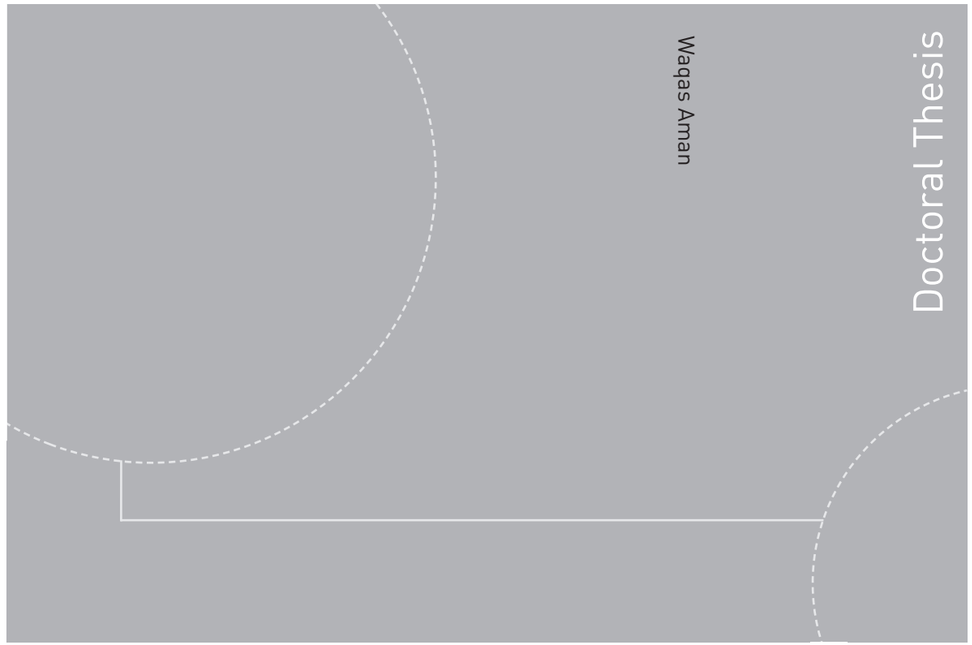
ISBN 978-82-326-1626-8 (printed version)
ISBN 978-82-326-1627-5 (electronic version)
ISSN 1503-8181



Doctoral theses at NTNU, 2016:144

Waqas Aman

Adaptive Security in the Internet of Things

NTNU
Norwegian University of
Science and Technology
Faculty of Computer Science and Media Technology
Norwegian Information Security Laboratory - NISLab

NTNU
Norwegian University of
Science and Technology

Doctoral theses at NTNU, 2016:144

NTNU

NTNU
Norwegian University of
Science and Technology

Waqas Aman

Adaptive Security in the Internet of Things

Thesis for the degree of Philosophiae Doctor

Gjøvik, April 2016

Norwegian University of Science and Technology
Faculty of Computer Science and Media Technology
Norwegian Information Security Laboratory - NISLab

NTNU
Norwegian University of
Science and Technology

NTNU
Norwegian University of Science and Technology

Thesis for the degree of Philosophiae Doctor

Faculty of Computer Science and Media Technology
Norwegian Information Security Laboratory - NISLab

ISBN 978-82-326-1626-8 (printed version)
ISBN 978-82-326-1627-5 (electronic version)
ISSN 1503-8181

Doctoral theses at NTNU, 2016:144

Printed by Skipnes Kommunikasjon as

Adaptive Security in the Internet of Things

Faculty of Computer Science and Media Technology
Norwegian University of Science and Technology

To my beloved parents.

Declaration of Authorship

I, Waqas Aman, hereby declare that this thesis and the work presented in it is entirely my own. Where I have consulted the work of others, this is always clearly stated.

Signed:

(Waqas Aman)

Date:

Summary

Internet of Things (IoT) is a progressively growing networking paradigm that connects various devices or *things* including sensors, software, electronics and other physical objects to collect and exchange data. Due to the emerging *things* concentration, heterogeneity, and dynamic changes in the IoT environment, achieving security efficacy in it has become a challenging task and critical concern.

Conventional security controls, such as intrusion detection and prevention system (ID/PS), firewalls, and anti-virus programs, can only monitor a particular infrastructure unit and safeguard a particular service, such as access control, with a limited context visibility. For instance, a network firewall, based on predefined rules, can only analyze packets at the perimeter and cannot observe a user or process activity or behavior on an endpoint to assess a situation holistically. From a design viewpoint, it may not be practical to implement them in resource-constrained *things*, e.g. in body sensors. These controls are platform-specific and are not feasible to be realized in a multi-vendor heterogeneous space as the IoT. Moreover, the literature concerning information security risk management (ISRM) models mostly focuses on a particular security service, e.g. confidentiality or authentication. The different phases in them are executed on an on-demand basis. Besides security, they do not consider any runtime objectives and lack efficient response strategies. The controls and ISRM models that support response mechanisms either utilize fixed or static approaches, i.e. they either implement predefined mitigation rules which might not address the dynamic threat landscape, or they tend to mitigate a risk manually and therefore, increase response latency. Furthermore, their mitigation strategies only focus on asset protection and do not assess other runtime factors, such as user and QoS preferences, that may be affected by a mitigation response. Hence, they may not be practical choices in IoT-driven systems particularly in a user-centric system, such as the IoT-enabled remote patient monitoring systems, which necessitates continuous and real-time services.

Adaptive security can be an effective tool to address threats in the IoT as it can observe, analyze and react to them dynamically on the fly. However, there is no clear evidence to establish how such a solution can be developed for this heterogeneous and lightweight objects driven network, and to what extent will it be feasible to take dynamic trade-off decisions.

These problems led this research to investigate the feasibility of a poten-

tial adaptive security solution for the IoT. This thesis contributes an Event-driven Adaptive Security (EDAS) model that satisfies the adaptive risk management requirements in IoT-driven smart spaces. It can observe, analyze and react to security changes (*things*-generated events) at the infrastructural level and offers a context-aware security adaptation approach. It utilizes a novel runtime adaptation ontology that enables the system to take a dynamic trade-off decision. Therefore, besides security, it evaluates other critical runtime objectives, such as the available resources, user preferences and QoS requirements to ensure optimized adaptation.

This thesis also contributes to the implementation and pre-development essentials of EDAS. A prototype has been developed that details the implementation blueprint of EDAS. The prototype demonstrates EDAS as a reusable, extendable, and flexible model, and evaluates it as a real-world artifact. A scenario-based evaluation method has been suggested that provides a pre-development tool to assess and realize the knowledge necessary for optimized adaptation. By using the evaluation method, this thesis provides clear evidence that EDAS can effectively address all the potential runtime factors or trade-offs in a particular adaptation decision.

Major limitations concerning, architectural constraints, scalability issues, and the use of security metrics have been identified, which are necessary for EDAS to be a robust and reliable solution for IoT security. Preliminary insights to approach these concerns in the future are also discussed.

Acknowledgments

This work has been supported by the Adaptive Security for Smart Internet of Things in eHealth (ASSET) Project. ASSET (2012-2015) has been sponsored by the Research Council of Norway in the VERDIKT program. This research has been carried out in the Norwegian Information Security Laboratory (NISLab) at the Norwegian University of Science and Technology, Gjøvik.

I am truly grateful to my supervisor Prof. Einar Arthur Snekkenes for his vigorous guidance and constant encouragement. I am also obliged to my co-supervisor Dr. Habtamu Abie of the Norwegian Computer Centre (NR) for his continuous support, guidance, and motivation. Moreover, my gratitude goes to Dr. Wolfgang Leister at NR, who have always induced me with great ideas to improve my work. They have always provided, above the planned schedule, their expertise and invaluable time whenever I needed them. They will always have my respect and sincere admiration.

I am also thankful to Hilde Bakke, Kathrine Huke, and Oddny Willassen for their immense support that they have provided me while settling in Gjøvik. They have warmly welcomed me and have been a great help throughout my studies.

I am grateful to my friends and family without whom it would not have been easy to achieve the planned objectives efficiently. It was their smiles, company, patience, and encouragement that have always created a relaxing environment and lots of motivation in stressful times.

Contents

List of Figures

List of Tables

Part I

Research Overview

Chapter 1

Introduction

This chapter details the problem statement and motivation of the dissertation. It introduces the research questions investigated in this thesis and highlights their relationship with the published articles. Furthermore, it describes the scope of this research and highlights the organization of the dissertation.

1.1 Motivation and Research Problem

We have experienced considerable technological improvements in the last decade. In the most recent years, we are introduced to a new concept called the Internet of Things (IoT). IoT is a global network infrastructure that links physical and virtual objects through data capturing and communication capabilities [34]. It was first introduced in 1999 by Kevin Ashton [35] who associated it with the idea of Radio-Frequency Identification (RFID) utilization in supply chain management. Since then, the potential of IoT has been widely studied in a multitude of areas including transportation, power and energy, and healthcare. It is envisioned that IoT will capture real-time information in critical infrastructure, create new business models, provide a global visibility platform and extend services offered by traditional communication modes [56]. Thus, it can add potential improvements and extensions to the services offered in the physical as well as in cyberspace.

Although, having the potential to bring significant improvements in the existing services, many critical concerns, such as standardization, networking, QoS issues, as well as security and privacy, are yet to be resolved for the IoT to be a more reliable platform [37]. From a security viewpoint, the threat spectrum of IoT environments is much wider than in the traditional information and communications technologies (ICTs). It is because IoT enables service extension to accommodate a variety of sensory and mobile technologies each having a set of inherited vulnerabilities with corresponding threats. By operating together, these heterogeneous *things* may add greater utility to the existing services but may also open new means and opportunities for the adversaries to target consumers, service providers, governmental assets. A recent research made by OWASP and HP® [12] details some se-

rious vulnerabilities in the IoT. The report highlights that 90% of the *things* collect at least one piece of personal information, 60% of the *things* web interfaces are prone to cross-site scripting (XSS) attacks, 70% of the devices are prone to account enumeration attacks, and 70% of the devices communicate via unencrypted channels. This report and related studies, such as [92, 135], remind us that there are critical security and privacy concerns in the IoT, which necessitate appropriate countermeasures.

Since an IoT ecosystem consists of heterogeneous devices with potentially different communication stacks and processing mechanisms, it has a rather complex networking and communication model. Therefore, analyzing the contextual information corresponding to an adverse situation is more complicated. Moreover, because of the presence of sensory and mobile elements, the environment is changing dynamically. Due to this dynamic and complex nature of IoT, the conventional preventive and detective security controls are not sufficient to protect it against the increasing threat sophistication [136, 137]. The countermeasures they provide are heavily dependent on static information and are insufficient to provide protection against the dynamically evolved advanced attacks [89]. Furthermore, they tend to rely on a particular piece of contextual information monitored in a particular infrastructure domain. For instance, some may analyze inbound traffic at the network perimeter and others may scan a filesystem on an endpoint for possible malware, but neither of them can collectively monitor and analyze both the situations in an extended context. Analyzing risk based on a part of a context or situation may yield to false alarms [88] which may trigger unnecessary reconfigurations and may cause adverse effects, such as service disruption in a continuous monitoring service. Hence, due to their limited scope, they may fail to ensure security in a multiplex and dynamic architecture, like the IoT. Furthermore, IoT and resultant services are mainly driven by wireless resource-constrained devices or things, which may not be able to host these conventional controls.

In most cases, IoT-enabled operations are performed in unattended real-time environments in which response to the risk faced is desired to be taken dynamically. Therefore, adaptation is considered a key desirable attribute in the IoT architectures [87]. Adaptation is the property of a system to autonomously regulate its behavior and reconfigure its settings according to the situation under investigation[101]. In a risk management context, adaptation or mitigation in the conventional preventive and detective controls is either inflexible, static, or is lacking entirely. Their mitigation mechanisms mainly focus on the asset protection and do not consider other critical parameters, such as the usability or performance. While executing these mechanisms, disregarding such objectives may result in adverse influences and might further yield to a security risk itself. Moreover, in most situations, they utilize static or manual mitigation approaches in a security incident,

which may increase response latency due to human interventions. Attended security management seems to be impractical in the IoT as the number of *things* per person are significantly increasing [57]. Hence, from a security adaptation perspective, traditional controls are not suitable to be utilized in the IoT as it does not regard critical attributes in mitigation decisions and involves exhausting manual management of the monitored assets.

By analyzing these shortcomings in conventional security solutions, it is evident that we need a computationally affordable adaptive security solution for the IoT which can dynamically analyze a threat situation in a holistic context. Furthermore, the solution should be able to adapt an optimal trade-off mitigation response to the risk faced. The current literature on adaptive security and ISRM models seems to be insufficient to achieve this objective in the IoT. It either focuses only one or a particular set of security service, such as authentication and confidentiality, for example, [73, 91, 107], or describes abstract frameworks and models without sufficient details, for instance, [20], [116]. Security adaptation models, such as [113], [67], [58], only emphasize a particular component, e.g. analysis of the adaptation loop, discuss specific objective, e.g. energy consumption, or only considers protection mechanisms and does not evaluate other non-security parameters that are essential to be addressed during adaptation. There are studies, such as [46, 52, 86], which provides comprehensive approaches towards dynamic and real-time risk analysis. However, they lack to investigate adaptation as a key risk management strategy.

1.2 Research Questions

The primary objective of this thesis to develop and assess the feasibility of a potential adaptive security solution that can ensure adequate protection in an IoT-based environment. This research is fundamentally based on the conjecture that existing security engineering and corresponding controls typically makes static mitigation decisions and are insufficient to address IoT security. Whereas, adaptive security can make trade-off decisions dynamically as per the risk situation. However, there is no sufficient evidence to establish how such a solution will look like in a heterogeneous and lightweight objects driven network like the IoT, and to what extent will it be feasible. The intention is to develop a context-aware adaptation model that can analyze a threat in an extended context to reduce any false alarms, and that it can adapt security changes autonomously in agreement with the user, QoS, and resource requirements. Therefore, this objective is captured in the fundamental research question as follows:

Main Research Question: *What is the feasibility of autonomic adaptive security in the Internet of Things?*

To achieve this objective systematically, the theme of the main question is further divided into sub-questions. Each of these questions investigates a particular aspect, i.e. requirements, design, demonstration, and evaluation, of a potential adaptive security model that can be effectively utilized in IoT related scenarios. These sub-questions along with a brief description of their objectives are stated as follow:

Research Question-1: *What are the key requirements for modeling automated risk management in an IoT-based service?*
Objective: Considering adaptive security as an automated risk management activity, this research was started with understanding the scope of the risk management and its requirements in IoT. eHealth was chosen as a potential service archetype to investigate what critical elements are to be recognized and evaluated for modeling automated ISRM in a continuous IoT-enabled service. The objective was to identify and understand functional, security, and risk management requirements that are essential for modeling adaptive security in IoT.

Research Question-2: *How can we develop an effective adaptive security solution for an IoT-based service?*
Objective: The study related to this question was focused on the development of the adaptive security architecture that can observe and react to security changes in the IoT-ecosystem. The intention was to identify what can be essentially characterized as a security change in a system, how it can be monitored, collected and analyzed in a holistic context, and how can security be adapted to it. Hence, this question investigated an autonomic adaptive security architecture that can ensure context-aware risk analysis, and reason to adapt an optimal mitigation action against a threat faced.

Research Question-3: *To what extent is the proposed model feasible in real-world scenarios?*
Objective: This question investigated the feasibility of the proposed model as a real-world artifact. The proposed model was extended to a working system architecture, and its concept and features were compared with various architectures related to conventional security controls to assess which of them is a more suitable candidate for IoT security. Furthermore, this question investigated the various challenges, limitations and benefits corresponding to the proposed architecture as a technical artifact.

Research Question-4: *How and to what extent does the adaptation loop of the proposed model add value to autonomic risk management in the IoT?*
Objective: This question further evaluated the proposed model. Since changes

in an IoT environment can be dynamic, it is potentially challenging to realize the adaptation loop for various threat scenarios. This question emphasized how typical security scenarios can be realized in the proposed model. Moreover, as an adaptation decision or mitigation response always involve one or more trade-offs, this question examined how and to what extent does these trade-offs are handled by utilizing the proposed model.

1.3 List of Publications

Article 1:
Waqas Aman and Einar Snekkenes. An Empirical Research on InfoSec Risk Management in IoT-based eHealth. In the third International Conference on Mobile Services, Resources, and Users (Mobility 2013), pages 99–107, 2013 [28]

Article 2:
Waqas Aman. Modeling Adaptive Security in IoT Driven eHealth. In the Sixth Norsk informasjonssikkerhetskonferanse (NISK), 2013:61–69, 2014 [27].

Article 3:
Waqas Aman and Einar Snekkenes. Event Driven Adaptive Security in Internet of Things. In the Eighth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM 2014), pages 7–15, 2014 [29]

Article 4:
Waqas Aman and Einar Snekkenes. EDAS: An Evaluation Prototype for Autonomic Event Driven Adaptive Security in the Internet of Things. Future Internet, 7(3. Pages:225–256, July 2015 [30]

Article 5:
Waqas Aman and Einar Snekkenes. Managing Security Trade-offs in the Internet of Things Using Adaptive Security. In the Tenth International Conference for Internet Technology and Secured Transactions (ICITST-2015), London UK, 2015. Pages 362–368 [31]

The research questions and their relationship with the published articles is shown in Figure 1.1. Moreover, Figure 1.2 depicts the entire research study in a context detailing the association of the articles, the research results, and how they connect with each other.

Requirements Analysis	**Article 1**
RQ. 1 - What are the key requirements for modeling automated risk management in an IoT-based service?	
Solution Design	**Article 2 & 3**
RQ. 2 - How can we develop an effective adaptive security solution for an IoT-based service?	
Evaluation & Feasibility	**Article 4 & 5**
RQ.3 - To what extent is the proposed model feasible in real-world scenarios?	
RQ.4 - How and to what extent does the adaptation loop of the proposed model add value to autonomic risk management in the IoT?	

Figure 1.1: Research Questions & Articles Relationship

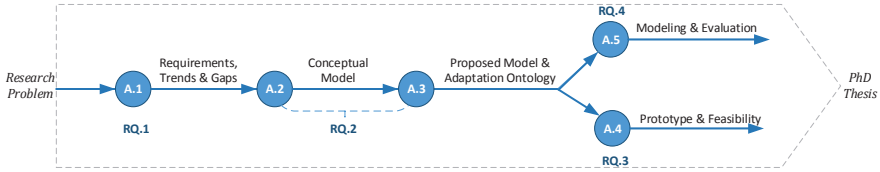


Figure 1.2: Articles Connection & Research Flow

1.4 Dissertation Scope

This dissertation is a part of the Adaptive Security for Smart Internet of Things in eHealth (ASSET) project*. ASSET aims to research and develop risk-based adaptive security models and methods for IoT-eHealth. Within the framework of the project, this dissertation focuses on the development and evaluation of a feasible adaptive security model where any appropriate monitoring and analysis methods or tools can be employed to ensure autonomous security adaptation.

Furthermore, this study mainly concentrates on the IoT in a particular application domain, i.e. an IoT-enabled eHealth infrastructure where remote patients, at home or outside wearing medical sensors, actuators, and other essential sensors, are continuously monitored from a hospital site. In the rest of the thesis, this setup will be referred to as IoT-eHealth. An ab-

*ASSET (2012-15) is a research project financed by the Norwegian Research Council under the grant agreement no. 213131/O70 in the VERDIKT (Core Competence and Value Creation in ICT) program. Project website: asset.nr.no

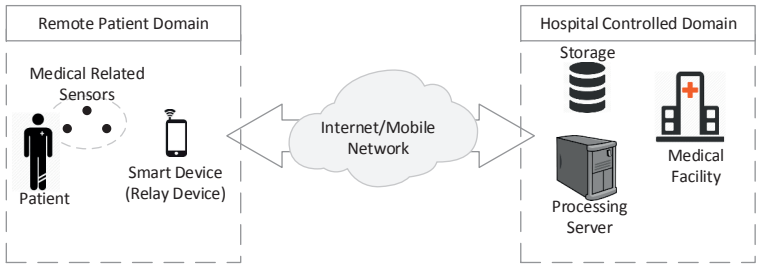


Figure 1.3: IoT-eHealth Abstract Context

stract context diagram of IoT-eHealth infrastructure is shown in Figure 1.3. Although, this research is primarily designed and validated for a restricted environment, i.e. eHealth, we suggest that the proposed architecture can be utilized in a similar IoT ecosystem, such as smart grids, sensors-based cloud services or other IoT-enabled smart environments. However, this proposition needs to be further investigated for the underlying environment.

This thesis evaluates an event-driven approach towards security adaptation where *thing*-generated events are considered as the primitive context available to characterize any change, i.e. a potential threat event, in a monitored environment. IoT and the corresponding *things*, being progressive concepts, are defined in the literature in a multitude of styles. This thesis perceives a *thing* in the IoT as an object that can autonomously react to any change (event) it senses within its internal or external environment. The reaction can be categorized as generating, storing and communicating the change information or actuating processes in response to the change. Our understanding of *things* in the IoT is more aligned with that of the Cluster of European Research projects on the Internet of Things (CERP-IoT). It realizes *things* as"active participants in business, information and social processes where they are enabled to interact and communicate among themselves and with the environment by exchanging data and information sensed about the environment, while reacting autonomously to the real/physical world events and influencing it by running processes that trigger actions and create services with or without direct human intervention" [122]. Hence, this thesis asserts that any object qualifying the stated realization can be adequately managed, in a security context, with the proposed architecture.

1.5 Dissertation Structure

This dissertation is organized into two parts. Part I details the thesis overview and includes four chapters. Chapter 1 introduces the thesis problem statement, research questions and scope. Chapter 2 details the related work. The research methodology used in this research is described in Chapter 3. Chapter 4 provides a summary of the publications and Chapter 5 summarizes the contributions of this thesis. Limitations and future work are discussed in Chapter 6. The Part II of this dissertation consists of the research publications appended as chapters.

Chapter 2

Related Work

The chapter provides an overview of the related work concerning this research work. The objective is to reflect on the major thematic areas conversed in this thesis and to discuss related methods and models.

Each section presents a brief introduction to a related topic. The introduction is followed by highlighting the literature that converses about the various concepts in the topic area. Moreover, each section provides a description of the related models, methods, and theories that describe how these studies address a given concern in that topic or approach it as a whole. Furthermore, under a particular topic, the concepts and methods on which this research is based and how it relates and connects with the related work are also detailed.

Moreover, adaptive security being the major objective of this thesis, a separate section on the related state-of-the-art is detailed in Section 2.4.1. It provides an overview of the various security adaptation approaches and highlights their shortcomings.

2.1 Internet of Things

Internet of Things (IoT) is a rapidly progressing concept in the academic, business, and social realms. Fundamentally, it is the ubiquitous presences of various objects or things including physical, wireless and wired sensor and mobile technologies, which interact with each other to fulfill common objectives [64]. Initially, it was used in improving the visibility of objects being transported in the trading networks by utilizing the RFID-tags in the Electronic Product Code^{TM}(EPC), a joint venture of Auto-ID Labs [84] and EPCglobal [4]. Semantically, IoT can be perceived as a combination of two concepts, i.e. the internet and things, and a worldwide interconnection of uniquely identifiable objects based on standard communication protocols [19]. Depending upon the particular interest of the stakeholders, IoT can be approached either from the thing or internet perspective [37].

Things in the IoT are also defined differently. For instance, they are realized as objects with identities and virtual personalities that are operating in a smart space using intelligent interfaces to communicate with social,

environment and user contexts [125]. The CERP-IoT defines things as *active participants in business, information and social processes where they are enabled to interact and communicate among themselves and with the environment by exchanging data, and information sensed about the environment, while reacting autonomously to the real/physical world events and influencing it by running processes that trigger actions and create services with or without direct human intervention* [122]. It is a more comprehensive definition and is more aligned with this thesis perspective as highlighted earlier in Section 1.4 describing the thesis scope.

This thesis recognizes IoT as a smart environment that can react to the changes or events it experiences in its ecosystem. A smart environment is a digital space that respond to the machine-readable information from the physical ecosystem independent of the smart device in use [102]. Therefore, it can be established that an IoT-based system is a smart space where interoperability among devices at the edge of the network, i.e. *things* in the IoT, is considered as the desired attribute.

Although, the RFID remains one of the notable driving technologies [106], a multitude of other sensing and mobile objects are introduced to extend the IoT vision. This extension enables a seamless integration of the physical, sensing, and mobile objects in the traditional ICT infrastructure thus, create new opportunities in social and business domains[96]. Mobile ticketing [42], eHealth [133], smart buildings [43], smart grids, environment sensing [47, 75], etc., are a few examples of IoT-enabled services and applications in different fields of cyber-physical settings.

Despite the huge potential and market value [74], many issues are yet to be addressed and evaluated to achieve the true benefits of IoT, such as, global visibility, real-time autonomous management of critical infrastructures, and other envisioned objectives as mentioned in [56]. These challenges, as highlighted in [37, 66, 96], include concerns related to networking and communication, resources consumption, such as power and computing, QoS support, standardization, security, and privacy. Some of these concerns, such as the QoS issues and resource consumptions, are ultimately a security problem as they are influencing or being influenced by it directly or indirectly. Thus, it can be established that security is one of the most critical issues that needs to be appropriately addressed. Ensuring security in IoT is a challenging task as the network is composed of different sensing, computing, and communicating devices. Such a diverse technology presence though offers service extensions and new business models, it may also introduce new means and opportunities for the adversaries to exploit assets at different level of a service architecture. These challenges, visions, and advantages motivate us to investigate effective security solutions to protect IoT from the emerging threats as the current traditional security controls are inefficient and insufficient to protect this evolving smart network

[69, 136, 137].

2.2 Context Awareness

Adaptation can have an adverse influence on the service offered, if the situation under observation is overlooked. Since adaptation, or more specifically security adaptation, heavily depends on the environmental input and investigating them holistically, it is important to ensure context awareness in the overall procedure. Context awareness is more critical in IoT scenarios, particularly in adaptation, as it is mainly a machine to machine communication without the intelligence (direct involvement) of the humans. Without making sense of the information captured in a context, adaptation might not be efficient.

The word context designates certain information [22]. However, the word *information* has also been explicitly used with context as *context information*. This thesis uses them synonymously. A context can be a set of different types of events that have a logical or timing relationship and enable us to understand a situation [24]. It characterizes a situation or entity that can be an object, place, or person [22]. Contexts can be categorized into two groups, i.e. primary and secondary. Primary context is the primitive information extracted from an entity, also called as raw data [114]. Secondary context is the information obtained after processing the raw data or primary context.

IoT being a heterogeneous environment can offer a variety of contexts. They may be describing security, location, mobility, or phenomena related to the physical environment. Each of the contexts can be managed with one or more corresponding context-aware system. A context-aware system utilizes one or more contexts and provides relevant services or information to the user [22]. Broadly, context schemes can be categorized into two groups, i.e. operational and conceptual [131]. The operational schemes emphasize how the context is captured and further processed. They can be grouped into sensed, derived, static, and/or profiled information [70]. The conceptual schemes explain how the various contexts relate to each other.

As a risk-based adaptive security approach, this thesis perceives the *thing* generated security-related events as the primary source of context for the event-based real-time risk analysis. They are generated by the software objects or applications of the monitored *things* (assets) using an event framework. This framework, usually, consists of a handler and a logger object [94]. The handler captures and pre-processes a certain context (changes or events), such as input/output exceptions or a login attempt, and the logger stores this context locally or sends them to external storage as an event log. These events describe the primitive changes in the environment and highlight the key context attributes [70]. As depicted in the Figure 2.1, a *thing*

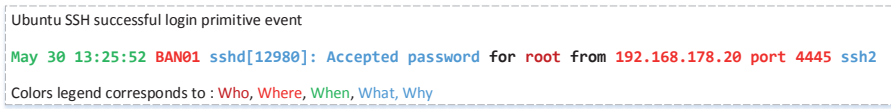


Figure 2.1: Primitive (raw) event example

generated event provides a list of fundamental context attributes that describe the who, what, when, why, and where of a change, and fully qualifies the definition of a context in a computing environment [23]. Primitive events from the monitored things are captured as the primary context. These primitive events are then filtered, normalized, and correlated to extract crucial information as secondary context using appropriate complex event processing (CEP) methods, like those highlighted in [29]. Moreover, in this thesis, an event-driven approach [93] is utilized as an operational scheme where the events form the basis of context-aware risk analysis. Whereas, an ontology ensures a context-aware adaptation as it includes all the contextual requirements necessary for risk adaptation (response).

2.3 Information Security Ontologies

Due to the presence of heterogeneous things, understanding, analyzing and accessing the knowledge to solve and approach various problems is a fundamental problem in the IoT. Ontology can be a useful tool to address this issue by organizing the knowledge in a universal form. It is used to capture, organize, communicate and reuse the knowledge of interest [51, 65]. To be more specific, an ontology defines the concepts and relations in a field of study and provides rules that explain how these concepts and relations can be utilized [53]. It can provide a basis for modeling the semantics among objects, which is an essential component to interrelate knowledge of the diverse things in the IoT [134]. Therefore, ontology assists us to understand and address a problem in a context-aware manner as it provides a platform to recognize the potential requirements and their relationships.

Literature provides a large set of proposals concerning ontologies in different field of information systems. The following text provides a brief description to ontologies in sensor networks (SN), IoT, and information security.

An ontology for adaptive SN has been proposed in [38]. Adaptive power management is the main subject of this ontology. It describes how the available nodes can adapt to an optimal power state by analyzing various environmental factors. Based on the sensor modeling language (SensorML) [16], Russomano et al., in [111, 112], proposed the OntoSensor ontology.

The OntoSensor utilizes concepts from the Suggest Upper Merged Ontology (SUMO) [100] and ISO 19115 standard (now revised as 19115-1:2014) [7]. It provides a general inference model and knowledge base for sensors. OntoSensor was later extended by Kim et al., in [80], to build a service-oriented ontology that can be utilized in the SN as a web service. Other (non-security) work captured as ontologies in SN that can potentially be used in IoT scenarios, includes search and classification of SN data [54, 99], service and data publishing and discovery [40], and task management [105].

Security and related concepts have also been the focus in modeling information security concepts as ontologies. Jeffrey et al., in [129], presented an intrusion detection ontology for computer systems. The top level concepts include host, attack, consequence, input, and means to highlight different attack vectors that can compromise a host. A similar approach has also been used in [63] in which the authors have limited the ontology scope to attacks and countermeasures concerning Session Initiation Protocol (SIP) and Voice over IP (VoIP).

Andreas et al., in [55], suggested a security ontology framework to conduct low-cost risk management and threat analysis in small and medium enterprises (SMEs). Their framework consists of four parts: a security and dependability taxonomy based on [39], a risk analysis methodology, the concepts describing the IT infrastructure, and a simulation environment. They have used the Annual Loss Expectancy (ALE) method to simulate a SME scenario.

A risk-based security ontology is proposed in [127]. The authors have extended the Common Information Model [3] to address information security related concepts in a risk assessment perspective. They have also suggested a four-phased framework to conduct risk management activities. Pekka et al., in [117] have proposed a taxonomy for service-centric systems. Their taxonomy has five major concepts including assets, attributes (the security services, such as confidentiality), threats, solutions, and metrics. Security metrics are used to measure the goodness of a system and can be related to functional operations (e.g. user login), control parameters (such as a key length), or control mechanism that utilizes the parameters [79].

Antti and Eila, in [58], have proposed an adaptation ontology for smart spaces in which they have utilized a risk-based approach. Risk levels are the only measured entities expressed in this ontology. The main problem of this ontology is its limited scope. Although claimed as a runtime ontology, it has only addressed security from a protection viewpoint and did not address other factors or non-security metrics influencing a given execution state. Similar information security ontologies can be found in [48, 61, 71].

On the application side, there are many technologies that support ontologies design, development, and implementation. Some of them are mentioned as follow. The Web Ontology Language (OWL) is a semantic web lan-

guage to represent knowledge about things [18]. OWL can be validated by applications called reasoners, such as FaCT++ [5], HermiT [6], and JFact DL [10] which are further utilized in various editors and related tools. Similar semantic technologies to model ontologies also exist. For instance, the Resource Data Format (RDF) [15] is a directed, labeled graph utilizing Uniform Resource Identifiers (URI) to name the concepts or things and their relationship in an ontology. The RDF graph is also called a triple as it model a given association as subject (first node or thing), predicate (relation), and object (second node) relationship. Different languages are used to query similar graphs or web URI. SPARQL [17] is one typical example of such languages that are used to traverse through RDF ontologies to retrieve related information. These languages can also be employed in various implementations of ontologies, such as the Protégé tool [14] and Apache Jena framework [1].

2.4 Security Adaptation

Adaptation is the attribute of a system that can autonomously monitor and regulate its behavior according to the situation or change under observation [101]. Systems or computing environments that have the ability to respond autonomically to the security threats or system failures are called autonomic or self-managing systems [62]. In terms of information security, adaptation is the ability of a system that can continuously observe the monitored environments, analyze any potential security threats faced and autonomously respond to the risk posed to reduce its consequences. Such a system helps to address the complexity by using technology to manage technology [62].

The presence of diverse and dynamic elements make the IoT-based systems more complex. It necessitates an adaptation mechanism to manage this complexity. Moreover, having a futuristic vision, IoT has an evolving, composite and non-traditional outlook thus, will create new attack vectors and threat dimensions. This evolution and complexity make the current traditional security controls and approaches impractical to be utilized in the IoT scenarios [136, 137] as they have a limited scope and have manual response mechanisms. Adaptive security can be seen as a potential candidate for the IoT security to overcome these lacking. It utilizes a feedback control loop [33, 62] to ensure the autonomic behavior. Using agents, such as sensors and actuators, and components to collect, analyze and respond (as a feedback) to the security-related information in a system, the control loop directs the security settings and reconfigurations.

To approach autonomic computing, IBM suggested the MAPE-K model [62], as shown in the Figure 2.2. The MAPE-K utilizes the Monitor, Analyze, Plan and Execute activities by employing a control loop. The Knowledge component provides the necessary information required to perform adaptation. According to IBM, an autonomic system should have the fol-

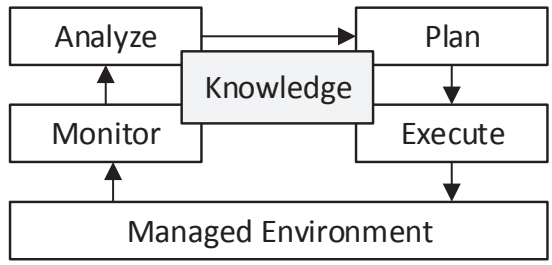


Figure 2.2: The IBM's MAPE-K Model with a Control Feedback Loop

lowing *self*-attributes: i) *self-configuration*, to adjust dynamic changes in the system components. ii) *self-healing*, adapting to the discovered system disruptions. iii) *self-optimizing*, to re-adjust the available resource parameters. iv) *self-protection*, responding to threats in a hostile situation. In [59], the authors added two more properties to this list, i.e. self-awareness and context-awareness. They defined the self-aware attribute as the system's capability to know and react to its behavior. Whereas, context-awareness is its ability to understand its operational ecosystem.

2.4.1 Adaptive Security Approaches – A state-of-the-art

A context adaptive framework has been proposed for mobile and cloud computing in [50]. It mainly emphasizes adaptive authentication of a mobile user performed in the cloud. The cloud system is represented as a finite state machine. The system has four states where each state utilizes a cognitive learning module to analyze a particular pattern, in the form a tuple, for potential intrusions. A tuple is assigned a set of probabilities and is composed of the information derived from a user request and the associated profile stored on the system. In the context of authentication, adaptation is performed in a fixed manner as the system can only allow or deny a request. The system implements only a single authentication mechanism, Message Digest and Location-based Authentication (MDLA) [49], and does not offer any parameter optimization within MDLA. The authors also suggest that the cloud infrastructure can be dynamically adapted to comply with service uptime requirements during a disruption. From user authentication perspective, the proposed system adaptation is inflexible and essentially describes an intrusion detection mechanism. Therefore, in authentication context, it doesn't have any self-properties as claimed. Availability is ensured with a self-configuration capability, which is not adequately explained. The

framework considers only authentication and availability concerns and does not regard user requirements. Addressing limited objectives also restrict the overall threat monitoring scope of the system.

A game-based adaptive security mechanism has been proposed for the IoT-eHealth scenarios in [67]. The authors have used the Markov game theory [26] to model and evaluate five adaptation strategies concerning communication channel, memory, energy, intruder, and a hybrid situation. They emphasize only a part of the IoT-eHealth, i.e. Body Area Network (BAN). No sufficient evidence has been provided on to what extent the model can be scaled to accommodate ex-BAN and future infrastructural components in the targeted application domain. Moreover, environment monitoring and adaptation response realization was disregarded. Although, the model is studied for the eHealth scenarios, it only supports device (sensor node) related changes and does not address any user (e.g. patient or physician) preferences. The model is fundamentally focused on self-optimization in authentication scenarios and self-healing at the communication level. Similar game theoretic approaches can also be found in [41, 120].

An Adaptive Security Manager (ASM) is proposed in a Genetic Messaging-Oriented Secure Middleware (GEMOM), a message oriented middleware (MOM) [21]. ASM performs the necessary tasks for security adaptation using a learning mechanism. Monitoring is facilitated by integrating external tools, such as anomaly detectors, vulnerability discovery tools, a QoS monitor and security measurement tool. The learning mechanism used by the ASM analyzer component and the type of information it utilizes are not described. Self-protection is enabled using the authorization component to protect against any intrusions. The authorization component also ensures confidentiality with a self-optimization capability. From context-awareness viewpoint, ASM focuses mainly on QoS and security related information. The authors did not explicate how user requirements are addressed in their design and lack to provide essential details of the analysis and adaptation components. Furthermore, the study mainly emphasized the monitoring aspects. The self-configuration, self-protection, and self-optimization properties are limited to particular security objective. Self-optimization is limited only to confidentiality and trust services. Self-protection is restricted to authorization only whereas self-configuration only addresses service availability.

An Ontology-based security adaptation model is proposed for smart environments [58]. The model uses security measures to collect information about the monitored environment using different agents. Details or example of the measures have not been provided. The risk faced is quantified by using a risk level based on a risk matrix define in [121] and a risk equation with a product of the threat likelihood and the asset value. Threat identification, which forms the basis of risk quantification, has not been addressed.

The proposed model uses runtime ontology to adapt. The ontology, as well as the overall design, mainly emphasizes security objectives and do not regard any non-security objectives, e.g. service or user requirements in the decision process. The security view is also limited to confidentiality and integrity related concerns and is therefore lack to monitor threats corresponding to other security objectives. Moreover, the monitoring and adaptation activities are performed on the object (device or thing) level. Such a strategy may not be feasible for resource-constrained devices, like body sensors. Similar design choices limit the information to be analyzed and might not be security efficient as the potential context from the neighboring and other associated objects is disregarded. However, the model fully realizes the self-configuration and self-optimization capabilities as an autonomous system. The same model is also utilized in [60]. These models are preliminary based on the MAPE-K model where the knowledge component is established by a security ontology based on [44].

Motivated by the fact that static security configuration cannot adapt to the dynamically changing requirements, a context-sensitive adaptive authentication approach has been proposed in [73]. The authors have extended the traditional three-factor authentication, i.e. what the user is, has, and knows, by adding situational a context. Two contexts, i.e. location and time, were used to evaluate the probability and authentication level required. Different sensed identity tokens and location information collected from the devices in the environment are fused together to assess the level of authentication required in various situations. The authors suggested a fusion algorithm that calculates a probability for a situation under observation, which is then used to determine the authentication level. The location information is obtained via a Context Management Framework (CMF) defined in [132]. Probabilities are calculated by a User Location Probabilistic Calculator (UPLC) that collects contextual data, i.e. location and timestamps, from the CMF and decides an authentication level. The adaptation control, at UPLC, seems to be implemented external to the object or application although, it is not entirely clear. This approach makes it suitable for resource-constrained objects as the required computations may be transferred to an external system with potential capacities. The authors have explicitly stated that they have utilized parameterization (or self-optimization) by offering different authentication levels. Other self-attributes are not supported. Like the other studied models, this approach is also limited to authentication related information and thus, may not be utilized in diverse threat scenarios.

Risk-based adaptive security management models have been proposed in [20, 116]. These models are based on the ISO/IEC 27005:2008 [9] risk management activities realized as the ISO Plan-Do-Check-Act (PDCA) model of the ISO/IEC 27001. The process name, PDCA, is not explicitly used in the new version, i.e. the ISO/IEC 27001:2013 [8]. These studies are generic

frameworks and highlight only a few methods and techniques, such as game theory, machine learning, context awareness, etc., that can be potentially be employed to achieve adaptive security in the IoT. The later study also details some security objectives, such as authentication and encryption, at the proposed communication layers as the core focus of adaptive security in IoT-eHealth scenarios.

Ashuman et al., in [118], have presented a software framework for autonomic security. At the top level, the framework realizes a control loop based on the control theory [36] by employing a sensor, analyzer, and responder model. It employs an event-driven communication model. The monitoring components act as event publishers to which the analyzers are subscribed. The analyzers assess the security context of the events, select a list of potential security configurations, suggest a single configuration having the lowest cost, and forward this decision to the responder as a high-level action, such as change encryption key. The responder maps this action to a particular security sub-system, e.g. authentication or cryptography, which validates and implements the decision. The framework supports self-configuration, self-optimization, and self-protection attributes. The suggested events are a part of a custom event schema and needed to be developed separately, which may require additional effort, time, and resources. A description of sample events are provided, but it is unclear how and to what extent they facilitate context-awareness in the system. Moreover, the details of the underlying analysis method, e.g. security context analysis, cost analysis, and the way optimum decisions are reached, are not provided. The framework reflects a reactive strategy towards the changes and lacks to provide any proactive approach.

A similar architectural view for self-managing security systems is proposed in [109]. It is fundamentally based on the GSpace model that the authors have previously proposed in [108]. The GSpace implements a distributed Shared Data Space (SDS). The SDS contains the necessary data for security services, core application operations, and communication. Data is stored in the form of tuples that are retrieved trough templates. A typical GSpace node has a GSpace Kernel, an application component, and a GSpace Proxy that connect the kernel with the application. The GSpace kernel has three major subsystems. The Operation Subsystem provides the core functionality and enables a node to participate in the GSpace architecture. A Context subsystem provides the context-related information and performs security analysis. Adaptation is achieved in the Security subsystem that implements Event-State-Condition-Action (ESCA) policies [110]. Communication among these subsystems is facilitated by an event bus. The study mentions that context is provided by various services, such as trust level, threat level, availability monitoring, memory monitoring, and bandwidth monitoring services. However, no further details are provided to elaborate

what type of information or methods are used to assess the corresponding contexts. Self-protection is enabled via the self-optimization and self-configuration mechanisms and can be activated at both the node and network level. Realizing the architecture's node level protection in the IoT might not be feasible due to the density of services offered. The network level strategy could be an option for IoT security. However, the authors lack to provide operational details of the monitoring and analysis components. Moreover, the proposal does not discuss how the non-security objectives or parameters that may be influenced by or influence the ESCA policy are addressed in the design, or in the adaptation process.

Tun et al., in [128] have proposed an adaptive information security (AIS) architecture that enables cloud services to respond dynamically to the changing user requirements. The AIS consist of two main elements: AIS Monitor and AIS Controls. The AIS monitor component resides on the user mobile and infers user location context and security requirements. The AIS Controls are implemented in the cloud (server) and adapt security strategies based on the inferred security requirements. The AIS Monitor logs user location, activities, and timestamps. Based on this information, a Requirements Monitor component in the AIS Monitor identifies probable security requirements. The authors asserted that more than one requirement could lead to conflicts, which may be dealt with. However, no conflicts resolution methods are specified. The AIS Monitor also includes an Application Adaptor component that will adapt the new changes it receives from the server component. On the server side, the AIS Controls consist of a Service Adaptation Engine that decides the adaptation based on a request tuple (Subject, Resource, Action, Requirement) from the client; a Policy Engine that defines a policy based on XACML schema [97] and enforces it for adaptation; and a Policy DB that stores the rule-based access control policy. The authors have presented a very abstract view of the architecture and do not provide any sufficient information to recognize how the contextual requirements or the analysis or adaptation processes are instrumented. The adaptation rendered is inflexible as only permitted and denied decisions are made. The approach only covers the access control objective and implements self-optimization in a restricted manner.

Salehie et al. in [113] proposed a requirements-driven adaptive security model. Requirements are captured as assets, threats and goals models that consist of the corresponding entities and their relationships. These requirements are used to build a casual network. The casual network is a Fuzzy Casual Network (FCN), based on Fuzzy Cognitive Maps [81] and Bayesian decision networks [76] to analyze the security changes and the impact of the potential analysis decisions. The model comprehensively addresses how threat can be analyzed using the FCN but and does not provide any information regarding asset monitoring and adaptation execution. Furthermore,

the model only emphasized how security mechanisms can be changed at the component level and thus only implements a self-configuration strategy.

Chapter 3

Research Methodology

This chapter provides a description of the research methodology employed in this thesis. Moreover, it explains the rationale for selecting the methodology used and highlights the particular research methods that have been utilized in the studies carried out in this research.

There are two types of research processes, namely inductive and deductive, that are used to develop knowledge while performing research activities [115]. Deductive approaches are used to infer knowledge from existing theories and are based on general ideas that are refined further towards a specific objective [115]. Inductive processes begin with a deeper understanding of a real-world problem and move towards the generalization of a research artifact [68, 115]. A complementary process to inductive research is an abductive approach. It is a research process that begins with rather a partial set of observations and move towards an artifact that is supported with a set of best possible decisions and explanation to address those observations [130].

As this thesis aims to develop and investigate the feasibility of a security adaptation approach based on the preliminary observation that a more intelligent and comprehensive solution is needed to address threats in the IoT dynamically, an abductive research approach was taken to initiate the research. Moreover, to establish rational scientific results and consistency, a research methodology was needed that would guide the artifact development and evaluation by allowing different methods and studies to be combined to address a particular problem. Hence, the Design Science Research (DSR) methodology [104] was adopted to steer this research as it is aligned with the criteria as mentioned above.

3.1 Design Science Research Methodology

DSR attempts to provide a platform to develop and investigate innovative artifacts and allow us to combine various scientific theories methods to inquire into a problem [82]. Artifacts can take different forms and can be methods, models, constructs, or instantiations [90]. A model represents a real-world problem and its solution, and utilizes constructs which may be the

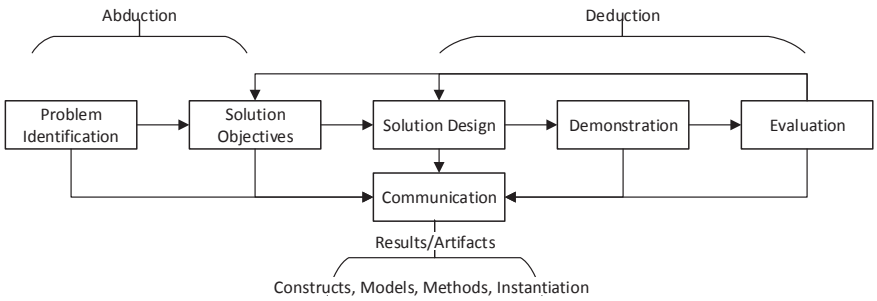


Figure 3.1: The Design Science Research Methodology (Compiled from [104])

desired attributes of a system, concepts or vocabulary used in the problem domain; a method provides a guideline to approach a particular problem, and an instantiation reflects an occurrence of the artifact [90].

This research employs the DSR process suggested by Peffers et al., in [104], as it provides a comprehensive guideline to perform scientific research. They have suggested this process after examining a comprehensive literature on design science in various fields of studies including engineering, computer sciences, and information systems. DSR starts with the problem identification and motivation that can be documented in a proposal. It is followed by inducing the solution's objectives from the problem identified and on the early study made. The objectives can be tentative design features expected in the desired solution. Based on this knowledge, an artifact is created, demonstrated and evaluated. These three activities are performed deductively. The results and knowledge obtained from these phases are documented and communicated to the relevant audience using different channels. The DSR process, as shown in Figure 3.1 is iteratively performed and can be initiated at any stage depending on the problem articulation [104].

3.1.1 DSRM Selection Rationale

A scientific research method is a set of activities that enable the researcher to perform systematic observations, experiments, formulations, evaluation and modifications of the hypothesis [13]. In general, there are two types of research methodologies, qualitative and quantitative. Quantitative methods are mostly used to observe and analyze natural phenomenon and utilize quantifiable data, i.e. numerical data. Qualitative studies collect data through observations, such as interviews or case studies, and are useful to investigate human or organizational behavior in a situation [98].

DSR Phase	R.Q	Artifact	Research Method	Article
Problem Identification	Main			Proposal
Solution Objectives	RQ 1	Constructs: ISRM Requirements, Trends & Gaps	Literature Review	Art.1
Solution Design	RQ 2	Models The EDAS Model, Adaptation Ontology	System Modeling, Survey	Art. 2, 3
Demonstration and Evaluation	RQ 3, 4	Instants IoT-eHealth Scenarios, EDAS Prototype Method: Trade-offs Evaluation Method	Case Study, Prototyping, Analytical Reasoning	Art. 3, 4, 5

Table 3.1: Research Methods and Artifacts w.r.t DSR Methodology

This research develops and evaluates an information system, i.e. an adaptive security system that addresses the technological as well as the behavioral aspects, such as user requirements and organizational policy, to protect the IoT ecosystem. As an applied research paradigm, information systems research encompasses studies related to diverse fields of knowledge including computer science, social and natural sciences, economics, and information technology [104]. DSR provides a guideline that enables us to combine various scientific theories concerning the mentioned knowledge circles and aims to design and evaluate artifacts that provide utility to human organizations [72]. As per the objective composition of this study, a qualitative-based DSR method was adapted to conduct this research work. An overview of the contributed artifacts, methods used, and associated articles and questions in alignment with the DSR methodology is depicted in Table 3.1 that is further elaborated in Section 3.1.2. The Case Study method in this work is synonymously used for a scenario-based approach that is utilized, in Articles 3–5, to illustrate and reflect on the different concepts and artifacts produced in this research.

3.1.2 An Overview of the Research Methods Used

The constructs, the ISRM requirements, trends, and gaps, were achieved in the first study, conducted by exploring RQ. 1, specified in Article 1 [28]. A literature review was performed to extract the knowledge related to the models and methods the are currently being practiced to address the IoT-eHealth's architecture, security, and ISRM essentials. Based on the architectural and security needs studied, the ISRM requirements were identified to be a set of necessary high-level attributes that should be considered by potential adaptive security solutions anticipated for the IoT security. Secondary sources, mainly the scientific work in the form of technical reports, books, and peer-review conferences and journal articles, were used to collect the data required for the study. The trends and gaps were identified by

reasoning about the comparison made. The ISRM requirements formed the basis of the proposed model.

System modeling was utilized in the second and third studies, motivated by RQ. 2, detailed in Articles 2 [27] and 3 [29], respectively. These articles preliminary addressed the (DSR) Models, i.e. the EDAS model and the Security Adaptation Ontology, proposed in this research. Article 2 conceptualizes a tentative system view of the suggested adaptive security model, EDAS, which was later detailed in Article 3. System modeling in the third study was supported by Literature Survey performed with secondary data sources (mentioned above) to highlight the various monitoring, analysis, and security adaptation methods that can be utilized in the proposed model to realize event-based security adaptation in the IoT. The Adaptation Ontology model was also defined using system modeling to visualize the construct of the various entities, members, attributes, and their relationships to facilitate context aware optimized adaptation. Article 3 also demonstrates adaptation in EDAS using a scenario-based illustration. Data required for the illustration was assumed. The EDAS model and the adaptation ontology is further explained in Article 4 [30].

The demonstration and evaluation of the proposed models were achieved in the fourth study detailed in Article 4 [30]. To achieve this objective planned in RQ. 3, a Prototype, as a DSR Instant, was developed by performing system simulation and emulation. Security events were generated using a simulated event framework developed for an Arduino-based eHealth sensing platform [2]. Open Source Security Information Management tool, OSSIM [11], was used to emulate the event-based risk monitoring and analysis components. The necessary modules, such as the monitoring plugins and analysis rules, for these components were exclusively developed for this prototyping. The ontology was created as a semantic web ontology resource in a Resource Description Format (RDF) [15]. The prototype implements a confidentiality-availability trade-off scenario developed to realize an adverse situation in the IoT-eHealth domain. Article 4, also evaluates the concept of EDAS as a security architecture using a qualitative comparative analysis. The comparison was made with a list of potential architectural styles upon which traditional security controls are typically based. Additionally, a list of security and architectural concerns was reasoned analytically to reflect on how EDAS can adequately provide a solution to face some of the challenges faced.

The feasibility of the proposed models was also validated using a scenario-based method that aimed to explore RQ. 4. This evaluation was reported in Article 5 [31]. This method was developed by employing a Unified Modeling Language (UML)-based system modeling. Multiple scenarios were developed to the UML schema utilization and system feasibility. Data was collected using the developed scenarios and was assumed for each of them.

Chapter 4

Research Articles Summary

This chapter provides a summary of the articles published during this thesis work. The summary is categorized into three sections, i.e. *requirements and gap analysis*, *solution design*, and *solution demonstration and feasibility*, reflecting the studies performed in accordance with the Design Science Research Methodology. Each section describes the work emphasizing a common theme, highlights the associated research question(s), methods used, and present and overview of the results obtained in the concerning articles.

4.1 Requirements and Gap Analysis

Identifying the requirements and analyzing trends in a topic under research are the primary elements to recognize in initiating a research activity. From a risk management perspective, they provide a better understanding of the architectural essentials in scope. This activity aimed to build an understanding of the topic and to identify research gaps in the related literature based on which further work could be developed. The research was initiated with analyzing architectural requirements with focus on information security risk management (ISRM) requirements that may be further explored to develop an effective adaptive risk management solution for a potential IoT-based service. Hence, the first research question was investigated, stated as follow:

Research Question-1: What are the key requirements for modeling automated risk management in an IoT-based service?

Relevant Article:
Article-1: WAQAS AMAN AND EINAR SNEKKENES. An Empirical Research on InfoSec Risk Management in IoT-based eHealth. In the third International Conference on Mobile Services, Resources, and Users (Mobility 2013), pages 99–107, 2013 [28]

To model an effective information security risk management (ISRM) solution for a given service, we need to understand its architectural requirements influencing or being influenced by it. These requirements were inves-

tigated in Article-1 [28] which is driven by the research problem that they were not comprehensively recognized in a unified manner while ISRM solutions are devised for IoT. Requirements were grouped into three sets: functional, security and privacy (S&P), and ISRM modeling essentials. The related literature was reviewed, and the various features offered were mapped to standard and proposed requirements to realize how and to what extent they address the highlighted requirements. Based on analytical reasoning, we concluded the current trends in modeling these requirements and identified the gaps indicating how some of them are disregarded or not appropriately addressed. The corresponding research approach is depicted in Figure 4.1.

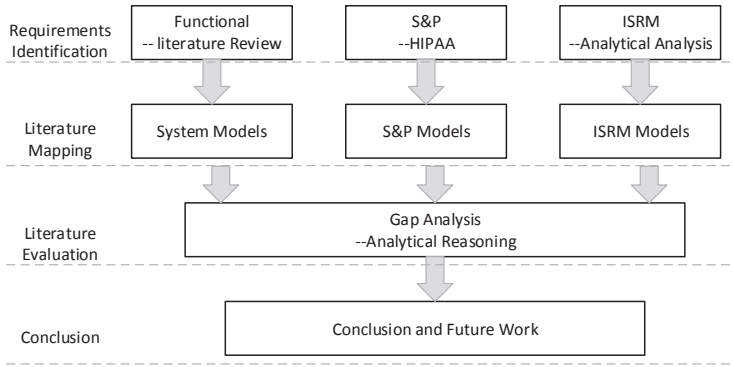


Figure 4.1: Research approach in used in Article-1 [28]

For IoT-eHealth to operate reliably, certain infrastructural essentials are needed to be satisfied. They can be identified as the different services and functionalities in the operational domain, therefore, define the scope of risk management in IoT-eHealth. During this study, it was realized that some of them have already been examined to an extent. Hence, among other work, the functional requirements identified in [103] are realized to be more inclusive and hence, were adopted as a benchmark for the review. However, one critical requirement, *mobility*, was lacking which we added to the list of functional requirements. Mobility is an essential functionality necessary for IoT-eHealth to be useful in mobility scenarios, such as outdoors activity monitoring and ambulatory needs. The functional requirements identified were: Collection and Processing, Real-time Delivery, Alarm Generation, Interpretation, Correlation, Data Request, Communication Interface, Actuation, and Mobility. Description of the individual requirement can be found in [28, 103].

Upon reviewing the related literature proposing various functional com-

ponents, we concluded that mobility and actuation were not appropriately addressed. On the majority, interpretation and correlation of vital signs are performed on the server-side. To reduce the number of requests made to the remote hospital site, these services should also be provided in the patient domain. This availability will make the system more scalable and would increase patient satisfaction. Furthermore, it was recommended that user-centric communication interfaces should be developed to make the system more adaptable to the user needs, which will also help in building technology usability.

S&P related literature was assessed against the networking and communication requirements as set by the U.S Health Insurance Portability and Accountability Act (HIPAA) [126]. They include data access, confidentiality, integrity, availability, alarm generation, identity management, privacy preservation, authentication, and event reporting. HIPAA requirements were selected based on the fact that they are comprehensive and are particularly suggested for the ICTs concerning healthcare. They also address administration and management elements, such as alarm generation, identity management, and event reporting, which are crucial to S&P as well. The literature assessment revealed that alarm generation, though a critical requirement, is predominantly disregarded. Multi-factor authentication was employed to exploit the already available determinants, such as vital signs and GPS location, instead of bringing new elements into the system which might have made it more complex. Availability modeling, to ensure continuous real-time service, was missing and needed to be explored. Largely, the literature focused on securing the domain external to the patient or external to the body area network (BAN), whereas protecting the personal area network (PAN) or the patient domain was overlooked.

A set of four requirements for ISRM modeling was identified. We emphasized it as a fitness criteria to be fulfilled by an ISRM model to address the dynamic nature of IoT-eHealth adequately. These requirements were:

- **Operational Nature:** It defines how often is the ISRM process executed. An *on-demand* operation implies a subjectively influenced ISRM execution performed as scheduled. Whereas, a *dynamic* ISRM process reflects a continuous risk management process conducted on real-time information. IoT-eHealth as a continuous real-time service necessitates a dynamic ISRM process and thus, needs a dynamic ISRM solution.

- **Context Awareness:** To reduced false alarms, threats should not be analyzed independently of their context. In real-world computing scenarios, a threat can be seen as a combination of different adverse events. Potential events need to be correlated in a broad context to ensure accurate analysis and response. Otherwise, the analysis may lead

to false alarms [119] and unnecessary reconfigurations. To avoid a possible adverse influence on a user-centric network driven by resource-constrained objects, such as the IoT-eHealth, a context-aware ISRM solution is necessitated to analyze the relationship of events in a possible threat situation.

- **Analysis Complexity:** To facilitate fast mitigation response in a real-time service and to ensure lightweight analysis to reduce the processing burden on the resource-limited *things* at the network edge, IoT-eHealth requires lightweight mechanisms. This requirement was later addressed as a lightweight architecture and discussed in Article-4 [30].

- **Self-Adaptation:** Managing security on each *thing* manually in an unattended environment and technology concentrated space, such as the IoT-eHealth, is a time and energy consuming task. It becomes more exhaustive when the number of monitored users is increased with each one using multiple *things*. Therefore, IoT-eHealth has to have self-adaptation properties. Self-adaptation refers to the effectual and autonomous reaction of a system to minimize the effect of a potential risk [20]. In ISRM terms, adaptation can be considered as an autonomic risk mitigation response to reduce a risk faced. Hence, the corresponding models or solutions should have the ability to react to an adverse situation and manage security autonomously to ensure adaptation

By analyzing the related ISRM literature with the mentioned criteria, it was realized that most of the models follow an on-demand process having qualitative analysis methods. Adaptation as a risk management strategy was lacking. Context awareness was either briefly discussed as a desirable attribute or was not addressed appropriately. Furthermore, the models investigated mainly focused on the analysis technique and rarely considered the influence of the solution on the monitored environment itself. It was concluded that the ISRM approaches reviewed are not feasible to be utilized in environments where continuous and dynamic monitoring is desirable, and corresponding services are driven by lightweight *things*.

4.2 Solution Design

As established in the Introduction chapter, conventional security controls, such as firewalls, antiviruses, access controls, intrusion detection systems (IDS), etc., are not suitable candidates for securing the IoT. Although they may provide some level of threat prevention or detection but in a long run and in a holistic context, they may show significant shortcomings. They might not withstand against the increasing threat sophistication in a diverse

technology outlook, such as the IoT. As standalone mechanisms, they tend to investigate only a particular type of information, e.g. a particular file type or network traffic, independent of their context and potential relationship, which may trigger false alarms. Moreover, they are not suitable for the resource-limited device, such as tiny sensors in the IoT ecosystem. Their risk mitigation strategies are inflexible and are executed by humans in the loop thus, they contradict the optimal and immediate mitigation response in a continuous real-time service. All these issues directed us to investigate the research and develop a new effective security solution for the IoT. This problem led to the examination of the second research question stated as follows:

Research Question-2: How can we develop an effective adaptive security solution for an IoT-based service?

Relevant Article(s):
Article-2: WAQAS AMAN. Modeling Adaptive Security in IoT Driven eHealth. In the Sixth Norsk informasjonssikkerhetskonferanse (NISK), 2013:61–69, 2014 [27].
Article 3: WAQAS AMAN AND EINAR SNEKKENES. Event Driven Adaptive Security in Internet of Things. In the Eighth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM 2014), pages 7–15, 2014 [29]

The IoT ISRM solution attributes identified in Article-1 [28], i.e. dynamic assessment, context awareness, a lightweight architecture, and adaptation, can pertinently overcome the lackings identified in the existing ISRM approaches and conventional security controls. Based on this criteria, in Article-2 [27], we proposed a conceptual adaptive security model as shown in the Figure 4.2. It was suggested that the *things*-generated events could be used to monitor the environmental influence or changes experienced by individual infrastructural elements (things). The captured influence can be analyzed by an Analyzer component with appropriate event correlation and context awareness capabilities to investigate potential threats in a holistic context. Furthermore, it was suggested that the Adapter component should not *just* adapt new security changes, but should decide an optimal mitigation action to ensure that factors affected by adaptation, e.g. user and QoS requirements, are appropriately addressed. The model proposition was built on the concept security event management (SEM) [88] which itself is an event-driven approach towards real-time security monitoring. The model extends the SEM concept further to accommodate adaptation.

In Article-3 [29], we extended the conceptual model and provided the details design specifications and methods of how adaptive security can be achieved in IoT by exploiting the event-driven architecture (EDA) concept [93]. The objective was to suggest an event-driven adaptive security model

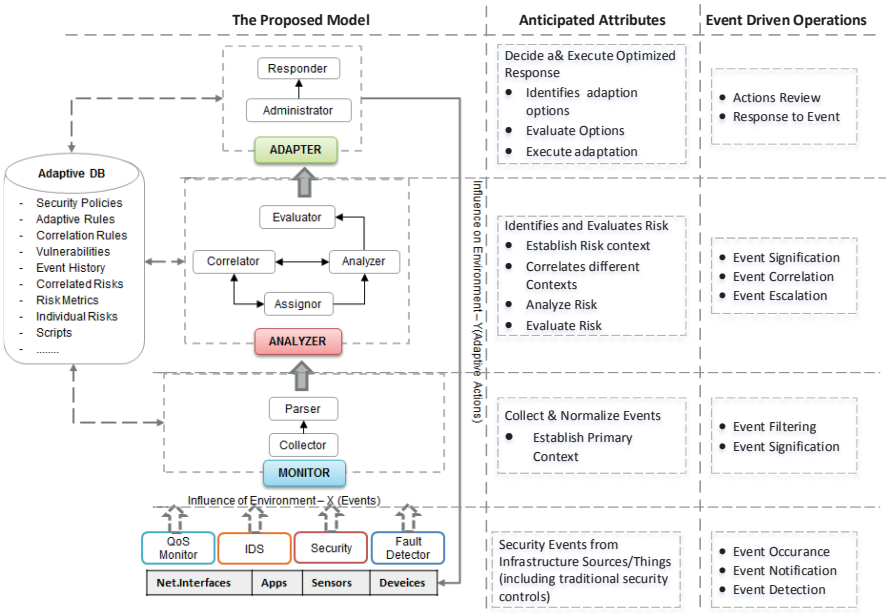


Figure 4.2: A conceptual view of the proposed adaptive security model (Enhanced)

(EDAS) that can observe, analyze and react to any security event. Thus, the model proposed had a holistic security view. The research approach used in this study is depicted in Figure 4.3. Article-3 was focused on two significant problems related to security adaptation modeling in IoT. They are highlighted as follows:

i) **Real-time Security Monitoring and Analysis**: How to monitor and collect security changes in real-time and analyzed the potential threat in a holistic context?

ii) **Security Adaptation:** How can the analyzed information be used to adapt security settings such that user and service preferences are appropriately addressed?

This first problem was approached by utilizing complex event processing (CEP) methods in an event-driven architecture style [93]. CEP aims to monitor, filter, classify, normalize and correlate system (thing) generated events in time and space to investigate a situation. As an example, the Open Source Security Information Management (OSSIM) [11] was used to illustrate how event monitoring and analysis can be employed in EDAS.

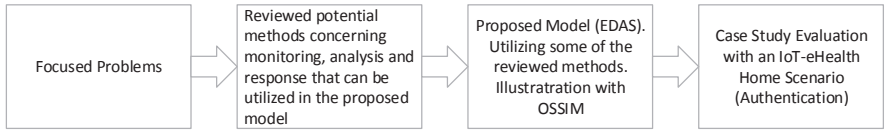


Figure 4.3: Article 3 Research Approach

Context awareness was demonstrated with an OSSIM correlation directive example to explain how individual events can be analyzed in a relationship, or in other words extended context, to investigate a potential threat scenario. It should be noticed that we intended to provide an open architectural approach where any appropriate event monitoring and analysis methods could be utilized.

The adaptation concern was addressed by proposing a novel security adaptation ontology. The ontology contained the necessary vocabulary, such as user and QoS requirements, device capabilities, security mechanisms, properties, and metrics, for deciding optimal mitigation response against the threat faced. The ontology, as shown in Figure 4.4 was projected as a runtime knowledge base that can be accessed in real-time during the decision process, shown in Figure 4.5. The proposed ontology ensures that the same knowledge, entities, members, properties, and relationships, modeled in the design-time can be readily used at run-time as it also addresses any changes that may arise during execution. Furthermore, an attack defense scenario in IoT-eHealth context was presented and detailed to comprehend the model illustration in a possible adverse scenario.

4.3 Solution Demonstration and Feasibility

Although, the ability of the proposed model to adapt to an adverse situation has been abstractly examined using a case study in Article-3 [29], its feasibility as a real-world artifact and its advantages over traditional security controls has yet to be investigated. To accomplish this purpose, the following question was examined.

Research Question-3: To what extent is the proposed model feasible in real-world scenarios?

Relevant Article:
Article-4: WAQAS AMAN AND EINAR SNEKKENES. EDAS: An Evaluation Prototype for Autonomic Event Driven Adaptive Security in the Internet of Things. Future Internet, 7(3). Pages 225–256, July 2015. [30].

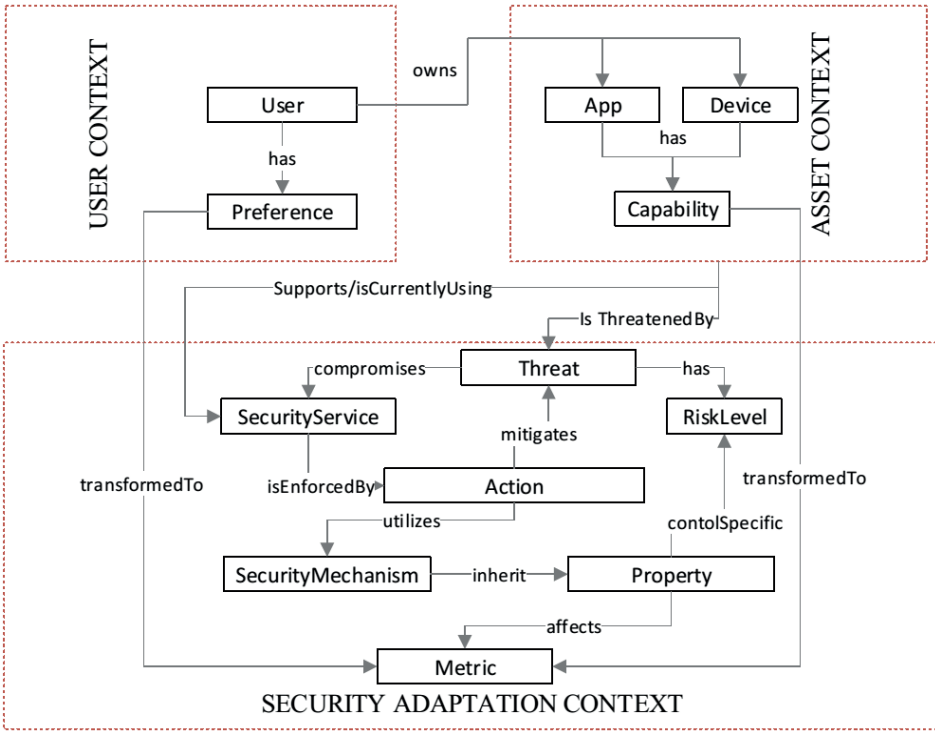


Figure 4.4: EDAS Adaptation Ontology

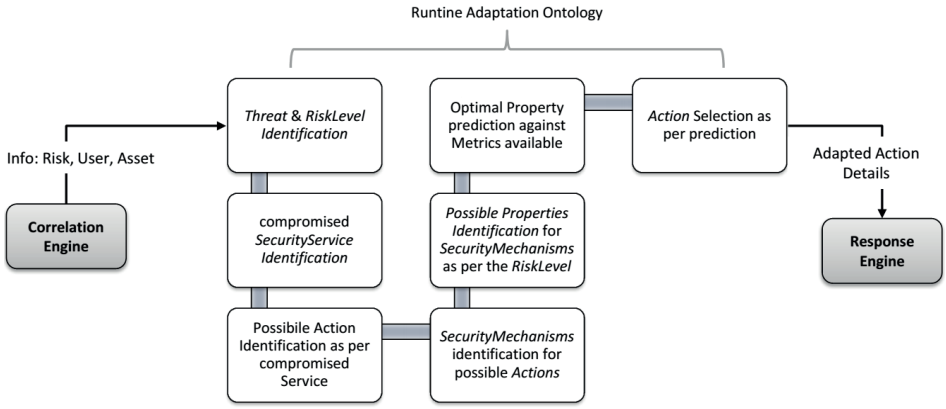


Figure 4.5: Adaptation Decision Process

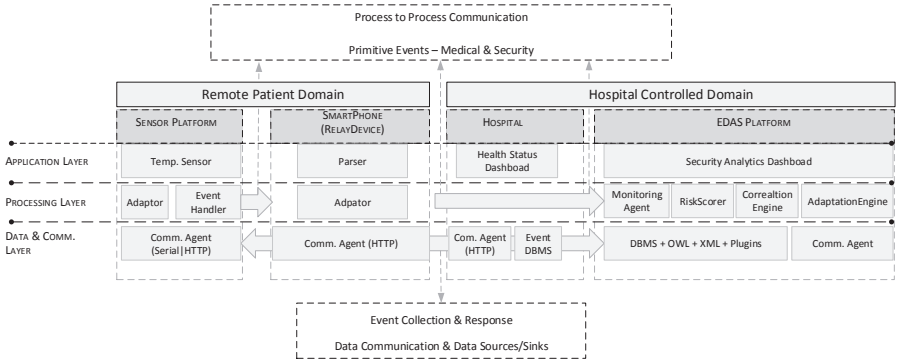


Figure 4.6: EDAS Prototype Layered Architecture

In Article-4 [30], the proposed model EDAS was demonstrated and evaluated as a real-world technical artifact,i.e. as a solution prototype. The research methods used in this study were system prototyping that realized a confidentiality-availability trade-off case study and analytical analysis to assess the overall feasibility of the proposed model as an architectural concept. The primary concerns investigated in this article were:

- What are the architectural strengths and limitations of EDAS as a real-world artifact?
- As a security architecture, how feasible is the EDAS concept and its adaptation loop if compared with the architectures related to traditional security controls?

This article explained the architectural schema of the EDAS prototype and provided its implemented specifications as a component-based model (CBA) [124]. A layered view of the prototype architecture can be seen in Figure 4.6. A description of the software and hardware tools used in the prototype development can be found in the related article. Technical details of the model's major component, i.e. the event sources (things), monitor, analyzer, and adapter, as well as their sub-components and corresponding input/output interfaces were also described.

Many significant insights have been developed from the prototyping activity. For instance, it was realized that event attributes can offer valuable and complete information for threat analysis. They contain the essential details about an event, such as the timestamps, type, severity levels, the generating process, user and device information, etc. These are critical attributes necessary for context-aware analysis as they provide primitive and complete information about the who, when, what, where, and how of an event

concerning a security change. Moreover, it was realized that the potential security events, if considered as the various steps towards a compromise situation, can be correlated with each other over a time span to assess whether they may yield to a probable risk. Thus, an event-driven analysis can be used to predict possible attacks and can take proactive measures. Therefore, the EDA approach enables us to combine attack prediction techniques to make the assessment of a risk context more reliable.

As an event-driven [93] and a component-based model [124], EDAS offers the flexibility that its components can be distributed over the network. In the context of healthcare, this flexibility enables the regional hospitals and their analysts (or analysis engines) to focus on their unique threat landscape and associated concerns.

Furthermore, computation resources needed for monitoring, analysis and adaptation are incorporated in a remote resource-full server. Therefore, *things* or event sources are only subjected to generation and communication of events. These tasks are already performed as built-in services in most things. Hence, the lightweight complexity attribute identified in the ISRM criteria discussed in Section 4.1 is addressed at the architectural level. The *things* have only to perform the local execution of the mitigation response sent by the remote Adaptation Engine, as a adaptation action request, to a Local Adopter which is merely a string parser and API caller.

A comprehensive analytical discussion was also presented on how EDAS can provide a dynamic real-time autonomous risk management platform that ensures a context aware as well as preference and capability-based security adaptation. This review validated that EDAS, as an architecture, fully complies with the ISRM criteria and requirements identified in the initial study in Article-1 [28]. Furthermore, Article-4 [30] provided a detailed discussion on various architectural aspects of EDAS, such as the architecture's self-protection; event communication; its adaptation scope; the way the model can be utilized in the protection against the advanced persistent threats (APT); and, how it can be employed in other prospective IoT system architectures. Exploring these aspects enable the readers to recognize and evaluate the concept of EDAS and its features inclusively in a big picture.

This study also investigate EDAS as an architectural solution and compares it with traditional engineered security controls in the IoT-eHealth context. The details can be found in [30] however, a tabular description is provided in Table 4.1 as an overview. The legends (++) implies that an attribute is positively qualified or supported, a (+) indicates partial qualification or support of the attribute suggesting that there is certain design dependency involved, and (-) indicates the attribute is not supported by a particular candidate.

	Attribute	EDAS	Host	Endpoint	Agent-Based	Centralized	Distributed
Execution	Interoperability	++	-	-	-	++	++
	Reliability	++	-	-	-	-	++
	Usability	++	+	+	-	-	-
	Latency	+	++	++	+	+	+
	Throughput	++	+	+	+	+	+
Security	Security	++	+	+	+	+	++
	Monitoring Scope	++	+	+	+	+	+
	Adaptability	++	-	-	-	-	-
	Threat Detection Accuracy	++	+	+	+	+	++
Design	Simplicity	+	++	++	+	+	+
	Extensibility	++	-	-	+	+	+
	Maintainability	+	++	+	+	+	+
Support	Supportability	++	++	++	++	++	++
	Testability	+	++	++	+	+	+

Table 4.1: EDAS *vs.* Conventional Security Artifacts

The feasibility of EDAS had been thoroughly validated from technical architecture viewpoint in Article-4 [30]. However, its evaluation was extended further to investigate how effectively it can manage the trade-offs in the concerning adaptation decisions. This issue was captured in the last research question as follows:
Research Question-4: How and to what extent does the adaptation loop of the proposed model add value to autonomic risk management in the IoT?

Relevant Article:
Article-5: WAQAS AMAN AND EINAR SNEKKENES. Managing Security Trade-offs in the Internet of Things Using Adaptive Security. In the Tenth International Conference for Internet Technology and Secured Transactions (ICITST-2015), London UK, 2015. Pages 362–368. [31]

A case study based approach was utilized in this study, which evaluated six eHealth-related scenarios. The scenarios were developed such that they reflected adverse situations in various operational contexts and any decision to mitigate the corresponding risk involved one or more trade-offs.

The related research question is approached categorically in two steps that are captured as sub-questions given below:

i) What typical trade-off situations exist in the IoT?

ii) To what extent does the EDAS adaptive security loop add value to autonomic risk management in the IoT?

These sub-questions were approached by proposing a scenario-based method towards adaptive security, as shown in the Figure 4.7. This method provides a two-phased approach and aims to identify, structure and eval-

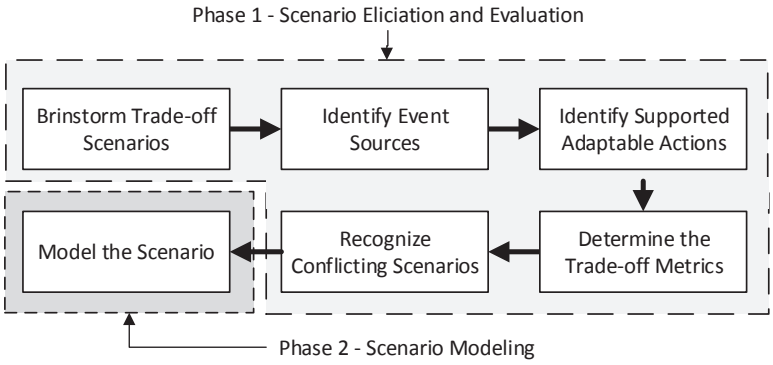


Figure 4.7: The proposed scenario-based method towards Adaptive Security

uate the knowledge necessary to realize potential security trade-off situations in EDAS. The trade-off represents a compromise between two or more runtime system attributes or metrics that may be negatively or positively influenced by realizing an adaptation response in a given scenario.

This study led to the conclusion that the proposed scenario-based method can be a useful tool for the analysts, architects, and developers to recognize and assess pre-development requirements, such as the trade-off metrics, conflicts and their resolution mechanisms, and development paradigms. It was also suggested that the method could be used as an implementation guideline for the developers as it highlights and structures the knowledge required to be implemented in EDAS related architectures.

Moreover, by using this method, it was made evident that EDAS evaluates all the potential trade-offs, including security and non-security runtime objectives, e.g. confidentiality, usability, availability, memory, and energy usage, while adaptation decisions are reviewed. These objectives are considered as utility metrics and are derived from the contextual requirements in the IoT ecosystem, i.e., the user preferences, QoS requirements, and the *things* available resources, in a particular operational context in which the scenario has occurred. For each possible adaptation response, EDAS evaluate these metrics and choose the one that has the maximum utility in a given threat scenario. Hence, this study validated the assumption that EDAS always takes an optimal trade-off decision while it adapts as it assesses all the potential runtime factors that can be affected by a decision. The validation was performed using arbitrary utilities for selected scenarios. A few concerns, such as utility assessment and architectural constraints were also explicated.

Chapter 5

Research Contributions

This chapters details the contributions of this thesis as the DSRM artifacts highlighted in Table 3.1(Chapter 1). They are grouped into models, instantiations, method, and constructs as defined in [90].The ISRM requirements identified and gap analysis extracted from the related literature are used as constructs to build the proposed models. A prototype, as the EDAS instance, and an evaluation method are then developed to assess the feasibility of the proposed models, i.e. the adaptive security model, EDAS, and the adaptation ontology model. The assessment involved multiple IoT-eHealth scenarios that are developed to support the demonstration and evaluation activities. The relationship between the contributions is depicted in Figure 5.1.

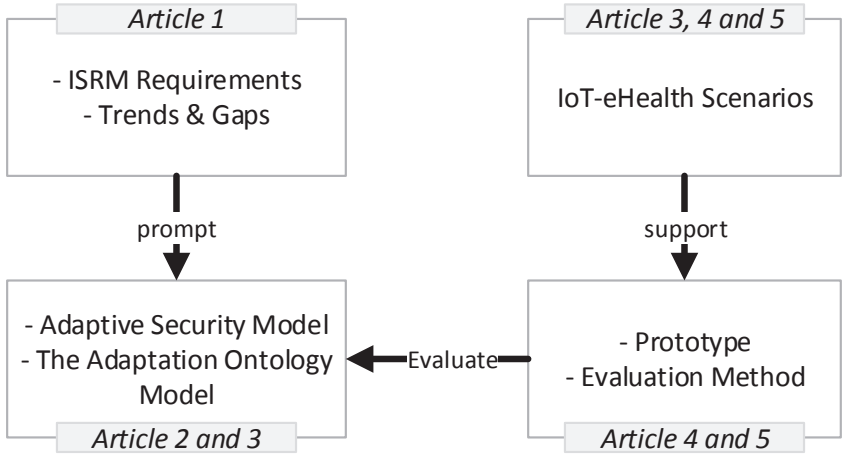


Figure 5.1: Contributions Relationship

5.1 Requirements, Trends and Gap Analysis

A comprehensive list of requirements is identified, collected and analyzed to understand the infrastructural, security and risk management essentials of IoT. These requirements have not been reviewed in a collective manner in the IoT before this study. From a risk management viewpoint, recognizing these essentials enable us to understand and evaluate: what infrastructural assets in the operational environment need protection?; what are the necessary security services to ensure protection?; and what may be the fundamental attributes of a potential adaptive risk management solution that can effectively oversee the security issues in the IoT at the infrastructural level? Furthermore, current literature is evaluated based on the identified requirements as a baseline. This assessment offers the opportunity to understand and analyze the current approaches, i.e. their strengths and weaknesses, and to what extent can they be a fit in the IoT. Moreover, the analyzed gaps enable the interested researchers to direct their study and focus on the investigation of related paradigms.

In the corresponding study Article-1 [28], made in the eHealth context, it was concluded that the identified services and functionalities are needed to be employed at the patient side to increase patient satisfaction and to encourage technology adoption. Also, it was recommended that security should not be limited to confidentiality and authentication. Related critical services, that directly or indirectly influence security, such as availability, alarm generation, and identity management, should also be investigated and modeled to ensure a more reliable and secure IoT-driven services. Furthermore, it was concluded that the studied ISRM models are not feasible for a real-time IoT-based service, as the majority of them are on-demand approaches and lack context-aware analysis, which does not qualify the major characteristics of IoT [123]. Such a complex environment with active participants necessitate a dynamic, context-aware, lightweight and adaptive ISRM solution to improvise effective protection.

5.2 The EDAS Model

The proposed event-driven adaptive security model, EDAS, aims to overcome the shortcomings of the existing traditional security and ISRM approaches. It provides an architectural approach towards dynamic, real-time, context-aware adaptive security solution that have a minimal processing burden on the resource-constrained *things* in the IoT. Existing architectural approaches, i.e, event-driven (EDA) [93] and component-based (CBA) [124] architectures, are utilized to address the adaptation problem. At the time of proposal, it was the first time that the feasibility of an EDA was evaluated thoroughly, particularly in IoT for adaptive security purposes. The EDA

approach has been studied in Security Event Monitoring (SEM) solutions [11, 32, 83], however, SEM has not been investigated further to the address autonomous adaptation, particularly in IoT scenarios.

EDAS suggests a risk-driven security assessment approach. The EDA mechanism provides essential knowledge for the risk to be determined, which includes the assets, critical events, and their impact in time and space. A comprehensive list of prospective event monitoring and analysis methods have been identified in Article-3 [29] which can be utilized in EDAS. Moreover, all the major pre-development [31] and implementation [30] details have been explicated.

Below are some of the key facts that reflect how EDAS, as an event-driven and component model, offers a more reliable and practical security architecture for IoT-based services than its counterparts:

- EDAS fulfills the adaptation system attributes recommended by IBM [62] and [59]. In information security perspective, EDAS provides/can provide:
  - A self-configuring platform as it can change a security component on the fly. For instance, autonomously changing an encryption or routing algorithms to another.
  - A self-optimizing tool as it can tune current or adapt new parameters of a security component, e.g. adapting key lengths or password types.
  - A self-healing mechanism as it monitors and reacts to disruption related events by considering them as changes affecting availability.
  - A self-protection and self-aware mechanism if its *EDAS Platform* is considered a critical event source. By doing so, it can monitor itself thus, becomes self-aware, and can respond to any adverse events within itself, therefore, ensures self-protection. This notion is further detailed in Article-4 [30].
  - A context-aware analysis by observing and assessing all the potential security context offered by thing-generated primitive events. The event correlation ensures risk analysis in a more extended context. Furthermore, it offers a context-aware optimal adaptation as all the necessary contextual requirements, such as operational environment, user preferences, QoS requirements and thing resources are evaluated.
- EDAS provides a holistic adaptive security approach:
  - Conventional security controls and the studied ISRM approaches implement fixed and static mitigation approaches. They implement a particular pre-defined security mechanisms even if other

choices exist, and react to a risk faced manually with an only focus of asset protection without assessing other runtime objectives. EDAS takes a dynamic trade-off adaptation action, therefore, addresses risk mitigation in an efficient manner.

- Unlike conventional controls that focus on a particular security services, such as confidentiality, etc., EDAS aims to observe *any* security-related changes whether they correspond to confidentiality, intrusions, errors, or resource consumption, etc.
- Most of the traditional controls emphasize inbound traffic, such as Firewalls or intrusion prevention system (IPS), and thus, lack to analyze outbound communications and activities [45]. EDAS focuses on the basic unit of change, event, irrespective of the direction and activity type. Hence, it encompasses a broader threat landscape.
- Fundamentally, EDAS implements detective and reactive approaches. However, if events are characterized as distinct steps towards a particular threat context, then with appropriate analysis techniques, such as behavioral and statistical approaches, it can be utilized to predict threats and can respond to them proactively.

- As IoT is driven by lightweight sensory *things*, embedding traditional controls, like anti-malware, etc., locally in a *thing* might not be feasible. EDAS utilizes minimal resources at a *thing* level. The monitoring, analysis, and adaptation processes are suggested to be performed on an enterprise server. The event sources (things) are only required to communicate the events they generate and adopt the adaptation instruction it receives. These are trivial tasks and require only nominal resources.

- From a system architecture point of view, specifically as an event-driven and component model, EDAS inherits loose coupling among the components which ensures reusability, extensibility and flexibility in development and deployment. EDAS automated adaptation loop enables prompt response to the risk faced hence, increases the overall throughput. Furthermore, to address the IoT heterogeneity, the architecture is made independent of any hardware or software specifications used in the monitored environment. Its only concern is the *thing*-generated event. Thus irrespective of the underlying platform, as long as a particular *thing* can generate and communicate events, it can be managed by the EDAS. This fact also lead us to the hypothesis that EDAS can be utilized in similar IoT-based smart ecosystems, such as Smart Grids, Smart Cities, and Vehicular networks, etc., where *things* can trigger, convey, and react to events about the changes they

experience. These tasks are commonly employed as an out-of-the-box event logging utility in almost all objects and is used for troubleshooting purposes. However, the conjecture of EDAS application has yet to be evaluated with appropriate domain specific information.

5.3 The Runtime Security Adaptation Ontology

The presence of heterogeneous and mobile elements make the IoT environment considerably volatile, frequently changing, and complex. Therefore, it is challenging to capture the runtime security requirements in such a dynamic and multivariate environment while the security system is being designed [58]. The proposed runtime ontology fills this gap by exploiting the requirements analysis and design-time knowledge during execution. It organizes and relates the contextual knowledge among *things*, users and security such that it can be accessed and modified during execution. Therefore, it reduces the time and effort needed in a design to product transition, as it provides a common platform for designers and developers to address a problem at the same level of development. Furthermore, this ontological approach provides a basis to model the semantics of the heterogeneous objects in universal format [134] thus, appropriately suits the IoT ecosystems.

Consisting of three knowledge domains, i.e. user, device (or thing), and security, the proposed ontology contains the concerning vocabulary that addresses all the potential contextual information and requirements. The ontology further realizes the corresponding domain requirements as the concerning utility metrics that enable the system to always decide an optimum trade-off decision.

5.4 Trade-offs Evaluation Method

The proposed trade-off evaluation method provides a tool for the analysts and developers to assess the various trade-offs involved in an adverse situation. The method utilizes a scenario-based approach to recognize, assess, and realize the trade-offs and related knowledge in an event-driven adaptive security system. It serves two objectives. Firstly, it identifies the potential trade-offs, conflicting scenarios, critical assets and corresponding events, and the supported adaptable actions in a particular threat scenario. Hence, it gathers and evaluates the knowledge necessary for system adaptation. Secondly, it emphasizes on the realization of the knowledge extracted by using the system (EDAS) schema, and highlights the relationship of individual elements in the knowledge domain, which further evaluates the knowledge.

The method can be used as pre-development platform where both the analyst and developers can evaluate adaptation essentials. Moreover, the knowledge realization provides a guideline for the developers as it can it

identify the different calls and procedures required during risk analysis, which further enables the developers to determine and utilize the appropriate development mechanisms and resources. The typical trade-offs that have been evaluated using this method includes, confidentiality, authentication, up-time (availability) efficiency, resource usage (energy and memory), accessibility, memorability, ease of use, and distress. Assessing these trade-offs provides an evidence that the proposed method is able to identify and evaluate security as well as usability and QoS related concerns in adaptation and thus, further evaluates the feasibility of EDAS.

Existing scenario-based evaluation approaches, for instance, [25, 77, 78, 85], either evaluates on quality attributes in software or system architectures, focus only on security attributes, or assess different approaches. The proposed method aims to address all the potential metrics in a trade-off whether they are security, user, performance or other architectural concerns. Moreover, the concerning knowledge is evaluated in a runtime perspective, i.e. what adaptation knowledge exist and how can it be realized during adaptation. Thus, the proposed method adds value to the realization and development of the artifact.

5.5 The EDAS Prototype

The developed prototype provides a proof of concept and explains the technical specifications of the EDAS model. The system design is built as a component-based model [124] that enables the proposed design to provide simplicity, extendability, independence, and reusability in the architectural components [95]. The prototype design details the design specifications of the major components, i.e. event source, monitor, analyzer, and adapter. These components were further decomposed into functional components with essential input and output interfaces to address the necessary objectives. Furthermore, a layered view consisting of application, processing, and data communication layers is also provided that gives a conceptual view of the system and enable the reader to understand the design at the infrastructural level. The Alienvault's Open Source Security Information Management (OSSIM) [11] was employed as an event collection and analysis platform in the IoT-eHealth context. However, the event source (a body temperature sensor), its local modules, its monitors (plugins), the event correlation rules and directives, and the adaptation modules were all custom developments tested in a developed adverse confidentiality-availability trade-off scenario.

5.6 The Case Study and Scenarios

A case study-based approach is used to evaluate and realize the feasibility of the proposed model. The case study reflected an IoT-eHealth arrangement. It describes a setting where a patient, utilizing various medical related sensors and equipment, is monitored from a remote hospital location. Numerous scenarios in different operational contexts,e.g. in home, hospital, and outdoor environments, were developed that described various accidental and intentional adverse security situations. These scenarios describe the utilization of various *things* and how can they become a target in the IoT, the typical trade-offs involved in potential adaptation decision, and provides a basis for a step-by-step system demonstration.Table 5.1 highlights these scenarios, their purpose, and the articles in which they appeared.

Scenario Description	Purpose	Appeared in
Repeated Password attempts	Adaptation illustration and Trade-off assessment	Article 3 and 5
Charging-Discharging of a sensor battery	Prototype implementation	Article 4
Resource optimization during mobility	System Modeling, Trade-off assessment	Article 5
Max. Confidentiality in Possible Intrusions	System Modeling, Trade-off assessment	
Handling a thing Compromise	Trade-off ilustration	
Physician Account Compromise	Trade-off Illustration	
Service Unavailability	Trade-off assessment	

Table 5.1: IoT-eHealth Scenarios

Chapter 6

Limitations and Future Work

The theme of the proposed model, EDAS, is to observe, analyze and react to security changes at the infrastructural level. With such a broad scope, it was challenging to propose and investigate the specific methods and tool required for all the major components in this thesis tenure. This thesis only evaluates the feasibility of adaptive security from an operational architecture viewpoint in a particular application area, i.e. eHealth, and emphasize mainly the adaptation part as it was not properly addressed in the IoT.

Therefore, EDAS, as an early stage development, needs to be further investigated to explore some of its limitations and concerns. This chapter details a few major issues, such as architectural constraints and operational concerns, that are still to be explored for EDAS. The following sections describe these concerns and provide initial observations that can be further pursued to make EDAS a more reliable and robust solution for IoT security.

6.1 Architectural Dependencies

EDAS is based on the assumption that every monitored thing in the IoT-ecosystem has already an event framework. The event framework is the primary pre-requisite in EDAS-based security adaptation. This facility is usually built in a thing as a troubleshooting, logging, and debugging tool. For example, being critical to human life, it is a vital component in body sensors to keep track of their operational reliability. However, in some cases, this utility might not be a design priority. Also, some objects are shipped with a single security mechanism, e.g. the encryption algorithm AES-256. In these situations, adaptation may not be practical as they cannot be monitored due to the absence of the event framework or are not adapted appropriately because of the inflexibility of the security component. To overcome these challenges, various middleware approaches can be added to the EDAS design. They can offer context-awareness and security components as services to the monitored things having the mentioned lacking.

6.2 Extending the Case Study

As an event-based solution, EDAS only focuses on the security-related events irrespective of underlying software or hardware specifications of a monitored thing. This notion led to the hypothesis that EDAS can be utilized in any IoT-based environment where things can generate and communicate events when they experience a related change. However, to validate the mentioned hypothesis, EDAS has to be further investigated in other IoT-driven systems. Due to the variations in the service architectures of some IoT-based systems, such as the IoT-based smart grids, the EDAS model may further be customized to meet targeted service or architectural requirements. Hence, the test scenarios have to be further extended, as for now only eHealth scenarios were evaluated. Such attempts will make the proposed model an inclusive solution for the IoT, particularly in the Internet of Service scenarios.

6.3 Scalability

EDAS provides a component-based architecture (CBA) which aims to decompose the functionality so that simplicity can be achieved. However, IoT is a progressive concept and envisions to accommodate future technologies. Therefore, the more things we have in the monitored environment, the more events will be generated, and therefore, more components and resources will be required. This may result in a scalability issue which is not addressed in this thesis. Being based on CBA, EDAS inherently solve this problem by offering the distribution of the processes and services as mentioned in Article [30]. However, the extent to which it is scalable still requires a thorough study. A good starting point to approach a potential solution will be to recognize: the event generation in normal and peak timings, how are the events communicated, i.e. best effort, guaranteed or secure delivery, event retention time, how frequently do the EDAS major components interact with each other. Understanding these and related concerns may provide vital information based on which the scalability concerns can be addressed, and will make EDAS a more reliable model from an architectural viewpoint.

6.4 Security Metrics

This thesis utilizes security metrics in two stages. First, during risk analysis when the secondary event context is investigated for the possible threats and corresponding risk exposure is quantified. Risk quantification is based on risk metrics, such as the event source, the event type, its importance, time, and frequency. Secondly, when adaptation actions are evaluated which was the prime focus of this research. Metrics at this stage are referred to as the

utility metrics and are derived from contextual requirements. They reflect the goodness of an action in a particular scenario. Although an integral part of the security process, this thesis has only exploited security metrics in adaptation illustration, as modeling new ones was not the thesis main objective. However, it is suggested that these metrics, particularly the utility metrics, need to be rigorously decomposed and classified to capture the actual requirements and to approach adaptation more realistically. Furthermore, the metric assessment in adaptation requires a more validated theory as the effectiveness of the proposed utility-based metric assessment is not properly evaluated.

Bibliography

[1] Apache jena-an open source java framework for building semantic web and linked data applications. Available from: https://jena.apache.org/index.html.

[2] The cooking hacks e-health sensor platform. Last accessed on 25-Apr-2016. Available from: http://tinyurl.com/eHealth-Platform.

[3] The dmtfs common information model (cim). Available from: http://www.dmtf.org/standards/cim.

[4] The epcglobal architecture framework, epcglobal final version 1.4 approved 15 december 2010. Tech. rep. Available from: http://tinyurl.com/nzszyvz.

[5] Fact++-a description logic reasoner. Available from: https://code.google.com/p/factplusplus/.

[6] Hermit owl reasoner. Available from: http://hermit-reasoner.com/.

[7] Iso 19115-1:2014. geographic information metadata schema. Available from: http://tinyurl.com/n945fdy.

[8] ISO/IEC 27001:2013. Information security management systems Requirements.

[9] ISO/IEC 27005:2008. Information Security Risk Management Guideline. This standard has been revised as ISO/IEC 27005:2011.

[10] Jfact description logic reasoner. Available from: http://jfact.sourceforge.net/.

[11] Ossim: the open source siem. Last access date: 19 Nov 2015. Available from: https://www.alienvault.com/products/ossim.

[12] Owasp internet of things top 10 project. Last accessed on: 24-Feb-2015. Available from: https://www.owasp.org/index.php/OWASP_Internet_of_Things_Top_Ten_Project.

[13] Oxford english dictionary. scientific method. Oxford English Dictionary.

[14] Protégé ontology editor. Available from: http://protege.stanford.edu/.

[15] Resource description framework (rdf). Available from: http://www.w3.org/2001/sw/wiki/RDF.

[16] Sensor model language (sensorml). Available from: http://www.opengeospatial.org/standards/sensorml.

[17] Sparql protocol and rdf query language (sparql). Available from: http://www.w3.org/TR/sparql11-query/.

[18] Web ontology language (owl). Available from: http://www.w3.org/2001/sw/wiki/OWL.

[19] Internet of things in 2020. a roadmap for the future. Tech. rep., The European Technology Platform on Smart Systems Integration (EPoSS), 2008.

[20] ABIE, H., AND BALASINGHAM, I. Risk-based adaptive security for smart iot in ehealth. In *Proceedings of the 7th International Conference on Body Area Networks* (2012), ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), pp. 269–275.

[21] ABIE, H., SAVOLA, R. M., BIGHAM, J., DATTANI, I., ROTONDI, D., AND DA BORMIDA, G. Self-healing and secure adaptive messaging middleware for business-critical systems. *International Journal On Advances in Security 3* (2010), 34–51.

[22] ABOWD, G. D., DEY, A. K., BROWN, P. J., DAVIES, N., SMITH, M., AND STEGGLES, P. Towards a better understanding of context and context-awareness. In *Handheld and ubiquitous computing* (1999), Springer, pp. 304–307.

[23] ABOWD, G. D., AND MYNATT, E. D. Charting past, present, and future research in ubiquitous computing. *ACM Transactions on Computer-Human Interaction (TOCHI) 7*, 1 (2000), 29–58.

[24] AHN, S., AND KIM, D. Proactive context-aware sensor networks. In *Wireless Sensor Networks*, vol. 3868 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2006, pp. 38–53.

[25] ALKUSSAYER, A., AND ALLEN, W. H. A scenario-based framework for the security evaluation of software architecture. In *Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on* (2010), vol. 5, IEEE, pp. 687–695.

[26] ALPCAN, T., AND BAŞAR, T. *Network security: A decision and game-theoretic approach*. Cambridge University Press, 2010.

[27] AMAN, W. Modeling Adaptive Security in IoT Driven eHealth. *The 6th Norsk informasjonssikkerhetskonferanse (NISK) 2013* (2014), 61–69.

[28] AMAN, W., AND SNEKKENES, E. An Empirical Research on InfoSec Risk Management in IoT-based eHealth. In *The Third International Conference on Mobile Services, Resources, and Users (Mobility 2013)* (2013), pp. 99–107.

[29] AMAN, W., AND SNEKKENES, E. Event Driven Adaptive Security in Internet of Things. In *UBICOMM 2014, The Eighth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies* (2014), pp. 7–15.

[30] AMAN, W., AND SNEKKENES, E. EDAS: An Evaluation Prototype for Autonomic Event Driven Adaptive Security in the Internet of Things. *Future Internet 7*, 3 (July 2015), 225–256.

[31] AMAN, W., AND SNEKKENES, E. Managing Security Trade-offs in the Internet of Things Using Adaptive Security. In *The 10th International Conference for Internet Technology and Secured Transactions (ICITST-2015), London UK* (2015). ACCEPTED.

[32] ARCSIGHT. Hp arcsight enterprise security: Product brief, 2012. [Online Accessed on 9 October 2012]. Available from: `http://www.hpenterprisesecurity.com/collateral/briefs/product/HPEnterpriseSecurity_ProductBrief_HPArcSightESM.pdf`.

[33] ASHBY, W. R. *An introduction to cybernetics*. Chapman & Hall Ltd., 1956.

[34] ASHTON, K. Rfid and the inclusive model for the internet of things. `http://www.grifs-project.eu/data/File/CASAGRASFinalReport(2).pdf`. The CASAGRAS Project Final Report.

[35] ASHTON, K. That internet of things thing. *RFiD Journal 22*, 7 (2009), 97–114.

[36] ÅSTRÖM, K. J., AND WITTENMARK, B. *Adaptive control*. Courier Corporation, 2013.

[37] ATZORI, L., IERA, A., AND MORABITO, G. The internet of things: A survey. *Computer Networks 54*, 15 (2010), 2787 – 2805.

[38] AVANCHA, S., PATEL, C., AND JOSHI, A. Ontology-driven adaptive sensor networks. In *MobiQuitous* (2004), vol. 4, pp. 194–202.

[39] AVIŽIENIS, A., LAPRIE, J.-C., RANDELL, B., AND LANDWEHR, C. Basic concepts and taxonomy of dependable and secure computing. *Dependable and Secure Computing, IEEE Transactions on 1*, 1 (2004), 11–33.

[40] BARNAGHI, P., PRESSER, M., AND MOESSNER, K. Publishing linked sensor data. In *CEUR Workshop Proceedings: Proceedings of the 3rd International Workshop on Semantic Sensor Networks (SSN), Organised in conjunction with the International Semantic Web Conference* (2010), vol. 668.

[41] BONACI, T., AND BUSHNELL, L. Node capture games: a game theoretic approach to modeling and mitigating node capture attacks. In *Decision and Game Theory for Security*. Springer, 2011, pp. 44–55.

[42] BROLL, G., RUKZIO, E., PAOLUCCI, M., WAGNER, M., SCHMIDT, A., AND HUSSMANN, H. Perci: Pervasive service interaction with the internet of things. *Internet Computing, IEEE 13*, 6 (2009), 74–81.

[43] BUCKL, C., SOMMER, S., SCHOLZ, A., KNOLL, A., KEMPER, A., HEUER, J., AND SCHMITT, A. Services to the field: An approach for resource constrained sensor/actor networks. In *Advanced Information Networking and Applications Workshops, 2009. WAINA'09. International Conference on* (2009), IEEE, pp. 476–481.

[44] CHEN, H., FININ, T., AND JOSHI, A. The soupa ontology for pervasive computing. In *Ontologies for agents: Theory and experiences*. Springer, 2005, pp. 233–258.

[45] COLE, E. Advanced persistent threat (apt) and insider threat, October 2012. Last accessed on 24 Apr 2015. Available from: `http://goo.gl/ZdOWvO`.

[46] CROLL, P. R., AND CROLL, J. Investigating risk exposure in e-health systems. *International Journal of Medical Informatics 76(5-6)* (2006), 460–465.

[47] DADA, A., AND THIESSE, F. Sensor applications in the supply chain: the example of quality-based issuing of perishables. In *The Internet of Things*. Springer, 2008, pp. 140–154.

[48] DENKER, G., KAGAL, L., AND FININ, T. Security in the semantic web using owl. *Information Security Technical Report 10*, 1 (2005), 51–58.

[49] DEY, S., SAMPALLI, S., AND YE, Q. A light-weight authentication scheme based on message digest and location for mobile cloud computing. In *Performance Computing and Communications Conference (IPCCC), 2014 IEEE International* (2014), IEEE, pp. 1–2.

[50] DEY, S., SAMPALLI, S., AND YE, Q. A context-adaptive security framework for mobile cloud computing. In *2015 11th International Conference on Mobile Ad-hoc and Sensor Networks (MSN)* (2015), IEEE, pp. 89–95.

[51] DOBSON, G., AND SAWYER, P. Revisiting ontology-based requirements engineering in the age of the semantic web. In *Proceedings of the International Seminar on Dependable Requirements Engineering of Computerised Systems at NPPs* (2006), pp. 27–29.

[52] DON, S., CHOI, E., AND MIN, D. A situation aware framework for activity based risk analysis of patient monitoring system. In *Awareness Science and Technology (iCAST), 2011 3rd International Conference on* (2011), pp. 15–19.

[53] DONNER, M. Toward a security ontology. *IEEE Security & Privacy 1*, 3 (2003), 0006–7.

[54] EID, M., LISCANO, R., AND EL SADDIK, A. A universal ontology for sensor networks data. In *Computational Intelligence for Measurement Systems and Applications, 2007. CIMSA 2007. IEEE International Conference on* (2007), IEEE, pp. 59–62.

[55] EKELHART, A., FENZ, S., KLEMEN, M., AND WEIPPL, E. Security ontologies: Improving quantitative risk analysis. In *System Sciences, 2007. HICSS 2007. 40th Annual Hawaii International Conference on* (2007), IEEE, pp. 156a–156a.

[56] EMC, AND CORPORATION, I. D. The digital universe of opportunities: Rich data and the increasing value of the internet of things. `http://www.emc.com/leadership/digital-universe/2014iview/internet-of-things.htm`, April 2014. Last accessed on 21 Dec 2014.

[57] EVANS, D. The internet of things:how the next evolution of the internet is changing everything. Tech. rep., CISCO, April 2011. Last accessed on: 31 Aug 2015. Available from: `http://bit.ly/1Inzh2Q`.

[58] EVESTI, A., AND OVASKA, E. Ontology-based security adaptation at run-time. In *Self-Adaptive and Self-Organizing Systems (SASO), 2010 4th IEEE International Conference on* (2010), IEEE, pp. 204–212.

[59] EVESTI, A., AND OVASKA, E. Comparison of adaptive information security approaches. *ISRN Artificial Intelligence 2013* (2013).

[60] EVESTI, A., AND PANTSAR-SYVÄNIEMI, S. Towards micro architecture for security adaptation. In *Proceedings of the Fourth European Conference on Software Architecture: Companion Volume* (New York, NY, USA, 2010), ECSA '10, ACM, pp. 181–188.

[61] EVESTI, A., SAVOLA, R., OVASKA, E., AND KUUSIJÄRVI, J. The design, instantiation, and usage of information security measuring ontology.

[62] GANEK, A. G., AND CORBI, T. A. The dawning of the autonomic computing era. *IBM systems Journal 42*, 1 (2003), 5–18.

[63] GENEIATAKIS, D., AND LAMBRINOUDAKIS, C. An ontology description for sip security flaws. *Computer Communications 30*, 6 (2007), 1367–1374.

[64] GIUSTO, D., IERA, A., MORABITO, G., AND ATZORI, L. *The internet of things: 20th Tyrrhenian workshop on digital communications*. Springer Science & Business Media, 2010. ISBN:978-1-4419-1674-7.

[65] GRUNINGER, M., AND LEE, J. Ontology applications and design. *Commun. ACM 45*, 2 (Feb. 2002), 39–41. Available from: `http://doi.acm.org/10.1145/503124.503146`.

[66] GUBBI, J., BUYYA, R., MARUSIC, S., AND PALANISWAMI, M. Internet of things (iot): A vision, architectural elements, and future directions. *Future Generation Computer Systems 29*, 7 (2013), 1645–1660.

[67] HAMDI, M., AND ABIE, H. Game-based adaptive security in the internet of things for ehealth. In *Communications (ICC), 2014 IEEE International Conference on* (2014), IEEE, pp. 920–925.

[68] HARSHORNE, C., AND WEISS, P. Pierce, c. s: Collected papers. Harvard University Press, 1931 - 1935.

[69] HEER, T., GARCIA-MORCHON, O., HUMMEN, R., KEOH, S., KUMAR, S., AND WEHRLE, K. Security challenges in the ip-based internet of things. *Wireless Personal Communications 61*, 3 (2011), 527–542. Available from: `http://dx.doi.org/10.1007/s11277-011-0385-5`.

[70] HENRICKSEN, K. *A Framework for Context-aware Pervasive Computing Applications*. Ph.D. thesis, The School of Information Technology and Electrical Engineering. The University of Queensland, 2003.

[71] HERZOG, A., SHAHMEHRI, N., AND DUMA, C. An ontology of information security. *International Journal of Information Security and Privacy 1*, 4 (2007), 1–23.

[72] HEVNER, A., AND CHATTERJEE, S. *Design research in information systems: theory and practice*, vol. 22. Springer Science & Business Media, 2010.

[73] HULSEBOSCH, R., BARGH, M. S., LENZINI, G., EBBEN, P., AND IACOB, S. M. Context sensitive adaptive authentication. In *Smart Sensing and Context*. Springer, 2007, pp. 93–109.

[74] IDC ITALIA S.R.L AND TXT E-SOLUTIONS S.P.A. Definition of a research and innovation policy leveraging cloud computing and iot combination. Tech. rep., European Commission DG Communications Networks, Content & Technology, May 2015.

[75] ILIC, A., STAAKE, T., AND FLEISCH, E. Using sensor information to reduce the carbon footprint of perishable goods. *IEEE Pervasive Computing*, 1 (2009), 22–29.

[76] JENSEN, F. V., AND NIELSEN, T. D. Bayesian networks and decision graphs. *Bayesian Networks and Decision Graphs: February 8, 2007, Information Science and Statistics. ISBN 978-0-387-68281-5. Springer New York, 2007 1* (2007).

[77] KAZMAN, R., BASS, L., WEBB, M., AND ABOWD, G. Saam: A method for analyzing the properties of software architectures. In *Proceedings of the 16th international conference on Software engineering* (1994), IEEE Computer Society Press, pp. 81–90.

[78] KAZMAN, R., KLEIN, M., BARBACCI, M., LONGSTAFF, T., LIPSON, H., AND CARRIERE, J. The architecture tradeoff analysis method. In *Fourth IEEE International Conference on Engineering of Complex Computer Systems. ICECCS'98.* (1998), pp. 68–78.

[79] KHAN, K. M., AND HAN, J. Composing security-aware software. *IEEE software*, 1 (2002), 34–41.

[80] KIM, J.-H., KWON, H., KIM, D.-H., KWAK, H.-Y., AND LEE, S.-J. Building a service-oriented ontology for wireless sensor networks. In *Computer and Information Science, 2008. ICIS 08. Seventh IEEE/ACIS International Conference on* (2008), IEEE, pp. 649–654.

[81] KOSKO, B. Fuzzy cognitive maps. *International Journal of man-machine studies 24*, 1 (1986), 65–75.

[82] KUECHLER, B., AND VAISHNAVI, V. On theory development in design science research: anatomy of a research project. *European Journal of Information Systems 17*, 5 (2008), 489–504.

[83] KUFEL, L. Security event monitoring in a distributed systems environment. *IEEE Security Privacy 11*, 1 (Jan. 2013), 36–43.

[84] LABS, A.-I. Available from: `http://autoidlabs.org`.

[85] LEISTER, W., HAMDI, M., ABIE, H., POSLAD, S., AND TORJUSEN, A. An evaluation framework for adaptive security for the iot in ehealth. *International Journal On Advances in Security 7*, 3 and 4 (2014), 93–109.

[86] LIU, C., ZHANG, Y., ZENG, J., PENG, L., AND CHEN, R. Research on dynamical security risk assessment for the internet of things inspired by immunology. In *Natural Computation (ICNC), 2012 Eighth International Conference on* (2012), pp. 874–878.

[87] MA, H.-D. Internet of things: Objectives and scientific challenges. *Journal of Computer science and Technology 26*, 6 (2011), 919–924.

[88] MACDONALD, N. The future of information security is context aware and adaptive. *Gartner RAS Core Research Note G 200385* (2010).

[89] MACDONALD, N., AND FIRSTBROOK, P. Designing an adaptive security architecture for protection from advanced attacks, February 2014.

[90] MARCH, S. T., AND SMITH, G. F. Design and natural science research on information technology. *Decision support systems 15*, 4 (1995), 251–266.

[91] MCGRAW, R. W. Risk adaptable access control, 2009. Last accessed on: 19 Nov 2015. Available from: `http://csrc.nist.gov/news_events/privilege-management-workshop/radac-Paper0001.pdf`.

[92] MEDAGLIA, C. M., AND SERBANATI, A. An overview of privacy and security issues in the internet of things. In *The Internet of Things*. Springer, 2010, pp. 389–395.

[93] MICHELSON, B. M. Event-driven architecture overview. *Patricia Seybold Group 2* (2006).

[94] MICROSOFT. Handling and raising events. Microsoft Developer Network. Last accessed on 21 Dec 2014. Available from: https://msdn.microsoft.com/en-us/library/edzehd2t(v=vs.110).aspx.

[95] MICROSOFT. *Architectural Patterns and Styles. Microsoft Application Architecture Guide*, 2nd edition ed. November 2009. ISBN: 9780735627109.

[96] MIORANDI, D., SICARI, S., PELLEGRINI, F. D., AND CHLAMTAC, I. Internet of things: Vision, applications and research challenges. *Ad Hoc Networks 10*, 7 (2012), 1497 – 1516.

[97] MOSES, T. OASIS Standard - eXtensible Access Control Markup Language (XACML) Version 2.0, Feb 2005. Last accessed on 24-April-2016. Available from: http://tinyurl.com/OASIS-XACML-Schema.

[98] MYERS, M. D., ET AL. Qualitative research in information systems. *Management Information Systems Quarterly 21* (1997), 241–242.

[99] NEUHAUS, H., AND COMPTON, M. The semantic sensor network ontology: A Generic Language to Describe Sensor Assets. In *AGILE workshop on challenges in geospatial data harmonisation, Hannover, Germany* (2009), pp. 1–33.

[100] NILES, I., AND PEASE, A. Towards a standard upper ontology. In *Proceedings of the international conference on Formal Ontology in Information Systems-Volume 2001* (2001), ACM, pp. 2–9.

[101] O'BRIEN, L., MERSON, P., AND BASS, L. Quality attributes for service-oriented architectures. In *Proceedings of the International Workshop on Systems Development in SOA Environments* (Washington, DC, USA, 2007), SDSOA '07, IEEE Computer Society, pp. 3–9. Available from: http://dx.doi.org/10.1109/SDSOA.2007.10.

[102] OVASKA, E., CINOTTI, T. S., AND TONINELLI, A. The design principles and practices of interoperable smart spaces. *Advanced Design Approaches to Emerging Software Systems: Principles, Methodology and Tools* (2012), 18–47.

[103] PALIWAL, G., AND KIWELEKAR, A. A comparison of mobile patient monitoring systems. In *Health Information Science*, G. Huang, X. Liu, J. He, F. Klawonn, and G. Yao, Eds., vol. 7798 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2013, pp. 198–209.

[104] PEFFERS, K., TUUNANEN, T., ROTHENBERGER, M. A., AND CHATTERJEE, S. A design science research methodology for information

systems research. *J. of Management Information Systems 24*, 3 (2008), 45–77.

[105] PREECE, A., GOMEZ, M., DE MEL, G., VASCONCELOS, W., SLEEMAN, D., COLLEY, S., PEARSON, G., PHAM, T., AND LA PORTA, T. Matching sensors to missions using a knowledge-based approach. In *SPIE Defense and Security Symposium* (2008), International Society for Optics and Photonics, pp. 698109–698109.

[106] PRESSER, M., AND GLUHAK, A. The internet of things: Connecting the real world with the digital world. *EURESCOM mess@ ge–The Magazine for Telecom Insiders 2* (2009).

[107] RSA. Rsa adaptive authentication. a comprehensive authentication and risk management platform, 2013. Accessed on: 19 Nov 2015. Available from: `http://www.emc.com/collateral/data-sheet/h11429-rsa-adaptive-authentication-ds.pdf`.

[108] RUSSELLO, G., CHAUDRON, M., AND VAN STEEN, M. Dynamic adaptation of data distribution policies in a shared data space system. In *On the Move to Meaningful Internet Systems 2004: CoopIS, DOA, and ODBASE*. Springer, 2004, pp. 1225–1242.

[109] RUSSELLO, G., AND DULAY, N. An architectural approach for self-managing security services. In *Advanced Information Networking and Applications Workshops, 2009. WAINA'09. International Conference on* (2009), IEEE, pp. 153–158.

[110] RUSSELLO, G., MOSTARDA, L., AND DULAY, N. Escape: A component-based policy framework for sense and react applications. In *Component-Based Software Engineering*. Springer, 2008, pp. 212–229.

[111] RUSSOMANNO, D. J., KOTHARI, C., AND THOMAS, O. Sensor ontologies: from shallow to deep models. In *System Theory, 2005. SSST'05. Proceedings of the Thirty-Seventh Southeastern Symposium on* (2005), IEEE, pp. 107–112.

[112] RUSSOMANNO, D. J., KOTHARI, C. R., AND THOMAS, O. A. Building a sensor ontology: A practical approach leveraging iso and ogc models. In *IC-AI* (2005), pp. 637–643.

[113] SALEHIE, M., PASQUALE, L., OMORONYIA, I., ALI, R., AND NUSEIBEH, B. Requirements-driven adaptive security: Protecting variable assets at runtime. In *Requirements Engineering Conference (RE), 2012 20th IEEE International* (2012), IEEE, pp. 111–120.

[114] SANCHEZ, L., LANZA, J., OLSEN, R., BAUER, M., AND GIROD-GENET, M. A generic context management framework for personal networking environments. In *Mobile and Ubiquitous Systems-Workshops, 2006. 3rd Annual International Conference on* (2006), IEEE, pp. 1–8.

[115] SAUNDERS, M.N.K., L. P. . T. A. *Research methods for business students*. FT Prentice Hall, 2003.

[116] SAVOLA, R. M., ABIE, H., AND SIHVONEN, M. Towards metrics-driven adaptive security management in e-health iot applications. In *Proceedings of the 7th International Conference on Body Area Networks* (ICST, Brussels, Belgium, Belgium, 2012), BodyNets '12, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), pp. 276–281.

[117] SAVOLAINEN, P., NIEMELÄ, E., AND SAVOLA, R. A taxonomy of information security for service-centric systems. In *Software Engineering and Advanced Applications, 2007. 33rd EUROMICRO Conference on* (2007), IEEE, pp. 5–12.

[118] SAXENA, A., LACOSTE, M., JARBOUI, T., LÜCKING, U., AND STEINKE, B. A software framework for autonomic security in pervasive environments. In *Information Systems Security*. Springer, 2007, pp. 91–109.

[119] SHACKLEFORD, D. Real-time adaptive security. Tech. rep., SANS, December 2008. Last Accessed on 4 April 2014. Available from: `http://www.sans.org/reading_room/analysts_program/adaptiveSec_Dec08.pdf`.

[120] SHEN, D., CHEN, G., BLASCH, E., AND TADDA, G. Adaptive markov game theoretic data fusion approach for cyber network defense. In *Military Communications Conference, 2007. MILCOM 2007. IEEE* (2007), IEEE, pp. 1–7.

[121] STONEBURNER, G., GOGUEN, A. Y., AND FERINGA, A. Sp 800-30. risk management guide for information technology systems.

[122] SUNDMAEKER, H., GUILLEMIN, P., FRIESS, P., AND WOELFFLÉ, S. Vision and challenges for realising the internet of things.

[123] SUNDMAEKER, H., GUILLEMIN, P., FRIESS, P., AND WOELFFLÉ, S. Vision and challenges for realising the internet of things.

[124] SZYPERSKI, C. *Component Software: Beyond Object-Oriented Programming*, 2nd ed. Addison-Wesley Longman Publishing Co., Inc., 2002.

[125] TAN, L., AND WANG, N. Future internet: The internet of things. In *Advanced Computer Theory and Engineering (ICACTE), 2010 3rd International Conference on* (2010), vol. 5, IEEE, pp. V5–376.

[126] TRIBBLE, D. A. The health insurance portability and accountability act: security and privacy requirements. *American Journal of Health-Systems Pharmacy 58* (2001), 763–770.

[127] TSOUMAS, B., AND GRITZALIS, D. Towards an ontology-based security management. In *Advanced Information Networking and Applications, 2006. AINA 2006. 20th International Conference on* (2006), vol. 1, IEEE, pp. 985–992.

[128] TUN, T., NHLABATSI, A., KHAN, N., BANDARA, A., KHAN, K., YU, Y., NUSEIBEH, B., NHLABATSI, A., TUN, T., KHAN, N., ET AL. An architecture for adaptive information security in cloud applications. Tech. rep., 2014.

[129] UNDERCOFFER, J., JOSHI, A., AND PINKSTON, J. Modeling computer attacks: An ontology for intrusion detection. In *Recent Advances in Intrusion Detection* (2003), Springer, pp. 113–135.

[130] VAISHNAVI, V., AND KUECHLER, W. Design research in information systems, Jan 2004. Available from: `http://ais.affiniscape.com/displaycommon.cfm?an=1&subarticlenbr=279`.

[131] VAN BUNNINGEN, A. H., FENG, L., AND APERS, P. M. Context for ubiquitous data management. In *Ubiquitous Data Management, 2005. UDM 2005. International Workshop on* (2005), IEEE, pp. 17–24.

[132] VAN KRANENBURG, H., BARGH, M. S., IACOB, S., AND PEDDEMORS, A. A context management framework for supporting context-aware distributed applications. *Communications Magazine, IEEE 44*, 8 (2006), 67–74.

[133] VILAMOVSKA, A.-M., HATZIANDREU, E., SCHINDLER, H. R., VAN ORANJE-NASSAU, C., DE VRIES, H., AND KRAPELS, J. Study on the requirements and options for rfid application in healthcare. Publisher RAND Europe.

[134] WANG, W., DE, S., TOENJES, R., REETZ, E., AND MOESSNER, K. A comprehensive ontology for knowledge representation in the internet of things. In *Trust, Security and Privacy in Computing and Communications (TrustCom), 2012 IEEE 11th International Conference on* (2012), IEEE, pp. 1793–1798.

[135] WEBER, R. H. Internet of things–new security and privacy challenges. *Computer Law & Security Review 26*, 1 (2010), 23–30.

[136] YANG, X., LI, Z., GENG, Z., AND ZHANG, H. A multi-layer security model for internet of things. In *Internet of Things*, Y. Wang and X. Zhang, Eds., vol. 312 of *Communications in Computer and Information Science*. Springer Berlin Heidelberg, 2012, pp. 388–393.

[137] ZHAO, K., AND GE, L. A survey on the internet of things security. In *Computational Intelligence and Security (CIS), 2013 9th International Conference on* (Dec 2013), pp. 663–667.

Part II

Published Research Articles

Article 1

An Empirical Research on InfoSec Risk Management in IoT-based eHealth

Waqas Aman and Einar Snekkenes
In the Third International Conference on Mobile Services, Resources, and Users (Mobility 2013). Pages: 99–107. Lisbon, Portugal. 2013

Chapter 7

An Empirical Research on InfoSec Risk Management in IoT-based eHealth

Abstract

Enabling the healthcare infrastructure with Internet of Things (IoT) will significantly improve quality of service, reduce the costs and efficiently manage remote and mobile patients. To be efficacious, IoT and eHealth infrastructure essentials as well as their associated security and privacy issues should be thoroughly recognized to effectively manage the InfoSec risks involved. Unfortunately, there has been a potential lack of research comprehensively addressing these issues jointly while InfoSec risk management solutions are devised for IoT-based eHealth. In this paper, we have highlighted the necessary knowledge while approaching InfoSec risk management in IoT-eHealth as per a standard process, assessed it against standard and proposed requirements and identified the current trends and gaps to set directions for future research.

Keywords - Internet of Things (IoT); Remote Patient Monitoring; Risk Management; Security & Privacy; eHealth.

7.1 Introduction

Internet of Things (IoT) is a global internet architecture connecting various wired and wireless technologies designed to meet specific objectives [43]. Beside its anticipated benefits in various private and business domains, enabling IoT in welfare spheres, such as healthcare, will greatly facilitate the society as a whole. A patient can now be monitored remotely in a continuous fashion thus making the health services more mobile, extendable and effective. Though offering a great deal of benefits, IoT is still facing a number of critical challenges such as networking, security and privacy, QoS, standardization, etc., which needs to be sorted out and yet remain open[2]. Among these challenges, the most threatening are the security and privacy concerns. Connecting diverse technologies may lead to new threats with much grander risk of security. These threats become more drastic when considered in the context of a continuous service, such as healthcare, where

the concern is not limited to a patient's privacy but, there is a threat to the breach of trust, leading to the exploitation of a welfare service.

Standards, guidelines and good practices concerning InfoSec Risk Management (ISRM), such as ISO 27005, NIST, CRAMM, ISACA RiskIT, etc., recommend to approach ISRM in a methodological fashion, i.e., understand the target business function, service or system, identify the security and privacy (S&P) concerns and threats, analyze the risk faced, and manage the risk to reduce it to an acceptable level. To qualify this process, IoT-driven eHealth as a continuous real-time service will need an intelligent security system that can dynamically predict and estimate the risk faced and mitigating it autonomously to be more resilient and adaptable in the face of changing security threats [1]. A number of architectural designs, security issues, risk management (RM) models and surveys are presented concerning eHealth [2, 8, 13, 14, 26, 43]. However, such studies are either focused on the mentioned individual topics, target a specific technology or presents abstract modeling. Hence, there is a lack of literature that provides a holistic study of the related topics as per the standard RM process to approach ISRM in IoT-based eHealth.

In this paper, we will highlight IoT-eHealth infrastructure essentials, the associated S&P issues and will explore various ISRM approaches to establish an understanding of how ISRM can be modeled in IoT driven eHealth. Existing literature is evaluated against standard and projected requirements and current trends and gaps are identified. We identified that the current system and S&P modeling are focused only on the primitive requirements and is done in an empirical manner. Whereas vital operations, key system components and necessary S&P services are overlooked. Suitability of various ISRM models and methods is explored and it was concluded that most of them have a subjective influence which makes them difficult to be adopted in a dynamic-real-time environments and lacks intelligent risk analysis and management capabilities, such as context awareness and self-adaptation, which are deemed to be essential for IoT driven eHealth[1]. We strongly believe that this contribution will provide a reference point for future researchers and will enable them to understand the requirements, challenges, options, methods and techniques necessary to consider while approaching ISRM in IoT driven eHealth.

The rest of the paper is organized as follow: In Section 7.2, an overview of the related work will be highlighted. Section 7.3 will elaborate the current literature highlighting architectural designs, S&P services, issues and threats and modeling security risks in the perspective of IoT-based eHealth. In Section 7.4, evaluation of the current literature will be highlighted by aligning them with a set of standard and proposed requirements. In Section 7.5, current trends and gaps will be identified by discussing the evaluated knowledge. Finally, concluding remarks and future research endeavors will be

underlined in Section 7.6.

7.2 Related Work

A summary of related efforts concerning IoT, remote eHealth, associated security challenges and ISRM modeling are highlighted in this section. The goal is to identify and converse the reviews which are aligned with the theme of ISRM and related topics in IoT-based eHealth.

A detailed description of networking and architectural characteristics of ubiquitous computing used for remote patient monitoring (RPM) is presented in [19]. Sunil et al. discuss the use of mobile networks and the utilization of their mobility features in RPM. 3G and 4G networks characteristics were compared and it was showed that 4G can provide magnified advantages in terms of QoS. QoS requirements concerning wireless networks were highlighted and respective suggestions were discussed to overcome some of the current shortcomings.

Wireless Sensor Networks (WSNs) play a vital role in remote eHealth setup. They enable the notion of a continuous monitoring in remote patient monitoring systems (RPMS). Murad et al.[33] stressed that preventive security measures are not sufficient for WSNs due to the presence of an internal attacker. They provided a comprehensive survey of different intrusion detection systems (IDS) categorizing rule, data mining, statistical and game theoretic based techniques as detective measures to comprehend internal and network attacks dynamically thus enabling a second layer of defense to preventive measures. Similar work is also done in[17] [6].

Latré et al.[20] discussed the importance of Wireless Body Area Network (WBAN) and its applications in remote monitoring of various diseases. Positioning of the WBAN in a RPMS setup is detailed and it is argued that most of the current research is focused on the extra-WBAN communication. Available MAC and Network layer protocols were highlighted and it was suggested that new MAC layer protocols need to be design to accommodate patient mobility. Latré et al. reasoned the current issues like QoS, usability and security are more studied in the WSN and should also be examined in WBAN being a more healthcare focused technology as compared to WSN. The survey however was more emphasized on the networking protocols.

A systematic literature review on S&P issues in an Electronic Health Record (EHR) system is presented in[12]. Literature appraisal was based on the requirements defined in ISO 27799 standard related to achieving security goals through cryptographic techniques, HR security measures, such as training and awareness, and its alignment with compliance and regulatory requirements. Luis et al. concluded that though most of the studies do explicate security controls but are not really implemented in health sectors.

A detail survey on IoT is given in[2]. Atzori et al. explain IoT from three different perspectives: *Things, Internet and Semantics* and converse its overlapping and diverse nature. Different technologies, such as Middleware, WSN, RFID, etc., are recognized to review their possibilities in enabling effective IoT. Extended opportunities of IoT in different application areas are explored and their benefits are traversed. Furthermore, a list of open issues, such as, security, privacy, networking, standardization, QoS and data integrity was highlighted and suggested to be researched to make IoT a more mature and promising technology.

To perform effective risk analysis, it is a difficult task to select the appropriate Risk Analysis (RA) methodology [42]. Vorster and Labuschagne presented a framework of evaluating RM methodologies to assist the business managers in selecting an appropriate method to conduct RA within an organization. A five-point common criterion was used for the comparison. A similar approach is also taken by [5] where RA methodologies were classified based on the involvement of risk analysts or stakeholders and the execution nature of the steps used in the RA process. RA methodologies can also be classified into two groups based on the approach adopted–*Traditional*: where a methodology have a subjective influence of the stakeholder involved and risk is analyzed by the appraisers; *Contemporary*: where risk is estimated based on the target system behavior by inspecting the events it creates, testing it and validating it with formal methods[29].

7.3 Approaches, Concepts & Issues

This section presents an overview of the current literature in accordance with the standard ISRM process. The selected literature encompasses systems overview, S&P services and threats and ISRM modeling approaches which are necessary to be understood while impending ISRM in IoT driven eHealth. A depiction of the literature organization in line with the standard ISRM guideline is shown in Table 7.1.

Table 7.1: Literature Organization & Standard ISRM Process

Standard ISRM Process	**Literature Organization**
Scope Identification	**IoT-eHealth Infrastructure** – System Overview & Functions – Key Assets – Comm. Medium
S&P Services/Threats	**S&P Modeling** – Threats & Security Services Modeling
Analyzing & Managing Risks	**Modeling InfoSec Risks** – Methods, Models & Frameworks for handling IoT-eHealth Risks

7.3.1 IoT-eHealth Infrastructure

IoT-based eHealth can be referred to as the global internet of wired and wireless technologies placed to monitor remote and mobile patients. Besides monitoring, patients can also be supervised over the internet and response to emergency situations can be made in a timely manner with the required aid. The infrastructure includes wearable sensors which collect various physiological sensed data from the patient as biosignals, forwards it to a smart device, such as a smartphone or tablet. Biosignals are filtered and are sent to a remote hospital site via mobile network or internet where the medical staff further investigate them and prescribe the patient accordingly. This concept is also portrayed in[28] in which Otto et al.explained a heart patient scenario while presenting their RMPS. A similar model is also described in[31] in which the proposed system, Tele Health Care, is used to monitor blood pressure and heart rate of a remote patient. In abnormal situations the patient is alerted with an alarm and a SMS is sent to the corresponding doctor for instant response. Ambulatory and emergency situations are also discussed. However, Rajan et al. did not discuss the notion of false alerts which may cause panic on both the patient and doctor sides.

Suh et al. proposed a RPMS, *WANDA*, for monitoring congestive heart failure patients[39]. The system is composed of three tiers: sensors, web and back end databases. Mobility is provided through the use of a smartphone carried by a patient. Via Bluetooth the biosignals are transmitted to a smartphone from the sensors and are sent to the second tier through GSM, 3G and/or Internet for further investigations. Health status can be accessed either by using the smartphone or the web services. The database tier is used only for backup and recovery procedures assisted with offline backup schedulers.

Based on the fact that a TV is still the most convenient way of interaction among the older adults, Santos et al. presented a TV based solution, *CareBox*, for RPM[35]. CareBox processes the vital signs only locally. Sensor data is sent to the monitoring unit attached to a TV where the patient can have a look at to his health status displayed on TV. The communication layer of the system is designed to support various protocols and technologies. A VoIP client is used where a patient can connect to a doctor for a video meeting. A survey form is programmed into the TV, which asks health related questions from the patient and can be sent upon submission to the doctor site via an internet connection.

Scacht et al. proposed *Fontane*[36]. In Fontane, medical data sensed by various sensors are transmitted to a home broker via Bluetooth. The home broker, implemented in a smartphone, sends the processed data to a telemedicine center (TMC) using GSM or UMTS. The live medical data received in the TMC is recorded as the patient's EHR. A J2EE–based SaPiMa module is used at the TMC to ensure EHR interoperability. Medical professionals

can access the EHRs via the internet to review health status. Based on specific prioritization rules set by a doctor, the system can review orders for the patients.

Sneha et al.[37] provided a comprehensive set of requirements for RPMS and suggested a three-step framework for RMPS: Sensing the vital signs, Analyzing them and if an anomaly is found, the analysis report is transmitted to the concerned site. A PDA equipped with different agents responsible for various tasks such as location update, collection and processing of vital signs, alarm generation, updating EHR and storage of personal data are utilized. These agents use ontology based on Descriptive Logic (DL) and implements various alerts and alarms as per the patient history. Sneha et al. however, did not discuss the patient-doctor communication within their model.

Wu et al.[44] presented an RIFD based Mobile Patient Monitoring System (MPMS) which they claimed to be the first of RFID driven RPMS. The sensor part of the network is composed of wearable ring-type pulse monitoring tags. The sensed data from the tags are sent to a reader where it is delivered to a smartphone via Bluetooth. The smartphone has the ability to process and analyze the data and anomalies are shared with a remote medical station. The smartphone is also equipped with a GPS, which sends out the patient location to the medical station in case of out-door emergencies. RFID is also used in[18] for an out-patient registration. Though, the title reflects a MPMS but is in-fact a model to facilitate the patient's check-in procedure in the hospital. A patient is registered into the system and an RFID bracelet is given to him. The doctor's PDA connects to the RFID server and retrieves the patient information. After the personal information is read, the corresponding patient history is extracted from the health system and advising is done accordingly.

Van et al. proposed *MobiHealth* system experimented in a number of countries [41]. In MobiHealth, the health information was transferred through the next generation wireless networks. Van et al. argued that beside wearable sensors, devices such as actuators and other wearable devices can also be integrated into the system. MobiHealth, however, was prone to major issues of data loss and low bandwidth drawn from the experiments conducted.

Kargl et al. presented a pervasive eHealth monitoring system, *ReMoteCare*[16]. ReMoteCare consists of a local processing and data collection units, which process and collect local data through sensor motes. The data is then forwarded to a remote or local analysis unit over a communication network through a gateway. A PC is used for local analysis from where analyzed data can be sent to a remote processing and collection unit via SNMP for further investigation.

7.3.2 Security & Privacy Modeling

eHealth involves critical information exchange and requires a number of security services to make this information reliable, confidential, available and trustworthy. The objective of this section is to understand the threat landscape, S&P issues and how various security services are modeled in remote/mobile patient monitoring.

RPMSs will no doubt greatly improve the quality of healthcare. However, it still have to face a number of challenges concerning S&P. Meingast et al.[26] discussed the issues concerning data access and storage such as authorization, data retention and the type of data to be stored to meet privacy objectives. Regulatory requirements and conflicts among regulations are also highlighted. They stressed that existing controls such as Role Based Access Control (RBAC), Encryption and Authentication mechanisms should be implemented to overcome these issues.

Extending the notion of threats posed in a MPMS, Leister et al. produced a threat assessment report stating the critical threats faced in an MPM environment using various scenarios [21]. Though, the main focus of the assessment is on the WSNs, they have also considered the long range wireless communication infrastructure and the corresponding threats. They also suggested a few countermeasures and security recommendations which can be considered to circumvent these threats.

A comprehensive analysis of threat faced by the WSNs is presented in[14]. The attacks and threats listed by Kalita and Kar are not specific to eHealth but as WSN plays a vital role in RPMS, these threats should be seriously considered when a secure design or risk analysis of RPMS are intended. The attacks identified are categorized in accordance with the TCP/IP network model so that appropriate measure can be taken at the specified layer. Countermeasures are suggested to avoid some of the common attacks.

Lin et al. presented a privacy protection scheme depicting how patient's privacy can be preserved in an MPMS setup [22]. Lin et al. demonstrated how the privacy of the patient medical information is protected from a global adversary trying to eavesdrop on the messages transferred between the patient and the doctor. Furthermore, they explained the preservation of patient's contextual privacy using the proposed scheme showing that an adversary cannot link a patient to a specific doctor by linking their sources and destinations. They also performed a thorough performance analysis of the proposed scheme demonstrating its efficiency in terms of transmission delays. Ramli et al.[32] provided an insight on four serious privacy issues in pervasive health monitoring systems; eavesdropping, prescription leakages, social implication and abuse of medical information. They argued that these concerns not only affect the health system but also greatly influence patient's life.

Frank et al. described different types of attacks that can be experienced

by various network components in RPM as well as the threats corresponding to the information shared between them [16]. They suggested a number of security measures that can be used to prevent internal and external attackers from compromising the confidentiality, integrity and availability of the network components and information. However, privacy and legal issues are just mentioned and are not well elaborated.

Apaporn et al.[4] presented a security framework for eHealth services using two mechanisms: Data and Channel security. Channel security is provided using the SSL on the HTTP layer and data security is provided on the SOAP layer constructed above the HTTP. Apaporn et al. emphasized that RBAC should be used along with multi-factor authentication to ensure proper authorization and authentication. Based on the roles of stakeholders and data sensitivity, communication is divided into different layers where various authentication and encryption settings can be adapted. The framework however dealt only with the web based eHealth services. Multi-factor authentication is also utilized in [38] where Sriram et al. used ECG and accelerometer features from the sensor to perform an activity based biometric authentication.

Elkhodar et al. proposed a Ubiquitous Health Trust Protocol (UHTP) in combination with TLS to authenticate a mobile doctor visiting patients at home[9]. Authentication is performed using three factors based on personal, device and environmental (location) information. During a request to a patient EHR, the doctor uses his smart phone to access the EHR system using his username and password. Beside these personal credentials, the SIM details, IMEI and GPS locations from doctor's phone are validated and access is granted accordingly. The rest of the communication security is ensured as per the TLS negotiated parameters. UHTP, however, doesn't have any application in a continuous RPM orientation.

Simple and secure RPMS is demonstrated in[15]. A mobile set is used as a pulse oximeter where pulse rates are transmitted to a smartphone. The smartphone is equipped with a symmetric cipher and a hashing algorithm to achieve confidentiality and integrity. Shortcomings of this model are ignoring the distribution concerns of the keys and the abstract knowledge of the model, which needs to be detailed.

Timestamps can provide valuable and fresh data for authentication and requires no active involvement of the user[10]. Elmufti et al. used packet timestamps to authenticate a patient/doctor (users) in RPMS. Users are assigned tokens based on timestamps signed by an authenticating server. These stamps are transmitted with individual messages and are compared with a sliding window maintained at the receiving end. User authentication itself is done with digital signature. Elmufti et al. although included sensors in their architecture but did not explore the proposed protocol applications in them.

QoS and event reporting are important requirements in information system. In eHealth, real-time delivery is a must and health status has to be monitored continuously[34]. Rikitake et al. presented an NGN/IMS based ubiquitous health monitoring system in which they addressed the issues of event notification, real-time transfer and data accumulation. Sensor's data is sent to an IMS Client from where it is sent to the observer's site using Realtime Transfer Protocol (RTP). For event notification a SIP base Subscribe/Notify module is utilized that records incidents in an event server connected to the hospital application server. An XML database management system (XDMS) is used that extracts the events from the event server and stores it in an XML format.

Malhotra et al. used Elliptic Curve Cryptography (ECC) to secure the exchange of medical data using mobile devices[25]. Basic ECC methods are used where encryption is done at the user level with a public-private key pair. User is authenticated through a username/password terminal and access to the data is granted based on the user (patient/doctor) role. ECC based digital signature to ensure non-repudiation while message integrity is provided through a cryptographic hash.

7.3.3 Modeling the InfoSec Risk

To detect and prevent accidental events regarding a patient's health, an activity based risk analysis framework is proposed in [8]. Collected vital signs events are matched with the patient's history already stored as EHR and the current situation of the patient is predicted. Based on the prediction, risk is calculated and an alert is generated to cope up with the situation. The proposed architecture, although only address the patient health, can be extended to the information security domain as a reference when modeling InfoSec risk analysis is desired.

There are several studies on general S&P issues in eHealth comprising ubiquitous systems. However, it is quite hard to understand and systematically listing down these key issues and design a risk mitigation strategy for them. Oladimeji et al.[27] proposed a framework to model security and privacy objectives, identifying threats and risks and approaching their mitigation strategies. They also discussed how information sensitivity can be characterized as well as how different administrative policies can be refined to protect the patient's privacy.

The attributes that are used to design IT solutions specifically in eHealth are usually complex and interdependent thus needed to be analyzed and prioritize to produce a reliable and trustworthy solution. In[7], it is discussed how these critical attributes and their inter-dependencies can be assessed to reduce the risk after the solution has been deployed. The study can be used for formulating the requirements of designing an automated

or real-time risk analysis model as it discuss both the quality and security issues at the requirement engineering level.

Bønes et al. proposed *ModIMob*, a model which can be used to discover the availability of the health experts where their presence is required for an expert opinion[3]. The Australian and New Zealand standard for RM (AS/NZS 4360:1999) is used to discover the risks associated with the use of IM and mobile services used in a healthcare. Though, the scope of their risk evaluation is limited to a specific domain of instant messaging but it can provide an understanding of conducting a RA process in a RPMS.

Abie and Ilangko proposed a risk based adaptive framework for IoT-based eHealth [1]. They argued that based on the real time data collected from the sensors and recent information history, a risk will be calculated, which will further be used in the decision making process of system adaptation. They also provide a detailed literature on various issue concerning system adaptation and risk management and it is deemed that using context awareness and Game Theory techniques, the faced risk can be effectively estimated and predicted.

To provide an appropriate level of privacy all the assets as well as the stakeholders involved in the target system must be considered[13]. A Privacy Risk Model is demonstrated specifically targeting the Ubicomp systems where risks concerning privacy are identified and analyzed by a series of questions. RM is performed by categorizing the risks analyzed and designing architectural strategies for them.

Maglogiannis et al. presented a detail risk analysis of RPMS. RA is performed through the CCTA Risk Analysis and Management Methodology (CRAMM) by considering a case study highlighting the associated key risks[24]. The results of the RA are used in developing a graph using Bayesian Network technique showing the interaction of various critical events that can cause system failure.

Beside the risk posture of the sensitive information processed by the health information systems, the devices used in healthcare have their own inherited risks. With the introduction of pervasive computing and IoTs this risk has grown rapidly. Zhao and Bai described how Failure Mode and Effect Analysis (FMEA) can help in analyzing and managing the risks associated with these devices to circumvent any potential hazards[45]. They showed that Risk Priority Number (RPN) can be used in the context of FEMA to reduce such potential casualties associated with medical devices.

ENISA, using EBIOS tool, performed a detailed RM process of a diabetes case study basing a RPMS[11]. EBIOS is a tool that incorporates the 5-steps RM process developed by the Central Information Systems Security Division of France. The report described a detailed step-by-step procedure of assessing and managing risks indicating the intended audience how to approach the overall process of risk management in MPMSs.

IoT comprises of a complex architecture composed of a variety of technologies due to which the overall threat faced becomes more drastic. There is a need for a sophisticated risk analysis method to assess the risk faced. Lui et al. proposed a mathematical dynamic risk assessment model, DRAMIA, to cope with the threat situation confronted in the IoT space[23]. Enthused by the Artificial Immune System (AIS) their proposed method consist of two components: a Detection Agent that sense and detect the attack environment and evolve accordingly; and a Dynamic Risk Assessment subsystem that computes the risk associated with the attack detected.

7.4 Evaluation

In this section, an evaluation of discussed literature is depicted. Evaluation is performed by mapping the reviewed articles onto a set of standard and proposed requirements.

7.4.1 System Models Evaluation

System models discussed in section 7.3.1 are evaluated against a set of functional requirements proposed in[30]. We believe that these are complete set of requirements which should be included in any RPMS and MPMS. However, we have added an important requirement of *mobility* as it is the only component that makes the health service mobile and assist in out-doors ambulatory and activity monitoring needs[37]. Functional requirements are described below whereas system models evaluation against the requirements is shown in Table 7.2. ($\surd$) mark indicates the presence of a specific function whereas an (–) implies that either the function is absent or not explicitly discussed.

- **Collection and Processing**: Collection and processing of vital signs from the body sensors by a Patient Cluster Head (PCH) or a wireless base station (BS)

- **Real Time Delivery**: PCH or BS should be able to deliver the processed data in real time for analysis to specified destination such a remote hospital site or a smart device.

- **Alarm generation**: The investigating node, a server at hospital or the smart device, should be able to generate alarms based on the real time data received both locally and remotely at hospital.

- **Interpretation**: Local and remote investigating nodes should be able to diagnose and interpret processed vital sign.

- **Correlation**: Local and remote investigating nodes should be cable of correlating various vital sign such as heart rate, diabetes level and blood pressure to diagnose the correct health status
- **Data Request**: Patient health history should be made available whenever requested
- **Communication Interface**: A communication interface should be incorporated locally to enable expert supervision for a remote patient.
- **Actuation**: To assist elder patients or on demand basis sensors or actuators should be able to saturate the essential medicine or trigger the required action.
- **Mobility**: The system should be able to support mobility services to the patient. This includes tracking the location and service availability while the patient is moving.

Table 7.2: IoT-based eHealth Systems Evaluation

Function/ Reference	[39]	[16]	[35]	[28]	.[41]	[31]	[36]	[37]	[18]	[44]
Collection & Processing	√	√	√	√	√	√	√	√	√	√
Realtime Delivery	√	√	√	√	√	√	√	√	√	√
Alarm Gen.	–	√	–	–	–	√	–	√	√	√
Interpretation	√	√	√	√	√	√	√	√	√	√
Correlation	–	–	–	√	–	–	–	√	–	√
Data Request	√	–	–	–	√	√	–	–	√	–
Comm. Interface	–	–	√	–	–	–	–	–	–	–
Actuation	–	–	–	–	√	–	–	–	–	–
Mobility	–	√	–	–	√	–	–	√	–	√

7.4.2 Evaluating S&P Modeling

S&P service modeling literature reviewed in section 7.3.2 is evaluated against the networking and communication requirements standardized by the U.S Health Insurance Portability and Accountability Act (HIPAA) of 1996 specified in [40]. Besides, ensuring health insurance coverage and simplification of administrative policies, HIPAA aims to standardize S&P mechanisms for electronic health information exchange. Requirements stated by HIPAA are: Data Access, Confidentiality, Integrity, Availability, Alarm Generation, Identity Management, Privacy Preservation, Authentication, and Event Reporting. Table 7.3 depicts the requirement(s) covered by each study as per HIPAA security requirements and how they are approached in individual study.

Table 7.3: Mapping S&P requirements onto HIPAA

Author	Data Access	Confidentiality	Integrity	Availability	AlarmGen.	Identity Mgt	Privacy	Authentication	Event Rep.
Lin et al.[22]	–	Symmetric Encryption	Hash	–	–	PKI based on Patient IDs	Symmetric Encryption & Pseudo ID	Shared Key	–
Apaporn et al.[4]	RBAC	Symmetric Encryption	–	–	–	–	–	Muliti-factor	–
Elkhodar et al.[9]	–	–	–	–	–	–	–	Multi-factor	–
Mona et al.[15]	–	Symmetric Encryption	SHA–1	–	–	–	–	Message Authentication Code(MAC) based on a Secret key	–
Khalid et al.[10]	–	–	–	–	–	–	–	User: Digital Signatures Message: Timestamps	–
Koichiro et al.[34]	AAA over NGN/IMS	–	–	Realtime Transfer Protocol (RTP)	–	–	–	AAA over NGN/IMS	SIP Event Subscribe/Notify Framework
Sriram et al.[38]	–	–	–	–	–	–	–	Multi-modal Biometrics	–
Malhotra[25]	RBAC	ECC	SHA-1	–	–	–	–	ECC based Digital Signatures	–

7.4.3 InfoSec RM Models Suitability

IoT-based eHealth is a continuous service in which response to an adverse situation should be made in a dynamic fashion. Hence, it requires an ISRM solution that can estimate and predict the security risk faced in real time and adapts appropriate security setting accordingly[1]. To capture the requirements of a real time RM in IoT-eHealth below we devise a fitness criteria, which we believe should be met by a given ISRM model in order to fulfill the operational needs of IoT-eHealth and efficiently manage the risk faced.

- **Operational Nature:** The time at which an ISRM process is executed. This can be *on-demand* basis where the process is activated when required. For instance, ISACA Risk IT method can be executed bi-annually or quarterly by an enterprise. ISRM can be performed in a *dynamic* manner where security risks are analyzed in a real-time fashion such as in military setups. For IoT driven eHealth the operational level should be dynamic in order to be in line with the continuous monitoring theme.
- **Context Awareness:** It corresponds to the understanding of an adverse situation in a given time. In most cases, risks are analyzed individually however; in real computing environment, risk can be seen as a combination of different adverse events. These events and risks need to be correlated to understand a given situation otherwise low impact risks might be tagged as critical leading to false positives and unwanted situations.
- **Analysis Complexity:** It should be taken care of that risk analysis method is lightweight and fast in response to facilitate the theme of real time service [1]. RA solutions having low computational complexity can also be integrated in devices with limited resources.
- **Self-Adaptation:** For IoT-eHealth to be dynamic and self-adaptive, an ISRM should have the ability to react to an adverse situation and manage security autonomously. Self-adaptation refers to the autonomous effective reaction of a system to minimize the effect of a risky situation [1].

In Table 7.4, we evaluate the suitability of the studied ISRM approaches against the above mentioned metrics to see how they address these metrics in order to be implemented in IoT-based eHealth.

7.5 Trends And Gaps

Key elements of ISRM concerning IoT-eHealth are reviewed in this paper as system, S&P and InfoSec risk modeling and are evaluated as per pro-

Table 7.4: ISRM Approaches Suitability in IoT-based eHealth

Author	Artifact	Analysis Method	Operational Nature	Context Awareness	Complexity	Self Adaptation
Don et al.[8]	Framework	Quantitative Analysis of patient activities	Dynamic	Event Correlation		No
Croll et al.[7]	Framework	Qualitative Investigation of Quality, Usability, Privacy and Safety (QUPS) Attributes	Dynamic & OnDemand	Investigating interdependent critical attributes and events		No
Hong et al.[13]	Model	Qualitative Assessment based on a questionnaire	On-Demand	No		No
Liu et al.[23]	Method	Quantitative RA based on attack detection in network packets using Artificial Immune System	Dynamic	No	Attack detection & RA are done by specific agents	Adaptation is performed only to enhance the detection capabilities.No mechanism of adapting a RM strategy
Maglog et l.[24]	Case Study	Threat Identification is performed using Bayesian Network Modeling whereas CRAMM is used as a RA method	On-Demand	Event dependencies are used to build the context of a specific threat	Unclear to evaluate the actual computation complexity just on the graphical model presented	No
Nes et al.[3]	Case Study	Methodology: Australian & New Zealand Standard for RM ASNZS 43601999. Qualitative Approach is used in the RA process	On-Demand	No		No
Abie et al.[1]	Framework	Monitor, Analyze & Adapt loop.	Dynamic	Game Theory & Context Awareness		Yes
Zhao et al.[45]	Model	Methodology: FMEA . Risk (RPN) is analyzed using Severity, Occurrence & Detection (SOD) values	On-Demand	No	Low: RPN = SxDxO	No
ENISA[11]	Case Study	Qualitative 5-Step EBIOS RA Methodology: Formulating Risk, Asset Valuation, Probability Calculation, Impact Valuation & Prioritizing Risk Levels	On-Demand	No	Low- Risk Calculation: Risk = (Threat x Vulnerability x Impact)	No

jected requirements. The objective was to understand and recognize the essential operations, S&P challenges and methodologies for effective ISRM in IoT driven eHealth. A brief discussion on the evaluated knowledge corresponding to individual domains is conferred below to reflect the current trends and gaps in the existing literature.

– System Models

A total of 10 models are studied and analyzed according to the required features in a RPMS or IoT driven eHealth. Some of the models reviewed are focused on monitoring generic vital signs such as ECG, Blood pressure and heart rate [36, 41] while a few targets specific heart [28, 39] and chronic diseases such as diabetes [37]. Systems corresponding to[28, 36, 37, 39, 41] emphasized the use of cellular network (GPRS, GSM and UMTS) for the transmission of sensed data to the hospital site through the use of smart phones. However, simultaneous transmissions on cellular networks can cause performance degradation and may affect continuous monitoring in critical situations[41]. Except for [37], the importance of local analysis of sensed data is ignored in the rest of the models, which enable a patient to view his health status locally and schedule the daily routines accordingly. Similarly, actuation of medical infusions is also overlooked. A vital functionality of RPM is to diagnose the patient at home to save the time and energy spent in regular checkups, i.e., the provisioning of communication interface between a doctor and patient however, an absence is experienced of this feature in most of the systems reviewed. Those who support this functionality did not explicate it in detail. Santos et al.[35] on the other hand fairly explained a patient-doctor communication over a VOIP client, which can also be used in calling the health facilities in case of emergencies as well. Alarm generation is merely explored, except for[37] who detailed each alarm as per the assigned agent's responsibilities.

It can be seen that most of the system models are focused on the basic functionalities of collection, processing and delivery of vital signs to the remote hospital site. Analysis and correlation of various bio-signals are limited to the server side, which is needed to be shifted to the patient side to increase patient satisfaction. Mobility features should be well designed to support both in and out door patient and to facilitate ambulatory services[37]. Security and safety alarms are needed to be designed intelligently to support critical patient monitoring. Communication interface and GUIs needed to be constructed in order to enrich a patient-doctor relationship and trust.

– Security & Privacy

Among the HIPAA required services for secure remote and mobile patient monitoring systems, the most addressed are the confidentiality and authentication. However, none of them addresses all the HIPPA requirements. Our objective here is not to criticize this fact but to recognize how these requirements can be approached and to identify the current focus of S&P modeling

and the necessary issues to be explored in future.

In most of the literature, Symmetric encryption is used to attain confidentiality [22] [4] [15]; however, asymmetric encryption using ECC is also explored[25]. Multi-factor authentication is used in a few studies where passwords, SIM credentials, GPS location, ID cards [4, 9] and vital signs (as biometrics) such as ECG and heart beats are used as various factors of authenticating patients and doctors[38]. Digital signatures are also used in authentication[10, 25]. Message authenticity is achieved by using packet timestamps and message authentication code [10, 15]. Hashing remained the only method of ensuring message integrity however, discussed by only a few[15, 22, 25]. Anonymity is only discussed in[22], where pseudo patient IDs are used to ensure identity privacy against global eavesdropping. Authorization through RBAC are conversed in[4, 25] but are not explicitly defined.

Some of the security services in a continuous RPM such as event reporting, alarm generation and availability are yet to be researched. These are the services which are used in real-time delivery and emergency situations and are the key attributes of RPMS. Most of the literature summarized targets the extra-Body Area Network (Ex-BAN) security, which includes traditional web services and back end database resources in eHealth. As per our knowledge, there is a very limited literature available on securing inter-BAN communication specific to medical information exchange. Research is necessary to be done to secure these networks as they are the core producers of the medical information in an IoT-based eHealth or RPM. Also, the resources used in such networks have limited capabilities thus there is a need to design lightweight cryptographic solutions as discussed in[25] to be aligned with sensors computational competencies.

– InfoSec RM Models

Managing InfoSec risk in IoT-based eHealth is a tough task because of the diverse nature of technology utilized in it. The evaluation of the studied literature in context of ISRM reveals that almost all of them can be used in an On-Demand basis most of which are analyzing the risk on qualitative grounds[3, 11, 13, 24, 45]. This is because of the subjective influence in RM process which makes it stiffer to be adapted in a dynamic environment. Those that can be executed in dynamic setups are suggested frameworks[1, 7] and still needs a keen and defined method of quantitative risk analysis. Liu et al.[23] on the other hand provide an effective method for analyzing the risk in a real time manner on a quantitative basis, which make it easier to program and usable for IoT-based eHealth. It also includes intelligent agents to adapt its attack detection capabilities and requires fewer resources as the threat detection and analysis is performed by specific agents. However, the suggested techniques are based on the inputs from signature based IDS, which makes it to generate false positives[33]. Self-adaptation as a risk

management strategy is completely absent and needed to be designed intelligently to make IoT-eHealth an autonomous technology.

IoT-based eHealth needs quantitative methods for predicting and estimating threats in a dynamic fashion and should be capable of understanding and analyzing the threat situation and transforming the system security autonomously [1]. Some of the methods and framework discussed such as [1, 8, 23] can be utilized as a reference point to design the desired InfoSec RM methods for IoT driven eHealth.

7.6 Conclusion and Future Work

In this paper, we have explored the existing literature in the context of approaching InfoSec risk management in IoT-based eHealth. A common knowledge of RMPSs, S&P issues, security and risk management modeling was established in the light of a standard risk management process. System models are evaluated against a set of required functionalities, models pertaining to security services are aligned with the standard HIPAA requirements and existing RM approaches in the context of IoT driven eHealth are weighed against a fitness criteria. An overall analysis is discussed and current trends and gaps are identified.

Our future work includes devising lightweight real-time InfoSec RM methods for IoT-based eHealth with the abilities of context awareness and self-adaptation. An adaptive security model will be developed that will address the mentioned InfoSec RM requirements. Security metrics and options necessary for the adaptation will be explored. To analyze the foreseen risk, Game and Utility theory will be used to model the dynamic and expected behaviors of adversaries and a comprehensive case study will be formulated to validate the model.

Acknowledgments

The work presented in this article is a part of the ASSET (Adaptive Security in Smart IoT in eHealth) project. ASSET (2012-2015) is sponsored by the Research Council of Norway under the grant agreement no: 213131/O70. Wishing thanks to the project colleagues and anonymous reviewers for their valuable suggestions and comments.

7.7 Bibliography

[1] ABIE, H., AND BALASINGHAM, I. Risk-based adaptive security for smart iot in ehealth. In *Proceedings of the 7th International Conference on Body Area Networks* (ICST, Brussels, Belgium, Belgium, 2012), Bo-

dyNets '12, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), pp. 269–275.

[2] ATZORI, L., IERA, A., AND MORABITO, G. The internet of things: A survey. *Computer Networks 54*, 15 (2010), 2787 – 2805.

[3] BNES E, HASVOLD P, H. E. S. T. Risk analysis of information security in a mobile instant messaging and presence system for healthcare. *International Journal of Medical Informatics 76(9)* (2007), 677–687.

[4] BOONYARATTAPHAN, A., BAI, Y., AND CHUNG, S. A security framework for e-health service authentication and e-health data transmission. In *Communications and Information Technology, 2009. ISCIT 2009. 9th International Symposium on* (2009), pp. 1213–1218.

[5] CAMPBELL, P. L., AND STAMP, J. E. A classification scheme for risk assessment methods. Last Accessed On: 13-Sept-2013. Available from: `http://prod.sandia.gov/techlib/access-control.cgi/2004/044233.pdf`.

[6] CHRISTIN, D., MOGRE, P. S., AND HOLLICK, M. Survey on wireless sensor network technologies for industrial automation: The security and quality of service perspectives. *Future Internet 2*, 2 (2010), 96–125.

[7] CROLL, P. R., AND CROLL, J. Investigating risk exposure in e-health systems. *International Journal of Medical Informatics 76(5-6)* (2006), 460–465.

[8] DON, S., CHOI, E., AND MIN, D. A situation aware framework for activity based risk analysis of patient monitoring system. In *Awareness Science and Technology (iCAST), 2011 3rd International Conference on* (2011), pp. 15–19.

[9] ELKHODR, M., SHAHRESTANI, S., AND CHEUNG, H. An approach to enhance the security of remote health monitoring systems. In *Proceedings of the 4th international conference on Security of information and networks* (New York, NY, USA, 2011), SIN '11, ACM, pp. 205–208.

[10] ELMUFTI, K., WEERASINGHE, D., RAJARAJAN, M., RAKOCEVIC, V., AND KHAN, S. Timestamp authentication protocol for remote monitoring in ehealth. In *Pervasive Computing Technologies for Healthcare, 2008. PervasiveHealth 2008. Second International Conference on* (2008), pp. 73–76.

[11] ENISA. Being diabetic in 2011 - identifying emerging and future risks in remote health monitoring and treatment. Technical Publication on ENISA website, 2009. Last Accessed On: 13-Sept-2013.

Available from: http://www.enisa.europa.eu/publications/archive/being-diabetic-2011/.

[12] FERNNDEZ-ALEMN, J. L., SEOR, I. C., NGEL OLIVER LOZOYA, P., AND TOVAL, A. Security and privacy in electronic health records: A systematic literature review. *Journal of Biomedical Informatics*, 0 (2013), –.

[13] HONG, J. I., NG, J. D., LEDERER, S., AND LANDAY, J. A. Privacy risk models for designing privacy-sensitive ubiquitous computing systems. In *Proceedings of the 5th conference on Designing interactive systems: processes, practices, methods, and techniques* (New York, NY, USA, 2004), DIS '04, ACM, pp. 91–100.

[14] KALITA, H. K., AND KAR, A. Wireless sensor network security analysis. *International Journal of Next-Generation Networks (IJNGN), 1* (December 2009), 1–10.

[15] KAMEL, M., FAWZY, S., EL-BIALY, A., AND KANDIL, A. Secure remote patient monitoring system. In *Biomedical Engineering (MECBME), 2011 1st Middle East Conference on* (2011), pp. 339–342.

[16] KARGL, F., LAWRENCE, E., FISCHER, M., AND LIM, Y. Y. Security, privacy and legal issues in pervasive ehealth monitoring systems. In *Mobile Business, 2008. ICMB '08. 7th International Conference on* (2008), pp. 296–304.

[17] KARLOF, C., AND WAGNER, D. Secure routing in wireless sensor networks: attacks and countermeasures. In *Sensor Network Protocols and Applications, 2003. Proceedings of the First IEEE. 2003 IEEE International Workshop on* (2003), pp. 113–127.

[18] KORKMAZ, I., ATAY, C., AND KYPARISIS, G. A mobile patient monitoring system using rfid. In *Proceedings of the 14th WSEAS international conference on Computers: part of the 14th WSEAS CSCC multiconference - Volume II* (Stevens Point, Wisconsin, USA, 2010), ICCOMP'10, World Scientific and Engineering Academy and Society (WSEAS), pp. 726–732.

[19] KUMAR, S., KAMBHATLA, K., HU, F., LIFSON, M., AND XIAO, Y. Ubiquitous computing for remote cardiac patient monitoring: a survey. *Int. J. Telemedicine Appl. 2008* (Jan. 2008), 3:1–3:19.

[20] LATRÉ, B., BRAEM, B., MOERMAN, I., BLONDIA, C., AND DEMEESTER, P. A survey on wireless body area networks. *Wirel. Netw. 17*, 1 (Jan. 2011), 1–18.

[21] LEISTER, W., ABIE, H., GROVEN, A.-K., FRETLAND, T., AND BALASINGHAM, I. Threat assessment of wireless patient monitoring systems. In *Information and Communication Technologies: From Theory to Applications, 2008. ICTTA 2008. 3rd International Conference on* (2008), pp. 1–6.

[22] LIN, X., LU, R., SHEN, X., NEMOTO, Y., AND KATO, N. Sage: a strong privacy-preserving scheme against global eavesdropping for ehealth systems. *Selected Areas in Communications, IEEE Journal on 27*, 4 (2009), 365–378.

[23] LIU, C., ZHANG, Y., ZENG, J., PENG, L., AND CHEN, R. Research on dynamical security risk assessment for the internet of things inspired by immunology. In *Natural Computation (ICNC), 2012 Eighth International Conference on* (2012), pp. 874–878.

[24] MAGLOGIANNIS, I., ZAFIROPOULOS, E., PLATIS, A., AND LAMBRINOUDAKIS, C. Risk analysis of a patient monitoring system using bayesian network modeling. *J. of Biomedical Informatics 39*, 6 (Dec. 2006), 637–647.

[25] MALHOTRA, K., GARDNER, S., AND PATZ, R. Implementation of elliptic-curve cryptography on mobile healthcare devices. In *Networking, Sensing and Control, 2007 IEEE International Conference on* (2007), pp. 239–244.

[26] MEINGAST, M., ROOSTA, T., AND SASTRY, S. Security and privacy issues with health care information technology. In *Engineering in Medicine and Biology Society, 2006. EMBS '06. 28th Annual International Conference of the IEEE* (2006), pp. 5453–5458.

[27] OLADIMEJI, E. A., CHUNG, L., JUNG, H. T., AND KIM, J. Managing security and privacy in ubiquitous ehealth information interchange. In *Proceedings of the 5th International Conference on Ubiquitous Information Management and Communication* (New York, NY, USA, 2011), ICUIMC '11, ACM, pp. 26:1–26:10.

[28] OTTO, C., MILENKOVIĆ, A., SANDERS, C., AND JOVANOV, E. System architecture of a wireless body area sensor network for ubiquitous health monitoring. *J. Mob. Multimed. 1*, 4 (Jan. 2005), 307–326.

[29] PAINTSIL, E. Taxonomy of security risk assessment approaches for researchers. In *Computational Aspects of Social Networks (CASoN), 2012 Fourth International Conference on* (2012), pp. 257–262.

[30] PALIWAL, G., AND KIWELEKAR, A. A comparison of mobile patient monitoring systems. In *Health Information Science*, G. Huang, X. Liu, J. He, F. Klawonn, and G. Yao, Eds., vol. 7798 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2013, pp. 198–209.

[31] Rajan, S.P., R. S., and Vijayprasath, S. Design and development of mobile based smart tele-health care system for remote patients. *European Journal of Scientific Research 70* (2012), 148158.

[32] Ramli Rusyaizila, Zakaria Nasriah, S. P. Privacy issues in pervasive healthcare monitoring system: A review. *World Academy of Science, Engineering & Technology 72* (2011), 741.

[33] Rassam, M. A., Maarof, M., and Zainal, A. A survey of intrusion detection schemes in wireless sensor networks. *American Journal of Applied Sciences 9*, 10 (2012), 1636.

[34] Rikitake, K., Araki, Y., Kawahara, Y., Minami, M., and Morikawa, H. Ngn/ims-based ubiquitous health monitoring system. In *Consumer Communications and Networking Conference, 2009. CCNC 2009. 6th IEEE* (2009), pp. 1–2.

[35] Santos, A., Castro, R., and Sousa, J. Carebox: A complete tv-based solution for remote patient monitoring and care. In *Wireless Mobile Communication and Healthcare*, B. Godara and K. Nikita, Eds., vol. 61 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*. Springer Berlin Heidelberg, 2013, pp. 1–10.

[36] Schacht, A., Wierschke, R., Wolf, M., von Lowis, M., and Polze, A. Live streaming of medical data - the fontane architecture for remote patient monitoring and its experimental evaluation. In *Object/Component/Service-Oriented Real-Time Distributed Computing Workshops (ISORCW), 2011 14th IEEE International Symposium on* (2011), pp. 306–312.

[37] Sneha, S., and Varshney, U. Enabling ubiquitous patient monitoring: Model, decision protocols, opportunities and challenges. *Decision Support Systems 46*, 3 (2009), 606 – 619.

[38] Sriram, J. C., Shin, M., Choudhury, T., and Kotz, D. Activity-aware ecg-based patient authentication for remote health monitoring. In *Proceedings of the 2009 international conference on Multimodal interfaces* (New York, NY, USA, 2009), ICMI-MLMI '09, ACM, pp. 297–304.

[39] Suh, M.-K., Chen, C.-A., Woodbridge, J., Tu, M. K., Kim, J. I., Nahapetian, A., Evangelista, L. S., and Sarrafzadeh, M. A remote patient monitoring system for congestive heart failure. *J. Med. Syst. 35*, 5 (Oct. 2011), 1165–1179.

[40] Tribble, D. A. The health insurance portability and accountability act: security and privacy requirements. *American Journal of Health-Systems Pharmacy 58* (2001), 763–770.

[41] VAN HALTEREN, A. T., BULTS, R. G. A., WAC, K. E., KONSTANTAS, D., WIDYA, I. A., DOKOVSKI, N. T., KOPRINKOV, G. T., JONES, V. M., AND HERZOG, R. Mobile patient monitoring: The mobihealth system. *The Journal on Information Technology in Healthcare 2*, 5 (October 2004), 365–373.

[42] VORSTER, A., AND LABUSCHAGNE, L. A framework for comparing different information security risk analysis methodologies. In *Proceedings of the 2005 annual research conference of the South African institute of computer scientists and information technologists on IT research in developing countries* (Republic of South Africa, 2005), SAICSIT '05, South African Institute for Computer Scientists and Information Technologists, pp. 95–103.

[43] WEBER, R. H. Internet of things: New security and privacy challenges. *Computer Law & Security Review 26*, 1 (2010), 23 – 30.

[44] WU, Y.-C., CHEN, P.-F., HU, Z.-H., CHANG, C.-H., LEE, G.-C., AND YU, W.-C. A mobile health monitoring system using rfid ring-type pulse sensor. In *Dependable, Autonomic and Secure Computing, 2009. DASC '09. Eighth IEEE International Conference on* (2009), pp. 317–322.

[45] ZHAO, X., AND BAI, X. The application of fmea method in the risk management of medical device during the lifecycle. In *e-Business and Information System Security (EBISS), 2010 2nd International Conference on* (2010), pp. 1–4.

Article 2

Modeling Adaptive Security in IoT Driven eHealth

Waqas Aman

In the 6th Norsk informasjonssikkerhetskonferanse (NISK), 2013:61–69, 2014

Chapter 8

Modeling Adaptive Security in IoT Driven eHealth

Abstract

The implementation of Internet of Things (IoT) in eHealth will indeed significantly enhance ubiquitous healthcare services. Existing research in these areas and corresponding systems are more focused on functional designing and developing preventive and detective security controls. However, threats faced in IoT are more complex due to the diverse nature of technologies involved and the evolving threat landscape. They have become more sophisticated and challenging for preventive and detective technologies. Hence, we need to develop adaptive security solutions for IoT-eHealth which can predict security threats and respond to them dynamically to protect personal health information. This paper presents an adaptive security model that will learn adverse influences in IoT-eHealth infrastructure, predict and estimate the risks involved in a context-aware manner and autonomously adapt security measures in order to minimize the risk faced. The model presented is a preliminary abstraction that reflects how adaptive security can be achieved in IoT-eHealth.

8.1 Rationale

IoT is a global network focusing on the interconnection of various technologies (things) to support services quality and their extensions [26]. IoT inherit intelligent and self-* capabilities, such as self-learning and self-adapting, which makes it favorable for dynamic environments [7]. However, due to the fact that IoT allows diverse technologies, wired and wireless, it is subjected to an array of threats as underlined by [11, 13, 16, 21]. This provides an adversary multiple means and opportunities to target personal health information that is transmitted from body sensors to remote hospital sites.

Traditional preventive and detective measures such as IDS, Access Control Lists (ACLs), firewalls, anti-viruses, etc., as stand-alone controls cannot provide the reality of an ongoing attack. They lack to add contextual information failing to distinguish a security event from a non-event thus

leading to high rate of false positives [25]. Adaptive security can provide a comprehensive security solution for IoT-eHealth where diverse technologies used are threatened by an array of ever changing security and privacy risks [7, 10, 24]. This approach can be seen in various security models, such as [6, 9, 15, 23] however, they are not intended for IoT-eHealth. To elaborate the process, adaptive security systems continuously monitors user, device and network related events (as environmental influence), establishes a context among them to analyze the situation (risk) reality and devise a new security strategy (as a response to the influence) as per the risk evaluated to defend against it. This mechanism provides a *predictive security* solution where the threats are apprehended before it becomes reality [22].

To be aligned with the nature of IoT-eHealth, which is a continuous monitoring service, security risks should also be assessed and responded dynamically. In the ASSET (Adaptive Security for Smart Internet of Things in eHealth) project [1] we aim to achieve this objective by developing risk based adaptive security methods to ensure predictive and autonomous security in IoT-eHealth. This paper presents an adaptive security model for IoT-eHealth based on the requirements we analyzed in [10]. These essentials entail that: **a).** Risks need to be dynamically assessed, **b).** The solution should provide context awareness to increase security intelligence required for risk analysis and to reduce false-positives, **c).** Autonomous adaptation needs to be incorporated to evolve security settings and responding to the analyzed risks autonomously and **d).** The solution should assimilate lightweight analysis methods to reduce the computational complexity. The objective of this paper is to answer the questions: How adaptive security can be modeled in IoT-eHealth? And, what are the necessary components and actions to accommodate the mentioned requirements? The model is still in the development phase and will be explored and evaluated in the near future. However, it illustrates the ground concept that addresses the requirements we analyzed earlier and gives an abstract solution for the adaptive security process in IoT-eHealth.

The rest of the paper is organized as follow: Section 8.2, presents a typical IoT-eHealth infrastructure. The proposed model will be detailed in Section 8.3. In Section 8.4, an objectives-based evaluation is presented aiming to recognize how the proposed model meets the risk management requirements in IoT-eHealth. Finally, the concluding remarks and our future research endeavors regarding the proposed model will be detailed in Section 8.5.

8.2 IoT-eHealth Infrastructure

A typical IoT-eHealth infrastructure, depicted in the figure 8.1, includes wearable body sensors which collects various bio-signals and transmits them to a hospital site for medical investigation via intermediary nodes and com-

munication paths. In ASSET's lab, we have used a planar architecture [17] at the edge of the Body Sensor Network (BSN) in which bio-signals are collected, interpreted and (locally) analyzed by a terminal node using single-hop communication. We have introduced the terminal node as a smart *thing*, a smartphone or a tablet, in order to utilize its features and resources to support eHealth services such as mobility, emergency calls, local analysis and reporting, patient-doctor communication and e-prescriptions etc. ZigBee and Near Field Communication (NFC) are used as channel protocols between the BSN and smartphone. The bio-signals are then sent to a remote hospital site using Internet or mobile network (3G/4G, GPRS) for further investigations. Currently, the Internet is used as a communication model between the patient and hospital sites.

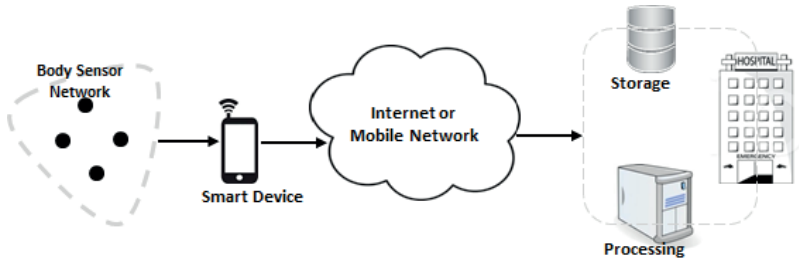


Figure 8.1: Typical IoT-eHealth Infrastructure

8.3 Proposed Model

This section elaborates our proposed risk based adaptive security model. The model is built on the concept of Security Event Management (SEM). SEM systems collects *interested* events from network devices, applications and systems and examine them to analyze the overall system security [18, 22]. They intend to provide a consolidated and centralized security management solution. The model consists of three major functional components: Monitor, Analyzer and Adapter. The entire Monitor-Analysis-Adaption process is done in a continuous real-time manner. Critical information, such as risk metrics, policies, analyzed risks, correlation rules, adaptive rules etc., are stored in the adaptive database, which are referenced and updated along the process. An abstract view of adaptive security process is depicted in figure 8.2 whereas, the proposed model is shown in figure 8.3. A description of the components and their functions is detailed in the subsequent subsections.

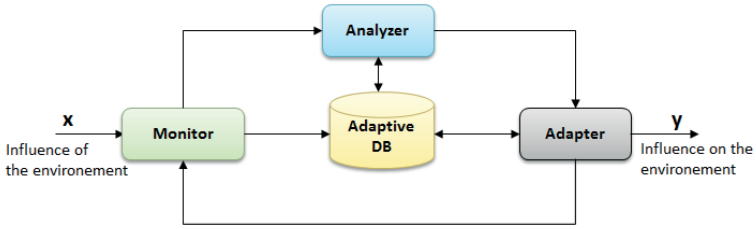


Figure 8.2: Continuous Adaptive Security Loop

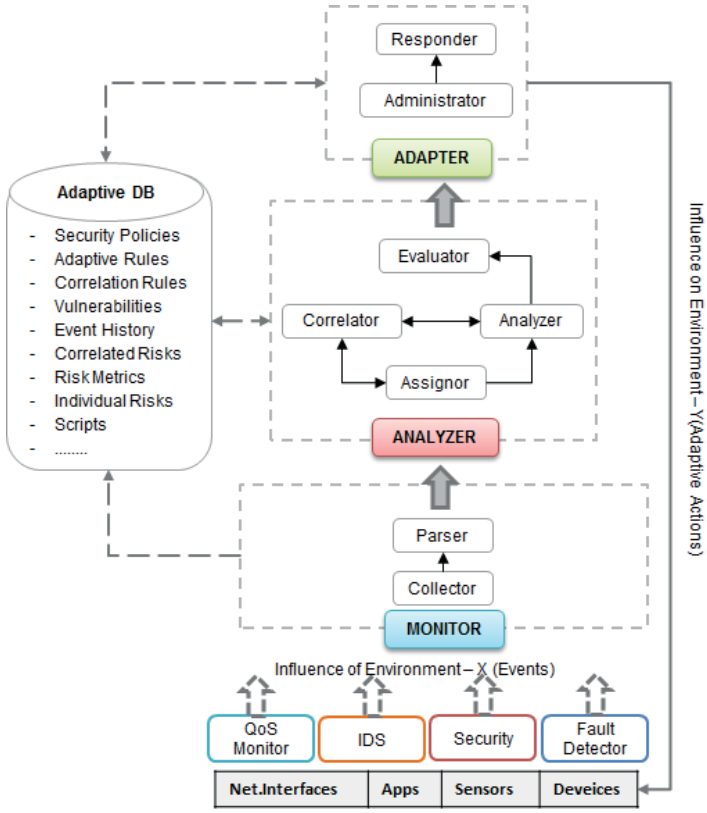


Figure 8.3: Proposed Adaptive Security Model

8.3.1 Monitoring

The *Monitor* component is used to capture the environmental influence on the individual infrastructural elements, such as the sensors, smart device, etc., as well as the associated network behavior and statistics. Depending upon the event syntax and semantics generated by a specific source, events can be sent to monitoring (Collector) unit using different transport proto-

cols, such as Syslog [2], which is the most common protocol used to collect events(logs) remotely. The collector component can be configured with the corresponding protocols' *sinks* in order to receive or collect the events in its integral form. To analyze events collected from different sources using different transport protocols, they must be transformed to a common format understandable by the *Analyzer* component. This transformation is done by the *Parser* which extracts the interested fields for analysis from the raw events and transform them to a universal format. We will be using regular expressions and XML libraries, such as [3, 4], as a standard universal techniques for events to be recognized and analyzed by the Analyzer effectively. Context of individual events are stored as XML tags which defines various attributes reflecting the situation of an event.

8.3.2 Analysis

The *Analyzer* establishes a context among related events and analyzes the impact and risk associated with them. It (*Assignor*) assigns risk metrics to the transformed events received from the monitor thus, adds more to the context of an event. Risk metrics can be pre-determined or dynamic values, for instance impact, likelihood, reliability or probability, which can be assigned to the event based on its sensitivity and the asset generating it. The *Correlator* relates different events (contexts) coming from different sources and provides an advance context that provision a true picture of the faced threat. This reduces the false-positives and increase security intelligence for the risk analysis and optimized adaptation [27]. Correlation can be achieved using pre-defined rules or using formal modeling techniques such as Bayesian network modeling or Machine Learning techniques.

8.3.3 Adaptation

Risks beyond acceptance are notified to the *Adapter* where a decision to circumvent the anticipated risk and a mitigation action is taken. Events from external security and performance tools, such as QoS monitors, Fault detectors, IDS and vulnerability management, can also be used to enrich the intelligence of security analysis process. The main objective of this function (*Administrator*) is to administer the decision of an optimal security response that reduces the analyzed risk to an acceptable level. It may consult stored adaptation rules or may use established approaches, such as Decision theories. Machine learning techniques can be used to complement the enrichment of existing adaptive and correlation knowledge. The adapter also directs the necessary steps to be taken to mitigate the faced risk. These directions are provided to the *Responder* which formulate them and consult the stored scripts to execute the security adaptation.

8.4 Objectives-Based Evaluation

In our previous study [10] we analyzed various requirements for dynamic risk management, which were formulated based on the nature and needs of IoT-eHealth. To revise, we identified IoT-eHealth as a continuous monitoring service where patient health information is transmitted, analyzed and responded to in a continuous manner. The main data producers in the infrastructure are the body sensors. Another critical resource is the smart device which is the first point of contact data collection, interpretation and (local) analysis for the sensor's data. These two resources are considered to be low-end devices.

Basing these facts, we conclude that IoT-eHealth needs an InfoSec risk management solution which should: assess the risks faced in a *dynamic* and *contextual* manner; its analysis needs to be *lightweight* to accommodate the computational constrains and to ensure immediate analysis and decisions; and that it should provide *autonomous adaptation* to mitigate the risk. In the subsequent sections, the proposed model is evaluated to ensure how it meets these requirements.

8.4.1 Dynamic Assessment

The model is designed as continuous-feedback loop which provides a dynamic and continuous mechanism for monitoring, analyzing and managing system and network behavior [12]. This property is a vital design consideration in self-* systems [14]. In the proposed model, events are collected, filtered, translated, analyzed, stored continuously and decisions on the analyzed events are performed dynamically. The environmental influences on the system are monitored and analyzed continuously and in realtime whereas, system influence (security adaptation) on the environment to mitigate an analyzed risk though, performed in realtime but is done when necessary.

8.4.2 Context Awareness

Context is the information required to characterize the situation of an entity [8]. In IoT-eHealth, an entity could be any of the infrastructural object as well as the users (patients, doctors or healthcare stakeholder in general). In the proposed model, we intend to achieve context awareness by modeling the information collected in Extensible Markup Language (XML). Events collected will be tagged with attributes that will define their situation and impact. When events are generated, there are certain attributes that are logged by the source. These attributes detail questions like who, where, when and what, which characterize the situation of an event. Context will

be represented and stored in XML which will be analyzed by the Analyzer during risk analysis.

Furthermore, a threat faced is an organized collaboration of different events and exploits which are experienced and logged by different sources as it progresses to the actual target(s). Thus, analyzing a risk based on a single event does not reveal the actual risk confronted and may results in false-positive redundancy [27]. It is therefore, necessary to relate events from different sources to provide holistic and contextual information for the risk analysis and to predict or detect the anticipated risks accurately. The correlation engine in the proposed model will correlate events from different sources with different context thus, providing new, advance and more refine context(s) based on reasoning (for instance, stored correlation knowledge) to make the risk analysis process more accurate.

8.4.3 Lightweight Analysis

To achieve fast response in the entire adaptive security process and to accommodate the computational resources of the low-end devices in the infrastructure, we are aiming to introduce lightweight mechanisms into the proposed model. The model is currently in the development phase where we haven't selected any specific methods. However, we intend to use simple tools and techniques, for instance, [3, 4, 5] for common jobs, such as event filtration, parsing, representation and their storage, to give enough time for the actual risk analysis and adaptation. Currently, we are exploring traditional risk metrics formulation as well as lightweight formal approaches to Game, Decision, Utility theories based approaches, such as [19, 20], as they tend to model the dynamic behavior of entities in a conflicting situation.

8.4.4 Autonomous Adaptation

The Adapter component in the proposed model will fulfill two objectives in the context of security adaptation:

1. Enhancing the stored correlation and adaptive knowledge by learning new trends and patterns from risk analysis and mitigation decision (performed by the *administrator*) processes. This will assist in accurate threat prediction, precise risk analysis and optimized mitigation strategies in future adversarial confrontations.

2. Adapting an optimized mitigation response to reduce the negative impact of a currently faced risk. Decision instructions (provided by the *Administrator* to the *Responder*) will be formulated which will trigger the stored scripts to execute the adaptive response. The response formulated can either be a security action for instance, blocking an unse-

cured port or employing a more secure protocol for future communications.

From design perspective we will be using Machine Learning techniques and Decision theories to meet this objective.

8.5 Conclusion & Future Work

IoT enabled eHealth will significantly enhance remote and mobile health monitoring. However, the introduction of IoT will increase the security risk as diverse technologies will be incorporated which can furnish multiple paths of attacks for the adversary. Furthermore, threat sophistication has also increased. Thus, traditional preventive and detective technologies seem to be ineffective to apprehend the risks faced. To overcome this problem, a risk-based adaptive security model is proposed in this paper that provides a continuous, realtime, context-aware assessment and an autonomous and optimized mitigation response to reduce an anticipated risk in IoT-eHealth. Its continuous and context-aware event correlation ensures to capture the threat before they become realistic. Thus, provides a predictive security solution to analyze the unknown threats.

In future, we intend to refine the proposed model which includes, identifying, detecting and categorizing security events in IoT, devising methods for correlating dependent events as well as risk analysis and identifying security metrics upon which the system will adapt. Beside formal methods, such as Game theory, Bayesian modeling and Utility theory, which are the primary design focus, we intend to achieve these objectives using lightweight approaches. Security metrics necessary for adaptation will be explored. An attack-defense case study will be formulated to validate the proposed model and a prototype will be developed on which formal tests and experimentation will be performed.

Acknowledgments

The work presented in this article is a part of the ASSET (Adaptive Security in Smart IoT in eHealth) project. ASSET (2012-2015) is sponsored by the Research Council of Norway under the grant agreement no: 213131/O70. Wishing thanks to the project colleagues and partners for their valuable suggestions and comments.

8.6 Bibliography

[1] Asset - adaptive security for smart internet of things in ehealth. Approve by Research Council of Norway under the grant agreement no: 213131/O70 (2012-2015).

[2] The bsd syslog protocol. RFC 3164: http://www.ietf.org/rfc/rfc3164.txt. Last Accessed on 1 Sept 2013.

[3] Python - fast xml parsing using expat. http://docs.python.org/2/library/pyexpat.html.Last Accessed on 1 Sept 2013.

[4] Python regular expressions library. http://docs.python.org/2/library/re.html. Last Accessed on 1 Sept 2013.

[5] Xml::parser - a perl module for parsing xml documents, 2011. http://search.cpan.org/~toddr/XML-Parser-2.41/Parser.pm Last Accessed on 20 July 2013.

[6] ABIE, H. Adaptive security and trust management for autonomic message-oriented middleware. In *Mobile Adhoc and Sensor Systems, 2009. MASS'09. IEEE 6th International Conference on* (2009), IEEE, pp. 810–817.

[7] ABIE, H., AND BALASINGHAM, I. Risk-based adaptive security for smart iot in ehealth. In *Proceedings of the 7th International Conference on Body Area Networks* (ICST, Brussels, Belgium, 2012), BodyNets '12, pp. 269–275. Available from: http://dl.acm.org/citation.cfm?id=2442691.2442752.

[8] ABOWD, G. D., DEY, A. K., BROWN, P. J., DAVIES, N., SMITH, M., AND STEGGLES, P. Towards a better understanding of context and context-awareness. In *Handheld and ubiquitous computing* (1999), Springer, pp. 304–307.

[9] ALAMPALAYAM, S., AND KUMAR, A. An adaptive and predictive security model for mobile ad hoc networks. *Wireless Personal Communications 29*, 3-4 (2004), 263–281.

[10] AMAN, W., AND SNEKKENES, E. An empirical research on infosec risk management in iot based ehealth. Accepted in: The Third International Conference on Mobile Services, Resources, and Users. MOBILITY 2013, Portugal, August 2013.

[11] ATZORI, L., IERA, A., AND MORABITO, G. The internet of things: A survey. *Computer Networks 54*, 15 (2010), 2787 – 2805. Available from: http://www.sciencedirect.com/science/article/pii/S1389128610001568.

[12] BRUN, Y., SERUGENDO, G. D. M., GACEK, C., GIESE, H., KIENLE, H., LITOIU, M., MÜLLER, H., PEZZÈ, M., AND SHAW, M. Engineering self-adaptive systems through feedback loops. In *Software Engineering for Self-Adaptive Systems*. Springer, 2009, pp. 48–70.

[13] KALITA, H. K., AND KAR, A. Wireless sensor network security analysis. *International Journal of Next-Generation Networks (IJNGN), 1* (December 2009), 1–10.

[14] KEPHART, J. O., AND CHESS, D. M. The vision of autonomic computing. *Computer 36*, 1 (Jan. 2003), 41–50.

[15] MCGRAW, R. W. Risk adaptable access control, 2009. `http://csrc.nist.gov/news_events/privilege-management-workshop/radac-Paper0001.pdf`. Online Accessed on 6 Sept 2013. Available from: `http://csrc.nist.gov/news_events/privilege-management-workshop/radac-Paper0001.pdf`.

[16] MEINGAST, M., ROOSTA, T., AND SASTRY, S. Security and privacy issues with health care information technology. In *Engineering in Medicine and Biology Society, 2006. EMBS '06. 28th Annual International Conference of the IEEE* (2006), pp. 5453–5458.

[17] MUNIR, S., DONGLIANG, X., CANFENG, C., AND MA, J. Mobile wireless sensor networks: Architects for pervasive computing, 2011.

[18] NICOLETT, M., AND KAVANAGH, K. M. Magic quadrant for security information and event management. *Gartner RAS Core Reasearch Note (May 2009)* (2011).

[19] RAJBHANDARI, L., AND SNEKKENES, E. A. Mapping between classical risk management and game theoretical approaches. In *Communications and Multimedia Security* (2011), Springer, pp. 147–154.

[20] RAJBHANDARI, L., AND SNEKKENES, E. A. Using game theory to analyze risk to privacy: An initial insight. In *Privacy and Identity Management for Life*. Springer, 2011, pp. 41–51.

[21] RAMLI RUSYAIZILA, ZAKARIA NASRIAH, S. P. Privacy issues in pervasive healthcare monitoring system: A review. *World Academy of Science, Engineering & Technology 72* (2011), 741.

[22] RIEKE, R., AND STOYNOVA, Z. Predictive security analysis for event-driven processes. In *Computer Network Security*. Springer, 2010, pp. 321–328.

[23] RSA. Rsa adaptive authentication. a comprehensive authentication and risk management platform, 2013. http://www.rsa.com/products/consumer/datasheets/6559_AA_DS_0511.pdf. Online accessed on: 19 July 2013.

[24] SAVOLA, R. M., AND ABIE, H. Metrics-driven security objective decomposition for an e-health application with adaptive security management. In *Proceedings of the International Workshop on Adaptive Security* (New York, NY, USA, 2013), ASPI '13, pp. 6:1–6:8.

[25] SHACKLEFORD, D. Real-time adaptive security. Tech. rep., SANS, December 2008. http://www.sans.org/reading_room/analysts_program/adaptiveSec_Dec08.pdf [Online Accessed on 16 Jul 2013].

[26] WEBER, R. H. Internet of things: New security and privacy challenges. *Computer Law & Security Review 26*, 1 (2010), 23 – 30. Available from: http://www.sciencedirect.com/science/article/pii/S0267364909001939.

[27] WEI, H. A correlation analysis method for network security events. In *Informatics and Management Science III*, W. Du, Ed., vol. 206 of *Lecture Notes in Electrical Engineering*. Springer London, 2013, pp. 269–277.

Article 3

Event Driven Adaptive Security in Internet of Things

Waqas Aman and Einar Snekkenes
In the Eighth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies, pages 7–15, 2014

Chapter 9

Event Driven Adaptive Security in Internet of Things

Abstract

With Internet of Things (IoT), new and improved personal, commercial and social opportunities can be explored and availed. However, with this extended network, the corresponding threat landscape will become more complex and much harder to control as vulnerabilities inherited by individual *things* will be multiplied. Conventional security controls, such as firewalls, intrusion detection systems (IDS) etc., may show some level of resistance to this self-organizing network but, as standalone mechanisms, are not sufficient to analyze the threat in a particular context. They fail to provide the essential context of a threat and yields false positives-negatives which can trigger pointless re-configurations, service unavailability and end user discomfort. Such unwanted events can be very catastrophic, for instance, in an IoT enabled eHealth services. We need to have an autonomous adaptive risk management solution for IoT, which can analyze an adverse situation in a distinct context and manage the risk involved intelligently so that the end user, service and security preferences are well-preserved. This paper details an event driven adaptive security model for IoT to approach the objective specified and explicates how it can be utilized in an eHealth scenario to protect against a threat faced at runtime.

Keywords–Adaptive Security; Internet of Things; Event Correlation; eHealth; Ontology.

9.1 Introduction

According to an analysis conducted by the International Data Corporation (IDC), the IoT expected install base will consist of approximately 212 billion things among which 30.1 billion will be autonomous [13]. Indeed, IoT has the potential to create new huge opportunities for personal, business and social services. However, the research this far is still inconclusive on var-

ious topics, such as standardization, networking, QoS, etc., among which security and privacy are the most challenging [16].

Things carry inherited vulnerabilities and corresponding threats. Physical exposure, user lack of knowledge, unattended management, remote implementation, communicating wirelessly, low resources, etc., are the common weaknesses which are mostly exploited when devices at the edge of the network are attacked. Bringing them to the IoT will make the threat faced more complex and hard to control. Traditional controls, such as IDS, Antiviruses, etc., as standalone measures may provide protection to some level but are limited in providing a clear context of a situation. As a result, false positives and negatives are triggered and create service disruptions, unnecessary changes and sometimes panic [41]. For instance, an IDS trigger a critical alarm that someone is trying a port scan looking for an open File Transfer Protocol (FTP) port and suggest to close that immediately. This might take the administrator to a total panic situation, and he might close the port on the file server without the fact that it is adequately protected by a strong password. Thus, a simple lack of contextual information might yield to service disruption and panic.

An effective way to approach this problem will be to collect the appropriate network and system information (status or any changes), analyze them in a context and decide an action accordingly. This approach is called adaptive security or adaptive risk management. It is the process of understanding, analyzing and reacting to an adverse situation in a particular context [36] and can be seen in a number of proposals, such as, [39][14]. Common problems with these models are, either they focus on only one security service, such as authentication, or provide a generic architecture without detailing the methods used within each architectural component. Also, existing approaches are either focused on threat analysis or adaptation individually. We realize an absence of a model with specific methods to address and connect both analysis and adaptation as a holistic solution to the problem. Hence, we approach these issues as a set of two questions, i.e., *how to monitor and collect security changes in a real time and analyzed them in a specific context?* And, *how can the analyzed information be used to adapt security settings such that user and service preferences are preserved?*

In this paper, we address the first question by utilizing Open Source Security Information Management (OSSIM) [9], which provides a platform to filter and normalize primitive events collected from *things* in the monitored scope. Correlation directives are specified to model adverse situations in which security events are correlated and analyzed in a particular context. The adaptation question is addressed by utilizing a proposed Adaptation Ontology which leverages on the risk information from the event correlation and adapt security settings accordingly. Using the ontology an optimum mitigation action is selected from an action pool in a manner such

that its utility, in terms of usability, QoS and security reliability, is maximum among the possible actions as per user requirements.

The main contribution of this paper is our autonomic security adaptation ontology. OSSIM does not provide such capability and relies on manual reconfigurations which may not address user and service requirements. Also, OSSIM is focused on the traditional computing environment including servers, desktops and corresponding applications where event processing is relatively a common task. This paper extends event driven security to the IoT where environment becomes more complex due to things diversity and mobility for which traditional protocols and tools seem to be inefficient to approach event processing. Hence, the concept of the paper itself can be considered as contribution.

The rest of the paper is structured as follows: In Section 9.2, work related to event monitoring, correlation and adaptation is presented. The proposed Event Driven Adaptive Security model is detailed in Section 9.3. In Section 9.4, an eHealth case study will be presented to show how the model can be utilized to protect against a threat at runtime. Finally, the paper will be concluded in Section 9.5 along with an overview of our near future plan.

9.2 Related Work

The related work is categorized into three major areas of relevance, i.e., event monitoring, event correlation and security adaptation in order to get a clear understanding of the specific methods used.

9.2.1 Event Monitoring

The objective of monitoring is to collect primitive events from various sources in the environment, filter out the unwanted, categorize them into interested areas of investigation, such as authentication, routing, confidentiality, etc., and normalize them to a common language specification for further analysis. In most of the event driven architecture (EDA), this phase is considered to be a typical task yet, requires knowledge of the target system event specification.

9.2.1.1 Event Collection

The two common approaches are agent-based and agent-less collection. An agent is a small additional program that is installed on the monitored source in order to collect and send events or log files remotely [31]. Agents can be customized to accomplish more specific objectives. The agent-less approach does not require any additional component to be installed. Instead, it utilizes built-in protocols and services, such as System Log (Syslog), Windows

Management Instrumentation (WMI), SNMP, etc., to store, access and communicate information at different levels of a monitored system in a standardized manner [17].

One has to address the attributes of flexibility, lightweight, platform independency and management when either of these approaches is adopted. With agent-based, the first three properties can be somehow achieved using expert skills, open source tools and libraries; however, it will be quite a challenge to manage agents across a complex network. The management and control issues can be complex when it comes to a network like IoT. Agentless approach faces the problem of detail customization thus lacks flexibility and might require additional tools for detail diagnosis [31].

Many commercial and open source event analysis tools, such as [9], use mixed strategies to overcome the flexibility and cross-platform issues. However, most of them use third party apps, for instance, [7][5], where updating and controlling is still a matter of discussion. In [31], the author presented an order-based approach which can provide all the mentioned properties by defining a monitoring scope and using system utilities. However, the method applies only to distributed computing environment where diagnosis utilities are supposed to be already in use. The approach apparently shows lacking when considered in the IoT environment where the monitored objects are more likely to be low-end and resource-less sensors.

9.2.1.2 Event Filtering

The objective of event filtering is to discard the redundant or unwanted events [22]. It defines the targeted event scope to be investigated. Filtering is normally achieved using regular expression where a pattern is matched against the collected events. Non matched events are dropped as redundant events. Two important issues that need to be addressed here are: what events are redundant and how to assure minimal information loss during the process? [48].

The authors in [37] explain that event redundancy scope can be defined using two approaches. Temporal filtration can be used to filter out events generated repeatedly over time with the same information. On the other hand, spatial filtration can provide a mechanism to remove similar event reported by a different system within a given time frame, t. They also propose casual filtration where events collected from different sources are removed based on the fact that they may have different syntax but conveys the same semantics.

Threshold values or time frames can be maintained in temporal or spatial filtration techniques to guarantee minimal information loss. Such flags and offsets will ensure that the information contained in the event will not change potentially and will also take into considerations, e.g., compression rates [48][18].

9.2.1.3 Event Classification

Event classification seems to be based on primitive knowledge about events. Every event generated and stored by a source has a unique set of attributes which can be used to classify an event, for instance, see event structures [12][4]. These attributes designate the event source/destination, timestamps, type, user IDs and the event severity level whose ranges changes as per the source event model and specification.

9.2.2 Event Correlation

Correlation is the heart of EDA. It aims to investigate a complex relationship among events and assist to provide enough contextual information to analyze errors, bugs and security threats . Broadly, correlation methods can be classified into two categories, Deterministic and Anomaly-based, either of which can observe events in spatial, temporal or both of the domains [29]. Both the approaches have their associated advantages and disadvantages. Thus, qualifying which of them is a better approach can be determined by evaluating them in a specific application domain [35].

9.2.2.1 Deterministic Approach

In deterministic approach, a predetermined knowledge is utilized to observe and evaluate a given situation. A knowledge base is maintained with application specific information, which is accessed whenever a particular event pattern is matched. So as a fact, a more expert knowledge can analyze a given security threat, problem or situation more precisely. The knowledge itself and the control to it can be characterized in a number of ways as discussed underneath:

Rule-base Correlation

Rule-based event correlation or threat analysis is the most common way to implement deterministic approaches. Most IDS and security event monitoring tools, for instance [9][11][2], uses a rule based correlation to analyze a threat faced. The knowledge is represented in the form of a predefined rule set which dictates defined alarms and alerts when a specific condition during analysis is met.

State Machine Automata based Correlation

Finite State Machine (FSM) is used to study the behavior and state of underlying systems. In the context of event correlation, various defined states for a system behavior (normal and abnormal) are designed and stored as knowledge base as FSM tuples [28]. A runtime diagnosis engine observes

user, application and device behavior and foresees the next system state. Alerts are generated as a flagged state is or about to be triggered. Some of the event correlation models proposed on FSM are [42][46].

The Codebook/Correlation Matrix Techniques

The codebook approach utilizes a symptom-problem relationship. Different suspected events (symptoms) are mapped to their associated abnormal behaviors (problems) and are stored as a knowledge base in a binary matrix, called correlation matrix or a codebook. Events generated are matched against this matrix to identify associated threats or problems. Event correlation models based on codebook techniques can be found in [29].

9.2.2.2 Anomaly-based Approach

Computing and networking environments are very dynamic and the attack vector changes frequently. Some events may not provide certain information and are thus subjected to probabilistic correlation and processing to resolve the uncertainty problem [35]. Unlike predetermined situations in deterministic methods, anomaly-based event correlation aims to identify anomalies without any prior knowledge and can be used to analyze unknown threats. However, they inherit the problems of generating false positive alarms.

Statistical Correlation

As mentioned earlier, events can be filtered, categorized and correlated in both time and space domains to extract rich contextual statistics. For instance, grouping the number of repeated login failure attempts events can provide credible statistics on whether the attempt is a legitimate or that somebody is trying to break-in using a guessing, dictionary or brute force method. High level events, such as alarm/alerts, generated by various security controls, such as IDS, can be used to perform statistical correlation. Statistical information can also be drawn from diverse events having similar attributes/parameters, such as event source, destination, timestamps, etc. Mostly used in anomaly based IDS, these attributes are used as random variables which are later utilized in statistical inferences [25][47].

Probabilistic Modeling

Bayesian networks tend to model relationship among interested random variables. Events can be mapped to random variables. Bayesian model can be illustrated as directed acyclic graphs where nodes represent events of interest and the connecting edges represent the relationships or interdependency between them. The probability of a node (situation or event) is

inferred by utilizing conditional probability assigned to each node (event) in a given network (scenario) [26]. In most cases Bayesian modeling is coupled with other models techniques, such as Hidden Markov Model and Kalman filters, to investigate complex events in depth [29].

9.2.3 Security Adaptation

Assuming that during the analysis an adverse situation or a risk has been discovered, what choices do we have to adapt the security in accordance? How can we utilize the information or context of the analyzed risk to adapt our security? Following is a list of approaches that can be used to answer these questions.

9.2.3.1 Security Policies

Policies remained one of the earliest methods to dictate an action against a given situation. They are a set of rules specifying how a particular situation should be tackled. Edwards et al. in [21] pointed that security policies can be divided into three groups, fixed (e.g., kernel level implementation), customizable (e.g., firewall, router ACLs, etc.) and dynamic, based on the flexibility they offer. Dynamic policies can be detailed on individual user or service level thus providing more flexible adaptation. Some related work include [32][34].

Utility and Probabilistic Models

Utility expresses the measure of efficacy or profit of a choice for a given user or service. In event driven adaptive security, adaptive decisions can be expressed in utilities on the basis of user acceptance, accuracy, power usage, etc. for a given analyzed risk (event). For instance, Alia and Lacosta in [15] used various QoS and security properties corresponding to a required security service to manipulate the utility of an autonomic adaptive response using a non-probabilistic (utility) predictor function. Probabilistic models of utility, such as, [38][19], provides a fair understanding of how security and trust adaptation can be modeled with utilities.

Besides utility theory, probabilistic models such as Bayesian Networks have also been used in a variety of adaptive applications. Bayesian models can be used to select a suitable algorithm from available list [27]. They can also be advantageous in rules discovery [44] to resolve a conflict where an analyzed risk (high level event) two different rules under a given policy [33]. Game theoretic models have also been proposed where intrusion and defense are modeled as games to adapt and defend system security [45][43][30].

Ontologies

Ontologies are used to capture and structure the knowledge about entities, instances and their relationship within an organization. They can be used both for design and runtime purposes [24]. In [23], the author describes an ontology where the knowledge required for security adaptation such as risk, security services and metrics, etc., are related to be assessed at runtime. Denker et. al [20], the authors used security ontologies for annotating functional aspects of electronic resources. However, these ontologies did not discuss how user requirements and preferences should be valued during the adaptation.

9.3 The Model

The model presented, Event Driven Adaptive Security (EDAS), addresses the notion of security adaptation in IoT as an EDA in feedback loop manner. We believe that the basic element of change available within the network is the event generated by various application and devices recorded into log files. They provide a primitive context about *who*, *when*, *where* and *what* of a change and contain vital information, such as timestamps, sources, destinations, user activity, severity levels, etc., necessary to reason about the risk situation associated with an event.

EDAS uses Open Source Security Information Management (OSSIM)[9] which provides a platform for writing scripts, called *plugins*, to filter and normalize primitive security events collected from the monitored sources. Correlation in OSSIM is supported with XML rules through which specific situations, in both temporal and spatial view, can be modeled to correlate and investigated events for potential security risks. The model utilizes a runtime adaptation ontology to adapt a best mitigation action from the available actions based on the stored user and service preferences and risk information produced by the correlation engine. A reference model is shown in Figure 9.1. It includes three major components Monitor, Analyzer and Adaptor. The input, method(s) utilized by individual component along with the details of the output they produced are explained below:

9.3.1 Monitor

The monitor, OSSIM Agent, collects various events (logs) from diverse things in the IoT, filters the unwanted events and normalizes them to a common language for correlation (analysis).

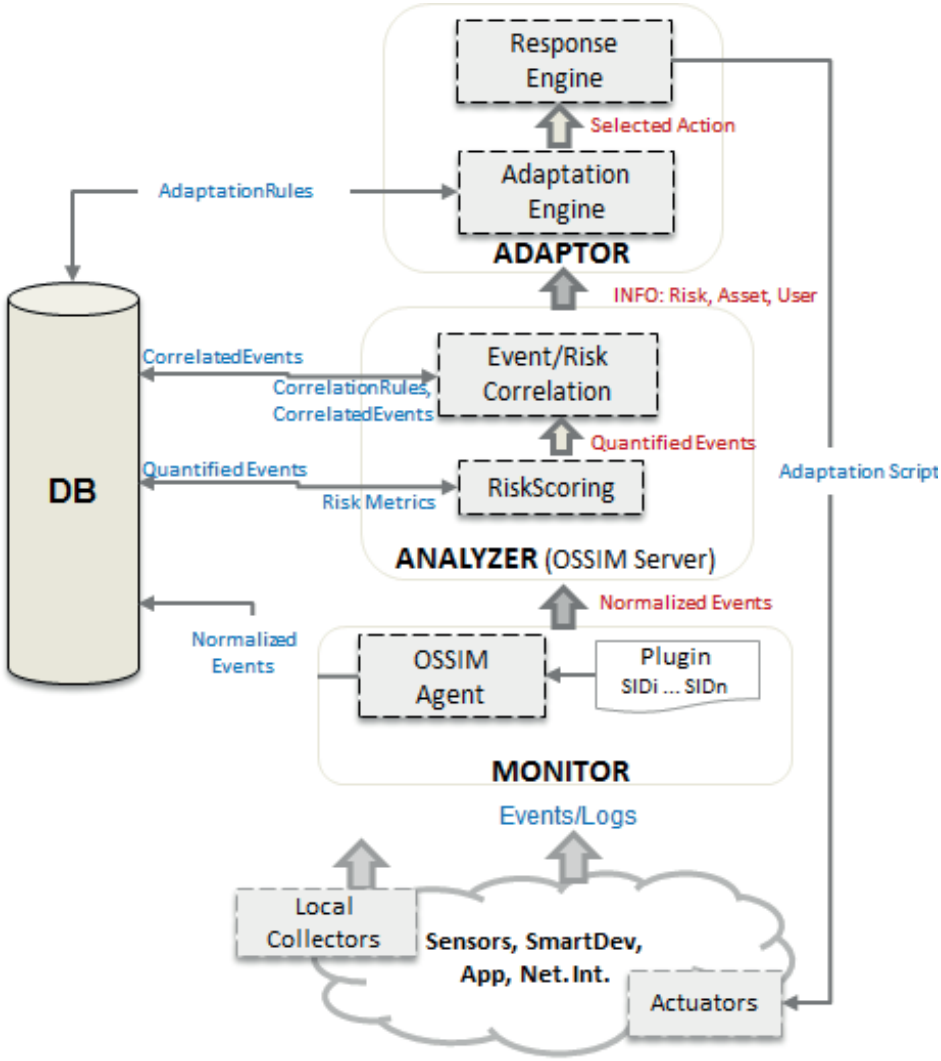


Figure 9.1: Event Driven Adaptive Security-Reference Model

9.3.1.1 Event Collection

Events generated by monitored *things*, e.g., devices, applications, security tools, are collected remotely by the Monitor enabled with OSSIM Agent. Both, agent and agent-less, methods are used to collect methods. OSSIM uses a variety of methods for remote collection including Syslog and SNMP. These two protocols are only used when a device or application supports them otherwise; an agent is installed on the monitored object. OSSIM does recommend some agents, such as Snare [5] and OSSEC [7], which translate

events onto the Syslog stream. However, these agents are not supported by devices at the edge of the network enabling IoT, for instance, smart devices and wireless or body sensors. Thus, we opt for an agent based on MQ Telemetry Transport (MQTT). MQTT is a lightweight M2M messaging transport protocol specifically designed for IoT with platform independence support [6]. The MQTT client hooks onto the event API of the device to collect security events generated and will transport them to the monitor component, the OSSIM Agent, where they are stored in a specific log file.

9.3.1.2 Event Filtration

Security events are extracted using a script, called *Plugin*, designed for individual event source. Writing the script requires some knowledge of the source and the events it is generating. Plugin, identified by a unique ID and other necessary parameters, is a configuration file that dictates from which queue events should be read and which of them needs to be filtered out. OSSIM utilizes a white-listing mechanism where only interested events are sent for further processing. A regular expression specifies these interested events. A match with the expressions is given a unique security ID (SID) which is further used in event correlation. An example plugin configuration is given in Figure 9.2 showing a specific SID corresponding to a login success event. A different SID can be defined for other events, for instance, a login failure event.

9.3.1.3 Event Normalization

Normalization is performed due to the fact that different *things* in the IoT will generate events in different formats. It is, therefore, necessary to transform them into a single common format for correlation and analysis. It is done during SIDs extraction and aims to extract vital attributes of an event transforming them into a common format for correlation. Attributes vary from event to event depending upon the primitive context they carry. In the above example, date and event source IP is normalized into a normalized common format and src_ip respectively.

9.3.2 Analyzer

9.3.2.1 Risk Scoring

Before the normalized events are correlated, they are assigned risk score. OSSIM uses three metrics used for the event (SID) risk quantification [8].

- Asset Value: Specifies the importance of event source or destination within the monitored scope. Ranges from 0-5.
- Priority: Specifies the impact of the event. Ranges from 0-5.

```
[DEFAULT]
plugin_id=1008

[config]
type=detector
enable=yes
source=log
location=/var/log/mydevice.log

[my-device-login-success]
#Apr 2 12:45:12 192.168.5.18 my device:192.168.30.18
login success
event_type=event
regexp="(?P<date>\w{3}\s+\d{1,2}\s\d\d:\d\d:\d\d)\s+(?P
<sensor>\S+)\s+my\device\[\d{1,2}\]:+(?P<src>\d{1,3}\.\d
{1,3}\.\d{1,3}\.\d{1,3})\s+login\success"
date={normalize_date($date)}
sensor= $sensor
plugin_sid=1
src_ip={$src}
...
```

Figure 9.2: Example Plugin

- Reliability: Determines the probability or confidence of the fact the event will corresponds to a compromise. Thus, gives a weight to it false positivity. Reliability ranges from 0-10.

For each event, X, risk is quantified as:
$Risk(X) = (Priority * AssetValue * Reliability)/25$
The division of 25 is made to keep the risk values in the range of 0-10 which reflects the risk level of each event. These values are stored in the DB against each SID and are assigned as they arrive in the Risk Scoring engine. They can be changed as required manually. However, priority and reliability values can take different values automatically during event correlation as per the rules.

9.3.2.2 Event Correlation

The correlation engine investigates normalized events coming from the Monitor. It is done using correlation directives stored in XML. They are triggered when a specific SID is encountered, and thus a new event is generated with a new reliability value. The engine increases and decreases this value with respective to defined attributes within the directive rules. Hence, risk is dy-

namically assessed when SIDs are correlated over time. An SSH login failure example taken (simplified) from OSSIM wiki [10] is given in Figure 9.3.

```
<directive id="500000" name="SSH Brute Force Attack Against DST_IP" priority="4">
  <rule type="detector" name="SSH Authentication failure" reliability="0"
      occurrence="1" from="ANY" to="ANY" port_from="ANY" port_to="ANY"
      plugin_id="4003" plugin_sid="1,2,3,4,5,6,9,10,12,13,14,15,16,20">
      <rules>
         <rule type="detector" name="SSH Successful Authentication (After 1 failed)"
                          reliability="1" occurrence="1"
                          from="1:SRC_IP" to="1:DST_IP"
                          port_from="ANY" time_out="15" port_to="ANY"
                          plugin_id="4003" plugin_sid="7,8"/>
         <rule type="detector" name="SSH Authentication failure (10 times)"
                          reliability="2" occurrence="10" from="1:SRC_IP"
                          to="1:DST_IP"
                          port_from="ANY" time_out="40" port_to="ANY"
                          plugin_id="4003"
                          plugin_sid="1,2,3,4,5,6,9,10,12,13,14,15,16,20"
                          sticky="true"/>
      </rules>

   </rule>
</directive>
```

Figure 9.3: Correlation Directive & Rules

It can be seen that rules can be defined up to *n*-levels of correlation depending upon the requirements. As the level is increased, more precise information is used, such as the time out, occurrence, source and destination, to validate the reliability and context of an event. In the mentioned example, reliability is increased which increases the risk level correspondingly. Similarly, using a rule, reliability during correlation can also be decreased if a login success event (SID) is encountered within the acceptable threshold range of the *occurrence* variable. Also, logical operators can be utilized when certain conditions are to be assured during the correlation.

Event correlation produces high level events which either goes for in-depth correlation or are flagged as alarms to be managed. Alarms are correlated events with risk level above risk acceptance threshold. Information carried by an alarm includes source and destination IDs, the user involved, risk level, threat details and the correlation directive responsible for generating it. This information is utilized during the adaptation process where the confronted risk is mitigated.

9.3.3 Adaptation

In order to utilize the available knowledge precisely and adapt security settings in an optimized manner, we propose an Adaptation Ontology. To be traversed at runtime, the ontology considers all the entities and their relationships necessary for optimal security adaptation. We will be utilizing this entire EDAS model in the IoT enabled eHealth scenario where a patient is remotely managed over the traditional internet or cellular network. To do

so, we establish three different contexts in the proposed ontology as shown in Figure 9.4.

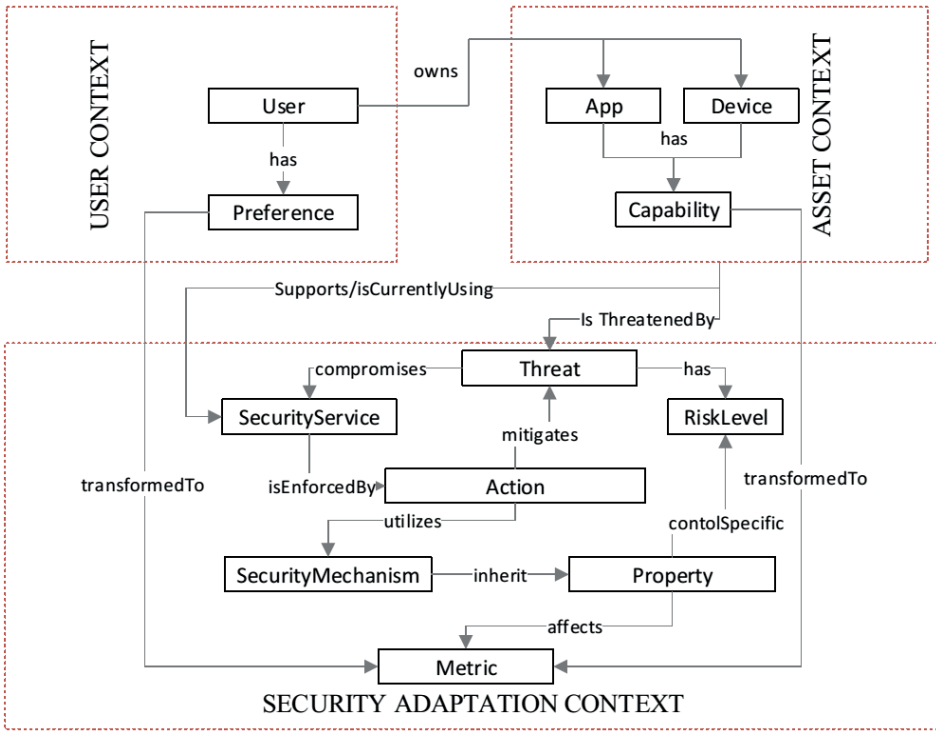


Figure 9.4: Security Adaptation Ontology

- *User Context* corresponds to the patient and medical staff preferences which have to be considered before the adaptation
- Each user owns or utilizes a set of application, such as the eHealth app, Skype for patient-doctor communication, etc. and devices, such as body sensors, smart device or desktop/Laptop, in the scope IoT-eHealth infrastructure. The corresponding information for instance, type, asset value, etc., along with their capabilities is contained in the *Asset Context*.
- The entities and associated settings required for optimized security adaption is grouped under the *Security Adaptation Context*.

An optimal mitigation action is selected from the actions pool following the procedure shown in Figure 9.5. The Response engine articulate a message based on the details of the action provided by the adaptation engine. Using

MQTT transport, the message is sent to an actuator (MQTT Client) installed on the monitored *thing*. The actuator is hooked the specific component API, for instance a login API, and passes the message as variables to be reconfigured.

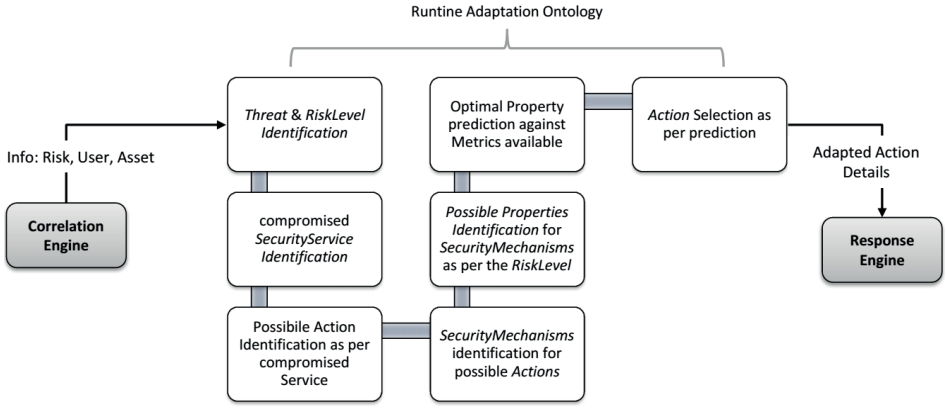


Figure 9.5: Security Adaptation Process

A predictor function chooses the action with maximum utility. Subjective weights are assigned to affected metrics against each property, which correspond the overall utility of the property (to be used in the adapted action) for a specific user. Metrics reflect parameters, such as usability, reliability, service cost, etc., which can be negatively or positively influenced by a security property selection. For the time being, metrics are grouped into three categories, User, QoS and Security, to capture influences concerning user preferences, overall QoS and security reliability. However, we are still exploring metrics and measures, such as described in [40], to make our adaptation process more focused and convincing for user and service requirements besides dealing with security issues. A description of individual entities along with example instances is listed in Table 9.1 whereas, relations among them are detailed in Table 9.2.

9.4 eHealth Case Study

IoT can substantially increase service quality and reduce cost, if enabled in the eHealth paradigm where patient vital signs are remotely diagnosed and managed via internet or cellular network. A number of projects, such as [1][3], aim to investigate different aspects of IoT-eHealth to make it more reliable and convenient. This section describes an IoT-eHealth home scenario in which a patient residing at home, Lynda, is equipped with various body sensors. Her vital signs are monitored through these sensors and are

Table 9.1: Ontology Entities

Context	Entity	Description	Example Instances
User	User	The registered user	Patient, Medical Staff, IT staff
	Preference	User preferences that affects or are affected by the adaptation decision	App/device usage knowledge, Current Health Status, Location, Environmental Context, etc.
Asset	App	Any soft components used in the IoT-eHealth infrastructure	eHealth app, communication software such as Skype, email, Security tools, etc.
	Device	Any hard components used to send receive and store User information	Body Sensors, Smart phones, Tablets, Laptops, Desktops
	Capability	The resources offered by individual **Asset**	Battery life time, CPU power, Memory, Supported Protocols etc.
Security Adaptation	SecurityService	The **security services** supported/Currently used by each **Asset**	e.g., Authentication, Encryption and Integrity modules
	RiskLevel	Event/Alarm Risk Level (analyzed by the event correlation/analysis engine) which threatens a **SecurityService** and **Asset**	Range(0-10)
	Threat	Threat information dictated by Correlation Directive	Brute Force, DoS, etc.
	Action	A list of adaptation actions (options) associated with a given **SecurityService**. Actions enforces a specific **SecurityService** in order to control a **Threat** faced	Changing Password, Locking a user for a specific time, changing encryption methods, Adapting a secure authentication protocol, etc.
	SecurityMechanism	Methods/algorithms associated with a given **Action** which are utilized in order to enforce a **SecurityService** challenged by a **Threat**	WEP, WP2, DES, AES, Captcha, SHA1, Disabling User Account etc.
	Property	Available attributes of a specific **SecurityMechanism** which can be adjusted for adaptation	AES (key length), Password (length, character type), captcha (image, audio), Account Locking time (seconds, minutes)
	Metric	Factors affecting security adaptation. Derived from user **Preferences**, device **capabilities** and the overall security against a given **Property** in terms of expected utilities.	Usability, PowerCost, ExecutionTime, ServiceLevelCost, Reliability, etc.

transmitted over a Wifi or cellular network to remote hospital site for further diagnosis. She frequently uses her smart phone, part of this infrastructure, installed with an eHealth app to keep track of health status as well as for billing payments besides personal use. We intend to explicate how our model fits into this scenario to defend against a security threat faced.

Home Scenario–Authentication: Lynda wants her credentials saved in the eHealth app to be protected. The app installed on her smart phone is protected with a password that is used to protect her credit card credentials, billing information and local Patient Health Information (PHI).

Table 9.2: Ontology Relations

Context	Relation	Classes Involved	Example Relations
User	has	User, Preference	Patient has a Preference of having easy to remember credentials Patient prefers service over security while being outside home Doctor prefers strict confidentiality while being outside hospital
User, Asset	owns	User, Asset	Patient owns a tablet to read his vital signs Patient owns (wears) ECG sensor Doctor owns a desktop machine to communicate with Patient over Skype
Asset	has	App, Device, Capability	Patient tablet has DualCore processor installed eHealth app installed on patient tablet has a medium level password ECG sensor does not support DES 128 bit algorithm Smart phone has 1 hour of talk time left
Asset, Security Adaptation	Supports, Currently Using	Asset, SecurityService	ECG Sensor supports/currentlyUsing Confidentiality, Authentication
	IsThreatenedBy	Asset, Threat	eHealth app is threatened by a password brute force attack In home Wifi network is threatened by DeAuth flooding
Security Adaptation	compromises	Threat, SecurityService	Password Brute force compromises eHealth app Authentication WifiDeAuth flooding compromises network integrity
	has	Threat, RiskLevel	Password Brute force on eHealth app has a HIGH Risk Level
	isEnforcedBy	SecurityService, Action	eHealth App is authentication is enforced by a medium strength password Wifi Network authentication is enforced by WPA policy
	mitigates	Action, Threat	Changing user password mitigates a password brute force threat Restricting user login attempts to t-seconds mitigates a password brute force
	utilizes	Action, SecurityMechanism	A password change action utilizes the password length & complexity Restricting user login attempts utilizes the time limit Increase encryption level action utilizes AES
	Inherit	SecurityMechanism, Property	Password length inherit the property of 6, 8 or 10 characters Password complexity inherit the property of character type
	controlSpecific	Property, RiskLevel	A password with 6 digit key length controls LOW level brute force attempts A password with 10 digit key length controls HIGH level brute force attempts
	affects	Property, Metric	10 character password affects (decreases) usability and (increase) security reliability 3G network affects (increases) Service Quality and (decreases) device battery
User, Security Adaptation	transformedTo	Preference, Metric	User preferences are transformed to Usability User location is transformed to QoS, Security \& Privacy attributes
Asset, Security Adaptation		Capability, Metric	Supported protocols (can be) transformed to QoS and Security metrices

Adverse Situation: An insider having access to Lynda's smart phone with the intention of stealing her credit card information is trying to login into the eHealth app by guessing different passwords repeatedly.
Preferences: Lynda prefers medium level password instead of a complex one. She does not want her account to be locked out as she has to check her diabetes level frequently.

A generalized message sequence of the whole adaptation process as per the scenario is given in Figure 9.6. The defense against the situation is detailed as follow:

Model Go-Through – The Runtime Defense:

Event Collection & Monitoring: Smart phone login failure events will be collected by the MQTT client and will be sent to the Monitor. Plugin, e.g., pluginID=20, specified for the smart phone will read these events on the OSSIM Agent. The *login failure on eHealth App* SID, with SID=3, will extract and normalize the important attributes such as timestamps, user, source, and will add other attributes, such as the number of attempts made.

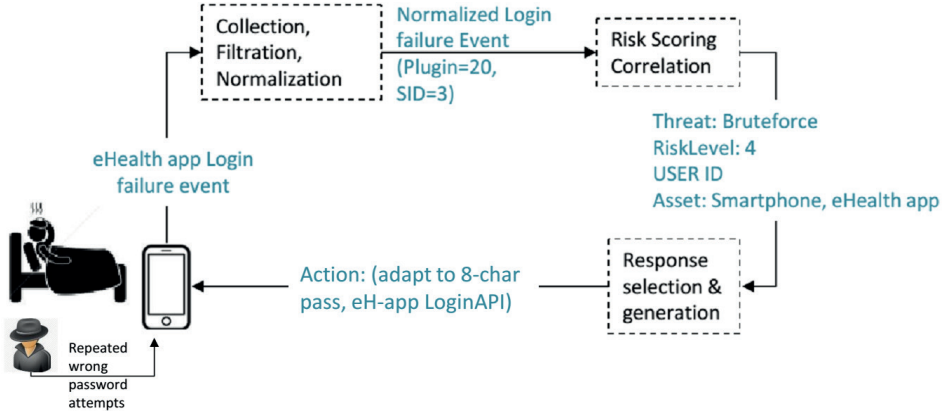


Figure 9.6: Attack-Defense Case Study Message Diagram

Risk Quantification: Considering the risk acceptance level for repeated login failure is 4 let the smart phone be a critical asset, so Asset Value=5. To give space to for the accidental wrong attempts, let the Reliability=0 for the first encounter and suppose the importance of the event is considerable so, Priority=5.

Event Correlation: The correlation directive shown in Figure 9.7 specifies 3 levels of correlation. The first wrong attempt is considered as normal so Reliability is not increased. For the next 5 wrong attempts, Reliability is increased to 2 and the engine waits for 10 seconds as a time out. Risk, as per the equation stated earlier, at this stage becomes 2. Similarly, after 6 wrong repeated attempts Reliability is increased to 3 and so does the associated risk level. Finally, an alarm will be generated a risk of level 4 is raised after consecutive 20 attempts when Reliability is increased to 4. Risk is assessed dynamically and instances of the same events are correlated over a period of time as context becomes more evident.

Security Adaptation: Proceeding logically with the procedure shown in Figure 9.5. An optimal mitigation action can be selected as:

```
<directive id="100" name="Password Brute Force against DST_IP" priority="5">
     <rule type="detector" name="eHealth APP Login failure" reliability="0"
        occurrence="1" from="ANY" to="ANY" port_from="ANY" port_to="ANY"
            plugin_id="20" sid="3">
            <rules>
                 <rule type="detector" name="eHealth App Successful Login (After 1 failed)"
                    reliability="2" occurrence="1"
                    from="ANY" to="DST_IP"
                    port_from="ANY" time_out="10" port_to="ANY"
                    plugin_id="20" plugin_sid="3"/>
                <rule type="detector" name="eHealth App Login failure (6 times)"
                    reliability="3" occurrence="6" from="ANY"
                    to="DST_IP"
                    port_from="ANY" time_out="40" port_to="ANY"
                    plugin_id="20"
                    sid="3" />
               <rule type="detector" name="eHealth App Login failure (20 times)"
                    reliability="4" occurrence="20" from="ANY"
                    to="DST_IP"
                    port_from="ANY" time_out="60" port_to="ANY"
                    plugin_id="20"
                    sid="3" />
            </rules>

    </rule>
</directive>
```

Figure 9.7: Correlation Directive & Rules for Repeated Login Failures

- *Threat & Risk Level:* Password Brute Force
- *Compromised Security Service:* Authentication
- *Possible Actions:* Suppose, Password Change, Account Lockout & Enforcing Captcha
- *Security Mechanisms:* As per each action, Password Change (keyLength), Enforcing Captcha (Captcha), Account Lockout (Time Restriction)
- *Security Properties Metrics & Utilities:* As a hypothesis, consider Table 9.3 showing the affected metrics by individual properties with associated utilities (ranging from 1-10). The properties listed are considered to mitigate risk level 4 or above for password brute force attempts on the smart phone. Furthermore, it is assumed that the utilities are assigned as per service and user preferences.

The predictor function will identify that the optimal action to circumvent this threat is to change the password on the smart phone eHealth app to an 8-characters. If it is already in use, it will go back and select the second best option. The selected action along with the user, concerned API and asset details will be given to the Response engine which will send a message containing the instructions as appropriate variables to the MQTT client residing on the smart phone as an actuator. The actuator will identify the API

Table 9.3: Properties, Metrics & Utilities

Metric	PROPERTIES					
	KeyLength		**Captcha**		**Time Restriction**	
	8-char,	10-char.	Audio	Visual	15 min.	30 min.
Usability	8	5	6	7	6	3
QoS	8	7	5	5	6	6
Reliability	7	8	4	4	7	8
Total Utility	23	20	15	16	19	17

mentioned and will pass the message variable. The API will implement the changes and will ask the user/adversary to enter a new 8-character password based on the older one.

9.5 Conclusion & Future Work

Existing detective and preventive controls as individual components seems to be inefficient in providing the required context to investigate security threats. We presented an event driven adaptive security model, EDAS, which leverages the capabilities of existing event models of diverse things in IoT and OSSIM correlation to adapt security settings by keeping the user and service utility at maximum. Primitive knowledge about security changes is collected and is analyzed in a definitive and established security context. The runtime adaptation ontology provides a structured knowledge of all the elements necessary to select appropriate mitigation action as user and service preferences. MQTT as a transport mechanism for the collection and actuation processes makes the model more extendable, platform independent and cost effective.

In the near future, we intend to develop a prototype for EDAS to test its processes as a real world IoT-eHealth artifact. Preliminary plans are to investigate the overall reliability, service response timings and building universal collectors and actuators for devices at the network edge, such as body sensors and personal smart devices. The prototype will be validated with confidentiality, availability, integrity and mobility scenarios as they are deemed to be the most critical aspects in remote patient management systems.

Acknowledgements

The work presented in this article is a part of the ASSET (Adaptive Security in Smart IoT in eHealth) project. ASSET (2012-2015) is sponsored by the Research Council of Norway under the grant agreement no: 213131/O70.

9.6 Bibliography

[1] Asset - adaptive security for smart internet of things in ehealth. Last access date: 31 May 2014. Available from: http://asset.nr.no/asset/index.php/ASSET_-_Adaptive_Security_for_Smart_Internet_of_Things_in_eHealth.

[2] The bro network security monitor. Last access date: 31 May 2014. Available from: https://www.bro.org.

[3] Butler ubiquitous, secure internet-of-things with location and contex-awareness. Last access date: 31 May 2014. Available from: http://www.iot-butler.eu/.

[4] Event schema elements (windows). Last access date: 31 May 2014. Available from: http://msdn.microsoft.com/en-us/library/windows/desktop/aa384367(v=vs.85).aspx.

[5] InterSect alliance - snare agents. Last access date: 31 May 2014. Available from: http://www.intersectalliance.com/snareagents/index.html.

[6] Mq telemetry transport, mqtt. Last access date: 31 May 2014. Available from: http://mqtt.org/.

[7] OSSEC: open source SECurity. Last access date: 31 May 2014. Available from: http://www.ossec.net/.

[8] Ossim risk calculation. Last access date: 31 May 2014. Available from: https://www.alienvault.com/wiki/doku.php?id=user_manual:dashboards:risk:risk_metrics#risk_calculation.

[9] Ossim: the open source siem. Last access date: 31 May 2014. Available from: http://www.alienvault.com/open-threat-exchange/projects.

[10] Ossim-writing correlation directives. Last access date: 31 May 2014. Available from: https://www.alienvault.com/wiki/doku.php?id=user_manual:intelligence:writing_correlation_directives.

[11] Snort : Open source IDS/IPS. Last access date: 31 May 2014. Available from: http://www.snort.org.

[12] System logger: Syslog linux man page. Last access date: 31 May 2014. Available from: http://linux.die.net/man/3/syslog.

[13] The internet of things is poised to change everything, says international data corporation. Press Release, October 2013. Last access date: 31 May 2014. Available from: `http://www.idc.com/getdoc.jsp?containerId=prUS24366813`.

[14] ABIE, H. Adaptive security and trust management for autonomic message-oriented middleware. In *Mobile Adhoc and Sensor Systems, 2009. MASS'09. IEEE 6th International Conference on* (2009), IEEE, pp. 810–817.

[15] ALIA, M., AND LACOSTE, M. A QoS and security adaptation model for autonomic pervasive systems. In *Computer Software and Applications, 2008. COMPSAC '08. 32nd Annual IEEE International* (July 2008), pp. 943–948.

[16] ATZORI, L., IERA, A., AND MORABITO, G. The internet of things: A survey. *Computer Networks 54*, 15 (2010), 2787–2805.

[17] BELLAVISTA, P., CORRADI, A., AND STEFANELLI, C. Java for on-line distributed monitoring of heterogeneous systems and services. *The Computer Journal 45* (2002), 595–607.

[18] BUCKLEY, M. F., AND SIEWIOREK, D. P. A comparative analysis of event tupling schemes. In *Fault Tolerant Computing, 1996., Proceedings of Annual Symposium on* (1996), IEEE, pp. 294–303.

[19] CHENG, S.-W., GARLAN, D., AND SCHMERL, B. Architecture-based self-adaptation in the presence of multiple objectives. In *Proceedings of the 2006 international workshop on Self-adaptation and self-managing systems* (2006), ACM, pp. 2–8.

[20] DENKER, G., KAGAL, L., AND FININ, T. Security in the semantic web using owl. *Information Security Technical Report 10*, 1 (2005), 51–58.

[21] EDWARDS, W. K., POOLE, E. S., AND STOLL, J. Security automation considered harmful? In *Proceedings of the 2007 Workshop on New Security Paradigms* (New York, NY, USA, 2008), NSPW '07, ACM, pp. 33–42.

[22] ETZION, O., AND NIBLETT, P. *Event Processing in Action*, 1st ed. Manning Publications Co., Greenwich, CT, USA, 2010.

[23] EVESTI, A., AND OVASKA, E. Ontology-based security adaptation at run-time. In *Self-Adaptive and Self-Organizing Systems (SASO), 2010 4th IEEE International Conference on* (Sept 2010), pp. 204–212.

[24] EVESTI, A., OVASKA, E., AND SAVOLA, R. From security modelling to run-time security monitoring. *Security in Model-Driven Architecture* (2009), 33–41.

[25] GARCA-TEODORO, P., DAZ-VERDEJO, J., MACI-FERNNDEZ, G., AND VZQUEZ, E. Anomaly-based network intrusion detection: Techniques, systems and challenges. *Computers & Security 28*, 12 (Feb. 2009), 18–28.

[26] GAUDIN, B., NIXON, P., BINES, K., BUSACCA, F., AND CASEY, N. Model bootstrapping for auto-diagnosis of enterprise systems. In *International Conference on Computational Intelligence and Software Engineering, 2009. CiSE 2009* (Dec. 2009), pp. 1–4.

[27] GUO, H. A bayesian approach for automatic algorithm selection. In *IJCAI 2003 Workshop on AI and Autonomic Computing, Mexico* (2003), Citeseer, pp. 1–5.

[28] HOPCROFT, J. E., MOTWANI, R., AND ULLMAN, J. D. *Introduction to automata theory, languages, and computation*. Addison-Wesley, 2001.

[29] JIANG, G., AND CYBENKO, G. Temporal and spatial distributed event correlation for network security. In *American Control Conference, 2004. Proceedings of the 2004* (June 2004), vol. 2, pp. 996–1001 vol.2.

[30] JIANG, W., FANG, B.-X., ZHANG, H.-L., TIAN, Z.-H., AND SONG, X.-F. Optimal network security strengthening using attack-defense game model. In *Information Technology: New Generations, 2009. ITNG'09. Sixth International Conference on* (2009), IEEE, pp. 475–480.

[31] KUFEL, L. Security event monitoring in a distributed systems environment. *IEEE Security Privacy 11*, 1 (Jan. 2013), 36–43.

[32] KULKARNI, D., AND TRIPATHI, A. Context-aware role-based access control in pervasive computing systems. In *Proceedings of the 13th ACM symposium on Access control models and technologies* (2008), ACM, pp. 113–122.

[33] LUPU, E., AND SLOMAN, M. Conflict analysis for management policies. In *Integrated Network Management V* (1997), Springer, pp. 430–443.

[34] MALIKI, T. E., AND SEIGNEUR, J.-M. A security adaptation reference monitor (SARM) for highly dynamic wireless environments. In *Proceedings of the 2010 Fourth International Conference on Emerging Security Information, Systems and Technologies* (Washington, DC, USA, 2010), SECURWARE '10, IEEE Computer Society, pp. 63–68.

[35] MARTIN-FLATIN, J. P., JAKOBSON, G., AND LEWIS, L. Event correlation in integrated management: Lessons learned and outlook. *Journal of Network and Systems Management 15*, 4 (Dec. 2007), 481–502.

[36] NICOLETT, M., AND KAVANAGH, K. M. Magic quadrant for security information and event management. *Gartner RAS Core Reasearch Note (May 2009)* (2011).

[37] OLINER, A., AND STEARLEY, J. What supercomputers say: A study of five system logs. In *37th Annual IEEE/IFIP International Conference on Dependable Systems and Networks, 2007. DSN '07* (June 2007), pp. 575–584.

[38] QUERCIA, D., AND HAILES, S. MATE: mobility and adaptation with trust and expected-utility. *International Journal of Internet Technology and Secured Transactions 1*, 1 (2007).

[39] RSA. Rsa adaptive authentication. a comprehensive authentication and risk management platform, 2013. Accessed on: 31 May 2014. Available from: `http://www.emc.com/collateral/data-sheet/h11429-rsa-adaptive-authentication-ds.pdf`.

[40] SAVOLA, R. M., AND ABIE, H. On-line and off-line security measurement framework for mobile ad hoc networks. *Journal of Networks 4*, 7 (2009).

[41] SHACKLEFORD, D. Real-time adaptive security. Tech. rep., SANS, December 2008. Last Accessed on 4 April 2014. Available from: `http://www.sans.org/reading_room/analysts_program/adaptiveSec_Dec08.pdf`.

[42] SIFALAKIS, M., FRY, M., AND HUTCHISON, D. Event detection and correlation for network environments. *IEEE Journal on Selected Areas in Communications 28*, 1 (Jan. 2010), 60–69.

[43] SIMMONS, C. B., SHIVA, S. G., BEDI, H. S., AND SHANDILYA, V. ADAPT: a game inspired attack-defense and performance metric taxonomy. In *Security and Privacy Protection in Information Processing Systems*. Springer, 2013, pp. 344–365.

[44] STERRITT, R. Autonomic networks: engineering the self-healing property. *Engineering Applications of Artificial Intelligence 17*, 7 (2004), 727–739.

[45] STIBOREK, J., GRILL, M., REHAK, M., BARTOS, K., AND JUSKO, J. Game theoretical adaptation model for intrusion detection system. In *Advances on Practical Applications of Agents and Multi-Agent Systems*. Springer, 2012, pp. 201–210.

[46] TAN, J., PAN, X., KAVULYA, S., GANDHI, R., AND NARASIMHAN, P. SALSA: analyzing logs as StAte machines. *WASL 8* (2008), 6–6.

[47] YE, N., MEMBER, S., EMRAN, S. M., CHEN, Q., AND VILBERT, S. Multivariate statistical analysis of audit trails for host-based intrusion detection. *IEEE Transactions on Computers 51* (2002), 810–820.

[48] ZHENG, Z., LAN, Z., PARK, B.-H., AND GEIST, A. System log preprocessing to improve failure prediction. In *IEEE/IFIP International Conference on Dependable Systems Networks, 2009. DSN '09* (June 2009), pp. 572–577.

Article 4

EDAS: An Evaluation Prototype for Autonomic Event Driven Adaptive Security in the Internet of Things

Waqas Aman and Einar Snekkenes
In Future Internet Journal , 7(3). Pages 225–256, July 2015

Chapter 10

EDAS: An Evaluation Prototype for Autonomic Event Driven Adaptive Security in the Internet of Things

Abstract

In Internet of Things (IoT), the main driving technologies are considered to be tiny sensory objects. These objects cannot host traditional preventive and detective technologies to provide protection against the increasing threat sophistication. Furthermore, these solutions are limited to analyzing particular contextual information, for instance network information or files, and do not provide holistic context for risk analysis and response. Analyzing a part of a situation may lead to false alarms and later to unnecessary and incorrect configurations. To overcome these concerns, we proposed an event-driven adaptive security (EDAS) model for IoT. EDAS aims to observe security events (changes) generated by various things in the monitored IoT environment, investigates any intentional or unintentional risks associated with the events and adapts to it autonomously. It correlates different events in time and space to reduce any false alarms and provides a mechanism to predict attacks before they are realized. Risks are responded to autonomically by utilizing a runtime adaptation ontology. The mitigation action is chosen after assessing essential information, such as the risk faced, user preferences, device capabilities and service requirements. Thus, it selects an optimal mitigation action in a particular adverse situation. The objective of this paper is to investigate EDAS feasibility and its aptitude as a real-world prototype in a remote patient monitoring context. It details how EDAS can be a practical choice for IoT-eHealth in terms of the security, design and implementation features it offers as compared to traditional security controls. We have explained the prototype's major components and have highlighted the key technical challenges.

Keywords–Internet of Things; adaptive security; eHealth; event-driven architecture; risk management; event correlation.

10.1 Introduction

A recent report by EMC and the International Data Cooperation (IDC) details that by the year 2020, 30 billion of the connected devices in the world, constituting IoT, will be small and smart wireless devices characterized by their autonomous behavior [20]. They further reason that IoT will create new business models, capture real-time information in critical infrastructure, extend services offered by traditional modes and will provide a global visibility platform. Analyzing this report and other similar research, one can conclude that IoT can bring phenomenal extensions to the ICT-based services offered in the business, as well as in public domains.

The potentials offered by IoT can become more efficacious and reliable if other challenges it faces, such as networking, standardization, security and privacy issues, are carefully sorted out [16]. From a security perspective, a recent study made by OWASP and HP® [8, 49] details a number of serious vulnerabilities that IoT has still to address. The report highlights that 60% of the things web interfaces are prone to web-related attacks, such as cross-site scripting (XSS) attacks; 90% of the things collect at least one piece of personal information; 70% of the devices communicate via unencrypted channels; and 70% of the devices are susceptible to account enumeration attacks. These are some severe concerns particularly for IoT-enabled health services, where the type of information communicated is mostly personal.

Current security solutions, like firewalls, intrusion detection systems (IDS), *etc.*, or even small anti-virus programs are not feasible for this sensory, tiny, low-resourced thing-driven network. Even if we somehow tailored a miniature version of them, they still would not achieve much. Because, as standalone mechanisms, they can only assess a particular set of vectors when a host is under a possible compromise situation. A security breach usually consists of multiple and associated attack vectors, means and targets. Analyzing only a part of them independently of their association in a context may yield false positives and negatives [48]. Analyzing risks irrespective of the context may further results in inappropriate mitigation decisions. Consider a scenario where a physician, currently on holiday, using her smartphone is given authorization by a role-based access control (RBAC) system to access patient personal information from an unusual place on a weekend. From an RBAC point of view, this activity seems to be legitimate, and the system should grant access. However, if the entire context, *i.e.*, the unusual place, current status and access time, is analyzed, one can conclude that there is a risk involved if access is granted, *i.e.*, the smartphone might have been compromised. Hence, current preventive and detective security solutions are either not feasible or do not have the intelligence to assess the situation in a holistic context.

In social services enabled by IoT, for instance eHealth concerning remote patient monitoring, mitigation decisions based on inaccurate risk informa-

tion may disrupt service availability and can be life threatening. Examining the stated facts, statistics and resource-constrained nature of things in IoT, we proposed an event-driven adaptive security architecture (EDAS) [15] in an eHealth scenario where remote patients are continuously monitored. EDAS is an autonomous risk-based adaptive security architecture that monitors and analyzes security events generated by things for potential threats and responds to the corresponding risks by adapting an optimal mitigation action. The mitigation response is selected in a way such that user preferences, service and security requirements are all assessed before a decision is made. For autonomous adaptation, we proposed a runtime security ontology containing the necessary knowledge for optimized adaptation. For event monitoring, collection, analysis and correlation, we listed a variety of techniques that can be utilized. The threat and risk analysis routines are suggested to be performed at a resourceful remote machine to avoid heavy computations on a resource-constrained thing. The things only use their out-of-the-box event framework to generate and communicate events. Events are collected and analyzed to investigate any potential threat that they pose and are correlated in a context to reduce any false positive or negatives.

This paper describes an IoT-eHealth prototype showing how EDAS can be implemented as a real-world artifact. The primary objective is to investigate the feasibility of EDAS as an event-driven security architecture and whether its adaptation control loop adds value to the security of IoT. From the prototyping activity, evaluation and comparison detailed in this paper, we conclude that EDAS can be a more practical choice for IoT security as compared to traditional security controls. It provides a holistic security solution, complies with the resource-constrained nature of things in IoT, provides extensibility, allows existing traditional systems to be monitored and substantially increases the overall throughput of electronic security operations. We detail the prototype's major components individually and how they are utilized collectively to ensure adaptive security. Furthermore, this paper highlights the key technical challenges and discusses how they can be approached. For demonstration purposes, we have adopted OSSIM [1] as an event monitoring and analysis tool. However, we suggest that any appropriate statistical, probabilistic, rule-based or other methods, tools or techniques can be utilized for event monitoring and correlation in the architecture, as long as it has complex event processing (CEP) [36] capabilities. We have already categorized these techniques in [15].

The rest of the paper is organized as follows. In Section 10.2, an overview of the proposed architecture will be briefly revised. Section 10.3 explains the major components of the prototype and describes their technical design and features. A case study demonstrating the prototype will be presented in Section 10.4. The prototype feasibility will be argued and detailed in Section 10.5 to evaluate the architecture aptitude. Related work is discussed in

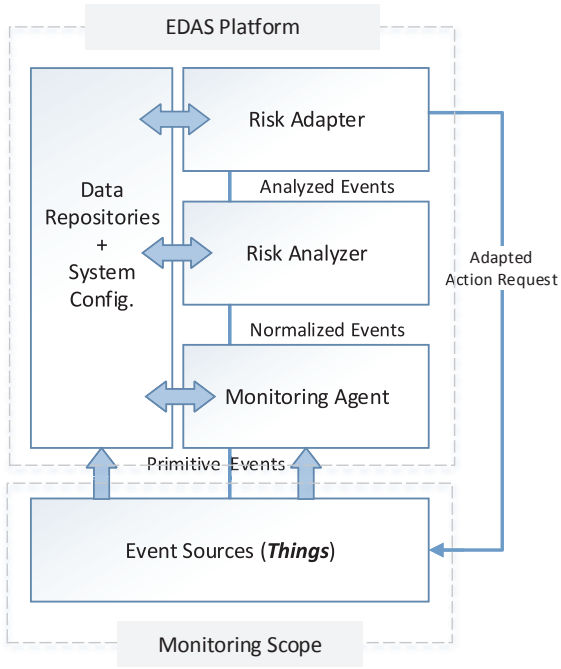


Figure 10.1: Abstract diagram of event-driven adaptive security (EDAS) architecture.

Section 10.6. In Section 10.7, key challenges concerning EDAS development, possible solutions and further work will be discussed. Finally, the paper will be concluded in Section 10.8.

10.2 Proposed Architecture

EDAS [15] is an autonomous risk-based event-driven adaptive security architecture for IoT. It monitors security changes, *i.e.*, thing-generated events, in the IoT environment, analyzes the associated threat(s) and adapts appropriate and optimized security configurations against the risk faced. At the abstract level, EDAS complies with the IBM autonomic control loop that consists of the sensing, analyzing, planning and execution components [24]. It consists of two major components: The EDAS platform and the event sources, as shown in Figure 10.1.

The event sources reflect the monitored environment consisting of all of the critical things in IoT, *i.e.*, the applications, devices, objects, *etc.*, that are crucial for service delivery. Thus, the scope of the monitored things defines the risk management scope for adaptive security. In a typical IoT-eHealth scenario, as shown in Figure 10.2, event sources correspond to all of the

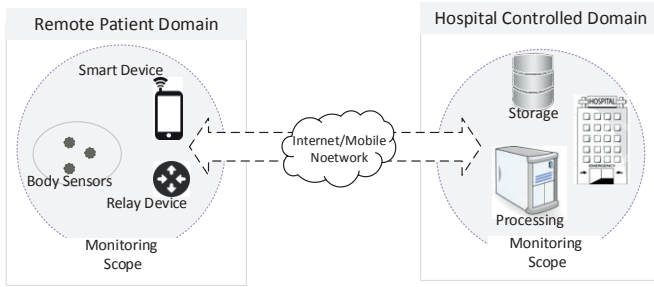


Figure 10.2: IoT-eHealth environment.

applications, sensors, smart devices and actuators, both in the patient and hospital domains, that are essential for the reliable, secure and efficacious operations of remote patient monitoring services.

In the proposed architecture context, the event source is any of the mentioned things that can react to any change or event it experiences within itself or in the environment in which it operates. Reaction refers to triggering appropriate actions and processes, generating or logging information that detail the actions taken and communicating these changes and information across the network. This elaboration makes our event source (thing) description more aligned with that of the Cluster of European Research Projects on the IoT (CERP-IoT). According to CERP [51], "things are active participants in business, information and social processes where they are enabled to interact and communicate among themselves and with the environment by exchanging data and information sensed about the environment, while reacting autonomously to the real/physical world events and influencing it by running processes that trigger actions and create services with or without direct human intervention".

The EDAS platform consists of a set of methods and tools necessary to continuously monitor and analyze these thing-generated primitive events in a context-aware manner to investigate any potential security threats and associated risks. As events arrive from different sources in different formats, they are filtered and normalized by the *Monitoring Agent* to remove any redundant events and to shape them into a universal format for risk analysis and adaptation. The *Risk Analyzer* investigates potential threats and risks associated with these events using a correlation engine, such that false alarms are avoided. This is ensured by correlating events in space and time in a context [33]. An unacceptable risk and its corresponding details are referred to the *Risk Adapter* where an adaptation engine utilizes a runtime security ontology to select a mitigation action from a pool of possible actions to reduce the risk impact. The selected action is sent to the local adaptor process embedded in the thing (event source) where the new security settings received

are applied. The effects of the adapted changes are also recorded, monitored and analyzed again. Hence, security monitoring, analysis and adaptation is done in a continuous autonomic control loop fashion. Necessary information regarding event correlation, risk quantification, system configurations, adaptation requirements, normalization scripts and event databases is accessed and updated along this process via the repository component.

10.3 EDAS Prototype Specifications

We have developed an evaluation prototype to investigate the feasibility of the EDAS architecture as a real-world implementation. The test settings were designed to reflect a working IoT-eHealth environment where a remote patient with wearable body sensors is monitored from a hospital site. The test environment consists of the following hardware and software components.

At the remote patient domain:

- Libelium open source eHealth Sensor Shield V.2.0
- Arduino Uno R2: A 16-MHz 32 K micro-controller for the eHealth Shield
- RN-XV 171 IEEE 802.11 b/g-compatible Wifi module
- XBee Communication Shield for communicating Arduino serial data over Wifi
- Samsung Galaxy SIII-Mini as a relay device

At the hospital-controlled domain:

- MySQL 5.6.12 as a storage for primitive medical and security events
- Apache 2.4.4 to display real-time patient vital signs enabled with High-Charts API
- Sixty four-bit OSSIM V.4.6.1 as the EDAS monitoring and analysis platform
- A Jena-SPARQL-enabled JAR as an adaptation engine for accessing, modifying and storing the RDF /XML adaptation ontology

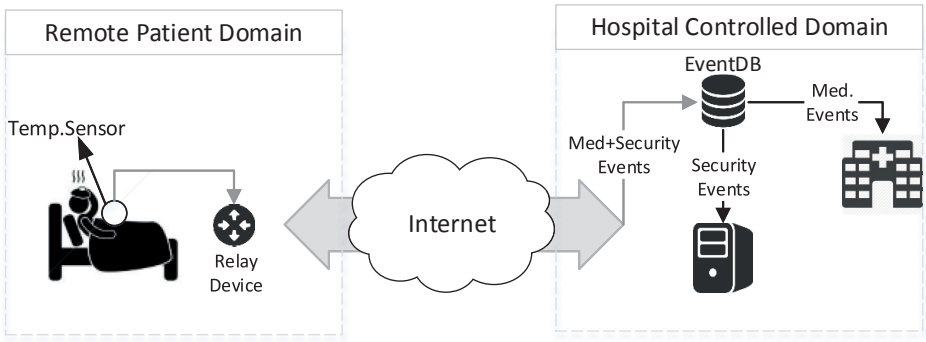


Figure 10.3: Prototype environment

10.3.1 Technical Setup

Encrypted patient body temperature values are collected using a wearable temperature sensor, which are communicated to the hospital site via a smartphone as a relay device. In the patient domain, an app on the smartphone receives, parses and directs temperature readings and any security event generated by the sensor to their respective event databases. The temperature values are extracted by the hospital health application, where they are decrypted and displayed as a continuous real-time graph for medical diagnosis. The security events are pulled by the EDAS platform to investigate and respond to any potential threats. The EDAS platform is a standalone system contained in the hospital-controlled domain. A context diagram of the environment is shown in Figure 10.3.

10.3.2 Architectural Overview

From a structural point of view, EDAS is designed as a component-based architecture (CBA) [52], where the design is fragmented into functional components necessary for achieving adaptive security. The components interact with each other using provided (output) and required (input) interfaces whenever an event is generated. Hence, from a communication perspective, EDAS utilizes an event-driven architecture (EDA) [36] in which events generated by a monitored component (thing) trigger concerned components. Events are considered as security changes and are monitored and correlated using a complex event processing (CEP) method. A CBA and EDA design make EDAS a reusable, replaceable, extensible, interoperable and independent architecture.

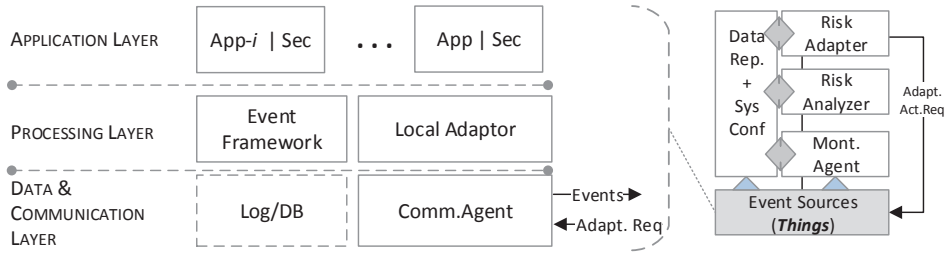


Figure 10.4: Event source abstraction.

10.3.3 Event Sources

An event source, a thing in the EDAS context, can be visualized in functional layers, as shown in Figure 10.4. The application layer contains the actual application(s), temperature sensing in our case, along with the necessary security components, tools and protocols. The processing layer must have at least two components for EDAS to work efficiently, an event framework and a local adaptor. The *Event Framework* typically comes as an out-of-the-box service with most applications and includes an event handler and a logging utility.

Microsoft defines an event as a message generated by an object to signal the occurrence of an action, which qualifies a user interaction or any system change, *etc.* [38]. Events are thing-specific (applications or devices) information with unique attributes describing a particular change. They can be status information notifying about the battery, CPU or memory levels; a thing internal change, such as a computation error caused within a body sensor; a particular type of user interaction, such as updating a bank account number, a password change or inserting incorrect information; or an external stimuli, e.g., a location change update notified by a GPS sensor. Events are received by a handler, for instance the Java Event Handler [5], which listens to a particular event raised and invokes further methods necessary for handling. One typical method is to log that event into a local or remote file or DB present on the data and communication (DC) layer.

The *Local Adaptor* parses the adaptation request and calls the particular application and security API to adopt the new settings locally. The adaptation request received is a string detailing the new security parameters to be adapted as a result of deemed risk and an *AppID* identifying the application that raised the event, which will adopt the new settings. The local adaptation process is shown in Figure 10.5a, and an example adaptation request is shown in Figure 10.5b. Most of the low-end devices, such as body sensors, are not equipped with the local event logging facility because of the limited storage capacity and sometimes do not have remote logging capabilities. The EDAS *Communication Agent* hooks onto the output stream of the

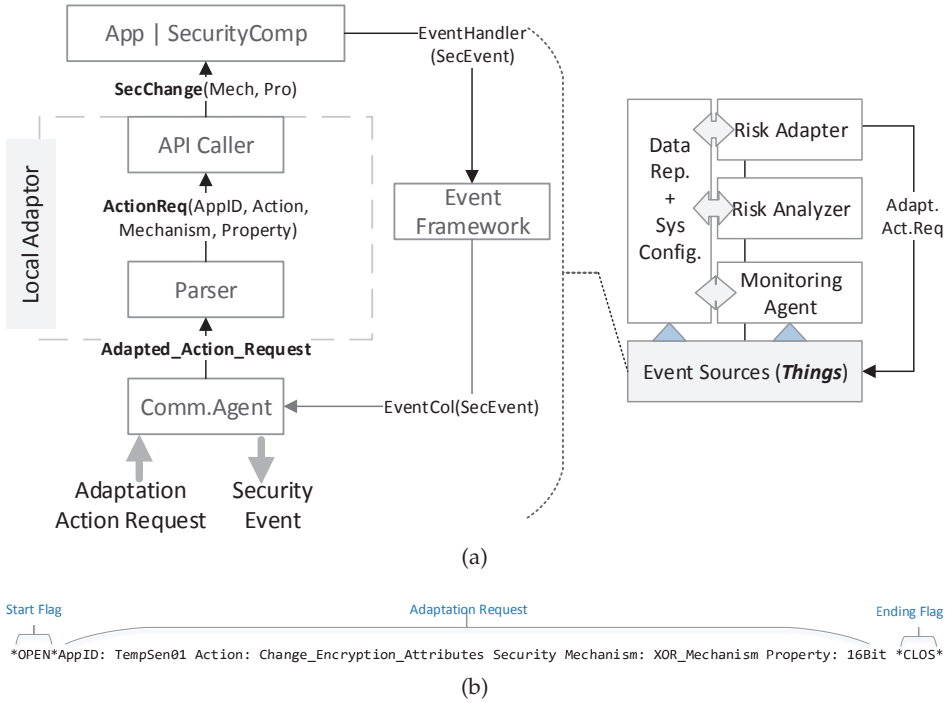


Figure 10.5: Local adaptation at the thing level. (**a**) Local adaptation process; (**b**) example adaptation request.

event framework via the *EventCol* interface, collects the events as they are raised locally and sends them to the EDAS platform via an event DB using HTTP request. Besides event communication, the agent also serves incoming connections. Thus, complying with the memory constraints, this design does not require any local storage for the events. Events from sources like a smartphone or those having a local logging facility, such as a file or DB, can be collected through the *LogCol* interface from the facility. Event source components and the interfaces between them can be seen in Figure 10.6.

10.3.4 The EDAS Platform

The objective of the EDAS platform is to monitor, filter, normalize and analyze primitive events coming from the monitored things in the IoT-environment. Furthermore, it decides a risk mitigation strategy per the risk faced, user and service preferences and thing capabilities. OSSIM is used as a monitoring and correlation platform in the prototype. The essential components and interfaces involved in this process are shown in its component diagram, Figure 10.7a, whereas a layered visualization of the platform is shown in Figure 10.7b. The components are explained as follows.

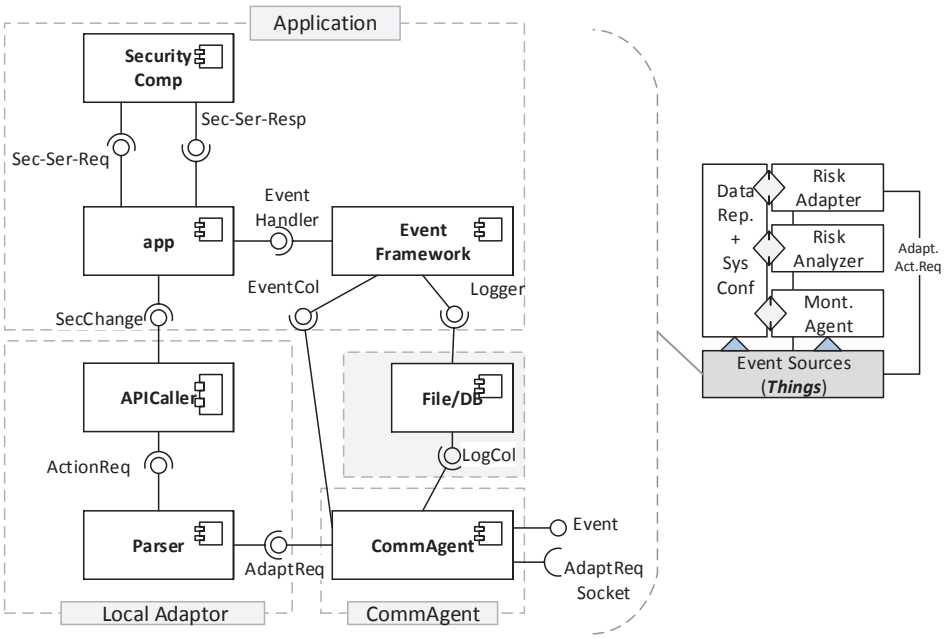


Figure 10.6: Event source component diagram.

10.3.4.1 Monitoring Agent

The *Monitoring Agent* reads, filters and normalizes events communicated by various sources. For each event source, there is a unique component called a plugin that performs the said operations. A plugin reads events from an event queue, a log file or DB, filters them using a white-listing technique and transform them into a standard format for analysis. Potential security events matching a regular expression (RegEx) or an SQL query are forwarded for normalization, considered as critical security events, while the unmatched ones are discarded. During normalization, essential event attributes are extracted as variables that are later used in the risk analysis process. Each event source and event type are assigned a unique plugin ID and plugin security ID (SID), respectively. These identifiers uniquely identify specific sources and event types in the IoT environment and are utilized in later processes. An example of an event before and after normalization with essential attributes for analysis and adaptation is shown in Figure 10.8.

Events can be distinguished based on the information they contain. Based on the type of events and their content, the monitoring agent's (the plugin) filtration criteria decide which of them should be considered for risk analysis. For instance, medical events concerning vital body signs are not to be classified as security events and may be filtered out unless they are involved in a related biometric authentication system. Furthermore, among other es-

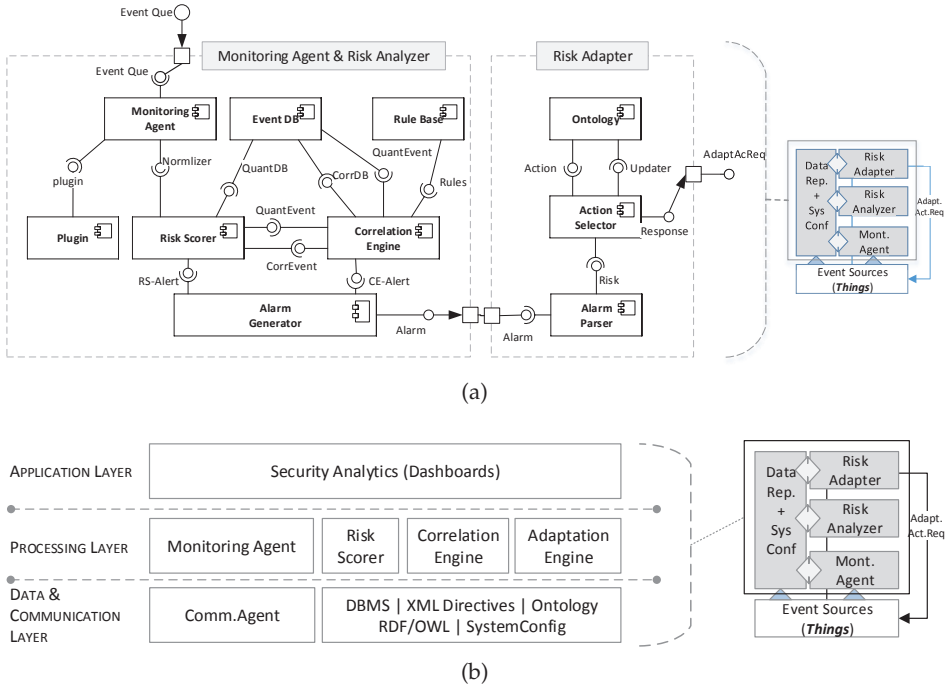


Figure 10.7: The EDAS Platform. (**a**) EDAS platform component diagram; (**b**) The EDAS platform layered architecture.

Primitive Raw Event:

```
May 30 13:15:52 dmz01 sshd[12980]: Accepted password for root from 192.168.178.20 port 4445
ssh2
```

Normalized Event:

2010-05-30 13:15:49,441 Output [INFO]: event **type** ="detector" **date**="1275239752" **sensor**="192.168.178.201" **interface**="eth0" **plugin_id**="4003" **plugin_sid**="7" **src_ip**="192.168.178.20" **src_port**="4445" **dst_ip**="192.168.178.200" **dst_port**="22" **username**="root" **log**="May 30 13:15:52 dmz01 sshd[12980]: Accepted password for root from 192.168.178.20 port 4445 ssh2" **fdate**="2010-05-30 13:15:52" **tzone**="0"

Figure 10.8: Example primitive and normalized events.

sential attributes, events are generated with a key property generally called level, for instance, see Microsoft event properties [37]. The level indicates an event's importance or its severity level. Although the level definition is thing specific, it can provide essential information to distinguish how critical a particular event is and how it should be assessed in the risk analysis process. In the prototype, OSSIM transformed this importance level into event priority, discussed in the next section.

10.3.4.2 Risk Scorer and Correlator

During normalization, the plugin defines which SID has to be assigned a particular event. An SID definition includes its ID, priority, reliability and a description. These fields are registered for a particular event type in a MySQL DB present at the data layer. Priority and reliability values together with the asset (event source or thing) value are used to quantify the risk associated with a particular event [26]. The *Risk Scorer* performs this quantification. In OSSIM, asset and priority values reflect the importance of the event source and the event respectively. A higher value implies a high event importance. For instance, a higher value can be given to an error or warning level event than to an information level event. Similarly, in a remote patient monitoring system, a critical asset, such as a wearable sensor, is given a higher asset value as compared to a smart device that the patient uses for other medical purposes. The reliability of an event (SID) specifies its probability as an actual attack. It is an attack probability level asserting the chances that a particular SID may yield to a real compromise and is used to deal with false alarms.

As we have suggested earlier that any CEP-supported analysis method can be utilized in EDAS, we cannot recommend any particular risk formulation equation because of the different risk perceptions. Below, we include the OSSIM risk equation [26] as an example and for the purpose of explaining the prototype. OSSIM calculates the risk of an event as follows:

$$Risk(Event) = (Asset \times Priority \times Reliability/25) \tag{10.1}$$

where Asset and Priority can take a value from [0, 5] and Reliability from [0, 10]. The division by 25 is made to restrict risk to 10 different risk levels.

Risk quantification is based on the event's primitive information. In some cases, several events over a period may contribute towards an incident. In such circumstances, if a risk is calculated on a single independent event, it may lead to false positives and negatives, which may further yield to the selection and implementation of inappropriate adaptation strategies. To avoid such situations, the *Correlation Engine* correlates different potential events over a period in a definitive context and decides whether there is a risk involved or not. It modifies the event's reliability as per a faced situation (context) when multiple potential events are detected in a specified

```
<directive id="500000" name="SSH Brute Force Attack Against DST_IP" priority="4">
            <rule type="detector" name="SSH Authentication failure" reliability="0"
                  occurrence="1" from="ANY" to="ANY" port_from="ANY" port_to="ANY"
                  plugin_id="4003" plugin_sid="1,2,3,4,5,6,9,10,12,13,14,15,16,20">
                  <rules>
                        <rule type="detector" name="SSH Successful Authentication (After 1 failed)"
                              reliability="1" occurrence="1"
                              from="1:SRC_IP" to="1:DST_IP"
                              port_from="ANY" time_out="15" port_to="ANY"
                              plugin_id="4003" plugin_sid="7,8"/>
                        <rule type="detector" name="SSH Authentication failure (10 times)"
                              reliability="2" occurrence="10" from="1:SRC_IP"
                              to="1:DST_IP"
                              port_from="ANY" time_out="40" port_to="ANY"
                              plugin_id="4003"
                              plugin_sid="1,2,3,4,5,6,9,10,12,13,14,15,16,20"
                              sticky="true"/>
                  </rules>

            </rule>
</directive>
```

Figure 10.9: Example OSSIM correlation directive.

interval. In other words, as more probable events are detected in the same correlation context, its reliability is increased. Thus, the correlation makes the overall threat reliability more accurate as more events occur in the same context.

In OSSIM, a context defined for event correlation is a sequence of different events observed in a particular time frame. It is stored as an XML directive in a file and is activated when a particular SID is detected [26]. An XML correlation directive contains a rule set. The first rule is called the triggered rule, as it activates the potential threat context to be analyzed. Each rule specifies an event occurrence and defines a new reliability for the associated threat context. Thus, a risk is analyzed in a context-aware manner, where events are correlated in time and space.

An example directive is shown in Figure 10.9. This example directive captures repeated SSH log-in attempts and the corresponding contextual events (SIDs) generated as a result of a failed attempt. It can be seen that the reliability of the threat context is analyzed with each rule. Correlation is performed in a particular time frame captured in the `time_out` variable. With each rule, it is made clearer whether the events correspond to authorized attempts where one can forget or mistype log-in credentials or to a compromise, such as brute force. Hence, with each rule (event occurrence), the threat context reliability is analyzed again, and risk alarms are raised accordingly. Event correlation makes analysis more accurate and reduces the possibility of false alarms.

10.3.4.3 Adaptation Engine

The *Adaptation Engine* decides a mitigation action for a particular user and thing in a given risk context. It takes the risk information from the *Risk Analyzer* and calls a runtime RDF/XML ontology. The proposed security adap-

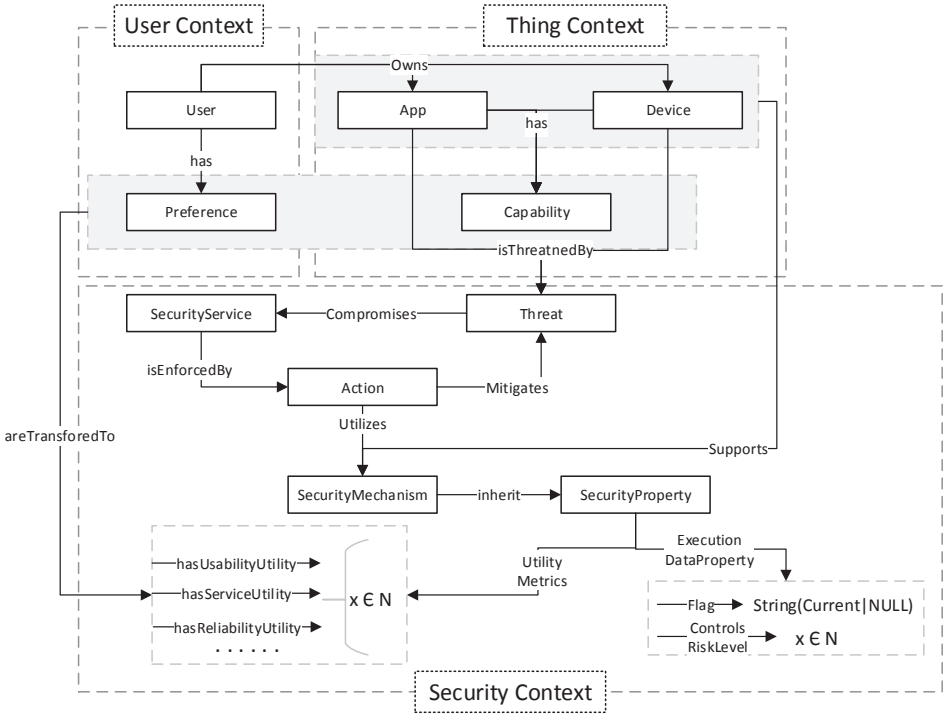


Figure 10.10: Security adaptation ontology.

tation ontology, shown in Figure 10.10, contains the security metrics, device capabilities and user preferences necessary to decide an optimized adaptation action from a pool of available actions. By optimized action, we refer to a response that is selected after assessing user, service and device requirements in a particular risk context. Entities and associated relationships in the ontology along with examples are detailed in our previous work [15]. The ontology entities are grouped into three contexts; user, thing and security. Each context captures the respective knowledge, as well as current runtime security settings necessary for deciding an adapted action. While selecting an action, the engine utilizes an Apache Jena-SPARQL API-enabled [3] script to query and update the stored ontology. Updating is performed to ensure that the ontology as a runtime knowledge platform is aware of all of the current configurations that may be required in possible succeeding adaptation decisions.

The adaptation engine is activated when it receives a risk alarm from the risk analyzer. The alarm is a token detailing the risk components, *i.e.*, the Threat-ID, Risk-Level, and Device-ID. The adaptation action is selected in a stepwise procedure as shown in Table 10.1. The table describes all of the signatures pertaining to the ontology elements, *i.e.*, subjects, predicates

Table 10.1: The adaptation action decision process.

Step No.	Type: Subject	Type: Predicate	Type: Object	Type: Return
Step 1	**Description:** *identifying a particular threat faced in the ontology*			
	`Class`: threats	`Data Property`: hasThreatID	`String`: Threat-ID, e.g., "DOS5001"	`Class Object`: a threat object
Step 2	**Description:** *listing possible actions that address the threat identified in Step 1*			
	`Class`: actions	`Object Property`: mitigates	`Class Object`: threat	`Class Object`: action objects
Step 3	**Description:** *identifying security mechanisms utilized by actions identified in Step 2*			
	`Class Object`: action objects	`Object Property`: utilizes	`Class`: security mechanism	`Class Object`: security mechanism objects
Step 4	**Description:** *identifying the device facing the threat*			
	`Class`: devices	`Data Property`: hasID	`String`: Device-ID e.g., "192.168.1.3"	`Class Object`: A Device object.
Step 5	**Description:** *extracting only device-supported security mechanisms from those identified in Step 3*			
	`Class Object`: a device object	`Object Property`: supports	`Class Object`: security mechanism objects	`Class Object`: security mechanism objects
Step 6	**Description:** *listing properties that are utilized by mechanism identified in Step 5*			
	`Class Object`: security mechanism objects	`Object Property`: inherit	`Class`: security property	`Class Object`: security property objects
Step 7	**Description:** *selecting properties addressing a particular risk level from the properties identified in Step 6*			
	`Class Object`: security property objects	`Data Property`: controlsRiskLevel	Integer: risk-Level e.g., 1, 2, 3...	`Class Object`: security property objects
Step 8	**Description:** *extracting utility metrics values for individual property identified in Step 6*			
	`Class Object`: security property objects	`Data Property`: hasUsabilityUtility, hasConfUtility, ...	`Integer`: Utility-value e.g., 1,2,3...	`Integer`: utility-value

and objects, which are accessed at each step of the procedure. The ontology is developed in an RDF/XML format using the Protege tool [44]. Different types of ontology elements used in the table are described as follows: a `Class` refers to a concept of interest in the ontology. `Class Objects` are the members of a class. `Object Property` is the relationship between one or more members of one class with one or more members of another class, and a `Data Property` refers to a particular attribute of a class object. Example objects and a description of classes used in the adaptation process are given in Table 10.2. For a detailed description of all of the concepts and relationships in the proposed ontology, refer to our previous work [15].

As the final step (Step 9), a security property object having the maximum weighted utility among those identified in Step 7 is selected as the most optimal property. The corresponding mechanism and action are extracted, and an adaptation request string is constructed. The request is sent to the event source as an optimal response to the faced threat and to be adopted locally. Metrics, such as hasUsabilityUtility, hasReliabilityUtility and hasConfUtility, *etc.*, are attributes associated with each security property object. They reflect the property utility in terms of user and service requirements and

Table 10.2: Classes description in the adaptation process.

Class	Description	Example class objects
Device	Monitored devices and their attributes	Sensor, smartphone, desktop
Threat	Threats known in the environment	DoS, brute force, malware infection
Actions	Mitigation responses to be adapted to defend against a threat	Enforce a CAPTCHA , change cipher, change routing algorithm
Security mechanism	Methods, tools, algorithms used by an action	DES, AES, XOR, CAPTCHA, password length
Security property	Adjustable attributes of a security mechanism	Eight-char password, image CAPTCHA, audio CAPTCHA, 128-bit key

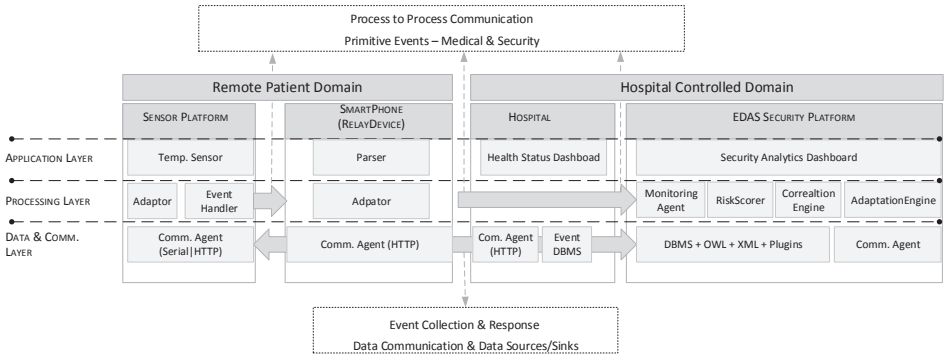


Figure 10.11: Prototype architecture categorized into functional layers.

device capabilities. These utility metrics are derived from the mentioned classes in the ontology.

The prototype test environment as a whole can be categorized into functional layers as shown in Figure 10.11. The relay device, a smart device, is used primarily for parsing medical data and security events arriving at the same HTTP connection. It uses a MySQL client to send these events to the respective DBs contained in the event DBMS for further investigation. Utilizing a smart device, such as a smartphone, as a relay also makes the entire system more usable. Patients will be able to monitor their health status locally with the help of an app installed without querying the health journals at the hospital site. Furthermore, it can make the patient monitoring more usable in mobility scenarios, therefore increasing the overall eHealth service utility. Figure 10.12 shows an abstract level message transfer between the major processes that we developed and designed in our prototype. It can be seen that data are collected using two loops, medical data collection and security adaptation, both executed in parallel.

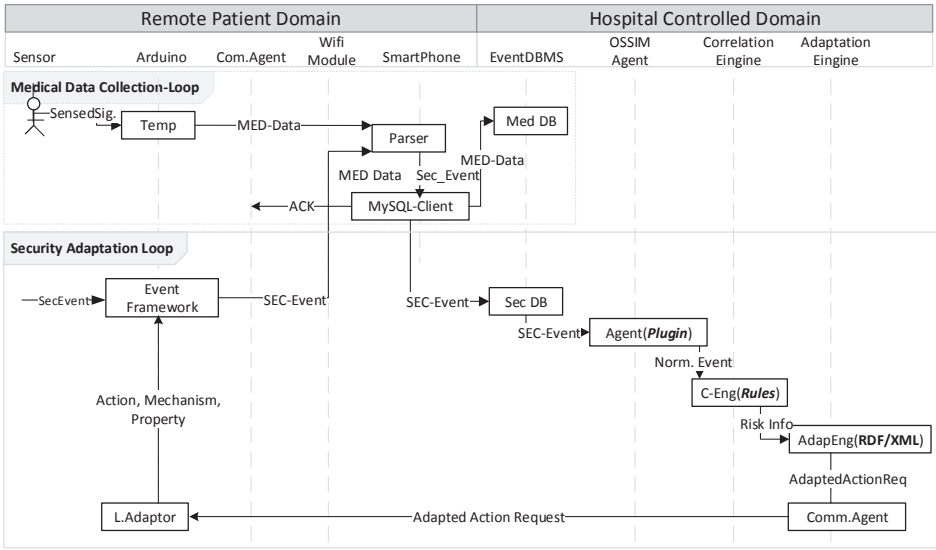


Figure 10.12: EDAS message sequence chart.

10.4 Case Study

A test case study concerning user confidentiality and service availability is developed to demonstrate EDAS as a real-world prototype. The case study characterizes a tradeoff between confidentiality and availability. The case study is based on a general phenomenon that encrypting messages consumes more energy if longer key lengths are utilized and *vice versa*. Preshared keys and respective indexes are used in the case study. A higher index corresponds to a key with increased length. The state transition diagram depicting the case study security adaptation is shown in Figure 10.13. *Stable State 1* is assumed to be the initial state. Different cipher key lengths for encrypting medical data are adapted when the *LowBattery* or *ChargingUp* events are generated by the temperature sensor.

The case study consists of two scenarios to reflect a confidentiality - availability tradeoff situation. In the first scenario, EDAS decides to ensure service availability as opposed to keeping a high confidentiality level when the sensor battery level drops below a certain threshold. Therefore, encryption keys with decreased lengths are adapted to meet the primary requirement, availability, of a continuous patient monitoring system. In the second scenario, confidentiality is preferred over availability. Confidentiality is regained and key lengths are increased as the battery is recharged to a particular threshold, indicating that the sensor is steadily available to meet a particular service level. Key lengths are gradually increased and decreased

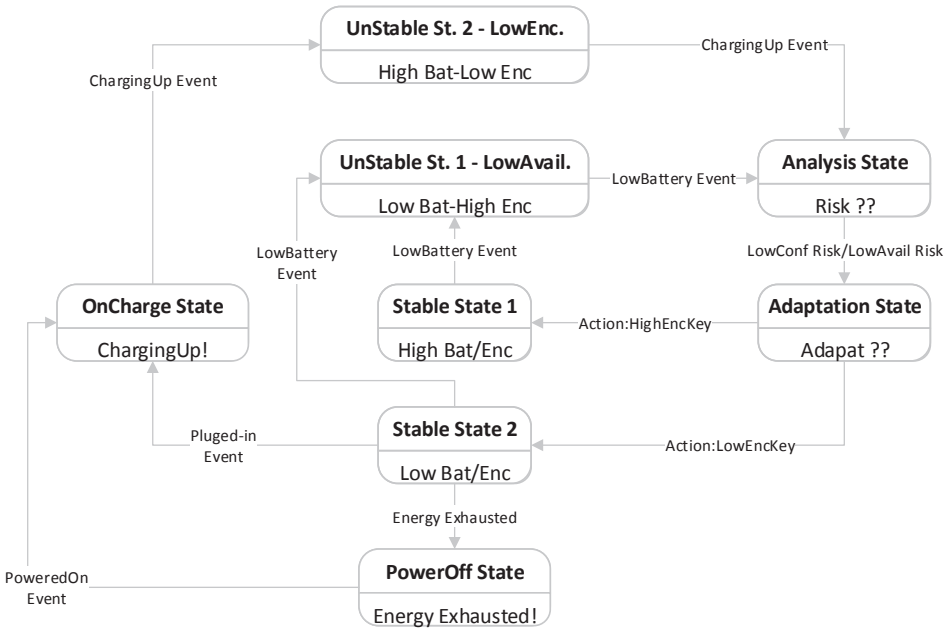


Figure 10.13: Adapting security to low availability/confidentiality risks.

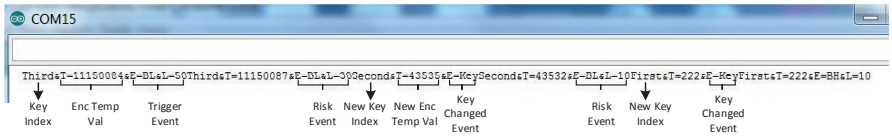


Figure 10.14: Scenario 1: sensor screen: decreased key lengths are adapted when battery level drops.

as per the observation of threshold battery level events. This adaptation process is performed continuously until the sensor changes a state.

- Scenario 1 (Low Availability Risk: High Encryption with Low Battery)

Low availability (context or directive) risk is raised when there is a *Low-Battery* event with the level being less than *X*%, and the encryption is still done with an increased key length. The corresponding risk is reduced when a *KeyChanged* event is generated by the event source (temperature sensor) after a reduced encryption key length is adapted. The corresponding screen-shots are displayed in Figures 10.14 and 10.15.

Events	Timestamp	M-Agent	Source(S)	Destination(D)	Asset.Val S→D	Risk
directive_event:BatteryLow-LowAvailability	2014-11-25 19:36:11	alienvault	192.168.1.2:2000	192.168.1.3	4→2	0
EDAS:EncryptionKeyChanged	2014-11-25 19:36:05	alienvault	192.168.1.2:2000	192.168.1.3	4→2	0
directive_event: BatteryLow-LowAvailability	2014-11-25 19:35:59	alienvault	192.168.1.2:2000	192.168.1.3	4→2	1
EDAS:SensorBatterLowEvent	2014-11-25 19:35:50	alienvault	192.168.1.2:2000	192.168.1.3	4→2	0
directive_event: BatteryLow-LowAvailability	2014-11-25 19:35:35	alienvault	192.168.1.2:2000	192.168.1.3	4→2	0
EDAS:EncryptionKeyChanged	2014-11-25 19:35:30	alienvault	192.168.1.2:2000	192.168.1.3	4→2	0
Directive_event: BatteryLow-LowAvailability	2014-11-25 19:35:23	alienvault	192.168.1.2:2000	192.168.1.3	4→2	1
EDAS:SensorBatterLowEvent	2014-11-25 19:35:18	alienvault	192.168.1.2:2000	192.168.1.3	4→2	0
EDAS:SensorBatterLowEvent	2014-11-25 19:34:48	alienvault	192.168.1.2:2000	192.168.1.3	4→2	0

Figure 10.15: Scenario 1: EDAS platform dashboard screen (modified): the *LowAvailability* alarm is raised (as risk = 1) whenever a *BatteryLow* event is detected and is reduced when a *KeyChanged* event is observed after adaptation. Color legend: yellow, trigger event; red, alarm (unacceptable risk); green, alarm (acceptable risk); white, event detected.

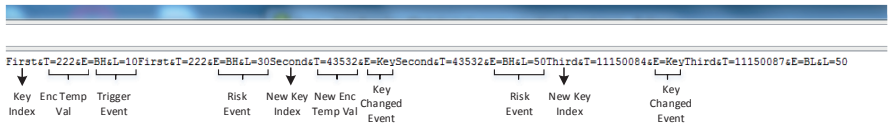


Figure 10.16: Scenario 2: sensor screen: encryption adapts to increased key lengths when the battery is recharged to a threshold level.

- Scenario 2 (Low Confidentiality Risk: Low Encryption with High Battery)

The low confidentiality alarm is raised when the *BatteryChargingUp* event is detected with a level greater than *Y*%, and the encryption is still done with a reduced key length. The corresponding risk is reduced when an increased key length is adapted and a *KeyChanged* event is discovered after increased key lengths are selected. See the corresponding screenshots in Figures 10.16 and 10.17.

10.5 Feasibility and Evaluation

This section evaluates EDAS as an event-driven security concept and system architecture and will detail lessons learned from the prototyping activity. We will discuss how EDAS can be the right tool for ensuring real-time risk management in dynamic environments, such as IoT, and how it complements existing ICT infrastructure. Moreover, EDAS is compared with architectures corresponding to traditional security controls to investigate its feasibility as a viable solution for IoT.

Events	Timestamp	M-Agent	Source	Destination	Asset.Val S→D	Risk
directive_event:BatteryCharging-LowConfidentiality	2014-11-25 19:36:11	alienvault	192.168.1.2:2000	192.168.1.3	4→2	0
EDAS:EncryptionKeyChanged	2014-11-25 19:36:05	alienvault	192.168.1.2:2000	192.168.1.3	4→2	0
directive_event:BatteryCharging-LowConfidentiality	2014-11-25 19:35:59	alienvault	192.168.1.2:2000	192.168.1.3	4→2	2
EDAS:ChargingUp	2014-11-25 19:35:50	alienvault	192.168.1.2:2000	192.168.1.3	4→2	0
directive_event:BatteryCharging-LowConfidentiality	2014-11-25 19:35:35	alienvault	192.168.1.2:2000	192.168.1.3	4→2	0
EDAS:EncryptionKeyChanged	2014-11-25 19:35:30	alienvault	192.168.1.2:2000	192.168.1.3	4→2	0
directive_event:BatteryCharging-LowConfidentiality	2014-11-25 19:35:23	alienvault	192.168.1.2:2000	192.168.1.3	4→2	1
EDAS:ChargingUp	2014-11-25 19:35:18	alienvault	192.168.1.2:2000	192.168.1.3	4→2	0
EDAS:ChargingUp	2014-11-25 19:34:48	alienvault	192.168.1.2:2000	192.168.1.3	4→2	0

Figure 10.17: Scenario 2: EDAS platform dashboard screen (modified): the *Low-Confidentiality* alarm is raised (as risk = 1, 2) whenever a *BatteryChargingUp* event is detected and is reduced when a *Key Changed* event is detected after adaptation. Color legend: yellow, trigger event; red, alarm (unacceptable risk); green, alarm (acceptable risk); white, event detected.

10.5.1 Dynamic Real-Time Autonomous Risk Management

The primary objective of EDAS is to ensure continuous and dynamic risk management capabilities in the IoT. Changes (events) in the monitored environment are collected and analyzed as soon as they are observed. IoT is thought to be a self-adaptive and self-organized network, and things are deemed to be autonomous [20]. Self- and autonomous adaptation capabilities are necessary in IoT [18, 32]. These properties ensure dynamic adaptation to avoid potential management delays caused by human intervention. EDAS overcomes this issue by placing the user preferences, service requirements and analyst knowledge in the correlation directives and adaptation ontology before the system start-up. Thus, it empowers the system to monitor, analyze and adapt to the risks faced autonomously and dynamically in an optimum manner.

10.5.2 Context Awareness

The term context has been defined differently by various authors. However, a more general definition is given by Abowd *et al.* [12] as any information that can be used to characterize and recognize the situation of an object, person or place. Context provides vital information regarding the who, what, where, when and why of a situation [13]. In a computing environment, this context or information is usually offered by the system- or application-generated events. It can be seen in Figure 10.8 that the information in the primitive events precisely provides essential attributes necessary to qualify both definitions stated. Hence, EDAS captures the fundamental unit of a system change, *i.e.*, the event, to set a clear and distinct ground for context-aware risk analysis.

Furthermore, in real-world scenarios, a compromise is usually a combination of different attack vectors, modifications, tools and targets, which

may trigger a series of different events originating from different sources as the compromise progresses. EDAS addresses such situations through event correlation. Using a correlation method, such as a rule-based OSSIM correlation directive, an analyst could define a particular compromise context defined as a rule set. The context captures all of the potential events that signify a potential compromise and can accurately qualify whether a risk is involved or not. Correlating events from different sources provides a broad view of understanding the context and holistic information. Thus, it reduces any false alarms that may be caused by analyzing events independently. Nonetheless, if events are seen as a series of steps towards a particular compromise situation (context), then by exploiting the precautionary principle [43, 50], EDAS can predict these steps and can respond to corresponding threats (events) before they are realized as actual attacks.

10.5.3 Preferences and Capability-Based Adaptation

Adapting security changes autonomously by only considering the impact of the risks faced is a security risk itself. In such cases, changing security parameters may negatively affect service attributes, such as throughput or latency, and thing resources. Re-configurations based only on the security context of a threat without assessing its impact on the service requirements can cause unnecessary adaptation that may cause serious problems [35]. Such strategies can cause service disruption in environments like IoT or WSN, where the main driving technologies are battery-powered and resource-limited devices. Therefore, other factors, such as the device and application capabilities, as well as service requirements, need to be considered while an adaptation strategy is decided, as performed in EDAS.

Nonetheless, in a user-centric service, such as IoT-eHealth, the user, *i.e.*, the patient, as well as the medical staff, preferences should also be assessed while new security settings are adapted. The EDAS runtime adaptation ontology stores these capabilities, requirements and preferences before system start-up and transforms them into metrics with respective utilities against a particular *SecurityProperty*. This enables the adaptation engine to choose a mitigation action from the available action pool, such that its weighted utility has a maximum value for a given user using a particular service in a specific risk situation.

10.5.4 Development and Deployment

Besides some component engineering, the EDAS development involves a technology integration process. It utilizes the thing event framework included in almost all mature applications and devices as a logging, troubleshooting and debugging facility. However, local adopters, which are merely a string parser and API caller, must be developed and should have

read and execute permissions in order to execute the call to a specific security component against a specified application. Local adaptation can also be performed via the application where the adaptation request received is passed as arguments to override the method that ensures security. As for the variety of technology used in the IoT, a platform-independent design, e.g., Java, can be opted to develop and integrate uniform local adopters across the monitored environment.

As most devices in IoT have low computational power and sometimes perform only specific sensing or actuation routines, the analysis burden has been taken away from the thing level to a resourceful machine (EDAS platform). Event sources can be monitored from anywhere. If the event source is addressable in the environment and if it can generate and communicate events, no matter where it is located (remote or on-site), it can be integrated with EDAS. This implies that traditional systems, such as firewalls, databases, file/application servers and other critical information systems, with built-in well-defined event frameworks, can also be integrated in and monitored by EDAS, as they have key roles in the overall service delivery.

Component-based architectures (CBA) and event-driven architectures (EDA) have the ability that their operational components can be distributed over the network as they are loosely coupled [36, 52]. EDAS can be used as a standalone platform to monitor, analyze and adapt to the risk faced. However, it inherits the CBA and EDA concepts, and thus, its major operations can be distributed over multiple locations in different hierarchical settings. For instance, in an IoT-eHealth perspective, a city hospital X can analyze the threats related to the environment it operates in by a local EDAS risk analyzer. Same settings can be established for a city hospital Y. However, a principal adaptation engine in location Z governed by a central security policy can be deployed to decide a mitigation response based on X's and Y's risk information to reduce the risk level in either of the domains. Such distributed settings enable security analysts to isolate and focus on a set of threats concerning a particular location context and may add more to the precision of the risk analysis process.

10.5.5 Architectural Comparison

This section provides a comparison of EDAS with traditional security architectures and corresponding controls. They are grouped into the following categories based on their architecture, however irrespective of the particular prevention and detection methods they utilize:

- Host-based: controls that manage security locally on an end-user machine, e.g., an anti-virus, host IDS, firewalls, *etc.*

- Endpoint: controls that protect end-user machines, but are managed by a central entity, e.g., Endpoints in Microsoft System Endpoint Protection.

- Agent-based: controls that protect and manage the monitored environment based on the information gathered by specialized agents; for example, an agent-based IDS.

- Centralized: controls that provide security as a central stand-alone entity; for instance, an enterprise firewall or a network-based IDS.

- Distributed: a security architecture that utilizes various prevention or detection systems distributed over a network to facilitate advanced security analysis; for example, a distributed firewall or IDS.

The objective is to highlight whether the concept of EDAS as an event-driven security architecture and adaptive security solution adds value to an IoT-based service, such as eHealth. There is comprehensive work concerning system and software architecture evaluation, e.g., [27, 28]. However, we present a simple comparison based on the architectural concept and on the prototyping exercise we performed to reflect on the extent to which these candidates qualify or support a given architecture quality attribute in the IoT-eHealth context.

One can find an extensive list of these attributes in the literature. However, we have selected a list of architecture quality attributes from [17, 22, 39, 41], which covers most of these attributes. Furthermore, we have used selected attributes from these sources as some of them, though having little differences, can be defined interchangeably. For instance, attributes like modifiability, evolvability, adaptability, configurability, reusability and customizability can be accumulated for a change in a system for which a single and, more common, word, maintainability, can be used, which we have adopted. Some attributes are intentionally dropped as they are more focused on software architecture as opposed to system architecture, e.g., portability in [22]. Hence, the attributes used here are considered and defined in a system perspective and not in a software context. Furthermore, we have added a few functional attributes, such as monitoring scope and threat detection accuracy, to reflect on a candidate aptitude as a security solution. These attributes are described in Table 10.3. Table 10.4 depicts the comparison where (++) implies that an attribute is positively qualified or supported, a (+) indicates partial qualification or support of the attribute indicating that there is some design dependency involved, and (-) indicates an absence of the attribute.

Table 10.3: System architecture quality attributes.

	S.No	Attributes	Ref.	Description
Runtime Execution	1	Interoperability	[17, 39, 41]	The ability of a system to be utilized in diverse environments
	2	Reliability	[17, 22, 39, 41]	The ability of a system to continue intended operations over time
	3	Usability	[39, 41]	The measure of how well the user requirements are met for using the system (in terms of the user's security requirements)
	4	Latency	[17, 22, 39, 41]	The reaction time of a system to an event/incident/threat
	5	Throughput	[17, 22, 39, 41]	The number of events/threats/incidents responded to in a given time interval
Security	6	Security	[17, 39, 41]	The capability of a system to stand against a potential threat concerning the C-I-A services
	7	Monitoring Scope		The various types of contextual information/assets the system can monitor and analyze
	8	Adaptability	[41]	The ability of a system to systematically and autonomously regulate its behavior and re-configure its settings (here Security Adaptation only)
	9	Threat Detection Accuracy		The capability of a system to accurately detect threats to avoid false positives/negatives
Design	10	Simplicity	[22]	A measure reflecting how functionalities are separated from one another to keep things clear and easy to understand, isolate and develop
	11	Extensibility	[22, 41]	The ease with which a system functionality can be extended by adding more components to it
	12	Maintainability	[17, 22, 39, 41]	The ability of a system to be easily modified when requirements are changed
Support	13	Supportability	[39, 41]	The ability of a system to provide helpful information to resolve errors, trace user activity and related issues
	14	Testability	[39, 41]	A measure depicting how well a test criteria can be created, executed and evaluated against the system

Table 10.4: EDAS *vs.* traditional security controls.

	Attribute	EDAS	Host	Endpoint	Agent-Based	Centralized	Distributed
Execution	Interoperability	++	-	-	-	++	++
	Reliability	++	-	-	-	-	++
	Usability	++	+	+	-	-	-
	Latency	+	++	++	+	+	+
	Throughput	++	+	+	+	+	+
Security	Security	++	+	+	+	+	++
	Monitoring Scope	++	+	+	+	+	+
	Adaptability	++	-	-	-	-	-
	Threat Detection Accuracy	++	+	+	+	+	++
Design	Simplicity	+	++	++	+	+	+
	Extensibility	++	-	-	+	+	+
	Maintainability	+	++	+	+	+	+
Support	Supportability	++	++	++	++	++	++
	Testability	+	++	++	+	+	+

10.5.5.1 Interoperability

A CBA design makes a component more independent and reusable [52]. EDAS is based on the CBA style, which makes its components interoperable. This is also true for distributed systems. Moreover, it is based on the event processing concept, which is a typical facility in almost all applications. Hence, irrespective of the type and nature of an application, device or network, it is an environment or platform-independent architecture that can be used in any context where the monitored objects can generate and communicate events. On the other hand, host, endpoints and agent-based solutions are designed for particular platforms performing specialized tasks. They may share data with external systems, but cannot be operated on different platforms.

10.5.5.2 Reliability

The CBA and EDA designs ensure loose coupling between the system components, which enables component distribution and redundancy to support reliability, separation of functionalities and avoids single point failures [36, 52]. These properties as desirable design attributes can also be found in distributed systems. Other architectures, being single entities, may expose and threaten the whole enterprise architecture or asset if compromised [23].

10.5.5.3 Usability

Traditional security architectures and corresponding controls are designed to protect resources, such as a server, files or subnets. They are driven by a resource-specific policy irrespective of the user requirements. EDAS offers a user- and service-centric security solution. All requirements pertaining to the user, service, as well as critical resources are considered and assessed in individual adverse contexts before any decisions are made. End-user solutions may accommodate user preferences to some extent, but overall, the emphasis is the resource.

10.5.5.4 Latency and Throughput

Architectures designed for end users, *i.e.*, host and endpoint architectures, perform analysis locally where the events of interest occur and, thus, have low latency. The other listed architectures, including EDAS, process events away from the point of occurrence and are subjected to delays caused by the network and communication. However, the EDAS autonomous adaptation property decreases any response management delays caused by a human in the loop. Thus, it results in maximum throughput as compared to the rest of the architectures.

10.5.5.5 Security

We have already established in Section 10.1 how traditional controls are not feasible for resource-constrained things and how they lack analysis of a context holistically, though they do provide security to a certain level. We have also detailed how EDAS addresses these issues by transferring computations required at a thing level to a resourceful machine (EDAS platform) and correlating different types of events in time to provide grounds for accurate analysis that reduces potential false alarms. Its cross event correlation feature can analyze a spectrum of threats, such as power exhaustion, confidentiality, intrusions, *etc.* Traditional controls can only defend against a defined set of threats. These can be the network, the web, local files or OS-related risks. Hence, their security aptitude is very restricted, and due to this lack of scope and context, they usually result in false alarms [33, 48].

10.5.5.6 Monitoring Scope

EDAS can monitor any event source as long as it is accessible and can communicate the events that it generates. This makes the monitoring scope of EDAS much broader as compared to traditional mechanisms that only monitor a specific concern.

10.5.5.7 Adaptability

To overcome potential management delays and to meet the dynamic nature of IoT, EDAS offers autonomic security adaptation. Adaptation is a key desirable attribute in IoT environments [18, 32]. Traditional architectures and solutions lack security adaptation and approach it in a manual manner where responses to threats are managed by a human in the loop. Of course, not all actions or configurations, for instance plugging in a wire or charging a smartphone to ensure availability, can be automated. However, all electronic operations can be automated, provided there are no physical engagements required. The objective of automation is to minimize the administrator or analyst interfacing with the system to increase throughput. In such circumstances, risk analysts may focus more on designing new criteria and rules for threat analysis, which can be added to the EDAS platform as security updates. Automation may also reduce the cost of the overall administration, as less effort will be required due to minimal manual configurations. The feasibility and degree to which an adapted action or activity can be automated is use case-, scenario- and risk context-specific and is beyond the scope of this study.

10.5.5.8 Threat Detection Accuracy

Please refer to Section 10.5.5.5 Security of the comparison discussing event correlation.

10.5.5.9 Simplicity

As a CBA, EDAS separates concerns into functional components. At the component level, separating concerns make it easier for developers to isolate, understand, verify and develop functionalities with fewer complexities [22]. However, at the system level and size of the monitored environment, its complexity is increased as more components are added. This notion also affects agent-based, centralized and distributed architectures. End-user architectures are comparatively simple to design, develop and implement due to the fewer number of components involved.

10.5.5.10 Extensibility and Maintainability

EDAS, as well as agent-based, centralized and distributed architectures are relatively hard to maintain and may demand increased cost due to the increased number of components involved. End-user solutions have a limited scope and fewer components. Thus, they take less effort and cost to maintain them. However, this maintenance advantage comes with a limitation of extensibility. Since end-user architectures are designed to meet specific and limited objectives, they cannot be extended to accommodate other systems. Agent-based systems can also be extended at the cost of an extra component, *i.e.*, the agent. Centralized and distributed architectures may also provide room for extension. However, the limited context they protect limits their extension scope. While EDAS can be extended to accommodate any system or thing that has a potential event framework.

10.5.5.11 Supportability

Almost all traditional security systems are equipped with mature event frameworks and logging mechanisms that can be used to resolve errors, failure and trace user activity. Since EDAS utilizes the same utility, it can potentially provide the same level of support.

10.5.5.12 Testability

While it is easy to create a test criterion against an individual component in a modular design, such as in a CBA, it is relatively complex to validate the entire system [39]. The increased number of components in EDAS, agent-based, centralized and distributed architectures makes it also hard to test and validate them. Interactions may be required in components distributed

across network locations, which may make testing more complex [41]. On the other hand, end-user systems can easily be tested, as they are considerably less complex.

10.6 Related Work

Since IoT comprises diverse technologies with varying capacities, there is a need to design appropriate architectures and mechanisms, such that information sharing and communication can be made more efficient. Credible work has been concluded in this regard, while others are still under research. Below, we discuss a few of them.

The OpenIoT project [7] details an open source architecture that aims to connect Internet-enabled things with cloud computing, thus enabling ICT companies to offer sensor-based solutions. The architecture of [42] consists of three layers, namely application, virtualization and physical plane. The cloud computing capabilities are placed in the virtualization plane. Sensor middlewares are placed in the physical plane, which collects, filters and normalizes sensors data and communicates them to the cloud. The application plane contains utilities that enable users to control and monitor the sensor. These utilities also control requests made from connected services.

A similar model is also produced in the IoT-A project [4]. However, the artifact, IoT Architectural Reference Model (ARM) [11], provides a more abstract and domain-specific description as opposed to detailing the technicalities. It consists of three major components. An IoT Reference Model stipulates the high abstraction level definitions, as well as information and communication models. These definitions and abstract models are driven by the business vision, scenarios and stakeholder requirements and serve as the second major component. Its IoT Reference Architecture provides a reference for developing IoT compliant architectures and mainly details various views, such as functional, deployment, operational and perspectives, such as security, resilience, performance and interoperability, derived from the scenarios and requirements.

Antii *et al.* [21] proposed a self-adaptive architecture for smart spaces, which utilizes an information security measurement ontology (ISMO) to carry out the adaptation process. The model is inspired from IBM's MAPE-K control loop [29]. Though the authors provide a detailed view of the architecture, they did not explicate how user requirements and things' hardware capacities should be addressed while new security strategies are adapted to a given situation.

Other adaptive security solutions and studies, for instance risk adaptable access control (RAdAC) [34], context-sensitive adaptive authentication [25], the RSA Adaptive Authentication platform [45], security event information management solutions, such as AlienVault Unified Security Management

[2] and HP ArcSight [9], *etc.*, either emphasize a single security service, e.g., authentication or confidentiality, or lack automated adaptation as an essential risk management component.

EDAS is focused on events corresponding to any security-related service or activity, *i.e.*, intrusion, confidentiality, energy, mobility, *etc.* Conceptually, EDAS can be related to the IoT-A ARM model, as it is driven by requirements and scenarios designed for IoT-enabled eHealth in remote patient monitoring settings. It is based on IoT-eHealth essentials that we have identified previously as functional, security and risk management requirements in [14]. Though designed primarily for IoT-eHealth, we suggest that EDAS, as an event-driven architecture, can be utilized for any IoT-enabled services where things can generate and communicate events. However, this proposition further needs to be investigated.

10.7 Discussion and Further Work

In this section, we discuss issues related to how the architecture itself can be made secure, its dependency on event frameworks, concerns related to event communication and how EDAS can provide a holistic approach towards the increasing risk sophistication. These are a few top level challenges, observations and possible further work that will be discussed in this section.

10.7.1 Securing the Architecture

Since EDAS aims to evaluate the security of a critical infrastructure by capturing and communicating security information (events), its adaptation loop also needs to be protected. The protection becomes more serious when its components are deployed in distributed settings. It must be ensured that the events, carrying security information, are communicated via well-protected channels and protocols and remain genuine and protected during the communication. Furthermore, access to the platform needs to be protected and well managed, and mechanisms should be provided to ensure its availability. To meet these requirements, we suggest the following:

(i) The EDAS platform should also monitor itself. This implies that there is a security monitoring and adaptation loop within the platform itself. Thus, EDAS should be considered as a critical event source in the architecture and needs to be monitored as other monitored objects in the scope.

(ii) Security tools, such as firewalls, intrusion detection, access controls and availability monitoring applications, should be installed on the platform to provide the required security services to it. The events

generated by these applications can also be integrated into the architecture to make the analysis process more accurate.

(iii) Event communication protocols should provide security mechanisms for confidentiality and integrity. Furthermore, it should be energy efficient to comply with the resources of low-end devices, such as sensors.

EDAS is primarily dependent on an event framework that can generate, handle and log meaningful events. In EDAS, we assume that event framework has already been programmed into the thing. This facility is typically available as an out-of-the-box solution for various purposes, including debugging, error tracking, security logging and other troubleshooting operations. In IoT-eHealth, it is a must have component for patient safety to ensure that the body sensors are working reliably.

10.7.2 Event Communication

While most applications (things) log events, they do not have mechanisms to communicate them to remote destinations. For this purposes, various protocols can be utilized that can read from the event log file or hook onto the event framework output stream and communicate them remotely in a timely manner. Some of these are logging specific, for instance SYSLOG [10] and SNMP, while general protocols, like HTTP, REST architecture [22] or MQTT [6], can also be utilized. However, the question to investigate here is: which of these will be more reliable and efficient to be used as communication agents in low-end devices as sensors in IoT? As already mentioned, these events contain critical information that should be protected during communication. Therefore, the event communication protocol needs to ensure that channel is well protected. Furthermore, to ensure energy efficiency and fast delivery factors, like the protocol's packet size, request-response time and other QoS attributes needs to be evaluated before it is adopted. For instance, just to reflect on this issue, Stephen in an experiment [40] shows that MQTT saves about 30% of battery as compared to HTTPS when 1024 messages each of one byte are sent over a period of 1 h. In the same experiment, he also shows that MQTT can send about 94-times more messages than HTTPS over a 3G network and 72-times more messages over a WiFi connection. An experiment like this implies that one must consider the communication protocol in an environment like IoT where resource-limited devices are the primary driving force.

In EDAS, we assume that the security analysts who design the threat correlation contexts have a keen understanding of the different types of events generated by a multitude of things in the monitored scope. It seems to be a complex task; however, most mature products (things) are provided with well-documented reference material of which they can take advantage.

10.7.3 The Security Adaptation Ontology

The security adaptation ontology is pre-configured and populated with the necessary knowledge before the system start-up. At this stage of development, we consider pre-determined ontology data based on the expert knowledge. Therefore, ontology modification at this stage refers to updating pointers in the existing knowledge contained as RDF data markups to reflect the current security posture of the monitored environment. Automated knowledge management, e.g., adding new threat members to the *Threat* entity in the ontology, can be done either by utilizing a machine learning technique, which can learn from the situations being dealt with, by employing update service resources or by other related methods. Such management methods are not considered in the study scope.

Furthermore, we have used only abstract level contextual requirements, such as patient's preferred privacy, availability, usability levels and a few service and device physical capabilities for each *Property* used in the ontology. However, these requirements are vital metrics for adaptation and must be carefully understood and designed. Further research is required to analyze and formulate such metrics and measurements that can reflect user preferences, security and services requirements and thing capabilities, as well as how they influence each other during the adaptation process. Potential work to approach in this regards can be security metric and quality of protection (QoP) modeling techniques, such as [30, 46, 47].

10.7.4 Dealing with Advanced Threats

Traditionally, mainly the inbound communications are considered the most concerning for which only preventive controls are employed. However, they may not be sufficient: first, because they may not be suitable for the increasing threat sophistication and, secondly, they mainly focus on the inbound communication [19]. For instance, traditional intrusion detection or prevention systems are based on the concept of intrusion and, in general, are used to detect and analyze inbound communications only. Thus, they lack protection against the insider threat. Furthermore, depending on their architecture, they either analyze network packets or host information. Hence, the scope and context they analyze is limited. Advanced threats, also known as advanced persistent threats (APT) [31], are considered to be highly sophisticated, possibly exploiting zero-days and can be very challenging for a complex and diverse network like the IoT. Therefore, relying on a single analysis technique or method may be insufficient to address the threat landscape faced. We should have appropriate multiple detection capabilities as a second line of defense, which should consider both the outsider and insider threat.

An in-/out-bound activity may trigger a number of events. If we focus only on the activity type, the chances are that we may omit critical events necessary for rigorous analysis. EDAS ensures event-driven analysis where events from multiple activities, irrespective of their type, can be investigated during event correlation. Furthermore, we suggested that any CEP-compatible methods and tools can be utilized, including the traditional solutions, as they may provide various security alerts (events) as an input. Hence, combining different analysis techniques and tools, such as rule-based, profiling and resource reputation measures, statistical methods, behavior analysis, *etc.*, to collectively analyze a situation may appropriately address advanced threats. Lastly, if the events handled in EDAS are adequately logged and managed, they can be utilized as key learning resources for enhancing the underlying analysis techniques.

10.7.5 Utilization in IoT System Architectures

Several system models and architectures have been proposed for IoT. We have discussed a few of them in Section 10.6, such as the Open IoT [42] and IoT-A [11] architectures. One may also find domain-specific system architectures in the literature related to grids, vehicular systems, eHealth, *etc.* These architectures detail technological, system or application perspectives of IoT. They categorize physical objects, applications and services to distinguish and model consumer and service operations. They may describe the means and methods that can be used to communicate between the possible interfaces and how the architecture can be employed as a cloud service or in a particular business or public domain, *etc.* EDAS proposes a security perspective and architecture detailing how a particular service, *i.e.*, security adaptation, can be modeled, such that it can be adopted in any of the mentioned or related IoT system architectures.

As an example, Figure 10.18 instantiates how EDAS can be utilized in the OpenIoT architecture [42] in the context of IoT-eHealth. The first row represents the OpenIoT architecture itself and the distribution of different things, services, applications and utilities it employs at each plane it describes. The second row shows the IoT-eHealth and its corresponding elements as a possible application archetype of the OpenIoT architecture. The last row specifies the possible implementation of EDAS and its components in the OpenIoT. The physical plane will encompass the event sources, *i.e.*, the monitored assets, such as sensors, smart devices and their event frameworks. The virtualized plane will consist of the methods and tools necessary for risk analysis and response, *i.e.*, the EDAS platform. It can either be implemented in the cloud or as a dedicated server inside the hospital-controlled environment. Utilities, such as local adopters, security dashboards and other admin tasks, can be implemented in the utility-app plane. The figure dictates that if adaptive security is desired in an OpenIoT-related architecture, EDAS and

Archirtecture	OpenIoT Architecture Planes: Physical Plane	Virtualized Plane	Utility-App Plane
OpenIoT	Physical Objects, Communication Middlwares, etc.	Cloud Storage/Services	Control/Monitor Utilities and apps
IoT-eHealth	eHealth Assets- Sensors, Actuators, Smart Devices, etc.	Vital Sign Diagnosis-Processing & Storage Resources	Health Dashboard, Smart apps, sensor management utilities
EDAS	Monitored Assets/Sensors/Things (Event Sources), Event Handlers, Communication Agents	The EDAS Platform – Methods, Ontologies, tools and computing resources required For threat (event) monitoring, analysis and adaptation decision processes	Local Adopters, Security Dashboard, Device management interfaces, etc.

Figure 10.18: EDAS utilization in OpenIoT architecture.

its components can be readily implemented in it as illustrated. Furthermore, as previously mentioned, EDAS offers independent and extensible components that can be distributed and employed as per the requirements of a given IoT system architecture, such as the OpenIoT.

10.8 Conclusions

We have explained and evaluated the feasibility of an autonomous event-driven adaptive security prototype based on our proposed architecture. We have expressed how our proposed concept of event-driven security and adaptation ontology ensures context-aware adaptation and complies with device capabilities, as well as user, QoS and security preferences. In the evaluation, there is potential evidence that EDAS, with its event-driven concept and adaptation control loop, is satisfying the requirements needed for managing risk in a dynamic environment, such as the IoT. It provides a real-time risk management platform and ensures autonomous and context-aware adaptive security. We have suggested that traditional security controls are not feasible for IoT in terms of resources and security. However, they can be still used in the traditional settings and can provide input to EDAS for security analysis, as they have a mature set of event frameworks. Hence, EDAS also motivates the merger of traditional enterprise architecture with the thing world: for instance, merging traditional eHealth systems with remote patient monitoring with wearable sensors to make the health system more reachable, accessible and efficient. Critical development and implementation issues, such as secure and reliable event communication, protecting the architecture and metrics required for adaptation, are highlighted, which can be further explored to make EDAS a more reliable solution for IoT-enabled services.

Acknowledgment

This work presented is sponsored by the Adaptive Security in Smart IoT in eHealth (ASSET, 2012–2015) project funded by the Research Council of Norway under Grant Agreement No. 213131/O70. We wish to thank our project colleagues and the anonymous reviewers for their valuable comments and suggestions.

Author Contributions

Waqas Aman and Einar Snekkenes designed the concept and structure of the article. Waqas Aman developed the prototype and performed the scenarios implementation. Waqas Aman and Einar Snekkenes wrote and reviewed the article.

Conflicts of Interest

The authors declare no conflict of interest.

10.9 Bibliography

[1] Alienvault ossim: the open source siem. http://www.alienvault.com/open-threat-exchange/projects. Last access on 21 Dec 2014. Available from: http://www.alienvault.com/open-threat-exchange/projects.

[2] Alienvault unified security management. Last accessed on 24 Apr 2015. Available from: www.alienvault.com/products.

[3] Arq - a sparql processor for jena. http://jena.apache.org/documentation/query/. Last accessed on 21 Dec 2014. Available from: http://jena.apache.org/documentation/query/.

[4] Internet of things architecture (iot-a) project. an eu fp7 project. Available from: www.iot-a.eu.

[5] Java class event handler. https://docs.oracle.com/javase/7/docs/api/java/beans/EventHandler.html. Last accessed on 21 Dec 2014. Available from: https://docs.oracle.com/javase/7/docs/api/java/beans/EventHandler.html.

[6] Mq telemetry transport, mqtt. http://mqtt.org/. Last accessed 21 Dec 2014. Available from: http://mqtt.org/.

[7] Open source cloud solution for the internet of things. openiot project. Available from: http://www.openiot.eu/.

[8] Owasp internet of things top 10 project. Last accessed on: 24-Feb-2015. Available from: https://www.owasp.org/index.php/OWASP_Internet_of_Things_Top_Ten_Project.

[9] Security information & event management (siem) solutions arcsight. Last accessed on: 24 Apr 2015. Available from: http://www8.hp.com/us/en/software-solutions/siem-security-information-event-management/.

[10] System logger: Syslog linux man page. http://linux.die.net/man/3/syslog. Last access date: 31 May 2014. Available from: http://linux.die.net/man/3/syslog.

[11] Project deliverable d1.2-initial architectural reference model for iot. Tech. rep., IoT-A Project, June 2011. Last accessed on 24 Apr 2015. Available from: http://www.iot-a.eu/public/public-documents/documents-1/1/1/d1.2/at_download/file.

[12] ABOWD, G. D., DEY, A. K., BROWN, P. J., DAVIES, N., SMITH, M., AND STEGGLES, P. Towards a better understanding of context and context-awareness. In *Handheld and ubiquitous computing* (1999), Springer, pp. 304–307.

[13] ABOWD, G. D., AND MYNATT, E. D. Charting past, present, and future research in ubiquitous computing. *ACM Transactions on Computer-Human Interaction (TOCHI) 7*, 1 (2000), 29–58.

[14] AMAN, W., AND SNEKKENES, E. An empirical research on infosec risk management in iot-based ehealth. In *MOBILITY 2013, The Third International Conference on Mobile Services, Resources, and Users* (2013), pp. 99–107.

[15] AMAN, W., AND SNEKKENES, E. Event driven adaptive security in internet of things. In *UBICOMM 2014, The Eighth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies* (2014), pp. 7–15.

[16] ATZORI, L., IERA, A., AND MORABITO, G. The internet of things: A survey. *Computer Networks 54*, 15 (2010), 2787–2805.

[17] BARBACCI, M., KLEIN, M. H., LONGSTAFF, T. A., AND WEINSTOCK, C. B. Quality attributes. Tech. rep., DTIC Document, 1995.

[18] CACERES, R., AND FRIDAY, A. Ubicomp systems at 20: Progress, opportunities, and challenges. *IEEE Pervasive Computing*, 1 (2011), 14–21.

[19] COLE, E. Advanced persistent threat (apt) and insider threat, October 2012. Last accessed on 24 Apr 2015. Available from: `http://goo.gl/ZdOWvO`.

[20] EMC, AND CORPORATION, I. D. The digital universe of opportunities: Rich data and the increasing value of the internet of things, April 2014. Last accessed on 21 Dec 2014. Available from: `http://www.emc.com/leadership/digital-universe/2014iview/internet-of-things.htm`.

[21] EVESTI, A., SUOMALAINEN, J., AND OVASKA, E. Architecture and knowledge-driven self-adaptive security in smart space. *Computers 2*, 1 (2013), 34–66.

[22] FIELDING, R. T. *Architectural styles and the design of network-based software architectures*. Ph.D. thesis, University of California, Irvine, 2000.

[23] FULP, E. Parallel firewall designs for high-speed networks. In *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings* (April 2006), pp. 1–4.

[24] GANEK, A. G., AND CORBI, T. A. The dawning of the autonomic computing era. *IBM systems Journal 42*, 1 (2003), 5–18.

[25] HULSEBOSCH, R., BARGH, M. S., LENZINI, G., EBBEN, P., AND IACOB, S. M. Context sensitive adaptive authentication. In *Smart Sensing and Context*. Springer, 2007, pp. 93–109.

[26] KARG, D. Ossim correlation engine explained. `https://www.alienvault.com/docs/correlation_engine_explained_worm_example.pdf`, August 2004. Last accessed on 21 Dec 2014. Available from: `https://www.alienvault.com/docs/correlation_engine_explained_worm_example.pdf`.

[27] KAZMAN, R., BASS, L., WEBB, M., AND ABOWD, G. Saam: A method for analyzing the properties of software architectures. In *Proceedings of the 16th international conference on Software engineering* (1994), IEEE Computer Society Press, pp. 81–90.

[28] KAZMAN, R., KLEIN, M., AND CLEMENTS, P. Atam: Method for architecture evaluation. Tech. rep., DTIC Document, 2000.

[29] KEPHART, J. O., AND CHESS, D. M. The vision of autonomic computing. *Computer 36*, 1 (2003), 41–50.

[30] KSIEZOPOLSKI, B., ZUREK, T., AND MOKKAS, M. Quality of protection evaluation of security mechanisms. *The Scientific World Journal 2014* (2014).

[31] LAWRENCE PINGREE, NEIL MACDONALD, P. F. Best practices for mitigating advance persistant threats. http://goo.gl/StNQEz, September 2013. Last accessed on: 24 Apr 2015.

[32] MA, H.-D. Internet of things: Objectives and scientific challenges. *Journal of Computer science and Technology 26*, 6 (2011), 919–924.

[33] MACDONALD, N. The future of information security is context aware and adaptive. *Gartner RAS Core Research Note G 200385* (2010).

[34] MCGRAW, R. W. Risk adaptable access control. http://csrc.nist.gov/news_events/privilege-management-workshop/radac-Paper0001.pdf, 2009. Last accessed on: 24 Apr 2015.

[35] METZGER, A., CHI, C.-H., ENGEL, Y., AND MARCONI, A. Research challenges on online service quality prediction for proactive adaptation. In *Software Services and Systems Research-Results and Challenges (S-Cube), 2012 Workshop on European* (2012), IEEE, pp. 51–57.

[36] MICHELSON, B. M. Event-driven architecture overview. *Patricia Seybold Group 2* (2006).

[37] MICROSOFT. Event properties. Last accessed on: 24 Apr 2015. Available from: https://technet.microsoft.com/en-us/library/cc765981.aspx.

[38] MICROSOFT. Handling and raising events. Microsoft Developer Network. Last accessed on 21 Dec 2014. Available from: https://msdn.microsoft.com/en-us/library/edzehd2t(v=vs.110).aspx.

[39] MICROSOFT. *Chapter 16- Quality Attributes. Microsoft Application Architecture Guide*, 2nd edition ed. November 2009. ISBN: 9780735627109.

[40] NICHOLAS, S. Power profiling: Https long polling vs. mqtt with ssl, on android. http://stephendnicholas.com/archives/1217, May 2012. Last accessed on 21 Dec 2014. Available from: http://stephendnicholas.com/archives/1217.

[41] O'BRIEN, L., MERSON, P., AND BASS, L. Quality attributes for service-oriented architectures. In *Proceedings of the International Workshop on Systems Development in SOA Environments* (Washington, DC, USA, 2007), SDSOA '07, IEEE Computer Society, pp. 3–9. Available from: http://dx.doi.org/10.1109/SDSOA.2007.10.

[42] PANAGIOTIS DIMITROPOULOS, JOHN SOLDATOS, N. K. J. E. B. A. G. D 2.3 openiot detailed architecture and proof-of-concept specifications. Tech. rep., March 2013. Last accessed on: 24 Apr 2015. Available from: http://www.openiot.eu/sites/all/themes/corporateclean/Files/OpenIoT_D23.pdf.

[43] PIETERS, W. Security and privacy in the clouds: a birds eye view. In *Computers, privacy and data protection: An element of choice*. Springer, 2011, pp. 445–457.

[44] PROTEGE. A free, open-source ontology editor and framework for building intelligent systems. Last Accessed on: 05-March-2015. Available from: http://protege.stanford.edu/.

[45] RSA. Rsa adaptive authentication. a comprehensive authentication and fraud detection platform, 2012. Last accessed on: 24 Apr 2015. Available from: http://www.emc.com/collateral/data-sheet/h11429-rsa-adaptive-authentication-ds.pdf.

[46] RUSINEK, D., KSIEZOPOLSKI, B., AND WIERZBICKI, A. Security trade-off and energy efficiency analysis in wireless sensor networks. *International Journal of Distributed Sensor Networks 501* (2015), 943475.

[47] SAVOLA, R. M., AND HEINONEN, P. Security-measurability-enhancing mechanisms for a distributed adaptive security monitoring system. In *Emerging Security Information Systems and Technologies (SECURWARE), 2010 Fourth International Conference on* (2010), IEEE, pp. 25–34.

[48] SHACKLEFORD, D. Real-time adaptive security. Tech. rep., SANS, December 2008. Last Accessed on 21 Dec 2014. Available from: http://www.sans.org/reading-room/whitepapers/analyst/real-time-adaptive-security-34740.

[49] SMITH, C., AND MIESSLER, D. The internet of things research study, September 2014. Last accessed on: 24-Feb-2015. Available from: http://www8.hp.com/h20195/V2/GetPDF.aspx/4AA5-4759ENW.pdf.

[50] STEWART, A. On risk: perception and direction. *Computers & Security 23*, 5 (2004), 362–370.

[51] SUNDMAEKER, H., GUILLEMIN, P., FRIESS, P., AND WOELFFLÉ, S. Vision and challenges for realising the internet of things.

[52] SZYPERSKI, C. *Component Software: Beyond Object-Oriented Programming*, 2nd ed. Addison-Wesley Longman Publishing Co., Inc., 2002.

Article 5

Managing Security Trade-offs in the Internet of Things Using Adaptive Security

Waqas Aman and Einar Snekkenes
In The 10th International Conference for Internet Technology and Secured Transactions (ICITST-2015), pages 362–368 London UK, 2015.

Chapter 11

Managing Security Trade-offs in the Internet of Things Using Adaptive Security

Abstract

Adaptive security can take dynamic trade-off decisions autonomously at runtime and is considered a key desirable attribute in the Internet of Things (IoT). However, there is no clear evidence that it can handle these trade-offs optimally to add value to such a complex and dynamic network. We present a scenario-based approach to recognize and evaluate typical security trade-off situations in the IoT. Using the Event-driven Adaptive Security (EDAS) model, we provide the assessment of dynamic trade-off decisions in the IoT. We have showed that an optimum trade-off mitigation response in the IoT can be automated by assessing various contextual requirements, such as the QoS and user preferences, thing capabilities, and the risk faced, at runtime. eHealth scenarios are examined to illustrate system application in IoT-based remote patient monitoring systems.

Keywords-Internet of Things; Adaptive Security; eHealth; Event Driven Architecture.

11.1 Introduction

IoT has a huge potential to facilitate the growth of our economy and society by digitizing commercial enterprises and public infrastructures. The European Commission envisions the market value of IoT to be one trillion euros by the year 2020 [7], yet alone in the Europe. IoT aims to connect diverse technologies, objects, services and people to achieve particular objectives. This interconnection introduces heterogeneity, complexity and dynamic elements in the concerning service architecture.

From a security perspective, these heterogeneous things in the IoT ecosystem have their inherited vulnerabilities and connecting them together will open a multitude of new means and opportunities for the adversaries. Hence,

this diversity makes the IoT threat landscape more complex though provides flexibility. Such a broad threat spectrum may not be addressed by the conventional security controls as they are designed to protect against a particular threat context, such as particular files or network packets. Their risk mitigation strategies are primarily focused on asset protection and do not consider other factors, such as resource capacity, QoS requirements, and user preferences, which are critical for a user-centric IoT-based service. The resulting decisions can be inflexible and inefficient and may negatively influence the monitored service. Furthermore, due to the increasing number of objects per user in the IoT [6], it will be relatively difficult to implement manual risk management activities.

The mentioned problems motivate autonomic security adaptation, a key desirable attribute in IoT-enabled smart environments [11]. In the IoT, adaptive security can be employed to achieve a cost-effective trade-off decision to reduce risks faced at runtime. Such attempts will significantly improve the overall service reliability as it would appraise all the potential factors affecting or affected by the decision. However, due to the IoT architectural complexity, it is challenging to recognize, assess and model potential trade-off situations using adaptive security. To address adaptive security in IoT, we have proposed and analyzed the feasibility of Event-driven Adaptive Security (EDAS) architecture in [3] and [4]. In this article, we explicate a scenario-based method to evaluate various security tradeoffs using EDAS. Our emphasis is to investigate two essential questions: i) What typical trade-off situations exist in the IoT? And, ii) To what extent does the EDAS adaptive security loop add value to autonomic risk management in the IoT?

We have found that by using EDAS, security adaptation in IoT can be effectively automated by utilizing a scenario-based approach. The mitigation response it adapts examines all the potential contextual requirements, i.e. QoS requirements, user preferences, resource capacity, and threat level. Hence, the response it adapts reflect an optimum trade-off decision as it weighs all the influencing factors and selects the one which has a maximum utility. Furthermore, the approach used in this article will empower system analysts and developers to identify and evaluate key pre-development requirements, e.g. context awareness essentials, trade-off metrics, and conflicts, programming aspects, etc., that are critical for engineering event-driven adaptive security. Moreover, it is realized that a more precise set of trade-off metrics need to be developed and analyzed to capture the contextual requirements accurately and for the adaptation decision to be more efficient. IoT-enable eHealth scenarios are investigated to reflect EDAS application.

The rest of the paper is organized as follows. A brief introduction to EDAS and the approach used in this paper is given in Section 11.2. The IoT-eHealth scenarios and corresponding trade-offs are briefly described in Section 11.3. Section 11.4 details a schema of how the scenarios and trade-

offs can be modeled in the EDAS. In Section 11.5, we will discuss some of the adaptation concerns and will relate them to work done in the literature. Finally, the article is concluded in Section 11.6.

11.2 Architecture and Approach

This section briefly introduces the EDAS model and describes the approach used to recognize and assess the potential trade-offs using EDAS.

11.2.1 The EDAS Model

An Event-driven Architecture (EDA) collects, analyzes and reacts to significant changes, events, in the monitored network. Monitoring these events provide a holistic visibility of the operations across the network. The primary feature offered by an EDA is loose-coupling which enables the system components to operate independently [13]. Hence, it offers flexibility, interoperability and extensibility in the design, which are highly desirable attributes in IoT-related architectures. The Event-driven Adaptive Security (EDAS) is an autonomic security adaptation model based on EDA [3]. Its reference model is depicted in Fig. 11.1. EDAS monitors, analyzes and responds to security threats (*thing*-generated events) using a continuous control feedback loop [5]. The *Risk Monitor* component collects, filters and normalizes events, before or after adaptation, emerging from the monitored *Event Sources* (*things*) in the IoT. The *Risk Analyzer* investigates different events in a context by correlating them for possible threats and raises a risk alarm when a threat discovered has a risk level beyond the threshold. The *Risk Adapter* utilizes a runtime adaptation ontology and responds to an alarm by selecting an optimal response as per the contextual requirements. The model is extended to a technical specification of a system architecture, and its feasibility is investigated as a real-world artifact in [4]. A description of its major components is given in Table 11.1. The relationships among these components is detailed in Section 11.4.

11.2.2 Towards Adaptive Security

We consider an adverse security scenario as a trade-off situation as there is always *security vs. some attribute* trade-off involved when mitigation actions are adapted to reduce the risk faced. The two-phased approach used to elicit and analyze engineering fundamentals of EDAS is depicted in Fig. 11.2 and is briefly described as follows:

The *Phase-1* focuses on the knowledge elicitation and evaluation required to identify, assess, and respond to potential threats in the corresponding scenarios. This knowledge includes threats, critical event sources, event correlation contexts, participating events, risk analysis methods, supported

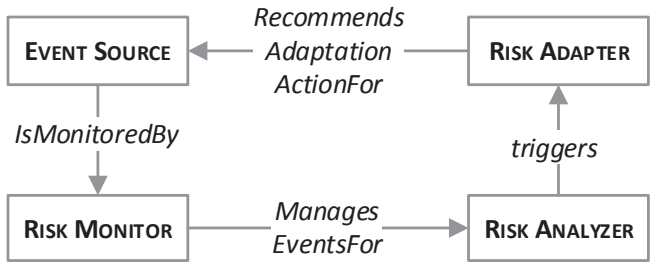


Figure 11.1: EDAS Reference Model

Table 11.1: A Description of EDAS Components

	Entity	Description
Event Source	Thing	A physical asset in the monitored IoT ecosystem
	Object	a software module of a *Thing* e.g. a temperature sensing module
	Security Component	Security Mechanism e.g. algorithms, used by an *Object*
	Event	A potential change in the *Thing* environment raised by an *Object*
	Event Framework	The *Event* handler and logger
	Local Adopter	A software module that instructs the execution the adaptation decision locally
	Adapt Request	The adaptation decision/action (risk mitigation response) to be adopted locally
Monitor	Monitoring Agent	A software component that collect, filter and transform events
	Filtration Criteria	An event filtration rules
	Normalization Criteria	Event transformation rules
Analyzer	Alarm	Risk alert detailing risk beyond acceptance
	Risk Scorer	Event risk quantifier and *Alarm* generator
	Risk Metric	A measure based on which risk is quantified, e.g. an asset or event importance value
	Threat Context	A marker specifying a particular risk situation
	Correlation Directive	A container for a rule set that directs risk manipulation for a *Threat Context*
	Correlation Criteria	Rules that correlate events in time and space
Adapter	Action	A possible risk mitigation response
	Mechanism	A vocabulary of the monitored ecosystem's security method, e.g. routing or encryption algorithms
	Property	A vocabulary of the attributes inherited by the *Mechanism*, e.g. key length
	Utility Metric	A trade-off factor influencing or influenced by a property to be adopted
	Utility	A positive integer indicating the extent to which a Metric is supported
	Risk Level	Risk impact level
	Contextual Requirement	Preferences and capabilities in a particular operational environment/context

adaptable actions, trade-off metrics, conflicting scenarios and their resolution approaches against the individual scenarios.

The *Phase-2*, scenario modeling, in this article, refers to the pre-development realization of the knowledge gathered in the Phase-1. It is performed by populating the adaptive security system model with the knowledge extracted. The realization can serve as an implementation guideline for the analysts and developers, and assist them in identifying and evaluating different development paradigms for each scenario.

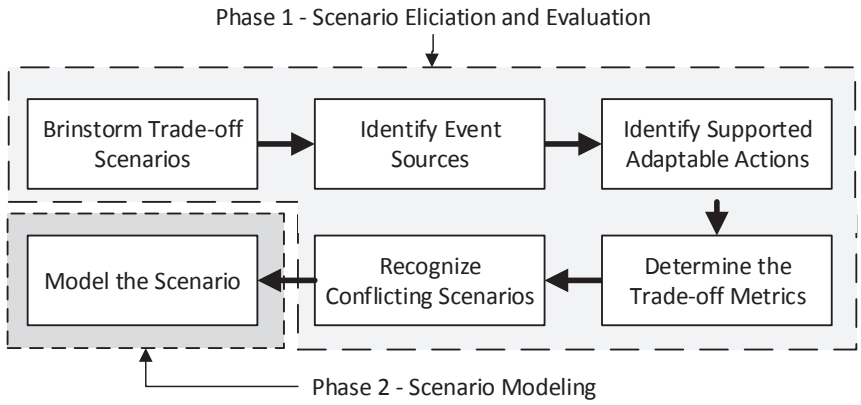


Figure 11.2: A Scenario-based Approach Towards Adaptive Security

11.3 Scenarios and Adaptation Trade-offs

In this section, we instantiate the Phase-1 of the approach with a few typical IoT-eHealth scenarios to highlight the trade-offs and to reflect on the overall process. We extend the IoT-eHealth case study in [3] and add different scenarios to narrate various real-world security incidents.

11.3.1 The IoT-eHealth Case Study and Scenarios

A hypoxemic patient at home, Lynda, equipped with an Oximeter is monitored from a remote hospital site. She has a smart device capable of mobile and internet-based communication. It has some general purpose sensors, such as a GPS sensor, and is used in activities like conferencing with the physicians, viewing health stats and prescriptions, billing and payments. Moreover, it acts as a relay access point between the sensors and the hospital and ensures that vital body signs are available during outdoor activities.

Scenario 1 Resource optimization during mobility: Before going outdoors for a prescribed exercise, Lynda changes the smartphone settings from WiFi to Mobile-Data indicating a change in operational context. As increased encryption consumes more power and memory, confidentiality has to be reduced as per the utility to ensure long-term data availability

Scenario 2 - Max. Confidentiality in Possible Intrusions: Assuming discovering unregistered radio devices as a threat to confidentiality, the patient requirements and the hospital policy dictate that confidentiality has to be increased in to avoid any possible compromise. This scenario is identified as *2a* and *2b* in the home and outdoor operational contexts respectively.

Scenario 3 - Handling a *thing* Compromise: The network component of the eHealth app on the patient smart device has somehow been compromised. The app has generated events indicating that a new destination has been added to the address list.

Scenario 4 - Repeated Wrong Login Attempts: An adversary having physical access to Lynda's smartphone is trying random passwords to login into the eHealth app installed to steal the banking information stored in it.

Scenario 5 - Physician Account Compromise: A Physician has successfully logged on to the Electronic Health Record (EHR) server from his machine. However, no such record is found in the employee attendance (RFID) server. Besides a technical fault, the situation indicates that the account might have been compromised.

Scenario 6 - Service Unavailability: The EHR server at the hospital, the primary destination for the remotely collected vital signs, suddenly goes down due to a technical fault. In such situations, the smart device has to store vital sign information locally.

Table 11.2 depicts the organization of the adaptation knowledge obtained in Phase-1. Fig. 11.3 shows a general view of the primary trade-offs involved in each scenario with the possible adaptation actions having distinct utilities in a trade-off as per the contextual requirements. In EDAS, an adaptable action comprises a *security mechanism* and its *property*, such as the AES encryption algorithm and its 128-bit key length property, supported by a particular event source. At a given time in a particular operational context, a property addresses a particular risk level. Metrics influenced in a trade-off, as shown in Table 11.2, are derived from the contextual requirements and are weighed against each property to reflect its overall utility and are different in different operational contexts. All these elements will be further explored in the next section to reflect on how they are addressed in EDAS.

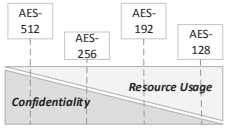
As an example, in Table 11.2, we have identified two conflicting scenarios (1 and 2b) as both will compete for the conflicting requirements, i.e. availability and confidentiality, in outdoor situations. Some conflicts may not be critical and can easily be resolved by simple *if-else* related techniques. For instance, one can ignore scenario 2b if scenario 1 has already occurred as it has, comparatively, more importance for service and user. However, other conflicts might need in-depth investigations requiring more sophisticated resolution mechanisms.

11.4 Scenario Modeling

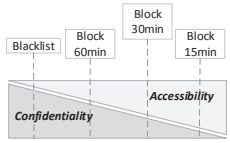
Scenario modeling serves two primary purposes. First, it provides a platform for the analysts to realize the knowledge evaluated in Phase-1 and assists in identifying any missing information. Thus, it further evaluates the adaptation knowledge required to analyze a threat scenario. Secondly,

Table 11.2: Scenario Elicitation and Evaluation

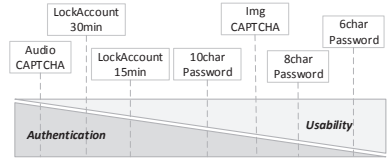
Sc. No.	Opr. Context	Associated Threat	Possible Event Sources	Supported Adaptable Actions	Supported Adaptable Mechanism[Properties]	Trade-off Metrics	Conflict
1	Outdoor	Data Unavailability	Oximeter, Smart phone	Change Cipher Change Cipher Key Length	Cipher[AES] Keylength [128, 192, 256, 512]	Efficiency, Resource Usage, Confidentiality	2b
2a	Home	Privacy & Confidentiality Breach	Oximeter, Dev Detector Sensor	Change Cipher Change Cipher Key Length	Cipher[AES] Keylength [128, 192, 256, 512]	Efficiency, Resource Usage, Confidentiality	
2b	Outdoor						1
3	Indoor, Outdoor Hospital	Information Hijacking	Smart phone, Management Server	Block ID/Address	Permanent[blacklist], Temporary[15min, 30min, 60min]	Accessibility, Confidentiality	
4	Indoor, Outdoor Hospital	Password Guess/Brute force Attack	Smart phone	Change Password Length, Lock Account, Enforce CAPTCHA	Length[8char, 10char], Lock Time[15min, 30min], CAPTCHA[Audio, Image]	Memorability, EaseOfUse, Accessibility, Authentication, Resource Usage	
5	Hospital	Intrusion	RFID Server, EHR Server	Change Account settings	LockAccount[15min, 30min],	Accessibility, Authentication	
6	Home, Outdoor, Hospital	Service Unavailability	Smart phone, EHR Server	Activate Local Cache	Cache Size[50MB, 100MB, 200MB]	Distress, Memory, Uptime, Energy usage	



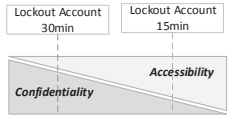
(a) Scenario 1 and 2a



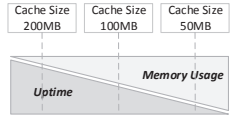
(b) Scenario 3



(c) Scenario 4



(d) Scenario 5



(e) Scenario 6

Figure 11.3: Scenarios, Primary Trade-offs, Adaptation actions & their utilities

it will provide a guideline for the developers to better understand problem (scenario) requirements for implementation and will facilitate them to identify and evaluate different programming techniques.

Taking scenario 1 and 2b as examples, we illustrate the scenario modeling and reflect on how the corresponding knowledge relates to each other. We present two illustration views: Fig. 11.4 depicts a tabular description of the concerning relations in Event Source whereas, Fig. 11.5-11.7 provides a conceptual view of the corresponding components. These figures extend the reference model (Fig. 11.1), provide a blueprint of the relationship between the major components, and describe how the extracted knowledge in Phase-1 can be structured for EDAS implementation.

The Event Source represents the monitored resource in the ecosystem. It consists of a physical asset (a *thing* in the IoT), and application specific objects. These objects generate events using their event framework facility and send them to the remote EDAS platform for threat analysis. The platform includes the risk monitor, analyzer and adapter components. An object does not take the adaptation decision by itself, but receives it as a request from the adapter via the local adopter and implements it locally. Using the scenarios knowledge, the relation between the Event Source components is shown in Fig. 11.4.

In Fig. 11.5, the strings starting from *Acpt* can be considered as regular expressions (RegEx) or rules to be designed to accept a particular event for further analysis. Normalization rules apply specific transformation rules to each event, depending upon its origin and importance, for further analysis. These strings and tags in the modeling provide a precise instruction set for the developers to construct the essential components. Therefore, this schematic modeling reduces communication gap between system analysts, architects, and developers, and speeds up the engineering process.

Each normalized event from the Monitor has associated risk metrics based on which the *Quantifier* object in the Risk Analyzer, see Fig. 11.6, calculates its risk. These metrics may also be modified during event correlation. Event correlation can also be used to investigate and resolve any conflicting scenarios. For instance, the *Correlation Criteria*, in Fig. 11.6, resolves the conflict between Scenario 1 and 2b by correlating the operational context. Moreover, it can be noticed that *Encrypt-Key-Change-Event* is also participating in the correlation contexts. Depending on the context, it represents the event that has been raised by the Oximeter sensing object after new encryption key lengths are adapted and is correlated in the same threat context to ensure that the threat has been addressed, and that the corresponding risk level has been reduced as per the contextual requirements. The `INCREASE` and `NORMALIZE` keywords specify the particular function calls or related equations that can be employed to manipulate the risk level as per the acceptance threshold. Furthermore, as event correlation intends to analyze events from

EVENT SOURCE

Relation	Components/Entities and Member/Objects	
has	**Entity: thing**	**Entity: Object**
	Oximeter	Sensing-Object
	Smart Device	StatusNotifier app
		DeviceDetector app
adopts	**Entity: Object**	**Entity: SecurityComp**
	Sensing-Object	AES[128, 192, 256, 512]
	StatusNotifier app	
	DeviceDetector app	
instructs	**Entity: Local Adopter**	**Entity: Object**
	Request Parser API Caller	Sensing Obj
		StatusNotifier app
		DeviceDetector app
handles	**Entity: Local Adopter**	**Entity: AdaptRequest**
	Request Parser API Caller	[***Action**: Change Cipher KeyLength, **Mechanism**: AES, **Property**: 192/512-Bit, **app_id***]
Triggers	**Entity: Object**	**Entity: Event Framework**
	Sensing app	OxiSens-Event Framework-obj
	StatusNotifier app	StatusNot- Event Framework-obj
	DeviceDetector app	DevDetector- Event Framework-obj
Generates	**Entity: Event Framework**	**Entity: Event**
	OxiSens-Event Framework-obj	Encrypt-Key Change-Event
	StatusNot- Event Framework-obj	Context-Change-Event
	DevDetector- Event Framework-obj	Unregistered-Dev-Found-Event

Figure 11.4: Event Source (tabular view)

different sources, it may include other sources which might not be a direct target in the threat faced but may provide essential information for correlation. Thus, the correlation criteria modeling enables the analysts to discover and assess other sources that may be critical in analyzing scenarios.

The Risk Adapter components, as shown in Fig. 11.7 (excluding the *Object* and *Alarm*), are the necessary vocabulary in the adaptation ontology proposed in [4]. It is accessed as per the scenario to formulate the adaptation (trade-off) decision which is sent to the Event Source as an adaptation request (*AdaptRequest*).

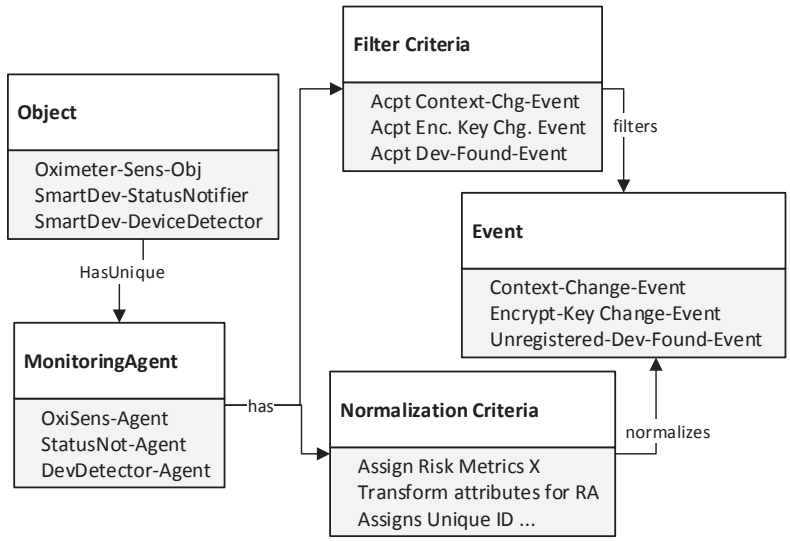


Figure 11.5: Risk Monitor (conceptual view)

11.4.1 Managing Trade-offs

Taking decisions always involves one or more trade-offs. The corresponding influences can sometimes be very low and can be ignored. For instance, while weighing various security metrics for an adaptation action to appropriately control access, e.g. changing a password length to 10 characters, the data integrity or confidentiality metrics can be disregarded as it has no significant influence on the decision. However, there will be situations that will require careful assessment of the influencing parameters to address all the potential requirements appropriately.

In EDAS, factors involved in a trade-off are considered as utility metrics, as shown in Fig. 11.7. They are derived from the contextual requirements identified in the monitored IoT ecosystem, i.e. user preferences, QoS requirements, and *thing* resources, and can have different utilities in different operational contexts. For instance, confidentiality, integrity, and availability requirements may differ significantly in outdoor contexts because of the adverse elements in the environment as compared to the home context. However, the usability requirements might remain the same in almost all contexts. For each property used in an action, these metrics are assessed, i.e. assigned a utility (a positive integer) by experts based on the property's competence against the threat and its influence on the contextual requirements. The greater the integer value is, higher is the utility of a metric. This

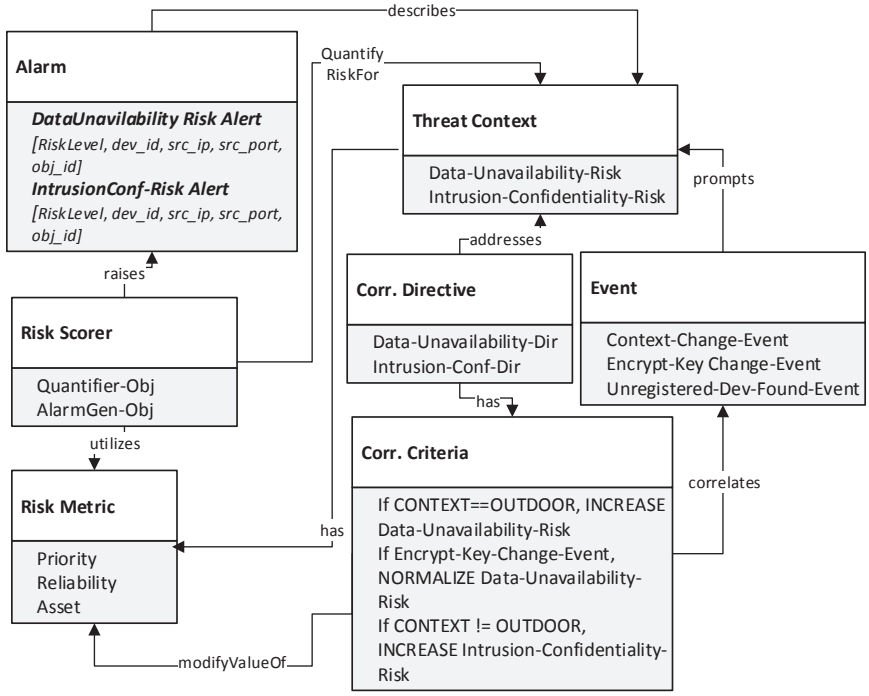


Figure 11.6: Risk Analyzer (conceptual view)

assessment facilitates the system, i.e. the Risk Adapter in EDAS, to take a trade-off decision that has a maximum utility in a particular threat scenario and is, thus, an optimum response.

As examples, depicted in Table 11.3-11.5, we illustrate how some trade-offs concerning Scenario 1, 2a, 4 and 6 can be handled in a security adaptation decisions. It can be noticed that we have expanded the primary trade-offs (in Fig. 11.3) to influencing metrics at the abstract level in each scenario to address the possible contextual requirements. However, in practice, these metrics should reflect all the influencing and influenced contextual pre-requisites for the decision to be more effective. The property with the highest total utility is selected, shaded out in gray, as the most cost-effective mitigation action to confront a threat in a scenario.

11.5 Discussion and Related Work

In this section, we discuss a few concerns, such as the trade-off metrics assessment and design restrictions, and relate them to similar work in the lit-

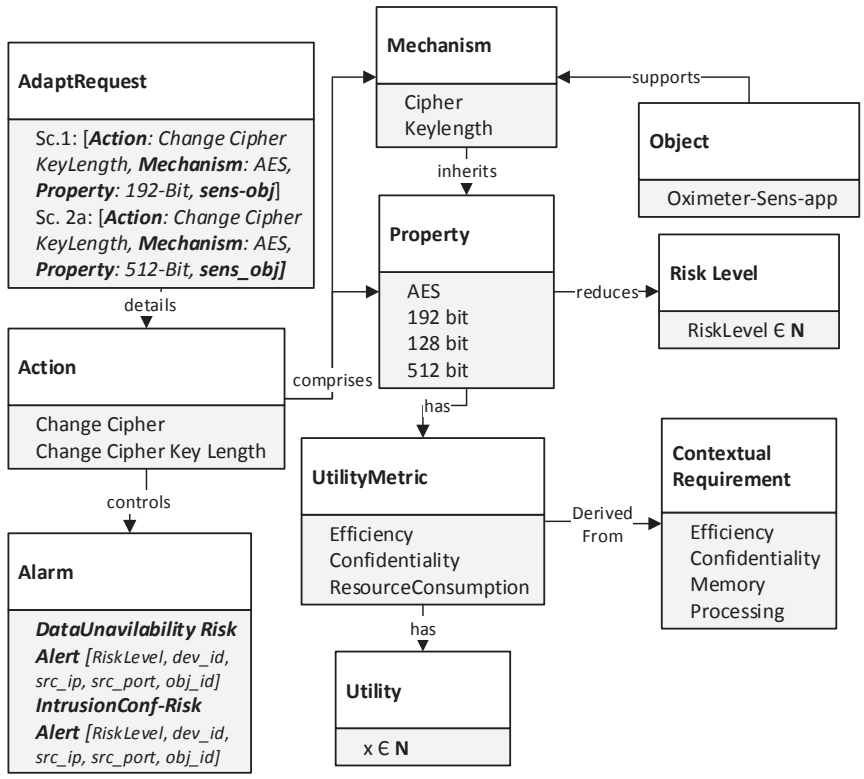


Figure 11.7: Risk Adapter (conceptual view)

erature to comprehend how the concepts proposed in the EDAS or related work can benefit from each other to make adaptive security a more reliable solution for the IoT.

11.5.1 Trade-off Metrics Assessment

At this stage of EDAS development, we have not investigated any particular trade-off metrics. However, our scenario-based approach suggests how they can be recognized in the IoT. We emphasize that all contextual requirements should be identified in potential operational contexts and should be categorized rigorously to capture the actual needs. A rigor classification of the requirements will result in a precise set of trade-off metrics and will make the adaptation decision more realistic and, therefore, effective. For instance, a patient usability preference should be further extended to other factors, such as learnability, memorability, ease of use, satisfaction, etc., to carefully

Table 11.3: Trade-off Assessment - Scenario 1 and 2a (Security = Confidentiality). Assuming 256-bit key is used before adaptation

Tradeoff/Utility Metric	***Scenario 2a Context = Home/Hospital***				***Scenario 1 Context = Outdoor***			
	Mechanism =AES-Key Length							
	Properties							
	128-bits	*192-bits*	*256-bits*	*512-bits*	*128-bits*	*192-bits*	*256-bits*	*512-bits*
Security	10	15	18	21	10	15	18	21
Efficiency	15	14	13	12	15	14	13	12
Resource Usage	17	16	15	14	17	14	10	6
Total Utility	***42***	***45***	***46***	***47***	***42***	***43***	***41***	***39***

Table 11.4: Trade-off Assessment - Scenario 4 (Security = Authentication)

Trade-off/Utility Metric	***Mechanisms***					
	Key Length		**CAPTCHA**		**Time Restriction**	
	Properties					
	8-Char	*10-Char*	*Image*	*Audio*	*15min*	*30min*
EaseOfUse	10	8	20	18	10	5
Memorability	15	10	2	2	2	2
Accessibility	10	7	20	10	10	5
Security	10	15	10	12	10	15
Resource Usage	12	12	8	5	12	12
Total Utility	***57***	***52***	***60***	***47***	***44***	***39***

Table 11.5: Trade-off Assessment - Scenario 6 (Uptime = Security)

Trade-off/Utility Metric	*Context = Home/Hospital*			*Context = Outdoor*		
	Mechanisms = Cache Size					
	Properties					
	50MB	*100MB*	*200MB*	*50 MB*	*100MB*	*200MB*
Distress	20	10	5	15	8	4
Uptime	15	25	30	10	15	20
Memory	20	10	5	20	15	10
Energy Usage	10	10	10	15	10	5
Total Utility	***65***	***55***	***50***	***60***	***48***	***39***

address his preferences in concerning scenarios.

The metrics assessment method during adaptation decision is also critical. Since the primary objective of EDAS was to provide a holistic autonomous security architecture, we did not investigate the effectiveness of its utility-based metric assessment. Although, it does offer a rationale for optimized adaptation decision, we have yet to explore it further for any improvements. In this context, methods from game theory [14], expected utility theories, machine learning, and related studies may provide significant and useful perspectives.

Depending on the organizational policy, the selection of a property can be approached in two ways. If the total utility of two or more properties has the same value, one of them can be randomly adapted as it implies that they all have the same maximum utility in a given context. Otherwise,

conflicts may arise due to utility overlapping which will necessitate more sophisticated assessment methods as mentioned earlier. Therefore, more meaningful and structured values (utilities) should be established to weigh individual metrics. In this regard, methods defined in [12], [15] and [9] can be potentially reviewed for developing and estimating metrics.

11.5.2 The Evaluation Approach

Similar evaluation frameworks can be found in the literature assessing different security and privacy aspects in information systems. The Architecture Tradeoff Analysis Method (ATAM) [8] suggested a scenario-based approach to analyze design approaches addressing various QoS attributes in software architectures. A similar approach is used in [2] where the authors utilized a scenarios-based method to evaluate the security of a software architecture. Recently, a more relevant evaluation framework is suggested by Liester et.al. [10]. The authors have provided an extensive list of IoT-eHealth scenarios as various system states. Linear and logarithmic approaches were utilized to assess and quantify their security and QoS requirements in an adaptive security system. Our approach complements their work and emphasizes to actively consider user preferences and devices capabilities besides QoS and security requirements to make the adaptation decision more effective. Furthermore, our approach tends to model the requirements in a way such that they can be easily and readily employed in the system development and implementation.

11.5.3 Architectural Constraints

From an architectural viewpoint, not every object is adaptable. In EDAS, only those objects can be adapted which utilize a flexible security component. Although, some objects are critical to security, they are only used to collect essential events for establishing context-aware analysis, e.g. a GPS module. Such objects may not use any security component. Others may have only a single supported security component, e.g. a DES-128 bits encryption algorithm. Apparently, in such cases, security adaptation does not seem to be practical. However, a possible trade-off in such scenarios can be that of a *zero encryption level* indicating an adaptation decision that instructs to drop any security mechanism in use. Evidently, this is not an efficient protection strategy, but can be useful in situations where confidentiality is not the primary objective, e.g. outdoor emergency scenarios where the patient's data availability is more critical than its confidentiality. To ensure flexible and more optimized adaptation, other design elements, such as the sensor middleware in the Global Sensor Network (GSN) [1] and related middlewares, could be introduced into the architecture. Such middlewares can

be used to offer flexible security components as services for objects having none or reduced security components.

11.6 Conclusion

Adaptive security is a desirable attribute in the IoT where the threat landscape is more complex and dynamic. In this paper, we have provided a scenario-based method that will facilitate system architects, analysts and developers to identify and evaluate different aspects of engineering event-driven adaptive security in the IoT. Using event-driven adaptive security (EDAS), and a few typical IoT-eHealth scenarios, we have provided essential knowledge that optimal trade-off adaptation decisions can be employed in the IoT to defend against a risk faced. Therefore, it is made evident that adaptive security can improve autonomous risk management in the IoT by adequately addressing the trade-offs. We have utilized a utility-based assessment method to deal with the trade-off metrics involved in a decision. Since these metrics are derived from the monitored ecosystem requirements, an adaptation decision evaluating these requirements results in a trade-off decision which is the most effective in a given scenario. The assessment provides a convincing basis for making dynamic trade-off decisions.

Acknowledgment

This work is supported by the Adaptive Security for Smart Internet of Things in eHealth(ASSET, 2012/15) project funded by the Norwegian Research Council under Grant Agreement no. 213131/O70. We wish to thank our colleagues and the anonymous reviewers for their expert opinion.

11.7 Bibliography

[1] ABERER, K., HAUSWIRTH, M., AND SALEHI, A. A middleware for fast and flexible sensor network deployment. In *Proceedings of the 32nd international conference on Very large data bases* (2006), VLDB Endowment, pp. 1199–1202.

[2] ALKUSSAYER, A., AND ALLEN, W. H. A scenario-based framework for the security evaluation of software architecture. In *Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on* (2010), vol. 5, IEEE, pp. 687–695.

[3] AMAN, W., AND SNEKKENES, E. Event driven adaptive security in internet of things. In *UBICOMM 2014, The Eighth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies* (2014), pp. 7–15.

[4] AMAN, W., AND SNEKKENES, E. EDAS: An evaluation prototype for autonomic event-driven adaptive security in the internet of things. *Future Internet 7*, 3 (2015), 225–256.

[5] ASHBY, W. R. *An introduction to cybernetics*. Chapman & Hall Ltd., 1956.

[6] EVANS, D. The internet of things:how the next evolution of the internet is changing everything. Tech. rep., CISCO, April 2011. Last accessed on: 31 Aug 2015. Available from: `http://bit.ly/1Inzh2Q`.

[7] IDC ITALIA S.R.L AND TXT E-SOLUTIONS S.P.A. Definition of a research and innovation policy leveraging cloud computing and iot combination. Tech. rep., European Commission DG Communications Networks, Content & Technology, May 2015.

[8] KAZMAN, R., KLEIN, M., BARBACCI, M., LONGSTAFF, T., LIPSON, H., AND CARRIERE, J. The architecture tradeoff analysis method. In *Fourth IEEE International Conference on Engineering of Complex Computer Systems. ICECCS'98.* (1998), pp. 68–78.

[9] KSIEZOPOLSKI, B. Qop-ml: Quality of protection modelling language for cryptographic protocols. *Computers & Security 31*, 4 (2012), 569–596.

[10] LEISTER, W., HAMDI, M., ABIE, H., POSLAD, S., AND TORJUSEN, A. An evaluation framework for adaptive security for the iot in ehealth. *International Journal On Advances in Security 7*, 3 and 4 (2014), 93–109.

[11] MA, H.-D. Internet of things: Objectives and scientific challenges. *Journal of Computer science and Technology 26*, 6 (2011), 919–924.

[12] MARCOT, B. G. Metrics for evaluating performance and uncertainty of bayesian network models. *Ecological modelling 230* (2012), 50–62.

[13] MICHELSON, B. M. Event-driven architecture overview. *Patricia Seybold Group 2* (2006). Available from: `http://bit.ly/1hadIqX`.

[14] MYERSON, R. B. *Game theory*. Harvard university press, 2013.

[15] SAVOLA, R., AND ABIE, H. Development of measurable security for a distributed messaging system. *International Journal on Advances in Security 2*, 4 (2009), 358–380.