



**INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH
TECHNOLOGY**

Enhanced Technique for Protecting Web Service Composition from Privacy Attack

Sabrabeebe.M^{#1}, Dhanalakshmi.V²

^{*1,2} M.E Computer Science Engineering, IFET College of Engineering, Villupuram, Tamil Nadu, India
sabrabeebe@gmail.com

Abstract

Web service composition is a web technology that combines information from more than one source into a single web application. This technique provides a special type of composition application that aims at integrating data from multiple data providers depending on the user's request. In addition, DaaS (Data as a Service) composition may reveal privacy sensitive information. When enforcing a traditional privacy preserving model, such as privacy model and negotiation, the composed data would suffer from the problem known as the curse of privacy attacks. In this paper is used to propose a new dynamic privacy model with cognitive algorithm in order to extend DaaS composition with privacy capabilities and to enable fast access to data resources on the Web. This dynamic privacy model makes it possible to dynamically reconcile the privacy capabilities of services when incompatibilities arise in DaaS composition. Validate the applicability of proposal through an implementation and a set of experiments.

Keywords: Web Service Composition, DaaS Composition, Dynamic Privacy Model.

Introduction

A web service is a software function provided at a network address over the web or the cloud, it is a service that is "always on" as in the concept of utility computing. Thereby providing a well-documented, platform independent and interoperable method of interacting with their data. Web services have recently emerged as a popular medium for data publishing and sharing on the Web [1]. DaaS (Data-as-a-Service) Services where services correspond to calls over the data sources. DaaS eliminates redundancy and reduces associated expenditures by accommodating vital data in a single location, allowing data use and/or modification by multiple users via a single update point. Initially used in Web mashups, the DaaS strategy is often used by commercial organizations.

Data as a Service or DaaS is a cousin of software as a service. The emergence of service-oriented architecture (SOA) has rendered the actual platform on which the data resides also irrelevant. This development has enabled the recent emergence of the relatively new concept of DaaS. A service provider that enables data access on demand to users regardless of their geographic location [8]. Also called Data as a Service (DaaS), data services are similar to Software as a Service in that the information is stored in the cloud and is accessible by a wide range of systems and devices. Data services

can enable the data to be accessed and/or updated by multiple users while ensuring a single point for updates. Potential drawbacks to data services include server downtime from the data service provider, data loss in the event of a disaster, and the security of the data, both in its stored location and in the transmission of the data among users.

The DaaS approach delivers the following benefits:

- **Agility:** Because data is easily accessible, customers can take immediate action and do not require in-depth understanding of actual data.
- **Affordability:** Providers can construct a outsource in the presentation layer, which helps build highly affordable user interfaces and allows more feasible presentation layer change requests.
- **Data quality:** Data accessibility is controlled through data services, which improves data quality, as there is a single update point.

Web Service composition

Web service composition is a web technology that combines information from more than one source into a single web application and it is depends on the user request. This technique provides

a special type of composition application that aims at integrating data from multiple data providers depending on the user's request. The automatic selection, composition, and interoperation of Web services to perform some task, given a high-level description of an objective. A web service is any piece of software that makes itself available over the internet and uses a standardized XML messaging system [7].

XML is used to encode all communications to a web service. For example, a client invokes a web service by sending an XML message, then waits for a corresponding XML response. Because all communication is in XML, web services are not tied to any one operating system or programming language--Java can talk with Perl; Windows applications can talk with Unix applications [6]. Web Services are self-contained, modular, distributed, dynamic applications that can be described, published, located, or invoked over the network to create products, processes, and supply chains.

These applications can be local, distributed, or Web-based. Web services are built on top of open standards such as TCP/IP, HTTP, Java, HTML, and XML. Web services are XML-based information exchange systems that use the Internet for direct application-to-application interaction.

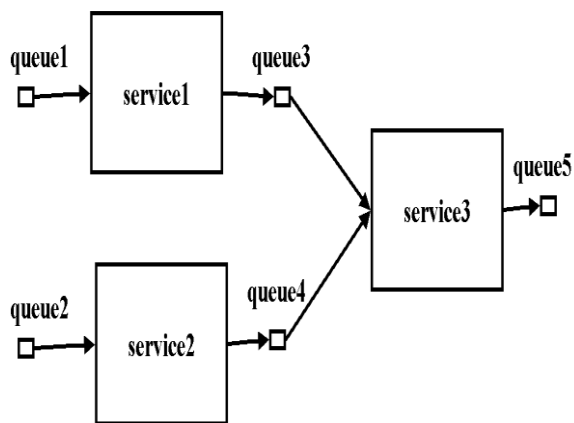


Fig.1 Service Composition

Disputes

There are two issues improve the problem of privacy in DaaS Composition. First, DaaS services collect and store a outsized amount of users information. Second, DaaS services are able to share this information with other services. By

sharing this type of information with third party service DaaS composition may reveal some personal information which will damage the privacy policy and privacy requirements of user.

1. **Privacy prerequisite.** Let us consider services S1 and S2 in web service composition. The administrator considers both input and output parameters of S1 (i.e., UserID and Dieses) as sensitive data. Let us now assume that this administrator states the "Health Insurance" service has an second service in DaaS service. In this service user got one unique insurance ID. The administrator may want to keep S1 invocation as private since this may disclose sensitive information to competitors.
2. **Privacy within compositions.**[1] Component services may require input data that cannot be disclosed by other services because of privacy concerns. They may also have conflicting privacy concerns regarding their ex- changed data. For instance, let us assume that S1 states to disclose its data (UserID) to a third-party service for use in limited time. S3 meanwhile attests that it uses collected data (UserID) for an unlimited time use. Then, S1 and S3 have different privacy constraints regarding the SSN. This will invalidate the composition in terms of privacy concerns.
3. **Operate with incompatible service.** The verification test will perform in between the privacy policy and privacy requirement in DaaS composition by using cognitive algorithm. If PP and PR is matched means it is compatible service if not means incompatible service.

Contributions

Dynamic Privacy Model: This model deals with privacy in the web service composition. To protecting the web service composition results from the privacy attacks before the final result is returned to the user. In this model extended with cognitive algorithm to improve the quality of service in web service composition. To access the resources form the web in the fast manner and providing the concealed information to users and third party service without revealing of privacy sensitive information of user and providing the compatibility services to the user and other service.

During registration, this model asks for your email address and certain other personal information.

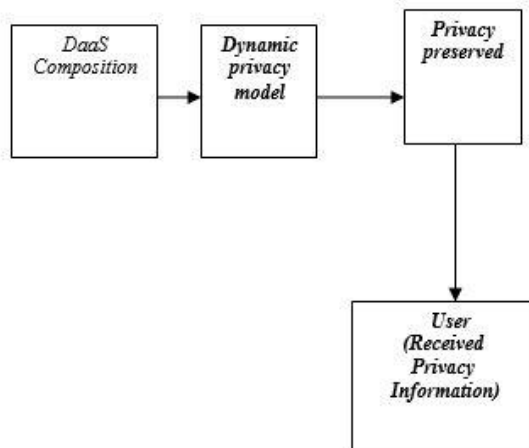


Fig.2 Protecting Privacy-sensitive information

DaaS Composition: Data as a Service (DaaS) is an information provision and distribution model in which data files (including text, images, sounds, and videos) are made available to customers over a network, typically the Internet. DaaS offers convenient and cost-effective solutions for customer- and client-oriented enterprises. DaaS eliminates redundancy and reduces associated expenditures by accommodating vital data in a single location, allowing data use and/or modification by multiple users via a single update point. Initially used in Web mashups, the DaaS strategy is often used by commercial organizations. Data as a Service or DaaS is a cousin of software as a service (SaaS). The emergence of service-oriented architecture (SOA) has rendered the actual platform on which the data resides also irrelevant.

This development has enabled the recent emergence of the relatively new concept of DaaS. The service provider that enables data access on demand to users regardless of their geographic location. Also called Data as a Service (DaaS), data services are similar to Software as a Service in that the information is stored in the cloud and is accessible by a wide range of systems and devices [3]. Data services can enable the data to be accessed and/or updated by multiple users while ensuring a single point for updates. Potential drawbacks to data services include server downtime from the data service provider, data loss in the event of a disaster, and the security of the data, both in its stored location and in the transmission of the data among users.

Privacy Compatibility Checking: This is capable of both assessing the compatibility and identifying incompatibility of service. Some personal information may be gathered when you register.

Privacy Compatibility Matching: To check the privacy compatibility of privacy policy and privacy requirement. PCM check the in assertions in Privacy Requirements with assertion in Privacy Policy for example assertion is service feature, preference, capability and requirements.

Negotiation to Reach Compatibility: The mediator basically discards any composition plan which is subject to privacy incompatibility from the set response. The main idea behind avoiding empty responses is to reach a compatible through a privacy-aware negotiation mechanism. Dynamically reconcile the privacy capabilities of services when incompatibilities arise in a composition.

Formal Model For Privacy Specification

A. Privacy Specification Model

In order to define an expressive model of privacy for web services, it is necessary first to examine the nature of data and to formally describe what we mean by privacy so that can argue that we protect such private data. In this case, the term privacy relates to the right of an entity to determine why, for whom, and for how long some information should be released. Due to the privacy subjectivity, each service has to identify which data are considered as *private* (noted as rs). If S provides some private data rs , then a set of privacy requirements applies to rs , and if S collects some private data rs , then a set of privacy policies also applies to rs .

1. **Privacy Rule:** A privacy rule R_i is defined by a tuple (H_i, I_i, Th_i) where H_i is the topic of R_i giving the privacy facet. For instance, the topic can describe: 3 *principle*, *beneficiary*, or *custody*. *Purpose* topic states the intent for which a given private data rs (collected or provided by S) will be used; the *beneficiary* topic mentions if and to whom rs can be revealed the *custody* topic specifies until when rs is stored by a third-party service. Then, for each topic H_i , a set I_i defines the value domain of the topic. The definition of I_i is based on an health insurance domain. For example, consider the privacy rule R_1 that corresponds to the topic $H_1 = \textit{beneficiary}$ and the domain $I_1 = \{\textit{public}, \textit{government}, \textit{private}, \textit{hospital}\}$. To define $G_i = \{\textit{total}, \textit{partial}\}$ as a granularity indicator, which states whether or not the data in rs , to which R_i applies, represent the

totality of the service input or output. In this paper, for the sake of simplicity, only consider the case where $G_i = \text{Partial}$. The definition of privacy rules, called *Rule Set* (RS), is described independently of any private data.

2. **Privacy Assertion:** The application of a rule $R_i = (H_i, I_i, T_i)$ to private data rs is a privacy assertion noted as $A(R_i, rs) = pf$. S specifies its privacy concerns for rs through $A(R_i, rs)$. For example, let consider $rs = \text{name}$ and $R1$ which corresponds to the topic $T1 = \text{beneficiary}$ and the domain $I1$. A privacy assertion applied to $rs = \text{address}$ through $R1$, which states that rs will be shared with *government agencies and hospital people*
3. **Privacy Requirements PRS:** "Personally identifiable information" is information that we can use to identify you as an individual. Personally identifiable information includes your name, address, telephone number and any other information that is connected with you personally. A service S providing some private data rs as its output specifies a set of privacy requirements, denoted as *PRS*, in terms of usage expectations.
4. **Privacy Policy PPS:** is a statement or a legal document that discloses some or all of the ways a party gathers, uses, discloses and manages a customer or client's data. Personal information can be anything that can be used to identify an individual, not limited to but including; name, address, date of birth, marital status, contact information, ID issue and expiry date, financial records, credit information, medical history, where you travel, and intentions to acquire goods and services. A service S requesting some private data rs' (where $rs' = rs$ referring output data of S) as input specifies its privacy policy, noted *PPS*, stating how S is going to use rs' . Privacy policy is important to the modern state, because grounded in it is the individual's physical and moral autonomy. For this reason, it is worthy of constitutional protection. The exact contents of a privacy policy will depend upon the applicable law and may need to address requirements across geographical boundaries and legal jurisdictions. Most countries have their own legislation and guidelines of who is covered, what information can be collected, and what it can be used for.

System Architecture

The approach presented in this paper is implemented as a project which deals with the privacy preservation issue in P2P data sharing environments [4]. This research where the need of data sharing is apparent for making better a health environment of people. To support the decision process multiple data sources such as the patient data, his social conditions, the geographical factors, etc. The data sources are provided by DaaS services and are organized with peers. DaaS services differ from traditional Web services, in that they are stateless i.e. they only provide information about the current state of the world but do not change that state. When such a service is executed, it accepts from a user an input data of a specified format ("typed data") and returns back to the user some information as an output. DaaS services are modeled by RDF views. Figure 3 summarizes the architecture of this project. The Multi-Peer Query Processing component is in charge of answering the global user query. The latter has to be split local queries (i.e., sub-queries) and has to determine which peer is able to solve a local query. Each sub-query is expressed in SPARQL [9]. Each peer handles a Mediator equipped with a Local Query Processing Engine component. The mediator exploits the defined RDF views within WSDL files to select the services that can be combined to answer the local query using an RDF a query rewriting algorithm [2]. Then, it carries out all the interactions between the composed services and generates a set of composition plans to provide the requested data.

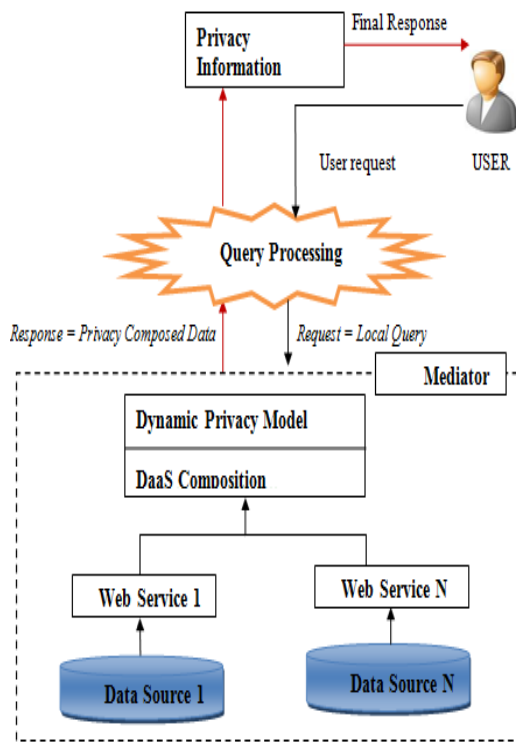


Fig.3 System Architecture

This figure 3 summarizes the architecture of this project. The Multi-Peer Query Processing component is in charge of answering the global user query. The latter has to be split local queries (i.e., sub-queries) and has to determine which peer is able to solve a local query. Each sub-query is expressed in SQL. Each peer handles a Mediator equipped with a Local Query Processing Engine component. Then, it carries out all the interactions between the composed services and generates a set of composition plans to provide the requested data. It deals with privacy not only at the data level (i.e., inputs and outputs) but also service level. In this paper to build upon this model two other extensions to address privacy issues during DaaS composition. The privacy model described in this paper is based on the model.

DAAS Composition

In DaaS composition services are Combines information from more than one service into a single web application. Combining data from different sources could potentially reveal privacy-sensitive information. To securely integrate data from different sources could potentially reveal privacy-sensitive information. To securely integrate private data from different data providers, whereas the integrated data still retains the essential information. A web service is any piece of software that makes itself available

over the internet and uses a standardized XML messaging system. XML is used to encode all communications to a web service. For example, a client invokes a web service by sending an XML message, then waits for a corresponding XML response. Because all communication is in XML, web services are not tied to any one operating system or programming language--Java can talk with Perl. Windows applications can talk with Unix applications. Web Services are self-contained, modular, distributed, dynamic applications that can be described, published, located, or invoked over the network to create products, processes, and supply chains.

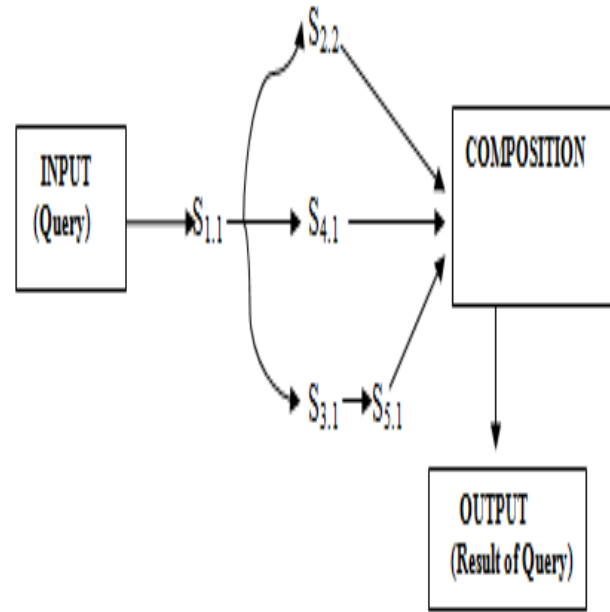


Fig.4 Composition Plan

Data as a Service or DaaS is a cousin of software as a service. The emergence of service oriented architecture (SOA) has rendered the actual platform on which the data resides also irrelevant. This development has enabled the recent emergence of the relatively new concept of DaaS. A service provider that enables data access on demand to users regardless of their geographic location. Also called Data as a Service (DaaS), data services are similar to Software as a Service in that the information is stored in the cloud and is accessible by a wide range of systems and devices. Data services can enable the data to be accessed and/or updated by multiple users while ensuring a single point for updates. Potential drawbacks to data services include server downtime from the data service provider, data loss in the event

of a disaster, and the security of the data, both in its stored location and in the transmission of the data among users.

The Privacy Compatibility Checking

This method is Capable of both assessing the compatibility and identifying incompatibility of service. Some personal information may be gathered when you register. During registration, this model asks for your email address and certain other personal information [5]. Introduce the notion of compatibility between privacy policies and requirements. Then define the notion of privacy subsumption and present our cost model based privacy matching mechanism.

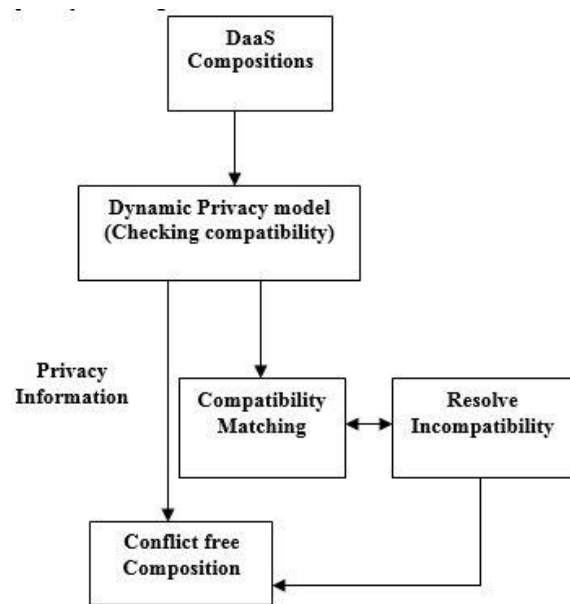


Fig.5 Compatibility Checking

Privacy Compatibility Matching

To check the privacy compatibility of privacy policy and privacy requirement in DaaS Composition Privacy Compatibility Matching (PCM) check the in assertions in Privacy Requirements(PR) with assertion in Privacy Policy (PP) The example of assertion are service feature, preference, capability and requirements. In order to check the compatibility between assertions of two services S and S' respectively[3], to introduce a Privacy Compatibility Matching algorithm (PCM). The semantics of PCM are described as follows. For each assertion $APRS$ and $A' PPS$ to be compatible, A must subsume A' in terms of privacy.

Dynamic privacy model with cognitive Algorithm

INPUT: Assertion of Privacy policy (PP) with match maker service

INPUT: Assertion of Privacy Requirement (PR) with selection service

OUTPUT: If there is an incompatibility between PR and PP for each rule set $rs = rs'$ to be compatible

for $i=1$, if $i < RS$ do

for $j=1$, $j < PR$ do

for $j'=1$, $j' < PP$ do

if A_j' subset of A_j then

A_j is compatible to A_j' and posses good level of trust else $Inc \leftarrow (A_j, A_j')$

Goto step 4

End.

Negotiation to Reach Compatibility

There are two different aspects to be considered when dealing with the privacy issue in the context of service composition. The first aspect is related to describing web service concerns with respect to privacy. The second aspect relates to evaluating how services can work together in the composition. The mediator discards any composition plan which is subject to privacy incompatibility from the set response. The main idea behind avoiding empty responses is to reach a compatible through a negotiation mechanism. Dynamically reconcile the privacy capabilities of services when incompatibilities arise in a composition. Compared to, in this paper, we revise the previous idea of negotiation and provide many improvements. The negotiation decision is taken according to a utility-based cost function defined by a service provider [1].

Conclusion and Future Work

In this paper, the proposed a dynamic privacy model for web services. This model deals with privacy at the DaaS. Composition to tackle the incompatibilities between privacy policies and requirements. Further, dynamic privacy must reveal with privacy sensitivity information whenever the privacy attack occur. Although privacy cannot be carelessly negotiated, it is still possible to negotiated apart of privacy policy for specific purposes. In any case, privacy policies always reflect the usage of private data as specified or agreed upon by service providers. As a future work, the aim at designing technique for protecting the composition results from privacy attacks in cloud computing.

Acknowledgment

The satisfaction that accompanies the successful completion of any task would be

incomplete without mentioning the names of people who made it possible, whose constant support and encouragement throughout success. I express my sincere thanks to my Project guide Ms.C. Nancy Nightingale M.Tech., for giving the constant support and valuable guidance during the time of my project. Finally, I extend my thanks to parents and friends who had giving their support throughout my course and encouragement during my project.

References

- [1] Salah-Eddine Tbahriti, Chirine Ghedira, Brahim Medjahed and Michael Mrissa. Privacy-Enhanced Web Service Composition. "IEEE Transactions on Services Computing", vol 3, March 2013 .
- [2] Salah-Eddine Tbahriti, Chirine Ghedira, Brahim Medjahed Member, IEEE, Michael Mrissa, and Djamel Benslimane. How to Enhance Privacy Within DaaS Service Composition . "IEEE systems journal", vol. 7, no. 3, september 2013.
- [3] M. Barhamgi, D. Benslimane, and B. Medjahed. A Query Rewriting Approach for Web Service Composition. "IEEE Transactions on Services Computing" (TSC), 3(3):206–222, 2010.
- [4] G. T. Duncan, T. B. Jabine, and V. A. de Wolf, editors. Private lives and public policies: confidentiality and accessibility of government statistics. "National Academy Press", Washington, DC, USA, 1993.
- [5] 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 Transactions on Services Computing", 99 (PrePrints), 2011.
- [6] Y. Gil, W. Cheung, V. Ratnakar, and K. kin Chan. Privacy enforcement in data analysis workflows. In T. Finin, L. Kagal, and D. Olmedilla, editors, "Proceedings of the Workshop on Privacy Enforcement and Accountability with Semantics" (PEAS2007) at ISWC/ASWC2007, Busan, South Korea, volume 320 of CEUR Workshop Proceedings. CEUR-WS.org, November 2007.
- [7] Y. Gil and C. Fritz. Reasoning about the appropriate use of private data through computational workflows. In Intelligent Information Privacy Management, "Papers from the AAAI Spring Symposium", pages 69–74, March 2010.
- [8] B. Hore, S. Mehrotra, and G. Tsudik. A privacy-preserving index for range queries. "In Proceedings of the Thirtieth international conference on Very large data bases" - Volume 30, VLDB '04, pages 720–731. VLDB Endowment, 2004.
- [9] M. K'ahmer, M. Gilliot, and G. M'uller. Automating privacy compliance with expdt. "In Proceedings of the 2008 10th IEEE Conference on E-Commerce Technology and the Fifth IEEE Conference on Enterprise Computing, E-Commerce and E-Services", pages 87–94, Washington, DC, USA, 2008. IEEE Computer Society.
- [10] H. Kargupta, K. Das, and K. Liu. Multi-party, privacy-preserving distributed data mining using a game theoretic framework. "In Proceedings of the 11th European conference on Principles and Practice of Knowledge Discovery in Databases", PKDD 2007, pages 523–531, Berlin, Heidelberg, 2007. Springer-Verlag.