

Cheng Shiu University

Department of Information Management

Master Thesis

Study on System Architecture of Information
Security Applications and Service IoT System

A large, faint watermark of the Cheng Shiu University logo is centered behind the title text. The logo is circular, with a pink outer ring containing the text 'CHENG SHIU UNIVERSITY' in white. Inside the ring is a blue stylized graphic resembling a flame or a torch, and below it, the Chinese characters '正修科技大學' are written in white.

Advisor: Dr. Wei-Ming Ma

Graduate: Ching-I Chen

May, 2017

正修科技大學
資訊管理系研究所

碩士論文

資訊安全應用和IoT服務系統的系統架構研究

指導教授：馬維銘博士

研 究 生：陳靜怡

中華民國一〇六年五月

正 修 科 技 大 學

研 究 所 碩 士 班

論 文 口 試 委 員 會 審 定 書

本校 資訊管理所 碩士班 陳靜怡 君

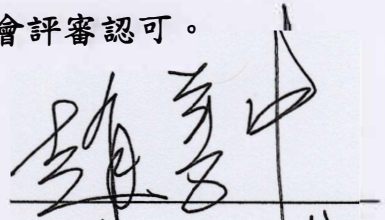
所提論文 Study on System Architecture of Information Security

Applications and Service IoT System

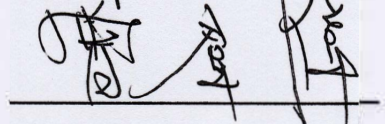
資訊安全應用和 IoT 服務系統的系統架構研究

合於碩士資格水準，業經本委員會評審認可。

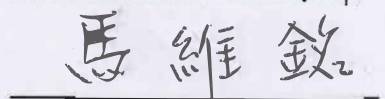
口試委員：趙善中



曾羣偉



馬維銘



指導教授：馬 維 銘

研究所所長：馬 維 銘

中華民國 106 年 5 月 6 日

摘要

本研究運用了結構行為合一的方法論與描述語言，建構一個資訊安全應用和 IoT 服務系統的系統架構模型（ISSHCASIS），它整合 IoT 服務系統與資訊安全管理模型中的結構和行為。ISSHCASIS 可以解決以結構導向為基礎的 IoT 參考模型導入資訊安全管理的三個難題，如：無線通訊驗證弱點、IoT 偵測設備安全問題、及資料庫安全缺失。本研究為強化整個系統的資訊安全管理，提出雲端服務提供者的安全篩選，物聯網安全與隱私管理者、識別授權與存取控制控制者兩項資訊安全結構，並配合物聯網安全與隱私管理行為，執行進出雲端資料庫安全與隱私資料監控，由識別授權與存取控制控制者負責符合法規的物聯網感測器的安全控管，達到提高無線通訊驗證強度、維護 IoT 偵測設備安全、及確保資料庫安全。

關鍵詞：資訊安全管理、結構行為合一、物聯網、無線通訊驗證、IoT 偵測設備

ABSTRACT

This research constructs a System Architecture of Information Security Applications and Service Internet of Thing System model (ISSHCASIS), which integrates the structure and behavior of Internet of Thing (IoT) service system and information security management model by using the methodology of Structure-behavior Coalescence Architecture description language. ISSHCASIS addresses three challenges of information security management based on the Structure-oriented IoT reference model, such as wireless communication verification vulnerabilities, IoT sensors security issues, and lack of database security. To strengthen the information security management of the ISSHCASIS, this research proposed the screened security of the cloud service providers; the IoT security and privacy manager; identification, authorization and access control controller are two important information security structures, and the Security and Privacy Management Behavior to implement accessing the cloud database security and privacy monitoring. The identification, authorization and access control controller is responsible for compliance with the law of the Internet security control of the IoT sensors, to improve the strength of wireless communication verification, maintenance IoT detection equipment security, and ensure database security.

Keywords: Information Security management, Structure-behavior Coalescence Architecture, Internet of things, Wireless Communication Verification, IoT Sensor.

ACKNOWLEDGMENTS

I am very proud of myself to pass two-year intensively study in the Cheng Shiu University. I have experienced a wonderful journey with my teachers and classmates. Now, I have my independent thinking and professional ability on my career.

First of all I would like to thank my thesis advisor Dr. Wei-ming Ma. He always takes the time while he is busy research and administration services in our school. Dr. Ma not only guided me how to do research, but also help me to write a good English papers. He just like a good friend to me and always on my side to watch my study progress and give me suggestions in professional services during my school life.

I would like to thank Dr. William S. Chao who built SBC theory and methodology of architecture. The SBC as a torch light shines on my study way and guide me through to study journey. The SBC inspires my creative thinking that I apply the information security management of IoT model. I also want to thank Dr. Chun-wei Tseng for recommending information security knowledge and skills.

I would like to thank all the teachers in Cheng Shiu University, who taught me information management expertise, enhanced my ability to face difficulties and perseverance. I have confidence and own ideas in the long run of my life. I also thank the classmates and followers for mutual encouragement and efforts.

Finally, I would like to thank to my dear parents and friends. They unmitigated support me, so that I can concentrate my study to achieve master's degree. Thanks for their encouragement to my successful leap forward.

DIRECTORY

CHINESE ABSTRACT	i
ABSTRACT	ii
ACKNOWLEDGMENTS	iii
DIRECTORY	iv
LIST OF TABLES	v
LIST OF FIGURES	vi
1. Introduction	1
1.1. Motivation	1
1.2. Study Goal	2
1.3. Study Method	2
2. Literature Review	3
2.1. Definition of IoT	3
2.2. IoT in the Ubiquitous Healthcare	4
2.3. Security and Privacy in the IoT	7
2.4. The IoT Security Frameworks	8
2.5. Architecture Description Language	13
2.6. Structure-Behavior Coalescence Architecture	15
3. Architecture-Oriented ISSHCASIS Model Design	18
3.1. Architecture Hierarchy Diagram	19
3.2. Framework Diagram	21
3.3. Component Operation Diagram	22
3.4. Component Connection Diagram	25
3.5. Structure Behavior Coalescence Diagram	26
3.6. Interaction Flow Diagrams	29
4. Results and Discussions	36
4.1. Comparison between Structure-Oriented Model and Architecture-Oriented Model	36
4.2. Useful Findings and Discussions	40
5. Conclusions and Recommendations	42
5.1. Conclusions	42
5.2. Information Security Managerial Implications	42
5.3. Recommendations	44

LIST OF TABLES

Table 2.1 Sensor for Monitoring of a Patient's Vital Signs.....	6
Table 2.2 IoT World Forum Reference Model.....	8
Table 3.1 Description of Operations in ISSHCASIS.....	23
Table 3.2 Description of Behaviors in ISSHCASIS.....	27
Table 3.3 Total of the External Environment Behaviors in ISSHCASIS.....	29
Table 4.1 Comparison between Structure-Oriented Model and Architecture-Oriented Model.....	38
Table 4.2 Difference between RAIoT and ISSHCASIS with Items.....	39



LIST OF FIGUERES

Figure 2.1 A Reference Architecture for IoT.....	11
Figure 2.2 Six Fundamental Diagrams of SBC-ADL.....	13
Figure 2.3 Completed SBC View Model.....	16
Figure 2.4 The six golden rules and their relationships.....	17
Figure 3.1 Architecture Hierarchy Diagram of ISSHCASIS.....	20
Figure 3.2 Framework Diagram of ISSHCASIS.....	21
Figure 3.3 Component Operation Diagram of ISSHCASIS.....	22
Figure 3.4 Component Connection Diagram of ISSHCASIS.....	25
Figure 3.5 Structure Behavior Coalescence Diagram of ISSHCASIS.....	27
Figure 3.6 Interaction Flow Diagrams of the Alerts Notifying behavior of ISSHCASIS.....	30
Figure 3.7 Interaction Flow Diagrams of the Registering_Patient_Account behavior of ISSHCASIS.....	31
Figure 3.8 Interaction Flow Diagrams of the Recording_Emergency_Response_Starting _Time behavior of ISSHCASIS.....	32
Figure 3.9 Interaction Flow Diagrams of the Recording_Emergency_Response_End _Time behavior of ISSHCASIS.....	33
Figure 3.10 Interaction Flow Diagrams of the IoT Security and Privacy Management Behavior of ISSHCASIS.....	34
Figure 3.11 Interaction Flow Diagrams of the Sensing_Patient_Vital_Signs behavior of ISSHCASIS.....	35

1. Introduction

In this chapter the study motivation, study goal, and study method, for Information Security of Smart Healthcare Cloud Applications and Services Internet of Things (IoT) System are described.

1.1. Motivation

The purpose of using Internet of Things (IoT) is to improve the quality of human life by automating some of the basic works that human must do. Creating and deploying smart objects allows homes, hospitals to become “intelligent” and smart environments. These smart environments and spaces and self-aware things will largely contribute to the improvement of the general population’s healthcare and wellbeing (Miller, 2015; Chao, 2016)

As a new wave of Internet-enabled technologies arrive, it is imperative to understand fully the security and privacy concerns (Thierer, 2015). Currently, there is a lack of guidance for securing IoT, IoE, and WoT as a cohesive unit (Dawson, 2016). There are several investigations done in the domains of IoT enabling technologies, applications, protocols, and security and privacy issues. The information security vulnerabilities of the IoT devices are including the low computing capabilities, low energy requirements, the unreliable nature of the wireless channel, and physical vulnerability (Eltayeb, 2017).

Healthcare applications represent the most outstanding application of IoT. The lack of confidence regarding privacy results in decreased adoption among users and is therefore one of the driving factors in the success of IoT. In fact, wireless channel increases the risk of violation due to the remote access capabilities, which potentially expose the system to eavesdropping and masking attacks.

Chellappan and Sivalingam (2016) studied the IoT revolution is expected to drive change in our society in an unprecedented way. They summarized recent research results in the area of IoT security. It emphasizes the challenges of privacy and security in IoT. The discussion considers open challenges in security and data privacy such as (1) scale and constrained network elements, (2) privacy in data collection as well as data sharing and management, and (3) identity management and authentication.

1.2. Study Goal

The purpose of this study is to explore and practice of construct an architecture-oriented of methodology for Information Security of Smart Healthcare Cloud Applications and Services IoT System (ISSHCASIS) to solve many difficulties caused by the Structure-oriented approach to the same system. This research will reach the goals of resolving the problem of high complexity of information transitions System of IoT, high cost of development, and low expandability of system.

1.3. Study Method

Enterprise architecture is complex that it comprises multiple views such as strategy, version, goal, object, concept, analysis, design, implementation, structure, behavior and input/output data views. Accordingly, an enterprise is defined as a set of interacting components forming an integrated whole of that enterprise's multiple views. Structure-Behavior Coalescence (SBC) results in the coalescence of multiple views. Therefore, it is concluded that the SBC architecture is so proper to model the multiple views of an architecture enterprise. Therefore, the SBC architecture is used to model the ISSHCASIS.

The Enterprise Architecture of Center of Excellence (EACOE) is also used to compare the practice of the enterprise architecture model for improving the speed of modeling.

2. Literature Review

The previous studies about Definition of IoT, IoT in the Ubiquitous Healthcare, Security and Privacy in the IoT, The IoT Security Frameworks, Architecture Description Language, Structure-Behavior Coalescence Architecture, and Enterprise Architecture Center of Excellent are described briefly.

2.1. Definition of IoT

Ashton (2009) is accredited for using the term “Internet of Things” for the first time during a presentation in 1999 on supply-chain management. He believes the “things” aspect of the way we interact and live within the physical world that surrounds us needs serious reconsideration, due to advances in computing, Internet, and data-generation rate by smart devices. At the time, he was an executive director at MIT’s Auto-ID Center, where he contributed to the extension of RFID applications into broader domains, which built the foundation for the current IoT vision (Bahga and Madiseti, 2014; Bhatnagar, 2015; Russell and Duren, 2016; Gilchrist, 2015; 2016).

New IoT definitions give more value to the need for ubiquitous and autonomous networks of objects where identification and service integration have an important and inevitable role. For example, Internet of Everything (IoE) is used by Cisco to refer to people, things, and places that can expose their services to other entities (Dhanjani, 2015).

International Telecommunication Union (2012) defined the IoT is a global infrastructure for information society enabling services by interconnecting physical and virtual things based on existing and evolving interoperable Information Communication Technologies.

Minerva, et al. (2015) defined An IoT is a network that connects uniquely identifiable "things" to the Internet. The "things" have sensing/actuation and potential programmability capabilities. Through the exploitation of the unique identification and sensing, information about the "thing" can be collected and the state of the "thing" can be changed from anywhere, anytime, by anything (L.R., 2013; Zhou, 2014; Holler, 2014; Elk, 2016).

2.2. IoT in the Ubiquitous Healthcare

In recent years, interests in Ubiquitous healthcare (U-Healthcare) which monitors man's health conditions using certain devices and mobile equipment and provides specialized healthcare services whenever and wherever it is needed, have been increased. The U-healthcare stands for Ubiquitous healthcare that provides healthcare services using remote medical technologies without any limitations in time and space (Kim et al., 2009). U-Healthcare services provide medical and healthcare services continuously through active participation and cooperation in all members employed in industries based on wire and wireless IT technologies through merging it to other advanced technologies such as IoT and Cloud service. To provide such healthcare services, those IoT systems must be able to learn about their environment over time, and react rapidly even if they encounter a new situation in which behavioral adaptation is required. With the IoT and healthcare there are many possible hypothetical scenarios that could be developed, from the critical, such as smart operating rooms, pharmacy services, and smart rooms for patient or psychiatric care units (Wears & Leveson 2008; Hameur and Brahimi, 2016).

In healthcare, using the IoT for patient care and using the IoT to reduce costs can co-exist as mutual goals to improve healthcare quality, as joint benefits emerge from streamlining for efficiency and improvement of service quality (Chaudhry et. al. 2006). The IoT strategies for healthcare should enhance and leverage legacy systems rather than reduce services as a by-product of automation. Connecting a device to the IoT framework requires transforming the external information a device produces and consumes into a form that can be transmitted over a network (Gubbi et. al. 2013; Penttinen, 2016).

Examples of relatively straightforward healthcare IoT applications enable scales, blood pressure monitors, temperature and other visit quantification devices to share data directly by transmitting on demand usable measurements to a requesting network agent. One or more network agents could manage the patient's record from each device. For example, as a patient enters a room, the room could be either activated by sensor, or could activate when a healthcare worker logs into the room's network and verifies the patient identity in the room. As the healthcare worker takes the measurements on various devices, the smart machines can send their readings to an open file, with buttons on the devices as options to skip logging the reading, or a way to do that in software in case there is a patient request to not update certain readings. Automating this data entry would save the time of the healthcare worker, who currently must scribe and re-enter the data into the

computer after completing the data collection with the patient.

Common to everyday living, wearable and wireless implantable medical devices, as well as home monitoring devices, are endowed with transmitting capabilities (Norris, 2015; Natarajan et al., 2016) that make information about a patient available for hospital staff analysis. For example, these devices may be wireless interconnected with sensors that measure the glucose level, the heart rate, the blood pressure, the weigh, and other medical parameters. These characteristics will turn these devices into a real part of IoT (Schwartz, 2014; 2015). In this sense, various applications are currently deployed, especially regarding the measurement and monitoring of a patient's vital signs, including glucose level sensing, electrocardiography, and blood pressure monitoring, as shown in Table 2.1:



Table 2.1 Sensor for Monitoring of a Patient's Vital Signs

Patient's Vital Signs	Sensor	Communication	Authors
Glucose	Glucose Meter	Wireless, Blue tooth	Li, 2014; Lu, 2015
Electrocardiography	Electrocardiography	Radio, Wireless	Anurag, 2014; Macala, 2016
Blood pressure	Blood pressure monitoring system	Bluetooth, Zigbee	Xin et al., 2013; NMazima, et al., 2013
Heart rate	Heart rate monitor	Wireless, BT, ZigBee	Natarajan, 2013
Body weight	Body scale (Kg)	Wireless, BT, ZigBee	McCallum, & Higgins, 2012; Tamura, et al., 1998
Body Temperature	Body Temperature sensor (C)	Xbee, Wireless, Zigbee	Mansor, et al., 2013; NMazima, et al., 2013
Respiration rate	Breath sensor	Wireless, Zigbee	Bachfischer, 2014

2.3. Security and Privacy in the IoT

The Internet of Things (IoT) promises to revolute communications on the Internet. The IoT enables numerous business opportunities in fields as diverse as electronical health, smart cities, smart homes, among many others. It incorporates multiple long-range, short-range, and personal area wireless networks and technologies into the designs of IoT applications. This will result in the IoT being pervasive in many areas which raise many challenges the IoT with regard to security, privacy, and management (Moolayil, 2016; Hu, 2016).

The challenges that must be overcome to resolve IoT security and privacy issues are immense. This is primarily because of the many constraints attached to the provision of security and privacy in IoT systems. The deployment of the IoT raises many security issues arising because of the following aspects: (1) the very nature of smart objects, for example, the adoption of lightweight cryptographic algorithms, in terms of processing and memory requirements. (2) the use of standard protocols, for example, the need to minimize the amount of data exchanged between nodes. (3) the bidirectional flow of information, for example, the need to build an end-to-end security architecture (Kellmerit and Daniel, 2013; Aslam, 2015; Chellappan and Sivalingam, 2016).

Confidentiality: transmitted data can be read only by the communication endpoints; availability: the communication endpoints can always be reached and cannot be made inaccessible; integrity: received data are not tampered with during transmission, and assured of the accuracy and completeness over its entire lifecycle; authenticity: data sender can always be verified and data receivers cannot be spoofed and authorization: data can be accessed only by those allowed to do so and should be made unavailable to others. The requirements for securing the IoT are complex, involving a blend of approaches from mobile and cloud architectures, combined with industrial control, automation, and physical security (Ren, et al., 2014; Romdhani, 2017).

However, the smart IoT devices expose much more sensitive information, and provide much less scope for this type of commercial model as it is largely back-end data. Hence users are likely to be both vulnerable and sensitive to privacy concerns. These challenges make it very complex to operationalize IoT in a secure way, while fully preserving privacy. There are several promising approaches that are being investigated to solve for each aspect of the privacy issues, and there is still some distance to go before we can see production ready commercial implementations that are standardized and

widely adopted (Wears and Leveson, 2008; Sabina, 2017).

2.4. The IoT Security Frameworks

Today, there is no standardized conceptual model that characterizes and standardizes the various functions of an IoT system. Cisco Systems Inc. has proposed an IoT reference model that comprises seven levels. The IoT reference model allows the processing occurring at each level to range from trivial to complex, depending on the situation. The model also describes how tasks at each level should be handled to maintain simplicity, allow high scalability, and ensure supportability. Finally, the model defines the functions required for an IoT system to be complete. The seven levels and their brief characteristics are shown in Table 2.2. The fundamental idea is to present a level of abstraction and appropriate functional interfaces to provide a complete system of IoT. It is the coherence of an end-to-end IoT architecture that allows one to process volume of context specific data points, make meaningful information, manage intrinsic feature of large scale, and ultimately design insightful responses (Green, 2014; Buyya and Dastjerdi, 2016).

Table 2.2 IoT World Forum Reference Model

Levels	Characteristics
Physical devices and	End point devices, exponential growth, diverse
Connectivity	Reliable, timely transmission, switching, and
Edge computing	Transform data into information, actionable data
Data accumulation	Data storage, persistent and transient data
Data abstraction	Semantics of data, data integrity to application, data
Application	Meaningful interpretations and actions of data
Collaboration and processes	People, process, empowerment, and collaboration

(From: Green, 2014)

The simplest type of passive threats in the IoT is that of eavesdropping or monitoring of transmissions with a goal to obtain information that is being transmitted. It is also referred to as capture attacks. Capture attacks are designed to gain control of physical or

logical systems or to gain access to information or data items from these systems. The ubiquity and physical distribution of the IoT objects and systems provide attackers with great opportunity to gain control of these systems. The distribution of smart objects, sensors, and systems results in self-advertisements, beacons, and mesh communications, providing attackers greater opportunity to intercept or intercede in information transmission within the environment. Moreover, the frequency of the data transmissions, data model, and formats help attackers in cryptanalysis.

Some of the well-known active threats are as follows: Masquerading: an entity pretends to be a different entity. This includes masquerading other objects, sensors, and users. Man-in-the-middle: when the attacker secretly relays and possibly alters the communication between two entities that believe that they are directly communicating with each other. Replay attacks: when an intruder sends some old (authentic) messages to the receiver. In the case of a broadcast link or beacon, access to previous transmitted data is easy. Denial-of-Service (DoS) attacks: when an entity fails to perform its proper functions, or acts in a way that prevents other entities from performing their proper functions.

Generically, an IoT deployment can consist of smart sensors, control systems and actuators, web and other cloud services, analytics, reporting, and a host of other components and services that satisfy a variety of business use cases. IoT services can be public or may be open to external agencies; as such, security can be an issue. Because of an increase in theft, privacy issues, misuse of information, lack of policy guidance, and ethical issues, it has become increasingly imperative to govern the use of information technology. This has increased the demand for security management, as shown in Figure 2.1.

Hardware and software manufacturers of IoT applications and peripherals need to be able to determine what impact their decisions will have on overall consumer satisfaction. The IoT provider and manufacturer should address privacy and security issues through adopting best practices for the development of risk management processes. Weber (2010) summarized the privacy and security requirements for protecting IoT systems as follows: (1) Resilience to Attacks: The system has to avoid single points of failure and should adjust itself to node failures. (2) Data Authentication: Access to objects' information must be authenticated as a principle. (3) Access Control: Information providers must be able to implement access control on the data provided. (4) Client Privacy: Measures need to be taken to ensure that only the information provider can infer from observing the use of the

lookup system related to a specific customer; at least, inference should be very hard to conduct (Waher, 2015; Eltayeb, 2017).

International Telecommunication Union (2012) shown the IoT reference model. It is composed of four layers as well as management capabilities and security capabilities which are associated with the four layers. The four layers are as follows: application layer, service support and application support layer, network layer, device layer. The application layer contains IoT applications. The service support and application support layer consists of two capability groups such as Generic support capabilities and Specific support capabilities. Network layer consists of Networking capabilities and Transport capabilities. Device layer capabilities can be logically categorized into Device capabilities and Gateway capabilities. The IoT management capabilities can be categorized into generic management capabilities and specific management capabilities.

There are two kinds of security capabilities: generic security capabilities and specific security capabilities. Generic security capabilities are independent of applications. They include: at the application layer: authorization, authentication, application data confidentiality and integrity protection, privacy protection, security audit and anti-virus; at the network layer: authorization, authentication, use data and signaling data confidentiality, and signaling integrity protection; at the device layer: authentication, authorization, device integrity validation, access control, data confidentiality and integrity protection. Specific security capabilities are closely coupled with application-specific requirements, e.g., mobile payment, security requirements (International Telecommunication Union, 2012).

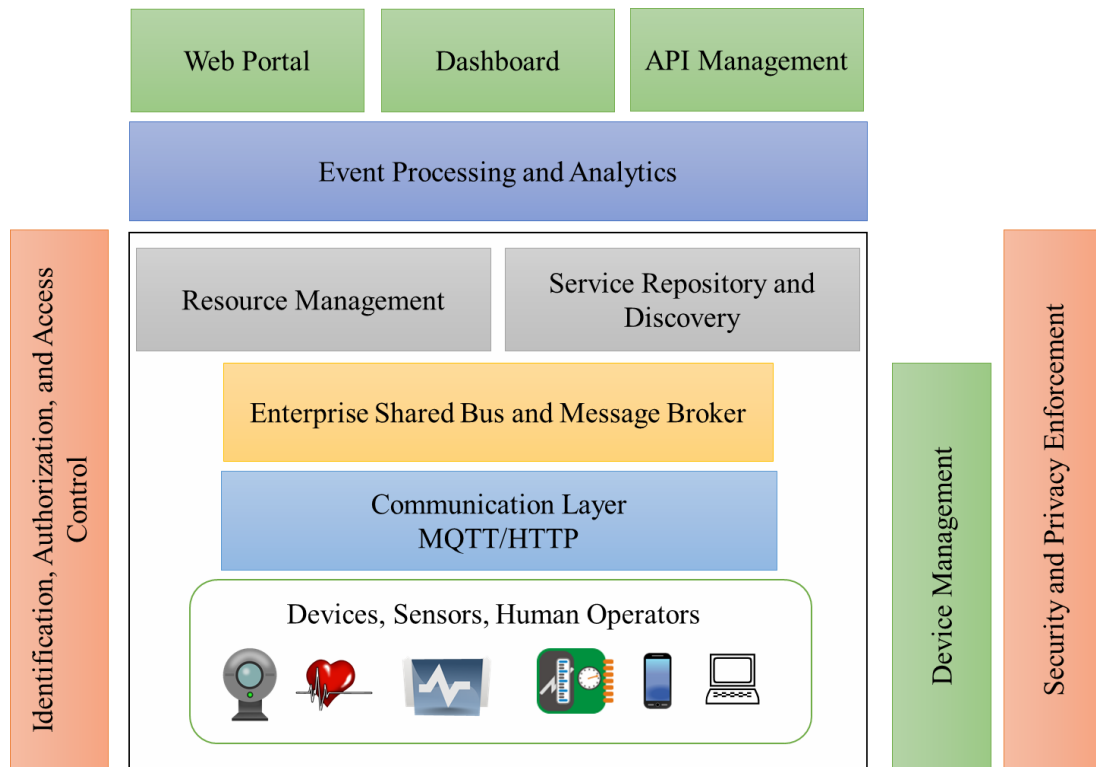


Figure 2.1 A Reference Architecture for IoT (Redraw from WSO2)

The reference architecture for IoT consists of a set of components. Layers can be realized by means of specific technologies. The layers are (1) Client/external communications - Web/Portal, Dashboard, APIs (2) Event processing and analytics (including data storage) (3) Aggregation/bus layer – ESB and message broker (4) Relevant transports - MQTT/HTTP/XMPP/CoAP/AMQP, etc. (5) Devices. There are also some cross-cutting/vertical layers such as access/identity management and device manager.

Numerous studies have examined the privacy and security implications of the IoT (Eltayeb, 2017; Zhang et al., 2015; Ren et al., 2014; Pohls et al., 2014), and the media commonly report on issues that have arisen because of breaches of security in this domain. The relevant IoT end-to-end security requirements including: (1) Identity verification entities at both ends of the communications to have known and verifiable identities. (2) Protocols are needed to negotiate session keys, and to provide the required security functions across several heterogeneous networks. (3) Algorithms to implement security functions e.g. encryption as in Secure Hash Algorithm. Simple, optimized and lightweight algorithms are needed in the IoT. (4) Secure implementations of the security protocols and algorithms should be free of bugs and security holes that could compromise security.

(5) Users and operators should understand security operations, which can be complex in the IoT given the diversity of things and the heterogeneous nature of IoT communications (Stackowiak and Licht, 2015; Elkhodr, et al., 2016; Spaanenburg, 2016)

The requirements for privacy in the IoT includes: (1) Collection Announcement is a need to find efficient ways to communicate and inform the user about collection procedures of his or her personal data in the IoT. (2) Choice and Consent is to give users a selection mechanism so they can indicate which service they which to use. (3) User should be able to check the logs of their devices and the data they collect. (4) Control over Contextual Data Disclosure is given that interactions in the IoT are multi-dimensional, contextual data can reveal sensitive information about the users such as location information (Elkhodr, et al., 2016).



2.5. Architecture Description Language

Chao (2012; 2016) studied an architecture description is a formal description and representation of a system. A description of the systems architecture must grasp the essence of the system and its details at the same time. In other words, an architecture description not only provides an overall picture that summarizes the whole system, but also contains enough detail that the system can be constructed and validated.

The language for architecture description is called the architecture description language (ADL) (Chao, 2012; 2016). An ADL is a special kind of language used in describing the architecture of a system. Since the architectural approach uses a coalescence model for all multiple views of a system, the foremost duty of ADL is to make the strategy/version n, strategy/version n+1, concept, analysis, designs, implementation, structure, behavior, and input/output data views all integrated and coalesced within this architecture description.

SBC-ADL uses six fundamental diagrams to describe the integration of systems structure and systems behavior of a system. These diagrams, as shown in Figure 2-20, are: a) architecture hierarchy diagram (AHD), b) framework diagram (FD), c) component operation diagram (COD), d) component connection diagram (CCD), e) structure-behavior coalescence diagram (SBCD), and f) interaction flow diagram (IFD), as shown in Figure 2.2:

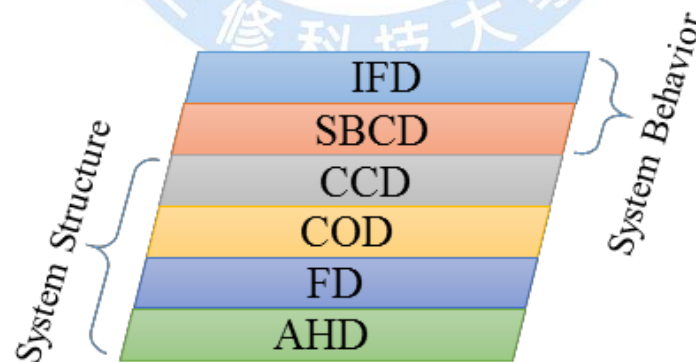


Figure 2.2 Six Fundamental Diagrams of SBC-ADL

Chao (2012) has developed Structure-Behavior Coalescence (SBC) EA since 1999 and used in large enterprises in Taiwan very well. Because structure view and behavior view are the two most prominent ones among multiple views of the EA, integrating

structure and behavior views is a way to integrate multiple views of an EA SBC-ADL uses AHD, FD, COD, CCD, SBCD, and IFD to depict the systems structure and systems behavior of a system.

Examining the SBC-ADL approach, we find out that it depicts the systems structure first and then depicts the systems behavior later, not the other way around. The reason SBC-ADL does so lies in that the systems behavior must be attached to or built on the systems structure. With the systems structure and attached systems behavior, then, we can smoothly get the systems architecture (Simon and Schmidt, 2015).

In the SBC-ADL, systems behavior must be attached to or built on systems structure. In other words, the systems behavior shall not exist alone, it must be loaded on the systems structure just like a cargo is loaded on a ship as shown in Figure 2-24. There will be no systems behavior if there is no systems structure. A stand-alone systems behavior is not meaningful. AHD, FD, COD, and CCD belong to systems structure. SBCD and IFD belong to systems behavior. Concluding the above discussion, we perceive that SBC-ADL will describe AHD, FD, COD, and CCD first then describe SBCD and IFD later when it constructs the systems architecture of a system.

2.6. Structure-Behavior Coalescence Architecture

Chao (2012, 2016) studied that a software system comprises multiple views such as analysis view, design view, implementation view, deployment view, structure view, behavior view, input data view, output data view. The non-architectural approach respectively picks a model for each view. The architectural approach, instead of using many unrelated model, will integrate all these multiple views into a single coalescence model of software architecture. Since structure view and behavior view are the two most prominent ones among multiple views, integrating structure and behavior views is a way to integrate multiple views of a software system. In other words, structure behavior coalescence (SBC) sets a path to achieve the multiple views coalescence and software architecture.

Among multiple views, the structure view and the behavior view are perceived as the two prominent ones. The structure view focuses on the software structure while the behavior view concentrates on the software behavior. Analysis, design, implementation, deployment, input data, and output data are considered. Chao (2012, 2016) defined software architecture is an integrated, holistic, coordinated, coherent, and coalescence model for a software's multiple views, i.e., structure view, behavior view, and other views, in which:

- A software comprises many structure elements;
- Multiple views such as structure view, behavior view, and other views are derivable from interactions among these structure elements; and
- Multiple views such as structure view, behavior view, and other views are all contained in this model.

Based on its definition, software architecture can be regarded as a knowledge repository of a software system. Each stakeholder, through structure view, behavior view, and other views, contributes his own knowledge or know-how to this repository when the software architecture is built up. Structure-behavior coalescence (SBC) architecture is an architecture-oriented model for personal information protection which integrates the structure and behavior views of personal information protect system. SBC view model is a three-dimensional matrix representation of a system's multiple views. Dimension 1

stands for the evolution & motivation view which contains the strategy/version 1, strategy/version 2, strategy/version 3, strategy/version 4, strategy/version views; dimension 2 stands for the multi-level (hierarchical) views which contain the concept, analysis, design, and implementation views; dimension 3 stands for the systemic view which contains the structure and behavior views (SBC Architecture International, 2015). According to the definition of SBC architecture description language (SBC-ADL), the structure view consists of Architecture Hierarchy Diagram (AHD), Framework Diagram (FD), Component Operation Diagram (COD), and Component Connection Diagram (CCD); the behavior view consists of Structure-Behavior Coalescence Diagram (SBCD) and Interaction Flow Diagram (IFD). Also, FD consists of business layer, application layer, data layer, and technology layer. Adding these ideas, we then get the complete SBC view model (SBC Architecture International, 2015), as shown in Figure 2.3:

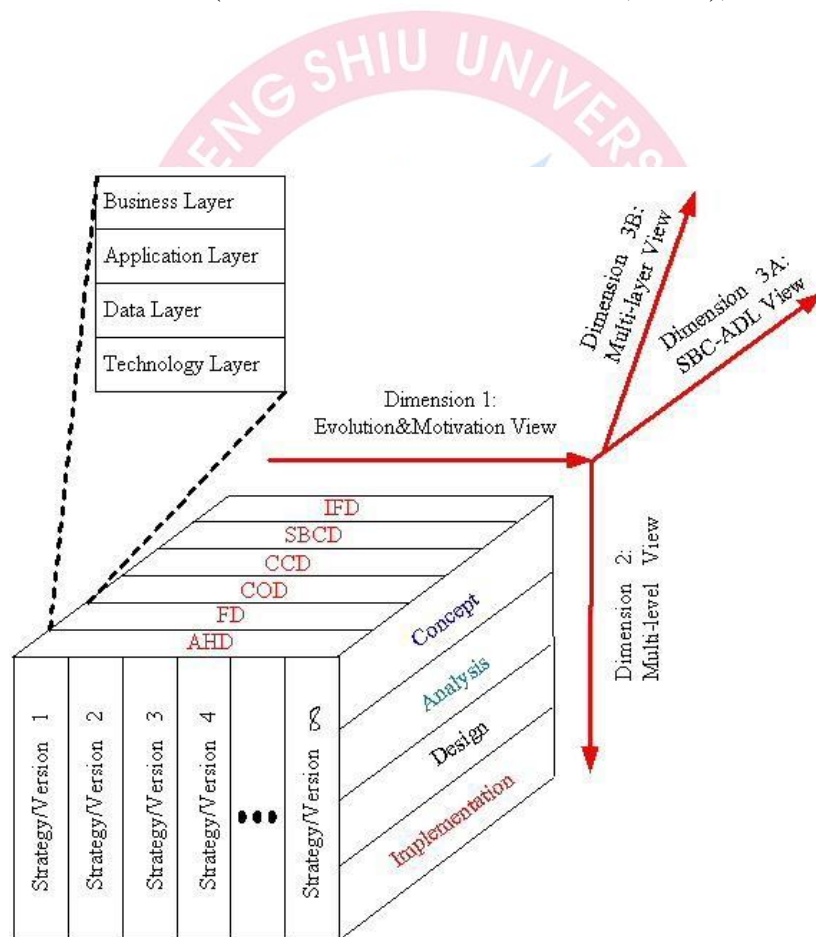


Figure 2.3 Completed SBC View Model (From: Chao, 2012)

SBC architecture consists of six fundamental diagrams, also known as six golden rules, for personal information protection. These diagrams are: Architecture Hierarchy

Diagram (AHD), Framework Diagram (FD), Component Operation Diagram (COD), Component Connection Diagram (CCD), Structure-Behavior Coalescence Diagram (SBCD), and Interaction Flow Diagram (IFD). The six golden rules and their relationship are shown in Figure 2.4:

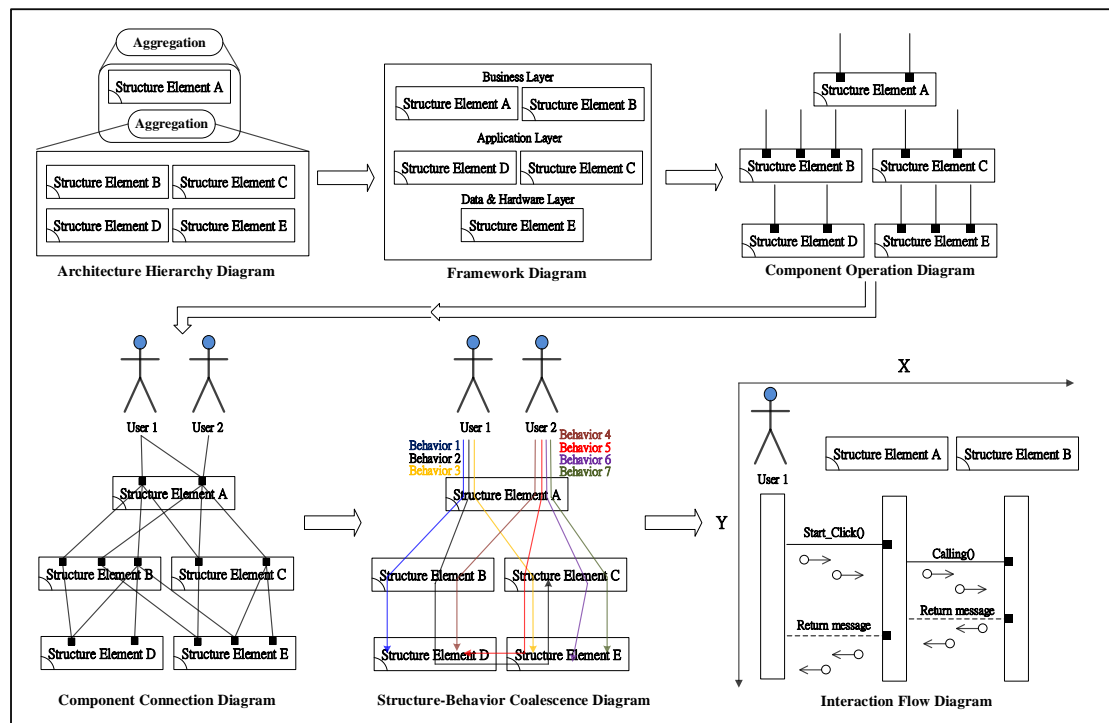


Figure 2.4 The six golden rules and their relationships (From Wei-ming Ma, 2010)

Chao (2012) intensively applied the SBC methodology to many case studies: auto mobile - hardware architecture, multi-tier personal data system - software architecture, Stanford University - enterprise architecture, Robot – knowledge architecture and strategic thinking of an airline – thinking architecture. He has fruitful research results. Ma (2010) adopted the SBC methodology to construct an architecture-oriented information security risk assessment model. He found out the information security consultant, project manager were the key roles for the success of the risk assessment from structure-behavior coalescence diagram. Therefore, the resources could be evenly distributed, implementing security performance can be increased, and reduced security risk for the enterprise (Ahlemann, et al., 2012; Smith, 2017)

Ma and Tsai (2012) used the SBC methodology to construct implementation of

the personal information protection act (PIPA) on Chen Shiu University (CSU) Campus, which was a prototype integrated structures and behaviors of the personal information protection model. This research will achieve a beneficial model and knowledge for the personal information protection.



3. Architecture-Oriented ISSHCASIS Model Design

In this research, we extended the Systems Architecture of Smart Healthcare Cloud Applications and Service IoT System (SHCASIS) (Chao, 2016) and emphasized on information security of IoT. The SBC methodology was used to design an architecture-oriented the systems architecture of Information Security of Smart Healthcare Cloud Applications and Services Internet of Things (IoT) System (ISSHCASIS). The ISSHCASIS included: architecture hierarchy diagram, structure element diagram, structure element service diagram, structure element connection diagram, structure behavior coalescence diagram, and interaction flow diagram.

3.1. Architecture Hierarchy Diagram

Any management model can be illustrated by an architecture hierarchy diagram (AHD) for the structure of a system's decomposition and combination to understand complex systems easily. AHD is obtained after finishing the architecture construction. Figure 3.1 shows an AHD of ISSHCASIS. In the figure, ISSHCASIS composed of Application_Layer, Data_Layer, and Technology_Layer. Application_Layer is composed of Presentation_Layer and Logic_Layer. Presentation_Layer is composed of Patient_Account_Registering_UI, Alerts_Notifying_UI, Emergency_Response_Starting_Time_UI, and Emergency_Response_End_Time_UI. Logic_Layer is composed of Patient_Vital_Signs_Deamon. Data_layer is composed of ISSHCASIS_Database. Technology_Layer is composed of Patient_Vital_Signs_Sensor_P, IAA(identification, Authorization, and Access Control)_Controller, and IoT_Security_&_Privacy_Manager.

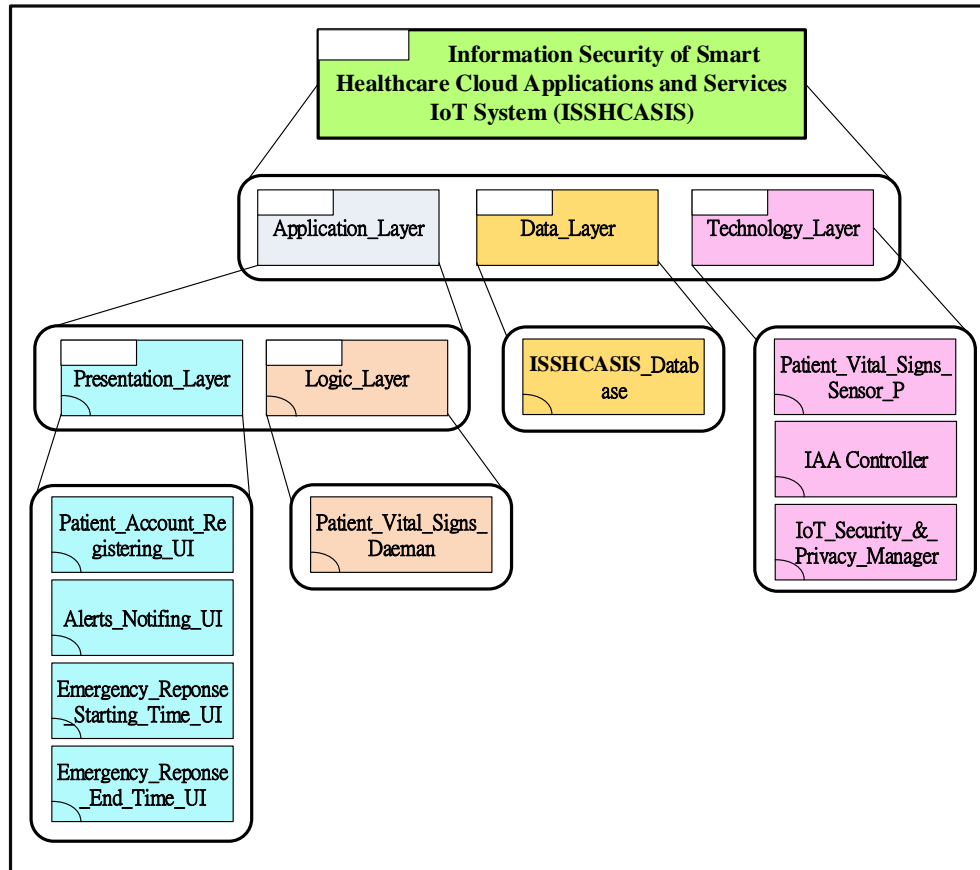


Figure 3.1 Architecture Hierarchy Diagram of ISSHCASIS

This is required to construct the ISSHCASIS model structure element diagram from a structural point of view. The structure elements of the ISSHCASIS model are the basic elements, and they compose of the ISSHCASIS structure. The necessary structure elements were analyzed from the ISSHCASIS model. That is identified all builders and destroyers of the ISSHCASIS model.

3.2. Framework Diagram

After collection of non-aggregated systems or structure elements of architecture hierarchy diagram, we obtain the Framework Diagram (FD). From Figure 3.1, Presentation_Layer and Logic_Layer are sub-layers of Application_Layer. Presentation_Layer contains the Patient_Account_Registering_UI, Alerts_Notifying_UI, Emergency_Response_Starting_Time_UI, and Emergency_Response_End_Time_UI components. Logic_Layer contains the Patient_Vital_Signs_Daemon component; Data_Layer contains the ISSHCASIS_Database component. o_Layer contains the Patient_Vital_Signs_Sensor_P, IAA_Controller, and IoT_Security_&_Privacy_Manager components, as shown in Figure 3.2:

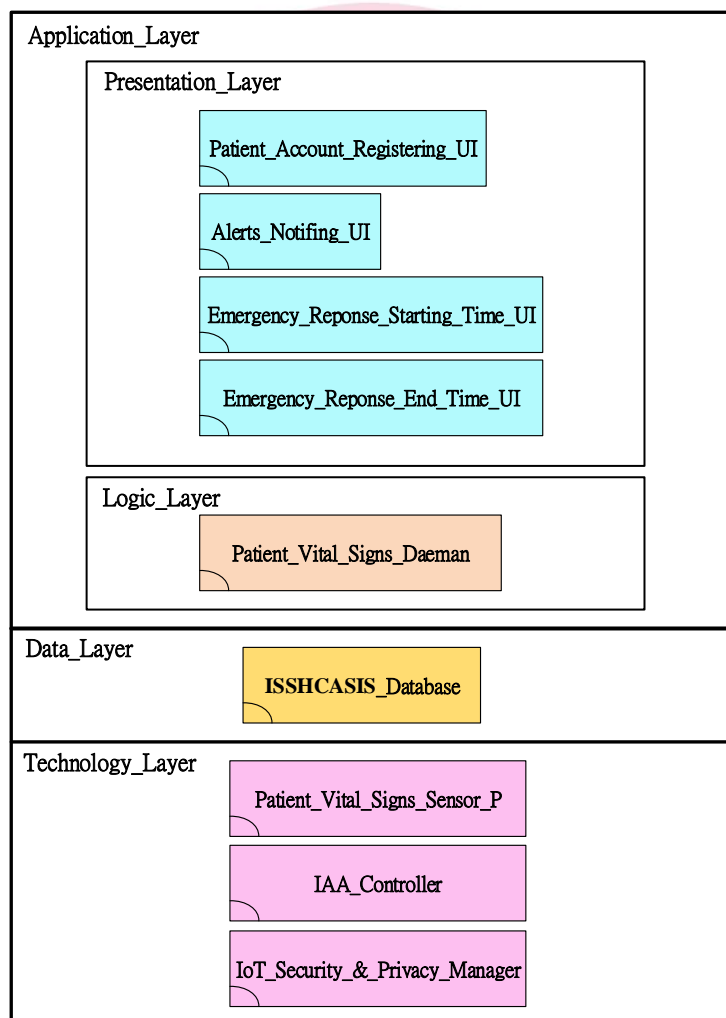


Figure 3.2 Framework Diagram of ISSHCASIS

3.3. Component Operation Diagram

For a system, we use component operation diagram (COD) to demonstrate all components operations. COD is the third fundamental diagram to achieve structure-behavior coalescence. The structure components provide many operations through the interface or work content of the structure components with input or output parameters is called a COD (Sweeney, 2010; Lawler and Howell-Barber, 2007). Input parameter of the service is denoted by an arrow symbol directed to structure element. Output parameters of the operation are denoted by an arrow symbol leave the component. Based on the collection of literature, standard operation procedure (SOP), and sorted out the structure components step by step, operations of nine structure elements were obtained for the ISSHCASIS, as shown in Figure 3.3:

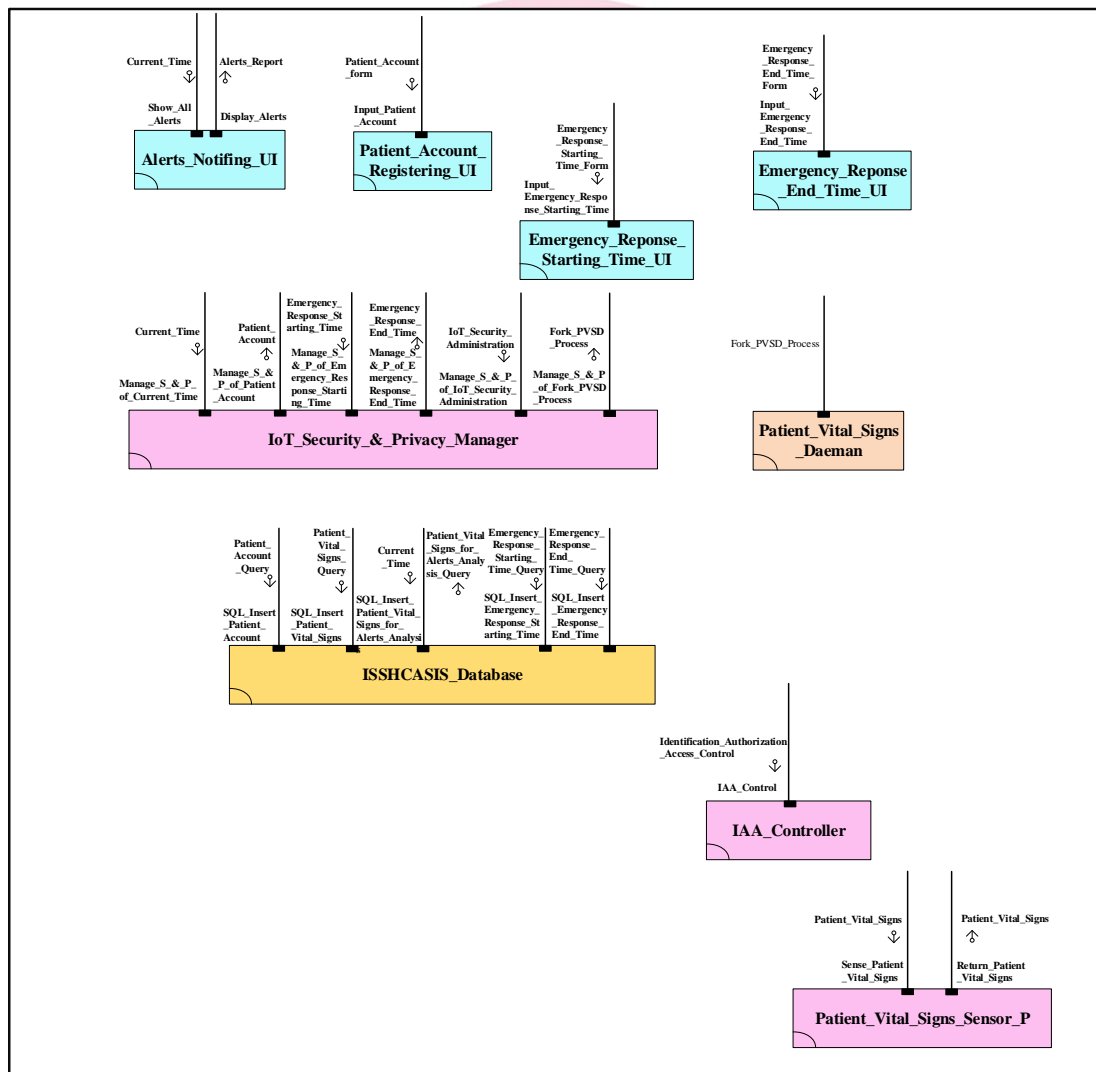


Figure 3.3 Component Operation Diagram of ISSHCASIS

Description of Operations for each component in ISSHCASIS are described in Table 3.1:

Table 3.1 Description of Operations in ISSHCASIS

Component	Operations
Patient_Account_Registering_UI	Input_Patient_Account Patient_Account_Form
Alerts_Notifying_UI	Show_All_Alerts Current_Time
	Display_Alerts Alerts_Report
Emergency_Response_Starting_Time_UI	Input_Emergency_Response_Starting_Time Emergency_Response_Starting_Time_Form
Emergency_Response_End_Time_UI	Input_Emergency_Response_End_Time Emergency_Response_End_Time_Form
Patient_Vital_Signs_Daemon	Fork_PVSD_Process
ISSHCASIS_Database	SQL_Insert_Patient_Account Patient_Account_Query
	SQL_Insert_Patient_Vital_Signs Patient_Vital_Signs_Query
	SQL_Insert_Patient_Vital_Signs_for_Alerts_Analysis Current_Time Patient_Vital_Signs_for_Alerts_Analysis_Query
	SQL_Insert_Emergency_Response_Starting_Time Emergency_Response_Starting_Time_Query
	SQL_Insert_Emergency_Response_End_Time Emergency_Response_End_Time_Query

Patient_Vital_Signs_Sensor_P	Sense_Patient_Vital_Signs Patient_Vital_Signs
	Return_Patient_Vital_Signs Patient_Vital_Signs
IAA_Controller	IAA_Control Identification, Authorization, and Access Control
IoT_Security_&_Privacy_Manager	Monitoring_IoT_Security_&_Privacy Current_Security_&_Privacy_Status
	Manage_S_&_P_of_Current_Time Current_Time
	Manage_S_&_P_of_Fork_PVSD_Process Fork_PVSD_Process
	Manage_S_&_P_of_IoT_Security_Administration IoT_Security_Administration
	Manage_S_&_P_of_Emergency_Response_Starting_Time Emergency_Response_Starting_Time
	Manage_S_&_P_of_Emergency_Response_End_Time Emergency_Response_End_Time
	Manage_S_&_P_of_Patient_Account Patient_Account

3.4. Component Connection Diagram

A structure component connection diagram (CCD) connects operations between the various structure components in accordance with its priorities. CCD is obtained after the analysis phase is finished. We use the CCD to describe how the components and actors (in the external environment) are connected within ISSHCASIS. CCD is the fourth fundamental diagram to achieve structure-behavior coalescence. Rectangular frame is the system boundary, and the Five_Minute_Interval, Healthcare_Provider, IoT_Security_Administrator, Server_Root, Patient_Vital_Signs are the external environment, as shown in Figure 3.4:

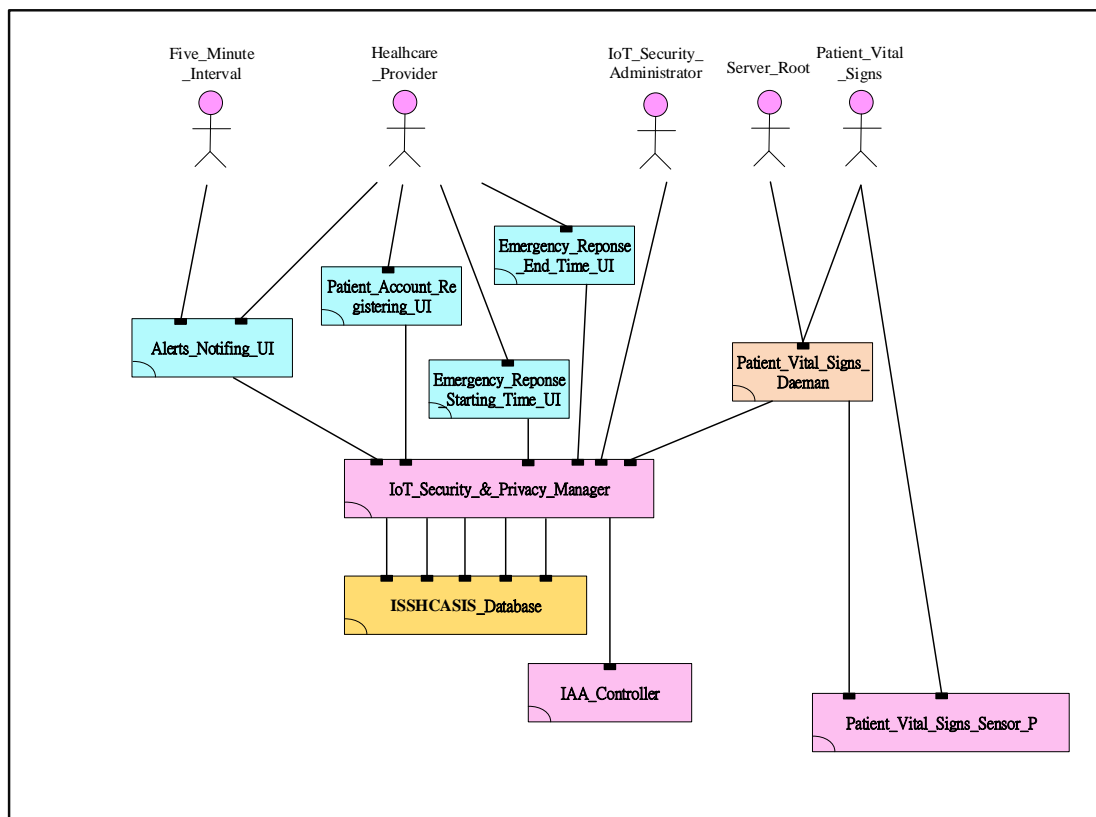


Figure 3.4 Component Connection Diagram of ISSHCASIS

3.5. Structure Behavior Coalescence Diagram

The major purpose of adopting the architectural approach, instead of separating the structure model from the behavior model, is to achieve one single coalesced model. In Figure 3.5, systems architect can see that systems structure and systems behavior coexist in the SBCD. That is, in the SBCD of ISSHCASIS, systems architect not only see its systems structure but also see its systems behavior simultaneously.

From the structure element diagram and structure element service diagram, we further derive out six behaviors of the ISSHCASIS model: (1) Alerts Notifying Behavior (2) Registering Patient Account Behavior (3) Recording Emergency Response Starting Time Behavior (4) Recording Emergency Response End Time Behavior (5) Sensing Patient Vital Signs Behavior, and (6) IoT Security and Privacy Management Behavior, as shown in Figure 3.5 and Table 3.2.

SBCD is the structure-behavior coalescence diagram we obtain after the architecture construction is finished. Figure 3.5 shows a SBCD of the ISSHCASIS in which interactions among the Five_Minute_Interval, Healthcare_Provider, IoT_Security_Administrator, Server_Root, Patient_Vital_Signs actors and the Alerts_Notifying_UI, Patient_Account_Registeritig_UI, Emergency_Response_Starting_Time_UI, Emergency_Response_End_Time_UI, Patient_Vital_Signs_Daemon, ISSHCASIS_Database, Patient_Vital_Signs_Sensor_P, IoT_Security_&_Privacy_Manager, IAA_Controller components shall draw forth Registering_Patient_Account, Sensing_Patient_Vital_Signs, Alerts_Notifying, Recording_Emergency_Response_Starting_Time, Recording_Emergency_Response_End_Time, IoT_Security_&_Privacy_Management behaviors. In other words, these six behaviors together provide the overall behavior of the ISSHCASIS.

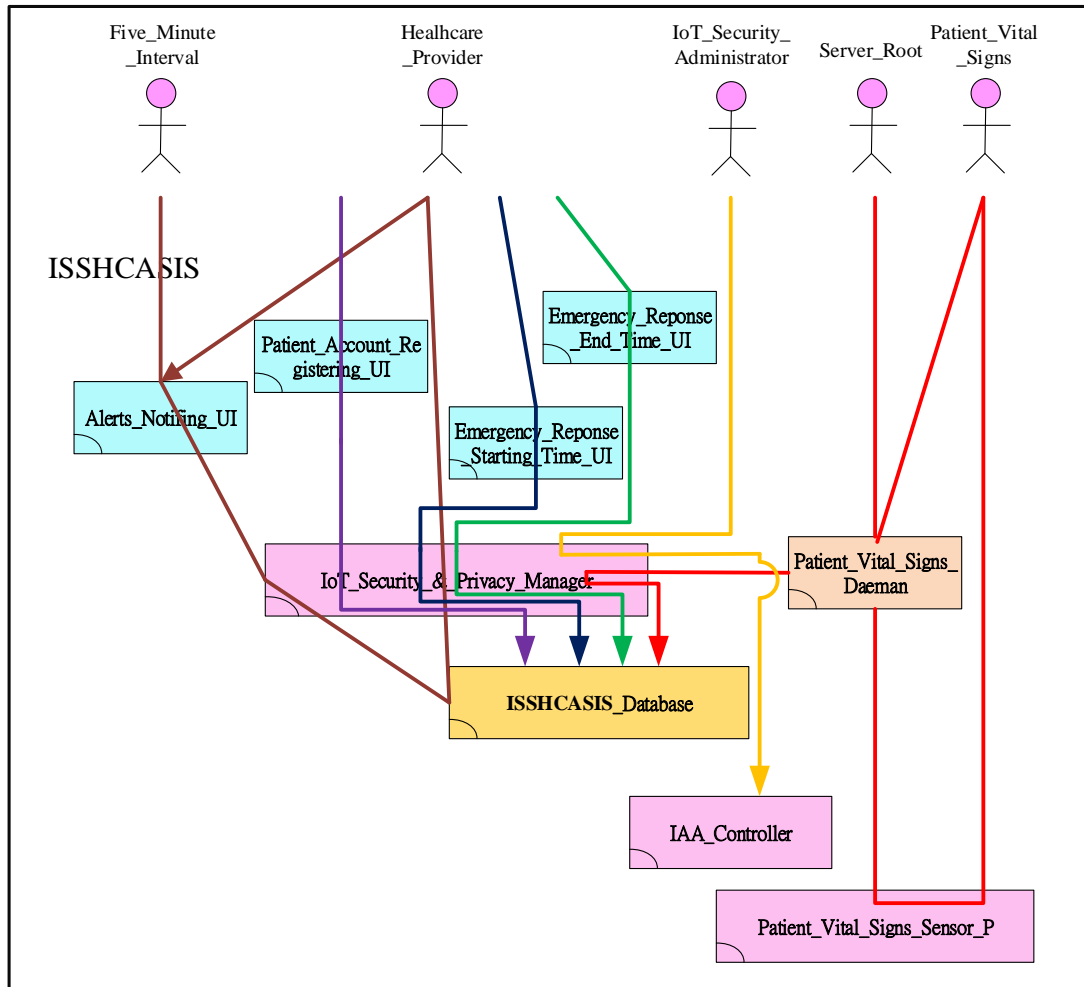


Figure 3.5 Structure Behavior Coalescence Diagram of ISSHCASIS

Table 3.2 Description of Behaviors in ISSHCASIS

External environment	Structure Elements	Interactive Behaviors	Behavior Number
Five_Minute_Interval Healthcare_Provider	Alerts_Notifying_UI	Alerts	1
	IoT_Security_& Privacy Manager	Notifying	
	ISSHCASIS_Database	Behavior	
Healthcare_Provider	Patient_Account_Registering_UI	Registering	2
	IoT_Security_& Privacy Manager	Patient	
	ISSHCASIS_Database	Account Behavior	
Healthcare_Provider	Emergency_Reponse_Starting_Time_UI	Recording	3

	IoT_Security_&_Privacy_Manager	Emergency Response Starting Time Behavior	
	ISSHCASIS_Database		
Healthcare_Provider	Emergency_Reponse_End_Time_UI	Recording Emergency Response End Time Behavior	4
	IoT_Security_&_Privacy_Manager		
	ISSHCASIS_Database		
IoT_Security_Administrator	IoT_Security_&_Privacy_Manager	Sensing Patient Vital Signs Behavior	5
	IAA_Controller		
Server_Root, Patient_Vital_Signs	Patient_Vital_Signs_Daemon	IoT Security and Privacy Management Behavior	6
	Patient_Vital_Signs_Sensor_P		
	IoT_Security_&_Privacy_Manager		
	ISSHCASIS_Database		

In Table 3.2 can find out Department Heads have 1 behavior, and Vice President has 8 in ISSHCASIS. We can see the Vice President initiated 8 interactive behaviors in the SBC Diagram of ISSHCASIS, that means the Vice President is the commanding officer of information security risk management and all of the activities of the risk management can be fulfilled to the requirements and continuously improvement of information security risk management. Total of the external environment behaviors in ISSHCASIS, as shown in Table 3.3:

Table 3.3 Total of the External Environment Behaviors in ISSHCASIS

External environment	Total of Interactive Behavior
Five_Minute_Interval	1
Healthcare_Provider	4
IoT_Security_Administrator	1
Server_Root	1
Patient_Vital_Signs	1

3.6. Interaction Flow Diagrams

We use interaction flow diagram (IFD) to demonstrate individual behavior. IFDs are the interaction flow diagrams we obtain after the architecture construction is finished. IFD is the sixth fundamental diagram uses in achieving structure-behavior coalescence. Each behavior presented on the SBCD of the ISSHCASIS can be drawn as an IFD. The construction of IFD of the ISSHCASIS describes the outside environment and structure elements, and their interactions according to the time. Each individual behavior is represented by an execution path. We use an IFD to define each one of these execution paths.

There are 6 interaction flow diagrams in total for the ISSHCASIS: (1) Interaction Flow Diagrams for Alerts Notifying of ISSHCASIS (2) Interaction Flow Diagrams for Registering Patient Account Behavior of ISSHCASIS (3) Interaction Flow Diagrams for Recording Emergency Response Starting Time Behavior of ISSHCASIS (4) Interaction Flow Diagrams for Recording Emergency Response End Time Behavior of ISSHCASIS (5) Interaction Flow Diagrams for Sensing Patient Vital Signs Behavior of ISSHCASIS, and (6) Interaction Flow Diagrams for IoT Security and Privacy Management Behavior of ISSHCASIS.

Figure 3.6 represents IFD for Alerts Notifying of ISSHCASIS. X-axis represents structure elements and the external environment in which information flow direction is from left to right. Y-axis represents the implementation of an interactive timeline from the top to the bottom in the time sequence. Figure 3.6 shows an IFD of the Alerts_Notifying behavior. First, actor Five_Minute_Interval interacts with the Alerts_Notifying_UI component through the Show_All_Alerts operation call interaction, carrying the

Current_Time input parameter. Next, component Alerts_Notifying_UI interacts with IoT_Security_&_Privacy_Manager component through the Manage_S_&_P_Vital_Signs_for_Alerts_Analysis carrying the Current_Time input parameter. Next, IoT_Security_&_Privacy_Manager interacts with ISSHCASIS_Database component through the SQL_Select_Patient_Vital_Signs_for_Alerts_Analysis operation call interaction, carrying the Current_Time input parameter and Patient_Vital_Signs_for_Alerts_Analysis_Query output parameter. Continuingly, IoT_Security_&_Privacy_Manager interacts with Alerts_Notifying_UI component through the Monitoring_IoT_Security_&_Privacy operation call interaction, carrying Current_Security_&_Privacy_Status. Finally, actor Healthcare_Provider interacts with the Alerts_Notifying_UI component through the Display_Alerts operation call interaction, carrying the Alerts_Report output parameter, as shown in Figure 3.6:

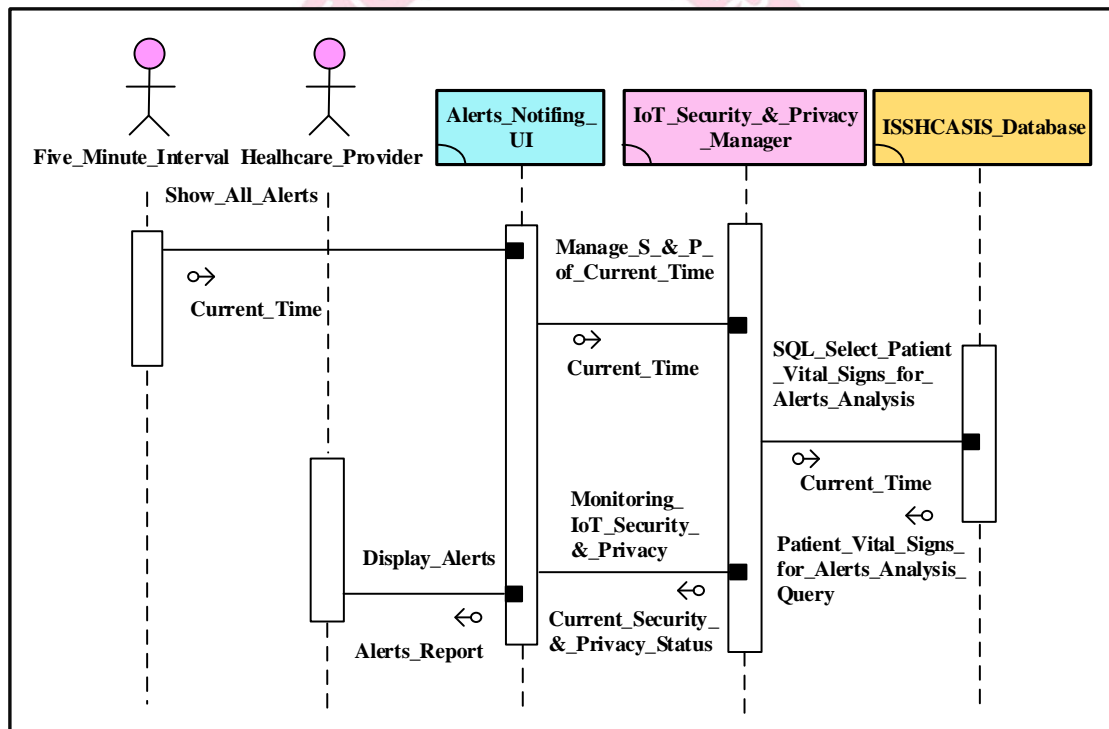


Figure 3.6 Interaction Flow Diagrams of the Alerts Notifying Behavior of ISSHCASIS

Figure 3.7 shows an IFD of the Registering_Patient_Account behavior. First, actor Healthcare_Provider interacts with the Patient_Account_Registering_UI component through the Input_Patient_Account operation call interaction, carrying the Patient_Account_Form input parameter. Next, component Patient_Account_Registering_UI interacts with the IoT_Security_&_Privacy_Manager component through the Manage_S_&_P_of_Patient_Account operation call interaction, carrying the Patient_Account input parameter. Finally, component IoT_Security_&_Privacy_Manager interacts with the ISSHCASIS_Database component through the SQL_Insert_Patient_Vital_Signs operation call interaction, carrying the Patient_Vital_Signs_Query input parameter.

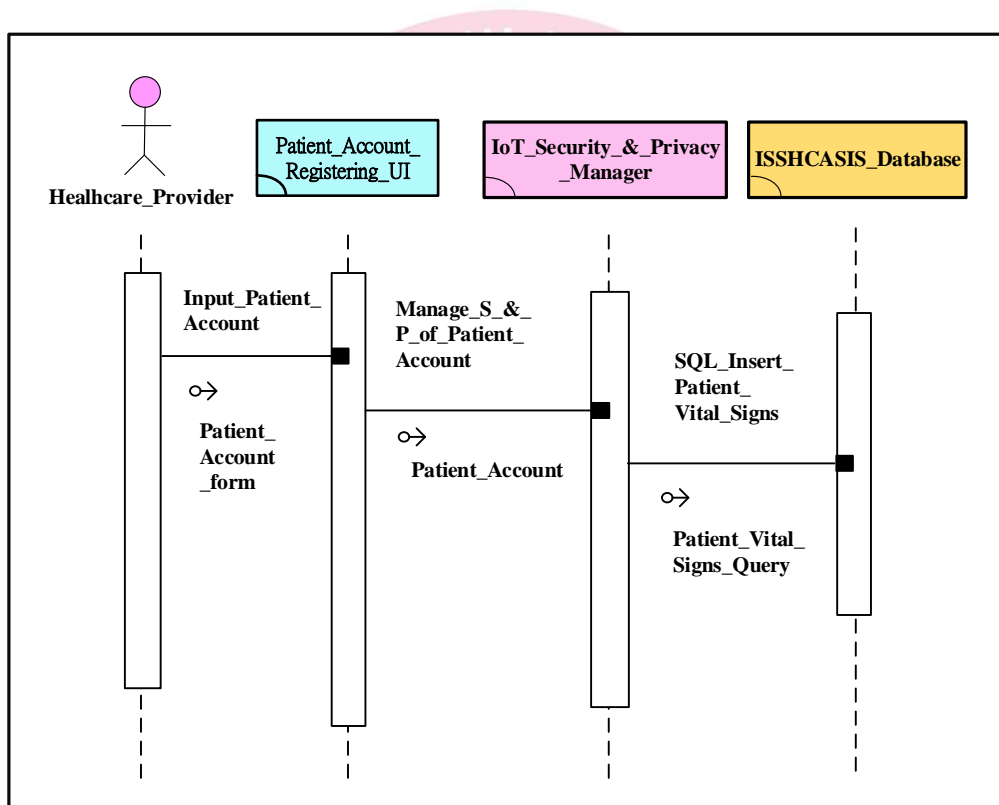


Figure 3.7 Interaction Flow Diagrams of the Registering_Patient_Account Behavior of
ISSHCASIS

Figure 3-8 shows an IFD of the Recording_Emergency_Response_Starting_Time behavior. First, actor Healthcare_Provider interacts with the Emergency_Response_Starting_Time_UI component through the Input_Emergency_Response_Starting_Time operation call interaction, carrying the Emergency_Response_Starting_Time_Form input parameter. Next, component Emergency_Response_Starting_Time_UI interacts with IoT_Security_&_Privacy_Manager component through Manage_S_&_P_of_Emergency_Response_Starting_Time operation call interaction, carrying input Emergency_Response_Starting_Time. Finally, component IoT_Security_&_Privacy_Manager interacts with the SHCASIS_Database component through the SQL_Insert_Emergency_Response_Starting_Time operation call interaction, carrying the Emergency_Response_Starting_Time_Query input parameter.

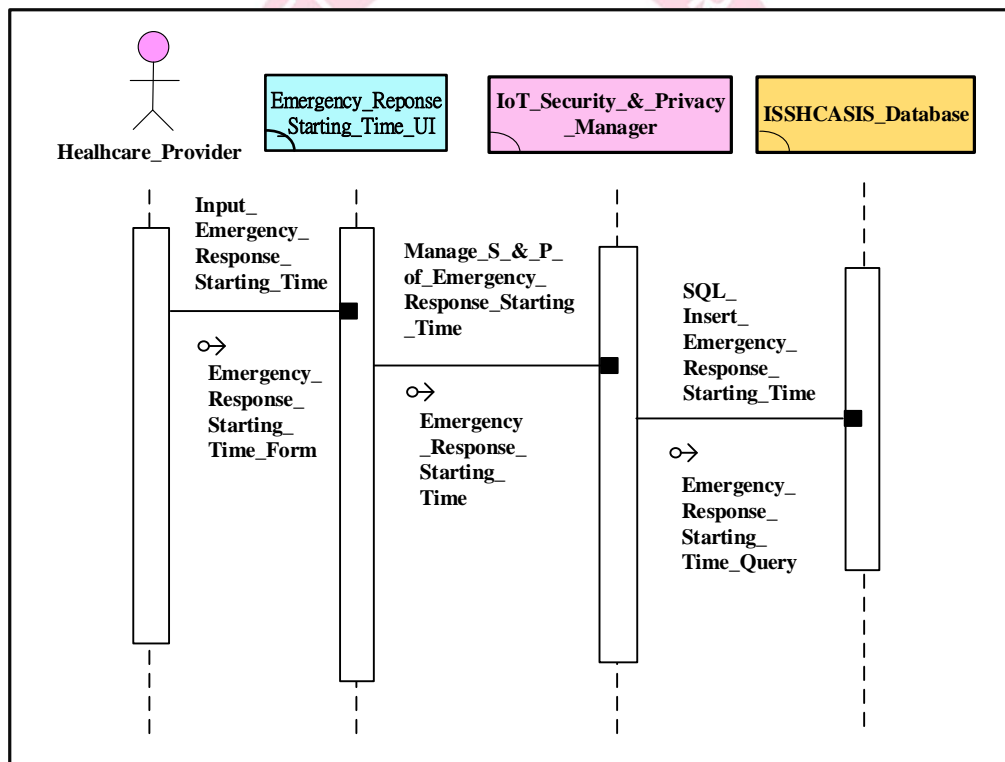


Figure 3.8 Interaction Flow Diagrams of the Recording_Emergency_Response_Starting_Time behavior of ISSHCASIS

Figure 3-9 shows an IFD of the Recording_Emergency_Response_End_Time behavior. First, actor Healthcare_Provider interacts with the Emergency_Response_End_Time_UI component through the Input_Emergency_Response_End_Time operation call interaction, carrying the Emergency_Response_End_Time_Form input parameter. Next, component Emergency_Response_End_Time_UI interacts with IoT_Security_&_Privacy_Manager component through Manage_S_&_P_of_Emergency_Response_End_Time operation call interaction, carrying input parameter Emergency_Response_End_Time. Finally, component IoT_Security_&_Privacy_Manager interacts with the ISSHCASIS_Database component through the SQL_Insert_Emergency_Response_End_Time operation call interaction, carrying the Emergency_Response_End_Time_Query input parameter.

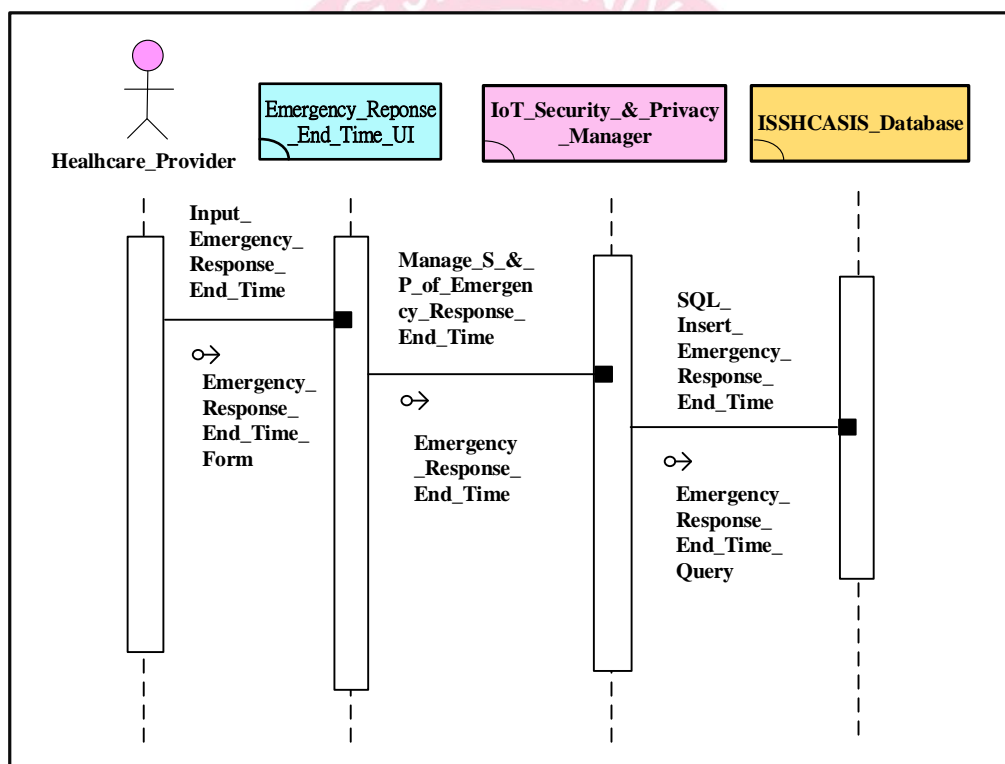


Figure 3.9 Interaction Flow Diagrams of the Recording_Emergency_Response_End_Time behavior of ISSHCASIS

Figure 3-10 shows an IFD of IoT Security and Privacy Management Behavior of ISSHCASIS. First, actor IoT_Security_Administrator interacts with IoT_Security_&_Privacy_Manager component through Manage_S_&_P_of_IoT_Security_Administration operation call interaction, carrying IoT_Security_Administration input parameter. Next, IoT_Security_&_Privacy_Manager component interacts with IAA_Controller component through the Identification_Authorization_Access_Control operation call interaction, carrying IAA_Control input parameter.

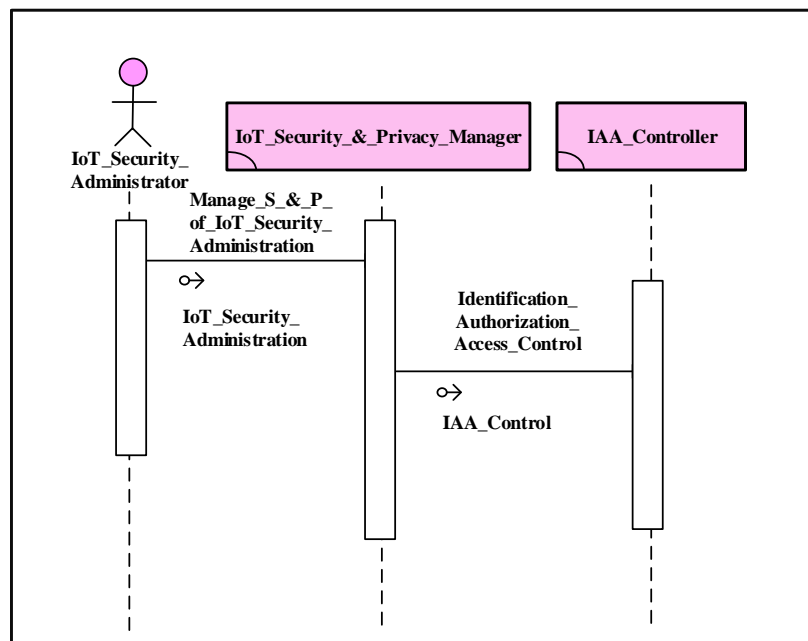


Figure 3-10 Interaction Flow Diagrams of the IoT Security and Privacy Management Behavior of ISSHCASIS

Figure 3-11 shows an IFD of the Sensing_Patient_Vital_Signs behavior. First, actor Server_Root interacts with the Patient_Vital_Signs_Daemon component through the Fork_PVSD_Process operation call interaction. Next, component Patient_Vital_Signs_Daemon interacts with the Patient_Vital_Signs_Sensor_P component through the Sense_Patient_Vital_Signs operation call interaction, carrying the Patient_Vital_Signs input parameter. And then, component Patient_Vital_Signs_Daemon interacts with the IoT_Security_&_Privacy_Manager component through the Manage_S_&_P_of_Fork_PVSD_Process operation call interaction, carrying Fork_PVSD_Process input parameter. Continuously, component IoT_Security_&_Privacy_Manager interacts with the ISSHCASIS_Database through the SQL_Insert_Patient_Vital_Signs operation call interaction, carrying the Patient_Vital_Signs_Query input parameter. Finally, actor Patient_Vital_Signs interacts with the Patient_Vital_Signs_Sensor_P component through the Sense_Patient_Vital_Signs operation call interaction, carrying the Patient_Vital_Signs input parameter.

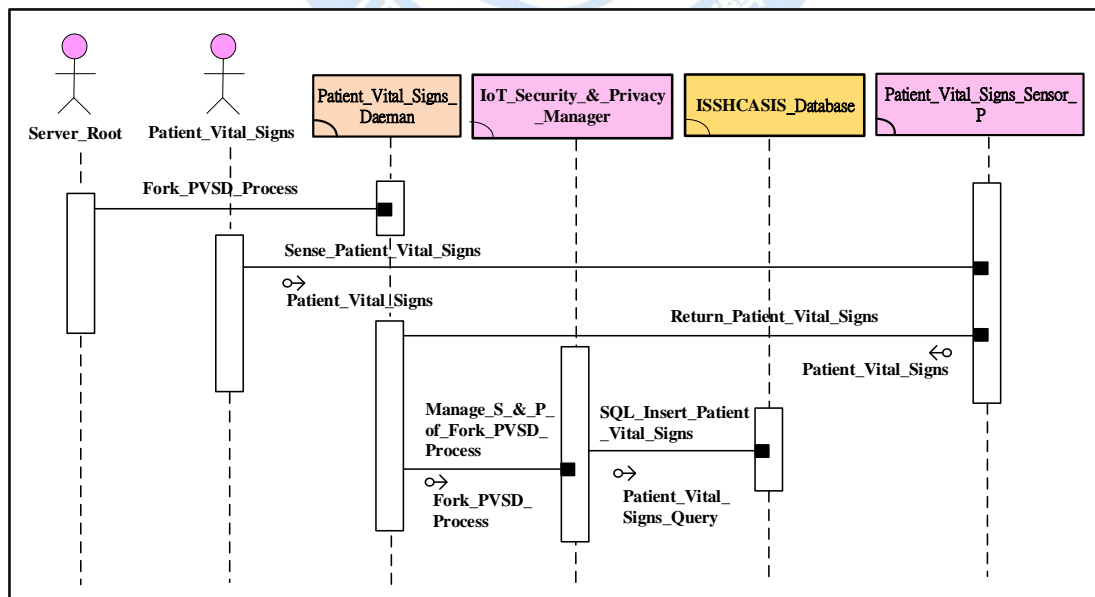


Figure 3-11 Interaction Flow Diagrams of the Sensing_Patient_Vital_Signs behavior of

4. Results and Discussions

In this chapter, we will explain Comparison between Structure-Oriented Model and Architecture-Oriented Model and Useful finding in the ISSHCASIS.

4.1.Comparison between Structure-Oriented Model and Architecture-Oriented Model

The innovation potential of IoT technology not only can improve human life such as smart healthcare, but also has numerous inherent vulnerabilities. Wireless sensor networks or radio frequency identification strengthen the privacy issue of IoT because they make information readily available and accessible through a wireless network. Information Security management for IoT that model the protection of security and privacy on the behavior of the components would be especially useful for the proper administration of cloud system and cloud services for Homecare. Security is one of the most important areas to be handled in the emerging area of cloud computing and IoT applications. If the security is not handled properly, the entire area of cloud computing and IoT applications would fail as it mainly involves managing personal sensitive information in a public network, or patient's vital signs in the healthcare.

ISSHCASIS can achieve a desirable blueprint for implementing information security management for IoT applications of an enterprise. Overall, since SBC architecture is good at describing complicated business processes and makes the business easy to understand and implement ISSHCASIS. ISSHCASIS helps an enterprise improving the communication efficiency, execution fast and increasing decision quality within an enterprise.

The ISSHCASIS provides a more holistic way, than a Reference Architecture for IoT (RAIoT) to describe how the information security management for IoT to be implemented. The ISSHCASIS not only states the process but also combine with the structures such as five-minute interval, healthcare provider, IoT security administrator, server root, and patient vital signs. The RAIoT gives us the ideas for “what should be done in the information security management for the ISSHCASIS?” but it does not provide us any information about “how to do in the information security management for the ISSHCASIS.” The ISSHCASIS achieves “what should be done in the information security management

for the ISSHCASIS?” and “how to progress in the information security management for the ISSHCASIS.” The ISSHCASIS is from architectural centric point of view to avoid unreasonable design from the requirements of the information security management for the ISSHCASIS.

From the structure and behavior of the information security management for the ISSHCASIS, we can clearly see who is responsible for the information security management and what processes should be evaluated for the information security management in transparent ways. The SBC diagram of ISSHCASIS shows the four interfaces, IoT Security & Privacy Manager, ISSHCASIS Database, Patient Vital Signs Daeman, and 6 behaviors, which going through the units. From the managerial point of view, we may see the weakness and pitfalls of the process, and then the higher priority may put attention on the IoT Security and Privacy Management Behavior and IoT Security & Privacy Manager to monitor the data writing or modifying to the ISSHCASIS Database. The ISSHCASIS architecture is compared to the Structure-oriented model RAIoT from WSO2 whitepaper, as shown in Table 4.1:

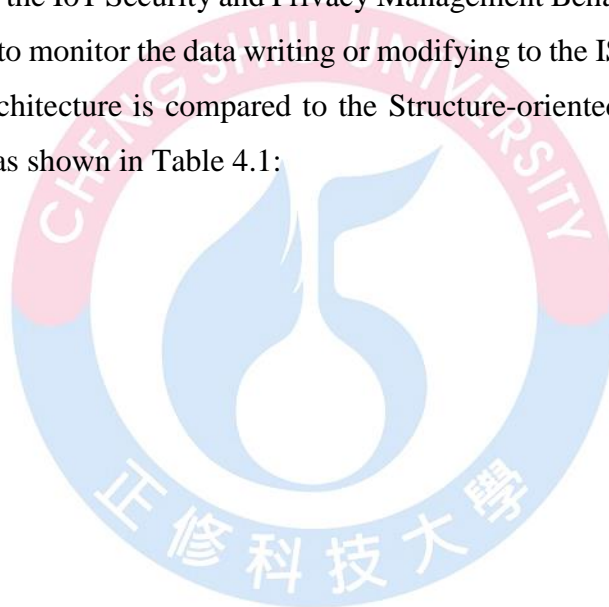
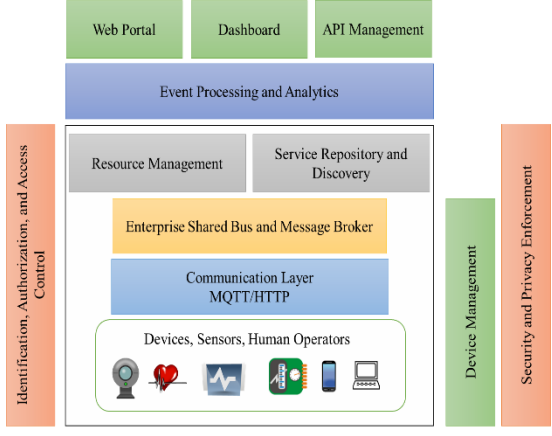
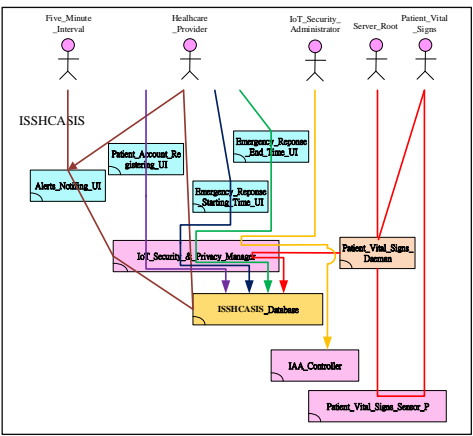


Table 4.1 Comparison between Structure-Oriented Model and Architecture-Oriented Model

RAIoT	ISSHCASIS
 <p>A Reference Architecture for IoT (Redraw from WSO2)</p>	 <p>Structure-Behavior Coalescence Diagram of ISSHCASIS (From This study)</p>

The RAIoT provided by WSO2, pre-described the information processes of the IoT only. Information security management is systematic method where results are obtained from pre-defined process. Therefore, the RAIoT only has organizational structures, but it does not have any guidance about organizational behaviors, and it will bring up difficulty in the organization to implement the information security management of ISSHCASIS.

Five external environment behaviors: Five_Minute_Interval, Healthcare_Provider, IoT_Security_Administrator, Server_Root, and Patient_Vital_Signs. The Healthcare_Provider initiates interaction with the four behavior: Alerts Notifying Behavior, Registering Patient Account Behavior, Recording Emergency Response Starting Time Behavior, and Recording Emergency Response End Time Behavior.

In this research, we proposed two information security mechanisms of the ISSHCASIS structure on IoT Security & Privacy Manager, and IAA Controller through the IoT Security and Privacy Management Behavior. The major task for the IoT Security & Privacy Manager is to protect of the patient's security and privacy that performs by guarding the patient's vital signs stored or access to the ISSHCASIS Database. The responsibilities for the IAA Controller are identification, authorization, and access control for patient's vital sign sensors to follow the IoT safety regulations in applied countries.

Table 4.2 Difference between RAIoT and ISSHCASIS with Items

Items	RAIoT	ISSHCASIS
Model Orientation	Structure-Oriented	Architecture-Oriented
Layers	Client/external communications, Event processing and analytics, Aggregation/bus layer, Relevant transports, Devices, and access/identity management and device manager.	Presentation, Logic, Data, Technology
Behaviors	No	6 behaviors: Alerts Notifying, Registering Patient Account, Recording Emergency Response Starting Time, Recording Emergency Response End Time Behavior, Sensing Patient Vital Signs, IoT Security and Privacy Management Behavior
Structures	12 Structures: Web/Portal, Dashboard, APIs, data storage, ESB, message broker, MQTT/HTTP/XMPP/CoAP/AMQP, Devices, access/identity management and device manager	9 Structures: Patient_Account_Registering_UI, Alerts_Notifying_UI, Emergency_Response_Starting_Time_UI, Emergency_Response_End_Time_UI, Patient_Vital_Signs_Daemon, ISSHCASIS_Database, Patient_Vital_Signs_Sensor_P, IAA_Controller, IoT_Security_&_Privacy_Manager
Security & Privacy mechanism	Not Clear	IoT_Security_&_Privacy_Manager
IoT sensor	Not Clear	IAA_Controller

control		
Security & Privacy Behavior	No	IoT Security Administration Behavior

4.2. Useful Findings and Discussions

In the comparison, we can see the Healthcare Provider playing a major role in the information security management of the ISSHCASIS. The Healthcare Provider initiates the four behavior for the information security management, which provides the ISSHCASIS database to access the patient's vital signs, and notifies the emergency notice to the other people to assist reducing the dangerous situations. The IoT Security Administrator, Server Root, Five Minute Interval, and Patient Vital Signs are also very important.

Five useful findings and discussions are stated as followings:

(1) Smart Healthcare Cloud Applications and Services Internet of Things (IoT) System, highlighting key information assurance IoT issues and identifying the associated security implications. Information assurance protection of IoT systems and information being stored, processed or transmitted from unauthorized access or modification of machine-to-machine devices, wireless sensor networks, and monitoring control and data acquisition systems.

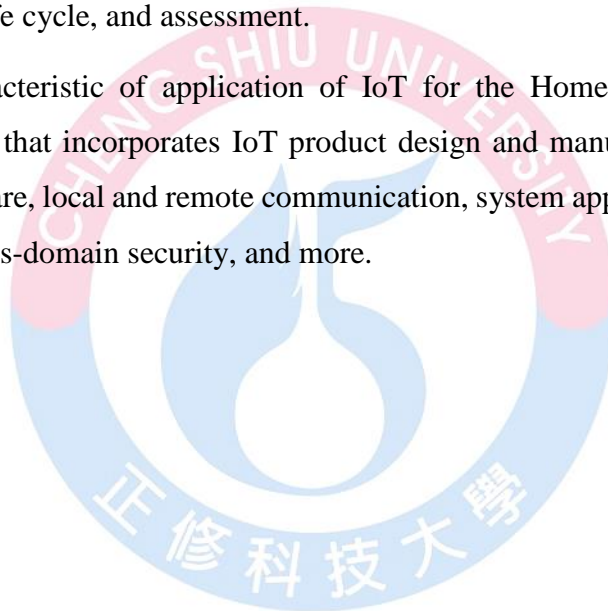
(2) In this study, we applied five-minute interval to alert patient's vital signs on current time to ensure message freshness from the IoT sensors. IoT Homecare applications might be more vulnerable facing this kind of attacks compared to other application scenarios; an outdated information could lead to inadequate medical interventions. To overcome this issue, it can be implemented using one of the following strategies: Random numbers, Sequence numbers, and Timestamps. Random numbers solution brings a drawback that the smart object has to maintain a list of the received nonce in its internal memory storage space. Sequence numbers solution brings a disadvantage, if one of the involved entities goes down e.g., hardware failure, the protection is no longer effective. On the other hand, timestamps solution is highly energy consuming to be implemented for constrained entities, as synchronized clocks should be maintained. To against replayed messages could be

achieved through the combination of the random numbers, sequence numbers, and timestamps solution strategies according to the network model specificities.

(3) As communication technology continues to evolve in organizations, it is vital to understand the impact that these advances IoT technology will have on different aspects of the business environment as well as the opportunity for further improvement. Effects of IT on system architecture, governance, and growth explores the influence of emerging technology such as IoT on different viewpoints associated with contemporary enterprise.

(4) Enterprise Cybersecurity empowers organizations of all sizes to defend themselves with next-generation cybersecurity programs against the escalating threat of modern targeted cyberattacks. It enables an enterprise to architect, design, implement, and operate a coherent cybersecurity program that is seamlessly coordinated with policy, programmatic, IT life cycle, and assessment.

(5) A key characteristic of application of IoT for the Homecare is that combine multiple disciplines that incorporates IoT product design and manufacturing, embedded hardware and software, local and remote communication, system application development and integration, cross-domain security, and more.



5. Conclusions and Recommendations

The Conclusions, Information Security Managerial Implications and Recommendations of this research are described in this chapter.

5.1. Conclusions

The information security incidents have been reported recently and the loss of enterprise operation becomes more and more serious problem. Risks are higher and higher for the enterprise because of information transformation and electronic commerce. Consequently, the requirement of an effective information security management framework for IoT is urgently need. In this study, we adopt the structure behavior coalescence methodology to construct an architecture-oriented Smart Healthcare Cloud Applications and Services Internet of Things (IoT) System (ISSHCASIS), which is integrated structure and behavior of the information security management model. ISSHCASIS solves many difficulties caused by the Structure-oriented approach the reference architecture for IoT in WSO2 such as uneven distribution of resources, poor safety performance, and high risk.

This research achieves “what should be done?” and “how to progress?” in information security management of ISSHCASIS model from architectural centric point of view to follow the IoT regulations. We find out the IoT security and privacy manager; identification, authorization and access control controller are two key roles of information security structures for the success of the information security management from structure behavior coalescence diagram. The feedback mechanism in the system is essential to report and respond to the incidents for reducing the risk.

5.2. Information Security Managerial Implications

After finishing building the ISSHCASIS model, we found out four information security management implications as followings:

- (1) ISSHCASIS is a structured and behavioral system integration model that provides a single interface for IT systems, emergency notification processes, and so on. Internet of things managerial personnel through ISSHCASIS to understand effectively of the information security management of the whole picture, to clear the responsibilities of the staff of various units, flexible allocation of enterprise resources. ISSHCASIS

also provides a good communication channel between the organization and the external environment.

- (2) To strengthen the information security management of the ISSHCASIS, this research proposed the screened security of the cloud service providers; the IoT security and privacy manager; identification, authorization and access control controller are two key roles for information security structures, and the Security and Privacy Management Behavior to implement accessing the cloud database security and privacy monitoring.
- (3) The identification, authorization and access control controller is responsible for compliance with the law of the Internet security control of the IoT sensors, to improve the strength of wireless communication verification, maintenance IoT detection equipment security, and ensure database security.
- (4) By this study introduction and elaboration of the enterprise architecture of protect security and privacy of personal information, we may understand clearly how the SBC helps architects effectively construct fruitful enterprise architectures. The ISSHCASIS enterprise architecture focus on: (1) Verifying input data for security and privacy checks before storing data in ISSHCASIS database. (2) Verify inputting emergency response starting or end time for security and privacy checks before updating data in ISSHCASIS database. (3) Verify PVS alerts data for security and privacy checks before updating data in ISSHCASIS database. (4) Manage IoT Security & Privacy is by configuring properly of IoT Security & Privacy manager, and managing IAA Controller for PVSSP. (5) IAA Controller manages identification, authorization, and access control of IoT for protection security and privacy. (6) IoT Security & Privacy Manager is used to manage IoT protocols, authentication, and encryption of patient vital signs.
- (5) This research reached the goals of resolving the problem of high complexity of information transitions system of IoT, high cost of development, and low expandability of system.

5.3. Recommendations

An understanding of cloud technology innovation with IoT applications become increasingly essential for IT practitioners, as entrepreneurs realize the business requirements fulfillment potential of an enterprise perspective of cloud computing and IoT applications. We recommend a SBC architecture development process, center is SBC framework and surrounded by nine architectures: cloud EA, cloud application architecture, cloud data architecture, cloud technology architecture, cloud integration architecture, cloud management architecture, cloud security architecture, cloud governance architecture and IoT application architecture. The information security management of enterprise can be achieved by using architecture-oriented information IoT model.



References

1. Ahlemann Frederik, Eric Stettiner, Marcus Messerschmidt and Christine Legner, 2012. *Strategic Enterprise Architecture Management: Challenges, Best Practices, and Future Developments*, New York, Springer, p. 314.
2. Ashton, K., 2009. *That 'internet of things' thing*. RFIJ, 22(7), pp. 97–114.
3. Aslam, Abdul, 2015. *Enterprise Cybersecurity: How To Build A Successful Cyberdefense Program Against Advanced Threats*, Apress.
4. Bahga, Arshdeep and Vijay Madiseti, 2014. *Internet of Things - A Hands-on-Approach*, 1st ed., VPT, pp. 446.
5. Bhatnagar, Rishi M, Frank Puhlmann, Dirk Slama, Jim Morrish, 2015. *Enterprise IoT*, O'Reilly Media.
6. Buyya, Rajkumar and Amir Vahid Dastjerdi, 2016. *Internet of Things*, New York: Morgan Kaufmann.
7. Chao, William S., 2012. *Systems Architecture: SBC Architecture at Work*, Taipei, LAP LAMBERT Academic Publishing, p. 344.
8. Chao, William S., 2016. *Systems Architecture of Smart Healthcare Cloud Applications and Services IoT System: General Architectural Theory at Work*, Amazon Digital Services LLC, pp. 116.
9. Chao, William S., 2016, *Systems Architecture of Smart Home Security Cloud Applications and Services IoT System: General Architectural Theory at Work*, Amazon Digital Services LLC, pp. 120.
10. Chellappan, V. and K.M. Sivalingam, 2016. *Security and privacy in the Internet of Things in Internet of Things*, Edited by Rajkumar Buyya and Amir Vahid Dastjerdi, New York: Morgan Kaufmann.
11. Dawson, Maurice, 2016. *Exploring Secure Computing for the Internet of Things*,

- Internet of Everything, Web of Things, and Hyperconnectivity*, Hershey: IGI Global.
12. Dhanjani, Nitesh, 2015. *Abusing the Internet of Things: Blackouts, Freakouts, And Stakeouts*, O'Reilly Media.
 13. Elk, Klaus, 2016. *Embedded Software Development for The Internet of Things*, Amazon Digital Services LLC, pp. 221.
 14. Elkhodr, Mahmoud, Seyed Shahrestani, Hon Cheung, 2016. *Internet of Things Research Challenges*, IGI.
 15. Eltayeb, Mohamed, 2017. *Privacy and Security in Security Solutions for Hyperconnectivity and the Internet of Things*, Edited by Maurice Dawson; Marwan Omar; Mohamed Eltayeb, Hershey: IGI Global.
 16. Gilchrist, Alasdair, 2015. *A Concise Guide to The Internet of Things for Executives*, RG Consulting, pp. 30.
 17. Gilchrist, Alasdair, 2016. *Industry 4.0: The Industrial Internet of Things*, Apress.
 - Greengard, Samuel, 2015. *The Internet of Things*, The MIT Press, pp. 232.
 18. Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M., 2013. Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems*, 29(7), pp. 1645–1660.
 19. Hameur, Amina, and Samiha Brahimi, 2016. *Background on Context-Aware Computing Systems in Internet of Things and Advanced Application in Healthcare*.
 20. Holler, Jan, Vlasios Tsiatsis, Catherine Mulligan, Stefan Avesand, Stamatis Karnouskos, David Boyle, 2014, *From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence*, Academic Press, pp. 352.
 21. Hu, Fei, 2016. *Security and Privacy in Internet of Things (IoTs): Model, Algorithms, and Implementations*, CRC Press, pp. 604.
 22. International Telecommunication Union, 2012. *ITU-T Recommendation Y.2060: Series Y: Global information infrastructure, internet protocol aspects and next-*

- generation networks: Frameworks and functional architecture model: Overview of the Internet of Things*. Geneva: International Telecommunication Union.
23. Kellmereit, Daniel, and Obodovski, Daniel, 2013. *The Silent Intelligence: The Internet of Things*, DND Ventures LLC, pp. 166.
 24. L.R. LLC., 2013. *An introduction to the Internet of Things (IoT)*. http://www.cisco.com/c/dam/en_us/solutions/trends/IoT/introduction_to_IoT_november.pdf, Browsed on Nov. 30, 2016.
 25. Ma, Wei-Ming, 2010. *Study on Architecture-Oriented Information Security Risk Assessment Model, Computational Collective Intelligence Technologies and Applications*, Volume 6423/2010, pp. 218-226.
 26. Ma, Wei-ming, Tsai, Cheng-F, 2012. *Study of Implementation of the Personal Information Protection Act Architecture on CSU Campus*, 2012 Symposium on Global Business Operation and Management, Kaohsiung, Taiwan.
 27. Ma, Wei-ming, 2013. Study on Enterprise Architecture Development, *Journal of Global Business Operation and Management*, 5, pp. 57-71.
 28. Miller, Michael, 2015. *The Internet of Things: How Smart TVs, Smart Cars, Smart Homes, and Smart Cities Are Changing the World*, Que Publishing, pp. 335.
 29. Minerva, Roberto, Abyi Biru, Domenico Rotondi, 2015. *Towards a definition of the Internet of Things (IoT)*, IEEE, http://IoT.ieee.org/images/files/pdf/IEEE_IoT_Towards_Definition_Internet_of_Things_Revision1_27MAY15.
 30. Moolayil, Jojo, 2016. *Smarter Decisions – The Intersection of Internet of Things and Decision Science*, Packt Publishing.
 31. Natarajan, K., Prasath, B., & Kokila, P., 2016. Smart Health Care System Using Internet of Things. *Journal of Network Communications and Emerging Technologies*, 6(3).
 32. Ning, Huansheng, 2013, *Unit and Ubiquitous Internet of Things*, CRC Press, pp.267.

33. NMazima, J.k., M. Kisangiri, D. Machuve, 2013. Design of Low Cost Blood Pressure and Body Temperature Interface”, *International Journal of Emerging Science& Engineering*, 1(10).
34. Norris, Donald, 2015. Norris, *The Internet of Things: Do-It-Yourself at Home Projects for Arduino, Raspberry Pi and BeagleBone Black*, McGraw-Hill Education TAB, pp. 352.
35. Penttinen, Jyrki T. J., 2016. *Wireless Communications Security: Solutions for The Internet of Things*, Wiley, pp.336.
36. Pohls, H. C., Angelakis, V., Suppan, S., Fischer, K., Oikonomou, G., Tragos, E. Z., Mouroutis, T., 2014. *RERUM: Building a reliable IoT upon privacy-and security-enabled smart objects*. In *Wireless Communications and Networking Conference Workshops (WCNCW)*, IEEE, pp. 122-127.
37. Ren, K., Samarati, P., Gruteser, M., Ning, P., & Liu, Y., 2014. Guest Editorial Special Issue on Security for IoT: The State of the Art. *Internet of Things Journal*, IEEE, 1(5), pp. 369–371.
38. Russell, Brian, and Drew Van Duren, 2016. *Practical Internet of Things Security*, Packt Publishing, pp. 336.
39. Romdhani, Ilmed, 2017, Confidentiality and Security for IoT Based Healthcare in *Securing the Internet of Things* by Li Da Xu, Shancang Li.
40. Sabina, Jeschke, Brecher, Christian, Houbing Song, and Danda B. Rawart, 2016. *Industrial Internet of Things: Cyber manufacturing Systems*, Springer.
41. Schwartz, Marco, 2014. *Internet of Things with the Arduino Yun*, Packt Publishing, pp. 112.
42. Schwartz, Marco, 2015. *Internet of Things with the Photon*, Amazon Digital Services LLC, pp. 66.
43. Schwartz, Marco, 2015. *Internet of Things with the Raspberry Pi-Build Internet*

- of Things Projects Using the Raspberry Pi Platform*, Open Home Automation, pp. 54.
44. Simon, Daniel and Christian Schmidt, 2015. *Business Architecture Management- Architecting the Business*, Springer.
 45. Smith, Sean, 2017. *The Internet of Risky Things*, O'Reilly Media.
 46. Spaanenburg, Lambert, 2016. *The Role of Time in Health IoT*, in *Internet of Things and Advanced Application in Healthcare*, IGI Global.
 47. Stackowiak, Robert, Art Licht, 2015. *Big Data and The Internet of Things: Enterprise Information Architecture for A New Age*, Apress.
 48. Tamura, T. et al., 1998. "Fully automated health monitoring system in the home," *Medical Engineering & Physics*, 20(8), pp. 573-579.
 49. Waher, Peter, 2015. *Learning Internet of Things*, Packt, pp. 242.
 50. Wears, R. L., & Leveson, N. G., 2008. *Safeware: Safety-critical computing and healthcare information technology. Advances in Patient Safety: New Directions and Alternative Approaches*, 4, pp. 1-10.
 51. Weber, R. H., 2010. Internet of Things - New security and privacy challenges. *Computer Law & Security Report*, 26(1), pp. 23-30.
 52. WSO2, 2014. A reference architecture for the Internet of Things. http://wso2.com/wso2_resources/wso2_whitepaper_a-reference-architecture-for-the-internet-of-things.pdf.
 53. Zhang, Z. K., Cho, M. C. Y. S., 2015. *Emerging Security Threats and Countermeasures in IoT*. In *Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security*, (pp. 1-6). ACM.
 54. Zhou, Honbo, 2014. *The Internet of Things in the Cloud: A Middleware Perspective*, CRC Press, pp. 348.