Physical Access Control with Smart Intrusion Tracking and Hunting Agent

António Leong, Simon Fong
Faculty of Science and Technology
University of Macau
Macao
Email: antleong@gmail.com, ccfong@umac.mo

Abstract

In our previous work [ASS04], a general framework for detection of irregular access patterns based on the concept of security merits is described and modeled using predicate logics. In this paper, we will focus the concerns on performance and scalability issues and introduce a new role in the system that is a real-time security agent called SMITH (SMart Intrusion Tracing and Hunting). SMITH is a distributed agent for detecting irregular access patterns in a region. For example, a legitimate user who just entered a room one minute ago should not have its physical access detected from another room 100 meters away. As a case study, the application of SMITH on a real-world scenario is analyzed. That is the physical access control for the complex for East Asian Games 2005 held in Macao, the Macao East Asian Games Dome. The venue offers a good example of how users' trails especially the suspicious ones can be detected and tracked in a secure closed environment. This work is believed to have significant relevance to maintaining the security of a place against terrorism.

Key words

Smart-card, contact-tracing, intrusion tracking, intrusion detection.

1. Introduction

In our previous work [ASS04], we have described and analytically modeled a general framework for irregular access patterns detection by extending the concept of static access control list, allowing early detection of abnormal access card usages by analyzing user's access patterns in real-time.

Commercial physical access control products evaluate access authorization based on defined security policies, which are usually a set of predefined static rules. Authorization process starts with identification of authentic of user based on verification of user's credential, which might varies from a smart card presented by the user, a password or the fingerprint of the user. After the authentication of the user succeeds, the system will verify user's access based upon a set of rules defined in the security policy. Access will be granted if all validations return a positive result; otherwise, access will be denied. Authorization decisions made by standard access control systems usually return a Boolean result. Either access is granted if no violations occurred, or denied if any violations are verified.

By extending this concept a logical model is developed in our last paper that allows a computerized system to interpret suspicious or abnormal access patterns and the system returns a quantitative alert level of a certain range that represents the possible risks derived from it. The concept of security merits is also introduced, which is real positive number that is first assigned to each user, and we assume that this value is stored in user's credential. If any violations were verified during the authorization process, this value is deducted according to a mathematical formula, which is a function that represents the level of criticalness of violations in a logical form. The new value will be stored in the card accordingly, and when it drops to lower than zero, access will be denied, otherwise, access will be granted. This way, the security control could exploit certain fuzzy nature in interpreting how critical is the violation.

Such logical model has been implemented in a software prototype for physical access control using smart card technology [SCT]. Unlike traditional access control scheme, which make permission decisions based on solely Boolean result returned from the authorization process, by integrating this model to a physical access control system, the system can

react and make decisions based upon different values of security merits. By adopting this scheme, potential intrusions and irregular activities can be detected as a pre-warning in preliminary phases and proper actions can be taken.

Practically, system scalability and performance is an important issue in real-world physical access control systems. Large scale physical access control systems must be able to response rapidly to user's requests; otherwise, it will induce signal latency and long waiting queues at the entrances of access points. In this paper, we will shed some light on the scalability and performance issue of the developed prototype and introduces a new role in the system, a security software agent called SMITH (SMart Intrusion Tracing and Hunting).We will then analyze the application of this model on a real-world application: physical access control, as an example, for the main complex for East Asian Games 2005 held in Macao, the Macao Dome.

The rest of the paper is organized as follows. Section 2 surveys on previous works and commercial access control solutions. Section 3 outlines our software model that incorporates the SMITH technology. Section 4 discusses how it integrates in a real-world scenario. Section 5 describes the future works that can be developed based on this model. A conclusion is followed at the end of this paper.

2. Previous Works

A Physical Access Control system, as the name implies, is built for verification and authentication of different human entities and grant them access to different facilities or services accordingly. Nowadays, these systems usually employ smart card technology, which is widely accepted as a medium that can hold securely user's personal identification data. As for our groundwork, the meaning of Physical Access Control System is defined by Smart Card Alliance [SCA] is that a system composed with the following elements:

- A card that is presented to a door reader
- The reader, that responds with a signal indicating a valid card, and
- The door or gate, that is unlocked if entry is authorized

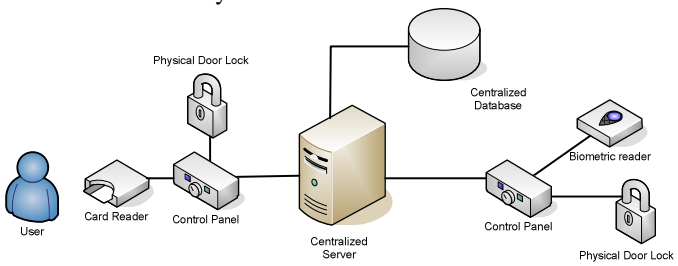


Figure 1 – Access Control System Schematic

A typical Physical Access Control System is shown in the above figure. The actual process starts when the user presents the card to the control panel, which is usually mounted next to a door. The reader retrieves and processes data from the card, then will send it to the control panel. The role of the control panel is to validate the card and accepts the data. Depending on the overall system design, control panel might either send the data to a centralized server or may determine the user's rights and grant him the access accordingly. If no irregularities were detected upon checking, access will be granted and the control panel will send a signal to unlock the physical door lock.

The response to an invalid card depends on the defined security policy and procedures. If an invalid card is presented, the centralized server and control panel will just ignore it and will not send an unlock code to the door lock control hardware. They may send a signal to the reader to emit a different sound, signaling that access was denied. Furthermore, they might also notify and activate other security systems (e.g. Closed-Circuit Television, alarms), indicating that an unauthorized card is being presented to the system.

Commercial products using smart cards are widely available on the market. From the references [GEM][IDC][SCSITE] they list out some of these typical commercial systems. We found out that most of these products usually focus on the following features: identity mechanism, flexibility, cost-of-ownership, scalability and security.

Due to various types and characteristics of commercial products available on the market, researchers tend to investigate how different products could integrate well with the existing database system adopted by the enterprise. As many companies already adopted their own database system, which holds the information of each individual employee, physical access control system design usually focuses on scalability and flexibility in order to integrate with these systems.

In an article of Information Storage + Security Journal [INTOS], it describes how a regional claims contractor for the U.S. Medicare program, builds an integrated platform that incorporate different technologies, from access control, digital video, analog CCTV switcher, intrusion system and intercom system to a secure security system.

In another article [TAPS], it discusses the trend of convergence of existing information technology and security system. New physical access control products can now integrate with existing network using TCP/IP. Many products now offer web-based control panels to provide an installation-free environment for management of control panels.

Another field of physical access control system research trends to investigate on further improvement on identification of authentic users by using additional properties of the card holders. Although user's unique biometric properties like fingerprint, eye scan, face or handprint, or voice pattern prove to be more reliable than traditional password-based system, due to cost consideration, only a few businesses and government agencies have deployed or currently testing for identification of users. [IDTHF]

2.1. Smart Card-Based Irregular Access Patterns Detection System

Traditional approaches for evaluating access permission are usually based on comparison over a set of static rules. By formulizing these rules into a generic mathematical model, we obtain the following equation:

$$vio(\{v_c\}) = c(\{v_c\})$$

Equation 1

where:
c – conditional function
$\{vc\}$ – input vector of conditions factors

The violation function performs a set of checks according to the security policy and returns a Boolean value. The false and true values of the Boolean results are represented as 0 and 1, respectively. If a value of 1 is returned, which corresponds to detection of critical violations, access will be denied. Otherwise, a value of 0 is returned and access can be granted. The return values of the violation function $vio(\{v_c\})$ are calculated based on the conditional function $c(\{vc\})$, which represents the validations rules in mathematical form. As input parameters vary according to different type of checks and validations, we use a vector $\{v_c\}$ to generalize the set of input parameters.

Validations performed using Equation 1 will simply return a Boolean result represented by the values 0 and 1. They do not take account of dynamic changes in the real-time access control system and cannot distinguish some possible type of actions that the model might consider as abnormal or irregular. To handle this type of behaviors, we developed a generalized model by extending Equation 1 and introduced the concept of security merits s, where $s \in S$. During the validation process, if any type of abnormal action is detected, the value stored in the card $s_{card} \in S$ will be deducted by the amount of security merits s_d calculated based on our logical model. We assume that access will be denied for any values of $s_{card} \leq 0$, and according to the access control policy, an alert can be sent to security officers if the value of security merit drops below a certain value. Security officers can also define the period of time when this value is reset to its original level.

The concept of security merit can be used to derive a mathematical model that reflects the abnormal usage behaviors and establishes a quantitative relationship between system alert levels. By extending Equation 1, we get the following formula defines the amount of security merits to be deducted when any violations are detected.

$$s_d(\{v_c\},\{v_b\},l) = c(\{v_c\}) \cdot b(\{v_b\}) \cdot r(l)$$

Equation 2

where:
s_d – Security merits to be deducted
c – conditional function, $\{v_c\}$ – input vector of conditions factors
b – behavior function, $\{v_b\}$ – input vector of behavior factors
r – reduction factor function, l – security level

Abnormal usage behaviors are rationalized and converted into a behavior function b, which is a function that accepts user's access information as input parameters and return the level of abnormality mathematically. Using this approach, abnormal usages behaviors can be classified based on the deducted value s_d. In other words, even if authorization is granted, there are post-conditions that apply.

Based on this approach, we developed a model based on smart card technology that incorporates the concept of security merit. When the user presents a card to the system, it will verify its authenticity and execute a set of procedures to verify if any violations occur. Once detected, the corresponding value s_d will be deducted, stored in the card and reported to the system so proper actions can be taken. The cumulative effect of abnormal usages will result a denial on user's access and reported to the system. Table 1 show two possible permission validations checks formulized using Equation 2. More examples are shown in the appendix.

Card expiration check (Critical) Description: Check whether card is expired $c(\{v_c\}) = \phi\left(t_{card\_expiry} > t_c\right), \{v_c\} = \{t_{card\_expiry}, t_c\}$ $b(\{v_b\}) = \lvert s_{card} \rvert, \{v_c\} = \{s_{card}\}$ $r(l) = l_{def}, l = l_{def}$ t_{card_expiry}: expiry time of the card	**User permission level check (Critical)** Description: Check whether user has the required access permission level $c(\{v_c\}) = \phi\left(lvl(uz) \le lvl_{req}\right), \{v_c\} = \{lvl(uz), lvl_{req}\}$ $b(\{v_b\}) = \lvert s_{card} \rvert, \{v_c\} = \{s_{card}\}$ $r(l) = l_{def}, l = l_{def}$ $lvl(uz)$: Function that returns the level of a user uz lvl_{req}: Required permission level (positive increasing integer represents higher level)

Table 1 – Permission validations checks

3. Security Agent SMITH

Real-world smart-card based access control system not only deals with a large number of online and offline access points, it also has to deliver simple and responsive permission validation services for cardholders. The user interface of the system should be designed to be simple enough and accessible to all type of users. Users simply need to present their credential to the access device for self identification and admission is conditionally granted for a particular facility.

On the other hand, the system should be highly responsive even when it handles a large amount of user accesses during a short period of time. General users expect that access will be granted almost immediately (no longer than a few seconds) after they present their ID card to the access verification device and even small delays during access verification by the system will cause long waiting queues when there is a high volume of users.

During peak usage hours, the number of concurrent users will increase dramatically. The system should be designed to handle this wave of users without compromising the performance. If access information is exchanged between a local access point and the centralized server, extra delays will introduce during transaction of information. Furthermore, transmission of access data for authentication and validation will introduce further delays to the process.

To reduce the waiting time and increase the system's usability, the system must be properly designed to cope with a large flow of users. Based on the model developed in our previous work, we add a new role as a security agent, as shown in the following figure:

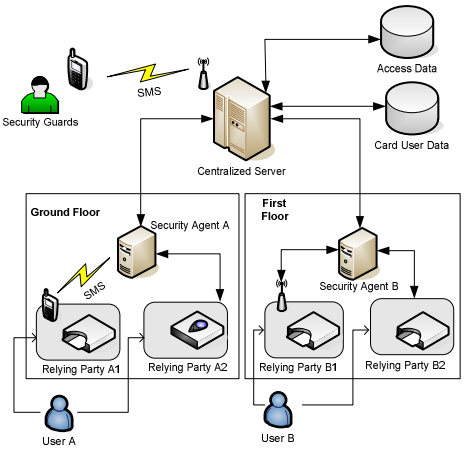


Figure 2 – Access Control System Schematic

The details of the model are shown in Figure 3. The authentication process starts when user presents their credential to the ***Relying Party*****.** Credential is something that can uniquely identifies the user and cannot be easily tampered and duplicated by attackers. Usually, it can be a smart card that stores securely personal identification of users or user's biometric properties like fingerprint or iris.

The relying party is an integrated device with basic computational power. The ***Access Data Module*** will first retrieve user's identification data from either a simple smart card reader or a biometrics reader. It will pass the information to the ***Security Check Module,*** which contain a set of ***security methods***. Security methods are the set of validation rules executed for verification of user access permissions, and as a class of objects implemented based on the logical model described in the latter section, they can be added or removed dynamically.

The security check module will first verify the authenticity of the user by matching its information with those stored in the ***Local User Information Cache***. Then, a basic set of security methods, as defined in ***Local Security Methods Table***, are

executed accordingly. In execution of some of the security methods, security check module might also request for additional access information of the user from the security agent if it is not available in ***Local Access Log***.

If any violations were verified, the deducted security merit value and the violation type will be reported immediately to the security agent through the ***Communications Module***. Depending on its geographical location and required security level, it might either establish a permanent connection with the security agent through a wire or wireless network or a temporary connection through the GSM network. Otherwise, access information will be stored in the ***Local Access Log***, which will be updated immediately or in batch through the ***Log Updater***, according to the defined security policy.

Relying parties, instead of passing all information to the centralized server as in our previous work, will pass them to the ***Security Agent.*** These agents have higher computational and storage capacities than the relying parties, and they are responsible for maintaining and controlling a group of relying parties, usually located within a specific zone. Based on this scheme, workload is distributed over several security agents and performance could be sustained.

Theoretically, performance is directly proportional to the number of security agents and it would be ideal to maximize the number of security agents. However, the cost-of-ownership for installing and maintaining a large number of agents will increase accordingly.

To resolve this issue, we have introduced the concept of ***SMart Intrusion Tracking and Hunting*** (abbreviated as SMITH) mechanism into our security agents. With this mechanism, security agents are not just mere devices with higher computational and storage capacities to distribute the computing capacities of a centralized server, but they are able to perform real-time analysis of usage patterns. Therefore the workloads from the centralized server will be shared by delegating them to the distributed agents.

General users are not able to accept security validations that take longer than a second, and most expect an immediate response once they present their credential to the system. We have to achieve a balance point between required level of security and the system's overall performance. Theoretically, the required level of security increases as the number of security methods increases, but on the other hand, this will impair the overall performance and might decrease system's overall usability.

The general rule for relying parties is to execute the most basic set of security methods to increase the system's overall response. The system might request for additional validations for some types of users. When less critical abnormalities or irregularities were detected as a user presents their credential, its security merit will be deducted, even though access would be granted. Relying parties will still report these violations to the security agent. As this information is processed by the ***Security Analysis Module***, it might increase the number of required security methods to be executed on this particular user and will inform all the relying parties nearby.

By using this scheme, the performance of others users will not be affected. But it results as a more severe checking on subsequent access for the targeted users who are the suspicious ones. Targeted users will be "tracked" and "hunted" (closely monitored) by the system for a certain period of time. Although they might find slight decrease in usability like longer check time during the course, other users' performance won't be affected. This condition will be on temporary basis and will return to normal after a certain period of time. If the system is connected to a surveillance camera or alert system, the security agent might even inform the situation to the ***Centralized Server***, which is responsible for informing these alert systems so that target users are not overlooked during the progress.

Security agent might also increase the number of security methods for certain group of users as sampling basis for a certain period of time. The results findings might be further analyzed and the system might increase the number of security checks dynamically as required. Apart from this functionality, security agent might also provide the required data request by relying agents and must be able to maintain a local database that contains all access information of the relying parties they are handling.

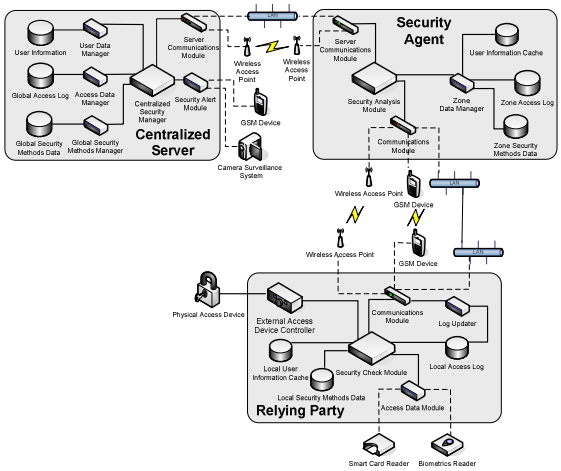


Figure 3 – Detailed Access Control System Schematic

Centralized server is mainly responsible for handling all user and access information. The ***Centralized Security Manager*** receives information from different security agents and sent to the corresponding data mangers for storage archival. It will also fetch any data requested by security agents. Furthermore, it is also responsible to inform the ***Security Alert Module if*** abnormalities were reported by any of the security agents, which will take appropriate actions according to the system's security policy.

To achieve higher performance, ***Centralized Security Manager*** is also in charge of distribution of user information, access data and security methods data to each of the security agents. Each security agent includes a ***Zone Data Manager,*** which is responsible for management of all these information. Security analysis module in each of the agents is responsible for deciding what information should be cached from the centralized server and distributed to each of the relying parties.

3.1. Security Agent

In this section, we describe the basic functionalities and the design of new role introduced in our model, the security agent. Basically, security agent performs the following functions:

- Perform analysis of usage pattern based on violations reported by relying parties and dynamically assign new security methods to different relying parties.
- Provides access information upon request from a particular relying party.
- Request access information from centralized server.
- Analyze what user information is frequently used by relying parties and distribute to them accordingly.

The following figure shows a class diagram from the design of our security agent model. As shown in the below figure, the class *SecurityMethod* performs the set of validations upon request of the *SecurityCheckModule*, and two example classes of *SecurityMethod* are in the figure. As seen in the figure, data structure in the *Card* and *Users* classes, as it is vital for comparison of data stored in cards with the data stored in the system to avoid possible alteration of card data by tampered users.

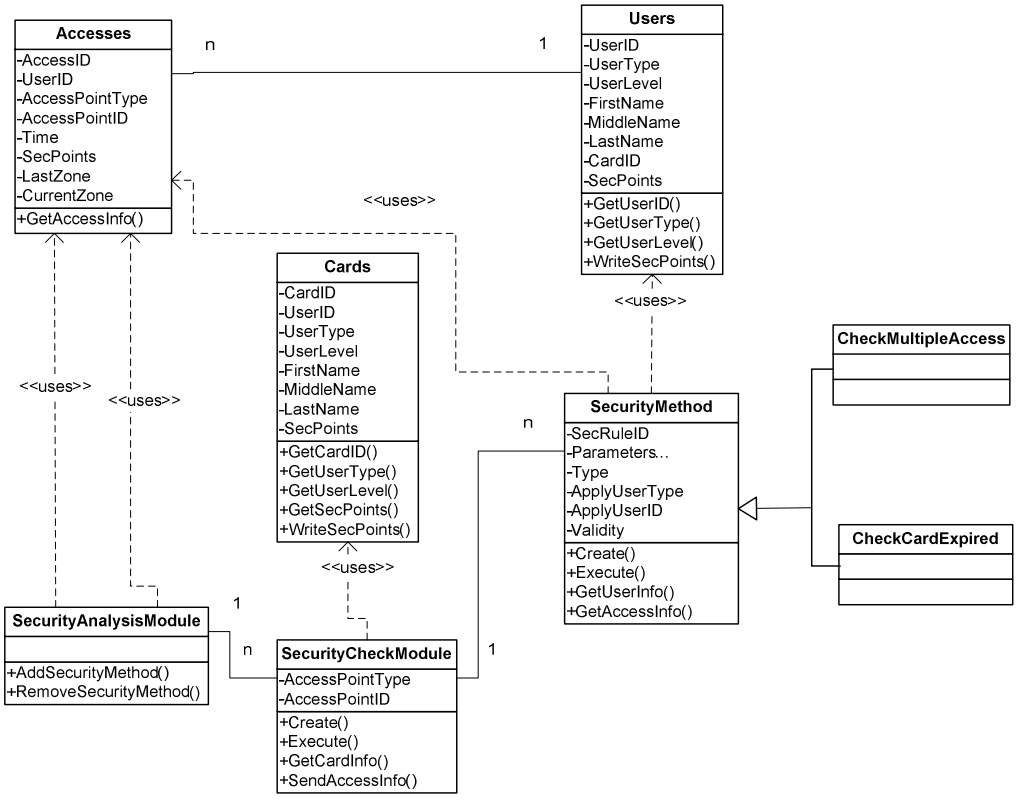


Figure 4 – Class Diagram

Figure 5 shows a sequence diagram and describe how classes interact when the system detects a possible misuse and a new security method is added dynamically for the targeted users. When the user presents a card, *SecurityCheckModule* will request the card's information and will initiate the validation process. The *SecurityMethod* class will execute the check accordingly and return the corresponding deducted security merit value. *SecurityCheckModule* will report the results to the *SecurityAnalysisModule* which will instruct all other *SecurityCheckModules* to add a new *SecurityMethod* dynamically.

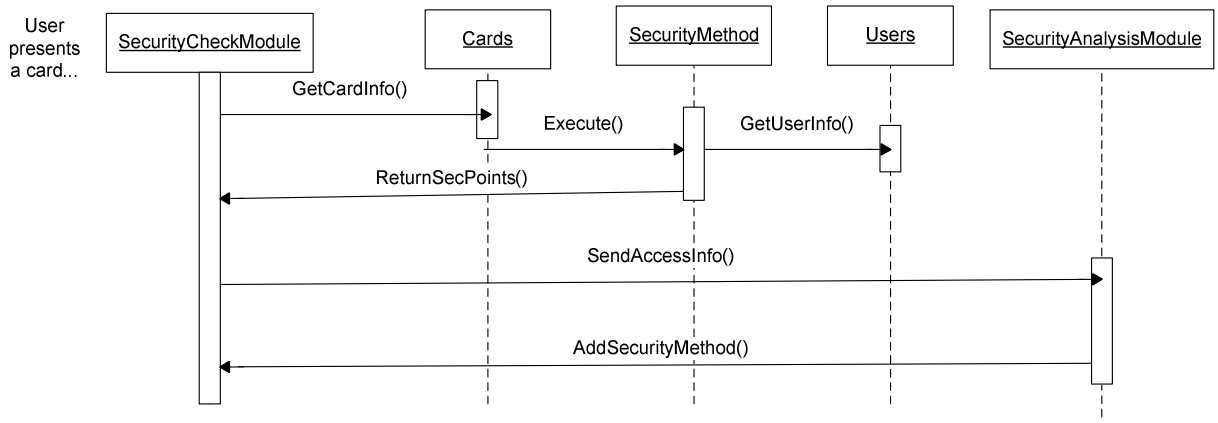


Figure 5 – Sample Sequence Diagram

4. Case Study

In this section, we describe how our model can be integrated in a real-world scenario. We will discuss on a possible implementation of a physical control system with SMITH mechanism of the main complex for East Asian Games 2005 held in Macao, the Macao East Asian Games Dome. [EAG05].

Macau East Asian Games Dome, situated in Cotai that is between the two islands of Macao, Taipa and Coloane, is now still under construction and will be Macao's largest and most modern sport facility when it opens in December 2004. It is a three-storey multi-purpose sport complex covering a total area of 139,960 m^2 and has been designed with two separate functional indoor pavilions ideal for different indoor sports and activities, as well as a large exhibition hall that can accommodate up to 2,000 people. There are two main pavilions, with Pavilion I of a total seating capacity of more than 7,000, of which the main feature is the dedicated indoor track and field set-up, Pavilion II, with 2,000 seats and is designed with a central stage that offers an U-shaped spectators seats setting, and a multi-purpose court that can hold 2,000 people for different types of activities.

With total participants of 3000 athletes from 9 countries and regions, over 10,000 staffs and expecting 1 million of visitors (based on currently statistics of visitors to Macau per month) for the East Asian games 2005, a proper physical access control for the main venue of the Games is required for preventing possible fraudulent uses and even terrorist attack.

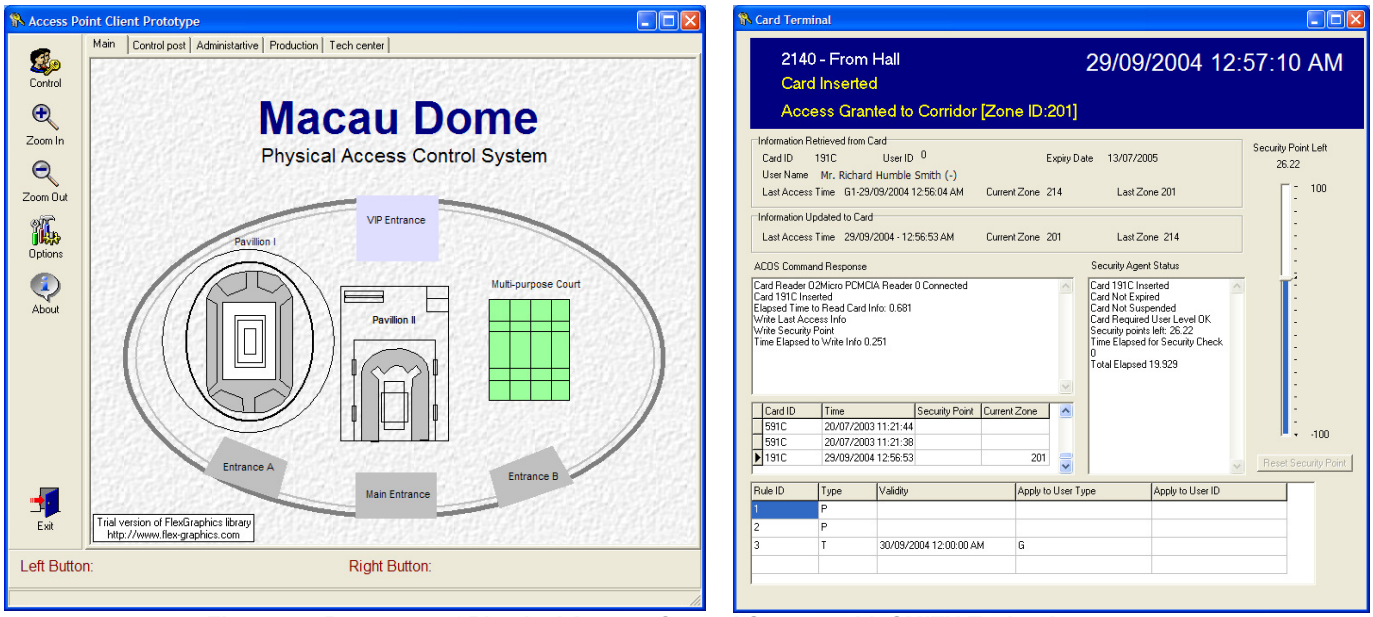


Figure 6 – Prototype of Physical Access Control System with SMITH Technology

We start by first defining the class of users that interact with the system. The following six classes of users are defined: staffs, athletes, security guards, VIP users, visitors, temporary/volunteers staffs. In our system, we must denote special

notice to two types of users, which are the security guards and temporary staffs. Due to the size and population limitations of Macao, the organizing committee has to hire a large number of volunteers staffs (up to 8000 for now) to participate in the event. And due to population limitation, the committee must focus on its quantity rather than its quality, so a low level entrance requirement is set for this type of staffs. As these classes of users has higher access levels than general visitors, and have access to different rooms and facilities, which might lead to security holes on the system.

Even for a proper entrance requirement system, we assume that it would not be feasible to rigidly verify the background of all these temporary staffs and is not possible to filter all possible intruders. Thus, the system should be able to track and hunt irregular usage behaviors of this class of users. So the infiltrated people who act suspiciously will be detected.

Figure 6 (right) shows a table with security methods to be executed in an access point. These methods might be either permanent or temporal. Temporal security methods are performed only during a certain period of time and are usually targeted to a group of a particular user. These rules are created based on the logical model described in the last section and usually comply of simple rules that relying parties with low computational and storage power can manipulate.

The role of security agent is to perform analysis of irregular access patterns and assign additional security methods to relying parties when necessary. Consider the following scenario. A volunteer is hired as technical staff of Pavilion I might be able to access most of the facilities in this Pavilion, but during the course of his work, he might need to access some facilities on Pavilion II. However, if the frequency of access for facilities Pavilion II increases, the system must alert the security guards to verify this irregular behavior. To achieve this, the security agent can introduce a permanent security method to all access points located to Pavilion II, where the security merits are deducted from users at different zones when they are trying to access their facilities. Using this scheme, the system has increased control for access of users from different areas while maintaining its flexibility, and will not completely deny the access of users from some sensitive areas.

Another possible scenario is to monitor general users' access after a VIP user or athlete accesses a particular room. Security agent can dynamically assign a new rule that deduct the security merits from staffs without special permission and trying to access zones after the access of a particular VIP user or athlete. Similarly, if a particular staff follows the trails of a VIP user or athlete, a new dynamic rule can be added to monitor this particular user and the cumulative effect of deductions might deny its access or alarm the system because of a potential assassination for example.

The ability of creating additional security methods dynamically not only allows greater flexibility to the security framework, it can also increase the system's overall performance. In this scenario, as it incorporates a large sum of users with higher access levels, possible attacks cannot be neglected. However, it is neither reasonable to deny all accesses of these classes of users as they might require a temporal access to some types of facilities. Using the concept of security merits, access could still be granted, but post-condition applies and this might give a preemptive alert to security guards. On the other hand, dynamic rules can be added during system usage, which will increase the system's security level while maintaining the overall performance level for general users.

5. Future Works

Using on data mining and visual mining technology, possible intrusions data can be gathered and analyzed, and the results could be represented graphically to show security officers where are the 'hot spots' on the screen. The new hidden rules discovered by data miners help defining new or modifying existing security policy. On the other hand, based on analysis of past usage patterns, new security rules could be immediately retrieved and incorporated to the system, and the level of criticalness of the rules can be derived by the pre-defined logical model.

Another add on to the current work could be RFID tags replacing contact ID cards for users. And for the security personnel they can be equipped with a wireless device with screen display showing the location information. With the development of wireless network [WIFI] technology, communications with security guards based on short messages through GSM network could be replaced by a more sophisticated technology. Security guards, holding a personal digital assistant with wireless network connections, have access to latest graphical information of different zones and access points.

6. Conclusion

In this paper, we have described a physical access control system based on SMITH mechanism. Using this approach, irregular or suspicious user access behaviors can be analyzed in real-time, while maintaining system flexibility and scalability. Access permission is granted based upon successful verification of all these rule validations. Unlike traditional access control system, a post-condition might apply, meaning an access might be granted but the user will be closely monitored. Based on the security merits concept, when the system alerts some possible irregular actions, security merits will be deducted but access is still granted to a user until its cumulative effect denial user's access.

Based on this approach, validations rules can be added or removed dynamic based on this approach. Instead of performing a whole set of security checks to all users, a set of additional validations are performed on targeted users. Hence a balance between performance and security can be achieved in a distributed agent environment.

Appendix

Sample security rules developed using Equation 2.

General Parameters use in the following equations:

t_c: current time, t_l: last access time
t_o: start time of validation period, t_e: end time of validation period
s_{card}: security merts stored in card
l_{def}: defined security level
$ac(u_z)$: function that returns the current access point of user u_z
$al(u_z)$: function that returns the last access point of user u_z

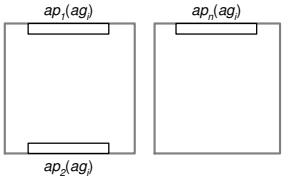
Check time period constrain (critical)

Description: Check if card is accessed during authorized time range $[t_o..t_e]$

$c(\{v_c\}) = (t_c < t_o)\ or\ (t_c > t_e), \{v_c\} = \{t_c, t_o, t_c\}$
$b(\{v_b\}) = |s_{card}|, \{v_c\} = \{s_{card}\}$
$r(l) = l_{def}, l = l_{def}$

Check irregular repetitive access

Description: Check whether the set of access points $ap_{1..n}$ belonging to the group ag_i is repetitively activated between the time period t_o and t_e

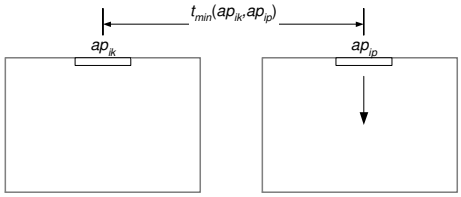


$c(\{v_c\}) = (ap_c \in ag_i) \cdot (ap_l \in ag_i) \cdot \left[(t_c - t_l) > (t_e - t_o)\right],$
$\{v_c\} = \{ap_c, ap_l, t_e, t_o, t_c, t_l\}$
$b(\{v_b\}) = \left[(t_e - t_o) - (t_c - t_l)\right] \cdot NumAc(CardID, ag_i, t_o, t_e) \cdot Status(CardID),$
$\{v_b\} = \{UserID, ag_i, t_e, t_o, t_c, t_l\}$
$r(l) = l_{def}, l = l_{def}$

CardID: Unique identification number of the card
ag_i: access group to be verified
Status(*CardID*) : Card status function
NumAc(*CardID*,ag_i,t_o,t_e): Function that returns the number of access of a particular time over the access points belonging to ag_i within the time period $[t_o..t_e]$

Check minimum access time between two points

Description: Check whether the access time of two access points ap_{ik} and ap_{ip}, is shorter than average minimum time required from moving between points ap_{ik} and ap_{ip}.



$c(\{v_c\}) = \phi\left(t_a(ap_{ip}) - t_a(ap_{ik}) > t_{\min}(ap_{ip}, ap_{ik})\right), \{v_c\} = \{ap_{ip}, ap_{ik}\}$
$b(\{v_b\}) = t_{\min}(ap_{ip}, ap_{ik}) - \left(t_a(ap_{ip}) - t_a(ap_{ik})\right), \{v_b\} = \{ap_{ip}, ap_{ik}\}$
$r(l) = l_{def}, l = l_{def}$
$t_{min}(ap_{ip}, ap_{ik})$: Function that returns average minimum time required from going between points ap_{ik} and ap_{ip}
$t_a(ap_{ix})$: Function that returns the access time of point ap_{ix}

References

[ASS04] Smart Card-Based Irregular Access Patterns Detection System – António Leong, Simon Fong, Shirley Siu, Proceedings of the 2004 IEEE International Conference on e-Technology, e-Commerce and e-Service (EEE'04),
[SCT] Smart Card Overview - http://java.sun.com/products/ javacard/smartcards.html
[GEM] ID & Security Solutions from Gemplus http://www.gemplus.com/pss/id_security/
[IDC] IDentiPASS Plus Access Control http://www.identicard.com/products/idpassplus.htm#
[SCSITE] Digital Persona: Developer Tools http://www.digitalpersona.com/developer/exisDevSol/physAccess.html#count
[SCA] Contactless Technology for Secure Physical Access: Technology and Standard Choices – Smart Card Alliance, October 2002
[MAG] Magnetic Stripe Cards and Card Readers - http://www.amtel-security.com/Components/MagStripe.htm
[WIE] Access Control: Wiegand Cards & Card Readers - http://www.amtel-security.com/Components/WiegandCard.htm
[INTOS] Integrating Access Control with Other Systems - The new necessity - Andrew (Andy) Bulkley – Information Storage + Security Journal , May 6, 2004/9/19
[TAPS] Trends in Access Products & Systems - Timothy O'Connell – Security Distributing and Marketing
[IDTHF] Biometrics: A Future Identity Solution? High-tech scanning of fingers and eyeballs may be the perfect weapon against identity theft, Identify Theft, September 2003
[EAG05] Macau 2005, 4th Asian Games - http://www.east-asian-games2005.com/en/
[WIFI] Wi-Fi Alliance – http://www.wi-fi.org