Computer Systems & Information SSR: a Unified Approach for Decision Making

Dov Shirtz, Zigmund Bluvband, Yuval Elovici, Peretz Shoval

Abstract

Safety, Security and Reliability of complex systems are the three interacting and most important risk related factors. In many cases of failure event, the Security function assumes charge, and manages the failure event and its resolution. But does the Security function consistently apply the optimal failure resolution methods? We propose that several organizational functions, including Information Security (IS), should analyze, manage, and resolve each case of failure in a coordinated effort, based on the failure classification and prioritization, and then apply appropriate Corrective Actions (CA). Such coordination may result in applying a CA that is sub-optimal by Security standards, yet optimal from the organization's perspective. In this paper we present an innovative composite methodology for identifying, prioritizing and selecting failures and incidents for appropriate treatment. The methodology is based on organizational priorities, knowledge and considers the analyses results of End Effects (EE), solutions and CAs.

1. Introduction

Safety, Security and Reliability (SSR) are the three most important interconnected risk-related elements of any software or hardware system, or any combination of both (Erland, 1998; Olovsson, 1992; Avizienis, 2004). Modern Information Security (IS) is a part of the total security system linked to information safety, software quality and reliability (Erland, 1998; Avizienis, 2004).

Many organizations strive to provide "24 x 7" up-time for different types of services, implying that no substantial failures occur, or any failure is repaired as quickly as possible (Doolin, 2003; Brown, 1999; Sheldon, 2005).

Organizational considerations center on finances and therefore, organizations must decide how to allocate their resources to provide optimal organizational benefits. In this study we assume that the correct prioritization of failures occurring in one of the organizational ICT infrastructures will increase organizational benefits. Adequate preparation to failures occurrence entails application of failure analysis methods, failure causes prioritization techniques methods of identification of all possible End Effects (EE induced by a specific failure. In addition, decisions must be taken on how to prepare the system to prevent the most damaging failures and implement recovery processes after failures occur (Bluvband, 1999) (see figure 1).

Kume (1985) describes five error-handling principles: Elimination, Detection, Replacement, Facilitation and Mitigation. From the organization's perspective, elimination of failures is the best principle. However, this is not always a feasible goal in the software industry, where business and trade considerations, such as capturing a larger market share, demonstrating innovativeness, etc, drive software manufacturers to introduce new software products, or new services based on software, before de-bugging

has been completed. Therefore, Detection, Replacement, Facilitation and Mitigation are the more conventional ways of error handling in software engineering.

It seems that the issue of decision-making for Failure Mode (FM) handling and identifying Failure Cause (FC) with respect to IS is somehow less popular then other IS topics. Some of the reasons for this are:

- Failure Analysis (FA) is a tedious and time-consuming task.
- Usually FA is not a regular process of system maintenance.
- There are always more "urgent" matters than FA and FC analysis that require attention.
- Fixing failures as soon as possible is perceived to be more important than allocating resources to an improvement process.

Edgington (2004) concludes that using FC analysis to build an organizational knowledge base increases the throughput of the organization. Increasing throughput is a consequence of a managerial decision process that distinguishes between two types of failures: failures for which an improvement processes is required, including application of FA methods and discover remedies to FC in advance, and failures which will be repaired as they emerge (and regarding which no improvement process is undertaken).

Fixing or repairing a failure without FC analysis is defined as a *reactive process*, while analyzing a failure, identifying its FC, and implementing corrective actions is generally defined as a *proactive process* (Grady, 1996). Notably, reactive actions do not reveal failure causes; only proactive actions set us on a track to discover the FC.

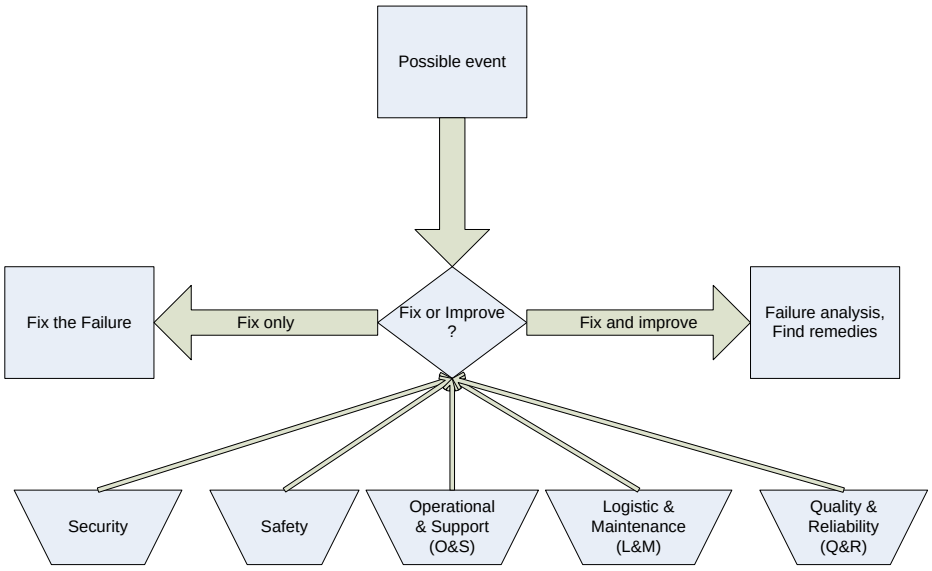

Figure 1: *A simple decision model for handling a failure*

Decision making processes can be classified according to the extent of knowledge available to decision makers: "decision-making under certainty," "decision-making under risk," and "decision-making under uncertainty." (Taha, 1987).

"Decision-making under certainty" is a term used to describe decision-making, which assumes that we have perfect knowledge about the future. The two remaining categories refer to the availability of partial or imperfect information. In "decision-making under risk," the degree of ignorance can be expressed in terms of probability density functions. In contrast, "decision-making under uncertainty" refers to problems with large components and substantial interactions among those components, or systems that are so complex that they defy understanding in entirety (Austin, 1993). Another view is to look at uncertainty as a property of our knowledge about an event (Winterfelt, 1986). From this perspective, "decision-making under risk" lies between the two extremes of certainty and uncertainty.

IS decisions are, by definition, decisions made under risk or uncertainty. To make decisions under risk, probabilities for IS failures occurrence are calculated (Ryan 2005,

Ekenberg 1995). However, Finne (1998) states that IS decisions are better characterized as decision making under uncertainty, and therefore require the application of conservative uncertainty techniques for decision making (Taha, 1987), or the use of modeling techniques, such as the Monte-Carlo approach, which capture uncertainty in security modeling parameters such as probability, rate of events, etc (Conrad, 2005). Game theory technique is another method for decision making under uncertainty In IS. The “game” is based on the benefits and costs for an "attacker" vs "defender" on the organizational networks or outer network (Cavusoglu, 2004; Alpcan, 2003).

The methodology proposed in this paper is a compound method, which adopts Safety, Security and Reliability (SSR) perspective to identify, scale, select, prioritize and resolve FCs and EEs to set the organization on an optimal performance track. This methodology combines the insights of the Failure Mode Effects and Criticality Analysis (FMECA) , End Effects (EE) studies , Analytic Hierarchy Process (AHP) , the Pareto-Priority Index approach and other statistical methodologies (Bluvband, 2005; Saaty, 1980).

The remainder of the paper is structured as follows: Section 2 introduces the basic definitions used in the paper and illustrates the relationship between information security and safety, Section 3 recommends the organizational functions that should be involved in failure solving, Section 4 outlines the new methodology, Section 5 presents a short case study using the presented methodology framework, Section 6 presents results of sensitivity analysis, and Section 7 summarizes the conclusions of the research.

2. Definitions

Although all three risk-related factors (reliability, safety and security) may be intuitively understood, a clear, unequivocal definition will help to avoid adopting misleading interpretations.

2.1 Reliability

Reliability is defined as "the duration or probability of failure-free performance under stated conditions" (MIL-Std-109C, 1994). Reliability and availability are closely related, and the term reliability often represents both concepts (Musa, 2004). However, strictly speaking, the term "availability" means percentage of the up time. For the case of irreparable systems, availability equals reliability. Otherwise, availability is greater than reliability for any given period of time, because availability also contains the restore time needed for the system to begin operating after the occurrence and resolution of a failure.

2.2 Safety

There are different definitions of safety. Safety can be defined as "freedom from those conditions that can cause death, injury, occupational illness, or damage to or loss of equipment or property, or damage to the environment" (MIL-Std-882C, 1993). IEEE Std-1228 (1994) defines software safety as "freedom from software hazard," where software hazard is defined as "a software condition that is a prerequisite to an accident," and an accident is defined as "an unplanned event or series of events that results in death, injury, illness, environmental damage, or damage to or loss of equipment or property." In this paper we assume that the term "property" also includes intellectual property.

For our purposes, we adopt the following definition of safety by Donald (2003), "the degree to which accidental harm is prevented, detected, and reacted to." What is important to emphasize is that the damages is unintentional.

2.3 Security

In this paper we narrow the discussion to Information Security (IS), rather than security in a broader sense. Information Security can be defined as "the concepts, techniques, technical measures, and administrative measures used to protect information assets from deliberate or inadvertent unauthorized acquisition, damage, disclosure, manipulation, modification, loss, or use." (McDaniel, 1994). Laprie (1992) defines information security from a preventative point of view, as "dependability with respect to prevention of unauthorized access and/or handling of information."

2.4 Dependability

We add dependability as the fourth dimension of SSR. Historically, dependability evolved from the perspective of reliability and availability (Erland 1998). Laprie (1992) defines dependability as "the trustworthiness of a computing system which allows reliance to be justifiable placed on the service it delivers." MIL-Std-109C (1994) defines dependability as "a measure of the degree to which an item is operable and capable of performing its function at any (random) time during a specified mission profile, given item availability at the start of the mission."

Dependability as a broad term (Avizienis, 2001; Erland, 1998) is defined as a compound term comprising Reliability, Availability, Maintainability, Safety and Security (RAMSS)

2.5 Complementary definitions

The following terms are in use throughout the paper:

Failure Cause (FC) is defined as the circumstances during design, manufacture or use which have lead to a failure (BSI, BS 4778-3.2, 1991).

Failure Mode (FM) is defined as the physical or functional manifestation of a failure (IEEE Std-610.12, 1991).

End Effect (EE) is defined as the consequence(s) of a failure mode for the operation, function, or status of the highest indenture level (MIL-Std- 1629, 1980). For all practical purposes, this definition can be extended to include system/equipment availability.

2.6. The Connection between Safety and Security in Software Engineering

The definitions presented in this section clearly indicate that IS has a broader perspective than safety. A safety problem as an unintentional internal action occurring in the system or software, may be performed by the system's operator, or caused by environmental factors. In contrast, a security failure is the result of a deliberate intentional act. This distinction is, however, of no consequence to the Chief Information Security Officer (CISO), who views every event as suspect, and strives to resolve problems independent of the their cause (The CISO is the only person in the IT departments who gets paid to be paranoid). Therefore, from the perspective of the CISO, any fault is defined as an IS problem unless proven otherwise.

Let us examine the following example. Without loss of generality we assume a computerized financial organization with an internal network connected to the Internet, using the IP protocol for communications. Let us further assume that one computer resides within the internal network starts to initiate a session with an organizational

forbidden address, such as a hacker web site. From the perspective of the CISO, this is a malicious, highly dangerous incident that requires an immediate response. Generally, the CISO responds according to the following steps: shut down all outbound communications, check firewall parameters, investigate the suspicious address, find and check the contents of the involved suspicious computer, and re-establish outbound communications after resolution. The reactive actions of the CISO comply with a binary decision matrix aimed at generation of an immediate response to events. However, the corrective actions may discover that it was just a safety problem caused by an internal non-deliberate software failure, such as a bug in the program running on the computer that caused a buffer overflow or a memory leakage, setting a random IP address.

3. Who deals with failures?

Generally, failure treatment in organizations is in the hands of numerous organizational functions (OF) dealing with Operation and RAMSS (Dependability). In many organizations the following OF's teams are required to analyze and manage system failures ((see figure 2).):

- Quality (Q&R) - responsible for Reliability (R) of equipment, software and service
- Logistics, Support and Operations (LSO) – responsible for responding to failures supporting Operation as well as Maintainability (M) and Availability (A) of organization's infrastructure, equipment and software
- Safety(S)
- Security (S)

Although different OFs are involved at different stages of failure management, all functions jointly perform the final two steps of failure management: corrective actions, and defining the lessons to be learned from the event. Sometimes, additional OFs, may participate in these final stages. The training department, for example, may be in charge of incorporating the lessons to be learned into new organizational procedures.

In the next two sub-sections, we will look into the typical OFs that participate in handling hardware and software failures.

3.1 Reacting and Analyzing hardware failures

To properly deal with a hardware event, a partnership among all responsible OFs is required.

For the purpose of reaction and analysis, the unique perspective of each organizational function is essential.

.

Reaching the needed Dependability – the optimal level of RAMSS – organization has to devise methods to minimize failure rates, increase employees' ability to correct errors and repair failures, reduce the time needed to correct a failure and minimize downtime. An example of a methodology to enhance Dependability is a highly coherent, easy to understand hardware operation and troubleshooting procedure manual. Clarity of explanations and instructions in the manual reduces the time required to fix a fault, and increases the ability to resolve failures.

The function of **LSO** is to correct or replace the failed hardware part and incorporate the event into future LSO actions relating to test and support equipment, support users, programmers, etc

The role of the **Safety** function is to determine the potential effect of the failure on people and hardware. The proactive aspect of **Safety** is to identify processes and methodologies in advance in order to avoid such failure events. This is done by incorporating information about the failure - probabilities of damage and losses, compliance with standards, failure rates, etc - into organization's failure reporting database.

The **Security** function is comprised of two parts: physical security and IS. The physical security function addresses possible physical break-ins, the responsible parties and the possible resulting damage. Information Security (IS) is dealt by the CISO.

The **Quality** (actually, **Q&R**) function is the center of event analysis. The Quality function investigates the root cause of the failure, and incorporates the lessons of the analysis by adding controls to the system to enhance future fault analysis. Generally, the Quality function is interested in investigating and prevention of the potential and observed errors, faults and failures.

3.2 Analyzing software failures

Software failure analysis is similar to hardware failure analysis with the exception of the following:

The LSO function (specifically, software programmers) is responsible for correcting the program, creating a new program, or bypassing the failure to restore proper operation.

The Safety function investigates the extent of any financial damage or danger to human life as a result of a software failure.

IS checks for possible surveillance of intrusions into organization's computers.

Software failure Quality activities are much the same as in the case of hardware failure.

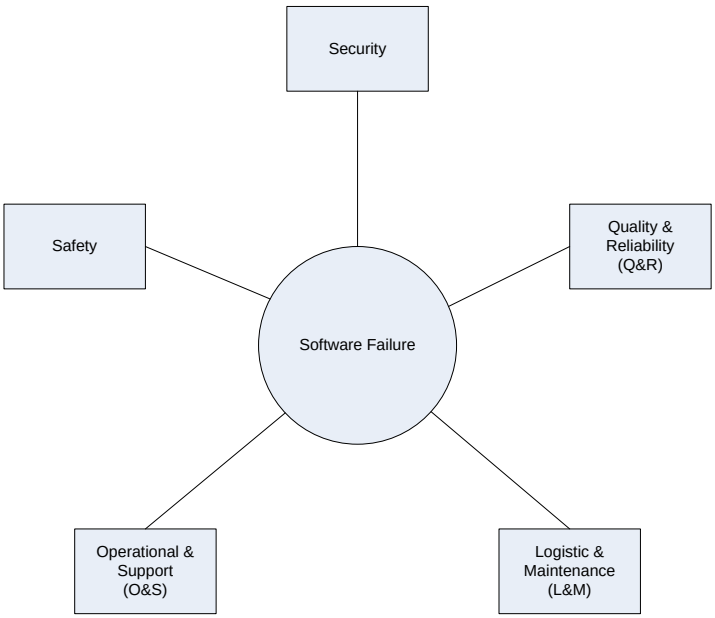

Figure 2: *Organizational Functions participating in failure management, handling and analysis***.**

4. Prioritization of Failure Management Based on End Effects

4.1 Outline of the New Methodology

In this section we propose a new organizational methodology for prioritizing failures based on End Effects. The methodology proposed herein is based on the AHP algorithm for multivariable decisions presented by Saaty (Saaty, 1980). In the methodology, we assume that collaboration among all OFs is necessary in analyzing and prioritizing End Effects (EE), Failure Mode (FM), Failure Cause (FC) in order to achieve harmonized solutions. The methodology differs from almost all currently proposed methodologies which ignore participants of the prioritization process (see figure 3).

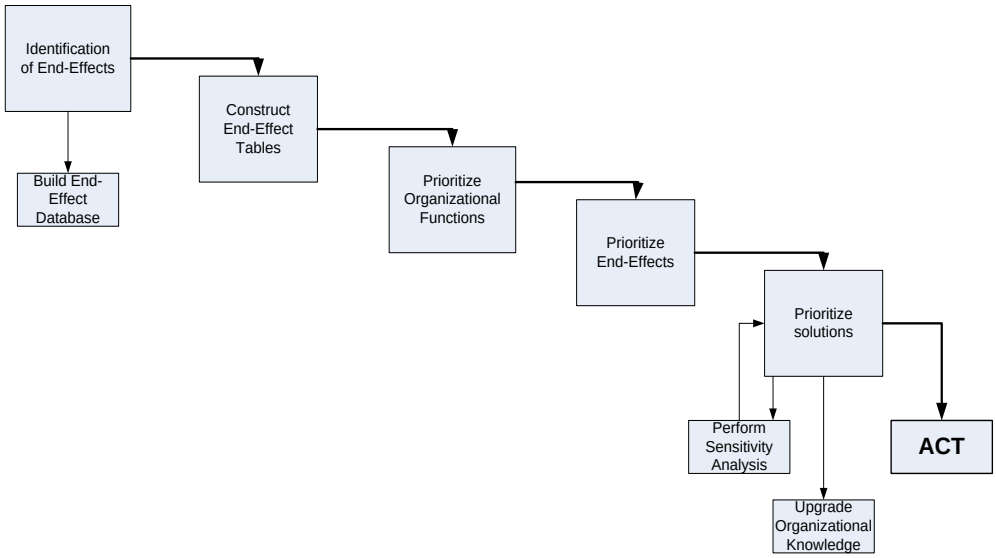

Figure 3: *Methodology steps.*

The methodology is comprised of the following six steps:

Step 1: Identification of all End Effects (EE)

Identifying a complete set of EEs is based on the following procedure:

- Data Collection - collection of all historical data about failures and failure causes; including lessons learned based on similar incidents in other organizations.

- Brainstorming - conduct a brainstorming session to ensure the set of EEs to be as exhaustive as possible.
- Conclusion - finalization of the EE list when all known EE are added. Making sure that when any new EE is identified in the future , it would be added to the list according the above procedure, generating a revised EE list.

Step 2: Construction of OF tables

Draft a two-dimensional table containing all EEs defined in the previous step for each OF. Using the AHP Algorithm, each cell represents a pair-wise comparison value from the perspective of each OF regarding each EE. Based on its unique responsibilities, views and expertise, each OF attributes a different weight to each EE.

Step 3: Prioritize Organizational Functions.

Every organization has a different view of a relative importance of its OFs. For example, a financial organization may prefer IS over Quality, an airline gives a major significance to Safety issues at the expense of IS. In this step, the AHP algorithm is used once again to prioritize OFs.

Some organizations may decide not to give preference to any OF. In this case, equal weight is given to all OFs.

Step 4: Building an EE Prioritization List

The EE prioritization list results from summing the product of EE weight (Step 2) and OF weight (step 3), for each EE. The prioritization list reflects a harmonized decision making tool, that reflects the relative importance of each EE to the organization, and the order in which the EEs should be managed and resolved.

Step 5: FC Solution Prioritization List

Failure Causes (FCs) are derived from Failure Modes (FMs), which are in turn, derived from EEs. FCs and FMs, and between FMs and EEs can relate to ach other as one-to-many, many-to-one and or many-to-many. Using separately such tools as Fault Tree Analysis (FTA) and Failure Mode Effect Analysis (FMEA) may bring results, but both methodologies would exhaust organization's resources.

Bluvband (2005), has proposed a new root failure analysis method named Bouncing Failure Analysis (BFA), BFA combines FTA and FMEA into one coherent, convenient, methodology. BFA methodology is comprised of the following four steps:

a. Define all possible EEs (See above step 1.)

b. Assign appropriate severity to each EE (See above steps 2-4.)

c. Define all possible FMs

   Identify all possible FMs linked to and deriving from each EE.

d. Analyze FMs by shifting between the FMEA method, for FM and EE analysis, to the FTA method for calculations and sensitivity analysis, and bouncing back to the FMEA method.

For each FC identified by the BFA, a remedy and an action are defined with the aim of preventing the failure, detecting its occurrence, recovering from the failure, or deciding on a partial or no implementation of a remedy, based on the management's acceptance of a specific risks.

Sometimes, in order to achieve an optimal organizational solution , additional OFs should be involved in defining solutions for failures. For example, the legal department is required to check legal options for changing a maintenance contract.

Each OF has its own set of solutions for any specific failure. Such a set of solutions may not be suitable from the perspective of other OFs, and may even be sub-optimal for the organization. To overcome these differences in perspective, a harmonization approach to failure resolution should be adopted.

Harmonization is achieved by collecting all the solutions proposed by the OFs and produced by application of AHP algorithm. The entire process is very similar to steps 2, 3, 4 above, this time using proposed solutions instead of EEs. The result of this final stage is an optimal organizational solution to the EE.

Step 6: Sensitivity/ validation analysis

Looking closer at Table 2 we find a small and non-significant weight difference between the pair "Firewall slowdown" vs "Virus in network ", and between "Firewall miss-configured" vs "Network flood." Obviously there is a need for a method that generates more significant results.. The first option is to run another AHP, only for these hard to distinguish pairs, asking a different group of OF's employees to give their comparison pairs. The second option is to use different scales for calculating weights, for example, a scale of [1, 5, 10] or a scale of [1,3, 9], followed by the use of scales such as [1,10,100] and [1,2,4] for each pair-wise comparison.

When different scales are applied, there are two possible outcomes. Either the same preference is validated in all cases (using different scales) with much more significant results, or the priorities are reversed. In the first case the result is final, but in the latter case, a statistical decision method should be applied to the same pair-wise comparisons, i.e. getting results from an entire team of experts (not just from one or two), so we'll have **n** comparative values in each cell. Thus**,** we'll perform the statistical analysis at each cell calculating the average and standard deviation for the results in each comparative cell and finally for the summarized results in the AHP table. In this case the importance and the

priority of a EEs will be characterized by the Priority Index (PI) calculated as a lower limit of the average PI= $\overline{X}$ -3S/$\sqrt{n}$, where $\overline{X}$ is the mean score value and S- calculated standard deviation of the resulting score values.

5. *Applying the Methodology to Prioritization of Software Failures: a Case study*

The goal of the prioritization process is to construct a harmonized prioritized list of EEs and solutions, clarify the preferred prioritization among different organizational functions responsible for resolving the issues. The following example illustrates the prioritization process described in section 5 for an information security software failure in an organization.

Step1: Let us assume that the following nine EEs have been identified in the organization (Table 1).

Table 1: End Effects.

End Effect (EE)	**End Effect description**
Communication Slowdown	Users sense reduced outer network response time.
Virus in network	A virus is detected on at least one PC or server.
Firewall slowdown	Sharp degradation in the firewall throughput.
Network flood	Network is flooded with messages of any type. For example, regular load of the internal network is 10% of the network bandwidth and over 20% in the case of a flooded network.
Stacked service	A service is stacked in the server.
Firewall miss-configured	Rules are not configured properly.
Misrouted messages	Messages are misrouted due to a bug in the messaging program or modification of routing tables by a hacker infiltrating the system.
Firewall leakage	In and Out messages penetrate the firewall.
Workstation hangs	Workstation hangs without any reasonable explanation.

Step 2: Each OF applied a Saaty pair-wise prioritization between each of the above EEs, using a scale of 1 to 9 ([1, 9]). Each OF defines priorities among the EEs based on its organizational role and duties (see appendix A, Table A.1, A.2, A.3, A.4 and A.5). Each cell C(i,j) represents the results of a pair-wise comparison performed by the decision maker on the relative importance of EEi (row) over EEj (column). C(i,j) = 1 means that the decision maker attributes equal importance to EEi and EEj. Each C(j,i) cell holds 1/C(i,j). Calculating the weight of each EE is performed in three phases: Phase 1: Calculating the sum of the jth column. Phase 2: Dividing each cell C(i,j) by the sum of the jth column (calculated at phase 1). Phase 3: Calculating the weight *wj* by summing up the calculated values in C(i,j) for each row.

Step 3: generates the organizational preference among the OFs. Using the AHP methodology, a list of the relative weights (reflecting the relative importance) of each OF is generated (see appendix A, Table A.6. The AHP calculation is explained in step 2.).

Step 4: prioritizes EEs, taking into consideration EE weight and relative OF weight. This step generates a value for each EE, reflecting its relative organizational importance, by summing the products of each EE weight and the weight attributed to each OF involved.

$$Wj = \sum_{i} wji \bullet Vi$$

Were

Wj is the relative weight of EE j among all End Effects.

wji is the weight of EE j calculated by OF i (see step2 above).

Vi (*i*=1,2,3,…,k) is the relative weight of the OF i, defined by the organization management.

By ordering EEs according there relative weights, *Wj*, the results in an ordered list of EEs, from the most important to the least important (see Table 2).

Table 2: *Ordered list of EEs by Wj.*

Sequence	End Effect	Weights
1	Communication Slowdown	0.208
2	Misrouted messages	0.151
3	Firewall slowdown	0.116
4	Virus in network	0.114
5	Stacked service	0.105
6	Firewall miss-configured	0.089
7	Network flood	0.085
8	Firewall leakage	0.075
9	Workstation hangs	0.058

The list in Table 2 expresses the organizational priority of EEs, that is, the sequence in which EEs should be managed and resolved, in the event of multiple EE occurrences.

Step 5: Prioritization of solutions is the decision making step (See Appendix C). This step is a two-phase one. First, AHP is applied to the solutions proposed by each OF for each FM or FC, and in the second sub-phase, a harmonized (optimal from the business perspective) solution is constructed or a solution that encompasses the preferences of all OF regarding the solution. (Remark: Without loss of generality, we assume that the same number of OFs, as mention above, are at this phase, although, in practice, the number of OF may vary.)

For example, a decision to use a new FireWall software product that answers some IS needs for securing the organization Internet site. The Firewall was configured to close inbound traffic used to uploading web seminars. From the IS aspect closing such inbound data transfer is an excellent decision as it eliminates viruses from entering into the servers. From the quality and reliability aspects it is a wrong decision as it causes users to

leave there sessions in the middle of a transaction execution, generates cpu-time overhead (to overcome a transaction failure), generates memory saturation, and a generates a total slowdown in the site response time. From the operational perspective such FireWall configuration is a wrong decision, as the users aren't using the site. From the marketing view there is a potential loosing of market share to competitors, decreasing revenue, loosing of customers etc.

The AHP calculation is the same as in previous steps. For each FM_k (or FC_k) there is a set S_k of solutions, $S_k=(s_1,s_2,s_3,\ldots,s_n)$. Let's assume that r OFs participate in the solution step. V is a vector containing the relative weight attributed to each OF by the organization,

$V= (v_1,v_2,v_3,\ldots,v_r)$. With r OFs, we execute the AHP process r times for each FM_k (or FC_k). For each set of solutions S_k , we define a two dimensional table, in which each cell C(i,j) represents a pair-wise preference of solution i over solution j by OF_t, (t=1,2,3,…,r). Using the AHP algorithm we calculate a vector of solution's weights generated by one specific OF (Same as described in previous section.). Then, the weight of each solution, s_i is calculated by the following:

$$Ws_i = \sum_{j=1}^{r} w_{si(forOFj)} * Vj$$

where Ws_i denotes the weight of solution s_i.

Step 6: Sensitivity analysis.

Getting results from an entire team of n experts, i.e. n comparative values in each cell, we'll calculate the priority indices PI (see Table 3):

Table 3: *Sensitivity analysis table for EEs*

Sequence	End Effect	Weights and Priority Indices (after the Sensitivity Analysis)	Weights (before the Sensitivity Analysis)
1	Communication Slowdown	$\bar{X}$ = .34; S/$\sqrt{n}$ = .023; **PI=.27**	0.208
2	Misrouted messages	$\bar{X}$ = .20; S/$\sqrt{n}$ =.024; **PI=.13**	0.151
3	Firewall slowdown	$\bar{X}$ = .13; S/ $\sqrt{n}$ =.012; **PI=.094**	0.116
4	Virus in network	$\bar{X}$ = .11; S/$\sqrt{n}$ =.011**; PI=.077**	0.114
5	Stacked service	$\bar{X}$ = .12; S/$\sqrt{n}$ =.016**; PI=.072**	0.105
6	Firewall miss-configured	$\bar{X}$ = .10; S/$\sqrt{n}$ =.021; **PI=.037**	0.089
7	Network flood	$\bar{X}$ = .08; S/$\sqrt{n}$ =.022; **PI=.014**	0.085
8	Firewall leakage	$\bar{X}$ = .062; S/$\sqrt{n}$ =.017; **PI=.011**	0.075
9	Workstation hangs	$\bar{X}$ = .046; S/$\sqrt{n}$ =.015; **PI=.001**	0.058

From the sensitivity analysis Table 3 above, one can see that in our case the same preferences are validated for all EEs.

6. Conclusions and Future Work

In this paper we presented an innovative methodology that can be applied to prioritize IS events sequence handling and remedies for those events, subject to a clear definition of OF importance at a specific organization. Implementing AHP as a unified approach, moves the organization towards an optimal solution that encompass organizational thinking about the importance of an IS event and remedies to that event.

The concept of using EE and prioritizing events, prioritizing and harmonizing solutions to potential risk factors, brings up and emphasizes the customers perspective instead of technical perspective. This perspective is in accordance with modern marketing that puts the customers in the center of customers-organization relationship, which is a

business-oriented approach. Using prioritization and harmonization throughout the EE view, sets a more coherent resource allocation methodology that still is in accordance with the business goals of the organization.

Decisions can be seen as reactive (tactical) or proactive (strategically). Both types / levels of decision must be aligned and take into consideration all known aspects of the IS event, OFs, consequences and possible remedies based on the prioritized process. It is obvious that for different organizations, the same IS event metrics will have different prioritization and different solution sequences. These prioritization metrics can be viewed as a knowledge base about the relative importance attributed by management to each IS event, and to the proactive, reactive and corrective actions, or remedies, that should be taken. From this point of view, it is obvious that the metrics are dynamic and may change in response to changes in organizational goals, market conditions, regulations or other factors, and should be rechecked and reevaluated periodically.
IS reliability, safety and dependability should not be treated as separate domains but should rather be addressed from a unified organizational perspective, using one of the methodologies discussed above to harmonize between different OF priorities.

There are some issues that require additional research attention. Issues like: Using and adopting the bootstrap statistical approach in case of decision conflicts or sensitivity analysis. In addition, the decision on the optimal organization solution should be broaden with the concepts of Pareto Priority Indices (PPI) comparison between suggested solutions, taking in account (Bluvband, 1999) Cost of solution COS, Time to problem solving TTPS, Probability of success POS, etc.

Appendixes

Appendix A. Prioritization process with unequal weights at O.F

Table A.1. ***Pair-wise comparison for the nine events from the programmer point of view.***

	comm. slow	virus in net	Fw slow	flood. net	Stacked service	FW miss-config	Miss routed massage	FW leakage	WS hangs	Weights*
comm. slow	1	10	4	4	10	10	10	10	5	0.371
virus in net	1/10	1	1/4	1/7	1	1	1	1	1/3	0.031
Fw slow	1/4	4	1	5	5	5	6	5	5	0.203
flood. net	1/4	7	1/5	1	8	8	8	8	4	0.192
Stacked service	1/10	1	1/5	1/8	1	1	1	1	1/4	0.029
FW miss-config	1/10	1	1/5	1/8	1	1	1	1	1/4	0.029
Miss routed massage	1/10	1	1/6	1/8	1	1	1	1	1/4	0.029
FW leakage	1/10	1	1/5	1/8	1	1	1	1	1/4	0.029
WS hangs	1/5	3	1/5	1/4	4	4	4	4	1	0.090

For explanation of each EE see section 6 table 1.

Table A.2. ***Pair-wise comparison for the nine events from the Information Security point of view.***

	comm. slow	virus in net	Fw slow	flood. net	Stacked service	FW miss-config	Miss routed massage	FW leakage	WS hangs	Weights*
comm. slow	1	4	6	2	8	8	7	5	10	0.267
virus in net	1/4	1	1/4	1/4	9	1/7	1/9	9	10	0.094
Fw slow	1/6	4	1	1/6	8	2	5	1/6	9	0.094
flood. net	1/2	4	6	1	10	10	10	10	10	0.264
Stacked service	1/8	1/9	1/8	1/10	1	1/5	1/6	1/10	5	0.021
FW miss-config	1/8	7	1/2	1/10	5	1	1	1	9	0.074
Miss routed	1/7	9	1/5	1/10	6	1	1	1	9	0.083

massage										
FW leakage	1/5	1/9	6	1/10	10	1	1	1	9	0.091
WS hangs	1/10	1/10	1/9	1/10	1/5	1/9	1/9	1/9	1	0.012

Table A.3. ***Pair-wise comparison for the nine events from Safety point of view.***

	comm. slow	virus in net	Fw slow	flood. net	Stacked service	FW miss-config	Miss routed massage	FW leakage	WS hangs	Weights*
comm. slow	1	1/5	1	1	1/10	1/10	1/10	1/10	1	0.031
virus in net	5	1	10	5	4	1/7	1/6	1/7	1	0.131
Fw slow	1	1/10	1	2	1/6	1/5	1/5	1/5	1/5	0.028
flood. net	1	1/5	1/2	1	1/10	1/5	1/5	1/5	1/5	0.024
Stacked service	10	1/4	6	10	1	1	1	1	1	0.156
FW miss-config	10	7	5	5	1	1	1	1	1	0.171
Miss routed massage	10	6	5	5	1	1	1	1	1	0.167
FW leakage	10	7	5	5	1	1	1	1	1	0.171
WS hangs	1	1	5	5	1	1	1	1	1	0.122

Table A.4. ***Pair-wise comparison for the nine events from Reliability point of view.***

	comm. slow	virus in net	Fw slow	flood. net	Stacked service	FW miss-config	Miss routed massage	FW leakage	WS hangs	Weights*
comm. slow	1	10	1	5	1	10	10	10	5	0.351
virus in net	1/10	1	1/10	1/5	1/5	1/5	1/5	1/5	1	0.028
Fw slow	1	10	1	1	1/5	1/6	1/6	1/5	1/5	0.069
flood. net	1/5	5	1	1	1/10	1/5	1/5	1/5	1/5	0.032
Stacked service	1	5	5	10	1	1	1	1	1	0.138
FW miss-	1/10	5	6	5	1	1	1	1	1	0.100

config										
Miss routed massage	1/10	5	6	5	1	1	1	1	1	0.100
FW leakage	1/10	5	5	5	1	1	1	1	1	0.096
WS hangs	1/5	1	5	5	1	1	1	1	1	0.090

Table A.5. ***Pair-wise comparison for the nine events from Quality point of view.***

	comm. slow	virus in net	Fw slow	flood. net	Stacked service	FW miss-config	Miss routed massage	FW leakage	WS hangs	Weights*
comm. slow	1	10	1	2	1/7	1/10	1/8	1	1	0.058
virus in net	1/10	1	1/10	1/5	1/5	1/5	1/5	1/5	1	0.026
Fw slow	1	10	1	1	1/5	1/6	1/6	1	1/5	0.055
flood. net	1/2	5	1	1	1/10	0.2	0.2	1	1/5	0.041
Stacked service	7	5	5	10	1	1	1	6	6	0.232
FW miss-config	10	5	6	5	1	1	1	7	8	0.242
Miss routed massage	8	5	6	5	1	1	1	5	7	0.222
FW leakage	1	5	1	1	1/6	1/7	1/5	1	1/5	0.043

Table A.6. ***Pair-wise comparison of the participants***

	Programmer	Security	Safety	Reliability	Quality	Weights*
Programmer	1	1/10	1/5	1/5	1/5	0.315
Security	10	1	5	5	5	0.496
Safety	5	1/5	1	5	5	0.242
Reliability	5	1/5	1/5	1	5	0.147
Quality	5	1/5	1/5	1/5	1	0.084

Appendix B. The case of equal OF weights

Table B.1. ***Pair-wise comparison of the participants with equal organizational weight***

	Programmer	Security	Safety	Reliability	Quality	Weights*
Programmer	1	1	1	1	1	0.315
Security	1	1	1	1	1	0.496
Safety	1	1	1	1	1	0.242
Reliability	1	1	1	1	1	0.147
Quality	1	1	1	1	1	0.084

Table B.2. ***Ordered list of EE by Wj for equal weight of O.F.***

Sequence	Failure	Weights
1	Communication Slowdown	0.216
2	Firewall miss-configured	0.123
3	Misrouted messages	0.120
4	Stacked service	0.115
5	Network flood	0.110
6	Firewall slowdown	0.090
7	Firewall leakage	0.087
8	Workstation hangs	0.080
9	Virus in network	0.061

Appendix C. Building EE, FM, FC and Suggested Solutions analysis

Table C.1 is an example of breaking down the EE "Virus in network", to the FM and FC.

Table C.2 represents some of the FC and the suggested recommended action solving that EE, suggested in table C.1.

Table C.3 sums an AHP process performed to the proposed solutions, and represents only proactive solutions. Only proactive solutions create a change in the system.

Table C.4 is the prioritized sequence of solutions to be implementation.

Table C.1: ***The breakdown of EE to FM and FC.***

EE	**FM**	**FC**
Virus in network	Virus in Work-Station (WS)	An infected media was inserted into the WS.
		The AntiVirus (AV) program doesn't check automatically inserted media on that drive.
		False alarm. A bit-string is wrongly interpreted as a virus.
	Virus in a server	An infected email received at the server.

Table C.2: ***FC and suggested recommended action solving an FC.***

FC	**Solutions**	
	Reactive	**Proactive**
An infected media was inserted into the WS.	Disconnect immediately the WS from net. Clean WS from virus.	Set an IS policy restricting insertion of unchecked media. Define a specific stand alone WS for virus checking and cleaning
The AntiVirus program doesn't check automatically inserted media into that drive.	Disconnect immediately the WS from net. Clean WS from virus.	Build new default parameters for the AntiVirus (AV).
False alarm. A bit-	Ignore. Call manufacturer for	Download manufacturer updates

string is wrongly interpreted as a virus.	getting a software update.	on a daily basis.
An infected email received at the server.	Disconnect immediately the server from net. Clean server from virus Check AV parameters at the mail server.	Contact the ISP and ask for AV activation at the ISP premises.

Table C.3: ***Summing up of an AHP process to various suggested solutions***.

#	**Solution**	**Solution Weights**
1	Set an IS policy restricting insertion of unchecked media	0.287
2	Define a specific stand-alone WS for virus checking and cleaning	0.177
3	Build new default parameters for the AntiVirus (AV).	0.133
4	Download manufacturer updates on a daily basis	0.194
5	Contact the ISP and ask for AV activation at the ISP premises.	0.207

Table C.4: ***Prioritized list of solutions*.**

#	**Solution**	**Solution Weights**
1	Set an IS policy restricting insertion of unchecked media	0.287
2	Contact the ISP and ask for AV activation at the ISP premises.	0.207
3	Download manufacturer updates on a daily basis	0.194
4	Define a specific stand alone WS for virus checking and cleaning	0.177
5	Build new default parameters for the AntiVirus (AV)	0.133

Bibliography

Alpcan (2003) Alpcan, T., and Basar, T., "A Game Theoretic Approach to Decision and Analysis in Network Intrusion Detection", *Proceedings of the 42nd IEEE conference on Decision and Control*, Maui, Hawaii USA, December 2003,

Austin (1993) Austin, L.M. and Grandforoush, P., "management science for Decision Making", *West Publishing Company*, USA, 1993

Avizienis **(**2001) A. Avizienis, J., C. Laprie and B. Randell, "Fundamental Concepts of Dependability", Research report NO. 1145, LASS-CNRS, April 2001.

Brown (1999) Brown, C., V., Ross, J., W., "The IT Organization of the 21th Century: Moving to a Process-Based Organization", *Center for Information Systems Research, Sloan Working-Paper No. 4078*, Massachusetts Institute of Technology, 1999

BSI, BS 4778-3.2 (1991) British Standards Institution, BS 4778, Part 3, Availability, Reliability and Maintanability terms; Section 3.2: 'Glossary of international terms', 1991 (equal to IEC 50-191,1990).

Cavusoglu (2004) Cavusoglu H., Birendra, M. and Srinivasan R., "A Model for Evaluating IT Security Investments", *CACM*, July 2004, Vol. 47, No. 7, pp. 87-92

Conrad (2005) Conrad, J. R., "Analyzing the Risk of Information Security Investment with Monte-Carlo Simulation", *Fourth Workshop on the Economics of the Information Security*, Harvard University, UK, June 2005

Donald (2003) Donald G. Firesmith, "Common Concepts Underlying Safety, Security, and Survivability Engineering", *technical Notes CMU/SEI-*

2003-TN-033, 2003, http://www.sei.cmu.edu/publications/pubweb.html.

Doolin (2003) Doolin, B., McQueen, B., and Watton, M., " Towards a Framework of Internet Strategies for Established Retailers", *7th Pacific Asia Conference in Information Systems*, July 2003, Adelaide, South Australia, 2003

Edgington (2004) Edgington , T., Choi, B., Henson, K., Raghu, T.S., and Vinze, A., "Adopting Ontology to Facilitate Knowledge Sharing", *CACM*, Vol. 47, No. 11, November 2004, pp. 85-90

Erland (1998) Erland, J.,"An Integrated Framework for Security and Dependability", *Proceedings of the 1998 workshop on new security paradigms*, January 1998

Finne (1998) Finne, T., "The Three Categories of Decision-Making and Information Security", *Computer & Security*, Vol. 17, 1998, pp 397-405

(Grady 1996) Grady, R., B., "Software Failure Analysis for High-Return Process Improvement Decision.", *Hewlett-Packard Journal*, Vol.47, No. 4, August 1996

IEEE Std-1228 (1994) Institute of Electrical and Electronics Engineers, IEEE-Std-1228, "IEEE Standard for software safety plans", *IEEE* Copyright, 1994. All rights reserved.

IEEE Std-610.12 (1991) Institute of Electrical and Electronics Engineers, IEEE-Std-610.12, "Glossary of software engineering terminology", *IEEE* Copyright, 1991. All rights reserved.

Kume (1985) Nakajo, T., and H.Kume, "The Principles of Fool proofing and there Application in Manufacturing", *Reports of Statistical Application Research* 32, No.2

McDaniel (1994), Mcdaniel, G.,ed.,(1994), IBM Dictionary of computing. New

York, NY: McGraw-Hill, Inc., 1994.

MIL-Std-109C (1994) USA Department of Defense, MIL-Std-109C, "Quality assurance terms and definitions", 1994

MIL-Std-882C (1993) USA Department of Defense, MIL-Std-882C, "System safety Program requirements", 1993

MIL-Std-1629A (1980) USA Department of Defense, MIL-Std-1629A, "Procedures for Performing A Failure Mode, Effects and Criticality Analysis", 1980

Musa (2004) Musa, j. d., "Software Reliability Engineering More Reliable Software Faster and Cheaper", *Author Hose*, 2'nd Ed 2004, ISBN 1-4184-9387-2 (*sc*)

Olovsson (1992), Olovsson, T., " A Structured Approach to Computer Security", Technical Report No 122, Department of Computer Engineering, Chalmers University of Technology Goteborg.

Ryan (2005) Ryan, J. J. H., Ryan D. J., "Proportional Hazards in Information Security", *Risk Analysis*, Vol. 24, No. 1, 2005

Saaty (1977) Saaty, T.,"A Scaling method for Priorities in Hierarchical Structures", *Journal of mathematical Psychology*, 15, pp. 234-281.

Saaty (1980) Saaty, T.,"The Analytic Hierarchy Process", McGraw Hill Company, New York

Sheldon (2005) Sheldon, F., T., Batsell, S., G., Prowell, S., J., Langston, M., A., "Position Statement: Methodology to Support Dependable Survivable Cyber-Secure Infrastructures", *Proceedings of the 38th Hawaii International Conference on System Sciences*, 2005

Taha (1987) Taha, H. A., "Operation Research, an Introduction", *Macmillan Publishing Company*, New York, 1987

Theodosios (2005) Theodosios, Tsiakis, and George Stephanides. "The economic

approach of information security, *Computers & security*, 24, 2005, pp. 105-108

UK DTI, ITSEC (1991) UK Department of Trade and Industry, ITSEC, Information Technology Security Evaluation Criteria, June 1991

Winterfeldt (1986) Von Winterfeldt, Detlof & Ward E., "Decision Analysis and Behavioral Research", *Cambridge University Press*, USA, 1986

Bluvband (1999) Bluvband, Z., Zilberberg, E., "Knowledge-based approach to integrated FMEA," *Proceedings of ASQ's 53rd Annual Quality Congress*, Anaheim, CA: ASQ, 1999.

Bluvband (2005) Bluvband, Z., Polak, R., Grabov, P., "Bouncing Failure Analysis (BFA): The Unified FTA-FMEA methodology", *RAMS 2005*, 2005