# HoS-ML: Socio-Technical System ADL Dedicated to Human Vulnerability Identification

Paul Perrotin, Nicolas Belloir, Salah Sadou, David Hairion, Antoine Beugnard

# HoS-ML: Socio-Technical System ADL Dedicated to Human Vulnerability Identification

Paul Perrotin[1], Nicolas Belloir [2], Salah Sadou[3], David Hairion[5], Antoine Beugnard[4]
[1]Chair of Naval Cyber Defense, Ecole navale - CC 600, F29240 Brest Cedex 9
[2]Military Academy of St-Cyr, CREC St-Cyr, IRISA, Vannes, France
[3]Université Bretagne Sud, IRISA, Vannes, France
[4]Institut Mines-Télécom Atlantique, Lab-STICC CNRS UMR 6285, Brest, France
[5]Naval Group, France
contact : paul.perrotin@ecole-navale.fr

***Abstract*—Due to the increasing complexity of modern systems, the level of responsibility dedicated to the human operator has grown, particularly in Socio-Technical Systems (STS) where humans are considered as subsystems. Like every system, the human operator can fail by behaving in undesired ways, and consequently have a negative impact on the system. Thus, to improve the resilience of the overall system, it is necessary to manage the vulnerability of humans. In this paper we present an approach to assess human vulnerabilities in an STS through its architecture. We propose a model that describes the STS, based on human characteristics having a significant impact on human vulnerabilities. We define an assessment metric for each characteristic. We propose an approach allowing not only to assess the vulnerability of a specific human in the system, but also to understand how a vulnerability propagates through the system. We implemented this approach with a dedicated architecture description language, called Hos-ML, allowing the architect to deal with STS vulnerabilities.**

***Index Terms*—Human vulnerability, human models, socio-technical systems, cybersecurity**

## I. Introduction

Nowadays, modern complex systems rely on many devices and other systems, which can be more or less interconnected and intelligent. They are mostly software intensive and they interact with a set of actors such as humans and organizations. A comprehensive vision on these complex systems must include these actors as parts of the system. For these reasons, such systems are often referred to as "Socio-Technical Systems" (STS) [9]. In the past, several studies have been dedicated to the rigorous design and verification of these systems [2]. They mainly focused on the reliability point of view: systems and subsystems were designed in order to be resistant to inner and accidental malfunctions. However, the last decade has seen an increase in a particular type of malfunction which is external and malicious: cyberattacks. Such external events try to access the STS in order to negatively impact it: allowing access to sensitive data, influencing it in a conductible way, destroying it... Such attacks have become possible because attackers used system vulnerabilities. A lot of work has focused on decreasing system vulnerabilities [6]. Nevertheless, as mentioned by [25], the number of attacks based on the influence of human behavior (social engineering) and resulting in a loss of integrity is increasing. This can be dramatic in STS because it adds another layer to the cathedral of vulnerability issues that may arise from the composition of subsystems. Indeed, human as a constituent system introduces a degree of uncertainty concerning the security of the services provided by the system.

The complexity of the challenge cited above also comes from the combination of two categories of vulnerabilities: systemic and human. To our knowledge, the second category seems to have been the focus of less research. Thus, before trying to deal with the problem of combination, we think that it is first necessary to better formalize the analysis of human vulnerability. Thus, the problem can be formulated as follows: how to improve the detection of security failures in human-centered systems? Taking into account human factors is thus fundamental. Several cognitive and emotional aspects were studied [4, 13, 22], such as the impact of "mental workload" or "stress" on the efficiency and performance of individuals and teams in decision making. We propose to introduce into the backbone of a human-centered STS the mechanisms allowing to analyze its vulnerability. Our approach is based on three elements: i) human vulnerability modeling. ii) architecture description language facilitating vulnerability analysis. iii) estimator of the STS's vulnerability based on the two previous elements. These constitute the contributions of this paper.

The remaining of the paper is organized as follows: Section II states the problem and gives the position of our work with respect to other current work. In section III we identify pertinent human factors, relevant to security, starting from a generic model. In section IV we present HoS-ML, a language using these factors in order to allow architects to describe the architecture of a human-centered STS. In the same section we also define the way to assess human vulnerability and its propagation to the rest of the system. Before concluding in section VI we discuss in section V a concrete case study that we conducted with our industrial partner

## II. State of the art

A human-centered STS corresponds to an organization where each person has specific assigned tasks for a common

goal. Studying the vulnerability of such a system (organization) leads to studying the vulnerability of each of its members and how one person's vulnerability can impact the others. Before that, we must be able to identify and then to formalize the human characteristics involved in the notion of vulnerability. Below, we describe the most current state of the art on these themes before giving an outline of our approach.

### A. Modeling Human Factors

Assessing the vulnerability of a human as an actor in an STS (human operator) requires taking her/his traits into account and considering each one as a vulnerability factor. For instance, a human operator who is resilient to stress will be able to work in stress-generating positions. Unlike traditional security analysis approaches, which rely on technical aspects, for humans we need to rely on psychological and sociological aspects. The latter is often related to her/his position in the STS.

In the literature, few studies have focused on this problem. We have identified two types of approaches: organizational approach [14] and more systemic one [6, 16]. In [14], the author established a model to describe humans in the management of risk-investment constructs, in security investments and in constructive feedback situations for security incidents. The approach presented by the author results in a feedback on incidents related to human factors. This feedback helps managers to have the best managerial policy according to the targeted person. The author described what he has called "human factors" in a model with an organization level management in order to be more resistant to social engineering attacks. The human model is organized into two parts: i) direct factors, describing the characteristics of a person. ii) indirect factors, describing the constraints of the human role in her/his environment. This approach describes the human operator with properties. It makes it possible to specify *a posteriori* which property is at the origin of a given incident. This approach is very interesting because the proposed model aims to characterize what is expected by a person at a given position. Nevertheless, it may be improved to make it useful in the detection of human vulnerabilities in STS. Indeed, we think that the list of properties under study needs to be expanded. Some proposed properties also must be refined. This is discussed with more detail in the next section. Another weakness is that the properties can not be expressed with values. So, it is not possible to have a fine-grained analysis of those properties. Moreover, some factors may influence others and those influences may have a significant impact on the human vulnerability. Thus, it is not easy to deduce the emergence of human vulnerability. For instance, a low value on a given human factor is not enough to make it a vulnerability. Generally, this is achieved through a combination of specific values for certain factors [21].

Human modeling is only a first step in modeling an STS. Indeed, a language that includes security requirements needs to be defined to design STS models. A design language called STS-ML [16] allows the modeling of an architecture from security requirements. The authors proposed an approach for modeling the system architecture including the definition of each person as a role. The system description also defines the data access of each role, as well as each role's objectives and the interaction between the roles. Finally this model allows to represent a threat in the modeled system. The result is a security requirement document based on the expression of the architecture and the threats identified in it. However, the definition of the roles consists only on specifying objectives and held documents. This is not a consistent definition of a human, as operator, with her/his properties. Thus, we can not define human vulnerability using this language alone.

### B. Vulnerability Propagation

A model representing each human operator is not enough to be able to evaluate the vulnerability of the whole system. It is also necessary to assess human vulnerability and evaluate how it may be propagated to the other operators of the system. To do that, once the operators have been described, we need to simulate human vulnerabilities of each operator and how they propagate through operators and roles. In [8] the authors proposed an approach based on Bayesian networks to simulate a technical vulnerability and its propagation in an information system. In this approach the internal and external factors of an information system are simulated to model the threat propagation. After the simulation, they obtain the path of the threat propagation with the highest probability. Such a result allows the architect to modify the architecture of the simulated information system. However, to our knowledge there is no work on the propagation of human vulnerability in an STS.

### C. Proposed Approach

The approach proposed through this paper aims at completing and combining the approaches described above in order to allow an efficient analysis of human-centered STS vulnerability. Our proposition consists in three steps:

1) Simplify the human operator description by focusing only on the factors having an impact on her/his vulnerability. This will lead to a vulnerability-oriented human model. Then, we define a scale of values for the identified factors and determine the combinations of factors that impact human vulnerability.
2) Propose an architectural description language which describes the STS in terms of operational connections between human operators, taking into account their factors.
3) Estimate the actual vulnerabilities of human operators and analyze the propagation of the assessed vulnerabilities in the human-centered STS.

## III. Security-Oriented Human Factors

The representation of a human operator through factors leads to identifying the important factors according to the desired point of view (here security) and then identifying the relationships between these factors. In the following, we start with the description of a generic human model containing only the categories of factors before describing these categories.

### A. Factor-Oriented Human Generic Model

Any given person is more or less resilient depending on her/his own factors. Variations on those factors may increase or decrease the risk of making an error. For instance, depending on her/his level of conscientious and tiredness, a human may have a more or less appropriate reaction to a specific situation. The environment in which the human operates is also a parameter having an impact on her/his reaction and therefore revealing certain vulnerabilities. For instance, in a stressful situation, some people may lose their capabilities to correctly evaluate a situation and thus make an incorrect decision.

In fact, in [14] the author showed that human vulnerability depends on direct and indirect factors. Direct factors characterize the human independently of any external influence. Indirect factors correspond to elements linked to the environment in which the person will operate, eg. the task assigned to her/him, which can influence her/his behavior. Direct and indirect factors can influence each other. For instance, a person who usually is easily affected by stressful situations can become resilient to these situations with specific management. So we can consider that there is a relationship between a direct factor "emotional stability" and an indirect factor "management".

Relationships are another element that must be considered. Often in their work, people are part of a team. In a group of people there are often those who influence and those who are influenced. So, each interaction may have an impact on some human's factors that we have to consider.

Below we describe these factors which come mainly from research results in the human sciences. We present them from the security point of view in an STS. In addition, we add other factors highlighted during our collaboration with our industrial partner (given in bold bellow).

### B. Security-Related Direct Factors

we identified a set of security-related direct factors. We grouped them depending on whether they are inherent to the person or dependent on the presence of other persons.

*1) Inherent Factors:* This kind of factors are inherent to the person in the sense that their assessment does not require facing the person with other people.

- Skill: defines the human operator's ability to perform certain tasks in a given area and her/his level of expertise in that area [14].
- Experience: defines the level of knowledge of the human on a specific position [17].
- Reliability: relates to the predisposition of a human to not commit mistakes in her/his position [2].
- Conscientious: allows us to take into account the ability of a human operator to respect the procedures in order to ensure a job well done. Indeed, if the human operator pays little attention to the process she/he is performing, this can lead to some vulnerabilities [23].
- **Confidence**: represents the level of trust an organization has in a human operator performing a task or occupying a position. It is an important factor in the definition of security. Indeed, this trust comes from necessity and does not represent in any way the trust that one has in the operator as a person. It is the trust that one must have in the fact that, given the context of the task, the human operator will do the job well.
- **Robustness**: represents the ability to handle a workload whether physical or mental. Indeed, certain positions and roles may require greater physical or mental strength and if the concerned human operator does not support them, this leads to a vulnerability.

As already said, the last two factors raised from the expertise of our industrial partner in the naval and military fields.

*2) Social-Oriented Factors:* These factors relate to people whose qualifications require them to be in contact with other people.

- Informational level: highlights the subject's ability to pay attention to the various security policies and best practices implemented in the company [26].
- Organizational Cooperation: defines the ability of a person to obey an organization's instructions even if they differ from her/his conception of the work. This factor is inspired by [10] and is an extension of the concept of human-machine cooperation to a wide view of cooperation in an organization.
- Relationship: represents the relationships that an individual may have with other members of the organization, without this being related to his/her tasks or position.
- Emotional stability: This is one of the measurements characterising the human in a stressful situation [4].

### C. Security-Related Indirect Factors

Indirect factors are those that may have an influence on a human operator while they are linked to her/his environment. Below we give the definition of the indirect factors from literature that we found relevant to security.

- Management: designates the level of security flaws allowed due to the fact that a human operates according to a given management team [19].
- Security Policy: represents the level of security good practices required by a task or a position [24].
- Culture: designates the level of security flaw associated with the culture of the company. Combined to other direct or indirect factors this factor may promote the emergence of vulnerabilities [20].
- Communication: refers to the level of communication needed by the position. If the position requires active communication but the human operator does not assume it, it can promote the emergence of a vulnerability [7].
- Task Exigency: represents all elements that the task need to be realized in good condition like the time constraint or the complexity of the task [18].
- Resource: Indeed, a mismatch between the provision and the needs for the performance of a task, can lead the human operator to a vulnerable situation. This concept generalize the concept of budget presented in [1].
- Professional Relationship: represents the human relationships related to her/his role in the STS [22].

- Position: defines the importance of the position in the organization. Indeed, this is not just limited to the person's place in the hierarchy but also to her/his technical role.

### D. Factor Assessment

Some of the previous factors are easily measurable, like skill or experience, while others are much less so. For example, giving a value to the emotional stability of a human operator is not easy. Moreover this evaluation could vary from one assessor to another as well as according to the context. Thus, it is necessary to propose a way to valuate them which limits this disparity. We have identified two types of characteristics: the so-called digital characteristics that could be noted on an ordinal scale and the adjective characteristics for which another method of assessment has to be applied. Concerning the digital characteristics, we decided to apply a simple scale: from 1 to 5, where 1 is set when the score is low and 5 when it is perfect.

When setting a digital value is not appropriate, we have decided to use an adjective usage for the notation. We applied the approach already used in the *Computer Security Handbook* [4].Indeed, it recommends considering the "quality" level of collaborators to be defined by adjectives chosen to allow precise gradation. We have therefore chosen this method corresponding to the notion of a gradient, which makes it possible to be less dependent on the context. Moreover, in the same study we found several characteristics quite close to ours. Thus, we have reused these qualifiers directly in our context.

So, for the adjective characteristics we use the following assessments:

- Conscientious: *efficient, responsible, compulsive, pompous, slavish*.
- Communication: *active, energetic, quiet, shy, silent*.
- Organizational cooperation: *Trusting, Deferent, Reticent, Untrusting*) proposed in [10].
- Emotional stability: *stable, unemotional, anxious, moody*.
- Culture: *Simple Structure, Machine Bureaucracy, Professional Bureaucracy, Divisionalised From, adhocracy* [11].
- Management: Based on the meta-analysis approach [13], we selected (*Organizational, Cognitive Behavioral, Relaxation, Multimodal, Individual Focus*).

For the relationship factor, we do not detail the level of relationship and we only need to know if it exists or not. So, the value is a boolean.

## IV. Human-Oriented Architecture Description and Vulnerability Assessment

In order to evaluate human vulnerability of an STS, we propose a combination of a human-oriented security architectural modeling language to describe the human architecture of the STS with a vulnerability detection approach.

### A. Human-Oriented Security Architecture Modeling Language

To make human factors usable when designing an STS, we propose a language called HoS-ML(Human-Oriented Security architecture Modeling Language), based on the existing language STS-ML [16]. This language allows the system engineer to describe an STS architecture, focusing on roles and human operators playing these roles. Each can be characterized by factors and relationships described in the previous section. The main idea is to provide a way for the system architect to evaluate the security of the ideal STS architecture represented by roles, and a more realistic architecture represented by human operators. Indeed, human operators may not be perfectly compliant within the factors modeled in roles. For instance, an operator may be impacted by stress or may be less vigilant due to a long working period. As we will see in the next section, considering variations between roles and operator characteristics will allow system engineers to evaluate the impacts of those variations on the overall security of the system. The architectural model also makes it possible to simulate the propagation of a threat into this STS.

### B. Estimating Individual Vulnerability

Since in our use case, the system is made up only of humans, the source of the vulnerability is human. The organizational structure will therefore allow the vulnerability to spread throughout the whole system. For this reason we need to assess individual vulnerabilities before estimating their propagation to the rest of the system.

The method we propose to estimate human operator vulnerability takes into account the two following points:

1) The difference between the operator's direct factors and those required by the corresponding role. The difference between the operator and the role for the same direct factors can increase the probability that the operator will be a source of vulnerability in the system. For example, if an operator has less skill than expected this will probably lead to a vulnerability if the indirect factor *Task Exigency* is high.
2) Indirect factors through the various links between different factors. For example, some management approaches can help a person to lower stress levels.

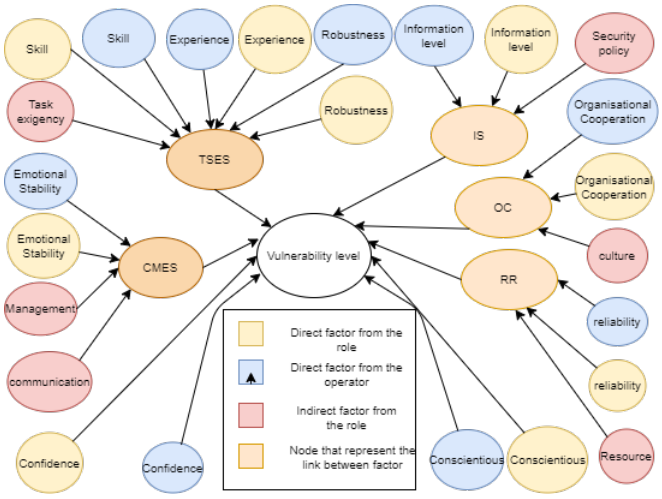


Fig. 1. Bayesian network for estimating operator's vulnerability level.

Figure 1 gives a representation of the Bayesian Network (BN) which implements all the retained human factors to estimate the human operator vulnerability. BN has the advantage to offer a distribution of probabilities on a model having many links and nodes. It can also learn from existing data that can be more accurate than using probabilities extracted only from the literature that may be too general or corresponding to a particular case.

The of kind of links, pointed out in point 2 above, are represented in Figure 1 by intermediate node (nodes labelled with initials of the factors). For each intermediate node we list here the influences that we have been able to extract from the literature. Some of them will decrease the vulnerability while others will increase it:

**IS**: According to [12] the security policy can have an impact on the informational level factor.

**CMES**: The management and the communication factors have an impact on the emotional. stability [13].

**TSES**: The task exigency factor is a source of vulnerability if the operator do not have enough skills, experience and robustness [18].

**OC**: This node represent the impact the human operator's cooperation ability on her/his vulnerability. Indeed, an human operator who has difficulty to cooperate will be more vulnerable in a company where the cooperation is needed [5].

**RR**: This node represent the impact of reliability on human vulnerability. Indeed, a lack of resources can be an element of vulnerability [15].

### C. Vulnerability Propagation

Vulnerability of operators is inferred from a graph, where operators are vertices, and their relationships are edges. Before analyzing the propagation of an operator's vulnerability through the STS, it is important to first estimate its level of potential impact on the STS. Indeed, the position of the vulnerable operator in the organization and the level of confidence, which the other members have in her/him, impact the level of propagation of the vulnerability in the STS. Thus, to each operator A we assign an impact level on the STS as follows:

$Impact(A) = (PositionOf(A) + ConfidenceIn(A))/2$

Our approach to estimate the propagation of a vulnerability from an operator A to an operator B, in case of direct or indirect relationship between them, is based on the work carried out on stress contagion presented in [3]. Thus, we estimate the propagation of vulnerability with 2 elements: i) the structural links between the two operators, which are explicit in the architecture of the STS, and ii) the relationships between operators as individuals, whether professional or personal. Indeed, a person can contaminate another by simple influence through their personal or professional relationship [3]. For structural links we consider the 2 links present in our ADL: the delegation of objectives and the transmission of documents.

## V. Case Study

To evaluate the applicability of our approach, we applied it to a real life industrial case study with our partner. It concerns a Maritime Piracy Control STS (MPC-STS) that takes place in a surveillance frigate deployed in a theater requiring active surveillance against piracy. Due to the lack of place, we describe only one scenario of the case study. It follows a recurring process based on four main steps: *detecting, identifying, classifying, proposing an operational response*. These four steps are broken down in the process into several tasks distributed over 5 roles: *Officer, Monitor Chief, Intervention Chief, First Operator and Monitoring Operator*. These roles shares data such as intelligence information, navigation flows, weapon system information…Both these roles and their interactions constitutes the architecture of the STS. All the different role profiles were designed with the help of our industrial partner as well as with the support of different maritime experts.

The objectives through this case study, were to test our approach against a real life case that has already been analyzed by the industrial experts. The attended feed-backs resulting of the case study were about the ability of our ADL to describe a real STS architecture as well as its ability to highlight vulnerabilities that experts could confirm.

The study involved simulating the different possible gaps between each role and its potential instantiated operator with regard to their specified human factors. From a certain gap between the values of the human factors of the operator and those expected for the role, all the operators become vulnerable. However, for the Monitor Operator role the vulnerability was reached with a minimal deviation. In addition, this role can contaminate another role (the Officer). he Monitor Operator's vulnerability does not directly imply a great threat to the STS, as her/his role has a low impact on the latter. However, the contamination the Officer leads to a more critical situation as her/his role has a great impact on STS.

The analysis of this situation by our industrial partner led to the conclusion that the vulnerability is due on the one hand to the architecture of the STS and on the other hand by the fact that the indirect factors for the concerned role do not help to compensate for the differences between the operator's factors compared to those of the role in order to reduce its vulnerability. The reaction of the industrial experts was the following: since it is not possible to improve the structural part of the STS's architecture, or to add a validation operator to face the flaw, the only solution is to train the Monitoring Operator to know the different types of cyber attacks and the potential impact on her/his activities. However, to make the architecture consistent, we need to change the required skill level for this role. Thus, the impact level associated with this role will increase.

Thus, this case study showed two things: i) our architecture description language is able to describe a real life STS. ii) our vulnerability assessment and propagation approach is able to identify flaws in an STS that the experts had not initially identified.Thus, our approach can help system architects to improve STS security.

## VI. Conclusion

In this paper, we have addressed the problem of detecting a human vulnerability in socio-technical systems. We have identified human factors relevant to security aspects and then we defined a human-oriented architecture description language called HoS-ML. We have also proposed an approach based on Bayesian networks in order to simulate human vulnerabilities, considering the specified factors, and estimating their impact on the modeled STS.

As we noted during our case study, this language and its associated approach together have enabled our industrial partner's experts to find unsuspected flaws in the specification of their own STS. This shows that the proposed approach makes it possible to identify unsuspected vulnerabilities in such complex systems. However, the literature we found in human science is not always very explicit on some possible links between human factors. So, in a future work we plan to collaborate with a human sciences team in order to improve our vulnerability detection tool. We mainly aim to investigate the problem of the simultaneous existence of several vulnerabilities carried by different individuals. Indeed, when several individuals are vulnerable at the same time, complex social mechanisms may interact, including group effects in contamination and propagation of vulnerabilities.

## References


[1] Maryam Al-Awadi. Success factors in information security implementation in organizations. 2008.

[2] A. Avizienis, J. . Laprie, B. Randell, and C. Landwehr. Basic concepts and taxonomy of dependable and secure computing. *IEEE Transactions on Dependable and Secure Computing*, 2004.

[3] Niall Bolger, Anita DeLongis, Ronald Kessler, and E. Wetherington. The contagion of stress across multiple roles. *J Marriage Fam*, 1989.

[4] S. Bosworth, Wiley Online Library (Service en ligne), M.E. Kabay, and E. Whyne. *Computer Security Handbook*. Wiley, 2015.

[5] Jennifer Chatman and Sigal Barsade. Personality, organizational culture and cooperation: Evidence from a business simulation. *Administrative Science Quarterly*, 1995.

[6] Fabiano Dalpiaz, Elda Paja, and Paolo Giorgini. *Security Requirements Engineering: Designing Secure Socio-Technical Systems*. MIT Press, 2016.

[7] Gurpreet Dhillon and James Backhouse. Technical opinion: Information system security management in the new millennium. *Journal of Health Psychology*, 2000.

[8] Nan Feng, Harry Wang, and Minqiang Li. A security risk analysis model for information systems: Causal relationships of risk factors and vulnerability propagation analysis. *Information Sciences*, 2014.

[9] A. Gregoriades and A. G. Sutcliffe. Automated assistance for human factors analysis in complex systems. *Ergonomics*, 2006.

[10] Jean-Michel Hoc and Christine Chauvin. Cooperative implications of the allocation of functions to humans and machines, 2011.

[11] Geert H. Hofstede. *Cultures and organizations: Software of the mind*. McGraw-Hill, 1991.

[12] Merete Hagen Janne, Albrechtsen Eirik, and Hovden Jan. Implementation and effectiveness of organizational information security measures, policy. *Information Management & Computer Security*, 2008.

[13] Jac Klink, R.W.B Blonk, AH Schene, and Frank Dijk. The benefit of interventions for work related stress. *American journal of public health*, 2001.

[14] Reza Mortazavi-Alavi. *A Risk-Driven Investment Model for Analysing Human Factors in Information Security*. PhD thesis, University of East London, 2016.

[15] H. Nouri and R.J. Parker. The relationship between budget participation and job performance: The roles of budget adequacy and organizational commitment. *Accounting, Organizations and Society*, 1998.

[16] Elda Paja, Fabiano Dalpiaz, and Paolo Giorgini. Modelling and reasoning about security requirements in socio-technical systems. *Data Knowl. Eng.*, 2015.

[17] Simon Parkin, Aad van Moorsel, and R. Coles. An information security ontology incorporating human-behavioral implications. In *Proc. of the 2nd International Conference on Security of Information and Networks*, 2009.

[18] Miguel A. Quinones, J. Kevin Ford, and Mark S. Teachout. The relationship between work experience and job performance: A conceptual and meta-analytic review. *Personnel Psychology*, 1995.

[19] Von Solms Rossouw. Information security management: why standards are important. *Information Management Computer Security*, 1999.

[20] Anthonie Ruighaver, Sean Maynard, and Shanton Chang. Organisational security culture: Extending the end-user perspective. *Computers Security*, 2007.

[21] Bruce Schneier. The psychology of security. In *Progress in Cryptology – AFRICACRYPT*. Springer, 2008.

[22] Markus Schumacher, Eduardo Fernandez, Duane Hybertson, and Frank Buschmann. *Security Patterns: Integrating Security and Systems Engineering*. John Wiley Sons, Inc., 2005.

[23] Kerry-Lynn Thomson and Johan van Niekerk. Combating information security apathy by encouraging prosocial organisational behaviour. *Information Management Computer Security*, 2012.

[24] Harold F. Tipton and Micki Krause. *Information Security Management Handbook, Volume 1*. Auerbach Publications, 2007.

[25] Verizon. Data Breach Investigations Report. Technical report, Verizon, 2016.

[26] Cheryl Vroom. Towards information security behavioral compliance. *Computers Security*, 2004.