Complex vs. Simple Asset Modeling Approaches for Information Security Risk Assessment Evaluation with MAGERIT methodology

Adrián Fernandez, Daniel F. Garcia
Department of Informatics
University of Oviedo
Gijon, Spain
UO170656@uniovi.es
dfgarcia@uniovi.es

ABSTRACT: *A proper asset modeling is essential to develop an information security risk assessment in any corporation. A too complex model will take a long development time and may require parameter values difficult to get. On the contrary, a too simple model will provide inaccurate estimations of risks, although it could be developed fast. One of the characteristic that most influences the complexity of the model is the way to characterize the dependence between assets, generally using a dependency graph. This work evaluates how slight variations in the complexity of the dependency graph affect to estimated risks. To carry out the evaluation, the MAGERIT methodology is used because it can handle graphs of variable complexity and allows qualitative or quantitative asset valuation. Finally, this work provides insights to select a proper complexity for the asset modeling approach used.*

Keywords: Information security, Risk assessment, Asset modeling, Dependency graph

Received: 15 July 2016, Revised 21 August 2016, Accepted 27 August 2016

1. Introduction

Today, information security is a primary concern for any corporation. Reaching a high level of security is very desirable, but it would consume a lot of resources and would be very expensive. A more feasible approach consists on providing the appropriate security level to each asset of the corporation, considering the different types of assets: information, services, software, hardware, facilities, etc.

The common method to select the appropriate protections (not insufficient, not excessive) for assets is carrying out a risk assessment process. There are many methodologies to carry out a risk assessment, but all of them requires a model of corporate assets. Typically, the model includes the value of each asset, or at least of the most important assets, and also the dependencies between assets.

The identification of assets and their valuation is the first step of any risk assessment process. The second step is the organizations of assets in layers and establishing dependencies between assets of different layers or in the same layer. Finally, a dependency graph between all assets is built. This process must be developed for each dimension of security: confidentiality, integrity, availability, authenticity and traceability. Generally, the graphs are similar for the five dimensions but they are not exactly the same.

The asset modeling process requires a deep understanding of the business process of the corporation, the information used to support them and the infrastructure required for storing and processing the information. Generally, the modeling process involves interviews with persons of many departments of the corporation and a lot of information must be collected and organized. Usually, the development of the model takes a lot of time and the engineers have to make a choice between slowly develop a complex model, which would be very accurate, or develop a simpler model quickly but accurate enough.

In this work we develop asset modeling approaches, more complex and simpler, and compare them. In order to evaluate and compare models of different complexity, we must use a risk assessment methodology that can easily accept models developed with different approaches and objectives. The MAGERIT methodology [1] fulfills this requirement very well, and furthermore, MAGERIT provides a simplified asset valuation mechanism that reduces noticeably the modeling efforts. For these reasons, the asset modeling approaches developed in this work will be integrated in MAGERIT risk assessment methodology.

The rest of the paper is organized as follows: the next section introduces the related work which is essential to understand the currently available modeling approaches; the third section introduces the MAGERIT methodology; the fourth section present the modeling approaches developed, and the conclusions are summarized in the final section.

2. Related Work

The asset modeling is always developed at the first stages of any security risk assessment. Therefore, the context for asset modeling is provided by the methods and techniques of risk assessment.

Shameli et al. [2] provide a taxonomy on information security risk assessment that includes the most relevant methodologies. One of the four categories used by the taxonomy is resource valuation, but other categories like appraisement (qualitative, quantitative) or perspective (asset, service or business driven) or risk measurement (propagated or non-propagated) are also very strongly related with the modeling of assets.

Corporations have been doing risk assessment analysis for many years, and therefore, they have had to model their assets properly. Next, a chronological review of most relevant works is developed.

Baskerville [3] provides a good compilation of the first generations of risk analysis methodologies. Although the compilation includes very little information on the modeling of assets, it provides references to papers with detailed information on methodologies which also includes the modeling of assets.

Shu and Han [4] proposed a model of two layers for asset modeling: business functions and support assets (like applications, hardware, etc.). A matrix defines the grade of dependency of each business function from each supporting asset. Then, the matrix defines the vertical dependencies between the elements of the two layers of the model. Furthermore a graph defines the dependencies between support assets. Then, the graph models the horizontal dependencies within this single layer. The model is used to assess quantitatively the risk of disruption of business functions provoked by risks suffered by supporting assets and propagated to the business. This methodology integrates many of the basic concepts of current methodologies, like the utilization of a business perspective, the consideration of dependencies and the propagation of risks, although in a simplified manner.

Innerhofer and Breu [5] proposed managing risks using an enterprise architecture, that is, a detailed meta-model of the enterprise based on four levels: business, application, technical and physical. In the business level there are roles (persons) that develop activities as part of business processes. The activities consume and generate information using software components of the application level. The components are executed on nodes (computers) of the technical level, which also storage the information. The nodes send and receive information through edges (communication equipments). The nodes and edges reside in locations of the physical layer. There is a clear dependency graph between the elements of the four layers. To carry out a risk analysis,

more information is attached to each element of this enterprise model, as security requirements or threats. Security controls are added to threats. The main drawback of this methodology is that assets are not really valuated. Only is defined the impact and frequency of each threat on assets using three qualitative values.

As part of the approach to implement ISO/IEC 27005 proposed by Leitner [6] there is a method to model the assets of a corporation using a service tree extracted from the Configuration Management Data-Base (CMDB) of the corporation. The connections (dependencies) in the tree are based in the DIN-25424 standard for fault tree analysis. The qualitative risk levels calculated for each element of the tree (each configuration item) are propagated to the top of the tree considering the two connection types. With the OR connection, used when there are redundant elements, the average of the risk levels is calculated. With the AND connection the maximum risk level is taken. This propagation scheme, based on the aggregation of levels, is too simple and it does not allow the independent analysis of the risk dependencies.

Zambon et al. [7] developed a qualitative model and technique for assessment of availability risks, called QualTD, to be used together with standard risk assessment methods. It is based on a dependency graph defined formally. The dependencies between the nodes of the graph can be of types AND and OR. Threats and vulnerabilities, with their likelihood are associated to the nodes. Incidents are also associated to nodes, selecting a threat exploiting a vulnerability to attack the node. The incident propagation and the nodes affected is also defined formally. Finally, the technique calculates the global impact and the aggregated risk of each incident in all businesses affected by the incident. This classic and well formalized qualitative technique uses a very few levels to define impact, likelihood, risk, etc., which generate coarse analysis results. Furthermore, results are aggregated for each incident generating global results, good to obtain a general picture of risks but hiding details required for a fine analysis.

Loloei et al. [8] use the meta-model proposed in [5] using only the three upper layers. Weights are assigned to the edges that define the dependencies between the elements integrated into the layers of the model, but dependencies are only defined for availability. Values are assigned to the elements of the model considering the cost of the element and the impact that could provoke in a business process. The valuation is qualitative in the examples provided. Next, an acyclic dependency graph is derived from the meta-model. Finally, the value of the elements is recalculated using an algorithm that considers the propagation of the value through the graph. This work follows the classic modeling approach based on dependency graphs but without considering the physical layer and focusing exclusively on availability.

Asset modeling methods developed recently look for simplicity because the analysts tend to avoid the use of cumbersome methods. Breier [9] has developed a relatively simple method for asset valuation. The dependencies are established considering the three main dimensions of security: availability, confidentiality and integrity. The model considers four main types of layers: Data, Applications, Equipments and Locations. In the top layer several types of data are considered: customers, company, user authentication, e-mail, private, etc. This model does not consider business processes, and they have to be modeled indirectly by software applications. The model includes two types of dependencies. The classic AND for an asset that depends on other asset, and OR for an asset that depends on any one of several redundant assets. The dependency between two assets has three weights to represent independently the grade of dependency in three basic security dimensions. The risk is propagated through the dependencies but using an adjusting formula. This method follows the current tendency towards simplicity but the lack of a business layer is contrary to current trend to mainly valuate the services provided and the information handled by them.

The conclusion of the analysis of related work is that we need an asset modeling method not too complex, but enough sophisticated to provide good results for risk analysis. The complex methods use to be cumbersome and requires long time to get the results of the analysis. On the contrary, the very simple methods are not capable to provide enough accurate results. For these reasons, MAGERIT methodology has been selected to develop our evaluation work due to its flexibility to use simple or complex dependency graphs, and qualitative or quantitative valuation for the same computer system and considering simultaneously five dimensions of security. The complexity of MAGERIT is similar to the other methodologies based on dependency graphs.

3. Introduction To Magerit Methodology

In Spain, the Council for Electronic Administration has decided to use MAGERIT (Methodology for Information Systems Risk Analysis and Management) as common methodology for risk management framework according to ISO/IEC 27001 standards. This methodology is simple and quick to apply, well as providing good results on the status of the risk assumed. It can also be

used as a basis for support when making decisions for improvement.

This methodology can be applied from two different approaches: qualitative or quantitative. The first seeks to know what is in Boolean terms. Furthermore, the qualitatively mode also includes how much there are.

MAGERIT is based on following concepts:

• **Assets:** Anything that is useful or valuable for a company, which may be material or intangible. Assets must be identified and valuated, however MAGERIT states that the criteria for evaluation must not be its "real value", it should be the estimated value of the damage caused to the organization by a malfunction in any of the security dimensions.

• **Dependencies:** In addition to the assets, we must also identify the dependencies between them. That is, the ways in which the impact and risk may spread from one asset to another. MAGERIT establishes that essential assets (principally the information) must be at the top level of the dependency graph, services under them, and at the last level, other assets such as hardware, software and communications. For each of the dependencies it must be indicated the dimensions of security involved and percentage of dependency.

• **Threats:** Threats should be identified for each asset in each dimension and evaluated according to their frequency and the degradation provoked on the asset if they materialize. MAGERIT provides several sets of threats related to each class of asset. So that the correct identification and class assignment to assets is essential to start from. Furthermore, it may be necessary to add specific threats as appropriate.

• **Safeguards:** As is the case with threats, MAGERIT provides a huge list of predefined safeguards associated with assets and threats. This way a checklist could be used to determine which are applied and their maturity. Again, depending on the particular corporation, it may be necessary to add custom safeguards.

Impact is defined as the loss of value of an asset when a threat materializes. Equation 1 defines this concept.

Figure 1 shows the components of risk assessment with the MAGERIT methodology.

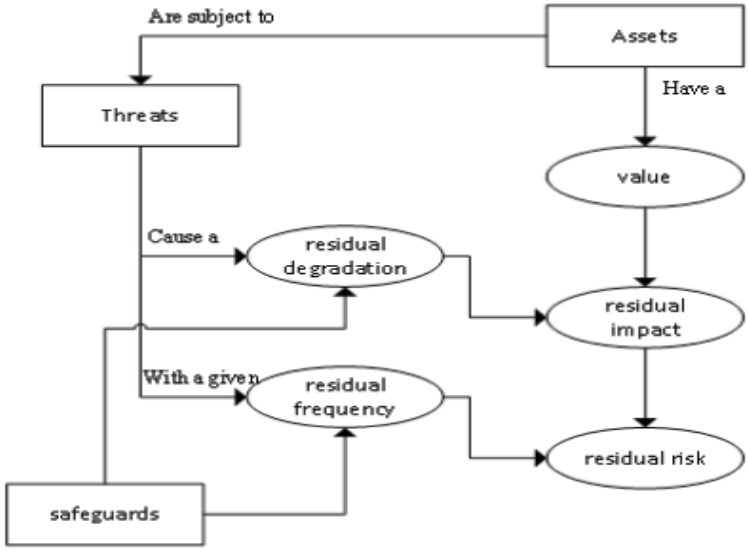

Figure 1. Elements for residual risk analysis

$$Impact = Asset\ Value \times Asset\ Degradation \quad (1)$$

After applying safeguards, the impact of threats, depending on their effectiveness, is reduced. This generates a residual impact of the threat that is calculated with equation 2.

$$Residuallmpact = Impact \times (1\text{- } Effectiveness) \quad (2)$$

where effectiveness is measured as real number from 0 (no effect on the impact) to 1 (total protection).

Risk is the measurement of the probable damage to system. So, residual risk calculation can be expressed by equation 3.

$$ResidualRisk = Residuallmpact \times Frequency \quad (3)$$

4. Asset Modeling Approach

The first step is asset identification and classification into groups provided by PILAR (Procedimiento Informatico-Logico para el Analisis de Riesgos) (Logical Computer Procedure for Risk Analysis), the software tool to help the use of the MAGERIT methodology. Then, it is needed to build the tree of dependencies between all assets previously defined. Finally, when the assets are defined and related it will be necessary to assign the value of each one. When the assets are well defined, MAGERIT determines potential threats and, using its impact valuation, the risk can be determined.

To show how to use MAGERIT in practice, we will conduct an example in which we analyze the risk of a simple model of case processing by two services, one in person and another one remotely, based on a cloud infrastructure.

This example will be used to perform dependencies modeling between assets in two ways, first by a classic or complex scheme and, on the other hand, following the simplified approach proposed by MAGERIT, in order to compare both results on the calculated risk in major services.

4.1 Asset identification

The assets are the resources in the information system or related to it that are necessary for the organization to operate correctly and achieve the objectives proposed by its management. Only those information system resources that have value to the organization, either by themselves or because they support other valuable assets, are of interest.

MAGERIT established a set of layers for asset classifying:

- **Business layer [B]:** Essential assets for the organization, usually they define herein both data, which contain information handled by system, and services, the essential business processes.
- **Internal services [IS]:** Ancillary services defined to group assets working together to achieve the same goal.
- **Equipment [E]:** The equipment layer integrates different groups as software [SW], which allow data to be handled; hardware [HW], that hosts the data, applications and services; communications [COM] allowing exchange of data and auxiliary elements [AUX] for complimenting the computer equipment.
- **Outsourced services [SS]:** Contracted services to third parties.
- **Locations [L]:** House the computer and communications equipment.
- **Personnel [P]:** Use or operate all the above elements.

An asset can be individual or collective. If, for example, an administrative unit has 10 PC it could be taken like unique asset or like ten various different assets. In this case the decision will be made according to the role played by these PCs. If they all perform the same tasks and have similar configuration, a unique asset can be used, to simplify. However, if they have different roles or tasks will be better define various assets.

As an example, we develop the model of the service for attention to citizens of a public administration. The service is provided in offices and also telematically through Internet.

All information of the citizens is stored in a data base server deployed in a cloud.

As all offices are similar, and therefore the model only uses one office to represent all them.

The model of the example include the assets showed in table 1:

Layer	Asset	Description
B	**Info**	Data of the cases
B	**In Person**	In person cases processing service
B	**Remote**	Remote cases processing service
IS	**Personal Attention**	Auxiliary service involving elements for in person attention to clients
IS	**Information processing**	Auxiliary service involving elements for data processing.
E/SW	**Local corp app**	Corporative application
E/SW	**Httpd**	Web server + Web page + Web services
E/SW	**Database**	Database management system
E/HW	**PC**	PCs located in the office
E/HW	**VM Cloud**	Virtual Machine hosting servers
E/COM	**LAN**	Local Area Network in the office
E/COM	**Virtual firewall**	Virtual firewall to control access to cloud assets
E/COM	**Local router**	Router providing communications to the office
SS	**ADSL**	ADSL Service
SS	**Energy**	Energy Service
SS	**IaaS**	IaaS Service
L	**Office**	Office where employees attend clients
P	**Employees**	Employees working for the company in clients' attention

Table 1. Assets identification of the example

4.2 Dependency building

Dependencies between assets are organized into a hierarchical tree structure. An upper asset A depends on a lower asset B means that the degradation suffered by B will affect A.

Due to its dependencies there are "upper assets" and "lower assets". Lower ones are considered the basis supporting higher ones. Thus, the occurrence of a security threat in a lower asset will spread the damage up through the dependencies.

Essential assets are the most important assets of the organization. Both information and services provided are the basic elements to be protected.

Internal services are sets of tasks carried out to support the provision of main services or information processing.

At the lowest level are the elements of hardware, software and communications that allow the execution of business processes.

The figure 2 shows the dependency diagram for the example under analysis.

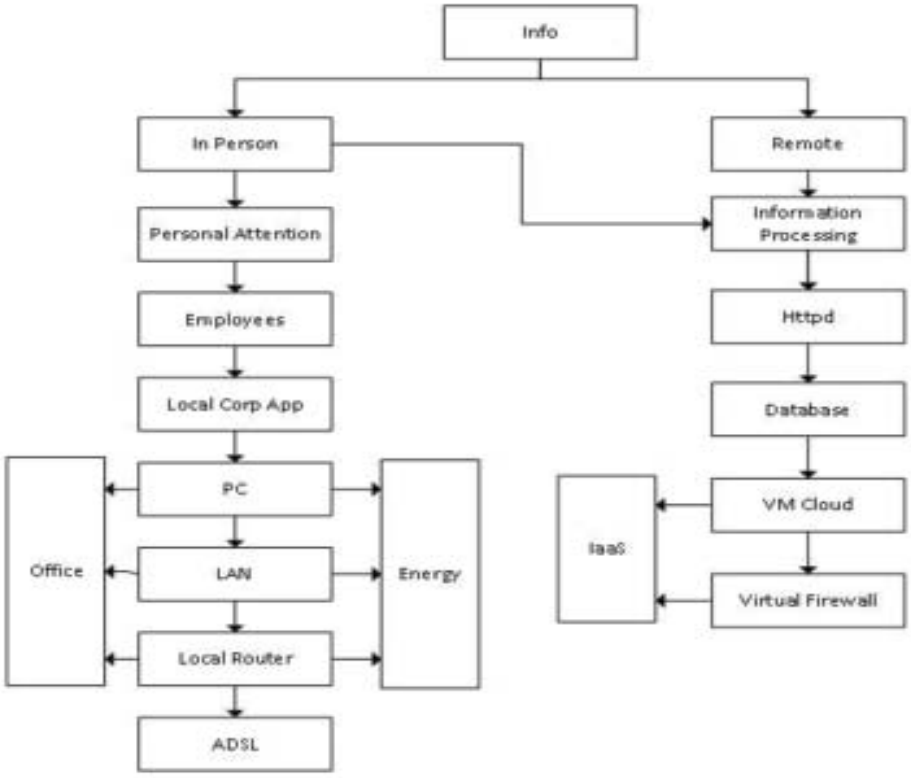

Figure 2. Dependency tree in case study (Classic approach)

Figure 3 shows an example of dependency diagram developed under MAGERIT methodology:

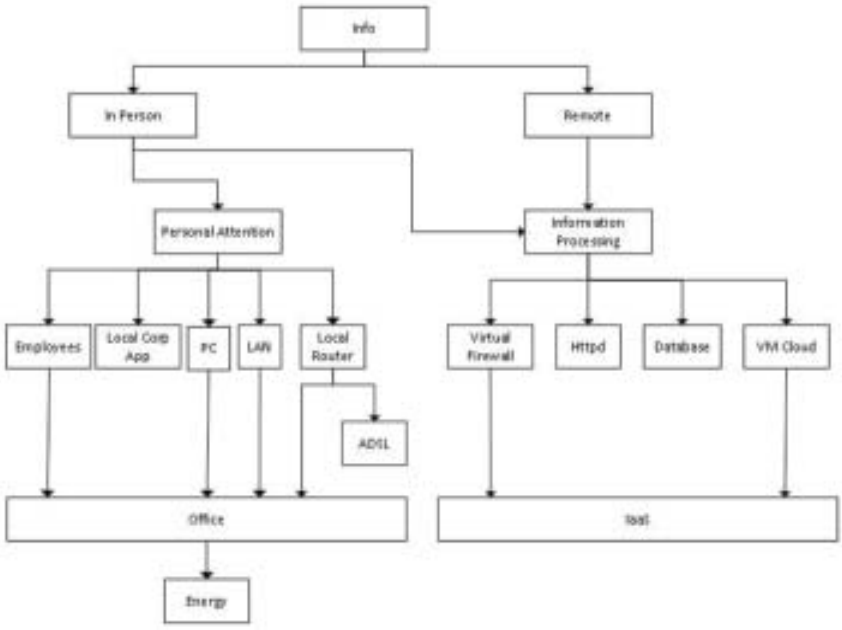

Figure 3. Dependency tree in case study (MAGERIT approach)

MAGERIT uses a simplified dependency model, instead of consider each asset needed by another asset in the five dimensions, it only thinks through those whose damage could spread to the asset concerned.

4.3 Calculation of dependencies

From the point of view of the qualitative mode dependencies are binary, exists or not exists. In case that a dependency exists, it will be total.

The calculation of transitive dependencies in qualitative mode is direct, as shows next equation.

$$A \to C \leftrightarrow \exists B, (A \to B) \wedge (B \to C) \tag{4}$$

Conversely, using quantitative mode dependencies must specify their degree of dependency. This degree is expressed as a percentage. It means how much depends on an asset from other.

For transitive relations of dependence, the degree is obtained as shows:

$$A \to C \leftrightarrow \exists B, (A \to B) \wedge (B \to C) \tag{5}$$

$$Degree\,(\mathrm{A} \to \mathrm{C}) = \sum_{i} \langle\, Degree\,(A \to B_i) \times Degree\,(B_i \to C) \,\rangle$$

The sums must be performed as follows.

$$a + b = 1 - (1 - a) \times (1 - b)^3 \tag{6}$$

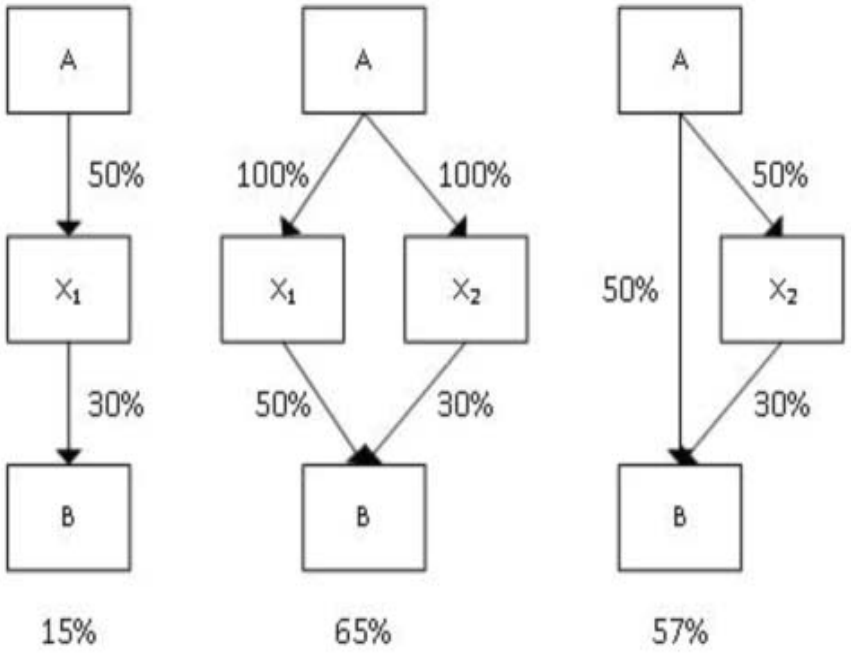

Figure 4. Examples of quantitative transitive dependencies calculation

This way of adding is taken of Bayes' theorem. It is given by the need to ensure that if an asset depends on another in various ways (diamond structure), total dependence can never exceed 100%.

The figure 4 shows three different examples of quantitative transitive dependencies between an asset A and an asset B through different assets X_i.

Dependencies must be analyzed in the five domains of security. An asset A does not have to depend on asset B in all domains, can have it only in a set of them. In the case of quantitative also can have dependencies with different degrees.

4.4 Assets valuation

Finally, the last task to model the assets is to value each one. It must be taken into account the five dimensions of security to assign value to assets. MAGERIT does not consider an economic or "real" value of the elements. Instead of, it uses the value that the organization would have to bear as a result of the materialization of a threat on an asset.

Before define values it is important to consider if working in a qualitative or quantitative mode.

In a qualitative mode assets are valuated with an integer between 0 and 10 in relative order with respect to others. The limitations of this mode are that they do not allow values to be compared beyond their relative order (for example, A is greater than B).

On the other hand, in a quantitative mode the values are amounts of money. For example, the economic cost of replacing corrupted application could be $100 (time of an employee for reinstalling software), but this failure in the application availability will be propagated to the service, whose interruption cost $4000. In this case 100 is the value of "PC" availability and 4000 the value of "In Person" service in its availability dimension.

It requires more effort to develop but gives more information.

A good technique to valuate assets is asking these questions:

- **Availability:** What damage would be caused if it were not available or could not be used?
- **Confidentiality:** What damage would be caused by unauthorized access?
- **Integrity:** What damage would be caused if it were damaged or corrupt?
- **Authenticity:** To what extent is harmful the lack of knowledge about who has done what?
- **Accountability (service):** What damage would be caused by not knowing to whom the service is provided?
- **Accountability (data):** What damage would be caused by not knowing who accessed the data and what they do with them?

In MAGERIT, each asset can have two values, own and accumulated. The own value of an asset is the value assigned to the asset by the analyst. The accumulated value of an asset comes from its own value plus the accumulated value of all assets that depend on it.

Being $A_1, A_2 \ldots A_n$, some assets that depend directly or indirectly from B, equation 1 represent the accumulated value of the asset B in qualitative mode:

$$Accumulated\ Value(B) = \max\big(value(B), value\,(\mathrm{A}i)\big)\ \forall \mathrm{i} \in [1, n] \tag{7}$$

The calculation of the accumulated value in quantitative mode requires knowledge of the degree of dependence.

Knowing the degree of dependency between assets, their accumulated value can be calculated with next equation:

$$AccumulatedValue\ (B) = Value\ (B) + \sum_{i} \langle\ Value\ (A_i) \times\ Level\ (A_i \Rightarrow B)\ \rangle \tag{8}$$

MAGERIT has several checklists to help asset valuation, classifying them in values like "Critical information", "Law-protected information", "Important information for business" and "Trivial information" it's easier to determine its value.

These checklists are a good starting point for a first quick valuation of assets, however later may be refined manually.

To assign a qualitative value such asset, in each of the dimensions of security, we used the following criteria:

- 10: Critical assets, must have proper operation.
- 8-9: Important for the proper provision of the service, but not essential.
- 6: Less important assets.
- 3: Without important implications.

Table 2 shows the valuation of the assets of the study case and their valuation in qualitative mode:

Assets	**Values**				
	A	**I**	**C**	**Auth**	**Acc**
Info	9	10	10	9	9
In Person	9	9	10	8	8
Remote	9	9	10	8	8
Personal Attention	7	7	7	6	6
Information processing	9	10	10	8	8
Local Corp App	6	6	7	6	8
Httpd	8	8	7	6	8
Database	9	10	10	8	8
PC	5	3	7	6	6
VM Cloud	9	10	10	8	8
LAN	6	3	6	6	-
Virtual Firewall	6	7	6	6	-
Local Router	6	3	4	5	-
ADSL	9	-	-	-	-
Energy	9	-	-	-	-
IaaS	9	9	10	8	-
Office	7	-	-	-	-
Employees	6	4	8	8	6

Table 2. Case study qualitative assets valuation

Table 3 shows the valuations determined automatically by PILAR, in quantitative mode, due to the assets type:

Assets	Values				
	A	**I**	**C**	**Auth**	**Acc**
Info	100K	210K	210K	100K	100K
In Person	100K	100K	210K	46K	46K
Remote	100K	100K	210K	46K	46K
Personal Attention	21.5K	21.5K	21.5K	10K	10K
Information processing	100K	210K	210K	46K	46K
Local Corp App	10K	10K	21.5K	10K	46K
Httpd	46K	46K	21.5K	10K	46K
Database	100K	210K	210K	46K	46K
PC	4K	1K	21.5K	10K	10K
VM Cloud	100K	210K	210K	46K	46K
LAN	10K	1K	10K	10K	-
Virtual Firewall	10K	21.5K	10K	10K	-
Local Router	10K	1K	2K	4.6K	-
ADSL	100K	-	-	-	-
Energy	100K	-	-	-	-
IaaS	100K	100K	210K	46K	-
Office	21.5K	-	-	-	-
Employees	10K	2K	46K	46K	10K

Table 3. Case study quantitative assets valuation

4.5 Threats

Threats must be related to assets. PILAR, the tool that provides support to use MAGERIT has integrated a standard library of threats. PILAR assigns threats to assets automatically. But it is important to determine which ones are really applicable and estimate the degradation that each threat would cause in each asset. This degradation is a percentage between 0% and 100% representing how much value losses the asset when the threat occurs.

In addition, the analyst must specify the frequency of occurrence of threats. In quantitative mode frequency takes one of this values: Very low (1), low (3), medium (5), high (7), very high (9). In quantitative mode, the estimated occurrences per year is usually used.

The accumulated impact is calculated for an asset taken into account the threats which it is exposed and accumulated value (it own plus the value of the assets that depend on it)

Table 4 and table 5 show an example of threats related to “PC” and “Local Corp App” (app) assets respectively.

4.6 Impact

MAGERIT considers three kinds of impact over assets: own impact, accumulated impact and deflated impact. All of these impacts are measured in same units as the assets.

Own impact is the one calculated in previous section. It refers to the impact of threats on the asset itself.

Accumulated impact is the degradation of the accumulated value of an asset. It can be calculated with a function $f(v, d)$, being v the accumulated value and d the degradation percentage. This function needs to meet the following requirements.

Asset	**Threat**	**F**	**Percentage of degradation**				
			A	**I**	**C**	**Auth**	**Acc**
PC	Fire	L	10%	-	-	-	-
PC	System errors	M	2%	20%	20%	-	-
PC	Masquerading of identity	M	-	50%	50%	100%	-
PC	Abuse of access privileges	M	1%	10%	50%	-	-
PC	Unauthorized access	M	1%	1%	50%	-	-
PC	Deliberate alteration of information	H	-	100%			
PC	Disclosure of information	M	-	-	10%	-	-
PC	Software manipulation	M	5%	100%	100%	-	-
PC	Denial of service	M	10%	-	-	-	-
PC	Theft	M	1%	-	100%	-	-
Max. degradation			**10%**	**100%**	**100%**	**100%**	**-**

Table 4. Example of threats related to the asset “PC”

Asset	**Threat**	**F**	**Percentage of degradation**				
			A	**I**	**C**	**Auth**	**Acc**
App	Hardware or software failure	L	50%	-	-	-	-
App	User errors	M	1%	10%	10%	-	-
App	System errors	M	20%	20%	20%	-	-
App	Software vulnerabilities	M	1%	20%	20%	-	-
App	Masquerading of identity	M	-	50%	50%	100%	-
App	DoS	H	-	-	-	-	100%
App	Software manipulation	M	50%	100%	100%	-	-
App	Malware diffusion	M	100%	100%	100%	-	-
App	Hardware or software failure	M	50%	-	-	-	-
App	User errors	M	1%	10%	10%		
Max. degradation			**100%**	**100%**	**100%**	**100%**	**100%**

Table 5. Example of threats related to the asset “Local Corp App”

$$Impact(0,0\%) = 0 \tag{9}$$

$$Impact(v, 0\%) = 0$$

$$Impact(v, 100\%) = v$$

$$\forall d, v_i < v_j \rightarrow Impact(v_i, d) < Impact(v_j, d)$$

$$\forall d, d_i < d_j \rightarrow Impact(v, d_i) < Impact(v, d_j)$$

Deflected impact is the degradation of value suffered by asset A that depends on asset B when B is degraded by the materialization of some threat.

4.7 Risk

The risk on an asset is the total loss of value suffered by the asset during the period of time considered in the analysis, typically one year.

There are two kinds of risks, accumulated and deflected. The accumulated risk is calculated on the assets that support the information system. However, the deflected risks are calculated on each asset in order to know the consequences of a threats coming from lower assets.

To obtain the accumulated risk of an asset, it must be used the accumulated impact on the asset. This value expresses the loss of value accumulated by threats to the asset itself.

Deflected risk is calculated by the deflected impact. This value represents the loss of value duo to threats suffered by lower assets.

	Asset	**A**	**I**	**C**	**Auth**	**Acc**
Qualitative	In Person	6.8	7.3	6.4	6.4	6.8
	Remote	6.8	7.3	6.4	6.4	6.8
	PC	4.9	7.5	7.0	6.9	-
	App	6.8	7.0	7.0	6.4	6.8
Quantitative	In Person	3M	2.4M	1.1M	600K	650K
	Remote	3M	2.4M	1.1M	600K	650K
	PC	550K	12.6M	4.5M	1.4M	-
	App	4.3M	1.9M	3.0M	870K	940K

Table 6. Accumulated risks related to “In Person”, “Remote”, “PC” and “Local Corp App” with classic dependency model

To calculate the deflected risk of an asset, MAGERIT uses the sum of all the deflected risks of the assets which affect the asset concerned plus the asset own risk.

In qualitative mode, MAGERIT provides a table to interpret the values of risk obtained: 9 – Catastrophic, 8 – Disaster, 7 – Extremely critical, 6 – Very critical, 5 – Critical, 4 – Very high, 4 – High, 2 – Medium, 1 – Low and 0 – Negligible.

In quantitative mode, risk must be calculated using economic values, as the estimated annual loss.

MAGERIT does not provide a “global risk” concept, but it can be easily calculated using accumulated risk, which is the risk got using accumulated impacts. It also provides deflected risk, which is the risk obtained using deflected impacts.

Table 6 shows accumulated risks calculated using the classic dependency model for “In Person”, “Remote”, “PC” and “Local

Corp App" assets.

Repeating the calculations about threats and risks, but using the proposed simplified dependency model instead of the classic one, the results of table 7 are obtained.

	Asset	A	I	C	Auth	Acc
Qualitative	In Person	6.8	7.3	6.4	6.4	6.8
	Remote	6.8	7.3	6.4	6.4	6.8
	PC	4.9	7.5	7.0	6.9	-
	App	6.8	7.0	7.0	6.4	6.8
Quantitative	In Person	3M	2.4M	1.1M	600K	650K
	Remote	3M	2.4M	1.1M	600K	650K
	PC	510K	11M	3.9M	1.2M	-
	App	4.2M	1.8M	2.8M	680K	900K

Table 7. Accumulated risks related to "In Person", "Remote", "PC" and "Local Corp App" with simplified dependency model

Both models provide the same results in qualitative mode because they are using accumulated values, so, what matters is the set of whole dependencies, not the specific way assets are related.

	Asset	A	I	C	Auth	Acc
Qualitative	In Person	6.8	7.5	8.1	6.9	6.9
	Remote	6.8	7.5	8.1	6.9	6.9
	PC	4.4	3.9	6.3	5.7	-
	App	5.0	5.7	6.3	5.7	6.2
Quantitative	In Person	14M	19.6M	47.4M	8.6M	4.6M
	Remote	6.8M	9.7M	22.9M	4.4M	3.8M
	PC	88.5K	11.7K	180K	12.6K	-
	App	370K	180K	580K	170K	210K

Table 8. Deflected risks related to "In Person", "Remote", "PC" and "Local Corp App" with classic dependency model

However, in qualitative mode there are minor differences. The core services still have the same risks, but in this model PC has

less risk. This occurs because in the simplified model “Local Corp App” does not depends on “PC”, so the accumulated value of the asset “PC” is lower.

In the case of deflected risks, the same applies. Table 8 shows the deflected risks when classic dependency model is used and Table 9 when simplified model is used.

	Asset	A	I	C	Auth	Acc
Qualitative	In Person	6.8	7.5	8.1	6.9	6.9
	Remote	6.8	7.5	8.1	6.9	6.9
	PC	3.9	3.4	5.2	4.6	-
	App	5.0	4.7	5.2	4.7	6.2
Quantitative	In Person	14M	18.5M	46.3M	8.6M	7.1M
	Remote	6.8M	10.3M	22.9M	4.4M	3.8M
	PC	56.6K	11.7K	180K	12.6K	-
	App	180K	53.2K	120K	41.0K	210K

Table 9. Deflected risks related to “In Person”, “Remote”, “PC” and “Local Corp App” with simplified dependency model

There are minor differences in quantitative mode, but the overview of risks is quite similar.

5. Conclusions

The main conclusion is that in qualitative mode the accumulated risks are the same for the two dependency modeling approaches. In the quantitative mode the accumulated risks are the same for the essential assets of the business layer, although the risks are a little different for the asset “PC” and “Local Corp App”. Similar conclusions are obtained for deflected risks.

The real important result is that risks of essential assets of business layer practically do not depend on the detailed dependencies between the support assets of the equipment layer and the other lower layers. This allows to da simplified modeling of the dependencies of support assets accelerating the development of the model, but retaining the accuracy of risks estimations for the business layer assets.

The simplified approach does not allow analyze the deflected risk from support assets to other support assets. But its great advantage is that is not necessary neither to model the dependencies among support assets nor valuate them, because they accumulate the value of their supported assets.

This work has shown the possibilities of modeling simple vs complex and the results obtained with each. It is a decision of the security engineer to use one approach or another depending on the information wanted and the time available to obtain it.

Acknowledgment

This work has been partially funded by the Spanish Program for Development of Scientific and Technical Research under the project TIN2014-56047-P.

References

[1] Amutio, M. A., Mañas, J. A. (2014). MAGERIT version 3.0 Methodology for Information Systems Risk Analysis and Management, Book I - The Method, Spanish Ministry of Finance and Public Administration, July 2014.

[2] Shameli-Sendi, A., Aghababaei-Barzegar, R., Cherie, M. (2016). Taxonomy of information security risk assessment (ISRA),

Computers & Security, V. 57, p. 14–30, March 2016.

[3] Baskerville, R. (1993). Information systems security design methods: implications for information systems development, *ACM Computing Surveys,* 25 (4) 375–414, December.

[4] Suh, B. Han, I. (2003). The IS risk analysis based on a business model, *Information & Management* V. 41, p. 149–158, December 2003.

[5] Innerhofer–Oberperfler, F., Breu, R (2006). Using an enterprise architecture for it risk management, *In:* Proc. of the ISSA 2006 from Insight to Foresight Conference, Sandton, South Africa, July 2006.

[6] Leitner, A. (2009). ARiMA - A new approach to implement ISO-IEC 27005, *In:* Proc. of 2nd International Symposium on Logistics and Industrial Informatics, Linz, Austria, September 2009.

[7] Zambon, E., Etalle, S., Wieringa R. J., Hartel, P (2011). Model-based qualitative risk assessment for availability of IT infrastructures, *Software & Systems Modeling*, 10 (4) 553–580, October 2011.

[8] Loloei, I., Shahriari H. R., Sadeghi, A (2012). A model for asset valuation in security risk analysis regarding assets' dependencies, *In:* Proc. of 20th Iranian Conference on Electrical Engineering, (ICEE'12), Tehran, Iran, p. 763-768, May 2012.

[9] Breier, J. (2014). Asset valuation method for dependent entities, *Journal of Internet Services and Information Security* (JISIS), 4 (3) 72–81, August.