Modeling and Assessment of Systems Security

Jonas Hallberg, Johan Bengtsson, and Niklas Hallberg

Swedish Defence Research Agency, Olaus Magnus väg 42, SE-583 30, Linköping, Sweden
{jonas.hallberg, johan.bengtsson, niklas.hallberg}@foi.se

Information technology (IT) is a crucial resource and enabler in almost every part of our society. However, there are severe risks associated with IT that may substantially decrease the potential benefits. To handle these risks, it is essential to be able to judge the security posture of systems. This requires the ability to perform security assessments. However, since security is an abstract, subjective, and non-tangible property, proper security assessment of non-trivial systems is hard. Currently, there is a lack of methods for efficient, reliable, and valid security assessments. In this paper, problems relating to the structural assessment of system security are addressed. In structural security assessments, the security of systems is quantified based on the security qualities of and inter-relations between sub-systems.

1 Introduction

Information technology (IT) has become an important tool for individuals as well as organizations. This results in dependencies, which implies vulnerabilities. Hence, to exploit the benefits that IT based systems can provide and concurrently avoid the negative effects of unpredictable and unwanted behavior, it is essential to be in control of the security. In this paper, the term security is used in the meaning of IT security, which consists of upholding the characteristics of confidentiality, integrity, and availability of IT systems and the data processed, transmitted, and stored in these systems. However, depending on the context of the system, the security assessor, and the users of the security assessment results, other characteristics may be included in the concept of security.

The design and implementation of efficient, reliable, and valid methods for security assessments poses major challenges [1-5]. Largely, this stems from the fact that security is an abstract, subjective, and non-tangible property. This results in:

- difficulties to decide what is actually meant with security,
- the belief in *secure* as an achievable, ever-lasting property of information systems,
- security not being possible to measure, instead other system characteristics and effects have to be measured in order to enable estimations of security levels, and
- difficulties to interpret the meaning of security-relevant system characteristics and effects.

An approach to manage complex problems, like security assessment, is to divide these problems, study the resulting sub-problems, and aggregate the results. In relation to security assessment, this approach results in the following questions that need to be addressed.

- How can methods starting from data on security-relevant characteristics of system entities produce system-level security values?
- Which specific security assessments user needs are addressed by such methods?
- What level of effort is required to produce useful results?

The results presented in this paper contribute to the first of the questions stated above through the design of the eXtended Method for Assessment of System Security,

XMASS. Further, initial studies relating to the third question is presented through the collection of data required for security assessment based on the XMASS.

2 Security Assessment

Security assessment is performed in order to establish how well systems meet specified security criteria. Hence, the aim of security assessment is to produce knowledge. This knowledge can be used to, e.g., determine the needs of improving the security levels of the assessed systems. Security assessments can provide insight into the security posture of systems, but (for complex systems) it cannot guarantee any level of security. Though, it provides a basis for confidence in the assessed system [6].

Although IT security deals with technical elements, comprehensive security assessments need to consider other interrelated aspects, such as the organizational, human, and contextual aspects. The inclusion of these aspects emphasizes the need to consider their influence on the security levels of systems. However, security assessment does not include the assessment of the security of organizations, persons, and contexts themselves.

2.1 Security Assessment Approaches

Several approaches to security assessment have been presented. Security metrics programs refer to the process of identifying measurable system effects, which supposedly are caused by relevant security characteristics of the system, measuring these security effects, and presenting them in an illustrative way [7-9]. Adequate security metrics should be consistently measured, inexpensive to collect, expressed by numbers, and have a unit, such as seconds [8]. The interpretation of specific security metrics is left to the user. Proponents of security metrics programs claim that the characteristic of triggering discussions on the meaning of the presented results is a key benefit. In contrast, the approach presented in this paper, the XMASS, aims at providing system-wide security assessment values including the effects of system structure and interconnections. Thus, the whole system is considered during the assessment rather than isolated system effects.

Attack-based methods assess systems based on the steps that adversaries have to complete in order to achieve their goals, e.g., [10-11]. The aim of the weakest-adversary security metric method is to enable the comparison of different system configurations based on the attributes required to breach their security [11]. Characteristics of network configurations and current attack stages, e.g. root-level shell access on host A, form the states of systems. The transition rules describe the requirements for and consequences of the transitions from system states into other system states. Describing the actual prerequisites of successful attacks, the presented results are intuitive. However, the analysis of results may not be as straightforward, e.g., in order to compare the system effects resulting from different system configurations. The XMASS does not require the knowledge of specific vulnerabilities that can be used to penetrate systems, instead assessments are based on the security qualities of systems.

System characteristics-based methods combine characteristics at the system-level to receive security values representing the security levels of complete systems. The Security Measurement (SM) framework is used to estimate scalar security values corresponding to relevant security characteristics [12]. For the transformation of relevant security characteristics into measurable system effects or characteristics a decomposition method is described. The outcome is a tree with measurable security characteristics as leaves. For the aggregation of security values, a method to calculate

weights in the resulting tree and functional relationships modeling the interactions between these factors are used. Because of the generality of the method large efforts are required to design specific methods based on the framework. While XMASS-based assessments can utilize different sets of security characteristics to capture the security levels of systems, the process of systems modeling is more clearly specified. Like security metrics programs, the SM framework lacks support for the capturing the security effects of system structure, which is explicitly supported by the XMASS.

3 From Modeling to Security Values Computation

In order to assess the security of systems, it is essential to capture the underlying characteristics and effects related to the system as well as to define how the computation of security values should be performed. Thus, systems and computations have to be modeled. Provided these models, the base data has to be captured and the aggregated values have to be computed to receive the final assessment results (Fig. 1).

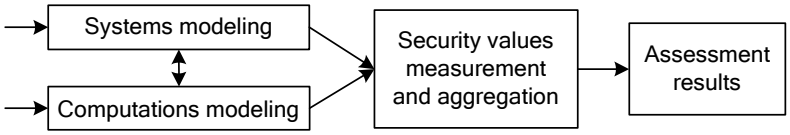

Fig. 1. The outline of methods for security assessment.

The eXtended Method for Assessment of System Security (XMASS) has been formulated according to the structure in Fig. 1. XMASS supports the security assessments of networked information systems. Assessments are based on the available knowledge regarding the security characteristics of the system entities and their relations [5].

The system modeling is supported by the possibility to create profiles for standardized system entities and their relations. There are no explicit limitations in the method disabling the inclusions of different system aspects. The computation of higher-level security values is controlled by the computations model, which can be specified by the user, but is tied to the structure of the system. Thus, the computation of aggregated security values, not just the input, depends on the system models. In the following subsections, the parts of the XMASS relevant for this paper are presented. The central concepts of XMASS are illustrated in Fig. 2.

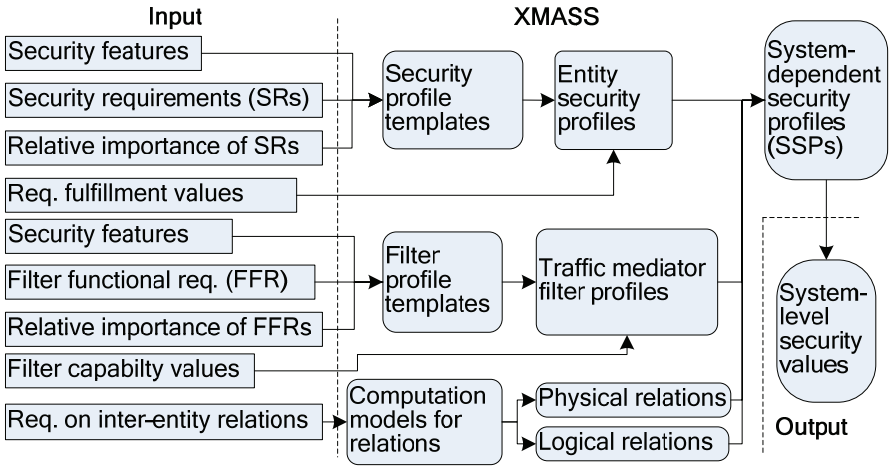

Fig. 2. The central concepts of XMASS and their relations.

3.1 Systems Modeling

In XMASS, systems are modeled as entities and their inter-relations. To reflect the difference between entities that produce and consume respectively mediates data, they are divided into traffic generators and traffic mediators. Traffic generators are, e.g., web servers, workstations, and printers. Traffic mediators are, e.g., hubs and firewalls. The security characteristics of system entities are described by security profiles consisting of vectors with security values. To capture the ability of traffic mediators to shield the generators from security weaknesses of their peers, filter profiles are associated to the traffic mediators. The security values are affected by the way entities are interconnected, both physically through networks and logically through the inter-entity dependencies between security functions. Thus, the inter-relations between entities are described through logical and physical relations.

The physical relations describe associations between entities through physical means, e.g., wired or wireless communication. The physical relations are bi-directional and symmetric. They are modeled as the physical layout and topology of the studied system and contain no properties except the connected entities. The effects of physical relations are modeled thorough a profile, which is common for all the physical relations in a system. The logical relations are used to model dependencies and interaction between the security mechanisms of the entities. Thus, the logical relations differ depending on the characteristics of the captured relation. Moreover, logical relations are uni-directional since their effects are asymmetric. Consequently, the logical relation profile is selected per relation. Thereby, it is possible to model different kinds of logical relations, such as those resulting from Virtual Private Networks and the relation between a workstation and a central server managing the anti-virus software. A relation profile, whether physical or logical, consists of a matrix of weights and a matrix of functions.

To model the security posture of systems, system-dependent security profiles (SSPs) are associated to the traffic generators. Like the security profiles, the SSPs consist of vectors with security values. However, the security values of the SSPs are influence by the security values of the neighbors of the current entity as well as the security values of the entity itself. The concept of *neighbors* is defined as entities that are connected (via relations) directly or through one or several traffic mediators. The system security posture can be assessed through the aggregation of the security values of the SSPs.

Although, the security and filtering characteristics of the individual system entities are described through the values included in the security and filter profiles, these values of the profiles are not necessarily directly measurable. Consequently, to support the creation of profiles, the concept of profile templates is introduced. Profile templates are used to define how the underlying characteristics of the entities affect the security values of their profiles. The creation of security and filter profile templates is presented in the following section.

3.2 Computations Modeling

This section presents the notation and central steps used in the specification of the computations model of XMASS.

Entity Security Profile Templates

A security profile template is a set of formulas used to compute the security values of the security profiles. The definition of security profile templates captures two main aspects of the assessment. Firstly, the selection of security features to be represented by the security profiles. This reflects the view on security taken by the method designers, which should be based on the needs of the end users. Secondly, the

prioritization of the underlying characteristics is used to compute the security values of the security profiles. There is a single security profile template active at any time during the computation of the security values of a modeled system. The security profile template is designed through the following steps:

1. decide the set of security features to be represented by the security profile,
2. for each security feature, decide on a set of entity security requirements describing what needs to be fulfilled in order to fulfill the security feature,
3. for each security feature, divide the entity security requirements into the sets of fundamental and important security requirements,
4. for each set of important security requirements, prioritize the security requirements based on their relative importance, and
5. calculate the security profile template based on the data produced in steps 1 to 4.

The method for criteria weighting from the Analytic Hierarchy Process (AHP) is used to derive the prioritization from the judged relative importance among the important security requirements [13]. Unfulfilled fundamental requirements results in security values of zero. Thus the fulfillment values of the fundamental requirements are multiplied with the weighted sum of the important security requirements.

For each security feature, *k*, the set of security requirements is divided into one set of fundamental and one set of important security requirements, $\mathbf{RF}_k$ and $\mathbf{RI}_k$ respectively. For each security feature the relative importance between each pair in the set of important requirements, ri_i, $ri_j \in \mathbf{RI}_k$, is decided according to Table 1 by assigning weights, a_{ij}. The weights result in matrices, $A_k = \{a_{ij}\}$, which describes the relative importance of the pairs of important security requirements, $\mathbf{RI}_k$ of the security feature *k*.

A requirement, ri_i, considered more important than another requirement, ri_j, results in a corresponding weight $a_{ij}>1$. Correspondingly, $a_{ij}<1$ express less importance of the former requirement compared to the latter, while a value of 1 represents equal importance. Values less than 1 are constructed by reversing the comparison, that is, comparing the latter with the former requirement, and using the reciprocal value, $a_{ij}=1/a_{ji}$.

The prioritization of the important requirements, $\mathbf{RI}_k$, of security feature k is decided by calculating and scaling the eigenvector, $e_k = \{e_{ki}\}$, which corresponds to the largest eigenvalue, λ_{max}, of the matrix A_k. The eigenvector is scaled so that $\sum_i e_{ki} = 1$. For matrices with properties like those that A_k has, it can be shown that λ_{max} will be slightly larger than the dimension of the matrix and the rest of the eigenvalues will be close to zero [14].

The values of the security profiles are computed from the fulfillment values that are assigned to entities considering the included fundamental and important requirements (Eq. 1). Since the fundamental requirements of each security feature should be decisive for the respective security value, the degree of fulfillment for each of these requirements is included as a factor (as in multiplication) in the security profile template. The fulfillment values corresponding to the important requirements are included as a weighted sum. The combined influence of the important security requirements should depend on the relation between the number of fundamental and important security requirements. Consequently, the weight of the important security requirements, corresponding to a security feature *k*, is defined as $n_k/(m_k+n_k)$, where n_k is the number of important requirements and m_k is the number of fundamental requirements for security feature *k*. Consequently, the weight of the fundamental security requirements becomes $1-(n_k/(m_k+n_k))= m_k/(m_k+n_k)$.

$$SP_k = \left[\prod_{j=1}^{m_k} rfv\left(rf_{kj}\right)\right] \cdot \left(\frac{m_k}{m_k + n_k} + \frac{n_k}{m_k + n_k} \cdot \left(\sum_{i=1}^{n_k} rfv\left(ri_{ki}\right) \cdot e_{ki}\right)\right) \quad (1)$$

Table 1. The weights used when deciding the relative importance of requirements [13].

Requirement weight	1	3	5	7	9
Meaning (importance)	Equal	Moderate	Strong	Very strong	Extreme

Traffic Mediator Filter Profiles Templates

During the computation of the system dependent security profiles (SSPs), filter profiles are used to specify how the filtering capabilities of the traffic mediators affect the security values of the SSPs. The filter profiles are computed using filter profile templates. To decide the filtering profile template, the filtering functionality of the network entities is characterized. This is done by creating a set of filter functional requirements, **FFR**, which is used to assess the filtering capability of all traffic mediators.

Like for the security profile templates, the requirements are prioritized regarding their relative importance by using the process for criteria weighting in the AHP [13]. However, the same set of filter functional requirements is used for all the filter profile values. Hence, for each filter profile value, FP_k, the filter functional requirements, ffr_i, $ffr_j \in$ **FFR**, are pair-wise prioritized according to their relative importance, resulting in weights, b_{ij}. The weights result in matrices, $B_k=\{b_{ij}\}$. Subsequently, the prioritization is decided by calculating and scaling the eigenvector, $e_k=\{e_{ki}\}$, which corresponds to the largest eigenvalue, λ_{max}, of B_k. The scaling of the eigenvector is done so that $\sum_i e_{ki}=1$.

Traffic mediators are unfortunately unable to block all the traffic affecting the security level of the traffic generators. This inability is modeled by including an influence factor, $S_k \in [0, 1]$, in the filter profile template. Thereby the maximum value of each element, k, in the filter profile is S_k. The traffic mediators are characterized in order to assess their influence between traffic generators in a system (Eq. 2).

$$FP_k = S_k \cdot \sum_{i=1}^{N} \left(e_{ki} \cdot fcv_k \left(ffr_i \right) \right) \tag{2}$$

The function $fcv_k(x)$ returns the filtering capability value of requirement x regarding security feature k.

Computations Model for System-Dependent Security Profiles

The security profile and filtering profile templates provide the means to transform the basic input to the XMASS into the security and filtering profiles which constitute the input for the calculation of the system-dependent security profiles (SSPs). The SSP of an entity represent the security posture considering the status of the neighboring entities and constitutes the base for the computations of system-level security values. The computation of the SSPs is performed by assessing each physical and logical relation to reveal the effect it has on the studied entity. Thereafter, the effects of all relations are combined, resulting in the SSP of the entity.

Combination of Multiple Paths between Traffic Generators

It is hard to exactly model the impact of physical connections on the security values of the entities. From a security perspective, it is the interactions between the entities, enabled by the physical connections, that are of interest. In XMASS, those interactions are modeled at the level of the security features. That is, the mathematical functions used to model the relations should capture the effects on the security values corresponding to each specific security feature. To capture the effects of the physical connection to a neighbor, a resulting security profile, RSP, is computed. RSPs include the effects of multiple paths and traffic mediators between the current entity and its

neighbors. The computation is based on the equivalent security profiles, ESPs, resulting from the effects of individual paths between neighbors (Eq. 3).

$$RSP_{i,nb} = \min\left(ESP_{i,nb_1},\ldots,ESP_{i,nb_n}\right) \tag{3}$$

Thereby the resulting security profile, RSP_{nb}, is the element-wise minimum of the equivalent security profiles, ESPs, of the paths.

Each element of the ESPs of a neighbor, *nb*, is calculated considering its security profile, SP_{nb}, and the effective filter profile, EFP_{tm}, of the traffic mediator *tm* (Eq. 4). This calculation can be carried out iteratively for any number of traffic mediators connected in series with a traffic generator. A path with no traffic mediator in it, that is, a direct connection between two traffic generators, results in an equivalent security profile of the neighbor identical to the security profile of the neighbor, that is $ESP^{nb}=SP^{nb}$.

$$ESP_{i,nb} = SP_{i,nb} + \left(1 - SP_{i,nb}\right)\cdot EFP_{i,tm} \tag{4}$$

The ability of a traffic mediator, *e*, to filter malicious traffic is represented by its filter profile, $FP_e \in [0, 1]^N$. However, to include the possible effects of the security weaknesses of traffic mediators themselves, the effective filtering abilities are modeled through a combination of the filter and security profiles of the traffic mediators and are represented by the *effective filter profile*, $EFP \in [0, 1]^N$. The effective filter profile is calculated as the filtering profile multiplied with the weighted average of the security profile of the traffic mediator (Eq. 5).

$$\mathrm{EFP_e} = \mathrm{FP_e}\cdot\sum_{i=1}^{N}\mathrm{w_i}\cdot\mathrm{SP_{e,i}} \tag{5}$$

w is a vector of the size of the profiles (*N*), where $\sum_{i=1}^{N}\mathrm{w_i} = 1$.

Logical Security Profile

The effect of logical relations between entities on the SSPs of the entities is modeled with the *logical security profile*, LSP (Eq. 6). A logical security profile represents the SSP that the entity would have if this logical relation was the only relation to other entities.

$$\mathrm{LSP_{e,lre,l}} = \mathrm{g_{e,lre,l}}(\mathrm{SP_e}, \mathrm{SP_{lre}}) \tag{6}$$

SP_e is the security profile of the entity under evaluation, *e*, and SP_{lre} is the security profile of the logically related entity, *lre*. The function $g_{e,lre,l}$ represents the logical relation *l* and is on the same form as the function *f* for calculating the physical security profile, but it is uniquely specified for each type of relation.

System-level Security Values

Based on the system-dependent security profiles of the entities, system-level security values can be calculated. How the results of a security assessment are best presented depends on the specific needs of the users. Some of the possible approaches are:

- A *system-wide security profile* calculated as the weighted average of the system-dependent security profiles of the entities. The weights represent the importance of each entity; an important server would have a high weight and a peripheral client would have a low weight.
- A scalar *system-wide security value* calculated as the weighted average of the system-wide security profile described above.

- An *entity categorization* using the categories low, medium, and high security value. This can be presented, for example, as a coloring of the system model or by the number of entities in each category.
- *Hot-spot identification* regarding the weakest entity or the weakest security feature to indicate where more security measures are needed.

3.3 Security Values Measurement and Aggregation

The first issue to be resolved when performing security assessments based on XMASS is the formulation of the necessary profile templates. The following examples regarding a security profile template relates to the security requirements specified by the Swedish Armed Forces for information systems [15], referred to as the KSF (Sw. Krav på SäkerhetsFunktioner, requirements on security functions).

The first step is to decide on the set of security features to use. The KSF defines a set of seven security functions. Five of these relates to the task of the XMASS and are therefore identified as security features. These are: Access Control, Security Logging, Protection against Intrusions, Intrusion Detection, and Protection against Malware.

The second step is to decide on the sets of security requirements to be used for the computation of the security values corresponding to the identified security features. Considering the security feature Intrusion Detection (ID), twelve security requirements are identified from the KSF, e.g. *No registered events are erased, overwritten or destroyed*.

The third step involves the division of the set of security requirements into the sets of fundamental and important security requirements for each security feature. If a security requirement is judged to be of such importance that failure to fulfill it would ruin the security feature, it is a fundamental security requirement, else it is considered to be an important security requirement. Nine security experts were asked to divide the security requirements for ID into the sets of fundamental and important security requirements. The aggregated result of the expert survey is four fundamental security requirements, e.g. *Interpretable presentation and possible inspection*, and eight important security requirements, e.g. *Automatic analysis*.

The fourth step is to prioritize the important security requirements according to their pair-wise relative importance. The individual prioritizations of seven security experts were aggregated by taking the median of the weights provided by the security experts for each pair-wise comparison (Table 2).

Table 2. The weights resulting from the comparisons of the important security requirements for intrusion detection.

	ID2	ID4	ID5	ID7	ID8	ID9	ID10	ID11
ID2	1	1/3	3	1/3	1/3	1/5	1/3	1/3
ID4	3	1	3	1	1	1/3	1	2
ID5	1/3	1/3	1	1	1/3	1/5	1	1
ID7	3	1	1	1	3	1/3	1	1
ID8	3	1	3	1/3	1	1/3	1	3
ID9	5	3	5	3	3	1	3	3
ID10	3	1	1	1	1	1/3	1	3
ID11	3	1/2	1	1	1/3	1/3	1/3	1

The fifth step is to calculate the scaled values of the eigenvector corresponding to the maximum eigenvalue of the matrix in Table 2. The calculations result in the

prioritization of the important security requirements and the security profile template (Eq. 7).

$$SP = rfv(ID1)\cdot rfv(ID3)\cdot rfv(ID6)\cdot rfv(ID12)\cdot\left(\frac{1}{3}+\frac{2}{3}\begin{pmatrix} rfv(ID2)\\ rfv(ID4)\\ rfv(ID5)\\ rfv(ID7)\\ rfv(ID8)\\ rfv(ID9)\\ rfv(ID10)\\ rfv(ID11)\end{pmatrix}^{T}\cdot\begin{pmatrix} 0.05642708\\ 0.12638000\\ 0.06286048\\ 0.13170758\\ 0.12523142\\ 0.29879861\\ 0.12094164\\ 0.07765318\end{pmatrix}\right) \quad (7)$$

To judge the consistency of the resulting priorities, a consistency ratio can be calculated ($CR=((\lambda_{max}-n/(n-1))/RI$) [16], where λ_{max} is the maximum eigenvalue, n is the dimension of the matrix, and RI is the random index for the specific matrix dimension. For this example, the dimension of the matrix is 8, which gives a random index of 1.40, and the maximum eigenvalue is 8.83865523. This results in a consistency ratio of 0.085577. As a guideline the CR should not be greater than 0.10 in order to have a consistent decision [16]. The priorities, based on the aggregated matrix, thereby seem to represent a consistent decision.

When the security profile template has been defined, security profiles can be computed for all the different kinds of entities included in the system model. To achieve this, fulfillment values have to be decided for each requirement of each security feature for each entity type. The fulfillment values state how well each security requirement is met by the considered entity. A fulfillment value of 1 represents complete fulfillment of the corresponding requirement, whereas a value of 0 denotes non-compliance. Partial fulfillment is expressed by a fulfillment value between 0 and 1.

5 Discussion

Assessing the security of networked information systems is difficult, but nevertheless important. Two of the main reasons for the importance of efficacious security assessment methods are (1) the ongoing integration of systems, which makes it impossible to comprehend the resulting systems and the security effects caused by all actions affecting the system without the aid of proper security assessment methods, and (2) the need to incorporate security mechanism and thinking in all processes relating to these systems, which results in security assessment needs relating to, for example, systems requirement engineering and configuration management.

In reality, all security assessments should be based on the assessment needs of the user of the produced results. The work described in this paper is based on the general needs, i.e., the need for efficient, reliable, and valid methods for security assessments. Future work should include the specification of and adaption to more specific user needs.

This paper addresses the question regarding how methods starting from data on security-relevant characteristics of system entities can produce system-level security values. With this purpose the eXtended Method for Assessment of System Security (XMASS) has been designed. The XMASS illustrates the possibility to capture the knowledge of security experts through the design of profile templates. To verify the relevance, validity, and reliability of XMASS tools supporting the use of the method are required. For this purpose the environment for the implementation of security assessment methods (NTE), and the Security AssessmeNT Application, SANTA,

have been implemented. The purpose of NTE is to reduce the effort required to implement tools supporting different security assessment methods. This is accomplished by providing common functionality, e.g. file handling. The software implementation of XMASS, called SANTA, has been implemented using NTE.

Further, to address the question on what level of effort is required to produce useful results, initial studies relating to the third question are presented through the collection of data required for security assessment based on the XMASS. These results illustrate how the knowledge of security experts can be captured in the formulation of the basic building blocks used to model systems in the XMASS.

There are numerous tasks that should be undertaken in order to further support the development of structural security assessment methods and tools, such as:

- a set of profiles for the XMASS should be assembled,
- a study of the underlying reasons for the assessment results provided by the XMASS should be undertaken,
- alternative methods for the selection of priorities for the important security requirements and the filter functional requirements used in the XMASS should be studied, and
- real-word assessment should be performed with XMASS.

References

1. ACSA, Ed. Proc. Workshop on Information Security System Scoring and Ranking (2002), Applied Computer Security Associates.
2. Vaughn, R., Henning, R., and Siraj, A. Information assurance measures and metrics – state of practice and proposed taxonomy. In Proceedings of the Hawaii International Conference on System Sciences (HICSS-36), Waikoloa, Hawaii (January 2003).
3. Seddigh, N., Pieda, P., Matrawy, A., Nandy, B., Lambadaris, J., and Hatfield, A. Current trends and advances in information assurance metrics. In Second Annual Conference on Privacy, Security and Trust (October 2004).
4. Geer, D. Measuring security, lecture notes, training program m3measuring security, lecture notes, training program m3. 15th usenix security symposium, vancouver, canada. 2006.
5. Hallberg, J., Hallberg, N., and Hunstad, A. Crossroads and XMASS: Framework and method for system it security assessment. Tech. rep., Swedish Defence Research Agency, FOI, 2006.
6. Bishop, M. Computer Security – Art and Science. Addison-Wesley, 2003.
7. Swanson, M., Bartol, N., Sabato, J., and Hash, J. Security metrics guide for information technology systems. Tech. rep., National Institute of Standards and Technology, July 2003.
8. Jaquith, A. Security metrics: replacing fear, uncertainty, and doubt. Addison-Wesley, 2007.
9. Herrmann, D. Complete guide to security and privacy metrics: measuring regulatory compliance, operational resilience, and ROI. Auerbach Publications, 2007.
10. Laing, B., Lloyd, M., and Mayer, A. Operational security risk metrics: Definitions, calculations, and visualiztions. Presentation at Metricon 2.0, August 7, 2007, Boston, 2007.
11. Pamula, J., Jajodia, S., Ammann, P., and Swarup, V. A weakest-adversary security metric for network configuration security analysis. In Proceedings of the 2nd ACM Workshop on Quality of Protection (October 2006).
12. Wang, C., and Wulf, W. A framework for security measurement. In Proceedings of the National Information Systems Security Conference (Oct 1997), pp. 522–533. Baltimore, MD.
13. Saaty, T. Fundamentals of Decision Making and Priority Theory – with the Analytic Hierarchy Process. RWS Publications. Pittsburgh, USA, 1994.
14. Forman, E., and Selly, M. Decisions by Objective - How to Convince Others That You are Right. World Scientific Publishing Company, 2002. ISBN 978-9810241438.
15. SWAF. Krav på säkerhetsfunktioner – grunder. Tech. rep., Swedish Armed Forces, 2004.
16. Saaty, T. Decision making - the analytic hierarchy and network processes (AHP/ANP). Journal of Systems Science and Systems Engineering 13, 1 (March 2004), 1–35.