Integrated Physical Security System in Army Units

Fernando Augusto da Costa Batista, 83381, MEEC
Academia Militar/Instituto Superior Técnico
e-mail : fernando200721@gmail.com

Abstract— Physical Protection Systems are used to achieve the desired levels of Physical Security in specific areas or buildings of the Army Units. These systems are composed by the functions of detection, delay and response, being used several types of sensors for the detection function. However, the lack of a mechanism capable of unifying the data produced by all sensors results in the fact that both the detection and the notification given to security managers are done in a fragmented and inefficient way.

This paper aims to describe and characterize a system developed for the collection, relation, storage and notification of the events generated by the various sensors, to identify anomalous situations and to notify them more efficiently to the security elements.

Since the Armed Forces have specific needs, it is important to consider their objectives, procedures and threat model. The integration of several monitoring systems and the possibility of using multiple sensor data fusion will be studied through models that have already been developed with application in other areas, allowing a transposition and application in security.

Index Terms—physical security, sensors, fusion, information

I. INTRODUCTION

Military units currently implement various mechanisms used to control physical access, both to the unit itself and to certain areas of the unit, to guarantee the safety levels defined in the SEG MIL military regulation [1]. These include sensors (smoke detection, motion detection, access card readers, video surveillance cameras, among others) and classic methods of physical security such as physical obstacles, specific building materials and patrols/rounds by the military to the perimeter of the unit and to the facilities. In each unit, there is a duty force responsible for security, intervening whenever any anomalous event is detected, which is reported through the channels established for that purpose.

Currently, if an intrusion is detected by one of the sensors, it triggers an alarm, which can be audible or visual (via a warning light on a monitoring console), to notify the security force and then it can intervene. However, the fact that the various existing sensors and video surveillance cameras are not integrated and managed through a central console and the occurrence of unstructured events, such as telephone calls, make monitoring increasingly complex.

The procedures used show that there is room for improvement: some systems are closed and it is not possible to collect and centralize the events; The notifications are rather incomplete and do not provide relevant information to the security team to enhance their response; There is no mechanism used capable of unifying alarm signals; The administration and control procedures are done manually and costly.

As a consequence of the growing difficulty of monitoring, there is an increase in the time elapsed between the existence of an anomalous event and the notification of a responsible force, which is a critical factor that should be sought to reduce, thus enabling a faster and more adequate response by those responsible.

Thus, the motivation to develop a solution that meets existing needs was created through a five-pronged platform:

- Collection of the events generated by the existing systems and equipment in the Units;
- Processing of the collected events and generation of alarms through the identification of anomalous situations;
- Efficient notification of the Duty Officer of the unit where the occurrence took place;

- Automation of the production of Counterinformation and Security reports;
- Centralization of the history of events in the unit responsible for storing occurrences and their reports.

In this way, it is possible to give a faster response to occurrences that currently take considerable time to be detected and in turn solved, providing an event management tool, which can be used by entities responsible for unit security and by the upper echelon for purposes of management and control. It also streamlines any existing bureaucratic process since an event occurs until the produced report is stored.

II. STATE OF THE ART

A. Military Security

The purpose of Military Security is to study and propose measures of local protection, internal and external, of its Unit, Establishment or Organ (U/E/O), against any actions that represent a threat, of intentional or accidental origin/catastrophe. The Portuguese Army addresses Military Security in four strands: Information Security, Personnel Security, Physical Security and Cyberspace Security (INFOSEC - Information Security). This paper is part of the Physical Security section, contributing per guidelines for the fulfillment of the Army's mission [1].

To better understand the context in which this work is inserted and the necessity of its accomplishment, it is important to know that the model developed by the Army defines that the threat may originate from Hostile Information Services (HOIS), subversive organizations or groups, terrorist groups or individuals, natural accidents, surveillance and information gathering devices among others. In this sense, all activities must be monitored and evaluated, insofar as they can indicate hostile capacities and intentions [1] [2].

In general, security is achieved when facilities, information, personnel, material and activities are protected against the threat, and it is important to consider that all Physical Security measures indirectly contribute to the security of the Information, Staff and INFOSEC.

Each U/E/O establishes a Physical Safety program along the lines of active and passive safety measures, deciding what degree of physical protection to adopt in each building, taking into account aspects such as [1]:

- the degree of classification and the nature of the information to be protected;
- the volume and type of substances to be safeguarded;
- The security clearance and the need to know the personnel;
- Evaluating threats posed by hostile intelligence services, sabotaging terrorist and/or subversive and criminal activities.

Physical security measures involve the use of guards and reaction forces, as well as structural and technical measures, such as physical obstacles placed in depth, to make it difficult to attempt to penetrate safety systems. Measures are also envisaged to avoid possible intrusions through electronic, acoustic and optical devices [2].

Whenever there is an anomalous safety event, the necessary action is taken to nullify/minimize its effects and a Security Report is drawn up which should include a description of the incident or activity and the assessment of its significance. This report is analyzed later at the level of the U/E/O where the event occurred.
and sent to the Security Coordinating Center of the Army (CCSE), unit responsible for centralizing all this information [1].

Considering the mentioned procedures, the proposed system in this paper will aid in the detection of possible intrusions and in the control of access to restricted or prohibited areas, increasing the internal and external physical protection.

B. Physical Protection System

Physical Protection Systems (PPS) are designed to combine elements such as fences, barriers, sensors, procedures, communication devices, and security personnel to meet Physical Security objectives. The primary functions of a PPS are the detection of an adversary, the delay of this adversary and the response triggered by the security personnel, as shown in Figure 1 [3].

![Fig. 1. Functions of a Physical Protection System [3].](image)

In order to evaluate a physical protection system, it is essential to consider and understand its functions, since these are the ones that allow to evaluate the system based on the effectiveness with which they are implemented. It is important to understand that the functions are sequential and that the protection system is as good as its weakest function, so there is no point in having a very efficient response function if the detection capability is weak or non-existent.

Focusing on the detection function, intrusion sensors play a big role creating overlapping layers of detection, supporting each other in the event of any failure, being of extreme importance to assure a good communication and evaluation of generated alarms to improve the system.

The proposed solution focuses on improving the functions of detection and response, through its components of alarm assessment and communication to response force, respectively.

C. Integration of Physical Protection Subsystems

The PPS are composed of different subsystems such as Video Management Systems (VMS), Intrusion Detection Systems (IDS), Access Control Systems (ACS), among others, being each one managed through an independent platform. The integration of the various sensors allows the analysis and interpretation of a larger amount of data together, making possible the identification of patterns that suggest abnormal behavior, resulting in a more adequate prevention and a faster response to any event.

In Tokyo, to anticipate the risk of terrorism that may arise in 2020 at the international sports event, Hitachi has proposed in [6] an integrated platform that allows the centralized management of information from various systems, such as IDS and ACS.

Through the use of an integrated management layer, this approach solves some of the problems pointed out, allowing the analysis and interpretation of the information together, which enables the identification of more complex situations while providing more information about them. During a fire at a facility, it is possible to know that a person is still inside one of the work rooms as a result of information intersection provided by the fire detector, the ACS and the VMS that registered that person's entry, making it a more efficient system and providing presentation of the results in an integrated way.

D. Multi Sensor Data Fusion

To take advantage of a platform that integrates multiple sensors in a cooperative way, the need arises to use a technique that can transform data provided by a sensor into more useful information through the Multiple Sensor Data Fusion (MSDF) [5]. The use of MSDF allows to make inference that can be used to turn the system performance superior to what the system components could achieve separately [6]. However, the implementation of the merger can be done in several ways because it is very specific to each problem.

Before establishing the system where the MSDF will be done, it is important to consider some aspects [7]:

- Sensor distribution: it influences aspects such as the type of communication used between sensors and central module, as well as the possibility of providing some type of redundancy in the event of any sensor failing or being necessary to confirm values;
- Nature of the sensors used: if the sensors used are of different types, then the information collected from these sensors needs to be formatted whenever possible so that the data is standardized;
- Processing capacity in the sensors: an aspect that can determine the type of architecture to be used, since if the sensors have sufficient processing capacity, it is possible to use a distributed architecture, which, despite being more complex in its implementation, offers a greater scalability;

E. Frameworks

Esteban proposed a framework in [7] for data fusion that uses three steps in the analysis of the system: identification, estimation and validation, represented in Figure 2.

![Fig. 2. Schematic representation of the framework proposed by Esteban [7].](image)

Mitchell's proposal is composed by modules, each one representing a function of the framework composed by three levels of domain, being: physical, informative and cognitive as presented on Figure 3. The control block is represented by the sensor manager [6].

![Fig. 3. MSDF framework proposed by Mitchell with sensor manager [6].](image)

The two proposed frameworks are composed of three levels, and in both the first level addresses the physical component and the sensors, the second focuses on the interpretation of the data fusion. However, the third level differs in the sense that Mitchell proposes the information to be presented via the Human Machine Interface (HMI) to a human operator and Esteban proposes a validation of the processed data through benchmark and performance assessment. In both models, it is represented the need to make a constant
adjustment of the sensors and a control of the system through a closed cycle.

F. Home Automation

Since home automation deals with different types of equipment as we do in the present work, exists the need to interconnect these equipments in order to facilitate their management. In this sense, two architectures proposed in home automation were analyzed to make an analogy of the problems encountered and thus help in the creation of a solution to the existing problems in the area of security systems.

It has been proposed in [8] a system capable of integrating heterogeneous devices of a home automation system and functioning as a Gateway that allows the exchange of information between devices, enabling their cooperation. Figure 4 represents the general architecture of a Domotic House Gateway (DHG).

![Architecture of a Domotic House Gateway](image)

The architecture represented can be divided into three main layers: the top layer that surrounds all the home automation devices that can be connected to the DHG through its drivers; The central layer that is responsible for routing low-level event messages between the DHG and the various devices; The last layer consists of the intelligence of the system and is dedicated to the management of events at the logical level, where there is a core based on rules that defines the reactions that must be triggered by the actuators for each received event. These rules can be adjusted dynamically by the system itself through user interfaces.

Considering that most of the approaches used are quite expensive, it was proposed by Renato Nunes in [9] a DomoBus architecture, a more economical solution because modules based on inexpensive microcontrollers that manage to control 16 or more devices are used. Figure 5 illustrates the DomoBus architecture and represents a network segment. The connection between the various segments is made from the backbone segment and the Routing Modules.

![DomoBus System Architecture](image)

The Control Module (CM) connects directly to physical devices and can run different applications that generate actions on devices automatically, but have memory restrictions which limits interaction with other control modules or the execution of tasks simultaneously.

In this sense, Supervision Modules (SM) are used, responsible for system management and supervision. When they receive information from the CMs, they process it and send the orders to the CMs. This approach allows each SM to be responsible for one segment of the DomoBus network and in turn can communicate with one another over an Ethernet network or wireless network.

By analyzing these and other existing solutions in the field of home automation, it is possible to perceive that there are some approaches that can take advantage of the reception of data from different equipment and relate them to have a more comprehensive analysis of the environment as a whole and thus create actions more complete and justified. In this sense, it is extremely important to analyze the architectures used in home automation since there are common problems and approaches that can help in the construction of a better solution for the security area.

There are also limitations in the transposition of these approaches to the desired solution in security, namely: the decision nucleus or supervision module are intended to take actions by the actuators based on events occurred. However, it is intended in security systems to extract the maximum information from the events and not the triggering of actions, since these should be performed by the security team; the interfaces used are directed essentially to configuration tasks and not so much for monitoring as the security area requires.

III. REQUIREMENTS

To be able to list the requirements, it is necessary to define the designation that the information will have throughout its flow through the system. To do so, the concepts of event, occurrence and security report are now defined:

**Event:** produced by the system with information provided by a sensor when it is activated, read or change state, which represents possible information to be used for the control and that may be a sign of a possible change in the safety state;

**Occurrence:** produced by the integrated security system and consists of the effective existence of an event or action that violates the rules defined in the security plan, resulting from the interpretation made by the system of the events generated;

**Security Report:** Document created automatically by the system when an occurrence is produced. It contains information that is automatically filled out by the system as well as information entered manually, resulting from the interpretation of the Security Officer after the occurrence has been resolved.

A. User Classes and Actions

System users are separated into different classes depending on the type of role they perform and the privileges that the same role requires. Table 1 summarizes the different classes and operations available to each one of them.

<table>
<thead>
<tr>
<th>User Class</th>
<th>Actions Allowed</th>
<th>Event Notifications</th>
<th>Occurrence Notifications</th>
<th>Security Report</th>
<th>Define Rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>Security Element</td>
<td>X</td>
<td></td>
<td></td>
<td>Security Report</td>
<td></td>
</tr>
<tr>
<td>Security Officer</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Day Officer</td>
<td>Just of his day</td>
<td>Just of his day</td>
<td>X</td>
<td>Classes: 5 days</td>
<td></td>
</tr>
<tr>
<td>Commander</td>
<td>X</td>
<td>X</td>
<td></td>
<td>Signs: All</td>
<td></td>
</tr>
<tr>
<td>Policy Maker</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 1. User classes and their allowed actions.
B. User Interfaces

Due to the existence of different types of users interacting with the information system, it is important to specify the different ways of interacting with the system and which users will use them.

- Monitoring: Through a Web application, the security elements must have access to a console where information about the existence of the alarms that occur and its graphical location is displayed.
- SMS (Short Message Service): The system must produce a notification whenever an occurrence takes place and it will be received on the device used by the Duty Officer in the form of a SMS.
- Reports: The Duty Officer will have access to a security report that must be completed after the existence and resolution of an occurrence. Information should be requested from the people involved, type of occurrence, location, date-time group and description of the solution adopted. The Commander must also have access to this interface to check and submit the final version of the reports.
- Configuration: The system administrator must be able to add new security rules, new sensors to the system, and change the roles and permissions of each user class.

C. Functional Requirements

During the transition of information since a signal is generated on one of the sensors until a security element is notified or the information is stored and the security report is done, there are several functionalities that must be provided by the system.

System Features:
- Communication with the sensors to receive the generated signals;
- Reception of unstructured events;
- Generation of occurrences related to physical security;
- Production of notifications to the security team;
- Generation and completion of a security report;
- Local store all events, occurrences and reports;
- Send events, occurrences and reports to a central repository.

D. Performance and Security Requirements

The only way for systems to meet their performance targets is for them to be specified clearly and unambiguously. In this sense, the following performance requirements were defined:

- To ensure prompt and adequate intervention by security elements, sensor events must be processed in useful time and ensuring no event is lost.
- The system must be developed in such a way that it is scalable without deteriorating performance.

The Security Requirements aim to maintain the 3 Rs (reliability, resiliency and recoverability), resulting in the software's operation as expected, in the non-violation of security policies, in the resistance to actions threatening agents and the possibility of recovery following a materialized threat [10]. Accordingly, the following security requirements have been defined:

- Only authenticated and authorized users can access system features and actions (LDAP + Roles);
- To provide data security, it must be encrypted;
- Creation of backups for critical data in isolation from the rest of the network;
- Secure Socket Layer (SSL) must be used for secure communication;
- There must be redundancy of all critical points of the system, both equipment (sensors, centrals and server) and communication channels.

IV. INTEGRATED SYSTEM OF PHYSICAL SECURITY

A. Logical Architecture

The architecture of the system at the organizational level is represented in Figure 6 and is composed by the sensors that operate in the Unit, the centrals to which the sensors are connected and the developed system to which the centrals communicate with. The system presents the information produced through a Web application and sends all the data produced to a central repository as is done by the system in each unit.

![Fig. 6. System Architecture at organization level.](image)

The proposed system consists of several modules, assigning a specific function to each one, facilitating the interpretation of the system and future changes or adaptation of some of the modules. The system architecture at the unit level is represented in Figure 7 and meets the framework proposed by Mitchell with three levels, being the physical, informative and cognitive.

![Fig. 7. System Architecture at Unit level.](image)

The module that communicates with the sensors is responsible for receiving the data generated by the sensors, normalizing them into events and sending them to the Data Fusion module and to the local database module where they are stored.

As soon as the normalized events arrive at the fusion module, it processes them with additional system information to identify anomalous situations and eventually generates an alarm in the form of an occurrence, which in addition to being stored locally is communicated to the Notification Module and generates the log for security report creation.

The Notification Module is responsible for transmitting the alarms to the security team and to the Duty Officer, communicating with the monitoring console module and the mobile device, respectively.

The reports module is responsible for, after being filled out and submitted, storing the security report in the local database. The Data Fusion and Communication modules are designed to allow the integration and treatment of new sensors or rules to come.

B. Data Model

The data model of the proposed solution is represented in Figure
8 in a UML form, where the most important classes through which the information flows throughout the system are represented, respecting the previously defined system requirements and architecture.

The sensor class represents the element that collects information from the physical medium, having an identifier, an associated location and other type of control information. This sensor is represented from the point of view of what the system needs to know to identify a sensor and not so much in terms of the information it emits, because that information is transmitted directly from the physical centrals to which the system is connected.

The central class, which is associated with one or more sensors (loaded from a configuration file), implements the Runnable interface and is responsible for collecting the information from the physical central. Each object of this class that is instantiated has a set of identification variables and initiates a thread that is in constant communication with the physical central receiving, in the established protocol, all the messages produced and transforming them into structured events.

The Event represents the simplest structured information object in the system. Each event is associated to the central that communicated it and contains pertinent information such as the identification of the sensor that generated it, location of the event, date-time group in which it was generated, a value associated with the physical property measured by the sensor and the descriptive type of event.

Once generated, all events are processed by occurrence-generating classes, responsible for identifying irregularities or security breaches. Each occurrence-generating instance traces a specific rule, pattern, or behavior, and produces an occurrence when it is violated.

An occurrence may result from the existence of one or more events and constitutes a very important information object since it implies a potential or effective breach of security and as such the security team must have immediate knowledge of its existence. The occurrence has a more objective description and besides the control fields that the events present, it also has information about its occurrence when it is violated. The Duty Officer must have immediate knowledge of the occurrence and the descriptive type of event.

After an occurrence is resolved by the security team, there is a need to produce a report explaining what happened. The report is associated with the occurrences reported in it and the events that gave rise to it, thus leaving the entire flow of information stored in a structured way. It is also included information reported by the Duty Officer that responds to the 5 W’s (who, what, when, where and for what). This information, once submitted, is centralized in a database to provide a better analysis of the data and to aid decision making.

C. Module of Communication with Sensors

In the communication module, occurs the initialization of the different subclasses of the “Central” class, each of which supports a different protocol, so that at any moment a new subclass can be defined for communication with sensors in a different protocol. The “Central” class defines the necessary variables and methods to be defined by its subclasses.

In the instantiation of each subclass, a configuration file containing the IP address of the central, the port in which it communicates and the information (identifier and location) of the various sensors connected to the central is read. Figure 9 represents the format of one of the configuration files loaded at startup.

![Image](image-url)

**Fig. 8. UML of implemented solution.**

The sensor class represents the element that collects information from the physical medium, having an identifier, an associated location and other type of control information. This sensor is represented from the point of view of what the system needs to know to identify a sensor and not so much in terms of the information it emits, because that information is transmitted directly from the physical centrals to which the system is connected.

**Fig. 9. Configuration file of a “Central”**.

In the example shown in the previous figure, it is defined that the system will connect to a central at address 127.0.0.1, that will communicate in port 2020 and that there are two sensors connected to it, one at the infirmary and the other in the command building, with the identifiers 111111 and 111112 respectively. Therefore, within the object of type "Fire" two objects of the type "Sensor" are created through the data provided.

To collect the data generated by the various sensors and gathered by a central, it is necessary to have an interaction between the system and the physical central, which can be done in two ways, or the data is sent without warning (push), or periodically the system will ask the data to the central (pull).

The use of one or another form of data collection depends on how the central has implemented the sending of data to external systems, but generally pull has advantages when the device or system from which the information is to be collected has many different types of data, but only a subset of this data is needed. A practical example that was used is the request of data from the fire central, the request can be made for the state to the central (existence of smoke, existence of fire, etc...), or knowing that the normal state of the central has changed, it can be asked for only the state of the sensors connected to it, since it is already known that there is a change and it is intended to know where this change occurred.

Devices or systems that push periodically have advantages when reporting a small subset of data at high rates and in real time. In an intrusion control system, periodic sending of "I'm alive" messages can be made, which means that the control unit keeps running and there are no changes to its status. In terms of implementing client-server communication, the fact that a sensor/central uses push or pull will reflect in the implementation insofar as it will play the role of client or server respectively.

The reading of the data is then done through the method “readSensor” according to the type of data collection, in case of push it is enough to analyze the sent messages, in the case of pull it is requested periodically the state of the central, being only necessary to request the state of the sensors when the state of the central is changed for some type of incident.

When a message is received with an incident, communication module is also responsible for normalizing these messages into events that become “Event” objects. Thus, from the moment the received event is normalized, it is represented by an object with the fields represented on the data model previously presented.

In the work developed, communications with fire, intrusion and
access control centrals were established, using Contact ID and ModBus protocols and a REST Web service respectively.

D. Data Fusion Module

When implementing the data fusion module, it is important to note that in a centralized architecture system, the Data Fusion Unit works as a central node that receives all the information from the different sensors and makes decisions with that information. The required processing capacity must be ensured, since the data arriving at it comes in the raw format from all sensors.

Since the number of sensors does not present an obstacle in terms of processing capacity, the data fusion module has been implemented centrally, however it is possible to view the fusion system as being distributed, since the physical centrals can process the information coming from their various sensors and add some knowledge or compact the data before they send it. This module is hosted on the server and was implemented in Java, receiving the structured events coming from the communication module (client) through the web service. Figure 10 shows the operation of the fusion module at high level.

![Fig. 10. Representation of the data fusion module.](image1)

As it is possible to verify in Figure 10, for occurrences to be generated, it is necessary that the events can be processed with some additional information to determine the violation of standards or rules implemented in the Unit. This process is done in the data fusion module where there are several “occurrence generator” classes, each of which is responsible for scanning and generate a specific occurrence. Figure 11 shows the fire occurrence class, which is the simplest to represent, since a fire event directly results in a fire occurrence.

![Fig. 11. Generation code of a fire occurrence from an event.](image2)

All generated events are sent to each of the different occurrence generating classes, which in turn filter these events and only process those that may be associated with the occurrences they produce. In the previous example, the class that generates fire occurrences is only permeable to fire events. Next, heuristics are used to determine the existence of an occurrence, through a set of rules and information from the local database and system. As soon as an occurrence is generated, it is stored in the local database and sent to the notification module, responsible for disseminating its existence.

The occurrences implemented are in line with the defined in the security plan and intended to make known when any rules/standards of conduct may be violated. For demonstration, a variety of occurrences including generation of out-of-hours access, expired access, smoke, fire, intrusion, flood and multiple access attempts were programmed. In terms of extensibility, it is easy to add new rules to existing ones by simply creating a new subclass of the abstract class “GenerationOccurrences” that implements the “processEvent” method and is permeable to the specific type of event that must be processed.

E. Notification Module

Incident notifications must reach the security team as quickly as possible to improve the response function, because with a faster communication, neutralizing the threat or resolving the incident is also faster.

To do this, in the Web application there is an occurrence tab that allows the entire security team to check the security status in real time as shown in Figure 12.

![Fig. 12. Web application: security state.](image3)

In this application, as shown in Figure 12, it is possible to verify the description of the occurrence, the time in which it occurred and the location, both descriptive and graphically signaled in the plant in real time. To complement this type of notification and considering that the Duty Officer is often outside his office, it is important to ensure that he is notified no matter the location. Thus, the notification module is also responsible for sending an SMS through a Gateway to the Duty Officer with the textual description of the occurrence so that decisions can be taken as soon as possible.

F. User Authentication

All information systems must be protected in some way against improper access attempts. Since the developed system visually displays and processes sensitive information and allows actions to be taken that influence the success of the security plan compliance, it is important to ensure that each person who has access to it is who they say they are. In this sense, there is a need for an authentication process to guarantee the security of information and actions that the system can take.

User authentication was done through LDAP. When the user accesses the application, a login page is presented for him to fill the username and password. The information is the submitted an a link is created with the LDAP server to which a bind operation with the authentication information is sent.

To configure authentication, it must be specified the properties of the environment in which it is to be made. This configuration was
done in Java through the servlet responsible for the login and using the class "javax.servlet.http.HttpServletRequest", which is represented in Figure 13:

```java
// InitialDirContext object
InitialDirContext initialContext = new InitialDirContext();

// Attributes used in the connection with the LDAP server
Attributes attributes = new Attributes();
attributes.put Wert("ContactID", "1234567890");
attributes.put Wert("ContactName", "John Doe");
attributes.put Wert("ContactAddress", "123 Main St, Anytown, USA");

// Connection to the LDAP server
String url = "ldap://example.com:389/";
String context = "ou=People, dc=example, dc=com";

// Creating a new directory context
DirectoryContext directoryContext = new DirectoryContext(url, attributes, context);
```

Fig. 13. Setting the initial authentication context.

After authentication, it is possible to verify the legitimacy of the user and which group they belong to, thus giving them permissions according to their Role. The permissions are then stored in session variables which are queried by the server to adjust the type of information that the user can access or what actions can be taken. In this way, it is also possible to change in real time the permissions of each group and consequently of each user through the LDAP server.

The LDAP server configuration was performed on a "VMware Workstation Pro" virtual machine running the "Windows Server 2016" operating system that allows the directory configuration of the folders for authentication.

G. Workflow and Report

One of the main objectives of the developed system is to automate the steps performed since a sensor collects a physical property until the security report is produced, so it is important to represent how the workflow is performed according to a set of rules that can be summarized in the diagram shown in Figure 14.

![Workflow diagram](image)

Security reports consist of the last piece of information present in the workflow and are formal written descriptions of unusual occurrences that have or may have adverse consequences. A complete and well-structured report should provide a factual description that answers the 5 Ws of who, what, when, where, and if possible the reason and/or how it was triggered, avoiding any type of additional information without importance.

The fields of the report should be completed in an objective and succinct manner, considering the following [11]:
- Who: Who were the people most involved? Who witnessed the incident?
-What: What actions and events happened? The description must be made in an objective and chronological way;
- When: Date and time of the incident;
- Where: Where did the incident occur? Include the address and/or reference buildings as close as possible to the location;
- What For: It should only be filled in if the security team has the possibility to interrogate those involved or witnesses to obtain the reason for the actions that led to the incident.

The implementation of the security report on the system was done using the visual editor iReport-3.0.0 which allows designing and creating the "JasperReports" report layout. This type of report has the advantage of being integrated into the server, where the report is called through a Java method, being passed as attributes the fields to fill in the report. The implementation of this type of report also makes possible for certain information to be automatically filled.

H. Integration with the CCSE

To improve the information flow, storage and process analysis, it is important that the entities responsible for receiving the reports can do so as soon as possible and the information must be structured, facilitating their methodological analysis. In this sense, a server has been developed that will be placed in the central repository whose function is to accept requests from the various Units to send all the information resulting from each incident, which consists of the report in PDF, a report type object, the occurrences that are attached to the report and finally the various objects of type event that gave rise to the previously mentioned occurrences.

In this way, the entity responsible for receiving, analyzing and storing the reports has more pertinent and detailed information about the incidents and enables them to execute new forms of data analysis, either statistics, allowing for example monthly reports or identification of patterns that may exist in the various occurrences they receive.

This information provides a basis for collecting performance goals and requirements, characterizing workload, creating performance testing strategies and plans, and evaluating project and system risks.

V. SYSTEM TESTS AND EVALUATION

Although system evaluation is a continuous process, it is important and of high value to conduct tests and validations on various stages of development to verify that the objectives and performance requirements are being met. To validate and evaluate the way the system was implemented, a set of evaluations and tests were performed on the system’s functionalities, security and performance. The system was also presented to entities that deal daily with development and with processes approached by this work, with the objective of perceiving their feedback through a questionnaire.

In order to conduct the tests, it was made use of Modbus PLC Simulator software to test the communication through ModBus protocol, an interface simulator was developed in Java for communication of the access control system and the ContactID protocol was implemented through a developed simulator as well.

A. Features

For the system to be a viable option to replace and dematerialize existing processes, it is necessary to meet the features previously defined. It is important to make a retrospective to perceive and validate the functionalities defined and implemented.

Event reception from sensors: This functionality has been implemented through the module of communication with the sensors and it is possible to observe the events and their information received through the Web application as shown in Figure 15.
Generation of occurrences and notification of the security team: the data fusion module is responsible for event processing and occurrence production as previously described, and the number of rules for occurrence production can be increased over time as seen. These occurrences are notified to the security team through the Web application where their real-time locations can be seen on the unit's plan, as suggested in Figure 12 and the Duty Officer is also notified by SMS.

Reception of unstructured events: this possibility has been implemented in a way that security elements can report incidents communicated to them via telephone call. To do this, a splitter is provided to fill in the incident and the security element can decide whether to create an occurrence based on the incident.

Generation and completion of a security report: on Workflow and Report was described how the Duty Officer completes a questionnaire that the system uses together with internal information to automatically generate the security report.

Regarding the Commander, the view assigned to him is shown in Figure 16, where he can verify the reports already sent by the Duty Officer, give his opinion and submit them. It is also possible to verify the information of the occurrences that have not yet been reported.

Locally store all events, occurrences and reports: this functionality has been implemented according to the defined in Data Model, ensuring that all information that is received and produced by the system is stored.

Send events, occurrences and reports to a central repository: once the security report is submitted by the Commander, the system automatically sends the generated PDF, report, occurrences and events to the central repository through a Web service provided.

B. Security

The security requirements that the system should comply with to ensure the software operation as expected have been defined earlier to prevent the violation of security policies, contributing to resist to actions of threatening agents and creating the possibility of recovery following a materialized damage [10]. These requirements are now reviewed and their implementation validated in case has it has been done, otherwise it is given an explanation on the way it could be implemented.

Security Requirement 1 (SR1): only authenticated and authorized users can have access to system features and information (LDAP + Roles) - as described in section of User Authentication, this requirement was implemented through LDAP to distinguish authorized users from unauthorized and through the definition of roles it is possible to manage in real time the actions that each one can take.

SR2: data is encrypted in the system to be secured - a security requirement not implemented, but that can be carried out using the Encrypting File System (EFS). This technology allows to encrypt the system data and control who can decrypt or recover it, since it is necessary to have an EFS certificate to perform these operations [12].

SR3: creation of backups for critical data in isolation from the rest of the network - this security requirement must be studied and implemented by the security team of the units, being defined the frequency with which data is stored and the storage location considering the level of physical security of the place chosen is a factor of extreme importance.

SR4: use of a Secure Socket Layer (SSL) for communication - this technology has been used and consists of implementing security at the transport layer, allowing Web browsers and Web servers to communicate over a secure connection. In this secure connection, the data is encrypted before being sent and then decrypted when received at the destination and before being processed. The use of this technology confers authentication and confidentiality to communication.

SR5: there should be redundancy of all critical points in the system - this safety requirement must be considered especially during the material acquisition phase of the system. Sensors that perform critical measurements must be purchased in duplicate, the server must also exist in at least two physical locations, having a generator capable of withstanding long periods of power outage. The need for redundant physical connections (communication channels) should also be studied, safeguarding that some adjustments should be made to the software due to the duplication of information that would be received by the system.

C. Performance

Since the focus of the developed system is its ability to receive information from the various sensors of a Unit and process it in possible occurrences, it is important to understand how the system behaves to perform these tasks and what are their possible limitations. A Lenovo computer with Intel (R) Core (TM) i7-5500U CPU @ 2.40GHz 64-bit processor and 16GB of RAM was used to carry out the performance tests.

Beginning with communication with sensors, the ability to communicate when many incidents occurs in a short period of time was tested. For this, 1000 events were generated from the same sensor (the communication channel will always be the same), for different time intervals between events.

Figure 17 shows a graphical representation of the results obtained. In all the tests performed, three measurements of each scenario were made and their mean value, upper and lower limits are represented with a 90% confidence level.
By the graphical interpretation of the obtained results it is possible to verify that there is a degradation of the reception capacity of the events when they have a time interval of less than 30 ms. Although the reception of events is done using TCP/IP communication, TCP retransmissions work well for short network interruptions, however they perform poorly when the network is subject to long interruption times as was caused by the sending of 1000 consecutive events. This phenomenon stems from the fact that retransmissions are made after a timer expires, which typically doubles after each unsuccessful retransmission [13].

Thus, it is possible to interpret the loss of events from intervals less than 30 ms as a result of the mechanism that prevents a flood from a network that is already overloaded with retransmissions. Since the system has dedicated sockets to establish communication with each of the sensors/centrals, it is safeguarded the occurrence of this type of event due to natural causes because the reception frequency of events is much lower than the one tested, so RS2 is fulfilled to the extent that makes it possible to scale the system.

This scenario may then occur as a consequence of a Denial of Service (DOS) attack, so a high number of events in a short period of time provides an indication of a possible attack, and an occurrence must be generated to represent this behavior.

After identifying the minimum interval between events supported by each connection, the quality of the communication was studied when subjected to a long period of communication to see if it deteriorates. In this scenario, it was tested the sending of an increasing number of events always spaced of 30 ms, and the results obtained are illustrated in Figure 18 where the percentage of received events is represented.

![Fig. 18. Received events for different number of events produced.](image)

As it is possible to verify, all the generated events were received by the system, meaning that there is no flood of the network capable of harming the sending of messages even when submitted to a large number of consecutive events if provided that the interval between them is at least 30 ms.

In order to determine the delay in the processing of events and the possibility of scaling up the implementation of more security rules, the time elapsed between the production of an event and the generation of its occurrence was studied as the number of rules applied to the system was increased. For that, only one event was produced and the rules that the system initialized were replicated, being the last one responsible of generating an occurrence resulting from the processing of the event. The result of the various scenarios tested is shown in Figure 19.

![Fig. 19. Elapsed time between event production and occurrence generation for different number of rules applied.](image)

By interpreting the above data, we can verify that there is no direct relation between the number of rules applied and the time elapsed between the production of an event and the generation of the corresponding occurrence. This is because the execution of the rules does not carry a computational load large enough that the system takes more time to execute them, and the small oscillations of elapsed times can be ignored. These results allow us to conclude that the performance requirements were reached, both the RD1 because the events are processed in a useful time (~ 500-600 ms), as the RD2, because it is verified that there is the possibility to scale the system, it is possible to implement new security rules without reducing system performance. It should be safeguarded that by increasing the complexity of the rules or introducing computationally heavy techniques, such as the use of machine learning, will naturally increase the processing time of events.

D. Inquiries

Constant contact was established with the entities responsible for the application development and for ensuring in the Army some of the processes that this work intended to improve or dematerialize. In this sense, it was constant the exchange of information for a correct definition of the requirements, for exchanges of experience in the implementation and finally in the evaluation. The developed system was presented to the Directorate of Communication and Information Systems (DCSI), the unit responsible for collecting and handling security reports (CCSE) and the Military Academy.

After explaining the scope of the work developed and presenting the system in operation, a questionnaire was made to several elements with the following functions:

- Head of the Department of Application Development (1);
- Head of the CCSE Security Section (1);
- Head of the Information Section of the CCSE (1);
- Officials who regularly serve as Duty Officer (2);
- Soldiers who regularly perform the function of a security element (2).

The questions posed were open-ended since they were not directed to specific system functionalities because the requirements were defined in conjunction with the entities with competence to do so, but rather to understand how the various parties described the impact that the system could have in the management of information and in the perception of security implemented in the Units.

The feedback received was very positive, and there was unanimous assertion that the developed system could promote an improvement in the management of information coming from sensors and consequently enhance the responsiveness of security teams. It was also strongly mentioned the possible change in the perception of the security paradigm, since there is a "need to complement physical security with information systems for their analysis and correlation of events of the various sensors" and that, in turn, security "should always contain in their systems the so-called redundancy, so this system, integrated with existing systems would facilitate and expedite processes in a timely manner."

Military personnel who are often the security team, either through the Duty Officer function or through the security element function, have indicated that this system would enable them to be notified more efficiently and in a timely manner, thus improving the response they can give as well as the subsequent management of all information that is facilitated by the high automation provided.

In short, the presentations and questionnaires were very useful for validating the importance and potential of the work developed, thus mirroring the way the requirements and the architecture were achieved and the perception that the entities involved in the processes approached are willing to use the system in their daily life, encouraging the continuation of its development and suggesting the possibility of integrating other systems within it.
VI. CONCLUSIONS

The redundancy and diversity of sensor types in the Military Units is essential for the existence of an efficient Physical Security monitoring system, however this requires the existence of several sensors and consequently a large number of alarms generated that need be managed. However, this is a subject that is underdeveloped to make it possible to integrate several security systems considering the particularities of the institution in question.

Problems that need a different approach to be solved were then identified: some security systems are isolated and it is not possible to efficiently collect and treat their data; the notifications produced are rather incomplete and do not provide relevant information to the security team to enhance their response; there is no mechanism capable of unifying the alarm signals; the administration and control procedures are done manually and costly.

This led to the need to study the particularities of the Military environment and a set of subjects that support the understanding of the sensors, the various physical security systems and the models developed to solve similar problems, thus allowing the acquisition of the necessary knowledge for the development of a solution capable of: collecting the events generated by the systems and equipment existing in the Units; process the collected events and generate alarms by identifying anomalous situations; efficiently notify the Unit Duty Officer and his team of the location and type of occurrence; automate the production of Security reports and centralize the history of events, occurrences and reports in the CCSE.

To develop this solution, we started by defining the requirements according to the existing needs, from which the architecture and the data model to be used were created. The system was then implemented in accordance with the established and with great extensibility, reason why it is possible to add new rules and sensors to the system easily.

After its implementation, tests and evaluations were defined and executed: to the functionalities, from which it was verified that all were successfully implemented; to security, where it was found that the security requirements that were possible to implement up to the system installation phase were met; to the performance, where it was possible to find the maximum frequency of events reception by each process that communicates with a central as well as to infer the high potential of scalability of the system in terms of rules applied to it.

The evaluation phase was further completed by presenting the system to Army officials working in areas where it may have direct or indirect influence. From this presentation and the resulting questionnaires, it was possible to confirm the high interest in the solution developed by allowing descriptive and predictive structured analysis of the collected data, resulting in the evaluation of the adjudication of the means to mitigate the threats. Also, the information management was pointed out as a great value conferred by the system for allowing the registration, data analysis and implementation of automated decisions based on the specific needs inherent to the Military environment. In addition to the advantages conferred in terms of processes, it was also identified by the head of the information section of the CCSE the potential of the work developed to change the perception of the institution to the security paradigm, mitigating the divergences that come from the individual experience of each decision maker with responsibilities in security.

As future work, it is proposed to use foundations created to integrate other technologies and systems. The determination of occurrences through events is done through rules, but the feasibility of using machine learning techniques to generate these same occurrences should be studied. Integration can be made with Army projects at the level of video surveillance, giving another vector to be integrated into the system.