A MODEL FOR ASSURANCE OF EXTERNAL PROPERTIES OF INFORMATION SECURITY REQUIREMENT BASES

Jussipekka Leiwo, Chandana Gamage and Yuliang Zheng
Peninsula School of Computing and Information Technology
Monash University
McMahons Road, Frankston, Vic 3199, AUSTRALIA
Tel. +61-(0)3-9904 4287, Fax +61-(0)3-9904 4124
E-mail: {skylark,chandag,yuliang}@mars.fcit.monash.edu.au

ABSTRACT

Harmonization of information security harmonization requirements is a process of transforming abstract information security objectives into concrete information security policies in an automated manner. The process is designed to also provide assurance from internal properties, such as consistency and correctness, of an requirement base, collection of information security requirements. Harmonization is, anyhow, not capable of dealing with external properties of a requirement base, such as comprehensiveness. A model shall be proposed to provide assurance for the external properties of a requirement base in the harmonization framework.

1 INTRODUCTION

Significant amount of research in computer security was triggered by formal access control models, the best known being the Bell-LaPadula model (BLP) for confidentiality [9]. Well established scientific foundation of access control models lead to several other models to protect confidentiality and integrity of data, and to investigate information flows in order to prevent unauthorized disclosure of information. The major attack scenario was a logical trojan horse attack where malicious software attempts to bypass rules of dealing with classified information, namely disclosing data to lower clearance levels. Further, several logics and models were established to reason about security of systems based on access control models. Other logics were develop to evaluate requirement bases, that is collections of access control requirements to be enforced by access control models. These models enabled evaluation of the security of computer systems according to a pre-defined criteria. The first efforts to formulate such criteria resulted in the U.S. DoD Trusted Computer System Evaluation Criteria (TCSEC) [1], also known as the "Orange Book". TCSEC was further followed by several other international and national evaluation criteria.

The core idea in the evaluation of security is to examine a formally specified security policy model and it's implementation against a fixed set of requirements. The higher the target level of security, the more convincing the evidence of correctness of the security policy model and it's implementation is required. Security policy models are usually expressed as mathematical models, and assurance is provided by different means of logically reasoning about the security. Even if the evaluation of security policy models has become a widely accepted technology that has also been adapted to distributed systems, the major draw back from this research and information systems security point of view is the limited scope. Evaluating security policy models never intended on providing security of comprehensive systems, only from the trusted computing base. Therefore, it is not reasonable to expect same paradigm to be applicable for dealing with entire information systems security.

Simultaneously to access control models, cryptographic techniques have developed rapidly, and distributed systems' security depends on them heavily. From the break through of Diffie and Hellman [10] and Merkle [16], and introduction of RSA [17], significant amount of research has been carried out in theory and applications of private and public key cryptosystems [18]. Therefore, any mechanism for expressing information security requirements that does not take into account nature of cryptographic requirements can not be considered adequate. Since cryptographic primitives, especially those employing public key ciphers, introduce a heavy computational overhead in communications, it is important that application of the primitives is well optimized. Various logics exist to reason about cryptographic protocols to assure from optimal protection with minimal overhead, and recently new types of cryptosystems are introduced that combine calculations for encryption and digital signatures resulting in significant reduction in both number of calculations and size of messages being protected [23].

As summarized by [7], tools for the management of information security have evolved from early check list based models to risk analysis and security evaluation. Even if risk analysis has been a flagship of the management of information security for more than 10 years, there are several

unsolved problems that reduce the capability of risk analysis to provide comprehensive support for the management of information security in the specification, administering, monitoring and implementing security measures on information systems. Due to these problems, different baseline criteria have been proposed for being used instead of risk analysis [20, 21]. Such codes of practise include, for example, British Code of Practise [3] and German IT Baseline Protection Manual [4]. These criteria set common standards for the level of security and costly risk analysis should be used only for the security work in systems with exceptional or high security requirements. Baseline criteria provide good solutions for many practical security development situations. From a scientific and high assurance system point of view, anyhow, this is not a desired step. Applying baseline criteria means returning to the check list based protection that is, even with extensive listing, neither scientifically sound method nor capable of providing high level assurance from the security of systems [7, 8]. Therefore, there is a need to develop new methods to evaluate security level of comprehensive systems instead of returning to old solutions.

One such idea is a proposed framework for harmonizing the information security requirements in organizations [14, 15]. Harmonization framework is a semi-formal specification of an information security development organization, information security requirements, assignment of these requirements into various components in the organization, and rules for merging requirements of various sources and transforming them into lower levels of abstraction. The process on which information security requirements are transformed from high level of abstraction into low levels of abstraction in a manner that provides assurance from internal properties of a requirement base, such as consistency, is called harmonization of information security requirements.

The contribution of this paper is to examine methods for providing external assurance of requirement bases. As harmonization only deals with internal properties of a requirement base, there is a need to study issues such as comprehensiveness to provide a comprehensive framework for dealing with security requirements in information systems. We propose a method for specifying criteria to evaluate requirements expressed using a specific notation given here. Further, different evaluation approaches shall be presented and compared in order to support identification of most suitable strategy for different security development situations and environment. Optimally, different strategies and methods are used in concert to divide evaluation into phases each building on previous phase and providing more detail about the status of a requirement base.

The paper begins by stating the modeling objectives and justifying the scope of chosen evaluation approach in section 2. A specification of the assurance model shall be given in section 3. Different evaluation strategies shall be identified and compared in section 4 and the model shall be critically evaluated in section 5. Finally, conclusions shall be drawn and directions highlighted for future work in section 6.

2 MODELING OBJECTIVES AND SCOPE OF RESEARCH

Models for assurance of security usually focus either on international evaluation of products or on internal evaluation of a requirement base. Internationally, products can be evaluated, for example, according to TCSEC [1], ITSEC [2] or Common Criteria [5]. These evaluation criteria are, anyhow, not to be considered within this paper. They are well established by international bodies and support universal determination of security levels of products by judging them by an international evaluation body. Instead, focus of this paper is on the provision of control on top of models for internal evaluation of a requirement base in the specific harmonization framework. Conceptually, the control can be seen as a set of constraints the requirement base must satisfy to meet the organizational assurance requirements.

Models for reasoning about requirement bases are usually considerably specific to the underlying security model. This is obvious, since different semantical conventions in methods to express requirements are the core of most advanced models for dealing with authorizations. Therefore, it is unlikely that the proposed framework would be directly applicable on other logics. Security logics are a widely studied area of research since foundation of formal access control models, and several results have been published. Proposed approaches include, for example, formal languages with rich semantics [6, 12, 22], deontic logic [11], theory of normative positions [13] and harmonization functions [14].

The strength of harmonization framework is, that the exact semantic of a requirement base is loosened and the core is on the specification of harmonization functions [15] that transform abstract requirements with loose semantics into more concrete implementation specifications with stricter semantics. This mechanism reduces the dependency between notation for expressing requirements and the exact semantic model underlying. Therefore, it makes the framework more flexible for dealing with a wider range of requirements, namely any specific security requirement in a distributed systems. Pervasive requirements are dealt in a way that assurance is provided from internal properties of a requirement base, and - as extended in this paper - also from compliance with organizational constraints, that is assurance from external properties of requirement bases.

The main objective of this research is to provide support for dealing with comprehensiveness of a requirement base. Comprehensiveness here

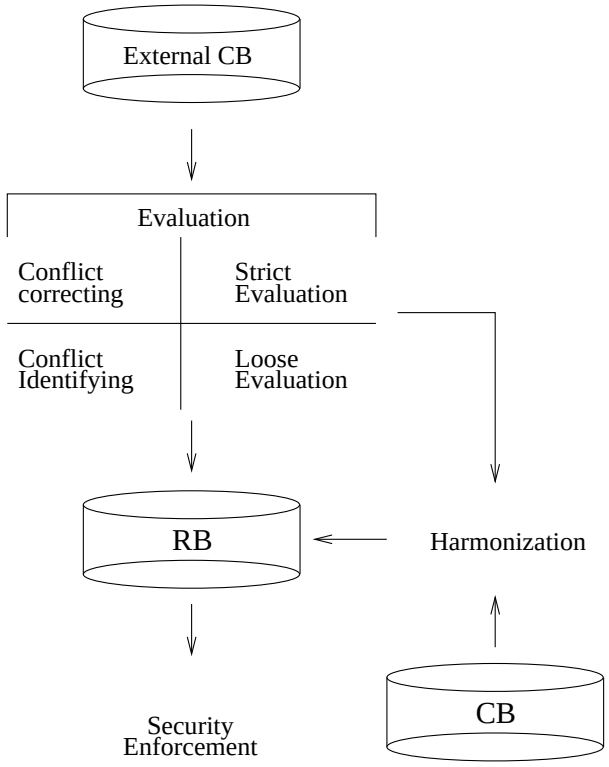

Figure 1: Modeling approach

refers to the property of a requirement base containing descriptions of all relevant security requirements. It is obvious that such a state of a requirement base can not be reached by internal evaluation. Therefore, a mechanism is provided to express constraints for a requirement base and to test the base against these constraints is required. Our modeling approach is illustrated in figure 1. We assume that an internal requirement base (RB) consists of formal specification of security requirements and a set of internal criteria (CB, criteria base) these requirements must satisfy in order to assure from consistency of a requirement base. Both requirements and criteria are formulated using any of the aforementioned notations.

Assurance from internal properties of a requirement base is therefore expanded also to cover external properties. An external constraint base (External CB) is assumed, and two strategies towards evaluation are proposed: loose and strict evaluation that can be enforced either by requirement identifying or requirement correcting evaluation. Evaluation refers to the process of determining whether a requirement base complies to the constraints. It is logical to first evaluate a requirement base using a loose evaluation strategy to assure that it satisfies the minimum level of security. Once assurance of minimal level of security is provided, strict evaluation can be performed to further improve consistency and cost-effectiveness of a requirement base. Additionally, conflict correcting and conflict identifying evaluation methods shall be identified to be used at different phases of security development.

3 ASSURANCE MODEL

A notation shall be given for requirements and individual criteria that the comparison criteria is formed of, and different alternative evaluation methods shall be compared. The following terminology shall be used throughout the paper:

Information security requirement (shortly, requirement) is any formally expressed complete or incomplete statement of the required protection of a component in a system.

Information security requirement base (shortly, requirement base) is a collection of requirements assigned to a specific component in the organization.

Information security evaluation criteria (shortly, criteria) is any formally expressed statement indicating a criteria that the organizational information security requirement base is evaluated against.

Information security evaluation (shortly, evaluation) is a process of comparing a set of information security requirements into the set of information security evaluation criteria and, depending on the evaluation method, reporting or correction mismatches between requirements and criteria.

The model for assurance consists of specification of security requirements within the organization. Independently of the requirements, an evaluation criteria shall be formulated to consist of statements regarding the required type and level of protection within the system. Then, according to a predefined method and strategy, security requirements are evaluated and exceptions are identified and further processed. Methods for exception processing are conflict identification where violations of security are only reported, and conflict correction where potential violations are also corrected as they occur. Strategies are loose and strict evaluation. In loose evaluation, violations are identified only if they reduce the level of security, but in strict evaluation all exceptions are reported. Typically, evaluation of security within organizations consists of both. At early phases, loose evaluation is used to assure from the baseline security within the organization. Strict evaluation is used later to optimize the protection. Material on sections 3.1 and 3.2 are modified from specifications in [15].

3.1 Formulation of information security requirements

Communications security can be seen as provision of secure associations [19]. An association is

an abstraction of communication channel used for transmitting data between processes. Let P and Q be processes. A process is usually a program on execution in a specific host but at higher levels of abstraction can represent a system, business unit or even an organizational division. If P and Q communicate, we say that exists an association A such that $(P, Q) \in A$. Association refers to a unidirectional data flow from P to Q. Each process has an attribute vector $P.\alpha$ where each attribute can be either a possession attribute or criteria attribute. Possession attributes are abstract possessions of a process, such as encryption keys, and are used as parameters for securing the association[1]. Criteria attributes, such as length of an encryption key, set constraints that possession attributes must satisfy. Similarly, an association A has an attribute vector $A.\alpha$.

To specify protection for an association, an additional protection vector $A.\gamma$ is specified for each association. Protection vector is of form (p, a, δ) where p is a communication protocol, a is an algorithm, and δ is a parameter vector for a. An intuitive interpretation of the protection vector is that transmission over A, using protocol p must be protected by algorithm a with a parameter vector δ. This satisfies the specification of both content and protection of a data flow. Security enforcement mechanism should then consult the requirement base when transmitting data and protect the flow accordingly. Information security requirement is formally specified in definition 1. Similarly, if not specification is given for a specific communication protocol, transmission using that protocol should be denied.

Definition 1. *Information security requirement base is a collection of information security requirements formulated as* (P, Q, A, p, a, δ) *where* P *and* Q *are processes and* A *is an association* $(P, Q) \in A$ *such that* $A.\gamma = (p, q, \delta)$.

Even when dealing with higher levels of abstractions and incomplete requirements, the final status of a requirement base is a collection of well-defined protection vectors of different associations within the system. This collection of protection vectors can then be evaluated by the criteria formulated according to rules given in the following.

3.2 Formulation of evaluation criteria

Evaluation criteria base consists of a set of individual criteria, as formally specified in definition 2. A criteria is a statement expressed as $[a : b]$, intuitively interpreted as "for each requirement in a requirement base where a is true, also b should be true". To allow efficient and feasible automation of comparison, both a and b should be reduced first order logic statements about process attributes and association protection specifications. Of form $c = [a : b]$, each a shall be called precondition of a criteria c and b shall be called postcondition of criteria c.

The following notation is used to formulate criteria:

1. Let P be a process and A be an association. Any process attribute $P.\alpha$ or association attribute $A.\alpha$, requirement attribute $A.\gamma.p$, $A.\gamma.a$ or $A.\gamma.\delta$ is a literal.

2. Using the given alphabet, any well formed construct using literals and logical symbols $\neg$, $=$, $\in$, $\wedge$ and $\vee$ is a formula. The logical symbols are assumed standard semantics.

3. If f and g are formulas, then $[f : g]$ is a criteria.

Definition 2. *An information security evaluation criteria base is a collection of individual criteria where each criteria is of form* $[f : g]$ *where each* f *and* g *are specified as above.*

Criteria base can be either an industry wide specification, or an internal self audit resource of an organization. In the case of evaluating a system, the process then consists of two steps. First, is the provision of assurance that the used criteria is a complete and correct representation of security requirements and, second, provision of assurance that the system security requirement base meets the criteria. The latter can be automated, due to a formal presentation of requirements and criteria, but the provision of assurance of the correctness of a requirement base is left on the experience and expertise of the security personnel within an organizations. As the evaluation can be automated, it is possible for clients comparing the security of different components to specify their own, context-specific criteria and use this in order to evaluate different alternative systems. Alternatively, industry-wide baseline can be applied and trust is based on the quality of that baseline criteria. It is, anyhow, not within the scope of this paper to discuss the trust in evaluation criteria further. The logic behind this is that it is assumed significantly easier to provide guidelines on what must be included in a requirement base than to test whether the (most likely very complicated) requirement base actually meets those criteria. Therefore, the organizational knowledge is assumed to play a key role in the specification of a criteria base.

3.3 Evaluation methods

Let $\mathcal{R} = \{r\}$ be a requirement base and $\mathcal{C} = \{c\}$ a criteria base. Simply, evaluation of security of the set of requirements R with respect to the criteria base C is calculation of function $\Theta : \mathcal{R} \times \mathcal{C} \rightarrow \{TRUE, FALSE\}$

[1] In the Harmonizer software, the notation has been simplified slightly to improve user friendliness. Fully documented software is publicly available in http://mars.fcit.monash.edu.au/~skylark/harm/ for evaluation.

$$\Theta(\mathcal{R},\mathcal{C}) = \begin{cases} TRUE & if \quad \forall c \in \mathcal{C} \exists r \in \mathcal{R}: \\ & (c.f(r) = TRUE) \wedge \\ & (c.g(r) = TRUE) \\ FALSE & otherwise \end{cases} \quad (1)$$

where the notation $c.f(r)$ refers to the evaluation of whether requirement r meets the precondition of a criteria c. Equation 1 specifies a simple method to report exceptions and mismatches between $\mathcal{R}$ and $\mathcal{C}$. It is, anyhow, that this may not always be enough to evaluate the security of systems. This type of evaluation, where conflicts are only identified and reported is called "Conflict identifying evaluation". Additionally, as the criteria notation specifies the post condition that a requirement base should satisfy, it is possible to automatically correct the conflicts between R and C. This is called "Requirement correcting evaluation". Both these approaches shall be studied and compared within the rest of this section.

Conflict identifying evaluation is where weaknesses are identified and reported but not corrected. A system is to be evaluated to meet a specific level of security, and in case there are weaknesses, the evaluation fails and the system is not certified to meet the target level of security. This approach has a clear advantage in it's simplicity and capacity of comparing the level of security of different products or systems to a minimum standard. This is often case when there are several potential systems to be purchased and the organization sets strict minimum level of security required from systems. The conflict identifying evaluation clearly states whether the system meets the required level of security.

There is, anyhow, a possibility of improvement. Assuming a self-audit situation, it is as essential to know that there is a weakness in the level of security in a system as to quickly identify mechanisms to correct the weakness. As conflict identifying evaluation only reports a failure of the requirement base to meet the evaluation criteria, it does not provide with adequate information for refining a requirement base to meet the criteria. Due to a formal presentation of [14] and [15], it is possible to automatically correct weaknesses and therefore, the evaluation can be integrated into the refinement and harmonization of information security requirements using conflict correcting evaluation.

Conflict correcting evaluation is where an attempt is made to automatically correct any weakness identified in the requirement base during the evaluation. Conflict correcting evaluation is enabled by the conditional nature of criteria, where the post condition (g) specifies the desired content of a requirement base in cases where pre condition (f) is satisfied. Simply, an automated transfer of a vulnerable requirement base into a free of conflict requirement base can be triggered when a vulnerability is identified.

Let $r \in \mathcal{R}$ be a requirement and $c \in \mathcal{C}$ be a criteria such that $c.f(r) = TRUE$ but $c.g(r) = FALSE$. This is, requirement r is in conflict with criteria c. Transformation $\mathcal{R} \rightarrow \mathcal{R}'$ is seen as a modification of requirement $r \in \mathcal{R}$ into requirement $r' \in \mathcal{R}'$ such that $c.f(r') = TRUE$ and $c.g(r') = TRUE$. The simple way to convert $\mathcal{R}'$ from $\mathcal{R}$ is to enforce equation 2. In the Harmonizer software, enforcement should be enforced by constructing a requirement generating harmonization function that is enforced to the relevant requirement subset. Merging of requirements has been successfully implemented using same strategy in the current implementation of Harmonizer. Basically, each requirement in the requirement base is assessed and, if there is a match with a pre condition but unmatch with a post condition, the requirement is replaced with the post condition. This is possible, since - due to the simple notation of requirements - each requirement is atomic, they can not be divided into smaller components. It is also assumed that requirement dependencies are considered when specifying the criteria. Each criteria should also be independent in a way that one criteria evaluates only one aspect of the enforcement of security. Therefore, requirement dependencies must also be enforced by separate criteria.

$$\forall r \in \mathcal{R} c \in \mathcal{C} : (c.f(r) = TRUE \wedge c.g.(r) = FALSE) \Rightarrow \mathcal{R} = (\mathcal{R} - \{r\}) \cup \{c.g\} \quad (2)$$

An additional question is the alignment of evaluation and security enforcement mechanisms within the organization. Evaluation can be used not only to identify and correct weaknesses but also excessive protection specifications that might lead to either inconsistency or increased cost of protection. The next section studies different evaluation strategies aiming on answering the question on whether only weaknesses or all conflicts should be identified and corrected.

4 EVALUATION STRATEGIES

Two basic strategies towards evaluation are identified. These cases are applicable to both conflict identifying and conflict correcting evaluation. Loose evaluation refers to the evaluation where only weaknesses are identified and processed (reported or corrected) and strict evaluation to the case where all exceptions of the criteria, whether weakening or strengthening protection, are further processed. Selection of the strategy depends on the objective of evaluation. The primary purpose of evaluation may be on the provision of assurance that the system satisfies the minimum level of information security. In this case, it is adequate to only identify potential weaknesses. Similarly, if the organization has a minimum level of security that each individual component in the information system is required to satisfy, it is enough to

evaluate potential purchases by loose evaluation strategy.

Security evaluation may, anyhow, have additional objectives than identification of weaknesses in the security of systems. Provision of assurance from the security of information systems is an early phase activity in the major improvement or update of organizational information security. Once this stage is completed, and systems satisfy the required security level, more fine grained evaluation, using strict evaluation strategy, can be carried out in order to identify potential inefficiencies in the protection. As the fundamental objective of information security is to provide strong protection with minimum cost, there may be a need to evaluate systems also in order to reduce the level of protection to align application of security enforcement mechanisms within the organization, thus improving cost-efficiency of security maintenance, or to reduce unnecessary processing overhead caused by enforcement of too strict security requirements.

Let $r \in \mathcal{R}$ be a requirement and $CL = \{\delta_1, \delta_2, \ldots, \delta_n\}$ is a set of security classes within the organization where $\delta_1 < \delta_2 < \ldots < \delta_n$. This is similar to the **dominates** relationship in the original BLP model. To enforce the loose evaluation, each requirement must be attached with an interpretation function $\mathcal{I}(r) \rightarrow CL$ that maps a requirement to a specific security level. Now evaluation refers to the specification of set Γ, as specified in equation 3, that contains all requirements r that fail to meet the evaluation criteria $c \in \mathcal{C}$.

$$\Gamma(\mathcal{R}, \mathcal{C}) = \{(r, c), where(c.f.(r) = TRUE) \wedge (\mathcal{I}(r) > \mathcal{I}(c.g))\} \quad (3)$$

For example, let $r = (Q, P, A, SMTP, RSA, (sk, pk, 768))$ where Q and P are processes and A is an association such that $(P, Q) \in A$ be a simple requirement to specify that when communicating by E-mail over Internet (SMTP protocol), RSA encryption must be used using secret key sk and public key pk of the length of 768 bits. Assume also that $CL = \{TS, S, C, U\}$ describing top secret, secret, classified and unclassified security classes within the organization. A simple interpretation function can be specified for r as in equation 4. The example is simplified in the sense that not all values are possible for RSA keys but intuitively highlights the nature of interpretation functions. It can also be debated whether top secret information should be transmitted over untrusted networks and whether they should be protected only using stronger techniques than public key cryptography.

$$\mathcal{I}(r) = \begin{cases} TS & if \quad kl \geq 2048 \\ S & if \quad 1578 \leq kl \geq 2048 \\ C & if \quad 1024 \leq kl \geq 1577 \\ U & if \quad kl \leq 1023 \end{cases} \quad (4)$$

Now, by loose evaluation, a criteria can be set as $c = [A.\alpha = SMTP : A.\gamma.a = RSA \wedge A.\gamma.kl = 1024]$. Criteria c sets the minimum key length for RSA-based e-mail protection to be 1024 bits (indicating information transmitted by e-mail is confidential by default). It is easy to see that requirement r when evaluated by criteria c leads to a conflict since $\mathcal{I}(r) = U$ and $\mathcal{I}(c.g) = C$ and $S < C$. Let c' be a modified requirement $c' = [A.\alpha = SMTP : A.\gamma.a = RSA \wedge A.\gamma.kl = 1024]$ there should not be conflict identified. Using the syntax of Harmonizer software, this can also be expressed as a harmonization function:

Protocol = SMTP AND Algorithm = RSA
AND kl < 1024 : A kl = 1024

For strict evaluation, a more fine grained interpretation function must be specified. Instead of dealing with security classes, interpretation must deal with absolute values regarding protection. This makes is more difficult to compare the level of security provided by different types of requirements but enables strict evaluation. In this case, the interpretation function for a requirement r must be specified as $\mathcal{I} : \mathcal{R} \rightarrow \mathcal{N}$. For example, the previous example can be simplified into an interpretation function $\mathcal{I}(r) = kl$ where the key length simply determines the level of protection. There are better ways to make different algorithms to be more comparable, such as work factors, but for the purposes of this example, the above interpretation is adequate. Now, evaluation refers to the specification of set Δ as specified in equation 5.

$$\Delta(\mathcal{R}, \mathcal{C}) = \{(r, c)\}, where(c.f(r) = TRUE) \wedge (\mathcal{I}(r) \neq \mathcal{I}(c.g)) \quad (5)$$

5 EVALUATION OF THE MODEL

There are several success factors for the proposed model. First, the most essential question is whether the model is capable of catching the nature of system evaluation without too strong restrictions in the formulation of requirements and evaluation criteria. The major restricting factor is the notation specified for formulating information security requirements. The flexibility of the notation improves chances of it being capable of catching the requirements. Also, tests carried out with the Harmonizer software support the assumption. This notation rules out pervasive and non-technical, such as educational requirements. There are two reasons for this not been seen as a too restricting factor. First, pervasive requirements are usually concerned with generic requirements concerning trusted implementation and verification of implementation of security sub system. Therefore, the evaluation model actually acts as a tool to set pervasive requirements, concerning non-specific security aspects of the system. Second, non-technical security requirements are dependent on the technical implementation of security subsystem. Generic security awareness can be raised

but without being bind into the existing security enforcement technology, it remains on a high level of abstraction and may fail to provide concrete guide lines on acting in a manner that enforces security. Therefore, we see that technical security must be properly enforced before non-technical requirements can be formulated and enforced.

Second, there is a question on whether there is a need for the proposed model. As stated in section 2, the purpose of the model is to enable light weight self audit of information systems in order to support various duties in the management of information security. It is not intended to replace formal security evaluation by international criteria but to be a commonly used tool used internally within organizations. Also, an intention is to provide support for different evaluation strategies, each carried out in different phases of the security work. There are several approaches to the evaluation of information security, such as ISO 9000, Capability maturity model (CMM) and Baseline approaches. The proposed model lies below these approaches and attempts on providing a concrete tool for setting concrete objectives as parts of the overall quality of the security of information systems. As there is a significant gap between approaches to the quality of information security and formal evaluation, proposed model intends on reducing this gap by providing a tool to be used as a part of the information security quality assessment.

Third, there is a question on whether the proposed model is applicable in the real world. The lack of empirical results on the application of the model on the evaluation of concrete systems is the major weakness of the proposed model. Anyhow, the underlying theory has been evaluated and it is up to versioning the Harmonizer software to provide assurance of external properties of a requirement base. This paper intentionally focuses on theoretical foundation behind assurance attempting to promote discussion regarding the issue. Due to the lack of light weights models, security evaluation has not typically been an essential component in the management of information security. Therefore, there is a need to comprehensively analyze the role of evaluation, if made possible by the proposed model, in the development of security of information systems. This discussion is then assumed to provide authors with feedback regarding the direction that should be taken when implementing a prototype and applying the model in case studies.

6 CONCLUSIONS AND FUTURE WORK

A mechanism has been proposed on providing assurance of external characteristics of a requirement base in the harmonization framework, especially that of comprehensiveness. The aim of the model is to aid designers of information systems security to assure from the fact that all relevant requirements are considered in the requirement base formulation, and to aid in internal security self-audits. Most existing security logics focus on internal assurance and optimization of a requirement base, and therefore lack provision of assurance of external properties. As most models for evaluating information security of products are not capable of providing tools for organizational self-audit of security, it is essential that models such as the proposed one are developed.

The modeling approach has been to provide with a means to formulate security requirements and requirement evaluation criteria and provide with a formal foundation on determining whether a requirement base meets the criteria. Additionally, several evaluation methods and strategies have been proposed in order to make the model more attractive by providing support for several phases of evaluation. It is logical to think that at first phases of evaluation, the focus is on provision of guarantee that a system satisfies the minimum level of security. Once this has been assured by a loose evaluation, and potential weaknesses have been identified and corrected, it is necessary to optimize the protection by reducing the security into optimal level in order to reduce cost without violating total security. This is when strict evaluation is applied.

The model itself is intended as a tool for organizational self audit and comparison of different alternative products that have not been internationally evaluated according to international criteria. There is still a huge gap between proposed model and formal security evaluation criteria. It has not been the intention of this proposal to fully bridge this gap but to provide with a new method to expand the control over an information security requirement base that is a step towards solution bringing together internal requirement base evaluation logics and international evaluation criteria and processes. This is where the major area of future work is seen. Practical case studies are an other area of important work in order to evaluate the model in practical security development situations, but theoretical work is needed to study interoperation of the proposed model and international security evaluation models.

References

[1] Trusted computer systems evaluation criteria. U.S. Department of Defence, 1983.

[2] Information technology security evaluation criteria (ITSEC). provisional harmonized criteria, version 1.2. Commossion of the European Communities COM(92) 298 final, Brussels, Belgium, Sept. 1992.

[3] Code of practise for information security management. British Standards Institute Standard BS 7799, UK, 1995.

[4] IT baseline protection manual. BSI, Germany, 1996.

[5] International standard ISO/IEC 15408 common criteria for information technology security evaluation (parts 1-3), version 2.0, CCIB-98-026, May 1998.

[6] M. Abadi, M. Burrows, L. B., and P. G. A calculus for access control in distributed systems. In *Advances in Cryptology - Crypto'91*, LNCS 576. Springer-Verlag, 1991.

[7] J. Backhouse and G. Dhillon. Structures of responsibility and security of information systems. *European Journal of Information Systems*, 5(1):2–9, 1996.

[8] R. Baskerville. Information systems security design methods: Implications for information systems development. *ACM Computing Surveys*, 25(4):375–414, 1993.

[9] D. Bell and L. J. LaPadula. Secure computer systems: Mathematical foundations and model. Technical Report M74-244, MITRE Corporation, Bedford, Massachusetts, USA, 1975.

[10] W. Diffie and M. E. Hellman. New Directions in Cryptography. *IEEE Transactions on Information Theory*, IT–22(6):644–654, Nov 1976.

[11] J. Glasgow, G. MacEwen, and P. Panangaden. A logic for reasoning about security. *ACM Transactions on Computer Systems*, 10(3):226–264, 1992.

[12] S. Jajodia, P. Samarati, and V. S. Subrahmanian. A logical language for expressing authoprizations. In *Proceedings of the IEEE Symposium on Security and Privacy*, 1997.

[13] A. J. I. Jones and M. Sergot. Formal specification of security requirements using the theory of normative positions. In *Computer Security - ESORICS'92*, LNCS 648. Springer-Verlag, 1992.

[14] J. Leiwo and Y. Zheng. A formal model to aid in documenting and harmonization of information security requirements. In *Proceedings of the IFIP TC11 13th International Conference on Information Systems Security*, 1997.

[15] J. Leiwo and Y. Zheng. A framework for the management of information security requirements. In *Information Security, Proceedings of the First International Workshop*, number 1396 in Lecture Notes in Computer Science, pages 232 – 245. Springer–Verlag, 1997.

[16] R. C. Merkle and M. E. Hellman. Hiding information and signatures in trap door knapsacks. *IEEE Transactions on Information Theory*, 24(5):525–530, September 1978.

[17] R. L. Rivest, A. Shamir, and L. M. Adleman. A Method for Obtaining Digital Signatures and Public–Key Cryptosystems. *Communications of the ACM*, 21(2):120–126, Feb 1978.

[18] B. Schneier. *Applied Cryptography*. John Wiley & Sons, New York, second edition, 1996.

[19] W. Stallings. *Network and Internetwork Security: Principles and Practise*. Prentice-Hall, Englewood Cliffs, N.J., 1995.

[20] R. von Solms. Information security management: The next generation. *Computers & Security*, 15(4):281–288, 1996.

[21] R. von Solms. Can security baseline replace risk analysis. In *Proceedings of the IFIP TC11 13th International Conference on Information Systems Security*, 1997.

[22] T. Y. Woo and S. S. Lam. Authorization in distributed systems: A formal approach. In *Proceedings of 1992 IEEE Symposium on Research in Security and Privacy*, 1992.

[23] Y. Zheng. Digital signcryption or how to achieve cost(signature & encryption) $\leq$ cost (signature) + cost(encryption). In *Advances in Cryptology – Crypto'97*, number 1294 in Lecture Notes in Computer Science. Springer-Verlag, 1997.