Risk-Driven Security Metrics Development for an e-Health IoT Application

Reijo M. Savola, Pekka Savolainen, Antti Evesti
VTT Technical Research Centre of Finland
Oulu, Finland

Habtamu Abie
Norwegian Computing Center
Oslo, Norway

Markus Sihvonen
MPY Palvelut Oyj
Helsinki, Finland

Abstract—**Security and privacy for e-health Internet-of-Things applications is a challenge arising due to the novelty and openness of the solutions. We analyze the security risks of an envisioned e-health application for elderly persons' day-to-day support and chronic disease self-care, from the perspectives of the service provider and end-user. In addition, we propose initial heuristics for security objective decomposition aimed at security metrics definition. Systematically defined and managed security metrics enable higher effectiveness of security controls, enabling informed risk-driven security decision-making.**

Keywords-Android; security metrics; risk analysis; security effectiveness

I. Introduction

The number of persons with chronic medical diseases is increasing dramatically worldwide. Their treatment is taking a major proportion of national health care budgets. Both from individual patient and budget-saving perspectives, the most effective treatment is informed self-care. Self-care includes medical measurements that can be realized with new Internet-of-Things (IoT) solutions. Sensor devices and well-managed data collection are central to self-care. In the emergence of new digital e-health applications, security and privacy are major concerns.

Security and privacy requirements are high in healthcare, even for intra-organizational communication. Attacks of various types render it possible to compromise an IoT sensor-based system and potentially even a centralized electronic health record (HER) system to which it has connections. Moreover, as software-intensive systems incorporate increasingly critical applications, grow more difficult to manage, and utilize ever more complex and networked software, they become exposed to security risks in increasing numbers [1].

Quantification techniques are widely used in engineering to enable informed decision-making [1]. Systematically and carefully developed and managed *security metrics* increase understanding of the security effectiveness (SE) level of the target system. Security effectiveness is the assurance that the stated *security objectives* (SOs) are met in the target system, and the expectations for resiliency in the use environment are satisfied, while at the same time the system does not behave contrary to the intended behavior [2–4]. Security objectives are high level statements of intent to counter the identified threats and/or to satisfy the organizational security policies and/or the assumptions identified [5].

To ensure sufficient SE, the developed metrics should be based on prioritized results of iterative risk analysis, rather than best practices.

The main contribution of this study is in analyzing the security risks and objectives of an e-health self-care system that contains medical IoT sensors, communication and storage solutions, processing and presentation of the data, and the appropriate interfaces in between. We also discuss risk impact assessment. Moreover, the study proposes initial heuristics for security metrics development via decomposition of security objectives. The proposed heuristics cover the main risk-driven security controls and strategies for the decomposition. In addition to the decomposition heuristics, initial measurement architecture development stages are proposed.

The paper continues with Section II, discussing the target system of study. Section III describes the risk analysis process used and presents prioritized results from its application. Section IV proposes heuristics for security metrics development, based on the risk analysis results. Section V examines related work, and Section VII offers conclusions and discusses future research questions.

II. Target System

The target system of this study is an envisioned e-health IoT (Internet of Things) system with care functions for elderly people and persons with chronic diseases, including monitoring and chronic disease self-care. The system includes various biomedical sensors (Biomedical Sensor Network), a gateway device at home and the service provider functions. The sensors include well-being meters, motion detectors, blood glucose meters, and blood pressure meters. Fig. 1 depicts the larger ecosystem, in which our system under investigation is a sub-system. In the figure, the scope of this study is Node 1 (end-user environment) and Node 4 (service provider).

978-1-4799-7755-0/15/$31.00 ©2015 IEEE

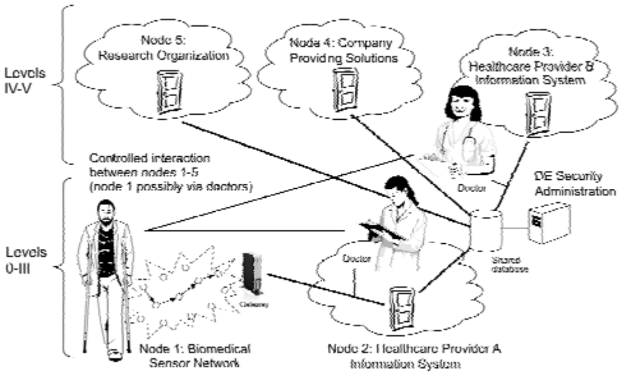
Levels IV-V
Node 5: Research Organization
Noda 4: Company Providing Solutions
Node 3: Healthcare Provider B Information System
Controlled interaction between nodes 1-5 (node 1 possibly via doctors)
Doctor
DE Security Administration
Levels 0-III
Doctor
Shared database
Gateway
Node 1: Biomedical Sensor Network
Node 2: Healthcare Provider A Information System

Figure 1. Ecosystem around the system under investigation. [6]

There are considerable security and privacy challenges in this environment. Wireless communications can be intercepted nowadays quite easily, and protecting the IoT sensors is more challenging than protecting devices with more computing power. Moreover, the introduction of fake sensor nodes is possible.

In the use cases resulting from Fig. 1, sensor data are stored in private databases. In this study, we assume that the databases reside in a well-managed shared database on the service provider's premises.

Security and privacy requirements are high in healthcare, even for intra-organizational information sharing. The privacy of patients is protected by general healthcare regulations such as the United States HIPAA (Health Insurance Portability and Accountability Act) [7].

In this study, the target system is analyzed from two viewpoints: Case 1: the service provider's business perspective, and Case 2: the end-user's perspective.

III. The Risk Analysis Process and its Results

A. *Iteration in Risk Analysis*

Risk management decisions, SO descriptions, and supporting security metrics development activities should be based on careful, sufficiently detailed risk analysis (RA) of the target system. To increase the reliability of the results of the RA, it should be carried out in several, iterative phases. For example, telecommunications company Ericsson uses an iterative RA process comprising three iterative instances of RA sessions [8]: (i) RA1, conducted when the product requirements are defined, (ii) RA2, when the product is being specified, and (iii) RA3, when the product is being designed and verified. The main focus in RA1 is on the points where risks reside in a business value chain, while RA2 focuses mainly on analyzing the risk environment from a product or solution feature perspective, and RA3 focuses on verifying how the identified risks have been mitigated and what the residual risks are [8].

In this study, we aim for high SE. Therefore, the SOs are based on prioritized RA results. In practice, however, the SOs are based on risk management (RM) *decisions*: RM can choose for a risk to be accepted, mitigated, or cancelled.

B. *The RA Process used for the Target System*

Below, we explain the RA process used for gaining the risk knowledge in the target system and give a panorama of its overall results. The purpose of the RA was to analyze the security risks of the envisioned target system from two perspectives: Case 1 focused on the service provider perspective, and Case 2 on the end-user perspective.

The RA was carried out in co-operation between the security researchers from VTT Technical Research Centre of Finland, and the security and business experts of the Finnish telecom company MPY Palvelut Oyj. The RA process consisted of two main phases: risk identification and risk prioritization. The latter phase included (i) severity and probability scoring and (ii) priority ordering of risks. The relations between risk impact and SE were analyzed in co-operation with the security experts at VTT and the Norwegian Computing Center (NR).

The RA commenced with a brainstorming meeting wherein participants were divided into two teams – both comprising persons from both organizations with enough expertise in the target system, business, and security. First, one team concentrated on Case 1 and the other on Case 2. After a time, the teams switched the cases. Therefore, two independent risk sets were obtained for each case. Next, both teams categorized these risks and presented the results to the full group. The process was continued in a plenary session. The independent risk sets were combined through the removal of duplicates and interpretation. Finally, the probability of each risk and the severity of its consequences were rated. In the latter phase, the 'raw' prioritization results were ordered in the light of expert opinions. It should be noted that risk prioritization is not unambiguous, and even small changes in the system's assumptions can change it.

C. *Overview of Prioritized Risks*

In the following, we present an overview of the RA results. Table I lists the prioritized risks for Case 1, and Table II for Case 2. The rank of each risk is indicated by the number in the first column. In the tables, 'S' refers to the severity of the consequences if the risk is actualized, and 'P' denotes the probability of the risk being realized. The scale for each is 0-3. The former number represents no risk, and the latter indicates extremely high severity or probability. In the tables, the abbreviation 'R:' means 'risk arising from'. It is used in connection with *attack types*, *vulnerabilities*, and *faults* that cause a risk. Most of the risks listed here are of these types. Naturally, the risk identification process also yields thoughts on threats and attack types.

All of the risks listed in the tables can be seen as important. However, further analysis, such as SO definition, should be carried out in priority order. It should be noted that the definition of SOs and security controls is not a one-to-one mapping to the list of risks. For example, security controls such as access control can be used to mitigate several risks, and some risks contribute to others. These interdependencies can be rather complex. Even though there are many interdependencies among risks, they should be listed in the way shown in the tables. Otherwise, information about the prioritization can be lost easily. This information is highly relevant for the application of the proper emphasis in SO definitions.

TABLE I. PRIORITISED RISKS FOR CASE 1 (BUSINESS), SECURITY RISK IMPACT BASED ON S AND P

R#	Description	S	P
1	R: infrastructure problems due to core network or power failures	3.00	3.00
2	R: unavailability of the system at a critical moment (DoS, denial of service)	3.00	3.00
3	R: network failure in sparsely populated area	3.00	2.00
4	R: update process in servers or customer devices	2.00	3.00
5	R: vulnerabilities in software (SW) or hardware (HW)	2.00	3.00
6	R: human errors	2.00	3.00
7	R: third parties not meeting requirements	2.00	2.00
8	R: data integrity problems	2.00	2.00
9	R: activation of malware at a critical moment	2.00	2.00
10	Personal injury	3.00	1.00

TABLE II. PRIORITISED RISKS FOR CASE 2 (END-USER), SECURITY RISK IMPACT BASED ON S AND P

R#	Description	S	P
1	R: infrastructure problems due to core network or power failures	3.00	3.00
2	R: server inavailability	3.00	2.00
3	R: configuration errors	2.00	3.00
4	R: wrong or missing medication due to system problems	2.00	3.00
5	R: vulnerabilities in devices	2.00	3.00
6	R: user errors	2.00	3.00
7	R: software bugs	2.00	3.00
8	R: empty batteries	2.00	2.00
9	R: death due to problems in system	3.00	1.00
10	R: wrong use of devices	1.00	3.00

The overall results of the RA have the following pattern: availability concerns are ranked as highest. Next in priority come configuration correctness concerns. After these, come SW and HW quality concerns, especially risks arising from vulnerabilities. Following this pattern, different types of risks are prioritized, some having interdependencies with the above mentioned ones.

Sufficient usability is one of the core design goals for the target system. The target systems will be deployed in an environment in which usability is very important. Decisions on trade-offs between security effectiveness and usability are needed, with support from adequate metrics depicting both dimensions. A sufficient usability to security ratio is needed to mitigate especially R1.6 (human errors) and R2.6 (user errors).

Authentication and authorization (AA) related concerns were not ranked in the Top-10 list, although R2.10 (wrong use of devices) might result from the lack of proper AA.

The prioritized risks have many interdependencies with each other, as always in risk analyses. Recognized interdependencies help during the controls' selection. The security control selected for any specific risk can also mitigate other risks.

Although criticized, security risk impact assessment, is still an often used technique for the measurement of the strength (SE) of protection mechanisms, or security controls.

The potential risk impacts of threats on the security controls can be calculated using appropriate metrics that measure the effectiveness of these security controls. Security metrics can be quantified using risk impact assessment techniques. A higher risk impact of a security control means that the effectiveness of it is weak. In this kind of situation, the security controls need to be adapted better to mitigate the corresponding risks. A lower risk impact indicates strong SE. Further action might not be needed in this case. However, if costs or resources need to be saved, one might want to analyze whether the level of protection can be decreased a bit, to an appropriate goal level.

D. *Availability Concerns*

According to the RA results, the main security risk category in the target system is security risks arising from problems due to the core network or power failures (R1.1, R2.1 and R1.3), and unavailability of the system at critical moment (R1.2) and server unavailability (R2.2). Since R1.1, R2.1 and R1.3 also address risks under the control of third parties (core network and power infrastructure), R1.7 has interdependencies with them. All of these risks cause availability problems to the service. R1.9 (activation of malware at a critical moment) and R2.8 (empty batteries) are availability risks, too.

E. *Lack of Configuration Correctness*

Many risks in both cases are related to configuration correctness. In Case 1, R1.4 (update process in servers or customer devices), and in Case 2, R2.3 (configuration errors) directly relate to it. In addition, many other risks can depend on configuration correctness, such as the availability risks (R1.1, R1.2, R1.3, R2.1, and R2.2).

R1.10 (personal injury) and R2.9 (death due to problems in the system) can result from other risks. Consequently, sufficient access control, confidentiality, integrity, and availability solutions are needed to mitigate this risk.

F. *Security Risks Arising from SW and HW Quality Problems*

Risks due to HW, SW and device vulnerabilities are in focus in the RA results (R1.5 and R2.5). Moreover, the concerns of SW quality (R2.7) were also ranked among the Top-10 risks in Case 2.

G. *Death Due to Problems in the System*

Death resulting from problems in the target system (R2.9) is a severe incident, which is possible in any healthcare support systems, although not seen as very likely. The problems from which death can result include service availability, configuration correctness, SW and HW errors and many others. This risk category can be seen as high-level, and mitigation of this risk can be achieved by developing and enforcing countermeasures for more specific risk categories. Furthermore, compliance with rules and regulations, and monitoring the compliance via metrics, is a specific objective.

IV. Heuristics for Security Metrics Development

In the following, we discuss the process of deriving risk-driven security metrics from the RA results, including initial considerations of heuristics for SO decomposition, aiming at security metrics development. We analyzed SO decomposition for a generic e-health IoT system in [9].

In a risk-driven approach, SOs are defined based on prioritized RA results. The actual security controls are based on SOs. Some high-level SOs can mitigate several risks. The initial considerations for decomposition heuristics can be applied separately to each SO resulting from the RA. The actual security measurement activity utilizing the metrics resulting from the process varies. Some work can be automated, while monitoring security policies involves assessment and interviews. In all cases, a risk-driven SO decomposition method should be used to define security metrics to ensure sufficient SE.

A. Background

Fig. 2 depicts a simplified example of how authentication-related SOs can be decomposed [10]. Basic measurable components (BMCs) are leaf components of a decomposition that clearly manifest a measurable property of the system [11]. High-level BMCs shown in Fig. 2 are Authentication Identity Structure Authentication Identity Uniqueness, Authentication Identity Integrity, Authentication Mechanism Integrity and Authentication Mechanism Reliability [11].

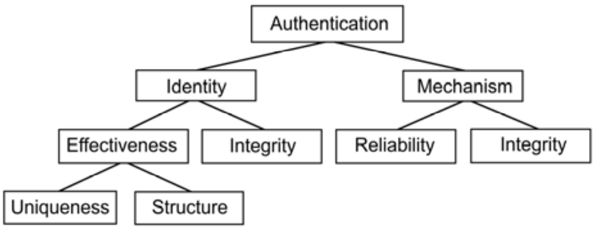

Figure 2. An example authentication decomposition based on [10]

Six strategies for security measurement objective decomposition, aimed at security metrics development and management, were proposed in [1]. The *basic strategies* proposed addressed security configuration correctness, direct partial security effectiveness, and software and system quality. *Integrated strategies* were proposed to support compliance with best practice and regulations, pure security effectiveness, and the security effectiveness versus efficiency trade-off. In the approach of [1], the security effectiveness goals are introduced using a Security Effectiveness Abstract Model (SEAM), a simplified model that encompasses the core knowledge of factors contributing to the SE of the target system. It is not possible to measure SE as a whole directly; however, it is possible to measure the factors contributing to it.

B. Heuristics for Availability Risks

Availability risks were identified as the main risk category in the Top-10 lists for both cases.

TABLE III. Heuristics for Availability

Stage	**Description**
A1	Identify applicable security measurement objectives for the SOs designed to: • R1.1, R2.1 – control infrastructure to prevent service unavailability due to core network or power failures • R1.2 – control service availability of the system at critical moments • R1.3 – control network failures in sparsely populated areas • R2.2 – control server availability • R1.9 – control prevention of malware activation at critical moment • R2.8 – control prevention of risks due to empty batteries Although these SOs all address the availability dimension, their specical objectives vary: • Since availability of the infrastructure includes third parties, attention should be paid to technical requirements and service level agreements (SLAs). • The SO related to R1.2 is more generic, including availability considerations of the target system. A suitable alarm system should be utilized. • Controlling server availability is a specific goal, and can be achieved by suitable procedures, maintenance and service mirroring techniques. • Malware prevention at critical moments requires antivirus program installation and management, on both the service provider's and the user's side. • Unavailabity of service due to empty batteries should be prevented by battery monitoring techniques in the home service, and alarm notifications. The security measurement objectives should involve monitoring how the procedures are obeyed, how the service-level is achieved with regard to SLAs, and proper management of alarms. Investigate available and attainable availability evidence and its relevance to SE. SE should be emphasized: priotize evidence with respect to SE.
A2–A3	Define SEAM, which includes mapping to security measurement objectives from Stage A1. If a reference model, such as a standard of availability relevant to the perspective of the risks listed, is available, analyze the correspondence of the security measurement objectives and the reference model.
A4	Identify the system components relevant to the availability objectives. Setting boundaries is important. The components are architectural components (like modules, devices, protocols, interfaces, platforms) [1].
A5	Carry out objective decomposition in an iterative way. Use SEAM to guide the process in order to ensure that the resulting metrics contribute to SE.
A6	Identify the measurement points in the metrics hierarchy. These are data structures, devices or files where the configuration data and the deployment control resides [1].
A7	In the decomposition, develop BMCs that aim for feasible metrics or the use of available metrics, with the goal being a conclusion of 'OK' or 'not OK' [1].

Table III proposes security metrics development heuristics for availability, following the approach in [1]. The stages in the table follow the stage numbering from [1]. The proposal can be followed without knowing the details from [1].

Note that if the target system is used in paramedic scenarios, the system should have heightened availability of critical information. However, paramedic use was not within the scope of the analysis. Potential paramedic use should be, however, considered when designing the envisioned system.

C. Heuristics for Configuration Correctness

As discussed earlier, security risks due to problems in configuration correctness are seen as an important category, as many prioritized risks are either directly or indirectly related to them. In addition to having a direct effect in controlling specific risks, various configuration correctness controls contribute indirectly to the overall SE of the system. Table IV proposes security metrics development heuristics for configuration correctness.

TABLE IV. HEURISTICS FOR CONFIGURATION CORRECTNESS

Stage	Description
A1	Identify applicable security measurement objectives for the SOs designed to: • R2.3 – control critical configuration errors (general) • R1.4 – control critical configuration errors during the course of the update process (specific) Configuration correctness can be achieved by developing and enforcing suitable configuration management practices and rules. Because of R1.4, special attention to the update process is needed. The security measurement objectives should involve monitoring how the rules are obeyed. Investigate available and attainable configuration evidence and its relevance to SE. SE should be emphasized: priotize evidence with respect to SE.
A2–A3	Define SEAM, which includes mapping to security measurement objectives from Stage A1. If a reference model, such as a standard of configuration correctness relevant from the R2.3 and R1.4 perspective, is available, analyze the correspondence of the security measurement objectives and the reference model.
A4	Identify the system components relevant to the configuration correctness objectives. Setting boundaries is important. In the case of R2.3, it is important to set limits to which parts of the system can be updated using particular kinds of updating procedures. The components are architectural components (like modules, devices, protocols, interfaces, platforms) [1].
A5	Carry out objective decomposition in an iterative way. Use SEAM to guide the process in order to ensure that the resulting metrics contribute to SE.
A6	Identify the measurement points in the metrics hierarchy. These are data structures, devices, or files in which the configuration data and the deployment control resides [1].
A7	In the decomposition, develop BMCs that aim for feasible metrics or the use of available metrics, with the goal being a conclusion of 'OK' or 'not OK' [1].

Although compliance was not seen as a risk in the RA, compliance with healthcare regulations is a requirement in the target system. Compliance issues can be handled in security metrics development in the same way as configuration correctness. Moreover, the compliance strategy of [1] can be utilized.

D. Heuristics for SW and HW Quality

Heuristics for SW and HW quality objective decomposition are proposed in Table V.

TABLE V. SW AND HW QUALITY

Stage	Explanation
A1	Security measurement objectives of SW and HW quality are based on good SW and HW quality objectives. However, their connection to SE should be analyzed. SOs based on R1.5 and R2.5 emphasize controlling vulnerabilities in SW and HW, whereas R2.7 is more general, concerning risks arising from SW bugs.
A2–A3	Define or utilize a pre-existing SEAM that emphasize adequate SE of the target system, and incorporate SW and HW quality objectives into it.
A3	Applicable vulnerability databases offer important knowledge of SW&SQ. A reference model based on the robustness to prioritized vulnerabilities can be used.
A4	Identify components where evidence for quality can be gathered.
A6	Measurement points are typically focused on SW & HW testing activities.

V. RELATED WORK

State-of-the-art information systems in healthcare organizations utilize architectural solutions such as service oriented architectures (SOA) [12]. Jafari et al. [13] discuss security metrics development goals for e-healthcare information systems. However, they do not propose heuristics or strategies for the actual security metrics development. Jaatun et al. investigate security for tablet-based e-health applications in [14]. Aman and Snekkenes [15] list empiral research efforts for risk management in IoT-based e-health environments.

Challenges of requirement decomposition were discussed by Kirkman [16] and Koopman [17]. They list the following challenges in decomposition: excessive subsystem decomposition, insufficient decomposition, 'gaming' promoted by too great a focus on goals, unattributed requirements, excessive hierarchy and issues of change management. Our SO decomposition approach is similar in general to the Goal Question Metrics (GQM) of Basili et al. [18], a three-level decomposition approach for refining software measurements specification. The GQM method lack strategies or heuristics to define their security-relevant content aimed at *security* metrics. There are already plenty of security metrics approaches proposed in the literature. Summaries of these can be found in [19–22]. However, the state of the art lacks widely accepted and well-validated approaches to security metrics. This is due to the facts that security is often considered to be an 'add-on' property, the security research field itself is in its infancy, and there is a lack of suitable real incident-based data for use in risk analysis and risk-driven security metrics development [23].

VI. Conclusions and Future Work

We analyzed security risks of an envisioned e-health Internet-of-Things system with functions for elderly people and persons with chronic diseases, including patient monitoring and chronic disease self-care. According to the risk analysis results, system availability risks are ranked as highest. Next in priority are risks related to erroneous configuration. Software and hardware quality concerns, especially risks arising from vulnerabilities are emphasized too. Risk impact analysis can be used in security metrics development to receive indications of security effectiveness.

We also proposed initial heuristics for security objective decomposition, aimed at security metrics development. Availability objective decompositions include considerations for alarm management, monitoring of procedures, rules, and agreements, and service mirroring. The decomposition of configuration correctness objectives is based on investigation of specific settings in the configuration that contribute essentially to security effectiveness. Decomposition of software and system quality objectives incorporates investigation of vulnerability databases.

In our future work, we plan to focus on defining more detailed security objectives for the target system, and developing a hierarchy of security metrics for it.

Acknowledgments

The work presented here has been carried out in four research projects: (i) ASSET – Adaptive Security for Smart Internet of Things in eHealth (2012–2015) funded by the Research Council of Norway in the VERDIKT program, (ii) SAICS – Situation Awareness in Informatics and Cyber Security, launched by the Academy of Finland (2014–2015), (iii) the IoT Program (2012–2015), launched by the Finnish Strategic Centre for Science, Technology and Innovation TIVIT Plc., and co-funded by the Finnish Funding Agency for Innovation (Tekes) and VTT, and (iv) the SASER-Siegfried Celtic-Plus project (2012–2015), co-funded by Tekes and VTT.

References

[1] R. Savola, “Strategies for Security Measurement Objective Decomposition,” ISSA 2012, 15–17 August 2012, Johannesburg, South Africa, 8 p.

[2] R. Savola, “Security Metrics Taxonomization Model for Software-Intensive Systems,” Journal of Information Processing Systems, Vol. 5, No. 4, Dec. 2009, pp. 197–206.

[3] W. Jansen, “Directions in Security Metrics Research,” U.S. National Institute of Standards and Technology, NISTIR 7564, Apr. 2009, 21 p.

[4] ITSEC. Information Technology Security Evaluation Criteria (ITSEC), Version 1.2, Commission for the European Communities, 1991.

[5] ISO/IEC 15408-1:2005. Common Criteria for Information Technology Security Evaluation – Part 1: Introduction and General Model, International Organization for Standardization and the International Electrotechnical Commission, 2005.

[6] R. Savola and M. Sihvonen, “Metrics Driven Security Management Framework for E-Health Digital Ecosystem Focusing on Chronic Diseases,” MEDES 2012, Oct. 28-31, 2013, Addis Ababa, Ethiopia, pp. 75–79.

[7] HIPAA. Health Insurance Portability and Accountability Act (HIPAA). U.S. Public Law 104—191. 1996.

[8] R. Savola, C. Frühwirth, and A. Pietikäinen, “Risk-driven Security Metrics in Agile Software Development – an Industrial Pilot Study”, Journal of Universal Computer Science, Vol. 18, No. 2, Sept. 2012, pp. 1679–1702.

[9] R. Savola and H. Abie, “Metrics-driven Security Objective Decomposition for an E-Health Application with Adaptive Security Management,” In Proc. ASPI 2013, Sept. 8, 2013, Zürich, Switzerland, 8 p.

[10] C. Wang and W.A. Wulf, “Towards a Framework for Security Measurement”, Proceedings of 20th National Information Systems Security Conference, 1997, pp. 522–533.

[11] R. Savola and H. Abie, “Development of Measurable Security for a Distributed Messaging System,” International Journal on Advances in Security, Vol. 2, No. 4, 2009, pp. 358–380 (published in March 2010).

[12] S.C. Chu, “From Component-based to Service Oriented Software Architecture for Healthcare”, *Proc. HEALTHCOM ’05*, pp. 96–100.

[13] S. Jafari, F. Mtenzi, R. Fitzpatrick, and B. O’Shea,. “Security Metrics for E-healthcare Information Systems: a Domain Specific Metrics Approach. Int. Journal of Digital Society, 1(4), 2010, 238–245.

[14] M. Jaatun, E. Jaatun, and R. Moser, “Security Considerations for Tablet-based eHealth Applications,” Proc. CEUR Workshop 2014, pp. 27–36.

[15] W. Aman and E. Snekkenes, “An Empirical Research on InfoSec Risk Management in IoT-based eHealth”, Proc. MOBILITY 2013, pp. 99–107.

[16] D. Kirkman, “Requirement Decomposition and Traceability,” Requirements Engineering, Vol. 3, No. 2, 1998, pp. 107–114.

[17] P. Koopman, “A Taxonomy of Decomposition Strategies Based on Structures, Behaviors, and Goals,” Design Theory & Methodology ’95, 1995.

[18] V. Basili, G. Caldiera, and H.D. Rombach, ”The Goal Question Metric Approach,” J. Marciniak (Ed.), Enclyclopedia of Software Engineering, Wiley, 1994.

[19] D. S. Herrmann, Complete Guide to Security and Privacy Metrics – Measuring Regulatory Compliance, Operational Resilience and ROI, Auerbach Publications, 2007, 824 p.

[20] A. Jaquith, Security Metrics: Replacing Fear, Uncertainty and Doubt, Addison-Wesley, 2007.

[21] N. Bartol, B. Bates, K.M. Goertzel, and T. Winograd, Measuring Cyber Security and Information Assurance: A State-of-the-art Report, Information Assurance Technology Analysis Center, May 2009.

[22] V. Verendel, “Quantified Security Is a Weak Hypothesis: A Critical Survey of Results and Assumptions,” New Security Paradigms Workshop, Oxford, U.K., 2009, pp. 37–50.

[23] R. Savola, “On the Feasibility of Utilizing Security Metrics in Software-Intensive Systems,” International Journal of Computer Science and Network Security, Vol. 10, No. 1, 2010, pp. 230–239.