Security in Physical Environments: Algorithms and System for Automated Detection of Suspicious Activity

Robert P. Biuk-Aghai, Yain-Whar Si, Simon Fong, and Peng-Fan Yan

Business Intelligence Group
Department of Computer and Information Science
Faculty of Science and Technology
University of Macau, Macau
robertb,fstasp,ccfong@umac.mo, franciswing@163.com

Abstract. Secure physical environments are vulnerable to misuse by authorized users. To protect against potentially suspicious actions, data about the movement of users can be captured through the use of RFID tags and sensors, and patterns of suspicious behaviour detected in the captured data. This paper presents four types of suspicious patterns, algorithms for their detection, and the design and implementation of an integrated system which uses our algorithms for the detection of suspicious patterns in access data of physical environments.

Keywords: access pattern, user behaviour, secure physical environment, detection, generation.

1 Introduction

In the wake of increased terrorist and criminal activity over the past decade, the security of physical environments has become an increasingly important topic. In many parts of the world the use of video surveillance technology has become widespread for detecting security breaches [1]. Moreover, electronic and information technology has been used to restrict access to physical environments. For example, smartcard-based access control systems have been used over the past decade to automate the identification and authentication of access to restricted physical environments such as buildings, rooms, etc. More recently, RFID (radio frequency identification) has enjoyed quick and widespread adoption in the security domain. RFID allows a person or object to be tagged with a unique identifier that can be wirelessly sensed when the RFID tag enters the range of an RFID sensor.

The low cost of RFID equipment coupled with the convenience of a wireless mode of operation and a fast detection rate makes this technology particularly suited for security applications. In 2005, the US Department of Homeland Security (DHS) announced the distribution of 40,000 RFID-based access cards to its employees and contractors to control access to both physical environments

and computer systems. Other US federal agencies also are making use of similar technology to strengthen the security of their physical environments, and this technology is being adopted by governments and private agencies around the world.

Using RFID technology allows the physical access of people to secure areas to be controlled. Moreover, given enough sensors in a secure environment, it also allows the movement of people within the environment to be tracked. Current use of this technology, however, is mainly restricted to disallow unauthorized access. Once a person has gained access to a secure physical environment, the actions of that person within that environment are usually not further monitored other than detecting outright breaches of security, e.g. through video surveillance. It is possible, however, that a given person within a secure environment behaves in a way that does not constitute an outright security breach, but that could be considered suspicious behaviour. Other security problems could arise if data from a valid RFID tag is surreptitiously obtained (RFID sniffing) and used to create a clone of the RFID tag which can then be used in RFID spoofing, replay attacks, or denial of service [2, 3]. If such suspicious behaviour could be detected, security personnel could be alerted to monitor the suspicious person closely to determine whether a security breach is about to be committed.

Extensive research on intrusion detection systems (IDS) for computer networks, which covers suspicious access detection (SAD), has paralleled the fast proliferation of Internet development and penetration. On the other hand, IDS and SAD for physical access security became an important worldwide concern in recent years after the September 11 disaster in the USA. Considering the characteristic of suspicious access detection, there are certain similarities in its application both in the digital and physical realm. The required techniques of SAD for the physical realm could be based on the ones developed for the digital realm. In computers, activities can be captured easily and comprehensively, resulting in large amounts of activity data. Data analysis and mining algorithms can be applied to this data to discover abnormal and suspicious activities among the considerable volume of data. Recent development of RFID technology enables tiny contact-less tags for physical object tracing and tracking. Practical implementation of object movement identification and registration becomes feasible, simple and convenient.

The research reported here has developed techniques for extracting and analysing information on suspicious patterns of movement from people's access logs in physical SAD, and developed algorithms, methods, and tools for analysis and visualization of data related to physical object movements, especially user behaviour patterns that are suspected to be security threats. The derived result can be used for early warning of suspicious activities in a closely monitored environment equipped with multiple sensors.

Intrusion Detection Systems (IDS) for computer networks and applications has been a popular research topic during the past decade. Various data mining techniques have been applied and proven to be effective, including association and frequent episode [4, 5], meta learning [5], classification [4] and clustering [6].

Research about location sensing of people or objects using radio frequency identification technology has been conducted recently. LANDMARC [7] is a location sensing prototype system that uses active RFID tags for locating objects inside buildings. Isoda et al. [8] proposed a user activity assistance system that employs a state sequence description scheme to describe the user's contexts. Willis and Helal [9] proposed a navigation and location determination system for the blind using an RFID tag grid. Leong et al. [10] developed a logical mathematical model to formulate a knowledge base of suspicious and irregular actions. Based on this model, they proposed a real-time suspicious access pattern detection prototype which allows rapid alert and reaction to irregular behaviour.

Based on the concepts and methodologies described above, we have modelled the physical environment developed an intrusion detection model for physical environments. Given the lack of availability of secure access event data, we have developed an access event generator for physical environments. The remainder of this paper is structured as follows. In Section 2, we outline four types of suspicious patterns we detect. In Section 3, we give an overview of the system we developed, and in Section 4 we discuss the design and implementation of our simulated access event generator. In Section 5 we discuss related work, and finally draw conclusions in Section 6.

2 Suspicious Pattern Detection

Here we describe the semantics of four suspicious patterns and corresponding method for detecting these using concrete algorithms. Our proposed techniques detect a person's suspicious behaviour by analysing movement patterns and identifying potential security threats in a secure physical environment. Suspicious behaviour consists of a collection of suspicious patterns. Each of these patterns is a sequence of actions performed by a person that may be completely legitimate when the level of analysis is a single event. However, when these events are combined over time and viewed together as a sequence they give rise to certain kinds of suspicion. The exact definition of the suspicious movement of people usually varies from one environment to another, and subjectively depends on the security requirements of each different situation. Given an existing physical environment with surveillance sensors installed, access events are captured and stored in a database together with related access right policies. Our detection functions access this data and evaluate it against administrator-defined thresholds for detection of suspicious patterns using concrete methods. We define following four suspicious patterns:

1. *Temporal pattern*: an unusually long period of stay by a person in a given area.
2. *Repetitive access pattern*: unusual repetitive accesses within a given period of time
3. *Displacement pattern*: consecutive accesses to distinct but distant neighbouring locations within an unusually short period of time.

4. *Out-of-sequence pattern*: consecutive accesses in an undefined sequence.

To detect these patterns in collected data, the following algorithms can be used in an existing physical environment that has surveillance sensors installed. Parameters for the detection algorithms are presented in Table 1.

Table 1. Parameters for detection model

Parameter	Description
$event_i$	ith access event
cid	$cid = cardID(event_i)$, access card ID of ith access event
AP_i	$AP_i = accessPoint(event_i)$, access point of ith access event
$repThreshold$	normal maximum allowable number of repeated accesses
$repAccMinDuration$	normal minimum allowable duration for a sequence of repeated accesses

Detection of Temporal Pattern: Let $timeStamp(AP_i, cid)$ be the function which returns the timestamp of the ith detected access point of the person holding card cid. Let $location(AP_i)$ be the function which returns the location of the ith access point, and let $maxStay(loc, cid)$ be the function which retrieves the predefined maximum duration that the person holding card cid is allowed to stay at the location loc. We define the algorithm for detecting temporal patterns as follows:

for all new detected $eventi$ **do**
  $t_{pre} = timeStamp(AP_{i-1}, cid)$
  $t_{cur} = timeStamp(AP_i, cid)$
  $t = t_{cur} - t_{pre}$
  $t_{max} = maxStay(location(AP_{i-1}), cid)$
  if $t_{max} < t$ **then**
    pattern = "Temporal"
  else
    pattern = "Normal"
  end if
end for

Detection of Repetitive Pattern: The detection of the repetitive pattern focuses on access events detected from a pair of access points (sensors) installed at two opposite sides of a door or entrance. In addition, two conditions must hold for a repetitive pattern: (1) the total number of repeated accesses should be greater than the predefined threshold, and (2) the total time spent during the repeated accesses must be shorter than the minimum allowable duration for a sequence of normal repeated accesses. First, the system derives the total number of repeated accesses from the last detected access event. For instance, two repeated accesses are detected from the sequence $AP_{i-4} \rightarrow AP_{i-3} \rightarrow AP_{i-2} \rightarrow AP_{i-1} \rightarrow AP_i$, where AP_i is the ith detected access point. Note that $AP_i = AP_{i-2} = AP_{i-4}$, and $AP_{i-1} = AP_{i-3}$. Let $repAccessCount(AP_n)$ be the function which counts the

total number of repetitive accesses for access point AP_n. For the example of the above sequence, $repAccessCount(AP_i)$ is equal to 2. Let $timeSpent(AP_x, AP_y)$ be the function that returns the time spent by the person when accessing point y after accessing x. Therefore, the total time spent by the person for the previous access sequence can be denoted as $timeSpent(AP_{i-4}, AP_i)$. Based on these functions, we define the algorithm for detecting repetitive access patterns as follows:

for all new detected $event_i$ **do**
  if $(repAccessCount(AP_i) >= repThreshold)$ and
  $(timeSpent(AP_i, AP_{2(repAccessCount(AP_i))}) < repAccMinDuration)$ **then**
    pattern = "Repetitive"
  else
    pattern = "Normal"
  end if
end for

Detection of Displacement Pattern: Let $minMove(AP_{i-1}, AP_i)$ be the function which returns the minimum time required to travel from $(i-1)$th access point to ith access point. We define the algorithm for detecting displacement patterns as follows:

for all new detected $event_i$ **do**
  $t_{pre} = timeStamp(AP_{i-1}, cid)$
  $t_{cur} = timeStamp(AP_i, cid)$
  $t = t_{cur} - t_{pre}$
  $t_{min} = minMove(AP_{i-1}, AP_i)$
  if $t < t_{min}$ **then**
    pattern = "Displacement"
  else
    pattern = "Normal"
  end if
end for

Detection of Out-of-Sequence Pattern: A pattern is considered to be out-of-sequence when it is detected that a person attempts consecutive accesses to two distinct locations whereby the second location is unreachable from the first one. Let $isNeighbor(AP_{i-1}, AP_i)$ be a Boolean function which returns true if AP_i can be reached from AP_{i-1}. We define the algorithm for detecting out-of-sequence patterns as follows:

for all new detected $event_i$ **do**
  if $isNeighbor(AP_{i-1}, AP_i)$ **then**
    pattern = "Normal"
  else
    pattern = "Out-of-sequence"
  end if
end for

Using the above four detection algorithms, a security system may decide to raise an alarm when a suspicious access pattern is detected. However, in some situations a sequence of access events may not be considered suspicious as its degree of suspicion does not exceed pre-defined threshold values. For instance, the total number of repeated accesses by a person may not exceed the limit and hence the system may not raise the alert. In such cases, the system may not be able to detect cases of slight suspicion. The prediction of future possible suspicious access patterns would be a straightforward extension of our algorithms.

3 System Design Overview

We have designed an integrated system for the capture of RFID sensor data, generation of simulated physical access data, training of our detection model, and real-time detection of suspicious access patterns. This system design consists of five modules arranged in three layers, as shown in Fig. 1.

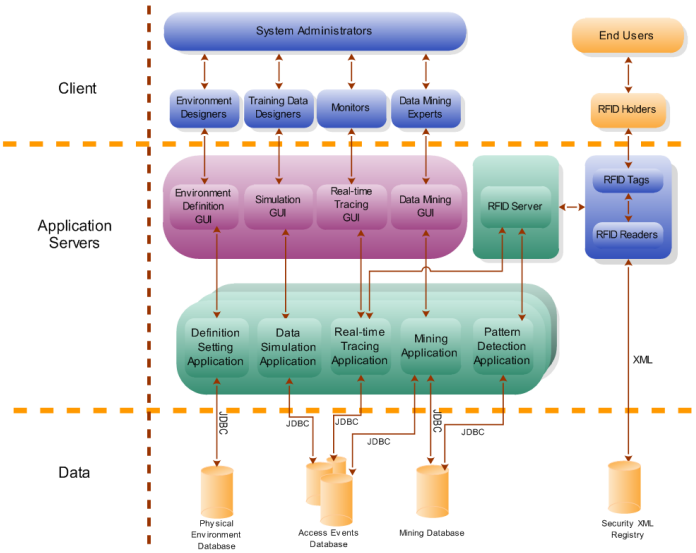

Fig. 1. System structure

3.1 System Layers

Data Layer: This layer consists of several databases. The Physical Environment Database stores data defining the physical environment, such as building layout, access point location etc. The Access Events Database stores the data about RFID access events, including real data obtained from RFID sensors installed in

the physical environment, or data generated by our simulator component. The Mining Database stores mined models and detection rules. It also records parameters for training the models. The Security XML Registry stores the security requirements.

Application Server Layer: This layer consists of several applications for defining the physical environment, generating simulated access event data, training detection models, real-time monitoring and tracking of suspicious patterns, and mining patterns from the captured access event data. For ease of use by non-expert security personnel, these applications are designed with graphical user interfaces. They comprise offline applications for creating detection models, and online applications for processing real-time access event and identifying suspicious patterns based on the models from the offline applications.

Client Layer: The client layer comprises two main parts. One part consists of the RFID holders, i.e. the persons whose movements are monitored through our system and who each hold an RFID tag that is the source of data when sensed by the RFID sensors placed in the environment. The other part consists of the users of our system's applications who are system administrators in charge of different aspects of the whole system's operation: the Environment Designer is responsible for defining the physical environment by setting environment-related parameters; the Training Data Designer is responsible for designing the simulation of physical access events by setting probability-related parameters; the Monitor is responsible for monitoring the real-time animation of real-time access events or simulated access events and looking for suspicious patterns among the users' actions; the Data Mining Expert is responsible for defining useful and efficient models/algorithms for detecting suspicious patterns.

3.2 System Modules

There are five modules: The *Physical Environment Module* is used to define the locations within the physical environment including the RFID sensors. The *Data Simulation Module* utilizes the defined physical environment to simulate access events and generate access event data based on user defined parameters. The *Real-time Tracing Module* visualizes RFID holders' actions in the physical environment. It can also visualize the simulated and historic data. The *Data Mining Module* extracts detection rules/models from given historical data. The *Pattern Detection Module* is used for detecting suspicious access patterns in real time.

Among the above modules, the data simulator is responsible for simulating access events with respect to specified parameters. For the sake of simplification, we adopt a fixed floor plan and simulate the movement of people from one area to another. We have devised two algorithms for generating paths for the simulator. The optimum path finding algorithm (VOP) is used to generate shortest paths from a given starting point to a target point, where the distance is regarded as the evaluation measure. A Random Path finding algorithm (VRP) is used to generate a path randomly so that random movement of RFID holders can be mimicked in the system. *User portion* (the ratio of users behaving suspiciously)

and *probability* (likelihood of a certain kind of suspicious action) parameters are used to describe the ratio of the suspicious pattern access events to be generated during the simulation.

4 Prototype System Implementation

Based on the design from Section 3, we have implemented a prototype system. This section briefly introduces our implementation. Due to space limitations, we only illustrate some of its many functions here. Initially the Environment Designer defines characteristics of the physical environment. An example of the physical layout of a given environment is shown in Fig. 2, here of a portion of the US White House. The environment consists of corridors, rooms, doors, passage ways, etc. Labelled with numbers at each door and on some of the walls are locations of RFID sensors that are installed in the physical environment. The Environment Designer records all the relevant information about areas (rooms, corridors), entrance ways, connections between areas, locations of RFID sensors etc. The environment design also includes the definition of parameters related to suspicious access events, illustrated in Fig. 3.

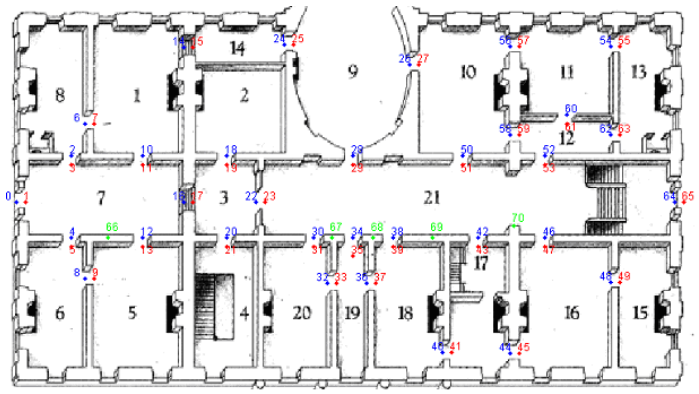

Fig. 2. Example physical environment layout: US White House (image is in the public domain)

In the example of Fig. 3 the environment designer is adjusting the minimum traversal time between an access point (RFID sensor) and one of its neighbours, as used in the detection of Displacement patterns. Given the information about the environment and the defined pattern parameters, the pattern detection is able to determine when a suspicious pattern has occurred, as explained in Section 2 above.

Once the environment is in operation, access events are captured from all connected RFID sensors and stored in the Access Events Database. Below is a

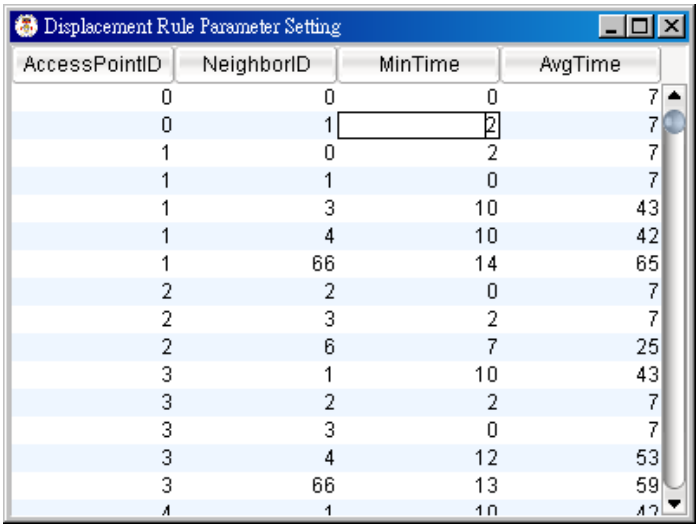

Fig. 3. Parameter settings for displacement pattern at a given RFID sensor location

sample of some raw access event data, showing access point IDs, access card IDs and timestamps:

```
66  23  2009-05-05 12:20:07
66  78  2009-05-05 12:21:01
19  71  2009-05-05 12:21:18
```

Once the pattern detection is in progress, it will search for and display any suspicious patterns found in the access event data. Fig. 4 shows an example of a detected Displacement pattern, with a panel for rule selection and configuration in the right of the same window. This allows rules to be customized at run-time in response to observed behavioural patterns.

We have evaluated our detection algorithms using our own simulated data. Not surprisingly, the detection works flawlessly. A more meaningful evaluation would use real data from an actual secure physical environment. However, given the sensitive nature of these environments and the data captured from them, we have to date not had the opportunity to get access to such data and thus evaluate our algorithms more fully. We welcome collaboration with any organization that would be interested in applying our research to their security-related data.

5 Related Work

The closest related work is in the area of intrusion detection systems (IDS). An IDS is designed as software or hardware for detecting unwanted attempts at accessing, manipulating, or disabling of computer systems, mainly through a computer network such as the Internet. In recent years a large number of

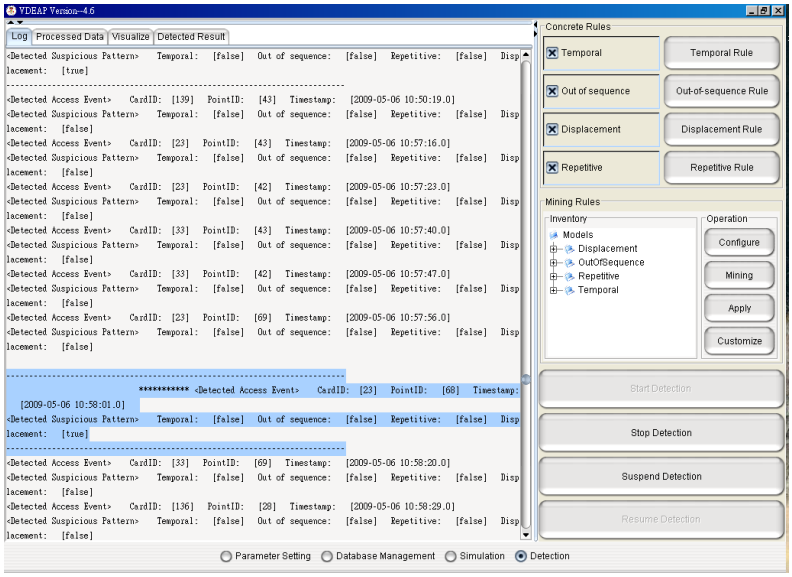

Fig. 4. Detection of a suspicious pattern among the access event data

IDS have been developed to address specific needs [11]. The most commonly used models for current IDS are host-based, network-based, and protocol-based IDS. In host-based IDS, there is a unique host used to detect the intrusion by analysing data packets that travel through that host. This host comprises an agent which identifies intrusions by analysing system calls, application logs, file-system modifications and other host activities and state. OSSEC [12] is an example of host-based IDS, as it performs log analysis, integrity checking, windows registry monitoring, rootkit detection, time-based alerting and active response. In network-based IDS, a computer network intrusion detection system (NIDS) is usually installed by connecting to a hub, network switches or network taps, and is an independent platform which keeps track of network traffic data. The data from the computer network is monitored against a database and the NIDS flags those which seem to be suspicious. The audit data from single or multiple hosts are also used to detect intrusion signs. Snort [13] is an example of NIDS that performs packet logging and real-time traffic analysis on IP networks. Protocol-based IDS (PIDS) [14] usually consists of a system or agent located at the very front end of a server to monitor and analyse the protocol which is used to communicate between a connected device and the server. PIDS monitors the dynamic behaviour or states of the protocol. Depending on the requirement, two or more types of IDS are combined together to construct a hybrid intrusion detection system.

The majority of IDS use either anomaly or misuse detection models. The principle of the anomaly detection model is to look for anomalous behaviour or deviations from the predefined baseline. Although this model is effective in

detecting unknown intrusions and new exploits, anomaly detection can result in a high false positive rate. For example, Qiao et al. [15] have discussed an anomaly intrusion detection method based on HMM. The intrusion detection system monitors the call trace of a UNIX privileged process, and passes it to a HMM to obtain state transition sequences. Preliminary experiments prove the state transition sequences can distinguish normal actions and intrusion behaviour in a more stable and simple manner. The misuse detection model has knowledge of suspicious patterns of behaviour and looks for activities that violate the standard policies. Misuse detection models have a lower false positive rate. Kumar and Spafford describe a generic model of matching that can be usefully applied to misuse intrusion detection [16]. Their model is based on Coloured Petri Nets.

IDS can collect a large amount of data without sufficient means to merge the data so as to extract the context for detecting attacks. Intellitactics Security Manager [17] allows users to prioritize and prevail across the full range of security threats in real time. It can capture and monitor real-time event activity and translate event codes into easy to understand terms. It can analyse complex security situations with customizable web-based reports, correlate data and prioritize threats. In another case, audit data analysis and mining (ADAM) [18] IDS used tcpdump to build profiles of rules for classification. ADAM adopts data mining technology to detect intrusions, including the combination of association rule mining and classification methodologies. Lee et al. [4] developed a data mining framework for building an intrusion detection model, which consists of programs for learning classifiers, association rules for link analysis and frequent episodes for sequence analysis. Portnoy [6] proposed an intrusion detection model with unlabelled data using clustering (unsupervised learning). The model can detect a large number of intrusions while keeping the false positive rate reasonably low.

6 Conclusion

In this paper, we describe a model for detecting suspicious patterns within a large volume of access events in secure physical environments. We have defined four types of suspicious patterns that may occur in common physical access environments, namely Temporal, Repetitive, Out-of-Sequence and Displacement, respectively. Using characteristics of each type of pattern we have defined algorithms for detecting these among a large set of logged access event data. Our presented integrated system allows the definition of a secure physical environment's features, the configuration of parameters related to suspicious patterns, and the detection of these patterns in collected data. For training purposes, an integrated simulator can generate large volumes of realistic access data. The use of our presented algorithms and system design can be of great use in providing an additional level of security to large physical environments in which the use of video surveillance alone is not sufficient to determine whether a sequence of valid actions performed by its users can be considered legitimate in the context of the user and location concerned. Our work is thus of particular relevance to the use in military installations, government facilities and other high-security

locations. We welcome contact by organizations wishing to apply our techniques in their secure physical environments.

Acknowledgments. This research was funded by the Research Committee, University of Macau under grant number RG076/04-05S/BARP/FST.

References

1. US Department of Justice: CCTV: Constant cameras track violators. National Institute of Justice Journal **249** (July 2003) 16–23
2. Thornton, F., Haines, B., Das, A., Campbell, A.: RFID Security. Syngress (2006)
3. Cook, D., Holder, L.: Graph-based data mining. IEEE Intelligent Systems **15**(2) (2000) 32–41
4. Lee, W., Stolfo, S., Mok, K.: A data mining framework for building intrusion detection models. In: IEEE Symposium on Security and Privacy, IEEE Press (1999) 120–132
5. Li, Q., Xiong, J., Yang, H.: An efficient mining algorithm for frequent pattern in intrusion detection. In: International Conference on Machine Learning and Cybernetic, IEEE Press (2003) 138–142
6. Portnoy, L.: Intrusion detection with unlabeled data using clustering. Undergraduate thesis, Data Mining Lab, Department of Computer Science, Columbia University (2000)
7. Ni, L., Liu, Y., Lau, Y., Patil, A.: LANDMARC: Indoor location sensing using active RFID. In: IEEE International Conference on Pervasive Computing and Communications, IEEE Computer Society Press (2003) 407
8. Isoda, Y., Kurakake, S., Nakano, H.: Ubiquitous sensors based human behavior modeling and recognition using a spatio-temporal representation of user states. In: 18th International Conference on Advanced Information Networking and Applications, IEEE Press (2004) 512–517
9. Willis, S., Helal, S.: A passive RFID information grid for location and proximity sensing for the blind user. Technical report, University of Florida (2004)
10. Leong, A., Fong, S., Siu, S.: Smart card-based irregular access patterns detection system. In: IEEE International Conference on e-Technology, e-Commerce and e-Service, IEEE Press (2004) 546–553
11. Brandenburg University of Technology: Intrusion detection systems list and bibliography. `http://www-rnks.informatik.tu-cottbus.de/en/node/209` (2004)
12. Trend Micro, Inc.: OSSEC manual. `http://www.ossec.net/main/manual` (2009)
13. Beale, J., Foster, J., Posluns, J., Russell, R., Caswell, B.: Snort 2.0 Intrusion Detection. Syngress Publishing (2003)
14. Wikipedia contributors: Protocol-based intrusion detection system. `http://en.wikipedia.org/wiki/Protocol-based_intrusion_detection_system` (2009)
15. Qiao, Y., Xin, X., Bin, Y., Ge, S.: Anomaly intrusion detection method based on HMM. IET Electronic Letters **38**(13) (2002) 663–664
16. Kumar, S., Spafford, E.: A pattern matching model for misuse intrusion detection. In: 17th National Computer Security Conference. (1994) 11–21
17. Intellitactics, Inc.: Intellitactics Security Manager. `http://www.intellitactics.com/int/products/securitymanager.asp` (2009)
18. Barbara, D., Couto, J., Jajodia, S., Popyack, L., Wu, N.: ADAM: Detecting intrusions by data mining. In: IEEE Workshop on Information Assurance and Security, IEEE Press (2001) 11–16