MANTECH Publications

Recent Trends in Computer Science and Software Technology

Volume 7, Issue 2, May-August 2022

Paradigm for Creating a Safer Internet Based on a Blend of Different Perspectives

Dr. V.K Lakshman[1], Arvind Yadav[2]

Assistant Professor[1], Student[2]

Department of Computer Science Engineering

Sharad Institute of Technology Polytechnic, Yadrav, Maharashtra, India

Corresponding Author's E-mail: *yadavaravindcse@gmail.com[2]*

Abstract

The needs are described from four separate points of view in this article: those of the owner, the administrator, the user, and the network. This study offers a paradigmatic approach to developing the design of secure systems. The process of eliciting requirements and creating an architecture involves jointly defining the problem and its structure. The viewpoints put out in this work are those that we think are necessary for their contributions as significant players in the development, application, use, and upkeep of secure systems. The remarkable expansion of Internet technology and the applications used on the World Wide Web has made it such that security is now a top priority in the creation of safe systems. Organizations today employ a variety of security methods. The security issue persists despite the broad adoption of several alternative security solutions. The construction of secure architecture using the methodology outlined in this work is claimed to be entirely realistic. The requirements model (owner and administrator viewpoints) and the operations model are the models that reflect these various points of view (views of user and network). This article explains the multiple view paradigm by first outlining the distinctive specifications and/or traits of secure systems (especially in the context of networks) and the secure architecture / system development process.

Keywords: *Multiple View Paradigm, Owner, Secure System, Administrator, User, Network, Model*

 Page 70-77 © MANTECH PUBLICATIONS 2022. All Rights Reserved

INTRODUCTION

Restricting physical resource access was used to offer security for early computing systems. Every programme, though, is now distributed in a networked environment since it is shared by many users in the Internet era. Additionally, the majority of these programmes' users are "outsiders" (customers) of the company hosting them online. The limitation on access to physical resources must be lifted in order to share the information or application with individuals outside the community. However, a number of security primitives, including encryption and digital signatures, are used to allow the security for the application and/or information.

Despite the difficulty in designing and implementing these security measures, there have been instances where they have been compromised. "Why shouldn't we adopt a systematic strategy to give security to the application and or information that is based on the requirements of the security policies, plans, and business processes of the organisation?" is a question that quickly enters the mind. This idea inspires the method outlined in this article, which involves first determining the needs from four distinct perspectives in order to offer security, and then developing a solution. This is what we refer to as a "many perspective paradigm" on the application's security requirements. Everyone in the modern world is aware that impossible security is frequently breached in ways that its creators never anticipated [1]. This makes it very evident that we require a method for simulating threats to computer and information systems.

It is feasible to build and create countermeasures to those assaults if we are able to comprehend all the numerous ways that a system might be attacked. This study examines the Multiple View Paradigm, a technique for creating safe software and information systems.

We employ a modified 'Concur Task Tree' [2] notation based requirements driven task modelling technique in order to construct a multiple view paradigm. When a system interacts with people, task models explain how actions might be carried out to achieve the system's objectives. When creating an interactive application, such as an e-commerce application or any other web-based application, they should take into account the needs anticipated by everyone who should be taken into account.

Page 70-77 © MANTECH PUBLICATIONS 2022. All Rights Reserved

MULTIPLE VIEW PARADIGM

It seems less hopeful to design safe systems from a single vantage point since hackers will always find a way into those systems. This is especially true when considering the bigger, more dispersed systems and applications that are used today on the Internet. This research employs a multiple view approach to secure system design in this situation. These systems are designed to operate in environments where actual, business-critical applications are running in enterprises.

Due to the involvement of numerous parties in the system development and deployment process, including the owner, the designer, the user, and the network with which the user accesses the application or system, the views of the system's security, specifically the requirements model and the operations model, have been occurring naturally. These models show how the owner, designer, user, and network may all comprehend the security issue with the system. This natural division of the security requirements into these two parties' and views' perspectives has technical merit as well; in order to enable the concept known as "task tree" in this research, it is necessary to separate security aspects of the system into individually comprehensible packages or modules.

In order to implement the security perspectives of the requirements model, this "task tree" is a framework that offers the operations (operations model).

In our case, the requirements for the secure system in an organisation have been identified from four different view points, and as a result, its architecture reflects these needs. A secure system and the design of its architecture aim to create a solution for the problem undertaken with the requirements identified for it.

THE REQUIREMENTS MODEL

Even though researchers interested in software security have recognized the significance of the awareness factor at the implementation level (in terms of algorithms, approaches, methodologies and solutions), it is essential to note that they have failed to see its other dimensions including the one at organizational level [3]. However, recent attempts have prompted the proper identification of various kinds of requirements of security to information and or software.

Page 70-77 © MANTECH PUBLICATIONS 2022. All Rights Reserved

Security properties and behaviors of a software system are categorized into 11 classes in ISO/IEC-15408 Common Criteria.[4]. These classes are made of members, called families, based on a set of security requirements. In addition to these requirements, this research reveals the identification of the following too, to be identified as requirements for the development of complete security solutions to the software.

a) Constituent organizational units – The organisational environment may be made up of several constituent parts that interact and work together and use their own sets of rules and regulations. To assess the security danger posed by these constituent units and create security solutions, it is critical to identify all such constituent units. In other words, a company that hosts software or applications on the web and seeks to secure access to that information must also identify and enrol the company's other component units, commonly known as sister-concerns, who would also utilise or share the information.

This is a necessary requirement because a "known" insider (colleague) who works for this organisational unit but does not have access rights to this information may try to gain access to the secured information through a third "unknown" insider (colleague) who may work in another constituent unit and has access rights to the information, which must be monitored and prevented.

b) Constituent Systems – An organisational environment's infrastructure may contain a variety of operating systems, networking technologies, databases and management systems, as well as other heterogeneous system components, services, and applications. To fully analyse the security requirements, all of these systems must also be recognised. The system that hosts the actual application, which needs to be protected from assaults, may be hacked through any one or more of these constituent systems.

The majority of software and information systems that need to be safeguarded against hacking are already accessible or have their services offered in environments with many different organisational constituents and systems. Therefore, it is rather logical to consider providing security or evaluating the need for security from these other organisational constituents. The development of software system security must also concentrate on such a complex environment.

Page 70-77 © MANTECH PUBLICATIONS 2022. All Rights Reserved

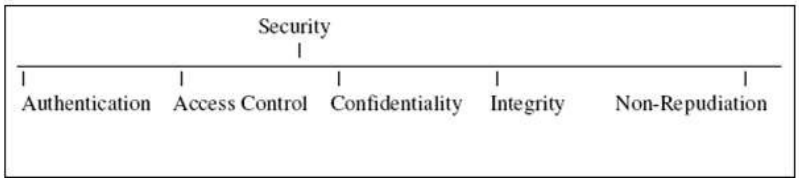


Fig. 1 A typical Requirements Model Task Tree

As an example, Fig. 1 shows a typical requirements model, as a safeguard that can be deployed to prevent the likelihood of threats being realized or to reduce the impact of a realized threat. The (Fig. 1) requirements model caters to the need of a multi-domain or multi-constituent environment where from the software information is available to the outside world for access. That is., this model spans all of the security requirements of different domains, including communications and or network security, computer systems security, personnel security, administrative security and media security.

THE OPERATIONS MODEL

Determining the vulnerabilities and threat exposures embedded in commercial software systems and networks shall be the first step, which is very critical in practice. If we know what we have to fix and how to get the fix, a simple patch, upgrade, or configuration change could be sufficient to eliminate even the most serious vulnerability. Several security policies and models have been proposed by researchers in the past [5,6]. In order to develop a security solution to the aforesaid multi-constituent environment and its requirements model, we prefer this 'operations model'. In other words, it is not possible to trace out the place / position of threat exposure in a multi-constituent environment as it has multi-domain nature in terms of organizational units and systems. Hence we require a security model which would cater to the 'trace' of all possible security exposures in a multi-domain / multi- constituent system. The operations model performs this trace using a framework, referred to as 'task tree' as said in section 2.

The 'task tree' has two levels: the top level correlates to the requirements model, and the lower level incorporates the 'trace' extractions based on the requirements model that is described in the previous section. The top / upper level is to specify the essential security mechanisms or solutions in order to meet the security requirements identified in the requirements model. Each of the security mechanisms

Page 70-77 © MANTECH PUBLICATIONS 2022. All Rights Reserved

listed in the top level is being broken down (in the bottom or lower level) into a set of sub-mechanisms, which are referred to as ‘tasks’. Thus the structure of the security model that is prescribed here looks like a ‘task tree’.

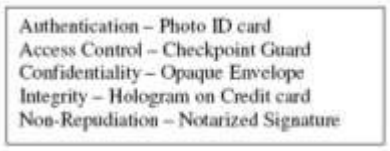


Fig. 2 Security Mechanisms – Top Level of Operations Model

The requirements model in Fig. 1 is represented by top level security methods in Fig. 2. These security measures could be necessary in common application software, such e-commerce software, which serves a large number of customers from a multi-constituent organisation. In this illustration, there is a mapping between the mechanisms that are expected to be employed to address the needs indicated in the requirements model.

The usual bottom-level "sub-mechanisms" needed in the scenario where the "mechanism" is "access control" are shown in Fig. 3. By reconsidering how a mechanism (security solution) is to be viewed, defined, and represented in a multi-constituent / domain environment, as well as how to capture the processes or tasks that need to be in the "task tree," in order to provide a complete secure solution to the mechanism in question, it becomes clear that these sub-mechanisms have been captured in order to provide a complete security solution to each top level mechanism. Relevant considerations included the requirement to identify the danger in the organization's individual units and systems, assurance of the subsystems' ongoing integrity, and the degree of security or protection for the selected mechanism's communications.

The whole security solution, which must be built on the insertion / inclusion of various units and systems of the environment / organisation, is therefore enabled by this task tree-based operations model through their crucial points and interfaces. It's true that the security measures that are already in place are in addition to the standard ones that are envisioned for the hosted software or information system.

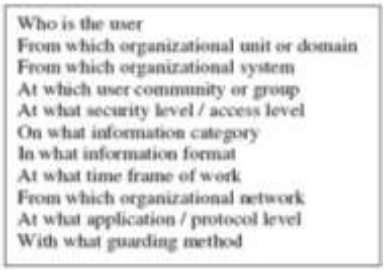


Fig. 3 Sub-mechanisms of Access Control- the Tasks- the bottom level of the Task Tree

Page 70-77 © MANTECH PUBLICATIONS 2022. All Rights Reserved

The security solution thus offered by the multiple view paradigm mentioned above allows the solution's area to be divided into the smallest acceptable secure functional elements or solutions at all feasible levels, with all data / information access through them. Only this security measure could be as tight and practical as the technique that must be created and included into the software systems of the modern world.

AN EXAMPLE

One of the problems with gauging the true business impact of security breaches is that many companies are understandably reluctant to publicly admit that they have suffered significant losses due to the failed security. It is not merely a network problem or an information technology problem. The development and implementation of a sound security policy must start with strategic business assessment from the owners of that business, then its design and development issues will have to be thoroughly studied and implemented by the designer; the requirements at the genuine users and the threats from the bogus users also must be taken care; the problems associated at the network – either at the protocols of networks or applications, and the users, thus must be the totality of the concerns that are to be considered before the security is deployed in that company. These views, which have been discussed in the above sections are being studied and experimented for this research purpose, with many organizations on a trial basis. A typical example of an e- business company is presented in this section, in brief.

Table I gives the analysis (requirements model) carried out for a business house that has its applications in the web. For this e-business to be a highly secured one, there will be different requirements from the four parties of the system. These have been illustrated in Table I. For example, the transaction processing requirements for the owner of the application is making the system ‘fully alert’; that of the designer is to enable all the possible methodologies using suitable algorithms or approaches; that of the user is to exploit the system for its fullest potential; and, for the network side, it is to monitor all the transaction based messages and provide security based information access through network. The operations model has also been developed using task tree representation. This task tree gives out the details of lower level security measures that need to be incorporated in the system.

Page 70-77 © MANTECH PUBLICATIONS 2022. All Rights Reserved

TABLE I

Views and Requirements Analysis for an E-Commerce Application

Views / Require-ments	Owner	Designer	User	Network
Business-related Security issues	List Security holes	List all threats & vulnerabilities	Methods to access	Protocols to support the access
Transaction Processing	Full-Alert Policy	Algorithms & Methodology for complete secure system	Full usage	Monitoring messages & secure them

CONCLUSION

The security solution thus offered by the multiple view paradigm mentioned above allows the solution's area to be divided into the smallest acceptable secure functional elements or solutions at all feasible levels, with all data / information access through them. Only this security measure could be as tight and practical as the technique that must be created and included into the software systems of the modern world.

REFERENCES

1. Khaled M.K., and Jun Han, Composing Security-aware Software, IEEE Software, pp.34-41, Jan-Feb 2002.
2. Theus V., and Ray H., Intrusion Detection Techniques and approaches, Computer Communications, Elsevier, No.25, pp.1356-1365, 2002.
3. Anderson, R, Why Cryptosystems Fail, Communications of the ACM, No.37, pp.32-40, 1994.
4. F. Paterno, Model based Design and Evaluation of Interactive Applications, Springer-Verlag, 1999, ISBN 1-85233-155-0.

Page 70-77 © MANTECH PUBLICATIONS 2022. All Rights Reserved