Crimson Publishers
Wings to the Research

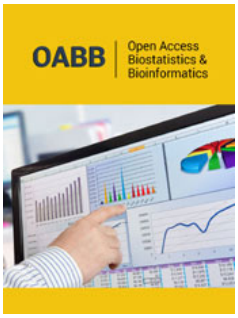
Research Article

Cyber Risk and Cyber Security: Cyber Access Control with Data Mining

Mariagrazia Fugini*

Department of Electronics, Information and Bioengineering, Italy

ISSN: 2578-0247



***Corresponding author:** Mariagrazia Fugini, Department of Electronics, Information and Bioengineering, Polytechnic of Milan, Italy

Submitted: August 07, 2024

Published: August 30, 2024

Volume 3 - Issue 5

How to cite this article: Mariagrazia Fugini*. Cyber Risk and Cyber Security: Cyber Access Control with Data Mining. Open Acc Biostat Bioinform. 3(5). OABB.000575. 2024.
DOI: 10.31031/OABB.2024.03.000575

Abstract

This paper presents concepts about adaptive cyber security for areas where risks and emergencies need to be managed and may require modifying authorizations dynamically due to risks. In scenarios where risks may occur, cyber security has to be managed dynamically granting and revoking access rights according to the needs-to-save *Safety of Smart cyberspaces*. Treating cyber security for risk is given in the paper via a *model of Cyber Resources in the Smart Cyberspace*. A spatial description allows for resource localization in the affected areas, and a set of physical and logical identifiers permits dynamic regulation of cyber access to them according to security and risk policies, combined together. *Adaptivity* of cyber access control rules apply to Cyber Subjects, who intervene to manage the risk. A review of existing approaches and a proposal are given.

Moreover, we consider the more general concept of *Cyber Physical Systems (CPS)* and the theme of Security Policy Checking, which is a hot topic in the cyber security field to generate and manage coherently secure policies about access to resources. In the second part of the paper, we propose a methodology for access policy checking. In particular, the contribution of this work consists in using a Data Mining approach to different kinds of access policies developed for managing both physical and logical resources in Smart cyberspaces. It is shown how Data Mining can individuate issues and unwanted access to systems, e.g., for violation of read/write policies, or of privacy policies, in an environment managed under the Role Based and Attribute Based Access Control models, which are nowadays the most popular and up to date models for security management, as we will explain in the work.

Keywords: Risk in cyberspace; Adaptive cyber access control; Attribute-based access control; Generalized world; Entities; Safety of smart cyberspace; Data mining for policy checking

Introduction

Risks due to disasters such as fires, floods, earthquakes, civil war, or terrorist attacks may cause crisis situations. Regardless of their origin, crisis situations are often accompanied by uncertainty of how the disaster will develop, and require a rapid pace of response operations given the possibility of serious loss of human lives and property if not responded to properly. Good situational awareness and decision-making support will greatly help minimizing property damage and injury, and saving people's lives. In scenarios where risks may occur, such as urban areas (parks, event locations, etc.) or industrial and building worksites, cyber security has to be managed dynamically granting and revoking access rights according to the needs-to- save [1]. *Safety of Smart cyberspaces* [2] needs models, methods and tools to intervene in face of a crisis, to react to dangerous events in the monitored areas, and to locate people at risk. However, resources and people's confidentiality and privacy have to be

preserved, while allowing for *dynamic adaptation* of cyber access control rules to face risks.

In recent years, due to advances in ICT, several infrastructures, technologies and methods have been introduced for risk and disaster management [3]. For instance, advanced cloud platforms and services are available to support emergency response and people tagging during big events [4]. Moreover, most modern cities have deployed technology solutions towards the concept of "smart cities": urban areas use different types of sensors for digital data collection (in the paradigm of Internet of Things) to supply information to manage assets and resources efficiently [5].

The damages caused by risks in a cyberspace, namely an environment where physical resources (sites, premises, devices, etc.) and logical resources (data, digitally mirrored resources, etc.) are exposed to risks, can be greatly reduced by preparedness to manage such cases. Meanwhile, considerable attention has been paid to cyber security, in particular to access control to data and resources and to privacy of people in such cases, showing that significant access control requirements exist for civilian government and corporations to meet their operational, control, and strategic goals. Civilian governments are concerned with confidentiality of data and privacy of people, namely with the protection of individuals' data, policies and intervention plans in economy, social, healthcare and other sectors of organizations [6].

In this paper, we address cyber access control rules that can be adapted to risks detected in the cyberspace and to natural disasters. The aim of the paper is to give a review of work in risk management and in cyber security. Then, we propose a model for dynamic management of cyber access control in risky cyberspaces.

In the second part of the paper, we consider the more general concept of *Cyber Physical Systems (CPS)* which are taking more and more importance and have become an integral part of the basic infrastructure of many Countries. This phenomenon has inevitably expanded the attack surface of the world of cybercrime exponentially. As can be seen from numerous studies, the number of attacks has grown dramatically. This increment is also due to some threats that have evolved in recent years. In particular, the advent of IoT devices supported by Cloud networks, core points of CPS, together with the spread of work from home, has highlighted the need to develop better and more dynamic Access Control (AC) models that allow the management of access to both logical and physical resources [7]. Many companies have migrated to the use of *dynamic Access Control Systems (ACS)* based on *Role Based Access Control (RBAC)* or *Attribute Based Access Control (ABAC)* [8]. Nevertheless, on the other hand, they are often difficult to structure, and when developed, they can often hide unintentional errors within their policies. For solving this problem, several tools and methodologies for *policy control* have been developed over the years. However, many do not boast high usability values and require users to go through a first important learning phase to take advantage of the methodology.

In this paper, we propose an alternative solution, keeping usability at the center of our model, so we develop a methodology to ease *the analysis of policies* by extracting them from logs. Our solution takes advantage of *Data Mining (DM)* technologies that allow inspection of logs generated on two Policy Sets, one applied to a purely logical system, such as the management of access to the medical records of two patients, and one to a cyber physical system that controls the access permissions to a surgical robot. The aim is to be able to semi-automatically identify a series of requests that obtained permissions not desired by the administrators during the policy development phase.

The two algorithms used have shown high accuracy values on the set of policies we have defined. In particular, by observing the average precision during the experiments, we could study the accuracy values for the obtained outputs. that the two chosen algorithms, namely JRip and Ridor, have respectively shown in the results of access policy checking. We want to show how the methodology developed so far maintains accuracy values while maintaining high usability. Usability comprehends the concept of efficiency and of learnability so that the system should be easy to learn, and the agent can rapidly start using the system and perform a high level of productivity.

The overall paper is organized as follows

In Section II, we define our context of cyber security with principles and trends. In Section III, we review related literature. In Section IV, we present our modeling approach to Cyber Access Control in presence of risks. In Section V, we present technology and scenarios, showing two applications of CPS (Industry and Healthcare 4.0) and their main threats in cyber security. In Section VI we present our Data Mining-based approach to AC policy analysis, with the structure of two experiments, their outputs and results. Finally, in Section VII we give the concluding remarks.

Cyber Security: Principles and Trends

Individuals, businesses, and states are becoming increasingly vulnerable to cyber threats. Such cyber-attacks are generally aimed at accessing, modifying, or destroying sensitive information, extorting money, or disrupting business processes. Cyber security generally refers to protecting information systems and information, from internal and external cyber attaks. The term cyber security is often used interchangeably with the expression Information Security, but although there are many common elements, these two concepts are not the same. Cyber security goes well beyond the boundaries of traditional Information Security. It encompasses protecting information assets and protecting environments, including protecting people, physical infrastructures, and economic assets [9].

Among the most critical points to defend are those services that are provided to all citizens, such as power plants, hospitals, financial services, and much more [10]. Potentially, any activity that creates a discrepancy between actual (de facto) and perceived (de jure) property rights, whether intentional, accidental, or simply incidental, is still a cyber security incident. Cyberspace typically involves an extensive computer network composed of many sub-networks, which allow for the development of communication and

data exchange activities. This concept is now expanding to include CPS. These systems integrate digital capabilities, including network connectivity, with physical devices and systems, especially with the advent of IoT used in many different domains, from energy transmission systems to smart houses, smart grids, military systems, robotics, and medical services [11].

The Cost of a Data Breach Report compiled by the Ponemon Institute for IBM in 2021 [12], the average cost of a data breach was $3.86 million. According to "The 2020 Official Annual Cybercrime Report" [13], cybercrime is the most significant economic threat to every business globally and one of the biggest problems facing humanity. Implementing effective cyber security measures is particularly challenging today, where there are more devices than people, and the world of cybercrime is becoming increasingly innovative and dangerous. Cyberattacks are increasing in frequency, impact, and sophistication. Behind that growth, there is the adoption of new technologies by cybercriminals, and because defenses are often permeable, cyber-attacks have a greater degree of success. Over the past decade, the number of breaches in the US has increased dramatically, up nearly 67%. [14]. A study of the Universiti Teknologi MARA [15] highlights the main technological threats recognized in cyber security nowadays deriving from IoT, Cloud Services, Fog Computing, etc.

i. Definitions

We use Cyber Subjects to model entities that take actions in the cyberspace, and CyberObjects to model: i) physical resources (e.g., areas, tools, sensors, video cameras), and people to be protected; ii) informative entities (e.g., data, maps, and sensor data) whose sensitivity needs be ensured even during a crisis.

Cyber access control can dynamically change the *Authorizations of CyberSubjects to access CyberObjects* based on the risk level and risk type, and on CyberSubject/CyberObjects access requirements. In fact, CyberSubjects can receive new access privileges, temporarily to handle the risk, and then return to the normal situation under revocation of emergency privileges. The principles of cyber access control regard aggregating intelligence from masses of data acquired from an area via sensors, GPS, monitoring devices, and so on, to observe risks and emergencies signaled in a given area. The model is based on ABAC [7], where attributes describe CyberSubjects, CyberObjects and the cyberspace, including geo-referenced information for positioning people, resources, events, which signal a risk and which have to be located to understand the dangerousness of the situation, and risk areas. Under the need-to-know access control policy, we define *flexible access control rules*, which are activated/deactivated to adapt access of CyberSubjects to CyberObjects according to risks. Adaptivity is modeled by introducing Access Control Domains (ACDs) for CyberSubjects on CyberObjects linked to each other by temporary privileges. The ABAC paradigm is used in an extended architecture model of XACML for risk management.

For risk and disaster recognition, we rely on the solutions proposed in [16] addressing cloud and fog systems which tend to leverage architectures with a strong central intelligence that controls various systems. We propose to leverage edge devices' cyber security capabilities to allow IoT systems to work also in emergency cases where the connection with the central services is interrupted while preserving cyber security properties of the cyberspace.

Literature Overview

Regardless of the origin of risks and disasters, crisis situations are often accompanied by uncertainty of how the disaster will develop, and require a rapid pace of response operations given the possibility of serious loss of human lives and property if not responded to properly. On the other side, cyber access control models are proposed in terms of Attribute- based Access Control for CPS in [17].

Good situational awareness and decision-making support will greatly help minimizing property damage and injury, and saving people's lives. Risk management in critical cyberspace is based on events that arise on the fly is still an open issue [18]. Considering that about 90% of injuries can be traced back to unsafe practices and behaviors [19], proper safety management is essential to treat the risks that arise based on unsafe activities and situations.

Until a few years ago, monitoring activities of persons and their safe behavior (e.g., usage of proper equipment during risky activities) was very challenging if not impossible [20]. Nowadays, the "Smart Environment" (SE), i.e., areas surveilled by sensor technologies, make it possible to monitor areas, activities, people, tools, and machinery, with many potential opportunities to obtain safety also in the appearance of emergencies in CPS [21], by providing the infrastructure that enables risk prevention and management through advanced devices and services, by interconnecting physical and virtual "things" based on the existing and evolving ubiquitous technologies.

In more recent years, the term autonomic computing has been often associated to (and sometimes replaced by) the term self-adaptive system [22]. Here, the focus is more in designing systems able to modify their behaviors autonomously (e.g., functionalities, operating conditions, actions) in either response to changes affecting the system itself or the cyberspace, the system is deployed in. The crucial shift in paradigm is that the cyberspace and (possibly) the users are now explicitly considered as part of the operational model, and reactions are then issued in response to changes in the system or variations in the operational cyberspace /users. The literature on autonomic and self-adaptive systems is highly fragmented, with research differentiating in the addressed problems, considered approaches, and proposed solutions. For instance, identification of when should we introduce adaptation can either follow a reactive approach in response to a change or a proactive one, by speculating that a change will occur in the near future. (e.g., see https://osha.europa.eu/en).

In SEs, smart Cyber Objects interact based on semantic services. From the architectural point of view, SEs evolve from the Service-Oriented Architecture (SOA), which provides a decentralized architecture, to facilitate the adoption of IoT Services to define the

interaction of the smart CyberObjects [23]. IoT Services include sensing and control of the physical "things". Therefore, in a SE that employs IoT Services, safety is an important issue that can currently be tackled in a decentralized way based on distributed devices, possibly connected to a cloud system, which can sometimes be unavailable due to critical events, to guarantee the safety of the assets while protecting persons, critical CyberObjects, and infrastructures.

In critical situations, *resilience* is defined as the capability to maintaining and regaining a stable state prior, during and after an event [24]. To achieve resilience, minimization of failure, early detection and treatment of hazards, minimization of consequences of a risk, and flexibility are required. By continuously monitoring the cyberspace, early detection of hazards is possible. Moreover, early and full assessment of the risk and consequences, planning of preventive strategies, and facilitating the collaboration between actors improves the treatment of risk. Finally, the flexibility and adaptiveness to the risk type facilitates the successful execution of the strategies.

In recent years, two main standards have been developed for emergency management, namely AS/NZS 4360 [25] and the ISO 31000:2009 standard on Risk Management [26]. AS/NZS 4360 defines risk as a chance of something that could happen, which will affect the previous Cyber objectives and can be measured by two dimensions, which are consequences and likelihood. Similarly, ISO 31000:2009 considers risk as the effect of uncertainty on Cyber objectives where this effect might be positive or negative [27].

Moving towards SE requires data capturing risks be considered as an entity. As time progresses, data are enriched with continuous information coming from devices interacting with other entities such as individuals, sensors, machinery, and so on. Thanks to the availability of a large variety of sensors and devices that both sense and integrate data into technological monitoring platforms, several information can be gathered from the cyberspace to monitor its status and possibly notify risks and critical events [28].

The cyberspace consisting of data items gathered in this way can be automatically processed to signal, usually in the form of events, the changes in the cyberspace conditions [29]. A step ahead consists in interpreting these events to understand the risk that is possibly arising, to elaborate a strategy to prevent the risk, and to support the execution of the strategy by involved actors such as risk managers.

Emergency management is strictly bound to the concept of vulnerability, which is currently a key determinant in every "pure" risk assessment [30]. More precisely:

1. *Vulnerability* is the propensity or predisposition to be adversely affected. Vulnerability encompasses a variety of concepts and elements including sensitivity or susceptibility to harm and lack of capacity to cope and adapt.

2. *Resilience* is the ability of a system to cope with an external factor that undermines it, with the system bouncing back to a previous state of safety.

3. *Susceptibility* is defined as the fragilities in a community that influence potential harm from particular hazard impacts.

However, multiple definitions and conceptual frameworks of vulnerability have emerged from several distinct groups (such as political economy, social-ecology, vulnerability, and disaster risk assessment, and adaptation to climate change) who have different views on the same concept. The difference in views on vulnerability is a result of the needs emerging from particular groups and peculiar issues regarding potential impacts of disasters [31] explores the widening of the vulnerability concept (see Figure 1) which indicates an improved understanding of the notion of vulnerability over years.

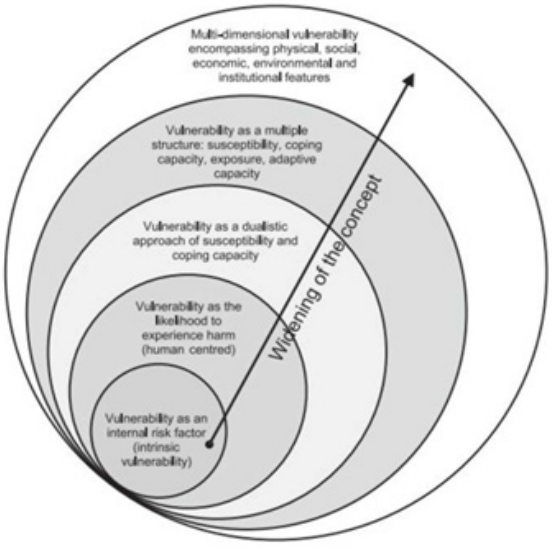


Figure 1: Key spheres of the concept of vulnerability (Source: [31]).

Despite the existence of various frameworks to define and assess vulnerability, some common causal factors of vulnerability have been identified [32], which include: 1) lack of resilience, which constitutes lack of coping and adaptive capacities, 2) susceptibility/fragility as known in disaster risk domain (or sensitivity in the climate change domain). The success of disaster and risk management efforts depends on the ability of multiple stakeholders to share disaster- related information. Modeling the SE for emergency management includes publishing and retrieval of relevant knowledge or information about hazards and vulnerabilities and requires semantic integration and enhancement of hazard resources to achieve meaningful results [33].

Multiple international agencies have since long emphasized the importance of a unified understanding of the disaster management domain through the creation of glossaries and vocabularies [34].

In the field of climate change, currently the so-called emergent risks have recently appeared in the scientific literature in sufficient detail to permit assessment, for example, of the potential impacts of geo-engineering and of solar radiation management on the monsoon. An *emergent risk* is defined as an event that can be identified as a key risk once sufficient understanding of it accumulates. Among the reasons for emergence of a risk is that its initial consequences have only recently been detected above the natural variability of the climate system, such as effects of ocean acidification on calcareous organisms. *Risks* may also arise *gradually* when they emerge from the interaction of phenomena in a complex system, for example the effect of populations shifting in response to climate change on the capacity of receiving regions to adapt to local climate changes [35-37].

In risk management, providing safety to people and physical objects according to what happens in the environment is an open issue, as discussed in [38] with the research moving under various labels: smart environments, smart cities. With the purpose of reducing the risk exposure of a physical environment, *resilience* is more and more considered as a feature to be added to the environment [39]. The purpose is to achieve a balance of a physical system by constantly adapting the information flow among sensors, risk management tools and human actors to meet the needs related to understanding the risks and incidents. A statement on the resilience of an environment corresponds to a particular incident and to the system ability to recover, within a certain response time, as well as to the analysis of composite costs and risks. *Resilience indicators* are studied with focus on defining the concept of risk, modeled as a first-level entity, and on how risk can be treated by substituting the concepts of risk probability by uncertainty, so paving the way to use the main ideas of *resilience engineering in risk management*.

Modeling Cyber Access Control in Presence of Risks

Coming to cyber access control modeling, under need-to-know access clearances, we have adapted the ABAC paradigm and XACML policy language to meet our needs for adaptive access control according to events in the cyberspace. We discussed the architecture of access control mechanism. that extends the XACML policy language to accommodate the adaptations of ABAC to risk management scenarios. Here, we use the Generalized World Entities (GWE), described in [40], to semantically model the components of the access control model,

A cyber access control rule is one defined on a cyber resource, be it physical (e.g., a device) or logical (e.g., a database table). Cyber Access Control on cyber resources (Cyber Objects), including people who need protection, depends on what happens in the cyberspace, and has various purposes depending on the Cyber Subjects and their organizational roles. There is a need for adaptivity of the cyber access control rule, to be able to manage risks, and to perform necessary actions to control the cyberspace while minimizing the threats to Confidentiality, Integrity and Availability (CIA) properties of such space.

ACDs are security policies which delimit the set of access control rules to be activated in case of risk and deactivated upon the conclusion of risk. For handling the conflicts between the access control rules due to run-time activation and deactivation, we adopt the XACML policy language [41] that includes many combining algorithms to avoid conflicts; we extend it to include the adaptivity to risks using ACDs to establish which rules apply in which context.

We assume we are in the discretionary-based access control policy (privileges can change for CyberSubjects on CyberObjects) and in the ABAC paradigm. A CyberSubject is any entity taking actions in the system. CyberObject (a resource and in general a geo-referenced element) is any entity to be protected from unauthorized use: data, devices, services, physical areas, persons, and so on.

Cyber Access Control Policies

A cyber access control policy is the set of rules about permissions or denial of cyber access from a Subject onto a set of CyberObjects. Privileges represent the "authorized behavior of a Subject"; they are defined by an Authority. In cyber systems, they represent both the right to perform elementary actions. For example, *read* on a CyberObject maps into ''view'', ''read'', ''zoom in/out'' privileges, depending on the CyberObject and the right to perform activities (e.g., process the images from a camera or rescue a person).

A further element is the *Cyberspace*, defined as a set of monitored factors used at decision time to influence an access control decision. Risks and Emergencies are recognized using a *Risk Management System (RMS)* [42], which monitors the cyberspace. Moreover, external crisis management systems can notify the RMS about the occurrence of natural disasters. In this paper, however, we do not deal with risk and disaster recognition. We assume that sensors and devices are in place for this purpose, and also assume recognition methods have a suitable level of accuracy, while deepening these issues will be CyberObject of our short-term research steps.

The RMS system will introduce a strategy to manage a recognized risk by proposing a set of risk actions. These actions, like

turning on an alarm, or opening an emergency door, can sometimes need to be undertaken by modifying the access control privileges that normally hold between CyberSubjects and CyberObjects. These modifications require adaptive access control to face a risk on a temporary basis. Access control rules are changed for example by granting more privileges to some CyberSubjects on some CyberObjects and therefore bypassing the existing access control, which is then rolled back to the previous access control privileges once the risk has been managed.

We consider the ECA (Event-Condition-Action) paradigm [43] to model adaptive access control policies: given an event (e.g. risk), we consider the conditions to dynamically activate the *Access Control Domain* (ACD) and hence access control rules. Also, Subject and CyberObject attributes might change due to an event that can affect the authorization decisions later on. ECA allow us to express cyberspace policies that affect security policies, which in turn are expressed by ACDs. In fact, an ACD is a policy that establishes how the access rules are allowed to change depending on what happens in the cyberspace areas, namely which rules apply, in a given situation and at a given time, to a set of Subject Attributes over a given set of CyberObject Attributes using which Privileges. This means that some ACDs are activated (e.g., the Fire ACD applies when the fire risk is detected), while others are deactivated (e.g., the *Security Check Point* of the ACD is deactivated upon the fire risk) in order to dynamically adjust the set of access control rules applied for the time necessary to handle the risky situation.

Finally, we use the term "access clearance" or "sensitivity" referring to a threshold or a range of admitted values for confidentiality or other access control elements in a discretionary style (no mandatory policies are obviously considered in our approach).

Cyber Access Control Model

The model components are defined as follows.

Cyberspace: We consider a composed cyberspace, including open and closed areas. Each area has at least one access point used as an entrance, an exit, and/or as an emergency exit. In the cyberspace, there are several factors that might cause risks and should be monitored, e.g. water and gas pipes, electrical wires and so on.

RMS and monitoring devices: These elements model the technological elements needed for the environment surveillance. The RMS is a dashboard where environmental data collected by devices (d) are conveyed, so that persons are notified about the current risks in the various areas. In our model, d are the various monitoring devices such as sensors for the temperature, light, smoke levels, smart phones and wearable devices that can be used for getting data about people's positions and health status like heartbeat. These d elements notify risks to the RMS which selects the strategies to be followed in a risky situation. The details about the specific kinds of sensors used, their costs, and the accuracy of the data given by the sensors are out of the scope of this paper. And for the sake of simplicity, we assume that the risks are reported based on accurate data. We also assume that the RMS can both deploy automatic actions (e.g., lock the doors, turn the electricity off, and so on) and suggest actions to humans who have to intervene or to decide what actions to undertake for risk.

CyberObject: this abstracts resources that a Subject can access or act on. Objects in our definition are any resource to be protected. Objects hold three groups of attributes (OA): 1) General Attributes can be Object specific and differ depending on the type of the object. 2) Geo Attributes, including geo referenced coordinates (latitude, longitude), and levels of granularity (they exist at various zooming levels) available in the repository, and the objects in the vicinity, 3) Security Attributes define restrictions on information in terms of privacy, owner, level of sensitivity, time restrictions, age/location restrictions, and groups:

CyberObject o: {{GeneralAttributes: ID, ObjType, ...}

{GeoAttributes: [Latitude, Longitude], ZoomingLevel, {VicinityObjs}, ...}

{SecurityAttributes: Privacy, Owner, SensitivityLevel, {TimeRestriction}, {AgeRestriction},

{LocationRestriction}, {Groups}}}

CyberSubject: this abstracts a Person, an application or a process wanting to perform an action on a resource/object. A Subject can hold many Attributes (SA). We consider the following two groups of SA: 1) General Attributes: define the general characteristics of a subject, such as its identity, name, and etc. 2) Geo Attributes: define geo properties of a subject, such as location, reachable positions, etc. Geo attributes can be given at various levels of granularity. For example, for privacy reasons, the exact location of the Subject might be hidden while the Subject's logical position, usual location, and the places that the Subject is allowed to have access to are visible;

Access control attributes: define the access control-related properties of the subject, such as access clearances, highest-possible access clearance roles which can be active at a given instant, and so on. Moreover, each Subject has been associated with a Personal Risk Level (PRL). The used PRL depends on the Physical location, expressing the presence of risk sources and persons in the environment, Role of the Subject, and the Objects that they need to access.

Formally: *PRL=f (s.Position, s.Role, {o.ID}).*

The Subject Role and accessed Objects (that are recognized by their IDs) are evaluated using an evaluation function, while the location is evaluated using an environmental risk map [44]; f is an opportune evaluation function. A possible choice is given by the following function (where eval stands for evaluation):

$PRL = \sum (eval(s.Location), eval(s.Role), eval(\{o.ID\}))$

To summarize, a *Subject* is defined as follows:

Subject s: {{GeneralAttributes: ID, Name, ...} {GeoAttributes:

[Latitude, Longitude], GranularityLevel, {AllowedLocations}, {DeniedLocations}, …} {SecurityAttributes: ClearanceLevel,

MaxClearanceLevel, {roles}, PRL}}

In our model, Subject Attributes (SA) and Object Attributes (OA) are assigned by an authoritative subject, or by a group of CyberSubjects, that collaborate and balance one another's decision/control.

Actions and activities: these are operations (i.e., privileges) that can be executed by CyberSubjects on objects in a given ACD (where the ACD is defined hereafter). We consider two types of operations: simple operations (actions), such as read, write, update, execute, zoom-in/out; and complex operations, called activities, which combine simple actions to model a task, a process, an application, or even a physical action. Examples of activities in an environment are "View persons localization data", or "Turn on the fire alarm protocol", "Guide an ambulance towards the crisis area center". Actions and activities are assigned with a unique id to be able to differentiate them. Activities can consist of different actions, and, for the purpose of Separation of Duties, each component action can be executed by a different CyberSubjects. However, for the sake of simplicity, we define actions and activities in the same way and assume that they can be performed by the same CyberSubjects. If a request for performing an activity is authorized, it means that all the sub actions that are included in that activity are also authorized to ensure that the activity can be completed.

Access control rule r, and ACD c: The ACD component of our model indicates a set of access control rules which are valid in a certain situation based on dynamic changes in the environment, in particular the occurrence of risks. In other words, ACDs are access control domains, in which rules are defined that are the operations allowable to CyberSubjects on specific objects in an environment under certain circumstances. We assume that the access control rules in each ACD are defined by the function *Define Rule (a, outcome)* where 'a' and 'outcome' are the operation and the authorization result, respectively. As an example, for the *EmergencyACD*, we can define the rule specifying that the CyberSubjects with the role of Security Manager and access clearance higher than L3 are permitted to turn on (activity) the alarms (Object o) whose sensitivity level is less than S4. Note that the clearance and sensitivity are shown with indexes and higher indexes are correlated to more clearance or higher sensitivity, respectively. Thus, can state:

R1: DefineRule (turnon, EmergencyACD) →

(s.Role ="SecurityManager") ∧ (s.AccessClearance > "L3") ∧ (o.Group = "alarm")

∧ (o.SensitivityLevel < "S4").

These rules can be addressed with the definition of <Rule> entity in XACML. The ACD then maps into the definition of the <Policy> entity. The <Policy> is constructed out of <Rule> entities following the specified rule combining algorithms (defined in XACML) to avoid overlaps for the rules belonging to an ACD.'

In order to receive a permission to execute an operation, a request should be submitted to the access control system as depicted in Figure 1. This request is specified by three elements: the requesting Subject (access subject), the operation to be permitted, and the Object to be accessed as in the standard XACML. Considering S, O, and A as a set of CyberSubjects, Objects, and Actions/Activities respectively, a request will be defined as follows: *Request (s:S, o:O, a:A).*

Considering the operation, attributes related to the Subject, Object and the rules in the active ACDs, Request returns Y, N, or NA, meaning that access is allowed, denied, or not applicable, respectively, based on the standard XACML.

Risk r/emergency em: Some factors that change dynamically and are monitored by the RMS can signal the occurrence of a risk situation, which can be recognized based on parameters such as: type, level, and location of the risk factors. It is then possible to decide how to adapt the access control rules when the risk r has to be handled. The RMS is able to recognize the risks that can happen in the environment. When a risk is recognized, the RMS identifies the root cause of the risk, its location and the persons at risk (according to their PRL), and tries to propose the optimal strategy to manage the risk. On the other hand, an emergency occurs when a risk exceeds a given threshold or when a parameter exceeds a threshold (e.g., if the environment temperature exceeds 70 °C).

Risk event ev: Identifications of risks in the environment can trigger risk events. Risk events, which can be localized due to their geo reference attribute, activate/deactivate ACDs. Events may also cause the modification of the attributes of CyberSubjects and Objects within the ACD, according to the strategies suggested by the RMS. The details of how we handle the impact of risk events on ACDs and Object and Subject attributes are discussed in the next section of this paper; risk events play a key role in dynamic adaptation of access control in response to changes in the environment.

Knowing the components of the access control model, and having the risks and emergencies that can cause risk events, we can specify our approach of adaptive access control in the following subsection.

Risk events are triggered due to environment factors out of range, which are identified as risks. Therefore, assignment of access control rules should be carried out dynamically as soon as the incident is identified to adapt the access control rules to the risky situation.

The RMS proposes a strategy to manage risks and emergencies that can take place to handle the risky situation. The location, type and intensity of the risk can affect the strategies suggested and/or deployed directly by the RMS.

For example, if smoke is detected as an out-of-range parameter, which means a possible fire event, the RMS evaluates where the source of the risk is. If the event occurs near gas pipes, it is potentially very dangerous and hence needs fast corrective strategies (evacuation, fire brigades, locking of doors and so on)

since it is an emergency. Instead, if the same factor (presence of smoke) occurs in locations where no special danger is present, this is a risk, which can be handled through preventive actions, such as smooth interventions to check what the origin of the smoke is, while evacuation is not needed.

In the first case, to give an example, the RMS is notified that a medium intensity fire risk exists near the gas pipes, and hence triggers a fire risk event. A sample strategy in this case would be: activating alarms, closing the gas pipes, notifying that evacuation is needed, and notifying the risk manager. This is specified as follows:

If (r.Type: "fire") ∧ (r.Level: "medium") ∧ (r.Location: "closeToGasPipes") : TriggerRiskEvent(r),

Strategy = {Activate ({alarms}), Close(gasPipes), Notify (Evacuation), Notify(s.RiskManager)}

The authorization to execute these actions might need a change in the access control rules. Therefore, access control rules should also get adapted according to the risk event by activating/deactivating necessary ACDs and changing subject/object attributes. Activation/deactivation is ruled out by ECA rules.

There can be multiple rules per ACD and different ACDs can share the same rules. Each ACD activation should enable the corresponding access control rules. Since we adopt XACML as the policy language, such conflicts can be avoided using the policy combining rules defined in XACML. As we mapped our ACDs into the definition of <Policy> entity in XACML, the set of activated ACDs are relevant to the <PolicySet> entity, and therefore, the policy combination algorithms can be applied to avoid conflicts between the activated ACDs.

ii. Adaptivity

We introduce the concept of adaptivity using the ECA paradigm. An ECA rule indicates that in case of an event, if the condition holds, then (a) certain action(s) should take place, where "action" denotes the activation/deactivation of ACD(s) or/and modifications in the attributes of CyberSubjects/Objects by the function ChangeAttr(attribute, condition, new- value). The event in the ECA paradigm is the risk event triggered by the risks identified in the environment and notified by the RMS.

To clarify, let us set some examples. Suppose we have the following ECA statement:

RiskEvent: (em.Type: 'Explosion') Conditions: (em.Level: 'high')

Actions: Activate (EvacuationACD) ∧ Deactive (CheckPointACD)

In this example, the conditions indicate the case of an occurrence of an explosion risk, with high level of danger. Therefore, the Evacuation ACD should be activated. And the CheckPoint ACD be deactivated to allow persons to evacuate the area quickly with no need to pass through security checkpoints.

The Evacuation ACD can include rules that facilitates the evacuation, like low sensitivity level for doors that need a access card to allow everyone to pass. Or letting the rescue teams to view the location data of the people at risk:

EvacuationACD.〖Rule〗_(1):Role(s)="SecurityStaff", p=read ,Group(o)= Location, effect=Permit.

EvacuationACD. 〖Rule〗_2:Role(s)= ANY,p=open ,Group(o)= doors, effect=Permit

Another example is the following:

RiskEvent : (r.Type: ANY)

Conditions: (e.locateCyberSubjects(r.Position) != 0)

Actions: ChangeAttr(s.GranularityLevel, s.Position: r.Position, 'Exact')

This specifies that, in case of risk, if there are persons in the risk area, their exact location should be available to the rescue teams, meaning that the CyberSubjects GranularityLevel at which the observations can be made on the risky area should be equal to the exact location that is available.

One further example is as follows:

RiskEvent : (r.Type: ANY) Conditions : s.PRL > Threshold

Actions: ChangeAttr(o.SensitivityLevel, ((o. ObjType: "HealthData") ∧ (o.Owner: s.ID)) , 'L2')

Where the sensitivity level of health data (o) of a person with PRL higher than a threshold, is reduced, so this data can be available for emergency doctors.

Technology and Scenarios

The proposed approach can be used to develop sensor networks that can be easily deployed on different territories (with various morphologies and infrastructures) and that are adaptable and scalable. The approach concentrates on the design of networks of sensors and actuators to overtake the limited memory, storage and computation of edge devices. This architecture is often referred as *fog computing*, to differentiate from cloud computing where computational power and logic is central. In the fog paradigm, computation occurs the closer to where data is generated and collected. This gives two advantages:

a. It permits to take *intelligent decisions* about when it is convenient to move the computation from the edge to the cloud and the other way around. In this way, it is possible to optimize network capabilities and deal with security and sensitivity of data, having them analyzed at the closest collection point.

b. It also permits to obtain *resilient systems*. These systems, in fact, can operate also in absence of an internet connection by applying local logic and react to the cyberspace situations. This characteristic is very helpful, of course, in the case of an emergency.

So, the approach will concentrate on designing and develop efficient, robust and secure fog computing solutions that can

combine cloud, IoT and data analysis to respond to emergencies, and to prevent them by tagging people when it is possible, e.g., in the occasion of large events (e.g., concerts) where people can be tagged.

The designed network of sensors and actuators could be deployed in unexplored geographical territories to have a mapping and to start collecting useful information about them, to be inserted into a Geographical Information Data Warehouse. This can help during emergencies (like flood, earthquake, fire, ...) to coordinate different interventions and to provide services to rescuing interventions. The designed network can be used also in the scenario of smart cities when, due to traumatic events, it may happen that its infrastructural backbone collapses. In this case, the fog computing network would be able to implement an intelligent edge that can continue to operate even the connection with a central intelligence.

The envisioned impacts for the approach are wide and include benefits for SMEs and public-sector players. The approach will help designing and developing a technological distributed ecosystem, based on Internet of Things, to better support the digitization needs of the near future. By leveraging experience from the academic world and from SMEs and industries, the approach will help public organizations and local administrations to implement state of the art systems for their citizens. In particular, the focus of the approach will be on creating resilient and robust systems that will be able to operate and produce added value also during strong emergencies that can compromise the operation of distributed technologies. Security and privacy of treated data will be a primary issue.

By working on the technology and the social and practical impact of fog computing systems, the approach will assist SMEs and public administrations to design, implement, and operate services and infrastructures based on the cloud and that leverages intelligent edge capabilities. By closing the gap between public sector administrations (e.g. regional administrations, forest and agriculture agencies, fire brigades, etc.) and industry players, the approach will help create the conditions for a positive cooperation and for producing solutions leveraging the contribution of both public and private sectors. This will encourage administrators to develop IoT solutions for their citizens e.g., to map unexplored (e.g., wild) territories and will encourage also private companies to leverage identification technologies to monitor large events. On the other hand, it will encourage private-sector companies and entrepreneurs to develop and offer cloud-based services leveraging the most advanced technologies, such as fog computers, edge technologies, and identification/tracking tags.

Use cases

The approach is designed to be easily deployed during emergencies. In recent years, for different root causes, the number and intensity of natural disasters increased. In many parts of Europe, we experienced critical situations due to catastrophic climatic events: earthquakes in southern Europe, flooding and twisters from seas becoming hotter, fire breakouts during summer. In addition to these events, recently, Europe also faced tensions due to terroristic attacks and mass panicking situations.

During all these events, a rapid and effective response is the key to minimize the impacts and favor the recovery to a normal situation. The approach, by providing the possibility to deploy resilient technological systems easily, will provide a support to the forces involved in the emergency response. The developed system, in fact, will be very quick and easy to deploy and will permit to monitor and collect data on different measures about the surrounding cyberspace. The system will be able to operate with fog computing principles and thus will be able to operate even in the case part of the internet infrastructure is non-operational due to the crisis. The developed system will also be able to communicate with the jackets and tokens (e.g., beacons) of the people involved in rescuing activities and this will give better information about the dislocation of the forces on the territory and will provide a tremendous improvement in their coordination. Also, a technology for tagging people participating in structured events (e.g., sport events, concerts, open air events) will be taken into consideration, with the possibility of tracking the number, position, activities of people who have been registered.

A use case concerns a "smart emergency system" scenario, where some firefighters are progressing inside a dangerous setting being equipped, in a wearable computing context, with special firefighter vests equipped with all sort of sensors. All the information collected is converted into GWEs at the command post; the GWEs can then be processed by the inference procedures proper in order to assure better and quicker reaction to emergency situations such as fire and attacks.

Smart emergency systems applications

Disasters such as fires, floods, earthquakes, civil war, or terrorist attacks may cause crisis situations. Regardless of their origin, crisis situations are often accompanied by uncertainty of how the disaster will develop, and require a rapid pace of response operations given the possibility of serious loss of human lives and property if not responded to properly. Good situational awareness and decision-making support will greatly help minimizing property damage and injury, and saving people's lives. The GWEs (Generalised World Entities) paradigm operating in a platform can be of significant help to provide adequate situational awareness and decision-making support in order to manage crisis situations.

Considering *fire*, the FRSs (Fire and Rescue Service) in Europe must follow a set of strict work procedures in their ER (emergency response) operations, from handling an emergency call, to dispatching emergency response forces, to on-site preliminary situation assessment, and then to crisis response. First responders (firemen) have different priorities during the different stages of their operations. Taking the UK case as an example, ER operations are triggered by "999" calls handled by a command centre of the FRS. The command centre has the ability to dispatch police cars,

ambulances and fire engines. A certain number of fire engines from the nearest and available fire brigades are dispatched to the incident site. The incident commander, or other staff member assigned to arrive on scene, is responsible for making the decisions for scene management and for calling in additional resources if required. Any incident site is physically separated into two parts – an inner and outer cordon. When the first responders arrive at the incident site, they mount an inner cordon around the rescue zone into which only specialy equipped and trained professionals are allowed. One on-site command post is established to control the ER operations and coordinate the interoperation between all of the organizations present including the FRS, police and medical services. The FRS coordinates its own operations within the inner cordon. Other organizations coordinate their activities together with the needs of the FRS within the outer cordon.

A smart emergency system is expected to support the entire emergency operations from the mobilization and the preliminary situation assessment phases to the intervention phase in terms of emergency operation stages. According to what expounded above, the functions of the smart emergency systems include:

i. generation of the original data streams of input entities from a variety of different hardware-based sensors, e.g., for fire, localization (RFID, WIFI, BT), temperature, lights, door positions, movement detection, etc., including instantly deployed or pre-installed wireless sensor nodes, visual and infrared signals, RFID etc.;

ii. a *data fusion and event detection phase*, where the raw data will be first processed (including noise removal and missing data compensation) and recognized, and their essential properties are automatically detected;

iii. a *situation awareness phase*, where the involved resources of any level of complexity are identified;

iv. a *final decision phase*, where the appropriate inference operations are executed for taking decisions about emergency operations, resource allocation, cyber security, and other activities.

A scenario regards *some firemen who are progressing inside a dangerous setting being equipped with fireman vests*, as shown in Figure 2. The essential part of the sensor network is made *mobile*, given that the sensors now equip the firemen's vests-i.e., the vests measure each fireman's pulse rate, breathing rate, body temperature, outside temperature, and collect three-axis gyro and accelerometer data. Each vest also provides geo-location information for the wearer, and measures the available air supply in the fireman's air tank. The vests have a self-contained PAN that interrogates each of the sensors and monitors. The vest codes the fireman's information with the fireman's ID, and then transmits the data to the firemen's command post.

Figure 2: General schema of a smart emergency scenario implying 'equipped' firemen.

A schematic representation of the application's architecture is shown in Figure 3. The results expected from the implementation of this use case concern a better and quicker reaction to a set of general emergency situations of the "fire" type. To give some simple examples:

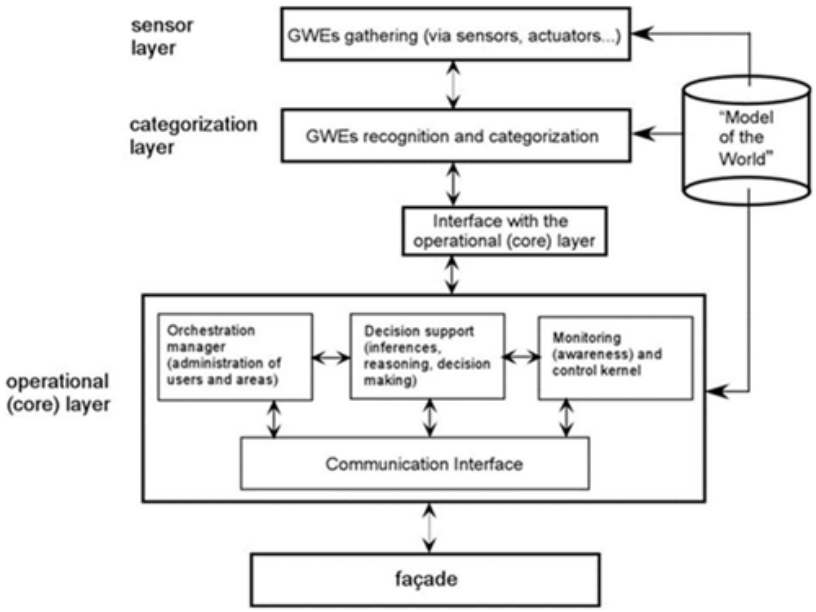


Figure 3: Logical architecture for the smart emergency scenario.

A. If the security officer has some information about the location of a fire (detected by sensors) on a map, including information concerning temperature changes in the neighboring rooms (above, below and alongside), she/he could easily and very quickly know if the fire is already starting to take over to other rooms and react accordingly. Firemen having a current and detailed operational picture could easily find the fastest and safest way to rescue people from a burning danger.

B. An infrared camera on the firemen' helmets could provide on-demand view of fire conditions within the building and the location of the hot spots.

C. The camera on the firemen' helmets could also be used to recognize objects, such as doors, stairs, emergency exits, fire-extinguishers and so on. This information will be used to improve the localization of the fireman and to provide context information of his surrounding environment.

D. All available data of a building will be used to analyze the situation before the operation itself. As an example, firemen need to know if there are still some people inside the building and, in particular, where they are.

AC Policy Checking using a Data Mining Approach

Flaws and problems can affect the Access Control (AC) Policies due to a specific problem that we tackle here: *inconsistency*. Due to coding errors, misunderstandings about access requirements, and so on, often common consistency properties for specific AC policies may not be met causing the discrepancy between what the policy authors intend to specify and what the specified policies reflect. There is an urgent need to find an efficient way to optimize policies and speed up policy evaluation in ACS. Therefore, we can find several studies in the literature that attempt to address the problem of policy checking. Different approaches have been used to fulfill this task. A milestone in the field has been the Margrave tool [45], a software suite for policy analysis. It provides to perform:

a. verification of the properties of a policy

b. elucidation of a policy

c. analysis of the impact of change between two policies, and

d. verification and clarification of a comparison between two policies.

Recently [46], authors have developed some coverage criteria to address policy quality. In [47] a hybrid paradigm is suggested for digital evidence storage that combines ABAC and RBAC using eXtensible Access Control Markup (XACML) throughout the policy statement creation process. The research demonstrates that the ABAC and RBAC models can function in accordance with the developed permit and deny test scenarios.

XDLEngine [48] is a Java-based policy evaluation engine based on three main steps: i) it converts all strings in an XACML policy to numeric values and the strings in a request into their corresponding numeric values; ii) it converts a numericized XACML policy, with a hierarchical structure and multiple complex conflict resolution mechanisms, to an equivalent policy with a flat structure and only one conflict resolution mechanism, which is First-Applicable; iii) it further converts a numerical, normalized policy into a tree structure used to process numerical requests efficiently.

CSRM [49] is another policy evaluation engine; it groups policies to optimize based on merging them through complex computational rules into policies clusters with lowered number of overall PolicySets.

Both tools, CSRM and XEngine, although powerful, show a high complexity in usage and understanding. This makes them not so widely common until now. In this streamline, this paper develops a methodology that would not require a high level of a priori knowledge on Data Mining and, in the meantime, does not require any elaborations on the log's dataset.

A. Using a data mining approach

Data Mining (DM) is the process of identifying correlations, patterns, and trends in large volumes of data. It is a multidisciplinary process that uses statistical techniques but also Artificial Intelligence (AI) and specifically, Machine Learning (ML) technologies to extract useful information for different purposes [50]. We now consider AI and ML in Policy Mining approaches.

Policy Mining

Policy Mining is often done to aid in the migration from one method of AC to another. In order to assist policy administrators when specifying policies, a particularly useful approach is to infer AC rules from existing logs. Among other information, these logs contain tuples denoting the agent, the sensitive resource, the right exercised, and a time-stamp. They may contain access logs that should not have occurred (i.e., under-allocations) in the event of an error in the application, or they may contain only partial permission information (i.e., overallocations).

In existing approaches, Policy Mining is discussed] using custom mining algorithms. For example, [51] presents Xu-Stoller, with an extensive evaluation based on both realistic and synthetic policies. However, the approach has the major limitation of considering only positive permissions ("who can access what"), while many ABAC and RBAC languages also allow negative permissions. Logs may also contain denied access requests due to auditing purposes which are negative examples in the extraction process.

Policy Mining is often done to support the migration from one method of AC to another. In order to assist policy administrators when specifying policies, a useful approach is to infer AC rules from existing logs. Among other information, these logs contain tuples denoting the agent, the sensitive resource, the right exercised, and a time-stamp. They may contain access logs that should not have occurred (i.e., under-allocations) in the event of an error in the application, or they may contain only partial permission information (i.e., overallocations).

In [39,51], we have studied dynamic and risk-prone environments, where security rules should be flexible enough to permit the treatment of risks, and to manage privileges on resources based on the situation at hand. This paper presents a risk-adaptive AC model that adopts hierarchies of contexts and security domains to make adaptations to risks at different levels of criticality. Since various risks may arise simultaneously, two or more security domains might be applicable at the same time incorporating various security rules which might lead to conflicts. Therefore, an approach to analyze conflicts is proposed as a conflict analysis algorithm based on set theory and we illustrate its usage with the proposed risk-adaptive access control model.

Experiments on policy conflicts analysis

The main objective of our approach here is to easily and effectively identify discrepancies between the permissions that the policies give and those that should be granted (see Figure 4). First, a Policy Decision Point (PDP) server and a Policy Administration Point (PAP) containing the policies were created ad hoc. Subsequently, a series of logs were generated from two different sets of policies and used to mine policies from them through ML algorithms. The mined policies should be easier to read and fewer than the original ones, but at the same time statistically correct. Ideally, any misclassified log should be analyzed because it could intrinsically hide a bug in the policy itself.

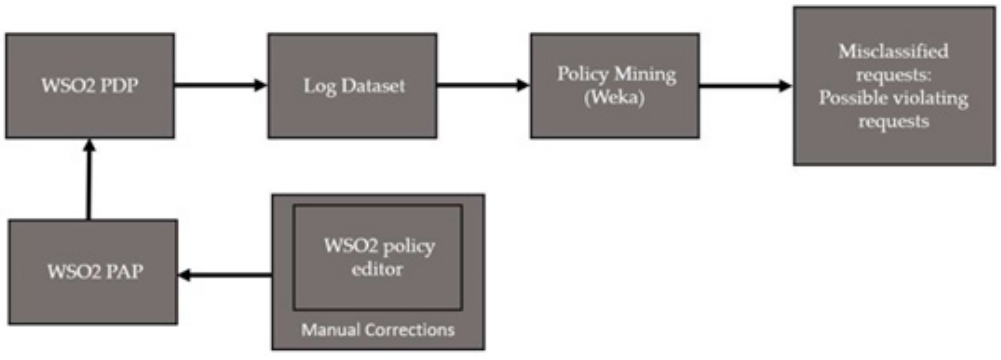


Figure 4: Experiment architecture.

To do this, the *WSO2* Identity Server was used to develop the PDP and PAP, and Weka (Waikato Environment for Knowledge Analysis) was used for the data mining phase on both logs' datasets. WSO2 was chosen because it is a free opensource service (released under Apache 2 license), and it is widely diffused and used by companies. It is also easy to use.

On a dedicated machine, the identity server has been installed and therein the policies uploaded. The policies have been created to be applied specifically in two scenarios:

A. AC for a surgical robot

B. AC in healthcare records management.

In the two scenarios we refer to an ABAC model while maintaining some characteristics of RBAC, such as the presence of some roles. However, our tool can be used with any kind of AC model and for managing access to Both/ logic and physical resources.

Policies in the experiments

In the Surgical Robot scenario, we define two roles: a *Medical Doctor* and an *Assistant*; one Resource: *SurgicalRobot*; and two Actions: Read and *Write*. The policy has to ensure "*that the robot is only accessed during our hospital's normal operating hours*". We therefore assume business hours 08:00 to 24:00. During this time, both the physician and the assistant should have Read privileges on the surgical robot. Therefore, the hypothesis is that surgery occurs between 4:00 pm and 6:00 pm. During these hours, the physician should also have to Write privileges on the robot to control it during surgery. All other access privileges should be denied.

The above is summarized in three predicates:

1. *Pr1: Requests from the Doctor and Assistant during operating room hours should always be allowed.*

2. *Pr2: During operation hours, the Doctor must also have writing privileges.*

3. *Pr3: No combination of attributes exists such that an agent can access the SurgicalRobot outside the allowed time.*

We developed a set of XACML rules that were then combined with the algorithm called *Deny Unless Permit*. This algorithm was selected because it eliminates the ambiguous *NotApplicable* and *Indeterminate cases*, allowing only Permit and Deny response, making the system simpler and possibly with less unwanted errors.

Such a model has been assumed as a more realistic access management model for in a hospital environment. The developed policy is reported in the policies specified at https://fugini.faculty.polimi.it/?page_id=18. In the policy, an error is included. If write and read access outside the operation is requested, the privilege of writing is guaranteed, although this violates Pr3.

In the second experiment, healthcare records management, we define a *MedicaStaff* role and a *Medical Administrator role*. We then define two *Electronic Health Records (EHR1 and EHR2)* as resources, and two actions *Read* and *Write*. We want that the policy grant access during the operating time, so from 08:00 to 24:00, to the Medical Staff only to his/her patient EHR with both right Read and Write. The Medical Administrator shall have only Read privilege to both EHRs.

We summarize this in predicates:

1. *Pr1 Request from Staff to access EHR1 should be granted during the operating time with both access types.*

2. *Pr2 Request from Administrator to access EHR1 and EHR2 with Read privileges should be granted during operating time.*

3. *Pr3 No combination of attributes exists such that an agent can have access outside the allowed time.*

Even in this case, the selected combining algorithm was *Deny If Not Permit*. In the policy, there is unwanted access in different lines. If someone from the Medical Staff asks for access for both EHRs simultaneously, the access is granted for both, even if for Pr1, it should be granted only for one of them.

Log generation

After loading the policies in the PDP, a series of logs have been generated considering all the possible combinations of roles, times, and actions. The total number of cases is: $2(s+a+r) \times t$ where s are subjects, a the actions, r the resources and t the different time slots. This approach guarantees that all possible combinations of the available attributes are generated. Then all possible requests have been evaluated by the PDPcomponent in WSO2, building a series of logs whose output will be: Deny or Permit.

Log analysis: Mining phase

For the mining phase, we selected the Weka (Waikato Environment for Knowledge Analysis) software [36], a collection of ML algorithms for data mining tasks. It contains tools for data preparation, classification, regression, clustering, association rules mining, and visualization. We used a series of ML algorithms dedicated to Classification Learning to categorize each request-response pair and develop new, more coincidental rules. After the first phase of Data Preparation, all logs have been analyzed by Weka.

Algorithm selection

As said previously, ML algorithms can be divided into supervised and unsupervised learning models. In our application model, we need to perform a classification; for this reason, we have excluded a priori any algorithm of the unsupervised type and have dedicated our search and subsequent selection to algorithms of the supervised type only.

Numerous algorithms can be used for classification. In particular, our tool relies on supervised algorithms with decision trees. In Weka, we have several applications of this classification technique. In "A comparative study of Reduced Error Pruning method in decision tree algorithms" compared the J48 REPTree PART RIDOR and JRip algorithms. The first two are tree- based classifiers generating as output a tree structure, while PART, RIDOR, and JRip are rule-based classifiers generating as output an if-then statement. For this reason, these three have been chosen to develop our framework; moreover, from the study, it results that for algorithms based on rules, Ridor gives the highest accuracy of classification while JRip gives the best performance in terms of complexity of tree structure (Figures 5 & 6; for details, see https://fugini.faculty.polimi.it/?page_id=18).


This class implements a propositional rule learner, Repeated Incremental Pruning to Produce Error Reduction (RIPPER), which was proposed by William W

Cohen as an optimized version of IREP.

The algorithm is briefly described as follows:

Initialize RS = {}, and for each class from the less prevalent one to the more frequent one, DO:

1. Building stage:
Repeat 1.1 and 1.2 until the descrition length (DL) of the ruleset and examples is 64 bits greater than the smallest DL met so far, or there are no positive examples, or the error rate >= 50%.

1.1. Grow phase:
Grow one rule by greedily adding antecedents (or conditions) to the rule until the rule is perfect (i.e. 100% accurate). The procedure tries every possible value of each attribute and selects the condition with highest information gain: p(log(p/t)-log(P/T))

1.2. Prune phase:
Incrementally prune each rule and allow the pruning of any final sequences of the antecedents;The pruning metric is (p-n)/(p+n) -- but it's actually 2p/(p+n) -1, so in this implementation we simply use p/(p+n) (actually (p+1)/(p+n+2), thus if p+n is 0, it's 0.5)

2. Optimization stage:
after generating the initial ruleset {Ri}, generate and prune two variants of each rule Ri from randomized data using procedure 1.1 and 1.2. But one variant is generated from an empty rule while the other is generated by greedily adding antecedents to the original rule. Moreover, the pruning metric used here is (TP+TN)/(P+N).Then the smallest possible DL for each variant and the original rule is computed. The variant with the minimal DL is selected as the final representative of Ri in the ruleset.After all the rules in {Ri} have been examined and if there are still residual positives, more rules are generated based on the residual positives using Building Stage again.
3. Delete the rules from the ruleset that would increase the DL of the whole ruleset if it were in it, and add resultant ruleset to RS.
ENDDO

Note that there seem to be 2 bugs in the original ripper program that would affect the ruleset size and accuracy slightly. This implementation avoids those bugs and thus is a little bit different from Cohen's original implementation. Even after fixing the bugs, since the order of classes with the same frequency is not defined in ripper, there still seems to be some trivial difference between this implementation and the original ripper, especially for auto&ology data in UCI repository, where there are lots of classes of few instances.

Details please see:

William W. Cohen: Fast Effective Rule Induction. In: Twelfth International Conference on Machine Learning, 115-123, 1995.

PS: We have compared this implementation with the original ripper implementation in aspects of accuracy, ruleset size and running time on both artificial data "ab+bcd+defg" and UCI datasets. In all these aspects it seems to be quite comparable to the original ripper implementation. However, we didn't consider memory consumption optimization in this implementation.

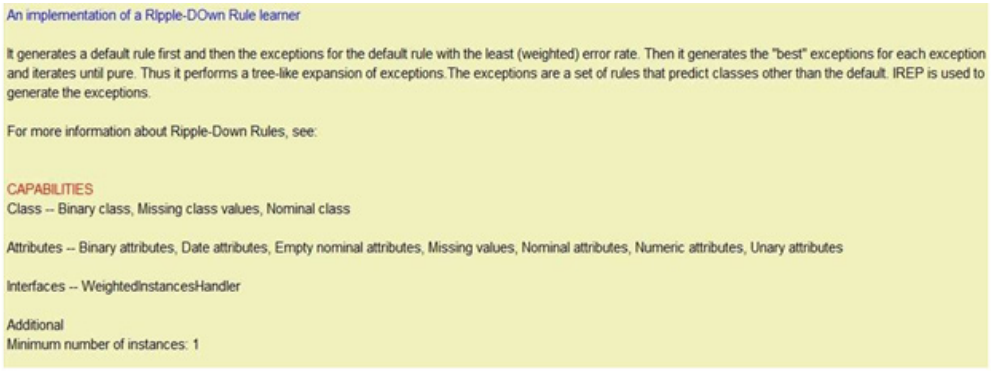
CAPABILITIES
Class -- Binary class, Missing class values, Nominal class

Attributes -- Binary attributes, Date attributes, Empty nominal attributes, Missing values, Nominal attributes, Numeric attributes, Unary attributes

Interfaces -- WeightedInstancesHandler

Additional
Minimum number of instances: 3

Figure 5: JRip algorithm.


An implementation of a Ripple-DOwn Rule learner

It generates a default rule first and then the exceptions for the default rule with the least (weighted) error rate. Then it generates the "best" exceptions for each exception and iterates until pure. Thus it performs a tree-like expansion of exceptions.The exceptions are a set of rules that predict classes other than the default. IREP is used to generate the exceptions.

For more information about Ripple-Down Rules, see:

CAPABILITIES
Class -- Binary class, Missing class values, Nominal class

Attributes -- Binary attributes, Date attributes, Empty nominal attributes, Missing values, Nominal attributes, Numeric attributes, Unary attributes

Interfaces -- WeightedInstancesHandler

Additional
Minimum number of instances: 1

Figure 6: Ridor algorithm.

Validation

Both algorithms were validated through cross-validation.

The total number of cases is: 2 $(s+a+r) \times t$ where s are subjects a the total number of actions, r are the resources, and t the number of time slots.. This approach guarantees that all possible combinations of the available attributes are generated. Then all possible requests have been evaluated by our PDP in WSO2, building a series of logs whose output will be: Deny or Permit. The chosen configuration is reported in Table 1-6, (Figure 7-19). All the specifications and the tests are contained in https://fugini.faculty.polimi.it/?page_id=18.

Table 1: Validation hardware configuration.

OS Name	Microsoft Windows 10 Pro
System Type	x64-based PC
CPU	Intel(R) Core (TM) i7-10875H CPU @ 2.30GHz, 8 Core(s)
Installed Physical Memory (RAM)	32.0 GB

Table 2: Accuracy of the two algorithms.

Correctly classified	Jrip	Ridor
Experiment 1	83.33%	83.33%
Experiment 2	91.66%	86.11%

Table 3: JRIP accuracy experiment 1.

JRIP	Precision	Recall	ROC Area	MCC	Class
	0.688	0.786	0.858	0.616	P
	0.906	0.853	0.858	0.616	D
Weighted Avg.	0.842	0.833	0.858	0.616	

Table 4: Ridor accuracy experiment 1.

RIDOR	Precision	Recall	ROC Area	MCC	Class
	0.75	0.643	0.777	0.582	P
	0.861	0.912	0.777	0.582	D
Weighted Avg.	0.829	0.833	0.777	0.582	

Table 5: JRIP accuracy experiment 2.

JRIP	Precision	Recall	ROCarea	MCC	Class
	0.833	0.909	0.936	0.81	P
	0.958	0.92	0.936	0.81	D
Weigthed Avg.	0.92	0.917	0.936	0.81	

Table 6: Ridor accuracy experiment 2.

RIDOR	Precision	Recall	ROCarea	MCC	Class
	0.731	0.864	0.862	0.694	P
	0.935	0.86	0.862	0.694	D
Weighted Avg.	0.872	0.861	0.862	0.694	

```
JRIP rules:
===========

(R = 1) and (T(8-00) = 1) => out=P (16.0/4.0)
(T(16-18) = 1) and (Doc = 1) and (W = 1) => out=P (2.0/0.0)
 => out=D (30.0/0.0)

Number of Rules : 3


Time taken to build model: 0 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances          40               83.3333 %
Incorrectly Classified Instances         8               16.6667 %
Kappa statistic                          0.6129
Mean absolute error                      0.1877
Root mean squared error                  0.3806
Relative absolute error                 44.8023 %
Root relative squared error             83.3106 %
Total Number of Instances               48
```

Figure 7: JRip Output.

```
RIpple DOwn Rule Learner(Ridor) rules
--------------------------------------

out = P  (48.0/34.0)
          Except (T(8-00) = 0) => out = D  (13.0/0.0) [3.0/0.0]
          Except (R = 0) and (W = 0) => out = D  (6.0/0.0) [2.0/0.0]
          Except (R = 0) and (Doc = 0) => out = D  (3.0/0.0) [1.0/0.0]

Total number of rules (incl. the default rule): 4

Time taken to build model: 0 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances          40               83.3333 %
Incorrectly Classified Instances         8               16.6667 %
Kappa statistic                          0.5789
Mean absolute error                      0.1667
Root mean squared error                  0.4082
Relative absolute error                 39.7886 %
Root relative squared error             89.3648 %
Total Number of Instances               48
```

Figure 8: Ridor Output.

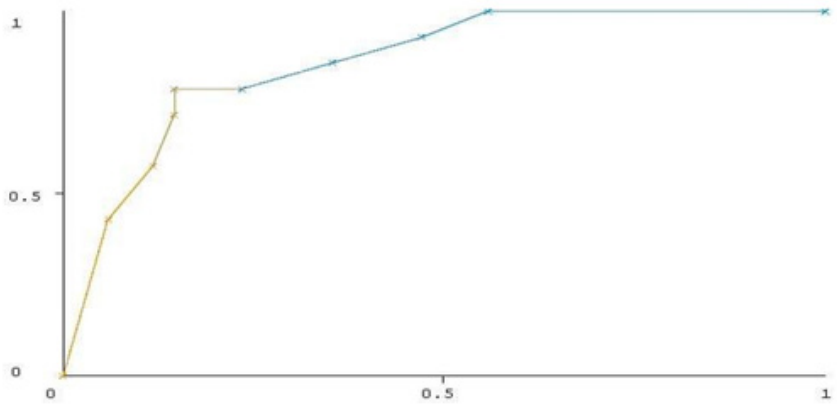


Figure 9: ROC curve JRip experiment 1.

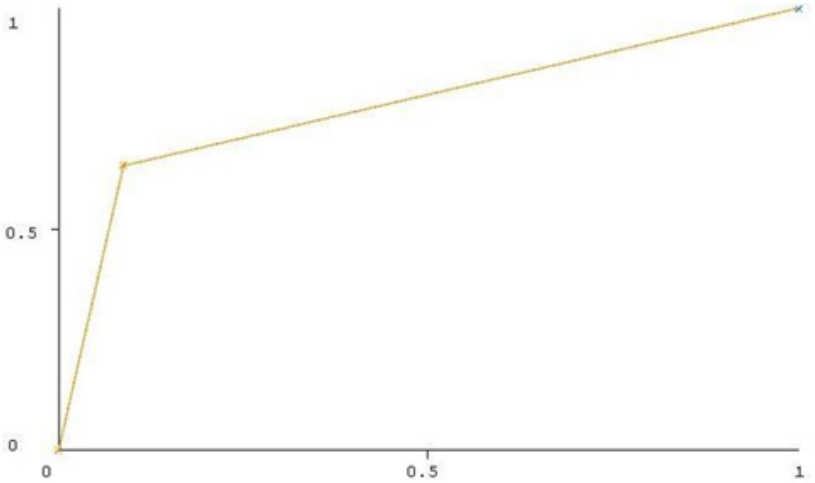


Figure 10: ROC curve Ridor experiment 1.

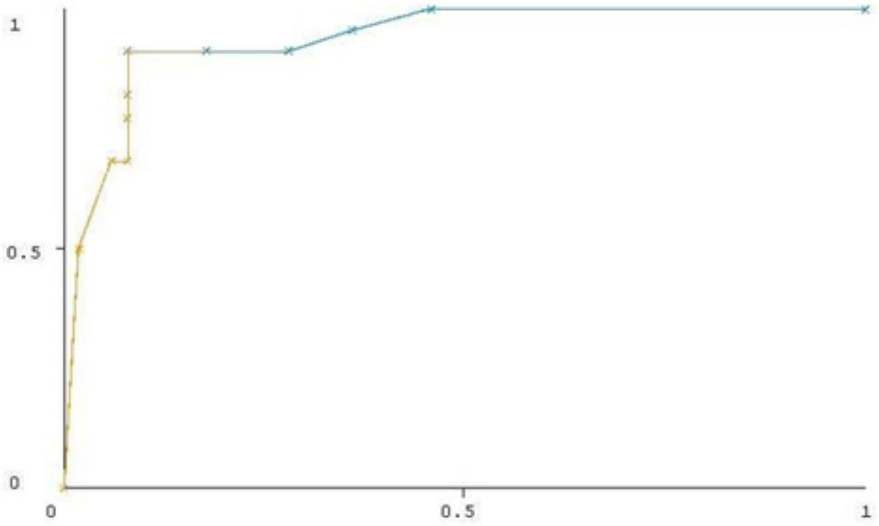


Figure 11: ROC curve JRip experiment 2.

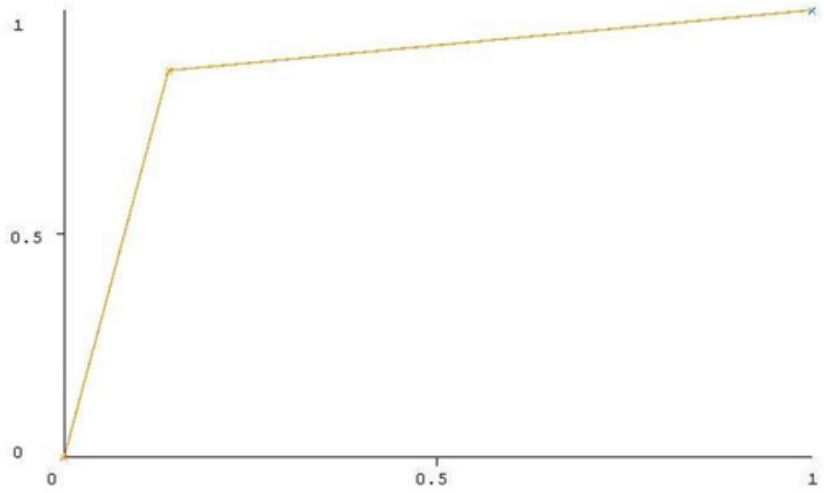


Figure 12: ROC curve Ridor experiment 2.

```
out = P  (48.0/34.0)
          Except (T(8-00) = 0) => out = D  (13.0/0.0) [3.0/0.0]
          Except (R = 0) and (W = 0) => out = D  (6.0/0.0) [2.0/0.0]
          Except (R = 0) and (Doc = 0) => out = D  (3.0/0.0) [1.0/0.0]
```

Figure 13: Ridor generated rules experiment 1.

```
(R = 1) and (T(8-00) = 1) => out=P (16.0/4.0)
(T(16-18) = 1) and (Doc = 1) and (W = 1) => out=P (2.0/0.0)
 => out=D (30.0/0.0)
```

Figure 14: JRip generated rules experiment 1.

```
Plot : weka.classifiers.rules.JRip
Instance: 26
                Doc : 1
                Ass : 0
                  R : 1
                  W : 1
           T(16-18) : 0
            T(8-00) : 1
            T(00-8) : 0
prediction margin : -0.9310344827586208
    predicted out : D
                out : P
```

Figure 15: JRip misclassified request.

```
Plot : weka.classifiers.rules.Ridor
Instance: 26
                Doc : 1
                Ass : 0
                  R : 1
                  W : 1
           T(16-18) : 0
            T(8-00) : 1
            T(00-8) : 0
prediction margin : -1.0
    predicted out : D
                out : P
```

Figure 16: Ridor misclassified request experiment 1.

```
Plot : weka.classifiers.rules.Ridor
Instance: 2
                Doc : 0
                Ass : 1
                  R : 1
                  W : 1
           T(16-18) : 0
            T(8-00) : 1
            T(00-8) : 0
prediction margin : -1.0
    predicted out : D
                out : P
```

Figure 17: Ridor misclassified request experiment 1.

```
Plot : weka.classifiers.rules.JRip
Instance: 11
     ï»¿MediStaff : 1
          MediAdm : 1
                R : 0
                W : 1
             EHR1 : 1
             EHR2 : 1
           T(8-24) : 1
prediction margin : -0.9555555555555
    predicted Out : D
              Out : P
```

Figure 18: JRip misclassified request experiment 2.

```
Plot : weka.classifiers.rules.Ridor
Instance: 53
      ï»¿MediStaff : 1
           MediAdm : 0
                 R : 0
                 W : 1
              EHR1 : 1
              EHR2 : 1
           T(8-24) : 1
prediction margin : -1.0
    predicted Out : D
               Out : P
```

Figure 19: Ridor misclassified request experiment 2.

Concluding Remarks

In the last years, the number of Internet-connected devices has dramatically increased. Nowadays, they are present in any environment, even in the critical structures, such as in the medical and hospital sector. This increase, combined with the spread of robotics and IoT technologies, led to the development of CPS systems. Such systems require more advanced access management methods than the traditional ones. Models such as ABAC and RBAC are currently gaining momentum in system development, mainly due to their flexibility and dynamism, which have become key features for system security. However, these access models require complex policies, and can easily hide bugs and vulnerabilities that are often difficult to detect without the use of complex tools. Research has been mobilizing for years to develop more advanced systems for creating and analyzing suitable policies. Unfortunately, despite their undoubted effectiveness, these tools are almost never easy or immediate to use. The novelty of the tool presented in this paper is mainly related to the simplicity and ease of use. A reliable, but at the same time agile and easy to use tool, can help the system administrators in the delicate task of identifying access requests that can hide bugs in the policies of the system itself. Exploiting the predictive power of ML algorithms, it was thus possible to identify bugs present in the examined policies that allowed unwanted accesses. Such bugs have been identified by our tool, in two sets of policies specifically developed in the medical-hospital sector, to manage a robot in a surgical environment and the consultation and compilation of medical records. The results show that using it DM is possible to identify requests that expose bugs reliably. This tool was developed to find a secure way to quickly and easily identify possible bugs in policies that could allow unwanted access to very sensitive resources. The limits of this work are associated with the impossibility, so far, to test the methodology developed directly in real case scenarios and not only in theoretical ones.

Testing in real environments is important to gain a broader vision of the real limits of the method in various environments of possible future use, and would allow to perform all the necessary adjustments to make the tool increasingly safe and versatile. In future work it would be interesting to test how the methodology presented here can really be used with more complex policies and in very different and heterogeneous environments, testing its ability to detect bugs and unwanted accesses in policies already in use and especially in real situations.

Moreover, the proposed algorithms should be tested on larger datasets and with more complex systems where accesses are defined by a greater number of attributes. A further interesting development would be to try to expand the capabilities of our tool to create models that, once identified the bag, can directly suggest possible changes and corrective actions.

The results expected from the implementation of the Cyber Security Model in these use cases will concern a better and quicker reaction to a set of general emergency situations of the “fire” type. To give some simple examples:

Concerning Risk, if the safety officers have some information about the location of a fire (detected by sensors) on a map, including information concerning temperature changes in the neighboring rooms, she/he could easily and very quickly know if the fire is already starting to take over to other rooms and react accordingly. Firemen having an updated and detailed operational picture could easily find the fastest and safest way to rescue people from a burning danger.

An infrared camera on the firemen’ helmets could provide on-demand view of fire conditions within the building and the location of the hot spots. The camera on the firemen’ helmets could also be used to recognize CyberObjects, such as doors, stairs, emergency exits, fire-extinguishers. This information will be used to improve the localization of the fireman and to provide context information of his surrounding cyberspace.’

Concerning *Cyber Security*, all available data of a building will be used to analyze the situation before the operation itself. As an example, firemen need to know if there are still some people inside the building and, in particular, where they are.

Eventually, a last demonstrator will concern the setup of an integrated "smart surveillance system" able to ensure optimal monitoring and control against threats or malicious attacks of highly secured buildings. These can correspond to critical infrastructures having very restricted access and high security requirements like embassies, government building, train stations, power plants, etc. The main idea is to take a step forward in the integration of security subsystems using a centralized command and control system that creates an 'awareness picture' of the situation in the building. This is achieved by collecting first all the information provided by the security subsystems-token/smart cards, vision (tracking, abnormal behavior), voice recognition, Indoor-Location Based-Systems (ILS), monitoring & control and also rule-based systems for decision making. The corresponding GWEs will be correlated then by inference to establish a snapshot of what is the real state of the building at a specific moment.

Acknowledgment

This work was partially supported by the European Union's Horizon 2020 research and innovation program me under grant agreement No. 826232, project Working Age (Smart Working environments for all Ages).]

References

1. Kitchin R, Dodge M (2017) The (in) security of smart cities: Vulnerabilities, risks, mitigation, and prevention. In Smart cities and innovative Urban technologies 26: 47-65.
2. Houichi M, Jaidi F, Bouhoula A (2022) Analysis of smart cities security: Challenges and advancements. 15th International Conference on Security of Information and Networks (SIN), IEEE.
3. Wu J (2022) Problems and solutions regarding generalized functional safety in cyberspace. Security and Safety 1(2022): 13.
4. Nanda S, Panigrahi CR, Pati B (2023) Emergency management systems using mobile cloud computing: A survey. International Journal of Communication Systems 36(12): e4619.
5. Fang Z, Yue P, Zhang M, Xie J, Wu D, et al. (2023) A service-oriented collaborative approach to disaster decision support by integrating geospatial resources and task chain. International Journal of Applied Earth Observation and Geoinformation 117: 103217.
6. Kumar A (2020) Framework for realization of green smart cities through the Internet of Things (IoT). Trends in Cloud-based IoT pp. 85-111.
7. Gupta M, Bhatt S, Alshehri AH, Sandhu R (2022) Access control models and architectures for IoT and cyber physical systems. Cham, Switzerland, Springer.
8. Hu, Vincent C (2015) Attribute-based access control. Computer 48(2): 85-88.
9. Paton D, Buergelt P (2019) Risk, transformation and adaptation: Ideas for reframing approaches to disaster risk reduction. International journal of Environmental Research and Public Health 16(14): 2594.
10. Doan TT (2024) Smart city cyber resilience: Your perception matters. Friesen Press.
11. Berg B, Kuipers S (2022) Vulnerabilities and cyberspace: A new kind of crisis. Oxford Research Encyclopedia of Politics.
12. Algarni AM, Thayananthan V, Malaiya YK (2021) Quantitative assessment of cyber security risks for mitigating data breaches in business systems. Applied Sciences 11(8): 3678.
13. Khiralla FAM (2020) Statistics of cybercrime from 2016 to the first half of 2020. International Journal of Computer Science and Network 9(5): 252-261.
14. Rodrigues GAP, Serrano ALM, Vergara GF, Albuquerque RO, Nze GDA (2024) Impact, compliance, and countermeasures in relation to data breaches in publicly traded us companies. Future Internet 16(6): 201.
15. Alli AA, Kassim K, Mutwalibi N, Hamid H, Ibrahim L (2021) Secure fog-cloud of things: Architectures, opportunities and challenges. Secure Edge Computing pp. 3-20.
16. Jamal N, Zain JM (2022) A review on nature, cybercrime and best practices of digital footprints. 2022 International Conference on Cyber Resilience (ICCR), IEEE pp. 1-6.
17. Chiquito A, Bodin U, Schelén O (2023) Attribute-based approaches for secure data sharing in industrial contexts. IEEE Access 11: 10180-10195.
18. Mishra A, Jha AV, Appasani B, Ray AK, Gupta DK, et al. Emerging technologies and design aspects of next generation cyber physical system with a smart city application perspective. International Journal of System Assurance 14: 699-721.
19. Swuste P, Groeneweg J, Gulijk CV, Zwaard W, Lemkowitz S, et al. (2020) The future of safety science. Safety science 125: 104593.
20. Toma A, Negura FG, Moraru RI, Mureşan F, Racautanu M (2024) Behavior-based safety, the right long-term approach to address workplace safety. MATEC Web of Conferences 389: 1-10.
21. Kampourakis V, Gkioulos V, Katsikas S (2023) A systematic literature review on wireless security testbeds in the cyber- physical realm. Computers & Security 133: 103383.
22. Weyns D (2020) An introduction to self-adaptive systems: A contemporary software engineering perspective. John Wiley & Sons.
23. Malathi C, Padmaja IN (2023) Identification of cyber-attacks using machine learning in smart IoT networks. Materials Today: Proceedings 80: 2518-2523.
24. Mentges A, Halekotte L, Schneider M, Demmer T, Lichte D (2023) A resilience glossary shaped by context: Reviewing resilience-related terms for critical infrastructures. International Journal of Disaster Risk Reduction 96: 103893.
25. Knight, Kevin W (2010) AS/NZS ISO 31000: 2009-the new standard for managing risk. Keeping Good Companies 62(2): 68- 69.
26. Purdy G (2010) ISO 31000: 2009--setting a new standard for risk management. Risk Analysis: An International Journal 30(6): 881-886.
27. Zamecka A, Buchanan G (1999) Book review applying risk management standards to disaster management.
28. Gupta M, Bhatt S, Alshehri AH, Sandhu R (2022) Authorization frameworks for smart and connected ecosystems. Access Control Models and Architectures for IoT and Cyber Physical Systems pp. 39-61.
29. Viktoriia K, Rodica TM, Lilia D, Nataliia T, Iulita B (2023) Development of critical infrastructure from the point of view of information security. Univers Strategic 53(1): 170.
30. Samarakkody A, Amaratunga D, Haigh R (2022) Characterising smartness to make smart cities resilient. Sustainability 14(19): 12716.
31. Shen B (2023) A survey of access control misconfiguration detection techniques. arXiv preprint arXiv, 2304.07704.
32. Xu D, Shrestha R, Shen N, Automated Coverage-Based Testing of XACML Policies. in Proceedings of the 23ndACM Symposium on Access Control Models and Technologies, Indianapolis Indiana USA, pp. 314.
33. Maulin A, Rasjid ZE (2024) Unified access management for digital evidence storage: Integrating attribute-based and role-based access control with XACML. International Journal of Advanced Computer Science & Applications 15(3): 1345-1353.

34. Liu AX, Chen F, Hwang J, Xie T (2008) Xengine: A fast and scalable XACML policy evaluation engine. ACM SIGMETRICS Performance Evaluation Review 36(1): 265-276.

35. Deng F, Yu Z, Zhang L, Ge X, Zhao R, et al. (2020) Clustering and supervised response for XACML policy evaluation and management. Knowledge-Based Systems 205: 106312.

36. Amiri Z, Heidari A, Navimipour NJ, Unal M, Mousavi A (2024) Adventures in data analysis: A systematic review of deep learning techniques for pattern recognition in cyber-physical-social systems. Multimedia Tools and Applications 83: 22909-22973.

37. Chhetri P, Bhatt S, Bhatt P, Nobi MN, Benson J, et al. (2024) Environment aware deep learning-based access control model. In Proceedings of the 2024 ACM Workshop on Secure and Trustworthy Cyber-Physical Systems pp. 81- 89.

38. Karimi L, Aldairi M, Joshi J, Abdelhakim M (2021) An automatic attribute-based access control policy extraction from access logs. IEEE Transactions on Dependable and Secure Computing 19(4): 2304-2317.

39. Teimourikia M, Marilli G, Fugini M (2016) Context-based risk-adaptive security model and conflict management. DEXA 2016: Proceedings, Part I, 27th International Conference on Database and Expert Systems Applications 9827: 121-135.

40. Amarilli F, Amigoni F, Fugini MG, Zarri GP (2017) A semantic-rich approach to IoT using the generalized world entities paradigm. Managing the Web of Things pp. 105-147.

41. Karjoth G, Schade A (2008) Implementing ACL-based policies in XACML. In 2008 Annual Computer Security Applications Conference (ACSAC), IEEE pp. 183-192.

42. Fugini M, Teimourikia M, Hadjichristofi G (2016) A web-based cooperative tool for risk management with adaptive security. Future Gener Comput Syst 54: 409-422.

43. Miculan M, Pasqua M (2022) Distributed programming of smart systems with event-condition-action rules. CEUR Workshop Proceedings 3284: 201-206.

44. MacPherrgrason-Krutsky CC, Brand BD, Lindell MK (2020) Does updating natural hazard maps to reflect best practices increase viewer comprehension of risk? International Journal of Disaster Risk Reduction 46: 101487.

45. Margrave (2021) http://www.margrave-tool.org/v1+v2/margrave/versions/01-01/index.html.

46. Tsohou A, Diamantopoulou V, Gritzalis S, Lambrinoudakis C (2023) Cyber insurance: State of the art, trends and future directions. International Journal of Information Security 22(3): 737-748.

47. Deng F, Yu Z, Liu W, Luo X, Fu Y, et al. (2021) An efficient policy evaluation engine for XACML policy management. Information Sciences 547: 1105-1121.

48. Kure H (2021) An Integrated cybersecurity risk management (I-CSRM) framework for critical infrastructure protection.

49. Delen D (2020) Predictive analytics: Data mining, machine learning and data science for practitioners. FT Press, USA.

50. Nobi MN, Gupta M, Praharaj L, Abdelsalam M, Krishnan R, et al. (2022) Machine learning in access control: A taxonomy and survey. arXiv preprint arXiv:2207.01739.

51. Teimourikia M, Fugini M (2017) Ontology development for run-time safety management methodology in smart work environments using ambient knowledge. Future Generation Computer Systems 68: 428-441.