-shops. In *3rd International Conference on Mobile Data Management (MDM2002)*, pages 53–60. IEEE Computer Society Press, Jan. 8-11 2002, Singapore.

[17] Y. Wang and J. Ren. Building internet marketplaces on the basis of mobile agents for parallel processing. In *3rd International Conference on Mobile Data Management (MDM2002)*, pages 61–68. IEEE Computer Society Press, Jan. 8-11 2002, Singapore.

[18] W. Zahreddine and Q. H. Mahmoud. An agent-based approach to composite mobile web services. In *Proceeding of the 19th International Conference on Advanced Information Networking and Applications (AINA'05)*. IEEE, 2005.