jamk.fi

IoT from cyber security perspective

Case study JYVSECTEC

Ville Sulkamo

Master's thesis
June 2018
School of Technology, Communication and Transport
Information Technology
Master's Degree Programme in Cyber Security

jamk.fi

Description

Author(s)	Type of publication	Date
Sulkamo, Ville	Master's thesis	June 2018
		Language of publication: English
	Number of pages: 98	Permission for web publication: x

Title of publication
IoT from cyber security perspective
Case study JYVSECTEC

Degree programme
Cyber Security

Supervisor(s)
Rantonen Mika

Assigned by
JYVSECTEC, Vatanen Marko

Abstract

The objective of the master's thesis was to focus on cyber security perspective on the environment and appliances of the Internet of Things. The number of IoT devices is increasing all the time, and these devices are used practically in all areas of everyday life. Therefore, securing the IoT devices is gaining more and more importance.

The selected research method was divided into two-parts. The first part was about conducting to study and investigating the environment and devices of the Internet of Things from architectural perspective; what is available on the market, what kind of appliances are used and for what purposes. This part also includes the basic rules for protecting such an environment against possible cyber vulnerabilities and attacks.

The second part of the thesis consists of a practical case study against a cloud based IoT appliance. That IoT appliance consist of active sensors and servers related to the collecting data from sensors and cooling using which is controlled by the sensors data. This part also included possible attack vectors against that live environment. The case study part also focused on different kinds of vulnerabilities are aimed at IoT environments.

The important part of case study was to define recommendations of basic functions for protecting the IoT environment. Additionally, the recommendations were defined for the management level to improve the security controls at organizational level against cyber security risks.

As a conclusion, the case study showed that the Internet of Things environments are also under substantial risk of cyber threats. Therefore, the needed security processes and their implementation are very highly recommended. The minimum recommendation is to keep software modules updated.

Keywords/tags (subjects)
Internet of things, IoT, cyber security, attack vector

Miscellaneous

jamk.fi

Kuvailulehti

Tekijä(t)	Julkaisun laji	Päivämäärä
Sulkamo, Ville	Opinnäytetyö, ylempi AMK	Kesäkuu 2018
	Sivumäärä 98	Julkaisun kieli Englanti
		Verkkojulkaisulupa myönnetty: x

Työn nimi
IoT from cyber security perspective
Case study JYVSECTEC

Tutkinto-ohjelma
Cyber Security

Työn ohjaaja(t)
Mika Rantonen

Toimeksiantaja(t)
Marko Vatanen, JYVSECTEC

Tiivistelmä

Opinnäytetyössä tutkittiin verkkoon kytkettävien laitteiden, Internet of Things, IoT -laitteiden turvallisuutta. Nykyisin IoT-laitteita on monilla elämänalueilla eri yhteyksissä, niin kuluttajatuotteissa kuin teollisuuden puolella.

Valittu tutkimusmetodi jakautui kahteen osaan. Ensimmäisessä osassa perehdyttiin IoT-laitteiden arkkitehtuuriin ja mahdollisiin käyttökohteisiin. Käytiin läpi yleisiä suojausmenetelmiä, joilla voidaan parantaa IoT-laitteiden kyberturvallisuutta.

Toisessa osassa työtä tutkittiin IoT-ympäristön kyberturvallisuutta, joka oli toteutettu JYVSECTEC:in RGCE-pilvipalvelussa. Kyseisessä ympäristössä oli kaksi lämpötila-anturia, joiden tuottamalla datalla ohjattiin jäähdytinyksikköä.Tutkimuksessa keskityttiin pohtimaan mahdollisia hyökkäysvektoreita kyseistä ympäristöä vastaan.

Tutkimuksessa suoritettiin käytännössä haavoittuvuustyökalujen ajaminen kyseistä ympäristöä vastaan. Tehtyjen haavoittuvuustestausten perusteella ympäristölle tehtiin suositukset ympäristön turvallisuutta parantamaan. Tutkimuksen perusteella tehtiin myös suosituksia hallintotasolle siitä, kuinka tarvittavat prosessit ja menettelytavat tulee olla etukäteen määriteltyinä ja kommunikointi ongelmatilanteissa sovittuna.

Johtopäätöksenä tutkimuksen perusteella voidaan todeta, että IoT-laitteet ovat nykyisin houkuttelevia kohteita mahdollisille ulkopuolelta tuleville verkkohyökkäksille. Siten niiden kyberturvallisuutta ei saa unohtaa eikä laiminlyödä. Vähintään suositellaan tehtäväksi ohjelmistojen säännölliset päivitykset kyberturvallisuutta parantamaan.

Avainsanat (asiasanat)
Internet of Things, IoT, kyberturvallisuus, hyökkäysvektorit

Contents

Figures

Tables

1 Introduction

In today's world, basically all devices are interconnected to each other via networks. There are multiple devices are in homes, offices, cars and production plants and they run various tasks to help with daily tasks. The number of connected devices is increasing all the time because manufactures present every day new internet connected devices for helping the users of these devices in their everyday life and creating new digital experiences. The existing and new Internet based devices are related to smart house appliances, smart cities, smart energy plants, automobiles, health care services, retail stores and transportation. Examples from those areas are home surveillance cameras and fridges, smart city applications for helping citizens to find a vacant parking slot and for health care sector's personal trainer appliances.

The key points with all those devices and appliances are that they are connected to the internet with the key purpose to improve the quality of life by offering digital experiences. Those devices produce different kind of information, raw data, and data and information are shared with other systems. The data which internet connected devices generate can be stored and then used for various purposes. For example, a fridge can tell the owner about the shortage of groceries which need to be ordered. One example could be from health care sector about the wearable sensor or monitor which tells the person's state of health. That information is then shared via network with professionals in health care.

There are plenty of different examples for different areas of life. One significant area using Internet connected devices is industrial production. The manufacturing business uses different kinds of sensors and monitoring tools for collecting important data from manufacturing machines and their conditions; based on that data the production is adjusted to be more effective. Also, one very important area is the collection of maintenance data from machines proactively; hence, maintenance breaks can be planned based on that data with shortest possible breaks. A very good live example about this appliance is "Train as Service" in the UK. The Azuma trains are manufactured by Hitachi and the idea is to collect sensor data from trains online and use that data as a base for maintenance and schedule the maintenance breaks

for individual trains based on that sensor data. (New Azuma trains arrive at UK port ahead of passenger services starting later this year 2018.)

As can be seen, the world of connected devices is versatile covering basically all aspects of life.

2 Defining the Internet of Things

As described in the previous chapter, the aim of the internet connected devices is to help with everyday living. What are internet connected devices? There is a term to describe the internet connected devices; they are called Internet of Things devices, i.e. in short, IoT.

2.1 General

The term Internet of Things as a concept was mentioned first time in the 1990s. The current format of the expression was proposed by Kevin Asthon in 1999. (Ashton 2009.)

What is Internet of Things? How is the term defined? What does it mean? According the Gartner (N.d.), the industrial Internet consists of physical devices capable to monitor surroundings and transmit the monitored data to other devices and perform actions based on the monitored data intelligently. In short, it can be said that Internet of Things, IoT, relates to devices connected to each other via different communication methods and channels. Those devices are capable to transmit data and communicate with each other. The communication method and transmit paths can either be wireless or wired data paths, depending on the IoT device and its purpose. According to different research institutes such as Gartner (2017), there will

be up to 25 billion IoT devices by 2020. However, the figure can be much higher than estimated.

The Internet of Things, IoT, is defined also by the Infrastructure of the International Telecommunication Union's (ITU) in Global Standard Initiative (Internet of Things Global Standards Initiative N.d.). The IoT is defined as global infrastructure for the information society for interconnecting physical and virtual assets, things, based on evolving interoperable communication channels and technologies. (ibid.).

Lawrence Miller in "IoT Security for Dummies, INSIDE Secure Edition" (2016) defined the IoT as follows: IoT covers devices and objects connected over different communications protocols to each other. The devices can be computing devices, laptops or desktop computers or even tablets and smartphones. The communication paths in IoT are typically Bluetooth and Long Range Wide Area, Lora WAN, or mobile phone based wireless transmission paths like 3G and 4G type of transportation paths. Those transmission paths are typically defined as low power protocols, because typically IoT devices send a small amount of data with low transmit speed and in low range. In the future, IoT devices will can communicate with long range infrastructure with modern technologies such as 5th generation mobile networks, abbreviated as 5G. (IoT Innovation 2018; Tarkoma 2017.)

2.2 Typical IoT devices nowadays

IoT devices can be divided into consumer-based devices such as domestic appliances and home automations, and appliances for industrial use such as different kinds of sensors for measuring temperature, humidity and movement. Typical domestic appliances are e.g. surveillance cameras, network switches, routers and Network Attached Storages (NAS), fridges, smart televisions and cars. For home automation, i.e. smart homes, typical appliances are e.g. heating and ventilation systems, lighting control systems and different sensors for monitoring humidity and CO2 levels inside a house or buildings. (Internet of Things 2018.)

When talking about the industrial IoT, the term “Industry 4.0” needs to be mentioned. Industry 4.0 covers different kind of automation and manufacturing data exchange technologies. It has also been called “Smart Factory”. Smart factory covers cyber physical systems, the Internet of Things, cloud computing and cognitive learning. The term Industry 4.0 was first time mentioned in German government memo released 2013 (Moore 2018.) and the aim of that memo was to introduce high-tech strategy for fully almost computerised manufacturing industry without the human involvement. (Moore 2018.) In summary, Industry 4.0 means that factory machines and sensors communicate with each other with less or no human’s involvement. Therefore, automation with Industry 4.0 together with intellectual monitoring provides higher level automated production, which improves the quality of end-user products. Based on monitoring the results of the sensors, machines can adjust their run in a more efficient way. Normally, home computers and laptops or mobile phones do not count as IoT devices. (Moore 2018.)

2.3 Future of IoT

According to Gartner, (2017) there will be as many as 20 billion connected IoT devices in the year 2020. However, the fact is that the number of connected IoT devices in 2020 could be even higher because of inventions of new IoT appliances. Earlier, the growth trend has been as high as 31% from year to year. Currently in 2017, the consumer segment presents 63% of the total number of IoT appliances. Also, the key factor will be on consumer side and low-cost devices playing a major part in the number of volumes. (ibid.)

When talking about the future of the IoT, also market value plays a significant role. Even though consumers are purchasing bigger numbers of IoT appliances, the business side invests more in IoT devices. In 2017, 57% of spending on the IoT market came from business side. The growing trend comprises IoT services; still a small but rapidly growing sector. Figure 1 below shows the market values for IoT market from year 2016 up to 2020 as estimated. (ibid.)

Category	2016	2017	2018	2020
Consumer	532,515	725,696	985,348	1,494,466
Business: Cross-Industry	212,069	280,059	372,989	567,659
Business: Vertical-Specific	634,921	683,817	736,543	863,662
Grand Total	**1,379,505**	**1,689,572**	**2,094,881**	**2,925,787**

Figure 1. IoT market value, 2016 – 2020 (Gartner 2017)

3 Cyber Security

The term “Cyber Security” needs to be defined first. The definition of this term is problematic, and the term can be written in several ways depending on speaker’s language. The word “cyber” itself is rather old, already seen in BBC’s fictional TV series Doctor Who when he battled against cyberman in 1966. (ENISA 2015, 10; Doctor Who N.d.)

In 1985 writer William Gibson published his novel “Neuromancer” where he defines the term “cyberspace” as

> *… a consensual hallucination experienced daily by billions of legitimate operators, in every nation, by children being taught mathematical concepts... A graphic representation of data abstracted from the banks of every computer in the human system. Unthinkable complexity. Lines of light ranged in the non-space of the mind, clusters and constellations of data. Like city lights, receding*.

(ENISA 2015, 10.)

As can be seen, the definition of the term “cyber” as such is not relevant. The term cyber is nowadays also connected to different kinds of areas such as cyberpunk, which is a fiction type of novels; cybergoth is related to the music genre, cyberbullying, i.e. bullying others via internet or social media; and cybercrime which

is crimes made with help of computers. There are also other areas with which the term cyber can be connected, however, the focus of this thesis is on cyber security.

The Oxford English Dictionary (N.d) defines cyber security in the following way: "*The state of being protected against the criminal or unauthorized use of electronic data, or the measures taken to achieve this.*" In practice, the above sentence means that any unauthorized or criminal based misuse of electronic data or device is understood as cyber threat. Manipulating physical assets can also be measured as jeopardizing cyber security. However, the connection link between cyber security and information security is quite small because in many cases, the cyber security issues can be converted to information security issues and vice versa. Many public sources list these terms as synonyms. (ibid.)

ENISA's (2015, 10) for cyber security includes different domains which all together are measured as cyber security. The following figure (Figure 2) defines the connections between domains (ibid., 11):

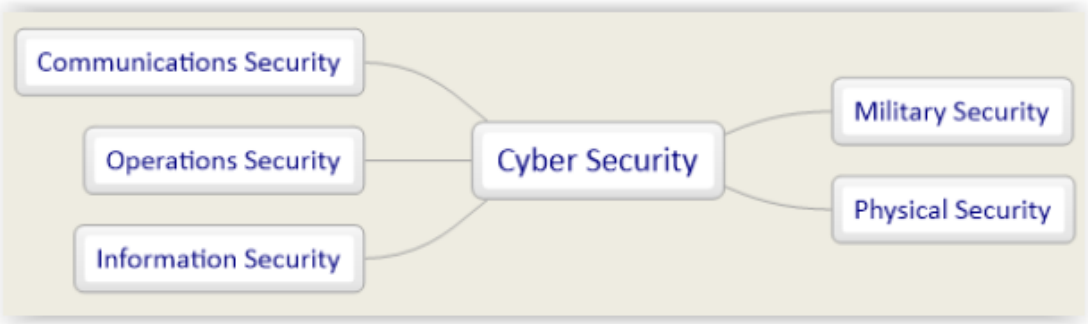


Figure 2. connections related to the cyber security

The purpose of different domains is to protect corresponding areas. The following table defines different domains and their purpose in a more detailed way (ibid.).

Table 1. Domain definitions in cyber security

Domain	**Definition**
Communications Security	protection against threat which tries to influence technical infrastructure and manipulate characteristic values in such a way which is not intended by owner or designer
Operations Security	protecting against a threat which tries to manipulate processes or workflows into unwanted results
Information Security	protecting the data stored in cyber system against threat of theft, manipulation or deletion; in short InfoSec.
Physical Security	protecting the physical assets related to a cyber system; the assets could be servers, storages or network components. Also, protecting against unauthorized access is included in this domain.
Military Security	protecting against threats which are against physical assets; however, have a flavour of political, military or strategic aspect.

As the figure and table above show, all those different security areas focus on their own speciality; however, they all are related to cyber security itself in the end. From that point of view, cyber security could also be understood as an umbrella term for different security domains.

The definition in the following Figure 3 (ENISA 2015) shows the relationships between different components included in cyber security. The key point is to notice that cyber security is not only for technical protection of environment, but it includes key elements related to organizations, e.g. CIA and it also involves assets and possible threat sources.

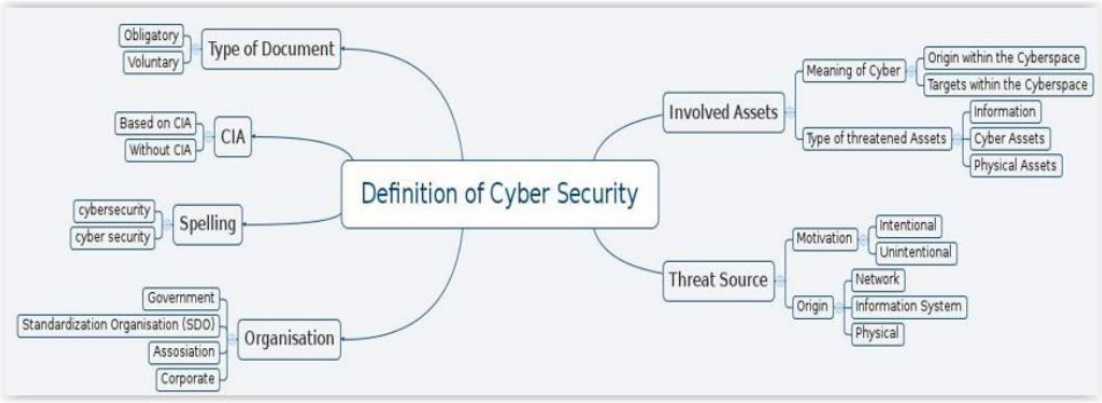


Figure 3. Connections related to the cyber security

One important definition related to cyber security is CIA. CIA stands for Confidentiality, Integrity and Availability. With those terms it is defined that confidential material should be kept safe and one should be sure that it is not changed by an unauthorized counterpart. It should be available when needed for authorized users. CIA should be a key element on organizational level when defining information security and cyber security policies.

There are few standards which define cyber security and its connections to information security. The ISO/IEC 27000 (2018) covers the information security which considers any assets related to cyber space under control. The ISO/IEC 27032 (2012), covers cyber security as its own subject; however, it intends to cover only assets connected to the internet. Other standardization organizations e.g. ITU-T, ETSI TC Cyber, National Institute of Standards and Technology (NIST), NATO Cooperative Cyber Defence Centre of Excellence and Committee on National Security Systems (NCSS) have defined the cyber security in their own documentations. The main principle in all these different organizations is the same; however, they have their own views of cyber security based on the intention of the organization.

The main sources from standardizing in this master's thesis are ISO/IEC 27000 and ISO/IEC 27032.

One basic term to be covered under this topic is Security Policy. In practice, it means securing the individual system, organization or other entity. The policy covers instruction and how to react in different types of security risks. One important term is also Cyber Security Policy, the aim of which is to provide instruction on how to protect public and private infrastructure against cyber threats and cyber-attacks. The Cyber Security policy can consist of controls against physical access to the hardware but also security controls for access via network, data or code injection. Like every other security policy, this also needs be agreed and defined by organization's or company's management together with security specialist. (Salonen 2017.)

4 Research questions

The purpose of this thesis is to research and study the cyber security level of IoT environment. The first part of thesis researches IoT environment from a theoretical perspective. The purpose is to gain a close understanding of the architectural concept and view of IoT environment. Also, the purpose is to understand what possible security methods protect the IoT environment. The second part of this thesis focuses on a case study environment which was built up to JYVSECTEC RGCE virtual environment. The environment consists of basic IoT elements such as temperature sensors and a cooling machine as well as data management applications. The following topics cover the test environment in a more detailed way.

The purpose of this thesis is to find the answers to the following research questions:

- What is the initial level of cyber security within the IoT environment?
- What is the minimum acceptable level of cyber security in IoT environment?
- What are the minimum implementations for improving the cyber security in the IoT environment?

Those questions together with thesis topic are purely out of the author's own interest.

5 The basic theory and key modules for the IoT

5.1 Basic theory

When discussing the Internet of Thing, IoT, devices or appliances, the commercial and industrial IoT appliances must be distinguished from each other. The commercial devices typically comprise smart house automation, surveillance cameras or sensors for providing more convenience for living. Nowadays, even automated vacuum cleaners and smart televisions can be counted as IoT devices. The number of IoT devices is increasing all the time as for smart homes, smart cities and the health care sector. (Miller 2016.)

The industrially based IoT appliances, Industry 4.0, are typically sensors or similar devices generating measurable raw data from factory machines for providing information to be based on rational decision making. The purpose of the industrial IoT devices is to provide more information about an industrial process or a machine for guiding the production. The aim with the Industry 4.0 is to go towards smart factories even with less human involvement in production. (Moore 2018.)

5.2 Key modules of the IoT

The basic IoT devices or appliances typically consist of following parts from the architecture point of view:

1) sensor or any other smart object
2) transport layer either aerial or wired
3) switches or routers
4) data collection server

The basic modules in IoT environment are typically sensors for collecting measurable data, transport layer for delivering sensor data, computing devices and data analysis application and storage for storing the data. Figure 4 highlights the architecture point of view of the IoT environment. (Miller 2016, 6.)

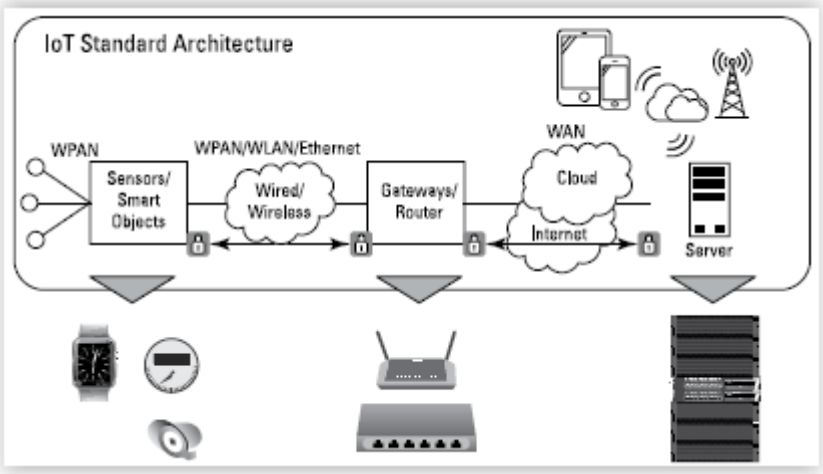


Figure 4. IoT system architecture and functional overview (Miller 2016, 6)

As can be seen above, the key components are not just sensors or smart objects and servers for collecting data but the data path, the network, has a very important role in this architectural concept. The network together with the used protocol plays a significant role within IoT appliances. (Miller 2016, 32). The basic IoT architecture components are described more in depth in the following chapter.

When considering the IoT device itself and excluding all external counterparts such as transmission paths, network components such as routers and data analysis

servers, the internal architecture of a simplified IoT device can be presented Figure 5. (Miller 2016, 7)

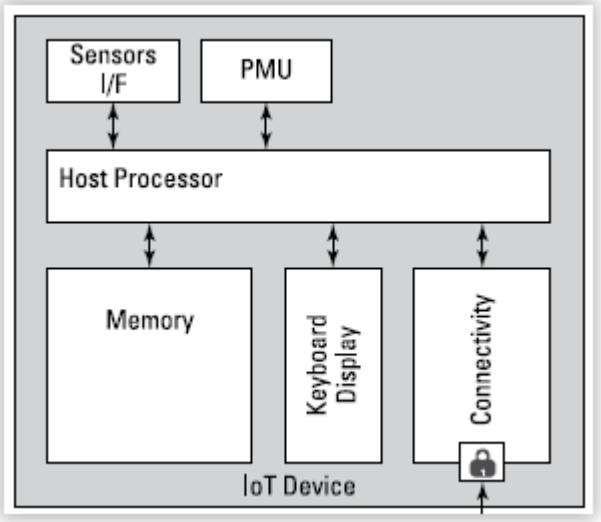


Figure 5. IoT device architecture (Miller 2016, 11)

There are similarities in IoT device architecture when it is compared to normal computers. As can be seen, the IoT device includes a host processor which is responsible for managing and controlling this device. The memory is used for storing data from sensors; however, also for storing application code for running purposes. The input/output devices such as keyboard and monitor are for user interfaces. The connectivity is implemented via network, wired or wireless. If the network is wireless, there are different methods to perform: WIFI Ethernet, Bluetooth or mobile connectivity via GSM 4G. The power source provides necessary power to the system. The sensors collect e.g. ambient temperature data from environment.

These are the same modules and components as the main part of a normal computer's architecture has. The difference between IoT devices and normal computers is sensors dedicated for collecting information from ambient environment.

Also, one key difference between IoT and normal computer is the size. Typically, IoT devices are much smaller and their power consumption is significantly lower than that of normal computers. IoT devices might be typically powered with small rechargeable or replaceable batteries. (Miller 2016, 32.)

5.3 Sensors

The purpose of sensors is to collect measurable data from a machine, device or ambient environment. The sensor is an electronic component which transforms temperature change, humidity or movement to electronic format and transmits the data to computer. Sensors are used widely in industry for measuring different parameters and in domestics IoT appliances. (Sensor 2012.)

The sensors are used in everyday appliances for various purposes. For example, sensors are used in cars for measuring the gas exhaust level and based on that information, they adjust the settings of gasoline and oxygen of the car's engine. One area for sensors are infrared sensors used for remote-control boxes for home entertainment, for motion detection and intrusion detectors. (Sensor 2012.)

5.4 Network

All devices need a network for transferring and receiving data for different purposes. When talking about IoT devices, there are basically two different options available, wired network with copper or fibre connections or wireless connections with different kinds of connections methods and protocols. The chosen transfer method always depends on what kind of IoT appliance it is, and what its purposes are. Also, the used protocol plays a significant role with IoT appliances. At this point, the power consumption should be mentioned because consumption differs between different

connections types. When counting all those different possibilities together, the transport layer with IoT devices is quite manifold. (Internet of Things 2018.)

The wired network is usually produced with copper or fibre cables. Those methods support different kinds of connectors such as RJ45, which is widely used in computers for networking devices. If IoT devices are used with fibre connectivity, there are various connectivity types for networking. A typical connector type nowadays is SC snap-in connector. The benefit for using copper instead of fibre for connections is the cost. Copper cables are cheaper, and they resist more robust handling compared to fibre cables. The benefit with fibre cables is the transfer speed (copper 10G, fibre 100G) when compared to copper cables. (Guide to Fiber Optics & Premises Cabling. N.d.)

For wireless connections, there are multiple choices available. The used transfer method should be selected based on the needed transfer distance because there are various categories for distance: short-range, medium-range and long-range wireless connections. Examples for short-range are Bluetooth, Z-Wave and ZigBee transfer methods. The reason for using the short-range method is that these IoT devices are used as a personal area network (PAN) device, and they only need short-range low-latency data transfer rate. The physical range is typically from few meters up to 100 meters. The key point is also low power consumption. (Internet of Things 2018.)

With medium-range wireless transfer methods, the low power consumption is also an important feature. The data rate might also be low; however, there might be a need for high-speed communication. Examples of these are HaLow, 802.11ah (IEEE 801.11ah 2018) and LTE Advanced which is the enhanced version of Long Term Evolution, LTE (LTE Advanced).

Examples of long-range communication channels are Low-Power wide-area networking, LPWAN, very small aperture terminal VSAT and long-range WIFI connectivity methods. The important design factor also in this category is low data rate with reduced power consumption. Some examples of LPWAN are Lora WAN, SIGFOX and NB-IoT. (Internet of Things 2018.)

A very common transport method today is to use 4G mobile network. It provides an effective method to transfer data with high speed, and the amount of data can also

be very high. In the future, the fifth generation of mobile network, 5G, will provide higher speed and high data range within a new frequency band to provide a more effective path for future IoT devices. (LTE Advanced 2018.)

The used protocols with IoT devices are e.g. message Queuing Telemetry Transport, MQTT, which works on top of the TCP/IP protocol. The benefit with MQTT is that it is lightweight with small code footprint and therefore suitable for remote applications such as IoT appliances. (MQTT N.d.)

One protocol to mention together with IoT appliances is Advanced Message Queuing Protocol, AMQP. It is based on open standard and its main purpose is to pass business messages between applications. (AMQP N.d.)

5.5 Servers and storages

From the architectural perspective, the servers in the IoT environment have multiple roles. The servers are needed from running application code to managing the necessary IoT applications. The IoT appliance may need various applications for running what it is intended to do. The application server could be either hardware based or a virtual installation on top of any hypervisor application. One key purpose with servers is to manage and handle raw sensor data. The term big data is often mentioned in this context. The big data applications and analysis are described shortly in the following chapter. Managing the updates for various parts in IoT environment is also vital. The IoT environment consists of various hardware modules including their own firmware in each module. Those modules should be up-to-date during the lifecycle of the IoT appliance and therefore, the dedicated update server could be configured in the IoT environment. This same is relevant also for all software modules.

The servers are needed for storing and managing the raw data generated by different kind of sensors. The amount of sensor data could be large; hence, storing the data only to server is not relevant and therefore, storing the data to separate disk storage makes sense. The storage system could be local disk storage based on

fibre channel connectivity, FC, or Network Attach Storage, NAS. One possibility is to use a cloud-based storage system from a well-known cloud vendor.

One key role with server is the user authentication to ensure that only valid users have privilege to access IoT applications and managing the IoT data. There are several choices and best practices for managing user's authentications. One possibility is to manage users and possible security keys locally in the IoT device itself or use a dedicated user management server or application for that purposes. The remote access possibilities and user authentications should also be handled within user management server and applications.

5.6 Big data and analytics application

One major part in IoT architecture is big data and analytics applications. There are currently various vendors that offer the applications for managing a large amount of data and manipulating the data in such a way that it is more human readable. The key point is also to understand whether the amount of data is growing all the time when more IoT appliances and devices are published for various purposes. (Hitachi Vantara Big Data Analytics 2017.)

The amount of data generated by IoT devices is huge. It does not matter if the data is generated by a sensor or surveillance camera or any other IoT hardware device. The huge amount of data is basically useless until it has been analysed and adapted in such way that human can understand it. In the worst case, the generated sensor data is just a basic Ascii text without any formulations. Below, in Figure 6 there is an example of basic sensor data without any formulation:

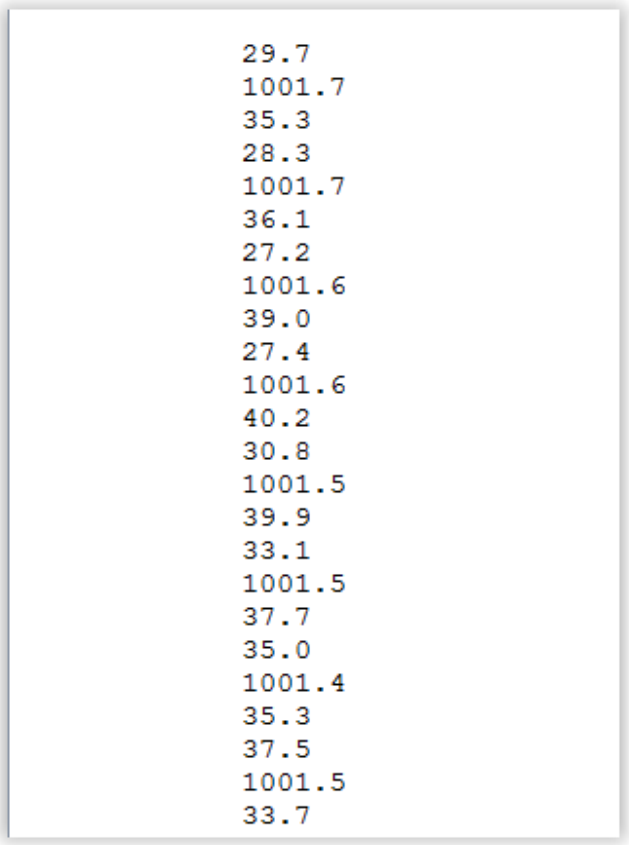


Figure 6. Sensor data, temperature, air pressure and humidity

The above data is generated with Sense Hat temperature and air pressure sensors running on top of Raspberry Pi 4 computer. As can be seen, the above numbers do not illustrate anything; they are just bunch of numbers. To visualize those numbers, the big data and analytics applications are needed.

Today, the application market for big data and analytics tools is huge. Some of the applications only focus on one area such as big data or analytics; however, some vendors provide a complete application suite for covering all big data and analytics needs in one package. The following Figure 7 gives an idea of the availability of applications on the market. (Hitachi Vantara Big Data Analytics 2017).

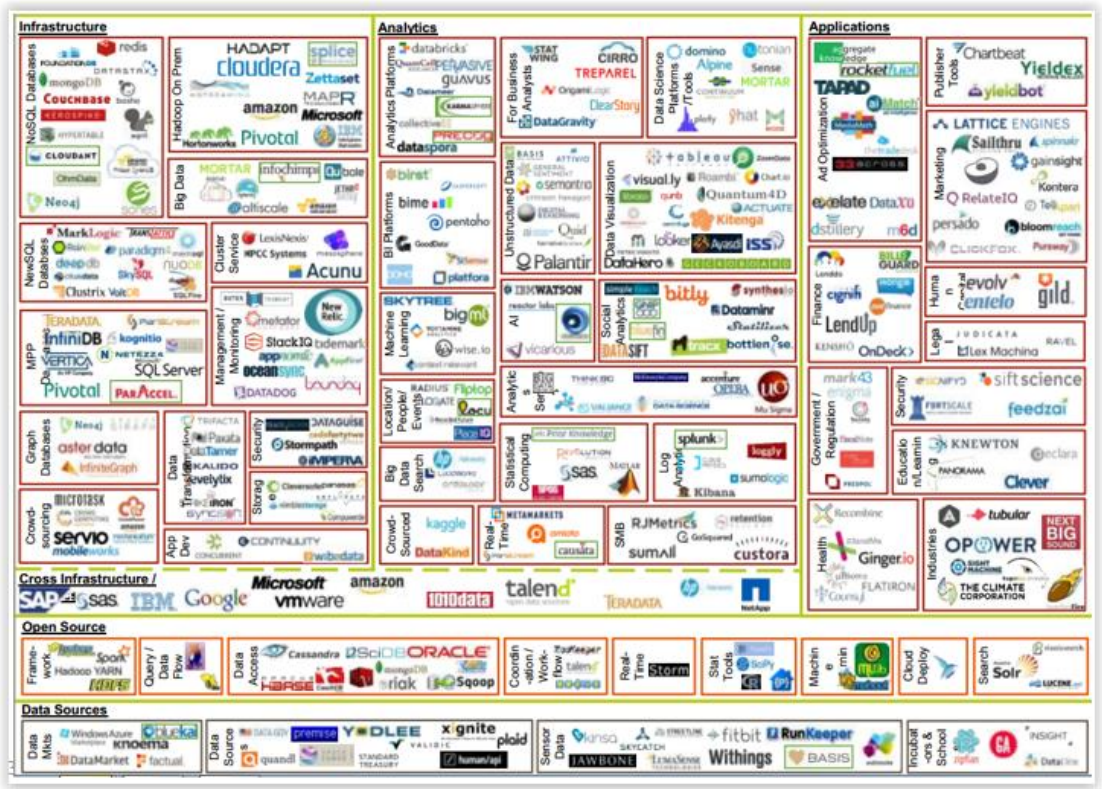


Figure 7. Big data and analytics application in market 2017

The purpose of a big data application is to help users to understand what is in the data, and more importantly, what is relevant in the data to meet the organization's needs. It can be said that data without any analysis is almost useless. One key role with big data tools is also the ability to store the data and keep it in safe. (Hitachi Vantara Big Data Analytics 2017.)

Figure 8 represents a solution for data flow from sensor side up to analytics application including the steps between (Hitachi Vantara Big Data Analytics 2017).

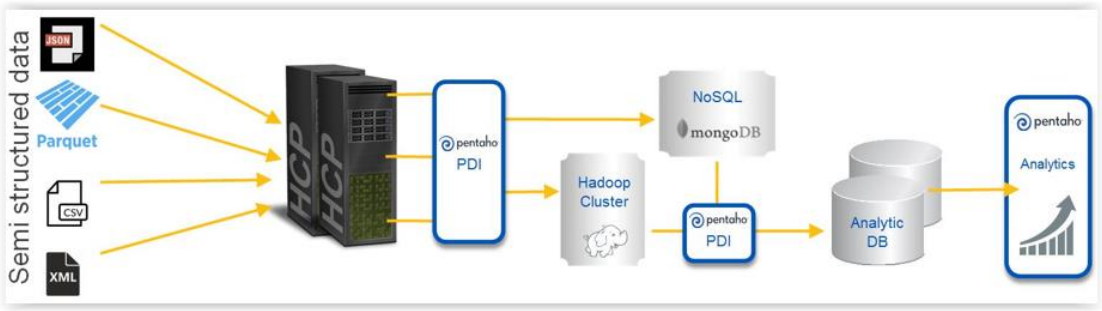


Figure 8. Data flow for sensor data, sample

The example above shows typical data flow from sensors and the application modules needed for data analytics. The sensors and network modules are excluded from this example.

This example is based on Hitachi Vantara's 2017 solution for big data and analytics application suite. The data flow is from left to right. The raw sensor data or any unstructured or semi structured data is collected from various locations to one major location for storing purposes, in this case Hitachi Content Platform, HCP. After collection the data is modified in such a way that it is usable with Pentaho PDI tools. Pentaho PDI transmits the newly formulated data to Hadoop Cluster and for MongoDB database for storing purposes. Pentaho's analytics tools fetch the data and modified and formulate it such a way that it is human readable like charts or graphs. (Hitachi Vantara Big Data Analytics 2017.)

This thesis does not cover any big data and analytics applications.

6 Protecting the IoT environment

6.1 General

When thinking about the Internet of Things environment, the protection against external threats and vulnerabilities must be considered as important as in a normal ICT environment. The reason for that is the vast amount of IoT devices and environments which could be used for building up the bot networks or used for any other hostile activities. (Fruhlinger 2018.)

There are many various public sources available which go through the possible scenarios related to various kinds of vulnerabilities and solutions for choosing the right protection mechanics for these various scenarios. The main sources in this chapter have been the following publications from public internet sources:

1) IoT Security for Dummies by Lawrence Miller, 2016
2) The 8 Biggest IoT Security Mistakes and How to Avoid Them, 2017
3) Definition of Cybersecurity, Gaps and Overlaps in Standardisations V1.0, 2015
4) Open Web Application Security Project, OWASP.

Those various sources have different approaches for cyber security challenges. The first three concentrate mainly on protecting IoT environments from hardware perspective including the drivers and firmware levels as protections mechanics. OWAPS's approach has a different angle compared to other publishers. Their focus is on protecting the IoT environment from a web application's perspective. The other major difference with OWASP is that they do not recommend any commercial applications or services; their approach is to provide guidelines and a cheat sheet for building up a protected IoT environment.

Gartner's publications are one motivator for designing and building up secured IoT environments. According to Gartner (2017), the quantity of the connected " things" will increase year by year by over 30 % and by 2020, the quantity of connected IoT

devices could be as high as 20.4 billion. The number of IoT devices by January 2017 was 8,380,600 (Millions of units) (Gartner 2017). This means that the number of cyber-attacks will grow rapidly against the IoT environment because most of the environment are not protected well enough (Kennedy 2017).

One aspect also related to the protection of the IoT environment against the cyber security threats is to think about an attacker's motivation and reason for invasion. There are hackers invading systems because it is fun, and they are testing their own capabilities against different systems. One group of hackers try to cause harm as much as possible in diverse ways such as creating bot networks of the hacked environment and using them for their own purposes. One group of attackers try to gain economic benefit by stealing valuable information, personal or corporate secrets. Then one group to mention are hackers guided by governmental organizations with the purpose to influence other governments or focus on possible corporate espionage. (Solarwinds msp 2017.)

The privacy and data security are the key questions today when considering the IoT devices on organizational level or in private use. The large-scale theft of information on personal identities or sensitive data from institution or organization is always a substantial risk in wrong hands. To ensure the appropriate level of protection for securing IoT ecosystems, the business organizations must perform a risk analysis. One part of performing a risk analysis is to implement appropriate safeguards. Also, the security management policy must be defined and implemented in internal processes. There is a risk that developers have not perceived the IoT environment as a target for invasions. That they might not consider the possible attack vectors during the development phase related to the environment. Therefore, the implementation of cyber security should be included in organizations procedures and processes to protect the organization's intellectual property. (Salonen 2017; Solarwinds msp 2017.)

6.2 Legislation, standards and guidelines

The IoT as a technology and concept is quite new, and therefore the legislation and standardization are still under development phases. However, some standards already exist and therefore they can provide guidance for designing cyber secure IoT ecosystems. Good examples of current standards and guidelines are:

1) Common Weakness Enumeration (CWE 2018), which describes possible weaknesses in IoT architecture and design
2) Common Attack Pattern Enumeration and Classification (2018) is source of resource for identifying attack methods
3) Computer Security Resource Center (2018) provides information about the United States Government computer security standard for accrediting cryptographic modules
4) Common Criteria (N.d.) provides information for computer security certification development for the smart card industry based on ISO/IEC 15408 International Standard.

There is one very important set of standards related to information technology. ISO/IEC 27001 gives information related to Information Security Management System, ISMS, and the needed requirements. ISO/IEC27002 provides information for Code of practice for information security controls. ISO/IEC 27032 is a standard for "Guidelines for Cybersecurity" which defines common cyber security risks and different controls for the risks. Those standards are provided by International Organization for Standardization (ISO homepage.)-The ISO/IEC27001 and ISO/IEC27001 standards include guidelines and requirements related to log management as well.

Legislation and guidance always differ from country to country. For example, in Finland the Ministry of Defence has published Katakri (2015) document, which is national guidance for auditing cyber security criteria (*Kansallinen turvallisuusauditointikriteeristö*). One important guidance to mention is VAHTI (N.d.), published by the Ministry of Finance. The purpose of this document is to improve

cyber security awareness in government organizations. An important national organization to mention in this section is *Viestintävirasto Kyberturvallisuuskeskus*, which is responsible for publishing cyber security threats and providing guidance. This organization provides service for Finnish public consumers; however, some of their services are only available for state of administration and security-critical organizations.

The legislation related to the intelligence legislation is currently under adoption in Finland. The purpose is to give mandate for authorizations to improve Finnish protection capabilities against serious threats against national security. This new law would improve Finnish Security Intelligent Service's capability to investigate and inquire network traffic in case of serious threats if it is allowed by court. (Ministry of the Interior 2018.)

6.3 Attack vector

The key term related to cyber security and attack is the attack vector. What is attack vector? Attack vector is a method and technique used by an external or internal counterpart with the purpose to assault or exploit the environment, network, compute or another device. Often the purpose is also to gain access to an environment to deliver a payload or malicious outcome. Usually the purposes for those actions are hostile and aim at collecting sensitive information, pecuniary benefit or carry out other harmful activities against the organization. It can be said that attack vector is the path or route to carry out hostile activities against the environment. The vulnerabilities are in key role in attack vectors because these weaknesses are exposed to the risks from the environment. There are plenty of public web sites on the internet which define the attack vectors. Figure 9 below explains the path from threat agent up to business impact very clearly:

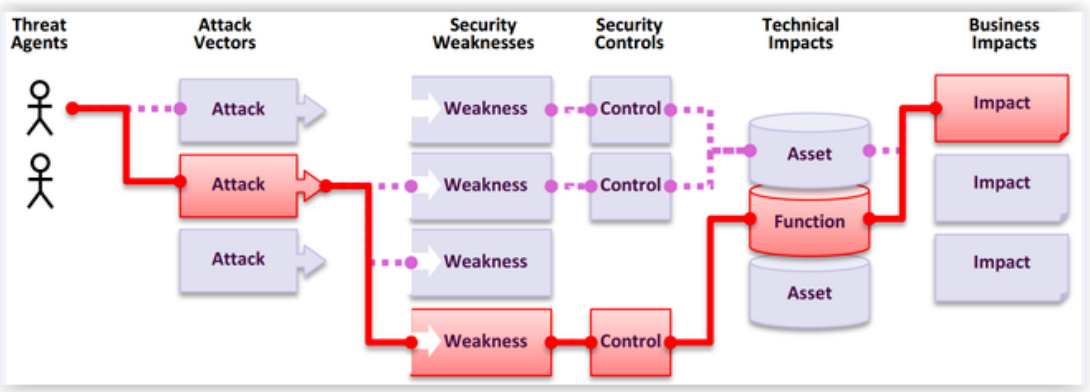


Figure 9. Threat agent (OWASP 2017)

This figure is taken from the web pages of Open Web Application Security, OWASP (2017). It shows clearly the relationship of attack vectors, security weaknesses, security controls, technical impacts and their influence on business. Typical attack vectors could include viruses, email attachment, web pages and pop-up windows; however, the actual list is much longer. Very common malicious payloads are viruses, Trojan horses, worms and different kind of spywares. The common factors with those malicious payloads are that programming skills and knowledge of different applications and operating systems are needed. (Salonen 2017.)

When considering the attack vectors against the environment, one major element related to security is the risk rating methodology related to different attacks and their severity. The following Figure 10 shows one version of risks related to exploitability, weakness prevalence, weakness detectability, technical impact and finally business impact. By using that table, the calculation of risk factors is easier and more illustrative. (OWASP 2017.)

Threat Agents	Exploitability	Weakness Prevalence	Weakness Detectability	Technical Impacts	Business Impacts
App Specific	EASY: 3	WIDESPREAD: 3	EASY: 3	SEVERE: 3	App / Business Specific
	AVERAGE: 2	COMMON: 2	AVERAGE: 2	MODERATE: 2	
	DIFFICULT: 1	UNCOMMON: 1	DIFFICULT: 1	MINOR: 1	

Figure 10. OWASP Risk Rating Methodology (2017)

When considering the attack vectors, it is important to keep in mind that attacks do not only come from the network via email and web application but also the physical attack vectors must be considered as important as attacks via network. (Salonen 2017.)

The following chapters will cover more in depth the key areas related to the vulnerabilities and cyber security recommendations for securing the IoT environment.

6.4 Protection of IoT

The basic question is why the intellectual data generated by IoT devices or IoT assets needs to be protected. One reason is to protect one's own intellectual property against other competitors. Also, the own data and environment need to be protected because that is a way of preventing others to compromise one's own environment and use it for malicious purposes. One important reason for security is the overall security, i.e. the internet comprises interconnected systems and any of its vulnerable parts may be used to harm the other parts of the internet. Therefore, it is extremely important to protect one's own intellectual properties. (Sagedhi 2017.)

One aspect to see how the cyber security is related to an IoT device can presented with the following triangle. The purpose of the triangle is to visualize the necessary steps needed to acquire a higher level of security awareness with IoT devices. (ibid.)

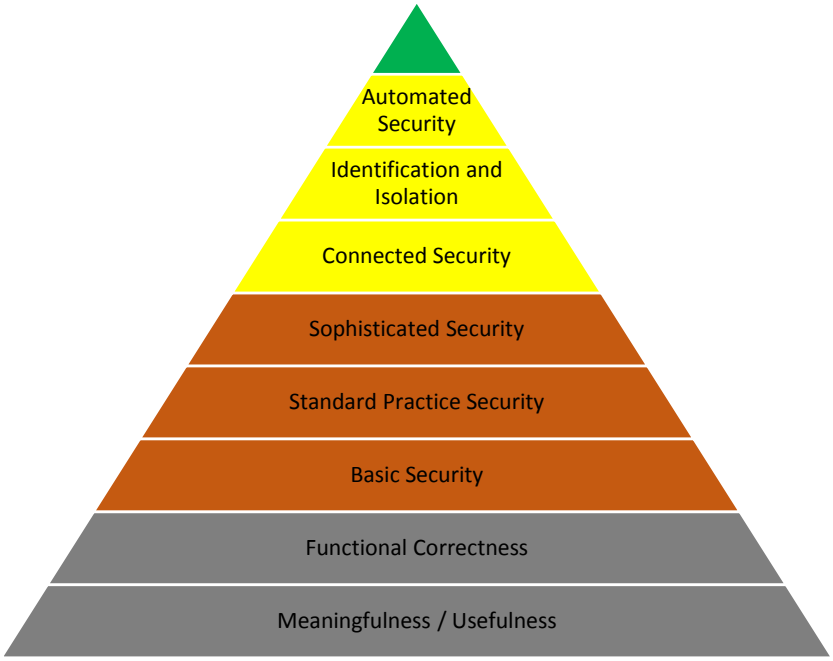


Figure 11. Cyber security awareness triangle (Sagedhi 2017)

The triangle should be observed from the bottom level it and one should go further up step by step with the top green level as target. (ibid.).

The first two grey steps are the starting point on the path to a higher security level. These two levels mean basically a normal internet without any security features and for example, there could duplicate MACs or IP addresses causing security issues. (ibid.)

The next three levels, basic security, standard practice security and sophisticated security are a higher level of security awareness. The protection methods implemented within these three steps are like unique passwords; the latest software and firmware versions, secured communication protocol such as HTTPS securing web

frontends and security architecture. These steps are the first implemented security procedures and therefore they enhance situation awareness. (ibid.)

The next three levels with yellow present a higher level of security implementation in an IoT environment. The overall security awareness is on a new level. The practical implementations with these levels are like swarm attestation, IoT sentinel and the implementation of automatic vulnerability and attack detection. One key element is a log management system, which enhances the knowledge of system usage by collecting and analysing data in the maintenance log, usage log and access log.

The top green triangle is the level where one should strive to whenever it is possible and reasonable as well as economically wise. (ibid.)

When considering the top green triangle, the cost level to achieve that security target should be defined by the organizations, and it should correlate with the value of the intellectual property. If the value of the IoT appliance and generated data is low and the benefit for organization is minor, it is not worth investing a great deal of resources in protection that appliance and data.

6.5 Points of vulnerabilities

When considering the basic IoT environment, Figure 12 below shows pre-digested schematics of possible vulnerable components. The IoT environment consists of many components and layers which possible are targets for attackers. (Miller 2016, 6).

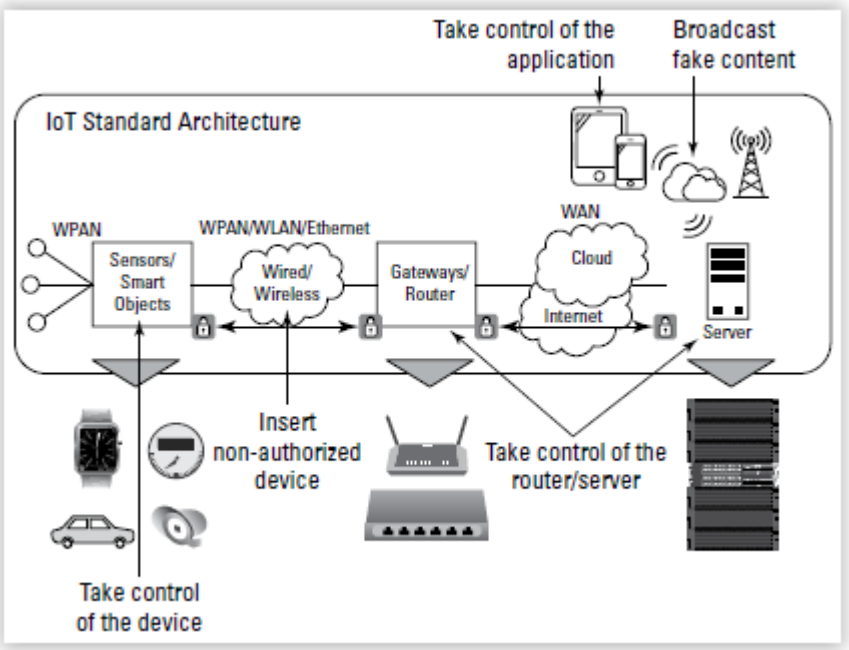


Figure 12. IoT points of vulnerabilities (Miller 2016, 6)

The key point in Figure 12 above is to notify that possible vulnerable points are not only hardware related but also software and application related attack targets. By analysing the IoT environment such as the one above, there is a possibility to define possible attack vectors used by attackers. The following chapter contains more information about the vulnerable points and how to protect against possible external threats.

6.5.1 Universal problems

The environment consists of a basic sensor which collects measurable data. The sensor's data is transmitted to IoT server via physical transmission path either in wireless or wired network. There are also other physical counterparts involved in the transmission path from sensor to server, such as gateways and public internet connections. The software part of IoT environment consists of e.g. user accounts with credentials, applications and operating systems. Finally, the sensor data is stored to mass storages devices.

According to Praetorian (2017), transport layer weaknesses, vulnerability to account compromise and shared default secrets count as universal problems.

Based on their assessment, the problems in transport layer weaknesses were caused by traffic with no encryption enabled for sensitive data, older protocols used and problems with certification and weak ciphers. Those altogether caused plenty of vulnerability risks in IoT environments. (Praetorian 2017.)

The compromise in the account was caused by weak password policy and lack of lockout feature of used accounts. Also, a major part was of security risks was caused by a possibility to enumerate active user accounts. (ibid.)

The last in universal problem by Praetorian (2017) is the shared default secret. That one is very critical in case of a cloud based IoT environment. In practice, it means that any new user might share the same user password and encryption key, which allows possible attackers to cause vulnerability to the complete IoT ecosystem by recovering the secret key from a single device. (Praetorian 2017.)

6.5.2 Device specific concerns

According to Praetorian (2017), the device-specific security concerns are like debug services enabled, missing patches and insecure updates. These items are defined based on assessments.

Debug service enabled means in practice that IoT hardware device has in system board the JTAG debug connector, serial connectors or USB ports enabled and available, where attackers can connect their own device to compromise the hardware. The purposes of those connectors are to provide more debug information for developers in the development phase and in many cases, those connectors and ports are left in ready-made devices. (Praetorian 2017.)

Missing patches are caused by cases where the IoT appliances are built up with certain levels of operating systems and patches, and by the time of production the

working combination is not updated anymore because of a risk to break something. Those patch versions might be months or even years old versions. (ibid.)

The insecure updates present a case where the update process and functionality are compromised. In such a situation where the compromise has occurred once or all following circumstances are true: the device has failed to be identified to update server, the connections are not secured, or signature verification is failed before update. (ibid.)

6.5.3 Warning narratives

Praetorian (2017), defines the warning narrative as follows: inheriting vendor's security and balancing security with user experience.

The typical example of inherited vendor's security is such where the client uses outsourced chip from a third-party vendor, and that chip is not secure enough to store or handle sensitive data. The data can be accessed from chip directly by reading the memory contents address by address. (Praetorian 2017.)

The user experience is also important. The usability of a device should be easy enough that it is meaningful to use. However, on the other side, the security should be at a certain level for the device to be safe enough to use. It is very common that usability goes over security as a result of unsecure devices and appliances. (ibid.)

6.5.4 Cyber security recommendations

When considering the securing recommendations for IoT environment, there are various sources with predefined recommendations and actions for protecting against possible external threat actors.

Based on Praetorian (2017), there are different protecting methods depending on the threats. For protecting and securing the transport layer vulnerabilities, the best

way to protect is to use secured communication protocols such as TLS or SSH. Additionally, using strong ciphers and trusted services secures the transport path between devices. Using ready-made test tools for verification of a secured transport layer is also recommended. One example is a tool made by OWASP, Transport Layer Protection Cheat Sheet (2017).

Protecting against the account compromising there are several steps to perform. First, the password policy should be defined in such a way that password length is set to minimum, complexity is required, and password aging is defined. Additionally, defining user lock out timer in case of failed login attempts is good practice. Good practice is also to prevent the account enumeration for external responses. OWASP also provides cheat sheet instructions for protecting user authentication. (OWASP 2017; Praetorian 2017.)

Protection against shared default secrets is made at the beginning of the deployment process of new installations. The method is to force a user to change and set their own passwords during the login process. One possibility to protection is to allow user or application to generate their own secret key at the beginning. (Praetorian 2017.)

Debugging services are needed for troubleshooting purposes; in particular at the development phase of the IoT appliance. The protection against misuse of those services is quite straightforward; all unnecessary services and ports must be disabled to prevent misuse. The hardware related ports are like JTAG, USB, serial connection and SD card reader, and processes are such as Telnet and SSH. (ibid.)

It is very common that readymade appliances are running with old, outdated levels of patches. That was also noted as one major possible vulnerability by Praetorian. The recommended protection method is to implement an automated updating process. One possible protection method is to build up custom made versions with the required functionality. That will also help with defining the required patch process in products. An important part is also to follow newly published vulnerabilities and then implement the necessary patches to protect the environment. Related to this, it is important to have an up-to-date product inventory which tells the different versions published earlier. (ibid.)

Related to the patch levels and update policies, the insecure updates have been defined as one possible vulnerability. The recommended method is to protect by using certificates to ensure secure connections. Also, the use of TSL ensures that connections between client and server are secured and there is no risk that intruders can hijack a connection such as in the man-in-the-middle attack. The downloadable package should be hashed and signed for protection purposes. It will ensure the integrity of the downloadable package. OWASP provides a cheat sheet to help to ensure the protected method for securing updates. (Praetorian 2017; OWASP 2017.)

The protection for the above cyber security risks is easily accomplished with few common best practices, e.g. regular updates for all software modules related to the IoT appliance, securing the transfer method by using encrypted and secure data path, and ensuring the integrity of downloadable software packages. One important best practice to mention is user management together with password policy. Most of the potential vulnerabilities could be avoided with those listed best practices. (Praetorian 2017.)

7 Case study, cyber security in cloud based IoT environment

7.1 Introduction of the case study

The case study focuses on the cyber security view of cloud-based Internet of Things environment with two sensors and one cooler unit. The environment is built up to JYVSECTEC's Cyber Range, RGCE, a cloud provider value chain case study. The configuration is described in the following chapter, Configuration architecture. This case study focused mainly on cyber security elements and how to make the system more secure. The basic IoT functionality, configuration and relationships of different modules were out of the scope of this case study.

The cyber security level after installation with default settings is not typical enough to give a decent level of security. Normally, all user credentials, e.g. user ids and passwords are default ones which are typically very well-known and searchable from the internet. Quite often the installed software modules are behind the latest patch and secure levels. Also, the used operating systems in IoT environment might be out-of-date what comes to the OS versions.

This part of the master's thesis focuses on a case study of Internet of Things environment, which is produced by RGCE cloud environment. The following chapter consists of the configuration overview of the used environment from hardware and software perspective.

When considering the IoT environment from cyber security point of view, the following key questions are the ones which this master's thesis answers based on the case study:

1) What is the initial level of cyber security within the IoT environment?
2) What is the minimum acceptable level of cyber security in the IoT environment?
3) What is the minimum implementation for improving the cyber security in the IoT environment?

7.2 Scope of the case study

This case study focuses only on cyber security view in JYVSECTEC's Internet of Things environment. The study does not focus on the installation and configuration of the IoT environment. Also, the functionality of the virtual environment, RGCE, is out of the scope of this study. This master's thesis does not cover installation steps and configuration details to build up an IoT environment for customer ready usage. More detailed description about the JYVSECTEC's cyber range environment is found in the online document provided by JYVSECTEC.

This case study also excludes commercial smart home IoT appliances such as surveillance cameras, network devices and smart televisions. Also, the hardening of the operating system in various modules in this environment is out of the scope. However, normal procedures for upgrading patches to operating systems is a part of this thesis.

7.3 Configuration architecture

The environment is built up with cloud based IoT implementation provided by the virtual machines of RGCE. The used IP address allocations are not based on the real public internet IP address space; the used IP addresses are virtual and without real connections to the public internet.

The purpose of the environment is to provide sensor data from two temperature sensors and based on that data, control and adjust one cooler unit.

The high-level overview of the environment from the perspective of software modules is following.

The IoT environment is built over ThingsBoard, which is an open-source IoT platform. The ThingsBoard needs two different basic service modules to handle the IoT environment: Apache Zookeeper, cluster orchestration software module and service coordination; and Apache Cassandra, a database for storing the IoT configuration and application data.

Figure 13 below gives the view of the connected modules and their relations to each other.

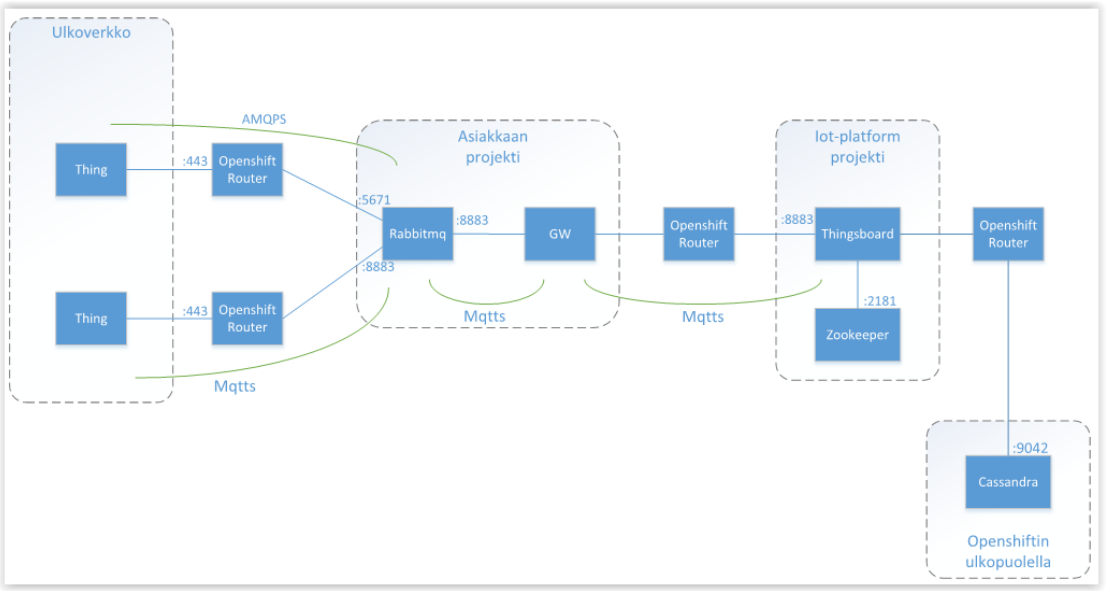


Figure 13. High level architecture view of environment

Figure 14 shows the overview of the ThingsBoard's dashboard view for sensor temperature data.

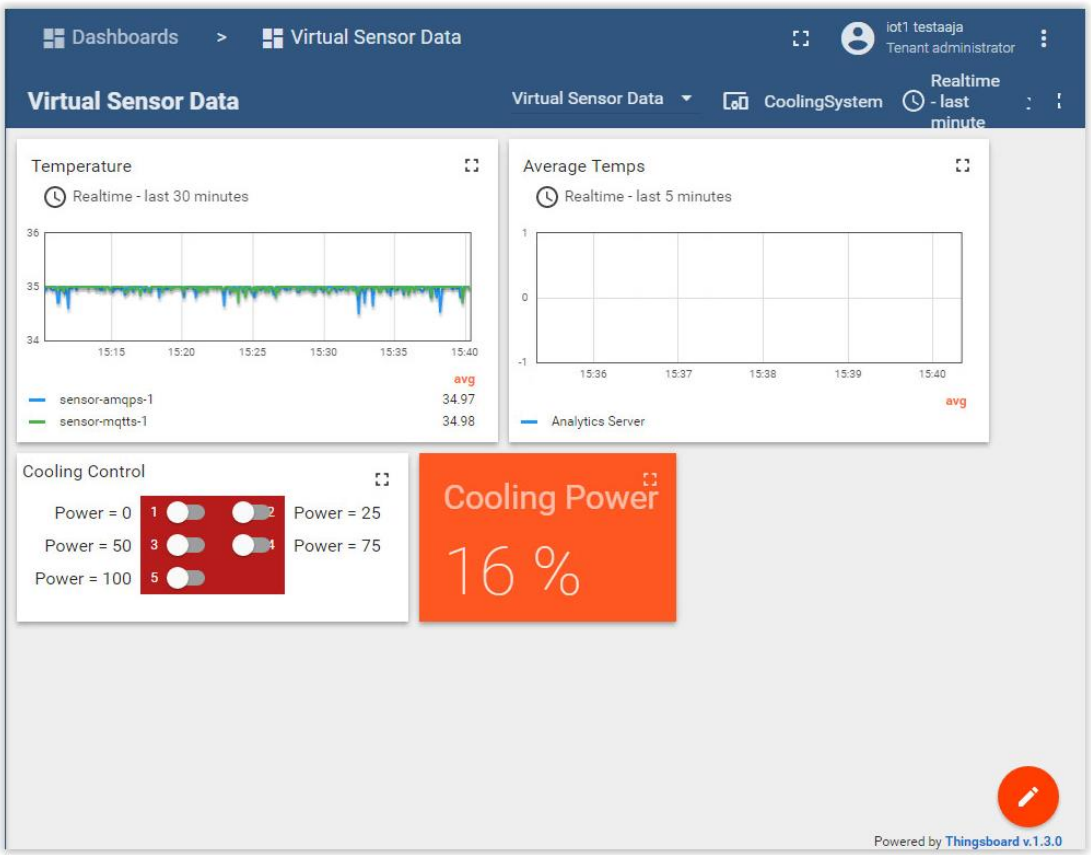


Figure 14. Dashboard view for sensor data by ThingsBoard

7.4 Defining security implementation

When considering the cyber security for the IoT environment, it is necessary to think what possible attack vectors there might be against the environment. Therefore, security administrator and system administrator need to think what the possible vulnerability of the environment could be. The methods for analysing the environment could be similar to the zero-day vulnerabilities analysis (Salonen 2017; Palmers 2013):

1) analyse environment
2) test environment
3) report/document findings
4) design mitigations activities.

The zero-day attack is an attack, that exploits computer appliances before the vulnerability is known or patched against the vulnerable. The quality assurance process like Unknown Vulnerability Management Process is to aid the administrator to identify possible weakness in system before appliance is released to market. The main reason is to identify possible critical points from the vulnerability view and design potential mitigation solution against them. (Salonen 2017; Palmers 2013.)

7.4.1 Attack vectors in case study environment

After considering the attack vectors against the environment, good knowledge of the architecture and data flow is an essential part of design. It should also be kept in mind that physical security is also generally very important part of security.

Typically, the IoT environment consists of physical devices such as sensors, servers and network data paths. If the physical immunity is compromised, there is a considerable risk that the complete IoT appliance is exploited by attackers and, in the worst case, it is used as part of botnet which will affect other appliances and counterparts.

When considering the attack vectors, a very important part are system level attack methods exploiting the flaws in the security design or general implementation.

The following two chapters describe possible attack vectors against this IoT environment; however, this is not a complete outcome of all possible attack scenarios.

7.4.2 Physical based attack vector

Quite often the hardware-based protection against offensives is not considered a substantial risk. The attackers can gain physical access to an IoT device, tamper the configuration information related to the device, reconfigure the device to send malicious data or attach a part of it to the botnet. The risk that attackers break down to the IoT device is also high. Table 2 describes the key elements for a physical attack vector and the table is built up collecting elements from OWASP's and MITRE's web pages.

Table 2. physical attack vector

Attack vector	
Unauthorized physical access to environment	
Security weakness	Physical security of the environment components
Mitigation	Physical installation of environment to separate data center with authorized access only. Place critical components to lockable racks or cabinets and design user access management procedure to access keys.
Technical impact	Feed incorrect measurement data to system, physical damages, uncontrolled change of configuration. Perform malicious firmware or microcode upgrade to IoT device itself.

Business impact	Feed incorrect measurement data to system, physical damages, uncontrolled change of configuration. Monetary loss, reputation loss.

When considering possible physical attack vectors against the case study environment, the probability of risk for a physical attack is rather low; however, it should be kept in mind as a potential risk of attack method.

7.4.3 Application based attack vectors

When considering the possible attack vector against the applications, servers and network, there are not only few attacks vectors against the environment but a considerable number of attack vectors. The security and system administrator should consider all those areas separately and if possible, use readymade cheat sheets to help to protect environment against vulnerabilities.

In this case study, the application-based attack vectors are based on Open Web Application Security Project (OWASP 2017). The reason for using OWASP as a source was because they have covered the most typical attack vectors against web applications. Also, those attack vectors could be relevant when considering possible threats against the IoT environment as such. Therefore, many of those risks are applicable to this case study. There are also two attack vectors not included in OWASP which could be possible attack vectors against this IoT environment. The following attack vectors presented in Table 3 have been considered to be possible threats against this IoT configuration:

Table 3. List of possible attack vectors (OWASP 2017; a part by the author)

Attack vectors
Broken Authentication, OWASP ref A2: 2017

Sensitive data exposure, OWASP ref A3: 2017
Security misconfiguration, OWASP ref A6: 2017
Cross-Site Scripting, OWASP ref A7: 2017
Using component with known vulnerabilities, OWASP ref A9: 2017
Insufficient logging & monitoring, OWASP ref A10: 2017
Manipulating user data in different tenant inside IoT framework
Denial of Service attack against IoT environment
Man-In-the-Middle attack against IoT environment

The following tables describe (4, 5, 6, 7, 8, 9, 10, 11, 12) the key elements for the possible attack vectors. Note that the tables below do not present a complete set of all attack combinations against this environment. The tables are built up by collecting the key elements from OWASP's and MITRE's web pages. (OWASP 2017; Mitre 2018).

Table 4. Broken authentication, (OWASP 2017; Mitre 2018)

Broken authentication, OWASP ref A2: 2017	
Attack vector	Attackers have access to hundreds of millions of valid username and password combinations for credential stuffing, default administrative account lists, automated brute force, and dictionary attack tools. Session management attacks are well understood, particularly in relation to unexpired session tokens.
Security weakness	The prevalence of broken authentication is widespread due to the design and implementation of most identity and access controls. Session management is the bedrock of authentication and access controls and is present in all stateful applications.

	Attackers can detect broken authentication using manual means and exploit them using automated tools with password lists and dictionary attacks.
Prevention, and mitigation measures	1) When possible implement multifactor authentication to prevent automated, credential stuffing, brute force and stolen credential re-use attacks. 2) Do not ship or deploy with any default credentials particularly for admin users. 3) Implement weak-password checks 4) Align password length, complexity and rotation policies with modern, evidence based password policies. 5) Ensure registration, credential recovery, and API pathways are hardened against account enumeration attacks by using the same messages for all outcomes. 6) Limit or increasingly delay failed login attempts. Log all failures and alert administrators when credential stuffing, brute force, or other attacks are detected. 7) Use a server-side, secure, built-in session manager that generates a new random session ID with high entropy after login. Session IDs should not be in the URL, be securely stored and invalidated after logout, idle, and absolute timeouts.
Technical impact	Attackers have to gain access to only a few accounts, or just one admin account to compromise the system.
Business impact	Depending on the domain of the application, this may allow money laundering, social security fraud, and identity theft, or disclose legally protected highly sensitive information.

The above scenario is possible threat because the used applications are build up the cloud based IoT environment. Quite often vendors for such applications have listed their default user ids and passwords in their product web pages. Therefore, this kind of attack vector is relevant in any environment using commonly available applications.

Table 5. Sensitive data exposure, (OWASP 2017; Mitre 2018)

Sensitive data exposure, OWASP ref A3: 2017	
Attack vector	Rather than directly attacking crypto, attackers steal keys, execute man-in-the-middle attacks, or steal clear text data off the server, while in transit, or from the user's client, e.g. browser. A manual attack is generally required. Previously retrieved password databases could be brute forced by Graphics Processing Units (GPUs).
Security weakness	Over the last few years, this has been the most common impactful attack. The most common flaw is simply not encrypting sensitive data. When crypto is employed, weak key generation and management, and weak algorithm, protocol and cipher usage is common, particularly for weak password hashing storage techniques. For data in transit, server-side weaknesses are mainly easy to detect, but hard for data at rest.
Prevention, and mitigation measures	1) Classify data processed, stored or transmitted by an application. Identify which data is sensitive according to privacy laws, regulatory requirements, or business needs. 2) Apply controls as per the classification. 3) Don't store sensitive data unnecessarily. Discard it as soon as possible or use PCI DSS compliant tokenization or even truncation. Data that is not retained cannot be stolen. 4) Make sure to encrypt all sensitive data at rest. 5) Ensure up-to-date and strong standard algorithms, protocols, and keys are in place; use proper key management. 6) Encrypt all data in transit with secure protocols such as TLS with perfect forward secrecy (PFS) ciphers, cipher prioritization by the server, and secure parameters. Enforce encryption using directives like HTTP Strict Transport Security (HSTS). 7) Disable caching for response that contain sensitive data. 8) Store passwords using strong adaptive and salted hashing functions with a work factor (delay factor), such as Argon2, scrypt, bcrypt or PBKDF2. 9) Verify independently the effectiveness of configuration and settings.
Technical impact	Failure frequently compromises all data that should have been protected.

Business impact	Typically, this information includes sensitive personal information (PII) data such as health records, credentials, personal data, and credit cards, which often require protection as defined by laws or regulations such as the EU GDPR or local privacy laws.

This attack vector is related to broken authentication, because both are related to weak password management. Therefore, the delivery method of sensitive data like a password or business secrets must be protected in proper way using SSL or TSL protocols. Also, valuable information should be encrypted with salted hashing functions.

Table 6. Security misconfiguration, (OWASP 2017; Mitre 2018)

Security misconfiguration, OWASP ref A6: 2017	
Attack vector	Attackers will often attempt to exploit unpatched flaws or access default accounts, unused pages, unprotected files and directories, etc to gain unauthorized access or knowledge of the system.
Security weakness	Security misconfiguration can happen at any level of an application stack, including the network services, platform, web server, application server, database, frameworks, custom code, and pre-installed virtual machines, containers, or storage. Automated scanners are useful for detecting misconfigurations, use of default accounts or configurations, unnecessary services, legacy options, etc.
Prevention, and mitigation measures	Secure installation processes should be implemented, including: 1) A repeatable hardening process that makes it fast and easy to deploy another environment that is properly locked down. Development, QA, and production environments should all be configured identically, with different credentials used in each

	environment. This process should be automated to minimize the effort required to setup a new secure environment. 2) A minimal platform without any unnecessary features, components, documentation, and samples. Remove or do not install unused features and frameworks. 3) A task to review and update the configurations appropriate to all security notes, updates and patches as part of the patch management process (see A9:2017-Using Components with Known Vulnerabilities). In particular, review cloud storage permissions (e.g. S3 bucket permissions). 4) A segmented application architecture that provides effective, secure separation between components or tenants, with segmentation, containerization, or cloud security groups (ACLs). 5) Sending security directives to clients, e.g. Security Headers. 6) An automated process to verify the effectiveness of the configurations and settings in all environments.
Technical impact	Such flaws frequently give attackers unauthorized access to some system data or functionality. Occasionally, such flaws result in a complete system compromise.
Business impact	The business impact depends on the protection needs of the application and data.

This attack vector is also relevant for the case study environment. The fact is that there could be an error in the configuration of software modules or transport protocols, which will leave the security weakness in the environment available for security flaws.

Table 7. Cross-site Scripting, (OWASP 2017; Mitre 2018)

Cross-Site Scripting, OWASP ref A7: 2017	
Attack vector	Automated tools can detect and exploit all three forms of XSS, and there are freely available exploitation frameworks.

Security weakness	XSS is the second most prevalent issue in the OWASP Top 10 and is found in around two thirds of all applications. Automated tools can find some XSS problems automatically, particularly in mature technologies such as PHP, J2EE / JSP, and ASP.NET.
Prevention, and mitigation measures	Preventing XSS requires separation of untrusted data from active browser content. This can be achieved by: 1) Using frameworks that automatically escape XSS by design, such as the latest Ruby on Rails, React JS. Learn the limitations of each framework's XSS protection and appropriately handle the use cases which are not covered. 2) Escaping untrusted HTTP request data based on the context in the HTML output (body, attribute, JavaScript, CSS, or URL) will resolve Reflected and Stored XSS vulnerabilities. The OWASP Cheat Sheet 'XSS Prevention' has details on the required data escaping techniques. 3) Applying context-sensitive encoding when modifying the browser document on the client-side acts against DOM XSS. When this cannot be avoided, similar context sensitive escaping techniques can be applied to browser APIs as described in the OWASP Cheat Sheet 'DOM based XSS Prevention'. 4) Enabling a <u>Content Security Policy (CSP) as a defense-in-depth mitigating control against XSS. It is effective if no other vulnerabilities exist that would allow placing malicious code via local file includes (e.g. path traversal overwrites or vulnerable libraries from permitted content delivery networks).
Technical impact	The impact of XSS is moderate for reflected and DOM XSS, and severe for stored XSS, with remote code execution on the victim's browser, such as stealing credentials, sessions, or delivering malware to the victim.
Business impact	Stealing user credentials, sessions or delivering malware to the victim.

The management software modules in the case study environment run on top of web browsers. There is a risk that using an unsecure HTTP protocol enables this possible vulnerability.

Table 8. Using component with known vulnerabilities, (OWASP 2017; Mitre 2018)

Using component with known vulnerabilities, OWASP ref A9: 2017	
Attack vector	While it is easy to find already-written exploits for many known vulnerabilities, other vulnerabilities require concentrated effort to develop a custom exploit.
Security weakness	Prevalence of this issue is very widespread. Component-heavy development patterns can lead to development teams not even understanding which components they use in their application or API, much less keeping them up to date. Some scanners such as retire.js help in detection, but determining exploitability requires additional effort.
Prevention, and mitigation measures	There should be a patch management process in place to: 1) Remove unused dependencies, unnecessary features, components, files, and documentation. 2) Continuously inventory the versions of both client-side and server-side components (e.g. frameworks, libraries) and their dependencies using tools like versions, DependencyCheck, retire.js, etc. Continuously monitor sources like CVE and NVD for vulnerabilities in the components. Use software composition analysis tools to automate the process. Subscribe to email alerts for security vulnerabilities related to components you use. 3) Only obtain components from official sources over secure links. Prefer signed packages to reduce the chance of including a modified, malicious component. 4) Monitor for libraries and components that are unmaintained or do not create security patches for older versions. If patching is not possible, consider deploying a virtual patch to monitor, detect, or protect against the discovered issue. Every organization must ensure that there is an ongoing plan for monitoring, triaging, and applying updates or configuration changes for the lifetime of the application or portfolio.
Technical impact	While some known vulnerabilities lead to only minor impacts, some of the largest breaches to date have relied on exploiting known vulnerabilities in components.

Business impact	Depending on the assets you are protecting, perhaps this risk should be at the top of the list.

The above attack vector A9 is one of the most critical attack vector to protect against. If the administrator does not know the used versions of the components, it generates considerable risk for application vulnerabilities. This includes all components such as operating system, web applications, database application, application programming interfaces (API) and possible libraries related to appliance. Typically, any out-of-date component generates a possible attack path for intruders. This is a potential risk in the case study environment.

Table 9. Insufficient logging & monitoring, (OWASP 2017; Mitre 2018)

Insufficient logging & monitoring, OWASP ref A10: 2017	
Attack vector	Exploitation of insufficient logging and monitoring is the bedrock of nearly every major incident. Attackers rely on the lack of monitoring and timely response to achieve their goals without being detected.
Security weakness	This issue is included in the Top 10 based on an industry survey. One strategy for determining if you have sufficient monitoring is to examine the logs following penetration testing. The testers' actions should be recorded sufficiently to understand what damages they may have inflicted.
Prevention, and mitigation measures	As per the risk of the data stored or processed by the application: 1) Ensure all login, access control failures, and server-side input validation failures can be logged with sufficient user context to identify suspicious or malicious accounts, and held for sufficient time to allow delayed forensic analysis. 2) Ensure that logs are generated in a format that can be easily consumed by a centralized log management solutions.

	3) Ensure high-value transactions have an audit trail with integrity controls to prevent tampering or deletion, such as append-only database tables or similar. 4) Establish effective monitoring and alerting such that suspicious activities are detected and responded to in a timely fashion. 5) Establish or adopt an incident response and recovery plan, such as NIST 800-61 rev 2 or later. There are commercial and open source application protection frameworks such as OWASP AppSensor, web application firewalls such as ModSecurity with the OWASP ModSecurity Core Rule Set, and log correlation software with custom dashboards and alerting.
Technical impact	Most successful attacks start with vulnerability probing. Allowing such probes to continue can raise the likelihood of successful exploit to nearly 100%.
Business impact	In 2016, identifying a breach took an average of 191 days – plenty of time for damage to be inflicted.

To protect against the above attack vector A10, the solution is to log and monitor the environment. The good practice of logging and monitoring environment does not protect against hostile offensives; however, it will give to the administrators' sufficient time frame to react to a possible attack and start necessary defence measurements. Also, predefined alerting threshold values, e.g. network traffic, give time to response in timely manner.

Table 10. Manipulating user data in different tenant

Manipulating user data in different tenant inside IoT framework	
Attack vector	Customer can manipulate sensitive information in different tenant inside the IoT framework.
Security weakness	This could be similar assumption like OWASP ref A2:2017 were the user credentials are compromised. Therefore, customer might be able to see confidential data from guest tenant.
Prevention, and mitigation measures	Protection would be to change default password's right after the system is configured. Also creating additional user credentials for daily tasks without administration right could be relevant task. And if possible to keep number of users with administration rights as low as possible.
Technical impact	Most successful attacks start with vulnerability probing. Allowing such probes to continue can raise the likelihood of successful exploit to nearly 100 %.
Business impact	Compromising sensitive data in IoT platform lead losses of reputation. The loss of reputation always leads to financial losses.

The attack vector above is a scenario where other IoT customer, tenant, can manipulate and compromising another customer's data, configuration and sensitive customer data. The above could be possible in such circumstances where a malicious customer starts scanning a network by using the configuration information based on their own environment. Typically, in such environment from service provider the basic settings are normally standardized and therefore, basic user IDs and other settings are easily feasible. The mitigation in such case is basically in the customer's own responsibility by changing the default credentials to fulfil standard for password policy.

Table 11. Denial of Service attack

Denial of Service attack against IoT environment	
Attack vector	Attack is focused on making the site, application or service unavailable for the purpose it was designed for legitimate users by generating network or computing load.
Security weakness	The possible areas for security weakness are TCP protocol retransmission timeout, servers overload with flooding memory or another component in environment.
Prevention, and mitigation measures	There are no exact tools to protect against DoS. Monitoring and logging mechanism have significant role to preventing DoS attacks.
Technical impact	Most successful attacks start with vulnerability probing. Allowing such probes to continue can raise the likelihood of successful exploit to nearly 100 %.
Business impact	The customer services could unavailable for legitimate users. That will cause monetary and reputation loses, also potential problems to production if attack target used as production management

The Denial-of-Service attack, DoS, could be a relevant attack against this case study environment. Possible targets for DoS attack could be application servers, sensors and the cooler unit. The Dos attack against application servers with considerable number of requests could cause the application to cease from normal operation, which could lead to a situation where temperature sensor data is queued or postponed, and the cooler unit is out of control. Also, one possible scenario could be that the cooler unit is not any more capable of performing its tasks, which would lead to overheating of the environment.

Table 12 shows the key elements for the Man-In-the-Middle attack vector. Also, possible prevention and mitigation solutions are listed in table.

Table 12. Man-In-the-Middle

Man-In-the-Middle attack against IoT environment, MitM (Infosec guide 2017.)	
Attack vector	The malicious counterpart is listening legitimate communication between different part of the organization. Attack purpose is to manipulate or steal the data so that legitimate user does not know or realize that data is manipulated or compromised. One possible aspect is also that Man-In-Middle attack purpose is to only steal sensitive data.
Security weakness	The MitM attack could be done via following methods and vulnerabilities: - Address resolution Protocol ARP cache poisoning - SSL and TSL hijacking - Domain name server, DNS, Spoofing
Prevention, and mitigation measures	The mitigation method for above weaknesses are following: adding static ARP table prevents attacker from using the ARP requests, implementing the HTTP Strict Transport Security, HSTS, allows user only access through SSL or TSL. Mitigation against the DNS spoofing is difficult one because it is hard to detect. By clearing the DNS cache will help and there is available application for Microsoft based servers for tightening DNS security, Domain Name Security System Extension, DNSSEC.
Technical impact	The realization of man-in-middle attack needs technical skills to perform. There should be also access to network for implementing necessary vulnerability modules for performing the MitM attack.
Business impact	The data which is transfer from sensor devices to servers could be compromised and therefore the legitimate use of data is out of date.

The Man-In-The-Middle attack is a likely threat in the IoT environment. Quite often there is a situation that after initial setup the IoT device, consumer devices, is left out of any monitoring or security enhancement, which provides a possible path for Man-In-The-Middle attacks. When considering the case study environment, the MitM

attack is possible, yet, quite hard to implement. The target with MitM attack would be to manipulate the data between sensors and analysing servers. Additionally, stealing the data would be the target in general.

7.5 Information gathering for case study environment

The basic scanning against the network devices in IoT environment was performed by using the Kali Linux servers with default applications included in Kali distribution. The IoT environment was configured with default settings.

The scanning of the environment started with network scanning for all possible devices in the network and it was run with Network Mapper, NMAP, application, with the following parameters as shown in Figure 15:

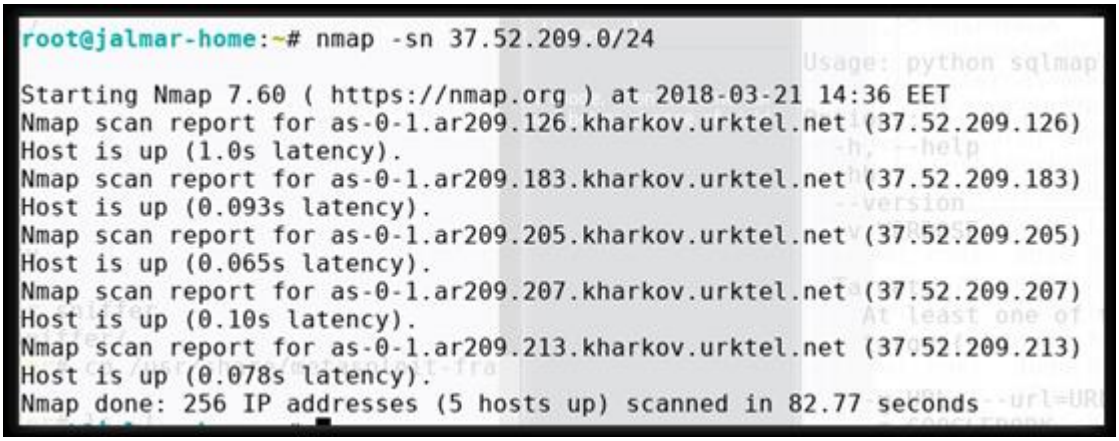


Figure 15. NMAP scan against sensors

The results indicate that there were three IoT devices within that network segment, with addresses 37.52.209.205, 37.52.209.207 and 37.52.209.213 even though the NMAP was able to find altogether five devices.

The application server's names were known. Therefore, the first scan was performed with DNS names to get an overview of the IP addresses. Additionally, the purpose was to collect information about open ports and running services in the servers. Figure 16 shows the results of the scan.

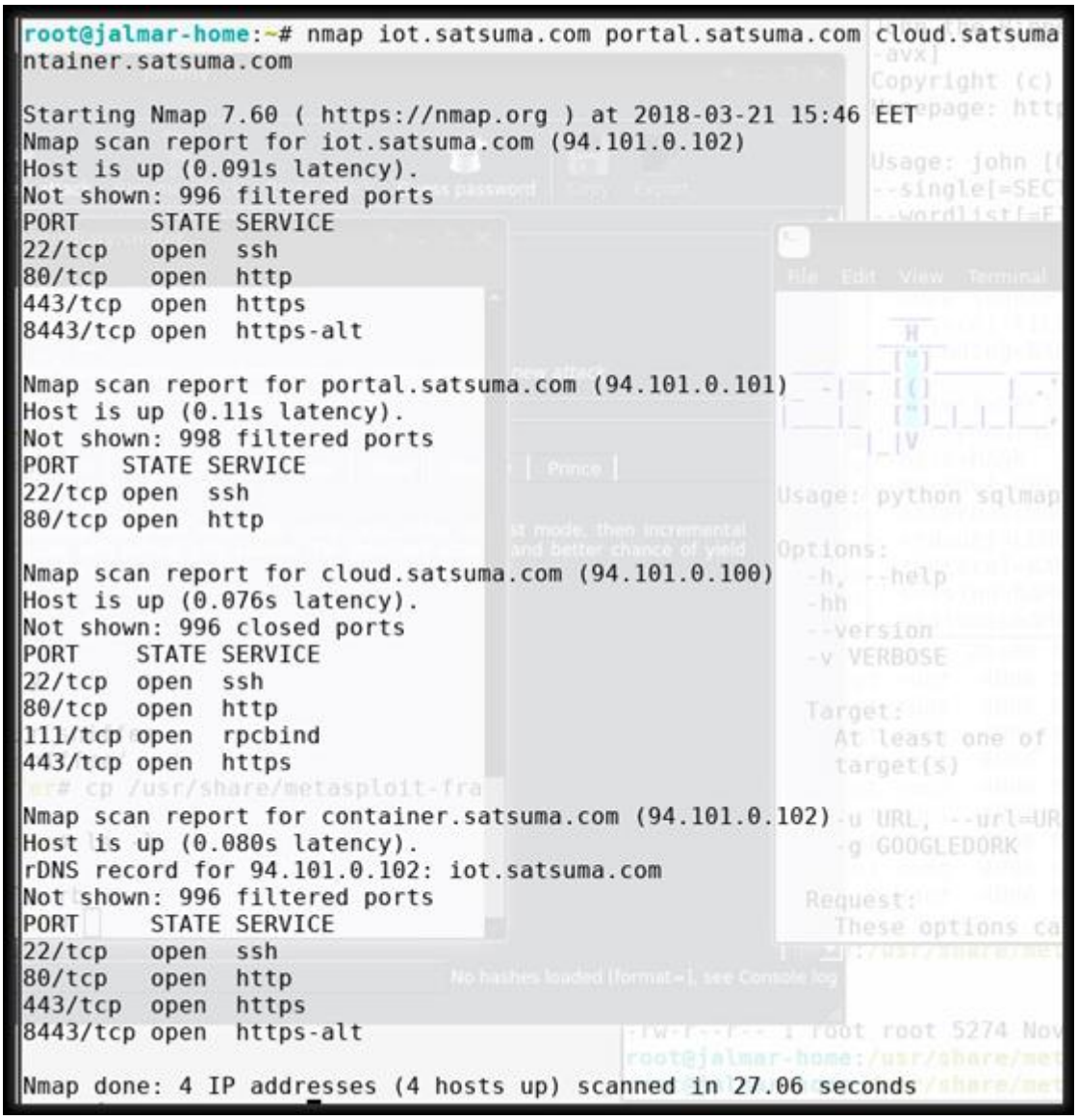


Figure 16. NMAP scan against application servers

Based on the above NMAP scanning, the detected IP addresses and hosts names are listed in Table 13 for this IoT environment.

Table 13. Name and IP address correlation

Device	**IP address**
iot.satsuma.com	94.101.0.102
portal.satsuma.com	94.101.0.101
cloud.satsuma.com	94.101.0.100
container.satsuma.com	94.101.0.102
Sensor AMQP	37.52.209.205
Sensor MQTT	37.52.209.207
Cooler MQTT	37.52.209.213

The names for application servers were already known and therefore the scanning against them was performed with basic syntax to get more information from them.

The second NMAP run against detected devices was performed with following syntax to get more information from the devices itself:

nmap -v -sS -sV -sU -A -oN /root/k3020/<scan target> <ip>

With those options, NMAP was able to scan more information from the devices, e.g. operating system version, open services and their versions. Figure 17 summarizes the NMAP findings; the complete outputs are to be found in the Appendices part.

The used operating systems and applications and their versions were following:

Targets	Address	Up	Port	State	Service version	OS
Cooler MQTT	3	Yes	22/tcp	open	ssh, Openssh 6.7p1 Debian 5	Linux 3.2 - 4.8
			111/tcp	open	rpcbind 2.4	
			68/udp	open	dhcpd	
			111/udp	open	rpcbind 2.4	
Sensors AMQP	5	Yes	22/tcp	open	ssh, Openssh 6.7p1 Debian 5	Linux 3.2 - 4.8
			111/tcp	open	rpcbind 2.4	
			68/udp	open	dhcpd	
			111/udp	open	rpcbind 2.4	
Sensors MQTT	7	Yes	22/tcp	open	ssh, Openssh 6.7p1 Debian 5	Linux 3.2 - 4.8
			111/tcp	open	rpcbind 2.4	
			68/udp	open	dhcpd	
			111/udp	open	rpcbind 2.4	
cloud.satsuma.com	94.101.0.100	Yes	22/tcp	open	ssh, Openssh 6.6.1, protocol 2.0	Linux 4.4
			80/tcp	open	http, Apache httpd 2.4.6, OpenSSL	
			111/tcp	open	rpcbind 2.4	
			443/tcp	open	ssl/https Apache httpd 2.4.6, OpenSSL	
			111/udp	open	rpcbind 2.4	
portal.satsuma.com	94.101.0.101	Yes	22/tcp	open	ssh, Openssh 6.6.1, protocol 2.0	Linux 4.4
			80/tcp	open	ssl/http Werkzeug/0.9.1 Python/2.7.5	
iot.satsuma.com	94.101.0.102	Yes	22/tcp	open	ssh, Openssh 6.6.1, protocol 2.0	Linux 4.4
			80/tcp	open	http proxy HAProxy http proxy 1.3.1	
			443/tcp	open	ssl/https proxy HAProxy http proxy 1.3.1	
			8443/tcp	open	ssl/https Golang net/http server	
Container.satsuma.co	94.101.0.102	yes	22/tcp	open	ssh, Openssh 6.6.1, protocol 2.0	Linux 4.4
			80/tcp	open	http proxy HAProxy http proxy 1.3.1	
			443/tcp	open	ssl/https proxy HAProxy http proxy 1.3.1	
			8443/tcp	open	ssl/https Golang net/http server	

Figure 17. NMAP detailed findings

The other main used tool was Nessus scanner installed in Kali Linux distribution. The purpose of using Nessus was to identify possible vulnerabilities, configuration issues and malware, which might be possible threats for the environment.

Nessus scanner allows the user to select a pre-defined network and device scanner options such as “Host Discovery”, “Basic Network Scan” and “Advanced Scan”. The selected scan type in this case was “Advanced Scan”. Advanced Scan allows the user to select different kinds of plugins to perform wide scans against the selected host names or IP addresses even though by default the major part of plugins is pre-selected.

Nessus scan results below were run against the IP addresses in the environment as an Advanced scan. The detailed overview for each device is seen in Figure 18 with the outputs.

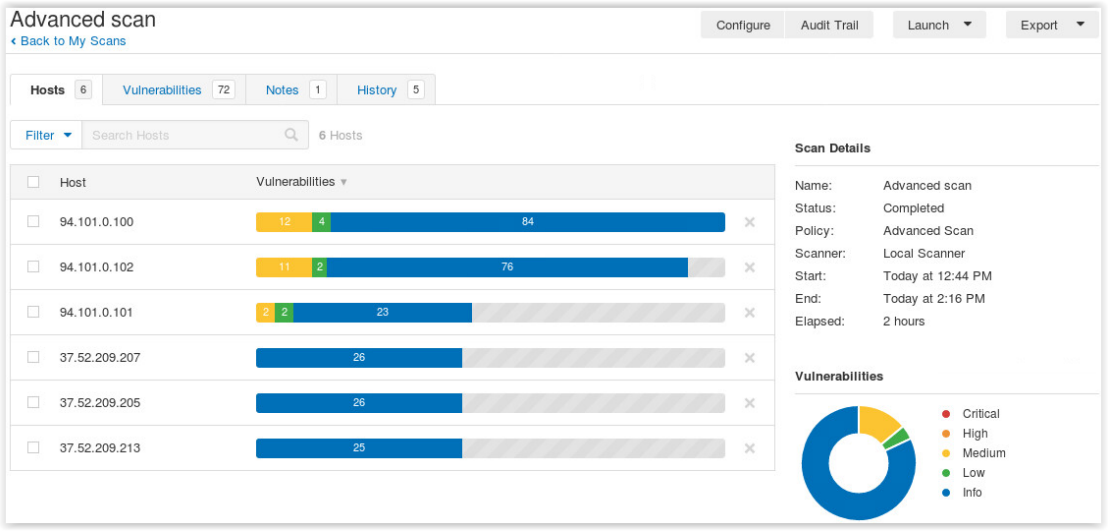


Figure 18. Nessus Advanced scan against devices

The Nessus Advanced scan gave interesting results from the environment.

The first perception was that Nessus Advanced scan was able to detect several “Medium” level classified vulnerabilities from the application hosts. These vulnerabilities need a more specified investigation. Secondly, Nessus Advanced scan was able to identify many “Info” classified type of vulnerabilities from the application hosts.

The sensors and cooler side findings with Nessus were only “Info” classified type of vulnerabilities. Vulnerabilities with category “Info” typically do not need any further analyses. The vulnerabilities with category “Medium” should be analysed more specifically because they are potential risks for offensives by vulnerabilities.

The detailed Nessus findings are seen below for each device scanned in network. After each figure there is a summary with more explanations about the findings. The recommended actions and tasks are listed in next chapter.

94.101.0.100

0	0	8	3	49
CRITICAL	HIGH	MEDIUM	LOW	INFO

Vulnerabilities — Total: 60

SEVERITY	CVSS	PLUGIN	NAME
MEDIUM	6.4	51192	SSL Certificate Cannot Be Trusted
MEDIUM	6.4	57582	SSL Self-Signed Certificate
MEDIUM	5.0	11213	HTTP TRACE / TRACK Methods Allowed
MEDIUM	5.0	20007	SSL Version 2 and 3 Protocol Detection
MEDIUM	5.0	40984	Browsable Web Directories
MEDIUM	5.0	42873	SSL Medium Strength Cipher Suites Supported
MEDIUM	4.3	85582	Web Application Potentially Vulnerable to Clickjacking
MEDIUM	4.3	90317	SSH Weak Algorithms Supported
LOW	2.6	65821	SSL RC4 Cipher Suites Supported (Bar Mitzvah)
LOW	2.6	70658	SSH Server CBC Mode Ciphers Enabled
LOW	2.6	71049	SSH Weak MAC Algorithms Enabled
INFO	N/A	10107	HTTP Server Type and Version
INFO	N/A	10114	ICMP Timestamp Request Remote Date Disclosure
INFO	N/A	10223	RPC portmapper Service Detection
INFO	N/A	10267	SSH Server Type and Version Information
INFO	N/A	10287	Traceroute Information
INFO	N/A	10335	Nessus TCP scanner
INFO	N/A	10386	Web Server No 404 Error Code Check
INFO	N/A	10662	Web mirroring

94.101.0.100 10

Figure 19. Cloud.satsuma.com Nessus Advanced scan result

Figure 19 presents the scan results for cloud.satsuma.com application server. As can be seen, there are eight vulnerabilities with category “Medium” which must be analysed more in detail. Five of those vulnerabilities are related to the SSL and one for HTTP and browsable web directories. The PLUGIN 85582 is vulnerable for Clickjacking.

94.101.0.101

0	0	2	2	21
CRITICAL	HIGH	MEDIUM	LOW	INFO

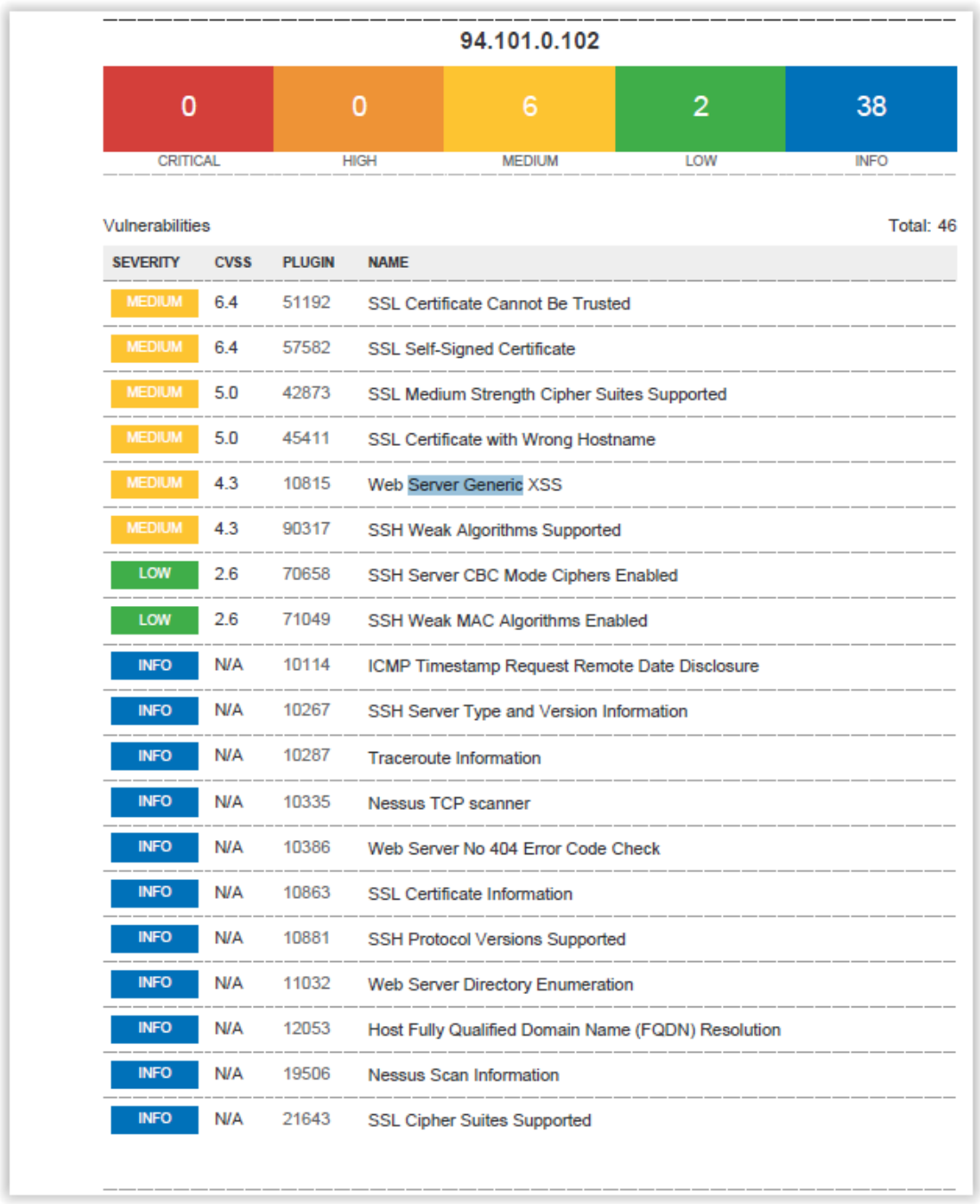
Vulnerabilities — Total: 25

SEVERITY	CVSS	PLUGIN	NAME
MEDIUM	4.3	85582	Web Application Potentially Vulnerable to Clickjacking
MEDIUM	4.3	90317	SSH Weak Algorithms Supported
LOW	2.6	70658	SSH Server CBC Mode Ciphers Enabled
LOW	2.6	71049	SSH Weak MAC Algorithms Enabled
INFO	N/A	10107	HTTP Server Type and Version
INFO	N/A	10114	ICMP Timestamp Request Remote Date Disclosure
INFO	N/A	10267	SSH Server Type and Version Information
INFO	N/A	10287	Traceroute Information
INFO	N/A	10335	Nessus TCP scanner
INFO	N/A	10662	Web mirroring
INFO	N/A	10881	SSH Protocol Versions Supported
INFO	N/A	12053	Host Fully Qualified Domain Name (FQDN) Resolution
INFO	N/A	19506	Nessus Scan Information
INFO	N/A	22964	Service Detection
INFO	N/A	24260	HyperText Transfer Protocol (HTTP) Information
INFO	N/A	25220	TCP/IP Timestamps Supported
INFO	N/A	33817	CGI Generic Tests Load Estimation (all tests)
INFO	N/A	34277	Nessus UDP Scanner
INFO	N/A	39520	Backported Security Patch Detection (SSH)

94.101.0.101 13

Figure 20. Portal.satsuma.com Nessus Advanced scan result

Figure 20 presents the scan results for portal.satsuma.com server. Nessus has detected the same possible vulnerability as with the cloud.satsuma.com, the Web Application Vulnerable to Clickjacking. One SSL based vulnerability was masked as medium, PLUGIN 90317. The rest of the findings for this server is more information type of data.



94.101.0.102

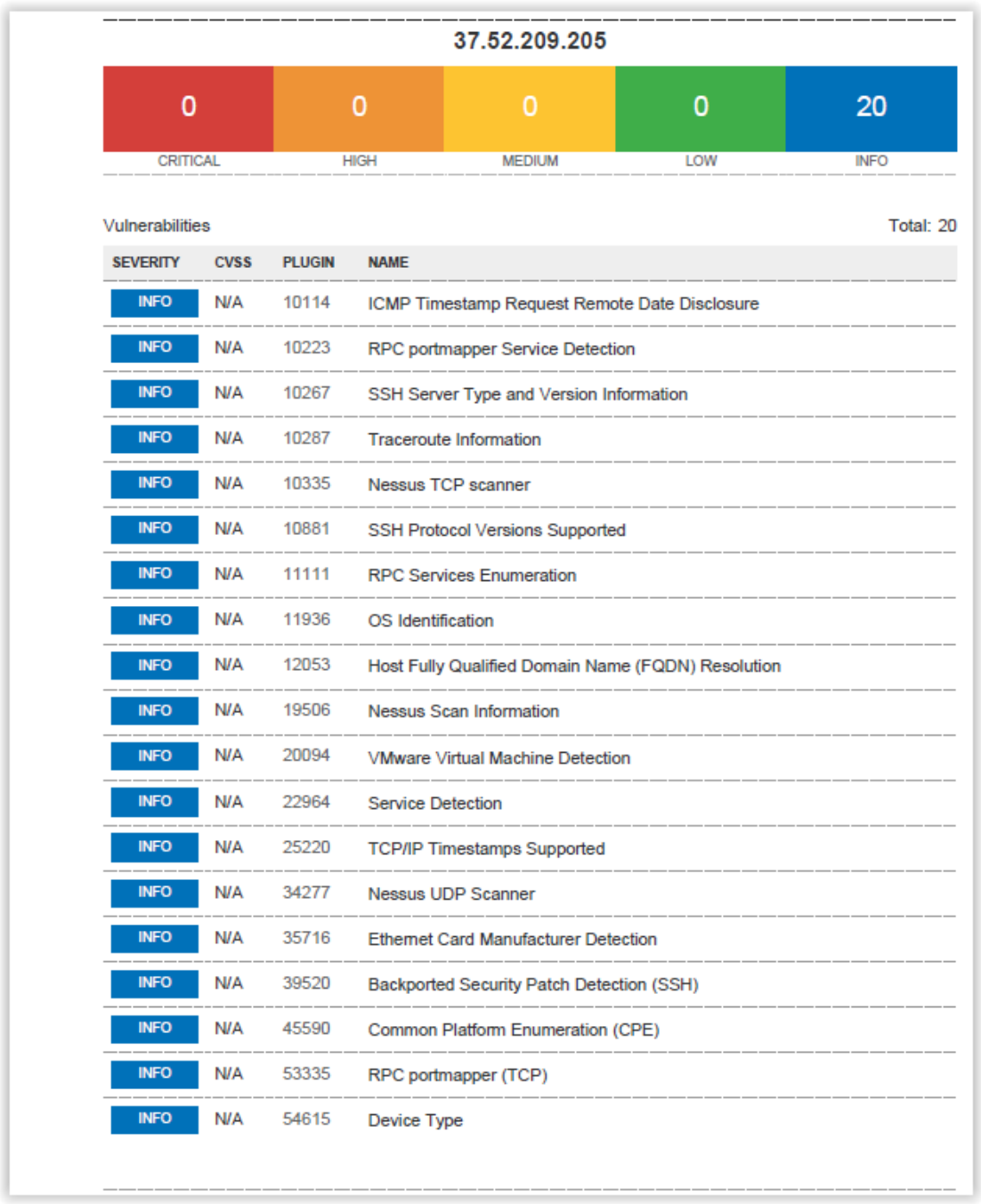
0	0	6	2	38
CRITICAL	HIGH	MEDIUM	LOW	INFO

Vulnerabilities — Total: 46

SEVERITY	CVSS	PLUGIN	NAME
MEDIUM	6.4	51192	SSL Certificate Cannot Be Trusted
MEDIUM	6.4	57582	SSL Self-Signed Certificate
MEDIUM	5.0	42873	SSL Medium Strength Cipher Suites Supported
MEDIUM	5.0	45411	SSL Certificate with Wrong Hostname
MEDIUM	4.3	10815	Web Server Generic XSS
MEDIUM	4.3	90317	SSH Weak Algorithms Supported
LOW	2.6	70658	SSH Server CBC Mode Ciphers Enabled
LOW	2.6	71049	SSH Weak MAC Algorithms Enabled
INFO	N/A	10114	ICMP Timestamp Request Remote Date Disclosure
INFO	N/A	10267	SSH Server Type and Version Information
INFO	N/A	10287	Traceroute Information
INFO	N/A	10335	Nessus TCP scanner
INFO	N/A	10386	Web Server No 404 Error Code Check
INFO	N/A	10863	SSL Certificate Information
INFO	N/A	10881	SSH Protocol Versions Supported
INFO	N/A	11032	Web Server Directory Enumeration
INFO	N/A	12053	Host Fully Qualified Domain Name (FQDN) Resolution
INFO	N/A	19506	Nessus Scan Information
INFO	N/A	21643	SSL Cipher Suites Supported

Figure 21. Iot.satsuma.com Nessus Advanced scan result

Figure 21 presents the scan results for iot.satsuma.com server. Nessus was able to detect six medium classified vulnerabilities and the rest of the findings is information type of data. The key point here is that for this device, five of those findings are related to the SSL protocol. The Web Server finding was related to the possibly malicious JavaScript.



37.52.209.205

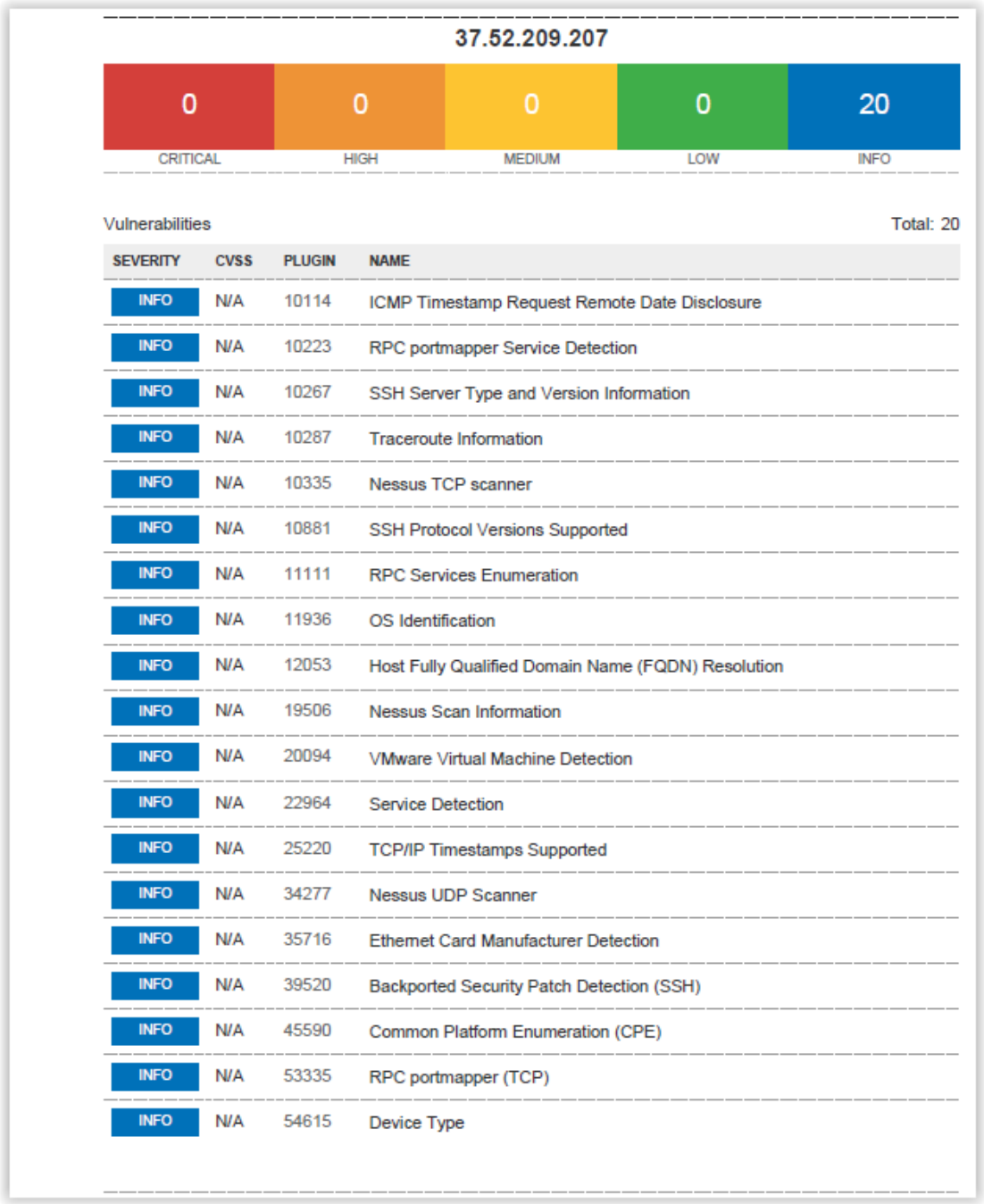
0	0	0	0	20
CRITICAL	HIGH	MEDIUM	LOW	INFO

Vulnerabilities — Total: 20

SEVERITY	CVSS	PLUGIN	NAME
INFO	N/A	10114	ICMP Timestamp Request Remote Date Disclosure
INFO	N/A	10223	RPC portmapper Service Detection
INFO	N/A	10267	SSH Server Type and Version Information
INFO	N/A	10287	Traceroute Information
INFO	N/A	10335	Nessus TCP scanner
INFO	N/A	10881	SSH Protocol Versions Supported
INFO	N/A	11111	RPC Services Enumeration
INFO	N/A	11936	OS Identification
INFO	N/A	12053	Host Fully Qualified Domain Name (FQDN) Resolution
INFO	N/A	19506	Nessus Scan Information
INFO	N/A	20094	VMware Virtual Machine Detection
INFO	N/A	22964	Service Detection
INFO	N/A	25220	TCP/IP Timestamps Supported
INFO	N/A	34277	Nessus UDP Scanner
INFO	N/A	35716	Ethernet Card Manufacturer Detection
INFO	N/A	39520	Backported Security Patch Detection (SSH)
INFO	N/A	45590	Common Platform Enumeration (CPE)
INFO	N/A	53335	RPC portmapper (TCP)
INFO	N/A	54615	Device Type

Figure 22. Sensor AMQP Nessus Advanced scan result

Figure 22 presents the scan results for temperature AMQP sensor. This device is one of the sensors for collecting ambient temperature data from the environment. Based on Nessus scanning, all notified vulnerabilities are classified as information type of data.



37.52.209.207

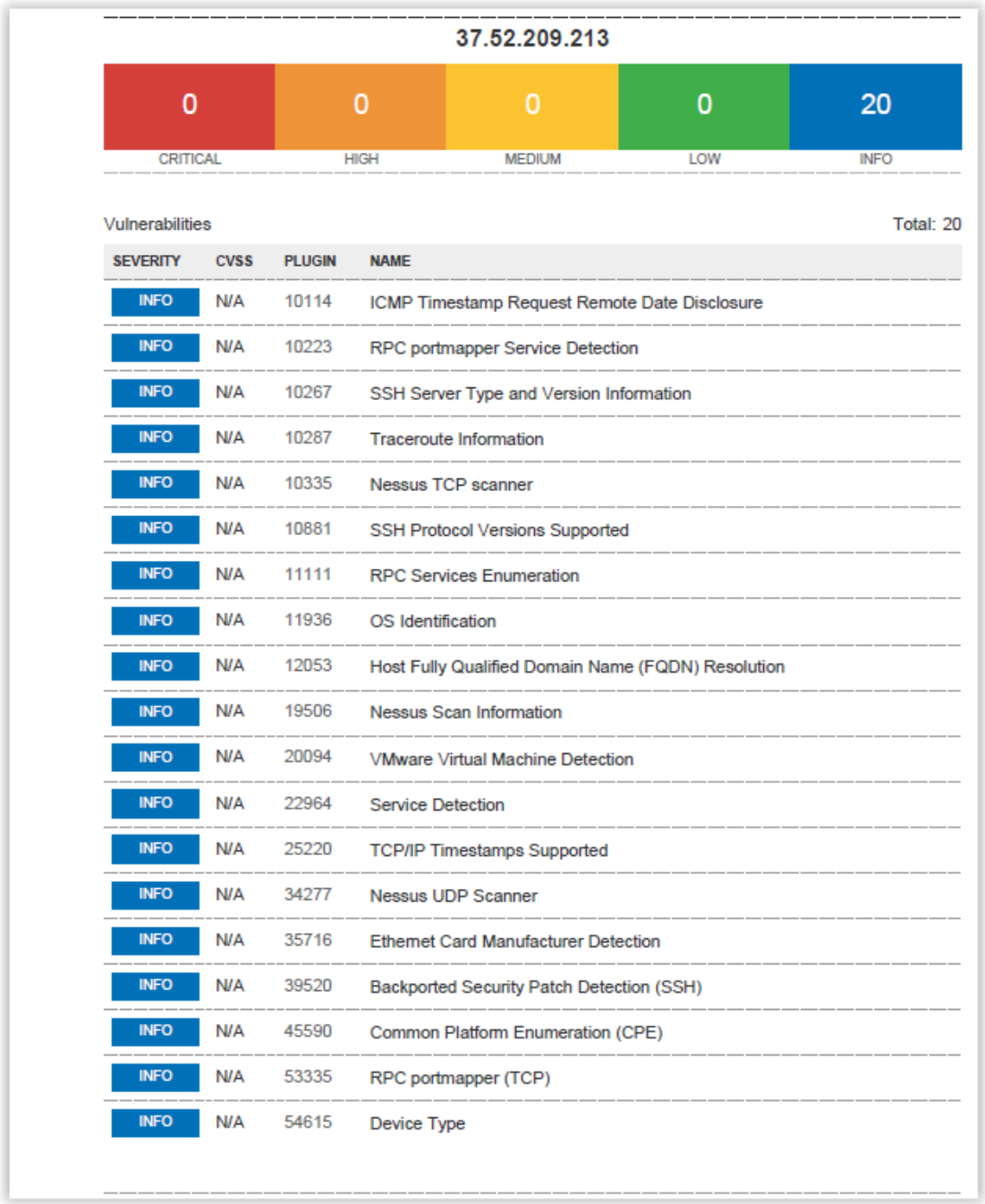
0	0	0	0	20
CRITICAL	HIGH	MEDIUM	LOW	INFO

Vulnerabilities — Total: 20

SEVERITY	CVSS	PLUGIN	NAME
INFO	N/A	10114	ICMP Timestamp Request Remote Date Disclosure
INFO	N/A	10223	RPC portmapper Service Detection
INFO	N/A	10267	SSH Server Type and Version Information
INFO	N/A	10287	Traceroute Information
INFO	N/A	10335	Nessus TCP scanner
INFO	N/A	10881	SSH Protocol Versions Supported
INFO	N/A	11111	RPC Services Enumeration
INFO	N/A	11936	OS Identification
INFO	N/A	12053	Host Fully Qualified Domain Name (FQDN) Resolution
INFO	N/A	19506	Nessus Scan Information
INFO	N/A	20094	VMware Virtual Machine Detection
INFO	N/A	22964	Service Detection
INFO	N/A	25220	TCP/IP Timestamps Supported
INFO	N/A	34277	Nessus UDP Scanner
INFO	N/A	35716	Ethernet Card Manufacturer Detection
INFO	N/A	39520	Backported Security Patch Detection (SSH)
INFO	N/A	45590	Common Platform Enumeration (CPE)
INFO	N/A	53335	RPC portmapper (TCP)
INFO	N/A	54615	Device Type

Figure 23. Sensor MQTT Nessus Advanced scan result

Figure 23 presents the scan results for temperature MMTT sensor. This device is also the sensor for collecting ambient temperature data. Based on Nessus scanner, twenty (20) findings are classified as information type of vulnerabilities.



37.52.209.213

0	0	0	0	20
CRITICAL	HIGH	MEDIUM	LOW	INFO

Vulnerabilities — Total: 20

SEVERITY	CVSS	PLUGIN	NAME
INFO	N/A	10114	ICMP Timestamp Request Remote Date Disclosure
INFO	N/A	10223	RPC portmapper Service Detection
INFO	N/A	10267	SSH Server Type and Version Information
INFO	N/A	10287	Traceroute Information
INFO	N/A	10335	Nessus TCP scanner
INFO	N/A	10881	SSH Protocol Versions Supported
INFO	N/A	11111	RPC Services Enumeration
INFO	N/A	11936	OS Identification
INFO	N/A	12053	Host Fully Qualified Domain Name (FQDN) Resolution
INFO	N/A	19506	Nessus Scan Information
INFO	N/A	20094	VMware Virtual Machine Detection
INFO	N/A	22964	Service Detection
INFO	N/A	25220	TCP/IP Timestamps Supported
INFO	N/A	34277	Nessus UDP Scanner
INFO	N/A	35716	Ethernet Card Manufacturer Detection
INFO	N/A	39520	Backported Security Patch Detection (SSH)
INFO	N/A	45590	Common Platform Enumeration (CPE)
INFO	N/A	53335	RPC portmapper (TCP)
INFO	N/A	54615	Device Type

Figure 24. Cooler MQTT Nessus Advanced scan result

Figure 23 presents the scan results for the cooler unit in this environment. As can be seen above, Nessus scanner was not able to detect any critical, high, medium or low types of vulnerabilities. Instead, only twenty (20) of information type observations were detected.

7.6 Recommendations based on scanning

When starting to consider possible actions for preventing attackers to compromise this case study IoT environment, there are different tasks and recommendation actions to perform.

7.6.1 NMAP recommendations

The output from NMAP command gave the basic information from the devices like the open ports, services and possible operating system level. The output from NMAP gives the attacker path and method to start considering a possible route to exploit the environment. The NMAP is typically used in the early state of an attack as an espionage tool. The key is to collect as much as possible valuable information from the environment to help attackers to build up techniques to exploit the known vulnerabilities against the environment.

Targets	Address	Up	Port	State	Service version	OS
Cooler MQTT	3	Yes	22/tcp	open	ssh, Openssh 6.7p1 Debian 5	Linux 3.2 - 4.8
			111/tcp	open	rpcbind 2.4	
			68/udp	open	dhcpd	
			111/udp	open	rpcbind 2.4	
Sensors AMQP	5	Yes	22/tcp	open	ssh, Openssh 6.7p1 Debian 5	Linux 3.2 - 4.8
			111/tcp	open	rpcbind 2.4	
			68/udp	open	dhcpd	
			111/udp	open	rpcbind 2.4	
Sensors MQTT	7	Yes	22/tcp	open	ssh, Openssh 6.7p1 Debian 5	Linux 3.2 - 4.8
			111/tcp	open	rpcbind 2.4	
			68/udp	open	dhcpd	
			111/udp	open	rpcbind 2.4	
cloud.satsuma.com	94.101.0.100	Yes	22/tcp	open	ssh, Openssh 6.6.1, protocol 2.0	Linux 4.4
			80/tcp	open	http, Apache httpd 2.4.6, OpenSSL	
			111/tcp	open	rpcbind 2.4	
			443/tcp	open	ssl/https Apache httpd 2.4.6, OpenSSL	
			111/udp	open	rpcbind 2.4	
portal.satsuma.com	94.101.0.101	Yes	22/tcp	open	ssh, Openssh 6.6.1, protocol 2.0	Linux 4.4
			80/tcp	open	ssl/http Werkzeug/0.9.1 Python/2.7.5	
iot.satsuma.com	94.101.0.102	Yes	22/tcp	open	ssh, Openssh 6.6.1, protocol 2.0	Linux 4.4
			80/tcp	open	http proxy HAProxy http proxy 1.3.1	
			443/tcp	open	ssl/https proxy HAProxy http proxy 1.3.1	
			8443/tcp	open	ssl/https Golang net/http server	
Container.satsuma.co	94.101.0.102	yes	22/tcp	open	ssh, Openssh 6.6.1, protocol 2.0	Linux 4.4
			80/tcp	open	http proxy HAProxy http proxy 1.3.1	
			443/tcp	open	ssl/https proxy HAProxy http proxy 1.3.1	
			8443/tcp	open	ssl/https Golang net/http server	

Figure 25. NMAP detailed findings

Figure 25, NMAP detailed findings, shows clearly the versions used in this environment. The used operating system versions are based on Linux kernel 3.x and 4.x. The latest available stable version currently is 4.16.5. One issue is with used version of OpenSSH 6.7p1. According to the source (CVE Details 2018), the OpenSSH version 6 up till 7 has severe vulnerabilities, which allows hackers to cause a Denial of Service, DoS, attack against OpenSSH. The latest available version is version 7.6. Based on NMAP scanning, also Apache httpd should be upgraded from 2.4.6 to 2.4.33 due to the severe vulnerabilities. (Apache HTTP Server Project N.d.)

Severe vulnerabilities were also detected with Python because the installed version is 2.7.5 and the latest version is 3.4.8, which has mitigations against vulnerabilities. (Python N.d.).

The recommendation based on findings with NMAP is to close unused ports and services to prevent any unwanted penetration activity. Also, it is strongly recommended to upgrade used services to the latest versions, because at this point those were far behind the safest versions. Table 14 summarizes the need for upgrading different modules.

Table 14. Versions to upgrade

Module	Installed version	Latest available
Linux kernel	3.x and 4.x	4.16.5
OpenSSH	6.7p1	7.6
Apache	2.4.6	2.4.33
HAproxy	1.3.1	1.8.4

7.6.2 Nessus recommendations

Nessus scanner results are quite clear and straightforward to implement. Nessus reported several vulnerabilities related to the SSL protocol for all application servers. Therefore, the recommended actions for all those applications servers are almost equal. The following tables (Tables 15, 16, 17) summarize the detected vulnerabilities for recommended mitigation actions based on Nessus scanned results.

Table 15. Vulnerabilities detected by Nessus

Host cloud.satsuma.com, 92.101.0.100		
Plugin	Name	Mitigation
51192	SSL Certificate Cannot Be Trusted	The X.509 certificate for this service cannot be trusted. Purchase or generate a proper certificate for this service.
57582	SSL Self-Signed Certificate	The SSL certificate chain for this service ends in an unrecognized self-signed certificate. Purchase or generate a proper certificate for this service.
11213	HTTP TRACE / TRACK Methods Allowed	Debugging functions are enabled on the remote web server. Disable these methods. Refer to the plugin output for more information.
20007	SSL Version 2 and 3 Protocol Detection	The remote service encrypts traffic using a protocol with known weaknesses. Consult the application's documentation to disable SSL 2.0 and 3.0. Use TLS 1.1 (with approved cipher suites) or higher instead.
40984	Browsable Web Directories	Some directories on the remote web server are browsable. Make sure that browsable directories do not leak confidential informative or give access to sensitive resources. Additionally, use access restrictions or disable directory indexing for any that do.
42873	SSL Medium Strength Cipher Suites Supported	The remote service supports the use of medium strength SSL ciphers. Reconfigure the affected application if possible to avoid use of medium strength ciphers. See also https://www.openssl.org/blog/blog/2016/08/24/sweet32/
85582	Web Application Potentially	The remote web server may fail to mitigate a class of web application vulnerabilities.

	Vulnerable to Clickjacking	Return the X-Frame-Options or Content-Security-Policy (with the 'frame-ancestors' directive) HTTP header with the page's response. This prevents the page's content from being rendered by another site when using the frame or iframe HTML tags.
90317	SSH Weak Algorithms Supported	The remote SSH server is configured to allow weak encryption algorithms or no algorithm at all. Contact the vendor or consult product documentation to remove the weak ciphers. See also https://tools.ietf.org/html/rfc4253#section-6.3

As can be seen above, by purchasing or regenerating the proper SSL certificate, (51192, 57582) some of the vulnerabilities would be prevented. Also, reconfiguring (42873) the SSL and by removing the weak cipher (90317) would solve two vulnerabilities. Also, disabling SSL 2.0 and 3.0 (20007) would prevent several cryptographic flaws detected in those versions. Implementing the TLS1.1 or higher with proper cipher version or higher would protect against the cryptographic flaws.

Vulnerability 40984 is related to the browsing directories in the web server. This vulnerability allows attacker to fetch sensitive information from web server. (Directory Indexing 2010; Common Weakness Enumeration 2018.)

Vulnerabilities 11213 and 85582 are both related to the remote web servers. The 11213 is a rather old vulnerability, which can be prevented by disabling the HTTP TRACE support from the web server. It should also be noted that this vulnerability is related to Apache, and therefore upgrading Apache to the latest version is essential. The 85582 vulnerability is also rather old. There are few solutions for protecting against that vulnerability. The first one is to include frame-breaking functionality which prevents other pages from framing the site to defend. Another possible solution is to use JavaScript frame-breaking code. (OWASP 2017.)

Table 16. Vulnerabilities detected by Nessus

Host portal.satsuma.com, 92.101.0.101		
Plugin	Name	Mitigation
85582	Web Application Potentially Vulnerable to Clickjacking	The remote web server may fail to mitigate a class of web application vulnerabilities. Return the X-Frame-Options or Content-Security-Policy (with the 'frame-ancestors' directive) HTTP header with the page's response. This prevents the page's content from being rendered by another site when using the frame or iframe HTML tags.
90317	SSH Weak Algorithms Supported	The remote SSH server is configured to allow weak encryption algorithms or no algorithm at all. Contact the vendor or consult product documentation to remove the weak ciphers. See also https://tools.ietf.org/html/rfc4253#section-6.3

The 85582 vulnerability is rather old. There are few solutions for protecting against that vulnerability. The first one is to include frame-breaking functionality, which prevents other pages from framing the site to defend. Another possible solution is to use JavaScript frame-breaking code. (OWASP 2017.)

The vulnerability 90317 is fixed by removing the weak cipher.

Table 17. Vulnerabilities detected by Nessus

Host iot.satsuma.com, 92.101.0.102		
Plugin	Name	Mitigation
51192	SSL Certificate	The X.509 certificate for this service cannot be trusted. Purchase or generate a proper certificate for this service.

	Cannot Be Trusted	
57582	SSL Self-Signed Certificate	The SSL certificate chain for this service ends in an unrecognized self-signed certificate. Purchase or generate a proper certificate for this service.
42873	SSL Medium Strength Cipher Suites Supported	The remote service supports the use of medium strength SSL ciphers. Reconfigure the affected application if possible to avoid use of medium strength ciphers. See also https://www.openssl.org/blog/blog/2016/08/24/sweet32/
45411	SSL Certificate With Wrong Hostname	The SSL certificate for this service is for a different host. Purchase or generate a proper certificate for this service.
10815	Web Server Generic XSS	The remote web server is affected by a cross-site scripting vulnerability. Contact the vendor for a patch or upgrade.
90317	SSH Weak Algorithms Supported	The remote SSH server is configured to allow weak encryption algorithms or no algorithm at all. Contact the vendor or consult product documentation to remove the weak ciphers. See also https://tools.ietf.org/html/rfc4253#section-6.3

The mitigation activities for this application server, iot.satsuma.com, are mainly the same as for other servers because Nessus detected the same vulnerabilities from all these servers. However, the vulnerabilities 45411 and 10815 need further investigation.

The solution for 45411 is to generate or purchase a proper certificate because currently the common name field identified a different host name than what the actual server was.

The 10815 is cross-site scripting related to the web server. This can occur when the attacker feeds malicious code by using the web application in form of browser side

script to a different end user. These kinds of flaws are quite common and by following several simple rules by OWASP's (2018) XSS Prevention Cheat Sheep pages, the most of those serious attacks can be avoided. This vulnerability is also listed as a possible attack vector, OWASP ref A7:2017. Therefore, preventing this defect is important.

Next, the sensors and cooler detections are to be discussed in detail. Nessus scanner notified an information classified type of vulnerabilities from both the temperature sensors and the cooler unit. Those vulnerabilities do not require any actions at this point because those can be counted as not critical ones. Nevertheless, it is good to be aware that these sensors and cooler unit are visible and open in the network for NMAP and Nessus scanning.

As a summary for recommended actions, the key tasks are to correct the SSL's certificate issues and configurations related to SSL. By fixing these issues, the overall security level could be much higher. Then, the next corrective action should be upgrading all software modules detected by NMAP. Also, it needs to be mentioned that the operating system version and patches should be upgraded in a regular manner. This is one of the key tasks to protect the environment against vulnerabilities.

7.6.3 Management level recommendations

When considering the needed actions to perform for enhancing the security level in the case study environment, the administrators should also take the needed workload against the risk level into account. If the risk level against a certain vulnerability is low, then the performed actions should be in proportion to that risk level. However, this is not only an administrator's responsibility because the management should be also involved in decision making. The management is responsible for the development of a valid security policy, and that it is followed. The scope of security policy should be compared to the organization's size and used resources. It is important to understand what is relevant to spend time on and what the relationship of the security policy is to business processes and core business.

There are several different methodologies to measure and estimate the risks caused by different vulnerabilities and threats. The demonstrative way to present a possible Risk Rating is to use OWASP's (2017) generated method used in this case study. It contains an easy and visual way to categorize different attack vectors against real risk factor to create following, where different modules are listed with relevant numerical values. The key here is also to realize that creating the numerical risk table is an estimate, and it might differ from real life situation. However, it will give view and guidance what to focus on with the actions against threats. The detailed Risk Rating Methodology definitions with numerical values are listed in Appendix 3.

First, there is a need to decide of the numerical scale for impact. Normally the scale is from 0-9 with following allocations for low, medium and high:

Table 18. Impact numerical allocations

Likelihood and impact	
0 – 3	Low
3 – 6	Medium
6 – 9	High

The potential risks have been defined using possible attack vectors and findings with NMAP and Nessus scanners. The next step is to define the likelihood, estimated impact with different key elements, and after that the calculations of severity of risk can be done.

The following calculation is based on the defined attack vector against this environment. The used attack vector in Table 19 is Broken Authentication, OWASP ref A2: 2017.

Table 19. Overall likelihood

Threat agent factor					Vulnerability factors			
Skill level	Motive	Opportunity	Size		Ease of discovery	Ease of exploit	Awareness	Intrusion detection
3	3	3	3		3	3	3	3
Overall likelihood = 3 (Medium)								

Table 20. Overall technical and business impact

Technical impact, CIA					Business impact			
Loss of confidentiality	Loss of integrity	Loss of availability	Loss of accountability		Financial damage	Reputation damage	Non-compliance	Privacy violation
3	3	3	3		3	3	3	3
Overall technical impact = 3 (Medium)					Overall business impact = 3 (Medium)			

After the determination and calculation of overall impact estimation values, these values can be compared to the overall risk severity in Table 21 to gain the best view of the final severity rating.

Table 21. Overall risk severity

Overall risk severity				
Impact	HIGH	Medium	High	Critical
	MEDIUM	Low	Medium	High
	LOW	Note	Low	Medium
		LOW	MEDIUM	HIGH
	Likelihood			

Based on the above calculations ((3+3+3)/3), the overall risk severity factor is 3 and it can be categorized as Medium. However, the calculations should always be done based on the worst scenario and compared to the core business. Meaning that if the technical impact is high but the impact to the business is low, then the protection mechanisms should be based on that business level. The summary of the calculated risk severity is in Table 22. The rest of the corresponding calculations for attack vectors are found in the Appendices part, Appendix 3.

The above calculation should serve as a guidance and aid to fulfil an organization's security policy defining which vulnerabilities and threats will be fixed and with what schedule: immediately, soon, or during the next service break.

Below is a complete table, Table 22 regarding the calculated Risk Ratings for different attack vectors. The rest of the calculations can be found in Appendix 3.

Table 22. Risk ratings

Possible risk	Risk Severity
Broken Authentication, OWASP ref A2: 2017	Medium
Sensitive data exposure, OWASP ref A3: 2017	Medium
Security misconfiguration, OWASP ref A6: 2017	Medium
Cross-Site Scripting, OWASP ref A7: 2017	Medium
Using component with known vulnerabilities, OWASP ref A9: 2017	Medium
Insufficient logging & monitoring, OWASP ref A10: 2017	Medium
Manipulating user data in different tenant inside IoT framework	Medium
Denial of Service attack against IoT environment	Medium
Man-In-the-Middle attack against IoT environment	Medium
NMAP scanner findings	Medium
NESSUS scanner findings	Medium

As can be seen, the calculated risk severity factor for those attack vectors was classified as Medium. Those are estimates which might differ based on a different kind of emphasis depending on the organization's requirements for cyber security. Medium classified risks need mitigation activities in near future to protect the environment. If any of those risk severities are calculated and defined higher than Medium, e.g. High or Critical, then that attack vector needs immediate mitigation action against that attack vector. Also, if any of the risk severities are lower than Medium, e.g. Low or Info, then the necessary mitigation actions should be performed in the future. This is the decision for the organization to do.

8 Conclusion and discussion

The Internet of Things, IoT, devices and appliances are getting more and more popular nowadays. Gartner (2017) estimates that the number of connected IoT devices could be 20 billion or more in year 2020. This will induce that the traditional ICT business decision makers are faced with a new kind of thinking. Quite often the IoT devices of the new era are installed to the premises supporting only one purpose. In the worst-case scenario, all normal maintenance procedures together with regular updates are ignored, which will lead to a situation where a significant group of IoT devices is available for possible hostile activities in the local premises or are to be a part of bigger attacks such as botnet.

Currently, almost daily there are news and information about severe attacks generated by hackers using the IoT device in various locations. Those compromised IoT devices typically present consumer appliances. The industrial based IoT devices are seldom an attack target for hostile activities. This will cause the need for preparing protection for domestic market devices against external and internal vulnerabilities and attacks.

Based on the theoretical part of thesis, an IoT device could be visualized as a computer from the architectural point of view. However, there are few differences in IoT devices compared to computers; one difference in general being the sensor part for measuring data and the second difference being the size. Typically, IoT devices are small form factor devices, which will have a positive effect on the device power consumption.

First, the theoretical part of this thesis is discussed. As mentioned above, the IoT environments are comparable to general ICT environments, i.e. even the smallest IoT environment needs life-cycle management approach from the development phase up to the production ready devices. Also, the complete data path from the raw data of the sensors to the visualization server and application should be defined clearly, and all components must be well-known. The responsibility about the life-cycle management for the IoT product is not only developers' task and responsibility but very likely also management's responsibility. The management should be thriving for the organization level security management policy, which should include elements such as securing the life-cycle of IoT environment, Information Security Management System (ISMS), disaster recovery processes and possible risk management calculations for assets based on possible attack vectors. Together with these elements, the information security must take confidentiality, integrity and availability (CIA) into consideration. These terms give the basic guidelines for handling, storing and delivering of the confidential data to other counterparts. Together with ISMS, the management should define and implement the security controls for different kinds of scenarios.

The Security Information and Event Management, SIEM, is also one toolset which the management should consider implementing. There are vendors who provide SIEM as a service and some vendors offer SIEM as software. The SIEM provides tools for information gathering from the complete environment, identity management, vulnerability management and logging capabilities from various parts of environment.

The security controls listed above must be implemented to all parts of an organization, and every single individual employee must know why such security controls have been defined and implemented in practice. They should also be aware

of actions and tasks which should be considered in case of emergency, such as communication path. The responsibilities related to the communication path must be defined and clearly communicated to all parties involved in it. If there is need to report deviation to authorities, the communication interface should be clearly defined, and a responsible person or organization defined.

If organizations use partners, the defined security controls should be included in their processes also. When an organization starts considering and designing different security controls, the processes and tasks should be in line with the value of the IoT assets and valuation of the data.

There are various toolsets to help and fulfil an organization's needs what comes to designing and implementing security controls. The most appropriate tools should be selected based on a consideration of the IoT environment because not all toolsets are suitable for all organizations. Some of the toolsets are technical security control implementations, and some of them are more related to the organization's way of operations.

When considering the IoT environment, the technical security control implementations are quite like the ones in a traditional ICT environment. The physical protection of the IoT devices should be taken into consideration. There is a risk that physical devices are compromised if the access to the device's ports, such as USB, serial or JTAG connectors, is not protected or is disabled by default. The risk of physical damage also should be taken into notice. The mitigation solution against a physical threat is to place the IoT device in secured location with access control implemented.

Related to IoT device's hardware solutions, the regular updates for firmware levels and possible core applications should be implemented. The updates should be configured in such a way that update packages are protected with checksums and possible certificates, and the source for the update is validated before downloading the packages. Also, one key point is to implement automated updates to ensure that software modules are always up-to-date with the latest level.

One important part to implement and consider is the user management. Only the user who must have access to a system should have it, and all other users must be

restricted. The password policy should be defined in such a way that at least the length, lifetime and logout value are defined for all users. Also, when adding a new user to the system, the user must change the password within the first login attempt. Implementing strong password policy will protect the IoT environment against possible password cracking tools.

The key point with an IoT device is to protect the network traffic. Typically, those devices use more often wireless network than wired network modules. This creates the demand to protect the transmission traffic between a sensor device and collecting point of data. Whenever it is possible, the secured transmission should be implemented to protect the data transmit.

To have a clear view of the network traffic, a firewall with defined security rules should be implemented. The firewall gives opportunity to allow or restrict the traffic between different network modules. For having more information from the network traffic, the implementation of Intrusion Detecting Systems, IDS, and Intrusion Prevention System, IPS, should also be considered. Those tools together with the firewall implementation give the administrators more visibility of the network traffic and usage and time to reach for possible DoS attacks.

Another key element also to mention is the logging system for the IoT environment. There could be different kinds of logs collected from systems such as system logs, usage logs, change logs and error logs. The proper implemented logging systems give a clear view of actions performed by different users. Also, any abnormal activities related to the environment could be notified from logs and then they can have a clear indication of hostile activities against environment. The important matter related to logs is log management. The logs should be stored and analysed is such a way that there is no risk of compromising the log data.

One basic part of the implementation of technical security controls is to disable and close all unnecessary and unused ports and services. The rule could be that only a minimum number of ports and services should be open and active, and these are the ones related to the running applications. All other ports and services must be closed.

When considering the case study environment used in this thesis, the findings with NMAP and Nessus scanners were convergent with the theoretical part of thesis. The

scanners were able to notify several out-of-date software modules which should be updated soon to mitigate possible threats generated by old software versions. The scanners also detected several issues related to old or weak SSL certificates. The issue with SSL is quite easily corrected by implementing a proper SSL certificate and configuring SSL in a proper way. Nessus was able to detect in the environment some more severe vulnerabilities, which should be corrected in the near future such as Web Application Clickjacking vulnerability and Web Server Generic XSS, cross-site scripting vulnerability.

The research question defined for this thesis was of the writer's own interest. The findings answering these questions are listed below as divided in the theoretical part and the case study. The thesis consists of a theoretical part on research architectural concept of IoT devices, and the practical part focused on finding answers for following research questions:

- What is the level of cyber security initially within the IoT environment?
- What is the minimum acceptable level of cyber security in IoT environment?
- What are the minimum implementations for improving the cyber security in the IoT environment?

When considering the first question, the answer is that the cyber security level of the IoT environment is not sufficient enough to protect environment from the cyber security perspective. The case study findings supported that view. From that perspective, the initially configured IoT environments need a security implementation.

The second question and answer are mainly meant for management level stakeholders. They should define the organization's cyber security policies and practices and make sure that the security controls are implemented in practice. They should take the monetary value and market values into notice when defining the security controls. Because investing a great deal of resources for a low level IoT implementation might not be relevant. However, on the other hand, the loss of market share and reputation are critical elements for a profit-making organization.

The third question and the answer for that are related to the second question. It is a management decision how much they will invest in protecting IoT environments or devices against possible cyber security threats. That decision defines the levels of security controls. For example, if the IoT appliance is the organization's main product and vital from the profit aspect, then the security controls must be implemented in such a way that there is no risk of the IoT appliance being jeopardized by external intruders in any circumstances.

As a summary, the outcome from thesis was that any of the IoT environments or appliances should be served as any other ICT or computer devices. The IoT appliances have the same modules as traditional computers and are in the same way also vulnerable for different kinds of external threats. Therefore, the protection of IoT appliances is as critical as any other computer system connected to the internet. Because by protection, the device itself is not the only thing but is protecting the other user on the internet as well.

References

5 Tools for IoT Security. 2017. Accessed 3 February 2018. Retrieved from http://www.ioti.com/security/5-tools-iot-security.

AMQP. N.d. AMQP is the Internet Protocol for Business Messaging. Accessed 3 February 2018. Retrieved from http://www.amqp.org/about/what.

Apache HTTP Server Project. N.d. Apache HTTP Server 2.4 vulnerabilities. Accessed 3 February 2018. Retrieved from https://httpd.apache.org/security/vulnerabilities_24.html.

Ashton, K. 2009. That 'Internet of Things' Thing, RFID Journal. Accessed 3 February 2018. Retrieved from http://www.rfidjournal.com/articles/view?4986.

Common Attack pattern Enumeration and Classification. 2018. A Community Resource for Identifying and Understanding Attacks. Accessed 17 March 2018. Retrieved from https://capec.mitre.org.

Common Criteria. The Common Criteria for Information Technology Security Evalaution. N.d. Accessed 17 March 2018. Retrieved from https://www.commoncriteriaportal.org.

Common Weakness Enumeration. 2018. CWE-548: Information Exposure Through Directory Listing. Accessed 17 March 2018. Retrieved from http://cwe.mitre.org/data/definitions/548.html.

Common Weakness Enumeration. 2018. Starting page. Accessed 31 March 2018. Retrieved from https://cwe.mitre.org.

Computer Security Resource Center. 2018. Starting page. Accessed 31 March 2018. Retrieved from http://csrc.nist.gov.

CVE Details. 2018. OpenSSH 6.0 Security Vulnerabilities. Accessed 31 March 2018. Retrieved from https://www.cvedetails.com/vulnerability-list/vendor_id-97/product_id-585/version_id-156382/Openbsd-Openssh-6.0.html.

Directory Indexing. 2010. WASC Threat Classification. Accessed 2 April 2018. Retrieved from http://projects.webappsec.org/w/page/13246922/Directory%20Indexing.

Doctor Who. N.d. Introduce of the TV series. Accessed 2 April 2018. Retrieved from https://en.wikipedia.org/wiki/Doctor_Who.

English Oxford Living Dictionaries. N.d. Accessed 2 April 2018. Retrieved from https://en.oxforddictionaries.com/definition/cybersecurity.

European Union Agency for Network and Information Security (ENISA). 2015. Definition of Cybersecurity. Gaps and overlaps in standardisation. Accessed 23 February 2018. Retrieved from https://www.enisa.europa.eu/publications/definition-of-cybersecurity.

Fruhlinger, J. 2018. The Mirai botnet explained: How teen scammers and CCTV cameras almost brought down the internet. Accessed 14 April 2018. Retrieved from

https://www.csoonline.com/article/3258748/security/the-mirai-botnet-explained-how-teen-scammers-and-cctv-cameras-almost-brought-down-the-internet.html.

Gartner. 2017. Gartner Says 8.4 Billion Connected "Things" Will Be in Use 2017, Up 31 Percent From 2016. 14 April February 2018. Retrieved from https://www.gartner.com/newsroom/id/3598917.

Gartner. N.d. Market Guide for IoT Platforms. Accessed 19 March 2018. Retrieved from https://www.gartner.com/technology/research/internet-of-things/.

Guide To Fiber Optics & Premises Cabling. N.d. The Fiber Optic Association – Tech Topics. Accessed 19 March 2018. Retrieved from http://www.thefoa.org/tech/connID.htm.

Hitachi Vantara Big Data Analytics 2017. Accessed 23 September 2017. Retrieved from corporation's web pages. Pdf document.

Hitachi Insight Group IoT. 2016. Lumada: Internet of Things Breakthrough. Accessed 20 March 2018. Retrieved from https://www.youtube.com/watch?v=LC9GdmaZAnk.

IEEE 802.11ah. 2018. Accessed 23 February 2018. Retrieved from https://en.wikipedia.org/wiki/IEEE_802.11ah.

Infosec Guide. 2017. Defending Against Man-In-the-Middle Attacks. Accessed 20 March 2018. Retrieved from https://www.trendmicro.com/vinfo/us/security/news/cybercrime-and-digital-threats/infosec-guide-defending-against-man-in-the-middle-attacks.

Infrared transmission. N.d. Accessed 20 March 2018. Retrieved from https://searchnetworking.techtarget.com/definition/infrared-transmission.

Internet of Things Global Standards Initiative. N.d. Accessed 22 March 2018. Retrieved from https://www.itu.int/en/ITU-T/gsi/iot/Pages/default.aspx.

Internet of Things. 2018. Accessed 22 March 2018. Retrieved from https://en.wikipedia.org/wiki/Internet_of_things.

IoT Innovation. 2018. What 5G Means for the Internet of Things. Accessed 23 April 2018. Retrieved from https://internet-of-things-innovation.com/insights/the-blog/5g-means-internet-things/#.Wu74KaSFNhE.

ISO homepage. N.d. Accessed 22 March 2018. Retrieved from https://www.iso.org/home.html.

ISO. 2012. ISO/IEC 27032:2012 Information technology – Security techniques – guidelines for cybersecurity. Accessed 23 March 2018. Retrieved from https://www.iso.org/standard/44375.html.

ISO. 2012. ISO/IEC 27000:2018 Information technology – Security techniques – Information security management systems – Overview and vocabulary. Accessed 23 March 2018. Retrieved from https://www.iso.org/standard/73906.html.

JYVSECTEC CYBER RANGE. RGCE and solution. N.d. Accessed 29 March 2018. Retrieved from https://jyvsectec.fi/wp-content/uploads/JYVSECTEC-cyber-range.pdf.

Kennedy, C. 2017. The internet of things: The cyber security risks and how to protect against them. Accessed 23 March 2018. Retrieved from https://www.itproportal.com/features/the-internet-of-things-the-cyber-security-risks-and-how-to-protect-against-them/.

LTE Advanced. 2018. Accessed 18 March 2018. Retrieved from https://en.wikipedia.org/wiki/LTE_Advanced.

Miller, L. 2016. IoT Security For Dummies, INSIDE Secure Edition. Accessed 10 August 2017. Retrieved from https://www.insidesecure.com/content/download/909/10066/file/IOT%2BSecurity%2Bfor%2BDummies.pdf.

Ministry of the Interior. 2018. Civilian intelligenc legistlationwould improve Finland's national security. Accessed 23 February 2018. Retrieved from http://intermin.fi/en/article/-/asset_publisher/siviilitiedustelulaki-parantaisi-suomen-kansallista-turvallisuutta.

Mitre. 2018. Adversarial Tactics, Techniques & Common Knowledge. Welcome to ATT&CK. Accessed 23 March 2018. Retrieved from https://attack.mitre.org/wiki/Main_Page.

Moore, M. 2018. What is Industry 4.0? Everything you need to know. The latest news, views and developments in the workd of Industry 4.0. Accessed 23 March 2018. Retrieved from https://www.techradar.com/news/what-is-industry-40-everything-you-need-to-know.

MQTT. N.d. MQTT is machine-to-machine (M2M)/2Internet of Things" connectivity protocol. Accessed 23 February 2018. Retrieved from http://mqtt.org/.

Nessus Professional. N.d. Nessus product info page. Accessed 23 February 2018. Retrieved from https://www.tenable.com/products/nessus/nessus-professional.

New Azuma trains arrive at UK port ahead of passenger services starting later this year. 2018. Accessed 20 March 2018. Retrieved from https://www.virgintrains.co.uk/about/media-room#/pressreleases/new-azuma-trains-arrive-at-uk-port-ahead-of-passenger-services-starting-later-this-year-2453824.

OWASP. 2017. Authentication Cheat Sheet. Accessed 9 December 2017. Retrieved from https://www.owasp.org/index.php/Authentication_Cheat_Sheet.

OWASP. 2017. Clickjacking Defence Cheat Sheet. Accessed 9 December 2017. Retrieved from https://www.owasp.org/index.php/Clickjacking_Defense_Cheat_Sheet.

OWASP. N.d. OWASP Risk Rating Methodology. Accessed 9 December 2017. Retrieved from https://www.owasp.org/index.php/OWASP_Risk_Rating_Methodology.

OWASP. 2017. Pinning Cheat Sheet. Accessed 13 January 2018. Retrieved from https://www.owasp.org/index.php/Pinning_Cheat_Sheet.

OWASP. 2017. Top 10-2017 Application Security Risks. Accessed 13 January 2018. Retrieved from https://www.owasp.org/index.php/Top_10-2017_Application_Security_Risks.

OWASP. 2017. Top 10-2017 Top 10. Accessed 13 January 2018. Retrieved from https://www.owasp.org/index.php/Top_10-2017_Top_10.

OWASP. 2017. Transport Layer Protection Cheat Sheet. Accessed 13 January 2018. Retrieved from https://www.owasp.org/index.php/Transport_Layer_Protection_Cheat_Sheet.

OWASP. N.d. Welcome to OWASP, the free and open software security community. Accessed 13 January 2018. Retrieved from https://www.owasp.org.

OWASP. 2018. XSS (Cross Site Scripting) Prevention Cheat Sheet. Accessed 13 January 2018. Retrieved from https://www.owasp.org/index.php/XSS_(Cross_Site_Scripting)_Prevention_Cheat_Sheet.

Palmers, T. 2013. Implementing a vulnerability management process. SANS Institute InfoSec Reading Room. Accessed 23 February 2018. Retrieved from https://www.sans.org/reading-room/whitepapers/threats/implementing-vulnerability-management-process-34180.

Praetorian. 2017. The 8 Biggest IoT Security Mistakes and How to Avoid Them. Accessed 10 August 2017. Retrieved from https://p16.praetorian.com/downloads/report/The_8_Biggest_IoT_Security_Mistakes_and_How_to_Avoid_Them.pdf.

Puolustusministeriö. 2015. Katakri 2015. Tietoturvallisuuden auditointityökalu viranomaisille. Accessed 20 January 2018. Retrieved from https://www.defmin.fi/files/3165/Katakri_2015_Tietoturvallisuuden_auditointityokalu_viranomaisille.pdf.

Python. N.d. Download the latest version for Windows. Accessed 23 February 2018. Retrieved from https://www.python.org/downloads/.

Salonen, J. 2017. Tietoturvaratkaisut (SSCP), TestOut Security Pro, web based online trainig. Helsinki Metropolia University of Applied Sciences. Accessed 9 September 2017. Retrieved from https://cdn.testout.com/client-v5-1-10-495/startlabsim.html?.

Sagedhi, A. 2017. Things, trouble, Trust: On Building Trust in IoT. IoT Security WorkShop 07.09.2017 in Aalto University.

Sensor. 2012. Accessed 24 March 2018. Retrieved from https://whatis.techtarget.com/definition/sensor.

Solarwinds msp. 2017. Think Like a Cybercriminal. eBook. Accessed 2 February 2018. Retrieved from https://www.solarwindsmsp.com/resources/eb-it-security-101-think-cybercriminal.

Tarkoma, S. 2017. The Internet of Things Research Program and Beyond. IoT Security WorkShop 07.09.2017 in Aalto University.

Valtionvarainministeriö. N.d. VM, VAHTI ja tietoturvallisuus. Valtiovarainministeriö ohjaa ja yhteensovittaa julkishallinnon ja erityisesti valtionhallinnon

tietoturvallisuuden kehittämistä. Accessed 20 January 2018. Retrieved from https://www.vahtiohje.fi/web/guest/vm-vahti-ja-tietoturvallisuus.

Viestintävirasto. N.d. Kyberturvallisuus.Viestintävirasto kehittää ja valvoo viestiverkkojen ja palveluiden toimintavarmuutta ja turvallisuutta. Accessed 20 January 2018. Retrieved from https://www.viestintavirasto.fi/kyberturvallisuus.html.

Appendices

Appendix 1.

Raspberry pi3 with SenseHat, temperature, air pressure and humidity measurement, Python code for collecting data

```
##Libraries##

from sense_hat import SenseHat
from time import sleep
from time import time
from datetime import datetime

##Main program##

sense = SenseHat()

counter = 0

while counter < 50:
 t = sense.get_temperature()
 p = sense.get_pressure()
 h = sense.get_humidity()

 t = round(t, 1)
 p = round(p, 1)
 h = round(h, 1)

 print(t)
 print(p)
 print(h)
 sleep(4)
 counter = counter + 1
#copyright VSulkamo/8.10.2017
```

Appendix 2.

Sensor output data from Raspberry and Sense HAT

29.7
1001.7
35.3
28.3
1001.7
36.1
27.2
1001.6
39.0
27.4
1001.6
40.2
30.8
1001.5
39.9
33.1
1001.5
37.7
35.0
1001.4
35.3
37.5
1001.5
33.7
40.2
1001.4
31.1
41.8
1001.1
29.0
41.8
1001.3
27.9
41.2
1001.2
27.3
38.7
1001.2
27.8
36.0
1001.6
28.8
33.8
1001.6
30.5

Appendix 3.

The OWASP Risk Rating Methology,

https://www.owasp.org/index.php/OWASP_Risk_Rating_Methodology

Below is summary of OWASP's factor estimates for calculate the risk factor. The factor should be always to be proportion to own organization's needs and requirements.

Threat Agent Factors

Skill level: How technically skilled is this group of threat agents? Security penetration skills (9), network and programming skills (6), advanced computer user (5), some technical skills (3), no technical skills (1)

Motive: How motivated is this group of threat agents to find and exploit this vulnerability? Low or no reward (1), possible reward (4), high reward (9)

Opportunity: What resources and opportunities are required for this group of threat agents to find and exploit this vulnerability? Full access or expensive resources required (0), special access or resources required (4), some access or resources required (7), no access or resources required (9)

Size: How large is this group of threat agents? Developers (2), system administrators (2), intranet users (4), partners (5), authenticated users (6), anonymous Internet users (9)

Vulnerability Factors

Ease of discovery: How easy is it for this group of threat agents to discover this vulnerability? Practically impossible (1), difficult (3), easy (7), automated tools available (9)

Ease of exploit: How easy is it for this group of threat agents to actually exploit this vulnerability? Theoretical (1), difficult (3), easy (5), automated tools available (9)

Awareness: How well known is this vulnerability to this group of threat agents? Unknown (1), hidden (4), obvious (6), public knowledge (9)

Intrusion detection: How likely is an exploit to be detected? Active detection in application (1), logged and reviewed (3), logged without review (8), not logged (9)

Technical Impact Factors

Loss of confidentiality: How much data could be disclosed and how sensitive is it? Minimal non-sensitive data disclosed (2), minimal critical data disclosed (6), extensive non-sensitive data disclosed (6), extensive critical data disclosed (7), all data disclosed (9)

Loss of integrity: How much data could be corrupted and how damaged is it? Minimal slightly corrupt data (1), minimal seriously corrupt data (3), extensive slightly corrupt data (5), extensive seriously corrupt data (7), all data totally corrupt (9)

Loss of availability: How much service could be lost and how vital is it? Minimal secondary services interrupted (1), minimal primary services interrupted (5), extensive secondary services interrupted (5), extensive primary services interrupted (7), all services completely lost (9)

Loss of accountability: Are the threat agents' actions traceable to an individual? Fully traceable (1), possibly traceable (7), completely anonymous (9)

Business Impact Factors

Financial damage: How much financial damage will result from an exploit? Less than the cost to fix the vulnerability (1), minor effect on annual profit (3), significant effect on annual profit (7), bankruptcy (9)

Reputation damage: Would an exploit result in reputation damage that would harm the business? Minimal damage (1), Loss of major accounts (4), loss of goodwill (5), brand damage (9)

Non-compliance: How much exposure does non-compliance introduce? Minor violation (2), clear violation (5), high profile violation (7)

Privacy violation: How much personally identifiable information could be disclosed? One individual (3), hundreds of people (5), thousands of people (7), millions of people (9)

Appendix 4.

Overall risk severity calculations

Sensitive data exposure, OWASP ref A3: 2017 (OWASP 2017)

Threat agent factor					Vulnerability factors			
Skill level	Motive	Opportunity	Size		Ease of discovery	Ease of exploit	Awareness	Intrusion detection
4	4	2	2		3	3	3	3
Overall likelihood = 3 (Medium)								
Technical impact, CIA					Business impact			
Loss of confidentiality	Loss of integrity	Loss of availability	Loss of accountability		Financial damage	Reputation damage	Non-compliance	Privacy violation
4	4	3	4		3	4	4	4
Overall technical impact = 3,75 (Medium)					Overall business impact = 3,75 (Medium)			

Overall risk severity				
Impact	HIGH	Medium	High	Critical
	MEDIUM	Low	Medium*	High
	LOW	Note	Low	Medium
		LOW	MEDIUM	HIGH
	Likelihood			

Table 1, overall likelihood, technical and business impact

Security misconfiguration, OWASP ref A6: 2017 (OWASP 2017)

Threat agent factor					Vulnerability factors			
Skill level	Motive	Opportunity	Size		Ease of discovery	Ease of exploit	Awareness	Intrusion detection
4	4	4	2		8	3	4	3
Overall likelihood = 4 (Medium)								
Technical impact, CIA					Business impact			
Loss of confidentiality	Loss of integrity	Loss of availability	Loss of accountability		Financial damage	Reputation damage	Non-compliance	Privacy

								violation
3	3	3	3		3	3	3	3
Overall technical impact = 3 (Medium)					Overall business impact = 3 (Medium)			
Overall risk severity								
Impact	HIGH	Medium	High	Critical				
	MEDIUM	Low	Medium*	High				
	LOW	Note	Low	Medium				
		LOW	MEDIUM	HIGH				
	Likelihood							

Table 2, overall likelihood, technical and business impact

Cross-Site Scripting, OWASP ref A7: 2017 (OWASP 2017)

Threat agent factor					Vulnerability factors			
Skill level	Motive	Opportunity	Size		Ease of discovery	Ease of exploit	Awareness	Intrusion detection
5	4	4	2		3	2	5	4
Overall likelihood = 3,625 (Medium)								
Technical impact, CIA					Business impact			
Loss of confidentiality	Loss of integrity	Loss of availability	Loss of accountability		Financial damage	Reputation damage	Non-compliance	Privacy violation
3	3	3	3		3	3	3	3
Overall technical impact = 3 (Medium)					Overall business impact = 3 (Medium)			
Overall risk severity								
Impact	HIGH	Medium	High	Critical				
	MEDIUM	Low	Medium*	High				
	LOW	Note	Low	Medium				
		LOW	MEDIUM	HIGH				
	Likelihood							

Table 3, overall likelihood, technical and business impact

Using component with known vulnerabilities, OWASP ref A9: 2017 (OWASP 2017)

Threat agent factor					Vulnerability factors			
Skill level	Motive	Opportunity	Size		Ease of discovery	Ease of exploit	Awareness	Intrusion detection
6	3	3	2		3	3	4	4
Overall likelihood = 3,5 (Medium)								
Technical impact, CIA					Business impact			
Loss of confidentiality	Loss of integrity	Loss of availability	Loss of accountability		Financial damage	Reputation damage	Non-compliance	Privacy violation
3	3	3	3		3	3	3	3
Overall technical impact = 3 (Medium)					Overall business impact = 3 (Medium)			

Overall risk severity				
Impact	HIGH	Medium	High	Critical
	MEDIUM	Low	Medium*	High
	LOW	Note	Low	Medium
		LOW	MEDIUM	HIGH
	Likelihood			

Table 4, overall likelihood, technical and business impact

Insufficient logging & monitoring, OWASP ref A10: 2017 (OWASP 2017)

Threat agent factor					Vulnerability factors			
Skill level	Motive	Opportunity	Size		Ease of discovery	Ease of exploit	Awareness	Intrusion detection
4	4	4	2		4	3	3	3
Overall likelihood = 3,375 (Medium)								
Technical impact, CIA					Business impact			
Loss of confidentiality	Loss of integrity	Loss of availability	Loss of accountability		Financial damage	Reputation damage	Non-compliance	Privacy violation
3	3	3	3		3	3	3	3
Overall technical impact = 3 (Medium)					Overall business impact = 3 (Medium)			

Overall risk severity				
Impact	HIGH	Medium	High	Critical
	MEDIUM	Low	Medium*	High
	LOW	Note	Low	Medium
		LOW	MEDIUM	HIGH
	Likelihood			

Table 5, overall likelihood, technical and business impact

Manipulating user data in different tenant inside IoT framework

Threat agent factor					Vulnerability factors			
Skill level	Motive	Opportunity	Size		Ease of discovery	Ease of exploit	Awareness	Intrusion detection
5	4	4	2		3	3	2	3
Overall likelihood = 3,25 (Medium)								
Technical impact, CIA					Business impact			
Loss of confidentiality	Loss of integrity	Loss of availability	Loss of accountability		Financial damage	Reputation damage	Non-compliance	Privacy violation
5	4	4	3		5	5	5	4
Overall technical impact = 4(Medium)					Overall business impact = 4,75 (Medium)			

Overall risk severity				
Impact	HIGH	Medium	High	Critical
	MEDIUM	Low	Medium*	High
	LOW	Note	Low	Medium
		LOW	MEDIUM	HIGH
	Likelihood			

Table 6, overall likelihood, technical and business impact

Denial of Service attack against IoT environment

Threat agent factor		Vulnerability factors

Skill level	Motive	Opportunity	Size		Ease of discovery	Ease of exploit	Awareness	Intrusion detection
5	4	5	2		4	5	4	3
Overall likelihood = 4 (Medium)								
Technical impact, CIA					Business impact			
Loss of confidentiality	Loss of integrity	Loss of availability	Loss of accountability		Financial damage	Reputation damage	Non-compliance	Privacy violation
3	3	5	3		3	3	3	3
Overall technical impact = 3,5 (Medium)					Overall business impact = 3 (Medium)			
Overall risk severity								

Impact	HIGH	Medium	High	Critical
	MEDIUM	Low	Medium*	High
	LOW	Note	Low	Medium
		LOW	MEDIUM	HIGH
	Likelihood			

Table 7, overall likelihood, technical and business impact

Man-In-the-Middle attack against IoT environment

Threat agent factor					Vulnerability factors			
Skill level	Motive	Opportunity	Size		Ease of discovery	Ease of exploit	Awareness	Intrusion detection
6	5	4	2		3	4	5	4
Overall likelihood = 4,125 (Medium)								
Technical impact, CIA					Business impact			
Loss of confidentiality	Loss of integrity	Loss of availability	Loss of accountability		Financial damage	Reputation damage	Non-compliance	Privacy violation
6	5	5	4		4	4	4	4
Overall technical impact = 5 (Medium)					Overall business impact = 4 (Medium)			
Overall risk severity								

	HIGH	Medium	High	Critical

Impact	MEDIUM	Low	Medium*	High
	LOW	Note	Low	Medium
		LOW	MEDIUM	HIGH
	Likelihood			

Table 8, overall likelihood, technical and business impact

NMAP findings

Threat agent factor					Vulnerability factors			
Skill level	Motive	Opportunity	Size		Ease of discovery	Ease of exploit	Awareness	Intrusion detection
4	2	3	2		4	4	3	3
Overall likelihood = 3,125 (Medium)								
Technical impact, CIA					Business impact			
Loss of confidentiality	Loss of integrity	Loss of availability	Loss of accountability		Financial damage	Reputation damage	Non-compliance	Privacy violation
3	3	3	3		3	3	3	3
Overall technical impact = 3 (Medium)					Overall business impact = 3 (Medium)			

Overall risk severity				
Impact	HIGH	Medium	High	Critical
	MEDIUM	Low	Medium*	High
	LOW	Note	Low	Medium
		LOW	MEDIUM	HIGH
	Likelihood			

Table 9, overall likelihood, technical and business impact

NESSUS findings

Threat agent factor					Vulnerability factors			
Skill level	Motive	Opportunity	Size		Ease of discovery	Ease of exploit	Awareness	Intrusion detection
6	5	3	2		6	5	5	4

Overall likelihood = 4,5 (Medium)								
Technical impact, CIA					Business impact			
Loss of confidentiality	Loss of integrity	Loss of availability	Loss of accountability		Financial damage	Reputation damage	Non-compliance	Privacy violation
4	4	4	5		4	4	3	4
Overall technical impact = 4,25 (Medium)					Overall business impact = 3,75 (Medium)			

Overall risk severity				
Impact	HIGH	Medium	High	Critical
	MEDIUM	Low	Medium*	High
	LOW	Note	Low	Medium
		LOW	MEDIUM	HIGH
	Likelihood			

Table 10, overall likelihood, technical and business impact