PKI and UDDI based trust centre: an attempt to improve web service security

Wiem Rekik, Maher Khemakhem, Abdelfettah Belghith, Jacques Fayolle

HAL Id: ujm-00400071

https://hal-ujm.archives-ouvertes.fr/ujm-00400071

Submitted on 29 Jun 2009

PKI and UDDI based trust centre: an attempt to improve web service security

Wiem REKIK1,2
1HANA Group research
University of Manouba,
ENSI, Tunisia
2DIOM Lab
University of
Jean Monnet,
Saint-Etienne, France
wiem.rekik@gmail.com

Maher Khemakhem3
3Miracl Lab
University of Sfax,
FSEGS, Tunisia
PO. Box, 1088,
3023, Sfax, Tunisia
maher.khemakhem@ensi.rnu.tn

Abdelfettah Belghith1
abdelfattah.belghith@ensi.rnu.tn

Jaques Fayolle2
jacques.fayolle@univ-st-etienne.fr

Abstract

Nowadays Internet becomes the most used tool for the ever increasing amount of various transactions between institutions, organizations and more generally between clients and providers. Conducted studies and experiments showed that it is more convenient to provide and achieve these transactions as web services (WS) to guarantee their flexibility and their reuse. So far these services and the corresponding providers' URLs are advertised on specific UD-DIs (Universal Description, Discovery and Integration). As such, after finding the requested service any given client contacts the right provider to negotiate the service access procedure. These first contacts between clients and providers are usually and commonly not protected (Encrypted) yielding enough room for Hackers to intrude into these unprotected messages.

In this paper, we propose a securing approach based on both the PKI infrastructure and some proposed improvements of the UDDI functioning in an attempt to provide adequate security for web services.

1. Introduction

A web service (WS) can be any given transaction or a function which achieves a specific task and can be accessed commonly through a remote procedure call (RPC) via the internet. Possibilities of composing dynamically several WS to achieve some complex tasks are one of the attractive sides of such WS [6], [8]. Researchers are attracted by such properties and started working on this field to satisfy several needs especially that are related to the remote cooperation and easy communication between geographically dispersed organizations and institutions [7]. The idea consists to implement for example any given daily transaction between any given organizations (a provider, clients) in the form of a WS which can be accessed by any authorized client just through a simple RPC. So far, WS are advertised over specific Directories named UDDI [15] where any client can first looking for the appropriate WS and then get the corresponding provider URL. After that, the client has to contact the adequate provider to get the access grant (which is a kind of certificate) to the requested WS. With this access grant, the client becomes able to access to this WS. Figure 1. illustrates the advertising mechanism of WS. Unfortunately, WS security still constitutes the big challenge; in fact, despite the multitude of security proposals done mainly by specialized consortium, organizations and researchers such as W3C, OASIS [7], [1], [16], [10], [14], [9], this problem seems to be not yet well solved. The non possession of public key infrastructure (PKI) especially by clients (customers) can be considered amongst the main causes behind this security problem. As illustrated in figure 1. if the client doesn't own a PKI, then any hacker can interfere in the exchanged messages between this client and the contacted provider and do what he wants.

Consequently, we, propose in this paper a detailed idea based on both the PKI and the improvement of the UDDI functioning which attempt to provide security for web services. The remainder of this paper is organized as follows: In section 2, we overview the existing solution for secure
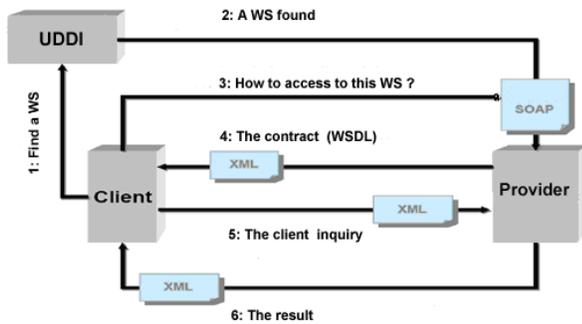
Figure 1. Web service advertising mechanism

Figure 2. WS Security framework

WS then we discuss the limit of the existing architectural scheme. In section 3, we present our contribution to provide secure and safe WS. Finally in section 4, we summarize the presented proposal and outline its perspective.

2. Background: Security of Web Services

Security of web services (WS) can be viewed from different sides because of the multitude of corresponding utilization. If we consider the case of utilization of WS for sharing information and services across organizations, then we can say that the corresponding security proposals and solutions are not bad but still require more improvements. But if we consider the case of public WS which can be provided to any given client by specialized providers then, we can say that the existing proposals and solutions lack a lot of security. This is mainly due to the non possession of the adequate security tools by most of the clients as we will explain next. Thus, our main concern in this paper focuses on this last case.

Specialized consortium and organization such as W3C, OASIS, ... have proposed several standards and solutions which attempt to provide security for WS. The framework illustrated by figure 2. is the most commonly used one to provide security for WS. This framework shows that the security problem is divided into two levels; the transport level and the application level.

2.1. Transport Layer Security (SSL/TLS):

Commonly, SSL/TLS [5] offers a secure channel on the transport channel. The server authentication is based on specific certificates and the client is authenticated via password or certificates.
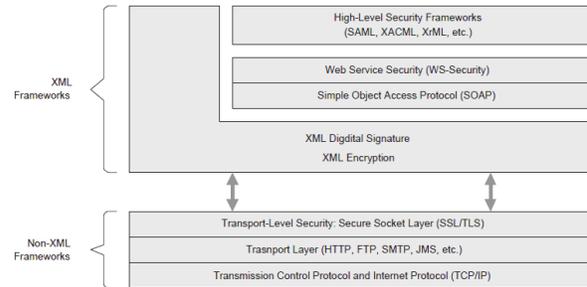
2.2. Web Services Security (WS-Security):

If we assume malicious hosts then the transport level security is possibly not enough. The transport level is controlled by the host, so the web service itself cannot verify the user credentials. The industry standard WS-Security [11] offers application level security as an extension to SOAP [2]. It defines how to integrate various XML Security concepts as XML Signature [4], XML Encryption [3] or the Security Assertion Meta Language (SAML) [12] into SOAP.

2.3. Security Assertion Meta Language (SAML):

SAML defines a XML-based framework for creating and exchanging authentication and authorization information. The standard purpose of using SAML is to realize Web Single-Sign-On. The user authenticates at the first site, retrieves an authentication and authorization token and subsequently uses this token to access further services without the need of re-authentication.

2.4. XML Access Control Specifications XACML:

XACML [13] is an extension to SAML that focuses on access control rights. XACML defines how to express access policies. Furthermore it specifies a request/response protocol between a policy decision and a policy enforcement point. XACML is considered the better way to implement role based access control (RBAC) [7] which restricts the WS accessibility according to predefined security policies and rules.

Unfortunately, despite the consistency and the robustness of these security tools, WS still require more protection especially if they are intended for the public as we will explain in the next subsection.

2.5. Limit of the existing architectural schema

So far, WS and the corresponding URL (of providers) are advertised over specific UDDI where users (clients) can look for then find the required services for their different needs as illustrated in figure 1. The first security problem that maybe encountered with this scheme concerns mainly the first contact between any given client which belongs to the great public and the provider. This contact can't be protected (encrypted) especially if this clients don't know the PKI of the provider and in addition he don't own a personal PKI. Consequently, any hacker can easily interfere in exchanged messages between this client and provider to do what he wants. To deal with this problem, we propose in the next section an idea based on two components that attempts to solve this limitation.

3. PKI and UDDI based trust center for secure WS

Our proposal requires first the possession of a personal PKI by every involved actor (a client or a provider) to encrypt all exchanged messages between them. Second, we suggest that the UDDI should achieve in addition to its actual missions the following roles:

- a prior registration of any provider or client of the advertised WS;

- the publication in an encrypted manner of the PKI of every party or actor (a client or a provider);

- the authentication of every party before any given access to the advertised WS.

In this manner, UDDI will play indeed the role of a trust centre and all exchanged messages between any given provider and client will be well protected.

3.1. The mechanism of the proposed solution

As mentioned earlier , every provider or client of the advertised WS must be previously subscribed on the UDDI server to guarantee the security of any given transaction between them. We suppose also that the UDDI server owns a personal PKI which allows it to communicate in a secured manner with every involved actor. Thus, if a given new provider would like to advertise any WS, he must achieve the following steps:

- register his complete identity which will serve to the authentication process over the UDDI centre;

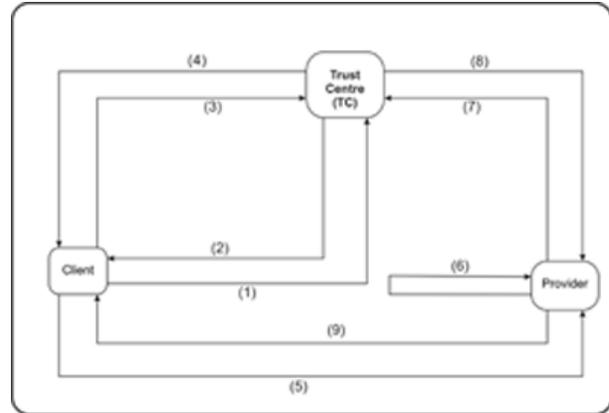- advertise both his PKI and the WS which can provide over the UDDI;

Figure 3. The mechanism of the proposed solution

in the same manner if a given new client would like to access (consume) any advertised WS, he must achieve the following steps:

- register his complete identity which will serve to the authentication process over the UDDI centre;

- advertise his PKI over the UDDI;

of course, to reduce the involvement of the UDDI centre during the authentication process and consequently the response time of our proposal, we suggest that only the first transaction or contact between any given provider and a new client which requires this involvement. It means that once a new client (customer) becomes known by any given provider, then he will be added automatically to a list of known customers with the corresponding PKI. So we suggest that every provider must own a list of customers where to store the required information to the authentication process of each of which and the corresponding PKI to encrypt every exchanged message with them. In this manner, hackers will not be able to interfere in exchanged messages.

Figure 3. illustrates the mechanism of the proposed solution; the corresponding steps are the following:

1. Looking for the required WS;

2. Request of registration and the personal public Key in an encrypted message using the public key of the Trust centre;

3. Registration + publication of his personal public key in an encrypted message using the public key of the Trust centre;

4. Getting the provider Public key and the adequate information for the authentication process in an encrypted message using the public key of the client;

5. Request of the required WS in an encrypted message by using the provider public key:

6. Authentication of the client from within the list of already known customers;

7. If the client is a new one then the provider authenticates him from the Trust centre in an encrypted message using the public key of the Trust centre;

8. The trust centre authenticates the client and turns back to the provider the adequate information of the client in an encrypted message using the public key of the provider;

9. After the authentication process, the provider updates his customers list and turns back to the client the requested WS in an encrypted message using the public key of this client.

Consequently, the proposed solution improves substantially the security of WS especially if these services are intended for the big public.

4. Conclusion

We proposed in this paper a new solution that attempt to improve the security of web services especially those intended for the big public. Our proposal is based on two components; the first one is the personal PKI which is required by every client or provider of WS. And the second one is some improvements of the UDDI functioning which should play the role of a trust centre in an attempt to provide adequate security for web services. What makes attractive our proposal is its easy adaptation to improve the security of any kind of distributed application intended especially for the big public such as e-health, e-learning, e-government ...

Several investigations are understudy, especially the implementation and the real evaluation of our proposal on a specific application.

References

[1] D. Booth, H. Haas, F. McCabe, and E. Newcome. Web services architecture. In *Available at, http://www.w3.org/TR/2003/WD-ws-arch-20030808/*, August 2003.

[2] D. Box, D. Ehnebuske, G. Kakivaya, A. Layman, N. Mendelsohn, H. Nielsen, S. Thatte, and D. Winer. Simple object access protocol (soap) 1.1. In *available at http://www.w3.org/TR/2000/NOTE-SOAP-20000508.*, W3C Note, 2000.

[3] D. Eastlake, J. Reagle, T. Imamura, B. Dillaway, and E. Simon. Xml encryption syntax and processing. In *available at http://www.w3.org/TR/xmlenc-core/*, W3C Recommendation, 2001.

[4] D. Eastlake, J. Reagle, D. Solo, M. Bartel, J. Boyer, B. Fox, B. Lamacchia, and E. Simon. Xml-signature syntax and processing. In *available at http://www.w3.org/TR/xmldsig-core/*, W3C Recommendation, 2002.

[5] A. Freier, P. Karlton, and P. Kocher. The ssl protocol, version 3.0. In *Internet draft, Netscape*, November 1996.

[6] G.Poulin, J.Soyer, and M.Trioullier. *Sécurité des architectures WEB*. WEB-Edition, 2004.

[7] D. A. Haidar, F. C. N. Cuppens-Boulahia, and H. Debar. An extended rbac profile of xacml. In *Proceedings of ACM SWS'06*, Alexandria, Virginia, USA, November 3 2006.

[8] K. Heather. Web services conceptual architecture. In *wsca 1.0*, May 2001.

[9] T. Imamura, B. Dillaway, and E. Simon. Xml cryption syntax and processing-w3c recommendation. In *Available at, http://www.w3.org/TR/xmlenc-core*, December 2002.

[10] M. Khemakhem, H. B. Abdallah, and A. Belghith. Towards an agent-based framework for the design of secure web services. In *Proceedings of ACM SWS'08*, Alexandria, Virginia, USA, October 2008.

[11] A. Nadalin, C. Kaler, r. P. Hallam-Bake, and R. Monzillo. Web services security: Soap message security 1.0. In *Proceedings of ACM SWS'04, OASIS*, Alexandria, Virginia, USA, 2004.

[12] OASIS. Security assertions markup language (saml), version 2.0. working draft. In *Organization for the Advancement of Structured Information Standards*, OASIS, 2004.

[13] OASIS. Xml access control markup language (xacml), version 2.0. committee draft. In *Organization for the Advancement of Structured Information Standards*, OASIS, 2004.

[14] J. Pamula and al. A framework for etablishing, assessing and managing trust in inter-organizational relationships. In *Proceedings of ACM SWS'06*, Alexandria, Virginia, USA, November 3 2006.

[15] P.Fremantle, D.Koeing, and C.Zenter. *Building Web Services with java: making Sense of XML, SOAP, WSDL and UDDI, 2nd Edition*. (Developer's Library), Sams,, 2004.

[16] M. A. Rahman, A. Schaad, and M. Rits. Towards secure soap message exchange in a soa. In *Proceedings of ACM SWS'06*, Alexandria, Virginia, USA, November 3 2006.