



Systems Engineering Architecture Framework for Physical Protection Systems

ISSE 2020 - 6th IEEE International Symposium on Systems Engineering, Proceedings

Tekinerdogan, Bedir; Ozcan, Kaan; Yagiz, Sevil; Yakin, Iskender

<https://doi.org/10.1109/ISSE49799.2020.9272215>

This publication is made publicly available in the institutional repository of Wageningen University and Research, under the terms of article 25fa of the Dutch Copyright Act, also known as the Amendment Taverne. This has been done with explicit consent by the author.

Article 25fa states that the author of a short scientific work funded either wholly or partially by Dutch public funds is entitled to make that work publicly available for no consideration following a reasonable period of time after the work was first published, provided that clear reference is made to the source of the first publication of the work.

This publication is distributed under The Association of Universities in the Netherlands (VSNU) 'Article 25fa implementation' project. In this project research outputs of researchers employed by Dutch Universities that comply with the legal requirements of Article 25fa of the Dutch Copyright Act are distributed online and free of cost or other barriers in institutional repositories. Research outputs are distributed six months after their first online publication in the original published version and with proper attribution to the source of the original publication.

You are permitted to download and use the publication for personal purposes. All rights remain with the author(s) and / or copyright owner(s) of this work. Any use of the publication or parts of it other than authorised under article 25fa of the Dutch Copyright act is prohibited. Wageningen University & Research and the author(s) of this publication shall not be held responsible or liable for any damages resulting from your (re)use of this publication.

For questions regarding the public availability of this publication please contact openscience.library@wur.nl

Systems Engineering Architecture Framework for Physical Protection Systems

Bedir Tekinerdoğan
Information Technology
Wageningen University
Wageningen, The Netherlands
bedir.tekinerdogan@wur.nl

Kaan Özcan
ASELSAN
Ankara, Turkey
mkozcan@aselsan.com.tr

Sevil Yağız
ASELSAN
Ankara, Turkey
syagiz@aselsan.com.tr

İskender Yakın
ASELSAN
Ankara, Turkey
iyakin@aselsan.com.tr

Abstract— A physical protection system (PPS) integrates people, procedures, and equipment for the protection of assets or facilities against theft, sabotage, or other malevolent intruder attacks. In this paper we focus on the architecture modeling of PPS to support the communication among stakeholders, analysis and guiding the systems development activities. A common practice for modeling architecture is by using an architecture framework that defines a coherent set of viewpoints. Existing systems engineering modeling approaches appear to be too general and fail to address the domain-specific aspects of PPSs. On the other hand, no dedicated architecture framework approach has been provided yet to address the specific concerns of PPS. In this paper, we present an architecture framework for PPS (PPSAF) that has been developed in a real industrial context focusing on the development of multiple PPSs. The architecture framework consists of six coherent set of viewpoints including facility viewpoint, threats and vulnerabilities viewpoint, deterrence viewpoint, detection viewpoint, delay viewpoint, and response viewpoint. We illustrate the application of the architecture framework for the design of a PPS architecture of a building.

Keywords— *Physical Protection Systems, Systems Engineering, Architecture Framework*

I. INTRODUCTION

A physical protection system (PPS) integrates people, procedures, and equipment for the protection of assets or facilities against theft, sabotage, or other malevolent intruder attacks [4][5][8]. A PPS provides deterrence and a combination of detection, delay and response measures to protect against an adversary's attempt to complete a malicious act. A PPS includes physical protection devices such as interior and exterior intrusion detection sensors, cameras, barriers, access control devices and response measures.

Developing a PPS typically requires a systems engineering approach that adopts an interdisciplinary focus to design, integrate, and manage complex systems over their life cycles [11]. The systems engineering approach has evolved from a document-centric approach to a model-based approach. Model-based systems engineering (MBSE) focuses on creating and exploiting domain models as the primary means of information exchange between engineers, rather than using informal documents. Transitioning to a model-based systems engineering approach enables engineering teams to enhance the understanding of the

design, enhance the communication of design decisions, and better analysis of the system before it is built.

An essential part of systems engineering in general and model-based systems engineering, in particular, form the systems architecture [6]. Systems architecture defines the gross-level organization of a system, including its major components, the relationships between them, and the way how they collaborate to meet system requirements, and principles guiding their design and evolution. The architecture design, together with the rationale of the design decision, is described in the architecture documentation that can be used as a guideline for the corresponding implementation. A well-documented architecture is crucial for supporting the communication among stakeholders, for guiding and analysis of the design decisions, and for guiding the organizational processes. A common practice for describing the architecture according to the stakeholders' concerns is to model different *architecture views* [2][10]. An architecture view is a representation of a set of system elements and relations associated with them to support a particular concern. Usually multiple architectural views are needed to focus on and separate the various stakeholder concerns. Each architecture view is a representation (model) that conforms to a well-defined viewpoint in the corresponding architecture framework. The architecture viewpoint represents the conventions for constructing and using the corresponding views.

For developing a proper systems architecture of PPS we could adopt a systems engineering approach. Yet, existing systems engineering approaches are agnostic to a specific domain and as such too general, and thus fail to address the domain-specific aspects of PPSs. On the other hand, so far, no dedicated architecture framework has been developed to model the architecture concerns of PPS.

Within the context of model-based systems engineering the overall objective of this study is therefore to derive an architecture framework that can be used to model the architecture of PPSs. The research has been carried out within the context of ASELSAN, a large scale systems engineering company. The architecture framework consists of six coherent set of viewpoints including facility viewpoint, threats and vulnerabilities viewpoint, deterrence viewpoint, detection viewpoint, delay viewpoint, and response viewpoint. We illustrate the application of the architecture framework for the design of a PPS architecture of a building.

The remainder of the paper is organized as follows. In section 2 we present the background on PPS and model-

based systems engineering. Section 3 presents process for selecting an architecture framework. Section 4 presents the metamodel of PPS that includes the key concepts. Section 5 presents PPSAF and describes the six viewpoints in detail. Finally, section 6 concludes the paper.

II. PRELIMINARIES

A. Physical Protection Systems

Due to the interdisciplinary and crosscutting concerns [1] developing a PPS typically requires a systems engineering approach. The traditional systems engineering lifecycle process is often presented as a V-model [9]. The left side of the V represents concept development and the decomposition of requirements into function and physical entities that can be architected, designed, and developed. The right side of the V represents integration of these entities (including appropriate testing to verify that they satisfy the requirements) and their ultimate transition into the field, where they are operated and maintained.

Obviously, the systems engineering lifecycle process is agnostic to specific domains and as such is generic and less useful for modeling the specific concerns of a particular domain. In the case of a PPS this requires the domain specific steps to focus on the detection, delay and response measures to protect a system against an adversary's attempt to complete a malicious act. Fig. 1. shows the top-level activities for the PPS design process [14]. In essence the PPS process consists of three key activities that include identifying the PPS objectives, designing the PPS, and evaluating the PPS. Determining the PPS objectives includes the facility characterization, the threat definition and the definition of the target that needs to be protected. Designing PPS focuses on three activities, detection, delay and response. The resulting PPS design should meet the defined objectives and operational, safety, legal, and economic constraints of the facility. The final step in the PPS lifecycle is the evaluation of the design PPS. Several techniques can be distinguished here, including Path Analysis, Scenario Analysis, and System Effectiveness Analysis [4][5].

The outcome of this process is a system vulnerability assessment. The analysis of the PPS design can lead to either to the conclusion that the design is feasible and effectively achieves the protection objectives, or it will still identify unnoticed weaknesses.

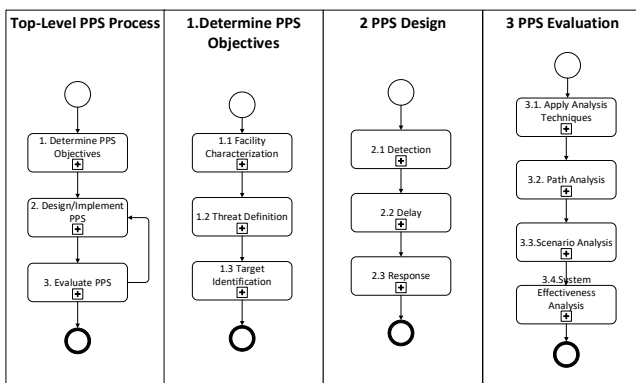


Fig. 1. PPS Design Process

B. Model-Based Systems Architecting

The MBSE approach was popularized by INCOSE when it kicked off its MBSE Initiative in January 2007. Goals included increased productivity, by minimizing unnecessary manual transcription of concepts when coordinating the work of large teams. MBSE is defined as follows: "Model-based systems engineering (MBSE) is the formalized application of modeling to support system requirements, design, analysis, verification and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycle phases." [9]

An important aspect of model-based systems engineering is model-based systems architecting. Architecture design is basically about *modeling* the system from different perspectives. Historically, *models* have had a long tradition in software engineering and have been widely used in software projects. The primary reason for modeling is usually defined as a means for communication, analysis or guiding the production process. The concepts related to architectural description are formalized and standardized in ISO/IEC 42010:2007, a fast-track adoption by ISO of IEEE-Std 1471-2000, *Recommended Practice for Architecture Description of Software-Intensive Systems* [10]. The standard holds that an architecture description consists of a set of *views*, each of which conforms to a *viewpoint*, but it has deliberately chosen not to define a particular viewpoint. An *Architecture Framework* is defined as the coordinated set of viewpoints that are used to define the views. A more precise definition of architecture framework is given in the ISO standard: "*Conventions, principles and practices for the description of architectures established within a specific domain of application and/or community of stakeholders*"

III. SELECTING ARCHITECTURE FRAMEWORK

Architecture frameworks can be reused, extended or designed from scratch. The decision process for this is shown in Fig. 2.

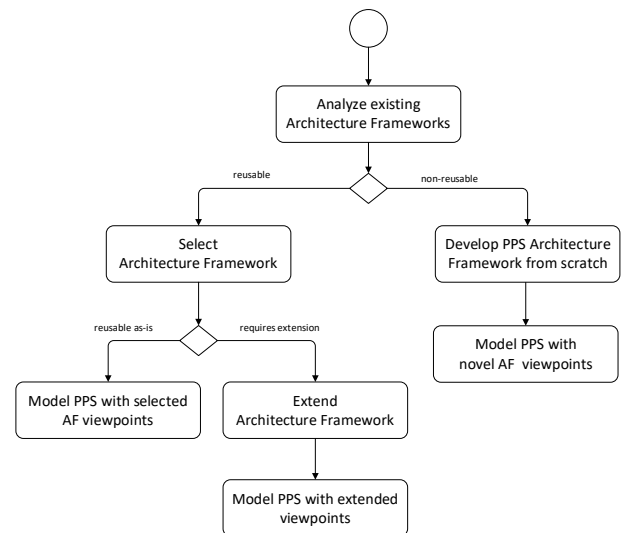


Fig. 2. Decision process for architecture framework for PPS

After an analysis of architecture frameworks in the literature a decision is made whether to reuse an existing architecture framework or develop an architecture framework from scratch. In case of the selection of an

architecture framework a decision needs to be made whether the viewpoints in the selected architecture framework are sufficient to model the domain specific PPS concerns. If this is the case, then the viewpoints of the selected architecture framework are reused and the PPS is modeled accordingly. If the selected architecture framework are not totally fit for purpose, that is, an extension is required, then novel viewpoints will be added to the architecture framework.

A PPS is highly software controlled and as such has also its own software architecture. Hence, to guide our decision making for the selection of architecture frameworks we have distinguished modeling the software architecture and systems architecture. This is shown in the conceptual model of Fig. 3. As such, the decision was split into which architecture framework to use for the systems architecture, and which for the software architecture.

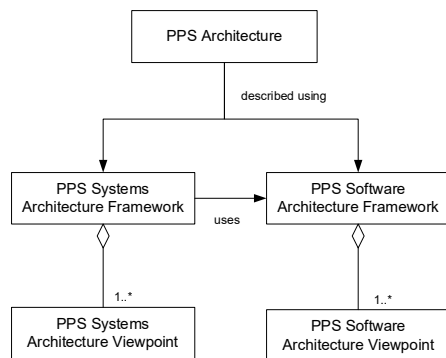


Fig. 3. The relation of architecture frameworks for PPS

A. Selection of Software Architecture Framework

For the software architecture we have selected the so-called Views and Beyond (V&B) approach [2]. The V&B approach defines the following view categories: *Module view* category that is used for documenting a system's principal units of implementation. *Component and Connector* category that is used for documenting the system's units of execution. *Deployment View* category that is used to document the relationships between a system's software and its development and execution environments. Viewpoints are defined as styles which are used to define views. The V&B approach appeared to be sufficient for modeling the software architecture. This is not the scope of this paper.

B. Selection of Systems Architecture Framework

Regarding the systems architecture framework we have been guided by the results of our earlier study [7] whereby we have provided the results of a domain analysis process to identify the current enterprise, system and software architecture frameworks. With respect to systems engineering architecture frameworks we concluded that the notion of viewpoint is less explicit here. Despite the few studies, overall we observed that the notion of architecture framework is still active research, while the proposed architecture frameworks are primarily domain specific and less applicable for architecting PPS. Hence for systems engineering we have decided to develop a PPS systems architecture framework from scratch (first option). We will elaborate on this in the following sections.

C. Approach for Developing Viewpoints

To develop viewpoints we will use the template as shown in Table 1. The template is based on the recommended standard for architecture description as it is defined in [10].

Table 1. Part of the Template for documenting viewpoints as defined by ISO/IEC 42010 [10]

Section	Description
Viewpoint Name	The name for the viewpoint, and any synonyms for the viewpoint
Overview	An abstract or brief overview of the viewpoint and its key features.
Concerns	A listing of the architecture related concerns framed by this viewpoint.
Anti-Concerns	It can be useful to document the kinds of issues a viewpoint is not appropriate for.
Typical Stakeholders	The typical audiences for views prepared using this viewpoint.
Architecture Elements	The key architecture components that are described in the architecture.
Architecture Relations	The key architecture relation types among the architecture elements.
Constraints	The configuration rules that impose constraints on the composition of the components
Notation	The adopted notation to be used for the model of this viewpoint
Operations on views	Operations define the methods which may be applied to views and their models. Operations can be divided into categories: <i>Creation methods</i> are the means by which views are prepared using the viewpoint. <i>Analysis methods</i> are used to check, reason about, transform, predict, apply and evaluate architectural results from this view.

IV. METAMODEL PHYSICAL PROTECTION SYSTEM

Every architecture framework defines a coherent set of viewpoints. The architecture framework as such has a well-defined conceptual basis that is typically reflected using a metamodel. Before developing the PPS architecture framework we have first developed the underlying metamodel, which is shown in Fig. 4.

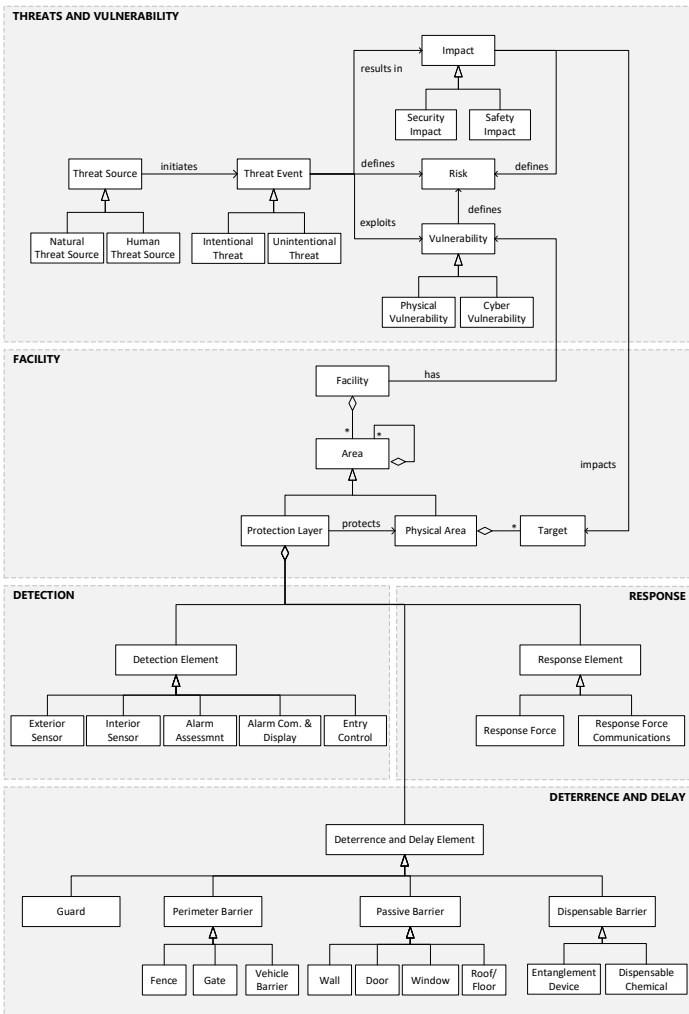


Fig. 4. PPS Meta-Model

Based on the provided PPS metamodel we define the architecture framework which consists of six viewpoints each addressing a specific concern of the PPS. The architecture framework is illustrated in Fig. 5. The details of the metamodel and the architecture framework with the six viewpoints will be explained in the next section.

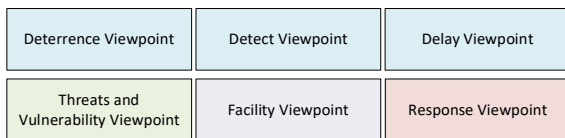


Fig. 5. Physical Protection System Architecture Framework (PPSAF)

V. ARCHITECTURE VIEWPOINTS

A. Facility Viewpoint

The facility viewpoint is used to design the structure of the facility that needs protection. Each facility includes the facility itself, the physical area in which it resides, the protection layers and the target in the facility that is the target of threats. Modeling the facility view is typically the first architecture design activity for the PPS. The modelled view can be used to support the facility analysis which is an important step of the PPS life cycle process. The Facility

Viewpoint is shown in Table 2. The adopted notation is using selected elements from MS Visio. This is also the case for the subsequent viewpoints.

Table 2. Facility Viewpoint

Section	Description
Viewpoint Name	Facility Viewpoint
Overview	Characterization of facility operations and conditions through description of the facility (the location of the site boundary, building location, building interior floor plans, access points).
Concerns	Model the facility that must be protected.
Anti-Concerns	This viewpoint only considers the facility and does not yet focus on threats and targets.
Typical Stakeholders	System engineers, architects, facility stakeholders
Architecture Element	Facility; Protection Layer; Physical Area; Target
Relations	Protects; Part of
Constraints	Physical area is surrounded by multiple protection layers. Target is located in Physical Area
Notation	<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 5px; margin: 2px;"> <<Facility>> </div> <div style="border: 1px solid black; padding: 5px; margin: 2px;"> <<Protection Layer>> </div> <div style="border: 1px solid black; padding: 5px; margin: 2px;"> <<Target>> </div> </div> <p>For other elements use stereotypes for UML class notation or well-defined visual elements.</p> <p>Elements are open-ended depending on the application domain. Below are example notations:</p> <div style="display: flex; flex-wrap: wrap; justify-content: space-between;"> <div style="width: 30%;"> <p>Room</p> <p>Window</p> <p>"L" Room</p> <p>Opening</p> <p>Pilaster</p> <p>Controller dimension</p> </div> <div style="width: 30%;"> <p>Wall</p> <p>"T" Room</p> <p>Double door</p> <p>Corner pilaster</p> <p>Room measurements</p> </div> <div style="width: 30%;"> <p>Door</p> <p>Curved wall</p> <p>Space</p> <p>Callout</p> </div> </div>
Operations on Views	<p>Creation Methods</p> <ul style="list-style-type: none"> - Identify the target that needs to be protected. Decide on the number and arrangement of protection layers and ensure that target is sufficiently protected <p>Analysis Methods</p> <ul style="list-style-type: none"> - The view can be used to analyse the threats and vulnerabilities, and herewith the risks of the facility that needs to be protected.

B. Threats and Vulnerabilities Viewpoint

The Threats and Vulnerabilities Viewpoint is shown in Table 3. This viewpoint is used to model the threats, and the vulnerabilities of the overall facility or the identified targets herein. The viewpoint is based on the top level part of the metamodel as shown in Fig. 4.

Table 3. Threats and Vulnerabilities Viewpoint

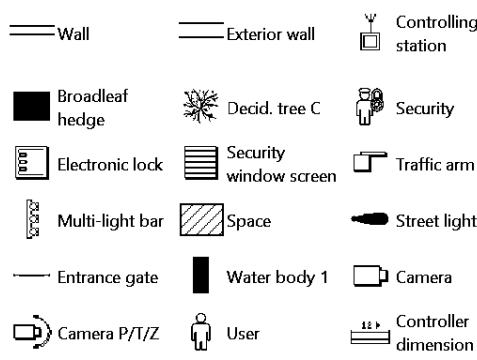
Section	Description															
Viewpoint Name	Threats and Vulnerabilities Viewpoint															
Overview	Describes the threats and the target.															
Concerns	Identify the internal and external threats Identify the vulnerabilities of the facility Identify the targets															
Anti-Concerns	This viewpoint only considers threats and targets, and does not focus on the design of the facility															
Typical Stakeholders	System engineers, architects, facility stakeholders, risk management professional, security experts															
Architecture Element	Threat Source; Threat Event; Target; Vulnerability															
Relations	Initiates (threat source initiates threat event) Exploits (threat event exploits vulnerability)															
Constraints	Threat events can exploit one or more vulnerabilities. Risk is defined as															
Notation	Using table format we with following column names: <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Threat Source</th> <th>Threat Event</th> <th>Likelihood</th> <th>Vulnerability</th> <th>Risk</th> </tr> </thead> <tbody> <tr> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> </tr> <tr> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> </tr> </tbody> </table>	Threat Source	Threat Event	Likelihood	Vulnerability	Risk										
Threat Source	Threat Event	Likelihood	Vulnerability	Risk												
Operations on Views	<p><i>Creation Methods</i></p> <ul style="list-style-type: none"> - Identify the threat sources and the possible adversarial actions of that source (threat event). - Calculate/define the likelihood - Identify the vulnerabilities of the facility that relate to the threat events - Calculate the risk <p><i>Analysis Methods</i></p> <ul style="list-style-type: none"> - The view can be used to systematically analyze and identify the threats and vulnerabilities. Based on the risks the view can guide also the required measures for detection, delay, and response actions. 															

The viewpoint shows the assets or the targets that need to be protected. A threat event is initiated by a threat source. The threat source can be natural or human-based. Natural threat source include threats, such as storms, floods, hurricanes, or tornadoes. A threat event is defined as anything that can exploit a vulnerability, intentionally or unintentionally, and as such lead to a security or safety impact. An example of an unintentional threat is an employee mistakenly accessing the wrong information. Intentional threats typically include planned adversary actions such as theft of or damage to a physical asset or digital asset. Vulnerabilities refer to weaknesses in a system and thus make a threat possible and potentially even more dangerous. In essence, the vulnerability can be considered as an internal factor of a system or a subject that is exposed to a threat. A risk is defined as the potential for loss, damage or destruction of an asset as a result of a threat exploiting a vulnerability. In general, reducing vulnerability will reduce risk and likewise increase resilience which in turn may reduce the impact of a threat. Note that for this viewpoint we primarily use a table representation to depict the threats and vulnerabilities.

C. Deterrence Viewpoint

To show the design that focuses on deterrence the Deterrence Viewpoint can be used, which is shown in Table 4. Here we focus on the elements that are integrated with the facility and do not consider legislative issues. The concepts for this viewpoint are shown in the lower part of the metamodel as shown in Fig. 4. Note that this part also refers to the delay elements. The reason for this is that deterrence and delay elements may look similar, but we distinguish these based on their role of usage.

Table 4. Deterrence Viewpoint

Section	Description
Viewpoint Name	Deterrence Viewpoint
Overview	Describes the architecture from the deterrence concern
Concerns	Show the deterrence elements in relation to the protected facility
Anti-Concerns	This viewpoint only considers deterrence concern and does not focus on detection, delay or response. If needed it can be combined with these.
Typical Stakeholders	System engineers, architects, facility stakeholders
Architecture Element	Deterrence Element; Facility; Protection Layer; Physical Area; Target
Relations	Deters; Part-Of
Constraints	-
Notation	Using stereotypes UML class notation for particular deterrence elements or visual elements. Elements are open-ended depending on the application domain. Example notations: 
Operations on Views	<p><i>Creation Methods</i></p> <ul style="list-style-type: none"> - Based on the facility structure and the target identify the means for deterrence <p><i>Analysis Methods</i></p> <p>The view can be used to analyse whether sufficient deterrence measures have been taken, or whether the existing deterrence approaches will meet the needs.</p>

The best way to protect a system is perhaps to avoid that the threat is ever triggered. Deterrence is the action of discouraging an action or event through instilling doubt or fear of the consequences. Deterrence occurs by adopting measures that are perceived by potential adversaries as too difficult to defeat [4]. An important aspect of deterrence is

legislation which performs penalties for adversarial actions such as unauthorized removal and sabotage. Other examples of measures that may enhance deterrence include presence of security guards, adequate lighting at night, and use of barriers. In general it is reported that the deterrence function of a PPS is difficult to measure, and reliance on deterrence only will be risky. In [5] it is not considered as explicit step of the comprehensive PPS design process, and only considered a secondary function.

D. Detection Viewpoint

Detection is the discovery of an adversary action and includes sensing of covert or overt actions. Table 5 shows the Detection Viewpoint. Detection starts with sensing a potentially adversarial act and is completed only when the cause of the alarm has been assessed. The assessment can decide whether the alarm is a valid alarm or a nuisance alarm. If an alarm is assessed as valid, a response will be initiated. Detection without timely and accurate assessment is meaningless [5]. The effectiveness of the detection function is defined as the probability of sensing adversary action and the time required for reporting and assessing the alarm.

An important element of PPS detection is entry control which refers to detecting the attempted entry of unauthorized personnel and material. Two metrics are identified here including false acceptance rate and false rejection rate. False acceptance rate defines the rate at which false identities or credentials are allowed entry. False rejection rate defines the frequency of denying access to authorized personnel. Detection can also be accomplished by human guards at fixed posts or response force.

Table 5. Detection Viewpoint

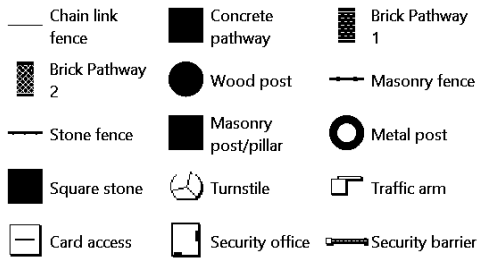
Section	Description
Viewpoint Name	Detection Viewpoint
Overview	Describes the architecture from the detection concern
Concerns	Show the detection elements in relation to the protected facility
Anti-Concerns	This viewpoint only considers detection concern and does not focus on the detailed description of threats
Typical Stakeholders	System engineers, architects, intrusion detection experts, facility stakeholders
Architecture Element	Detection Element (General) Interior Sensor; Exterior Sensor; Alarm Assessment; Alarm Communication; Entry Control
Relations	Detects
Constraints	-
Notation	Using stereotypes UML class notation for particular deterrence elements or visual elements or selected MS Visio Elements

	<p>Video Surveillance</p> <p>Alarm and Access Control</p>
Operations on Views	<p><i>Creation Methods</i></p> <ul style="list-style-type: none"> - Identify and locate the required interior sensors; - Identify and locate the required exterior sensor; - Identify and locate alarm Assessment - Define alarm communication; - Decide on elements and locations of entry control <p><i>Analysis Methods</i></p> <ul style="list-style-type: none"> - The view can be used to systematically analyze and identify whether the existing detection mechanisms are sufficient and effective.

E. Delay Viewpoint

The Delay Viewpoint is shown in Table 6. Delay is the slowing down of adversary progress. Delay can be accomplished by people, barriers, locks, and activated delays. Similar to the detection, response force can also be considered elements of delay. The primary measure for the performance of delay is the delay effectiveness which defines the time required by the adversary (after detection) to bypass each delay element. As stated before, delay before detection is a deterrence action, and not delay. Delay elements include passive barriers, dispensable barriers, and guards. Passive barriers include structural elements such as doors, walls, floors, locks, vents, ducts, and fences. Dispensable barriers are those that are deployed only when necessary during an attack. Guards can provide delay to adversaries using additional tactics to gain entry. Each type of barrier has advantages, and as such, to provide an effective PPS typically all three types of delay elements are used.

Table 6. Delay Viewpoint

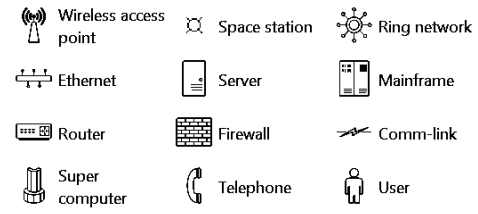
Section	Description
Viewpoint Name	Delay Viewpoint
Overview	Describes the architecture focusing on delay elements that are needed to delay detected adversary actions
Concerns	Show the delay elements in relation to the protected facility
Anti-Concerns	This viewpoint only considers delay concern and does not focus on detection or response. If needed a combined view can be provided
Typical Stakeholders	System engineers, architects, delay element experts, facility stakeholders
Architecture Element	Passive barrier Perimeter barrier Dispensable barrier Guards
Relations	delays
Constraints	-
Notation	Using stereotypes UML class notation for particular deterrence elements or visual elements 
Operations on Views	<p><i>Creation Methods</i></p> <ul style="list-style-type: none"> - Identify and locate the passive and active barriers; - Identify the number and location of guards. <p><i>Analysis Methods</i></p> <ul style="list-style-type: none"> - The view can be used to systematically analyze and identify whether the existing delay mechanisms are sufficient and effective.

F. Response Viewpoint

The final viewpoint of the PPSAF is the Response Viewpoint which is shown in Table 7. Response function includes the responding personnel and the response communication systems. Response force defines any personnel who may be involved in the response at a particular facility to respond on the adversarial action. Response forces can be located on-site or off-site, and include proprietary or contract guards, local and state police, armed forces, medical personnel, fire personnel, safety personnel, alarm notification, or unmanned aerial vehicles [4]. The action of response can be either immediate, timely response and/or after-the fact recovery.

Depending on the needs and objectives of a facility, the response is planned beforehand based on the needs and conditions. Different facilities and targets will require different response plans.

Table 7. Response Viewpoint

Section	Description
Viewpoint Name	Response Viewpoint
Overview	Describes the architecture focusing on response elements that are needed to provide a response to adversary actions
Concerns	Show the response elements in relation to the identified adversary actions
Anti-Concerns	This viewpoint only considers response concern and does not focus on deterrence or detection. If needed a combined view can be provided
Typical Stakeholders	System engineers, architects, communication systems experts, response force, facility stakeholders
Architecture Element	Response force Response force communications
Relations	Allocate response force Communicate Respond
Constraints	-
Notation	Using stereotypes UML class notation for particular deterrence elements or visual elements or selected MS Visio Elements. 
Operations on Views	<p><i>Creation Methods</i></p> <ul style="list-style-type: none"> - Identify and locate response force; - Identify and define response communications <p><i>Analysis Methods</i></p> <p>The view can be used to systematically analyze and identify whether the existing response mechanisms are sufficient and effective.</p>

G. Guidelines for Adopting PPSAF Views

In the previous sub-sections we have described the PPSAF Viewpoints. These viewpoints together form a coherent set of viewpoints, as it is also directed by the term of architecture framework. For designing the views we have provided the process as it is shown in Fig. 6. As shown in the figure the first step is the design of the facility view which defines the overall arrangement of the facility together with the protection layers. This is followed by the threats and vulnerabilities view which identifies the possible internal and external threat sources, the threat events and the vulnerabilities of the facility itself. The next step is design of the deterrence and detection views which can be done in parallel. This is followed by the design of the delay view, and response view. After this, all designed views are checked for consistency and if necessary aligned for internal consistency and the realization of the protection needs [12]. To respect the separation of concerns principle and herewith the better understanding and focus of a single concern, it is preferable to provide indeed a view for each concern. However, as it is the cases also for the other architecture frameworks in the literature, design of hybrid views may be required as well. Fig. 7 shows an illustrative example of a PPS hybrid view in which we show the facility (building) together with the deterrence, and detection elements.

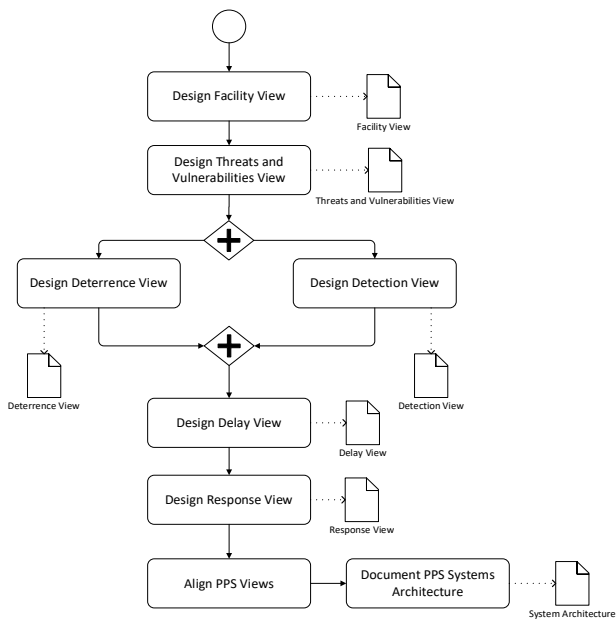


Fig. 6. Workflow for modeling the PPS Views.

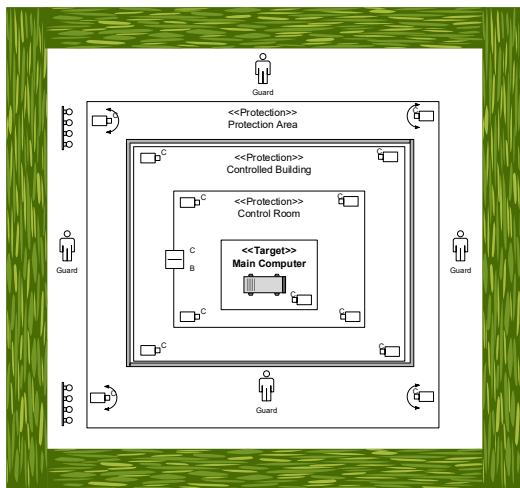


Fig. 7. PPS Hybrid View with elements of deterrence, detection, and delay.

VI. CONCLUSION

Physical protection systems (PPS) have been broadly discussed and applied to ensure the protection of various types of systems. Developing a PPS is in essence a systems engineering approach but existing systems engineering processes are limited to address the domain-specific aspects of PPSs. As such dedicated processes have been proposed in the literature to develop PPSs. An important artefact of the design process is the systems architecture which represents the fundamental structure of the PPS. To design a proper PPS it is necessary to adopt a well-defined architecture framework with the corresponding viewpoints addressing the required concerns. Unfortunately, neither in systems engineering nor in the dedicated PPS life cycle processes, a suitable architecture framework has been proposed yet to model PPS architectures. Hence, in alignment with the vision of model-based systems engineering, we have provided an architecture framework (PPSAF) for designing physical protection system architectures. To develop the PPSAF we have performed a

thorough domain analysis to PPS and provided a metamodel that defines the PPS key concepts. Subsequently, based on the metamodel we have derived a coherent set of six architecture viewpoints including facility viewpoint, threats and vulnerabilities viewpoint, deterrence viewpoint, detection viewpoint, delay viewpoint and response viewpoint. Each viewpoint can be used to model the systems architecture from the perspective of a particular concern that is held by the corresponding stakeholders. The fundamental concepts (abstract syntax) of the viewpoints are solid. However, due to the broad domain of PPSs the provided notation (concrete syntax) appeared to be open-ended. Further, PPS can be applied beyond facilities and include, for example, coast protection, border protection, city protection. This would require the extension of the framework for different infrastructures. Overall, we believe that PPSAF provides a valuable contribution to the PPS domain, and complements the research on systems architecture from the general systems engineering perspective. PPSAF is already applied in a real industrial context to document the PPS architectures. In our future work we will further apply these viewpoints for architecting various PPSs.

REFERENCES

- [1] J. Bakker, B. Tekinerdogan, M. Aksit. Characterization of early aspects approaches. In Proceedings of the Early Aspects Workshop at AOSD, The Netherlands, 2005.
- [2] P. Clements, F. Bachmann, L. Bass, D. Garlan, J. Ivers, R. Little, P. Merson, R. Nord, J. Stafford. Documenting Software Architectures: Views and Beyond. Second Edition. Addison-Wesley, 2010.
- [3] E. Demirli, B. Tekinerdogan. Software language engineering of architectural viewpoints. in Proc. of European Conference on Software Architecture, p. 336-343, Springer, 2011.
- [4] ML. Garcia. Vulnerability assessment of physical protection systems. Amsterdam: Elsevier Butterworth-Heinemann; 2006.
- [5] ML. Garcia. The design and evaluation of physical protection systems. 2nd ed. Amsterdam: Elsevier Butterworth-Heinemann; 2008.
- [6] Guide to the Systems Engineering Body of Knowledge (SEBoK), October 2016.
- [7] H. G. Gurbuz and B. Tekinerdogan. Analyzing Systems Engineering Concerns in Architecture Frameworks – A Survey Study, IEEE International Systems Engineering Symposium (ISSE), Rome, 2018, pp. 1-8, doi: 10.1109/SysEng.2018.8544385.
- [8] IAEA, Handbook on the Physical Protection of Nuclear Material and Facilities, IAEA-TECDOC-127, March 2000.
- [9] INCOSE Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities, 4th Ed.. John Wiley & Sons, 2015.
- [10] ISO/IEC 42010:2007 Recommended practice for architectural description of software-intensive systems (ISO/IEC 42010), 2011.
- [11] System Engineering Handbook, A Guide For System Life Cycle Process and Activities, 2015
- [12] B. Tekinerdogan, C. Hofmann, & M. Aksit. Modeling Traceability of Concerns for Synchronizing Architectural Views. Journal of object technology, 6(LNCS4549/7), 7-25, 2007.
- [13] B. Tekinerdogan., S. Bilir S., C. Abatlevi. Integrating Platform Selection Rules in the Model Driven Architecture Approach. In: Aßmann U., Aksit M., Rensink A. (eds) Model Driven Architecture., Springer LNCS. vol 3599, Berlin, Heidelberg, 2005.
- [14] B. Tekinerdogan, S. Yağız, K. Özcan, İskender Yakın. Integrated Process Model for Systems Product Line Engineering of Physical Protection Systems, in: Proc. Of 10th Int. Symposium on Business Modeling and Software Design, Springer LNBIP, Berlin, 2020.
- [15] E. Tüzün, B. Tekinerdogan, M.E. Kalender, S. Bilgen. Empirical Evaluation of a Decision Support Model for Adopting Software Product Line Engineering, Information and Software Technology, Elsevier, Vol. 60, Pages 77–101, April 2015.