Security Risk Scoring Incorporating Computers' Environment

Eli Weintraub
Department of Industrial Engineering and Management
Afeka Tel Aviv Academic College of Engineering
Tel Aviv, Israel

***Abstract*—A framework of a Continuous Monitoring System (CMS) is presented, having new improved capabilities. The system uses the actual real-time configuration of the system and environment characterized by a Configuration Management Data Base (CMDB) which includes detailed information of organizational database contents, security and privacy specifications. The Common Vulnerability Scoring Systems' (CVSS) algorithm produces risk scores incorporating information from the CMDB. By using the real updated environmental characteristics the system enables achieving accurate scores compared to existing practices. Framework presentation includes systems' design and an illustration of scoring computations.**

***Keywords*—CVSS; Security; Risk Management; Configuration Management; CMDB; Continuous Monitoring System; Vulnerability**

I. Introduction

Computing systems are subject to cyber-attacks which may cause damage to organizational and personal data, software and hardware [1]. Vulnerabilities are weaknesses or exposures stemming from bugs that are potential causes of security failures: loss of confidentiality, integrity or availability. Attackers are exploiting target systems making use of software vulnerabilities existing in systems' components. Attacks on users' computers cause them damages of many kinds such as stealing organizations' information or changing customers' data. The quality of knowledge an organization has of systems' weaknesses influences heavily on the success of organizations' defense activities. This work focuses on gaining accurate knowledge of computers' vulnerabilities, thus enabling improved organizational risk measures, which enable the development of efficient mitigation activities to defend computers from hostile attackers. Reference [2] states that Stuxnet worm included a process of checking hardware models and configuration details, and also downloads code from the controller to check if it was the “right” program before launching an attack. Both, attackers and security managers are interested in gaining accurate and detailed information of the target system, but for the opposite reasons. Organizations make decisions on actions they have to take, to limit their risks according to the amount of potential damage and vulnerability characteristics [3].

Risk has many definitions in research publications. We use the definition of [4]: "An event where the outcome in uncertain". According to this definition, this work is aimed at lessening risk uncertainty. The proposed model focuses on gaining accurate real-time information on systems' configuration, components and the environment which the system interfaces.

Several software products are aimed at defending computers from cyber attackers. Antivirus software, antispyware and firewalls are examples to some of these tools which usually perform periodic assessment of the target computer by comparing computers' software to the known published vulnerabilities. Antivirus software and firewalls use hash signatures to identify attacks on assets. In cases the defense software recognizes a hash signature in computers' software it reports to the computers' owner the existence of the attacking software. Those tools are aimed at identifying known threats but not new unpublished threats. Continuous Monitoring Systems (CMS) monitor computer systems in a near real time process aimed at detecting vulnerabilities and notifying organizations' security managers. Contemporary systems use vulnerabilities databases which are continually updated as new vulnerabilities are identified and a scoring algorithm which predicts potential business losses. CMS's are essential tools for limiting the time-frames organizations are exposed to risks, thus enabling organizations taking measures for risk mitigation.

Computers are defenseless to known threats as long as no patch exits to protect against the vulnerability. Preparing such a patch needs efforts of design, programming and testing activities that may last weeks or months. Only after the software vendor prepares a working patch, computers' owner has to load it to the operational system, which is the moment the computer ceases to be vulnerable. Loading patches to computer systems are usually performed as a periodical process, not continuously. The reason for this is avoiding too many interrupts required for uploading and activating the patch on the organizational computers. Other software tools usually use heuristic algorithms which are programmed to detect irregular suspicious activities of the software running on the computers. Those tools are programmed to detect deviations from the "normal" profile of computers' activities. In today's environment of zero-day exploits, conventional systems updating for security vulnerabilities has become a cumbersome process. There is an urgent need for a solution that can rapidly evaluate system vulnerabilities' potential damages and immediately fix them. [5].

Security Continuous Monitoring (SCM) tools are operating techniques for monitoring, detecting and alerting of security threats on a regular basis. After identifying these risks, tools evaluate the potential impacts on the organization, sometimes suggesting risk mitigation activities to the organization to support organizational risk management decisions. Reference [6] states that SCM systems which are running on computers, continuously try to detect systems' vulnerabilities aim to close the gap between the zero-day of identifying the vulnerability, until the moment computers' owner loads the corresponding patch fixing the vulnerability. The time frame may be considerably long.

In this paper, we describe the mechanisms of a new SCM framework of a system that will produce better detection and prevention than current known SCM systems. Frameworks' capabilities makes use of two main resources: knowledge concerning the specific computers' organizational environment of the target system, and a prediction algorithm which runs continuously evaluating risk scores.

The rest of the paper includes the following sections: In section 2 a description of current known existing solutions. In section 3 a presentation of the proposed framework including systems' architecture. In section 4 a description of the risk scoring algorithm which computes risk scores. In section 5 results. In section 6 conclusions and future research directions.

II. EXISTING SOLUTIONS

SCM systems are using external vulnerabilities databases for evaluation of the target computers' risk. There are several owners of vulnerability databases [5]: The Sans Internet Storm Center services and The National Vulnerability Database (NVD). Vulnerability Identification Systems (VIS) aimed to identify vulnerabilities according to three categories: code, design, and architecture. Examples for VIS are the Common Vulnerabilities and Exposures (CVE), and The Common Weakness Enumeration (CWE).

In this work, we shall use NVD vulnerabilities database as an illustration of the proposed model.

Risk evaluation uses scoring systems for assessing the impacts of vulnerabilities on the organization. The Common Vulnerability Scoring System (CVSS) is a framework that enables user organizations to receive IT vulnerabilities characteristics [1].

CVSS uses three groups of parameters to score potential risks: basic parameters, temporal parameters, and environmental parameters. Each group is represented by several score compound parameters ordered as a vector, used to compute the score. Basic parameters represent the intrinsic specifications of the vulnerability. Temporal parameters represent the specifications of vulnerabilities that might change over time due to technical changes. Environmental parameters represent the specifications of vulnerabilities derived from the local IT environment used by the organization. CVSS enables omitting the environmental metrics from score calculations in cases they have no effect on the score and in cases the users' does not specify the detailed description of environments' structure, and it's components.

CVSS is a common framework for characterizing vulnerabilities and predicting risks, used by IT managers, risk managers, researchers and IT vendors, for several aspects of risk management.

CVSS is an open framework which enables managers to deal with organizations' risks and make decisions based on facts rather than evaluations. Organizations adopting CVSS framework may gain the following benefits:

- A standard scale for characterizing vulnerabilities and risks scoring.
- Normalizing vulnerabilities according to specific IT platforms. The computed scores enable users getting rational decisions in correlation to vulnerability risks.
- CVSS uses an open framework. Organizations can see the characteristics of vulnerabilities and the logical process of the scoring evaluation.
- Environmental scores. Organizations using the environmental parameters may benefit by considering changes in its IT configuration according to predicted risk scores. The specification of systems' configuration is defined using only high-level parameters such as system.

There are few other vulnerability scoring systems besides CVSS differing by what they measure. CERT/CC emphases internet infrastructure risks. SANS vulnerability system considers users' IT configuration and uses default parameter definitions. Microsoft’s scoring system emphasizes attack vectors and the impacts of the vulnerability. Using CVSS scoring system, basic and temporal parameters are specified and published by products' vendors who have the best knowledge of their product. Users make estimates of environmental parameters since they have the best knowledge of their environments and vulnerability business impacts.

This paper focuses mainly on environmental metrics.

An exploit of a vulnerable component may cause major or minor damage to a system, depending on the technological and business characteristics of the configuration and of systems' users. CVSS environmental parameters specify the characteristics of vulnerabilities in correlation with systems' components. Environmental parameters are of three groups:

- Collateral Damage Potential (CDP).

A group of parameters which measure the economic potential loss caused by an exploit of a vulnerable component.

- Target Distribution (TD).

Parameters indicating the percentage of vulnerable components in users' environment. A large proportion might have more impacts on organizational potential damages.

- Security Requirements (CR, IR, AR).

Parameters indicating the security importance measures in users' organization. This group of parameters include parameters indicating the confidentiality (CR), integrity (IR), and availability (AR) of the vulnerable component. Higher security requirements might cause more security damages, thus causing more business losses.

Organizations' users should categorize all IT assets according to security requirement measures. Doing so raises the possibility to predict the organizational losses. Federal Information Processing Standards (FIPS) requirements demand implementation of a categorization [6] but does not require using any particular scale, thus risk comparison of users' systems is difficult.

III. The Proposed Framework

Federal organizations are moving from periodic to continuous monitoring implementing SCM's which will improve national cyber security posture [7]. The proposed framework includes two advantages over current practices. First, the environmental parameters are based on the components of the system as updated in the systems' Configuration Management Data Base (CMDB) [8]. This capability enables basing the scoring models to perform predictions of organizational damages on real IT environment rather than on user's evaluations. According to [9] it is impossible for organizations to make precise estimates of the economic losses caused by an attack without having full knowledge of users' IT environment. Reference [10] states that organizations should monitor their network continually, and analyze available vulnerabilities to provide the necessary security levels. Secondly, the information of the environmental components is described in this research is in resolution of data items rather than entire systems, thus enabling focused information in relevance to each data item. The proposed CMS examines a database of published asset vulnerabilities, compares in real time computers' assets for existing exposures and calculates computers' potential losses. Loss evaluation algorithm considers vulnerabilities at the moment they are identified even before software vendors prepares patches and before the organization loads the patches to the operational environment. The CMS's proposed architecture is described in fig. 1. Following, a description of systems' components and processes.

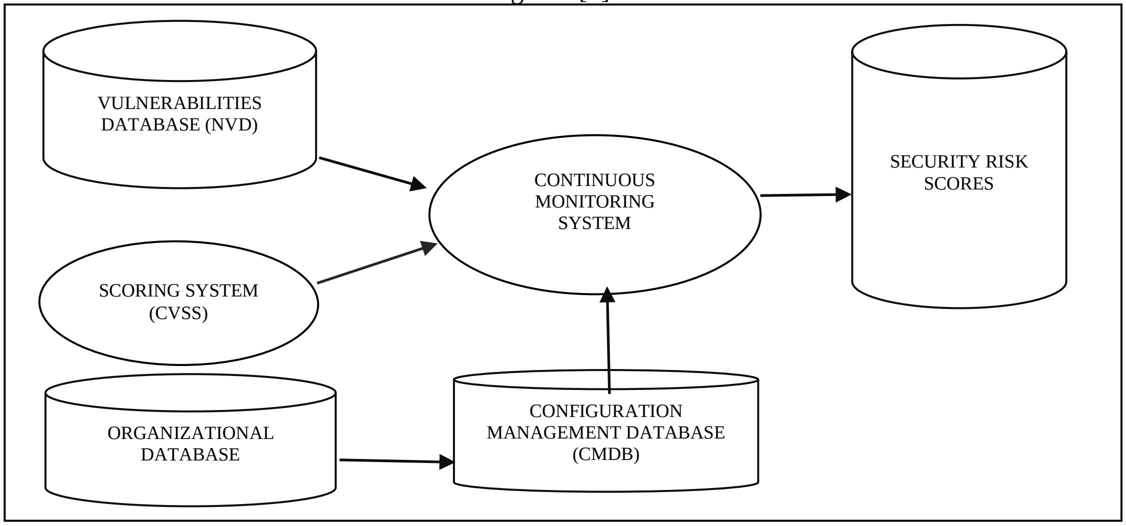
VULNERABILITIES DATABASE (NVD)
CONTINUOUS MONITORING SYSTEM
SECURITY RISK SCORES
SCORING SYSTEM (CVSS)
ORGANIZATIONAL DATABASE
CONFIGURATION MANAGEMENT DATABASE (CMDB)

Fig. 1. Continuous Monitoring System architecture

Vulnerabilities Database includes all known vulnerabilities and their specification as published by Database owners. Examples of vulnerability specifications used by NVD are vulnerability category, vendor name, product name, published start and end dates, vulnerability update dates, vulnerability severity, access vector, and access complexity [6]. Scoring system (CVSS) is the algorithm this research uses for illustration of the proposed model, which computes security risk scores according to parameter groups: basic, temporal and environmental. As stated above there are other known scoring algorithms, some of them for public use other commercial. CMDB is a database which includes all hardware and software components of the target system. According to the proposed model the CMDB manages high resolution information of the organizational database. A frequently running process populates the CMDB, through reading the organizations' database contents. The CMDB contains information of all modules, components, and relationships among the components. The design of the CMDB includes software in the resolution of programs, services and parameters. The design of data is in the resolution of database, tables and data items. Input/output design includes screen-names and output messages in the resolution of data items. The target system might be one computer or a group of organizations' computers. The CMDB also includes all the components which interface with the system directly or indirectly up to external and end-users' interfaces. The CMDB also includes the security requirements (CR, IR, AR) of each component in the resolution of data items. Users define security requirement measures according to business security levels' definitions. The CMDB includes also all interfaces among components. For each interface are indicated the direction of data transfer between the components and probability of connections' occurrences.

The system runs continuously and starts computing losses in two cases: first is whenever a software vendor publishes in NVD a new vulnerability or a change in vulnerability status, second, is whenever systems' owner makes a change in a systems' component or the systems' environment or interface. The system performs evaluations of damage potential using NVD, CVSS, and CMDB. In each case the system identifies a new vulnerable component according to NVD, the system evaluates the new damage potential score and informs the organization. The system writes the computed risk scores on the risk scores database for risk management organizational usage.

In this work, the CMDB includes only a subgroup of all kinds of information of the target computer: high-resolution knowledge of the data entities included in the organizational database, and security requirements of the data entities of the organizational database.

IV. The Scoring Algorithm

CVSS's framework makes use of three kinds of parameters. Vendors who have the best knowledge of their products make estimates of the basic and temporal parameters. Users, who have the best knowledge of their IT configuration, interfaces, and vulnerabilities' business impacts specify the environmental parameters. This work deals with the environmental parameters. According to [6], in many organizations IT resources are labeled with criticality ratings based on network location, business function, and the potential for loss of revenue or life. For example, the U.S. government assigns every unclassified IT asset to a system which is a grouping of assets. Every governmental agency has to categorize systems according to “potential impact” ratings to show the potential impact of systems' compromises on the organization. The categorization should relate to three security objectives: confidentiality, integrity, and availability. Thus, every IT asset in the U.S. government has a potential impact rating of low, moderate, or high with respect to the three security objectives. The Federal Information Processing Standards (FIPS) 199.5 describes this security rating system [11]. CVSS follows this general model of FIPS 199 but does not require organizations to use any particular system for assigning the low, medium, and high impact ratings. Reference [12] states that organizations should define the specifications of security risks of their environment, but does not outline the ways organizations have to specify that information. The Department of State (State) has implemented an application called iPost and a risk scoring program that is intended to provide continuous monitoring capabilities of information security risk to elements of its information technology (IT) infrastructure. According to [13] the iPOST scoring model does not refine the base scores of CVSS to reflect the unique characteristics of its environment. Instead, it applied a mathematical formula to the base scores to provide greater separation between the scores for higher-risk vulnerabilities and the scores for lower-risk vulnerabilities. This work is targeted to close this gap.

The CMDB defined in this work handles the environmental information included in the organizational database in the highest resolution of data items to be able to assign scoring measures to all entities: data items, database tables, and the entire organizational database. The CMDB manages five kinds of environmental information for every data item in the organizational database. Table I describes the parameters defined for each data item. The values for each parameter are based on [11] definitions which categorize parameters according to security information types which are the following: privacy, medical, proprietary, financial, investigative, contractor sensitive, and security management. Reference [11] states that the potential impact is low if the loss of confidentiality, integrity, or availability could be expected to have a limited adverse effect on organizational operations, organizational assets, or individuals. The potential impact is moderate if the loss of confidentiality, integrity, or availability could be expected to have a serious adverse effect on organizational operations, organizational assets, or individuals. The potential impact is high if the loss of confidentiality, integrity, or availability could be expected to have a severe or catastrophic adverse effect on organizational operations, organizational assets, or individuals.

TABLE I. Configuration Management Table Columns

Column ID	Column Name	Column Description	Values (*)
CDP	Collateral	This metric measures the	N, L,

	Damage Potential	potential for loss of life or physical assets through damage or theft of property. The metric may also measure an economic loss of productivity or revenue.	M, MH, H
TD	Target Distribution	This metric measures the proportion of vulnerable systems. It is an environment-specific indicator to approximate the percentage of systems that could be affected by the vulnerability.	N,L,M, H
CR	Confidentiality Requirement	The importance of the affected IT asset to a user's organization, measured regarding confidentiality. "Preserving authorized restrictions on information access and disclosure, including means for protecting personal privacy and proprietary information…"	L,M,H
IR	Integrity Requirement	"Guarding against improper information modification or destruction, and includes ensuring information non-repudiation and authenticity…".	L,M,H
AR	Availability Requirement	"Ensuring timely and reliable access to and use of information…".	L,M,H

(*) N=none, L=low, LM=low medium, M=medium, MH=medium high, H=high

To present the proposed rating system, an introduction of a use case database follows, which will help explain and demonstrate the computations. Tables II and III describe the organizational database and contents. The use case consists of a bank accounts database containing two tables: customers table which contains customers' details and accounts table containing all the details of the loan accounts given to the customers. Each customer may have several loans. Customers' details include customer identification number, customers' name, customers' address, customers' telephone number, customers' salary, and customers' total amount of bank deposits. The details which are categorized as private information items according to FIPS categorization [11] are customer identification and customer name. Items categorized as financial are customers' salary and customers' total deposits. Customers' loan accounts details include account number for identification, account's balance, accounts' date which indicates accounts' opening date, accounts number of months until end loan, and accounts' customer identification. Table IV describes the structure and contents of the CMDB. Each row in the CMDB describes one data item and environmental parameters. Data items of the entire database are in sequential order.

TABLE II. CUSTOMERS TABLE COLUMNS

Cus ID	Cus name	Cus address	Cus telephone	Cus salary	Cus deposit
100	John	Washington	2031234	70000	200000
200	Dan	New York	2025688	90000	50000
300	Scott	Philadelphia	7059876	20000	40000
360	Ben	Boston	7027654	40000	70000
450	Gary	Yale	80175324	60000	8000

TABLE III. ACCOUNTS TABLE COLUMNS

Acc number	Acc balance	Acc date	Acc months	Acc cus ID
3	1000	16.07.15	36	100
5	2400	12.10.14	12	100
8	20	10.8.15	48	300
11	599	19.07.10	100	360
16	50	30.03.13	66	100
23	2000	18.07.12	8	450

TABLE IV. CONFIGURATION MANAGEMENT DATABASE COLUMNS

Item no'	Item name	CDP	TD	CR	IR	AR	Env' Score
1	Cus ID	H	H	H	H	H	5
2	Cus name	MH	H	H	M	M	4
3	Cus address	L	L	M	L	L	0.3
4	Cus telephone	LM	M	H	L	L	2.3
5	Cus salary	LM	L	M	L	L	0.8
6	Cus deposits	MH	L	M	L	L	1
7	Acc number	MH	M	H	H	H	3
8	Acc balance	MH	L	M	L	L	1
9	Acc date	L	L	L	L	L	0.3
10	Acc months	L	L	L	L	L	0.3
11	Acc cus ID	LM	L	L	H	L	0.8

Following, an illustration of data items information security types categorization rational. Security type categorization leads to rational estimations of the environmental parameters. In this use case the security categorization is as follows: The account number data item is private information since it is an important data which hackers often use for identifying customers' information. The account balance data item is clearly financial information. Data items customer ID and account ID are integrity information since their function in the database is primary and foreign keys used for records identification and for keeping on integrity constraints.

The CM Database Columns Table includes all the information of the data items. Table IV includes values of five environmental parameters for each data item in the database: CDP, TD, CR, IR, and AR. Each parameter value indicates the security risk environmental specifications for data items. For illustration, data item customer salary is specified as low-medium CDP, TD is specified as low, confidentiality requirement is medium since the data item is categorizes as financial, IR is specified as low, and AR is specified as low. Following, calculations of the environmental scores using CVSS calculator [14]. The calculator computes the environmental score producing a real number ranging in the interval [0, 5], whereas 0 indicates the minimal impact of the environment on organizational risk and 5 indicates the maximal risk. Using configuration management parameters values and CVSS calculator, the environmental scores are computed and presented in the rightmost column of Table IV. For each data item the calculator uses the environmental parameter values to compute the environmental risk score. For example for the data item customer name the evaluated score is 4 (out of 5), which is an indication of the high impact on organizations' risk. This score is a computed result of a medium-high value of the CDP, high values of TD and CR, and medium values of IR and AR. The rational for the high security score of customer name item follows the high and medium security environmental parameter values. Following, computations of the environmental scores of each table. Computations of all data item environmental scores are presented in the CM table.

Following a formalization of the environmental risk scores.

The symbol i denotes table number.

Column j (i) indicates column number j in table i.

SCORE-TABLE (i) indicates the total environmental risk score of table i. Evaluation of tables' environmental score involves computation of the maximal risk scores of all table columns. The underlying assumption is that in case the organization is facing a damage to the table, all columns cannot be used. Table scores evaluation formula follows in fig. 2.

SCORE-TABLE (i) = MAX (SCORE (COLUMN (j (i))))

For all j ε Table (i)

Fig. 2. Table scores formula

As an illustration SCORE-TABLE (1) indicates the maximal risk score of Customers table which are: 5, 4, 0.3, 2.3, 0.8, 1. Thus, tables' score is 5.

Calculating the environmental risk score of accounts table number 2 involves computing the maximal scores of the columns 3, 1, 0.3, 0.3, and 0.8 which yields a score of 3.

This result represents a higher risk in cases of damage to customers table (4) than to accounts table (3), which may lead to organizational decisions of implementing improved mitigation strategies for the defense of customers table. Assuming that the organization manages several tables in one database, the environmental score of the database will be the maximal score of all tables' scores. This score indicates the environmental impact on the organization in cases of damage to the database as one integral entity. Such cases are relevant when no possibility exists or no knowledge exists concerning which parts of the database were damaged.

Assuming an occurrence of a compromise event to the database, evaluation of the environmental score yields 5 which is the maximum of table environmental scores 3 and 5. Database score evaluation formula follows in fig. 3.

SCORE (DATABASE) = MAX (SCORE-TABLE (i)

For all i Tables in the Database

Fig. 3. Database scores formula

In case an organization manages several databases, the above formula may be applied to evaluate the environmental scores of all databases, yielding different risk scores, leading to varying mitigation actions for certain databases.

V. Results

Fig. 4 illustrates the environmental scores computed by the scoring algorithm. The horizontal axis presents data items from 1 to 11 in our use case. The blue line shows the environmental score of the database as one entity. The red line shows the environmental tables' scores, which are 3 and 5. The green line shows data items environmental scores. In a case when a data item was stolen or damaged the score indicates the risk reflecting the damage to the organization caused by that data item. Data items' scoring calculations yields different scores. For example the computed score of data item number 1 is maximal (5) while the score of data item 3 is minimal (0.3) which leads to the conclusion that scoring all data items with one identical score is clearly far away from the specific characteristics as were specified by the users to the environmental parameters. In cases the organization can be specific about the damaged table then the scores evaluated will be 3 or 5 depending on the table. In cases the organization is unable to identify which data was damaged then a general score of 5 will be assigned, knowing that that score does not reflect the varying tables' risk scores and data items' risk scores included in the database.

As illustrated in this work, basing risks evaluations on high-level information yields higher risk scores which do not reflect the actual real environment and overestimate potential risk scores. The proposed framework enables organizations' risk managers getting improved risk scores based on high-resolution information of their configuration, thus allocating lower budgets to risk mitigation activities. Moreover, risk managers are now able to design risk management work plans based on accurate risk scores, enables being more efficient in risk management and allocating appropriate budgets to mitigate actual risks. Another benefit of the proposed model is using it by system managers and database administrators for database design. They can now use enhanced defense tools to protect their sensitive data from risks, for example, improved encryption techniques for higher security risk scored data items.

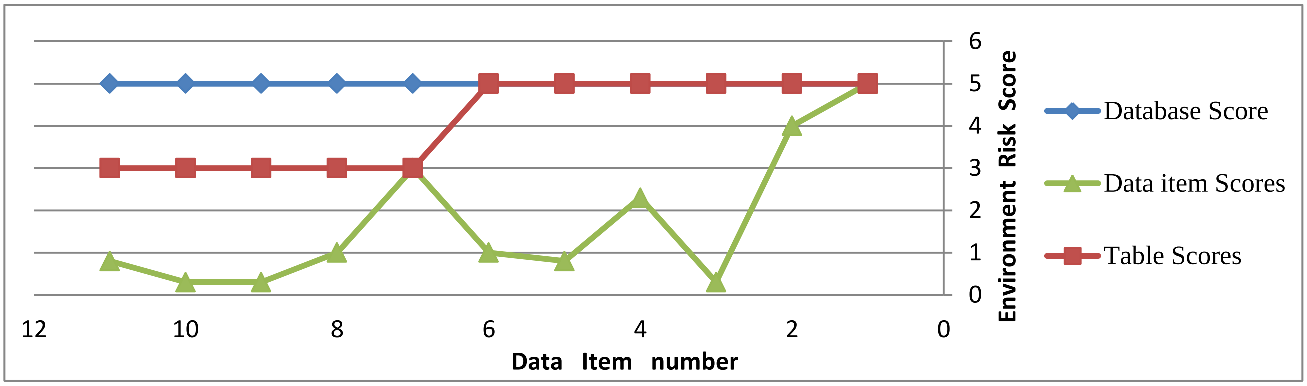

Fig. 4. Relative Environmental Scores

VI. Conclusions

In this work we described a framework of a Security Continuous Monitoring System, structure and mechanisms. The system introduces two new capabilities. First, the proposed model is basing risk scores on the actual environment and configuration of the target system, while existing models use users' estimations to environmental risk scores. Second, risk scoring is based on detailed information of the database in the resolution of data items. According to existing models and known practices, the resolution of the environmental information is defined by means of a whole database or a whole system, thus causing un-accuracies. Current practices produce high-level risk scores while the proposed model produces scores based on detailed information in the highest resolution levels. Accurate risk scoring enables efficient management of risk organizational budgets.

Future research direction may be the design of additional environmental parameters of the system such as hardware, software and communication components, for incorporation in the security scoring model. Another research direction is improving the CMS mechanisms by designing the interface between the organizational database and the CMDB, and automating the interfacing process.

References

[1] P. Mell, K. Scarfone, and S. Romanosky, “CVSS – A complete guide to the common vulnerability scoring system, version 2.0”, 2007.

[2] L. Langer, "Stuxnet: dissecting a cyber warfare weapon, security and privacy", IEEE, Volume 9 Issue 3, pages 49-51, NJ, USA, 2011.

[3] S. Tom and D. Berrett, "Recommended practice for patch management of control systems", DHS National Cyber Security Division Control Systems Security Program, 2008.

[4] A. Terje and R. Ortwin, "On risk defined as an event where the outcome is uncertain", Journal of Risk Research Vol. 12, 2009.

[5] Y. F. Nũez, "Maximizing an organizations' security posture by distributedly assessing and remeding system vulnerabilities", IEEE – International Conference on Networking, Sensing and Control, China, April 6-8, 2008.

[6] K. Dempsey, N. S. Chawia, A. Johnson, R. Johnson, A. C. Jones, A. Orebaugh, M. Scholl and K. Stine, "Information security continuous monitoring (ISCM) for federal information systems and organizations", NIST, 2011.

[7] M. G. Hardy, "Beyond continuous monitoring: threat modeling for real-time response", SANS Institute, 2012.

[8] A. Keller and S. Subramanianm, "Best practices for deploying a CMDB in large-scale environments", Proceedings of the IFIP/IEEE International conference and Symposium on Integrated Network Management, pages 732-745, NJ, IEEE Press Piscataway, 2009.

[9] M. R. Grimalia, L. W. Fortson and J. L. Sutton, "Design considerations for a cyber incident mission impact assessment process", Proceedings of the Intrnational Conference on Security and Management (SAM09), Las Vegas, 2009.

[10] I. Kotenko and A. Chechulin, "Fast network attack modeling and security evaluation based on attack graphs", Journal of Cyber Security and Mobility Vol. 3 No. 1 pp 27-46, 2014.

[11] FIPS Publication 199 - Federal Information proccessing standards publication, "Standards for security categorization of federal information and information systems", Department of Commerce, USA, February, 2004.

[12] E. Weintraub and Y. Cohen, "Continuous monitoring system based on systems' environment", ADFSL - Conference on Digital Forensics, Security and Law, May 19, 2015, Florida, USA.

[13] GAO – United States Government Accountability Office Report to Congressional Request, "Information security – state has taken steps to implement a continuous monitoring application but key challenges remain", July, 2011.

[14] National Vulnerability Database, "Common vulnerability scoring system version 2.0 calculator", https://nvd.nist.gov/CVSS/v2-calculator, retrieved March, 3, 2016.