SECURE SERVICE PUBLISHING WITH UNTRUSTED REGISTRIES

Securing Service Discovery

Slim Trabelsi and Yves Roudier
Institut Eurecom, 2229 route des Cretes, BP 193, 06904 Sophia-Antipolis, France

Keywords: Security, Service Discovery, Attribute Based Encryption, Trust, Privacy, Untrusted Registry.

Abstract: Service Discovery becomes an essential phase during the service deployment in Ubiquitous system. Applications and services tend to be more dynamic and flexible. Users need to adapt in order to locate these pervasive applications. Service mobility introduces new security challenges relating to trust and privacy. Existing solutions to secure the service discovery cannot provide any solution without relying on a trusted third party. In this paper we purport to use Attribute Based Encryption so as to protect the publishing and binding messages with untrusted registries.

1 INTRODUCTION

Service Oriented Architectures (SOA) introduce a loosely coupled interaction model which serves as the basis to define protocols and procedures that enable an efficient interconnection between different applicative systems or software components. SOA consists mainly of services, which are applicative elements providing elaborate functions (database access, data processing, business logic…), useful for clients requesting such services.

Orchestration is becoming an essential feature to develop softwares for increasingly pervasive systems, in particular with the fast development of ubiquitous computing. The orchestration technique becomes mandatory to locate previously unknown services. The first orchestration technique generally applied is the service discovery that allows dynamic detection of the available services in the network. The service mobility introduces new security challenges regarding trust and privacy. Private data exchanged during the discovery process can be re-used for illegal purposes. Failures in the discovery protocol can facilitate denial of service attacks. Most of the existing solutions to secure the discovery rely on trusted third party such as security modules, secure proxies or trusted registries. These modules are in charge of the encryption and of establishing trust among users. Such additional modules are not deployable on a large scale (without agreements) and are not realistic for pervasive computing scenarios where mobile clients and services do not have any a-priory knowledge of the ambient environment. In this paper we propose a new approach based on a particular cryptographic scheme that allows a secure service discovery using untrusted registries. Our solution offers the possibility for servers (clients) to publish (bind) in a secure manner their services with untrusted registry.

This paper is organized as follows. In section 2, we define the service discovery; then we provide a threat model. In section 3, we show in full detail how we can secure service discovery using Attribute Based Encryption scheme. Finally, we compare our approach with related work.

2 SERVICE DISCOVERY AND SECURITY CHALLENGES

2.1 Service Discovery Definition

With the emergence of new dynamic networks and services where devices are ubiquitous, the discovery techniques are being adapted in order to find mobile services rather than devices. This adaptation in particular shows how to combine services, as a logical layer in such systems, with the specification of environmental constraints.

Trabelsi S. and Roudier Y. (2007).
SECURE SERVICE PUBLISHING WITH UNTRUSTED REGISTRIES - Securing Service Discovery.
In *Proceedings of the Second International Conference on Security and Cryptography*, pages 175-179
DOI: 10.5220/0002127301750179

Centralized discovery approaches rely on a registry which plays the role of yellow pages, which clients can refer to. A service advertises its capabilities to the registry, which then stores them for a certain amount of time. A client solicits the registry to find a service by sending a request containing service preferences; the registry tries to match the requested service with the most suitable provider found from the stored advertisements. In that approach, registries have to be considered by the services and the clients as a third trusted party. An alternative approach to the service discovery mechanisms exists: it relies on peer to peer advertisements between services and clients. In this paper we only focus on the registry based model.

2.2 Registries: Threats and Attacks

Usually, during the service discovery execution a lot of private information (identities, location, addresses, URI, owner, domain …) are exchanged, and are permanently exposed to an illegal use (profiling, fishing …). If a server publishes its services in a Registry and if a client binds a Registry for a service, a trust relationship must initially be established between users (clients and servers) and Registries. This trust relationship will guarantee that private data related to service properties are kept in a protected database accessed only by the Registry. Clients must also be sure that the Registry will provide the right service, not fake ones.

In some cases, users cannot trust a Registry (belonging to an unknown domain). They have the possibility to protect the communications by encrypting exchanged messages using PKI public keys, but valid PKI Certificated does not represent a trust proof. In this case, users are left with no other alternative but to believe the registry and hope that their personal data will not be used illegally.

What are the threats related to the service discovery information display?

- Client's intention: A malicious registry has the possibility to establish an "intentional" profile about each user, and re-use it for commercial purposes (without any authorisation). It may thus act like a spyware in an infected computer.
- Illegal competition: service providers may want to prevent potential commercial competitors or malware from gathering information about their offers too easily.
- Wrong matching: A malicious Registry has the possibility to perform wrong matching with the client's request in order to re-direct it to malicious services (or other services that do not have anything to do with the client's wish).
- Fake registrations (fishing): Fake services have the possibility to register and trap putative clients. A service could register as a banking service in order to obtain from users their confidential banking numbers.

Trusting a registry becomes fundamental for a correct execution of a service discovery mechanism. Unfortunately, in some cases, it is very hard for a user to establish a trust relationship with an unknown registry. This difficulty becomes important particularly for mobile and pervasive applications in which users are accessing services from foreign domains. Some essential security requirements are needed, concerning these Registry-related vulnerabilities:

- Confidentiality: Exchanged data must be protected against any external access (from other clients, other services, un-trusted registries …)
- Privacy: Private data related to clients and services must be disclosed only to authorised entities.
- Authentication: Every entity must be able to authenticate its counterpart before disclosing personal data.
- Access Control: Services must be able to restrict the possibility of being discovered only to a restricted class of clients. Clients must also be able to limit the scope of their discovery request to trusted (certified) services.
- Integrity: All exchanged messages must be checked to verify the authenticity of the content and detect illegal modifications.

In this paper we explain how this compromise can be effective in order to perform a secure service discovery with non-trusted Registries.

3 SECURING SERVICE DISCOVERY

In (Trabelsi, 2006) authors proposed a solution based on an extension of the Identity Based Encryption (IBE) (Boneh, 2001) called Attribute Based Encryption (ABE) (Sahai, 2005) in order to secure the Web Service Discovery in a distributed configuration. In their solution, the authors propose to extend the WS-Discovery protocol by adding ABE functionalities. In this section we describe this security mechanism, and we explain how it could be extended in order to comply with our security requirements.

3.1 Attribute based Encryption

Using ABE, a user has the possibility to encrypt a message or a document using the identity (composed by a set of attributes) of its intended recipient as a public encryption key, without any need for the public key certificate of that recipient. Contrary to PKI, the ABE public key has a semantic meaning (a name, a mail address, an identifier …), and it does not require to be verified with a Certification Authority.

If we combine these attributes, we can use them as public key encryption, and we obtain the Attribute Based Mechanism (ABE) (Figure1).

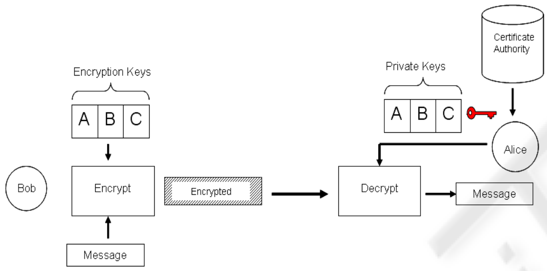

Figure 1: Attribute Based Encryption.

3.2 Extending Service Discovery with ABE

WS-Discovery is essentially used for a service discovery in a decentralized fashion and initially dedicated to LANs. This protocol relies on a Multicast diffusion to locate services connected to a restricted sub-network. Proxies are used to extend the scope of the discovery to other networks. The principle of this mechanism is quite simple; Clients multicast service query messages (Probe Message) and concerned services that are listening to the same multicast address will respond directly to the requester by sending a response message (ProbeMatch Message) containing the necessary information to access the requested service. Services have also the possibility to announce their existence by multicasting messages (Hello Message) containing their service capabilities. Clients that are listening to the same multicast address have the possibility to cache these information in order to join the desired services. The default configuration, found in the WS-Discovery specification, recommends the use of two matching attributes that are: Type (an identifier of the service endpoint), and Scope (An extensibility point that may be used to organize the services into logical groups).

(Trabelsi, 2006) in order to protect and restrict the service binding to only certified servers, a client has the possibility to encrypt his "Probe" message using Type and Scope Attributes as Public encryption Key like this $Encrypt[ProbeMessage]_{\{Type,Scope\}}$. Only mandatory key holders (with the correct values of the attributes Type and Scope) will be able to decrypt the query message. The same concept is used to protect the service response message sent to clients. In fact, the service has the possibility to restrict its responses to a restricted group of users by encrypting the "ProbeMarch" message, using particular attributes related to the restricted group of users. For example, in a university an administrative service could be restricted only to professors. In this case, the message sent will correspond to this $Encrypt[ProbeMatch]_{\{Bob,Professor\}}$.

In WS-Discovery we also have the possibility to rely on a centralized registry called "Proxy". The matching between a client's request and service profile is performed by this Proxy. This configuration extends the scope of the discovery to other networks and domains without the necessity to share the same multicast address.

With no modification involved, the previous solution will prevent a Proxy to perform a correct matchmaking. If this Proxy does not keep the mandatory private keys related to the attributes used to encrypt the messages, it will be impossible to process these messages. For this reason, we decided to extend this solution by replacing the encryption of the messages by a partial encryption of the sensitive data contained. Mandatory information (attributes) needed by the proxy to perform a correct matchmaking are contained in the client's query messages and those contained in the service's publish messages. In WS-Discovery of such information is limited to *Type* and *Scope*, but it could be easily extended by other attributes. The rest of the information contained in the messages is not really useful for the Proxy; it could thus be kept hidden. This is why we propose to partially encrypt the publish (Hello) and bind (Probe) messages in order to keep the matching attributes clear for the Proxy.

3.3 New Message Format

In WS-Discovery a “Hello” message is composed of two parts: the header (containing session information related to the protocol) and the body (containing information about the service). Only some attributes of the body are useful for the Proxy. In order to protect its private information and restrict the discovery of its profile to some allowed user, the service provider can encrypt the entire message except for the type and scope attributes (Figure 2).

```
<s:Envelope>
  <s:Header>
   Encrypt[Header]{Professor}
  </s:Header>
  <s:Body>
   <d:Hello>
    <a:EndpointReference>
 Encrypt[EndpointReference]{Professor}
    </a:EndpointReference>
    <d:Types>
     Printer
    </d:Types>
    <d:Scopes>
     University
    </d:Scopes>
    <d:XAddrs>
     Encrypt[XAddrs]{Professor}
    </d:XAddrs>
  </d:Hello>
  </s:Body>
</s:Envelope>
```

Figure 2: Encrypted Hello Message.

The “Probe” message, sent by the client to the Proxy in order to query for a service, also contains a header providing session information and client’s endpoint reference (for the reply message), and a body describing the attributes corresponding to the requested service. As seen previously, the client has the possibility to protect the “Probe” message against unauthorised access by encrypting the sensitive part of the message and keeping the matching attributes in clear (Figure 3). Only services with the correct keys (attributes) will be able to give a response.

```
<s:Envelope>
<s:Header ... >
Encrypt[Header]{Printer,University}
</s:Header >
<s:Body>
  <d:Probe>
    <d:Types>
      Printer
    </d:Types>
    <d:Scopes >
     University
    </d:Scopes >
   </d:Probe>
</s:Body>
</s:Envelope>
```

Figure 3: Encrypted Probe Message.

3.4 Towards a Hybrid Solution

After the modification of this part of the WS-Discovery protocol, an efficient large scale service discovery can be performed in a secure manner, without the obligation to establish a trust relationship with the Proxy. In a centralized configuration, if the service does not exist locally, the client will stop sending its binding message or will retry the binding process later. On the contrary, with a proxy-based configuration, if the service is not found locally, the local Proxy can forward the query to other Proxies belonging to other domains and networks. Proxies do not necessarily know one another, but they can communicate via a multicast address. With the proxy based solution, we can avoid bottlenecks on the service side. In fact, with the decentralized solution, when a client multicasts an encrypted probe message, all the servers that are listening to the multicast channel will try to decrypt the message at the same time. During the decryption period, if another client sends another Probe message, it could be dropped or cached until the end of the previous message decryption. This phenomenon generates a bottleneck on the service side that could be avoided with the Proxy-based solution. With this proxy-based solution, the secure service discovery is extended to other LANs and solves the bottleneck problem created by the decentralised solution. This performance improvement is conditioned by a privacy relaxation. With the ABE scheme the client has the possibility to verify if the returned services correspond to the requested services because the binding message is encrypted according to the attributes of the query, and only servers holding private keys corresponding to these attributes are able to reply to the client.

4 RELATED WORK

(Carminati, 2005) raised the privacy issues in Web Services with trusted UDDI-based Discovery Agencies (equivalent to a foreign agency providing a registry service). After describing the privacy requirements related to the discovery mechanisms, they provide five UDDI-based registries scenarios (Internal enterprise application, Portal, Partner catalog, and e-Marketplace). For each scenario, they proposed the application of three privacy enforcement strategies: Access-Control based solution using a third trusted party (a trusted UDDI registry) that is in charge of the access-control policy enforcement to the registry. Cryptographic-Based Solution, also relying on a trusted third party called encryption module in charge of encrypting sensitive data (XML encryption), according to a specific privacy policy provided by clients and services. Hash-Based solution where service providers publish hashed services in an untrusted registry. Compared to our ABE solution, Carminati's solution must rely on a trusted third party or a trusted registry to secure the service discovery; otherwise, they use insecure hash mechanisms.

(Czerwinski, 1999) proposed an architecture relying on an additional component, called Service Discovery Service (SDS), which plays the role of a secure information repository (secure registry). This SDS helps clients and servers set up a trust relationship and secure channels among them: using a PKI, it provides authentication, access control, encryption, signature verification, and privacy protection. The main idea is to create a kind of VPN in which clients, servers and registries could communicate in a secure manner. In order to encrypt the exchanged messages, the SDS uses a hybrid public/symmetric key system. Trust establishment between the SDS and other entities is limited to a simple verification of the SDS public certificate validity. This kind of infrastructure is based only on certificate verification; in this case, every user with a valid certificate is able to discover all existing services without any restriction.

5 CONCLUSION

In this paper we proposed an encryption-based solution to secure the service discovery mechanism with untrusted registries. Our solution is based on the Attribute Based Encryption scheme that enables a selective publication and binding without exposing private and sensitive data to an illegal usage by the untrusted registry (or a possible attacker). Using attributes for the encryption provides an access control system coupled to a confidentiality protection. Each user (clients and services) has the possibility to define security preferences by choosing the appropriate attributes required to decrypt the discovery message. Untrusted registries can only access these attributes in order to perform a matching between service's descriptions and client's query; the rest of the data remains hidden.

REFERENCES

Boneh, D., Franklin, M., 2001, "Identity-based encryption from the weil pairing", 21st Annual International Cryptology Conference on Advances in Cryptology.

Carminati, B., Ferrari, E., Hung, P.C.K, 2005, "Exploring Privacy Issues in Web Services Discovery Agencies", IEEE Security and Privacy .Volume 3, Issue 5

Czerwinski, S.E. et al, 1999, "An Architecture for a Secure Service Discovery Service" , In Proceedings of MobiCom '99.

SUN Microsystems, 2005, "Jini Specifications", http://java.sun.com/products/jini/

Martin, D et al, 2004, "Bringing Semantics to Web Services: The OWL-S Approach", Proceedings of the 1st SWSWPC.

Trabelsi, S., Pazzaglia, J.C, Roudier, Y., 2006, "Secure Web service discovery: overcoming challenges of ubiquitous computing", 4th IEEE European Conference on Web Services, ECOWS 2006.

Sahai, A., Waters, B., 2005, "Fuzzy Identity-Based Encryption", Advances in Cryptology-Eurocrypt'05.

WS-Discovery Specifications 2004, http://msdn.microsoft.com/ws/2005/04/ws-discovery/