UNIVERSITETET I AGDER

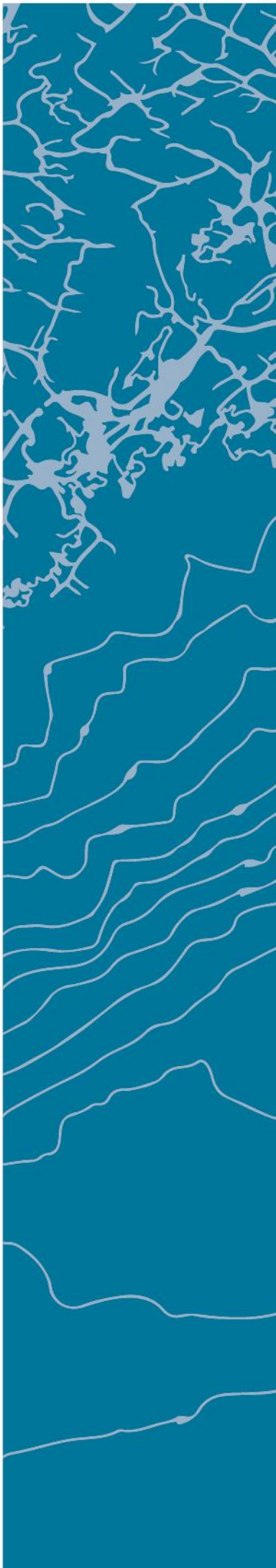
Analysis of security for IoT

A review of architecture and cryptographic algorithms

CHRISTER ERIKSEN HOLE

SUPERVISOR
Nils Ulltveit-Moe

University of Agder, 2017
Faculty of Engineering and Science
Department of Information and Communication Technology



Abstract

Internet of things has been a hot topic for some time now. The huge number of new devices and the limited size of these devices introduce some new problems. Recent event show a trend where IoT devices are being utilized more frequently in cyber-attacks. Home automation is becoming highly attractive and more and more people see the benefits of connecting household objects to the internet. This report will take a closer look at the cryptographic part of securing communication between IoT devices for smart homes as well as reviewing the architecture. A literature review will be performed to highlight previous work and determining the state of IoT. Based on this, an analysis of different security measures will be performed to find the best suited solutions for ensuring integrity, confidentiality and availability. The report focuses on the topics of public key infrastructure, data encryption and network architecture, where different solutions will be discussed to propose set of security measures that are best suited for securing smart homes.

Preface

This report was written as a Master's thesis for the subject IKT590 at University of Agder(UiA), Grimstad, Norway. The subject concludes the master's degree programme of Information and Communication Technology at the faculty of Engineering and Science.

I would like to thank my supervisor Nils Ulltveit-Moe for his guidance and motivation during my work.

Grimstad, Norway
21st May 2017
Christer Eriksen Hole

Contents

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

1 Introduction

The internet as we know it is always evolving, and in recent years an enormous increase in number of devices connected to the internet has occurred. Now, ordinary objects like TVs, watches and smoke detectors are given the feature to connect to the internet. This is what we call the Internet of Things. Interconnection of smart objects or “things” enable them to collect and transmit data. With this comes a wide range of possibilities in almost all fields and industry. The concept of IoT marks a new chapter in the history of the Internet. Cars, watches, cameras and medical equipment can now communicate over the internet. This enables more remote controlling, automation and monitoring of regular objects and processes. While the technology itself might not be brand new, now is the time where we see it be implemented in almost any object to create a network of “things”.

Connecting so many smaller devices to the internet clearly has its benefits in productivity and added features, but also some new problems. That fact that the devices themselves are so different in sizes and uses, hint to a problem we will cover later in this report, namely the lack of standardizations. Another problem which has been the main concern for many, is the lack of security. Security is costly and gives little enjoyment to the general end-user. It is first at the lack of security that users see the value of sufficient security measures. With restricted space comes limited hardware, and thus drawbacks in computational power. Although such devices usually do not need a lot of computational power to do their intended tasks, it becomes an issue when we add security measures like encryption and digital signing [1].

Although there certainly are a lot of IoT devices that are adequately secured, there are still many aspects where there is need for improvements. This is not only a problem for those who chooses to use such devices, but can also affect the general users of the internet. How severe the outcome of a compromised devices is, is highly dependent on the type of device, and what security measures that are implemented to handle this. Within the IoT field, there are multiple sub fields such as home automation, industrial controllers and sensors, smart healthcare and so on. Some of these contain or transmit

vital information, thus the importance of confidentiality is critical. This could be systems in E-health or the electronics in a car. Here, a compromised device or tampered data could have fatal outcomes. Others might not handle such important information, but the control of the device is still crucial to protect from hackers.

As mentioned there are various fields within IoT, and within these fields, different types of devices. This results in a huge number of settings with different requirements and conditions. This means that IoT will have to handle a new spectre of security threats in addition to the known dangers of internet communication.

In this report we will discuss some of the problems with security that arises with the IoT devices. Because IoT is a very vague and wide concept, it covers a lot of different devices which have different uses and ways of operating. This makes it difficult to find one solution which is best suited for all IoT devices. We will therefore take a closer look at home automation and security measures which can be applied to such devices. Also, we will evaluate how a network of such devices is best structured. The findings in this review will then be discussed to propose what solutions should be used. By looking at these topics we hope to shed some light on the need for improved security in IoT devices, and also propose solutions to help the work with creating common standards for IoT.

1.1 Background

Everyday tools are augmented to be connected to the internet. Objects like smoke detectors, thermostats and lightbulbs can now be monitored and controlled for added functionality and convenience**.** Home automation is becoming highly attractive and more and more people see the benefits of connecting household objects to the internet. New products are constantly entering the market, providing added security, comfort and convenience. Common features include: monitor security camera, remote control of lights and appliances and energy saving.

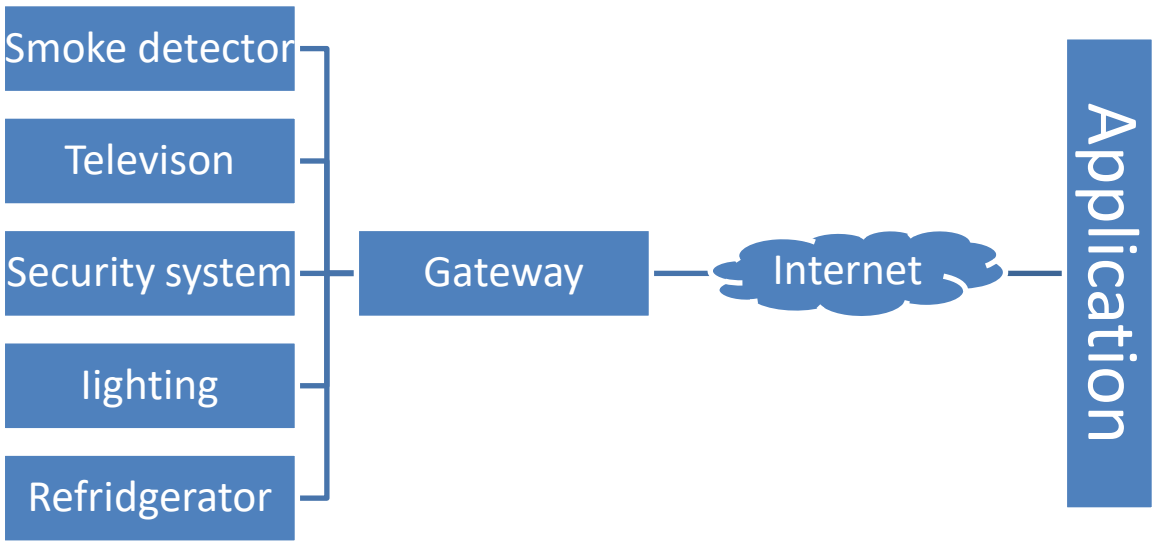


Figure 1 - Smart home example

The above model shows an example of a typical IoT environment for home automation. Numerous sensors and controlling devices are connected to one or more applications over the Internet via a gateway device.

Increased size is an example of something we wish to avoid, even when we are adding functionality. This is especially noticeable in the home-automation area, where convenience and aesthetics are highly valued attributes. Even though objects like smoke

detectors and baby monitors are given features to be remotely managed, we do not want them to be any bigger than before. And although we are willing to pay a little extra for this, the price should still be within a reasonable limit relative to a “non-smart” version. Due to these limitations, several problems occur with the IoT devices. Size limitations and cost efficiency makes for limited processing power, and manufacturers must make some compromises.

IoT face many of the same problems as regular computers, but not all the problems can be solved in the same way. Many devices are built to have long lifetime and to operate without much human intervention after deployment. The hardware of the devices is also quite varying. Some of the devices are able to use the standard well-known algorithms for securing communications. But for devices with limited resources this might not be practical in terms of time and energy consumption. For this reason, some algorithms might be better suited than others in different situations. The problem is to know what algorithms will perform best under specific circumstances.

With the rapid development in new devices and their way of operating, it has been a difficult task to keep up standard ways of implementation and communication for IoT devices. Also, the definition of IoT itself has been proven to be quite difficult. There are numerous ways of describing the concept. One definition, from IEEE, goes as follows: “A network of items—each embedded with sensors—which are connected to the Internet” [2]. As we can see here, the definition is not so specific, but it captures the essence of what IoT is. “Things” can be interpreted to mean basically whatever you would like it to mean. And also, it is used in so many different settings. With this, the term covers an endless number of different types of “devices”, with different uses and ways of operating. Together with the definition comes a three-layered model of the architecture of IoT.

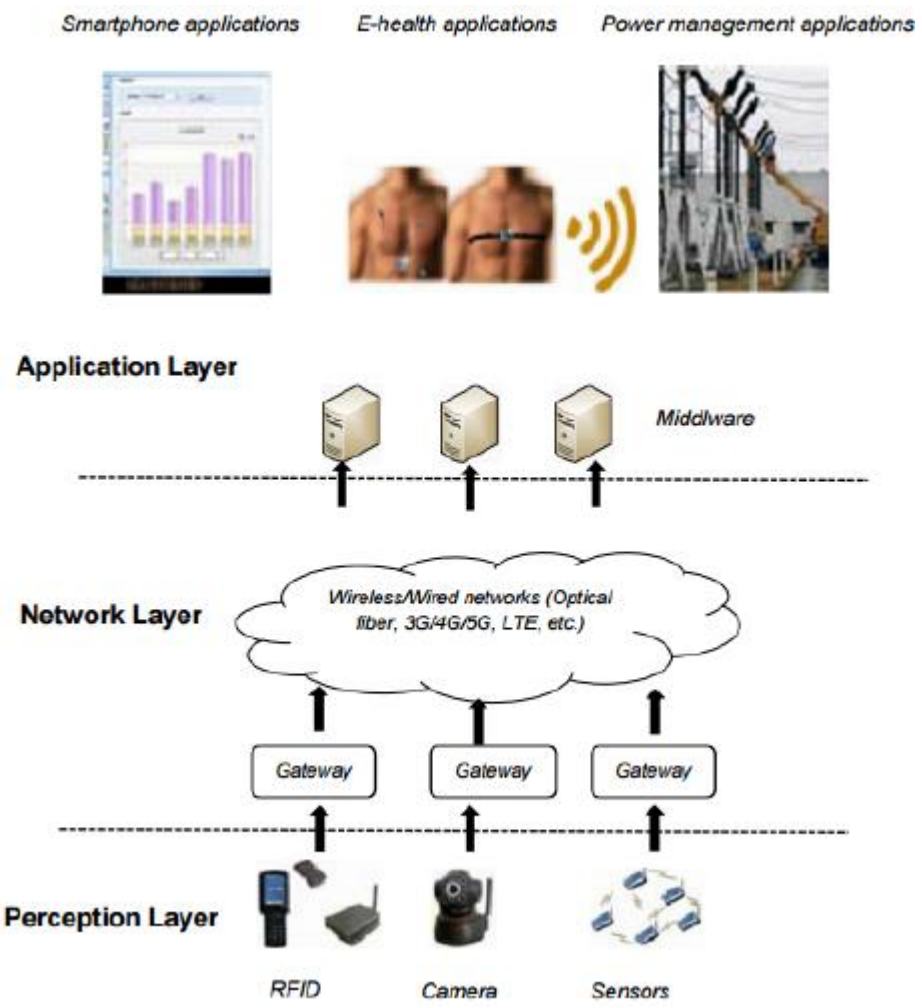


Figure 2 - 3-layered model of IoT[3 Fig. 3]

This shows a representation of the 3-layered model with examples of technologies and protocols that are used in each layer. The layers in this model are: Application layer, network layer and perception layer [3]. Here, the perception layer consists of the physical tasks of the devices, like sensing or controlling. This is also where the information is transformed to digital form. The network layer is responsible for transportation of the information between the perception layer and application layer. This layer uses protocols like Bluetooth, Wi-Fi and ZigBee. Application layer is what we as users see of the IoT process. This is where the information is presented and stored in various IoT applications.

1.2 Problem statement

As the last years have shown, attacks that utilize insecure IoT devices are becoming a common trend. It is clear, that in general the security of IoT devices is not good enough. As the number of such devices increase, it is safe to assume that IoT will continue to be a target and an attack vector for years to come. The growth is not only in terms of number of devices, but also in terms of various types and formats. With such variety in use, size and available resources, standardization becomes a difficult task.

More and more people are seeing the benefits that these smart objects bring, and are now willing to pay for it. While some smart products, like smart TVs, have been around for a while, more and more smart object are finding their way into the regular household. To handle this, home automation systems must make use of new and existing security measures. The system must be secure and scalable, to safely manage the increasing number of internet connected devices in your home. The problem is that the process of standardization is not keeping up with the rate of development. The result is a variety of solutions with different technologies, and some without sufficient security.

To find and propose better solutions for the security in smart home networks, this report will consist of a literature review and an analysis different security measures and solutions for the following topics:

- Public key infrastructure
- Data encryption
- Network architecture

.

The system needs to be scalable, and be able to handle a huge number of off- and on boarding of devices in a secure manner. It also needs to uphold the fundamental elements of AIC triad; Availability, Integrity and Confidentiality. With this we mean that the data and devices must be available to the user at all times. The user must also be sure that the data has not been tampered with and that it comes from legitimate sources. And all sensitive information must be handled in a way to protect the privacy of the user.

1.3 Security Assumptions

Given the security flaws in many IoT devices there is need for improvement [4] [5]. With the vast differences in need for security for all the types of devices it is difficult to specify requirements that hold for all. Providing a set of security measures that fits all possible scenarios is therefore not practical. Limitations like computational power calls for use of lightweight algorithms. While these are necessary for devices with extreme hardware limitations, other kinds will not have any benefit of this implementation and might have need for a stronger solution.

This chapter will describe a typical setup for a smart home and define some assumptions and requirements for a possible solution. The analysis of this report will assume the following set of devices as a baseline for the evaluation of different solutions:

Figure 3 - Assumed model of smart home

The network consists of IoT devices that are sold commercially and are popular among those who wish to implement home automation elements. The devices in this model are a smoke detector, a smart watch, an alarm system and a thermostat, are devices that represents IoT devices with minor limitations in hardware. The hardware capabilities of

such devises are comparable to the capabilities of other commonly used IoT devices for smart homes and are therefore a good representative selection for the field. In this setup, the devices are connected to the Internet via the home router. The user can connect to, and manage the devices with applications from his mobile phone.

To maintain a secure communication between the nodes, encryption is a must. Cryptographic keys must then be safely transferred between the parties to enable encryption and decryption. How the data is encrypted is also important to look at. The algorithm must not be so heavy that a device with limited computational power cannot run it. And it must also have a sufficient level of security to ensure confidentiality for the users. The structure of the communication itself is also necessary to review. The network needs to be structured in a way that supports a massive increase of number of devices, and it needs a secure way of discovering and managing devices.

1.4 Literature review

IoT is a field that has gained increasing momentum in the later years. With the components becoming cheaper, IoT devices for home automation are becoming more and more popular. Consequently, there is a lot of research being done in the field. In this report we will look at some of the trends in IoT and what direction things are going. IoT is a concept that with different definition and descriptions depending on who you ask. Because of this, it is natural that there are different opinions on what solutions are the best.

With a wide variety of communication technologies in use with IoT. Wi-Fi, Bluetooth, ZigBee and GSM/3G/4G all have their benefits in specific use-cases. But this becomes a problem when maintaining a network of different devices. Routers and gateways solve the problem with protocol translation, but this makes the communication path more complex than it needs to be. And more middleware operations enable more possible exploits along the path. To solve this, we are moving towards all-IP based networks with IPv6 and 5G.

There have been developed protocols that try to standardize communication on the network layer to provide end-to-end IP connection. IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs), IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL), are two standards being developed and maintained by IETF [6] [7]. These standards aim to enable the use of IPv6 over Low power and Lossy Networks(LLN) such as IEEE 802.15.4.

A report from McAfee predicts that IoT will become an important target for cyber criminals in the near future [8]. What types of attacks and how cyber criminals will make money on it is still unclear. There are numerous vulnerabilities and exploits related to IoT in general. But hackers must be able to perform them on a large scale to be able to make profit of it.

Last year, the number ransomware attacks and ransomware families had a massive increase. While this was mainly focused on regular computers analysts believe it will migrate to IoT soon. This has proven to be an easy way for hackers to make money, and given the sheer numbers and possible exploits it is reasonable to assume that we will see much more of this targeted to IoT devices.

The report also predicts that IoT also will be a target for hacktivist. Controlling and sensor devices for critical infrastructure and manufacturing systems in bigger companies could be a major target for political and environmental activists who are willing to use more extreme means to accomplish their goal. This could be an even more serious threat than cyber criminals as criminals want to make money on their actions and are thus usually not so destructive. Activist who go as far as to use criminal acts to help their cause, could be much more dangerous. Disrupting a voting poll or a security sensor in critical infrastructure could have catastrophic effects.

Rather than targeting single devices, new attacks will likely emerge that target a higher plane in the IoT network infrastructure. Targeting IoT devices by themselves will in most cases not be very profitable for attackers. By targeting the gateways and data collectors instead, attackers could gain control of huge amount of data or devices at once. We will likely also see a development in these types of devices to enable monitoring of IoT devices themselves for security reasons. IDS and behavioral monitoring will be a way to enable security measures for devices that have limited hardware resource.

Another topic, which is one of the reasons why there is so many security flaws tied to IoT. More and more products are being IP-enabled, meaning they can connect to the internet. Manufacturers with no prior experience with internet connected devices, are now starting to develop products that are going to be exposed to the internet. This will result in some of them making mistakes that more experienced companies would not do.

As consumers realize that more and more of the data they are creating is used by various vendors, privacy protection will become more desired by consumers. One way to

ensure user privacy is to make use of the blockchain technology. The Blockchain technology has previously been used in the cryptocurrency Bitcoin with great success, and uses a peer-to-peer model rather than the classic server-client infrastructure. This technology will still provide the vendors with valuable data, but it will not be identifiable to a user or device to protect the privacy of the consumer.

Another report of predictions of the threat picture for the years to come by Fortinet [9]. In this report they expect the botnets like Mirai to grow and become more sophisticated. It is speculated that the attacks we saw in the later part of 2016 could just be a test to see the abilities of an IoT powered botnet. Now that the capabilities are demonstrated we will see these botnets being used in more advanced attacks.

The previous report by McAfee suggested a that inexperience of manufacturers in internet communication could be a reason for the many security flaws related to IoT. This report highlights the same issue, but also points to another impact of the problem. Manufacturers outsource the development of the components that are responsible for the network communication to other technology companies. An example of this could be a company that produces smoke detectors. Since smoke detectors previously had no need to be connected to the internet, such a company will have little to no experience in securing the hardware and communication of the product. This is then outsourced to a company which has experience such devices. The problem is when multiple vendors have multiple product using components from the same company. If vulnerabilities are found in the components from that company, it will affect a huge number of devices.

Another organization that has done some valuable work in this field is The Open Web Application Security Project (OWASP). The OWASP Internet of Things Project works to provide information about security related topics regarding IoT [10]. The project offers information about vulnerabilities, attack surfaces, as well as general guidelines for improving security when deploying IoT devices.

A report on smart home systems proposes an approach on how to implement better security in home automation with internet of things [11]. The system is based on a centralized model where the nodes are connected and managed via a home gateway. The devices are connected to the gateway over Wi-Fi. The base of this implementation is the framework called AllJoyn [12]. The framework enables development of apps for easy management and security for IoT.

Streembit is another approach to the same problem [13]. With this solution, the developers also want to provide security for IoT networks, but use a different approach with the architectural structure. Nodes in the system are connected in a decentralized peer-to-peer network. The developers view this model as the best suited to handle IoT devices because of the scalability, easy on-boarding and security it delivers.

1.4.1 Vulnerabilites

While the new devices come with endless uses and improvements to industries and at home, there are also a lot of challenges that occur. Compromises have to be made to make them both functional and sufficiently secure. Internet of things devices and communication is susceptible to many of the same vulnerabilities as regular computers and networks are. Here we will list some of the major threats that IoT devices and networks face. The threats will be divided into 3 sections derived from the 3-layered model of IoT architecture proposed by IEEE [2]. Physical threats include vulnerabilities and attacks that target the physical part of the devices or the device's firmware i.e. correlated to the perception layer. The threats under the network section will be similar to many of the attack vectors we already see on the internet as many IoT devices utilize the same protocol and means of communication as regular computers. The network layer handles the transmitting of data between the devices. This is also the main focus area of this report. The last section contains threats related to the applications and middleware that is used in relation with IoT. These attacks target application that are controlling, monitoring and presenting data from end-nodes.

Physical

- Tampering of the physical device itself. This could be removing parts of the devices, like SD-cards or removing the whole devices. Devices with physical interfaces could also be accessed by attackers. Keeping devices hidden or locked away will prevent unauthorized access in such ways.

- Outdated firmware is been shown to be a major problem with routers and other IoT devices [14]. Despite the solution being both easy and obvious. Simply update the devices regularly. A study from Ubuntu [15] surveying 2000 users, shows that an alarming amount of user don't apply updates as they are released, or even make use of automatic updating. In some cases, the users believe the responsibility to be either the manufacturer or software developer or just don't do it out of simple laziness. But with IoT, firmware updates can be tricky. Many of the devices are deployed with the expectation that they would operate for long periods of time with human interactions. Failure to apply critical updates can result in hackers exploiting known issues that could have been patched. Last year thousands of router running outdated firmware were infected by a worm that exploited an old vulnerability [16].

- Differential power analysis is a type of side-channel attack. An attacker analyses the power consumption of the device over time. The power trace can then be analysed to find patterns for computing values during the steps of for example an encryption algorithm. This can then be used further to find cryptographic keys. This kind of attack requires the attacker to have physical access to the devices.

- Pre-computed cryptographic keys are quite common in IoT devices [14]. These keys are implemented in the firmware of the devices before shipping. The purpose of these keys is to work as a seed in other cryptographic algorithms. The problem, however, is that a lot of these keys are the same or not random enough. This results in an attacker being able to guess or even know what cryptographic key a device is using.

Network

- A lot of IoT devices transmit data even without any security measures to protect the communication.

- Man in the Middle Attack (MITM). In a man in the middle attack, the attacker intercepts communication between two entities in a communication network. If performed well, neither of the parties will be aware that the communication is transferred between them via a third party. A typical example of a MITM attack is when an attacker intercepts the communication between a client and a server. The attacker pretends to be the server that the client is trying to connect to. There are various methods to do this, like ARP Poisoning or Sybil attack. The client then sends the request to the attacker which forwards it to the server. The response from the server is then forwarded the same way in the other direction. This allows the attacker to not only see, but also make changes in the data which is transferred. The best way of protecting against MITM attacks is digital certificates. Digital certificates let devices identify themselves to ensure that the other end of the communication is legitimate.

- Sybil attacks are performed by use of fake identities or devices in a network. One or more malicious nodes acts as multiple nodes, to influence, control or spy on other legitimate nodes [17].

- DDOS This is not a weakness like the others listed here. The vulnerability here is not a single device being targeted by such an attack. The danger is that using exploits and weaknesses like the ones that are mentioned here, attackers could infect thousands of devices to perform massive DDoS attacks. Recent events have shown that IoT devices can be used to execute powerful DDoS-attacks. The attackers make use of botnets, which are networks of devices that are infected

with a small malicious software. These infected devices usually stay dormant until it receives commands from a command and control centre (C&C). The malicious code is small, hidden and therefore difficult to detect, aside from the communication to the C&C. In 2016 the Mirai botnet was the primary actor in two record breaking DDoS attacks. The first attack happened in September, targeting krebsonsecurity.com, a security blog by Brian Krebs [18]. The second attack targeted the infrastructure of DNS provider Dyn in October the same year [19]. This one lead to services from companies like Amazon, Spotify and many others being unavailable for several hours. The infection method of the Mirai botnet is to scan the internet of connected devices. A database of default usernames and passwords is then used to attempt to gain access to the scanned devices. Many such devices still use the default username and password combination from the manufacturer, making it easy for the attacker to gain access. This just shows that even though it might not seem like the device need a high level of security at first glance, it is crucial to hinder unwanted access to them to avoid attacks like these.

- Exposing a device to the internet enables you to access and control it remotely. While this is a major benefit it can also be used against you.

Application

While this report is mainly focused on the network layer of IoT, all aspects has to be worked on to provide sufficient level of security. An analysis of a newer, programmable smart home application highlights weaknesses in the application layer of IoT [20]. Multiple design flaws in applications for smart homes were uncovered, and then used to perform attacks to retrieve pin codes for locks, disable vacation mode and raise false alarm in smart home security systems. Some of the vulnerabilities found included Overpriviligation. Overprivileged default users seems to be a common problem with IoT devices. Because of the SmartThings framework, which was used in this case, applications can get more capabilities than they require. An example showed that an

auto-lock application which required the ability to lock the devices, also automatically obtained the ability to unlock it, even if it didn’t use it.

1.4.2 Obstacles

There are many known vulnerabilities related to internet of things. Some of them have solutions that could be used with a regular computer network. With IoT, there are some limitation in the fundamental way the devices work that makes it difficult to handle them. While a lot of the vulnerabilities mentioned earlier can be fixed by more security focused development, these obstacles are the things that make implementing IoT in a secure manner difficult. To further develop IoT for the future, security solutions must take this into account and find ways to handle them.

1.4.2.1 Size

Size poses a challenge in the design of IoT devices. In all areas there are devices which has a need to be as small as possible. This leaves little room for components, and as a result, devices have limitations in how fast they are and how much storage they have. Manufacturer must then choose wether or not they want to use more expensive, size-efficient hardware. Depending of the type of devices, this can drive the price up to levels above what the consumer regards as a reasonable price. A lot of the devices released in todays market only have hardware that is sufficent to perform their primary tasks.

1.4.2.2 Key Handling

Public key infrastructure is a set of actions and policies that enable distribution and storage of cryptographic keys. These are used for encryption and identification of devices to ensure secure communcation over a network. Cryptographic keys in public key infrastructures are asymmetric. This means that each part in the communcation has its own key as opposed to symmetric algorithms where both encryption and decryption key is the same. Operations are performed by use of a combination of private and public keys. To make use of symmetric algorithms, the involved nodes have to agree on a common secret key to handle decryption and encryption. To make sure that no third party actor is able to access this key, the agreement has to be done in a secure manner. Because communication channels like WiFi are succeptable to traffic sniffing, methods like RSA or Diffie-Hellman should be used. Failure to do so can result in attackers being

able to view encrypted data and also digitally sign transmission to act as a legitimate sender. This can be difficult in environments with limited computational power as shown above.

Handling of asymmetric cryptographic keys is very important but these keys are often very large. Use of heavy algorithms such as RSA for key exchange, digital signing and trust certificates will then become too slow and impractical to use [48].

1.4.2.3 Need for security

The need for security is not the same in all devices. Some devices process sensitive data and have a strong need for confidentiality. Data which identifies the user or information like pins for security systems must be encrypted to protect the user. Others devices, that control appliences in your home, are dependent on correct information from sensors. These devices will then have a high need for integrety to ensure that the data is correct. A coffee machine which can be controlled over the internet might not directly have a high need of security as it does not process any important data or control critical parts of you home. But in the case that someone gains access to this devices it could be used further as a spring board to launch an attack on the rest of your network.

1.4.2.4 Battery powered devices

With smaller and sometimes portable devices, it is quite common for them to be battery powered. While the power provided by the battery may be sufficent to do its intended task, computing keys and encrypting data can sometimes draw too much power. Many IoT devices are deployed with the intention that they operate for longer periods of time without much human interaction. If the security measures are too demanding in terms of power usage, the battery will run empty more frequently and make the device impractical to use.

1.4.2.5 Scalability

This obstacle occurs when joining more “things” together. As we are seeing an increase in the number of devices, it is important to have a system that is able to handle frequent off- and onboarding of devices. For the home automation field, a user must be able to easily connect and disconnect his devices without needing to upgrade infrastructure devices.

1.5 Problem solution

Finding the optimal solutions for securing smart homes requires a knowledge of the state of IoT. A literature study has been done to review the current situation on some aspects of security for IoT, and look at some new solutions that are being developed.

Based the problem statement and security assumptions, this report will solve the problem by answering the following questions:

- Is Elliptic Curve Cryptography more suitable for smart home devices than RSA?
- Should the ultra-lightweight encryption algorithm PRESENT replace AES in such a setting?
- Which type of network architecture is best suited for the future of IoT? Centralized client-server, or decentralized peer-to-peer?

At first, the different solutions will be described to highlight their properties and how they work. The solutions in each of the topics will then be compared to each other to analyse their strengths and weaknesses. The findings will be discussed, to answer the questions above and with regards to the use-case.

The final result will be a proposal of a set of solutions that secures the communication in home automation and ensures availability, integrity and confidentiality. The choices should be made by considering both the current state of smart home devices, as well as envisions of the future of IoT.

1.6 Report outline

The rest of the report will be structured as follows. Chapter 2 contains Theoretical background where topics that will be analysed further in the report will be explained. Section 3 will be about differentiating the possible solutions. What are their pros and cons, and when are they most beneficial to use. This knowledge will then be used to compare them to each other. Findings from the comparison will be used in a discussion in section 4 which will lead to the conclusion of the report in section 5. At the end, chapter 6 will suggest possible further work that builds on the discussed topics in this report.

2 Theoretical background

As a technology, IoT is not really something new. Sensor networks and Machine-to-machine communication has been around for years. In recent years, the concept has become more wide-spread as the number of devices increases. The number of devices connected to the internet is expected to grow exponentially towards 50 billion devices by 2020. Additionally it is expected to reach 20 billion by the end of 2017 [21]. This will result in an average of around 6 devices per person worldwide.

This chapter will highlight the different solutions that will be reviewed later in the report. Some of them are the currently used standards in regular computer networks. These have shown to be highly valuable and have then been implemented for IoT purposes as well. The others will be solutions that serve the same purpose but may be better suited for use when operating with IoT devices. We see that the security of IoT poses a lot of challenges. Many of these can be directly tied to algorithms for encryption requiring too much rescourses. But adding more powerful hardware to such limited space can be difficult without driving the cost of the product up to an unreasonable level. Therefore, we should look to newer ways of doing things in terms of algorithms. Lightweight cryptographic algortims are already being developed and some are even in use.

2.1 Public key

Asymmetric, or public key systems uses a pair of keys for each node. A public key which is published and a secret one that only the user knows. The key pair is derived from a large random number to make it difficult for an attacker to find out the values of the keys. Such keys have the property that a user can then encrypt data with the public key of the receiver, and the receiver then uses his private key to decrypt. Another usage is that the sender can use its private key to ensure the recipient of his identity. These actions are usually called encryption and digital signing respectively.

The security of this algorithm is based on the problem of factoring a product of two large prime numbers. It is considered a hard problem and there has yet to be found a solution which can solve it in a reasonable time, to intercept the communication. Although it is a secure way to encrypt and transfer data, the algorithm is to slow to be used on all communication. It is usually just used to transmit keys to a symmetric algorithm like AES, which is then used to encrypt the data.

2.1.1 RSA

RSA is an asymmetric algorithm which is widely used today for creating a secure way to transmit. It is one of the first public-key cryptosystems to be used. As an asymmetric algorithm, it uses key pairs for encryption and decryption. RSA is embedded in the SSL/TLS protocol which is used for secure communication. NIST recommends key sizes of at least 2048 bits for use before 2030 [22]. After that at least 3072 should be used. The currently most used key sizes are now 2048 and 4096 bits.

The keys that used in RSA are generated in 5 steps [23]:

1. Choose two large, distinct primes *p* and *q*.
2. A value *n*, is calculated as a product of *p* and *q*. *n* = *pq*
3. The totient of *n,* $\phi(n) = (p-1)(q-1)$*,* is calculated.
4. A key, *e* is a chosen integer which is coprime to $\phi(n)$. This can be any integer in the range $[3, \phi(n))$, but usually *65537* is chosen because it is and adequately high number that often fulfils the condition of being coprime to $\phi(n)$.
5. The other key is then determined by finding the modular multiplicative inverse of *e* modulo $\phi(n)$. This is done using $d \equiv e^{-1} \, (mod \, \phi(n))$

The fact that encryption key *e* can be chosen, and often to the same number, makes it not random at all. While this might seem like a bad idea for a cryptographic key, it is not a problem. This is the public key, so it will be publicly available anyway. Also, it is not possible to efficiently determine the private key from the public key. This is called the discrete logarithm problem, and is what RSA builds it security on. To find *e* from *d* you need $\phi(n)$, and to find this you need the prime factors of *n*. Since *n* is composed of large primes, this is hard to find.

Encryption and decryption in RSA is performed by raising the message to the power of the keys. In other words, to create a cipher text *c* we perform:

$$m^e \ (mod\ n) = c$$

Equations 1 - Encryption RSA

Since *d* is the multiplicative inverse of *e* with respect to $\phi(n)$, we get have:

$$e * d = 1 \ (mod\ \phi(n))$$

Because of this we can decrypt *c* by raising to key as such:

$$c^d = (m^e)^d = m^{e*d} = m^1 \left(mod\ \phi(n) \right) = m$$

Equations 2 - Decryption RSA

As we see, decryption is done by raising the cipher text to the other key. By doing this we multiply the exponents which gives us the original message. Because of the operation is commutative we can do this both ways.

$$m^{e*d} = \ m^{d*e}$$

The first application of RSA enables the sender to encrypt a message with the recipients public key and the receiver to decrypt it with the private key. With the property shown above, RSA also has another use. If the sender decrypts with his private key, anyone can decrypt the message with the public key. This is used in digital signing which ensures the receiver that the message originated from the holder of the private key.

A full proof can be found here [23].

2.1.2 ECC

Elliptic curve cryptography is one of the solutions which can replace RSA. Elliptic curve cryptography is performed by doing operations on an elliptic curve over finite fields. Studies have shown that decryption and digital signing can be done significantly faster with ECC than with RSA, and therefor require less computational power [48]. It is more favorable because it requires a smaller key size to achieve the same level of security as long as the parameters are chosen correctly. This means it can do the same job, but requires less power. This is an excellent attribute, seeing as so many of the IoT devices have limitations in hardware.

The functions of elliptic curve cryptography are defined as point operations on an elliptic curve, $y2 = x3 + ax + b$, where $4a3 + 27b2 \neq 0$. *a* and *b* are variables, and different values gives different curves. Usually an elliptic curve used for cryptography would look something like this.

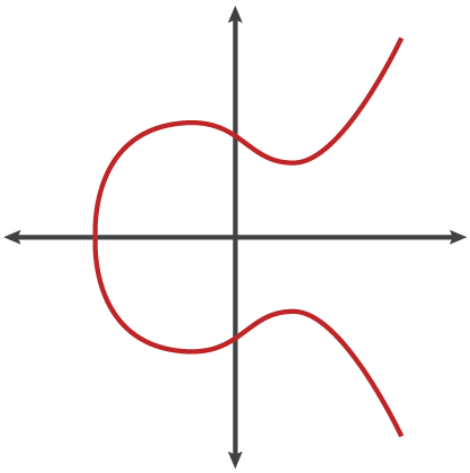


Figure 4 - Elliptic curve [21]

The basic operations with the elliptic curve is point addition and point doubling. Point addition is done by drawing a line between the two points. The line intersects in some third point along the curve. The sums of these two points is the inverse of the intersection point. The inverse point is the reflection with respect to the x-axis.

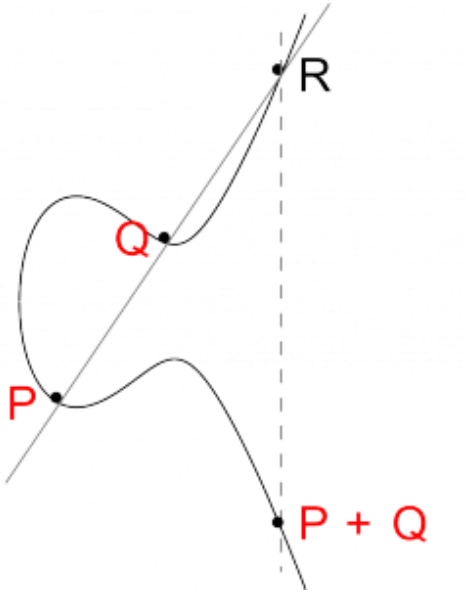


Figure 5 - Point addition on an elliptic curve [24]

Point doubling is essentially the same operation as the point addition. In this case, we only have one point, but in this case the line is drawn as a tangent to that point. The result of the doubling is again the inverse of the intersection point.

With these two operations, we can perform point multiplication. This is the main operation used in Elliptic curve cryptography. In point multiplication, a point *P* is multiplied with a scalar *k* to get a new point *Q* on the elliptic curve. This is where the discrete logarithm problem, and the security of ECC comes in. The discrete logarithm problem is in this case defined as the problem of finding *k* from the equation $kP = Q$ given *P and Q*.

Point multiplication can be performed in various ways, but the simplest one is an algorithm called “double and add”. This method uses repeated point additions and doublings. Finding $kP = Q$, where $k = 13$, we first find the binary expansion of 13 which is 1101. Then the algorithm iterates through each digit and doubles for each digit and adds *P* if there is a 1.

$$Q = k \cdot P = 13 \cdot P = 2\big(2(2P + P)\big) + P$$

While this example is using real numbers for sake of simplicity, elliptic curves in cryptography are defined over finite fields. The keys are generated by choosing a random number less than the order of the curve to be the private key *d***.** The public key *Q* is a point on the curve, generated by multiplying the private key with a generating point. This gives the key pair (d, Q). With this, communicating nodes can perform 2 actions; key exchange (Elliptic curve Diffie-Hellman key exchange) and digital signing (Elliptic curve Digital Signing).

The purpose of the key exchange is to agree on a common secret key, over an insecure channel. Each party performs point multiplication with his private key and the other party's public key.

A computes: $A_k = d_a * Q_b$

B computes: $B_k = d_b * Q_a$

Since the public keys are generated by a generating point *G,* we have:

$$d_a * Q_b = d_a * d_b * G = d_b * d_a * G = d_b * Q_a$$

This implies that $A_k = B_k$. A and B now have the coordinates to a point on the curve that only they know, and this can be used as a secret key.

2.2 Symmetric algorithms

In symmetric algorithms, the key used for encryption and decryption is the same. Therefore, it is necessary for the sender and receiver to obtain a common secret key which only they know about. This is often done by using the RSA or Diffie-Hellman key exchange.

2.2.1 AES

AES is a symmetric block cipher [25]. This algorithm has taken over after DES and Triple DES and become the default use for encryption of data. Today it is used in many protocols and application that offer encryptions, such as the SSL/TLS protocol [26]. Like mentioned above it is used to encrypt larger bulks of data to achieve confidentiality. AES encrypts blocks of 128 bits at a time, with keys of 128, 192 or 256 bits. Depending on the key length, AES performs encryption with 4-bit transformation in 10, 12 or 14 rounds.

As of now, there has not been proven any attack that can break AES. Various attacks a have been demonstrated but none of them are threatening the use of AES as they are either to computationally demanding or not much faster than brute-force. Side-channel attacks like differential power analysis is possible [27]. But these do not target the mathematical properties of AES, but rather a weakness in the way it is implemented implementation.

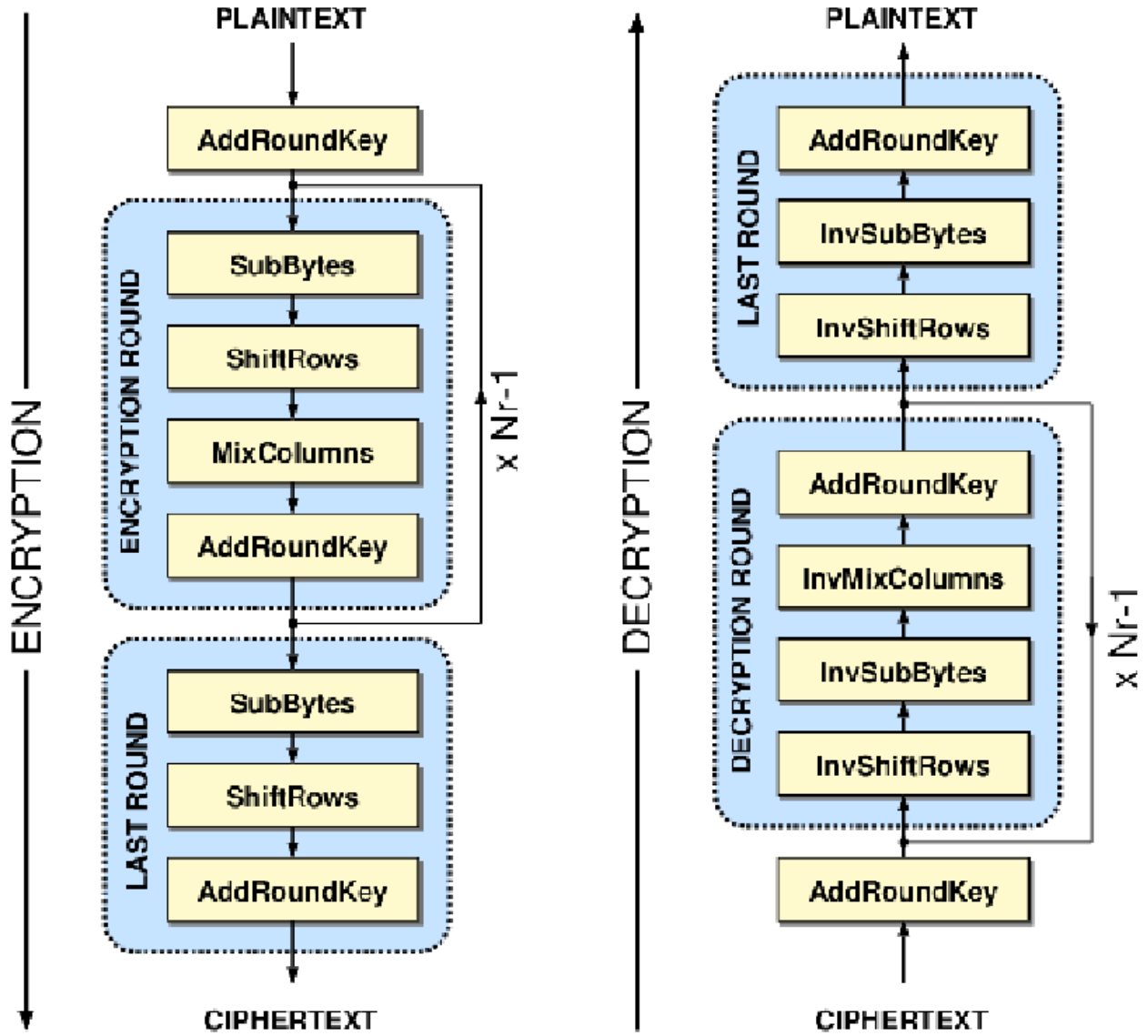


Figure 6 - High-level structure of AES [28]

The picture shows the operations done in decryption and encryption with AES. The encryption consists of 4 functions: AddRoundKey, SubBytes, ShiftRows and MixColumns. When encrypting the process starts with adding the round key. The 4 functions are then done repeatedly for a number of times depending on the length of the key. The last of these rounds are then done without the MixColumns function, and the ciphertext is produced. Decryption is done the in the same way but with the inverse of the functions. A more detailed explanation of the functions follows below.

As a block cipher, AES operates on blocks of 128 bits at the time. These are structured in a matrix of 4x4 bytes. This matrix is called the state. Before the encryption starts, a key expansion is performed. Information on how this is performed can be found here [29].

2.2.1.1 AddRoundKey

This is the first function of both encryption and decryption. Here, each of the 16 bytes of the state is XORed with 16 bytes of the expanded key. The first iteration uses bytes 1 – 16 from the expansion key. The next time 17-32 is used, to never use the same byte twice.

2.2.1.2 SubBytes

The next step is to substitute each byte in the state with a corresponding value according to the following S-box lookup table. The values are shown in hexadecimal. One byte is represented by two hexadecimal digits. The new value is found by matching the first and second digit to the corresponding column and row.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

Table 1 - AES S-box lookup table [30]

2.2.1.3 ShiftRows

This is where the matrix format of the state is important. In this function, each row is shifted to the left. The number of rotations to the left is determined by the row number. Row 0 is shifted 0 times to the left (stays the same). Row 1 is shifted 1 position to the left, and so on. To illustrate this an example of a state with values 1 – 16 would look like this before and after ShiftRows.

Before

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

After

1	2	3	4
6	7	8	5
11	12	9	10
16	13	14	15

2.2.1.4 MixColumn

In the mix column function, the state matrix is multiplied with a matrix with values from 1 to 3 as seen below. This matrix is a representation of a polynomial $a(x) = \{03\}x^3 + \{01\}x^2 + \{01\}x + \{02\}$. The result gives a new matrix *r*, which is the new state after the MixColumns.

$$\begin{pmatrix} s_1 & s_5 & s_9 & s_{13} \\ s_2 & s_6 & s_{10} & s_{14} \\ s_3 & s_7 & s_{11} & s_{15} \\ s_4 & s_8 & s_{12} & s_{16} \end{pmatrix} * \begin{pmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{pmatrix} = \begin{pmatrix} r_1 & r_5 & r_9 & r_{13} \\ r_2 & r_6 & r_{10} & r_{14} \\ r_3 & r_7 & r_{11} & r_{15} \\ r_4 & r_8 & r_{12} & r_{16} \end{pmatrix}$$

Equations 3 - MixColumns operation example

The operation is matrix multiplication, where the sum of the products in the first column and row form the first value in the result. The first operation would look like this:

$$(s_1 * 2) + (s_2 * 3) + (s_3 * 1) + (s_4 * 1) = r_1$$

Equations 4 - MixColumns operation, first value of the new state

To make the algorithm easier for hardware implementation, operations are done over the finite field $GF(2^8)$. The values are converted to an 8-bit binary representation. There are then 3 multiplications to define. Multiplication by 1 is the same. Multiplication by 2 can be done by shifting the number to the left. If the leftmost bit is 1 before shifting, we get an overflow and must reduce it by the modulo of the field.

$$A4_{16} = 1010\ 0100_2$$
$$1010\ 0100 \ll 1 = 0100\ 1000$$
$$0100\ 1000 \oplus 0001\ 1011 = 0101\ 0011$$

The example shows multiplication with 2 where the leftmost bit is one. The result of the shift is then XORed with $1B_{16}$, where 1B is the hexadecimal representation of the irreducible polynomial of the field.

To multiply with 3, we use the same method. Then we XOR the result with the original value because $3 = 2 \oplus 1$. So we have:

$$3 * x = (2 \oplus 1) * x = (2 * x) \oplus x$$

Now that the multiplication is defined, the addition in Equations 4 must be done. Since the operations are performed over $GF(2^8)$, addition is done by XOR. This equation gives the result for one of the values in the result matrix. The operation must then be done 15 more times. The following iterations perform the operation on column 2 - row 1, column 3 – row 1, and so on.

2.2.1.5 Last round

The last round is done the same as the others with the exception that MixColumns is not performed in this round. It is believed that the MixColumns in the last round does not have any effect on the security of the cipher, and is therefore not used. Some studies do however show that the absence of the last MixColumns reduce the time complexity of known attacks against the cipher [31].

2.2.2 PRESENT

PRESENT is another block cipher like AES, but this one is design to be used with limited computational power available [32]. Thus, it is considered an ultra-lightweight algorithm. It is based on the AES and has very similar operations. PRESENT was standardized by ISO/IEC in 2012 as “block ciphers suitable for lightweight cryptography, which are tailored for implementation in constrained environments”. PRESENT takes 64 bit blocks of data as input and encrypts with keys of 80, or 128 bits.

PRESENT encryption has 31 rounds where each round consists of 3 parts. The functions are addRoundKey, S-box modification and permutation of data. Like AES, PRESENT also operates with a state, which is the values of the data at any given time during the operation.

2.2.2.1 addRoundKey

AddRoundKey is done in the same way as with AES. The state is XORed bit by bit with the round key.

2.2.2.2 S-Box

For the S-Box, the state is divided into 16 4-bit words. These are then substituted according to the following table. Values in the table are shown in hexadecimal.

x	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
$S[x]$	C	5	6	B	9	0	A	D	3	E	F	8	4	7	1	2

Table 2 - PRESENT S-Box

2.2.2.3 Permutation

In this function the state is permutated bit by bit. The permutation the PRESENT uses is given by the following table:

i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$P(i)$	0	16	32	48	1	17	33	49	2	18	34	50	3	19	35	51
i	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
$P(i)$	4	20	36	52	5	21	37	53	6	22	38	54	7	23	39	55
i	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
$P(i)$	8	24	40	56	9	25	41	57	10	26	42	58	11	27	43	59
i	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
$P(i)$	12	28	44	60	13	29	45	61	14	30	46	62	15	31	47	63

Table 3 - PRESENT permutation table

Each bit i of the state is moved to another position according to the value given in $\mathrm{P(i)}$. A visual representation of the whole substitution-permutation network looks like this:

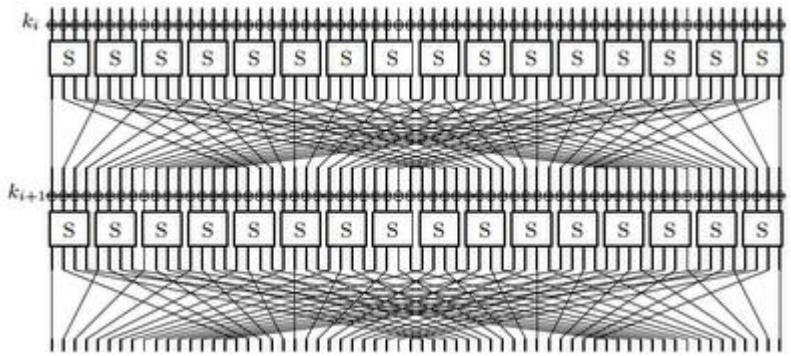


Figure 7 - SPN PRESENT

2.3 Architecture

The structure model of an IoT solution determines how the nodes in the network are connected to each other. There are of course many ways to do this, but this report will differentiate on two different ways; a centralized, and a decentralized model. In many systems there could be a combination of the two solutions. Where user devices are communicating with a common (centralized) server to request information, but this server has its databases spread out in different (decentralized) locations. In this report, we will distinguish the two, by how the IoT devices are communicating with each other. Are they connected and communicate via a centralized server over the internet, or do they build up a network of interconnected IoT devices themselves?

2.3.1 Centralized

A centralized, or server-client structure consists of central infrasturcture nodes with which each of the end nodes communicate. The server side can consist of a single instance that handles a small amount of devices, to large data centers with millions of clients. The use-case in the security assumption chapter, is a typical example of a centralized architecture in IoT for home automation. The IoT devices are connected over the internet to application hosted by some company.

The centralized architecture, which has been used for years with regular computers, clearly has its benefits. The networks are easy to manage and visualize as the connections between the communicating nodes are usually static. In this model most of the computation and processing is typically done on the server side. which relieves the end nodes of a lot of work. Smaller nodes can just gather and transmit data, which is handled on the server side.

In terms of IoT for smart homes this would mean that the IoT devices are connected to and managed by an infrastructure device. This could be a gateway such as shown in this system proposed by Freddy K. Santoso, and Nicholas C. H. Vun [11].

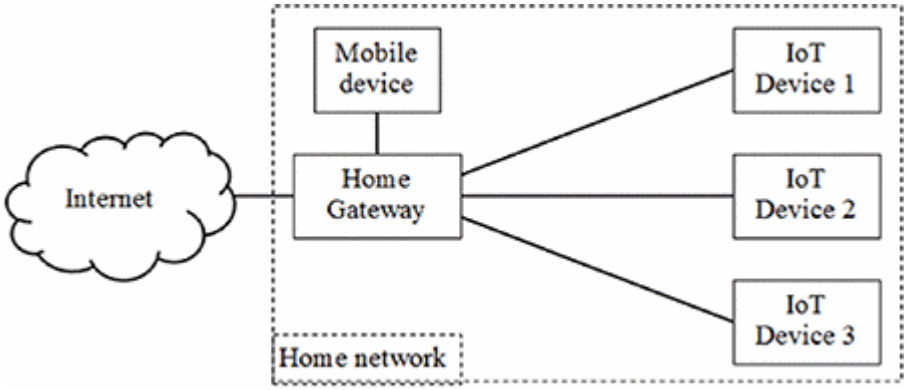


Figure 8 - Model of proposed architecture in [11]

By connecting devices like so, it creates a single point of contact for the user to manage all his devices.

2.3.2 Decentralized

A decentralized system, or a peer-to-peer system, has a flat hierarchy. The nodes in such a network communicate with each other directly rather than having all communication go through some infrastructure devices. The work load and storage can also be distributed between the devices themselves. A way of distributing the stored data is by using a distributed hash table. The nodes in the network store some part of the data each. The hash table functions as a lookup table where, data is stored in (key, value) pairs. Where nodes can request a hashed filename, or key, and the nodes that is responsible will reply with the corresponding data.

Similar to what the model shown for the centralized structure, a hybrid model consists of a decentralized network between the devices and with a stronger node acting as a gateway to the internet.

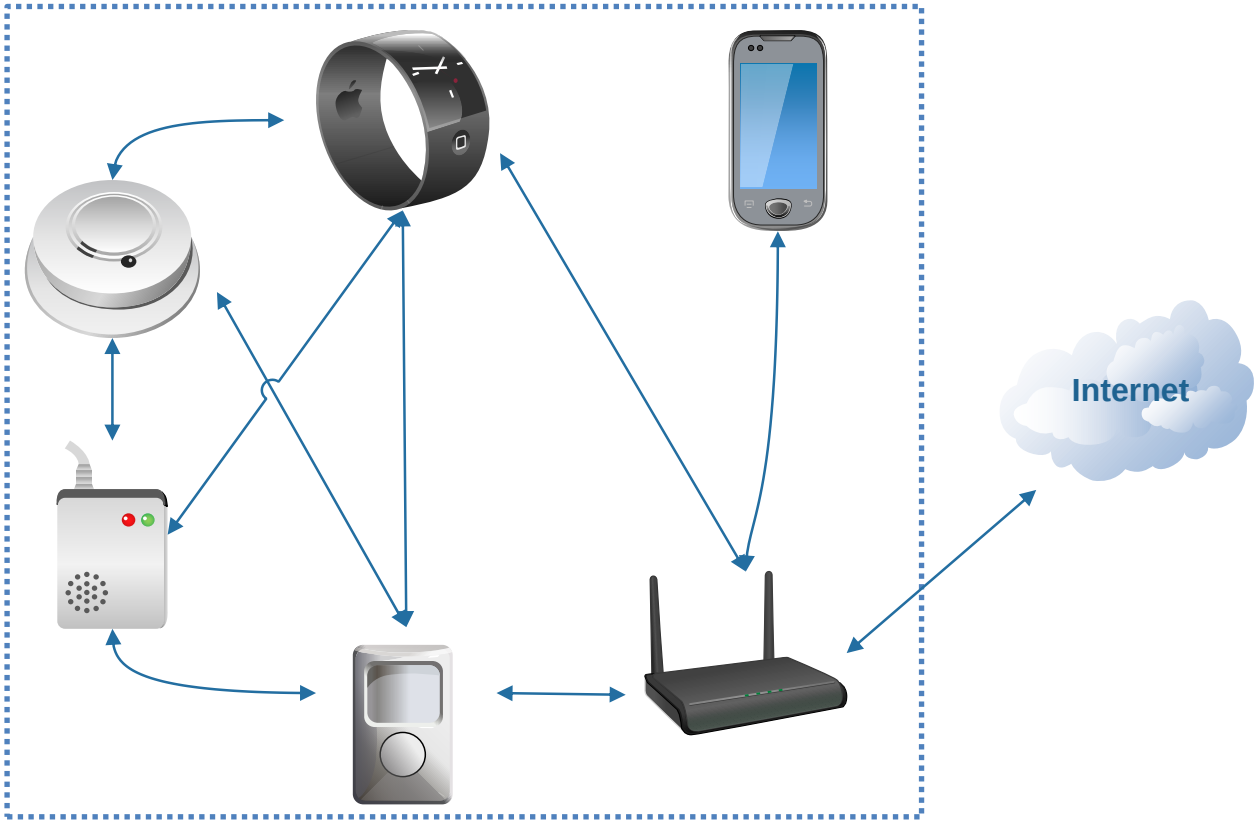


Figure 9 - P2P hybrid model

3 Analysis

The purpose of this report is to analyse the different solutions available for securing home automation systems. The different solutions that have been highlighted in this report, will be compared to each other to find the ones that are best suited for the given case. To do this we divide them into three categories based on what problem they are handling. The problems we will examine in this report are PKI, encryption and architecture. These are important parts for securing the communication network in a home automation system. Public key infrastructure provides authentication for the nodes in the network. Encryption algorithms like AES ensure privacy of the users. And the architecture determines how the network is structured and managed.

3.1.1 PKI

Public key infrastructure allows users to exchange keys and verify the identity of correspondents. These are two very important tasks to maintain the integrity of the communication. We must know that the data is coming from a legitimate source. This is usually obtained by digital signing. Also, since encryption of data is usually performed with a symmetric algorithm; we need a method to obtain a common secret key for both parties.

RSA has for a long time been the default algorithm used to perform tasks like these. However, RSA has been proven to be too demanding for smaller devices to run efficiently. In recent years, Elliptic curve cryptography has gained some attention. Although the algorithm is not new, RSA still was the first to emerge. This could be the reason why RSA has become the most used option.

When looking at how secure an algorithm is we review it with “bits of security”. This is a notion that combines the specifications of the algorithm and the used key length. This determines a level of security to compare algorithms with varying key sizes. AES with 128-bit key length offers 128 bits of security. To achieve this level of security with ECC and RSA we need key sizes of 256 and 3072 bits respectively [33]. The following table shows the comparison of the key sizes for RSA and ECC. The key sizes are also matched with the equivalent level of security symmetric key size.

Symmetric Key Size (bits)	RSA and Diffie-Hellman Key Size (bits)	Elliptic Curve Key Size (bits)
80	1024	160
112	2048	224
128	3072	256
192	7680	384
256	15360	521

Table 1: NIST Recommended Key Sizes

Table 4 - RSA vs. ECC key size

With this we see that the RSA requires a lot larger key to achieve the same level of security. This is, and will continue to be a problem for smaller devices with limited storage and computational power. On devices that are not limited by this, ECC will have the benefit of being much faster than RSA. Smaller keys will greatly improve the speed of signing, handshake etc. In fact, RSA is generally estimated to be 10 times slower than ECC with 128 bits of security levels and around 50 – 100 on 256 bits [34].

As shown in the description of the solutions in chapter 0, the mathematics behind ECC is a little more complicated than with the RSA. Although they have a lot of similarities, the operations can seems a bit confusing. There are also some parameters that has to be set, such as the choice of underlying field and basis, as well as the parameters for the

curve. Without proper knowledge, this can be a difficult task, and can result in insufficient security. Choosing the wrong parameters can result in the curve being supersingular. This results in the curve being susceptible to attacks like the MOV attack [35]. NIST has made a collection of elliptic curves and parameters [36]. This collection contains description and suggested values for all parameters that are used for defining elliptic curves for cryptographic uses.

3.1.2 Encryption

While RSA and ECC can be used for encryption, longer streams of data would be to exhausting. Therefor symmetric key algorithms are usually used instead. Given that first, a shared secret key is obtained, they are able to encrypt at a much faster rate. For some time, AES has been the standard algorithm for this. In IoT we need an algorithm that can handle encryption of data even with very limited hardware in a reasonable time. PRESENT was standardized in 2012 to satisfy the need for a lightweight crypto algorithm.

As PRESENT is specifically designed to be ultra-lightweight, we can assume that it will require minimal resources. In this paper [37], AES and PRESENT are compared when implemented on a low cost smart phone. In this analysis, we see that that AES outshines PRESENT in almost all categories in the results. Because PRESENT is developed for very constrained environments like RFID tags and contactless chips, an analysis on this kind of devices would provide different results.

Another, more thorough research shows similar results [38]. The conclusion is here that the lightweight algorithms do have their benefits when run on restricted hardware compared to AES. But the differences are mostly not enough to justify implementing anything else than AES.

Another thing to take note of when comparing the different algorithms is the security. The algorithm has to provide sufficient level of security if it is to be considered. A report published by the Czech Technical University evaluated PRESENT's resistance to brute force attacks [39]. The results show that their high performance, code breaking machine was able to verify 24 billion keys per second. Even with this high performance machine a brute force attack was estimated to last on average 800 thousand years.

From this we see that in most cases AES is not too demanding of the hardware and only in devices with very limited computational power will the benefits of PRESENT start to show. Taken into account that AES is more secure than PRESENT, we can says with certainty that AES will in almost any case be a better option for the given setup we assumed in this report.

3.1.3 Architecture

Today's smart homes need a system to manage the communication between the IoT. This system must handle frequent off- and on boarding of devices. It must also provide a secure communication between a variety of devices, in a smaller network. To determine how the network in a smart home is best structured we will look at some pros and cons of the two different models. In the centralized model computing and data storage will typically be on the server side. Services like data storage and management will then be hosted by a vendor. In the decentralized or peer-to-peer model, communication will be between nodes.

Centralized system

- **Pros**
  - Central system for control and management of nodes
  - Storing data in one place makes it easier to back up
- **Cons**
  - More expensive equipment needed for handling more devices
  - Single point of failure

Decentralized system

- **Pros**
  - Does not rely on a corporate server for functionality
  - User is in control of data and resources
  - Can scale without need of expensive equipment
  - Easy on-boarding of devices
- **Cons**
  - Infected clients can more easily harm the whole network [40]
  - Lack of overview on where data is stored

As we can see both systems have their advantages and disadvantages. But what is more important for the future of IoT? With the rate of increase we are seeing in the number of devices, scalability clearly is needed. And confidentiality and availability are

highly valued by the end user for any service. Seeing as a peer-to-peer network becomes stronger with increasing number of devices, this is a great way to handle an increasing number of devices.

Client-server systems require infrastructure that can handle all the nodes in the network. Such infrastructure devices can be quite exensive, and with the rapid increase of number of devices the network needs to be scaleable.

P2P avoids the danger of having services hosted on a server, in case the server should become unavailable for some reason. Typical scenarios where this could happen are when the devices communicate with a cloud-based application or data storage. If there is a problem on the server side or with your infrastructure devices, this service becomes unavailable for all devices. In case of a DDoS attack, a centralized system will be unavailable for all nodes that rely on this server. A P2P network will be a lot harder to take down. This is because in practice the attackers would have to target every device in the network to have the same effect. As a result, attackers would most likely never even try to perform a DDoS attack on such a network seeing as it would be too costly. Confidentiality can be achieved in both cases, but again the centralized system could possibly lose huge amounts of sensitive data if there is a breach in the central database.

With the server-client approach all communication between the devices pass through the servers. These cloud servers have to handle massive amount of data coming in from all the IoT devices. Even the devices that are located in the same household would have to connect over the internet and via the server farms to communicate.

You could argue that data storage and other services provided by third parties usually are secure enough. And that a security breach rarely happened and such services have almost 100% uptime. But in the event of an incident, the results are much worse than if a part of your P2P network becomes unavailable. If a node for some reason falls out, the rest of the network simply just works around it.

Recently there have been events that raise concerns about the usage of P2P networks for IoT. In 2016 it was uncovered that security cameras and other devices from Foscam, A Chinese manufacturer, was communicating secretly over the a P2P network hosted by the company [41]. When it became known that this communication was not possible to turn off, many security researchers and users alike became quite worried. Foscam later revealed that the purpose of this communication was heartbeat monitoring, to check the connectivity and that no personal data was stored on the company servers. The fact that the devices communicated with the manufacturer without the users knowing left many with a concern about the usage of P2P networks.

Decentralization will solve many of the problems that IoT currently face, but it will also introduce some new ones. Since there is such variation in performance of IoT devices, the ability to handle a decentralized model can vary. Both encryption and storage will become a problem for smaller devices with limited storage and computational power. Decentralized models such as blockchain removes the central storage servers. The nodes in the network are themselves responsible for the storage. The chain of information will increase over time, which in turn will become a problem for the smaller devices.

Streembit is a good example on how to handle IoT by use of a decentralized architecture. By not using the standard centralized architecture; Streembit is developed to overcome the problems these systems have with scalability and also becomes more scalable as more devices are added. The nodes are connected in a peer-to-peer network. On-boarding and distribution of data is provided by use of a distributed hash table called Kademlia. Data is stored in (key, value) pairs the value can then be retrieved by using this key. The Kademlia distributed hash table is already used in various other settings. In file-sharing networks the BitTorrent protocol is using the Kademlia DHT with great success.

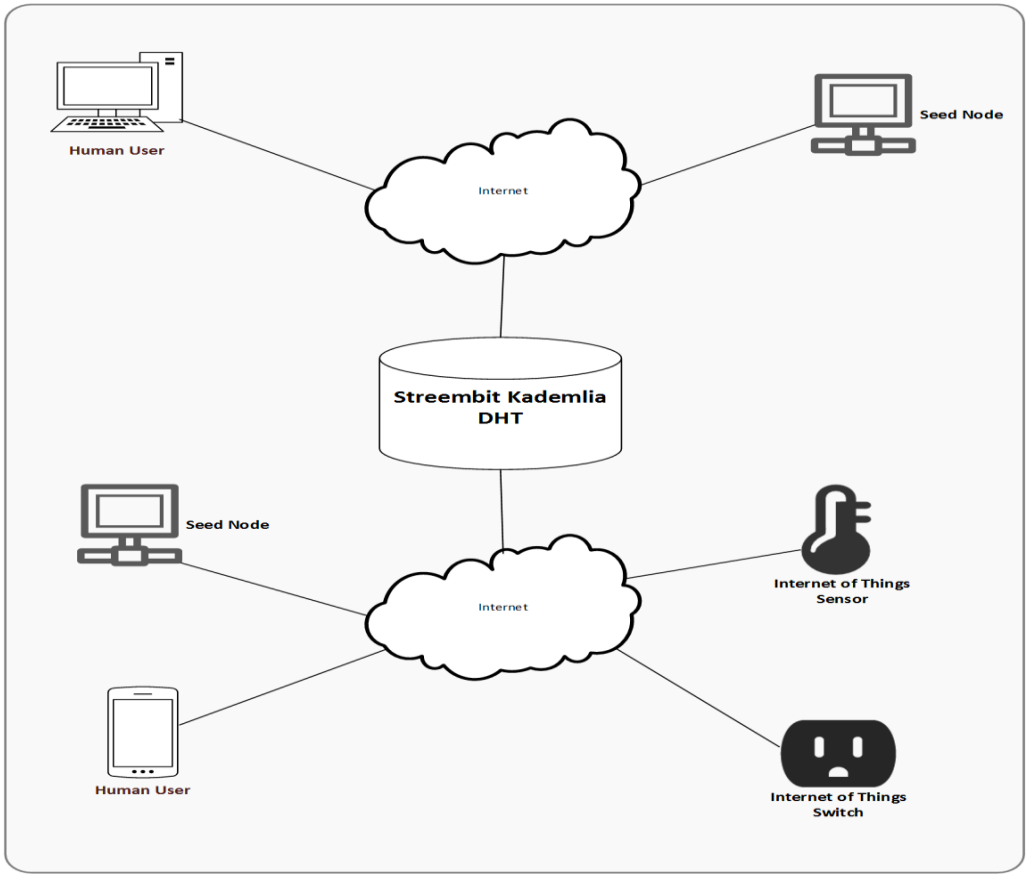


Figure 10 - Streembit architecture [42]

The nodes communicate directly with each other in a peer-to-peer network. Communication is secured by using Elliptic curve for key exchange (ECDH) and digital signing (ECDSA), and AES for data encryption. To further secure the communication between the nodes of the network, Streembit supports the option the make private networks. This allows the nodes to only communicate with your trusted devices within your home. On-boarding devices are connected to the public Streembit network to publish their encrypted IP address. They are then disconnected to await connection from the other private network participants.

Another approach to a peer-to-peer architecture of IoT, is the Blockchain solution. IBM and Samsung have been collaborating with Blockchain to develop an IoT solution called ADEPT that utilize the advantages of using a decentralized architecture like Blockchain

[43]. Blockchain is the same technology that Bitcoin uses. The idea is to use this technology to let devices establish a network, where they can publish data, much like how transactions are published to the chain in Bitcoin. With this shared information, other devices can act accordingly. Security cameras, sensors, locks, and alarm are all devices that communicates with each other in a smart home environment. With Blockchain such devices will be able to share their data without going through a server outside of the home network.

When storing the data in a blockchain, like ADPET does, there occurs a problem with storage for smaller devices. As time passes, there will be a lot of information to handle, and with the flat hierarchy of a peer-to-peer network, devices that have limited storage capacity will be congested with unnecessary information. ADEPT solves this by categorizing the peers of the network in 3 types: Light peer, standard peer and peer exchange [43]. These categories represent how powerful the device is and what should be its responsibilities.

The light peers are devices such as smart light bulbs which have limited storage and memory. These should only perform massaging and minimal storage of the distributed data. More demanding actions will be assigned to a more powerful trusted peer.

The standard peer is other IoT devices that have more powerful hardware and can store data and perform actions for themselves as well as the light peers. These devices will also forward messages for other devices. This will be quite demanding in terms of computational power for such devices. But IBM believes that with the current rate of development we will see devices capable of this as a standard in the near future.

The last category is the peer exchanges. Peer exchanges are powerful devices operated by companies essentially replacing the centralized servers that host cloud storage and application now. These can not only store a complete copy of a blockchain, but also host services that tie the consumer to a “marketplace”. These “marketplaces” are IBMs envisions of connections from a device to a vendor. With this, a washing machine can

order and pay for new detergent when it runs empty, with only machine-to-machine communication as demonstrated by IBM [43]. The categories also determine the trust level of the devices. Where light peers like smart watches and other wearables should not be able to make order or make any changes for other devices unless specifically noted.

Solutions like Blockchain and Streembit are still in development, and it could take years before we see a fully decentralized system be commercially available for home automation.

4 Discussion

From the literature review we can see that there is room for some improvements in the field of IoT. What is most important is to not to determine whether improvements must be made by the manufacturers or education of the end users. The answer is both. IoT devices must not be view as insignificant small devices that face no great threat to the web. Manufacturers should both enable devices and require users to make use of security measures.

As the comparison show, a decentralized architecture with ECC for PKI and AES for encryption is the preferred way to go. Elliptic curve cryptography shows better performance in all cases when applied in restricted environments and should be used as a standard way of implementing public key infrastructure in home automation systems as well as other fields of IoT. RSA is the most used solution, but as there is a need for a lighter algorithm, more and more systems are starting to use ECC. The fact that the math behind it is easier to understand, should not hinder the drive to implement better solutions. ECC is able to provide the same level of security with a shorter key length. For devices with limited computational power, this is a huge benefit.

In a smart home like the one modelled in the use-case, devices that have extreme limitation in computational power are rare. Even light bulbs like the LIFX smart lightbulb are able to use AES. An exploit which made an attacker able to obtain the WIFI password was discovered in 2014, but this was because the pre-shared keys never changed, not because of the AES implementation [44]. AES is the standard block cipher for many protocols and security suites. And given the marginal benefits that PRESENT shows in smaller semi-restricted devices, there is little that suggests there is a need to change that. AES performs well in all but the most extreme cases, such as RFID sensors. The benefits of PRESENT do not outweigh the fact that on such devices as in the use-case, AES performs better and provides more security. The continuation of using AES as a standard for home automation systems is therefore suggested.

As for the architecture, the centralized model is the most well-known and mature option. Because of the way it is structured, it is easier to both visualize and manage. But with the rate of increase in number of devices, a peer-to-peer architecture is the best suited solution when considering the future of IoT. Having a device accessible from the internet can be useful, but the real power in home automation is the cooperation of the devices themselves. As devices are added to a smart home, it enables cooperation between the devices without human intervention. Such as security systems with cameras, lock ad alarms. This kind of M2M communication within your home should be directly between the devices. A peer-to-peer architecture also provides the added security against DDoS attacks. While a home network is not really a high value target for such an attack, the peer-to-peer structure makes it much more robust against a DDoS attack. This kind of attack is more likely to target larger companies, and as the P2P option is less reliant on communication going via external parties, it is less likely to be affected by this kind of attack.

Blockchain is one of the solutions to create P2P networks in IoT. When using blockchain at the current state, a fully decentralized network is not the most efficient. Smaller devices will likely encounter storage problems. With solutions like this, the chain will eventually become quite large as it is storing all transactions that are being made. With devices of variable sizes, a solution where the stronger devices handle more of the storage and communication, would be more efficient. ADEPT by IBM fixes this problem by distributing the workload and storage to more powerful devices. The only problem is that this solution requires devices powerful enough to be used as standard peers, and handle some of the tasks of the light peers. According to IBM, such hardware capabilities will be standard in a few years. But as of now a smart home like the one in our use-case would consist of only light peers, which defeats the purpose of implementing such a model.

A way to to work around this problem is to add a device that handles the distributed hash table. In short term this works against the idea of decentralization, but it solves the problem with hardware limitations in a way that prepares for the future. The smaller

devices would not be overloaded with information that they don't need. This solution can utilize the benefits of decentralization, with the minus of having a single point of failure at the devices that handles the hash table. But in contrast to a centralized architecture, the failure point is in your home and in your control. As IoT devices with more computational power enters the market these will then be able to function as standard peers, and contribute along with the device. Then, the purpose of the device will shift from handling the workload of all your devices, to adding stability to your network. With a more powerful device in place, other, more demanding security measures could be implemented as well. With behaviour analysis, the peers could be monitored for signs of infection or malicious activity based on their behaviour.

The Streembit solution handles the problem of low power devices in a similar, but more realistic way. The developer acknowledges that in its current state, blockchain operation are simply too demanding for most IoT devices. Since the distributed hash table Kadmlia, is not storing every transaction like the with blockchain, it is much less demanding. In Streembit, low power devices are excluded from actions that are not relevant to them by a gateway [45]. A garage door opener does not need to take part in routing and other such actions.

For security Streembit is also able to establish private network where only authorised accounts and devices are able to connect. This, combined with the fact that it utilizes security measures like AES and ECC to provide integrity and confidentiality, provides the network and its devices with strong security against Sybil attacks and other malicious activities.

In other words, Streembit fulfils the requirement for ensure Availablility, confidentiality and intergretiy for users and devices in the network. It also implements features corresponding to the answers of the questions in the problem statement.

- Is Elliptic Curve Cryptography more suitable for smart home devices than RSA?
    - From the research covered in this report ECC is concluded to be the best suited option for smart homes
- Should the ultra-lightweight encryption algorithm PRESENT replace AES in such a setting?
    - Given that the type of devices commonly used in smart homes are able to run AES efficiently, there is no need to change it for PRESENT.
- Which type of network architecture is best suited for the future of IoT? Centralized client-server, or decentralized peer-to-peer?
    - Peer-to-Peer seems to be the architecture that is best suited for the future of IoT. While most of the solutions for this is still in development they show great promise.

5 Conclusion

A literature review of the current situation in the field of IoT. Based on this an analysis of possible solution has been done to find and propose the security measures that are best suited for public key infrastructure, data encryption and network architecture in smart homes. As a result of the reviewed literature in this report, a decentralized peer-to-peer communication network that uses AES and ECC, is suggested as a set of solutions best suited for home automation. These solutions alone are by no means a fix to the many problems that IoT face, but based on what is shown in this report this is the way to go for further improvement of the security of smart homes. The Streembit system implements the suggested security measures, and based on the analysis this is suggested as the way forward for improving IoT and home automation.

6 Further work

Securing network is a continuous process. New vulnerabilities and exploits will emerge and new ways of handling them must be found. While IoT and home automation has been around for a while, it is still an emerging technology. There is a lot of development driven in different directions to find the best solutions. With systems like Streembit and ADEPT, we see a shift from the server-client model towards decentralization and peer-to-peer networks. These systems are still in the development phase, but they show great promise. This report suggested the Streembit solution as the best way forward. A possible improvement for the Streembit system would be to implement a system like ADEPT, where the nodes are categorized based on the computational power. This relieves the low power nodes of a lot of work.

With the future in mind, other aspects of securing home automation should be analysed as well. This report covers some of the vital aspects for securing network communication. But for a well-functioning system other topics and other layers, like the application layer should see some improvements. Stricter policies for changing and strength of passwords and methods for rolling out security updates for all devices in your home network are things that definitely could be useful for home automation.

Bibliography

[1] “Security Challenges For the Internet Of Things.” https://www.iab.org/wp-content/IAB-uploads/2011/03/Turner.pdf.

[2] “IEEE IoT Towards Definition Internet of Things.” http://iot.ieee.org/images/files/pdf/IEEE_IoT_Towards_Definition_Internet_of_Things_Revision1_27MAY15.pdf.

[3] “Architecting the Internet of Things: State of the Art (PDF Download Available),” *ResearchGate*. [Online]. Available: https://www.researchgate.net/publication/274718805_Architecting_the_Internet_of_Things_State_of_the_Art. [Accessed: 11-May-2017].

[4] “The Current State of Security in Smart Home Systems.” https://www.sec-consult.com/fxdata/seccons/prod/media/SEC_Consult_Whitepaper_The_Current_State_of_Security_in_Smart_Home_Syste....pdf.

[5] “AT&T Exploring IoT security.” https://www.business.att.com/cybersecurity/docs/exploringiotsecurity.pdf.

[6] C. P. P. Schumacher, N. Kushalnagar, and G. Montenegro, “IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals.” [Online]. Available: https://tools.ietf.org/html/rfc4919. [Accessed: 04-May-2017].

[7] T. W. <wintert@acm.org>, “RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks.” [Online]. Available: https://tools.ietf.org/html/rfc6550. [Accessed: 04-May-2017].

[8] “McAfee threats predictions 2017.” https://www.mcafee.com/au/resources/reports/rp-threats-predictions-2017.pdf.

[9] S. Biddle, “Fortinet 2017 Cybersecurity Predictions: Accountability Takes the Stage,” *Fortinet Blog*, 21-Nov-2016. [Online]. Available: http://blog.fortinet.com/2016/11/21/fortinet-2017-cybersecurity-predictions-accountability-takes-the-stage. [Accessed: 09-May-2017].

[10] “OWASP Internet of Things Project - OWASP.” [Online]. Available: https://www.owasp.org/index.php/OWASP_Internet_of_Things_Project. [Accessed: 19-Apr-2017].

[11] F. K. Santoso and N. C. H. Vun, “Securing IoT for smart home system,” in *2015 International Symposium on Consumer Electronics (ISCE)*, 2015, pp. 1–2.

[12] “Framework | AllSeen Alliance.” [Online]. Available: https://allseenalliance.org/framework/. [Accessed: 29-Mar-2017].

[13] streembit, “Streembit.” [Online]. Available: http://streembit.github.io/. [Accessed: 28-Mar-2017].

[14] S. E. C. Consult, “House of Keys: Industry-Wide HTTPS Certificate and SSH Key Reuse Endangers Millions of Devices Worldwide.” http://blog.sec-consult.com/2015/11/house-of-keys-industry-wide-https.html.

[15] Canonical, “Research: Consumers are terrible at updating their connected devices,” *Ubuntu Insights*. [Online]. Available: https://insights.ubuntu.com/2016/12/15/research-consumers-are-terrible-at-updating-their-connected-devices/. [Accessed: 10-Apr-2017].

[16] “Thousands of Ubiquiti AirOS routers hit with worm attacks,” *Symantec Security Response*. [Online]. Available: http://www.symantec.com/connect/blogs/thousands-ubiquiti-airos-routers-hit-worm-attacks. [Accessed: 21-May-2017].

[17] K. Zhang, X. Liang, R. Lu, and X. Shen, “Sybil Attacks and Their Defenses in the Internet of Things,” *IEEE Internet Things J.*, vol. 1, no. 5, pp. 372–383, Oct. 2014.

[18] “KrebsOnSecurity Hit With Record DDoS — Krebs on Security.” https://krebsonsecurity.com/2016/09/krebsonsecurity-hit-with-record-ddos/.

[19] “Dyn Analysis Summary Of Friday October 21 Attack | Dyn Blog.” [Online]. Available: http://dyn.com/blog/dyn-analysis-summary-of-friday-october-21-attack/. [Accessed: 25-Mar-2017].

[20] E. Fernandes, J. Jung, and A. Prakash, “Security Analysis of Emerging Smart Home Applications,” in *2016 IEEE Symposium on Security and Privacy (SP)*, 2016, pp. 636–654.

[21] “20 Billion Connected Internet of Things Devices in 2017, IHS Markit Says | Electronics360.” [Online]. Available:

http://electronics360.globalspec.com/article/8032/20-billion-connected-internet-of-things-devices-in-2017-ihs-markit-says. [Accessed: 25-Mar-2017].

[22] “Keylength - NIST Report on Cryptographic Key Length and Cryptoperiod (2016).” [Online]. Available: https://www.keylength.com/en/4/. [Accessed: 19-May-2017].

[23] “Doctrina - Why RSA Works: Three Fundamental Questions Answered.” [Online]. Available: http://doctrina.org/Why-RSA-Works-Three-Fundamental-Questions-Answered.html. [Accessed: 28-Apr-2017].

[24] “E is for Elliptic Curves | Mathematical Institute.” [Online]. Available: https://www.maths.ox.ac.uk/about-us/life-oxford-mathematics/oxford-mathematics-alphabet/e-elliptic-curves. [Accessed: 29-Apr-2017].

[25] R. Materese, “Advanced Encryption Standard (AES),” *NIST*, 26-Nov-2001. [Online]. Available: https://www.nist.gov/publications/advanced-encryption-standard-aes. [Accessed: 29-Mar-2017].

[26] <> A. F., <> P. K., and <> P. K., “The Secure Sockets Layer (SSL) Protocol Version 3.0.” [Online]. Available: https://tools.ietf.org/html/rfc6101. [Accessed: 20-May-2017].

[27] “AES sidechannel attacks IEEE2.” http://spiegel.cs.rit.edu/~hpb/public_html/Lectures/20112/S_T/Src/36/AES_sidechannel_IEEE2.pdf.

[28] “Design and Implementation A different Architectures of mixcolumn in FPGA (PDF Download Available),” *ResearchGate*. [Online]. Available: https://www.researchgate.net/publication/230853805_Design_and_Implementation_A_different_Architectures_of_mixcolumn_in_FPGA. [Accessed: 14-May-2017].

[29] “AES by Example.” http://www.adamberent.com/documents/AESbyExample.pdf.

[30] “POWER8 in-core cryptography,” 21-Sep-2015. [Online]. Available: http://www.ibm.com/developerworks/library/se-power8-in-core-cryptography/index.html. [Accessed: 20-May-2017].

[31] K. AlMarashda, Y. AlSalami, K. Salah, and T. Martin, “On the security of inclusion or omission of MixColumns in AES cipher,” in *2011 International Conference for Internet Technology and Secured Transactions*, 2011, pp. 34–39.

[32] “PRESENT: An Ultra-Lightweight Block Cipher.” http://www.lightweightcrypto.org/present/present_ches2007.pdf.

[33] “ECC 101: What is ECC and why would I want to use it?” [Online]. Available: https://www.globalsign.com/en/blog/elliptic-curve-cryptography/. [Accessed: 18-Apr-2017].

[34] “Atmel 8951 CryptoAuth RSA ECC Comparison Embedded Systems WhitePaper.” http://www.atmel.com/Images/Atmel-8951-CryptoAuth-RSA-ECC-Comparison-Embedded-Systems-WhitePaper.pdf.

[35] “Elliptic Curve Cryptography.” https://ocw.mit.edu/courses/mathematics/18-704-seminar-in-algebra-and-number-theory-rational-points-on-elliptic-curves-fall-2004/projects/asarina.pdf.

[36] “NIST Recommended elliptic curves for federal government use.” http://csrc.nist.gov/groups/ST/toolkit/documents/dss/NISTReCur.pdf.

[37] Lara-Niño, C. Andrés, M. S. Miguel, and D. P. Arturo, “An evaluation of AES and present ciphers for lightweight cryptography on smartphones,” in *2016 International Conference on Electronics, Communications and Computers (CONIELECOMP)*, 2016, pp. 87–93.

[38] “Towards Green Cryptography: a Comparison of Lightweight Ciphers from the Energy Viewpoint.” https://perso.uclouvain.be/fstandae/PUBLIS/115.pdf.

[39] J. Pospíšil and M. Novotný, “Lightweight cipher resistivity against brute-force attack: Analysis of PRESENT,” in *2012 IEEE 15th International Symposium on Design and Diagnostics of Electronic Circuits Systems (DDECS)*, 2012, pp. 197–198.

[40] Z. Trifa and M. Khemakhem, “Analysis of malicious peers in structured P2P Overlay Networks,” in *2014 World Congress on Computer Applications and Information Systems (WCCAIS)*, 2014, pp. 1–6.

[41] “This is Why People Fear the ‘Internet of Things’ — Krebs on Security.” https://krebsonsecurity.com/2016/02/this-is-why-people-fear-the-internet-of-things/.

[42] streembit, “Streembit - Documentation.” [Online]. Available: http://streembit.github.io/documentation/. [Accessed: 04-May-2017].

[43] “Empowering the edge - Practical insights on a decentralized Internet of Things.” https://www-935.ibm.com/services/multimedia/GBE03662USEN.pdf.

[44] D. Goodin, “Crypto weakness in smart LED lightbulbs exposes Wi-Fi passwords,” *Ars Technica*, 07-Jul-2014. [Online]. Available: https://arstechnica.com/security/2014/07/crypto-weakness-in-smart-led-lightbulbs-exposes-wi-fi-passwords/. [Accessed: 19-May-2017].

[45] “streembit_whitepaper_v1.0.2.pdf.” http://streembit.github.io/downloads/streembit_whitepaper_v1.0.2.pdf.

[46] “Securing the Internet of Things: A Proposed Framework,” *Cisco*. [Online]. Available: http://www.cisco.com/c/en/us/about/security-center/secure-iot-proposed-framework.html. [Accessed: 25-Mar-2017].

[47] B. Anggorojati, N. R. Prasad, and R. Prasad, “Elliptic curve cryptography based key management for the M2M local cloud platform,” in *2016 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, 2016, pp. 73–78.

[48] S. R. Singh, A. K. Khan, and T. S. Singh, “A critical review on Elliptic Curve Cryptography,” in *2016 International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT)*, 2016, pp. 13–18.

Appendices

Appendix A Glossary and Abbreviations

ECC	Elliptic Curve Cryptography
RSA	Cryptosystem designed by Ron Rivest, Adi Shamir, and Leonard Adleman
AES	Advanced Encryption Standard
LLN	Low-Power and Lossy Networks
6lowPAN	IPv6 over Low-power Wireless Personal Area Networks
PKI	Public key infrastructure
MITM	Man in the middle
ECCDSA	Elliptic Curve Digital Signature Algorithm
ECCDH	Elliptic curve Diffie–Hellman
IoT	Internet of Things
P2P	Peer-to-Peer
M2M	Machine-to-Machine
H2M	Human-to-Machine
RFID	Radio Frequency Identification
IP	Internet Protocol
SPN	Radio Frequency Identification