A Decentralized Context-aware Cross-domain Authorization Scheme for Pervasive Computing

Bin Song[1, 2]
[1]Henan University of Science and Technology
[2] Henan International Joint Laboratory of Cyberspace Security Applications
263, Kaiyuan Avenue, Luoyang, China
+86-18625560900, 471023
songbin@haust.edu.cn

Min Gao[3]
[3]Luoyang Institute of Science and Technology
90, Wangcheng Avenue, Luoyang, China
+86-17737989187, 471023
litgaomin@qq.com

ABSTRACT
Context-aware access control is one of the most frequently used methods for making authorization decisions in pervasive computing environments. To the best of our knowledge, most previous relevant researches resorted to centralized schemes to preserve all the contextual information. As a result, they neglected actual circumstances where the sources of contextual information are generally decentralized among multiple management domains with different security policies. For the sake of cross-domain access control, in this paper we present a distributed context-aware authorization mechanism for pervasive computing applications. With the help of logical language theory, we demonstrate how the proposed model can attain the goal of effective reliability assurance and privacy protection by way of constructing a decision tree dynamically, according to the current contextual information.

Keywords
context-awareness; cross-domain access control; pervasive computing; security; privacy

1. INTRODUCTION
The emerging pervasive computing environments require security services that are easily adaptable to changing users or environmental contexts [1]. Context-awareness is actually a central aspect of pervasive computing applications. Various definitions for context and context-awareness have been proposed in the literature. Broadly speaking, context refers to any information that can be used to characterize the situation of an entity, and a system is context-aware if it uses context to provide relevant information and/or services to the user, where relevancy depends on the user's task [2]. For example, a mobile phone should always vibrate and never ring in a concert hall, if it somehow has knowledge about its current location and the activity going on.

ICNCC 2019, December 13–15, 2019, Luoyang, China

ACM ISBN 978-1-4503-7702-7/19/12…$15.00
https://doi.org/10.1145/3375998.3376026

Over the last few years, there has already been a tremendous growth in context-aware applications, including a hospital system [3], a shopping assistance [4], an exhibition guide [5], a tour guide [6], an active campus [7], a conference assistant [8], an office assistant [9], a smart classroom [10], and so on. In the meantime, there has also been an increasing concern towards security and privacy requirements in the realization of context-aware access control for pervasive computing [11]. In fact, the major goal of context-aware access control is exactly to allow users to define flexible security policies and provide a non-intrusive way to access to important physical or information resources for users. In order to decrease the management complexity of context-based pervasive computing applications, the concept of domain is proposed from a realistic point of view that security policies do not have to be specified individually for each principal but in a set for a collection of principal part of a domain. Security policies are essentially a set of rules for authorization, access control, and trust in a certain domain [12].

Although quite a few context-aware authorization mechanisms for pervasive computing applications have already been proposed in recent years, the existing models always resort to centralized schemes to preserve all the contextual information, and neglected actual circumstances where the sources of contextual information are usually decentralized among multiple management domains with different security policies. It is quite obvious that the traditional identity-based authorization approach does not work well in pervasive computing environments because the mobile users are likely to move across domains constantly. In order to solve the cross-domain access control problem effectively, in this paper, we present a distributed context-aware authorization mechanism by means of logical language theory, which is flexible enough to supply users with the ability to define their security policies at their discretion, thereby making authorization decisions dynamically, according to the current contextual information.

The rest of this paper is organized as follows. In the next section, we review some related work on context-aware access control and also point out their disadvantages. In section 3, we first introduce logical language theory briefly. Based on the theory, we then present context-aware cross-domain access control model and describe a detailed procedure to show how the propose model can be used to make authorization decisions effectively. Finally, we conclude this paper in section 4 where we also discuss our future work.

2. RELATED WORK

To the best of our knowledge, there are already some previous related researches that focus on context-aware access control in pervasive computing environments. These existing methods, however, always employ a centralized solution to gather contextual information and to evaluate security policies in making authorization decisions on behalf of a resource owner.

For example, on the foundation of traditional role-based access control, generalized role-based access control paradigm [13] incorporates the concept of environment roles to capture environmental information, which enables us to define contextual constraints on environmental variables, thus offering more expressiveness and making it suitable for context-aware authorization schemes [14]. However, since the paradigm adopts a centralized server to maintain the potentially large amounts of contextual information, it may not be feasible in practice.

Open architecture for secure interworking services is another proposal for a more general design of a role-based access control system, which allows arbitrary environmental constraints to be used in rules for enabling role activations and for maintaining role membership. Furthermore, role activation and role maintenance rules are definite Horn clauses and thus can represent environmental conditions as contextual predicates. Similarly, the architecture also makes use of a centralized pattern to collect contextual information and fails to address privacy protection effectively.

3. PRELIMINARIES

In this section, we first introduce basic concepts of logical language in a nutshell. With the help of logical language theory, we then present our context-aware cross-domain access control model for pervasive computing applications. Finally, we demonstrate how the proposed model can be used to attain the goal of effective reliability assurance and privacy protection through the process of making access control decisions.

As is known to all, logical language facilitates authorization decision by defining a set of rules and facts that change the behavior of the systems dynamically. Therefore, it is often applied in context-aware access control for pervasive computing application where authorization policies are represented as logical expressions composed of a set of Horn clauses.

In general, a Horn clause is a clause $A \leftarrow B_1, B_2, \ldots\ldots, B_n$ with at most one positive literal, where the positive literal A is called the head and the negative literals B_i are called the body. A unit positive Horn clause $A \leftarrow$ is called a fact, and a Horn clause with no positive literals $\leftarrow B_1, B_2, \ldots\ldots, B_n$ is called a goal clause. A Horn clause with one positive literal and one or more negative literals is called a program clause [15].

For example, health care services demonstrate a rich set of context-based security and resource access requirements where the complex authorization policies can be described with a logic program. Considering the situations in a patient monitoring activity case, suppose a person is allowed to access a patient's medical records only when she possesses a nurse role membership, and at that time she is in the patient's ward. Moreover, a patient's medical records should only be accessible via the nurse's personal smart devices such as iphone when the nurse is present close to the devices. For these requirements, we can express the medical records authorization policies that are dependent on the location-based contextual information of a person as a set of Horn clauses in Prolog.

As a matter of fact, Prolog is the first logic programming language and extensive implementation efforts have already transformed it into a practical tool for software development. The notation of Prolog is different from the mathematical notation that we have been using: (1) variables begin with upper-case letters, (2) predicates and constants begin with lower-case letters, and (3) the symbol :- is used for ← [16]. Figure 1 demonstrates the patient monitoring activity case with a set of Horn clauses in Prolog that define authorization policies to offer access to the medical records.

access(Person, medical records) :- role(Person, nurse),
in(Person, ward). *(a)*

in(Person, ward) :- possess(Person, Device), in(Device, ward). *(b)*

role(alice, nurse). *(c)*

possess(alice, iphone). *(d)*

in(iphone, ward). *(e)*

Figure 1. The medical records authorization policies.

Here, we cannot derive the left literal unless the literals on the right side of the Horn clauses are met at the same time. Therefore, in order to obtain authorization result, it is very necessary to verify whether a decision tree for the request *:- access(Person, medical records)* could be constructed or not with a given set of rules and facts. To illustrate this, we take a user *alice* who makes a request to access medical records for example.

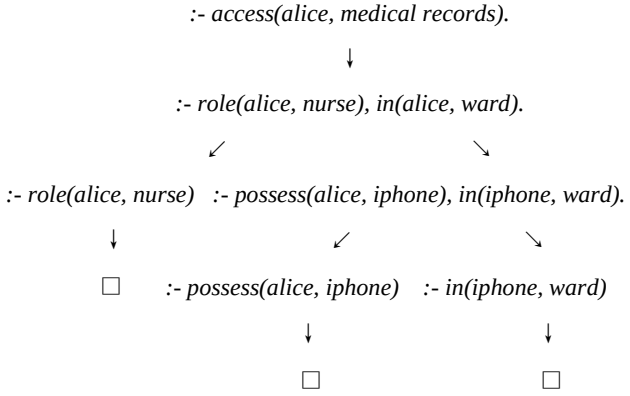

Figure 2. The construction process of a decision tree.

Based on the above rules and facts formulated in Figure 1, we can construct the decision tree by replacing variable Person in the rules with constant alice. Figure 2 shows the construction process where the non-leaf nodes of the decision tree represent the rules, and the leaf nodes represent the facts, respectively.

4. DECENTRALIZED CONTEXT-AWARE CROSS-DOMAIN AUTHORIZATION SCHEME

In this section, with the help of logical language theory, we present our decentralized context-aware cross-domain access control model for pervasive computing applications. Moreover, we demonstrate how the proposed model can be used to attain the goal of effective reliability assurance and privacy protection through the process of making authorization decisions.

4.1 Decentralized Context-Aware Cross-Domain Authorization Model

Based on logical language theory, in order to remedy the deficiency of the existing authorization mechanisms, we propose a context-aware cross-domain access control model for pervasive computing applications where authorization procedure generally involves multiple authorization servers in different management domains, which employ peer-to-peer method to construct a decision tree for an access request in a cooperative way. Therefore, an authorization server usually has to make a remote request to another authorization server if there does not exist the required information in its local policy repository.

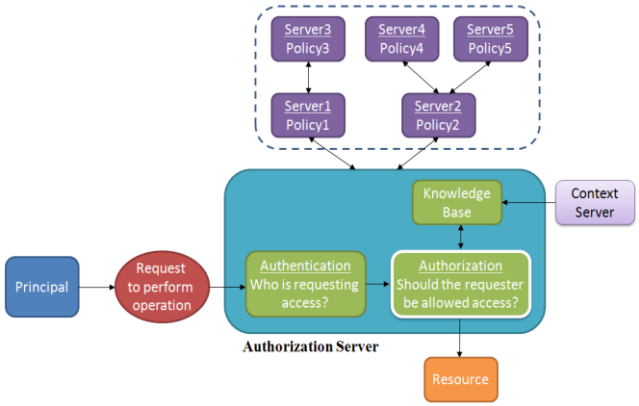

Figure 3. Context-aware cross-domain access control model.

Figure 3 shows the process of how the context-aware cross-domain access control model makes an authorization decision in a distributed environment. The proposed model consists of a set of servers, each of which only maintains partial knowledge about security policies and contextual facts. For every authorization server, the context provider offers current contextual information and updates facts in the policy repository dynamically. The policy repository preserves not only current contextual information but also access control rules and facts. As soon as the reference monitor receives a principal's request to perform operation on resource, it takes the initiative in trying to construct a decision tree that derives the fact in the request by retrieving information in the policy repository.

4.2 Decentralized Context-Aware Cross-Domain Authorization Implementation

Based on the proposed model, in order to achieve contextual information sharing among multiple management domains with different security policies, the authorization implementation should consider security and privacy requirements in the realization of context-aware cross-domain access control for pervasive computing applications. These requirements can be considered an opportunity to enhance the available security techniques. The enhancements may include less intrusive access control methods where the use of context-information requires trust in the context-source to assure the reliability of the contextual information used in the access control policies. The privacy requirement arises mainly due to the highly private sensitive nature of user contextual information, and the implicit gathering and combining of the contextual information in a pervasive service provisioning environment.

In view of the above-mentioned problems, we put forward the following pertinent solution. Above all, every management domain should formulate reliability policies that specify whether to believe in information from other domains with regard to the correctness of that information. It is important for each domain to choose reliable sources of information to derive correct contextual information. In addition, every management domain should formulate privacy policies to protect information in that domain. Therefore, if a principal wants to access the requested information, it must satisfy the privacy policies of the information provider.

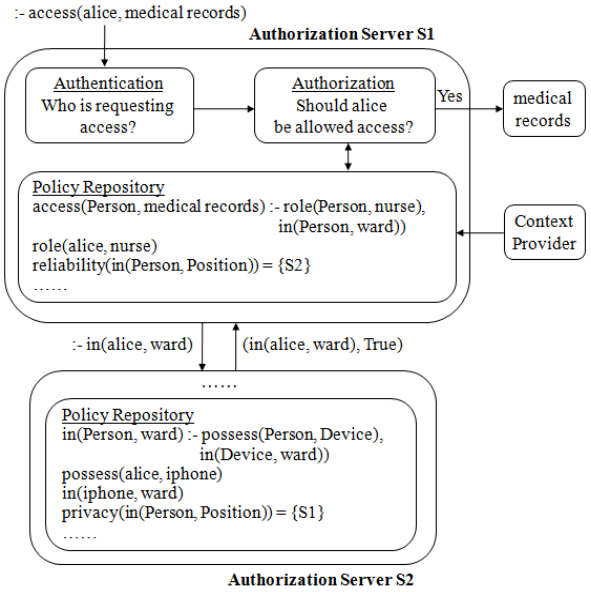

Figure 4. Example of reliability assurance and privacy protection.

With the help of the previous instance concerning health care services, Figure 4 demonstrates how the proposed authorization mechanism can be implemented to attain the goal of effective reliability assurance and privacy protection by way of constructing a decision tree dynamically, according to the current contextual information.

Suppose that authorization server *S1* receives an access request *:- access(alice, medical records)* that inquires whether or not *alice* is permitted to access medical records. According to server *S1*'s authorization policy in its policy repository that requires a requester's nurse membership and position information, *S1* has to make a request *:- in(alice, ward)* to another reference monitor maintained by authorization server *S2*. The reason why *S1* prefers *S2* is that *S2* satisfies *S1*'s reliability policies for request of the predicate *in(Person, Position)* (i.e., *reliability(in(Person, Position)) = {S2}*). In reality, every authorization server will reach a decision on the issue that which server a request should be sent by means of making inquires about its reliability policies. At the same time, since *S1* meets *S2*'s requirements of privacy policies for request of the predicate *in(Person, Position)* as defined in *S2*'s policy repository (i.e. *privacy(in(Person, Position)) = {S1}*), *S2* can process the request from *S1* successfully, and then draw the conclusion that *alice* is indeed in the ward by reasoning, according to the two facts *possess(alice, iphone)* and *in(iphone, ward)*. Furthermore, since *S1* trusts *S2*'s statement about a person's position information in the light of *S1*'s reliability policies, *S2* only needs to return a result that contains a single root node that states that *in(alice, ward)* is true. Obviously, the medical records access request will be authorized jointly by *S1* and *S2*. Generally speaking, *S2* could return a decision tree that contains

multiple authorization servers with different reliability and privacy policies. Here, we should pay attention to every principal who involves in constructing a decision tree should execute its privacy policies by way of encrypting a request result with a receiver principal's public key. Moreover, every request result should be signed with a sender principal's public key so that a principal who receives a request result that contains sub-results produced by multiple principals can check its reliability.

5. CONCLUSION AND FUTURE WORK

In this paper, we first analyzed the problems of the existing methods in the realization of context-aware cross-domain access control in pervasive computing environments, and then proposed an effective authorization mechanism with the help of logical language theory. Finally, we illuminated that the proposed method is practical as well as flexible by use of specific examples. Compared with previous access control models that adopted centralized solution to preserve all the contextual information, the distinguishing feature of the proposed method is reliability assurance and privacy protection through the process of the decentralized construction and evaluation of authorization decisions according to the current contextual information, involving multiple principals from different management domains.

However, the proposed authorization mechanism in this paper is still in a very primitive stage and further research is very necessary in the future. Moreover, our prospective technical issues will also involve the implementation and application of the context-aware cross-domain access control model that we have developed for pervasive computing environments.

6. ACKNOWLEDGMENTS

The work was sponsored by National Natural Science Foundation of China Grant No.61972133 and 61772174, Plan for Scientific Innovation Talent of Henan Province Grant No.174200510011, Key R&D and Promotion Special Projects (Tackling Hard-nut Problems in Science and Technology) of Henan Provincial Science and Technology Development Plan under Grant No. 192102210130 and the Key Scientific Research Projects Plan of Henan Provincial Higher Education Institutions under Grant No. 19B520008. The authors would like to thank the anonymous reviewers and the editor for the very instructive suggestions.

7. REFERENCES

[1] Dey, Anind K. 2018. Context-Aware Computing. *Ubiquitous computing fundamentals*. Chapman and Hall/CRC, 335-366.

[2] Snidaro, L., Garc á, Jes ús, and Llinas, J. 2015. Context-based information fusion: a survey and discussion. *Information Fusion, 25, 16-31*. DOI= http://dx.doi.org/10.1016/j.inffus.2015.01.002.

[3] Bonte, P., Ongenae, F., Schaballie, J., Vancroonenburg, W., Vankeirsbilck, B., and Turck, F. D. 2017. Context-Aware and Self-learning Dynamic Transport Scheduling in Hospitals. *European Semantic Web Conference, Springer, Cham.* DOI= http://dx.doi.org/10.1007/978-3-319-70407-4_31.

[4] Orciuoli, F., and Parente, M. 2016. An ontology-driven context-aware recommender system for indoor shopping based on cellular automata. *Journal of Ambient Intelligence and Humanized Computing*, 937-955. DOI= http://dx.doi.org/10.1007/s12652-016-0411-2.

[5] Vahdat-Nejad, H., Navabi, M. S., and Khosravi-Mahmouei, H. 2018. A context-aware museum-guide system based on cloud computing. *International Journal of Cloud Applications and Computing*, 8(4), 1-19. DOI= http://dx.doi.org/10.4018/IJCAC.2018100101.

[6] Meehan, K., Lunney, T., Curran, K., and Mccaughey, A. 2016. Aggregating social media data with temporal and environmental context for recommendation in a mobile tour guide system. *Journal of Hospitality and Tourism Technology*, 7(3), 281-299. DOI= http://dx.doi.org/10.1108/JHTT-10-2014-0064.

[7] Dou, E., Eklund, P. W., and Gretzel, U. 2016. Location privacy acceptance: attitudes to transport-based location-aware mobile applications on University campus. *The 27th Australasian Conference on Information Systems*.

[8] Koay, K. L., Syrdal, D., Bormann, R., Saunders, J., Walters, M. L., and Dautenhahn, K. 2017. Initial design, implementation and technical evaluation of a context-aware proxemics planner for a social robot. DOI= http://dx.doi.org/10.1007/978-3-319-70022-9_2.

[9] Bradley, N. C., Fritz T., and Holmes R. 2018. Context-aware conversational developer assistants. In *Proceedings of the 40th International Conference on Software Engineering*. ACM. DOI= http://dx.doi.org/10.1145/3180155.3180238.

[10] Huang, L. S., Su J. Y., and Pao T. L. 2019. A Context Aware Smart Classroom Architecture for Smart Campuses. *Applied Sciences*. DOI= http://dx.doi.org/10.3390/app9091837.

[11] Hosseinzadeh, S., Virtanen, S., Natalia D áz-Rodr guez, and Lilius, J. 2016. A semantic security framework and context-aware role-based access control ontology for smart spaces. In *Proceedings of the International Workshop on Semantic Big Data*. ACM. DOI= http://dx.doi.org/10.1145/2928294.2928300.

[12] Singh, K. K. 2016. Context-aware permission control of hybrid mobile applications.

[13] Iyer, P., and Masoumzadeh A. 2019. Generalized Mining of Relationship-Based Access Control Policies in Evolving Systems. In *Proceedings of the 24th ACM Symposium on Access Control Models and Technologies*. ACM. DOI= http://dx.doi.org/10.1145/3322431.3325419.

[14] Arfaoui, A., Cherkaoui, S., Kribeche, A. and Senouci S. M. 2019. Context-Aware Adaptive Authentication and Authorization in Internet of Things. *IEEE International Conference on Communications (ICC)*. IEEE. DOI= http://dx.doi.org/10.1109/ICC.2019.8761830.

[15] Marques-Silva, J., Ignatiev, A., Carlos Menc á, and Rafael Pe ñaloza. 2016. Efficient Reasoning for Inconsistent Horn Formulae. *European Conference on Logics in Artificial Intelligence*. Springer, Cham. DOI= http://dx.doi.org/10.1007/978-3-319-48758-8_22.