A multi-criteria ranking of security countermeasures

A Multi-Criteria Ranking of Security Countermeasures [*]

Nicola Nostro[⊖], Ilaria Matteucci[♮], Andrea Ceccarelli[♯], Francesco Santini[◇], Felicita Di Giandomenico[♭], Fabio Martinelli[♮], Andrea Bondavalli[♯]
[⊖]ResilTech s.r.l, Pontedera, Italy, [♯]University of Firenze, [♭]ISTI-CNR, Pisa, Italy, [♮]IIT-CNR, Pisa, Italy, [◇]University of Perugia
nicola.nostro@resiltech.com, [andrea.ceccarelli, bondavalli]@unifi.it
[ilaria.matteucci, fabio.martinelli]@iit.cnr.it
f.digiandomenico@isti.cnr.it, francesco.santin@dmi.unipg.it

ABSTRACT

We propose a multi-criteria framework for ranking controlling strategies according to several weights, such as *delay-time*, *resource cost*, and *success-probability of attacks* defined via quantitative threat analysis. Therefore, by assigning a different priority to weight-dimensions, we can rank controllers in an adaptive way. We exemplify our approach on the Customer Energy Management System (CEMS), that acting as an interface among different systems, is exposed to attacks. We consider both the Man in the Middle (MiM) and the Denial of Service (DoS) attacks.

1. INTRODUCTION

This paper presents an adaptive multi-criteria framework for ranking controlling strategies to minimize the impact of attacks, thus helping in defining controllers that favour security, cost-effectiveness, or energy efficiency. We enhance the results in [12] by introducing a *multi-criteria* analysis to drive the countermeasures ranking.

The proposed framework consists of two main steps: i) a security analysis, performed by *ADversary VIew Security Evaluation* (*ADVISE*) formalism [8] in order to evaluate attack points and paths with respect to several criteria and to identify those that are feasible, and to rank attack weights, from the perspective of an attacker; ii) an evaluation of different run-time execution traces, corresponding to several countermeasures on the basis of the multi-criteria analysis results. The classification is driven by the system's requirements hierarchy and it is modelled by the order of the considered semiring [1]. We exemplify the proposed framework on the *Customer Energy Management System* [7], a service of the low voltage grid for an advanced energy management.

The paper is structured as follows: Section 2 describes the multi-criteria framework for the ranking of countermeasures. Section 3 recalls controllers and formally defines the multi-criteria ranking. Section 4 exemplifies our approach on the CEMS reference scenario. Section 5 discusses related work and Sec. 6 draws conclusions and future work.

2. RANKING SECURITY STRATEGIES VIA SECURITY ANALYSIS

System requirements are expressed in the form of a hierarchy, with stringent requirements at top level, and less stringent ones in lower positions. In our framework, the ranking of controllers drives the selection of the best one with respect to the current hierarchy of requirements. In case of reorganization of the requirements hierarchy (*e.g.*, in multiple-phased systems, whose operational life spans multiple phases characterized by different functional and non-functional requirements), a new ranking is easily provided without performing the analysis of the whole system again. The framework is composed by the following steps:
- the system specificationin terms of its functional and non-functional requirements;
- once the system has been modelled, a security assessment is performed, thus providing a series of possible attacks that can be perpetrated on the system. The analysis considers several criteria and provides an estimation of each attack according to all criteria under investigation. Since the definition of the profile of an attacker is at the basis of the risk&threat evaluation processes [11], we defined two attackers' profiles: *hacker* and *civil activist*;
- the exploitation of functional and non-functional system's requirements to classify countermeasures according to possible trade-off among different criteria. Given the variety of potential attackers behaviour, we define several controllers that follow the attacker's behaviour step by step;
- the adaptive ranking of countermeasures driven by the requirements' hierarchy. Indeed, we use the information on the attack, obtained through the security analysis, to classify and rank countermeasures also according to the system's requirements.

3. MULTI-CRITERIA CLASSIFICATION OF QUANTITATIVE COUNTERMEASURES

We adopt *semirings* as the algebraic formalism to represent metrics used to rank the countermeasures.

Definition 3.1 (c-semiring [1]). *A c-semiring is a five-tuple* $\mathbb{K} = \langle K, +, \times, \bot, \top \rangle$ *such that* K *is a set,* $\top, \bot \in$

[*]This work has been partially supported by the TENACE PRIN Project (n. 20103P34XC) funded by the Italian Ministry of Education, University and Research, and by the European FP7 Project SmartC2Net under grant agreement n. 318023.

SAC 2016, April 04-08, 2016, Pisa, Italy
© 2016 ACM. ISBN 978-1-4503-3739-7/16/04…$15.00
DOI: http://dx.doi.org/10.1145/2851613.2851878

K, and $+, \times : K \times K \to K$ are binary operators making the triples $\langle K, +, \bot \rangle$ and $\langle K, \times, \top \rangle$ commutative monoids (semigroups with identity), satisfying i) *(distributivity)* $\forall a, b, c \in K.a \times (b + c) = (a \times b) + (a \times c)$, ii) *(annihilator)* $\forall a \in A.a \times \bot = \bot$, *and* iii) *(top element)* $\forall a \in K.a + \top = \top$.

The idempotency of $+$ leads to the definition of a partial ordering $\leq_K$ over the set K (K is a poset). Such complete partial order is defined as $a \leq_K b$ if and only if $a + b = b$, and $+$ becomes the *least upper bound* (*lub*) of the lattice $\langle K, \leq_K \rangle$. This intuitively means that b is "better" than a. As a consequence, we can use $+$ as an optimisation operator and always choose the best available solution. Other derived properties are [1]: *i)* both $+$ and $\times$ are monotone over $\leq_K$, *ii)* $\times$ is intensive (i.e., $a \times b \leq_K a$), iii) $\times$ is closed (i.e., $a \times b \in K$), and *iv)* $\langle K, \leq_K \rangle$ is a complete lattice where $\bot$ and $\top$ are its bottom and top elements, respectively.

A countermeasure (or controlling strategy) [3] is a run-time execution trace of a controller E that follows the behaviour of a target F step by step acting according to control rules in Tab. 1. The resulting behaviour is denoted by $E \triangleright^{\mathbb{K}} F$, where $\mathbb{K}$ is the semiring used for specifying quantities to quantitatively estimate the contribution of each countermeasure on the system workflow. The alphabets of E, F, and of the resulting process $E \triangleright^{\mathbb{K}} F$ are different, as E may perform *control actions* of the form a, $\boxplus a.b$, $\boxminus a$ for $a, b \in Act$, denoting respectively the actions of *acceptance*, that means that the action of F is accepted by the controller E, *suppression*, that means that the action of F is hidden (becomes τ) by E, and *insertion*, that introduces correct action in front of the action of F. Each action of both the controller and the target is associated to a value of the semiring $\mathbb{K}$, i.e., we have a couple (a, k) as label, where $k \in \mathbb{K}$ is a quantity associated to the effect a.

Given an execution trace $t = (a_1, k_1) \cdots (a_n, k_n)$, we define *label* $l(t) = a_1 \cdots a_n$, and *run weight* $|t| = k_1 \times \ldots \times k_n$. Hence, we are able to rank different strategies and, eventually, select the "best" one as follows.

Definition 3.2. *[3] Given an agent F, and a semiring $\mathbb{K}$, a controller E_2 is* better *than a controller E_1 w.r.t. F, $E_1 \leq_{\mathbb{K},F} E_2$, iff $\|E_1 \triangleright_{\mathbb{K}} F\| \leq_{\mathbb{K}} \|E_2 \triangleright_{\mathbb{K}} F\|$. E_2 is* always better *than E_1, $E_1 \leq_{\mathbb{K}} E_2$, iff $E_1 \leq_{\mathbb{K},F} E_2$, for any F.*

Using the fact that the Cartesian product of semirings is still a semiring, we can rank countermeasures according to several criteria by exploiting the lexicographic order among the considered semirings.

Definition 3.3. *Let $\langle K_1, +_1, \times_1, \bot_1, \top_1 \rangle$ and $\langle K_2, +_2, \times_2, \bot_2, \top_2 \rangle$ be c-semirings. Then, the associated lexicographic order $\leq_l$ on $K_1 \times K_2$ is given by:*

$$\langle k_1', k_2' \rangle \leq_l \langle k_1'', k_2'' \rangle \textit{ if } \begin{cases} k_1' <_1 k_1'' & \textit{or} \\ k_1' = k_1'' \wedge k_2' \leq_1 k_2'' & \end{cases}$$

4. RANKING OF COUNTERMEASURES IN THE CEMS CASE STUDY

The *Customer Energy Management System* (*CEMS*) is an application service or device of low voltage grids for an advanced energy management, based on tariff information and an integration of *Distributed Energy Resources* (*DER*) for a more balanced grid stability. A basis for this control network is established by the deployment of a comprehensive *Advanced Metering Infrastructure* (*AMI*) for *Automated Meter Reading* (*AMR*), able to monitor the electricity consumption of households collected by smart meters (see [7]).

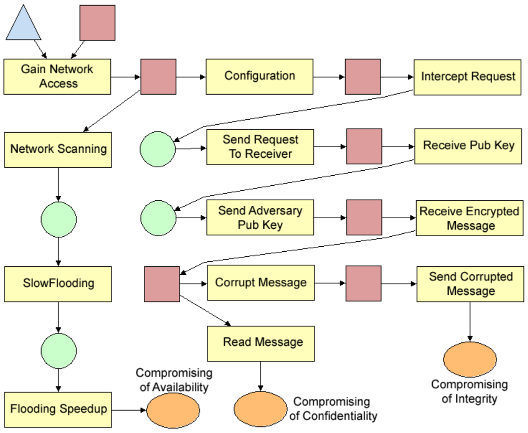

Figure 1: ADVISE Attack Execution Graph for Man in the Middle and DoS attacks.

Since CEMS may operate in a very hostile environment, our security analysis focuses on two well-known attacks potentially harmful for the CEMS functionalities, namely: 1) a *Denial of Service (DoS)* attack, consisting in the introduction of relevant quantity of noise in the bi-directional flow of information between CEMS and the AMR gateway. This may lead to a halt failure of the AMR gateway or CEMS or to a delay of the Energy Management Gateway (EMG) activity, thus reducing availability of the energy distribution system; 2) a *Man in the Middle (MiM)* attack, capturing messages exchanged between EMG and CEMS or between CEMS and the higher control layer. For instance, the attacker can delay the messages or alter their content to produce an undesired effect, or simply collect data, thus causing a violation of integrity or confidentiality.

The first step of the *Security Model Based Assessment* is the definition of the profile of an attacker. We assume two profiles, a hacker and a civil activist, in which the hacker has high technical skill, and might operate on commission driven by gain, while the civil activist, moved by ideological motivations, has a lower technical skill with respect to the hacker. We exploit the ADVISE formalism and the related simulator. Fig. 1 shows the *ADVISE Attack Execution Graph* (*AEG*) of both the DoS and MiM attacks, which aim at achieving the three attack goals: availability, confidentiality, and integrity violations. The AEG represents the sequence of attack steps (rectangles in the figure) the attacker has to perform in order to realize the goal (the three ovals in figure). Squares represent different access domains owned and triangles are the attacker skills regarding the next attack step. Circles denote the knowledge the attacker acquires while perpetrating the attack. We also define a set of access domains, knowledge, attack skills, and attack preferences, initially owned by different profiles of attackers, which represent the input to the ADVISE model. Moreover, the attack steps of the model have a specific time duration, cost, success probability, and detection probability, which are different according to the attacker competence and technical skill, and represent the additional set of input to the model.

Table 1: Semantics definitions for quantitative control rules.

$$\frac{E \xrightarrow{a,k} E' \quad F \xrightarrow{a,k'} F'}{E \triangleright^{\mathbb{K}} F \xrightarrow{a,k*k'} E' \triangleright^{\mathbb{K}} F'}\ (\mathrm{A}) \qquad \frac{E \xrightarrow{\boxminus a,k} E' \quad F \xrightarrow{a,k'} F'}{E \triangleright^{\mathbb{K}} F \xrightarrow{\tau,k*k'} E' \triangleright^{\mathbb{K}} F'}\ (\mathrm{S}) \qquad \frac{E \xrightarrow{\boxplus a.b,k} E' \quad F \xrightarrow{a,k'} F'}{E \triangleright^{\mathbb{K}} F \xrightarrow{b,k} E' \triangleright^{\mathbb{K}} F}\ (\mathrm{I})$$

Thus the described AEG applies both to the hacker and to the civil activist. Due to the different abilities of the considered attackers, the probability to successfully get through the attack steps is higher for the hacker than for the civil activist. Note that the setting values used in the model are for illustrative purpose and should be used only for relative comparison between the two profiles.

Initially the attacker has to gain the network access, *Gain Network Access* in Fig. 1, a common step for the three goals.

The DoS attack is attempted by performing the *Network Scanning* step to identify potential vulnerabilities of the system. Once the network scanning has been successfully passed, the attacker attempts the *SlowFlooding* step. In the meantime, the attacker floods CEMS with a number of packets, thus overloading the target's bandwidth and resources, to achieve degradation of availability. However, to prevent the system from recognizing the attack and defending itself, the flooding is started slowly, and only after a preliminary degradation of the performance, represented by successfully overcoming the *SlowFlooding* step, the next *Flooding Speedup* attack step is carried out.

The MiM attack performs the *Configuration* step, which is the activity required by the attacker to intercept messages on the network. The next attack steps are related to the Public-Key encryption: i) the attacker intercepts a conversation request from CEMS (EMG) with its public key; ii) attacker sends a conversation request to EMG (CEMS) with its own public key; iii) EMG (CEMS) receives a request, and sends a reply encrypted with the attacker's key; iv) attacker sends a reply encrypted with the CEMS's key (EMG's key), intercepted at step *i)*; v) attacker receives from CEMS (EMG) a message encrypted with the attacker's key. At this point the attacker can decrypt the message, thus obtaining sensitive information, and realizing the confidentiality violation goal, or s/he can modify the message and send it to EMG (CEMS), thus achieving the integrity violation goal.

4.1 Evaluation of Countermeasures on CEMS

The execution of the ADVISE model provides information related to: *i)* cost, *ii)* time, and *iii)* success probability to achieve a specific attack goal by one of the two attackers as shown in Tab. 2, Tab. 3, and Tab. 4. The cost measure is modelled by the weighted semiring $\mathcal{W} = \langle \mathbb{R}^+ \cup \{\infty\}, \min, +, \infty, 0\rangle$. This cost represents the total amount of hardware/software resources spent by the attacker (to attack) and by the defender (to defend). The delay suffered by the system (again, considering attacks and countermeasures) needs to be reduced as much as possible. To accomplish this, if t is time-cost of an action, then we model it as $1/t$ and we adopt the *fuzzy semiring* $\mathcal{F} = \langle [0..1], \max, \min, 0, 1\rangle$ (we suppose always $t \neq 0$). In this way, we compose two delays $t_2 > t_1$ by selecting the higher one ($1/t_2$ with respect to min), and we prefer the lower delay ($1/t_1$ with respect to max). Hence we minimize the bottleneck delay during the system execution. Finally, the success probability of an attack is represented by the probabilistic semiring $\mathcal{P} = \langle [0..1], \max, \hat{\times}, 0, 1\rangle$. On the attacker's side, such score represents the probability to be successful, while on the defender's side it models the effectiveness probability of stopping the relative attack: if the countermeasure is 100% effective, then the probability p of an attack is annihilated, i.e., $p\hat{\times}0 = 0$ (equal to $\perp$).

Table 2: *DoS* attack: total average time (T), success probability (P), and cost (C).

Time Unit	Success Prob.		Cost Unit	
	Hacker	*Civil activist*	*Hacker*	*Civil activist*
180	0.8664	0.6170	47.8555	45.4275
330	1	0.9962	54.9945	74.7150
440	-	0.9996	-	74.9700

Table 3: *Corrupt Messages* attack.

Time Unit	Success Prob.		Cost Unit	
	Hacker	*Civil activist*	*Hacker*	*Civil activist*
170	0.85104	0.3366	51.0624	35.3430
350	1	0.9994	60	104.9370
430	-	1	-	105

Table 4: *Read Messages* attack.

Time Unit	Success Prob.		Cost Unit	
	Hacker	*Civil activist*	*Hacker*	*Civil activist*
140	0.88589	0.42010	48.72395	39.9095
320	1	0.99960	55	94.9620
400	-	1	-	95

Depending on which measure is prioritized, we have the following categories. **Secure countermeasures:** the controllers are ordered based on their security, measured as the probability of being attacked, modelled by the *probabilistic semiring*. **Economical countermeasures:** a controller can be said to be *economical* when the priority is given to the dimension of cost, modelled by the *weighted semiring*. Hence, the obtained order on controllers considers the cost on each trace, and the optimal controller is the one that costs less. **Ecological countermeasures:** the *fuzzy semiring* models the measure of consumed time, interpreted as ecological impact. Hence the optimal controller is the lowest with respect to the amount of consumed energy.

Referring to the attacks' graph in Fig. 1, the action `GainNetworkAccess` represents a possible first step of an attack. Hence, the *controlling strategies* has a hook before this action in such a way to be prepared to activate a possible countermeasure. Let us consider three countermeasures:

$$\begin{aligned} C_1 &= (\boxplus\texttt{GainNetworkAccess}.\\ &\qquad \texttt{SendAccessRequest}, \langle x_1, y_1, z_1\rangle).C_1' \\ C_2 &= (\boxminus\texttt{GainNetworkAccess}, \langle x_2, y_2, z_2\rangle).C_2' \\ C_3 &= (\texttt{GainNetworkAccess}, \langle x_3, y_3, z_3\rangle).C_3' \end{aligned}$$

where both C_1' and C_2' behave according to the suppression rule on all the other attacker's action, while C_3' behaves by suppressing the `NetworkScanning` action. Then we consider that also this countermeasure works by accepting all the other actions. C_1 and C_2 act differently only on the first action `GainNetAccess`: C_1 modifies the behaviour by introducing an access request to clearly identify the user as an authorized one. C_2 intercepts `GainNetAccess` and suppresses it to avoid communication between attacker and CEMS.

The triple of weights $\langle x_i, y_i, z_i\rangle \in \{\langle t,c,p\rangle, \langle c,t,p\rangle, \langle p,c,t\rangle, \langle p,\ t,c\rangle, \langle c,p,t\rangle,\ \langle t,p,c\rangle\}$ represents the action costs of a countermeasure, and depends on the lexicographic order we select for computation. Referring to Def. 3.2, to compare C_1, C_2, and C_3, we have to apply them on the attacker's behaviour. According to the property of $\times$ operation and to the definition of evaluation of a process, we have:

PROPERTY 4.1. *For each execution trace t_C of a countermeasure C and t_A of attack A $\llbracket t_C \rhd_{\mathbb{K}} t_A\rrbracket = \llbracket t_C\rrbracket \times \llbracket t_A\rrbracket$.*

MiM attack. Let us consider $\langle t,c,p\rangle \in \mathbb{K}$ where $\mathbb{K}$ is the Cartesian product of $\mathcal{F}$, $\mathcal{W}$ and $\mathcal{P}$ (ecological controller) as lexicographic order to rank countermeasures Let us also consider the MiM attacker corrupts the message (Tab. 3), denoted by A_{MiMC}. Then we evaluate all C_1, C_2, and C_3 according to the considered ranking. $\llbracket C_1 \rhd_{\mathbb{K}} A_{MiMC}\rrbracket$ is $\langle x_1, y_1, z_1\rangle \times \llbracket C_1' \rhd_{\mathbb{K}} A_{MiMC}\rrbracket$. Note that, by definition of the countermeasure $\llbracket C_1' \rhd_{\mathbb{K}} A_{MiMC}\rrbracket = \llbracket C_2 \rhd_{\mathbb{K}} A_{MiMC}\rrbracket$. According to Prop. 4.1, being $\llbracket C_2\rrbracket = \langle X_2, Y_2, Z_2\rangle$ where uppercase variables denote the weight of the whole execution trace of C_2 and $\llbracket A_{MiMC}\rrbracket$ are in Tab. 3, then $\llbracket C_2 \rhd_{\mathbb{K}} A_{MiMC}\rrbracket = \langle \min(X_2, C_t), Y_2 \hat{+} C_c,\ Z_2 \hat{\times} C_p\rangle$ where C_t, C_c, and C_p are the delay, the cost, and the success probability of A_{MiMC}
$\llbracket C_1 \rhd_{\mathbb{K}} A_{MiMC}\rrbracket = \langle \min(x_1, \min(X_2, C_t)), y_1 \hat{+} Y_2 \hat{+} C_c, z_1 \hat{\times} Z_2 \hat{\times} C_p\rangle$
According to Def. 3.2, $C_1 \leq_{\mathbb{K}} C_2$. Let us now compare C_2 and C_3: C_3 accepts all the actions, $\langle X_3, Y_3, Z_3\rangle = \langle \perp, \top, \perp\rangle$.
$\llbracket C_3 \rhd_{\mathbb{K}} A_{MiMC}\rrbracket = \langle \min(X_3, C_t), Y_3 \hat{+} C_c, Z_3 \hat{\times} C_p\rangle = \langle \perp, C_c, \perp\rangle$
Hence C_3 is more ecological (less time) than C_2.

Let us now consider to change the hierarchy of requirements by privileging those about the probability of attack, then time, and finally cost requirements, $\langle p,t,c\rangle$ (secure countermeasure). The new ranking leads to a new classification of countermeasures. Being $Z_2 \hat{\times} C_p >_{\mathcal{P}} \perp$, we have $C_3 \leq_{\mathbb{K}, A_{MiMC}} C_2$. The new classification is made starting from the existing evaluation (no need to perform it again).

DoS attack. Let us now consider a DoS attack, C_1 is again always worse than C_2, due to the monotonicity of the semiring. C_2 suppresses all the actions of the DoS attack while C_3 suppresses the `NetworkingAccess` action. This increases the probability of preventing the attack, time, and cost of the trace. Hence,

$$\begin{aligned} \llbracket C_2 \rhd_{\mathbb{K}} A_{Dos}\rrbracket &= \langle \min(X_2, C_t), Y_2 \hat{+} C_c, Z_2 \hat{\times} C_p\rangle \\ \llbracket C_3 \rhd_{\mathbb{K}} A_{DoS}\rrbracket &= \langle \min(X_3, C_t), Y_3 \hat{+} C_c, Z_3 \hat{\times} C_p\rangle \end{aligned}$$

In this case, the hierarchy of requirements is crucial for ranking countermeasures. For example, let us consider the first line of Tab. 2 for the civil activist profile: $C_t = 1/180$, $C_c = 45.4275$, and $C_p = 0.6170$. If $X_2 <_F 1/180$ and $X_3 <_F 1/180$ then $\min(X_2, C_t) = X_2 = \min(X_3, C_t)$. Hence, if $X_2 <_{\mathcal{F}} X_3$ then C_3 is more ecological than C_2, otherwise, if the vice versa holds then C_2 is more ecological than C_3. Finally, if $X_2 = X_3$ then they are equally ecological, and we have to rank them according to their costs. The same reasoning can be done for the cost as well: if their cost is the same, we compare their probability.

5. RELATED WORK

In this section we survey the related work on quantitative measures driving the ranking of countermeasure strategies. Some industrial approaches already exist, such as the Security Analytics of HP Labs [10] that are mostly related to the usage of possible private information and big data. Other works in literature face a similar issue but, from the best of our knowledge, they consider only one criterion. In [6], the authors propose an enhancement of the software engineering approach by considering security solution at design time. The basic idea is to define security solution according to a risk analysis in order to make them more efficient. In [2], the authors introduce the notion of lazy controllers, which only control the security of a system at some points in time, and based on a probabilistic modelling of the system, quantify the expected risk. In [9], the authors deal with probabilistic cost enforcement based on input/output automata to model complex and interactive systems. Associating to each execution trace a probability and a cost measure as a unique weight, it is possible to evaluate the expected cost of the monitor and of the monitored systems. In [4], a notion of cost is used to compare correct enforcement mechanisms (defined as state machines) with different strategies. [5] evaluates controlled strategies to find the optimal one by using dynamic programming, taking into account rewards and penalties with correcting actions.

6. CONCLUSION AND FUTURE WORK

This work presents an adaptive multi-criteria framework for the ranking of countermeasures to select the one that satisfies the system requirements (arranged in a hierarchy) in presence of an attack. We exemplified the proposed approach in terms of the CEMS use case. As future work, we plan to introduce countermeasures as part of the system model to evaluate their impact on the initial system according to the criteria under analysis, and adaptively change the ranking according to the obtained measurements.

7. REFERENCES

[1] S. Bistarelli, U. Montanari, and F. Rossi. Semiring-based Constraint Solving and Optimization. *JACM*, 44(2):201–236, 1997.

[2] G. Caravagna, G. Costa, and G. Pardini. Lazy security controllers. In *STM*, pages 33–48, 2012.

[3] V. Ciancia, F. Martinelli, I. Matteucci, and C. Morisset. Quantitative evaluation of enforcement strategies - position paper. In *FPS*, pages 178–186, 2013.

[4] P. Drábik, F. Martinelli, and C. Morisset. Cost-aware runtime enforcement of security policies. In *STM*, pages 1–16, 2012.

[5] A. Easwaran, S. Kannan, and I. Lee. Optimal control of software ensuring safety and functionality. Technical Report MS-CIS-05-20, University of Pennsylvania, 2005.

[6] G. Elahi, E. Yu, and N. Zannone. Security risk management by qualitative vulnerability analysis. In *Proceedings of METRISEC '11*, pages 1–10. IEEE Computer Society, 2011.

[7] C. Hägerling, F. M. Kurtz, C. Wietfeld, D. Iacono, A. Daidone, and F. Di Giandomenico. Security Risk Analysis and Evaluation of Integrating Customer Energy Management Systems into Smart Distribution Grids. In *CIRED Workshop Proc.*, 2014.

[8] E. LeMay, M. D. Ford, K. Keefe, W. H. Sanders, and C. Muehrcke. Model-based Security Metrics Using ADversary VIew Security Evaluation (ADVISE). In *Proc. of QEST*, pages 191–200, 2011.

[9] Y. Mallios, L. Bauer, D. K. Kaynar, F. Martinelli, and C. Morisset. Probabilistic cost enforcement of security policies. In *STM*, pages 144–159, 2013.

[10] M. Mont, R. Brown, S. Arnell, and N. Passingham. Security analytics: risk analysis for an organisation's incident management process. *HP Lab., TR HPL-2012-206*, 2012.

[11] NIST. Framework for improving critical infrastructure cybersecurity, February 12, 2014. http://goo.gl/X3Uvtj, (accessed Dec. 2015).

[12] N. Nostro, I. Matteucci, A. Ceccarelli, F. Di Giandomenico, F. Martinelli, and A. Bondavalli. On security countermeasures ranking through threat analysis. In *SAFECOMP 2014 Workshops*, pages 243–254, 2014.