Requirement-Oriented Privacy Protection Analysis Architecture in Cloud Computing

Changbo Ke[1, 2]*, Ruchuan Wang[1, 2], Fu Xiao[1, 2, 4], and Zhiqiu Huang[3]
[1]School of Computer Sci. & Tech./School of Software, Nanjing Univ. of Posts and Telecom., Nanjing 210023, China
[2]Jiangsu High Tech. Research Key Laboratory for Wireless Sensor Networks, Nanjing, Jiangsu, 210003, China
[3]College of computer Sci. and Tech., Nanjing Univ. of Aeronautics and Astronautics, Nanjing, Jiangsu, 210016 China
[4]Key Lab of Broadband Wireless Communication and Sensor Network Tech. (Nanjing University of Posts and Telecom.), Ministry of Education Jiangsu Province, Nanjing, Jiangsu, 210003, China
Email: brobo.ke@njupt.edu.cn; wangrc@njupt.edu.cn, xiaof@njupt.edu.cn; zhqhuang@nuaa.edu.cn

Abstract —As a new software paradigm, cloud computing provides services dynamically according to user requirements. However, it is difficult to control personal privacy information because of the opening, virtualization, multi-tenancy and service outsourcing characters. Therefore, how to protect user privacy information has become a research focus. In this paper, we propose requirement-oriented privacy protection theory analysis architecture and implementation platform. Firstly, the theory analysis architecture is depicted as layers, and we analyze the function and key technologies of every layer. Secondly, we address the privacy property description method with ontology and description logic, and then analyze the theory modules of privacy items conflict checking layer, privacy policy negotiation layer and privacy agreement forensics layer. Thirdly, according to theory architecture, we design the implementation platform of requirement-oriented privacy protection, and discuss the function and workflow. In the end, we conclude and point out the future work.

Index Terms—Privacy protection, privacy items, privacy policy, privacy property, cloud computing

I. INTRODUCTION

Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction [1]. There are some characters, such as service outsourcing, virtualization, distribution and multi-tenancy. These characters enhance the service quality and save the computing resources, for example, service outsourcing enhances the service capability and specialization through service composition [2]. However, the transparency of privacy information to the outsourcing service provider, make users worry that privacy data be illegally propagated and used. For example, Google was sued by some users in America and was investigated by European Union, because of its new unified privacy policy implemented from Mar. 1st, 2012. According to the analysis by America Electronic Privacy Information Center, Google new privacy policies do not take into account how to use privacy data in the product, and to whom privacy data be propagated according to user privacy requirement, and may have conflicts with local laws. Therefore, privacy protection in cloud computing has become research focus.

Privacy was proposed as the human right in the beginning [3]. In domain of software engineering, privacy protection means the capability of preventing individual information from being collected, disclosed and stored by others [4]. The Platform for Privacy Preferences (P3P) [5] provides a standard and machine-understandable privacy policy, which matches with user privacy preference. According to the matched results, user can select services. However, the P3P lacks semantic information and only applies to Web Site, not supporting service composition. Therefore, P3P does not apply to cloud computing, since all entitles in cloud computing are service or composite service. Extensible Access Control Markup Language (XACML) 2.0 [6] [7] extends the privacy policy through profile and applies to the cloud service. However, it hardly guarantees the composite service satisfying user privacy requirement. Pearson, S *et al*. [8] [9] defined privacy protection in cloud computing as the capability of user controlling Personal Sensitive Information (PSI) not be collected, used, disclosed and stored by cloud service provider. They provided certain theoretical guidance, but do not put forward specific solution method.

In order to satisfy user privacy requirement and protect user privacy data in the business process, we propose requirement-oriented privacy protection analysis architecture in cloud computing, and develop a prototype system and run a case to prove the feasibility and practicability of the architecture.

The other parts of this paper are structured as the follows: Section 2 we introduce related works. Section 3 we address description of privacy requirement. Section 4 we depict privacy protection architecture and analysis, including privacy items conflict checking, privacy policy

Manuscript received September 2, 2014; revised January 30, 2015.
This work is supported in part by the National Science Foundation of China under grants (No.61272083, No.61373137, No.61373017), Major Program of Jiangsu Higher Education Institutions under grant No.14KJA520002, Six Industries Talent Peaks Plan of Jiangsu under grant No.2013-DZXX-014 and Jiangsu Qinglan Project.
Corresponding author email: brobo.ke@njupt.edu.cn.
doi:10.12720/jcm.10.1.55-63

negotiation and privacy agreement supervising. Section 5 we put forward privacy protection implementation platform. In the end, we conclude and point out the future work in section 6.

II. Related Works

We classify the related works of privacy protection as computing process oriented and data oriented privacy protection. The former is classified into five categories, which are modeling and verification of privacy requirement, matching and negotiation of privacy policy, disclosure and risk. In the mean time, we compare them from contribution, applied computing paradigm, whether supporting service composition and whether supporting semantic, and highlight our work in the table. The privacy policies are defined by different users or service providers, which may cause various understanding of the same words. The privacy policies supporting semantic can improve the matching chance. Most of services in cloud computing are composite services. Therefore, whether supporting service composition is very important. We specially compare related works from aspects of semantic and service composition.

In this paper, we major discuss the requirement-oriented privacy protection analysis architecture in cloud computing. Our work applies to service composition and contains semantic information, while most of other works do not. Details showed as below Table I.

TABLE. I: COMPARISON OF RELATED WORKS

Methods	Authors	Contributions	Computing Paradigm	Support for Service Composition	Support for Semantic
Modeling and Verification	Jiajun Lu *et al*. [10]	Verification of Behavior-aware Privacy Requirements in Web Services Composition	Service Computing	√	×
	N. Guermouche, S *et al*. [11]	Privacy-aware Web service protocol replaceability	Service Computing	×	√
	Mokhtari K *et al*. [12]	Verification of Privacy Timed Properties	Service Computing	×	×
	LinYuan Liu *et al*. [13]	Minimal privacy authorization in web services collaboration	Service Computing	√	×
Matching	Changbo ke *et al*. [14]	Service Outsourcing Character Oriented Privacy Conflict Detection Method in Cloud Computing	Cloud Computing	√	√
	WEI Zhiqiang *et al*. [15]	Privacy-Protection Policy for Pervasive Computing	Pervasive Computing	×	√
Negotiation	Changbo ke *et al*. [16]	Supporting negotiation mechanism privacy authority method in cloud computing	Cloud Computing	√	√
	Tbahriti S *et al*. [17]	Privacy-Enhanced Web Service Composition	Service Computing	×	×
	ZHANG Yan *et al*. [18]	Parsimonious Semantic Trust Negotiation	N/A	N/A	√
Disclosure	Jan Kolter *et a*l. [19]	Visualizing Past Personal Data Disclosures	Service Computing	×	×
	LinYuan Liu *et al*. [20]	Analysis of the minimal privacy disclosure	Service Computing	√	×
Risk	T. Yu, Y et al. [21]	Modeling and Measuring Privacy Risks	Service Computing	√	×
	Dan Svantesson *et al*. [22]	Privacy and consumer risks in cloud computing	Cloud Computing	N/A	N/A
Analysis Architecture	**Our work**	**Requirement-oriented Privacy Protection Analysis Architecture in Cloud Computing**	**Cloud Computing**	**√**	**√**

N/A: NOT APPLICABLE √: SUPPORT ×: NOT SUPPORT

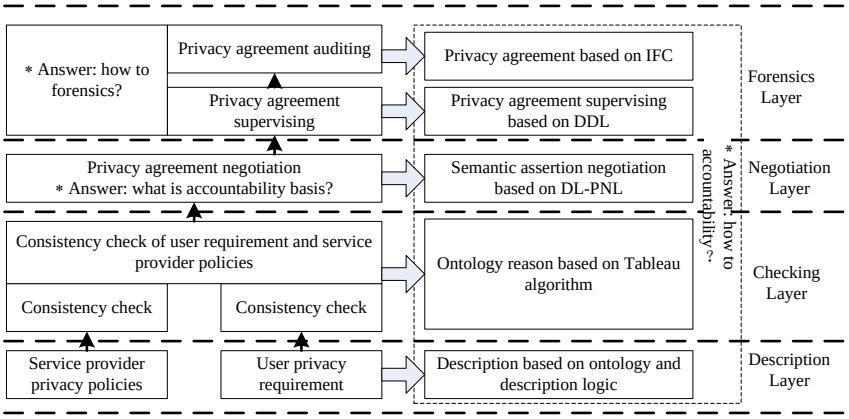

Fig. 1. Requirement-oriented privacy protection architecture

III. Privacy Protection Theory Analysis Architecture

A. Privacy Protection Architecture

Requirement-oriented privacy protection architecture includes description layer, checking layer, negotiation layer and forensics layer.

Description Layer: Privacy requirement of user and service provider is depicted with ontology and description logic, so that formal expression have context semantic and machine-understandable. This layer is basis of other layers.

Checking Layer: According to context semantic, check the consistency of both the privacy requirement of user and privacy policy of service provider respectively. Then check the consistency between the privacy requirement of user and privacy policy of service provider, to gain the privacy items sequence.

Negotiation Layer: To match corresponding privacy policy of privacy items with privacy negotiation language based on description logic. Mismatched privacy policy will be negotiated with privacy assertion negotiation mechanism, to obtain the privacy agreement that meet both user and service provider.

Forensics Layer: To instrument for cloud service composition execution flow BPEL, and then verify with dynamic description logic to supervise requirement. For privacy requirement with time property, analyze and audit the supervising log after BPEL is executed, in order to find the service providers that violate the privacy agreement, namely, to forensic the violations.

Details as showed in Fig. 1.

B. Description of Privacy Requirement

Through analyzing the description document of atom service, which joins in service composing in cloud computing, negotiation engine obtains the user privacy information to be used or disclosed from input and pre-condition. In the mean time, through analyzing user privacy requirement, negotiation engine obtains the privacy information to be protected by user. Both obtained privacy information are known as privacy attributes.

Under the framework of privacy policy negotiation, conflict detection of privacy attributes and exchange of privacy disclosure assertion between negotiating parties are automatically done by respective negotiation engine. Therefore, privacy policy assertion needs to be machine-understandable, supporting context semantic reasoning. In this paper, we take advantage of ontology and description logic, to describe and reason the privacy policy at the bottom of the framework. Before elaborate the relative theory of privacy description, we present an instance that applied throughout the text.

Instance: There are following participants in this instance, Seller, Buyer, ES (Service Composer), Bank, Shipper, Post office. ES has the business license issued by Industrial and Commercial Bureau, certificated as legal E-commerce platform. ES can issue reputation certification for those VIP sellers who have reputation value over 600, or have credit over 6000 issued by bank. Bank can issue credit card certification to both Seller and Buyer.

One day, Tom wants to purchase some Furniture from service provider corporation S via cloud service composer CSC. Tom has the following privacy requirements. Furniture Corporation S needs better to be VIP seller. In transaction, only VIP seller can obtain Tom *realName* and *phoneNumber* (Mobile phone), while non-VIP seller can only obtain Tom *nickName* or *officePhoneNumber*. All phone number can only be provided to corporation S and Shipper. Tom bank account can only be provided to Bank. Address and zip code can only be provided to Shipper. In certain period after transaction is done, all participants have to delete all privacy information automatically. Maintenance service is provided through tracking number and phone number. Corporation S especially requires getting feedback about service quality through phone number and address, from those customers with amount over $20,000. Considering the above requirements, the point is that whether Tom can successfully negotiate with S to obtain interaction sequence, and then place an order or not.

We obtain the relationship of privacy attributes through mapping among knowledge ontology, and then describe it with ontology tree. The ontology tree of privacy attributes is showed in Fig. 2. Thing is root node, which is super class of all privacy attributes. Except the bottom layer, the relationship between all the other layers is subsuming, namely, relationship of class and subclass. For example, class *name* includes subclass *realname* and *nickname*. Only the bottom layer belongs to its upper layer, for example, *Country*, *Province* and *City* belong to class *Address*.

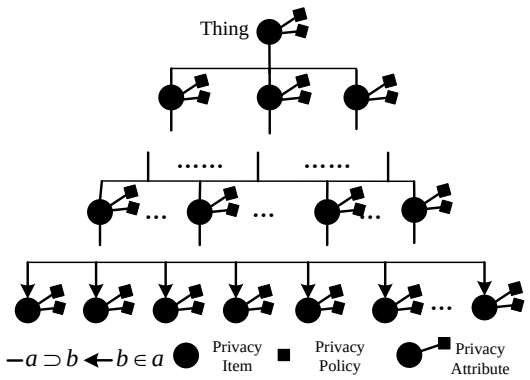

Fig. 2. Ontology Description of Privacy Attributes

Definition 1 Privacy Attribute: We describe privacy attribute as 5-tuple, namely,

$$PA-S/PA-U =< Issuer, Owner, Subject, PDA-S/PDA-U, Signature >$$

Issuer represents the mark of privacy attribute, to record the farther class and son class of a privacy attribute class. *Owner* represents participants of service

composition and also the owner of privacy attribute. *Subject* represents name of the privacy attribute. *PDA-S/PDA-U* represents Privacy Disclosure Assertion of Services or users. *Signature* is digital signature. Each privacy attribute corresponds to one concept in privacy attribute ontology. *Subject* can be described as follows:

$$Subject = C(op_1 : I_1, op_2 : I_2, op_3 : I_3 ... op_n : I_n) \vee C(dp_1 : D_1, dp_2 : D_2, dp_3 : D_3 ... dp_n : D_n)$$

where C represents class of privacy attribute ontology, op_i and dp_i represents object property and digital property respectively, belonging to class, satisfying $\forall(i \neq j) \rightarrow (op_i \neq op_j) \wedge (dp_i \neq dp_j)$, I_i represents instance of privacy attribute ontology, and D_i is one certain value or constant.

Definition 2 Delegation of Authority Statement DAS: DAS is part of privacy disclosure assertion, namely, DAS must be satisfied for each disclosure of privacy attribute, to prove if service participants have the authority of privacy attribute or not. It can be expressed as below:

$$DAS : \left[(serverConstr \otimes authorizer) \Leftarrow assertion \right]$$

Suppose *authorizer* = {*official*, *serviceComp*}, whereas *official* represents official organization, *serviceComp* represents service composer, *serverConstr* represents the rules or law constraint on service participants issued by official organization or service composer. For example, ES is e-commerce platform. DAS clarified that service participants must satisfy the assertion expression. Valid DAS must include signature on the assertion by official organization or service composer, and in the mean time, the signature is included in the privacy attribute.

Example 1: The DAS issued by ES is as follows.

To be VIP user of ES, the user must have reputation value over 600 or have credit over 6000 issued by bank. It can be express as below:

$$VIP \otimes EBay \Leftarrow credit\left[> 6000(rating) \right] \otimes Bank \vee reputation\left[> 600(value) \right] \otimes ES \quad (1)$$

Definition 3 Privacy Disclosure Assertion PDA-S: Privacy Disclosure Assertion is a constraint assertion for certain service participant to own certain privacy attribute, it can be expressed as below:

$$PDA-S = subjConstr\ \Theta\ ownerConstr;$$
$$subjConstr = C\left(op_1 : \{ow_1 \cdots ow_n\} ... op_n : \{ow_1 \cdots ow_n\}\right) \wedge C\left(op_1 : ow_1(vt_1) \cdots op_n : ow_n(vt_n)\right);$$
$$ownerConstr = DAS;$$

Privacy Disclosure Assertion PDA-S is composed of two parts, namely, constraint on subject of privacy attribute *subjConstr* , and constraint on service participant who own the privacy attribute *ownerConstr* . The former *subjConstr* mainly set constraints on the ownership of privacy attribute and validity period of using the privacy attribute by service participants. C represents class of privacy attribute, specifying that privacy attribute exchange can only be processed among corresponding class or subclass. $op_1 : \{ow_1 ... ow_n\}$, in which op_1 represents some instance of class and $\{ow_1 ... ow_n\}$ represents service participants corresponding to the instance. $op_1 : \{ow_1 \cdots ow_n\} \cdots op_n : \{ow_1 \cdots ow_n\}$ represents all instances of class that owned by certain service participants. $op_1 : ow_1(vt_1) \cdots op_n : ow_n(vt_n)$ represents validity time for service participants owning the privacy attribute instance. If data domain of vt_i is integer, then ow_i is a 1-tuple predicate based on data domain of vt_i . Common predicate is $\geq a$ or $\leq a$, in which *a* is constant.

The latter *ownerConstr* mainly set constraints on service participants who own privacy attributes, $ownerConstr = DAS$ means that service participants who own privacy attributes must also meet DAS requirement.

Example 2: Corporation S plans to sell furniture via ES on internet. The privacy disclosure assertion of user address issued by ES is as follows:

Supposing corporation S is ES VIP seller. ES requires buyer address can only be disclosed to shipper, and be deleted within 2 hours after goods delivered and deal is finished, we can express it as below,

$$address : shipper \wedge \left[address : \ \leq 2hours\ (vaildtime)\ \Theta\ Seller : VIP \otimes ES \right] \quad (2)$$

Take formula (1) into formula (2), we can obtain:

$$address : shipper \wedge \{address : \ \leq 2hours\ (vaildtime) \Theta credit\left[> 6000(rating) \right] \otimes\ Bank \vee reputation\left[> 600(value) \right] \otimes ES\} \quad (3)$$

Definition 4 Privacy Discloser Strategy (PDS): PDS is an ordinal sequence of privacy discloser assertion, namely,

$$PDS = PDA-S_1 \wedge PDA-S_2 \wedge PDA-S_3 \wedge \cdots \wedge PDA-S_n$$

Definition 5 Privacy Disclosure Assertion of User (PDA-U): Privacy Disclosure Assertion of User is a constraint assertion for certain service provider to own certain privacy attribute, it can be expressed as below:

$$PDA-U = subjConstr\ \Theta\ serverConstr;$$
$$subjConstr = C\left(op_1 : \{ow_1 \cdots ow_n\} \cdots op_n : \{ow_1 \cdots ow_n\}\right) \wedge C\left(op_1 : ow_1(vt_1) \cdots op_n : ow_n(vt_n)\right);$$
$$serverConstr = trustDegree;$$

Privacy Disclosure Assertion PDA-U is composed of two parts, namely, constraint on subject of privacy attribute *subjConstr*, and constraint on service provider who own the privacy attribute *serverConstr*. The expression of *subjConstr is* the same as that in Definition 3, while *serverConstr* is determined by user trust degree on service or service provider. Trust degree can be expressed as *trustDegree* = {*S*, *DAS*, *Re*}, whereas S

represents Security, to certify the truth and integrity of data and trustworthy of QOS. *DAS* is delegation authority statement that owned by service or service provider. Re represents reputation of service or service provider. *DAS* usually is issued by official organization or service composer to service provider, for example, stating whether Seller is VIP, whether Shipper or Bank have business license issued by official organization or not. We assume all shipper and bank have business license in this paper.

Definition 6 Privacy Preference PP: PP is an ordinal sequence of privacy discloser assertion. Namely,

$$PP = PDA-U_1(C_1(op_1)) \wedge PDA-U_2(C_2(op_2)) \wedge PDA-U_3(C_3(op_3)) \wedge \cdots \wedge PDA-U_n(C_n(op_n))$$

C. Privacy Items Conflict Checking

There are two layers for privacy conflict detection framework, as showed in Fig. 3.

Pre-detection Layer: The part with slash background in Fig. 3 represents Privacy Conflict Pre-detection Layer. This part mainly implements three functions as follows:

1) User privacy requirement is translated into privacy preference assertion $\{\varphi_1, \varphi_2, \varphi_3, \cdots, \varphi_n\}$ by user privacy preference editor (PPE).

2) User comment information and Qos in service description document (SDD) are evaluated by trust degree calculator (TDC), so as to obtain the trust degree value for services.

3) The input and precondition in service description document is captured by Xpath, and input and precondition is refined into privacy property.

At last, the privacy preference assertion, trust degree and privacy property are saved into privacy conflict detection knowledge base.

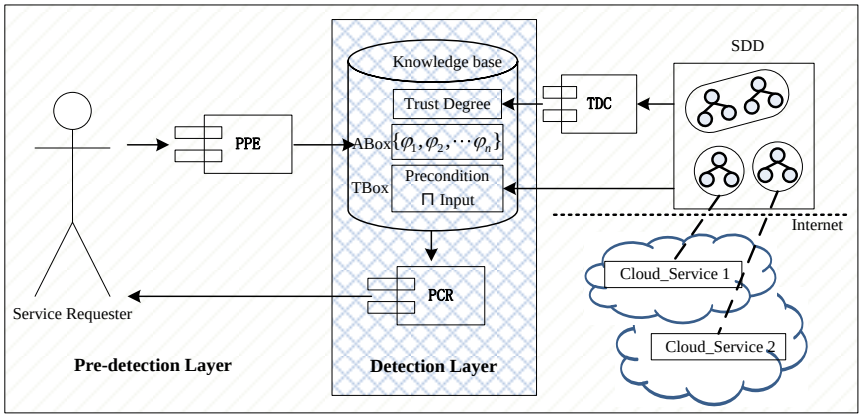

Fig. 3. Framework of privacy conflict detection

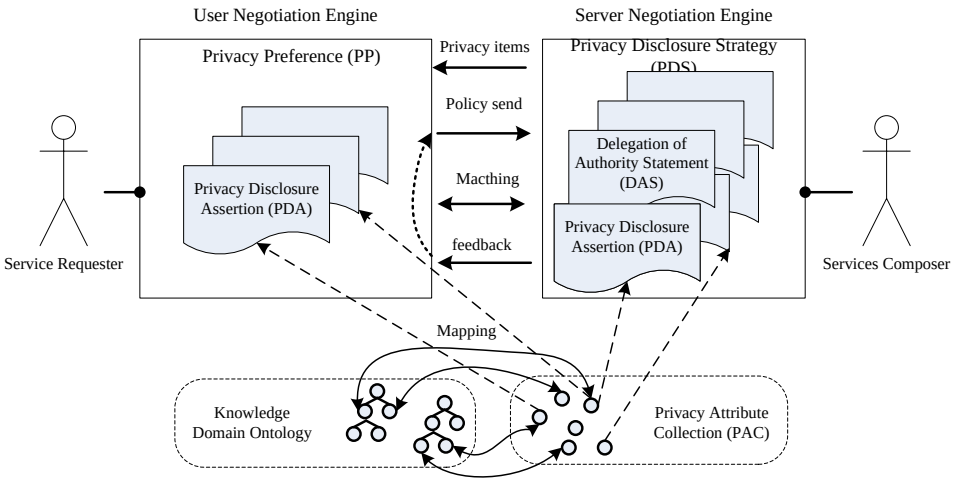

Fig. 4. Privacy policy negotiation framework

Detection Layer: The part with grid background in Fig. 3 represents Privacy Conflict Detection Layer.

Privacy conflict detection layer contains knowledge base and privacy conflict reasoner (PCR), in which knowledge base is made up of privacy preference assertion, trust degree and privacy property. In this layer, privacy conflict detection for knowledge base is implemented by privacy conflict reasoner and the detection result is returned to user.

D. Privacy Policy Negotiation

The framework of privacy policy negotiation has two layers, as showed in Fig.4.

1) Mapping layer, which supports the mapping

between Privacy Attribute Collection (PAC) and Knowledge Domain Ontology (KDO), so that the semantic relationship among privacy attributes can be determined and privacy attribute ontology can be set up. During the period of the privacy policy pre-negotiation, once conflicts are detected, negotiation engine can substitute the conflicted attribute with brother attribute of ontology tree, which is found by semantic relationship among privacy attributes, and find the privacy attribute sequence that satisfying user privacy requirement.

2) Negotiation layer, which has two periods, namely, pre-negotiating period and Privacy Disclosure Assertion (PDA) exchange period.

During the period of pre-negotiation, firstly, negotiation engine analyzes the user requirement document and service input and pre-condition provided by service provider, respectively, obtaining user Privacy Preference (PP) and service Privacy Attribute Collection (PAC). Secondly, detects the conflict between PP and PAC, to discover privacy attribute that not satisfying user privacy requirement. Thirdly, search engine substitutes the discovered privacy attribute with brother attribute of ontology tree through calling the mapping layer.

During the PDA exchange period, exchange the corresponding PDA of service privacy attribute and user privacy requirement, and iterate this exchange process. Through this process, PDA collection that satisfying both service provider and user, namely, Privacy Disclosure Strategy (PDS) may be found. Then PDS is included in Service Level Agreement (SLA).

E. Privacy Agreement Supervising

Privacy agreement-oriented supervising framework can be expressed with two layers, as showed in Fig.5.

1) Analysis layer:

Static analyze the BPEL process with BPEL Analysis Engine before execution. Detailed analysis process as follows: at first, we capture invoke mark by using Xpath, then insert probe DAS as precondition of calling outsourcing service. If precondition DAS is unsatisfied, then terminate BPEL process. If satisfied, then supervise the execution of outsourcing service with Supervisor, in which each privacy item {*subject: name*} corresponds with PDA in SLA.

2) Supervising layer:

With BPEL Execution Engine execute the BPEL process that have been analyzed and instrumented, and in the mean time save the supervision log into Supervision Log Repository. If precondition DAS is unsatisfied for an outsourcing service, then terminate the process and substitute the outsourcing service with candidate by using BPEL Analysis Engine. If one or more PDAs in SLA are violated, then punish the outsourcing service according to corresponding punishment rule in SLA and decrease the trust degree.

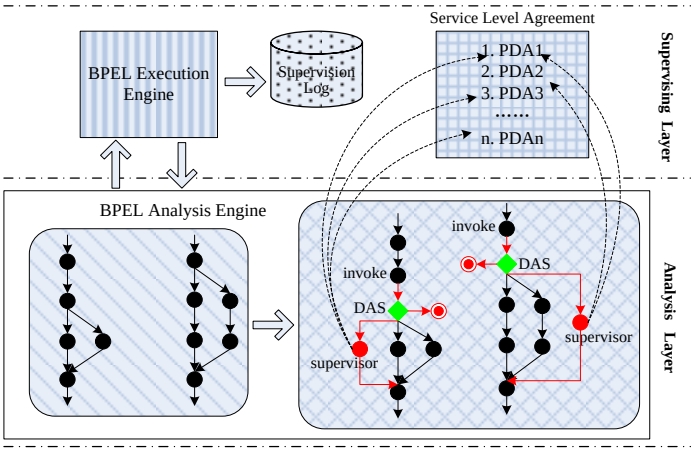

Fig. 5. Privacy agreement-oriented supervising framework

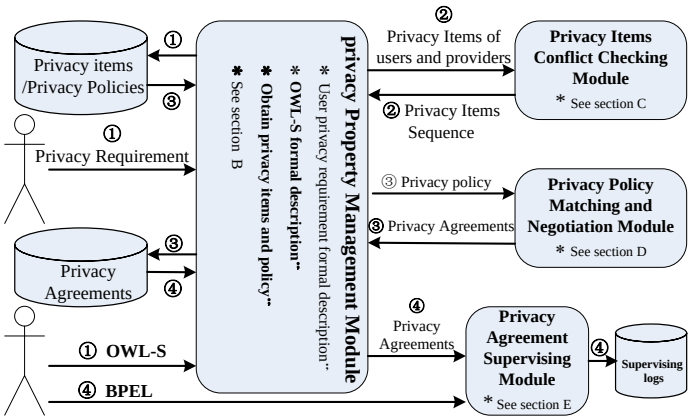

Fig. 6. Privacy protection implementation platform

IV. Privacy Protection Implementation Platform

Privacy protection implementation platform is made up of four modules, including privacy property management module, privacy items conflict checking module, privacy policy matching and negotiation module and privacy agreement supervising module. As showed in Fig. 6.

Privacy Property Management Module: Resolving user privacy requirement and service description document OWL-S, to obtain the privacy property, namely privacy items and corresponding privacy policy assertion, and then describing them with ontology and description logic.

Privacy Items Conflict Checking Module: Detecting the conflict of both user and service provider privacy item, and then detecting the conflict between user and service provider privacy item, namely check the consistency with Tableau algorithm.

Privacy Policy Matching and Negotiation Module: According to matching rules, matching module match the corresponding privacy policies of privacy item between user and service provider. If mismatch is triggered, matching module will invoke the negotiation module to obtain the privacy agreement satisfying user and service provider.

Privacy Agreement Supervising Module: Firstly, with dynamic description logic we verify whether outsourcing service is authorized to obtain the user privacy attribute, to prevent unauthorized outsourcing service from obtaining user privacy information. Secondly, we supervise authorized outsourcing service according to the privacy agreement, to assure the privacy agreement is kept.

Execution flow of privacy protection implementation platform as follows:

Step 1: Privacy property management module obtain the user privacy requirement and service description document from user and service provider, and resolve them to get privacy items and privacy policies. Then save them to privacy items/privacy policies database.

Step 2: Privacy conflict checking module gain the privacy items through privacy property management module, check the consistency between user and service provider privacy items. And save the privacy items sequence to privacy items/privacy policies database.

Step 3: privacy property management module obtain corresponding privacy policy according to privacy items sequence. Privacy policy negotiation module matches the corresponding privacy policy of every privacy item. When mismatch is found, Negotiation module will trigger negotiation mechanism to gain the privacy agreements, and save them to privacy agreement database.

Step 4: When service composer provide the service for user, privacy agreement supervising module obtain the service composition execution process BPEL, instrument privacy agreement assertion into BPEL according to privacy agreement and BPEL context semantic. So as to supervise the execution process, save the supervising logs into database.

In this paper, we develop a prototype system CloPP and run a case to verify our method feasibility and practicability. The prototype system is developed with java; the graphic interface is designed with MyEclipse Swing Matisse and the privacy data is stored with MySql. We build privacy conflict detection ontology with ontology editing tool Protégé based on java language, which is developed by Stanford University. And then the knowledge base is reasoned with reasoner Pellet to obtain the privacy items sequence. Pellet was developed by Mind Swap lab in University of Maryland. Pellet version number used in this experiment is V.2.3.0. The system negotiates the corresponding privacy policies of the privacy item to gain the privacy agreement satisfying user and service provider. In the end, the BPEL is supervised by privacy agreement supervising module, to discover the service provider violating the privacy agreement.

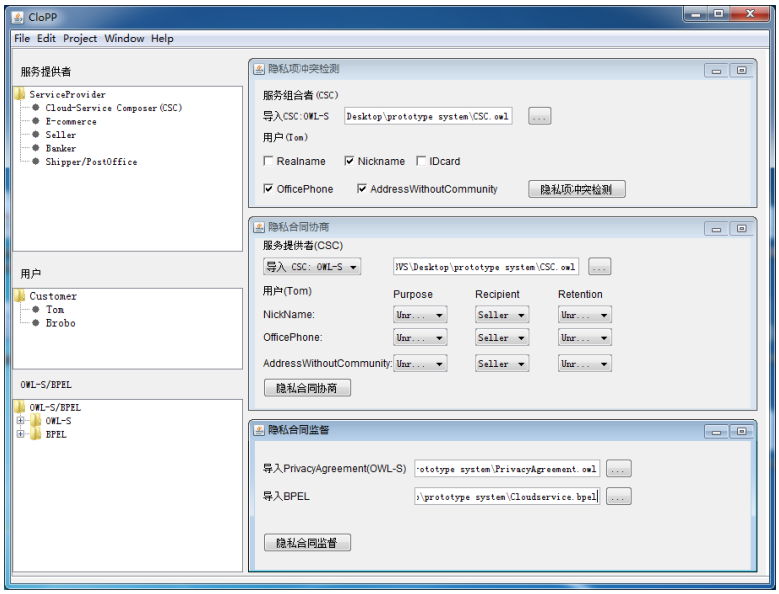

Fig. 7. Operation view of the prototype system

Fig. 7. is operation view of prototype system. The user and service partners are listed on the figure top-left. The cloud services description document and BPEL are listed on the figure bottom-left. The window of privacy items conflict detection, privacy agreement negotiation and supervising are listed on the figure right. Through analyze the privacy requirement, service description document OWL-S and business process execution language BPEL, we can gain the privacy item sequence, privacy agreement and service partners violating privacy agreement.

V. Conclusions and Future Work

This paper proposes a requirement-oriented privacy protection methods in cloud computing. According to the analysis of privacy protection theory architecture, we design the privacy protection implementation platform. Future works is to implement the theory frame of every module.

Acknowledgment

This work is supported in part by the National Science Foundation of China under grants (No.61272083, No.61373137, No.61373017), Major Program of Jiangsu Higher Education Institutions under grant No.14KJA520002, Six Industries Talent Peaks Plan of Jiangsu under grant No.2013-DZXX-014 and Jiangsu Qinglan Project.

References

[1] V. Rajaraman, "Cloud computing," *Resonance*, vol.19, pp. 242-258, March. 2014.

[2] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. H. Katz, A. Konwinski, *et al.*, "Above the clouds: A berkeley view of cloud computing," University of California, Berkeley, Tech. Rep. UCB-EECS-2009-28, Feb. 2009.

[3] K. Baghai, "Privacy as a human right: A sociological theory," *Sociology*, vol. 46, pp. 951-965, 2012.

[4] I. Goldberg, D. Wagner, and E. Brewer, "Privacy-enhancing technologies for the internet," in *Proc. 42nd IEEE International Computer Conf.*, New Jersey, 1997, pp. 103-109.

[5] F. Xiao, Z. Huang, Z. Cao, J. Hu, and L, Liu. "Modeling cost-aware Web services composition using PTCCS," in *Proc International Conf. on Web Service*, New Jersey, 2009, pp.461-468.

[6] G. Yee and L. Korba, "Privacy policy compliance for Web services," in *Proc. 2004 IEEE International Conf on Web Services*, New Jersey, 2004, pp. 158-165.

[7] J. Zhang, C. K. Chang, L. J. Zhang, and P. C. K. Hung, "Toward a service-oriented development through a case study," *IEEE Transaction on Systems, Man, Cybernetics, Part A*, vol. 37, pp. 955-969, 2007.

[8] S. Pearson, "Taking account of privacy when designing cloud computing services," HP Labs Technical Report, HPL-2009-54, London, HP Labs, 2009.

[9] S. Pearson and A. Charlesworth, "Accountability as a way forward for privacy protection in the cloud," HP Labs Technical Report, HPL-2009-178, London, HP Labs, 2009.

[10] J. J. Lu, Z. Q. Huang, and C. Ke, "Verification of behavior-aware privacy requirements in web services composition," *Journal of Software*, vol. 9, pp. 944-951, 2014.

[11] N. Guermouche, S. Benbernou, E. Coquery and M. S. Hacid, "Privacy-aware Web service protocol replaceability," in *Proc. of the International Conf. on Web Services*, New Jersey, 2007, pp. 1048-1055.

[12] K. Mokhtari, S. Benbernou, M. Hacid, E. Coquery, and F. Leymann, "Verification of privacy timed properties in Web service protocols," in *Proc. IEEE international Conf. on Services Computing*, New Jersey, 2008, pp. 593-594.

[13] L. Y. Liu, H. B. Zhu, Z. Q. Huang, and D. Q. Xie, "Minimal privacy authorization in web services collaboration," *Computer Standards & Interfaces*, vol. 33, pp. 332-343, 2011.

[14] C. Ke, Z. Q. Huang, W. W. Li, Y. Sun, and F. X. Xiao, "Service outsourcing character oriented privacy conflict detection method in cloud computing," *Journal of Applied Mathematics*, vol. 2014, May 2014.

[15] Z. Q. Wei, M. J. Kang, *et al.*, "Research on Privacy-Protection Policy for Pervasive Computing," *Chinese Journal of Computers*, vol. 33, pp. 28-138, 2010.

[16] C. Ke, Z. Q. Huang, and M. Tang, "Supporting negotiation mechanism privacy authority method in cloud computing," *Knowledge-Based Systems*, vol. 51, pp. 48–59, Oct. 2013.

[17] S. Tbahriti, C. Ghedira, B. Medjahed, and M. Mrissa, "Privacy-enhanced web service composition," *IEEE Transactions on Services Computing*, vol. 7, pp. 210-222, June 2014.

[18] Y. Zhang and D. G. Fen, "Parsimonious semantic trust negotiation," *Chinese Journal of Computers*, vol. 32, pp. 1989-2003, 2009.

[19] J. Kolter, M. Netter, and G. Pernul, "Visualizing past personal data disclosures," in *Proc. 17th Int. Conf. on Availability, Reliability and Security*, New Jersey, 2010, pp. 131-139.

[20] L. Y. Liu, H. B. Zhu, and Z. Q. Huang, "Analysis of the minimal privacy disclosure for web services collaborations with role mechanisms," *Expert Syst. Appl*, vol. 38, pp. 4540-4549, 2011.

[21] T. Yu, Y. Zhang, and K. J. Lin, "Modeling and measuring privacy risks in QoS web services," in *Proc. 8th IEEE International Conf on E-Commerce Technology/3th IEEE International Conf on Enterprise Computing, E-Commerce and E-Services*, New Jersey, 2006, pp. 4.

[22] D. Svantesson and Roger, "Clarke privacy and consumer risks in cloud computing," *Computer Law and Security Review*, vol. 26, pp. 391-397, 2010.

Changbo Ke was born in Shanxi Province, China, in 1984. He received the B.S. and M.S. degree from the Kunming University of Science and Technology of China, Kunming, in 2008 and 2010 respectively, and received the PH.D degree from the Nanjing University of Aeronautics and Astronautics, Nanjing, in 2014. Now is lecturer of Nanjing University of Posts and Telecommunications. Major research interests include security and privacy of information system, cloud computing and ontology-based software engineering.

Ruchuan Wang was born in Anhui Province, China. He researched on graphic processing at University of Bremen and program design theory Ludwig Maximilian Muenchen Unitversitaet from 1984 to 1992. He is a professor and tutor of Ph.D. candidate in Nanjing University of Posts and Telecommunications since 1992. Major

research interests include wireless sensor networks, information security.

Fu Xiao was born in Hunan Province, China, in 1980. He received the B.S. degree and the Ph.D. degree from the Nanjing University of Science and Technology of China in 2002 and 2007 respectively. Now he is professor of Nanjing University of Posts and Telecommunications. Major research interests include wireless sensor networks.

Zhiqiu Huang was born in Jiangsu Province, China, in 1965. He received the B.S. degree and the M.S. degree from the National University of Defense Technology of China, received the Ph.D. degree from Nanjing University of Aeronautics and Astronautics of China. Now he is a professor and tutor of Ph.D. candidate in Nanjing University of Aeronautics and Astronautics of China. Major research interests include formal method, software engineering, cloud computing.