Secure Spaces: Physically Protected Environments for Information Security

Shun Hattori, Taro Tezuka and Katsumi Tanaka
Department of Social Informatics, Graduate School of Informatics, Kyoto University
Yoshida-Honmachi, Sakyo-ku, Kyoto 606-8501, Japan
email:{hattori, tezuka, tanaka}@dl.kuis.kyoto-u.ac.jp

***Abstract*— We introduce the novel concept of *"Secure Spaces"*, physical environments in which any resource is always protected from its unauthorized users' eyes or ears by assuredly enforcing its access control policies for pairs of it and each user inside them. Aiming to build such secure spaces, this paper proposes a model and an architecture for space entry control based on its dynamically changing contents, such as users, physical resources and virtual resources outputted by embedded devices. We firstly formalize the content-based entry control model and mechanism, and then describe the architecture for building secure spaces.**

I. Introduction

In recent years, access control systems have become very significant for protecting computer security in diverse scenes, especially in business companies, educational facilities, healthcare centers and so forth. Regardless of being physical or virtual, the amount of sensitive information resources which should be protected in the real world, keeps growing exponentially. Therefore, we have researched aiming to build secure spaces in the real world. Here, we define a *"Secure Space"* as a physical environment in which any resource is always protected from its unauthorized viewers by assuredly enforcing its authorization policies for pairs of itself and each user inside the physical environment.

There are two kinds of conventional access controls for the purpose of protecting resource security. One is information access control. When a user requests to perform an action on a virtual resource such as a data file on computer, information access control systems make an authorization decision on whether the access request should be granted or denied, in order to protect the virtual resource from its unauthorized users. However, they are not aware of the other users who are surrounding a device outputting it, and thus there may exist its unauthorized user in the surrounding area. If that's the case, its unauthorized user becomes able to view it, and thus its confidentiality is not always protected. In order to protect it from its unauthorized users assuedly, they have to ensure that the area surrounding a device which will be granted to output the requested virtual resource is secure, that is, there is nobody unauthorized in the surrounding area. Another is physical entry control. When a user requests to enter a physically isolated space such as a room and a building, physical entry control systems make an entry decision on whether the entry request should be granted or denied, in order to protect any physical resource inside the space from its unauthorized users. However, they determine statically regardless of what resources there are actually in the physical space, and thus there may not exist any resource which should be protected.

While outputting a virtual resource via a device embedded in a physical space, when its unauthorized user requests to enter the area surrounding the device, we can have the following two approaches to keep protecting assuedly information security of the virtual resource:

- revoking the output session of the virtual resource, or
- preventing the unauthorized user from entering the physical space which contains the virtual resource.

In order to enforce the latter approach also, we have to assume an physically isolated space with electrical lock facilities.

As contrasted with these conventional access control systems, in this paper, we propose a method for space entry control based on its dynamically changing contents, such as users, physical resources and virtual resources outputted by some embedded devices. Our proposed method allows information access control systems to be aware of not only a user who is directly requesting to access a virtual resource but also the other users who are surrounding a device which will be granted to output the requested virtual resource, and also allows physical entry control systems to make decision on whether a user should be granted or denied to enter a physical space, dynamically according to what resources there are actually in the physical space.

The remainder of this paper is organized as follows. Section 2 presents a requirement for access control in mobile and ubiquitous computing environments, Section 3 gives an overview of entry control for building secure spaces, Section 4 formalizes a content-based entry control model, Section 5 describes an architecture to realize content-based entry control for secure spaces, and Section 6 introduces some related researches. Finally, we conclude this paper in Section 7.

II. Requirement for Mobile Computing Security

Our computing environments have changed from immobile and personal ones to mobile and ubiquitous ones. With this paradigm shift, the places where we access information in the real world have changed from private spaces to public spaces.

In immobile and personal computing environments where a computer is used by a single person, because we can assume naturally that some information transmitted by a device is received by only the user who is operating the device, information access control systems have to take only the single user directly operating the device and requesting to access an information resource into account while making an authorization decision on whether the access request should be granted or denied (in Figure 1). Of cource, an information resource outputted by a device might be received not only by

the user directly requesting the access but also by the other users nearby surrounding the device, especially in pseudo-private spaces such as shared living rooms at home. However, even if in the worst case, those who can receive the outputted information resource are limited to those who can enter such a pseudo-private space where the output device is located.

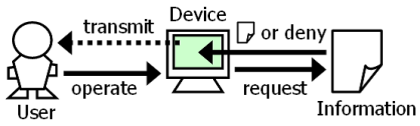

Fig. 1. Single Receiver of Information Resource Transmitted by Output Device in Immobile and Personal Computing Environments

However, in mobile and ubiquitous computing environments, we can access information anywhere at any time in our daily life, by carrying arround with our personal mobile/wearable devices or utilizing public immobile device embedded in every corner of the real world, and thus some information transmitted by such an output device might be received not only by the single user directly operating the device and requesting to access it but also by the other users nearby surrounding the device, especially in public spaces (shown in Figure 2). For information security, an information resource has its authorization policies which indicate the set of its authorized users who are permitted to access (e.g. read, write) it. But only by checking on whether a single user directly requesting to access an information resource is in the set of its authorized users, we cannot guarantee any longer that the information resource transmitted via an output device for an access request by its authorized user is not being viewed by its unauthorized users, because there might be its unauthorized user in the area surrounding the output device and it might be being viewed by its unauthorized user.

Therefore, unlike for traditional immobile and personal computing environments, access control systems for mobile and ubiquitous computing environments have to take into account not only a single user directly requesting to access an information resource via an output device but also the other users in the area nearby surrounding the output device while making an authorization decision on whether the access request should be granted or denied, in order to assure that any information resource can never be receive by its unauthorized users who do not have access rights to view it.

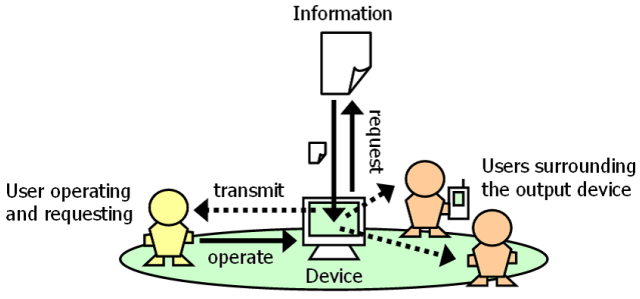

Fig. 2. Multiple Receivers of Information Resource Transmitted by Output Device in Mobile and Ubiquitous Computing Environments

III. Entry Control for Secure Spaces

We have been aiming to build *"Secure Spaces"* in which any resource is always protected from its unauthorized viewers by assuredly enforcing its authorization policies for pairs of itself and each user inside them. In order for access control systems to take into account not only a single user directly requesting to access an information resource via an output device but also the other users in the area nearby surrounding the output device while making an authorization decision on whether the access request should be granted or denied, it is necessary for them to be always aware of who there are in the area nearby surrounding the output device which will be granted to output the information resource. Our adopted strategy for it is to utilize electrically lockable environments which are physically isolated by opaque walls or partitions, such as rooms or buildings. In addition, we assume that any user inside such a secure space can view any resource inside the secure space, and any user outside the secure space cannot view any resource inside the secure space.

When a user requests to enter a secure space, our entry control system will make an entry decision on whether the entry request should be granted or denied, according to whether or not the requester satisfies all authorization policies of any resource inside the secure space, in order to protect information security for all contents of the secure space.

If the requester does not satisfy some authorization policies of the physical resources inside the secure space, our entry control system has only one approach of preventing the requester from entering the secure space that contains at least one physical resource which the requester does not have access right to view (shown in Figure 3). Reversely, when a physical resource requests to enter a secure space, our entry control system will also prevent the physical resource from entering the secure space that contains at least one user who does not have access right to view it.

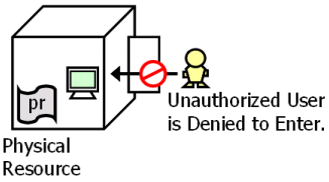

Fig. 3. Only one approach for protecting physical resource's authorization policies: preventing its unauthorized users from entering the secure space containing it.

Meanwhile, if the requester does not satisfy some authorization policies of the virtual resources outputted already via the embedded devices inside the secure space, our entry control system might have to be also denied in most cases. However, in special cases, that is, only if the requester satisfies all authorization policies of any physical resources inside the secure space and does not satisfy some authorization policies of virtual resources inside the secure space, our entry control system can also choose the alternative approach to grant the requester to enter the secure space after revoking the virtual resources which the requester does not have access right to view (shown in Figure 4). Reversely, when a virtual resource requests to be outputted via a device embedded in a secure

space, if in the secure space there exists at least one user who does not have access right to view the virtual resource, our entry control system has only one approach of denying the output request because our entry control system cannot expunge its unauthorized users inside the secure space.

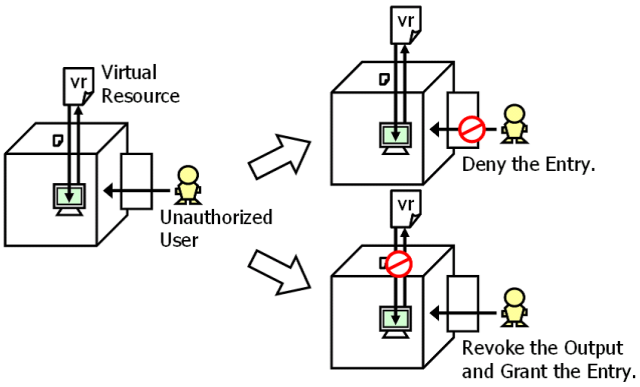

Fig. 4. Two approaches for protecting virtual resource's authorization policies: preventing its unauthorized users from entering the physical space containing an embedded device outputting it, or revoking its output session before granting its unauthorized users to enter the physical space.

IV. Formalized Model

In this 4th section, we introduce a formalized model and mechanism for space entry control based on its dynamically changing contents, such as users, physical resources and virtual resources outputted by some embedded devices. At the following, we formalize content-based entry control model by listing component primitives and by defining the syntax and semantics of the model components.

Definition 1: Model Entities

This model has the following four kinds of entities.

- User:
  is a physical entity who requests to enter or exit a secure space. We assume that a user can view any physical or virtual resource inside his/her current secure space.

- Space:
  is a physical environment isolated by opaque walls, such as a closed room or a building, and its doors' opening/shutting can be controlled electrically.

- Physical Resource:
  is a physical entity which should be protected by this model, such as a sensitive document of a big company. We assume that once the secure space containing a physical resource is entered by its unauthorized user, we can no longer prevent its unauthorized user from viewing it. Therefore, we have to prevent its unauthorized users from entering the secure space containing it at any cost, in order to assuredly enforce its authorization policies.

- Virtual Resource:
  is a virtual entity which should be also protected by this model, such as a data file on computer. Unlike a physical resource, even if the secure space containing a device which is outputting a virtual resource is entered by its unauthorized user, we can prevent its unauthorized user from viewing it by revoking its all output sessions in the secure space. Therefore, the secure space's administrator can (or must) choose whether to prevent its unauthorized users from entering the secure space containing it, or to revoke its all output sessions in the secure space before granting the entry request, by comparing two weights of the contents of the secure space after each choice.

Definition 2: Component Primitives

The entry control system based on this model is always stateful for the following component primitives.

- U: is the universal set of User identities.
- S: is the universal set of Spaces.
- PR: is the universal set of Physical Resources.
- VR: is the universal set of Virtual Resources.
- PRAP: is a set of Physical Resource's Authorization Policies stored in the system.
- VRAP: is a set of Virtual Resource's Authorization Policies stored in the system.
- $\mathsf{PERM} = \{\mathtt{grant}, \mathtt{deny}\}$: is a set of PERMissions that indicate whether an entry request be granted or denied.

Definition 3: Model Functions

This model uses the following functions in order to keep up on the set of entities in each secure space at each time and evaluate the weight of its contents.

- $\mathtt{cu}$ (Contained Users): $\mathsf{S} \rightarrow 2^{\mathsf{U}}$,
  is a function mapping each secure space s_i to the set of users inside the secure space, $\mathtt{cu}(s_i)$.

- $\mathtt{cpr}$ (Contained Physical Resources): $\mathsf{S} \rightarrow 2^{\mathsf{PR}}$,
  is a function mapping each secure space s_i to the set of physical resources inside the secure space, $\mathtt{cpr}(s_i)$.

- $\mathtt{cvr}$ (Contained Virtual Resources): $\mathsf{S} \rightarrow 2^{\mathsf{VR}}$,
  is a function mapping each secure space s_i to the set of virtual resources inside the secure space, $\mathtt{cvr}(s_i)$.

- $\mathtt{w}$ (Weight): $\mathsf{S} \times 2^{\mathsf{U}} \times 2^{\mathsf{PR}} \times 2^{\mathsf{VR}} \rightarrow \mathcal{R}$,
  is a function mapping a set of contents in each secure space s_i to its evaluated value for the secure space, $\mathtt{w}(s_i, \mathtt{cu}(s_i), \mathtt{cpr}(s_i), \mathtt{cvr}(s_i))$.

- $\mathtt{authU}$ (Authorization for User): $\mathsf{UER} \rightarrow \mathsf{PERM}$,
  is a function mapping each user's entry request uer_i to the authorization decision, $\mathtt{authU}(uer_i)$.

- $\mathtt{authR}$ (Authorization for Resource): $\mathsf{RER} \rightarrow \mathsf{PERM}$,
  is a function mapping each resource's entry request rer_i to the authorization decision, $\mathtt{authR}(rer_i)$.

Definition 4.1: Physical Resource's Authorization Policy

An authorization policy for a physical resource is defined as a 2-tuple of a physical resource and its authorized user,

$$\mathsf{PRAP} \subseteq \mathsf{PR} \times \mathsf{U}.$$

If $(pr, u) \in \mathsf{PRAP}$ where $pr \in \mathsf{PR}$ and $u \in \mathsf{U}$, then it states that the physical resource pr grants the user u to view itself in the same secure space.

Definition 4.2: Virtual Resource's Authorization Policy

An authorization policy for a virtual resource is defined as a 2-tuple of a virtual resource and its authorized user,

$$\mathsf{VRAP} \subseteq \mathsf{VR} \times \mathsf{U} \ \ (\times \mathsf{A}),$$

where A stands for the universal set of available actions performed on virtual resources, such as `read` or `write`.
If $(vr, u) \in \mathsf{VRAP}$ where $vr \in \mathsf{VR}$ and $u \in \mathsf{U}$, then it states that the virtual resource vr grants the user u to view itself outputted by some device embedded in the same secure space.

Definition 5.1: User's Entry Request

An entry request by a user is defined as a 2-tuple of a user and a secure space which he/she is requesting to enter,

$$\mathsf{UER} \subseteq \mathsf{U} \times \mathsf{S}.$$

If $(u, s) \in \mathsf{UER}$ where $u \in \mathsf{U}$ and $s \in \mathsf{S}$, then it states that the user u requests to enter the secure space s and to view the contents inside the secure space.

Definition 5.2: Physical Resource's Entry Request

An entry request by a physical resource is defined as a 2-tuple of a physical resource and a secure space which it is requesting to enter,

$$\mathsf{PRER} \subseteq \mathsf{PR} \times \mathsf{S}.$$

If $(pr, s) \in \mathsf{PRER}$ where $pr \in \mathsf{PR}$ and $s \in \mathsf{S}$, then it states that the physical resource pr requests to enter the secure space s and to be viewed by any user inside the secure space.

Definition 5.3: Virtual Resource's Entry Request

An entry request by a virtual resource is defined as a 2-tuple of a virtual resource and a secure space which it is requesting to be outputted via a device embedded in,

$$\mathsf{VRER} \subseteq \mathsf{VR} \times \mathsf{S}.$$

If $(vr, s) \in \mathsf{VRER}$ where $vr \in \mathsf{VR}$ and $s \in \mathsf{S}$, then it states that the virtual resource vr requests to be outputted by some device embedded in the secure space s and to be viewed by any users inside the secure space.

Definition 6: Space's Contents Weighting

The weight that evaluates contents in a secure space could be defined by several manners. Here, we introduce one of them which seems to be most often used by space administrators.

The weight $\mathtt{w}(s, us, prs, vrs)$ that evaluates a set of entities such as users, physical resources and virtual resources in a secure space according to its administrator is defined as the summation of each positive weight $\mathtt{w}(s, u, r)$ that evaluate a pair of each user and each resource inside the secure space,

$$\mathtt{w}(s, us, prs, vrs) = \sum_{u \in us, r \in prs \cup vrs} \mathtt{w}(s, u, r),$$

where $s \in \mathsf{S}$, $us \in 2^{\mathsf{U}}$, $prs \in 2^{\mathsf{PR}}$ and $vrs \in 2^{\mathsf{VR}}$.

Algorithm 2.1: Authorization for User

An entry request that a user u wants to enter a secure space s is granted, if and only if any physical resource in the secure space grants the user to view itself and if any virtual resource outputted by some device embedded in the secure space grants the user to view itself or the evaluated weight in the case of granting the user to enter the secure space after revoking all of its virtual resources which deny the user to view itself is higher than the evaluated weight in the case of denying the user to enter the secure space.

$$\begin{aligned} &\forall u \in \mathsf{U}, \forall s \in \mathsf{S}, \mathtt{authU}(u, s) = \mathtt{grant} \\ &\Leftrightarrow (\mathtt{apr}(s, u) = \mathtt{cpr}(s)) \\ &\quad \wedge \{(\mathtt{avr}(s, u) = \mathtt{cvr}(s)) \vee (\mathtt{w}(s, u) \geq \mathtt{w}(s))\} \end{aligned}$$

where $\mathtt{au}(s, u)$, $\mathtt{apr}(s, u)$ or $\mathtt{avr}(s, u)$ is the assumptive set of users, physical resources or virtual resources inside the secure space after granting the user u to enter the space s and regulating its contents to keep secure, respectively.

$$\begin{aligned} \mathtt{au}(s, u) &= \mathtt{cu}(s) \cup \{u\} \\ \mathtt{apr}(s, u) &= \{pr_i \in \mathtt{cpr}(s) | (u, pr_i) \in \mathsf{PRAP}\} \\ \mathtt{avr}(s, u) &= \{vr_j \in \mathtt{cvr}(s) | (u, vr_j) \in \mathsf{VRAP}\} \\ \mathtt{w}(s) &= \mathtt{w}(s, \mathtt{cu}(s), \mathtt{cpr}(s), \mathtt{cvr}(s)) \\ \mathtt{w}(s, u) &= \mathtt{w}(s, \mathtt{au}(s, u), \mathtt{apr}(s, u), \mathtt{avr}(s, u)) \end{aligned}$$

Algorithm 2.2: Authorization for Resource

An entry request that a physical or virtual resource r wants to enter a space secure s is granted, if and only if any user in the secure space is granted to view the resource.

$$\begin{aligned} &\forall r \in \mathsf{PR} \cup \mathsf{VR}, \forall s \in \mathsf{S}, \mathtt{authR}(r, s) = \mathtt{grant} \\ &\Leftrightarrow \mathtt{au}(s, r) = \mathtt{cu}(s) \end{aligned}$$

where $\mathtt{au}(s, r)$ is the assumptive set of authorized users who have access right to view any resource in the secure space even after granting the resource to enter the space.

$$\mathtt{au}(s, r) = \{u_i \in \mathtt{cu}(s) | (u_i, r) \in \mathsf{PRAP} \cup \mathsf{VRAP}\}$$

V. Architecture

In this 5th section, we describe a system architecture to realize content-based entry control for secure spaces in the real world. A *"Secure Space"* consists of the following facilities and is shown in Figure 5:

- Space Management:
  is responsible for figuring out its contents such as its users, its physical resouces and virtual resources outputted via its embedded devices, and for making an authorization decision on whether an entry request should be granted or denied.

- User Authentication:
  is responsible for authenticating who requests to enter or exit the secure space (e.g., RFID reader or biometrics), and for notifying the space management of it.

- Object Authentication:
  is responsible for authenticating what physical resource requests to enter or exit the secure space (e.g., RFID reader), and for notifying the space management of it.

- Electrically Lockable Door:
  is responsible for assuredly enforcing entry control over physical entities such as users and physical resources, according to the instructions by the space management.

- Isolating Opaque Walls:
  We assume that any user inside the secure space can view any resource inside the secure space, and that any user outside the secure space cannot any resource inside the secure space.

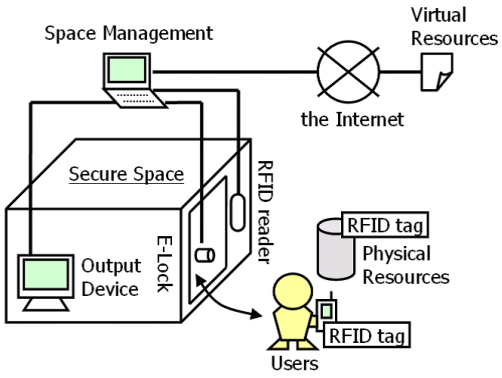

Fig. 5. Architecture of Secure Spaces

VI. Related Work

This our research is related very well to the field of access control for *"Smart Spaces"*. Smart spaces are physically isolated environments which heterogeneous computing resources such as output devices, various sensors or communication apparatus are embedded in and provide advanced services for their visitors by cooperating with each other. They are also called Active Spaces [1], [2], Intelligent Rooms [3], Aware Homes [4] and so forth.

In [2], an active space has four access modes such as Individual, Shared, Collaborative and Supervisor-mode, switching dependent on the presences of all users and the activities being performed in the active space. The set of permissions valid in the active space is calculated as the intersection set of their individual-assigned permissions in Shared-mode, the union set of them in Collaborative-mode. In Supervisor-mode, a supervisor such as a lecturer for students, acquires more permissions than in Shared-mode, but does not obtain more permissions than in his/her Individual-mode. In [5], utilizing a room with two chambers for entry and exit, allows us to figure out those who enter and exit the room, that is, to identify who is in the room. While a device embedded in the room is outputting virtual information resource, its unauthorized user cannot enter there until all of its output sessions are revoked, but his/her entry request to the room is never denied unlike our proposed model. These above-mentioned researches have tackled access control for physical spaces like our research, but have not supported physical entry control based on their contents, such as users and physical/virtual resources.

VII. Conclusion and Future Work

In this paper, we proposed a model and an architecture for space entry control based on its dynamically changing contents, such as users, physical resources and virtual resources, in order to build *"Secure Spaces"* in which any resource is always protected from its unauthorized viewers.

In the future, we plan to develop the prototype of our proposed entry control system and evaluate its effectivity or functionality by applying it to actual use cases in the real world. Moreover, we would like to formulate the hierarchical and more flexible model as the next step for *"Secure Spaces"*, in order to allow space administrators to configure their space more easily and more flexibly.

Acknowledgement

This work was supported in part by MEXT The 21st Century COE Program "Informatics Research Center for Development of Knowledge Society Infrastructure" (Leader: Katsumi Tanaka, 2002-2006), and MEXT Grant-in-Aid for Scientific Research on Priority Areas: "Cyber Infrastructure for the Information-explosion Era", Planning Research: "Contents Fusion and Seamless Search for Information Explosion" (Project Leader: Katsumi Tanaka, A01-00-02, Grant#: 18049041).

References

[1] J. Al-Muhtadi, A. Ranganathan, R. Campbell, and M. D. Mickunas. Cerberus: A Context-aware Security Scheme for Smart Spaces, In *Proceedings of the 1st IEEE International Conference on Pervasive Computing and Communications (PerCom'03)*, pp.489–496, March 2003.

[2] G. Sampemane, P. Naldurg, and R. H. Campbell. Access Control for Active Spaces, In *Proceedings of the 18th Annual Computer Security Applications Conference (ACSAC'02)*, pp.343–352, December 2002.

[3] Y. J. Song, W. Tobagus, D. Y. Leong, B. Johanson, and A. Fox. iSecurity: A Security Framework for Interactive Workspaces, *Technical report*, Stanford University, September 2003.

[4] M. J. Covington,W. Long, S. Srinivasan, A. Dey,M. Ahamad, and G. Abowd. Securing Context-aware Applications Using Environment Roles, In *Proceedings of the 6th ACM Symposium on Access Control Models and Technologies (SACMAT'01)*, pp.10–20, May 2001.

[5] P. G. McLean. A Secure Pervasive Environment, In *Proceedings of the Australasian Information Security Workshop 2003 (AISW'03), Conferences in Research and Practice in Information Technology*, pp.67–75, January 2003.