Chapter 26

A SERVICE-ORIENTED APPROACH FOR ASSESSING INFRASTRUCTURE SECURITY

Marcelo Masera and Igor Nai Fovino

Abstract The pervasive use of information and communication technologies (ICT) in critical infrastructures requires security assessment approaches that consider the highly interconnected nature of ICT systems. Several approaches incorporate the relationships between structural and functional descriptions and security goals, and associate vulnerabilities with known attacks. However, these methodologies are typically based on the analysis of local problems. This paper proposes a methodology that systematically correlates and analyzes structural, functional and security information. The security assessment of critical infrastructure systems is enhanced using a service-oriented perspective, which focuses the analysis on the concept of service, linking the interactions among services – modeled as service chains – with vulnerabilities, threats and attacks.

Keywords: Security assessment, vulnerabilities, threats, attacks, services, system-of-systems

1. Introduction

Security threats are a serious problem in this computer-based era. Any system that makes use of information and communication technologies (ICT) is prone to failures and vulnerabilities that can be exploited by malicious software and agents. Critical infrastructure components, especially industrial installations, have key features that differentiate them from more conventional ICT systems. In particular, industrial facilities combine traditional information systems (e.g., databases) with real-time elements that implement process control functions. Recently, these hybrid infrastructures have begun to be connected to internal and external communication networks, which raises serious security concerns.

Risk assessment and management in critical infrastructures is a relatively new discipline. Most efforts concentrate on corporate information systems. But industrial systems have certain unique features – the co-existence of heterogeneous environments (e.g., real-time and desktop applications) and constraints deriving from physical phenomena (e.g., power stability) and business objectives (e.g., productivity and performance). These factors determine how IT systems in industrial environments can be handled. For example, it may not be possible to stop industrial operations in order to install security patches.

An effective security assessment and management methodology must take into account information about system characteristics (vulnerabilities, assets, security policies), threats (intentions, resources, capabilities), and potential attack mechanisms and countermeasures. In addition, it is necessary to consider the interactions of malicious actions with accidental failures and human error.

The evolution of systems into infrastructures adds a further level of complexity. Infrastructures are systems-of-systems, greatly interconnected (mainly due to the pervasive use of ICT) and characterized by interdependencies that induce system-wide propagations of negative effects. Several approaches have been proposed for analyzing critical infrastructure systems. These approaches generally focus on linking structural and functional descriptions to security goals, and associating vulnerabilities with known attacks.

This paper describes a novel approach, which builds on the embryonic security assessment methodology of Masera and Nai [14]. The approach, which involves the systematic correlation and analysis of security-relevant information, reveals dependencies within infrastructure systems and relationships between different "information sets" that describe a system-of-systems from the security standpoint. The security assessment of critical infrastructure systems is enhanced using a service-oriented perspective, which links the interactions among services with vulnerabilities, threats and attacks.

2. The State of The Art

The scientific literature has very limited work tailored to the comprehensive assessment of industrial ICT security. However, there is relevant work in the field of ICT security, system modeling and system safety. This section provides an overview of the principal approaches related to ICT security.

Safety and risk have traditionally been the focus of assessments of industrial systems. Only recently have security issues begun to be considered. Keeney, *et al.* [12] have conducted a study on computer system sabotage in critical infrastructures. Stoneburner, Goguen and Feringa [24] have developed a nine-step procedure for risk assessment of information systems. Swiderski and Snyder [25] introduced the concept of threat modeling, and a structured approach for identifying, evaluating and mitigating risks to system security. A similar approach has been proposed for web application environments [4]. Several general purpose tools have been developed, including Microsoft's Security Assessment Tool [20] and Citicus [6]. The first tool supports a traditional "check list" assessment process, which goes through a series of question-and-answer

sessions, guiding the analysis through an iterative process and producing a set of recommendations and best practices. The process is quick and easy, but can only provide rough results. The second tool, Citicus, is based on the concept of perceived information risk, categorizing risk according to customized criteria.

The OCTAVE approach [1] introduced in late 1990s is an exhaustive methodology for information systems, but it has not been used for industrial applications. The CORAS methodology [7] was developed in the early 2000s to perform model-based risk analyses of security-critical systems – but the methodology has been applied to e-government and e-commerce systems, not to industrial control systems or, more generally, complex heterogeneous systems.

In our opinion, a security assessment is inadequate if it does not rely on a comprehensive description of the system of interest. The description should cover all relevant perspectives: policies and operations, structure and function, physical links and information flows, among others.

Infrastructure modeling has used mainly for design and operational purposes, but the analysis of security requires additional considerations. Alberts and Dorofee [1] have proposed a risk assessment methodology based on a system description. However, the description is relatively informal; more importantly, it cannot deal with complex systems. den Braber, *et al.* [7] have also presented a risk assessment approach that is partially based on a system description. The approach attempts to capture the concept of an adverse environment by introducing the concept of a "threat scenario." This, of course, represents an advance in system representation that could be adapted to modeling interacting systems (although this was not the intention of the authors).

Masera and Nai Fovino [15–17] have presented an approach based on the concept of a "system-of-systems," which preserves the operational and managerial independence of the individual components while capturing the relationship between components, services and subsystems. The present work adopts this approach as a starting point.

A security assessment has limited effectiveness unless it considers attack scenarios. An early approach to incorporating attack information was the creation of vulnerability databases (e.g., Bugtraq [22]). However, these databases merely describe vulnerabilities, not how they can be exploited in a successful attack. Graph-based attack models [23], which include Petri net models and attack trees models, are popular approaches for modeling attacks. The attack net model introduced by McDermott [19] is an exemplar; in this model, the places of a Petri net represent the attack steps and the transitions capture the actions performed by an attacker. Attack trees proposed by Schneier [21] use expansion trees to show the different attack lines that could affect a system, describing their steps and their interrelationships. The attack tree approach has been extended by Masera and Nai [18] who introduced the concept of an attack projection. This paper adopts this method of representing attacks as a reference.

3. Preliminary Definitions

A risk assessment of industrial ICT infrastructures requires two types of characterizations: security definitions and system definitions. These characterizations and related concepts are described below.

3.1 Security Description

A security description involves security-related concepts such as "threat," "vulnerability," "attack" and "risk." A "threat" is defined in [11] and in the Internet RFC glossary of terms as a potential for violation of security, which exists when there is a circumstance, capability, action or event that could breach security and cause harm. A "vulnerability" a weakness in the architectural design or implementation of an application or a service [2, 5]. As a direct consequence, an "attack" is the entire process implemented by a threat agent to exploit a system by taking advantage of one or more vulnerabilities. Finally, "risk," according to the ISO/IEC 17799:2000 [10], is the probability that a damaging incident is happening (i.e., when a threat is actualized by exploiting a vulnerability) times the potential damage.

3.2 System Description

A system description involves concepts required for system modeling such as "system," "subsystem," "component," "service," "dependency," "information flow" and "asset." A "system" is a collection of entities that collaborate to realize a set of objectives [9]. The same definition holds for a "subsystem" using an inheritance principle. Masera and Nai [16, 17] define a "component" as an atomic object able to fulfill actively- or passively-defined tasks. "Services" are tasks performed by components or subsystems (such services can be "on request").

The same authors define the concept of a "dependency" – a system object A depends on a system object B if B is required by A to accomplish its mission. "Information flow" is a set of point-to-point relationships describing the entire lifecycle of an information item [16]. Finally, an "asset" is any element with value to the relevant stakeholders of the system of interest [14].

As the loss (or impairment) of an asset will negatively affect its value, the objective of security management is to protect assets. Assets are security-relevant entities of a system because (i) their destruction, inability to perform the intended functions, or disclosure to unauthorized agents might cause a detrimental effect, (ii) malicious threats agents might have an interest in targeting them, and (iii) they can be exposed to malicious actions by component vulnerabilities and faults, or by errors on the part of system operators.

In general, an asset could take two main forms: (i) an internal set of components whose loss will cause detriment to the owner/operator of the system, or (ii) an external service supplied to users of the system. Examples of internal assets are a costly component or a set of sensitive data. A control function is

an example of external asset. IT-based systems are distinguished from physical systems because of the presence of "information assets" [16].

4. Service-Oriented Paradigm

Beyond the basic descriptions of components, vulnerabilities, attacks, etc., there is a need for a paradigm to capture the interconnections of the different elements that need to be analyzed. It is necessary to identify and examine, for example, the potential effect that a component vulnerability might have on the entire system (e.g., on the business objectives of an industrial facility). As the elements to be considered in a "system-of-systems" situation are manifold, a key challenge is to avoid an excess of data that would hamper the analysis by obfuscating the significant aspects.

To deal with this issue, we make use of the concept of a "service" [16, 17]. Viewed in this light, objects in a system are producers/consumers of services. This concept permits the creation of detailed descriptions of the relationships and dependence mechanisms, which are at the core of the security issues for infrastructure systems.

DEFINITION 1 *A service s is a tuple $< name, description, ID, sdr >$ where name identifies the service, description is a brief functional description of the service, ID is the identifier of the producer of the service, and sdr represents information about the dependencies of the service (i.e., in order to fulfill its duty a service x needs the direct support of the services k, z, m).*

The converse of service is "disservice," the lack of provision of the service. The concept of disservice is used in the field of dependability, but its importance has not yet been recognized in the field of ICT security assessment. Upon applying a service-oriented description, the system assumes a "network" aspect. In particular, components and subsystems are directly or indirectly interconnected by what we call "service chains," where all the components/subsystems are in some way necessary for the proper provision of the intended services.

As far as assets are concerned, it is possible to describe them as a mix of internal and external services – more than just a set of hardware and software elements. This is a more operative approach that permits linking the system and the security descriptions of a system. Information, components and subsystems provide/require services to/form other information, components and subsystems. Certain services coalesce through service chains in elements of specific value to the stakeholders of a system; these are called "system assets."

To clarify the concept, consider a system, which provides an "information service" (IS) to external customers (e.g., a power plant might supply data about the energy it produced). This is performed by a "web application service" (WAS) at the subsystem level. The data forwarded to the customers are stored in a database, which provides a "storage service" (SS). The data are the results of computations based on raw data retrieved by remote field sensors, which provide a "field monitoring service" (FMS). The high-level service IS is linked to WAS in a functional way. Moreover, information flow links exist between

WAS and SS, and between SS and FMS. In other words, there is an indirect service link between IS, WAS and FMS. This set of links, which constitutes a service chain, could show how a failure of FMS could affect IS.

DEFINITION 2 *A system S_n is defined by $\{s_1, \ldots, s_n, desc\}$, where s_1, ..., s_n are services provided by S_n, and desc is the general description of the system. The concepts of subsystem and component can be defined in the same way without as loss of generality.*

DEFINITION 3 *Let SoS be a system-of-systems defined by $\{S_a, S_b, \ldots, S_n\}$ (i.e., set of systems, subsystems and components in SoS), and let Serv be the set of services of SoS. A service dependency record sdr is a tuple $< s, s_{id}, inset, outset, lf >$, where s is a service, S_{id} is the identifier of the system, subsystem or component S_a in SoS (which "produces" the service); inset $= \{< d, w > | d \in Serv, w \in \aleph\}$ represents the collection of services directly contributing to the realization of the service with an associated relevance w; outset is the list of services to which the service s directly provides a contribution; lf is a second-order logic expression that describes (when combined with the weights w of inset) the manner and relevance to which the contributing services are logically linked. For example, the provision of a service A may require the combination of services B, C and D according to the logical expression $[(w_b.B \wedge w_c.C) \vee w_d.D)]$.*

Applying this definition to a system-of-systems, SoS, it is possible to reconstruct all the links between services. We call this the "service chain" of the object under analysis. This is an oriented graph describing the direct and indirect links between all the services provided by and within SoS. From a security perspective, service chains help identify all the dependencies that play a role in a security event (e.g., propagation of failures, cascading effects, etc.). As defined in Masera [13], a "security dependency" exists when there is a relationship between two systems A and B such that an internal fault in B can be propagated through a chain of faults, errors and failures to system A. Drawing from [3], we refer to such chains as "pathological chains."

Pathological chains can be caused by (i) accidental events due to internal faults or human errors, or (ii) malicious attacks. Since every component in a system description has an associated set of known vulnerabilities (each vulnerability affects a target component with a certain plausibility y), we can enrich the description of the pathological chain by adding information related to the vulnerabilities. In this way, the appraisal of service chains considering dependencies and vulnerabilities result in what we call "vulnerability chains" (see Figure 1). The notion of a vulnerability chain offers three main advantages:

- It allows the identification of low-level vulnerabilities (associated with low-level components) that can have an effect on high-level services (typically services provided by the system to the external world).
- It permits the capture of the potential non-negligible side effects of an identified vulnerability.

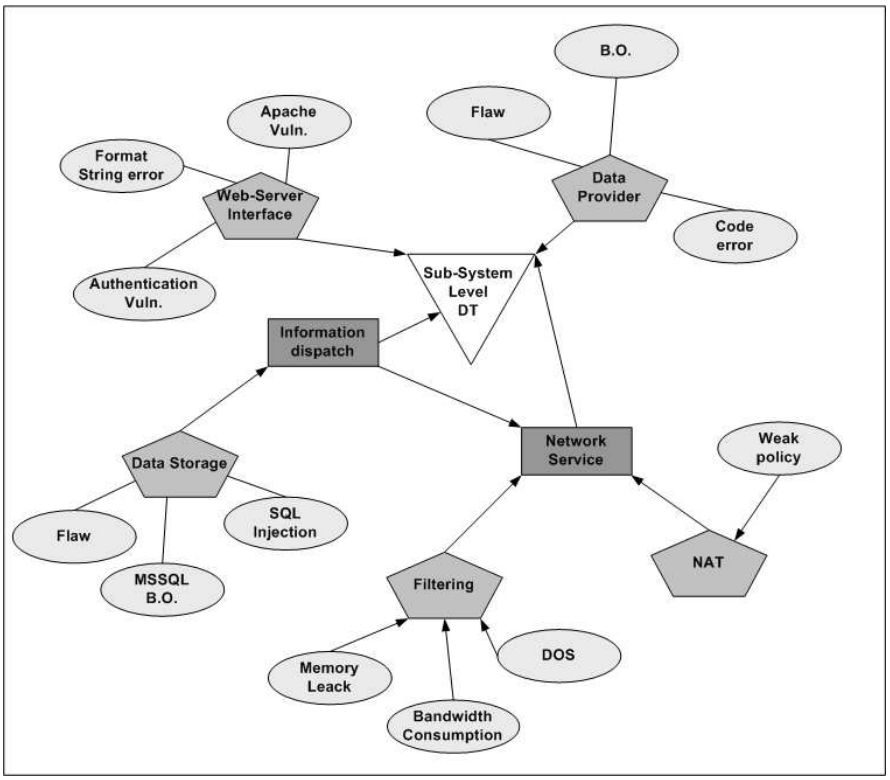

Figure 1. Vulnerability chains.

- It constitutes the glue that links system description knowledge (components, services, assets, etc.) with security knowledge (vulnerabilities, attacks and threats)

4.1 Service-Oriented Vulnerability Analysis

By adopting the description paradigm presented above, it is possible to identify which low-level vulnerabilities (i.e., those affecting low-level components) can have a negative security effect on the assets. The approach involves the following steps:

- The dependencies are computed for each asset.

- For each element in the asset, the services it provides are retrieved. Note that an asset may be composed of physical and logical subsystems, services and components.

```
Input: Set of System Assets (SA)
Output: Set SA enriched with information about associated vulnerabilities

Main
{
Select Asset from SA
For each element i in Asset do
J=i
If i is a service then J=service.ID
Inspect (J)
}
Function Inspect (J)
{
If check_cycle(J)=false
then
if J.vulnset ≠ φ then Asset.vulnset=Asset.vulnset+J.vulnset
for each service provided by J
{
sdr=retrieve s.sdr
for each service in s.sdr.inset Inspect(s.sdr.inset.ID)
}}
```

Figure 2. Asset-vulnerability association pseudo code.

- By exploring the service relationships associated with each service (while taking care of possible cyclic dependencies), the low-level components that contribute to the service in some way are identified.

- The vulnerabilities that potentially affect the low-level components are associated with the asset.

Applying this procedure, we can identify which vulnerability associated a component may have an impact on the asset and to what degree. This knowledge facilitates the analysis of the effects of threats and attacks, decisions about the effectiveness of current policies, the benchmarking of potential solutions and the running of security scenarios.

In our approach, vulnerabilities are classified according to their estimated relevance following an identification of potential threats. Figure 2 presents the pseudo code for this procedure.

4.2 Service-Oriented Threat Assessment

Determining the vulnerability of a system is not enough. It is also necessary to analyze the threats that might exploit the identified vulnerabilities. In this section, we expand the security analysis process described by Masera and Nai [14] into a "service-oriented threat assessment process," whose objective is

to determine the vulnerable assets that are exposed to different types of threats taking into consideration the vulnerability chains.

When this kind of analysis is applied to relatively small systems, it is usually solved by assigning known threats to possible target assets on the basis of some (possibly not well-documented) hypotheses made by the analyst. But such an approach, in addition to not being systematic, says nothing about the proportion of security situations being considered from the total number of possible negative events.

Threat analysis can be improved by using information derived from vulnerability analysis. A threat is relevant only if there is the real possibility that it can be realized. In particular, ICT threats have to be correlated with the assets that – from the vulnerability viewpoint – can be affected by them. Service-oriented threat analysis proceeds in a similar manner to service-oriented vulnerability analysis that was presented in the previous section:

- A subset of assets affected by significant vulnerabilities (identified by service-oriented vulnerability analysis) is selected from the set of assets.
- The subsystem services involved for each element of the subset of vulnerable assets are identified.
- The hypothesized threats (derived from some parallel identification of plausible threats applicable to the type of system under analysis) are instantiated and correlated with the previously-identified subsystem services (an effective way to conduct this analysis is to assign threats only to subsystem services [14]).
- The threats whose effects can be propagated to the vulnerable assets are verified by exploring the service dependencies and relationships.

Using this procedure, it is possible to obtain a focused, motivated and documented set of "exposed" subsystem services. Also, it is possible to demonstrate how threats can affect services – and therefore assets – directly by targeting components, or indirectly by effects on correlated assets that propagate through service dependency chains.

4.3 Service-Oriented Attack Analysis

Attack analysis involves identifying the potential attacks that can be successfully developed by the previously identified threats. Validating the possibility that an attack can take place against a target system is not a simple task, especially in the case of large or complex systems.

Attack trees are a popular means for representing the steps and the conditions required to perpetrate offensive actions against vulnerable assets. However, attack trees are usually too abstract because they refer to the types of components. To validate attacks, attack trees have to be instantiated for the specific elements and characteristics of the system under analysis. This requires the consideration of all the interconnections and relative interdependencies, and

potential alternative paths. An attack validation conducted without this knowledge will produce a large number of false positives (i.e., valid attacks that are not exploitable), or it will discard potential attacks derived from vulnerabilities that are coupled in non-obvious ways.

The information derived from service-oriented analysis is useful for mitigation efforts as it helps focus the examination of attacks on vulnerable disservice chains. In other words, by applying this knowledge to disservice chains, it is possible to identify whether there is some connection between vulnerable components that can be attacked by one of the verified threats. All the other chains may be considered safe with respect to potential attacks.

Attack validation using a service-oriented perspective proceeds as follows:

- The attacks that can be associated to the verified threats are identified and presented as hypotheses to be validated.
- The associated subsystems and all the respective relationships are identified for each verified threat.
- The attack trees associated with each verified threat are validated by applying existing information about disservice chains, dependency relationships and conditional assertions (i.e., assertions describing some additional conditions needed to realize the attack).
- The potential impact on the assets due to validated attacks are computed by considering all direct and indirect effects on the affected subsystems.

The procedure described above identifies the set of realizable attacks, while minimizing the number of false positives.

5. Preliminary Results

To test the performance, quality and benefits of the service-oriented approach, we developed a software tool named InSAW (Industrial Security Assessment Workbench). InSAW implements the analysis steps presented in this paper (system description, vulnerability assessment, threat assessment and attack assessment). In addition, it implements an additional phase for overall risk assessment. InSAW uses a MSSQL relational database with an intermediate object-oriented layer based on Hibernate and a set of modular analysis engines developed using Microsoft .Net technology.

The testing phase involved the following steps:

- Selection of a set of industrial case studies (remote control of primary substations, control of power plants), and performance of security assessments using the methodology with desktop tools.
- Application of InSAW to the automatic determination of service/disservice chains and related vulnerability, threat and attack analyses (it is, of course, necessary to input a description of the target system).

- Comparison of the results obtained by manual and automatic analyses.

Although the tests are preliminary in nature, the results (see, e.g., [8]) are promising. They enable us to make the following observations:

- A service-oriented approach provides an analyst with a better, more comprehensive understanding of the relations, connections and dependencies between system components.
- Service-oriented vulnerability and threat assessments benefit from the analysis of service dependencies as it is possible to identify side-effect connections between vulnerabilities and assets that are not readily observable by manual means.
- The service-oriented approach greatly augments the precision of attack validation. This is because each attack step can be related to all the aspects that might influence it.

6. Conclusions

The service-oriented methodology described in this paper is a novel approach for assessing the security of critical infrastructure systems. The methodology has as its core the concept of service and the description of service dependencies, which greatly facilitate vulnerability analysis, threat assessment and attack analysis and verification. Automating security assessment procedures is undoubtedly of value to analysts, mainly because of the dynamic nature of security events and the need to consider new information about vulnerabilities, threats, exploits and countermeasures. Our future work will concentrate on conducting extensive tests of the methodology and its implementation. In addition, we will attempt to link the approach with other security-relevant activities such as early warning, diagnostics and information sharing.

References

[1] C. Alberts and A. Dorofee, *Managing Information Security Risks: The OCTAVE (SM) Approach*, Addison-Wesley, Boston, Massachusetts, 2002.

[2] O. Alhazmi, Y. Malaiya and I. Ray, Security vulnerabilities in software systems: A quantitative perspective, in *Data and Applications Security XIX (LNCS 3654)*, S. Jajodia and D. Wijesekera (Eds.), Springer, Berlin-Heidelberg, Germany, pp. 281–294, 2005.

[3] 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*, vol. 1(1), pp 11–33, 2004.

[4] E. Bertino, D. Bruschi, S. Franzoni, I. Nai Fovino and S. Valtolina, Threat modeling for SQL servers, *Proceedings of the Eighth IFIP TC-6 TC-11 Conference on Communications and Multimedia Security*, pp. 189–201, 2004.

[5] M. Bishop, *Computer Security: Art and Science*, Addison-Wesley, Boston, Massachusetts, 2003.

[6] Citicus, Citicus ONE (www.citicus.com).

[7] F. den Braber, T. Dimitrakos, B. Gran, M. Lund, K. Stølen and J. Aagedal, The CORAS methodology: Model-based risk management using UML and UP, in *UML and the Unified Process*, L. Favre (Ed.), IGI Publishing, Hershey, Pennsylvania, pp. 332–357, 2003.

[8] G. Dondossola, J. Szanto, M. Masera and I. Nai Fovino, Evaluation of the effects of intentional threats to power substation control systems, *Proceedings of the International Workshop on Complex Network and Infrastructure Protection*, 2006.

[9] Institute of Electrical and Electronics Engineers, IEEE Standard Glossary of Software Engineering Terminology (IEEE Standard 610.12-1990), Piscataway, New Jersey, 1990.

[10] International Organization for Standardization, Code of Practice for Information Security Management (ISO/IEC 17799:2000), Geneva, Switzerland, 2000.

[11] A. Jones and D. Ashenden, *Risk Management for Computer Security: Protecting Your Network and Information Assets*, Elsevier Butterworth-Heinemann, Oxford, United Kingdom, 2005.

[12] M. Keeney, E. Kowalski, D. Cappelli, A. Moore, T. Shimeall and S. Rogers, Insider Threat Study: Computer System Sabotage in Critical Infrastructure Sectors, Technical Report, U.S. Secret Service and CERT Coordination Center, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania, 2005.

[13] M. Masera, Interdependencies and security assessment: A dependability view, *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, Taipei, 2006.

[14] M. Masera and I. Nai Fovino, A framework for the security assessment of remote control applications of critical infrastructures, *Proceedings of the Twenty-Ninth ESReDA Seminar*, 2005.

[15] M. Masera and I. Nai Fovino, Emergent disservices in interdependent systems and systems-of-systems, *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, 2006.

[16] M. Masera and I. Nai Fovino, Modeling information assets for security risk assessment in industrial settings, *Proceedings of the Fifteenth EICAR Annual Conference*, 2006.

[17] M. Masera and I. Nai Fovino, Models for security assessment and management, *Proceedings of the International Workshop on Complex Network and Infrastructure Protection*, 2006.

[18] M. Masera and I. Nai Fovino, Through the description of attacks: A multidimensional view, *Proceedings of the Twenty-Fifth International Conference on Computer Safety, Reliability and Security*, pp. 15–28, 2006.

[19] J. McDermott, Attack net penetration testing, *Proceedings of the New Security Paradigms Workshop*, pp. 15–22, 2002.

[20] Microsoft Corporation, Microsoft Security Assessment Tool (www.security guidance.com).

[21] B. Schneier, Attack trees: Modeling security threats, *Dr. Dobb's Journal*, December 1999.

[22] SecurityFocus, Bugtraq vulnerability database (securityfocus.com).

[23] J. Steffan and M. Schumacher, Collaborative attack modeling, *Proceedings of the ACM Symposium on Applied Computing*, pp. 253–259, 2002.

[24] G. Stoneburner, A. Goguen and A. Feringa, Risk Management Guide for Information Technology Systems, Special Publication 800-30, National Institute of Standards and Technology, U.S. Department of Commerce, Gaithersburg, Maryland, 2002.

[25] F. Swiderski and W. Snyder, *Threat Modeling*, Microsoft Press, Redmond, Washington, 2004.