

SECOMO: An Estimation Cost Model for Risk Management Projects

Jihène Krichène, Noureddine Boudriga, Sihem Guemara El Fatmi

Communications, Network And Security Research Lab

Higher School of Communication

Tunis, Tunisia

jnk@certification.tn, {nab,sihem.guemara}@supcom.rnu.tn

Abstract— We present in this paper an estimation cost model for risk management projects, called SECOMO. This model helps managers reasoning about the cost and schedule implications of network security decisions that security teams may need to make. It aims to achieve several objectives including: (1) providing accurate cost and scheduling estimates for currently security projects, and (2) providing a normative method for the allocation of resources necessary for the development and maintenance of network security solution.

Index Terms—Network Security, Cost Estimation Model, Project Management, Risk Management.

I. INTRODUCTION

SECURITY cost estimation is important because it aims to provide accurate cost and schedule for current (and likely future) security solutions to organizations. It also enables security teams to easily recalibrate, customize, and extend the cost model the estimation may produce. An accurate cost-estimation capability provides security teams with a solid basis for determining how much time, cost and personnel each risk management process should take. This helps managers to plan the securing activities, to perform competitive security contract bids, and to tell whether or not a security project is proceeding according to plan.

Efficient security cost models should provide a normative mechanism to allocate the resources necessary for effective security solutions development and maintenance. It should be evolving to integrate new capabilities to address needs for protection. Moreover, security cost models should provide an efficient and easy to understand set of definitions of the inputs, assumptions and outputs required for the estimations.

Several estimation techniques have been proposed and used during the late decades. However, to our knowledge, these techniques have considered that security projects are aggregate of sub-projects, which may be addressed separately. Estimation techniques include COCOMO, COCOTS, COQUALMO, and Expert COCOMO (see [1] for a description of these models). These techniques have addressed the cost estimation of the development of software, constructive integration, expert-determined defect introduction and removal, and risk assessment, respectively.

We believe that a security cost model should be based on the joint estimation of the cost of a set of processes including, but not limited to, the following inter-related processes:

- 1) building systems with commercial-off-the-shelf solutions,
- 2) risk analysis and monitoring,
- 3) security quality assurance,
- 4) project planning and
- 5) security policy definition.

To this end, we have developed a security cost model that we describe in this paper.

The objective of this paper is then to propose an estimation technique security oriented. This technique, which is inspired from those used successfully in software engineering, aims to develop an estimation cost model allowing managers to estimate the effort needed to set up a security solution. Because of the similarity existing between the security engineering management and the software engineering management, we have chosen to found the development of security cost model on the **Constructive Cost Model** (COCOMO II version).

COCOMO, which represents a basis for the model presented in this paper, can be defined as an objective cost model for planning and executing software projects [1]. COCOMO refers to a parametric software cost model for planning and executing software projects. It supports bottom-up algorithmic model estimates. COCOMO advantages include generality, efficiency, and extensibility [1], and its computations are based on the estimation of a project's size.

Like COCOMO, the estimation model proposed in this paper, which we referred to as SECOMO, supports algorithmic model estimates. The estimation is made for the whole security project. Due to the lack of security data statistics, the model initialization is based on expert judgment. SECOMO is specific to security projects. It is defined as an objective model for planning and performing risk management projects in networked environments.

The remaining of this paper is organized as follows: Section 2 presents the SECOMO model, its estimating equations and the method used to estimate the network size. Section 3 details the effort multipliers and scale factors definition. In Section 4, the SECOMO methodology is described, the model is validated using a questionnaire submitted to a set of security experts. In this

section, the *a priori* model is also defined. Section 5 presents the model refinement. It defines the *a posteriori* model and shows how it is adjusted. Section 6 concludes the paper.

II. SECOMO DESCRIPTION

Security Cost Model (SECOMO) comes within the framework of risk management project in a telecommunication network developed¹ at the National Digital Certification Agency (NDCA, Tunisia). It is used to estimate the effort required to conduct a risk management project in a networked environment. The effort estimation serves as the basis of other tasks estimations such as the project's duration, the number of persons that will carry out the project, and the project's cost. These estimations are performed using the concept of network size and various parameters, called scale factors and effort multipliers, that give a measure of the security task complexity. The next subsection presents the underlying equations for effort in SECOMO. These are the effort, duration, manpower and cost equations. Subsection II-B also defines the notion of system size estimation and the parameters that are related to it.

A. SECOMO equations

SECOMO estimations are based on the effort required to secure a network and the similarity that the network security effort has with the effort estimation in the software development field. The notion of system size is however more complex to be addressed. This will be considered in the next subsection.

We chose to express the effort needed to conduct risk management projects by (1), which is expressed in *man * time - unit*:

$$E = a \times EAF \times S^b \quad (1)$$

where a is a constant, EAF is an Effort Adjustment Factor, $EAF = \prod_i EM_i$, the EM_i 's are the Effort Multipliers, S is the size of the network, and $b = \beta + \sum W_i$, where β is a constant and the W_i 's are the scale factors. Equation (1) is similar to the effort equation developed for the effort estimation in COCOMO. However, constants a and b , effort multipliers, and scale factors are different.

The duration D of a security project is a function of the effort estimated E . We chose to consider that the relationship between D and E is based on the fact the D should increase with E and that this growth is amplified by the scale factors. The duration of risk management projects can be then determined as follows:

$$D = c \times E^d \quad (2)$$

where c is a constant, E is the effort estimated previously, and $d = \delta + \sum W_i$ where δ is another constant. For the effort equation as well as for the duration equation, a , β , c and δ are constants used to calibrate the model.

The number of persons involved in the security project is therefore equal to the ratio between the effort and the duration

¹SECOMO was designed and developed with NDCA by the Communication Networks and Security Research Lab, University of Carthage, Tunisia.

as illustrated by the equation: $P = \frac{E}{D}$. The total cost of the risk management project is then estimated using the equation $C = MS \times P$, where MS is the Mean Salary of the team members.

B. Network size estimation

The size S of the security system is the basic parameter in the effort estimation. It plays in our model a similar role to the one achieved by the code size in the software development projects (used particularly in COCOMO). Two approaches can be applied to define S : while the first approach considers the number of the network components, the second considers the security tasks that are performed during the risk management project.

The determination of an estimation of the size using both approaches separately presents some disadvantages. Two problems can occur with the first approach:

- Because the network can be made of heterogeneous components, counting these components poses the problem of choosing the unit count.
- Because the network components affect differently the security effort, counting only the number of components may lead to inaccurate estimation.

The major disadvantage of the second approach is its inefficiency to really reflect the needed effort. For example, vulnerabilities scanning needs more effort when it is applied to 30-component network than to a 10-component network. The number of occurrences of a given task influences the required effort and should not be neglected.

However, the two approaches can be combined. For this reason, we have chosen to estimate the size of the network on the basis of the security tasks and network components at the same time. We have classified the security tasks in two categories: general and specific. General tasks are related to common network elements, while specific tasks are restrictive to several kinds of network elements. For the sake of simplicity that size S is given by the basic equation:

$$S = S_g + S_s \quad (3)$$

Where S_g refers to the general tasks and S_s refers to the specific tasks.

In this estimation, we consider both the nature of the task and its occurrence number. Notice that the security level required for the network and its components should be considered when estimating S , because the level of security has an affect on the effort. The security level is then considered in the S_g and S_s estimation.

1) S_s estimation: S_s is given by the following equations:

$$S_s = \sum_i \alpha_i v_{si} \quad (4)$$

$$v_{si} = \beta_i \sum_j n_{t_{ij}} t_{ij} \quad (5)$$

where i is a given component in the network, α_i is the security level of component i , v_{si} is the value of the tasks related to component i , β_i is the number of copies of component i in the network, t_{ij} is the value of the task j for component i , and $n_{t_{ij}}$ is the occurrence number of the task j for the same component i .

2) S_g estimation: S_g is given by the following equation:

$$S_g = \alpha \sum_k n_k v_{gk} \quad (6)$$

where α is the security level of the network, k is a given task provided through the network, n_k is the number of occurrences of task k , and v_{gk} is the value of the task k .

Once the network size estimation model is established, we have to define the network elements, the security tasks and their values and the security levels.

3) *Network elements definition*: The network is made up of three essential categories of components : network components, data, and human resources. Some elements have been defined for each category. Each element is specified by its name and description.

Network components take their values in {Routers, Hubs, Switches, Proxy, Wireless Equipments, Work Stations, Laptops, Servers, Firewalls, IDS and Other Security Solutions}.

Data take their values in {Databases, Archives, Security Strategy, Security Policy and Procedures}.

Human resources take their values in {Technical Personnel and Administrative Personnel}.

4) *Security tasks definition*: The security tasks definition was based on certain reports [3], [4], [5] published on SANS site² and the audit projects conducted by the NDCA. The tasks are classified into two categories: general and specific. General tasks are those related to common network elements. They take their values in {Information Gathering, Threats Identification, Security Strategy Conformance Validation, Risk Analysis, Countermeasures Proposal, Security Strategy Definition, Implementation and Reports Generation}.

Specific tasks are restrictive to several kinds of network elements. They take their values in {Configuration Analysis, File System Analysis, Logs/Alarms Analysis, Administration Practices Analysis, SQL Injection, Policy analysis, Access rights Analysis, Access rules analysis, Interface with other systems Analysis, Physical Security Analysis, Vulnerabilities Scanning, Skills Tests and Behavior Analysis}.

The task values are assigned according to the difficulty level of the task. Three difficulty levels are considered in SECOMO: easy, meanly difficult and difficult.

5) *Security levels definition*: Three levels of security are defined according to the security zones defined in [2]: low security level related to the public zone (level 1), medium security level related to the zone not open to the public but open to the company staff (level 2), and high security level related to the protected zone (level 3).

The effort estimation can not be made only by the means of the network size. It can also be influenced by other factors related to the analyzed system, to the security project, to the team members and to the security policy to be produced. All these factors will be defined in the next section.

III. SECOMO PARAMETERS DEFINITION

The aim of this section is to detail the factors that influence the effort estimation. We have found it useful to classify such factors in two categories: Effort Multipliers and Scale Factors.

A. Effort Multipliers Definition

The Effort Adjustment Factor (EAF) is the product of the Effort Multipliers ($EM_i, i = 1..n$). Some effort multipliers we have adopted are deduced from the COCOMO II model. They are classified in three categories: Product, Personnel and Project. Two other multipliers related to the security projects were introduced: Attack Frequency and Audit Frequency. A fourth category, called Information System, specific to SECOMO is also defined.

The Product category is related to the reliability and the reuse of the security project outputs i.e. the security policy and the generated reports. The Personnel category deals with a capability and the experience of the team members. The Project category involves the complexity of the security projects and the constraints imposed on these projects. The Information System category includes geographical distribution of the system and security informations related to the system.

Each effort multiplier is specified by its name, its description and its significance. These effort multipliers are ranked into five levels: very low (VL), low (L), nominal (N), high (H) and very high (VH). The following list specifies for each category its related parameters and their description:

- Product factors
 - Required Security solution Reliability (RELY) : This is a measure of the extent to which the solution must perform its intended function. It represents the error consequences on the network securing process.
 - Required Re-usability (RUSE) : This is a measure of the additional effort needed to construct components intended when certain previous results have to be reused.
- Personnel factors
 - Team Capability (TCAP) : Reflects the design ability, and efficiency of the team members, and their ability to cooperate and communicate.
 - Team Experience (TEXP) : This factor measures the level of experience of the team members in the field of risk management.
 - Platform Experience (PEXP) : Measures the knowledge and the experience of the team members in using the development platform.

² www.sans.org

- Tool Experience (TEX) : Measures the knowledge and the experience of the team members concerning the use of security tools.
- Personnel Continuity (PCON) : Reflects the annual turnover of the personnel involved in the risk management project.
- Project factors
 - Complexity (CPLX) : Quantifies the complexity of the security tasks that constitute the risk management project. For instance, active vulnerability analysis should have a greater weight than passive vulnerability analysis.
 - Use of Software Tools (TOOL) : Measures the level of improvements of the tools to develop the project, including capability, maturity and integration.
 - Required development Schedule (SCED) : Measures the schedule constraint imposed on the project team. Time constraints may increase the required effort. Accelerated schedule tends to produce more effort.
- Information System factors
 - Multi-site Information System (SITE) : Reflects the impact of geographical distribution on the effort. Security tasks, when applied in a distributed environment, need more effort.
 - Audit Frequency (AFRE) : Measures the rate of audit operations. This may impact the complexity of the effort needed to secure the network.
 - Attack Frequency (ATTF) : Measures the rate of attacks against the network to be secured. A high attack rate implies that many actions should be taken to fill in the numerous breaches.

As previously stated, the effort multipliers definition has concerned various characteristics of the analyzed system, which may not be detected in a first walkthrough, as it can be illustrated by some of the factors in the Personnel factors category.

B. Scale Factors Definition

We have chosen some of the Scale Factors (SF) defined in the COCOMO II model and we have introduced an other scale factor related to the security projects: the presence of a security strategy. Each scale factor is specified by its name, description and significance. These scale factors have the same levels as the effort multipliers. The following list describes the considered scale factors and gives their significance:

- Precedentedness (PREC) : Existent security assessment reports can be used to implement the current project and hence decreases the required effort. This factor measures the level of similarity of the project with previously developed projects.
- Team Cohesion (TEAM) : Measures the ability of the team members to work in group (e.g., experience in working together).
- Project Maturity (PMAT) : Reflects the level of maturity of

the project. The rate used is developed based on the schedule and the tasks to be performed.

- Security Strategy (STRA) : Measures the security strategy efficiency, the compliance of the current security procedures (if any) with the security strategy, and complexity of tests for compliance.

IV. SECOMO METHODOLOGY

SECOMO use in a given organization needs to follow 2-fold process: the system is initialized in the first phase, refined in the second phase. The model initialization is based on the expert judgment. A Delphi tour will be conducted in order to assign values to the different SECOMO parameters.

After the Delphi tour, the *a priori* model is defined. The model refinement combines the expert judgment and the statistical collected data. The *a posteriori* model is then defined using the Bayesian method which allows the combination of expert and statistical data. The model is then periodically adjusted with new collected security projects. These steps are summarized in waterfall-like model and described in Fig.IV.

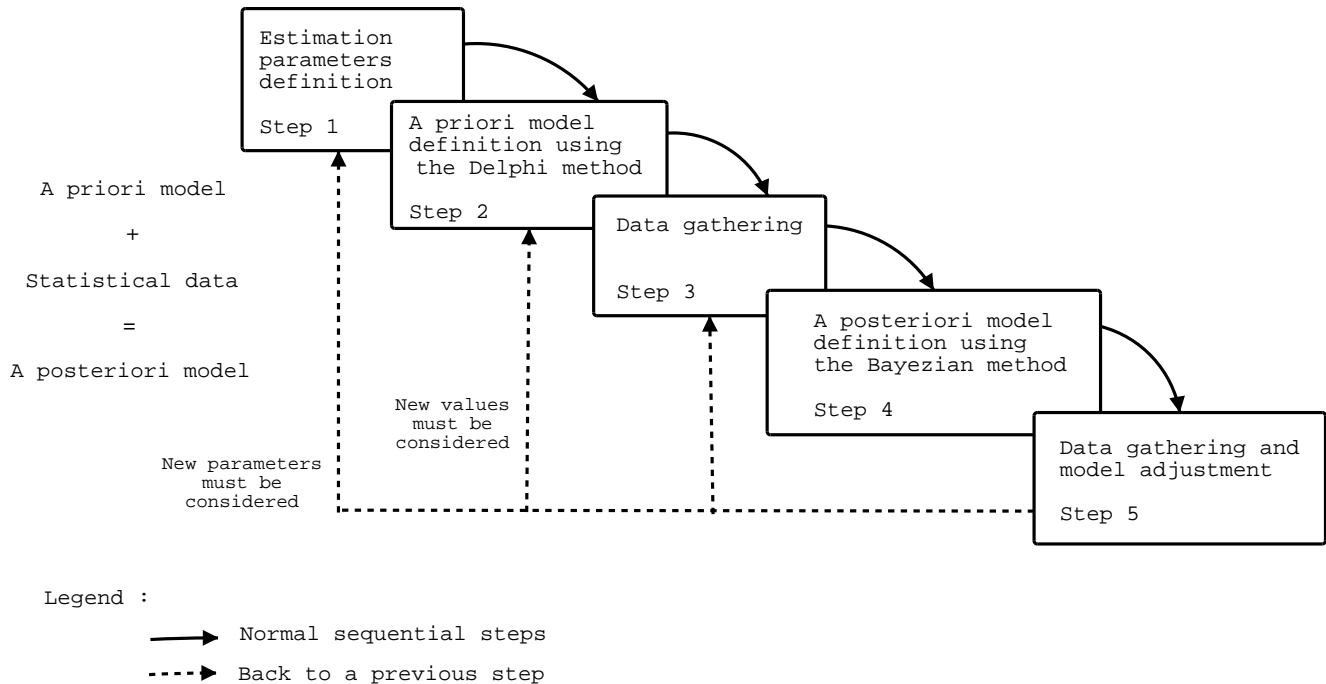
- **Step 1** : In the first step, are defined the different parameters that should be considered to estimate the security project cost. This step was described in the previous section.
- **Step 2** : The second step aims to initialize and validate the model based on the expert opinion. The initialization is made by the Delphi process which is one among the most popular methods.
- **Step 3** : Data gathering is performed in the third step. This activity is continuously realized in order to adjust the model.
- **Step 4** : The *a priori* model and the statistical data are combined during this step in order to perform the *a posteriori* model. In this step, the Bayesian method is used to allow the combination of the expert judgment and statistical data.
- **Step 5** : Security data gathering is continuously performed in order to adjust periodically the *a posteriori* model using the previous *a posteriori* model and the data newly collected.

The steps described above are performed in a sequential way. However, certain flashbacks are planned in order to refine the model definition, when needed. For example, the consideration of new parameters leads to a return to the first step. The integration of new values for the parameters under consideration may lead to a return to the second step, as well as.

In the following, two important aspects involved in above steps will be addressed. The first one concerns the model validation and the second is related to the *a priori* model definition.

A. Model and questionnaire validation

The *a priori* model will be refined by security experts through a Delphi questionnaire that permits the parameters initialization. We have referred to expert judgment for the definition of these parameters and for the establishment of the questionnaire that is used for the initialization procedure.



SECOMO methodology

To achieve this, a survey has been performed in order to validate the different model parameters and a questionnaire has been developed and distributed among a selected set of experts. We found that the experts critics have targeted two essential points: the network size estimation and the significance of the values assigned to the EM and SF levels.

Concerning the network size estimation, the experts have noticed the absence of the proxy and wireless equipments as network components. They also proposed to consider two sub-categories of data : informational and organizational data. As informational data, they consider databases and archives. As organizational data, they consider the security strategy, the security policy and the procedures. These critics have been taken into account. The sets presented in II-B.3 have been modified consequently.

The experts have also mentioned the absence of two security tasks in the network size estimation. The first task is classified as general and is called “Security Strategy Conformance Validation”. The set of general tasks presented in II-B.4 was then modified consequently. The second task is classified as specific and consists in analyzing the access rules. The set of specific security tasks presented in II-B.4 has then been updated.

Concerning the effort multipliers and the scale factors, the experts have mentioned some remarks related to the values assigned to the different levels of these factors. They all agreed that the numeric values must be assigned as intervals. The updated values related to the effort multipliers according to their levels (VL, L, N, H and VH) are given below:

- RELY takes its values in {lossless, low losses, moderate losses and high losses}. For example, if RELY takes its

value as “lossless”, this means that a minimal level of precision is required.

- CPLX takes its values in {passive analysis and active analysis}. For example, if CPLX takes its value as “active analysis”, this implies that an active analysis must characterize a set of security activities (e.g. penetration tests must be performed in the case of vulnerability analysis task).
- RUSE takes its values in {undocumented results and documented results}. For example, if RUSE takes its value in “documented results”, this means that the analysis results must be documented for a reuse intention.
- SCED takes its values in {without temporal constraints and with temporal constraints}. For example, if SCED takes its value in “with temporal constraints”, this states that the security project achievement is subject to severe temporal constraints.
- SITE takes its values in {mono-site system and geographically distributed systems}. For example, if SITE takes its value as “geographically distributed systems”, this means that the team members will have to move from a site to another, which increases their effort.
- TCAP, PCON and TOOL are expressed in percentage. Their values belong to fixed intervals. For example, if we consider the TOOL parameter, we find five intervals respectively corresponding to (VL, L, N, H, VH) and defined by ([0%, 20%[, [20%, 40%[, [40%, 60%[, [60%, 80%[and [80%, 1000%]). If this parameter takes its value in [80%, 1000%], this means that more than 80% of the security tasks are performed using automated tools.
- TEXP, PEXP and TEX are expressed in terms of duration.

Their values belong also to fixed intervals. For example, if we consider the TEXP parameter, we find five intervals respectively corresponding to (VL, L, N, H, VH) and defined by $([0\text{ months}, 6\text{ months}]$, $[6\text{ months}, 1\text{ year}]$, $[1\text{ year}, 3\text{ years}]$, $[3\text{ years}, 5\text{ years}]$ and up to five years). If this parameter takes its value as “up to five years”, this means that all the team members have been involved in security projects for more than five years.

- AFRE is expressed in terms of temporal frequency (n times duration) of the security audit. For this parameter, we define five intervals respectively corresponding to (VL, L, N, H, VH) and stating that the audit is performed only once, once every three years, once a year, and once a semester, respectively. If this parameter takes its value as “once a semester”, this means that a security audit has to be performed every three months.
- ATTF is expressed in terms of attacks number per day. For this parameter, we find five intervals respectively corresponding to (VL, L, N, H, VH) and defined by $([0\text{ attacks/day}, 250\text{ attacks/day}]$, $[250\text{ attacks/day}, 500\text{ attacks/day}]$, $[500\text{ attacks/day}, 750\text{ attacks/day}]$, $[750\text{ attacks/day}, 1000\text{ attacks/day}]$ and more than one thousand attacks per day). If this parameter takes its value as “more than one thousand attacks per day”, this means that a analyzed system can be subject to more than one thousand attacks a day.

The updated values related to the scale factors according to their levels (VL, L, N, H and VH) are given below :

- PREC takes its values in {without anterior reports and with anterior reports}. For example, if this parameter takes its value as “with anterior reports”, this means that the results located in anterior reports can be used.
- STRA takes its values in {without security strategy and with security strategy}. For example, if this parameter takes its value as “with security strategy”, this means that a security strategy must be evaluated and the Security Strategy Conformance Validation task should be performed.
- TEAM and PMAT are expressed in percentage. Their values belong to fixed intervals. For example, if we consider the PMAT parameter, we find five intervals respectively corresponding to (VL, L, N, H, VH) and defined by $([0\%, 10\%]$, $[10\%, 30\%]$, $[30\%, 50\%]$, $[50\%, 70\%]$ and $[70\%, 100\%]$). If this parameter takes its value in $[70\%, 100\%]$, this means that more than 70% of the security tasks have yet been performed.

The validation process performed with the help of security experts, allowed us to make some changes concerning the network size estimation method (e.g. components addition), and the effort multipliers and scale factors (value updates).

B. A priori model

SECOMO initialization is based on the expert opinion, because of the lack of statistical security data³. During this step, we use the Delphi method [6] which is the most used method in the fields of science and technology. The Delphi process aims to obtain the most reliable consensus among a group of experts by a series of questionnaires submitted over a series of rounds with controlled feedback. The a priori model proceeds as follows:

First round:

- 1) Submit a questionnaire to the expert panel.
- 2) Receive the experts responses.
- 3) Check the responses validity.
- 4) Analyze the given responses.

Next rounds:

- 1) Submit a questionnaire based on the results of the previous round to the expert panel.
- 2) Repeat the steps number 2, 3 and 4 of the first round.
- 3) If the opinions converge to a final result, then stop the Delphi process, else return to the first step.

The three latest steps presented in the SECOMO methodology need some additional precisions concerning the *a posteriori* model definition and the model adjustment process. This will be done in the following section.

V. MODEL REFINEMENT

A. A posteriori model

Security data are gathered at organizations that conduct an auditing activity. A questionnaire is used in order to gather the information needed in the *a posteriori* model definition. A wide variety of audit projects will be considered in order to not be limited to a certain organizations category. We choose to consider audit projects conducted at different sized organizations, acting in different fields and having different levels of public network use.

Three organizations sizes are considered: little, mean and large. Three activity fields are also considered: activity field lowly concerned by security, activity field meanly concerned by security, and activity field highly concerned by security. Three levels of public network use are defined: no public network use, public network use for advertising purposes and full public network use.

The *a posteriori* model definition is based on the Bayesian approach [7] which allows the combination of the *prior* information (*a priori* model) and sampling information (from security data gathering). Using Bayes’s theorem, we can combine our two information sources as follows:

$$f(\beta|y) = \frac{f(y|\beta) * f(\beta)}{f(y)} \quad (7)$$

where β is the vector of parameters in which we are interested, and y is the vector of sample observations from the joint density

³Data information of risk management projects will be collected in the future in order to adjust the model.

function $f(\beta|y)$. In (7), $f(\beta|y)$, the posterior distribution for β , summarizes all the information we have about β and $f(\beta)$ represents the *prior* information obtained from experts.

In the Bayesian context, the *prior* probabilities are “unconditional” to the sample information, while the *posterior* probabilities are the “conditional” probabilities, given sample and *prior* information.

This approach favors the experts opinion if they are in strong agreement and the statistical data is weak, and favors the statistical data if they are strong and the experts have disagreed. The Bayesian approach provides then an optimal combination of the two sources of information. According to the Bayesian analysis, the *posterior* variance is defined as:

$$Var(b^{**}) = \left[\frac{1}{s^2} X'X + H^* \right]^{-1} \quad (8)$$

where b^{**} is the *posterior* mean, X is the matrix of predictor variables, s^2 is the variance of the residual for the sample data and H^* is the precision of the *prior* information. Equation (8) can be written as follows:

$$\frac{1}{Var(b^{**})} = \frac{1}{s^2} X'X + H^* \quad (9)$$

As it can be seen in (9), the *posterior* precision will always be greater than the *a priori* precision or the sample data precision since it is equal to their sum.

B. Model adjustment

The model adjustment consists in collecting the security data continuously in the purpose of calibrating the model. Therefore, the following steps are performed:

- 1) Consider the last calibrated model as *a priori* model.
- 2) Insert the newly collected data in the security database.
- 3) Use the *a priori* model and the gathered data in order to adjust the parameters using the Bayesian technique and define then the new *a posteriori* model.

The problem that is encountered here deals with the adjustment periodicity. This periodicity can be function of time or projects' number. If we consider a temporal periodicity, new security data may not be available during the fixed period. If we consider a periodicity based on the realized projects' number, we may take long time before adjusting the model which must be avoided because it will have a negative consequence on the estimation precision. In order to overcome these problems, we have opted for a combined solution defined as follows:

- The model adjustment starts after the realization of p projects.
- If a period of y months has passed without reaching the fixed number of projects, then the model is calibrated using the available data.

The model adjustment allows us to have the highest estimation's precision since it can update the model parameters based on performed risk management projects.

VI. CONCLUSION

We developed in this paper a security cost model to help security solutions developers reasoning about the cost and schedule implications of network security decisions that may need to make. This model, which we referred to as SECOMO, aims to provide accurate cost and scheduling estimates of security projects. SECOMO is based on the network size and several parameters (e.g. effort multipliers and scale factors). The model was validated by a team of security experts. An *a priori* and an *a posteriori* model are defined for SECOMO. The first one is based on expert judgment and aims to initialize the model. Whereas, the *a posteriori* model combines expert opinion and sampling data and offers a higher precision. SECOMO is adjusted periodically so that cost and scheduling estimates are maintained accurate.

Future work with SECOMO will address task scheduling (fixing the problem of optimizing the tasks' allocation for the team members during the realization of security project), security assurance quality (defining a set of parameters aiming to measure the security quality), and data information collection of risk management projects for various types of organizations and business activities.

REFERENCES

- [1] Barry W. Boehm, Chris Abts, A. Winsor Brown, Sunita Chulani, Bradford K. Clark, Ellis Horowitz, Ray Madachy, Donald Reifer, and Bert Steece, “Software Cost Estimation with COCOMO II”, Prentice Hall, 2000.
- [2] Odd Nilsen, “Protection of information assets”, Technical report, SANS Institute, March 2002.
- [3] Michael Roberti, “Building an Enterprise security architecture”, Technical report, SANS Institute, April 2001.
- [4] James Bayne, “An overview of threat and risk assessment”, Technical report, SANS Institute, January 2002.
- [5] Ted Mina, “Application security, information assurance's neglected stepchild - a blue-print for risk assessment”, Technical report, SANS Institute, July 2001.
- [6] O. Helmer, “Social Technology”, Basic Books, New York, 1966.
- [7] E. E. Leamer, “Specification Searchers, Ad hoc Inference with Non experimental Data”, Wiley series, 1978.