A Comparison Analysis Study of Tools Support to Analysis Security Requirements for Internet of Things (IoT) Application

A.A. Ibrahim
Postgraduate Student, Faculty of Information and Communication Technology, Universiti Teknikal Malaysia Melaka
asmaasdayana@gmail.com

Massila Kamalrudin
PhD in Electrical and Electronic Engineering, Associate Professor, Faculty of Information and Communication technology, Universiti Teknikal Malaysia Melaka, Melaka, Malaysia.
massila@utem.edu.my

Safiah Sidek
PhD in Education, Associate Professor
Innovative Software System and Services Group, Universiti Teknikal Malaysia, Melaka, Malaysia
safiahsidek@utem.edu.my

ABSTRACT
The growth of IoT has resulted in the need for analysing security requirements to ensure safe and secure Internet of Things (IoT) application experience to the users. However, poor elicited security requirements can lead to a failure in IoT development. Thus, this study evaluates various security requirement and analyses the existing gaps in security requirement tools for IoT application that exist in the market for commercial or research purposes. We report our findings from review and analysis of different studies on security requirement engineering for IoT application. The strengths and weaknesses of the features and utility are also presented to provide further understanding of the gaps and weaknesses of each tool. We conclude that these tools are still immature and need further improvements

Keywords: Internet of things (IoT), security requirements, analyse, elicitation

Introduction
Complexity in security requirements requires Requirement Engineers (REngrs.) to have a security knowledge and experience in the process of eliciting and analysing requirements. Requirements captured by REngrs. are commonly inconsistent with their client’s intended requirements and are often error prone. There is limited tool support providing end-to-end support between the REngrs. and their client for the validation and improvement of these requirements [1].

A security of an application depends greatly on how an application was designed, so it is very important to capture security requirements at the requirements engineering phase. It is usually applied to functional requirements, and can be extended to quality and security requirements, traditionally considered non-functional. By integrating security requirements into Requirements Engineering (RE), a big improvement can be made in term of security vulnerabilities, application maintenance efforts and development costs [2]. This paper reports the review on existing works and tool support to analyse security requirements for IoT applications are presented. IoT applications cover variety of different aspects, such as personal, social, societal, medical, environmental, logistics and many others.

This paper is organized in five sections. After the introduction section, the second section presented the research background of this study. This followed by the third section that summarizes the related works of
this study. The fourth section describes the results and discussion of this research. Lastly, this paper ends with a section on the conclusion and future works.

I. Research Background

Security Requirements Engineering includes the process of specifying, eliciting, and analysing the security requirements for an application or system. Its primary concern will be for prevention of any potential harm in the real world and this is considered as the main constraint of functional requirements. A security requirement serves as a complementary to the functional requirement of a system. Security requirements are usually based on analysis of the assets and services to be ensured and the security threats from which these assets and services should be protected. Also, Therefore, it may be fundamental should consider those security of the requirements right from the starting of the development process [3]. However, it is difficult to guarantee a security requirement as there is a scarcity of security expertise available to the development group. REngrs. tend to have limited knowledge of security issues. They without expertise in security are at risk of overlooking security requirement, which frequently all the prompts security vulnerabilities that camwood after the act make misused to act.

The input Security Requirements is defined as imperatives on the functions of the system, and these imperatives operationalize one or more security objectives. A few requirements engineers could remember to specify, elicit, analyse, and manage such quality requirements as interoperability, operational availability, performance, portability, reliability, and usability. However, many studies fail when it comes to security requirements [3]. The majority of requirements engineers are not well prepared and trained at all in security, and the few that have been trained have only been given an overview of security structural imperatives for example, such as passwords and encryption rather than in actual security requirements. Thus, the most common problem with security requirements, when they are specified at all, is that they tend to be incidentally displaced with security-specific structural imperatives that may unnecessarily constrain the security team from using the most appropriate security components for gathering the true underlying security requirements [4].

The Common Criteria (CC) was developed to facilitate consistent evaluations of security products and systems. It is an global effort to define an IT Security evaluation methodology, which might accept mutual recognition between clients and vendors throughout the global economy [5]. In brief, the CC is a useful guide for the development of products and systems with IT security functions and a guide for procurement of commercial products and systems with security functions. Yet, the issue is that the existing standards, for example the ISO and Common Criteria for security requirements are extensive and developers found it hard to comprehend. There are also instances whereby project stakeholders do not have technical understanding of information system.

Additionally, in cases where security experts are obtainable in the project, formal methods for secure systems engineering tend to diverge from the typical approaches for requirements engineering. This brings concerning challenges for the security experts to contribute effectively with the rest of the requirements engineering effort. While there will be a reasonably useful understanding of security threats and vulnerabilities on the one hand, and on various technical security solutions on the other hand, the understanding of how to formulate good security requirements is at present constrained. In this respect, security problems and issues are often dismissed and this can lead to substantial security problems later. In order to overcome the major problem as discussed above, many methods, approaches, techniques and supporting tools have been developed to ease the requirement engineers and developers in managing the security requirements. However, according to the existing studies, there is a necessity for enhancements on the tools that help security requirements.

II. Related Works

The studied on supporting tools in IoT application that have been used to analyse security requirements will be discussed in this section. Additionally, a review on the related works of the developed tools to facilitate the security requirement task is presented. For this study, we compare six tools based on IoT applications and security requirements aspect that provided by the tools. We also look for tools that has
purpose in Requirements Engineering process. There are GARMDROID [6], AVISPA [7], ElicitO [8], HeRA [9], BSN-Care [10] and Haier SmartCare [11].

The first tool, GARMDROID [6] is a tool that aimed to help IoT software developers and integrators to evaluate IoT security threats supported the visualization of Android application hardware requests. GARMDROID relies on the capabilities provided by the Android SDK tool set, specifically the Android Asset Packaging Tool (AAPT) that is contained as a part of the platform tools set. During this execution client could transfer malware samples and request analysis by a Web interface. This procedure is going to be dependent upon the static analysis of permissions asked by Android applications. Throughout analysis, once an android application file (.apk) has been uploaded by a user, GARMDROID uses a set of bash and python scripts to command AAPT to extract the contents of the app’s AndroidManifest.xml file and to filter out the important strings. To every case GARMDROID displays an inference of the set of hardware features requested by the app under analysis, besides the set of permissions requests. These cases serve a two-fold purpose: to indicate GARMDROID operation and direct the discussion towards observations which might cause establish security threats in IoT-oriented Android applications. However, this tool analyse IoT security threats upon visualization of an Android application hardware request only, which will not cover the aspects of security requirements of IoT applications.

Next, AVISPA [7] is a push-button tool developed for analysing large scale Internet security protocols and applications. The protocols are coded in a language called the HLPSL (High Level Protocol Specification Language). HLPSL includes essential parts that represents different participants and composition of roles for representing scenarios of fundamental parts. Each role is independent from the other role, getting some initial information by parameters, communicating with the other roles by channels. In this study, a lightweight biometric based remote user authentication and key agreement scheme for secure access to IoT services has been proposed. The protocol makes use of lightweight hash operations and XOR (exclusive OR) operation.

The security analysis proves that it is robust against multiple security attacks. The formal verification is performed using AVISPA tool, which confirms its security in the presence of a possible intruder. The proposed multifactor biometric user authentication consists of four phases - User registration phase, Login phase, Authentication phase, and Password change phase. The proposed protocol is lightweight, because it uses only one-way hash, perceptual hash functions and XOR operations which are computationally less expensive, thereby, making the protocol highly suitable for the resource constrained devices in IoT. The security analysis proves that it is robust against multiple security attacks. The formal verification is performed using AVISPA tool, which confirms its security in the vicinity of a possible intruder. However, this work did not cover the elicitation and analysing of IoT application in early phase of development.

Another tool that supports a standardized quality terminology and ontological constructs to capture NFRs (Non-Functional Requirements) throughout the RE process is ElicitO [8]. Knowledge-based reasoning techniques are also supported by ElicitO enabling the semi-automation of RE activities; such as, conflict identification during prioritization and authenticity checking during requirements validation. The tool has two main layers. First layer is the ontology layer where the quality and domain ontologies are encoded as OWL (Web Ontology Language) constructs in the Protege database. The second layer is the user communication layer, which will be decayed under application and GUI (Graphical User Interface) components.

The application component communicates with the ontology layer when querying for domain knowledge and the related quality attributes by the Protege API. All query results and information displayed to the user is done via the graphical user interface. ElicitO additionally gives constructs that enable the explicit description of relationships between functional and NFRs and the integration of quality concerns across the early software engineering stages. Yet, this tool is used to capture NFRs throughout RE activities. It is not a tool to elicit the security requirements. Moreover, ElicitO not able to use across different projects or application by variety of stakeholders and does not support security mechanism for IoT applications.
Meanwhile, The Heuristic Requirements Assistant (HeRA) [9] tool applies security-relevant heuristics to requirements and service descriptions in order to identify possible security issues or problems. HeRA raises awareness and provides feedback while people write requirements. The HeRA tool supports technical experts, as well as security experts, in identifying potential security issues. HeRA is integrated with the CC-based requirements method and together these two techniques make up the elicitation phase of SecReq. SecReq is a security requirements elicitation and tracing methodology that is based upon the methodology connected toward ETSI. SecReq enhances the ETSI methodology with security requirements elicitation and writing support, as well as requirements analysis and tracing capabilities.

Furthermore, HeRA provides a requirements editor who allows technicians to enter system functional information for example, service requirements. The input to this editor is checked against security-related heuristics. In particular, these heuristics search for keywords and patterns that may indicate security-relatedness. This search for security keywords is in SecReq used, among other things, to assist a developer in selecting suitable parts of the CC security requirements knowledge, and thus, HeRA works closely together with the CC-based method. However, this tool does not offer elicitation of security requirement of IoT application.

The other tool is BSN-Care [10]. This tool is secure IoT-based healthcare system, using BSN (Body Sensor Network). BSN technology is one of the core technologies of IoT developments in healthcare system, wherever a patient are often monitored using a assortment of tiny-powered and light-weight wireless sensing element nodes. The development of this new technology in healthcare applications without considering security makes patient privacy vulnerable. In this proposed system, LPU plays a significant role. It collects the sensor data and firmly sends to the BSN-Care server. BSN-Care system can satisfy all the essential security requirements of IoT based healthcare system using BSN. However, this tool only monitored security in healthcare system but does not propose the elicitation and analysing security requirements for other IoT based applications.

The Haier SmartCare [11] is a smart device designed to control and read information from variety sensors placed throughout a user’s home which include a smoke detector, a water leakage sensor, a sensor to check whether doors are open or closed, and a remote power switch. These sensors are connected through the ZigBee protocol. The primary function of this device is to enable the user to better monitor their homes when they are away and to get alerts based on sensor information. In order to help better understand the security vulnerabilities of existing IoT devices and promote the development of low-cost IoT security methods, they use both commercial and industrial IoT devices such as from which the security of hardware, software, and networks are analysed and backdoors are distinguished.

A detailed security analysis procedure will be explained on a home automation system and a smart meter proving that security vulnerabilities are a regular issue for most devices. Security solutions and mitigation methods will also be discussed to help IoT manufacturers secure their products. Provided for that IoT devices are widely used in both business and industrial applications, the selected sample IoT devices include a smart controller for a home automation system and a smart meter for modern power grids.

Through these analysis, they substantiate and demonstrate the limitations of current IoT device design methodologies when countering different cyberattacks from the hardware, software, and network levels. They will further develop countermeasures to alleviate security threats to existing IoT devices so that more secure devices can be deployed in the coming IoT era. However, this tool is used primarily for monitor a home and get alerts using sensor information but not cover eliciting security requirements.

**III. Results and Discussion**

We compare the six existing tools related to security requirements of IoT application. The comparison features based on methodology/approach/technique, IoT domain based applications, requirement representation, IoT security aspects and purposes in requirements as shown in Table 1.
Security Requirements Engineering is one of the most important aspects in the achievement of secure software systems in the software development process. Analysing security requirements at the early phase contributes to the success of secure IoT application. However, current works and tools do not provide a proper means to analyse security requirements for IoT application and the current techniques are tedious, expensive and time consuming. Thus, it is necessary to have automated analysing and on security requirement especially on IoT application. According to the analysis as shown in Table 1, we found that there are various methodologies or approaches or techniques used by existing security tools for IoT application. We also found that all IoT applications domain namely: industrial, smart city and health well-being are concern in security issues when developing a tools. We also found that in terms of security requirements, most of the tools cover the aspect of authentication, and confidentiality, then followed by availability and integrity. All of these aspects are the important security elements that need to be considered in the IoT application to ensure the safety, security and privacy of the data to the users. Surprisingly, none of the tools cover authorization aspects although it is one of the important attribute among security requirements. Based on our review, we also found that elicitation and analysis are the most important process in requirement engineering rather than other process. So, this analysis support that elicitation and analysis are the required process in developing a secure IoT applications.

Table 1: Comparison of Tools Support for IoT Application

<table>
<thead>
<tr>
<th>Tools Name</th>
<th>Methodology/Approach / Technique</th>
<th>IoT Domain</th>
<th>Requirement Representation</th>
<th>IoT Security Requirement Aspect</th>
<th>Process in Requirement Engineering</th>
</tr>
</thead>
<tbody>
<tr>
<td>GARMISR OID</td>
<td>Android Asset Packaging Tool (AAPT) Bash and python scripts</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>/</td>
</tr>
<tr>
<td>AVISPA</td>
<td>Multi-factor biometric user authentication</td>
<td>X</td>
<td>X</td>
<td>/</td>
<td>X / X /</td>
</tr>
<tr>
<td>EliciT</td>
<td>Ontology-guided</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X / X /</td>
</tr>
<tr>
<td>HeRA</td>
<td>SecReq UMLsec</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>/</td>
</tr>
<tr>
<td>BSN-Care</td>
<td>Lightweight anonymous authentication protocol</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>/</td>
</tr>
<tr>
<td>Haier SmartCare</td>
<td>Mitigation methods</td>
<td>/</td>
<td>X</td>
<td>X</td>
<td>/</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>0</td>
</tr>
</tbody>
</table>

Submit Date: 05.07. 2018, Acceptance Date: 22.08.2018, DOI NO: 10.7456/1080SSE/319

Research Article - This article was checked by Turnitin

Copyright © The Turkish Online Journal of Design, Art and Communication
V. Future Works
The engineering of the requirements for a business, system or programming application, component, or (contact, data, or reuse) center includes distant more than just engineering its functional requirements. One must also engineer its quality, data, and interface requirements as well as its architectural, design, implementation, and testing constraints. However, most requirements engineers are poorly trained and not well prepared to elicit, analyse, and specify security requirements, often confusing them with the architectural security mechanisms that are traditionally used to fulfill them. They thus end up specifying architecture and design constraints rather than the correct security requirements. Analysing security requirements for IoT application during the early phase of application development is important. Due to the rapid development of IoT application, analysing of security requirements is seriously needed to ensure the safety, privacy and confidentiality of data. For future works, we plan to develop an approach and tool that is able to support elicitation and analysing of security requirements of IoT applications. This tool will help the developer to elicit security requirements in early stage of IoT application development.

VI. Acknowledgements
The authors would like to acknowledge Universiti Teknikal Malaysia Melaka (UTeM) and Ministry of Education (MoE) for its support and the funding of this FRGS research grant: FRGS/1/2016/ICT01/FTMK-CACT/F00325.

REFERENCES