Key issues in the development of risk analysis methodologies and tools

G. Carducci, P. Migliaccio & E. Montolivo
Securteam, Italy

Abstract

This paper describes the real-world experience of the authors that in the past three years have been involved in a project aiming at the definition of a risk analysis methodology and at the development of an automated risk management tool (named *Defender Manager*©) which is suitable for information security applications.

Keywords: risk analysis, risk management, information security, threats, vulnerabilities.

1 Introduction

Risk analysis is the process of estimating potential losses that may result from the occurrence of certain threats. It forms the basis for establishing a cost-effective risk management program suitable to reduce these losses to an acceptable level. Despite risk analysis usually being considered the only consistent approach to the selection of the most appropriate safeguards, a well defined and largely accepted risk analysis methodology suitable for information security applications (including the information and communication technology scenarios) is still lacking and even taxonomy in this sector is often a little bit confusing. This paper describes the real-world experience of the authors that in the past three years have been involved in a project aiming at the definition of a risk analysis methodology and at the development of a proprietary automated risk management tool (named *Defender Manager*©) suitable for information security applications. Key issues in the definition of risk analysis methodologies, as they arose during the project, are analysed. These include: defining a taxonomy for threats, attacks, vulnerabilities and risk; defining a metric for rating vulnerabilities and safeguards; building a database of threats, attacks and security measures; modelling security perimeters. A brief description of the tool Defender Manager© concludes the paper.

2 Security taxonomy and model

Defining or embracing a sound taxonomy is the first challenge to overcome for devising a consistent risk analysis methodology and model suitable for information security applications. In fact, even terms used in the field of information security are ambiguous and very often defined (or interpreted) in different ways in the relevant standards and scientific literature.

In our work, we assumed that assuring information security means to protect three information (and information systems) qualities:

- *confidentiality* (the quality of not being disclosed to unauthorised persons)
- *integrity* (the quality of not being altered or destructed by unauthorised persons or accidental events)
- *availability* (the quality of being timely and reliably accessible).

These three qualities (in the following referred to as CIA qualities) may overlap and even conflict. For example, providing strong confidentiality may adversely affect availability.

The definitions that follow play an important role in our model. They are somehow derived from similar definitions proposed in [1, 2, 3, 4], adapted to the context this paper deals with.

A *system* is a collection of interconnected entities (the *components* of the system), e.g., people, machines, infrastructures (basic facilities, services, and installations needed for the functioning of a community or society), that act and interact together towards accomplishment of some logical end.

A *security perimeter* is the collection of *systems* the risk analysis refers to.

An *information asset* is a piece of information (as a whole) whose value with reference to CIA makes sense to be assessed for an organisation inside a given *security perimeter*.

A *threat* is a potential event able to compromise *information assets* integrity, availability, or confidentiality. Both *natural threats* (fire, flood, Murphy laws, etc.) and *human threats* are included in this definition.

A *vulnerability* is the manifestation of the inherent states of a system (e.g., physical, technical, organisational, cultural) that can be exploited by an adversary (or by an adverse circumstance) to compromise information assets CIA qualities. Note that this definition includes technical and non-technical aspects such as people ignorance, lack of documented security procedures, etc.

An *attack* is the attempt by an adversary (or by an adverse circumstance) to exploit a system *vulnerability*. An *attack* is a way for a *threat* to materialise. Usually for each threat many *attacks* exist. A given attack can result in a loss of just one of the CIA qualities of an information asset or in more than one (in arbitrary combinations). We name *CIA qualities of the attack* the list of the information asset CIA qualities the attack impact on. Depending on attack peculiarities and on the safeguards implemented, this loss can be equal to or less than 100% of the value of the attacked information asset security quality. When the value of the information asset is assessed using a few level qualitative scale, it is usually appropriate to conservatively assume that a successful attack always results in a 100% loss.

The *attack potential* [4] is the potential for success of an attack, should an attack be launched. The *attack potential* can be expressed in terms of an attacker's motivation, expertise and resources (e.g. time available to identify and exploit a vulnerability, specialist technical expertise, knowledge of the attacked system design and operation, opportunity to access the attacked system, equipment required for launching the attack). *Vulnerabilities* can be rated on the basis of the *attack potential* required to an attacker to exploit them (the higher the *attack potential* required, the lower the corresponding *vulnerability level*).

A *security measure* is any hardware or physical device, software function, procedure, every form of human surveillance, etc. suitable to eliminate or reduce system vulnerabilities. *Security measure strength* (i.e. their ability to withstand an attack) can be rated on the basis of the *attack potential* required to launch a successful attack notwithstanding that security measures are in place (the higher the *attack potential* required, the higher the corresponding *security measures strength*).

The relationships among the above entities are graphically represented in Figure 1 that also provides a high level view of the security model used in the tool Defender Manager©.

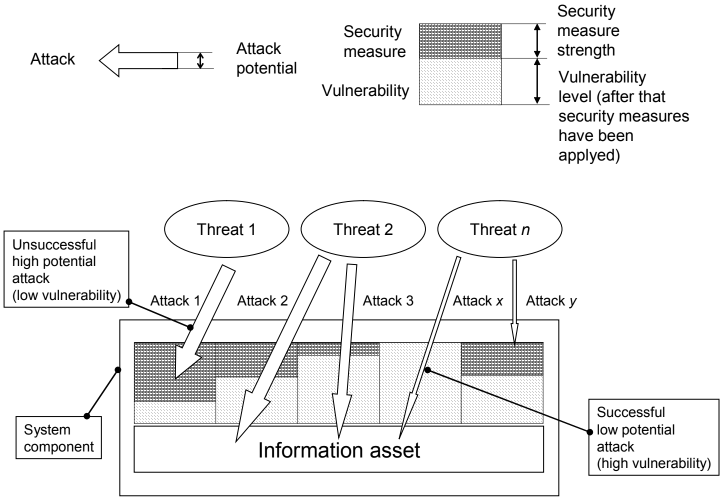
Attack
Attack potential
Security measure
Vulnerability
Security measure strength
Vulnerability level (after that security measures have been applyed)
Threat 1
Threat 2
Threat *n*
Unsuccessful high potential attack (low vulnerability)
Attack 1
Attack 2
Attack 3
Attack *x*
Attack *y*
System component
Information asset
Successful low potential attack (high vulnerability)

Figure 1: A high level view of the security model used in the tool Defender Manager©.

We named the security model represented in Figure 1 the *brick and straw* model after the famous traditional tale *The three little pigs*. The information asset processed in a system (or in a system component) is protected from the various attacks by means of a wall of a fixed thickness. This wall can be completely built of straw, partially of straw and partially of bricks, or completely of bricks. The overall thickness of the wall represents the maximum (theoretical) security measure strength that makes sense to think of. Bricks represent the security measures in place (with the thickness of the brick wall representing the security measure strength) while straw represents all the room that in principle could have been filled of bricks (providing more effective but more costly protection) but it was not (the thickness of the wall made of straw represents the residual vulnerability level).

Based on the above definitions, a threat scenario $S_{i,j}$ is defined as the 7-plet

$$\{\mathrm{Sp}, \mathrm{Sy}, \mathrm{Co}, \mathrm{T}_i, \mathrm{A}_{i,j}, \mathrm{AP}(\mathrm{A}_{i,j}), \mathrm{LV}(\mathrm{Co}, \mathrm{A}_{i,j})\} \tag{1}$$

where Sp, Sy and Co are the identifiers of a security perimeter, a system belonging to that perimeter and a component of that system respectively; T_i is a threat; $\mathrm{A}_{i,j}$ is one of the attacks implementing that threat; $\mathrm{AP}(\mathrm{A}_{i,j})$ is the expected attack potential of $\mathrm{A}_{i,j}$ and $\mathrm{LV}(\mathrm{Co}, \mathrm{A}_{i,j})$ is the vulnerability level of the component Co with respect to $\mathrm{A}_{i,j}$. Only those scenarios where $\mathrm{AP}(\mathrm{A}_{i,j})$ is high enough to allow the vulnerability to be exploited result in non zero consequences.

3 Risk model

Kaplan and Garrick [5] proposed that risk is a multidimensional entity depending on the answers to three questions: a) What could go wrong? b) How likely is it to go wrong? c) Given that it happens, what are the consequences? In line with this view and taking into account (1), we found appropriate to define risk associate to the system component Co of the system Sy in the perimeter Sp as a the set of triplets

$$\{\mathbf{R}_{i,j}(RC_{i,j}, RI_{i,j}, RA_{i,j})\} = \{S_{i,j}, f_{i,j}, \mathbf{X}_{i,j}\} \quad i=1, 2, \ldots j=1,2, \ldots \tag{2}$$

where

- $\mathbf{R}_{i,j}$ is a vector whose components represent risks associated with the scenario $S_{i,j}$ with respect to CIA qualities losses;
- $S_{i,j}$ is a threat scenario identification or description (the description of what could go wrong);
- $f_{i,j}$ is the expected frequency of that scenario (how likely the scenario is, e.g. how many times per century, year, day, etc. the scenario is expected to happen);
- $\mathbf{X}_{i,j}$ is the measure of damage with respect to CIA losses resulting from the occurrence of that scenario (the consequences of the scenario).

We named the set of 2-plets $\{f_{i,j}, \mathbf{X}_{i,j}\}$ *intrinsic risk.* After having statistically modelled the attacker population in terms of the attack potential of the attacks it generates, the intrinsic risk can be associated to the most appropriate security measures and security measure strength that is to those security measures that according to best practice are suitable to reduce risk to an acceptable level.

4 Building a database of threats, attacks and security measures

Our automated risk analysis and management tool, based on the security model described in the previous section, requires for its operation a database of threats, attacks and security measures adequately associated to the applicable system components and the CIA qualities they are relevant to. Designing such a database and populating it proved to be a very challenging task. Failing in this task causes meaningless risk analysis results and wrong risk management decision. Our experience has indicated that satisfactory database should fulfil the following requirements:

- all entities should be described at the same level of abstraction;
- threats should be mutually exclusive;
- attacks should be mutually exclusive;
- security measures should be mutually exclusive and non ambiguously linkable to the threats and the attacks they counter;
- threats and attacks should be easily distinguished (regardless who is asked to decide).

Meeting these requirements with reference to the database tables relevant to threats and attacks was the hardest part of the job. People asked to perform this job produced in the beginning hardly usable very inhomogeneous results. A systematic approach was needed. Howard and Longstaff, proposed in [6] a computer and network security incidents taxonomy that suggested us a way for establishing a precise template for threat and attack statement. This template is shown in Figure 2.

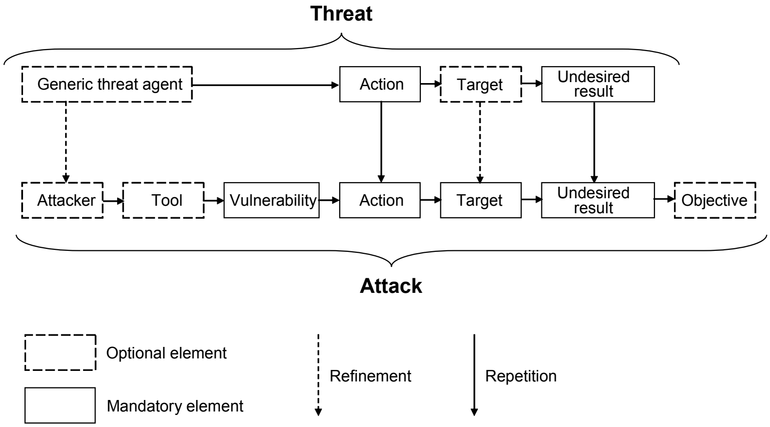

Figure 2: A template for threat and attack statement.

By using this template, threats are expressed using phrases with a pre-defined structure of the type: an adverse entity or an accidental event (*generic threat agent*) does something (*action*) involving a *target* which results in the compromising of one or more information CIA qualities (*unauthorised result*). The elements of the phrase enclosed in a dashed rectangle are not mandatory. For example, a threat relevant to virus or similar malicious code could be correctly expressed using the following statement

> *Adverse entities could introduce malicious code (viruses, Trojan horses, etc.) into an organisation IT system, causing loss of information.*

As an example of a threat of an accidental nature consider the following statement

> *An accident could cause a fire having as a consequence the unavailability of information assets.*

An attack is a way for a threat to materialise. Usually for each threat many attacks exist. In our security model, attacks are specific to the various system components (representing the various ways in which threats can be realized upon them). In specifying the attack the specification of the corresponding threat is refined with the provision of additional indications on how the former is realized.

More precisely, as shown in Figure 2, an attack statement will refine a threat statement by a mandatory specification of

- the vulnerability exploited by the threat agent to materialise the threat;
- the target of the attack.

In order to give a better characterisation of the attack and of the attacker, an attack statement can optionally contain

- a refinement of the threat agent;
- a description of the tools used to perform the attack;
- the objective of the attacker.

For example, an attack implementing the threat relevant to virus or similar malicious code could be correctly expressed using the following statement

> *A hacker, using e-mail messages and exploiting the lack of policies on virus protection, introduces malicious code (viruses, trojan horses, etc.) into an organisation IT system, causing loss of information, for self gratification.*

In a similar way, an attack deriving from the accidental threat of fire could be expressed as follows

> *A short circuit, due to inadequate fire prevention, detection and extinguishing system, causes a fire in a server room having as a consequence the unavailability of information assets.*

Once a clear structure for threat and attack statements was defined, it was easier to populate the security measure tables of the database in such a way that it was possible to understand exactly their effects in countering attacks and threats. In the Defender Manager© database, the security measures are defined at two levels of detail called respectively, *control* and *detailed control*. The first of these relates to threats while the second relates to attacks and involves a levelling over three levels corresponding to the increasing effectiveness of the security measures. Templates similar to those presented in Figure 2 for threats and attacks were devised also for controls and detailed controls in order to help keeping specification consistency and homogeneity.

Security measure levelling has been performed on the basis of the attack potential required to launch a successful attack notwithstanding the presence of the security measure itself. The algorithm used to estimate the security measure strength is similar to the one proposed in [4] adequately extended and adapted in other to make it applicable to security measures relevant to attacks that exploits vulnerabilities depending on procedural, physical and personnel related issues.

The Defender Manager© database associates threats, attacks, controls and detailed controls with other characteristics such as their impact on information asset confidentiality, integrity and availability, the category of system components these entities are applicable to, and other parameters (mainly weights) whose discussion is beyond the scope of this paper. This set of cross-referenced data provides an amount of information which can be used during the risk assessment process, and that other models available today often lack.

5 Modelling a security perimeter

According to the taxonomy introduced in Sec. 2, systems are the building blocks of security perimeters. Systems are collections of highly interconnected technical and social components (e.g. PC, operating systems, web servers, networks, application software, buildings, offices, power distribution systems, personnel, organizational infrastructures, etc.). In the beginning we tried to model systems by means of an oriented graph. A very simplified an incomplete example of the result achievable by using this technique is shown in Figure 3 (a). A path from a *triangle* (an attacker) to a *rectangle* (an information asset), passing through one ore more *circles* (the system components), represents a way for an attacker to compromise one ore more of the target information asset CIA qualities (which one depending on the CIA qualities of the attack). Unauthorised walking around the graph is made hard by the security measures associated to the system components. A single arc represents an attack. Following a successful attack, the attacker gains some control on the compromised component and from there he can try to attack (with an attack potential that can be different from the original one) other components and the relevant information assets. Given an attack A (consisting in the exploitation of a vulnerability V) and a node N being the target of the attack, it can be noted that some attackers can directly attempt to attack N while other can attempt A only after having successfully attacked other system components including a component represented by one of the nodes connected to N. A direct attack is represented by an arc going from a triangle to a circle and is not countered by the security measures implemented to protect nodes other than the attacked one. During the risk assessment process, the expected frequency of the attacks identified by A (i.e. the expected frequency of exploitation of the vulnerability *V*) must be estimated. This frequency can be thought of as the sum of the expected frequency of direct attacks exploiting *V* and that of the attacks of the same kind passing through other nodes connected to *N* (*indirect attacks*). While the first addendum can be estimated without taking into account the security measures in place (or planned), the second cannot be. Modelling in such a precise way a system, its components and their interconnections would require a large quantity of data that make impractical using this technique for characterising most real systems. In the end, we had to define a simpler modelling technique aiming at balancing complexity with the quality of the results.

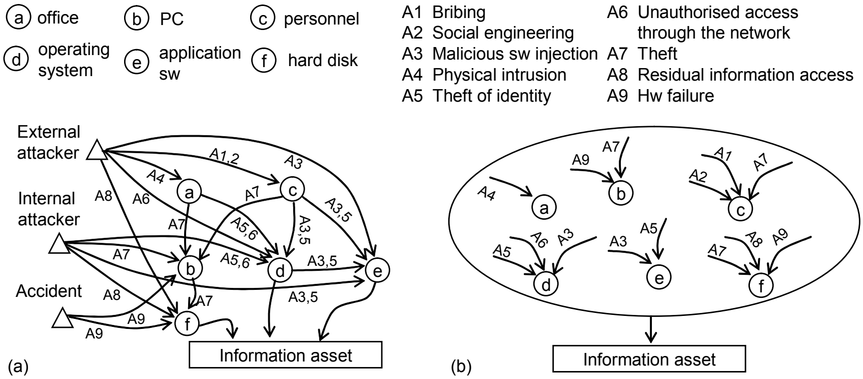

Figure 3: Modelling a system by (a) an oriented graph, (b) a list of components.

The selected approach consists in modelling a system by simply listing its components without any attempt to precisely represent component interconnections. Risk assessment is therefore performed considering each component independently of the others (Figure 3 (b)). In order to calculate the risk related to a component and to a specific attack the expected frequency of this attack and its consequences must be estimated. As previously discussed, only the expected frequency of direct attacks can be estimated when information about components interconnections lacks. Nevertheless, a rough estimation of the expected frequency of indirect attack can be worked out assuming that every components in the system have or will receive a reasonable and homogeneous level of protection and that, as a whole, they play a role in countering any indirect attack. Note that under the above assumptions in most cases the expected frequency of the direct attacks is predominant. This approach seems to be sufficiently accurate at least when a qualitative metric is used for rating over a few level scale the model variables. As far as the attack consequences is regarded this can be assumed to be a weighted sum of the value (with reference to the CIA qualities) of the information assets directly associated to the component and of those information assets that can be indirectly attacked passing through the component itself. Note that this does not require a complete view of component interconnections but just +a general understanding of how the system works.

6 The tool Defender Manager©

The methodology and the model discussed in this paper has been used as the basis for the development of a proprietary automated risk management tool, named Defender Manager©. This tool consists of two major parts: the *engine* and the *knowledge base*. The *engine*, using both information provided by the users and retrieved from the knowledge base, assists the users in all typical risk analysis and management activities (information asset identification and assessment, threat and vulnerability analysis, safeguards selection, etc.). The knowledge base contains a customizable list of general-purpose *component categories* (operating systems, web servers, networks, application software, buildings, offices, personnel, infrastructural components, etc.) that can be *instantiated* and used as building blocks for modelling real systems and security perimeters. It also contains a large (fully customizable) list of threats security, attacks, and security measures. (The knowledge base may accommodate specific sets of threats and countermeasures, in order to be compliant with standards such as ITSEC (Information Technology Security Evaluation Criteria, Common Criteria, ISO IS17799, military norms or custom security policies. It may also be customised to fit specific contexts other than information and ICT (Information and Communications Technology) security, such as site security, critical infrastructures security and so on.) It is managed by a relational database management system that interlinks threats with attacks, attacks with security measures and each of these entities with the security qualities (CIA) they impact on and the component categories they apply to. Security measures of different strengths (rated over a three level scale) are provided so that the engine can suggest the most appropriate choice depending on risk analysis results.

The tool assists during the whole process of designing a cost effective protection system. The first phase of this process concerns *modelling the security perimeters*. To this end, Defender Manager© provides the functionality to describe each system in the security perimeter by instantiating and characterising the general purpose building blocks provided in the *component categories* knowledge base. If a *component category* is required but it is not present in the knowledge base, and consequently no information about the relevant *threats/attacks* and *security measures* is included in the database, the tool allows the user to create new *component category* by introducing the required data.

The second phase is relevant to the *identification and classification of the information assets* and their association to the various systems and components of the security perimeters. To this end, the tool proposes a fully customizable *questionnaire* that can be used to interview the *information asset* owners. The tool allows to record the answer to the various questions and, depending on this answers, automatically assigns a *CIA criticality* to every information asset, representing the value of the asset with respect to the three security qualities. A 4 level qualitative scale is used in this case (*negligible*, *low*, *medium* and *high* criticality).

The third phase is concerned with *assessing and managing risk*. In this phase, the tool uses its own database to identify threats and attacks which are pertinent to the various *components* in the perimeter and requests the users to estimate the *expected frequency of occurrence* of the various *attacks* (regardless their *expected potential* and consequently their possibility of success). For each attack and each component, the tool automatically calculates a parameter called the *level of intrinsic risk*. This parameter is a function of the CIA criticality of the information asset stored or processed in the component and of the expected frequency of occurrence of the analysed attack. The level of intrinsic risk is used by the tool to identify a set of optimal countermeasures whose strength is appropriate for that level of intrinsic risk. The user can describe the security measures which he has already implemented or planned to implement using appropriate input forms provided by the tool itself. By comparing the optimal suggested solution and the actual situation described by the user, the tool lastly calculates the *level of residual risk* to which the component, system or the entire security perimeter is exposed.

In conclusion, Defender Manager© constitutes a security control panel providing all the information necessary to make informed decisions on which actions to take, to justify these decisions, to understand the consequences of every decision. For example, Defender Manager® makes it possible to verify, with reference to each system within the security perimeter, which are the relevant threats, the corresponding level of risk, recommended security measures and those effectively implemented, which attacks are adequately countered and which are not. All this information can be exported in textual form and/or as graphics.

References

[1] Law, W. and Kelton, W., *Simulation Modeling and Analysis*, McGraw Hill, 2000.

[2] Haimes Y.Y. and Horowitz B.M., Adaptive Two-Player Hierarchical Holographic Modeling Game for Counterterrorism Intelligence Analysis, *Journal of Homeland Security and Emergency Management*, Volume 1, Issue 3, Article 302, 2004.

[3] Committee on National Security Systems, *National Information Assurance (IA) Glossary*, CNSS Instruction No. 4009, 2003.

[4] CCIMB, *Common Methodology for Information Technology Security Evaluation -Evaluation Methodology*, Version 2.2, Revision 256, 2004, http://www.commoncriteriaportal.org.

[5] Kaplan, S. and Garrick, B.J., On the Quantitative Definition of Risk*, Risk. Analysis*, Vol 1 No 1, pp. 11-27, 1981.

[6] Howard, J.D., Longstaff, T.A., A Common language for Computer Security Incidents, *SANDIA REPORT SAND98-8667*, 1998, http://www.cert.org/research/taxonomy_988667.pdf.