International Journal of Information Technology and Knowledge Management
July-December 2010, Volume 2, No. 2, pp. 567-571

Security Issue – A Metrics Perspective

Mukta Narang[1] & Monica Mehrotra[2]

Security measurement in software is becoming a somewhat mature field, as evidenced by professional and international standards, specialized conferences, and several decades of literature and research. In spite of this history, security till date is a qualitative measure. For the past 20 years, the International computer security community has been developing criteria and methodologies for the security evaluation of IT products and systems The evaluation processes are highly qualitative as all the evaluation evidence, evaluator's qualification and experience, and evaluation methods are often difficult to quantify. Though every organization in some way or another is measuring the security of software, but still no standard metrics for measuring security of an information system is defined. This paper is an attempt to analyze security measurement methods by different people at different places and bring them all to a common point, which in the future can lead to a standard security metrics.

Keywords: Secure Software, Security Metrics, Security Measurement

1. Introduction

In this era of ever-growing technology, things are changing very fast. The era of software started with the prime focus on developing software. Since last few years the focus has shifted to developing secure software. The recent trend is to quantify security in these software's which for so many years was dealt in a qualitative manner. Security is important at all levels of a system and hence requires a better understanding and management. By security we mean the ability of the system to deliver a service in which the frequency and severity of defined types of service failure are acceptably low.

A more complete and thorough understanding of a subject can be obtained through measurement. As very rightly said by Lord Kelvin in 1883 " When you can measure what you are speaking about and express it in numbers you know something about it." These days we find different security frameworks and standards available but not much work has been done in terms of measuring the same. We can suggest a system to be secure but 'how secure' is still an unanswered question. Realizing the importance of security in software intensive systems all the organizations in some way are taking care of it. The competency in this area has gone that high that people have starting measuring security, but yet no standard has been defined for the same. In this paper we are trying to map different metrics given by different people to one single metrics, which in the future may lead to defining a standard security metric.

[1,2]Jamia Milia Islamia

Email: [1]mukta.narang@gmail.com, [2]monica_mehrotra2000@yahoo.com

2. Current Scenario

Security field is diverse and evolving rapidly for software intensive system. This diversity shows in the different approaches of measuring security, in different organizations. We need indicators of security properties that are good enough to support the types of decision that have to be made at aggregated levels of systems. A popular dictum states, "What gets measured gets done". Hence to measure security we need 'Security Metrics', which transforms policy into action and measures performance. They report how well policies, processes and controls are functioning and whether or not desired performance outcomes are being achieved. In the era where security is becoming the major concern for all software intensive systems, just stating that my system is secure is not enough. There is a need to measure this security to justify the performance of the system. Various people in different organization have made an attempt for the same. Some of these are stated here.

1. Andrew Jaquith, author of the book 'Security Metrics: Replacing Fear, Uncertainty, and Doubt' and originator of Securitymetrics.org, a website open to all security professionals for sharing, contributing and advancing the use of metrics in information security. He believes that security metrics don't have to rely on heavy-duty math to be effective, but they also don't have to be dumbed down to red, yellow, green. He gave few sample questions to check how secure your system [1] is.
2. Microsoft proposed [5] Relative Attack Surface Quotient (RASQ), which attempts to address at least one component of the security assessment process by attempting to mathematically quantify the

relative attackability of IT, assets. The RASQ of a product is calculated by adding together the effective attack surface value for all root attack vectors. The effective attack surface value is further defined as the product of the number of attack surfaces within a root attack vector and the attack bias. An attack bias is the value between 0 and 1, representing the risk of compromise for an attack vector weight given to a particular attack vector that represents the threat of compromise of that attack vector. Assumptions might be made that the attackable surfaces are potential weak spots within a product, such as network services running by default, weakly protected accounts and files, or improperly written code. However, an attackable surface does not necessarily mean a surface is vulnerable. For purposes of this model, an attackable surface is defined as a target for a potential attacker.

Michael Howard, Jon Pincus, and Jeannette M. Wing worked further on RASQ to propose [7] a metric for determining whether one version of a system is more secure than another with respect to a fixed set of dimensions. Howard identified 17 "attack vectors," i.e., likely opportunities of attack. Examples of his attack vectors were open sockets, weak ACLs, dynamic web pages, and enabled guest accounts. Based on these 17 attack vectors, he computed a "measure" of the attack surface, which he called the Relative Attack Surface Quotient (RASQ), for seven running versions of Windows. The other writers added three attack vectors to Howard's original 17 and showed the RASQ calculation for five versions of Windows.

3. Two very famous writers Chenxi Wang and William A. Wulf from the Department of Computer Science, University of Virginia in their paper 'Towards a framework for security measurement' have proposed a framework for security measurement [4]. They have given a systematic way to best approximate the security strength of a system. Security, according to them is system dependent; so it must identify a set of security-related attributes that are important to the use of the system. It must also decide whether the system security is to be represented as a vector or a single value. If a single value is desired, a model to relate the different attributes must also be defined. In some cases, a simple addition of the various ratings can render a sufficient measure while others may require a more sophisticated model such as weighted sum to calculate the final measure.

Table. 1
Sample Questions for Finding
Information Security Weaknesses

Parameters	Diagnostic Questions
The network perimeter is porous, permitting easy access to any outsider.	• How many sites are connected directly to the core network without intermediate firewalls?· How many of these sites have deployed unsecured wireless networks?
An outsider can readily obtain access to internal systems because password policies are weak.	• Starting with zero knowledge, how many minutes are required to gain full access to network domain controllers?· What percentage of user accounts could be compromised in 15 minutes or less?
Once on the network, attackers can easily obtain administrator credentials.	• How many administrative-level passwords could be compromised in the same time frame?
An intruder finding a hole somewhere in the network could easily jump straight to the core transactional systems.	• How many internal "zones" exist to compartmentalize users, workgroup servers, transactional systems, partner systems, retail stores, and Internet-facing servers?
Workstations are at risk for virus or worm attacks.	• How many missing operating system patches are on each system?
Viruses and worms can spread quickly to large numbers of computers.	• How many network ports are open on each workstation computer? • How many of these are "risky" ports?
The firms deployments of applications are much riskier than those made by leaders in the field (for example, investment banking).	• Where does each application rank relative to other enterprise applications [we have] stake have examined for other clients?
Application security is weak and relies too heavily on the "out of the box" defaults.	• How many security defects exist in each business application? • What is the relative "risk score" of each application compared to the others?

This methodology uses a decomposition method to develop such models, starting with high-level security properties of the system, work our way down to the basic components of the system and their interactions. The methodology has five basic steps:

- Decomposition: Identify a set of security-related goal(s) and decompose them at every level to their contributing factors, till the nodes can't be

decomposed any further and all leaf nodes are measurable components that are independent of each other. Note that in such a breakdown, a component can be either a physical subsystem or a logical function, which consists of a set of security properties.

- Functional Relationships: To provide adequate functionality a set of logical relations among system components and the composite rules associated with them are given.
- Weighting and Priorities: While decomposing, sometimes it is necessary to differentiate the relative importance or weights among components. A correct weight assignment is critical because the weights are used to compute the combinatorial effect of the various elements on the overall system.
- Basic Measurements: Most of the security attributes such, as confidentiality and integrity are terms of qualities. In measuring such quality terms, an inherent difficulty is that there might be many different interpretations of what they really mean. Because the overall estimate largely depends upon the basic metrics. Care must be taken in implementing them.
- Component Sensitivity Analysis: The last element of the estimation methodology is a component sensitivity analysis. A sensitivity analysis is performed to assess the impact of variations to the individual components. It allows us to identify a component or set of components' contribution to the overall system security.

3. ANALYSIS OF DIFFERENT METHODOLOGIES

An attempt is being made to give a standard metrics which can be mapped on each of the work specified in Section 2 and still achieve the same or better performance. One can look at this in a way that to measure security we need two things, one a parameter and second how much impact does this parameter have on the security. Since security is not just dependent on any one parameter, so we first have to decide on different parameters. These parameters may be fixed by an organization or may differ as per the project. Once the parameters have been identified the next task is to find ways to measure them. In other words, we have to map the qualitative parameters to some quantitative parameters, which can be used to calculate the security metrics. Thus we try and identify the measurable potential weak spots for the parameter. Depending on the project, these potential weak spots may be a continuous function dependent on some variable or it may simply be a discrete value, which can be measured, based on some criteria. Secondly the impact of every parameter will vary as per the project, so we need to assign some weighted value to the parameter. Based on this a standard security metrics would be:

$$SM = \sum_{i=1}^{n} W_i X_i \quad (1)$$

Where

SM = Security Metrics

n = Number of parameters

W = Weight value

X = Measurable potential weak spots for a parameter

All research work done in this field moves around this metrics. The basic difference lies in which scenario the metrics is working. Based on the project what may differ is, the parameters of security, the weighted values and the method of calculating it. Whatever the case may be but the end result is that security metrics is nothing but the weighted sum of the number of potential weak spots identified within a project. An attempt has been made to map the above mentioned research studies to this security metrics.

1. Consider the sample questions given by Andrew Jaquith as 'n' different parameters in table (1). This is a good example of qualitatively evaluating the security. The same table has been modified in order to map the qualitative parameters to its measurable components. Andrew Jaquith has identified some of the parameters. The diagnostic questions in this table are nothing but the potential weak spots that represent these parameters and the criteria to measure them. This modified version is represented in the following table.

Table. 2
Mapping the Parameters in Table(1) to Measurable Potential Weak Spots(X)

i	Parameters	Criteria for measuring X	Potential weak spots (X)
1.	The network perimeter is porous, permitting easy access to any outsider.	How many sites are connected directly to the core network without intermediate firewalls and have deployed unsecured wireless networks?	The number of sites.

Contd...

Contd...

i	Parameters	Criteria for measuring X	Potential weak spots (X)
2.	An outsider can readily obtain access to internal systems because password policies are weak.	Time	Percentage of user accounts that could be compromised.
3.	Once on the network, attackers can easily obtain administrator credentials.	Time frame	Number of administrative level passwords.
4.	An intruder finding a hole somewhere in the network could easily jump straight to the core transactional systems.	Different types of systems and servers.	How many internal "zones" exist.
5.	Workstations are at risk for virus or worm attacks.	Cross-reference the missing patch with a patch clearing house service.	How many missing operating system patches are on each system?
6.	Viruses and worms can spread quickly to large numbers of computers.	Risky ports.	How many network ports are open on each workstation computer?
7.	The firms deployments of applications are much riskier than those made by leaders in the field (for example, investment banking.	Where does each application rank relative to other enterprise applications [we have] stake has examined for other clients?	Number of applications.
8.	Application security is weak and relies too heavily on the "out of the box" defaults.	Relative risk score.	How many security defects exist in each business application?

Once all these values have been identified, depending on the impact these parameters have on the project some weighted value (W) can be attached to each one of them and the overall security would be a weighted sum of these values, that is:

$$SM = \sum_{i=1}^{n} W_i X_i, \text{ which is same as (1).}$$

2. Microsoft's proposed RASQ can be directly mapped to our security metrics. Let's first simplify the metrics given by Howard. They have identified 20 'attack vectors', which are actually the likely opportunities of attack. According to them:

RASQ = sum of effective attack surface value for all root attack vectors

Effective attack surface value = (number of attack surface within a root vector * Attack bias)

Thus

RASQ = sum of (number of attack surface within a root vector * attack bias)

Mapping this to SM:

n = number of attack vectors

Since 20 attack vectors have been identified, n = 20

Number of attack surface within a root vector = number of potential weak spots(X)

Attack bias (threat of compromise) = a weighted value attached to a parameter (W)

$$RASQ = SM = \sum_{1}^{20} X_i W_i, \text{ which is same as (1)}$$

3. The estimation methodology discussed by Chenxi Wang and William A. Wulf is in sink with the concept discussed in this paper. If S is the overall security represented by root of a hierarchal tree, then level 1 of this tree represents the parameters that would be considered to measure the security of this system. All consecutive levels represent the mapping of these qualitative parameters to their measurable potential weak spots. Thus the leaf nodes finally would represent the measurable components of these parameters, which are independent of each other. According to the authors any system can be divided into a hierarchal process, where in all the lower nodes are contributing in measuring the assessment score of security of the parent node. The nodes in the hierarchy have some functional relationship, which is given by some logical operations. These operations are somewhere a weighted contribution of all the child nodes to their parent nodes. This process moves up the hierarchy till a definite number at the root is known, which is the security measure of the system as a whole. Thus if we can prove that the overall security is some weighted sum of the leaf nodes then this goes with the security metrics proposed in (1).

We prove this using a functional relationship defined by Chenxi Wang and William A. Wulf, called Prioritized Siblings (PS) which is a relation existing among siblings each contributing to an independent aspect of the parent function. The failure of a single sibling will not necessarily cause the functional failure of the parent. Formally, PS can be described as:

$$S(\text{parent}) = \sum_{i=1}^{n} (S_i * W_i) \quad (2)$$

where S is the assessment score and W is the weight for n number of child nodes. Thus moving up the tree we can say that each level is a combined effect of the nodes in the underlying level. Thus the root is nothing but a weighted sum of all the nodes in the tree. Since we consider all the nodes as some factor contributing to the security of a system, decomposed at different levels, the overall security can be considered as equivalent to:

$S = \sum_{i=1}^{n} W_i X_i$, where i represents the levels in the tree, $1 \rightarrow n$. (3)

W represents the combined effect of weight of the child nodes

X represents the total effect of the scores(S) of child nodes

That is: $S = S_{root} = W_1 X_1 = W_1 S_1$ (4)

$= W_1$(weighted sum of the child nodes S_2 and S_3)

$= W_1(W_2 S_2 + W_3 S_3)$

$= W_1 W_2 S_2 + W_1 W_3 S_3$

$= W_v S_2 + W_y S_3$

(Expanding the scores S_2, S_3 to the next level where S_4, S_5 are child nodes of S_2 and S_6, S_7 are child nodes to S_3)

$$= W_u S_4 + W_w S_5 + W_x S_6 + W_z S_7$$

Where, W_v, W_y, W_u, W_w, W_x, W_z are a combined effect of weights of respective parent nodes of higher levels.

We can further expand these scores till we reach the leaf nodes of the tree, where further decomposition of the tree is not possible. Thus simplifying this we can say that security metrics is a weighted sum of the scores of the leaf nodes of the tree. The leaf nodes are the independent measurable components of the attributes that contribute to the overall security.

$$\text{Thus security (S)} = \sum_{\text{leaf nodes}} WX \quad (5)$$

Where W is the combined effect of weights at all higher levels

X is the score that can be measured.

This goes with (1), that is, security is a weighted sum of different parameters.

4. CONCLUSION

There's software everywhere, and when you put some faith into the idea that software is going to do what you expect it to do, you end up with the opportunity for a security problem. Traditionally, quantifying information technology security elements and risk exposure was a difficult exercise, due to the number of intangible factors involved, and lack of common standards and vendor support. The good news now is that there is a market for software security and software security tools. That means people are starting to realize they have got new risks in the form of security problems that they have to take into account when they build software. Though much is being done in the field of security, but we are losing ground in its complexity. We are building bigger and bigger systems at an astounding rate, so our complexity is growing exponentially, and our ability to manage that complexity is not growing with it. You can see the evidence of that as new types of vulnerabilities start to arise, without fixing what came before them.

Security managers look for a magic formula that they can apply to measure security of a system. This is not as easy at it appears. This paper brings you closer to an analytic process of security measurement. Though we have tried to give a metrics which is taking care of different metrics being used at different organizations but still a lot has to be done in this area. The challenge lies in standardizing a method to assign weights to different parameters and identifying the measurable potential weak spots for different parameters.

REFERENCES

[1] A. Jaquith, "Sample Questions for Finding Information Security Weaknesses", CSO Online, May 2007.

[2] A. Jaquith, "Security Metrics: Replacing Fear, Uncertainty, and Doubt", Addison Wesley April 2007.

[3] A.J.Wang, "Information Security Models and Metrics", ACM Southeast Regional Conference Proceedings, 2005.

[4] C.Wang, W.A.Wulf, "A Framework for Security Measurement", 20th NISSC Proceedings, October 1997.

[5] Ernst & Young, "Using Attack Surface Area and Relative Attack Surface Quotient to identify Attackability", March 2003.

[6] M.Betts, "Knowledge Center Security", Computerworld April 2006.

[7] M.Howard, J.Pincus, J.M.Wing, "Measuring Relative Attack Surfaces", August 2003.