Dynamic Aspects of Security Management of Information System.

Abstract

The dependency of enterprises on information systems makes security of information systems one of the mayor concerns for enterprises. An incorrect management of these information systems can increase the number of vulnerabilities in an enterprise, becoming sensitive to problems and attacks. By presenting and analyzing a vulnerabilities model, this paper provides insights to the problem that poor security management combine with vulnerabilities can harm an information system. By implementing robust *Technical Controls* (mechanisms that protect the system from incidents or attacks), *Formal Controls* (business structures that allow a proper use of tecchnical controls), and *Informal Controls* (security controls that deal with the workforce), vulnerabilities can be eliminated improving security management of information systems. These security controls could minimize the risk of security failures originated by the existence of vulnerabilities on the system.

Introduction

In today's world where enterprises depend on information systems and also where technology is increasing exponentially, poor information technologies security (ITS) can have humongous economic impacts on enterprises. In the 70´s software developing and computer-based information systems occupied a high priority ranking for computers experts and engineers. In 1978, Boehm classified software quality characteristics into four levels organized by hierarchy with seven main characteristics (Boehm, 1978). Then in 1993, Davis provided extensive discussion on each of the seven characteristics included at the third level of the hierarchy (Davis, 1993). Even though this classification was accepted and adopted, it is important to realize that security was never mentioned among these software quality characteristics. Recently, security has been included in similar classifications. Lawrence Chung and Brian A. Nixon pointed that some non-functioning requirements such as accuracy, security and performance were extremely important for the software quality classification (Chung and Nixon, 2000). Security nowadays has become one of the most important software quality characteristics.

These facts mentioned above are an incentive to put time and effort on this project. This paper tries to explain how technical, formal and informal security controls, which will be explain later on, can decrease the number of vulnerabilities in an information system making it more robust and secure. Even though security is a non-functioning requirement, it is indispensable because the quality and high performance of an information system depends on the quality of both functioning and non-functioning requirements. (Chung and Nixon, 2000)

Before to continue on, it is necessary to define what does security mean and why it is so important. Security according to Bruce Schneier, is about preventing adverse consequences from the intentional and unwarranted actions of others (Schneier, 2003). Attacks are one of the main responsible for the existence of security. An attack can be defined as a series of steps intended to result in something that is not authorized to happen (Firesmith, 2003). By the use of tools, the attacker exploits a vulnerability in order to achieve an unauthorized result. (Howard, J.H, Longstaff, T.A, 1998)

Superstitious learning behavior is another reason that can explain why a good implementation of technical, formal and informal security controls is crucial in order to achieve high level of security. People develop strong, but false and often harmful believes (Sterman, J. 1998). For example, let us assume that in a given enterprise, every member of a group is required to produce backups once a week about information processed during that time. Management rarely keeps track of these backups, and workers are not doing any. Because of these superstitious learning behaviors, workers´ idea about the low importance of backups is reinforced more and more and therefore the system becomes vulnerable. These psychological behaviors produce the absence of these backups, and security gets compromised. In case of an attack or a problem, the data of this enterprise could be lost forever.

Developing a vulnerability model, which contains all the variables involved in managing security and their interrelations could be helpful to gain a better understanding about this problem.

Problem Analysis

In some corporations security systems and security controls are left in a static state after their implementation. Static state is a situation where security is unattended and not updated, believing that security work properly by itself. Corporations keep experiencing security problems and being exposed to successful attacks because of the implementation of security controls (hardware or software) on enterprises is in a static state.

It appears to be a common factor among enterprises that have suffered attacks. This common factor is the fact that the security administrators and managers of these enterprises have undeveloped ability to detect vulnerabilities. This undeveloped ability hides the size, the number, and the potential impact that these vulnerabilities can have on the protected system. This behavior leads to a feeling of being secure when in reality the information system is weak and vulnerable. Small amount of detected vulnerabilities, in combination with a static state of security, does not alert security administrators. The opposite situation can also seen, the bigger the amount of detected vulnerabilities, the worrier security administrators and managers become about security and therefore, it is clear that this behavior or positive feedback loop has a great impact on how security works in a given enterprise.

In order for enterprises to detect vulnerabilities, eliminate them, and achieve high level of security, it is essential to ensure high quality implementation of technical, formal, and informal security controls. The hypothesis of this paper, tries to explain how security of information systems is the product of these three security controls. In order to achieve high quality of security, these controls have to be perfectly implemented but first, it is necessary to define these security controls.

- ✓ Technical security controls are any type of software or hardware that a corporation can purchase or develop in order to secure their goods. Mechanisms that protect the system from incidents or attacks: Antivirus software, access controls, backups, recovery and audit software. (Melara, et al. 2003).
- ✓ Formal controls are any type of training or understanding processes that ensure that administrators and users of these technical controls know how they work and their reason for them being there. They are basically, business structures and

processes that ensure the correct general conduct of a business and reduce the probability of an incident or an attack. (Melara, et al. 2003).

- ✓ Finally, informal controls are security controls that deal with the workforce, their culture and believe on the system. These informal security controls are well implemented if and only if the people understand management's intentions and so, they become more committed with their responsibilities. In other words, how well the workforce follows the formal security controls established.

If one of these security controls is not implemented and followed up properly, then the security of a given information system can be endanger. For example, having an antivirus, well implemented but users do not update it, then the probability of the system getting infected becomes very high. The same scenario can be seen if the enterprise buys the best antivirus but it is installed incorrectly. These controls represent a security chain. As soon as one of these controls becomes vulnerable, then the chain can easily be broken. Security of information systems in corporations is only as good as the weakest link in its global network.

By utilizing a vulnerabilities model, we are trying to see the dependency of security on these three security controls and also to see if the performance of security of enterprises can be estimated and evaluated by multiplying technical, formal and informal security control policies. The formula,

Security = technical*formal*informal,

proposes the following reasoning: by setting values on a range from zero to one, if in a corporation, one of these 3 values equal zero, then the security will equal zero as well. Zero means non effective or not well implemented security controls and one meaning optimal implementation of security controls. If one of these three values approaches zero, meaning that policies are not being well followed or implemented, then the security of the given enterprise is taken as poor or none.

As an example, let us assume that a given enterprise purchases the latest antivirus on the market. In addition, this enterprise takes the time to hire an expert to set the antivirus in the system. However, they do not invest money on explaining to people the risk that opening attached files represents and neither the need of frequently update the antivirus version. The information system has become highly exposed to the latest viruses

because users could not update the antivirus. Then the security level of the information system in this enterprise can approximately be calculated by the product of (0.9*0.8*0.1) = 0.072. This value of 0.072 indicates that even though the enterprise purchased a high quality antivirus and hired an expert to install it, the level of security is almost none.

Now, the same enterprise has purchase an excellent backup device. It has good technical security controls but, poor formal security policies due to the fact that almost no workers know how to use it correctly. Then the security level of that enterprise can be represented as (0.6*0.1*0.9) = 0.054. This value can also be seen as a poor and vulnerable information security system.

Economical aspect

Researching and experimenting with the economical aspect of security could be very interesting in reaching a better understanding about vulnerability problems in enterprises. The greater the budget invested on security the safer and more productive the system could be, as long as the system is updated and monitored in a daily basis.

Not all enterprises require the same level of security. The optimal way to manage security is to find an equilibrium between security and money invested. Too much security can be a problem. It can interfere with daily work activities affecting productivity. A good way to find the optimal security level can be done by properly implement technical security controls, having knowledge of how these technical resources work and finally to continuously implement informal security control policies. Security is not about trying to reach the value of one according to the formula mentioned above. Security is about finding the optimal level (an intermediate value between 0 and 1) that ensure the safety of the information system and also that do not interfere with the productivity of the enterprise.

One of the main goals of this work is to provide an understanding of the dependency of security on these three security controls. Through the implementation and enforcement of formal and informal policies, proper security can be achieved and vulnerabilities can be eliminated. Also it can be a good opportunity to analyze the relations between money invested in security and productivity of the enterprise.

Causal Loop Diagram

The following causal loop diagram has three controlling loops and two positive feedback loops. *Technical Innovations* as well as *Process Automatization* increases the number of new information technologies and the number of updates. The increase of these two variables creates *New Technical Vulnerabilities.*

The first controlling loop starts when an increase of *Technical Vulnerabilities* increases the fraction of the security budget called in this vulnerabilities model *Expenses in Fixing Technical Vulnerabilities.* Managers' attention is raised due to this increase and therefore, *New Technical Controls* are implemented and *Technical Vulnerabilities* are decreased or eliminated.

New Technical Controls, originates *New Formal Vulnerabilities* and the behavior of the second and third controlling loops are similar to the behavior observed in the controlling technical vulnerabilities loop.

The total budget designated to security is divided into three non equal fractions in order to satisfy the three expenses to fix technical, formal and informal vulnerabilities. These relationships create two positive feedback loops.

The first positive feedback loop, *Improving Technical Controls through formal controls*, starts when an increase of *Formal Controls,* increases the *Auditing and Measurement Capabilities.* Because of this auditing and measurements, more vulnerabilities are detected, and therefore the *Commitment to Security* is greater. This commitment makes the S*ecurity Budget* to increase as well as the fraction designated to fix technical vulnerabilities. With the increase of this fraction, there is an increase of N*ew Technical Controls.*

The second positive feedback loop, *Detecting through Commitments*, starts the same way than the first positive feedback loop with the difference that the *Security Budget* increases the fraction of the budget to fix formal vulnerabilities. The increase of this fraction creates more *New Formal Controls* and therefore, there are more *Formal Controls,* which increases the overall security level.

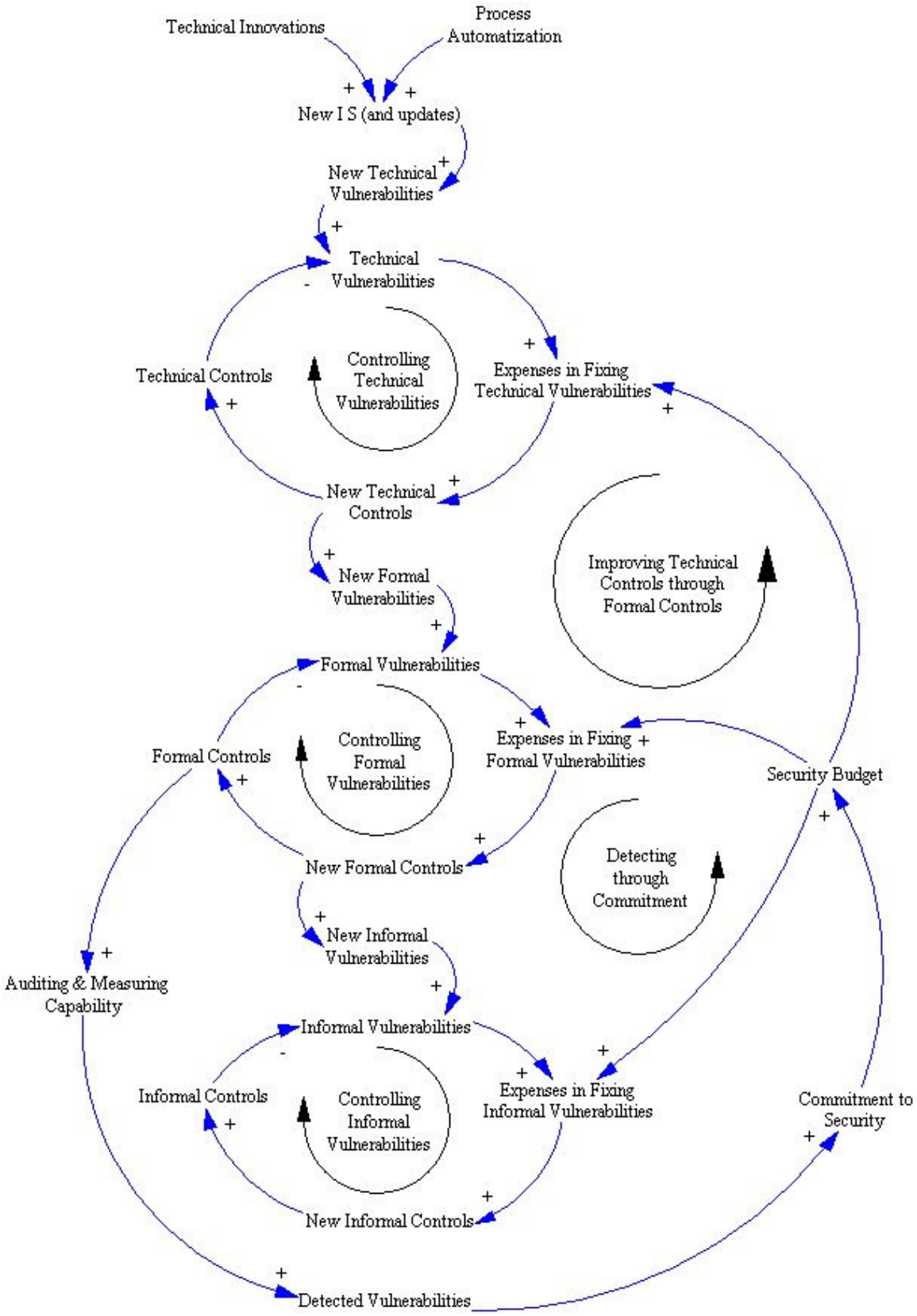


Figure 1: Causal loop diagram

Stock and Flow Diagram (Vensim Model)

The following vulnerabilities model has three subsystems and each one represents the process of technical, formal, and informal vulnerabilities elimination. In the beginning of the first subsystem, *New Technical Vulnerabilities* appear due to *Technical Innovations* and *Process Automatization*. Then, these vulnerabilities stay in the *Vulnerabilities Waiting for Verification* level for a period of time, until they get verified to see how much impact they can have on the information system. In order for corporations to realize and separate vulnerabilities with impact from vulnerabilities without impact, an analysis is required and money needs to be invested *(Expenses for Vulnerabilities analysis and Expenses for Vulnerabilities Analysis without Impact).* After this analysis, *Possible Technical Vulnerabilities with Impact* go to the next level where they wait for further analysis.

A second deeper analysis is required because; out of all possible impact vulnerabilities, the enterprise has to separate, existing real vulnerabilities from possible vulnerabilities that do not affect the system. These vulnerabilities that security administrators think the system has but it does not, are called false positives. After this differentiation, a fraction of the security budget will be designed to eliminate detected vulnerabilities (*Expenses to Eliminate Detected Vulnerabilities).*

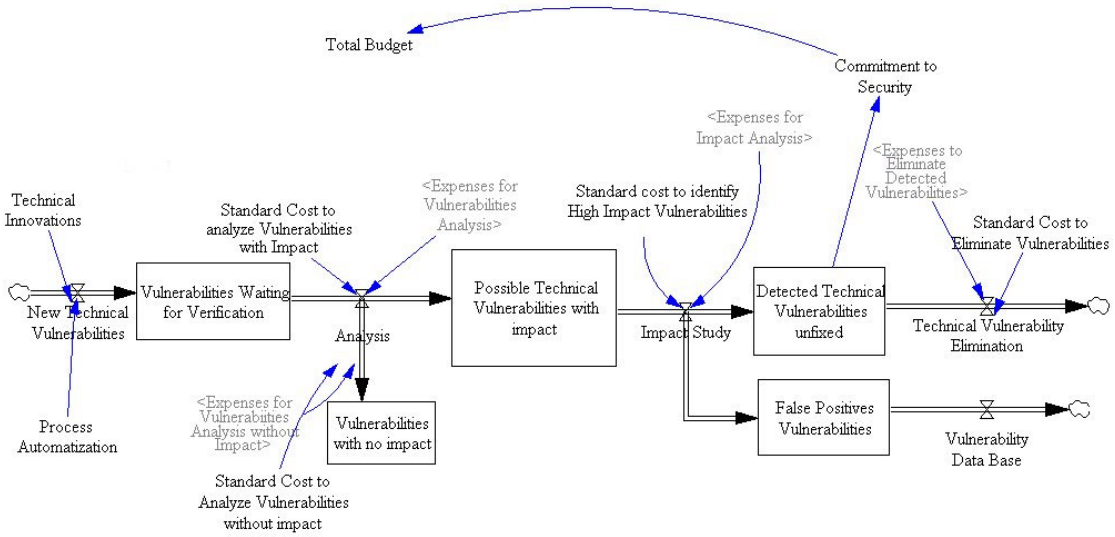


Figure 2: Technical subsystem

The same approach was utilized to represent the second level of the model, the formal security controls. In enterprises, the detection of *Technical Vulnerabilities*

produces *New Formal Policies Needs*. These new formal policies enter the *Unimplemented Formal Security Policies* level and the formal policies that are needed, are implemented with a cost that is calculated with a variable called *Expenses to Implement Formal Policies*, (a fraction of the security budget). After a certain period of time, these formal policies become obsolete and then they are eliminated.

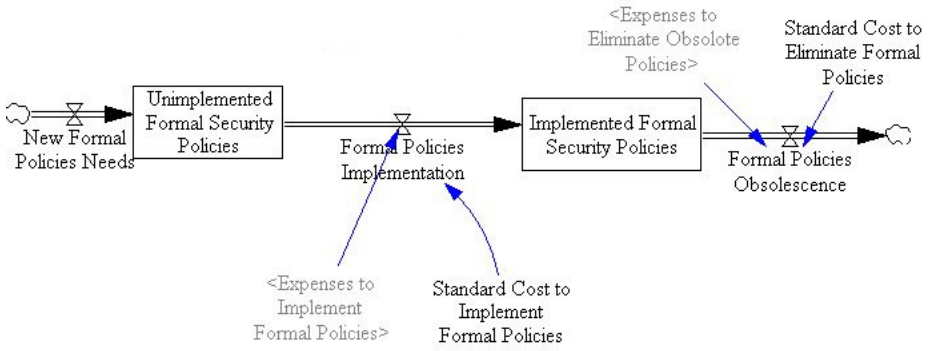


Figure 3: Formal subsystem

Informal security controls and formal security controls are represented into this vulnerability model somewhat similar. As mentioned above, in order for security to efficiently work, these three security controls have to be implemented and dynamically updated. When *Formal Policies* are implemented, *New Informal Policies* are going to be required. Those *Unimplemented Informal Security Policies* that apply to the enterprise are implemented based on the same procedure utilized for formal controls.

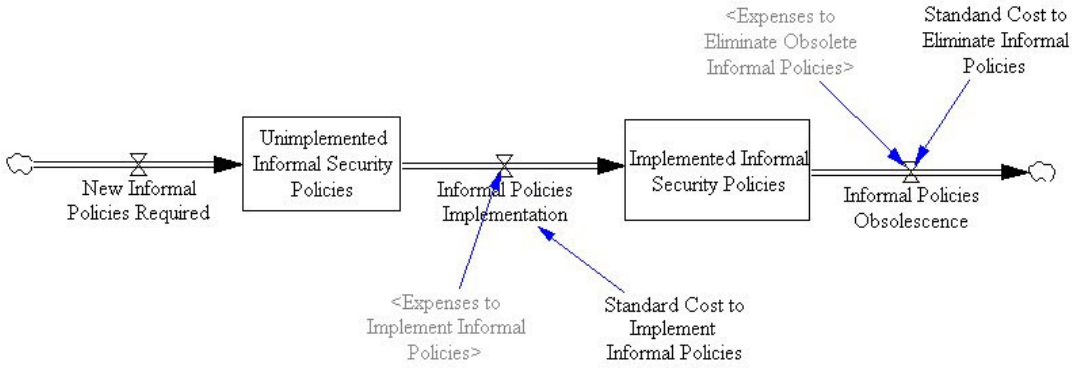


Figure 4: Informal subsystem

Project Future and Objectives

Currently with some help provided by s21sec (www.s21sec.com, a company specialized in vulnerability detection, located in San Sebastian, Spain), this model is being analyzed to possibly in the future create system's vulnerabilities learning environments. Creating learning environments that illustrate different security scenarios, could provide the opportunity to experiment with the interrelations between economic and security variables. Also, this vulnerability model could be incorporated into a corporation in order to eventually improve security decision making skills.

This model is in an early developing stage. However, by creating a robust and calibrated model, it could be possible to see the close relationship between good implementation of technical, formal and informal security controls, and high level of information system security.

Conclusions

The absence of a correct management of the information systems security could create big problems to firms, which have become highly dependent on their information systems. But firms still have not built robust structures to protect efficiently their systems. The reasons for this behavior are not clear yet.

It is necessary to understand why firms usually underestimate the risk that they are exposed to and why they have difficulties to adopt a broader perspective about security, including not only technical aspects but also formal and informal security aspects.

Security of information systems is a complex problem, involving many highly interrelated variables and delays. Systems dynamics might be a very useful perspective to analyze these relations, as it is able of capture both structural and dynamical complexity.

References

Boehm, B. W. 1978. *Characteristics of Software Quality*. NY: Elsevier North-Holland.

Chung, L, Nixon, B. 2000. *Non-Functional Requirements in Software Engineering*: Kluwer Academic Publishing.

Davis, A. M. 1993. *Software Requirements: Objects, Functions and States*. Englewood Cliffs, NJ: PTR Pretince-Hall.

Firesmith, D. 2003. Common Concepts Underlying Safety, Security, and Survivability Engineering. Pittsburg, PA: Carnegie Mellon University.

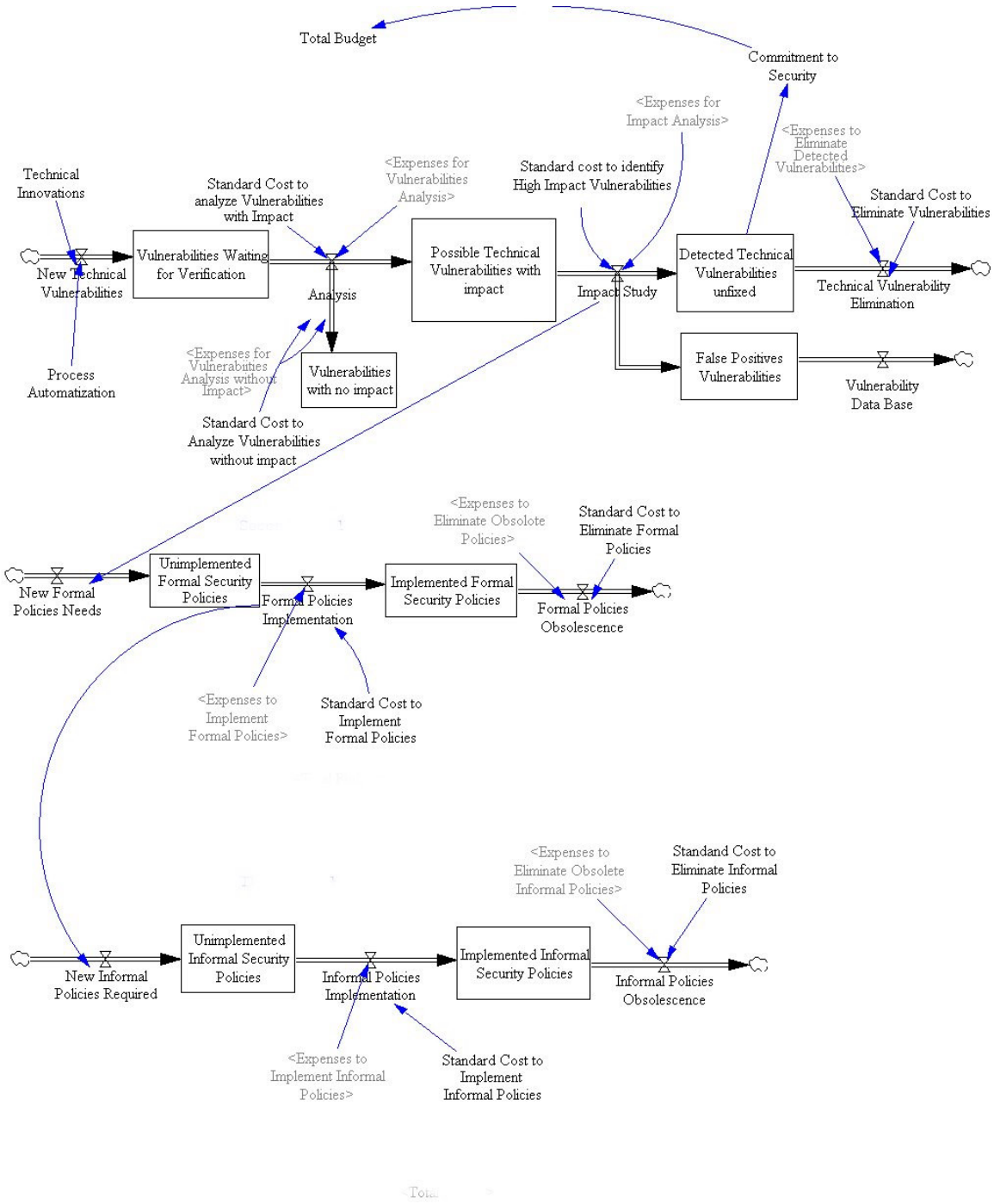
Howard, J.D, Longstaff, T.A. 1998. A common Language for Computer Security Incidents. Albuquerque, New Mexico: Sandia National Laboratories.

Melara, C, Sarriegui, J.M, Gonzalez, J.J. 2003. *From modeling to managing security*. Norway: Hoyskoleforlaget.

Schneier, Bruce. 2003. *Beyond fear*. 1 ed. New York, NY: Copernicus Books.

Sterman, J, 1998, Supertitious Learning (in *Organizational Learning at Work*), Pegasus Communications.

Appendix



Back to the Top