Virtual environments for penetration testing of IoT devices

UNIVERSITY OF TURKU
Department of Future Technologies
Master of Science in Technology Thesis
Networked Systems Security
March 2020
Sami Granö

Supervisors:
Assoc Prof. Seppo Virtanen
Dr. Antti Hakkala

UNIVERSITY OF TURKU
Department of Future Technologies

SAMI GRANÖ: Virtual environments for penetration testing of IoT devices

Master of Science in Technology Thesis, 62 p.
Networked Systems Security
March 2020

The aim of this thesis is to build a virtual penetration testing environment in order to test cyber security of IoT devices and provide material for teaching penetration testing in cyber security courses at the University of Turku. This thesis utilises the VMWare ESXI server rented by the University of Turku and also those Linux operating systems which have open-source code licences. In addition, this study exploits the Windows XP licence, held by the University of Turku, as the target machine in the attack. The IoT devices in this study have been coded from scratch by author of this thesis, and they work automatically when the virtual machine, where the device is installed, is switched on.

The virtual laboratory is built so that it is possible to install new devices if required. The beginning of the thesis is as comprehensive as possible so that the reader can comprehend the idea of cyber security in IoT devices and smart home system easily. This thesis covers the most common data transfer methods, their history, and the strength of their cyber security. In addition, the thesis covers the most used attack types and tools for implementing these attacks, and also example cases where these particular attacks have been used.

Thesis also presents an architecture model for a smart home in order for the reader to get a better view of the different possibilities of a smart home. Possible risk factors concerning cyber security are also considered with each device. This thesis also touches upon the role of smart cities and industry in the IoT market. The main focus is, however, on IoT devices used in smart home architecture and on researching their cyber security.

Keywords: Internet of Things, smart house, penetration testing, virtualization

Contents

1 Introduction 1

2 Internet of Things 4

2.1 Brief history of IoT 5

2.2 Smart home 6

2.3 Smart cities 8

2.4 IoT in industry 10

3 IoT network technologies 12

3.1 Ethernet 12

3.1.1 History of Ethernet 13

3.1.2 Security of Ethernet 13

3.2 Wireless local area network 14

3.2.1 WLAN standards and security 15

3.3 Bluetooth and BLE 17

3.4 Low-Power Wide-Area network 20

3.4.1 Narrowband IoT 20

3.4.2 LoRaWAN 21

3.4.3 SigFox 23

4 The current state of IoT cyber security 24

4.1 Devices in smart home system 24

4.2 Cyber attacks against IoT devices 27

4.2.1 DoS and DDoS 27

4.2.2 MITM 28

4.2.3 Software attacks 30

4.3 Pentesting tools and platforms 32

4.3.1 Kali Linux 33

4.3.2 Burp Suite 33

4.3.3 WireShark 36

4.3.4 Shodan 36

5 Designing smart home system architecture 38

5.1 Router 38

5.2 Controller 40

5.3 Smart lights 41

5.4 Smart lock 42

5.5 Smart sensors 42

5.6 Surveillance camera 43

6 Designing and building a virtual smart home IoT environment for penetration testing education 45

6.1 VmWARE 46

6.2 IoT devices 50

6.2.1 Smart lock 51

6.2.2 Surveillance camera interface 52

7 Conclusion 54

Bibliography 56

Chapter 1

Introduction

The number of Internet of Things (IoT) devices has grown exponentially in the last decade. Originally these devices became increasingly popular in industry, but later they have been manufactured for private use and for the needs of smart cities as well. Devices targeted for the industry are mainly different types of measuring sensors and minicomputers needed for transmitting their data. Devices targeted for smart cities use the same sensor techniques that are used in the industry, however, there are some additions: street lighting control devices, measuring techniques for smart parking, devices for measuring cleanliness of swimming water, and different types of collectors for targeted marketing. The needs of smart homes slightly deviate from these two previously mentioned, as the environment which is measured is much smaller and it is desirable that the controlling environment is easy and quick and can be directed to the person living there.

Together with the increasing number of IoT devices in the society, a question of their security is raised. When installing devices in public sector buildings, such as hospitals and schools, it is of utmost importance to consider privacy and safety of those who use the building. These people might not have the opportunity to affect the functioning of the devices, thus, they need to be able to trust that the devices are safe. When installing IoT devices in private homes, yet new aspects of safety must be taken into consideration. The information that might be recorded on the devices in private homes must be kept safe, and also data transfer to the network outside the apartment has to be limited as carefully as possible. The user needs to be able to do this limiting him- or herself, and it cannot be predefined by the device manufacturer.

In this thesis, the safety of IoT devices is examined and a virtual testing environment is built for testing of cyber security in these devices. The thesis touches upon devices used in industry and smart cities, but the main focus is on the smart home IoT architecture in the private sector. The purpose of this virtual penetration testing environment is to model the IoT device environment of private homes and examine how the safety of the devices installed in this environment can be tested. The aim of this thesis is to examine whether IoT devices can be tested reliably in a virtual environment, or do they require a physical network environment for reliable testing. Another aim of this thesis is to build a closed practice environment for penetration testing of IoT devices in the cyber security education at the university of Turku.

Chapter 2 explains what is an IoT device and their most common uses and operating principles. After this, the history of IoT devices is covered briefly. The history begins with the introduction of the term IoT in 1999 and from where the term came [2]. In addition, its development to this day is examined, ending up in future prospects in 2020 [7]. After covering history, the use of IoT devices in smart homes, smart cities and in industry is explored. Section 2.2 examines the IoT environment in general and the architecture built around it in a smart home. Later, chapter 5 delves more deeply into this environment. Section 2.3 covers IoT devices in a smart city on a more general level and section 2.4 examines their functions in the industry.

Chapter 3 examines network techniques used in IoT devices. The network techniques chosen for this study were the most common techniques in the sector of IoT devices at the time this study was made. Section 3.1 focuses on a wire connection, Ethernet, examining its history and safety. Section 3.2 is focused on WLAN standards and their safety, whereas section 3.3 examines the standards of Bluetooth and Bluetooth low energy. The most common techniques of LPWAN techniques used in IoT devices are dealt with in section 3.4. These include, for example, Narrowband IoT, LoRaWAN and SigFox.

In chapter 4, the safety of IoT devices is considered, along with attack types targeted at them and tools with which these attacks can be implemented. There is also an example case of each attack type, explaining where that particular attack was used and how this could have been prevented by improving safety in the devices. The latter part of chapter 4 presents attacking tools and what kind of attacking techniques it is possible to produce with these tools.

Chapter 5 delves into the architecture of a smart home. When the goal is to change the home into a smart home, many things need to be taken into consideration in the planning stage. This chapter explores various devices, what are the minimum requirements for a functioning smart home, and other generally used devices and their safety in a smart home.

Chapter 6 presents the virtual penetration testing laboratory built in this thesis and introduces the softwares needed for building this laboratory. In addition, information on the first capture the flag (CTF) exercises is given, and it is intended that these are used in cyber security courses in the future.

Chapter 2

Internet of Things

The term Internet of Things (IoT) means devices that are connected to the Internet [1]. For example, new smart refrigerators, which can connect to the Internet, are IoT devices. The user is able to program the refrigerator so that it orders groceries from the shop: the order is forwarded via Internet to the shop, which then sends the products to the customer using a delivery service. The number of IoT devices in the society has grown and will continue to grow steadily as the economic situation of people improve and they have more free time. Increased free time means that people also want more conveniences to help their everyday life. The benefits of these smart devices in the society and its functions are enormous and for this reason the number of start-up companies in the field of IoT devices is increasingly growing.

As the number of IoT devices increases, their information security must also be taken into consideration. Many devices are made using the plug in principle and hence they can be plugged into the network straight from the package. Usually these devices have a factory-configured password, which can be simple, for example “1234”, or long and complicated. These passwords are usually found from the product’s packaging and people take them from there. If the factory-configured password is too weak, there should be directions for changing the password. However, this is not always the case. Then the user who buys and installs the product can cause a threat in the information security of the whole system into which the device is plugged in.

IoT devices can be plugged into personal environments such as smart homes, cars, cottages, and boats. Alternatively, the devices can be plugged into a larger environment, for example into a whole network of a city. IoT devices are pioneering in the industry; several production lines have been automated with the help of IoT devices. IoT devices can be used to monitor access control, parking, and other simple production line functions. This chapter deals with the history of IoT, followed by utilising IoT devices in smart home systems, planning of smart cities, and in the industry.

2.1 Brief history of IoT

Although IoT devices have become widespread very quickly and their use has diversified, for a number of sectors the history of the Internet of Things is still very short. The term Internet of Things was first used by Kevin Ashton in 1999 in a title of his presentation [2]. This title was meant to impress the senior partners of his organization by becoming through as a term aimed for the future and at the same time being mysterious. The presentation was about an identification method using RFID technology [2]. Internet of Things is, however, much older than the term used to define it, and its beginnings take place much further in history. It is very hard to define the exact time when the technology, nowadays called as the Internet of Things, was created. But when thinking of devices which function independently and are connected to each other via internet, these kinds of devices are already found in the 1970s [3]. The first official IoT device was developed by John Romkey: he introduced it in 1990 in the INTERCOP conference [4]. The device was a toaster which was possible to switch on and off via internet. The device was plugged into the internet using TCP/IP protocol suite.

In 2005 a new step was taken in IoT technology. The International Telecommunication Union (ITU) published a report which dealt with the future of Internet and devices which are plugged into the internet. “from anytime, any place connectivity for anyone, we will now have connectivity for anything” ITU report [5]. In the same year, Rafi Haladjian and Olivier Mével introduced a robotic rabbit which was able to do alerts of the state of stock market and report the newest headlines to users using a WiFi connection [4]. According to the Cisco Internet Business Solutions Group (IBSG), IoT was created between the years 2008-2009 [4]. At that time, according to their report, there were more devices plugged into the internet than there were users. To be more exact, there were 1.84 devices plugged into the internet per one person, according to them [4].

The new version of Internet Protocol, called IPv6, was published in December 1998 [6], after which the universe of address locations grew so enormously that it was possible to plug in all devices in the internet and still there would have been addresses left. After this, Cisco, IBM and Ericsson began to develop education and marketing systems for these devices. In addition, Arduino began developing new platforms suitable for IoT devices together with other platform developers. In a briefing published in 2013, there was an estimation that IoT device market would expand to a value of 8.9 billion by the year 2020 [7].

2.2 Smart home

When people think about smart homes, they immediately think about science fiction movies situated in the future, where a home is functioning like a robot and is capable of conforming to all of the tenant's wishes and can also function as an interlocutor with the tenant when he or she is at home. These movies can cause people to fear that they lose their privacy; they start to think whether they trust this technology and will their personal and sensitive information leak outside home without their knowing. People equally fear the kind of world depicted by futuristic movies. For example, in the film ”Terminator”, artificial intelligence called ”Skynet” invades mankind. Fearing these kinds of incidents is never unjustified, but considering the technology nowadays, a smart home with its units is ultimately a computer software and it can easily be switched off.

In reality, when the number of IoT devices grow, these devices functioning in the internet can be harnessed to function better in a contiguous network environment [8]. People do not need to conform to the increase in amenities provided by smart phones, but these same properties can be used to govern lighting or heating systems in homes. It is possible to install an electronic lock to the front door, programmed in a way that when somebody who uses the apartment has forgotten their key, the door can be opened remotely; with a software installed in the phone. If the tenant wants to sleep in a cooler room, but still wants the room to be warm when he or she wakes up, it is possible to adjust the temperature automatically according to certain times of day.

It can be challenging to build a smart home without any technical knowledge. However, nowadays there are more and more companies that offer a so-called turnkey solution; a service where the tenant can get these kinds of services adjusted to the tenants needs. The home is normally governed by a central computer, which usually is an IoT device equipped with a touch pad (see figure 2.1).

Figure 2.1: The control panel of IoT devices in a smart home system

Systems installed at home, for example sockets, thermoregulators, surveillance camera systems, electronic locks and other internet-connectable sensors, in addition to IoT devices, are plugged into the internet via a central control unit and programmed according to users wishes. Furthermore, a smart phone can be plugged into the control unit with a separate application to make it easier for the user to govern devices remotely.

After the devices have been installed and the user is instructed to use them, the biggest challenge is security. As was already mentioned in the beginning of this section, the user usually has a realistic fear that devices might leak sensitive information outside home. Therefore, is it possible that some unrelated instance could take control over the devices in somebody's home and cause harmful or even dangerous activity with these devices? These questions and their answers are the subject of the fifth chapter of this thesis, where building of a smart home system is examined with an eye especially on information security. In the next section it is examined how IoT devices have enabled the building of entire smart cities.

2.3 Smart cities

A smart city is a city where normal services of a city, for example public transport, schools, libraries, hospitals, power stations, authorities' services, water, plumbing and electricity supply are all combined into a single system operating with electronic devices and sensors [9]. These devices and sensors are meant to serve the inhabitants of the city and improve their quality of life by making services better available and improve their use.

When these kinds of systems are planned to be built in a city, the first thing that should be thought of, is that services are supposed to improve citizens' quality of life and security. However, as surveillance increases, citizens probably cannot help but think that a "big brother" is watching them. Disproving this assumption should be the first priority when making plans. The best ways to spread knowledge of smart cities, is to make the gathered data open and give the citizens themselves a chance to participate in developing systems in the city together with the authorities [10].

Already there are several smart city systems operating in the world and these have proved to be useful in the operation systems of cities. For example, a Danish research with a published guide ”Think Denmark” [11] dealt in detail with different stages of how to plan a smart city and how it will improve the lives of citizens (see figure 2.2).

According to the Danish framework, the most important thing in designing a smart city is to change "smart vision to a smart society". The planning begins by changing all the systems in the city to a digital form. This means planning and developing sensors, data gathering services and systems for their management. Infrastructure dealing with different sectors of the city, for example variables targeted at energy production, waste management, traffic, and air quality, must be built on top of the basic systems [11]. Air quality, temperature and efficiency of these equipment can be affected by installing sensors to gather data from the air quality, for example from schools, nurseries, hospitals and other public buildings. The data from these sensors can be gathered to a unitary system, from which subsystems can be governed and information can be distributed to

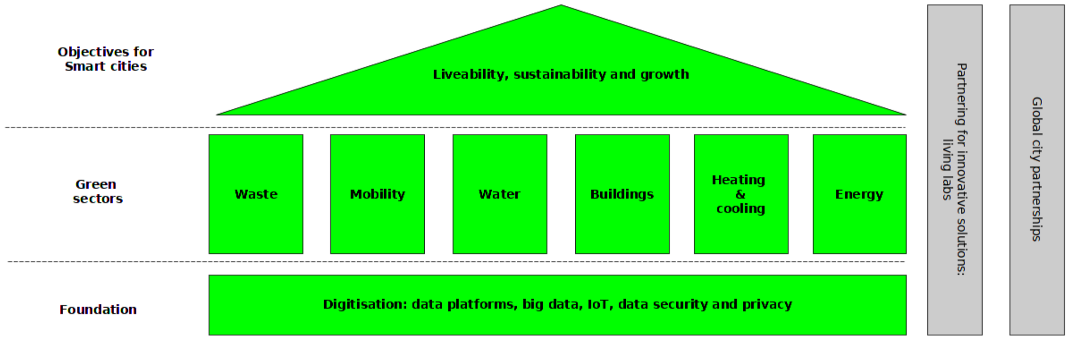

Objectives for Smart cities
Liveability, sustainability and growth
Green sectors
Waste
Mobility
Water
Buildings
Heating & cooling
Energy
Foundation
Digitisation: data platforms, big data, IoT, data security and privacy
Partnering for innovative solutions: living labs
Global city partnerships

Figure 2.2: A framework for smart cities [11]

inhabitants. Also, if a weakening in the quality of air is detected, it is easy to attend to this problem early on, in order to avoid exposures. Furthermore, sensors can be used to monitor temperature in buildings and when they are not in use, the temperature can be lowered, which brings savings in heating costs.

Emptying of garbage bins can be organized more efficiently in big cities with the help of sensors installed in waste management systems [12]. When bins are emptied regularly, the number of vermin in the city is easier to control. In addition, consumption of gasoline can be optimized as the drivers can follow filling of the garbage bins in real time using an application. Sensors can be installed to monitor traffic and to decrease traffic jams [13]. Also, inhabitants can be offered services with which they can easily follow rush hour times using their own mobile application and therefore avoid rush hour traffic. Sensor systems can be installed to the city's own car parks, which would give real time information to the citizens about free parking spaces, as it would be helpful to have this information already before entering the city. If parking spaces were easier to find, it would bring customers to the services more quickly and it would also reduce pollution caused by fumes in the city's air. By installing a wireless network and a monitoring system in vehicles of public transport, citizens get better information about the real time location of busses, trains, trams, and underground trains, making their use more easily optimized for the citizens.

However, the most important aspect in planning a smart city is its safety [14]. Devices, which are installed to gather data, can be easily damaged physically and their operation can be disturbed electrically. When installing sensors, it must be taken into consideration that it is possible to protect them from natural forces, but also from vandalism. Usually the sensors do not send sensitive data, but if, for example, a signal sent by a parking system is disturbed, it can cause problems for the citizens in finding a parking space. This problem can also become bigger if a citizen loses his or her trust in the system. From a safety point of view, the damage can be quite large if somebody messes up air conditioning systems, therefore causing economic losses in the form of energy consumption, and in the worst-case scenario permanent damage to the building. Disturbing systems in hospitals or rest homes can cause serious health problems. Thus, when connecting into the systems of a smart city, it should be carefully considered which equipment are included in the system. These safety issues and related questions are examined in detail in chapter 4. The following section considers IoT systems used in the industry.

2.4 IoT in industry

Among the major users of IoT devices are various industries [15]. Sensors in various machinery of industrial plants is already the norm. Thus, when they are connected to the Internet, also the industry moves into the digital era and begins to use IoT devices. With these sensors the reliability of maintenance can be improved and the need for repairs can also be predicted better. If a device with an installed sensor has a part which is about to break, the device sends this information directly to the central computer. The maintenance team gets the information and can begin preparing its repair even before a problem occurs.

When sensors are installed to production lines, information can be transferred directly from there to controlling units, which improves optimization of production [15]. These systems require a large amount of processing power, data transfer, and saving space. At this stage cloud services are also involved. Cloud services enable to break away from physical machine halls which require large spaces in the industrial plants and require their own maintenance team. New types of problems occur when factories begin to use IoT systems. An example of these kinds of problems is network overloading. In the planning stage of the factory's systems, one must indeed consider which information is relevant to transfer from the production line to be calculated in a cloud ser-

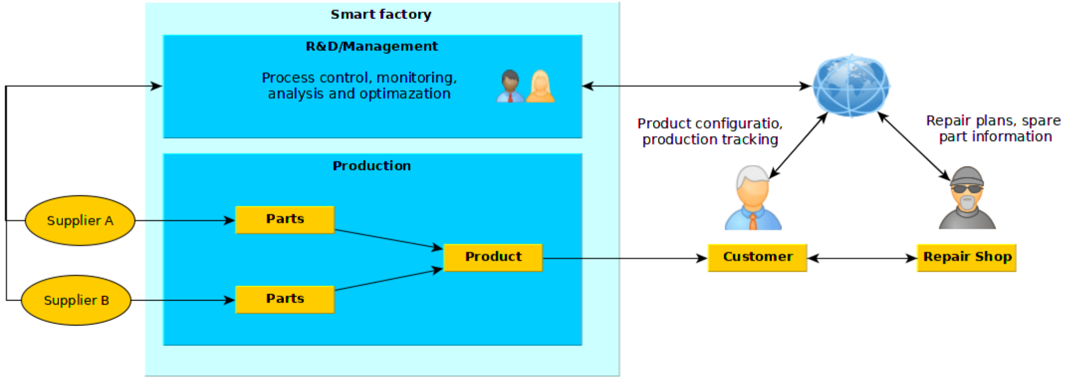

Smart factory
R&D/Management
Process control, monitoring, analysis and optimazation
Production
Supplier A
Supplier B
Parts
Parts
Product
Product configuratio, production tracking
Repair plans, spare part information
Customer
Repair Shop

Figure 2.3: Transferring information from the production line to be calculated in the cloud service

vice. It is useful to leave part of the information for the devices to calculate and only transfer the final, already processed data to the cloud service. This way the amount of data overloading the network could be reduced, at least to some extent. When moving to using IoT devices, one must also consider the changing conditions of information security. Data of the devices is usually sent making use of wireless networks, for example WiFi. If the device sends sensitive data from the perspective of the factory's production, the factory must ensure that this information does not leak into the wrong hands. Furthermore, a denial of service (DoS) attack is possible when using wireless network technology between devices and the controlling unit. This type of attack can cause a major economic setback for the factory, or even casualties [15].

Chapter 4 examines problems related to information security in IoT devices used in industry more carefully. However, the main goal of this study is the planning of a smart home and penetration testing of its IoT device network in a virtual environment, which is why IoT devices in the industry is only covered superficially. IoT network technologies are examined in the next chapter, where the most common network technologies used in IoT devices are explore.

Chapter 3

IoT network technologies

This chapter examines the most common network technologies IoT devices are using, beginning with a wired technology called Ethernet [16]. Wired technologies are, in general, more reliable than wireless, however, it is usually more cost-efficient to install wireless devices. Moreover, wireless devices are also easier to change and replace. In relation to wireless networks, the first one presented here is a technology called wireless local area network (WLAN) [17], which is still the most common wireless technology in use regarding the IoT device market. The second wireless technology examined in this chapter is Bluetooth, which is the most popular technology in headphones and smartwatches. LPWAN, which is becoming increasingly popular in the IoT device market, is presented fourth in this chapter. This section covers those low-power wide-area network (LPWAN) technologies which are the most utilized, for example NB-IoT, LoRaWAN and SigFox. Each of these technologies are briefly introduced; how they have been developed and what kind of qualities they possess in relation to IoT devices. Furthermore, it is also examined whether their information security is on a level where it is considered to be.

3.1 Ethernet

Ethernet, or more formally known IEEE 802.3, is a standardised data transfer technology and nowadays the most popular computer local area network. The first standard of this technology was publicly introduced in 1983. Competing standards were Token Ring, FDDI (Fiber Distributed

Data Interface) and ARCNET (Attached Resource Computer NETwork).

3.1.1 History of Ethernet

The first stages of Ethernet can be traced to a research centre in Palo Alto, where it was developed for the use of Xerox Alto computer in 1973 [18]. The network, which nowadays has a data transfer speed of over 10 Gb/s, then had a data transfer speed of only 2,94 Mb/s. The development of this network proceeded quickly and already in 1995, the data transfer speed was 100 Mb/s. At the time, different versions of the network were developed under the name Fast Ethernet. The version called 100baseTX remains in use today [19].

The Fast Ethernet technology was standardised by the IEEE. The problem was that when data transfer became faster, the length of a single cable was reduced 100 metres from 2,5 kilometres. With the Fast ethernet, the data transfer speed of network interface cards in home computers was increased from 10 Mb/s to 100 Mb/s [19].

3.1.2 Security of Ethernet

The Ethernet technology is very reliable when the goal is to connect IoT devices to the same network. The advantages of this technology are that it is free to use, it does not need a separate license, it is fast, and a distance of over 100 meters can be reached with only one cable. In addition, when considering the safety of Ethernet, it is much harder to eavesdrop Ethernet than a wireless connection. The Ethernet can be used in local area networks (LAN), metropolitan area networks (MAN), and wide area networks (WAN), depending on the network architecture and network area size [20]. Disadvantages are that Ethernet always requires cabling and large connectors.

Ethernet is useful at least in home and office environments. When devices are connected to each other with cables, it is considerably harder for the attacker to listen to the data traffic. In this case the attacker needs to connect his or her own computer to the same internal network with a physical connection or to penetrate a firewall. In wireless communication systems, the attacker can listen data traffic by simply capturing the wireless signal. By connecting low energy IoT devices with an Ethernet cable, the same cable can be used for network traffic and for taking the energy

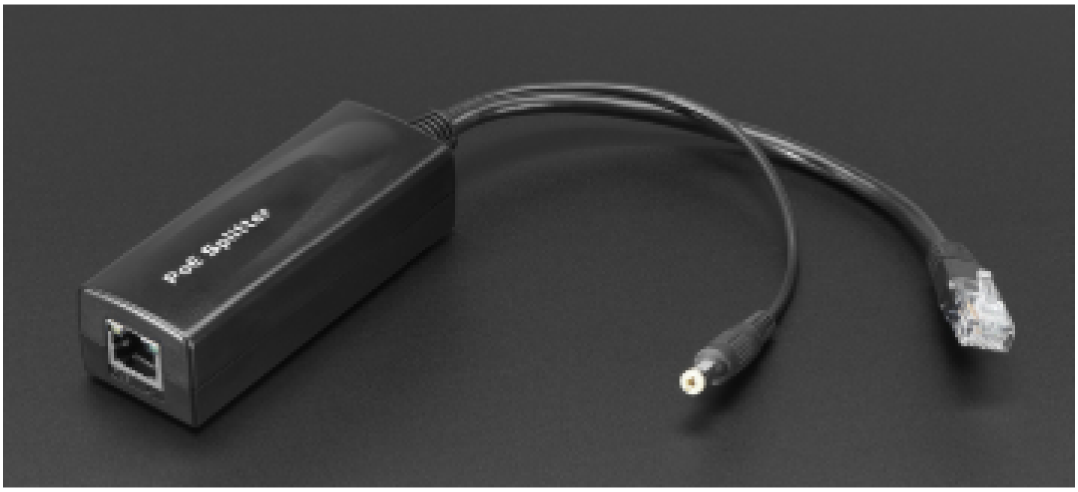


Figure 3.1: Power over Ethernet adapter (PoE)

for the use of the device. For this purpose, specific PoE (Power over Ethernet) devices have been developed (see figure 3.1) [21].

3.2 Wireless local area network

Wireless local area network (WLAN) is a wireless data transfer technology which is based on the IEEE 802.11 standard [22, p. vii]. It is more commonly known by its commercial name WiFi. When a household is connected to this network, a modem is the first device to be installed. The modem can also include a wireless router with which a wireless local area network (WLAN) can be created. If the modem does not include such technology, a separate wireless router can be connected to it, thus creating a WLAN.

A wireless network is usually the easiest way to connect devices such as laptops and smart phones to the internet. Nowadays laptops and smartphones have integrated wireless adapters, and it is therefore easier to make a connection to the router in the household. Desktop computers are usually connected to the router with an Ethernet cable, but if this is not possible for some reason, there is a separate wireless adapter working with USB or wireless adapters which can be installed directly to the desktop computer's PCI channel (see figure 3.2). With these adapters, it is possible to connect this device to a WLAN. When wireless technology is used to connect devices to the network, extra wiring in an apartment is avoided and the connected devices can also be moved

Figure 3.2: TP-LINK wireless USB adapter

within the network's range without losing their network connection.

The usability of the technology is also one reason for this technology being nowadays the most common technology for data transfer, for sensors to report temperature in homes, and for adjusting brightness in lighting. With a wireless technology the information from sensors can be sent directly to the main device at home, from where the user can control their functions. If the main device is connected to the heating system and electrical distribution centre in the apartment, the information from these sensors can easily be used for adjusting the temperature and lighting in the apartment. Furthermore, these functions can be automatized.

3.2.1 WLAN standards and security

In September 1999, the IEEE published amendments 802.11a and 802.11b, which were improvements to the original standard 802.11, published in 1997 [22, pp. 7-8]. The original standard 802.11 works in the 2,4 GHz frequency band and it has a maximum transmission speed of 1 Mbit/s or 2 Mbit/s, depending on the modulation used. The new version 802.11b is working in the same frequency band 2,4 GHz, but transmission speed has been increased to 5,5 Mbit/s or 11 Mbit/s. The next amendments 802.11a used the 5,5 GHz channel and the transmission speed was increased to 54 Mbit/s. The cost of usability of wireless technology is also one reason why the amendments

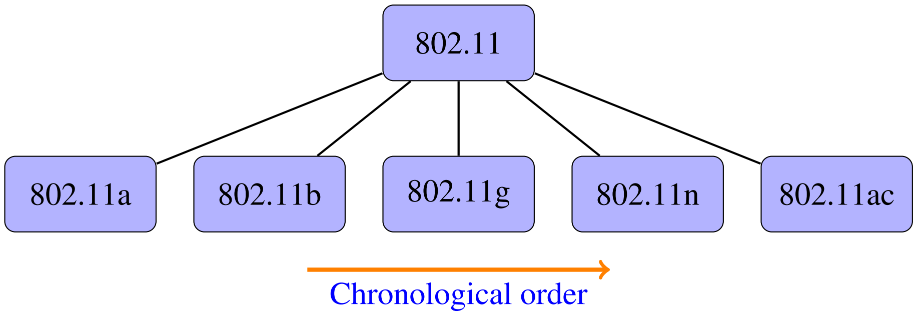


Figure 3.3: 802.11 family amendments from the oldest to the newest

802.11a did not become the main WLAN technology. When the frequency band moved from 2,4 GHz to 5,5 GHz, the network range decreased [22, pp. 7-8]. When amendments 802.11b were published, the IEEE developed a new encryption security algorithm called Wired Equivalent Privacy (WEP) [23, pp. 216-218]. The main idea of this new encryption algorithm was to encrypt the transmission between a wireless device and the router so that it could not be eavesdropped. WEP is based on the RC4 encryption algorithm which was included in RSA Security [23, pp. 216-218]. The WEP encryption technology is already an outdated encryption technology and it is very easy to break with basic penetration testing tools. The IEEE published a new amendment 802.11g in summer 2003 [22, pp. 7-8]. 802.11g has the same transmission speed of 54 Mbit/s as standard 802.11a, but this speed is available in the 2,4 GHz frequency band. When 802.11g was published it replaced 802.11b in the market [22, pp. 7-8]. 802.11g was also back compatible with 802.11b and it had a lower manufacturing cost than 802.11a. In the same year that the 802.11g amendment was published, the IEEE also developed a new encryption technology. This new encryption technology was needed because there were already so many vulnerabilities in the WEP encryption that it was not secure anymore. The new encryption technology called Wi-Fi Protected Access (WPA) used the Temporal Key Integrity Protocol (TKIP) to increase security [23, pp. 219-220]. This new encryption technology used the same RC4 encryption algorithm than WEP, but with TKIP, and the size of encryption keys could be increased from 64-bit to 128-bit [24, pp. 55-57]. A year later, in 2004, the IEEE published amendments 802.11i which included a new WiFi Protected Access II (WPA2) was published at the same time [22, pp. 7-8]. The new encryption algorithm called

WPA2 included support for Advanced Encryption Standard (AES) which replaced the old RC4 encryption algorithm. In October 2009, the IEEE published a new 802.11n which supported transmission speed up to 600 Mbit/s [22, pp. 7-8]. When the IEEE published this new amendment, they had to decrease the transmission speed back to 54 Mbit/s because old devices did not support the new speed. The new amendments 802.11n is more commonly known as WiFi 4. The next new amendment that the IEEE published in 2013 was 802.11ac, more commonly known as WiFi 5 [22, p. 3]. In WIFI 5, the technology was improved, making the maximum transmission speed over 1 Gbt/s. In 2019, the IEEE published a new amendment 802.11ax, more commonly known as WiFi 6 [22, pp. 33]. WiFi 6 included an improved encryption algorithm WPA3, but it will take some time for it to become the main encryption technology in the market.

3.3 Bluetooth and BLE

Bluetooth (BT) is the most used technology between two devices which are in short range [25, p. 3]. Bluetooth is used, for example, between a smart phone and wireless speakers. BT was created to replace the old infrared technology. The range in this old technology was almost the same than in the new BT technology, but the problem in that technology was the connection. When two devices using infrared technology try to connect and keep the connection up, these devices have to stay in a line-of-sight the whole time. Bluetooth uses radio frequency (RF) and therefore does not need line-of-sight. Bluetooh Low Energy (BLE) is a lightened version of the original BT. BLE is very useful if the IoT devices are using low energy and the data amounts sent by these devices are small. These kinds of devices include, for example, remote controllers and heart rate monitors [26, p. 3]. The founder of BT was the Swedish mobile phone manufacturer Ericsson. Ericsson created the BT technology together with the BT founder group in 1998 [25, p. 3]. Ericson, Nokia, IBM, Intel and Toshiba belong to the Special Interest Group (SIG) and the primary goal of this group was to create a new standard *de facto*. The name Bluetooth comes from the Viking king Harald Bluetooth, who managed to unite several families under one kingdom. The SIG had the same idea when they wanted to unite several communication protocols under one standard. The first device that utilized BT technology were mobile phone speakers developed in 1999. Such a device won the prize for ”Best of Show Technology Award” in the COMDEX competition which was organized also in 1999 [27]. After the new BT technology became more common, the first mobile phones which utilized BT technology also began coming to the market in 2001 [27].

BT works with a master-slave principle. One of the devices is the host, which will share the network to the public, and the other device, which is the slave, tries to connect to this shared network. A BT device can share a piconet (see figure 3.5), and eight devices can be in the same piconet at the same time. In this case, one device must be the master and the seven others will be slaves. A device can be a master only in one piconet at the same time, and if the device is a slave, it can only communicate with one master at a time [25, pp. 8-11].

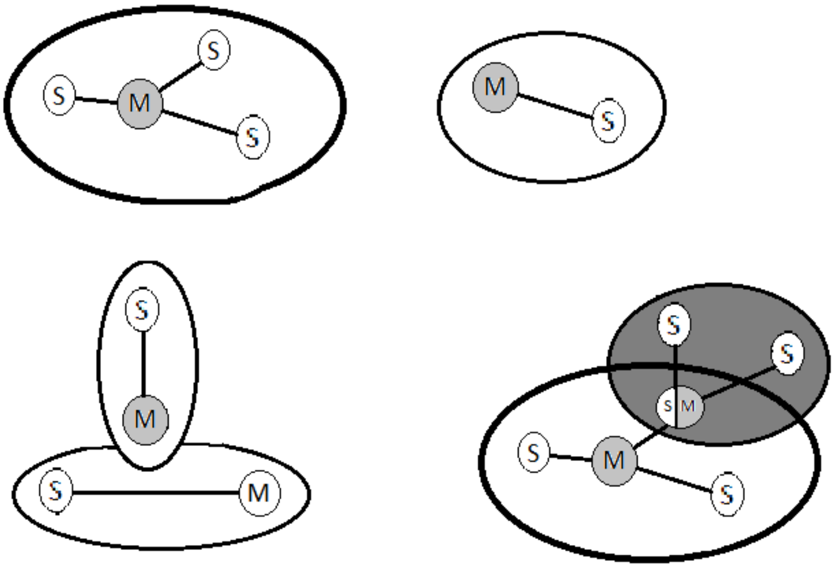


Figure 3.4: Different kinds of piconets [25, p. 9]

The first thing when trying to connect BT devices is to turn on the BT signal. A BT device sends an inquiry message to the network and repeats it for a couple of minutes. When another BT device turns on the BT and starts scanning the environment, the unconnected device enter in the inquiry scan state which means that the device wants to be discovered and these devices can find each other. Both devices are supposed to be in the same area when messages are sent. If the devices already know each other, this stage can be bypassed. In this case, the devices find each other when the signal is turned on.

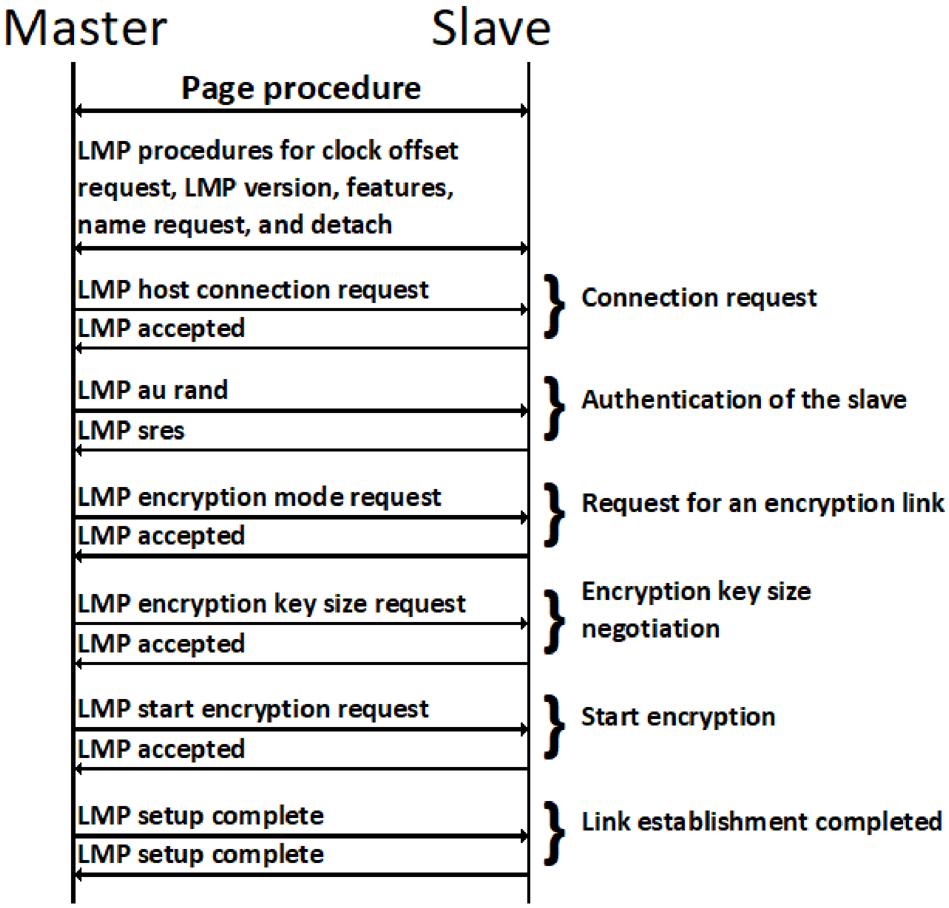

Master
Slave
Page procedure
LMP procedures for clock offset request, LMP version, features, name request, and detach
LMP host connection request
LMP accepted
Connection request
LMP au rand
LMP sres
Authentication of the slave
LMP encryption mode request
LMP accepted
Request for an encryption link
LMP encryption key size request
LMP accepted
Encryption key size negotiation
LMP start encryption request
LMP accepted
Start encryption
LMP setup complete
LMP setup complete
Link establishment completed

Figure 3.5: An example of a Bluetooth connection establishment[28]

After two BT devices have found each other in the network, the Link Manager Protocol (LMP) begins creating a connection between them (see figure 3.6). First, the master device sends an LMP connection request message to the slave device, which, in turn, accepts the connection request. Next, the master device sends an authentication request to the slave device and the slave device accepts it. If the authentication is successful, the master device sends an encryption request to the slave device which the slave device then accepts. Now the encryption can start, and the master-slave connection has been made.

Several versions of BT have been developed throughout the years. The BT version 5.0, published in 2016, is the newest version of BT technology [29]. The earlier version BT 4.0 was published in 2010 and it is the first version that includes BLE technology [30, p. 26]. BLE is the most used BT technology in the IoT sector, because of low energy consumption. Short signal range is a problem in BLE technology, but the new version 5.0 has fixed this problem with an almost four times longer signal range [29]. The BT version 4.2 is still the most used version of BT technology because the old devices do not have the technology with which to use the new BT version 5.0. The BT versions 4.1 and 4.2 have also updated the BT security protocol and firmware [30, p. 27]. BT uses frequency band 2,4 GHz, which is the same frequency band that WiFi technology uses. If there is a BT speaker and a strong WiFi adapter in the same space, there might be some interference in connection.

The transmission speed of BT technology can be almost 3 Mbit/s. The BT version 3.0 has new technology in which to add a WLAN connection, increasing the transmission speed to 24 Mbit/s. BT technology is divided to three different classes [31]. Class 1 has an ability to send a signal with a signal strength of over 20 dBm and the signal range can be over 100 metres. Common devices in this class are laptop computers and BT dongles. However, the BT class 1 can operate with classes 2 and 3, but the signal range will be shorter when mixing classes. The signal range of class 2 is under 10 metres. Class 3 is a low energy class and the signal range is under 1 metre. The technology of class 3 is normally used in smart watches and heart rate monitors.

3.4 Low-Power Wide-Area network

Low-Power Wide-Area network (LPWAN) refers to long-range wireless communication technology. These technologies have low energy consumption and they transfer a trace amount of data with to long range. These technologies can be used to build a network which covers a large area and transfers only some bits. LPWAN is used, for example, in smart parking systems or temperature surveillance systems in the industry.

3.4.1 Narrowband IoT

Narrowband IoT (NB-IoT) is a wireless communication technology based on 3GPP standard of LPWAN solutions [32, pp. 16-17]. It is a 4G communication technology designed together by

Nokia, Ericsson, Huawei and Intel in 2016. NB-IoT can utilize the same 4G network as other mobile devices. The newest version of NB-IoT technology was published in 2018.

The data transfer speed of NB-IoT technology is under 250 kbit/s and normally it is used in sensors. However, the signal strength of NB-IoT is over 20 dBm and it can easily penetrate walls and doors. Another advantage compared to WiFi is low energy consumption; an NB-IoT device can operate over ten years with two AA batteries [33]. The ability to use normal 4G communication network gives an advantage to NB-IoT against other LPWAN technologies [33]. With NB-IoT, several hundred of devices can be connected in the same network at the same time. That is why it is very popular in smart parking systems. When an NB-IoT base station is installed in the right place, one base station can cover the whole parking area. In addition, garbage management is using NB-IoT technology to save gasoline costs and decrease pest problems in cities.

3.4.2 LoRaWAN

Low-Range Wide-Area-Network (LoRaWAN) is another LPWAN communication technology that is used in wireless communication in IoT devices [32, pp. 17]. The difference between NB-IoT and LoRaWAN is that LoRaWAN does not have to use public network to communicate. When using the LoRaWAN technology, there is a communication terminal which creates its own network signal. LoRaWAN has three different classes of communication terminals; Class-A, Class-B and Class-C [34].

The class-A communication terminal works with bidirectional communication technology, which is based on ALOHA type communication protocol [35]. The devices in this class have the lowest power consumptions compared to the other two classes, and it answers to the server only when asked.

The class-B communication terminal also has bidirectional communication technology. However, devices in this class can also create their own answering frame. This answering frame is time synchronized and the server is capable to know when this frame is open in a device. This technology needs slightly more power than the class-A technology, but it is still a low energy device.

The class-C communication terminal uses more power than the other two classes because it is awake the whole time. Only when a device in this class sends a message to the server, it cannot listen to the traffic in the other direction. When a device in class-C starts sending a message to the server, it will close the answering frame and open it again after the message is sent.

LoRaWAN devices are extremely energy-effective and still able to send messages over long ranges [35]. Because of their energy efficiency, the amount of data sent by LoRaWAN devices is typically relatively small. The data transfer speed of LoRaWAN devices is limited to 50 Kbit/s and the amount of data to be sent is usually some tens of kilobits at one time. Data transfer of this size is, however, sufficient enough for sensors.

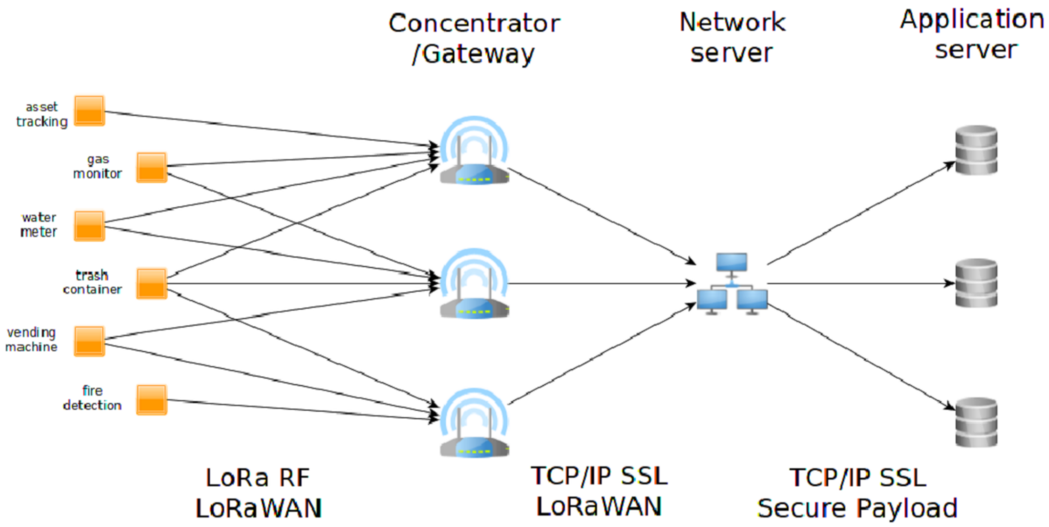


Figure 3.6: The basic structure of a LoRaWAN network

As can be seen in Figure 3.7, the basic structure of LoRaWAN is very simple. The sensors send data to the gateway, from where the data is transferred to the wired network. There are servers in the network which gather and process the data. Information can be sent from the server to the application server or it can be received directly from the application server. The application server in question can be installed to a computer or to a mobile phone.

LoRaWAN is a very simple and safe solution that can be used, for example, in air concentration measurement, smart parking, garbage surveillance, or monitoring fire-sensitive sites. As the sensors themselves do not work directly in the Internet network, devices based on the same technology have to be used in attacking against them. However, when the information is transferred from the gateway to the Network server, normal attacks against the IP protocol have to be taken into consideration as well.

3.4.3 SigFox

SigFox is the third most common information transfer protocol working with LPWAN technology [32, pp. 17]. Originally, it competed with the LoRaWAN protocol, but when NB-IoT came to the market, it has lost its position to some extent [36]. It works with the same radiotechnology as NB-IoT and LoRaWAN but is under a paid licence. SigFox has, however, been on the market since 2009, for the longest of these three technologies. SigFox communication terminals are cheaper than the communication terminals of the other two, but they only work in their own network, making the SigFox protocol much more limited.

SigFox protocol will not be discussed in more detail in this thesis, as its use in smart homes is not as common as the use of LoRaWAN and NB-IoT. LoRaWAN and NB-IoT technologies are also more flexible when they are combined together. It can be concluded that SigFox is a useful technology, which is becoming more common also in the IoT market, but at the moment it is still simpler to operate in a smart home without it.

Chapter 4

The current state of IoT cyber security

In the previous chapters it was discussed what is a smart home and a smart city and how IoT is related to the industry sector. Different data transfer technologies for this environment were also discussed. In this chapter, the first section focuses on the different devices that the smart house environment can include, and examines the IoT data traffic threats, which may arise when designing these environments. When designing a smart house environment, there should be different systems and their subsystems, and all these systems must work together. The various subsystems contain different technologies; thus, the attackers have also several different technologies with which to brake inside the environment. The second section focuses on the current state of IoT security and on different technologies how an attack against these devices works and which programs the attacker possible uses.

4.1 Devices in smart home system

The first things that come to mind when talking about smart houses are sensors and speakers that talk back. There are a lot of different sensors that can be installed to the house. These sensors can measure temperature, moisture, and motion in the house. However, installing sensors to survey a house is not the same thing as a smart house. When starting to build a smart house, the whole environment has to be made smart. There should be an advisor who tells the user what is secure and how the devices should be installed when designing this kind of house.

When a new user starts designing a smart house, the first device that the user buys is normally a smart television. The prices of smart televisions are lower than a couple years ago, and that is one reason why more people decide to buy it nowadays. A smart television is similar to a normal television, with an addition that there is a possibility to use the Internet via the television. A smart television is like a smart phone with a bigger screen; the user can install applications, games or streaming services on it. Many smart televisions already have a Netflix button included in the remote controller. If the user wants to use these new technologies in the smart television, there has to be an Internet connection. A connection can be made with an Ethernet cable or a wireless technology. Even though the user can attach a smart television to the Internet, it is not a pure IoT device.

When the user needs a wireless network for the devices, there has to be a router which has the ability to support wireless technology. Usually, devices such as laptops, smartphones, and smart televisions need a wireless connection. These devices, with other connected electronic devices, form a LAN system to a smart home. There should be malware programs and antivirus programs along with a secure password to secure these devices. Also, when the user has a possibility to check updates from these devices, it is highly recommended to do it every month.

Another part of the smart house system is independent devices, for example temperature sensors, alarm systems and motion sensors. These devices also need a control device which is connected to the smart house router. The heating system can have sensors controlling the temperature of the smart house and sensors in the water supply system alarm if there is a leak. The user can attach a motor in a window shade and make it work automatically, and lights can be changed to smart lights which can also be an automatic. There is also a smart lock system that can be attached to the outdoor of a smart house and a smart wall plug which can be attached to a normal wall plug system changing it to a remote-controlled system. The advantage in these smart wall plugs is that when the user is away from home, he or she can check if there are any devices switched on and switch them off remotely.

In a smart house system, the security is increased compared to a normal house. The user can attach motion sensors to windows and outdoors. There is also a surveillance camera system which the user can attach outside and inside of the house. The advantage of these security devices is that the user can notice if there is an unwanted person on the yard, possibly trying to get inside the house. The system can be modified to make an alarm to the police or to the firefighters if the situation requires it.

When all these devices are connected to the main controlling system, the main infrastructure of smart house architecture is in place. The user can attach more devices to the smart house system afterwards. However, when doing this, it always has to be checked that every device installed before, works without conflict with the new devices. When the whole system is built, the last thing is to connect the smart house to the global Internet. The connection between the smart house and the Internet should be made secure and if it is possible, it should be a VPN connection.

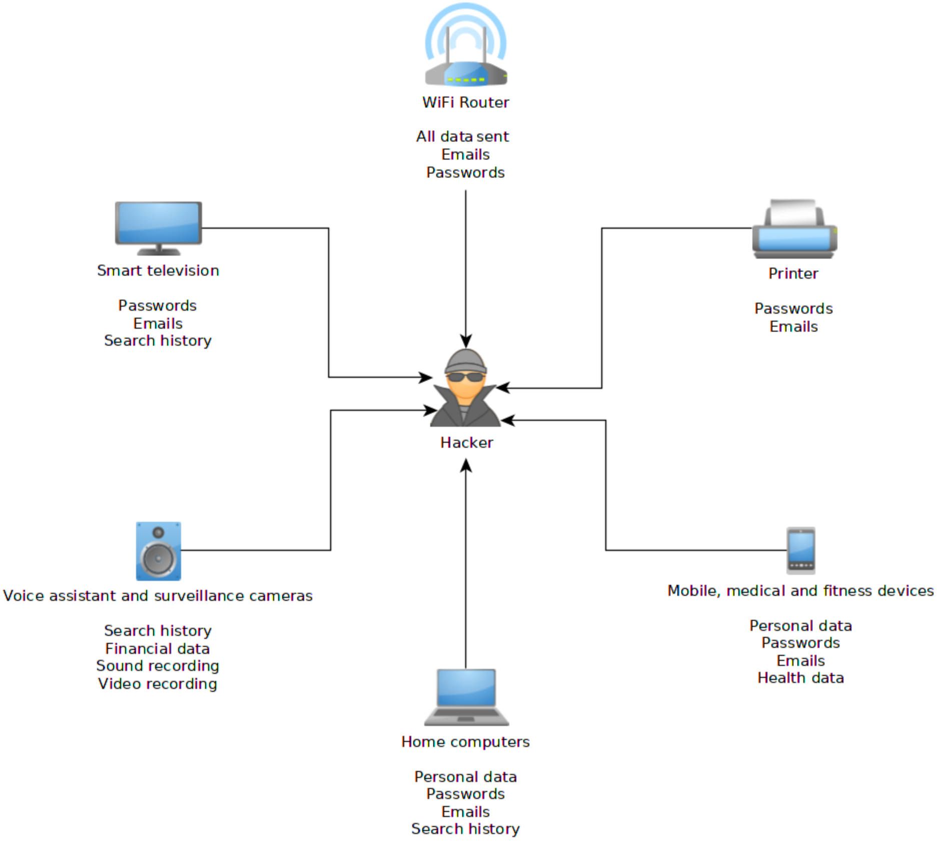


Figure 4.1: Sensitive information that an attacker could steal from an IoT device

4.2 Cyber attacks against IoT devices

Cyber attack is an attack that happens in the cyber network (see figure 4.1). The attack can shut down the devices, make a denial-of-service (DoS) against the devices, the devices could be attached as part of a botnet or the attack can be a man-in-the-middle attack (MITM). The purpose of an MITM attack is to steal sensitive information from the network users. Attacks against IoT devices have increased at the same time as the number of IoT devices has increased. The biggest problem in IoT devices is the security of these devices. The manufacturers do not make them very secure and the people who buy IoT devices do not understand information security enough to demand better security from the manufacturers. Setting some standards for the manufacturers has been under discussion in Finland, and in November 2019, the first manufacturer earned a certification mark for their smart devices. This new certification mark is based on the EN 303 645 standard and guarantees to the customer that the smart device is secure [37].

4.2.1 DoS and DDoS

DoS and Distributed DoS are one of the most used attack types [38, p. 304]. These are very easy to execute and do not need a lot of resources. The main purpose of these attack types is to make so much traffic to the network system that it crashes down. The attack can be made by a young hacker who is testing his or her skills, by an activist group, by some country's government, or it can be a diversion and the real attack is happening somewhere else.

If the purpose of a DoS attack is to shut down a bank network or some other big company network, there must be enough resources to do that. To make more resources for an attack, the attacker normally spreads malwares to the public network and these malwares are waiting in the devices for the attacker's command. When the attacker has many devices infected, it is called a Zombie army or a Botnet [39]. The reason for making a botnet is that an attacker wants more units with which to make a connection to the target. If the attacker makes one request to the target, it will not make any harm, but if the attacker makes 100 000 requests in the same millisecond it probably makes the target server crash down.

Building a Botnet is based on infecting thousands of devices. Nowadays, when the number of

IoT devices has grown, there are smart refrigerators, routers and surveillance cameras which are connected to the Internet. These new IoT devices are not protected enough and there are default passwords and bad security standards in the devices. An attacker can use these devices to make malwares spread faster and to a larger area. A malicious program stays hidden in a device and is waiting for a launching command. When the botnet is large enough, it can be used to a DoS or DDoS attack.

In 2016, one of the biggest DDoS attacks in history was launched. The Botnet, which was used in the attack, was prepared for a long time and the attacker used a malicious program called Mirai to do the attack. Mirai has been made to scan the network automatically and look for vulnerable IoT devices. It uses a default password which the users have not changed when they bought the device. When Mirai finds a vulnerable IoT device, it tries to connect with it, and if it successful in connecting with a device, it infects it with a malicious program and attaches it to as part of a botnet.[40]

4.2.2 MITM

An MITM attack is an attack type where an attacker makes an illegal connection between two devices to eavesdrop traffic [38, p. 306]. The attack is based on the attacker hijacking traffic between devices A and B. When the device A sends a message to the device B, the attacker hijacks this message and reads it or modifies it before sending it to the device B. When the attacker sends it to the device B, the attacker pretends to be the device A. Device B does not know that there has happened anything and thinks that the message has come from the device A. These attacks can be made using different technologies, for example Rogue Access Point (RAP), Address Resolution Protocol (ARP), or Multicast domain names system (mDNS) [38, pp. 306-308].

RAP is a technology where an attacker needs a specific router to make a honeypot [38, p. 190]. A RAP device creates a WiFi network in the target area and waits for a random device to connect to it. For example, if there is a public WiFi, like restaurant or bus station WiFi, it could clone the network SSID and hide the real network. When the victim is looking for a WiFi network with the victim device, there is a network that sounds familiar and safe. The victim connects to the WiFi and thinks that it is a normal public WiFi. In reality, that is the attacker's RAP device and it will connect the victim normally to the Internet, but at same time it will collect all data sent and received by the victim. For example, usernames and passwords to different websites.

The attacker can use the ARP attack technology when he is already connected between two devices. The ARP attack works in an Ethernet network. Every device connected to the Internet has an IP address and a MAC address [38, pp. 306-308]. An IP address is the device‘s network address in the Internet. A MAC address is the device’s own unique identifier; it consists of twelve hexadecimal numbers which are divided to two groups of numbers. The first six numbers tell the name of the manufacturer and the rest are unique number groups for network card address of that particular device. When the device sends a request to the Internet, it makes an ARP-request. Every device which is in the same LAN network with the device that sent the ARP request, sends its own MAC address in response to the ARP request. When the first ARP request has been made and the device which sent the request got an answer, it will be saved to that MAC address in the ARP cache and then it does not have to make the ARP request again. If there is an attacker inside that network, the attacker can hijack that ARP request and read it. After that the attacker can pretend to be someone else and use it to send a false message to the network.

The mDNS spoofing attack is a similar type of attack than the DNS attack, but is working inside LAN [38, p. 305]. It is very easy to be spoofing, because there is a security problem in the mDNS protocol. The mDNS makes a multicast address query and sends it to clients. The clients who listen this request send a name back to the questionnaire. If there is the same name twice, it will take that name who responses first. So, if the user sends data to the IoT device inside LAN, and the attacker machine inside LAN has already cloned the name of that device in an mDNS query, then the data will go to the attacker’s device and not to the user own IoT device.

An MITM attack is a very popular attack when the attacker wants to steal sensitive information from the victim. For example, there was an MITM attack against the VISA card company. The attacker made an MITM attack between the users VISA card and a store payment terminal. When the victim tried to make a payment with a VISA card, the attacker hijacked the card traffic and sent a message to the payment terminal which confirms that contactless payment has already been accepted from the card and at the same time it sends a message to the card that the payment terminal does not need a contactless payment confirmation. The normal limit of VISA contactless payment in Finland is 50 euros, but after doing an MITM attack, the attacker can change the limit to whatever he or she wants [41].

4.2.3 Software attacks

Software attacks is a third type of attack that can be used against IoT devices. This attack type includes, for example, malware, trojan, keylogger, remote control, and buffer overflow style attacks [24, pp. 403-437]. These attacks can be done over the network with an Internet connection or they can connect straight to an IoT device with WiFi or an Ethernet cable. The attacker can find programs from the Internet which can be used to find open ports or to make a brute force attack against the login screen. The attack type depends on the IoT device. If the IoT device has a web interface, like a surveillance camera login screen, brute force might be one way to do it. But if it is a local sensor which does not have a web interface or any other GUI, it is much harder to attack against it and port scanning might give some info on what to do.

When the user wants to increase security in a smart house, he or she will probably buy a surveillance system. Surveillance cameras are a very good way to see if there is are unwanted people on the yard or inside the house. The problem is the vulnerabilities that these devices almost always have. When the surveillance cameras are installed outside of the house and the camera uses a wireless connection, it might be the weakest link of the smart house system. If the attacker can pass the security of this camera and hijack its data traffic, the attacker can also make a link via the camera to the router in the house. If this router has a weak security, the attacker can break into the whole system. Normally, there is a windows pc or a mac computer that is connected to the router, and if the attacker is able to penetrate to the security of the router, he or she has a straight access to the computer. After the attacker has got an access to the computer, he or she can easily install a backdoor to the computer which gives right to remote control that computer any time the attacker wants [42].

When this kind of connection is made, the attacker has an unlimited access to the house computer. The attacker could install a malware or a trojan to the computer and, depending on the software code of the malicious program, it can do some damage to the computer, keeping some ports open or just waiting start up commands from a botnet. The attacker can also install a keylogger to the computer with this malware program and every time the user uses the computer and logins to some private website, the attacker gets all information that the user writes with the keyboard.

A Buffer overflow attack is a technique where the attacker utilizes an unsecure code of a target IoT device for an attack against that IoT device [24, p. 407]. If the attacker had information about the target IoT device, the attacker could use this information to make a script which strains this IoT device. The Buffer overflow technique is based on a program memory buffer overloading of the target IoT device. When there is weak security in the program code and the attacker overloads the memory part of the program, it can leak some data from the memory to the terminal. If the attacker finds a right target, there might be some sensitive information which leaks to the attacker. Another reason to do this attack is to disturb the target system. If the target system is a ventilation system and the attacker stops it from working in the smart house, it can be irritating, or, if the target is in the industry, a ventilation shutdown at a wrong time can result to large expenses, as happened in the Stuxnet case against Iran Nuclear Facilities [43].

A smart lock system can be used on the outdoors of a smart house, which make using the lock system much easier. A smart lock is an electronic lock which the user can control with a smart phone application, RFID tags or pin code [44]. If the smart lock works with a smart phone application, there has to be WiFi or Bluetooth technology inside the smart lock. With WiFi and Bluetooth technologies, there is always the possibility of hacking and if this smart lock had a weak security, it would be very easy to open without permission. There is a website in the Internet where people teach how to hack a smart lock which has a wireless connection [45]. Another technology a smart lock can use is RFID tags. RFID is more secure than a weak WiFi connection, but there are cases where the attacker has successfully cloned an RFID tag and opened a smart lock [46]. The third technology is a pin code. Pin code is the most secure technology after the real physical key, but there are still vulnerabilities, and if the programming code of a smart lock has a weak security, there is a possibility to make a brute force attack or a buffer overflow attack against this smart lock [47].

When the smart house architecture is designed well and every part of the system is protected against physical and cyber attacks, the user gets very much comfort in having a smart house and does not have to fear that someone attacks on the property. However, only one part of the system can be the weakest link and open the door to the attacker. Normally the attackers do not use only one technique, rather, they will use multiple techniques when they want to find a weak point in the system. Many of these techniques they are using, support each other. For example, when using a DoS attack, the attacker can shut down the router and open some vulnerabilities. With these vulnerabilities the attacker can install a malware, a backdoor or a keylogger, or the attacker can use an MITM attack to first open an IoT device and then infect the whole system via that device.

4.3 Pentesting tools and platforms

There are three different kinds of hacker groups and they are commonly known as black hat, white hat and grey hat hackers [48]. Cyber attacks are attacks against telecommunication systems in companies or in the public sector or against private users. The attackers can be part of political groups and they might have some political agenda they want to spread, they can be part of an organized crime business and they want blackmail money or some other financial or personal information, they can be part of a government agency and they are spying other nations, or, they are just young amateurs who are testing their skills.

Cyber defence is the other side of the coin. People who work in this field are called white hat hackers. They are also skilled computer users and their knowledge is similar to that of the black hat hackers. The difference between a white hat and a black hat hacker is what they are doing with their skills. White hat hackers are ethical hackers who are trying to defence telecommunication systems. They are building new antivirus softwares, IDS, IPS and firewalls for the security sector [38, p. 70]. They can also try to compromise telecommunication system security with their penetration testing skills, but there must be a permission to do this from the owner of the system. Companies who sell penetration testing services to other companies are usually hiring white hack hackers. The last group is called grey hat hackers. They are people from both sides; they are not cyber criminals, but they are doing some questionable actions in the network system.

Both the attacker and the defender need good computer skills, but also good tools. For example, if there is a locksmith and he has excellent skills, he still cannot do anything without tools.

The same thing applies for hackers working in the area of telecommunication systems. Already, there are many tools developed for this area, but a good hacker still has skills to make tools him- or herself. The most used operation system in the area of penetration security is Kali Linux [49, p. 2]. Many tools, which the hacker needs to make a penetration test on the websites, against WiFi security, scanning ports or just making phishing sites for testing purposes, are already installed to the Kali Linux operating system. Moreover, there are many websites with online programs that hackers can use for testing.

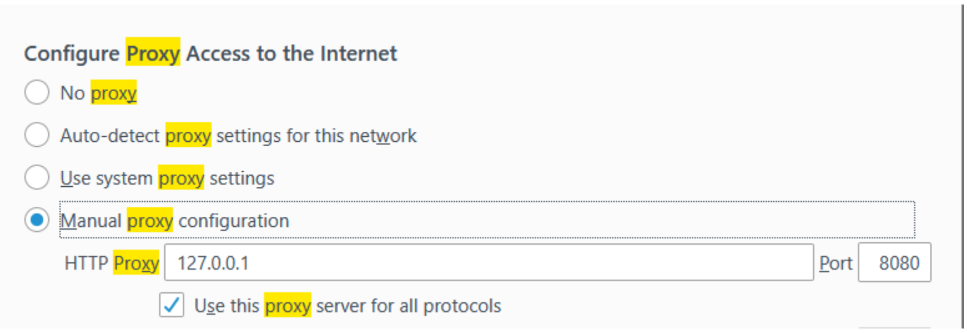
4.3.1 Kali Linux

The Kali Linux distribution is funded by the Offensive Security Ltd and founders of Kali Linux are Mati Aharoni and Devon Kearns [50]. The user can install Kali Linux to the computer's main operating system, to the second operating system with dual boot ability, or, it can be used directly from a CD-ROM or a USB stick. Kali Linux supports many penetration testing tools developed and it is very easy to install new tools to it. Kali Linux also supports most of the WiFi adapters and Bluetooth adapters in the market.

The Kali Linux distribution includes tools for information gathering, vulnerability analysis, wireless attacks, application testing, exploitation, making stress testing, and it also has forensics tools. These tools are made for finding vulnerabilities from websites and softwares. Many of the IoT devices are using wireless technology and when a device is using wireless technology there is always the possibility that someone tries to hack it.

4.3.2 Burp Suite

The Burp Suite is a special software for penetration testing. The users can use it for monitoring, modifying and stopping Internet traffic from the user's computer. It is a particular kind of MITM proxy, including special tools that can be used for attacking websites. There are three different versions of Burp Suite: free community version, a Professional Edition and an Enterprise Edition [51]. The free community version has the basic tools for penetration testing. The Enterprise Edition has a Web vulnerability scanner and Scheduled and repeat scanners in addition to the basic tools. The Professional version has a Web Vulnerability scanner and advanced tools for penetration testing. In November 2019, the price for the Enterprise Edition was 3499 euros per year and for the Professional Edition the price was 349 euros per year. The community version is free of charge [51].



Configure Proxy Access to the Internet

○ No proxy

○ Auto-detect proxy settings for this network

○ Use system proxy settings

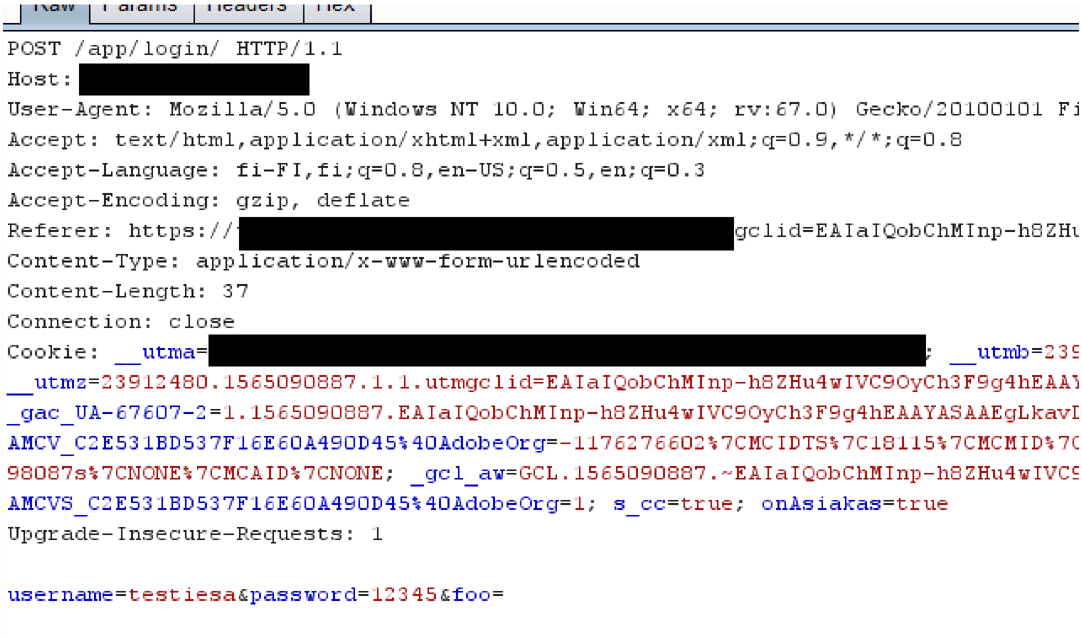
● Manual proxy configuration

HTTP Proxy 127.0.0.1 Port 8080

☑ Use this proxy server for all protocols

Figure 4.2: Firefox proxy settings

The first thing when starting to use Burp Suite, is to setup a proxy in the browser's settings. The proxy address is normally setup to localhost 127.0.0.1 and the port to where it is connected can be 8080 (see figure 4.2). These settings allow the Burp Suite software to read data traffic between the browser and the Internet. There is an ability to stop traffic, repeat it and modify it



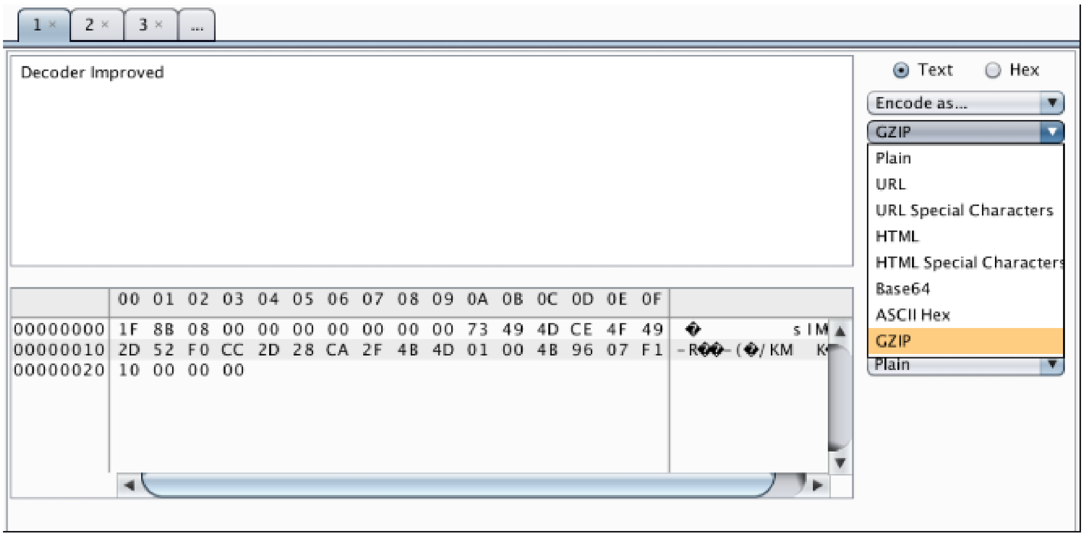
```
POST /app/login/ HTTP/1.1
Host: [redacted]
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:67.0) Gecko/20100101 Fi
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: fi-FI,fi;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: https://[redacted]gclid=EAIaIQobChMInp-h8ZHu
Content-Type: application/x-www-form-urlencoded
Content-Length: 37
Connection: close
Cookie: __utma=[redacted]; __utmb=239
__utmz=23912480.1565090887.1.1.utmgclid=EAIaIQobChMInp-h8ZHu4wIVC9OyCh3F9g4hEAAY
_gac_UA-67607-2=1.1565090887.EAIaIQobChMInp-h8ZHu4wIVC9OyCh3F9g4hEAAYASAAEgLkavI
AMCV_C2E531BD537F16E60A490D45%40AdobeOrg=-1176276602%7CMCIDTS%7C18115%7CMCMID%7C
98087s%7CNONE%7CMCAID%7CNONE; _gcl_aw=GCL.1565090887.~EAIaIQobChMInp-h8ZHu4wIVC9
AMCVS_C2E531BD537F16E60A490D45%40AdobeOrg=1; s_cc=true; onAsiakas=true
Upgrade-Insecure-Requests: 1

username=testiesa&password=12345&foo=
```

Figure 4.3: Parameters in Burpsuite

before sending. For example, when the browser gets the login page from the Internet and there are username and password textboxes, the user can send a fake username and password back to the server with the Burp Suite software. Then, when the server answers to the user, there can be a plaintext or an encoded answer from the server whether the authentication is correct or not. If the message is not encrypted well enough, the user can open it and change it before it arrives and get an access to the site without permission.

The Burp Suite software can also be used to make a brute force attack against the login pages of websites. When the user has already stopped traffic between the browser and the server (see figure 4.3), the user can send this HTML page to the repeater tool. With the repeater the user can change the username and password in the textboxes and repeat the send command after this. If the user has a password library which is large enough, he or she can use intruder tools in the Burp Suite and test all of these passwords automatically. The problem in this method is that it will take a lot of resources and time to crack the password via the brute force attack. If the password is made according to good password standards, it will take several years to solve it.

One of the useful tools in the Burp Suite software is a decoder (see figure 4.4). The user can copy part of the encoded text and send it to the decoder. The decoder has eight different encoding/decoding methods that the user can use to encode/decode a text. With spider tool the user can check what kind of different folders there are in a website folder tree, which makes



Decoder Improved

```
          00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
00000000  1F 8B 08 00 00 00 00 00 00 00 73 49 4D CE 4F 49   �          s IM
00000010  2D 52 F0 CC 2D 28 CA 2F 4B 4D 01 00 4B 96 07 F1   -R��-(�/KM   K
00000020  10 00 00 00
```

Figure 4.4: Decoder

attacking to a website much easier. Furthermore, it is possible to check open ports from a website by using an intruder tool, which works like an NMAP tool. This software is very useful against IoT devices if an IoT device has a web interface.

4.3.3 WireShark

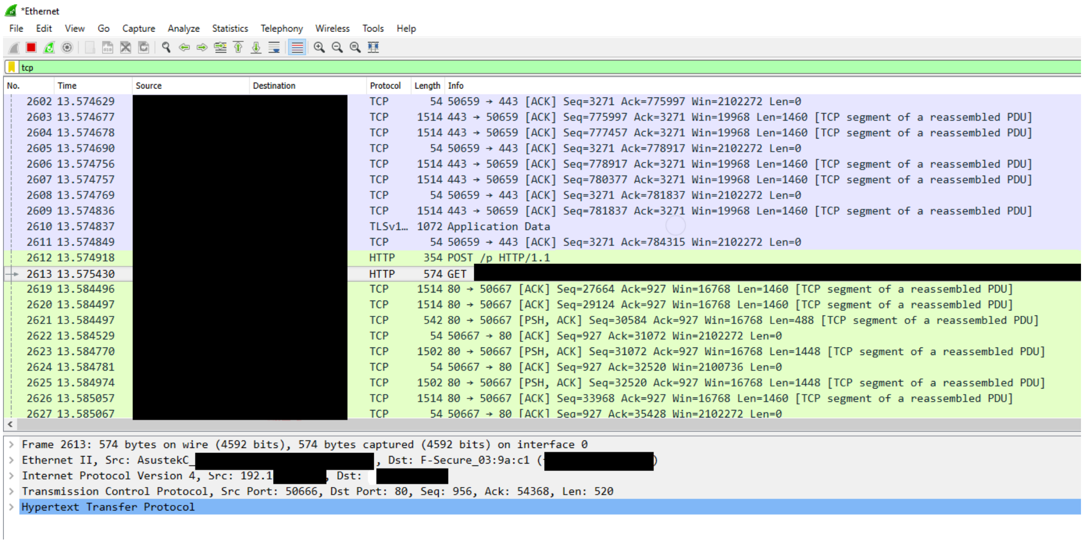
WireShark is a software which is made for monitoring and analysing Internet traffic [52]. It is an open source software founded in 1998 [52]. WireShark is based on the same technique than the Tcpdump software, but a graphical interface has been added for easier use. With the WireShark software the user can monitor Internet traffic and save data to a PCAP file. After WireShark version 1.4, an ability to monitor a wireless network has been included in WireShark [52]. To monitor a wireless network, the user has to change his or her wireless adapter to the monitoring mode before starting the WireShark monitoring. With this ability it is also possible to monitor the network traffic of an IoT device if the IoT device has a wireless connection.

The amount of data is huge when using the WireShark packet capture tool. When the user wants to read data from the packet, it will be a very painful task. Luckily, there is a filtering tool to make it easier. The WireShark filtering tool includes many different filters, for example an IP address filter, a protocol filter and a port filter. With the IP filter the user can filter either the source or the destination address. With the protocol filter, the user can choose for example HTTP, TCP or UDP protocol (see figure 4.5). If there is not good encryption or not encryption at all in the data, the attacker can use that and read the user's username and password easily from the data.

4.3.4 Shodan

Shodan is the most dangerous search engine in the Internet [53]. Every device connected to the Internet can be found with Shodan. With Shodan, an attacker can search open and unprotected ports from the Internet. Registration is not necessary if someone wants to start using Shodan. However, if the user wants to use all abilities in the Shodan webpage, registering is required.

Because of the abilities Shodan includes, it can be used to penetration testing purposes. However, it can also be used as a very advanced attacking tool. The user only has to put a search word


*Ethernet
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
tcp

```
No.    Time       Source  Destination  Protocol Length Info
[redacted]
2602 13.574629  TCP      54 50659 → 443 [ACK] Seq=3271 Ack=775997 Win=2102272 Len=0
2603 13.574677  TCP    1514 443 → 50659 [ACK] Seq=775997 Ack=3271 Win=19968 Len=1460 [TCP segment of a reassembled PDU]
2604 13.574678  TCP    1514 443 → 50659 [ACK] Seq=777457 Ack=3271 Win=19968 Len=1460 [TCP segment of a reassembled PDU]
2605 13.574690  TCP      54 50659 → 443 [ACK] Seq=3271 Ack=778917 Win=2102272 Len=0
2606 13.574756  TCP    1514 443 → 50659 [ACK] Seq=778917 Ack=3271 Win=19968 Len=1460 [TCP segment of a reassembled PDU]
2607 13.574757  TCP    1514 443 → 50659 [ACK] Seq=780377 Ack=3271 Win=19968 Len=1460 [TCP segment of a reassembled PDU]
2608 13.574769  TCP      54 50659 → 443 [ACK] Seq=3271 Ack=781837 Win=2102272 Len=0
2609 13.574836  TCP    1514 443 → 50659 [ACK] Seq=781837 Ack=3271 Win=19968 Len=1460 [TCP segment of a reassembled PDU]
2610 13.574837  TLSv1… 1072 Application Data
2611 13.574849  TCP      54 50659 → 443 [ACK] Seq=3271 Ack=784315 Win=2102272 Len=0
2612 13.574918  HTTP    354 POST /p HTTP/1.1
2613 13.575430  HTTP    574 GET [redacted]
2619 13.584496  TCP    1514 80 → 50667 [ACK] Seq=27664 Ack=927 Win=16768 Len=1460 [TCP segment of a reassembled PDU]
2620 13.584497  TCP    1514 80 → 50667 [ACK] Seq=29124 Ack=927 Win=16768 Len=1460 [TCP segment of a reassembled PDU]
2621 13.584497  TCP     542 80 → 50667 [PSH, ACK] Seq=30584 Ack=927 Win=16768 Len=488 [TCP segment of a reassembled PDU]
2622 13.584529  TCP      54 50667 → 80 [ACK] Seq=927 Ack=31072 Win=2102272 Len=0
2623 13.584770  TCP    1502 80 → 50667 [PSH, ACK] Seq=31072 Ack=927 Win=16768 Len=1448 [TCP segment of a reassembled PDU]
2624 13.584781  TCP      54 50667 → 80 [ACK] Seq=927 Ack=32520 Win=2100736 Len=0
2625 13.584974  TCP    1502 80 → 50667 [PSH, ACK] Seq=32520 Ack=927 Win=16768 Len=1448 [TCP segment of a reassembled PDU]
2626 13.585057  TCP    1514 80 → 50667 [ACK] Seq=33968 Ack=927 Win=16768 Len=1460 [TCP segment of a reassembled PDU]
2627 13.585067  TCP      54 50667 → 80 [ACK] Seq=927 Ack=35428 Win=2102272 Len=0

> Frame 2613: 574 bytes on wire (4592 bits), 574 bytes captured (4592 bits) on interface 0
> Ethernet II, Src: AsustekC_[redacted], Dst: F-Secure_03:9a:c1 ([redacted])
> Internet Protocol Version 4, Src: 192.1[redacted], Dst: [redacted]
> Transmission Control Protocol, Src Port: 50666, Dst Port: 80, Seq: 956, Ack: 54368, Len: 520
> Hypertext Transfer Protocol
```

Figure 4.5: WireShark network analyser

to the Shodan filter and Shodan searches it from the Internet. The search retrieves all words which include that word in the device's name. For example, if the word 'camera' is used as the search word, Shodan searches all devices including the word 'camera' in their name. When it finds a device which includes the word 'camera' in its name, it searches for an open port in this device and lists it to the person who made the search. Now this person can check whether these ports have passwords or other authentication methods. Without security the ports can be penetrated, and the attacker can use the interface of these devices to his or her own purposes. Many devices have an ability to install firmware updates via Internet. If the attacker manages to install a new firmware update to a device, the attacker can join the device to his or her bot network with this update.

Chapter 5

Designing smart home system architecture

The purpose of this thesis is to develop a virtual penetration testing environment for IoT devices. This environment can be used to test whether the connected IoT devices have cyber security problems or vulnerabilities by using penetration testing tools. This chapter focuses on the architecture of smart home IoT systems in order to offer a general view of the multifacetedness of a smart home environment and the network technologies used normally in smart homes. First, the focus is on the router, which is the centre device of a smart home system. The router is also an interface between the Internet and the LAN. The second important device of a smart home is a controller. The main purpose of a controller is being the control panel of all the devices in a smart home. Finally, this chapter examines the most used smart devices in smart homes, for example smart lights, smart locks, smart sensors and surveillance cameras.

5.1 Router

The first aspect of choosing a router to a smart home system architecture is to consider how many different network technologies is included in the new smart home system. Basic home computers are normally connected to the environment with an Ethernet cable, but other devices, for example smart sensors or smart lights, might need a wireless technology with which to connect to the router.

Normally families have also tablet computers, laptop computers and smart phones; these are all devices which probably need a wireless connectivity, as tablets and smart phones do not even have an integrated Ethernet port.

The second aspect is usability: when choosing a router to a smart home system, there has to be a good user interface, so it is easy to learn how to use it. Moreover, almost everyone who use a smart home system is not very interested in the technical details of the system. If the router has an ability to connect to a smart phone application, it will give more usability to the user and the user has a possibility to remote control the system when the user is not at home [54]. There is also a possibility to use a web-based user interface, but that cannot be recommended to anyone. The web-based user interface is much more unsecure than smart phone applications and an unauthorized person could break into the user interface via Ethernet more easily [54].

The third aspect of choosing a router is the range of the area that the router has to cover. There is a big difference in wireless technology if the apartment where the smart home system is located is 40 square meters or 200 square meters. If the wireless technology of the router does not cover the whole apartment, there are devices that can be used to strengthen the wireless signal. The prices of these devices start from 20 euros, and the strength of the signal could be increased by 30 to 50 percent. Another alternative is to use the Wireless mesh network (WMN) [55]. The WMN consists of two or more routers which are connected to each other. Together these routers construct a larger connected network area. The WMN helps the user to move inside an apartment without losing the network connection.

The main purpose of the router is to control the network traffic of a smart home system. It is a firewall between the global Internet and the local area network (LAN) at home. When installing a new router to a smart home system, the user should check the router's firewall configuration, because there can be some rules that the user wants to change. Furthermore, it is highly recommendable to check the wireless network settings of the router before connecting it to the global Internet and change the default password on the router interface and the wireless network if there is too weak a password.

5.2 Controller

A controller is a device where the interfaces of all other devices should be connected in a smart home system. If the controller needs a graphical user interface (GUI), it can be a tablet which is attached, for example, to wall at home, or it can be a smart phone application. The router has to have an ability to control the devices if the controller is a smart phone application. Still, if the router has an ability to control the network traffic of the devices, the main user interface of the devices is in the application and the data stream goes from the application via router to the end devices and from there back to the application. When building a new apartment and there are plans to make it a smart home, it is wise to put the Ethernet cable inside a wall and connect it to the tablet version of the controller. It is more secure than using a wireless mobile application because, as always with wireless technology, there is a possibility that someone makes an MITM attack between the devices and the application.

Sometimes there is a controller without a GUI. In this case, there has to be some other way to control the devices. Voice-controlled smart devices have been developed by different manufacturers. For example, Apple's device is called Siri, Google has created its own version called Google Assistant and Amazon has a voice controlled speaker called Alexa. Amazon was, however, the first manufacturer who produced a voice-controlled smart speaker for private customers. Alexa starts to work when the user gives the command "Alexa" after which the user commands what Alexa should do [56]. For example, the user can give the following command: "Alexa, what is the temperature outside". Alexa is connected to the Amazon server via Internet and starts looking for an answer from there. After Alexa has found an answer to the user's question, it tells it to the user with a voice that the user has installed to Alexa's program beforehand. Another example is controlling the devices in a smart home environment. When the user gives a command like "Alexa start coffee machine", it starts a conversation with the coffee machine and tells the user what he or she should do next [56]. There is a security risk in voice-controlled devices. If the data that these devices collect from the user's home stayed in LAN, there would not be problems such as data leakage. But, for example, Alexa always sends all the data that it gathers from the user's home to the Amazon's server. How can we be sure that Amazon does not sell this data to various enterprises?

The security of the controller is the most important part of the smart home architecture. The reason why the controller is such a security risk, is that it is the centre of the smart home system. All data traffic that is being transferred inside LAN goes through the controller components. The login authentication information of the controller has to be secured well enough. When the user buys a new controlling device, changing the password from the controller settings should be the first thing to do. Usually, there is no need to install a web-based interface to the controller. But, if there is no other way to configure the controller settings, a web-based interface has to be installed. In this case, the user should make a localhost or setup a VPN tunnel between the controller and the interface to ensure security and to avoid the possibility of an MITM attack.

5.3 Smart lights

The first smart device bought is usually smart lights. Normally, the user has a smart television before smart lights, but a smart television is a connected device, not an IoT device. Therefore, it is more like a computer connected to the Internet. The number of different smart lights is increasing on the market. Most of these smart lights need a low range wireless technology. For example, Ikea's Trådfri smart lights use a ZigBee technology between the smart bulb and the remote controller [57, p. 6]. This technology is very simple, as the user only has to change a bulb and connect the controller and the bulb together. There is also a mobile phone application, which the user can use to control the light bulbs [57, p. 14]. When using this application, the user has to buy a gateway that is developed for the Trådfri lights. This gateway needs to be connected with WLAN technology to the smart home controller.

Information transfer between the gateway and smart light bulbs are using similar technology than the information transfer between smart light bulbs and the remote controller. This technology has good information security and it is not a reasonable target to attack. The gateway is connected with WLAN technique to the controller and this is one part of LAN network traffic. If the user does not have good security in his or her LAN, there is a possibility to make an MITM attack to information traffic occurring between the controller and the gateway. An attack to the smart lights is not a big expense to the user, but it can be very irritating if someone blinks the user's lights without permission.

5.4 Smart lock

A smart lock is a lock which can be attached to the outdoor of a smart house. It can be remote controlled or a normal, physically controlled lock. The number of users of smart locks is increasing nowadays because the security of smart locks has improved lately. Some locks are working with an RFID tag, some with a remote-control application and some locks have a normal physical panel. Those locks that are using a remote-control application, has to have wireless technology installed. This remote-control application working with a mobile phone can be a security risk.

Vulnerabilities of smart locks can be scanned with special tools if there is WiFi on. There is a possibility to increase the security of WiFi. Users should use a strong enough password and the encryption technology should be at least WPA2. If there is WPS technology, it should be turned off. The WPA2 encryption technology is not enough if there is a weak password. The user should change the password if it is a factory password and check other settings as well. There is a possibility that someone makes an MITM attack to traffic data between the phone and the lock and gets the authentication data from there. When the attacker makes an MITM attack to a smart lock, the attacker can save that data to a PCAP file and use that to decrypt the password from the data. If there was good security and the encryption technology was updated, it would be much harder.

5.5 Smart sensors

Smart sensors are a hidden part of the smart home architecture. The smart sensors can be used to control the heating system and air ventilation at home, detect fire gas or create an alarm if an unauthorized person is moving in the house [58]. The sensor itself does not have to be smart, but when the user puts many sensors in the house, the sensor network can collect information data from the house and steer this data to the user's control panel. With this information the user can be wiser than without it and this is part of the smart house content. For example, the user can collect data from the heating system and use that to control all heating radiators in the house. There can be, for example, a different heating temperature in the bedroom than in the living room. Also, with the controller the user can automatically adjust the temperature of the room to be lower in the morning if the user so wishes. When the user is not at home, the temperature can be slightly lower which is one way to save money in heating costs.

The motion detection sensors have been used many years in people's houses to turn on the outdoor lights, but motion detection sensors in an IoT environment are a new thing. A motion detection sensor can be attached to the outdoor or window of a house and connected to the user's phone application. This is useful when the user is not at home and wants to check that everything is in order in the house. When there is a senior housing community, a motion detection sensor can be attached to the gateway of the smart home system and this gateway can send an alarm to the authorities if there has happened an accident. Another place where the user can use the sensors is in the water supply system. The sensors can monitor that there are no leaks or any other kinds of problems in the water supply system. If a problem occurs, an alarm can be sent to the user. It can save a lot of money if the water damage can be reduced with this warning system.

5.6 Surveillance camera

The smart house architecture which includes a surveillance system could have several outdoor and indoor cameras or only one surveillance camera on the front yard. If the user is not at home, he or she can easily follow what happens inside the house using a phone application. For example, if there is a pet alone at home, the user can check that everything is alright, or if the user is going on a holiday and the house is left empty, he or she can keep an eye on the house remotely and have peace of mind. Indoor and outdoor cameras installed to the part of the surveillance system. If the user cannot attach a camera to the system with an Ethernet cable, it has to be attached with wireless technology. When a camera is using wireless technology to send data to the smart house system, there is always a possibility to an MITM attack.

If a surveillance camera uses a wireless technology and is installed outside of the house, the surveillance camera can be the weakest link of the smart house architecture. When a camera is on the outside of a smart house, there is a possibility that some unauthorized person examines the model and technology of the camera. When the attacker knows the camera's model, he or she can do more research on the Internet and find a possible vulnerability concerning that particular model. For this purpose, there are programs on the Internet, for example Shodan, which was introduced in more detail in section 4.3.4.

When designing a new smart house architecture, it is important to make a route for an Ethernet cable in the surveillance system. If the surveillance camera is connected to LAN in the smart house, the user is supposed to turn off the wireless technology in the surveillance camera. If there is no possibility to attach the new camera to LAN, it has to be part of the wireless system. When the camera is in a wireless network, it is important to check the camera's settings and make sure there is secure configuration. A more secure way is making another isolated LAN and install the surveillance camera to that LAN. It might cause problems if the camera is installed on the outside wall of a house, for example on the front wall of a house to monitor movement on the front yard. When the camera is installed in the house's LAN system, a new wired connection can easily be connected to this device from outside of the house and get access to the house's LAN network if it is not isolated to a separate LAN system. When doing these kinds of different networks inside the house, there is an important question which needs to be considered: what is the balance between usability and security?

Chapter 6

Designing and building a virtual smart home IoT environment for penetration testing education

In this chapter is presented the architecture of the virtual penetration testing environment developed in this study. VMWare virtualization program was used for virtualising the penetration testing environment. This system simulates a simple smart home environment which includes a router equipped with a firewall, a home computer and two IoT devices (see figure 6.1). An old Windows XP system was chosen for the home computer because its vulnerabilities are already commonly known. To be able to use these vulnerabilities in this thesis, the vulnerabilities must be already published, and their patches must be publicly available.

The router chosen for this study is the PfSense virtual router. It was chosen because it is easy to configure, and its firewall is easy to validate appropriately. A router is used to create LAN (Local Area Network) and WAN (Wide Area Network) addresses, to which other virtual machines in the system have been attached with static IP addresses. To the router in the WAN side, the router gives one IP address and open ports for different protocols. Internal traffic of the smart home is connected to the router in the LAN side, and also new IoT devices can be easily connected to it when developing the simulation further. A connection to the simulation can also be opened via a real physical router, to which existing IoT devices can be attached for testing purposes. This is, however, beyond the scope of this thesis, because it is used in demonstrating the functioning of a network, but it is not necessary for research and teaching purposes. The virtual machines installed in the environment are supposed to reflect IoT devices that communicate through the router. The environment is easy to modify to resemble for example Raspberry Pi device, which is commonly used in IoT devices. The whole environment has been set to function with VmWARE virtual ESX server, which has been installed to the university server machines. VmWARE was chosen for the environment partly because of its support services, but also partly because the university already had this system. The system is meant to be used in teaching purposes in the future, so keeping it in the university's own network is thus reasonable.

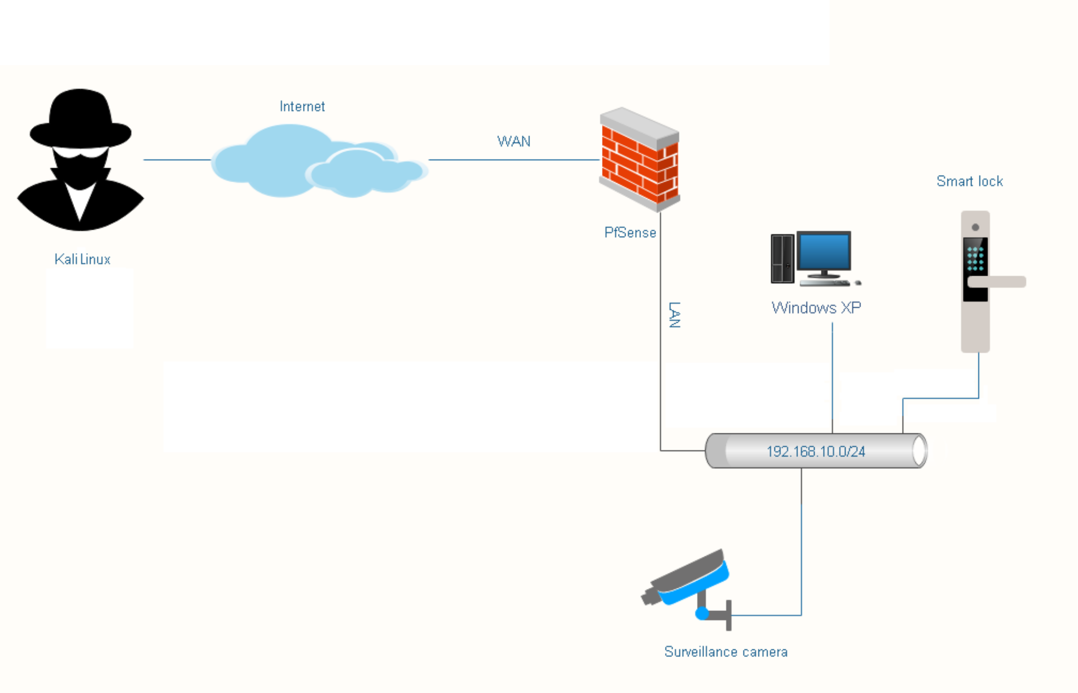


Figure 6.1: Virtual penetration testing environment architecture

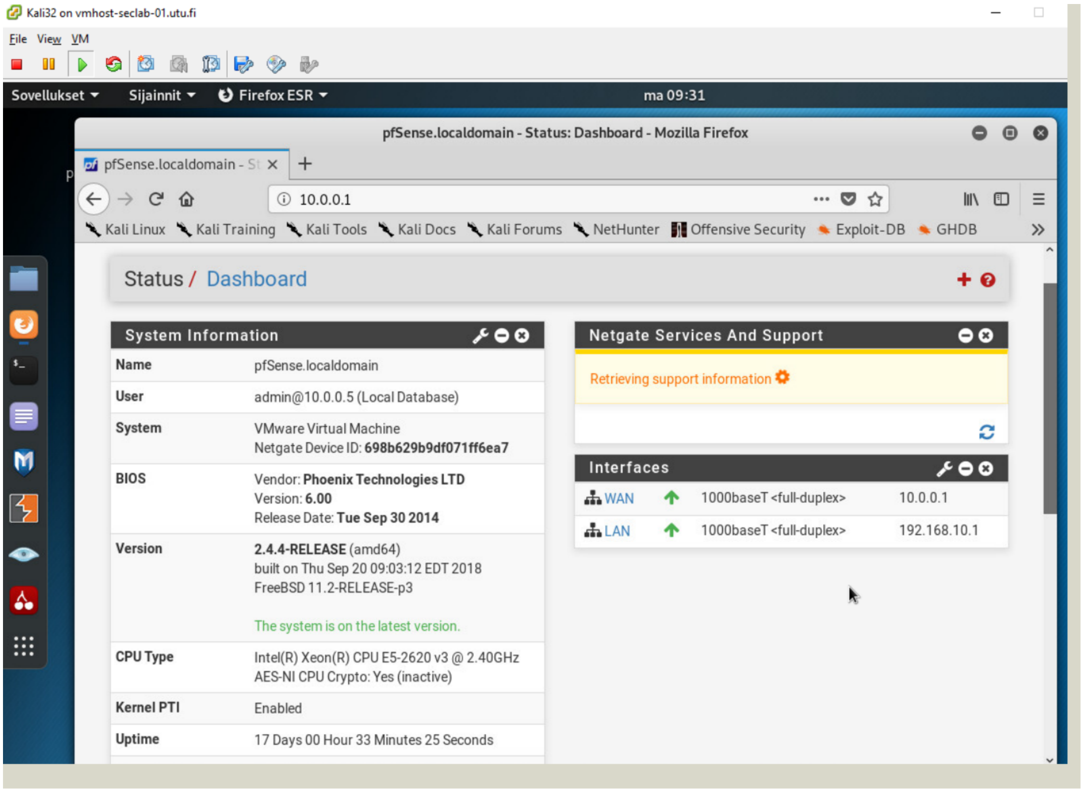
6.1 VmWARE

The simulator has been built to the ESX server produced by VMware. The ESX server is an independent virtual software, in which one can drive several different virtual machines at the same time. Users can be added to the server and users' rights can be governed as desired. By using the ESX server environment, a closed internal network can be built, into which a sufficient number of machines and devices can be connected. For each individual user can be classified a set of rights regarding which virtual machines they have the right to use in the Internet.

Firstly, an own network is created in the environment, and the virtual machines are linked to it. The building of the network begins from its core; therefore, a router is needed. With the router, separate WAN and LAN networks are created in the network, so that it is possible to simulate an environment which resembles a smart home. When the environment is ready, the network in question can be cloned and copied to a requisite number of students. For example, if there are 20 students in a course and it is desired that hands-on-exercises are used in teaching, these students can be divided into groups of two and each group gets a network environment for their use. Next, the student gets a username for VSphere and when registering to the network with a given username and password, the student or group sees only that virtual machine which is meant for them.

PfSense is a virtual router and a firewall which is based on an open source code. It can be installed to function as its own firewall and transfer all the network traffic of the machine through that particular firewall. It can also be installed to function as a router, which is the main purpose of this study. In this study, PfSense has been installed in one virtual machine and it has been given two network cards to function. The other network card can be used to create an outward bound WAN network, wanted to reflect the open world of the Internet.

PfSense can be used to create IP addresses either automatically or manually. A network 10.0.0.0, which is a simulation of an outside Internet, has been created in this study. The attacking machine used in this study is connected to the address 10.0.0.5 in this network. For safety reasons, the network is otherwise closed from outside network traffic. This WAN network would also be possible to connect directly to the right server, but the purpose of this study is to create a safe, simulated testing environment, and therefore it is unnecessary at this stage. The inner network which models the smart house has been created with the other network card of the PfSense router. A LAN network of 25 addresses has been created with the network card. The address of the inner network is 192.168.10.1/24; the /24 at the end of the IP address means that the IP address has a



Status / Dashboard

System Information

Name	pfSense.localdomain
User	admin@10.0.0.5 (Local Database)
System	VMware Virtual Machine Netgate Device ID: **698b629b9df071ff6ea7**
BIOS	Vendor: **Phoenix Technologies LTD** Version: **6.00** Release Date: **Tue Sep 30 2014**
Version	**2.4.4-RELEASE** (amd64) built on Thu Sep 20 09:03:12 EDT 2018 FreeBSD 11.2-RELEASE-p3 The system is on the latest version.
CPU Type	Intel(R) Xeon(R) CPU E5-2620 v3 @ 2.40GHz AES-NI CPU Crypto: Yes (inactive)
Kernel PTI	Enabled
Uptime	17 Days 00 Hour 33 Minutes 25 Seconds

Netgate Services And Support

Retrieving support information

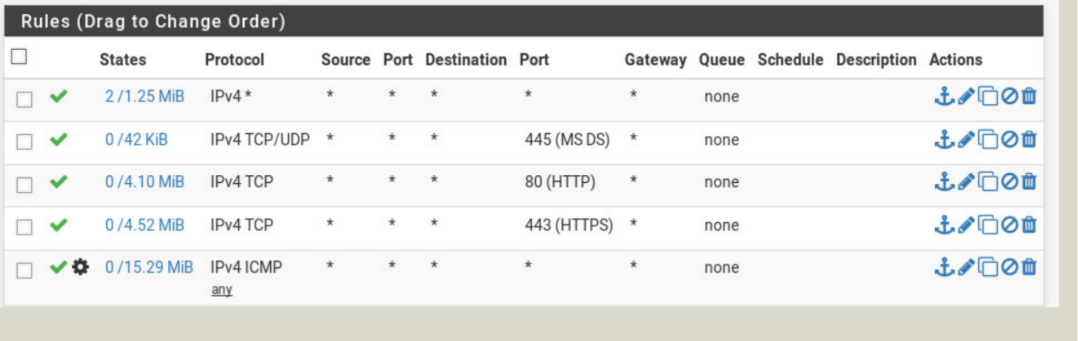
Interfaces

WAN	1000baseT <full-duplex>	10.0.0.1
LAN	1000baseT <full-duplex>	192.168.10.1

Figure 6.2: PfSense dashboard

DHCP server and it divides the network into 25 parts. When a new device is connected to the LAN network it gets its own subnetwork directly from the DHCP server, for example 192.168.10.2. If desired, the size of the network can be easily increased by giving more addresses to the network, but 25 subnetwork addresses is an adequate number due to the number of devices in the study.

The settings of the network are made from the terminal after installing PfSense. However, if more exact firewall settings and rules are needed, it is easier to do that via web interface, as long as the presettings have been made from the terminal. In order to be able to modify the settings from the web interface, a virtual machine with an existing operating system must be connected to the network. With the installed server of the operating system signing directly into the server's web interface is possible. It is good to have a proper JavaScript support in the operating system as it makes scrolling the web interface more user-friendly. When a connection has been made with the web interface, new rules and other firewall settings can be set to PfSense from there. Figure 6.2 presents the front page of the PfSense web interface. From this particular page can be seen how the router has divided the WAN and LAN sides to separate networks. Figure 6.3 shows the firewall settings in the WAN side. When the firewall is switched on, the WAN side of the PfSense ruletable is empty with default settings. Therefore, all traffic from outside the network is automatically blocked. By writing certain rules, the attacking machine is given the possibility to connect to the network.



Rules (Drag to Change Order)

□		States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
□	✓	2 /1.25 MiB	IPv4 *	*	*	*	*	*	none			⚓✎⧉⊘🗑
□	✓	0 /42 KiB	IPv4 TCP/UDP	*	*	*	445 (MS DS)	*	none			⚓✎⧉⊘🗑
□	✓	0 /4.10 MiB	IPv4 TCP	*	*	*	80 (HTTP)	*	none			⚓✎⧉⊘🗑
□	✓	0 /4.52 MiB	IPv4 TCP	*	*	*	443 (HTTPS)	*	none			⚓✎⧉⊘🗑
□	✓ ⚙	0 /15.29 MiB	IPv4 ICMP any	*	*	*	*	*	none			⚓✎⧉⊘🗑

Figure 6.3: Firewall WAN tablerules

Normally, the rules of the WAN side should be written as strictly as possible so that only the needed traffic can get through the firewall to the inner network. In this study, however, the rules have been written very loosely, so that a penetration exercise to the network can be done. In the rules, ports 80 (HTTP) are opened, enabling also penetrating into PfSense web interface. Port 443 (HTTPS) opens the connection for a better protected version in the web interface, port 445 (MS DS) is opened, so that the vulnerability of Windows XP computer can be utilised to get a remote control access to the target machine. Furthermore, port 22 (SSH) also needs to be opened from the firewall, so that a connection can be made with the SmartLock programme in the network. Kali Linux system was chosen for the attacking machine due its tools and properties that fit for penetration testing. These tools and their properties were presented in section 4.3.1 Kali Linux is also the most used Linux distribution package by cyber security criminals, so in that sense it is also a good fit for the attacker's tool Windows XP (experience) is a Windows distribution published by Microsoft in 2001. The first versions of XP were the Home edition and Professional, followed by SP updates (Service Pack): SP1 versions in 2002, SP2 in 2004 and SP3 in 2008. The version used in this thesis is Windows XP SP3 because it is the newest member of the XP family and still widely used in computers, both in factories and in home computers. Microsoft ended the XP distribution support in 2015, after which Microsoft has recommended that everyone should use newer versions of Windows.

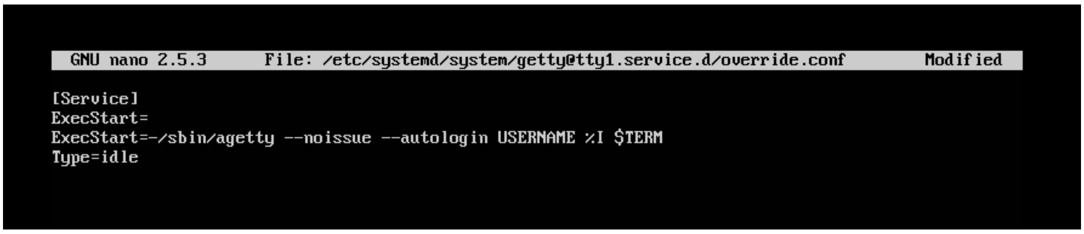
6.2 IoT devices

Several vulnerabilities to which Windows has published a patch package have been found from the Windows XP SP3 distribution. These patch packages, however, must be downloaded to the computer by the user him- or herself, and often the user has not done this. That is why it is still possible to use already found vulnerabilities to attacking purposes. In this thesis, one vulnerability detected in Windows XP SP3 has been used. This vulnerability enables making a remote control connection without any action required from the user. In this thesis, penetrating to the Windows XP target machine is even easier. After installation its firewall has been switched off and port 445 (MS DS) has been opened, originally meant for the port of the remote-controlled printer. Port 445 has also been detected to be a passage of well-known ransomwares, such as WannaCry and Petya, to the target machine, so this is a suitable vulnerability for an exercise [59].

The aim is to do a so-called general model so that needed devices can later be easily added to the simulation. First, a virtual machine is created, and to this machine is given a needed amount of memory, processor capacity (CPU) and hard drive space for use. The resources of the device are determined by the needs of the IoT device. The device is connected directly to the simulated network with a network card and it seeks itself a DHCP (Dynamic Host Configuration Protocol) address distributed by PfSense. The necessary ports are opened from the device so that devices operating with different interfaces can be created. The virtual machine requires some kind of operating system. The operating system is installed to the computer with an ISO file using a virtual station. In the devices used in this study, Ubuntu 16.04.2 server ISO icon has been used to install the operating system into the virtual machine. In this study, two separate virtual machines have been installed and both drive Ubuntu Server Linux as the operating system.

First, a working network interface is given to the virtual machine so that updates to the operating system can be done and the system can be otherwise altered as required. Logging in resulting from rebooting the computer has been deleted by creating a new service in the file

/etc/systemd/system/getty@tty1.service.d/override.conf, with changes shown in figure 6.4. When the necessary changes have been done in the virtual machine, the network card can be changed back to a closed environment from the settings of the virtual machine.



```
  GNU nano 2.5.3      File: /etc/systemd/system/getty@tty1.service.d/override.conf        Modified

[Service]
ExecStart=
ExecStart=-/sbin/agetty --noissue --autologin USERNAME %I $TERM
Type=idle
```

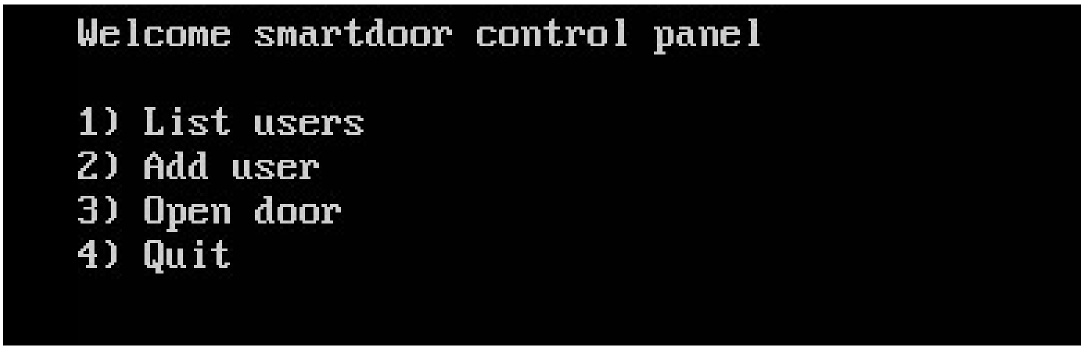
Figure 6.4: Override.conf file modifying

6.2.1 Smart lock

The first IoT device in this study is a smart lock control panel written in C++. The software has been installed directly to the root of Linux. The Linux system has been configured so that when it is switched on, it starts the software directly with a BASH script. In addition, port 22 has been opened for SSH connection and protection measures have been removed to make the task easier.

Normally, there is not a graphic user interface (GUI) in the most common models of smart locks, therefore, controlling of this software is also done from the terminal. The programme gives the user a menu when it opens, as seen in figure 6.5. The purpose of this menu is to demonstrate buttons in a real smart lock. A list of users can be read from the device's panel and a certain number of users can be added to the device. The number of users is limited in the programming code of the device. The third option of the device is to “Open door”. To open the lock in the door the user must be an authorised user on the list.

However, an error has happened in the device when programming and it is not possible to define rights for the user by clicking “add user”. Thus, opening the door requires the rights of a “Master” user at the moment. In this particular device the task is to find out whether it is possible to open the lock without having to disassemble the device. Solving this exercise requires knowhow about the C++ programming language and understanding of Buffer overflow errors in the use of IoT devices.



```
Welcome smartdoor control panel

1) List users
2) Add user
3) Open door
4) Quit
```

Figure 6.5: Smart lock system control panel

Several smart devices operate in the Internet and if there have been errors in the cyber security of the devices already in the programming stage, it is very difficult for the user to patch these errors independently. Furthermore, when the device has been installed in the network at home, it is possible that redundant ports have been left open in the device, for which reason the operating of the device can possibly be disturbed or fully penetrated. Smart locks must have some kind of user interface for it to function properly. The manufacturers should pay more attention specifically to this aspect so that the functioning of the lock cannot be manipulated by outsiders. A publication in the TechCrunch site demonstrates how easily a badly designed smart lock can be penetrated [60].

6.2.2 Surveillance camera interface

The other IoT device used in this study is a web interface of a surveillance camera installed at home. The interface of the camera has been done using the programming language React.JS and the ready application of the software has been installed to a virtual machine with Ubuntu server Linux as the operating system. The virtual machine has been configurated so that when it is switched on, it drives a BASH script, which first opens its own server to run in a certain port for the use of the application. After this, the application is switched on in the script, and then the application connects to a server which is already switched on and opens a graphic user interface to an IP address. For the functioning of the application, port 80 must be opened from the system for HTTP connection. In this study, port 80, which is the port of the HTTP protocol, is used because the benefit of the exercise does not increase even though a better protected connection was used, such as port 443, which is the port of the HTTPS protocol.

The interface of the device is opened to port 80 (HTTP) and its protection has been left to default values. The task is to understand the importance of password protection and that many device manufacturers leave the protection to default values; standard passwords and usernames are easily found from the Internet. The user has been given a BASE64 encoded file, which contains a set of usernames, and another BASE64 encoded file, which contains a set of passwords. These usernames are simple and very common, and manufacturers may use them as standard in their devices. The user must be able to master tools in BurpSuite programme, so that it is possible to penetrate through the protection of the surveillance camera to the interface of the device.

Surveilllance cameras are one of the most popular targets when wanting to penetrate into the network of a smart home. Typically, the cameras are placed outside of the house, for which reason an outsider can easily recognize the brand and model of the camera in question. When the attacker knows the brand and model, it is possible to prepare the attack carefully by searching detected vulnerabilities from the Internet related to that particular camera. Often cameras also have some kind of web interface with which to make settings to the camera or download data from the camera's memory. If the protection of a smart home network is jeopardised, the camera can be penetrated from inside the network as well, and if the passwords of the camera's web interface are too weak or if the camera's interface has other vulnerabilities, they can be used in this kind of situation. For penetrating the security of a camera, one possible option is to use the Shodan search engine, which is covered more fully in section 4.3.4.

Chapter 7

Conclusion

In this thesis, the safety of IoT devices is examined and a virtual penetration testing environment is built for testing cyber security of these devices. The environment built in this study are also going to be used in educating new cyber security students for better understanding of the basics of penetration testing and for teaching them the meaning of penetration testing in cyber security related work. The testing environment in this study is intended to resemble the basic architecture of a smart home system. The attacking machine used in the environment was Kali Linux penetration testing operation system and the router and firewall was PfSense virtual firewall. The attacking machine was installed to its outer network (WAN) and target machines to its inner network (LAN). The desktop computer of the smart home user was Windows XP SP3 version system, because its vulnerabilities are already generally known and this way the cyber security of softwares of third parties was not endangered during research. The IoT models used in the laboratory work with Linux Ubuntu operating system. A surveillance camera system, with its interface opened for the attack target, and a Smart Lock system which has a control panel in the device's IP address and includes a CTF exercise, were chosen for the IoT devices.

The environment built in this study can be utilised for education purposes if more exercises are added. This environment cannot be used as a virtual testing system for IoT devices as such, but physical IoT devices can, however, be installed in the environment and their testing is possible with the environment in question. The biggest challenge was to get the IoT devices virtualised when building the virtual penetration testing environment for testing of IoT devices. Already virtualised IoT devices were very hard to come by and coding them from the start by myself is out of the scope of this thesis. Therefore, it was decided to examine how to install real physical devices in the environment. Attaching physical devices in a virtual testing environment is the best way to test the cyber security of IoT devices. If it is desired that the environment is used as teaching material, much more exercises needs to be added to it. Doing these exercises in the scope of this thesis would be too extensive, as the exercises need to be done manually from the start. Although, already completed CTF competitions could be utilised and ways which would speed up the process of doing the environment exercises could be sought.

In the future, the laboratory in question could be extended to apply also to WEP penetration testing and certain physical IoT devices could be attached in the laboratory, in addition to the virtual device photos. Also, newer operating systems than Windows XP and other penetration devices could be attached in the laboratory. When the foundation of the laboratory is done, it is easy to add exercises in it and set them to work with new virtual machines. If needed, power can be added to the machines and 255 devices can be installed to merely PfSense LAN side without any changes.

Bibliography

[1] I. Lakshmi. The internet of things (iot) needs to become a reality in it world. *SSRG International Journal of Computer Science and Engineering*, 6(2):23–33, 2019.

[2] K. Ashton. That "Internet of Things" Thing. Available online at `http://www.itrco.jp/libraries/RFIDjournal-That%20Internet%20of%20Things%20Thing.pdf`, (2009). [Accessed 2019-11-11].

[3] D. Vinik. The Internet of Things: An oral history. Available online at `https://www.politico.com/agenda/story/2015/06/history-of-internet-of-things-000104`, (2015). [Accessed 2019-06-4].

[4] T. Harwood. Internet of Things (IoT) History. Available online at `https://www.postscapes.com/internet-of-things-history/`, (2019). [Accessed 2019-13-11].

[5] ITU Strategy and Policy Unit. The Internet of Things. Technical report, International Telecommunication Union, Geneva, 2005.

[6] S. Deering and R. Hinden. Internet Protocol, Version 6 (IPv6) Specification. RFC 2460, July 2017.

[7] K. Lueth. Why the Internet of Things is called Internet of Things: Definition, history, disambiguation. Available online at `https://iot-analytics.com/internet-of-things-definition/`, (2014). [Accessed 2019-8-5].

[8] J. Ravolainen. Kodit digitalisoituvat - oletko jo tutustunut älykodin tarjoamiin ratkaisuihin? Available online at https://blogi.mpy.fi/kuluttajat/ajankohtaista/kotien-digitalisoituminen-alykoti, (2018). [Accessed 2019-06-15].

[9] S. Musa. Smart City Roadmap. Available online at https://www.academia.edu/21181336/Smart_City_Roadmap, (2006). [Accessed 2019-05-25].

[10] H. Kangasniemi. Älykaupunki perustuu tiedon keräämiseen ja jakamiseen. Available online at https://yksityisille.hub.elisa.fi/mika-on-alykaupunki/, (2019). [Accessed 2019-05-5].

[11] State of Green. Think Denmark - White paper for green transition. Available online at https://stateofgreen.com/en/uploads/2018/05/Smart-Grid.pdf?time=1540804392, (2018). [Accessed 2019-10-5].

[12] S. Cifani. The 6 Most Innovative Waste Technology Systems. Available online at https://www.dumpsters.com/blog/smart-waste-management-technology, (2018). [Accessed 2019-06-8].

[13] G. Shubhankar. How Smart Parking Technology will Reduce Pollution in the City. Available online at http://www.parking-net.com/parking-industry-blog/get-my-parking/how-smart-parking-technology-will-reduce-pollution-in-the-city, (2018). [Accessed 2019-07-12].

[14] P. Piyush, D. Golden, S. Peasley, and M. Kelkar. Making smart cities cybersecure. Available online at https://www2.deloitte.com/insights/us/en/focus/smart-city/making-smart-cities-cyber-secure.html, (2019). [Accessed 2019-06-22].

[15] A. Sadeghi, C. Wachsmann, and M. Waidner. Security and privacy challenges in industrial internet of things. In *2015 52nd ACM/EDAC/IEEE Design Automation Conference (DAC)*, pages 1–6, 2015.

[16] IEEE Standard for Ethernet. *IEEE Std 802.3-2018 (Revision of IEEE Std 802.3-2015)*, pages 1–5600, 2018.

[17] IEEE Standard for Information technology—Telecommunications and information exchange between systems Local and metropolitan area networks—Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. *IEEE Std 802.11-2016 (Revision of IEEE Std 802.11-2012)*, pages 1–3534, 2016.

[18] B. Metcalfe. Metcalfe's law after 40 years of ethernet. *Computer*, 46(12):26–31, 2013.

[19] S. Nog and D. Kotz. A performance comparison of tcp/ip and mpi on fddi, fast ethernet, and ethernet. *Open Dartmouth: Faculty Open Access Scholarship*, 3210, 1995.

[20] R. Santitoro. Metro ethernet services - a technical overview. Technical report, Metro Ethernet Forum (MEF) Technical Committee, April 2003.

[21] K.L. Addy. Power Over Ethernet-prioritized active splitter, US Patent 7,081,827 B2, Jul. 2006.

[22] A. Holt and C. Huang. *802.11 Wireless Networks: Security and Analysis*. Springer, London, 2010.

[23] O. Santos. *End-to-End Network Security*. Cisco press, UK, 2007.

[24] M. Stamp. *Information security Principles and Practice*. John Wiley and Sons, Inc, Hoboken, 2011.

[25] C. Gehrmann, J. Persson, and B. Smeets. *Bluetooth Security*. Artech House, London, 2004.

[26] N. Gupta. *Inside Bluetooth Low Energy*. Artech House, United States, 2013.

[27] Bluetooth studio. History of bluetooth the evolution of bluetooth technology. Available online at `http://bluetoothinsight.blogspot.com/2008/01/bluetooth-power-classes.html`, (2016). [Accessed 2019-07-12].

[28] K. Ren. Bluetooth Pairing Part 1 – Pairing Feature Exchange. Available online at `https://www.bluetooth.com/blog/bluetooth-pairing-part-1-pairing-feature-exchange/`, (2016). [Accessed 2019-07-16].

[29] A. Hernandez-Solana, D-Perez-Diaz de Cerio, A. Valdovinos, and J. Valenzueala. Proposal and Evaluation of BLE Discovery Process Based on New Features of Bluetooth 5.0. *Sensors*, 17(9):1988, 2017.

[30] M. Aftab. *Building Bluetooth Low Energy Systems*. Packt Publishing, Limited, UK, 2017.

[31] DJ. Bluetooth power classes. Available online at `http://bluetoothinsight.blogspot.com/2008/01/bluetooth-power-classes.html`, (2008). [Accessed 2019-07-12].

[32] H. Sun, C. Wang, and B. Ahmad. *From Internet of Things to Smart Cities : Enabling Technologies*. Chapman and Hall/CRC, New York, 2017.

[33] E. Shin and G. Jo. Structure of nb-iot nodeb system. In *2017 International Conference on Information and Communication Technology Convergence (ICTC)*, pages 1269–1271, 2017.

[34] Digita. Mikä on LoRaWAN. Available online at `https://www.digita.fi/yrityksille/iot/mika_on_lorawan`, (2019). [Accessed 2019-05-20].

[35] F. Adelantado, X. Vilajosana, P. Tuset-Peiro, B. Martinez, J. Melia-Segui, and T. Watteyne. Understanding the Limits of LoRaWAN. *IEEE Communications Magazine*, 55(9):34–40, 2017.

[36] B. Ray. SigFox Vs. LoRa: A Comparison Between Technologies and Business Models. Available online at `https://www.link-labs.com/blog/sigfox-vs-lora`, (2018). [Accessed 2019-06-5].

[37] Traficom. Suomi aloittaa älylaitteiden turvallisuuden varmistamisen ensimmäisenä euroopassa – uusi tietoturvamerkki auttaa kuluttajia tekemään turvallisempia kodin älylaitehankintoja. Available online at `https://www.kyberturvallisuuskeskus.fi/fi/ajankohtaista/suomi-aloittaa-alylaitteiden-turvallisuuden-varmistamisen-ensimmaisena-euroopassa`, (2019). [Accessed 2019-2-12].

[38] D. Gibson. *CompTIA Security+ Get Certified Get Ahead SYO-501 Study Guide*. Sybex Inc, United States, 2017.

[39] J. Liu, Y. Xiao, K. Ghaboosi, H. Deng, and J.Zhang. Botnet: Classification, attacks, detection, tracing, and preventive measures. *EURASIP Journal on Wireless Communications and Networking*, 115(1):1687–1499, 2009.

[40] Cloudflare. What is Mirai? Available online at `https://www.cloudflare.com/learning/ddos/glossary/mirai-botnet/`, (2019). [Accessed 2019-07-12].

[41] T. Brewster. Exclusive: Hack Breaks Your Visa Card's Contactless Limit For Big Frauds. Available online at `https://www.forbes.com/sites/thomasbrewster/2019/07/29/exclusive-hackers-can-break-your-credit-cards-30-contactless-limit/#2d1ed3c941e1`, (2019). [Accessed 2019-08-5].

[42] Malwarebytes. Backdoor. Available online at `https://www.malwarebytes.com/backdoor/`, (2019). [Accessed 2019-09-7].

[43] M. Holloway. Stuxnet Worm Attack on Iranian Nuclear Facilities. Available online at `http://large.stanford.edu/courses/2015/ph241/holloway1/`, (2015). [Accessed 2019-09-20].

[44] J.F. Scalisi. Smartlock systems and methods, US Patent 8,947,530 B1, Feb. 2015.

[45] C. Paras. Can Smart Locks Be Hacked and How to Prevent It. Available online at `https://www.diysmarthomesolutions.com/smart-locks-hacked-how-to-prevent-it/`, (2019). [Accessed 2019-10-23].

[46] B.Mehl. Step-by-Step Tutorial: How to Copy or Clone Access Cards and Key Fobs. Available online at `https://www.getkisi.com/blog/how-to-copy-access-cards-and-keyfobs`, (2018). [Accessed 2019-10-23].

[47] T. Spring. Smart Lock Turns Out to be Not So Smart, or Secure. Available online at `https://threatpost.com/smart-lock-turns-out-to-be-not-so-smart-or-secure/146091/`, (2019). [Accessed 2019-10-23].

[48] Symantec employee. What is the Difference Between Black, White and Grey Hat Hackers? Available online at https://us.norton.com/internetsecurity-emerging-threats-what-is-the-difference-between-black-white-and-grey-hat-hackers.html, (2019). [Accessed 2019-10-15].

[49] H. Wolf and W. Bo. *Kali Linux 2018: Windows Penetration Testing: Conduct network testing, surveillance, and pen testing on MS Windows using Kali Linux 2018, 2nd Edition*. Packt Publishing Limited, Birmingham, 2018.

[50] A. Orin. Behind the App: The Story of Kali Linux. Available online at https://lifehacker.com/behind-the-app-the-story-of-kali-linux-1666168491, (2014). [Accessed 2019-10-15].

[51] PortSwigger. The Burp Suite family. Available online at https://portswigger.net/burp, (2019). [Accessed 2019-07-16].

[52] awasthi7xenextt. Introduction to Wireshark. Available online at https://www.geeksforgeeks.org/introduction-to-wireshark/, (2019). [Accessed 2019-09-17].

[53] A. Dobhal. The World's most dangerous search engine:Shodan. Available online at https://dev.to/ankitdobhal/the-world-s-most-dangerous-search-engine-shodan-1ja5, (2015). [Accessed 2019-08-20].

[54] S. Cheng, C. Wang, and G. Horng. Osgi-based smart home architecture for heterogeneous network. *Expert Systems with Applications*, 39(16):12418 – 12429, 2012.

[55] G. Fleishman. Wireless mesh networks: Everything you need to know. Available online at https://www.pcworld.com/article/3212444/mesh-network-explained.html, (2017). [Accessed 2019-08-21].

[56] Matthew B. Hoy. Alexa, Siri, Cortana, and More: An Introduction to Voice Assistants. *Medical Reference Services Quarterly*, 37(1):81–88, 2018.

[57] M. Szreder. IoT Security in Practice – A computer security analysis of the IKEA “TRÅDFRI” platformn. Master’s thesis, Linköping University, Linköping, Sweden, 2019.

[58] Hamed and Basil. Design and implementation of smart house control using labview. *International Journal of Soft Computing and Engineering*, 1:98–106, 2012.

[59] H. Rasmussen. Vältä digikaapparit – sulje avoimet portit Windows 10:ssä. Available online at https://kotimikro.fi/tietoturva/pc-suojaus/valta-digikaapparit-sulje-avoimet-portit-windows-10-ssa, (2018). [Accessed 2020-03-1].

[60] Z. Whittaker. Security flaws in a popular smart home hub let hackers unlock front doors. Available online at https://techcrunch.com/2019/07/02/smart-home-hub-flaws-unlock-doors/?guccounter=1&guce_referrer_us=aHR0cHM6Ly93d3cuZ29vZ2xlLmNvbS8&guce_referrer_cs=O64wkdVOYX9SH7DRdZMjrw, (2019). [Accessed 2020-02-1].