Time-constraint Access Control in Pervasive Computing Environments

Jun-qing Li
College of Computer Science, Liaocheng University, Liaocheng, 252059, People's Republic of China
lijunqing@lcu.edu.cn

Quan-ke Pan
College of Computer Science, Liaocheng University, Liaocheng, 252059, People's Republic of China
panquanke@lcu.edu.cn

Sheng-xian Xie
College of Computer Science, Liaocheng University, Liaocheng, 252059, People's Republic of China
xsx@lcu.edu.cn

Yu-ting Wang
College of Computer Science, Liaocheng University, Liaocheng, 252059, People's Republic of China
wangyuting@lcu.edu.cn

ABSTRACT

This paper presents a novel access control model named PC-TRBAC (Time-constraint Role-based access control for Pervasive Computing). We give a detail analysis of the access control characteristics in the pervasive computing environments and the overview of the traditional access control model. In order to adapt to the pervasive computing, we add some time constraints into our new access control. Some new time constraints including active time span, active time length, activation time interval, average active duration and entity dependency duration were proposed. At last, we give the framework of our new access control model, and the component definitions are also proposed.

Key Words: Role-Based Access Control; Pervasive computing; Time-constraint

1. Introduction

With the development of the network technology, we have entered into an age named pervasive computing after the Web stage and P2P stage. Pervasive computing is a new technology which will also named ubiquitous computing, connecting every kind of computing devices such as PDA, PC, Wireless sensor, Mobile phone and other wireless devices in a open and heterogeneous environment [1]. Pervasive computing has gained more and more focus in recent yeas and been improved in many aspects. However, there are lots of problems in pervasive computing environments, most of which fall into five main classifications: attention, complexity, privacy, security, and extensibility [2,3]. Access control infrastructure is the most important security component of pervasive computing environments. How to assign suitable permissions to the needed entities who want to access certain resources in the pervasive computing environments becomes a series problem.

Access control is a very important component of network and information security. The focus of access control is to authorize proper users with minimum but enough privilege to accomplish its work [5]. We can divide current access control model into six categories: DAC (Discretionary access control), MAC (Mandatory access control), RBAC (Role based access control) [6], TBAC (Task-based authorization control) [7,9], UC (Usage control) [8] and EDAC (Enterprise dynamic access control) [10]. RBAC is widely used in present enterprise environments, which can adapt to open and dynamic collaborative environments with controllable managements messages. However, there are lots of directions which should be complemented in RBAC standard model, including dynamic fine-grained features and the time-constraints characteristic.

2. Access Control in Pervasive Computing

In recent years, pervasive computing has got more and more focus and improved our life deeply. With many kinds of smart devices such as PDA, smart gadgets, mobile devices and wireless sensors, out life have become more interesting and intelligent than before. However, one key challenge in pervasive computing applications is managing security and access control. The traditional access control such as RBAC and TBAC has been used widely for many years. However, these access control model can not be transplanted into pervasive computing applications directly, and minor changes must be made to make traditional access control adapt to pervasive computing environments. The main access control characteristics in pervasive computing are as follows [2,3,4]:

(1) Context-sensitive

The user-to-role and role-to-permission mapping processes should be context-sensitive in pervasive computing environments. For example, let us assume a scene that a teacher is teaching in a network classroom, when the teacher permit his students download some files from his hard-disk, in this context, his students will be assigned with some roles like *downloader*. At other time, these students can not access the teacher's hard-disk. Therefore, our access control must resolve how to distinguish the entire context events.

(2) Mobility

There are lots of mobile devices that will attend in the pervasive computing applications, such as mobile phone, PDA, wireless sensor. When these devices move from one site to another, our access control must capture this situation and adjust the condition database to assign suitable roles for the requestors.

(3) Multi-region

A requestor in a pervasive computing environment will access multiple mobile devices at a certain time. For example, a student may use his PDA to download some resources when he requests for printing some documents. The devices may come from different regions with different access control rules, how to map different roles among different regions is the key problem in the access control mechanism of the pervasive computing applications.

3. Overview of Traditional Access Controls

In this section, we describe the most popular access control model as follows.

3.1. RBAC

RBAC is the first access control model which can adapt to flexible and open collaborative environment. In the RBAC model, users are associated with roles, and roles are associated with permissions. As we know, there are lots of users who will access resources in a same collaborative system, and their behaviors are dynamic. The core RBAC model includes five basic elements, i.e. U (users), R (roles), O (objects), OP (Operations) and P (permissions). The core RBAC also has some useful session sets, which is the mapping relationship between some users and some adaptable roles. The components in the core RBAC model as follows:

$Users = \{u_1, u_2, \cdots, u_m\}$, is the set of all the users;

$Roles = \{r_1, r_2, \cdots, r_n\}$, is the set of all the roles;

$Ops = \{op_1, op_2, \cdots, op_k\}$, is the set of all the operations;

$Objects = \{ob_1, ob_2, \cdots, ob_l\}$, is the set of all the objects;

$Perms = 2^{(Ops \times Objects)}$, is the set of all the permissions;

$Sessions = \{s_1, s_2, \cdots, s_p\}$, is the set of all the sessions;

$UA \subseteq Users \times Roles$, is a many-to-many user-to-role assignment relationship;

$PA \subseteq Perms \times Roles$, is a many-to-many permission-to-role assignment relationship;

$assigned\_users : (r : Roles) \rightarrow 2^{Users}$, is an operation which can return the users with some certain roles;

$assigned\_perms : (r : Roles) \rightarrow 2^{Perms}$, is an operation which can return certain permissions be assigned to the given roles;

$assigned\_roles : (u : Users) \rightarrow 2^{Roles}$, is an operation which can return certain roles be assigned to the given users;

$op(p : Perms) \rightarrow Ops$, is an operation which can return the related operation with the given permission;

$ob(p : Perms) \rightarrow Objects$, is an operation which can return the related operation with the given objects;

$user\_sessions(u : Users) \rightarrow 2^{Sessions}$, is an operation which can return the related sessions with the given object users;

$session\_users(s : Sessions) \rightarrow 2^{Users}$, is an operation which can return the related users with the given sessions;

$session\_roles(s:sessions) \rightarrow 2^{Roles}$, is an operation which can return the related roles with the given sessions;

$session\_perms(s:sessions) \rightarrow 2^{Perms}$, is an operation which can return the related permissions with the given sessions;

3.2. EDAC

The Enterprise Dynamic Access Control (EDAC) represents an access control model that adheres to the basic principles of Role-based Access Control standard. The EDAC accommodates complex and scalable access control situations many government and civilian organizations are experiencing when managing resource access. The main components of the EDAC model include Customer Personnel Database (CPD), Customer Object Profile Manager Service (OPMS), Customer Meta-Database (CMD), Customer portal, Customer resources, Customer environmental interfaces, Condition Manager Service (CMS), Rules Engine Service (RES), Repository Service (RS), Administrative Service (AS), Structure Format Service (SFS) and Condition Deprecator Service (CDS).

The most known advantages of the EDAC model mainly fall into three aspects: (1) The EDAC model establishes an effective security policy and accommodates enterprise implementations among regions; (2) The EDAC can evaluate inheritance on every user characteristic and environmental; (3) It is very convenient to configure condition constraints in the EDAC model.

4. Time constraint

4.1. Time definition

We define two time granularity unit as follows:

Definition 1: Time unit

The time unit is used to describe the minimum unit of time. For example, we can describe a time point as year-month-day-hour-minute-second, therefore, second will be the time unit of this time point. We can use a time unit set as follows:

$TU = \{t_1, t_2, \cdots t_k\}$

The popular time unit can be minute, second, hour, day, week, month and year.

Definition 2: Time space

Time space is all the time point that are divided by the same time unit in a certain period of time, the definition is as follows:

$$TS = \bigcup_{i \in N, t \in TU} p_t(i)$$

Here, $p_t(i)$ is the i^{th} time point with the t time unit.

For example, a session start at 2009-01-01 07:30:30, therefore, we can set $p_1(1)$=2009-01-01 07:30:30,$p_2(1)$=2009-01-01 07:30,$p_3(1)$=2009-01-01 07,$p_4(1)$=200809-01,$p_5(1)$=2009-01,$p_6(1)$=2009.

4.2. Time constraint definition

(1) Activation time length

Activation time length constraint was used to make a rule that an operation can only be active during a fixed length of time, e.g. one hour or one minute.

(2) Activation time interval

Activation time interval constraint was used to make a rule that an operation should be re-started after a certain period of time between its two runtime phases.

(3) Entity dependency duration

There are lots of dependencies among many entities. For example, a user can not have the permission to print his phone call list until he has paid his owed phone call fee. Therefore, a user can be assigned with a *printer* role after he has accomplished his *payer* role. There are also other roles dependencies. For example, a user can be assigned with a role *A* after he has gotten his role *B*; however, there are lots of accidental events that make the user waiting for getting the role *A*. After a certain period of time, the user is still waiting for the role *A*. We must make a rule to prohibit this situation. Therefore, we propose a new term named EDD to indicate that if a user waits a role for a given time length, we should ask the user abandon his waiting activity and make a try in another time.

5. The Access Control Framework

Considering the access control characteristics of the pervasive computing applications, we design a new access control named PC-TRBAC.

5.1. Components definition

(1) Enforcer. It represents the external interface for the requestor. The enforcer is responsible for the management and enforcement of the access control decision returned from the evaluator component. The enforcer component consists of three sub-components, i.e. customer meta-database directory, customer portal, environmental interface.

(2) Evaluator. The decision component of our access control whose duty is to collect the ambient-based policies and match them against the requests submitted by the requestor.

(3) Customer meta-database directory (Directory service). The directory service is used to construct a general and uniform data type for different request from different requestors.

(4) Repository Service. The repository service stores the resource access conditions such as time-constraint conditions, role dependency conditions, conflicting conditions.

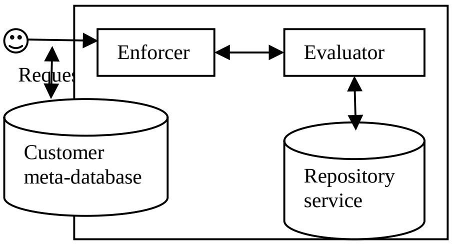

Figure 1. Structure of the PC-TRBAC model

5.2. Model definition

The definition of PC-TRBAC is a five- elements set <*U, R, P, S, C*>. Here, C (Conditions) represents the trust conditions. We also add a new attribute named *t* to U. The definitions of the entities in our system are as follows:

- ● U, R, P, S and C (users, roles, permissions, sessions and conditions respectively);
- ● $PA \subseteq P \times R$, a many-to-many permission-to-role assignment relationship;
- ● $UA \subseteq U \times R$, a many-to-many user-to-role assignment relationship;

The definitions of the operations are as follows:

- ● *users:* ($S \rightarrow U$), a mapping operation which assign a users to a session;
- ● *CR*: ($C \rightarrow R$), a one-to-one time-constraint condition-to-role assignment relationship, a peer with suitable time-constraint condition will be assigned a role with a given permission.
- ● *roles:* $S \rightarrow 2^R$, a mapping operation which assign a set of roles $roles\ (S_i) \subseteq \{ r|(users(S_i), r) \in UA) \wedge (users(S_i).t,\ r) \in CR \}$。 The permissions set that S_i has is $\bigcup_{r \in roles(Si)} \{p|(p,\ r) \in PA\}$.

6. Conclusion

There are lots of security problems in the pervasive computing environments. The differences between the pervasive computing and other computing environments decide that we can not apply the traditional security mechanism into the pervasive computing directly. In this paper, we first give some key access control characteristics of the pervasive computing and some key traditional access control mechanism. Then, we propose a new access control with many novel time constraints which can adapt to the access control characteristics in the pervasive computing excellently. The future work is to make our new access model adapt to the dynamic characteristic in the pervasive computing environments and make it more robust.

7. Acknowledgement

This research was partially supported by grants 2006AA01Z455 from the National High-Tech Research and Development Plan of China, 60874075, 70871065 from National Science Found from China, 2004ZX17 and 2004ZX14 from the Natural Science Foundation of Shandong Province, J08LJ20 from Science Research and Development of Provincial Department of Public Education of Shandong and X061015 from the Science Foundation of Liaocheng University.

References

[1] Weiser M. “Some Computer Science Issues in Ubiquitous Computing”. Comm. ACM, 1993, 36 (7): 72～84.

[2] Henricksen K,Indulsks J,Rakotonirainy A. “Modeling Context Information in Pervasive Computing Systems”. In: Proc. of the 1st International Conference, Pervasive, Lecture Notes in Computer Science, Springer Verlag , Vol 2414, 2002, pp. 167～180.

[3] Chrysanthis P K, Pitoura E. “Tutorial 2: Mobile and Wireless Database Access for Pervasive Computing”. In. Proc. of the 16th Intl. Conf. on Data Engineering, 1998.

[4] Driessen P F,Gabert J,et al. “An Internet protocol for flexible. scalable, and secure interaction with ubiquitous computing devices”. IEEE, 1997.

[5] William Tolone, Gail-Joon Ahn, Tanusree Pai and Seng-Phil Hong, “Access control in collaborative systems”, ACM Computing Surveys (CSUR), vol. 37, no. 1, pp. 29-41, Mar. 2005.

[6] Sandhu R S, Coyne E J, Feinstein H L, et al, “Role-based Access Control Models”, IEEE Computer, vol. 29, no. 2, pp. 38-47, 1996.

[7] Thomas R K,Sandhu R, “Task-based authentication controls(TABC):a family of models for active and enterprise-orien-ted authentication management”. in Proceedings of the 11th IFIP WG11.3 Workshop on Database Security, Lake Tahoe, California, London: Chapman Hall, 1997, pp.166-181.

[8] Martinelli F, Mori P, and Vaccarelli A, “Towards Continuous Usage Control on Grid Computational Services”, in Proceedings of the Joint International Conference on Autonomic and Autonomous Systems and International Conference on Networking and Services, Papeete, Tahiti, Los Alamitos, CA: IEEE CS Press, 2005, pp. 82-89.

[9] George Coulouris, Jean Dollimore and Marcus Roberts, “Role and task-based access control in the PerDiS groupware platform”, in Proceedings of the third ACM workshop on Role-based access control, Fairfax, Virginia, United States, 1998, pp.115-121.

[10]http://csrc.nist.gov/groups/SNS/rbac/documents/standards/EDACv2overview.pdf.

[11] Bertino E, Bonatti PA, Ferrari E. “TRBAC: A temporal role-based access control model”. ACM Trans. on Information and System Security, 2001,4(3):191-233.