XACML Policy Evaluation With Dynamic Context Handling (Extended Abstract)

Nariman Ammar, Zaki Malik
Department of Computer Science
Wayne State University
Detroit, Michigan, USA
{nammar,zaki}@wayne.edu

Abdelmounaam Rezgui
Department of Computer Science and Engineering
New Mexico Tech
Socorro, New Mexico, USA
rezgui@cs.nmt.edu

Elisa Bertino
Department of Computer Science
Purdue University
West Lafayette, IN, 47907
bertino@cerias.purdue.edu

I. Introduction

In collaborative service-based health data sharing environments, participating services may host different sets of data about the same individuals, identified by some common properties. Each organization in such environments (e.g., testing labs, research institutes, etc.) manages it's data access and usage through a specialized Web service end point through which users can submit queries. For instance, the Bio2RDF project incorporates data from several ontologies (e.g., NCBI-Gene,PharmKGB, DrugBank, CDT, and GeneCDS) using RDF (as a universal healthcare exchange language). Each repository defines an ontology (in OWL format) of all the concepts that can be searched for in a user's query. OWL defines *classes* as a generic concept of individuals (e.g., Patient) and *data type properties* to link individuals of those classes to their data values (e.g., hasDisease). PharmGKB repository, for example, can be identified by the set of data type properties (drug, disease, gene, etc.) defined on the set of classes (Dosage, Drug, DrugGeneAssociations, etc.). To query instances in those repositories, a user submit different queries through SPARQL endpoints dedicated for each service. In each query, he can can ask for different data type properties by which those instances are identified. Several privacy issues may arise in such environments. First, transforming such data sets into semantic data makes data linkage easier and machine processable. Second, *dynamic composition* of different data items (retrieved through participating Web services) may be misused by adversaries to reveal sensitive information, which was not deemed as such by the data owner at the time of data collection. For instance, a Clinical service may store the data items *(Age, Gender,..., patientStatus)*, a Genomic service may store *(Age, Gender, ...,Gene, SNP)*), and a Demographic service may store *(Age, Gender, ..., Employer, Address)*. Atomically, these data items may not reveal personally identifiable information, but linking those items may lead to unintended breach of privacy. Thus, the patient's consent that is statically defined in a privacy policy may not be enough for data disclosure. These issues call for a privacy management solution that is, *dynamic*, *context-sensitive*, and *semantic-based*. Several researchers have provided enhancements to the performance of XACML PEP, such as efficiency, scalability, and adaptation [1], [2], [3], [4], [5], but few works provided enhancements to the PEP accuracy by enhancing the context handler. This work focus on enhancing the XACML PEP component accuracy by adding dynamic context handling. Few researchers have looked into dynamic policy evaluation as opposed to static policies [6]. Others have proposed context-aware privacy management systems [7], [8], [9], [10]. Our work is different from previous approaches for dynamic privacy management in that it does not dynamically update the original policy definitions, but implicitly incorporate context into rule evaluation. They also regulate rather than prevent the data access.

II. Dynamic Privacy Management

We implemented a dynamic privacy management framework on the top of the XACML reference architecture. According to a standard XACML engine, upon query submission, the *PEP* wraps the query into an XACML request and forwards it to the *PDP*, which communicates with the *PIP* to fetch the required attributes. In our system, the *PIP* communicates with a *Semantic Handler (SH)*, which looks up the required attributes in the service's repository. The PDP then uses the attribute values to evaluate the request. If a *permit* decision is returned, the PDP consults the semantic handler (via the PIP) for previously recorded context of the matching instances. The PDP then wraps the retrieved context set as an XACML obligation element and sends it over to the PEP together with the context handling obligation logic to be performed. The PEP uses the obligation to perform further check by communicating with the *Semantic Handler* (Fig. 1, 1). The Semantic handler passes the set of instances I_j that match the query together with the query Q_j to the *Context Handler* (Fig. 1, 2). The context handler consists of two sub components: the *Classifier*, which dynamically classifies a query as being potentially $malicious$ or $legitimate$, and the *Sensitive Data Detector*, which dynamically determines the subset of data type properties in a query that could potentially be sensitive. The PEP uses the context $CTXT$ to update the context block of each instance i_k in the set I_j that matches Q_j via the *Semantic Handler* (Fig. 1, 4). The PEP then uses $CTXT$ to make the final decision through the *Dynamic Rule Evaluator* (Fig. 1, 5). Finally, the PEP sends the final response back to the user. For mathematical details and further description, we refer the reader to the extended work at [11].

III. Evaluation

We conducted two case studies on two systems using two XACML engines and two performance testing frameworks (Table I). Our results indicate that on average, the percentage of the *permited* decisions through a standard PEP is always higher than that of PEP with our context handler incorporated

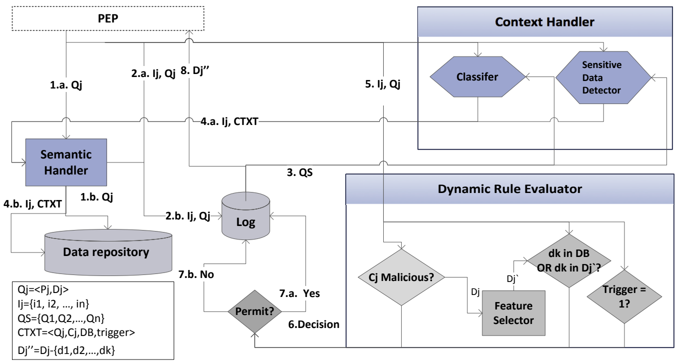

Fig. 1. Approach for dynamic privacy management.

TABLE I. EXPERIMENTAL SETUP.

Test Plan	**Java metrics**	**Jmeter**
Subject system	E-HIP HPMS	Bio2RDF
Environment	2GHz Intel Dual-Core i7 Mac, 8GB RAM, 64bit OS X 10.8.	2.8GHz Intel Quad-Core CPU, 8GB RAM, 32bit Windows Server 2007
XACML Engine	SunXACML	WSO2 IS
Threads (Users)	1, 2, 4, 8, 16, 32	100
Iterations	100	5, 10, 20
Policies	19 policies	10
Requests	3	10, 50, 100
Properties	3, 6, 11	14, 28

(Fig. 2). We observed that the cost incurred by the context handler (Fig. 3) in general and the semantic handler in particular (Fig. 4) does not significantly affect the throughput and evaluation time of a standard XACML-based framework. As for scalability, we found that the throughput increased until it reached more than 50 hits/sec on average with 50 active users (Fig. 5).

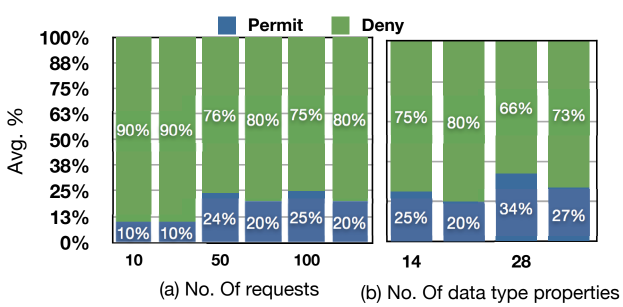

Fig. 2. PEP Accuracy for PEP (left) vs. PEP+CH (right)

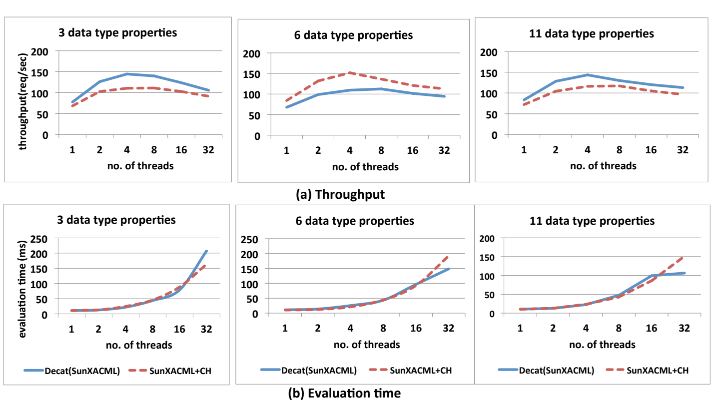

Fig. 3. PEP performance for different number of data type properties

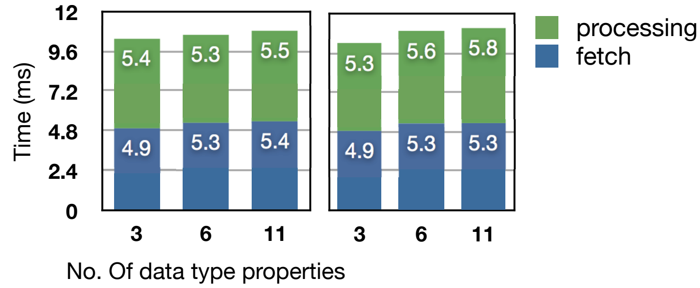

Fig. 4. Attribute fetch time in Decat et al (left) compared to ours.

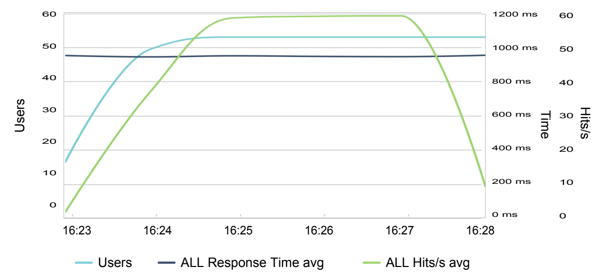

Fig. 5. Scalability

IV. CONCLUSION

We provided an XACML-based implementation of a semantic-based privacy management framework that incorporates context into dynamic rule evaluation and decision enforcement. Our evaluation results are promising, and we believe that future enhancements on the current implementation can provide a foundation for modern health records infrastructures and inspire collaborative data sharing.

REFERENCES

[1] M. Decat, B. Lagaisse, and W. Joosen, "Middleware for efficient and confidentiality-aware federation of access control policies," *Journal of Internet Services and Applications*, 2014.

[2] A. Mourad and H. Jebbaoui, "Towards efficient evaluation of xacml policies," in *PST*, 2014.

[3] A. X. Liu, F. Chen, J. Hwang, and T. Xie, "Designing fast and scalable xacml policy evaluation engines," *Computers, IEEE Transactions on*, 2011.

[4] A. Erradi, P. Maheshwari, and V. Tosic, "Policy-driven middleware for self-adaptation of web services compositions," in *Middleware*, 2006.

[5] R. Laborde, B. Kabbani, F. Barrère, and A. Benzekri, "An adaptive xacmlv3 policy enforcement point," in *COMPSAC Workshop*, 2014.

[6] B. Kabbani, R. Laborde, F. Barrère, and A. Benzekri, "Specification and enforcement of dynamic authorization policies oriented by situations," in *NTMS*, 2014.

[7] R. Ferrini and E. Bertino, "Supporting rbac with xacml+ owl," in *Symposium on Access control models and technologies*, 2009.

[8] T. Grandison, S. R. Ganta, U. Braun, J. Kaufman *et al.*, "Protecting privacy while sharing medical data between regional healthcare entities," *Studies in health technology and informatics*, 2007.

[9] R. Agrawal, C. Johnson *et al.*, "Securing electronic health records without impeding the flow of information," *International journal of medical informatics*, 2007.

[10] K. LeFevre, R. Agrawal, V. Ercegovac, R. Ramakrishnan, Y. Xu, and D. DeWitt, "Limiting disclosure in hippocratic databases," in *VLDB*, 2004.

[11] N. Ammar, Z. Malik, E. Bertino, and A. Rezgui, "Xacml policy evaluation with dynamic context handling," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 27, no. 9, 2015.