TOWARDS A CONCEPTUAL FOUNDATION FOR PHYSICAL SECURITY: CASE STUDY OF AN IT DEPARTMENT

S. AL-FEDAGHI[1] & O. ALSUMAIT[2]
[1] Department of Computer Engineering, Kuwait University, Kuwait.
[2] Information Technology Department, Ministry of Defense, Kuwait.

ABSTRACT
Protecting physical data, networks, and systems has become difficult, increasingly costly, and tougher to manage as technology and environments become more complex and dynamic. This paper presents a theoretical foundation for physical information technology (IT) security by developing a logical description based on a flow-based model. Within this model, a security machine is defined as a sequence of stages in which flow is identified and blocked in a multilevel blockage machine. The main focusses of the paper are the importance of having appropriate physical security in place, discussed with so-called onion/garlic models, and the notion of physical containment. The proposed representation is applied to an actual security plan for an IT department of a government ministry. The results suggest a viable approach to designing physical security strategies.
Keywords: Conceptual model, diagrammatic representation, physical access control, physical security, systems modeling language.

1 INTRODUCTION

According to the International Systems Security Engineering Association (ISSEA), the concerns of security engineering include understanding security risks, establishing security needs, developing security guidance, and establishing assurance. The field of security engineering is within the domains of enterprise engineering, systems engineering, software engineering, communications engineering, hardware engineering, human factors engineering, and systems administration [1]. According to one definition, 'security is a system condition in which system resources are free from unauthorized access and from unauthorized or accidental change, destruction or loss' [2]. The ISSEA classifies several types of security, including operations security, information security, network security, physical security, personnel security, administrative security, and communications security, among others [3].

Physical security is the oldest feature of security and a vital component of any overall security system [4]. The National Computer Security Center (NCSC) [5] defines physical security as 'The application of physical barriers and control procedures as preventive measures or countermeasures against threats to resources and sensitive information'. Physical security addresses events that can threaten an enterprise's resources and sensitive data. It covers all the devices, technologies, and materials related to perimeter, external, and internal protection (e.g. sensors, closed-circuit television, barriers, lighting, access controls) [6].

1.1 Importance of physical security

According to Gregg [4], 'Physical security is the point at which protection should begin. Physical security combined with technical security and an administrative control provides a holistic view of security'. St Sauver [7]-[8] that relatively little attention has been paid to the physical security of systems and networks; instead, the focus has been on 'online' information threats. Information security is concerned with 'how a person may penetrate the network using

ISSN: 2041-9031 (paper format), ISSN: 2041-904X (online), http://www.witpress.com/journals
DOI: 10.2495/SAFE-V9-N2-137-156

unauthorized means through wireless, software exploits or open ports. Security professionals with physical security in mind are concerned about the physical entrance of a building or environment and what damages that person may cause' [9]. Physical security may also (incorrectly) be viewed as 'someone else's problem' (e.g. the police). It is often overlooked because most organizations focus on technology-oriented security issues to prevent hacking attacks [10].

Hacking into network systems is not the only way that sensitive information can be stolen or used against an organization. Physical security must be implemented correctly to prevent attackers from gaining physical access and taking what they want. All the firewalls, cryptography, and other security measures would be useless if that were to occur [9].

Now more than ever, organizations need to defend against 'physical theft' [11]. In 2014, Coca-Cola Inc. announced that the personal information of about 74,000 employees might have been compromised when several laptops containing personal data were stolen from its headquarters [12]. Sensitive data can also be compromised if an attacker connects a flash drive to an unsecured computer [9].

1.2 Examples of physical security attacks

In physical security (in general), attackers *must be physically present* to achieve their goals and exploit physical vulnerabilities. Examples of actual physical attacks include fiber-optic cables being severed; the theft of data servers; thefts of safes from a locked room, including DVDs containing personal identifying information (PII) on millions of users [7]; theft of communication equipment, including routers and network cards; and access through an unmarked computer found in a spare room, connected directly to the Internet system [7]. System components can be stolen, including small parts such as memory and peripherals such as keyboards, mice, power bricks, and cables [7]. Some stolen physical equipment (e.g. cables, servers) can be crucial for *life-/safety-critical* systems (e.g. hospital and police connections) [7]. Perpetrators of reported security breaches and misuses of *physical access* to systems who bypass defences can include the following agents:

- A recently discharged employee whose system credentials have not yet been revoked;
- A person who finds a computer already logged in and uses it without the knowledge of the authenticated user; and
- A janitor who has privileged physical access, if not logical access [13].

According to a research study (Homeland Security Research Corporation [14]), the country of Kuwait, where the authors are based, spends more than 1% of its GDP on homeland security and public safety. According to the HSRC's latest report, *Global Homeland Security & Public Safety Market – 2015-2022*, 'in relative terms of GDP share, the Middle Eastern countries spend two to four times as much as the international superpowers. In addition, the urgency resulting from turmoil makes these regions some of the fastest-growing homeland security and public safety markets in the world' [14].

Great advancements have been made in new technologies for access control and security management, resulting in the development of security tools that serve a variety of physical security and related control functions [15], including physical access authorizations, physical access control, access control for output devices, physical access monitoring, visitor control, and access to records. Defence techniques include physical intrusion detection systems, alarm systems, man traps, RFID systems, and cable locks.

1.3 Problem in physical security

Protecting physical assets (data, networks and systems) has become *difficult to implement*, is *increasing in cost* and becomes *harder to manage* as technology and environments are becoming more complex and dynamic [9]-[10]. Organizations find themselves at risk of lawsuits because they can be held monetarily and criminally liable for a lack of due diligence in this area [9].

According to Huang [16], most developers and users *do not understand security*. Security cornerstones require: (i) *preventing* security breaches from occurring, (ii) *minimizing* the impact of a security breach, (iii) *detecting* vulnerabilities and security breaches, and (iv) *reacting* to vulnerabilities and security breaches quickly. Hutter [9] stated that 'Professionals that work in this space *do not always have a holistic understanding of physical security* because of specialized variables and components that are needed to secure an organization' (italics added). Schiavone [17] reported, in a study of data breach incidents, that the great majority of successful attacks resulted from an organization's poor understanding of security and poor deployment of its measures.

These problems have increased as a result of technological progress. Intruders likely prefer committing mixed cyber/physical intrusions over purely physical ones, including by blocking or saturating alarms, freezing video of digital cameras, remotely producing an access card, or creating a direct outage or damage to the power or elevators [18].

1.4 Methodology adopted in this paper

Physical security is typically based on *physical controls*, in the form of access control systems and monitoring, detection, and auditing systems. Access control relies on physical elements to keep unauthorized persons out of protected places. Properly managing these areas decreases criminal opportunities.

1.4.1 Adopting security as a process (machine)

A process is defined as 'a systematic series of actions directed to some end: to devise a process for homogenizing milk' [19]. According to blogger Franklin-Witter [20], 'Security is a process, not a point project (or product) … most security programs are not built and managed in a way that reflects this belief. … Security processes are not documented and often performed in an ad hoc manner'.

Ferraiolo [1] modelled an organization's security engineering (including physical security) in terms of processes 'to identify combinations of threat, vulnerability, and impact', using notions such as *events*: 'threat-vulnerability pairs that lead to unwanted outcomes'. The risk model involves: *assess threats* → *assess vulnerability* → *assess impact* → *assess security risk*. However, the method lacks a methodology for identifying the *events* involved.

Proposed in this paper is a security system based on the notion of security as a process *and* conceived as an *abstract machine*. The machine is an apparatus built on a synchronic order of five stages: creation, release, transfer, receipt, and process. The paper introduces a case study using such a machine to represent an actual IT department in Kuwait's Ministry of Defense (the workplace of the second author).

1.4.2 Flow-based design

A fundamental notion introduced in this paper is that of security-related *flows*. According to the adopted model, a security system is viewed in terms of flows and uses these flows as the

basis of a system-based representation. The paper focusses on flow-based physical security, specifically on *having appropriate physical flows in place*. Several *flows* are involved in this context, including flows with controlled *access* to the site by such entities as persons, equipment, and even trash involving entry and exit directions, evacuation, and alarm usage. Specifically, 'physical access control [of] the media is used to prevent unauthorized personnel from accessing the media' [3].

To keep the focus within reasonable limits for an academic paper, other physical *flows* involving power, electricity, gas, and water, despite being applicable to the flow-based approach, will not be discussed in this paper. The proposed solution is to monitor flow (e.g. of persons) to detect abnormalities and respond immediately to any irregular activity. This rapid response can be achieved with an information system that guards flow.

1.5 Contribution of this paper: A new approach to modelling physical security

Despite abundant advances in security-related technologies, the need exists for a theoretical platform that replaces informal and incomplete descriptions with systematic specification of a security system. *Systematization* refers to a logical and orderly depiction of occurrences of security-related events.

The next section briefly reviews some works on physical security modelling. Section 3 introduces a diagrammatic language (see Refs. [21]–[28]) as a vehicle for depicting security machines with illustrative examples that are a new contribution. Section 4 utilizes this language to model the security aspects of the case study. In Section 5, we analyse some of the modes reviewed in section 2 using the proposed new diagrammatic method.

2 RELATED WORKS

Developing security begins during the design phase, when *location* is a key consideration [4]. Presented in this paper is a theoretical foundation for physical security in the process of developing a logical description via a flow-based model. Within this *model*, an abstract security machine is defined as a sequence of stages in which a flow is identified and blocked in a multilevel blockage machine. Without loss of generality, a main focus of this paper is access control by *persons* in a cyber-physical system. The proposed model is built on a model that includes physical, human, and engineered elements and a description of how the world works (Simon [29], p. 119ff). Additionally, 'Human behavior needs to be included in assessments of insider threat, referring both to the abilities of the attacker as well as the weaknesses of others in enforcing security policies (e.g. in the case of social engineering)' [30]. A 'Model … can be used to analyse which sequences of actions in a network an attacker can exploit to achieve a certain goal. *Physical infrastructure* can be included in such models as well. This can be called the access-oriented approach' [31] (italics added).

'Modeling' access vulnerabilities at the physical level includes developing a security policy, monitoring areas, and installing a mechanism for limiting access to resources. The main ingredients are security barriers that prevent the unauthorized access or alteration of assets, and physical barriers combined with software barriers [32].

2.1 Onion ring model

The onion ring model is one of the basic physical security models. Like the layers of an onion (see Fig. 1), different layers of protection are built around the nested components of an

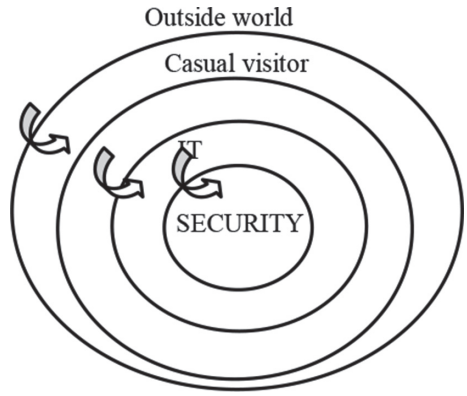

Figure 1: Onion model of physical security. Source: Partially redrawn from Forcht and Kruck [33]. Many similar publicly available diagrams can be found online (e.g. Wikipedia, 'Defense in depth').

information system. For example, Mobbs [32] emphasized the following issues: The building where equipment and files are located;

- The room where equipment and/or files are located;
- The computer hardware;
- The computers' operating system; and
- Files and data (including paper information).

An extension of the onion model is the so-called garlic model, which includes 'pockets' within the layers [33]. The purpose is to regulate physical access to areas, control panels, devices, cabling, and control rooms [34].

2.2 Architectural map-based model

The most common platform for modelling physical security is a description of the facility's physical *blueprints*. '*The first step is drawing a map* of the physical facility and identifying the areas and entry points that need different rules of access, or levels of security' (Philpott and Einstein [35]; italics added).

> Anyone who's built a house knows that blueprints are vital to the process. Construction doesn't start until the blueprints are drawn up, inspected by the builder and future homeowner, adjustments made and approval given … The same can be said of an *organization's security policy*. (Woodbury [36]; italics added)

Physical maps are typically adapted for use by police and fire departments to provide evacuation maps, wayfinding, facility management, and so on. They are also prepared as part of a security plan to reveal restricted areas as well as the physical security layouts that protect such areas. They are also utilized in deploying security objects and their physical layout inside the protected physical perimeter; however, security design needs a firmer foundation than a mere physical map and flowchart sketches.

To illustrate what we mean by an architectural map-based model, consider a typical warehouse security model and access plan [37] such as the one shown in Fig. 2, reflecting the locations of different access control and monitoring devices. Lincke [38] used the architectural platform to represent the sensitivity classifications assigned to different spaces of an IT department, as shown in Fig. 3, which includes three levels of security classification.

Imagine that you have some type of process, typically defined, as mentioned previously, in terms of a series of security-related actions (e.g. monitoring, granting access permission/blocking, preventing breaches, detecting, reacting); it is clear that an architectural map-based model is not suitable in this situation because it ties actions to specific physical locations, shapes, and areas of spaces in buildings. Entering and leaving a space are not necessarily accomplished through doors (e.g. entering a theatre stage). Doors themselves are as important as the spaces they separate.

The need exists for a theoretical platform that replaces informal and incomplete descriptions with a systematic specification of a security system. *Systematization* refers to a logical and orderly depiction of occurrences of security-related events. A *systematic* system would view spaces, subspaces, parking lots, doors, locks, search areas, monitoring cameras, fingerprint processors, entrance barriers, and gates uniformly as 'units' in the construction of security, much as one makes constructions from Lego pieces. Here, systematization is achieved by developing a *logical construction* of these flows instead of depending on their physical features. Pieters [30] suggested a different kind of space, sometimes termed an *infosphere* (see Fig. 4). According to Pieters [30], the *infosphere* requires thinking in terms of a space filled with informational things (i.e. pieces of information) instead of physical things.

2.3 Modelling cyber-physical systems

A current approach in security is to adopt top-down model-driven methods, including UML profiles for multifaceted security for cyber-physical systems requirements. This is because 'security of critical infrastructures … is a multi-faceted problem that requires an integrated

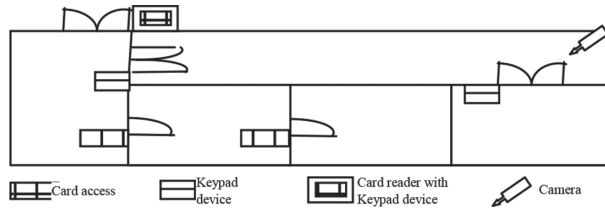

Figure 2: Typical security and access plan.

Source: Partial and incomplete, redrawn from Max [37].

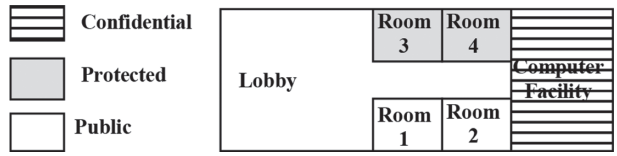

Figure 3: Sensitivity classifications.

Source: Partial and incomplete, redrawn from Lincke [38].

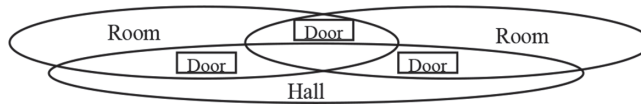

Figure 4: Infospheres. The ovals denote groups and their contained entities; they have no spatial meaning (partial, redrawn from Pieters [30]).

approach taking into account digital (i.e. cyber) security as well as physical security, which is strictly related to system protection against intentional threats of [a] physical nature' [39].

3 SECURITY AS A MACHINE

In developing a philosophy of information security, Vuorinen and Tetri [40]-[41] delineated an ontology to provide a theoretical base for understanding how an asset can be secured. They adopt the concept of a *machine* as proposed by Deleuze and Guattari [42], for whom 'a machine has a function, which is to produce and interrupt'. It is a machine of order, a machine that seeks to exclude noise and impurities [40]-[41].

We adopt the notion of security as a machine to facilitate security design. In our model, the machine is an abstract apparatus comprising five states (stages): creation, releasing, transferring, receiving, and processing of *s*. The paper applies such a concept in a case study of an actual installation.

The flow machine (FM) model was inspired by the traditional input-process-output model (Figs. 5 and 6), but the term 'process' used in that model seems overloaded, embedding several different phases of the life cycle of diverse things. For example, *creation* of a thing is not a 'process', since processing a thing implies that it already exists. It seems illogical to speak in terms of processing something that does not exist. Accordingly, creation is a preprocessing stage in a thing's life cycle. Additionally, *receiving* a thing is different from 'input'. It is possible for a thing to be output but not actually received (e.g. a transmitted packet reaching a port but failing to reach the input buffer). Furthermore, receiving does not guarantee 'input', since an arriving thing to the machine might be rejected.

Accordingly, we define a thing as what can be created, processed, received, released, and transferred in FM. Or, a thing is a byproduct that has been created, released, transferred,

Figure 5: Input-process-output model.

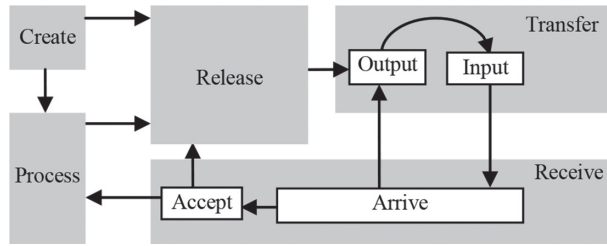

Figure 6: Flow machine.

received, and processed in FM. In computer science, the terms 'things' and 'thing-orientation' date back to the so-called 'ThingLab' (1979) and Self (1987), the programming language [43]. More recently, Wate*r*, a prototype-based language, links every XML tag with its top-level ancestor, 'Thing'. Thing-oriented programming is an approach that constructs software composed of things [43].

These stages in the 'movement' of *s* can be conceptualized in terms of *flows*. The resulting model is a diagrammatic schema that uses *flows* to represent a range of items that circulate or move through a system, for example, electrical, mechanical, chemical, and thermal signals, blood, food, concepts, pieces of data, activity, and so on. *Flows* are defined as what can be *created*, *released*, *transferred*, *processed*, and *received* as stages of FM, hereafter a *machine* (see Fig. 6).

- *Arrive:* A flow reaches a new machine.
- *Accepted:* A flow is permitted to enter the machine.
- *Received:* If arriving flows are also always accepted, *arrive* and *accept* can be combined as a single *received* stage.
- *Processed (changed in form):* The flow passes through some kind of transformation that changes its form but not its identity (e.g. compressed, coloured).
- *Released:* A flow is marked as being ready to be transferred (e.g. passengers cleared and waiting to board).
- *Created:* A new flow originates (is created) in the system (e.g. a data-mining programme generates the conclusion that an application should be rejected as input data).
- *Transferred:* The flow is transported somewhere outside the flow system (e.g. packets reaching ports in a router but still not in the arrival buffer).

The stages are mutually exclusive; that is, a flow in the *process* stage cannot be in the *created* or *released* stage at the same time. An additional *stored* stage can be added to any FM model to represent the storage of flows; however, storage is a generic, non-exclusive stage because there can be stored processed flows, stored created flows, and so on.

Figure 6 shows the structure of a flow system and its internal flows along with the six stages and transactions among them, assuming irreversibility of flow. A flow system may not need to include all the stages; for example, an archiving system might only use the stages *arrive*, *accept*, and *release*. Multiple systems captured by FM can interact with each other by triggering events related to one another in their machines and stages.

FM uses the following basic concepts:

- *Flow:* A thing is arriving or being created, released, transferred, accepted, and processed while flowing within and between machines. Flows can be material objects, concepts, actions, or information. Information communication involves creating, releasing, transferring, receiving, and processing information.
- *machines and submachines:* These are the flow's environments. A machine can have multiple flow systems in its construction, if needed. It can be an entity (e.g. a hospital and the departments within it; a person or class of persons such as nurses; a transistor; a logic gate, channel, wire, etc.).
- *Triggering:* This is an activation instrument (denoted by a dashed arrow). The mechanism of triggering can control the movement of flows in the system (e.g. in the process stage,

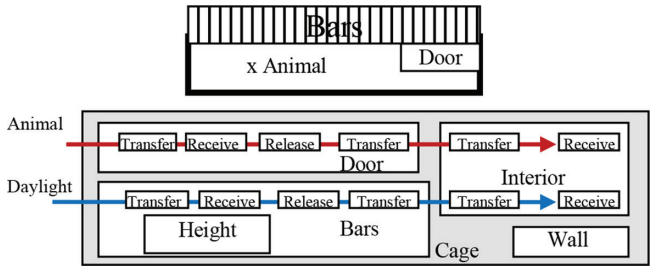

Figure 7: Physical depiction of a zoo animal cage (top) and its FM representation (bottom).

if a flow satisfies some condition, it is allowed to flow to the release stage). A flow is said to be triggered if it is created by another flow (e.g. a flow of electricity triggers a flow of heat) or is activated when a condition in the flow is satisfied (e.g. processing records x and y triggers the creation of record z in a flow system of records).

Example: The top of Fig. 7 depicts a cage in a zoo, including its physical boundaries: a wall, a door, and bars.
The lower part of Fig. 7 shows the *machines* of the cage that defines them as the bars, interior, wall, and door. The (surpassing) animal is a physical thing that flows through these machines. It flows to the interior from the outside through the door. The door and interior are flow machines. Note the 'logical' containment relationship between a machine (cage) and its submachine (door and interior). Other 'attributes' can be added, such as height as a submachine of bars, which in turn are a submachine of the cage. Bars represent a submachine of the cage, through which daylight flows to the interior.

The whole FM diagram is a machine with two types of flows. The 'operational semantics' of the machine are specified by means of *events*. This notion is exemplified in Fig. 8, where the *event* of the appearance of a new day triggers daylight that flows to the cage. (Note that, for simplicity, the stages of *create* and *process* of *day* are not enclosed in a box.)

This paper proposes that this 'logical' description of physical *space* can be used as a foundation for security specifications in machines, instead of basing the specifications on the architectural blueprints of spaces.

Multiple machines can exist in a machine, if needed. A machine can be a person, an entity (e.g. a company, a customer), a location (e.g. a laboratory, a waiting room), or a communication medium (e.g. a channel, a wire). Note that a thing, e.g. a person, can be conceptualized as a machine that contains such machines as a digestive system.

In an FM, every component is a machine or submachine of things that flow, including information, hardware, programmes, buildings, data, emotions, time, and events. The result is the ability to link machines and submachines to capture complex tasks. With everything treated as a machine, consistency emerges. Secure access to a physical place is no less important than secure access to an online asset. An FM diagram forms a scheme of machines and submachines that handle flows. *Security* is the protection of such an ecosystem of biotic and abiotic machines against any undesirable flow. The basic security machine filters incoming flow and prohibits objectionable things from flowing to a restricted area, as shown in Fig. 9.

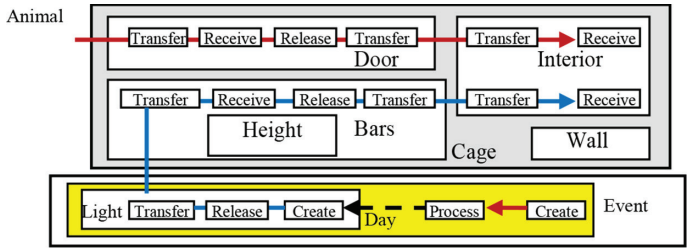

Figure 8: When daylight appears ('is created') and runs its course ('is processed') in time, this triggers the creation of light, which flows to the interior of the cage.

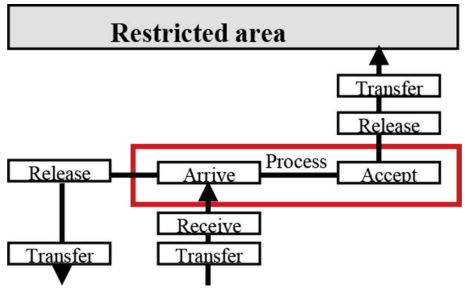

Figure 9: Security machine.

4 CASE STUDY: IT DEPARTMENT

To analyse the FM-based notion of security, we restrict our analysis, without loss of generality, to a physical intrusion into an IT department in an actual government ministry as a case study. The department has 160 employees assigned to diverse functions, including programming, systems analysis, network engineering operations with contractor employees in hardware and software maintenance. The main hardware includes high-end servers that run mostly Microsoft Server. The main software used includes Oracle Database. The department is also responsible for maintaining the ministry's network, built mostly of Cisco devices.

Currently, control over physical access in the department is facilitated through the positioning of security tools (e.g. ID card readers, fingerprint recognition machines, keys). Security monitoring involves several cameras positioned in select places, such as entrances. This paper reports on a study project to replace the ad-hoc system developed over years with a semi-automated system by using a design approach based on a systematic understanding of physical security.

Fig. 10 shows an FM diagram of the actual access control for department employees. First, a car enters the main gate of the department (circle 1 in the figure). Being received at the gate (2) triggers the driver to produce an ID (3; here, 'create' denotes the first appearance) that flows to the guard, who processes it (4) and returns it to the driver. Upon receiving his/her ID (5), the driver is triggered (6) to move the car to the barrier area, now in the *open* state (7). When the car leaves the parking lot (8), the barrier gate is closed (9). Note that the flow of a car implies the flow of the person(s) inside it.

Assume the person parks the car; he or she then moves on foot to the entrance of the building (10, 11). There, when received, the person enters his/her card (12) into a card reader

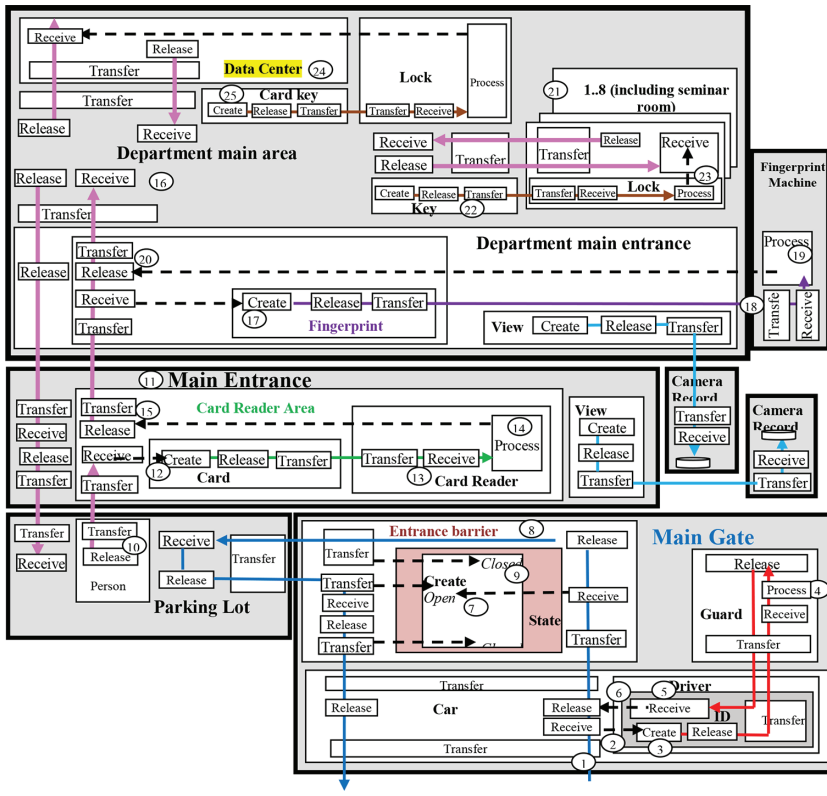

Figure 10: FM security machine of the case study.

(13), where it is processed (14) to permit releasing the person (15) to the main area's (16) entrance. The entry to the main area requires a fingerprint (17) that flows to a machine (18) that processes (19) it to permit (or deny) releasing the person from the main area (20). In this area, the person can enter one of eight offices (21) using a key (22) to unlock the room (23); or, he/she can enter the data centre (24) by using a special key card (25).

The schema depicted in Fig. 10 can be used as a base for constructing a security analysis of the department, its policies, and its plans. The analysis would include the nature and arrangement of the events and personnel related to access control. This would also involve materials and equipment-related activities (e.g. receiving or delivering materials), but, without loss of generality, we focus here on *the movement of personnel*. Additionally, information security, if the information is not in hardcopy form, will be studied in another paper that follows the same approach taken here. The plan involves establishing priorities for area protection. It includes how to handle intrusion: by positioning surveillance tools, monitoring areas, and coordinating and receiving instructions in case of intrusion.

The figure identifies five security machines to which flows are filtered:

1. The machine that controls the flow of cars into the parking lot. Note that we assume here that the flow of a car implies the flow of persons along with it, since the concern in this picture is with monitoring the movement of persons, not the movement of cars;
2. The machine that controls the flow of persons from the parking lot to the main entrance;

3. The machine that controls the flow of persons from the entrance area into the main area of the department;
4. The machine that controls the flow of persons from the main area of the department into various offices and the seminar room; and
5. The machine that controls the flow of persons from the main area of the department into the data centre.

Compare the FM security approach with the approaches of the modelling methods mentioned in the introduction, such as the warehouse security and access plan published by Edraw Max [37], which locates items such as 'card access' and cameras in the architectural drawing of the building (Fig. 2). This diagram is a simplified description that can easily be produced from an FM diagram. It should be noted, however, that the FM diagram of Fig. 2 is a *logical* description (e.g. not dependent on the physical characteristics or dimensions of a space); thus, it can provide a *conceptual* foundation for designing physical security and its control centre, as will be discussed in the next section.

5 MODELLING DYNAMIC ASPECTS

The FM representation provides a 'continuous' portrait of flows in the system. It is a static structure and a static map of things and their flows. Like network maps, it can serve as a reference to identify all possible flows, overlaid with all types of monitors and detection devices, in the context of access control.

The *dynamic* aspects of the static FM representation can also be developed. Dynamism is defined in terms of *events*. Events (occurrences) are things that can be created, processed, and so on. An operational semantics can be defined to designate the scheduling of events in the FM diagram, thereby specifying the thread of security control. Events have a different ontological status from the *static* FM diagram. When the diagram (or sub-diagram) comes alive (is actualized), it forms the content of an event: the diagram to *be executed* and the event *to execute*.

Each stage in the machine of FM representation can be counted as a separate event; however, at the administrative level, we are interested in "meaningful" events. For example, *a car enters the gate area* is represented in Fig. 11a by two potential stages: transfer and receive. This is a static specification of possible activity at the gate. In reality, the event content is the actualization of this possible activity in a certain period of time, as shown in Fig. 11b. The event in the figure (dark box) is the entry of a specific car at a certain time. Note that processing of time denotes that the event runs its course over time. The two stages create and process denote the event *itself,* beginning (create) and proceeding (process). The semantics of the shaded box are as follows: *An event appears (is created) and runs its course (is processed) over a certain period of time during which a car is transferred and received.* Of course, this event can occur many times; however, the description in (a) is a static, fixed specification. Note that, for purposes of simplification, we will use an event with a minimum specification of content.

Consider the events at the main gate of the IT department, shown in Fig. 12. There are 13 events to distinguish (shaded boxes), as follows:

Event 1: An automobile arrives outside the gate.
Event 2: The driver's ID is checked.
Event 3: The automobile proceeds to the entrance barrier.
Event 4: The automobile enters the entrance barrier area.

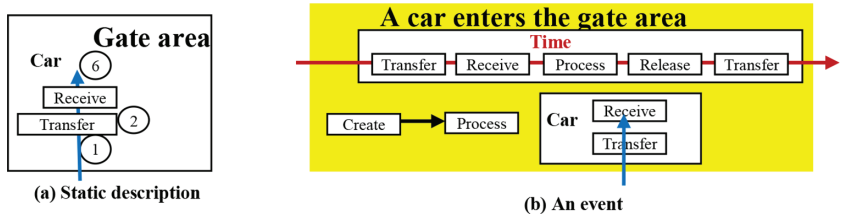

Figure 11: Actual event running its course through time at the main gate.

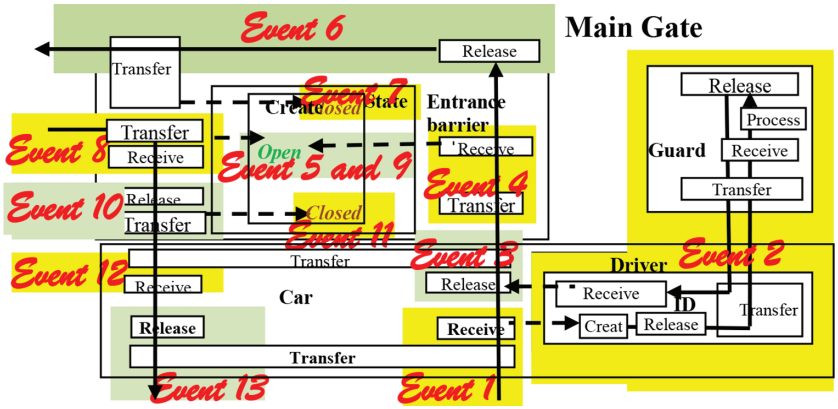

Figure 12: Possible events at the main gate.

Event 5: The entrance barrier is opened.
Event 6: The automobile proceeds beyond the barrier towards the parking lot.
Event 7: The entrance barrier is closed.
Event 8: The automobile arrives at the gate from the parking area.
Event 9: The entrance barrier is opened.
Event 10: The automobile leaves the entrance barrier area.
Event 11: The entrance barrier is closed.
Event 12: The automobile enters the area beyond the barrier.
Event 13: The automobile leaves the gate area.

Accordingly, the dynamic behaviours involved in either monitoring or executing the security policy can be specified in terms of acceptable *events*, including the following:

- The sequence (*event 1, event 3*) is not acceptable because it implies that *event 2* does not occur – that the driver of the vehicle did not give his/her ID to the guard.
- A state of alarm is triggered if event 5 (opening the barrier) does not occur within a certain period of time.
- The total number of entries by cars should equal the total number of exits.

Note that there are four types of events, as shown in Fig. 13:

- The two types of events of crossing the boundary of two machines (e.g. (1) transfer, receive and (2) release, transfer)

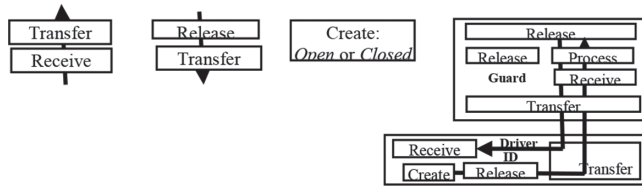

Figure 13: Types of events (with the same structure) at the main gate.

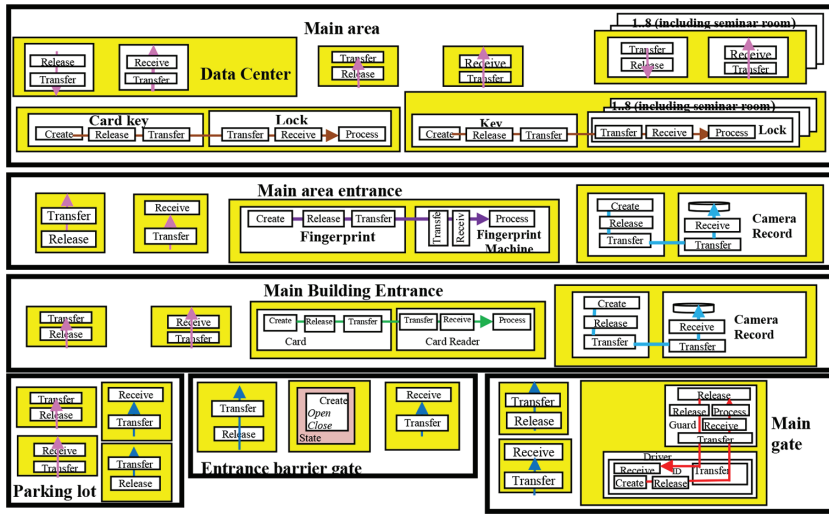

Figure 14: Events in the IT department.

- The event of changing the state between open and closed
- A secondary flow (with respect to the main flow of automobile) of approval of the driver's ID

Note that events are identified by factors such as triggers, independent flows, and change of submachines.

Figure 14 shows all of the events occurring within the IT department under study. The events are positioned according to their submachines; Fig. 15 shows the types of events.

An FM diagram can be utilized as a base for many types of projects, such as the building of a security monitoring system, simulation of certain situations (e.g. traffic flow), and theoretical analysis. It can also reflect the levels of security classification that are usually specified

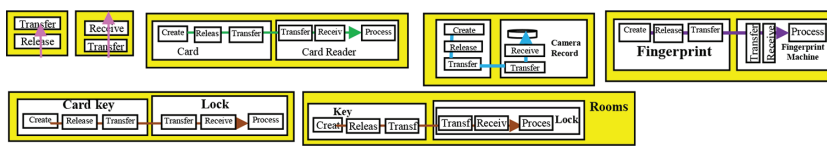

Figure 15: Types of events in the IT department beyond the main gate.

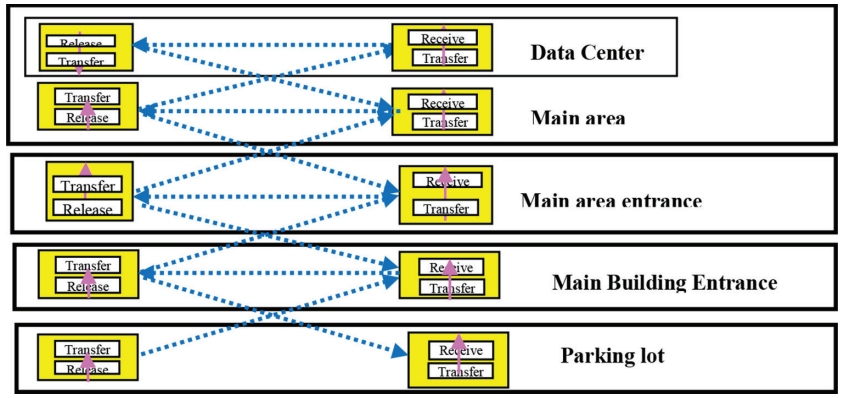

Figure 16: Entrance/exit events related to the data center.

by ad-hoc methods. Suppose we are interested in events of persons *entering into* and *exiting from* (assuming different doors) related to the data centre in the study case. In this case, the security system should monitor and record all events of these types. In designing such a security subsystem, it is clear that 10 cameras are needed, as shown in Fig. 16. The dotted arrows in the figure denote blind spots.

6 CONTRAST AND APPLICATIONS

Consider Lincke's [38] assignment of sensitivity classifications (Fig. 3), in which classifications are assigned to *different spaces* of the IT department, comprising *three* levels (confidential, protected, and public). This system does not include a mechanism to realize such distinctions. As shown in Fig. 17, different security classifications can be assigned security-related functions, regarded as events, such as checking one's ID or fingerprints. The classification is assigned according to the security events that 'block' flow of the event. The number and type of these events partially reflect the *strength* of the defence mechanism against intrusion. Accordingly, the FM approach provides a way to start the design of a security system, both automated and manual, from a simple beginning, such as Lincke's [38] assignment of sensitivity classification to spaces.

The FM description can accommodate all types of machines, including informational, physical, and virtual. Going back to Pieters's [30] infospheres discussed in the introduction (Fig. 4), according to Pieters [30],

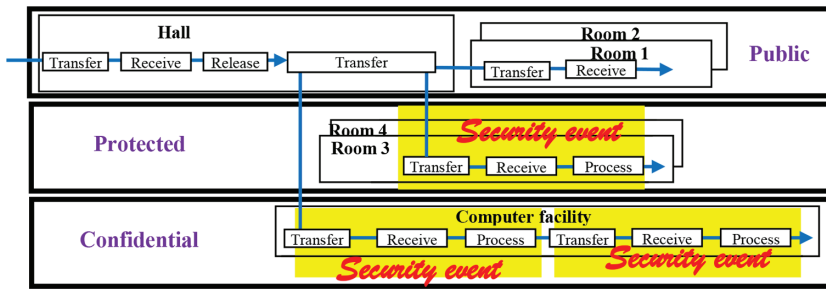

Figure 17: FM representation of Lincke's (2009) assignment of sensitivity classification (see Figure 3).

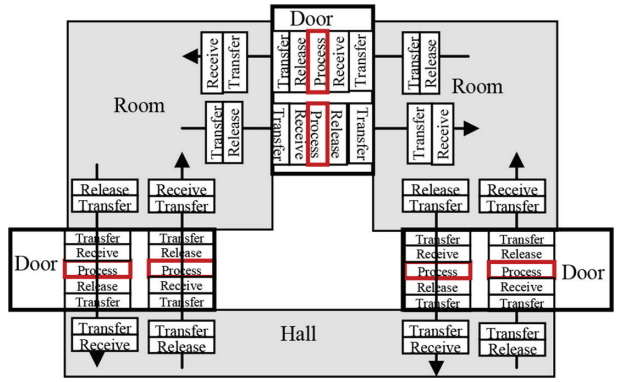

Figure 18: FM representation of Pieters's [30] infosphere.

The spatial arrangements in the physical world are only relevant as far as they enable or limit interaction between entities. For example, the situation where different entities are in the same room is only relevant for the consequence that this allows these entities to interact. The general structure of the model is then represented by which entities can access each other, interact and collaborate.

The general structure of the FM diagram is based on machines and machines as basins of flows. Pieters's [30] infospheres (Fig. 4) can easily be represented in the richer FM representation shown in Fig. 18.

The interesting aspect of Pieters's [30] infosphere concept is its mixed representation of physical and cyber spheres. For example, Pieters considers the so-called *road-apple attack*. In the IT world, this involves an infected dongle with the organization's logo left, say, in the organization's canteen. When an employee finds the dongle, he/she may plug the dongle into his/her laptop. If the employee does, the dongle will install a rootkit on the hard disk drive without the employee's knowledge (see Fig. 19). 'The rootkit can intercept any Input/Output to and from the disk or the disk's firmware. It uses this to its advantage by modifying data being sent back to the host computer. When the computer requests data from a sector on the disk, that data is first loaded into the disk's cache. The firmware can modify the data sitting in the cache before notifying the host computer that the data is ready' [44].

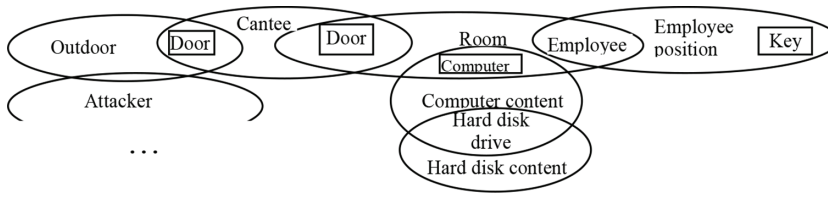

Figure 19: Infospheres of the road-apple attack (partial and modified, redrawn from Pieters [30]).

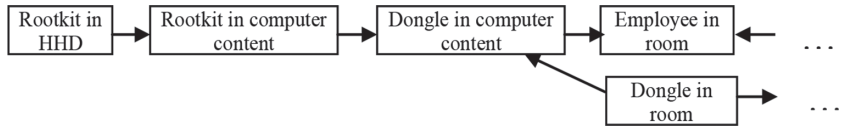

Figure 20: Attack graph of the second stage of analysis in the example (partial and modified, redrawn from Pieters [30].

It is clear that Pieters's [30] infosphere is not sufficient to describe the road-apple attack; hence, it is complemented with the diagram shown in Fig. 20 for analysis purposes.

An FM representation offers an integrated depiction in which the *physical* sphere is interwoven with the *digital* sphere (Fig. 21) in a smooth transition between the notions of *s* and *flow*. An attacker goes to the canteen (1) carrying the dongle (2) and leaves it there (3). An employee mistakenly takes the dongle (4) and plugs it into a computer (5), causing the appearance of the rootkit (6). Accordingly, any IO received or sent to the disk firmware (7) is intercepted and processed (8) by the rootkit.

7 CONCLUSION

Most of the reported research in the field of physical security has been driven by practical objectives. Yet, the value of these studies is limited because of their stationary representations based on static conceptions of space. This paper has presented a theoretical foundation for physical security through development of a logical description by means of a flow-based model based on the concept that a security system is a machine. The logical flow model is a unifying method with potentially diverse applications. The results indicate that the FM diagrammatic methodology can provide a systematic base for descriptions of a security process.

The proposed representation is demonstrated by applying it to an actual security plan of a government ministry's IT department. The results seem promising as a vehicle for modelling attacks and a security plan, and as a predesign schematic specification. Also, it seems suitable for security training and planning.

Further research will involve using the FM security diagram in the initial phase of developing an automated security monitoring system.

One possible weakness is the complexity of the resulting diagram, which might seem like too much to follow and control; however, this drawback is not significant in light of the large schemata of high-rise buildings and multifaceted specifications of airplanes and similar technical projects. From another point of view, such a centralized cross-level diagrammatic language seems to be an advantage in comparison with the complexity of multilevel diagrammatic languages such as UML and SysML.

REFERENCES

[1] Ferraiolo, K., The systems security engineering capability maturity model (SSE-CMM). *Proceedings of the International Systems Security Engineering Association*, 2000. http://csrc.nist.gov/nissc/2000/proceedings/papers/916slide.pdf (accessed 15 February 2017).

[2] Shirey, R., *Internet Security Glossary, Version 2*. Internet Engineering Task Force (IETF), RFC 4949, 2007.

[3] Krutz, R.L. & Vines, R.D., *The CISM Prep Guide: Mastering the Five Domains of Information Security Management*, John Wiley & Sons, 2003.
[4] Gregg, M., Hack the Stack: Layer 1: The Physical Layer, *Hack the Stack: Using Snort and Ethereal to Master the 8 Layers of an Insecure Network*, Syngress Publishing. 2006.
[5] National Computer Security Center (NCSC). US glossary of computer security terms, NCSC-TG-004, version 1. NIST computer security resource center. http://csrc.nist.gov/publications/secpubs/rainbow/tg004.txt (accessed 14 September 2017).
[6] Niles, S., *Physical Security In Mission Critical Facilities*, Schneider Electric White Paper 82, Revision 2, American Power Conversion, 2004. http://apcmedia.com/salestools/SADE-5TNRPL/SADE-5TNRPL_R2_EN.pdf
[7] St Sauver, J., *Physical Security of Advanced Network and Systems Infrastructure*, Presented at Spring 2011 Internet 2 Members Meeting, Arlington, Virginia, April 19, 2011.
[8] St Sauver, J., *Physical Security: A Crucial (But Often Neglected) Part of Cybersecurity*, SlidePlayer.com, 2017 (accessed 11 April 2017).
[9] Hutter, D., *Physical Security and Why It Is Important*, SANS Institute. https://www.sans.org/reading-room/whitepapers/physical/physical-security-important-37120 (accessed 5 March 2017).
[10] Harris, S., Physical and environmental security, *CISSP Exam Guide*, 6th ed., USA McGraw-Hill, pp. 427–502. 2013.
[11] Oriyano, S., Physical security. *CEHV8: Certified Ethical Hacker Version 8 Study Guide*. Wiley: Indianapolis, pp. 393–409, 2014.
[12] Scott, M., Coca-cola data breach highlights: importance of laptop security. ACFE Website, 2014, December 1. http://acfe.com/fraud-examiner.aspx?id=4294986501 (accessed 8 April 2017).
[13] Hunker, J. & Probst, C.W., Insiders and insider threats: an overview of definitions and mitigation techniques. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA)*, **2(1)**, pp. 4–27, March 2011.
[14] Homeland Security Research Corporation., China, UAE, Kuwait and Saudi Arabia are fastest growing homeland security markets. *Homeland Security Research Corporation Website*, January 30, 2014. http://homelandsecurityresearch.com/blog/category/cctv/ (accessed 21 March 2017).
[15] Federal Information Security Management Act (FISMA), PE1-PE19, Appendix F, NIST Special Publication pp. 800–53 Rev 3, n.d.
[16] Huang, J., *Brief Tour about Android Security*, December 7, 2012 [slides].
[17] Schiavone, S., Garg, L. & Summers, K., Ontology of information security in enterprises. *Electronic Journal of Information Systems Evaluation*, **17(1)**, pp. 71–87, 2014.
[18] Senstar Cyber. Threats in physical security: understanding and mitigating the risk. *senstarcyber.com* (accessed 11 February 2017).
[19] Dictionary.com. http://dictionary.com/browse/process?s=t (accessed 11 February, 2017).
[20] franklin-witter, If security is a process, why don't we manage it like one? Thought Leadership Website. https://symantec.com/connect/blogs/if-security-process-why-dont-we-manage-it-one (accessed 10 March 2017).
[21] Al-Fedaghi, S. & Moein, S., Modeling attacks. *International Journal of Safety and Security Engineering*, **4(2)**, 2014.
[22] Al-Fedaghi S., New conceptual representation of collision attack in wireless sensor networks. *International Journal of Safety and Security Engineering*, **3(4)**, 2013.

[23] Al-Fedaghi S. & AlMeshari, H., Social networks in which users are not small circles. *Informing Science*, **18**, pp. 205–24, 2015.
[24] Al-Fedaghi, S., Conceptualization of various and conflicting notions of information. *Informing Science*, **17**, pp. 295–308, 2014.
[25] Al-Fedaghi, S. Alsaqa, A., & Fadel, Z., Conceptual model for communication. *International Journal of Computer Science and Information Security*, **6(2)**, 2009.
[26] Al-Fedaghi, S., Software requirements as narratives. *Third International Conference on Information, Process, and Knowledge Management*, Gosier, Guadeloupe, February 2011.
[27] Al-Fedaghi S. & Mahdi, F., Events classification in log audit. *International Journal of Network Security & Its Applications*, **2(2)**, 2010.
[28] Al-Fedaghi, S., Flow-based description of conceptual and design levels. *IEEE International Conference on Computer Engineering and Technology 2009*, Singapore, January 2009.
[29] Simon, H. A., *The Sciences of the Artificial*, MIT Press: Cambridge, 1996.
[30] Pieters, W., Representing humans in system security models: an actor-network approach. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA)*, **2(1)**, pp. 75–92, 2011.
[31] Bishop, M., Coles-Kemp, L., Gollmann, D., Hunker, J. & Probst, C., 10341 report – insider threats: strategies for prevention, mitigation, and response. *Insider Threats: Strategies for Prevention, Mitigation, and Response, Dagstuhl Seminar Proceedings*, no. 10341, 2010.
[32] Mobbs, P., Introducing information security. A series of briefings on information security and on-line safety for civil society organisations, http://fraw.org.uk/mei/archive/handouts/apc-pws/pws-01.html, 2002 (accessed 10 March 2017).
[33] Forcht, K.A. & Kruck, S.E., Physical security models, philosophies, and context. *Journal of Information Management*, **10(2)**, article 9, 2001.
[34] Robbins, P., CISSP & physical and environmental security & information security. Presentation at *ISA 400 Management*, Information Security & Assurance Program University of Hawai'i West Oahu, 2015, January 17.
[35] Philpott, D. & Einstein, S., The Integrated Physical Security Handbook. *The Counter Terrorist Magazine web site*, http://thecounterterroristmag.com/pdf/IntegratedPhysicalSecurityHandbook.pdf (accessed 2 April 2017).
[36] Woodbury, C., Security blueprint [Online]. IBMSystems website, http:// ibmsystems-mag.com/aix/administrator/security/Security-Blueprint/ (accessed 1 April 2017).
[37] Edraw M., *Warehouse Security and Access Plan Template* [software], 2004–2017.
[38] Lincke, S.J., Physical & Personnel Security, *CISA Review Manual 2009*, PhD thesis, Univ. of Wisconsin, USA.
[39] Marrone, S., Rodríguez, R.J., Nardone, R., Flammini, F. & Vittorini, V., On synergies of cyber and physical security modelling in vulnerability assessment of railway systems. *Computers and Electrical Engineering* **47**, pp. 275–285, October 2015, August. https://doi.org/10.1016/j.compeleceng.2015.07.011
[40] Vuorinen, J. & Tetri, P., Security as a machine: struggling between order and chaos. *Pacific Asia Conference on Information Systems (PACIS) 2009 Proceedings*, paper 113, 2009. http://aisel.aisnet.org/pacis2009/113
[41] Vuorinen, J. & Tetri, P., The order machine: the ontology of information security, *Journal of the Association for Information Systems*, **13(9)**, pp. 695–713, 2012.

[42] Deleuze, G. & Guattari, F., *Anti-Oedipus*, Capitalism and Schizophrenia vol. 1, Continuum: London, 2004.
[43] Imbusch, O., Langhammer, F. & von Walter G., Ercatons: thing-oriented programming. Presented at *5th Annual International Conference on Object-Oriented and Internet-Based Technologies, Concepts, and Applications for a Networked World, Net. ObjectDays 2004*, Erfurt, Germany, pp. 27–30, September 2004. DOI:10.1007/978-3-540-30196-7_16
[44] Osgood, R., Hard Drive Rootkit Is Frighteningly Persistent. *Hackaday Blog web site*. http://hackaday.com/2015/06/08/hard-drive-rootkit-is-frighteningly-persistent/ (accessed 14 March 2017).