INTERNATIONAL JOURNAL OF MERGING TECHNOLOGY AND ADVANCED RESEARCH IN COMPUTING

ISSN: 2320-1363

Dynamic Privacy model for protecting Web services Composition

Lalitha devi Katikala*1, Veera Swamy Anaparthi*2

1*M.Tech Scholar, Dept of CSE, Universal College of Engineering & Technology, Perecherla, Dist: Guntur, AP, India.

2*Assistant Professor, Dept of CSE, Universal College of Engineering & Technology, Perecherla, Dist: Guntur, AP, India

ABSTRACT:

The composition of DaaS (Data-as-a-Service) services is a powerful solution for building value-added applications on top of existing ones. However, privacy concerns are still among the key challenges that keep hampering DaaS composition. Indeed, services may follow different, conflicting privacy specification with respect to the data they use and provide. The chance of revealing sensitive information is one of the key challenging issues in Daas Composition. we are proposing a novel dynamic privacy model in order to extend DaaS descriptions with privacy capabilities [1] [2] . The privacy model allows a service to define a privacy policy and a set of privacy requirements. We also propose a privacy-preserving DaaS composition approach allowing verifying the compatibility between privacy requirements and policies in DaaS composition [2] [4].

KEYWORDS: Service composition, service oriented architecture, dynamic privacy model, DaaS services, privacy.

I.INTRODUCTION

A web service is a software system identified by a URL, whose public interfaces and bindings are defined and described using XML. Its definition can be discovered by other software systems. These systems may then interact with the web service in a manner prescribed by its definition, using XML-based messages conveyed by internet protocols. This definition has been published by the World Wide Web consortium W3C, in the Web Services Architecture document

ISSN: 2320-1363

(Booth et al., 2004). The web service model consists of three entities, the service provider, the service registry and the service consumer [4] [2]. Other models, such as a peer-to-peer structure, exist as will be discussed later in this paper. Figure 1 shows a graphical representation of the traditional web service model:

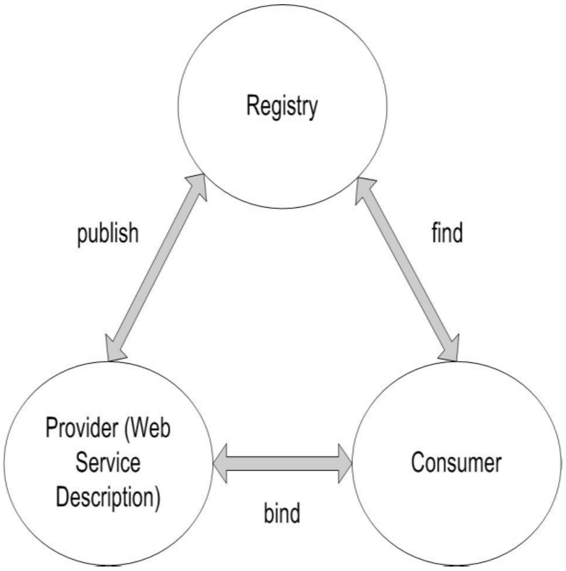

Figure 1.1 Web service model

The service provider creates or simply offers the web service. The service provider needs to describe the web service in a standard format, which in turn is XML and publish it in a central Service Registry. The service registry contains additional information about the service provider, such as address and contact of the providing company, and technical details about the service. The Service Consumer retrieves the information from the registry and uses the service description obtained to bind to and invoke the web service. The appropriate methods are depicted in Figure 1 by the keywords 'publish', 'bind' and 'find'. In order to achieve communication among applications running on different platforms and written in different programming languages, standards are needed for each of these operations [4] [5] [6] .

Web services architecture is loosely coupled, service oriented. The Web Service Description Language WSDL uses the XML format to describe the methods provided by a web service, including input and output parameters, data types and the transport protocol, which is typically HTTP, to be used. The Universal Description Discovery and Integration standard UDDI suggests means to publish details about a service provider, the services that are stored and the opportunity for service consumers to find service providers and web service details.

Besides UDDI, other standards have been developed as well. Dustdar and Treiber (2004) deals with web service registries in greater detail.The Simple Object Access Protocol SOAP is used for XML formatted information exchange among the entities involved in the web service model [5] [6].

Data as a Service is a new type of service oriented service sits between services-based applications (i.e., SOA-based business process) and an enterprise's heterogeneous data sources. The application developers can directly interact with various data sources that gives access to the business objects while individual services may provide interesting information/functionality alone, in most cases, users' queries require the combination of several Web services through service composition. Privacy is the right of an entity to determine when, how, and to what extent it will release private information [6] [4] [7] [8].

II.RELATED WORK

A typical example of modeling privacy is the Platform for Privacy Preferences (P3P) [2]. However, the major focus of P3P is to enable only Web sites to convey their privacy policies. In [3] privacy only takes into account a limited set of data fields and rights. Data providers specify how to use the service (mandatory and optional data for querying the service), while individuals specify the type of access for each part of their personal data contained in the service: free, limited, or not given using a DAML-S ontology. In [4], Ran propose a discovery model that takes into account functional and QoS-related requirements, and in which QoS claims of services are checked with external components that act as certifiers. The authors refer to the privacy concern with the term confidentiality, and some questions are raised about how the service makes sure that the data are accessed and modified only by authorized personals. Some policy languages, such as XACML [5], ExPDT [6] are proposed and deployed over a variety of enforcement architectures.

The works in services composition are closely inspired from workflow and Data mashups composition. In [7] a framework for enforcing data privacy in workflows is described. In [8], the use of private data is reasoned for workflows. Privacy-preserving mechanism for data mashup is represented in [9]. It aims at integrating private data from different data providers in secure manner. The authors in [10] discuss the integration and verification of privacy policies in SOA-based workflows. The previous approaches, related to data mash up and workflows, focus on using algorithms (such as k-anonymity) for preserving privacy of data in a given table, while in our work we go further and propose a model that also takes into account usage restrictions and client requirements. The work [11] proposes using third parties as database service providers without the need for expensive cryptographic operations. However the proposed schemes do not allow queries to execute over the data of multiple providers and do not take into account the privacy issue regarding service provider and data consumer, which is the main focus of our work. In [12], privacy leakage in multi-party environment has been investigated. The approach takes a game-theoretic approach to analysis some of privacy assumption in the presence of colluding parties. It consists of a light-weight method to let each participant estimate the percentage of colluders in the environment. However, the secure multiparty based-methods involve a high computational cost in distributed system. One appealing approach is described in [13] and aims at preserving privacy of private data mash up with the social networks. The issue this approach resolves is to dynamically integrate data from different sources for the joint data analysis in the presence of privacy concerns.

The proposal of [14] is based on privacy policy lattice which is created for mining privacy preference-service item correlations. Using this lattice, privacy policies can be visualized and privacy negotiation rules can then be generated. The Privacy Advocate approach [15] consists of three main units: the privacy policy evaluation, the signature and the entities preferences unit. The negotiation focuses on data recipients and purpose only. An extension of P3P is proposed in [16]. It aims at adjusting a pervasive P3P-based negotiation mechanism for a privacy control. It implements a multi-agent negotiation

ISSN: 2320-1363

mechanism on top of a pervasive P3P system. The approach proposed in [26] aims at accomplishing privacy-aware access control by adding negotiation protocol and encrypting data under the classified level. Previous work are suffering from two major short comings: The first one is the ''take-it-or-leave-it'' principle, i.e., a service can only accept or refuse the other service's proposal as a whole. The second is the ''one-size-fits-all'' principle: once the service producer has designed its privacy policy, it will be proposed to all interested services no matter what their requirements are. Our privacy model goes beyond previous privacy approaches and aims at ensuring privacy compatibility of involved services in the composition without any additional overload. Moreover, it reconciles the incompatibility of privacy concerns using a negotiation protocol.

III.PRIVACY MODEL

Each service S has a privacy policy specifying the set of privacy practices applicable on any collected data and privacy requirements specifying the set of privacy conditions that a third-party service T must meet to consume S's data. The privacy model can be described as the following modules [7] [8] [9]

3.1 Privacy level

3.2 Privacy rule

3.3Privacy Assertion

3.4 Privacy Policy

3.5 Privacy Requirements

3.1 Privacy level

The privacy level can be defined for two resource called data and operation. The data level deals with data privacy. Resources refer to input and output parameters of a service (e.g., defined in WSDL). The operation level copes with the privacy about operation's invocation. Information about operation invocation may be perceived as private independently on whether their input/output parameters are confidential or not [10] [11] [12].

3.2 Privacy rule

Privacy rule is used to define how much sensitive the resource is. We can define privacy rule for both data and operation. We define a privacy rule by a topic, domain, level, and scope [12] [13].

3.3Privacy Assertion

The services will use privacy rules to define the privacy features of their resources. The application of a rule $R_i = (T_i, L_i, D_i, Sc_i)$ on rs is a privacy assertion $A(R_i, rs)$ where rs has Li as a level. $A(R_i, rs)$ States the granularity of rs that is subject to privacy. The granularity g belongs to the scope Si of the rule. g is equal to partial if only the ID of the operation invoker is private. $A(R_i, rs)$ Also indicates Di's values that are attributed to rs [10].

3.4 Privacy Policy

A service S will define a privacy policy, PPS, that specifies the set of practices applicable to the collected resources. Defining the privacy policy PPS of S is performed in two steps. First, the service S identifies the set (noted Pp) of all privacy resources. Second, S specifies assertions for each resource rs in Pp. Deciding about the content of Pp and the rules (from RS) to apply to each resource in Pp varies from a service to another. PPS specifies the way S treats the collected resources (i.e., received through the mediator) [11] [12].

3.5 Privacy Requirements

A service S will define a Privacy Requirements PR^sstating S's assertions describing how S expects and requires a third-party service should use its resources. Through privacy requirements, S applies its the right to conceal their data (i.e., output) [11] [12].

IV. Privacy Compatibility Checking

The privacy compatibility checking includes the modules called Privacy Subsumption and Privacy Compatibility algorithm [13] [10].

4.1 Privacy Subsumption

Let us consider a rule $R_i = (T_i, L_i, D_i, Sc_i)$ Defining an assertion A(Ri,rs) =(pf,g) for rs involving assigning value(s) from Di to the propositional formula pf of A. The values in Di are related to each other.

4.2 Privacy Compatibility Matching Algorithm

Here we propose an algorithm Privacy Compatibility Matching Algorithm to check the privacy compatibility of PR and

PP. The aim of PCM is to check that assertions in $PR^{S/T}$ and PP^{T} Are related via subsumption ships. PCM makes expectations in $PR^{S/T}$ to practice PP^{T} and expectations in PP^{T} to practices in $PR^{S/T}$.

PCM deals with the following three cases

Case 1: PCM matches a $PR^{S/T}$ assertion $A(R_i, rs)$ where rs is an input or operation usage to an assertion A` $(R_i', rs')\ in\ PP^{T}$ . In this case $A(R_i, rs)$ is S`s expectation and A`$(R_i', rs')$ is a $PP^{T}$ practice. If A`$\subseteq A$ and A` and A are matched.

Case 2: PCM is matches a $PR^{S/T}$ assertion $A(R_i, rs^{E})$ where rs^{E} is an output to an assertion $A`(R_i`, rs`^{P})$ in $PP^{T}$. In this case, $A(R_i, rs^{E})$ is a S`s expectation and $A`(R_i`, rs`^{P})$ is a $PP^{T}$ practice. If A`$\subseteq A$ and A` and A are matched.

Case 3: PCM is matches a $PR^{S/T}$ assertion $A(R_i, rs^{P})$ where rs^{P} is an output to an assertion $A`(R_i`, rs`^{E})$ in $PP^{T}$. In this case, $A(R_i, rs^{P})$ is a S`s expectation and $A`(R_i`, rs`^{E})$ is a $PP^{T}$ practice. If A`$\subseteq A$ and A` and A are matched [1] [10] [14].

Two options are possible while matching PR^{s} and $PP^{S`}$. The first option is to require full matching and the second is partial matching. Indeed, the mediator may opt for the second matching type in case when some services are willing to sacrifice their privacy constraints. For that purpose, we present a cost model-based solution to enable partial matching. The cost model combines the notions of privacy matching degree and threshold. Due to the large number and heterogeneity of DaaS services, it is not always possible to find policy PR^{s} that fully matches a S's requirementPR^{s}. The privacy matching degree gives an estimate about the ratio of PR^{s} assertions that match $PP^{S`}$ assertions. We refer to M $\subset$ $PP^{S`}$as the set of all such $PP^{S`}$assertions. The degree is obtained by adding the weights of all assertions in M: Degree $(PP^{S`}, PP^{S`})$ =$\sum w_i$ for all assertions $A_j(R_i, rs_k), w_j, M_j) \in M$.The privacy matching threshold T gives the minimum value allowed for a matching degree. The value of T is given by the service and gives

ISSN: 2320-1363

an estimate of how much privacy the service is willing to sacrifice [8] [9].

V.RESULTS AND DISCUSSION

We had implemented the prototype using GWT(Google Web Tool Kit) and apache tomcat server. We had experimented on many web services which includes include services providing medical information about patients, their hospital visits, diagnosed diseases, lab tests, prescribed medications, etc. we evaluate the efficiency and scalability of our compatibility algorithm. For each service deployed in our architecture, we randomly generated PR and PP files regarding its manipulated resources (i.e., inputs and outputs). Assertions in PR and PP were generated randomly and stored in XML files. All services were deployed over an Apache Tomcat 6 server on the Internet. We implemented our PCM algorithm in Java and run the composition system with and without checking compatibility. To evaluate the impact of PCM on the composition processing, we performed two sets of experiments.

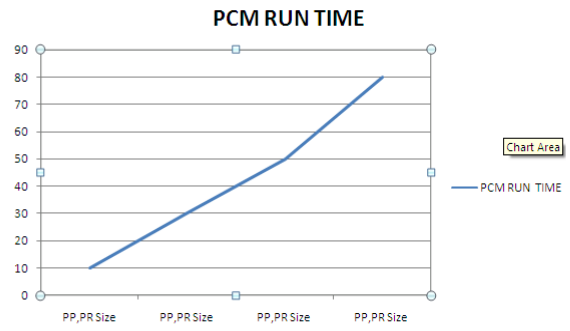

Fig 5.1 PCM Run time Analysis graph

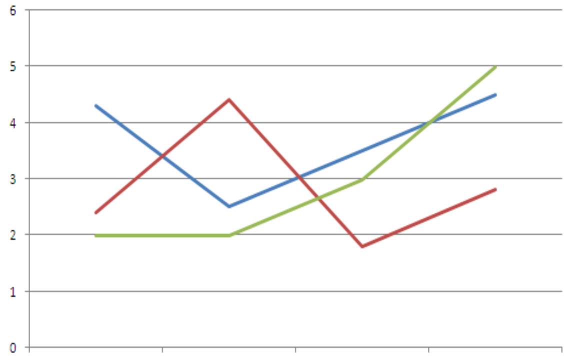

Fig5.2: Composition Plan

From the above figures we can observe that the performance of the PCM increases if the size of the composition (PP and PR).The time taken to process the request depends on the size of the offset.

VI.CONCLUSION AND FUTURE WORK

Dynamic privacy model for Web services deals with privacy at the data and operation levels. The model contains set of privacy policies and privacy rules that enhances the privacy matching compatibility. In any case, privacy policies always reflect the usage of private data as specified or agreed upon by service providers. As a future work, we aim at designing techniques for protecting the composition results from privacy attacks before the final result is returned by the mediator. Still the composition plan can predicts the target individual t contained in Tcp has target sensitive value s. we can extend our future work to prevent different types of attacks.

VII.REFERENCES

1. Salah-Eddine Tbahriti, Chirine Ghedira, Brahim Medjahed, and Michael Mrissa "Privacy-Enhanced Web Service Composition" IEEE TRANSACTIONS ON SERVICES COMPUTING, VOL. 7, NO. 2, APRIL-JUNE 2014.
2. W3C, The Platform for Privacy Preference Specification, 2004
3. A. Tumer, A. Dogac, and I.H. Toroslu, ''A Semantic-Based User Privacy Protection Framework for Web Services,'' in Proc.ITWP, vol. 3169, Lecture Notes in Computer Science, B. Mobasher and S.S. Anand, Eds., 2003, pp. 289-305.
4. S. Ran, ''A model forWeb services discovery with QoS,'' SIGecom Exchanges, vol. 4, no. 1, pp. 1-10, 2003.
5. Oasis. Extensible Access Control Markup Language (XACML). Identity, (v1.1):134, 2006.
6. M. Ka¨hmer, M. Gilliot, and G. Mu¨ ller, ''Automating Privacy Compliance with ExPDT,'' in Proc. 10th IEEE Conf. E-Commerce Technol./5th IEEE Conf. Enterprise Comput., E-Commerce and E-Serv., Washington, DC, USA, 2008, pp. 87-94.
7. Y. Gil, W. Cheung, V. Ratnakar, and K.K. Chan, ''Privacy Enforcement in Data Analysis Workflows,'' in Proc. Workshop PEAS ISWC/ASWC, vol. 320, CEUR Workshop Proceedings, T.Finin,L. Kagal, and D. Olmedilla, Eds., Busan, South Korea, Nov. 2007,CEUR-WS.org.
8. 8 .Y. Gil and C. Fritz, ''Reasoning About the Appropriate Use of

Private Data Through ComputationalWorkflows,'' in Proc. Intell.Inf. Privacy Manage., Mar. 2010, pp. 69-74, Papers from the AAAI Spring Symposium.

9. N. Mohammed, B.C.M. Fung, K. Wang, and P.C.K. Hung, ''Privacy-Preserving Data Mashup,'' in Proc. 12th Int'l Conf.EDBT, 2009, pp. 228-239.
10. Y. Lee, J. Werner, and J. Sztipanovits, ''Integration and Verification of Privacy Policies Using DSML's Structural Semantics in a SOA-Based Workflow Environment,'' J. Korean Soc. Internet Inf., vol. 10, no. 149, pp. 139-149, Aug. 2009.
11. J. Kawamoto and M. Yoshikawa, ''Security of Social Information from Query Analysis in DaaS,'' in Proc. EDBT/ICDT Workshops,2009, pp. 148-152.
12. H. Kargupta, K. Das, and K. Liu, ''Multi-party, Privacy-Preserving Distributed Data Mining Using a Game Theoretic Framework,'' in Proc. 11th Eur. Conf. Principles PKDD, 2007,pp. 523-531.
13. B.C.M. Fung, T. Trojer, P.C.K. Hung, L. Xiong, K. Al-Hussaeni,and R. Dssouli, ''Service-oriented Architecture for High-Dimensional Private Data Mashup,'' IEEE Trans. Serv. Comput., vol. 5, no. 3, pp. 373-386, 2012.
14. Y. Lee, D. Sarangi, O. Kwon, and M.-Y. Kim, ''Lattice Based Privacy Negotiation Rule Generation for Context-Aware Service,''in Proc. 6th Int'l Conf. UIC, 2009, pp. 340-352.
15. M. Maaser, S. Ortmann, and P. Langendo¨ rfer, ''The Privacy Advocate: Assertion of Privacy by Personalised Contracts,'' in Proc. WEBIST, vol. 8, Lecture Notes in Business Information Processing, J. Filipe and J.A.M. Cordeiro, Eds., 2007, pp. 85-97.