CC-Case Based on System Development Life-Cycle Process

1 Tomoko Kaneko, 2 Shuichiro Yamamoto, 3 Hidehiko Tanaka
1, 3 Institute of Information Security, 2 Nagoya University
1, 3 2-14-1 Tsuruya-cho, Kanagawa-ku, Yokohama, Kanagawa 221-0835, JAPAN
2 464-8601Furo-cho, chikusa-ku, Nagoya, Aichi, JAPAN
1 dgs128101@iisec.ac.jp, 2 yamamotosui@icts.nagoya-u.ac.jp, 3 tanaka@iisec.ac.jp

ABSTRACT

Secure system design faces many risks such as information leakage and denial of service. We propose a method named CC-Case to describe security assurance cases based on the security structures and thereat analysis. CC-Case uses Common Criteria (ISO/IEC15408) and Assurance Case (ISO/IEC15026 part2). While the scope of CC-Case mainly focuses to the requirement stage, CC-Case can handle the life-cycle process of system design, which contains the requirement, design, implementation, test and the maintenance stages. Risks in system development are categorized 3 types: Customer agreement risk, Business continuity risk, and System risk. The life-cycle process of CC-Case strengthens the treatment for system risk and business continuity risk by life-cycle support.

KEYWORDS


1 INTRODUCTION

Customers expect that IT products and systems satisfy the necessary conditions not to fall into any dangerous situations. Developers must build up IT products and systems avoiding many risks. Then risk management is important to support the development. We face many risks at every stage of system development such as requirement analysis, design, development, test and service provision stage. It is really important to assure the countermeasure against risks in the each process of system development.

It is important to handle security risks because security accidents have serious influences. However, there are no established methods to assure the validity of security risk up to this time. In this paper, we propose a security assurance case method against risks. This assurance case makes clear the elements to assure against system risks, and the process to argue with a customer and to get an agreement. It gives the way to make systems trustworthy by effective arguments with stakeholders.

In Chapter 2, we explain assurance case, security assurance case and risk management. In Chapter 3, we show significance of assurance case. Chapter 4, we show the concept of proposed method and its provision to the life-cycle process. In Chapter 5, we explain future tasks.

2 RELATED STUDIES

2.1 Assurance case

Assurance case, which is defined in ISO/IEC15026 part2, is a method for describing a system’s critical security level. Standards are proposed by ISO/IEC15026 [1] and OMG’s Argument Metamodel (ARM) and [2] Software Assurance Evidence Metamodel (SAEM) [3]. ISO/IEC 15026 specifies scopes, adaptability, application, assurance case’s structure and contents, and deliverables. Minimum requirements for assurance case’s structure and contents are: to describe claims of IT products and systems properties, systematic argumentations of the claims, evidence and explicit asumptions of the argumentations; to structurally associate evidence and assumptions with the highest-level claims by introducing supplementary claims in the middle of
a discussion. One common notation is Goal Structuring Notation (GSN) [4], which widely used in Europe for about ten years to verify system risk and validity after identifying risk requirements. Contents of GSN is shown below.

Table 1. Contents of GSN

<table>
<thead>
<tr>
<th>Contents</th>
<th>Figure</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Goal (Claim)</td>
<td></td>
<td>To describe claims of system and product properties</td>
</tr>
<tr>
<td>Strategy (Argumentation)</td>
<td></td>
<td>Systematic argumentations of the claims</td>
</tr>
<tr>
<td>Context (Assumption)</td>
<td></td>
<td>Explicit assumptions of the argumentations</td>
</tr>
<tr>
<td>Undefined</td>
<td></td>
<td>Undefined claims and explanation</td>
</tr>
<tr>
<td>Evidence</td>
<td></td>
<td>Evidence of the argumentations</td>
</tr>
</tbody>
</table>

Minimum requirements for assurance case’s structure and contents are: to describe claims of system and product properties, systematic argumentations of the claims, evidence and explicit assumptions of the argumentations; to structurally associate evidence and assumptions with the highest-level claims by introducing supplementary claims in the middle of a discussion.

2.2 Security assurance case

Goodenough, Lipson and others proposed a method to create Security Assurance case [5]. They described that the Common Criteria provides catalogs of standard Security Functional Requirements and Security assurance Requirements. They decomposed Security assurance case by focusing on the process, such as requirements, design, coding, and operation. The approach did not use the Security Target structure of the CC to describe Security assurance case. Alexander, Hawkins and Kelly overviewed the state of the art on the Security Assurance cases [6]. They showed the practical aspects and benefits to describe Security assurance case in relation to security target documents. However they did not provide any patterns to describe Security assurance case using CC.

Kaneko, Yamamoto and Tanaka recently proposed a security countermeasure decision method using Assurance case and CC [7]. Their method is based on a goal oriented security requirements analysis [8]. Although the method showed a way to describe security assurance case, it did not provide Security assurance case graphical notations and the seamless relationship between security structure and security functional requirements.

2.3 Common Criteria

Common Criteria (CC: equivalent to ISO/IEC15408) [9] specifies a framework for evaluating reliability of the security assurance level defined by a system developer. In Japan, the Japan Information Technology Security Evaluation and Certification Scheme (JISEC) is implemented to evaluate and authenticate IT products (software and hardware) and information systems. In addition, based on CC Recognition Arrangement (CCRA), which recognizes certifications granted by other countries’ evaluation and authorization schemes, CC accredited IT products are recognized and distributed internationally. As an international standard, CC is used to evaluate reliability of security requirements of functions built using IT components (including security functions). CC establishes a precise model of Target of Evaluation (TOE) and the operation environment. And based on the security concept and relationship of assets, threats, and objectives, CC defines ST (Security Target) as a framework for evaluating TOE’s Security Functional Requirement (SFR) and Security Assurance Requirement (SAR). ST is a document that accurately and properly defines security functions implemented in the target system and prescribes targets of security assurance. ST is required for security evaluation and shows levels of adequacy in TOE’s security functions and security assurance.

2.4 Risk Management
Risk Management's goal is to increase the impact and probability of positive risks and decrease them for negative risks. The point is not only avoiding failure, but to bring about opportunities. Time and energy can be spent avoiding, transferring to a third party, and mitigating potential failures. They can be similarly spent on accepting, sharing with third parties and enhancing opportunities. It is task of Risk Management to determine how much time and energy should be on avoiding failures and promoting opportunities.

Risk management includes six main processes in the theory of PMBOK[10]. These are risk management planning, risk identification, qualitative risk analysis, quantitative risk analysis, risk response planning, and risk monitoring and control.

3 ASSURANCE CASE FOR CURRENT SYSTEM RISK

3.1 Significance of Assurance case

Assurance case has been applied to the safety field[11] mainly. The largest benefit which we will enjoy by using the assurance case can perform the agreement argument for the demand between stakeholders enough. It is the point that can record the process of the reason / argument that reached the agreement result / conclusion. By certainly describing the following four points, assurance case offers the framework for building an argument more deeply.
· Claim
· Argumentation
· Evidence
· Explicit Assumption

We show significance of assurance case. Assurance case makes easy to confirm requirements by structured documentation. Requirements are verified that systems or services of target for evaluation are confirmed by assurance case. In addition, assurance case makes the basis clarified by the evidence that can achieve a goal. You can confirm the basis for judgment when a problem occurs. If the basis for judgment was validated by customer, you can use the evidence for legal basis.

The true difficulty of risk management in developing software comes from invisibility of software. Developers must build up software system avoiding many risks. Although software is invisible, developers must show that software system can work correctly as customer needs. It is important that customer’s need records correctly as an agreement. Assurance case record verified evidence with stakeholder. Therefore, assurance case is useful as a consensus building tool for risk management.

3.2 Current System Risks and Treatments by Assurance Case

Risks in system development are categorized 3 types: Customer agreement risk, Business continuity risk, and System risk (Table2). Customer agreement risk has the risk of suits. Business continuity risk has the risk of communication. System risk has explanation risk of validation in developing activity.

However, in conventional development, only system risks are considered, the others are not considered. A brand is suffered a big economic loss when 3 type of risks are mixed. Therefore appropriate treatment for negative risks (avoid, transfer, mitigate) is important. Assurance case for risk management is a method to avoid current risks in system development.
4 SECURITY RISKS AND CC-CASE

4.1 Difficulty of Treatment for Security Risks

It is especially important to handle security risks. If any security accidents happened, the brand would suffered a big economic and honor loss. It is also difficult to treat security risks.

Customer agreement risk of security of table 2 corresponds to suits risk when security accidents occur. To treat with this risk, the evidences to show authentication of contractual customer’s agreements are necessary. Business continuity risk of security mainly is caused by service damages by attackers. To counter with this risk, the evidences to show authentication of risk monitoring and control are necessary. System risk of security is the risk which occurs within development activities of IT products or security function of systems. To counter with this risk, the evidences to show authentication of risk identification, analysis and counter plan are necessary.

It is important to show objective evidences that the customer recognizes that his request such as “The system is acceptably secure” is satisfied.

4.2 Concept of CC-Case

We propose a method to solve difficulty of handling security risks. We developed a method named CC-Case by using Assurance Case and Common Criteria as the evidence. We show the description of countermeasures and procedures which clarify the scope of assurance for the menace, and which obtain an agreement on the assurance level with the customer. CC-Case can provide not only security requirement analysis method but also assurance according to the standard of Common Criteria.

Targets of CC-Case are IT products or systems. CC-Case is also a method to make agreements with customers and developers.

The procedures of CC-Case have dual-layer. Upper layer is named logical model. Under layer is named concrete model. Logical model and concrete model is shown Fig.1. Logical model shows the process structure developed in detail as much as possible independently of specific system. Logical model has life-cycle process and each stage’s process. Concrete model contains real cases corresponding to the specific system. Fig.3 shows the life-cycle process. Concrete model is decomposed logically until it describes evidences at the bottom layer. Concrete model remains evidences as real case and approval results of customers. These evidences recorded in sequence can be used for verification. Risks may change frequently. It is necessary to keep evidences depending on changes. CC-Case supports changes through storing all evidences in a DB. Fig.2 shows an example of concrete model.

![Logical model and concrete model](image-url)
Figure 2. An example of concrete model

Figure 3. The life-cycle process of CC-Case
4.3 Life-Cycle Version of CC-Case

In this paper, we show the support of the life-cycle process of CC-Case. The life-cycle process of CC-Case contains whole processes of requirement, design, implementation, test, maintenance stages. The life-cycle process of CC-Case should handle whole risks of security including business continuity risk of security.

Fig.3 is the life-cycle process of CC-Case. Using this assurance case, we explain the concept of the life-cycle process of CC-Case. In this case, top goal of assurance case is “IT products and systems using CC-Case are secure.” Explicit assumption of the argumentations is “CC”. The strategy shows to verify the process of life-cycle development.

The strategy can be divided into 4 processes of the second goal, “Requirements using CC-Case is secure.”, “Designs using CC-Case is secure.”, “Implementations using CC-Case is secure.”, “Tests and deliveries using CC-Case is secure.”. These goals need evidence which can verify the goals.

The second goal of “Requirements using CC-Case is secure.” can be divided into 2 processes of the third goal, “Security specification using CC-Case is secure.”, and “Definition of development environment using CC-Case is secure.” through the strategy of “Verify security functions of requirements.” The third goal, “Security specification using CC-Case is secure.” is equivalent to the top goal of CC-Case at the requirement stage.

4.4 The Merits of Life-Cycle Version of CC-Case

By the extension from requirement stage to life-cycle process, CC-Case has merits as mentioned below.

It makes easier to treat additionally the change of risks. Security risks changes incessantly because an invisible enemy exists, and an unexpected new menace occurs.

It makes possible to improve the development method. By defining development processes at each stage, CC-Case would be improved to a development method with assurance of life-cycle.

CC-Case at requirement stage has only assurance at the time. However, CC-Case of life-cycle extension has assurance with real products of long span.

The life-cycle process of CC-Case can be expected to establish the discipline and control in the processes of refinement of the IT products and systems during its development and maintenance. It strengthens the treatment for system risk and business continuity risk by life-cycle support. For example, at design stage CC-Case makes easier to accept specification changes by using its logical traceability and evidences. At maintenance stage, CC-Case can be expected to improve reusability, productivity by reusing evidences stored according to defined process.

5 CONCLUSION

In this paper, we showed the assurance case named CC-Case. We explained the general concept of the CC-Case and life-cycle process of the CC-Case.

The life-cycle process of CC-Case makes easier to treat additionally the change of risks. It makes possible to improve the development method. CC-Case of life-cycle extension has assurance with real products of long span. It can be expected to establish the discipline and control in the processes of refinement of the IT products and systems during its development and maintenance. Therefore the life-cycle process of CC-Case can handle current security risks properly.

6 FUTURE TASKS

There are some unsolved issues in the CC-Case presented in this paper.

(1) We need to show detailed processes of stages over life-cycle except requirement.

(2) A detailed selection process for remaining risks is important as a future task, because it implies specific assurance. In addition, describing measures in detail applied when unknown threats actually occur is important as a future task. We
need to define remaining threats and reaction procedures for events caused by the remaining risks.

7 REFERENCES