Petri Net Modelling of Physical Vulnerability

Francesco Flammini[1], Stefano Marrone[2], Nicola Mazzocca[3],
and Valeria Vittorini[3]

[1] AnsaldoSTS, Innovation and Competitiveness Unit (Italy)
francesco.flammini@ansaldo-sts.com
[2] Seconda Università di Napoli, Dip. di Matematica e Fisica (Italy)
stefano.marrone@unina2.it
[3] Università di Napoli "Federico II", Dip. di Ingegneria Elettrica e Tecnologie dell'Informazione (Italy)
{nicola.mazzocca,valeria.vittorini}@unina.it

Abstract. Several multi-disciplinary aspects need to be addressed in security risk evaluation, including the estimation of risk attributes. One of the most widespread definitions of security risk relates it to the attributes of: probability of occurrence (or rather "frequency") of threats, system vulnerability with respect to the threat (or rather "probability of success of the threat"), and expected consequences (or rather "damage"). In this paper we provide a straightforward generic model based on Stochastic Petri Nets which can be adopted for the quantitative evaluation of physical vulnerability. The model allows to evaluate besides effectiveness parameters (e.g. probability of sensing, assessment, neutralization, etc.) also efficiency related ones (e.g. time to sense, assess, neutralize, etc.). Some examples will be provided in order to show how the model can be used in real-world protection systems applications.

Keywords: Risk Analysis, Model-Based Vulnerability Assessment, Stochastic Petri Nets, Physical Security.

1 Introduction

Nowadays security risk analysis of critical systems and infrastructures is a primary issue. One of the most widespread and simple mathematical model for the quantitative evaluation of the risk associated with a certain threat accounts for: threat occurrence rate, system vulnerability with respect to the threat and expected damage caused by the threat. In particular, the vulnerability parameter represents the (conditional) probability that the attack is successful, that is to say the threat finally damages the target asset. If the asset is not hardened nor protected, than the vulnerability is 1, otherwise it is less than 1. Therefore, in this paper vulnerability is not defined as "a flaw of the system which can be exploited", a widespread qualitative definition especially in computer security.

A foremost problem with the aforementioned risk model is that the vulnerability is very difficult to evaluate. Several ad hoc models have been proposed for risk evaluation, however most of them fail to answer the simple question "Given

S. Bologna et al. (Eds.): CRITIS 2011, LNCS 6983, pp. 128–139, 2013.

a certain threat and a certain protection system, which is the probability that the threat succeeds?" (or in other words, "which is system vulnerability with respect to the threat?"). In this paper we provide a stochastic model based on a certain class of Petri Nets, which allows to give an approximate answer to that question in a way which is as simple as possible. The objective is that the model can be easily customized by only slight modifications to its structure and/or parameters. While (complex) models have been proposed in the scientific literature for risk evaluation in specific applications, to the best of our knowledge no simple generic model exists allowing quantitative vulnerability evaluation based on threat characterization and on the effectiveness and efficiency parameters of the protection system. Furthermore, often risk models are not described in detail for confidentiality reasons [16]. We believe that the generic customizable model described in this paper can help in supporting quantitative vulnerability evaluation in many real-world applications, as demonstrated by the example case-studies we provide.

The rest of this paper is organized as follows. Section 2 provides a brief risk taxonomy and pointers to the related literature. Section 3 describes and discusses the vulnerability model. Section 4 provides some evaluation examples using parameters of real-world applications. Finally, Section 5 draws conclusions and provides some hints about future developments.

2 Basic Definitions and Related Works

The Department of Homeland Security Risk Steering Committee has provided a publicly accessible document which represents a comprehensive reference of risk taxonomy [9]. In the remainder of this section, we will concentrate on definitions which are most related to the topic of this paper.

With reference to a specific threat, the quantitative risk R can be formally defined as follows [14]:

$$R = P \cdot V \cdot D$$

- P (sometimes found as "T", from the initial of *Threat*) is the expected frequency of occurrence of the threat, which can be measured in [events/year];
- V is the vulnerability of the asset with respect to the threat, that is to say the likelihood that an attack is successful, given that it is attempted;
- D (sometimes found as "C", from the initial of *Consequences*) is an estimate of the measure of the expected damage occurring after a successful attack, which can be quantified and expressed in any currencies, e.g. Euros [€].

The vulnerability V is a non-dimensional parameter, since it represents the conditional probability:

$$P(success|threat)$$

Therefore, a quantitative way to express the risk associated to a specific threat is to measure it in lost Euros per year: [€/year]. Though subject to criticism in some applications [13], the risk model defined above has been widely accepted by risk analysts, including the ones belonging to US national laboratories (see e.g. [2]). Nevertheless, the model is so simple to be nearly useless without a supporting methodology for the evaluation of the parameters involved in the analysis. Further details about practical applications and security surveys for vulnerability assessment can be found in references [6,4,1] and in [7,8] in the context of information security. In addition, reference [10] provides the description of a tool to automatically compute the expected annual benefit of a security system starting from the quantitative attributes of threats and protection mechanisms (and their interrelationships) using an extension of the basic risk model described above.

While many different definitions of Vulnerability Assessment may be found in the scientific literature (see e.g. [11]), in this paper we will only refer to the quantitative model-based evaluation of the V parameter of the risk formula. Generally speaking, evaluating the vulnerability corresponds to assessing the effectiveness of protection systems, which poses many challenges. For instance, in [18] a framework is described which addresses (but does not solve) several issues related to the evaluation of deployed security systems, considering both game theory and reliability theory. A simpler model which can be used to assess the vulnerability of a facility with respect to a threat has been adopted by Hennessey et al. [12]. In that model:

$$V = 1 - P_E \qquad P_E = P_D \cdot P_I \cdot P_N \qquad P_D = P_S \cdot P_A$$

Where:

- P_E (probability of effectiveness) is the probability that the physical protection system is effective against the threat;
- P_D (probability of detection) is the probability that the intruder has been detected;
- P_S (probability of sensing) is the probability that a sensor detects the intrusion;
- P_A (probability of an assessment) is the probability that the control room operator correctly assesses the situation and reacts accordingly;
- P_I (probability of interruption) is the probability that the response force gets to the scene in time to neutralise the threat;
- P_N (probability of neutralization) is the probability that the response force successfully neutralises the threat.

In simple words, in order to defeat an attacker, a series of activities must be successfully completed, including sensing, assessment and neutralization. How to quantify such probabilities is out of the scope of this work (some hints on stochastic modeling approaches are provided in [15]); however, estimations of sensing, assessment and neutralization probabilities may be sometimes derived by historical data, simulations and/or or expert judgment. Once such estimations are available, a simple multiplication would be enough to evaluate P_E, with the exception of the P_I parameter, which being time-dependent is more complicated to evaluate. In fact, even if it were 100% effective in terms of detection and neutralization success rate, the security system would be completely useless in case the response force would be unable to stop the perpetrators before they have the possibility to strike. Many real-world systems suffer from such a limitation, which has been raised as a major criticism against security technologies, which often only serve as a means to improve the sense of security instead of actually reducing the vulnerability. In this paper we will use the vulnerability definition reported above to focus on the evaluation of the P_I parameter.

Even though, as mentioned above, usually the term vulnerability has a different meaning when used in the context of computer security, that does not mean that the method described in this paper cannot be employed in order to evaluate computer security risks. In fact, according to the computer dependability taxonomy, physical attacks belong to the class of human-made deliberate malicious threats; as such, they are relevant in the evaluation of overall system resiliency against physical attacks and/or hacker penetration/access to networked terminals [19]. In this regard, some surveys of stochastic modeling techniques which can be employed also for security evaluation are provided in [15,17].

We have chosen to use the Stochastic Petri Net (SPN) formalism in the TimeNET tool [20] since it has a virtually unlimited expressive power, so that the basic models (which are easy to understand even to non skilled modelers) may be customized and/or extended in order to account for behaviors of any type and complexity.

3 The Petri Net Vulnerability Model

Before starting the description of the vulnerability model, we would like to remark that the following assumptions hold: (1) the model does not account for possible deterrent effects (influencing the threat occurrence rate) nor for consequence mitigation effects (influencing the expected damage), which should be considered only in higher level risk models; (2) due to its stochastic nature and simple structure, the model only provides a rough approximation of the result (which is what is needed in practice); as a consequence, its parameters do not require a very high precision (which would be nearly impossible to achieve in practice); (3) since failure parameters (both for threat and protection mechanisms) can be accounted for by a simple multiplication (as explained in the previous section), it is not necessary to complicate the model in order to consider them; (4) the model describes a single threat scenario: in case multiple scenarios need to be modeled, more models should be evaluated and their results summed or combined somehow; in case concurrent scenarios need to be modeled, also possible limitations in the number of active responders should be modeled; (5) the basic model does not account for (a) multiple levels of threat progression and/or detection and (b) any intelligent/adaptive behaviors of attackers/defenders however, it may be extended to account for them if required; (6) as a constraint of the SPN formalism, the completion time of the basic activities is distributed as a negative exponential stochastic variable, whose mean should be chosen as the expected delay in “nominal” or “standard” operational conditions, or rather as an appropriate mean among the most common scenarios.

All the assumptions listed above are essential to simplify the model in order to make it easy to use in practical applications and still meaningful. Regarding the last assumption (no. 6), please note that it introduces the necessary non-determinism which allows to account for variations in the activity completion times, which is especially important since the system is a “human-in-the-loop” type (but also sensing times of technological devices are not always deterministic).

In such assumptions, the resulting vulnerability model is the one depicted in Figure 1 with its elements described in Table 1, where:

- L_T is the threat latency, that is the mean time for the threat to reach the target asset starting from the sensing point;
- L_S is the sensing latency, that is the mean time for the sensors to generate[1] and transmit to the control center a warning event or an alarm;
- L_A is the (remote) assessment latency, that is the mean time for the control room operator(s) to assess the situation and react accordingly;
- L_R is the response latency, that is the mean time for the response force to get to the scene in order to neutralize the threat.

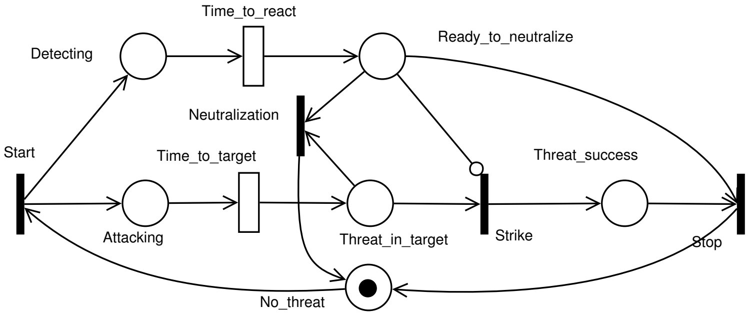

Fig. 1. The basic SPN model for vulnerability evaluation

To the best of our knowledge, all those latencies have not been explicitly taken into account in any generic models (like the one addressed in [12] and already discussed in the previous section); however, it is easy to understand that - together with the fail probabilities - they are essential parameters for the stochastic evaluation of physical vulnerability. The model works as follows. First of all, since vulnerability evaluation is conditional to the presence of a threat, the initial state in which the place *No_threat* has 1 token (enabling the *Start* transition) is evanescent (that is, the mean number of tokens in that place is

[1] Please note that not all sensors provide instantaneous outputs. For instance, smart-sensors like intelligent cameras or trace detectors include classifiers which require several seconds for the elaboration of input data.

Table 1. Description of the SPN Nodes

Node Name	Type	Description	Parameter Value
No_threat	Place	Initial status	Initial Marking = 1
Start	Immediate transition	Threat start trigger	Priority = 1, Weight = 1
Attacking	Place	Threat started the attack scenario	Initial Marking = 0
Detecting	Place	Sensor(s) started to detect	Initial Marking = 0
Time_to_target	Stochastic Transition	It models the threat delay to get to the target asset	Delay = L_T
Time_to_react	Stochastic Transition	It models the overall reaction delay including sensing, assessment and response latencies	Delay = $L_S + L_A + L_R$
Threat_in_target	Place	Threat has reached the target asset	Initial Marking = 0
Ready_to_neutralise	Place	Countermeasure(s) ready to neutralize the threat	Initial Marking = 0
Strike	Immediate transition	Threat strike trigger	Priority = 1, Weight = 1
Neutralization	Immediate transition	Threat neutralization	Priority = 1, Weight = 1
Threat_success	Place	The attack has been successful	Initial Marking = 0
Stop	Immediate transition	Attack scenario ends	Priority = 1, Weight = 1

0). The scenario always starts from the left, with 2 tokens generated by the *Start* transition, one in the *Attacking* place and one in the *Detecting* place: that models the situation in which the threat starts moving from the sensing point to the target asset. Moving to the right of the model, the two parallel stochastic transitions *Time_to_react* and *Time_to_target* are meant to model the concurrent actions of the attacker(s) and the defender(s):

- If the attacker arrives first to the target (1 token in *Threat_in_target* and no token in *Ready_to_neutralise*), then it has the possibility to strike (*Strike* transition is enabled, firing one token in *Threat_success*). Here the threat success probability is assumed to be 1, which is a sort of worst case which could be adjusted to get a more precise result in case threat failures are not negligible. Finally, the *Stop* transition resets the network to its initial state.
- In case the defender arrives first (1 token in *Ready_to_neutralise* and no token in *Threat_in_target*), then the *Strike* transition is disabled due to the inhibitor arc connecting it to the *Ready_to_neutralise* place, while the *Neutralization* transition is enabled, firing a token in *No_threat* and thus completing the

scenario. Here the detection probability is assumed to be 1 since, as mentioned above, detection failures can be simply evaluated by multiplying the result by P_S and P_A (see Section 2).

Figure 2 shows how to separately model the detection, assessment and response latencies. A similar approach could be used for threat latency modeling as well (including e.g. time to deploy, time to activate, etc.). That influences the kind of probability distribution of the overall latencies, since a sum of exponentially distributed stochastic variables features another type of distribution. Though in some cases it could make sense to go into these details, for the sake of simplicity we will not specify into our reference model any sub-activities (we will come back to discussing such an aspect later in this section).

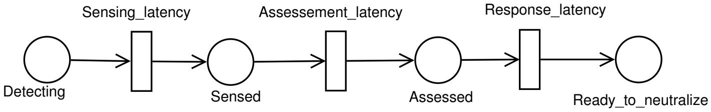

Fig. 2. PN modeling for distinct sensing, assessment and response latencies

Instead, explicit failure modeling complicates the network, increases the size of the reachability graph, and hence it can significantly slow-down model evaluation due to the state-space growth. As an example, we report in Figure 3 how to model the detection failure: two additional immediate transitions *Detect_success* and *Detect_failure* are enabled by tokens in place *Detecting*, with their weights representing the complementary probabilities, e.g. if $P_D = 0.9$ then:

$$weight(Detect\_success) = 0.9 \quad and \quad weight(Detect\_failure) = 0.1$$

It is easy to prove that the required reward expression to evaluate vulnerability (in the assumption $P_S = P_A = P_N = 1$) is as follows:

$$V = 1{-}P_E = 1{-}P\{\#Ready\_to\_neutralise = 1 \quad IF \quad \#Threat\_in\_target = 0\}$$

In fact, in order to neutralize the threat, the response must be ready before the threat has the possibility to strike. In such a condition, the inhibitor arc from the place *Ready_to_neutralise* prevents the transition *Strike* to fire, giving priority to the other enabled transition named *Neutralization*. Given the above assumptions, it is straightforward to understand that in case we need to know P_I, that can be simply obtained after model evaluation as $(1{-}V)$.

A basic validation of the reward expression may be performed by applying a boundary analysis to its parameters. As expected:

$$\begin{array}{l} Delay(Time\_to\_react) = 0 \quad AND \quad Delay(Time\_to\_target) > 0 \Rightarrow V = 0 \\ Delay(Time\_to\_react) > 0 \quad AND \quad Delay(Time\_to\_target) = 0 \Rightarrow V = 1 \\ Delay(Time\_to\_react) \gg Delay(Time\_to\_target) \Rightarrow V \backsimeq 1 \\ Delay(Time\_to\_react) \ll Delay(Time\_to\_target) \Rightarrow V \backsimeq 0 \\ Delay(Time\_to\_react) = Delay(Time\_to\_target) \Rightarrow V = 0.5 \end{array}$$

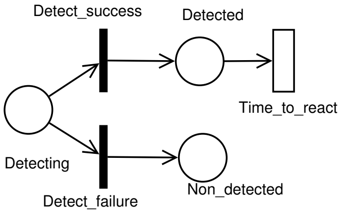

Fig. 3. Example SPN failure modeling

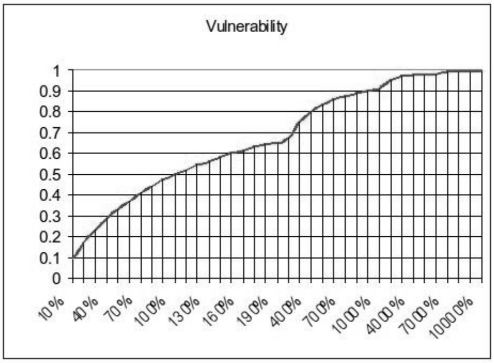

Fig. 4. Vulnerability as a function of the percentage ratio: delay(Time_to_react) / delay(Time_to_target)

In Figure 4 we report the results of a generic model evaluation. A single model execution (i.e. stationary analysis) in the TimeNET tool (Windows XP version) running on a typical office laptop (Intel Core 2 CPU 1.83 GHz with 2GB RAM) lasts only a few seconds. Since absolute timings are not relevant, we evaluated Vulnerability with respect to the following percentage ratio: delay(*Time_to_react*)/delay(*Time_to_target*). Neglecting approximations which alter curve regularity, the shape is parabolic tending asymptotically to 1 as the ratio tends to infinite, as expectable.

Regarding the probability distributions for the activities, though the exponential model is the most convenient in practice, it is important to highlight that (citation from [3], p. 165, 7.2): "The possibility of including timed transitions with general firing time distributions in GSPN (Generalized Stochastic Petri Nets) models is provided by the phase expansion that allows the behaviour of a random variable with general distribution to be approximated by means of a combination of negative exponential random variables with appropriate parameters. These distributions are called Phase-Type (PH) distributions. This means that an activity that lasts for a generally distributed random amount of time can be modelled by breaking it down into a set of stages (phases), all lasting for exponentially distributed periods of time."

As an example, consider the 2-phase attack vulnerability model depicted in Figure 5. In that model, the *Time_to_target* is split into two contributions: *Phase1* and *Phase2*. With such a model, the following result holds:

$$\begin{aligned}&Delay(Phase1) = Delay(Phase2) \quad AND\\&Delay(Phase1) + Delay(Phase2) = Delay(Time_t o_r eact) \Rightarrow V = 0.56\end{aligned}$$

In other words, the PH distribution assumption on the attacker side has increased (i.e. worsened) the vulnerability of about the 12%. In case of non homogenous bipartitions, the vulnerability increases a little bit less. More in detail:

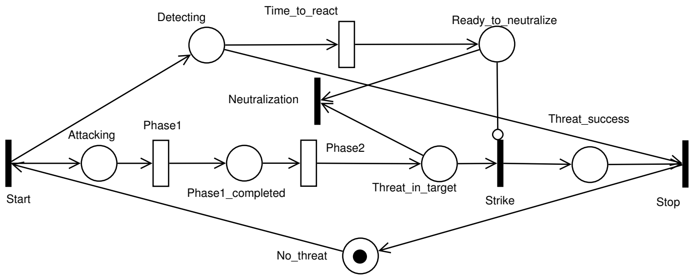

Fig. 5. SPN model of a 2-phase attack

$$Delay(Phase1) \neq Delay(Phase2) \quad AND$$
$$Delay(Phase1) + Delay(Phase2) = Delay(Time_to_react) \Rightarrow 0.5 < V < 0.56$$

Generally speaking, it could be shown that the result worsens as more attack phases are added (e.g. 4-phases with homogenous delays imply V = 0.66). However, excluding extreme cases, the impact on the results of considering more phases is generally limited and, nevertheless, it can be evaluated on a case by case basis by means of proper sensitivity analyses on the overall risk model.

4 Vulnerability Evaluation Examples

4.1 Case 1: Anti-theft and Intrusion Detection Systems

Valuable assets which are not continuously guarded are usually protected by means of active intrusion detection sensors which are part of surveillance systems featuring operators in remote control rooms or directly connected to the police stations. Let us assume we need to evaluate the vulnerability to thefts of a mission-critical server in a technical room which has an access door from the outside. Let us further assume that there is no active local siren, which is realistic in many industrial surveillance cases, to avoid disturbing people in case of false/nuisance alarms. Access control devices and magnetic contacts are used to detect unauthorized door openings. The magnetic contact has a very high reliability, lets suppose 98% (which usually gives the wrong perception that the overall protection system is very effective). The alarm is propagated to the control room in a few seconds, lets suppose 5s, and usually (in 95% of the cases) assessed in another bounce of seconds, say 15s, plus the time to call the responders and communicate the situation, say 30s. The responders are available and effective in 95% of the cases, needing about 3 minutes (180s) to get to the location. Once the door has been opened, the thief needs about 2 minutes (120 s) to disconnect the server and take it out.

Therefore: $P_S = 98\%, P_A = 95\%, P_N = 95\%, L_T = 120s, L_S = 5s, L_A = 15 + 30 = 45s, L_R = 180s$. P_I can be evaluated using the model in Figure 1 with: Delay(Time_to_react) = 230s, Delay(Time_to_target) = 120s. With the above parameters we obtain: $P_I = 34\%$. Therefore: $V = 1 - P_S \cdot P_A \cdot P_I \cdot P_N = 0.7$.

Therefore, despite of the reliability of the detection device, in more than 2 out of 3 cases, the theft will be successful in its intent. That suggests to install additional stand-off detection devices (e.g. motion detection cameras in the external area), quick response countermeasures (e.g. fog generators to blind thieves without damaging electronic devices), or even to guard the asset locally, depending on the other risk parameters (frequency of theft attempts, criticality of the asset, etc.).

4.2 Case 2: CBRNe Detection

The protection against Chemical Biological Radiological Nuclear and explosive (CBRNe) threats is often required in infrastructure security applications. In that case more than in others, the presence of detectors is not enough to decrease system vulnerability. In fact, the response strategy is essential, as we will formally demonstrate in the following. Consider a metro railway application in which detectors are installed before the turnstile barriers and no people/baggage screening is performed by dedicated security staff. Let us assume that (see also previous example): $P_S = P_A = P_N = 95\%$.

The average times to get to the target asset (e.g. a crowded area, like platform or train) and drop the substance/device is around 30s (a little bit more if the perpetrator needs to completely leave the station before the explosive device activates), and about the same time holds for the response latency (assuming local guards in the station). Using ad-hoc radio communications, the time needed to operators to assess the alarm and contact guards can be as low as 30s, but sensing times of CBRNe are usually higher (about 15s). Therefore: $L_T = 30s, L_S = 15s, L_A = 30s, L_R = 30s$.

Hence: Delay(Time_to_react) = 75s, Delay(Time_to_target) = 30s. Model evaluation provides the following result: $P_I = 29\%$. Therefore: $V = 0.75$, that is to say on average that in 3 out of 4 attacks the perpetrators will be successful. In those conditions, despite of the “perceived security”, the CBRNe detection system is almost useless. However, a simple countermeasure can make it much more effective: the automatic blocking of the entrance turnstile doors in case of detected alarms. If such a countermeasure is adopted, with the only drawback of slowing down the passenger flow in case of false-alarms, the response latency becomes a few seconds, say 3s, hence some latencies change as follows: $L_A = 2s$ (computer elaboration), $L_R = 3s$ (actuator command).

Also, since there is no human-in-the-loop, reliability parameters change as follows: $P_A = P_N = 99\%$. Thus: Delay(Time_to_react) $= L_S + L_A + L_R = 15 + 2 + 3 = 20s$, and the result becomes: $P_I = 60\%$. Therefore: $V = 0.44$. That is to say, in more than one half of the cases the CBRNe protection mechanism is able to neutralize the perpetrators.

5 Conclusions and Future Developments

In this paper we have presented a simple, generic and customizable model for the quantitative evaluation of physical vulnerability starting from parameters characterizing threat and countermeasure dynamics. The model is based on a certain class of SPNs allowing a very high expressive power; despite of that, it has an easy to understand basic structure which can be enriched in order to model more complex scenarios whenever required. In practical applications, however, just a rough approximation of the vulnerability is needed, since input parameters are not known with a very high precision. Therefore, the basic model can be more than enough to evaluate the effect of response latencies versus the time dynamics of the threat. That is required to populate risk models like the one presented in [10], which has mainly motivated the work presented in this paper. The effectiveness of any risk modeling approach is questionable under several points of view, including type (qualitative vs quantitative) and complexity (simple vs extensive). The approach is based on the three pillars which are well summarized by the following quotes:

1. "You can't control what you can not measure", Tom DeMarco
2. "Make things as simple as possible, but no simpler", Albert Einstein
3. "All models are wrong but some are useful", George E. P. Box

The first one suggests that quantitative models need to be adopted in order to govern the security risk. The second and third suggest to build models which are easy to manage and quick to evaluate as far as they provide us with usable results. Besides that, the work presented in this paper can be a starting point to build libraries of models for the modular/compositional development (e.g. by superposition of different nets) of more complex risk models, in which more threats and more protections are concurrently considered, together with the interrelationships of vulnerability with threat frequency and expected consequences. A further work is related to sthe definition of a model-driven automatic generatation of formal models from high level descriptions as successfully done in the reliability field [5].

Acknowledgements. The work of Stefano Marrone is funded by Campania region within the *AGRIGENET* project (PSR Campania 07/13).

References

1. Journal of physical security, `http://jps.anl.gov/`
2. A risk assessment methodology for physical security. White paper. Technical report, SANDIA National Laboratories (2008)
3. Ajmone Marsan, M., Balbo, G., Conte, G., Donatelli, S., Franceschinis, G.: Modelling with generalized stochastic petri nets. SIGMETRICS Perform. Eval. Rev. 26, 2 (1998)
4. Baker, G.H.: A vulnerability assessment methodology for critical infrastructure sites. In: DHS Symposium: Rand D Partnerships in Homeland Security (2005)
5. Bernardi, S., Flammini, F., Marrone, S., Merseguer, J., Papa, C., Vittorini, V.: Model-driven availability evaluation of railway control systems. In: Flammini, F., Bologna, S., Vittorini, V. (eds.) SAFECOMP 2011. LNCS, vol. 6894, pp. 15–28. Springer, Heidelberg (2011)

6. Broder, J.F.: Risk Analysis and the Security Survey. Butterworth-Heinemann (2006)
7. Casola, V., Mazzeo, A., Mazzocca, N., Vittorini, V.: A policy-based methodology for security evaluation: A security metric for public key infrastructures. Journal of Computer Security 15(2), 197–229 (2007)
8. Casola, V., Preziosi, R., Rak, M., Troiano, L.: A reference model for security level evaluation: Policy and fuzzy techniques. Journal of Universal Computer Science 11(1), 150–174 (2005)
9. Risk Steering Committee. DHS risk lexicon, `http://www.dhs.gov/xlibrary/assets/dhs_risk_lexicon.pdf`
10. Flammini, F., Gaglione, A., Mazzocca, N., Pragliola, C.: Quantitative security risk assessment and management for railway transportation infrastructures. In: Setola, R., Geretshuber, S. (eds.) CRITIS 2008. LNCS, vol. 5508, pp. 180–189. Springer, Heidelberg (2009)
11. Garcia, M.L.: Vulnerability Assessment of Physical Protection Systems. Butterworth-Heinemann (2005)
12. Hennessey, B., Wesson, R.B., Norman, B.: Security simulation for vulnerability assessment. IEEE Aerospace and Electronic Systems Magazine 22(9), 11–16 (2007)
13. Cox Jr., L.A.: Some limitations of risk = threat x vulnerability x consequence for risk analysis of terrorist attacks. Risk Analysis 28(6) (2008)
14. Lewis, T.G., Darken, R.P., Mackin, T., Dudenhoeffer, D.: Model-Based Risk Analysis for Critical Infrastructures. Critical Infrastructure Security - WIT Press (2011)
15. Nicol, D.M., Sanders, W.H., Trivedi, K.S.: Model-based evaluation: From dependability to security. IEEE Trans. Dependable Secur. Comput. 1, 48–65 (2004)
16. Rinaldi, S.M.: Modeling and simulating critical infrastructures and their interdependencies. In: Proceedings of the 37th HICSS 2004 - Track 2, vol. 2. IEEE Computer Society, Washington, DC (2004)
17. Sallhammar, K.: Stochastic Models for Combined Security and Dependability Evaluation. PhD thesis, Norwegian University of Science and Technology (2007)
18. Taylor, M.E., Kiekintveld, C., Western, C., Tambe, M.: A framework for evaluating deployed security systems: Is there a chink in your armor? Informatica 34 (2010), Special Issue on Quantitative Risk Analysis Techniques for Security Applications
19. Weingart, S.H.: Physical security devices for computer subsystems: A survey of attacks and defenses. In: Paar, C., Koç, Ç.K. (eds.) CHES 2000. LNCS, vol. 1965, pp. 302–317. Springer, Heidelberg (2000)
20. Zimmermann, A., Freiheit, J., German, R., Hommel, G.: Petri net modelling and performability evaluation with timeNET 3.0. In: Haverkort, B.R., Bohnenkamp, H.C., Smith, C.U. (eds.) TOOLS 2000. LNCS, vol. 1786, pp. 188–202. Springer, Heidelberg (2000)