ENHANCING SYSTEM SECURITY USING PATTERNS

Renuka Venkitesan,

Pondicherry Engineering College,

Abstract

In this paper, we are suggesting a combined consideration of security design patterns and Threat modeling for the enhancement of security in the software systems. Software Security helps in building software without security holes thereby preventing the security issues. The best way to do this is to apply specific patterns in software architecture. Security design patterns are the most widely used solutions to security problems with structural measures in designing some parts of a software system. The use of patterns at different levels all together make a system secure. Security Modeling or Threat Modeling also prevents security problem at an early stage of development. In our paper we also deals with the security analysis of software system based on the Combinations of security patterns during the design phase. We also demonstrated the combined considered approach using the developed application namely “Candidate Enrollment at the Faculty System”.

Index Terms—Security patterns, Software Architecture, Patterns, Threat model.

1. Introduction

The importance of software security is intense, because much of the attacks to software systems are based on poorly designed and developed software. Security is often introduced on an Ad–Hoc basis in design rather than employing it in architecture. Security Modeling is a method that prevent security problem at an early stage of development. A security pattern is a well-known solution to the recurring information security problem that arises in specific perspective. Security patterns are software patterns for the security services or goals. The combination of software patterns and the security services together makes the security patterns. Security patterns provide additional guarantee that software product is correct. Additional one can add new design pattern or security pattern to achieve some security goals. The threat modeling introduced in the SDLC helps to ensure that applications are developed with security. Threat modeling [17] is used to identify threats and vulnerability that affect the system. The identified threats and their ratings on the basis of the architecture and implementation of the application, allows to address countermeasures. The primary factor of threat model is that building secure system is achievable until understanding the possible threats to the application. The use of patterns at different levels all together make a system secure and intended to use the patterns as a guideline while constructing a secure system, also applied the patterns to the existing models at various levels and proven the fact. The main objective of this project is to enhance Security in Software Architecture by using security patterns and providing an security analysis for the software architecture.

2. Related Work

Security patterns categorized for different phases in software development and the ways to work with security patterns effectively are analyzed [1]. Eduardo B. Fernandez enforces the use of patterns at different levels all together make a system secure and intended to use the patterns as a guideline while constructing a secure system, the author also applied the patterns to the existing models at various levels and proven his fact [4]. The evaluation of the security pattern based on analysis of how well they overcome the software hole and possible attacks and did qualitative evaluation by the good combinations of the security pattern selectively to provide secure software system [7]. The usage of security patterns throughout the software lifecycle so as to meet the security requirements needed for the systems with a proposed unique methodology [5]. In the paper [8] they examined the security of the system using Security patterns with the
developed two systems, (i.e., one with security patterns and another one without security patterns,) and finally evaluated that combination of security pattern to each category of attack provides more security. This paper deals with the risk analysis of software system based on the Combinations or missing security patterns during the design phase which reduces the cost while compared to the introduction of security during implementation phase [10]. This paper [12] analyzed many classification schemes for security and also introduces one new security pattern, Further it organizes the patterns in hierarchy and briefly how to apply multiple patterns in sequence. A new taxonomy based on attack patterns to enhance the security design patterns especially for beginners in software security and also suggested the combination of attack patterns, security design patterns used for the validation and evaluation of security design patterns [18].

3. Deploying Security Patterns In Architecture

In enterprises some of the resources in the developed application should be available to certain users or certain people based on their roles. In order to protect those resources, it is essential to make sure that the authorized users only access to the said protected resources. Additionally in Software Development process, Architecture is the main artifact of the Software design.

The need for software security increases day by day, because most attacks to software systems are based on vulnerabilities caused by poorly designed and developed software. Moreover, [23] the implementation of security in software systems at the design phase can reduce the high cost and effort associated with the introduction of security during implementation. For this purpose, security patterns that offer security at the architectural level have been proposed in analogy to the well-known design patterns.

It is strongly believe that the Threat Modeling a design necessary. Without the Threat Model in place, the developer may not know that they have mitigated the most pressing threats.

3.1 Stride Threat Model:

A System called STRIDE is developed by Microsoft for classifying security threats. It prompts for Six categories of security threats. They are a) Spoofing threat is an attempt of gaining access to the system using false identity, b) Tampering threats deals with the modification of data or information maliciously, c) Repudiation is a kind of threat that the user may claim to have not performed that act, d) Information disclosure (privacy breach or Data leak) involves exposing the data or information to some unauthorized person, e) Denial of Service (D.o.S.) is an attack which denies the service to the valid users, f) Elevation of privilege, this type of threat gives the unprivileged user to gain privileged access thereby sufficient to destroy the system.

a. Patterns Classified For Stride:

The STRIDE threat model is used to categorize different types of threats that systems face. Classification process based on threat modeling is more intuitive, because it uses the security problems that the patterns solve.

Lists the classification of the security patterns based on the STRIDE model[3]
b. Patterns Deployed In Proposed System

Based on the classification [3] of Security patterns for the STRIDE, the following Security patterns are needed to provide solution for the system in accordance with the STRIDE Threat model. (a) **Check Point System**, it helps to check the incoming requests, monitors the communication and also take countermeasures in case of violations. (b) **Secure Logger** can be used to log sensitive data and also ensuring tamper-proof storage. c. **Secure Pre-forking** is mostly used for improved performance by running pre-forked processes for a limited time. d. **Replicated System** pattern describes a structure of system which allows distribute workload to the available components in case of failure. These Security patterns must be implemented in the proposed system so as to provide more Security. The addition of Security patterns in the Software component enhances the Software Architecture.

4. Case Study

The Developed system called “Candidates Enrollment at the Faculty System” (CEFS). The Candidates can register in the CEFS system in order to take an entry test. The Administrative workers also register the candidates into the system. The commissioner makes the entries of the test results into database. The commissioner prepare rank list of candidates.

5. Conclusion

Software Security has received much attention during the last years. It aims at preventing security problems by building software without the so-called security holes. One of the ways to do this is to apply specific patterns in software architecture. In the same way that the well-known design patterns for building the well-structured software have been used. Security patterns is a tool that helps developers, architects and security specialists utilize security experts knowledge and standardize how they respond to security threats. The inclusion of threat modeling in the SDLC can help to ensure that applications are
being developed with security built-in from the very beginning.

Fig.2 System Architecture of the CEFS

References


[18] Ian Gorton, “Essential Software Architecture “
