Evaluation of the impact of encryption on video transmissions (in e-health applications)

Björn Lindström

Björn Lindström
VT 2016
Examensarbete, 15 hp
Supervisor: Juan Carlos Nieves
Examiner: Jan Pedher Johansson
Kandidatprogrammet i datavetenskap, 180 hp

Abstract

E-health has been growing steadily with an increasing benefit for medical care and patients. One part of e-health is telemedicine where video transmissions could be used for medical treatment. To be able to reach out to the public it has to use the available infrastructure which involves the Internet. Doing so introduces the need for encryption in order to protect the confidentiality of the users. It was investigated what the performance cost would be for a broad set of hardware related to a Persona that does not have the most recent technology available. For each hardware, the corresponding utilization of the standardized symmetric algorithm Advanced Encryption Standard has been visualized in relation to the key size and three different video resolutions. The results show that utilization of a single core ranges from irrelevant, 0,84%, to low, 24%, when encrypting a high definition 1080p 2-way video transmission. The scope covers a smartphone, a laptop and a desktop computer with results for each of them. These results could be used when implementing an application based on the Advanced Encryption Standard, which is an approved standard in the USA for an e-health application. The results could also be used in other contexts when encryption is needed and are not specifically bound to e-health.

Evaluering av hur kryptering påverkar videoöverföringar(i en e-hälso applikation)

Sammanfattning

E-hälsa växer stadigt med ökande fördelar för både sjukvård och patienter. Telemedicin är en del av e-hälsa där videoöverföringar kan användas vid behandling och för att kunna nå ut till allmänheten finns det ett behov av att använda sig av befintlig infrastruktur vilket innebär nyttjandet av Internet. Detta betyder att det finns ett behov att skydda användarna genom kryptering. En analys gjordes för att bestämma prestandakostnaden av denna kryptering för ett brett urval av hårdvara gällande för ett Persona som inte har den nyaste teknologin, och prestandan för denna hårdvara har visualiserats relaterat till nyckellängder och olika videoupplösningar. Resultatet visar att prestandakostnaden sträcker sig från obetydlig till låg vid en högupplöst videöverföring i 1080p som är dubbelriktad. Dessa resultat kan användas vid implementering av en applikation som är baserad på Advanced Encryption Standard, vilket är en i USA godkänd standard för att användas i en e-hälso kontext. Resultatet är inte begränsad till e-hälsa utan kan även användas i andra sammanhang när kryptering är nödvändig.

Acknowledgements

I want to thank my supervisor Juan Carlos Nieves for valuable support, interesting discussions and never with a short supply of criticism of my work. Thank you. I would also like to thank Joan Daemen for taking his time and pointing me to the eBACS project. A big hug to my family for the never ending support.

Contents

1 Introduction

The following chapter will explain the problem and the benefits with telemedicine, explain the methods that were used and the main purpose of this thesis. It will also look back in history to give some guidance to the later chapters.

1.1 Problem statement

Moving the medical field into the public space of the Internet is a challenging task. Not only is there a need to protect sensitive information, e.g. the integrity of the patients, but also to balance performance and quality so that it is sensible to use for both the patient and the medical side of the activity performed. If handling security in a proper way impacts the quality of the activity performed it could have negative effects such as reduced effectiveness or in the worst case, not useful at all and contradicting the set goal of that activity. For example, if that means a low resolution, poor synchronization of video and audio or latency related problems would hamper medical assessments or consultations. This puts things on its edge when data-intensive tasks, like video transmissions, are being handled [1][2].

There are many possible benefits of expanding health care beyond common physical restraints. Doing so may benefit medical providers, patients and society in general as there is always a merit to be able to do more with less. From the medical perspective, it could have a cost saving and time reducing effect which would imply that more patients could be treated which benefits the society at large. It could improve efficiency by streamlining demands so that patients shorten the time for treatment they need which benefits both medical providers and patients. It can improve treatment in the form of availability and by that not only reduce cost related expenditures, otherwise associated with a physical visit, but also save time for the patients. Health care becomes more accessible. Patients might also benefit from quicker diagnoses, better treatment or advice on many medical related issues [1][3][4][5].

Protecting users exposed to malign adversaries in a public environment is a daunting task. Even a well formed and robust system are susceptible to exploits as the point of having such a system is to grant certain people access and deny others. Designing that system is even harder as there are many ways one could do it wrong [6, c1]. Users have to, especially so when it comes to medical environments, rest assure that they can trust the system. Therefore, it has to rely on security measures that protect this trust, that is built on proven and validated procedures. On the other side, there will always be the need for better performance, to be able to do more with higher fidelity. This would imply that there would be some form of trade off as it could be hard to do both with a good outcome. In a medical scenario, there should not be a trade-off. Trust should always be the priority when it comes to personal integrity otherwise there is a possibility that the system itself will undermine its purpose [1][6, p14-16]. There are already commercial products available that claim to

use secure systems. SKYPE[1], for example, states that it is using the National Institute of Standards and Technology (NIST) standard AES-256 cipher algorithm for encryption of their video transmissions. In that sense, encrypted video transmissions are already available for usage. However, there is always a risk with systems where the design is not clear or properly validated [7]. There are also many different hardware related issues. Especially when the user base might have everything from smartphones, tablets, laptops and stationary computers. Does that affect the service provided during a secure transmission and to what extent? Isolating different parts in a solution is an often used method of finding particular limitations or constraints that affect the overarching solution to a problem. This thesis is focusing on the performance of a standardized symmetric algorithm approved for e-health and telemedicine [7].

1.2 Expected outcome

In this thesis, I expect to find support for choosing a symmetric cipher algorithm that follows a recognized standard based on performance and security it provides. I also aim to show the predicted performance for that symmetric cryptographic algorithm for an average user in a telemedicine application by reducing the overall problem, i.e. sending secure video transmissions in a 2-way nature and focusing on the performance of the encryption it should then be possible to visualize the cost of that operation and subsequently the resources available for other tasks.

1.3 Research question

What performance, clocks per byte, are there for video transmission adhering to NIST PUB 800-21 symmetric encryption specifications for use in e-health applications [7]?

This would clarify the computational cost for encrypting data transmitted and in its implication visualize what available resources there are during encryption that could be used on other tasks.

1.4 Method

The original hypothesis, that led to the research question, was that encryption will have a cost in terms of available computational power. That cost should set a limit on other tasks that is needed for a transmission to function as intended. More specifically that it would be a limit in a profound way, an exhaustive task, with measurable implications that would affect the video transmission and ultimately the quality of service provided in video transmission in general and telemedicine in particular.

To narrow the scope of the thesis a persona of a typical user was created. Pruitt et al.[8] describes the usefulness of a persona in that it would give a more clear goal on what to achieve and helps focusing on a specific user group, rather than all possible users. The

[1] SKYPE is an application that provides video calls on desktop computers and mobile platforms

persona that is created will set the range of the specific hardware that will be covered in the results. It will guide the scope of the thesis in that the persona would not have the best and newest available technology, but rather equipment that dates a few years back and was not state of the art even then. The reason for this is to better represent a larger population with an conservative approach on the hardware itself.

There is a need to find a cipher algorithm that protects the patients integrity and that follows a current and advised standard in such an environment. Therefore a literature study is necessary to collect facts that would support my hypothesis as well as enhance the understanding of the field of crytography in general and particularly its relation to e-health and telemedicine. That also includes consultation of experts in the field if needed or adressing a mathematical solution to the problem if the necessary facts is hard to find.

1.4.1 Persona of a typical user

John is 49 years old and lives in an apartment in a medium sized city. He works at the local post office sorting mail normally between 06.00 - 15.00 hours Monday to Friday. He lives alone but regularly attends the local bridge club, an interest he also pursues over the Internet with friends abroad. That was the reason he bought his HP Envy 6z 1100 for a good price in 2012. He settled for a moderate laptop, carefully comparing the prices, equipped with an AMD 4655m processor. He also has a stationary PC[2] from 2007 with an Intel Core 2 Quad Q6600 processor, which he does not use that often since he bought the laptop. The laptop was the most recently technical device he bought as his smartphone, an HTC Vivid with a Snapdragon S3 processor, is really showing its age. John has no need for a new phone as he mostly makes phone calls and has no particular interest in technology or computers but uses his devices almost daily. He also has a duplex 10Mbps Internet connection with a wireless router installed in his apartment.

1.5 Related work

Studies regarding telemedicine and suggestions of how to implement video transmissions for health care involving encryption with Advanced Encryption Standard (AES) is rare. Kiah et al.[9] made a framework for real-time telemedicine using a secure group based communication architecture. They combined asymmetric and symmetric cryptography and verified its performance for different 15 frames 352x388 video encodings. Their result showed that the AES-256 encryption had insignificant increase in CPU load when comparing unencrypted traffic with encrypted traffic.

Thampi et al.[10] showed in a review of triple Data Encryption Standard (3DES) and AES a comparison of the running times for different data sizes. Singh et al.[11] made a study in which they compared Blowfish, AES, and 3DES throughput for different data sizes. Their main conclusion where the relative high amount of throughput of Blowfish and the resource demanding AES. Silva et al.[12] performed a study on the impact of different key sizes for AES and 3DES and how it affected response time in relation to different data files. They showed the relative small effect of an increasing key size compared to the data file inputed. Rihan et al.[13] presented in their study a comparison of AES and DES in regards to

[2]Personal Computer (PC)

throughput, processing time and CPU usage for different data sizes. Patil et al.[14] showed in a comparison study of AES, 3DES, Blowfish and RSA the individual characteristics of each algorithm.

One thing that was very obvious going through the articles were the variation in results. Mainly due to different setups, implementations, and hardware related issues. Even though Kiah et al.[9] had shown that the net cost of encryption was low, it was not obviously translatable to other hardware platforms.

1.6 History of Cryptography

Cryptography is an old technique of making the readable unreadable for anyone not entitled to the content. The ways to accomplish that has changed during the course of human history. From the early manual transition and substitution ciphers to the more elaborate ”Bazeries cylinder” in the late 1900 century, to the electro-mechanical Enigma machine during the Second World War. The common interest has always been to safeguard the information from unintended consequences, an evolution and battle between the ciphers and cryptanalysts. During the years there have been several ciphers claimed to be unbreakable only to succumb and be replaced by a new one [15].

It is of interest that Alan Turing, together with the many extraordinary cryptanalysts at Bletchley Park, successfully deciphered the Enigma cryptosystem with the help of an algorithm Turing had designed called the Bombe. Turing made many contributions to the field of cryptography and formalized the principle for the universal Turing machine before the war in 1936. His theories would later essentially become the computers of today [16][17].

1.6.1 Kerckhoffs principle

In the late 1900 century, Aguste Kerckhoffs formalized a set of requirements that a good field crypto cipher should use. Khan D. in his book The Codebreakers: The story of secret writing [15] states: ”These requirements still comprise the ideal which military ciphers aim at. They have been rephrased, and qualities that lie implicit have been made explicit. But any modern cryptographer would be very happy if any cipher fulfilled all six” [15]. Remember, Kahn originally wrote his book in 1967 and should be read in that context. However the first requirement ”the system should be, if not theoretically unbreakable, unbreakable in practice” are the main principles in which today’s cipher rely upon. The second requirement is the one Kerckhoffs is most known for: ”Compromise of the system should not inconvenience the correspondents” [15, p123].

This principle has been widely known as Kerckhoffs’s principle, but in a rephrased form to fit the modern day cryptology. Assume that the cipher is known, in any case, an adversary will eventually learn every detail of it. If it falls into unintended hands it should not cause any harm to the other users of the same system. This principle has been transformed to the separation of key and cipher. If the cipher is known, but reasonably safe, then the key is the only component to keep safe from an adversary. As an extension, it has also to separate the notion of secrecy as a means for security, known as security from obscurity. The argument is that a cipher that is open and transparent in the long run will prevail as it

is constantly probed for weakness. There have been many ciphers that have tried to hide their inner workings, but inevitably been exposed. Secrecy is just another hurdle and can be dangerously contradicting of what it originally intended to achieve [6, p24-25][18, p6-7][15, p124].

The four remaining requirements, ”The key should be memorable without notes and should be easily changeable; the cryptograms should be transmissible by telegraph; the apparatus or documents should be portable and operable by a single person; the system should be easy, neither requiring knowledge of a long list of rules nor involving mental strain” [15, p123], may be valid from the point of a military field cipher but they hardly apply in today's computer cryptography.

1.6.2 The Data Encryption Standard

DES has to be mentioned in its own right. Horst Feistel had already led a project at IBM in creating an encryption scheme in the late 60's when The National Bureau of Standards(NBS, later NIST) issued an open competition for an encryption scheme to be used in the government. IBM participated with an adapted version of Feistel's block cipher that had a key size of 56-bits. In 1977 the DES was approved by NBS and thus available to the public. The key length had some voiced controversy as it was assumed that the length would invite a brute force attack. It also had a criticism of the inner workings of the algorithm as not all parts were disclosed openly. Nevertheless, DES lasted for many years and was finally ended in 1998 when Electronic Frontier Foundation had broken a DES cipher with a dedicated machine. Computational power made it unsafe and not the inner workings of it [18, p168][19, p78].

1.7 Outline

The following is an outline of this thesis

2. Background
Describes the criteria in which the ciphers have been chosen and the basic parts involved to achieve confidentiality.

2.1 Telemedicine
Outlines the World Health Organisation definition of what telemedicine is, presents a basic end to end model of a video transmission and security services most likely involved.

2.2 Telemedicine requirements
Outlines necessary criteria for telemedicine drawn from experience of implementing successful enterprise-level clinical telemedicine program.

2.3 National Institute of Standards and Technology
Explains the basic process in which standards are approved by the NIST.

2.4 Health Insurance Portability and Accountability Act
Outlines the NIST recommendations for HIPAA and the underlying approved symmetric cipher algorithms approved for this context.

2.5 Encryption, decryption and cryptographic key

Explains the basic concept of transforming a plaintext into an unreadable ciphertext and the reverse process. It will also explain the different cryptographic keys and their usage and how to safely generate them.

2.6 Block cipher
Explains the basic structure of a block cipher. It will describe the two NIST approved ciphers for an e-health context.

2.7 Block cipher modes
Explains different block cipher modes and how they extend the block ciphers and essentially become part of the encryption/decryption of messages longer than the block cipher.

2.8 ECRYPT Benchmarking of Cryptographic Systems, eBACS
Explains the ECRYPT project and the VAMPIRE virtual laboratory. It will also explain the cryptographic benchmarking suite SUPERCOP.

3. Results
Presents collected data in regards to AES in counter mode and calculate its performance relative to different video resolutions outlined in 2.2.

3.1 TDEA
Explains why TDEA/3DES cipher was removed from the performance comparison.

3.2 AES performance values
Outlines collected performance values for AES in counter mode from the eBACS project and states the calculated bits per second values derived from the performance values.

3.3 Core utilization during encryption/decryption
Charts that visualizes the different hardware derived from 3.2 in relation to saturating different video resolutions outlined in 2.2.

4. Conclusion
Summarize the main conclusions related to the research question in 1.3.

4.1 Discussion
A discussion of the findings from the conclusions in relation to the original hypothesis.

4.2 Future Work
Outlines interesting subjects that might benefit from further studies.

2 Background

The following chapter will present the underlying criteria in which the algorithms have been chosen and on what merit it is founded. It will also explain the basic parts and their usage in a cryptographic environment. The principle of encryption relies on several subparts that combined can be strong, but as any system has inherited weaknesses. Most importantly, what has been stressed in the previous chapter, is the safekeeping and proper management of the cryptographic key.

2.1 Telemedicine

The use of telemedicine might be thought to be an information technology invention, but there are cases where consultation has been appropriated visa vi other means of communication like letter correspondence. However, the earliest record is from the late 19th century where a dutch physiologist used an electrocardiograph, a galvanometer and a telephone wire to record electric cardiac signals 1.5km away [20]. This matter today has taken great leaps forward but the definition has many meanings.

The World Health Organization has the following definition: ”The delivery of healthcare services, where distance is a critical factor, by all health care professionals using information and communication technologies for the exchange of valid information for diagnosis, treatment and prevention of disease and injuries, research and evaluation, and for the continuing education of health care providers, all in the interests of advancing the health of individuals and their communities.” [5].

2.1.1 Basic end to end video transmission model

The basic model will give a better understanding of what steps are typically involved when transmitting in a broad sense and specifically what subpart that is the focus of this thesis.

Figure 1 is the start point of the transmission. The user captures live video with a camera. The captured video is then encoded, compressed, with a video encoding scheme and sent through the cryptographic module. The data is encrypted with a symmetric block cipher and sent through a network. This network could be interconnected with the Internet.

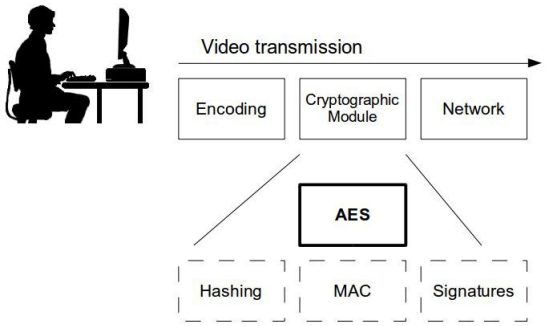

Figure 1: Start point of video transmission

Figure 2 is the end point of the transmission where the routed transmission will arrive. The network payload is decrypted with an equal symmetric block cipher and decoded before ending up on the users screen.

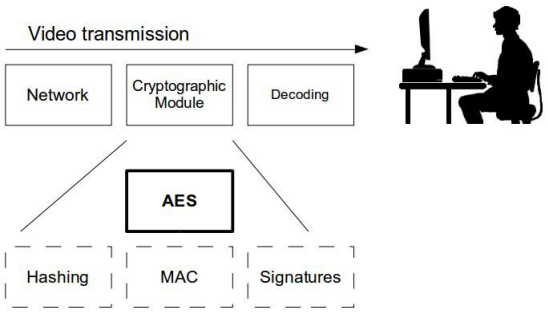

Figure 2: End point of video transmission

The model describes a 1-way transmission. For a 2-way transmission, it also has to be performed in the opposite direction, but in the order outlined above.

2.1.2 Security services

There is an intentional unclarity in the basic model outlined in 2.1.1 where other security measures have been left out. That was done to highlight the focus on confidentiality achieved by encrypting with a symmetric block cipher. Confidentiality means that the information itself is protected from anyone ”listening” somewhere between the endpoints. They are denied and can not convert the encrypted message into something intelligible and will not make any sense to the observer. Even though that is the main concern, there might be other services needed [21][22].

The first thing that needs to be addressed is that since a symmetric cipher relies on a single key for encryption and decryption of the message, it is important that this key is handled in a secure way. Even more importantly, we have to give access to the one intended by identifying the participating parties, so that a connection is established and verified to be correct. That is called authentication and only after this, the key can be exchanged in a secure manner, usually by encrypting the key itself with an asymmetric key. Remember, if the observer would get the key that also means that they can decrypt the message [21][22].

The integrity of the data itself can be protected with hashing, making a small message, that

correlates and only correlates to one specific information. So if the hash does not equal the information then someone could have tampered with it or exchanged it [21][22].

There are many ways to handle these and other issues, but for the sake of this thesis, the focus is the impact of encryption, confidentiality, and not the other parts most likely involved in an application such as these.

2.2 Telemedicine requirements

Meyer et.al[3] outlines a set of criteria that they consider is essential for telemedicine. They draw their conclusions from experience with the University of California, San Diego Medical Center's successful enterprise-level clinical telemedicine program. They state that the video transmission should be: "two-way, full screen, full resolution, full 29.97 frames per second. It should have a compression algorithm that enables transmission of large amounts of data through limited-capacity networks, with minimal delay; quality of service algorithms to ensure data quality. And; It should have an encryption of at least 128-bit."[3]. The interpretation of these criteria is outlined in the section below.

2.2.1 Video compression and bit rate

There are numerous video encodings that can be used when streaming video, however the focus of this thesis is limited to a video encoding commonly used today which has a good compression rate and quality loss rate, the h.264/AVC standard from 2003. Compared with mpeg2 it has a 50% better compression rate and is perfectly suitable for streaming video [23]. A recent study made successful e-health video consultations over a 4G mobile network using h.264 encoding [24]. According to Sullivan et al.[25] the size of the maximum bitstreams for various resolutions are outlined in Table 1 below that should satisfy the set criteria. However, these high rates might present a network problem, especially in a 3G mobile network. The lowest bit rate of h.264 goes as low as 64Kbps. A bit rate is the amount of bits per second that needs to be saturated for the video to be seamlessly playable.

Table 1 H.264 bitstreams for different resolutions

Resolution	Frames	Bit rate
480p or 576i	30	4Mbps
1280x720p	30	14Mbps
1920x1080p	60p	50Mbps

2.3 National Institute of Standards and Technology

The National Institute of Standards and Technology (NIST) is a body set under the USA Department of Commerce. The NIST is a well-known standardization institution and many of its standards are implemented in a global context. Their expertise is well known and accepted as many of their standards are used in a broad set of technologies and industries. The procedures to reach this stature relies on five principles. "The standards need to be

formed on a general Consensus, that satisfies that all views are heard. It has to rely on Transparency of the process. This includes for any standard to publicly advance a notice, identify the scope, provide information on participation and satisfy all interest before final approval. It has to Balance influence including government involvement. Satisfy that that Due process allows that anyone with a direct and material interest has a right to express their position. And finally, it needs to have Openness in that all parties with a materially affected interest may participate."[26].

2.4 Health Insurance Portability and Accountability Act, HIPAA

In order to answer the essential part of encryption, there has to be a framework of trusted and approved ciphers for use in e-health. This may, of course, vary between countries. The USA approved HIPAA in 1996 regulates the procedures for handling patient information in a secure and safe manner. NIST has published the SP-800-66 revision 1: An Introductory Resource Guide for Implementing the Health Insurance Portability and Accountability Act (HIPAA) Security Rule [27], with recommendations on which symmetric algorithms that are approved for this context.

2.4.1 NIST SP 800-21

NIST Special Publication: Guideline for Implementing Cryptography In the Federal Government, SP 800-21 specifies the approved cipher algorithms allowed in a government application. The approved symmetric ciphers are TDEA(3DES) and AES with the key size of 128, 192 and 256 bits. Furthermore, it outlines the need for a proper validation of the implementation both of the Cipher algorithm and the module in which it resides. NIST FIPS 140-2 is the validation process in which all cryptographic applications has to pass before approval [7].

2.5 Encryption, decryption and cryptographic key

The process of transforming a plaintext of any type of data that is readable by a human or a computer machine, into a ciphertext which is unreadable, is called encryption. A special key is added in the process and this key has to be hidden, or encrypted itself, in such a way that an adversary cannot read it, as it is uniquely bound to the ciphertext that was transformed from the original plaintext. Consequently, it should also work inversely, namely by decrypting the ciphertext with the same key. A consequence of finding the key effectively unlocks the ciphertext and reveals its content. So, not only is it of importance to safeguard the key, but also using a long enough key, in order to minimize the probability of finding it just by simply guessing [6, p23-24].

In Figure 3 "Plaintext" is a text readable to a human or a computer machine. "Key" is a cryptographic key. "Encryption" is the process in which the plaintext together with a key is generated into a ciphertext. Arrows are representing a transition. "Decryption" is the reverse process of "Encryption" and used to generate the original plaintext from a ciphertext.

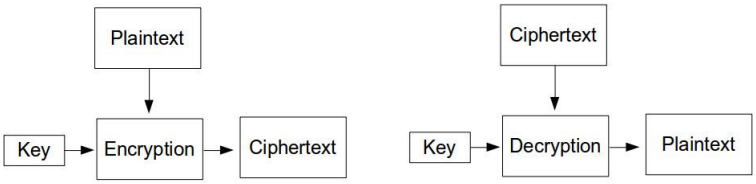

Figure 3: Encryption and decryption

2.5.1 Symmetric key

The symmetric key is also known as the private key, the use of a single key both for encryption and decryption. Commonly used in a single set environment e.g. encrypting files stored locally. If two separate parties are involved there has to be an exchange of the key either physically or by some other secure manner. For example with an asymmetric key handling [19, p33-34].

2.5.2 Asymmetric key

The asymmetric key, commonly known as the public key. It involves two keys, a public and a private key. With these keys, one can receive encrypted messages and also decrypt them without the need to exchange keys beforehand. This is done with the aid of a public key known to all and a private key, both unique to each user. If person A wants to send an encrypted message to person B it is done as the following. Person A encrypts a message with the public key of B and sends it to B. Person B can then decrypt the message with their private key which is the only key that works for decrypting the message [19, p239].

2.5.3 Random Number Generator, RNG

A significant part of maintaining a safe and unpredictable nature is that it is vital that the key is constructed from a source guaranteed to generate uniquely random numbers, i.e. generated in a way that cannot be emulated or mimicked by an adversary. This could involve generating the random numbers with the aid of mouse movement, the access time of hard disks, atomic decay [18, p55] or anything that can not be copied or emulated by an adversary.

2.5.4 Security related to key length

A key that is constructed from 128 bits has the property of 2^{128} possible combinations which translates to $3.402 * 10^{38}$ different keys. An adversary, therefore, has many combinations to try before eventually finding the right one. A real world example relevant to this number is the seconds since the big bang, estimated to be $4.343 * 10^{17}$ seconds [28]. Let us recall Auguste Kerckhoffs' old but still valid argument that the cipher should be, if not theoretically, but at least practically unsolvable. Even though this number presents a significant challenge, we can calculate how long it would take to try all those values with a computer. Assuming that one key can be generated each clock cycle then a 1 Ghz single core CPU can try 10^9 combinations every second which would take $1,078 * 10^{22}$ years to go through all possible key combinations. A 4 core 3 Ghz CPU can try $4 * 3 * 10^9$ which would take

$8,983 * 10^{20}$ years. Even if one was lucky and found it in half that time it would still amount to an incredibly long time [18, p49]. Even with the use of supercomputers or a distributed net, i.e. the use of many computers that could try many more combinations each second, the computational power would eventually make it unsafe, but computational power is not near that point yet. A cryptographic scheme is secure if it is computationally secure, even though it is not theoretically secure if the probability of an adversary to find the correct key is negligible. A 128-bits key is safe from this standpoint [6, p36].

2.6 Block cipher

A block cipher is a block of fixed length, AES uses 128-bits. Its essence is a construction, for any input combined with a key, that will create a pseudo-random permutation indistinguishable from a random permutation [18, p152-153]. Ideally, the block cipher should be a random permutation but that is more of a philosophical dream. More intuitively is that for each key and the block cipher length, there exists a mapping such that it only appears once, otherwise it would not be possible to reverse the ciphertext. That is true for all possible ciphertexts generated with the block cipher. Given an input and a key it will generate a permutation which will never be the same if the key is changed [6, p44-46]. Ferguson et al.[6] argue that one should never trust a block cipher that is not public, that it has been analyzed and reviewed by the cryptographic community in depth before one considers using it [6, p50].

2.6.1 Advanced Encryption Standard, AES

AES was originally open for all competition set about by NIST for a new, secure and fast encryption algorithm to replace the ageing DES cipher. During a period of 5 years of rigid and public testing, the Rivandjel contribution was chosen as the new AES[29, p1-3]. The AES is a block cipher that uses a Substitution-permutation network together with a symmetric key to accomplish a pseudo-random permutation. The key length can be 128, 192 or 256 bits and the rounds changes accordingly to the key as 10, 12 or 14 rounds. Each round performs four steps: the Sub Bytes step, the Shift Row step, the mix columns step and the add Round key step. On the last round, it ends with four additional steps before it is completed [30]. Bernstein et al.[31] points out that the relation of the key lengths and performance is not exactly linear, but a fair approximation would be a 40% increase comparing the 128-bits key with the 256-bits key due to the increased rounds.

2.6.2 Triple Data Encryption Algorithm, TDEA

There are many names for this cipher. It is commonly known as 3DES but NIST uses the name 3TDEA, to point out the three unique keys. 3TDEA is an improvement on the Data Encryption standard (DES) from 1977 in that it extends the key size by using three passes of DES with different 56-bit keys. First encrypts with key 1, then decrypts with key 2 and finally encrypts with key 3, meaning key1, key2 and key3 are not equal. The three key combination will give it a key length of 168-bits. For decryption it is the inverse order, decrypts with key 1, encrypts with key 2 and decrypts with key 3. This handles the problem of the brute force attack on the DES 56-bit key [32]. ”However, there is an attack on 3TDEA

that reduces the strength to the work that would be involved in exhausting a 112-bit key." [21]. This attack involves collecting a large number of plaintext/ciphertext pairs. The more pairs collected, the less work has to be done to break the cipher. NIST mentions 2^{40} pairs [21], which is a daunting task in itself.

2.7 Block cipher modes

For all plaintexts with lengths that are larger than the fixed block cipher length, an additional set of actions has to be applied and essentially becomes part of the encryption. Any plaintext length has to be padded so that the length equals a multiple of the given block cipher size [6, p64]. There are many different ways of combining block ciphers to accommodate the increased length. Some also incorporate authentication. The ones outlined below only handles confidentiality.

2.7.1 Electronic Code Book mode, ECB

Each plaintext message is encrypted with the block cipher as individual blocks. If identical blocks are repeated in the plaintext it is also repeated in the ciphertext. ECB mode is not an advised mode of operation as it leaks information due to repeating patterns [33][6, p65].

2.7.2 Cipher Block Chaining mode, CBC

The initial plaintext block is XORed with a random initial vector(IV) before encryption. The resulting ciphertext block is then XORed with the next plaintext block and then encrypted. This procedure continues until all plaintext blocks are XORed with the previous cipher block. By this operation, a randomness is introduced that is not present in the ECB mode [33][6, p65-66].

2.7.3 Output Feedback mode, OFB

At the start, it is initialized from a random IV which is used to generate a key stream that is independent of the plaintext itself, by using the block cipher to generate a pseudo-random stream of bytes. So the plaintext does not pass the block cipher, instead, the ciphertext is created with the key stream when it is XORed with the plaintext. This is called a stream cipher and one benefit is that it requires no padding [33][6, p68-69].

2.7.4 Counter mode, CTR

The Counter mode uses a Nonce, number only used once, that needs to be uniquely generated and never used with the same key more than once. The Nonce is concatenated with a counter value, usually a number 1-k and has to fit the block cipher length. The Nonce and counter is encrypted with the block cipher with a key and used to generate a keystream. Each keystream block is then XORed with the corresponding plaintext block. For each cipher block the counter increments. One benefit is the random access of the cipher blocks

making it possible to access any cipher block without any previous cipher block. Both encryption and decryption can be parallelized and the keystream can be generated as a pre-process. CTR mode does not require padding as the remaining space in the last block will be discarded [33][6, p70-71]. Lipmaa et al.[34] points out that Counter mode has compared to CBC mode a significant increase in speed, which could be up to four times that of the CBC mode.

A general representation of the steps, outlined above, is shown in Figure 4 for encryption and in Figure 5 for decryption. ”Counter” is the Nonce and counter value concatenated. ”Key” is the cryptographic key generated for the block cipher. ”Cipher block” is the chosen block cipher algorithm and arrows represents a transition. The crossed circle represents XOR.

Encryption, Figure 4, is where key and counter are encrypted into a keystream block and XORed with the corresponding plaintext block thus generating a ciphertext block. This procedure continues until the last nth plaintext block has been encrypted to the nth ciphertext block.

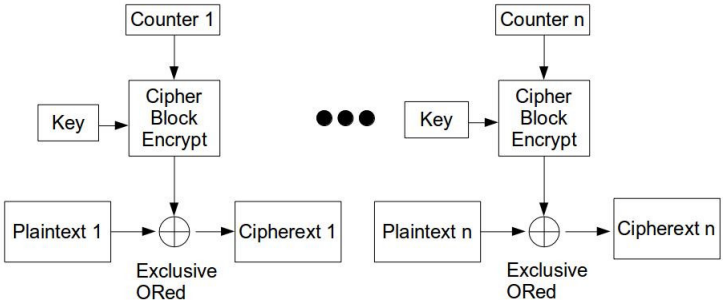

Figure 4: CTR mode encryption

Decryption shown in Figure 5 is done almost in the same order as encryption, but now the ciphertext is XORed with the keystream block and generating the plaintext block. This procedure continues until the nth ciphertext block has been decrypted to the nth plaintext block.

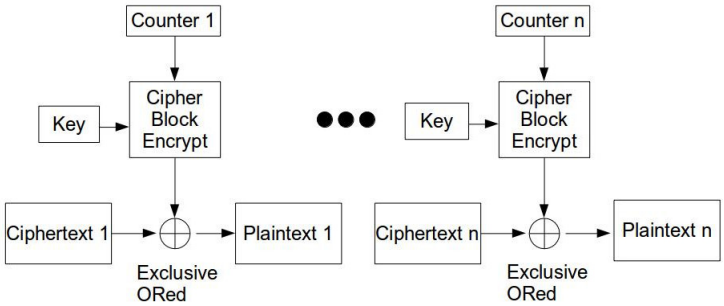

Figure 5: CTR mode decryption

2.8 ECRYPT Benchmarking of Cryptographic Systems, eBACS

eBACS[1] provides benchmarking for a wide range of hashes, stream ciphers and asymmetric systems with a benchmarking toolkit called SUPERCOP, System for Unified Performance Evaluation Related to Cryptographic Operations and Primitives. This toolkit is developed by the Virtual Applications and Implementations Research Lab (VAMPIRE) which aims to:" Research new techniques that are related to efficient and secure implementation and; Provide a bridge between research and the user community. VAMPIRE is a part of the European Network of Excellence for Cryptology II (ECRYPT II) which are funded within the Information and Communication Technologies Programme of the European Commission's Seventh Framework Programme (FP7)". More information regarding ECRYPT II and VAMPIRE can be found on their website [2].

SUPERCOP measures values from cryptographic primitives generated on real machines and stores these in a publicly accessible database. It tries to avoid any randomness that may occur during the benchmark by running each computation within the measuring program several times and then run the program several times to produce a stable distribution. Any discrepancies are flagged with red and a question mark. In theory, it could continue to run the test to remove the discrepancies when a stable outcome is verified. It performs the benchmark in an idle state, which means that there is a minimum of other processes that could affect the result. SUPERCOP is available for download for any user that wants to contribute to the project. That allows performance benchmarks for any specific hardware that is not in the database. Further information is available at the SUPERCOP website [3]. In relation to throughput, the data collected are the median values of cycles/byte on a 4096-byte message to be able to calculate the throughput in Mb/s for each hardware platform. An important note, benchmarks on computers with multiple CPUs only uses one CPU and CPUs with multiple cores only uses one core.

[1] http://bench.cr.yp.to/
[2] http://www.ecrypt.eu.org/ecrypt2/
[3] http://bench.cr.yp.to/supercop.html

3 Result

The following chapter will present the collected data and calculated median throughput. It will also show the core utilization for the different hardware that the persona John has during encryption or decryption of video transmission outlined in Table 1. It will also present the conclusions from the results and the discussion from it.

3.1 TDEA

There are a number of reasons to discard TDEA/3DES as a good option. Silva et al.[12] showed in their comparison of AES and 3DES that the throughput is considerably lower than of AES. In fact, it is less than a third. This is indeed expected as 3DES is an extension of the ageing DES and has to make three DES passes to complete its cycle, effectively slowing it down. Similar results, if not worse, were shown in another report [10]. NIST [21] states that there is a security issue that effectively reduces the exhaustive work of finding a DES 168-bit key reduced to finding an 112-bit key which does not meet the minimum security criteria of a 128-bit key. NIST also states that TDEA(3DES) will not be used after the year 2030, effectively meaning that 2026 is the last year of usage since according to NIST recommendations it only has a 4 year lifespan [21]. By these accounts, 3DES has been removed and will not be used in the performance analysis.

3.2 AES performance values

The data values were retrieved from eBACS [35] on the 13 of May 2016. The throughput is calculated as follows:

$$\frac{\text{processor clock frequency}}{\text{clocks per byte}} = \text{bytes per second} \tag{3.1}$$

Tables 2, 3 and 4 shows the values which are the basis for this evaluation. They contains three different values to support a stable distribution from a statistical point. The used data length for each AES and hardware platform is a 4096 bytes message. Longer messages might reduce the clocks per bytes even further. The median values are used when calculating the maximum throughput for each core. It is calculated by the stated processor speed and the throughput value is rounded off. The 4096 bytes message and the rounding off where chosen as a conservative approach so that the results could be interpreted as minimum values and better performance could be the case.

Table 2 Data values for AES in counter mode for 2012 AMD 4655m

Distribution	**AES-128**	**AES-192**	**AES-256**
Upper quartile (Uq)	1,79 cpb	2,03 cpb	1,33 cpb
Median	1,79 cpb	2,02 cbp	1,33 cpb
Lower quartile (Lq)	1,78 cpb	2 cpb	1,33 cpb
Diff median to (Uq/Lq)	0% / 0.56%	0.49% / 0.99%	0% / 0%
Median throughput	8936 Mbps	7920 Mbps	12032 Mbps

Clocks per byte (cpb) Mega bits per second (Mbps)

Table 3 Data values for AES in counter mode for 2011 Snapdragon S3

Distribution	**AES-128**	**AES-192**	**AES-256**
Upper quartile (Uq)	19,49 cbp	56,22 cpb	34,12 cpb
Median	19,49 cpb	56,14 cpb	34,05 cpb
Lower quartile (Lq)	19,49 cpb	56,11 cpb	34,04 cpb
Diff median to (Uq/Lq)	0% / 0%	0.14% / 0.05%	0.21% / 0.03%
Median throughput	728 Mbps	256 Mbps	416 Mbps

Table 4 Data values for AES in counter mode for 2007 Intel Q6600

Distribution	**AES-128**	**AES-192**	**AES-256**
Upper quartile (Uq)	12,65 cpb	15,32 cpb	18,02 cpb
Median	12,63 cpb	15,31 cpb	18,01 cpb
Lower quartile (Lq)	12,63 cpb	15,3 cpb	18 cpb
Diff median to (Uq/Lq)	0.16% / 0%	0.07% / 0.07%	0.06% / 0.06%
Median throughput Mb/s	1520 Mbps	1256 Mbps	1072 Mbps

3.3 Core utilization during encryption/decryption

The following charts show the utilization that the encryption/decryption process demands to saturate and maintain a stable video stream related to Table 1. Each chart, Figure 6-8, is derived from the throughput from Tables 2, 3 and 4.

The charts, Figure 6-8, shows different utilizations for the hardware stated on the top of each chart. The y-axis shows single core utilization, where 100% are the full usage of that core. The x-axis has the different bit rates corresponding to Table 1. Each AES with the key lengths 128, 192 and 256 bits are represented with the colors gray, white and black. The utilization for the respective AES is printed above each corresponding staple. The block cipher mode used are Counter mode, CTR.

The Intel core2, Figure 6, follows the expected difference regarding key lengths with a 20% increase from 128 to 192-bit keys and 20% from 192 to 256-bit keys. The utilization will be 9,32% on a single core in a 2-way full HD 1080p video transmission which has to be considered minimal.

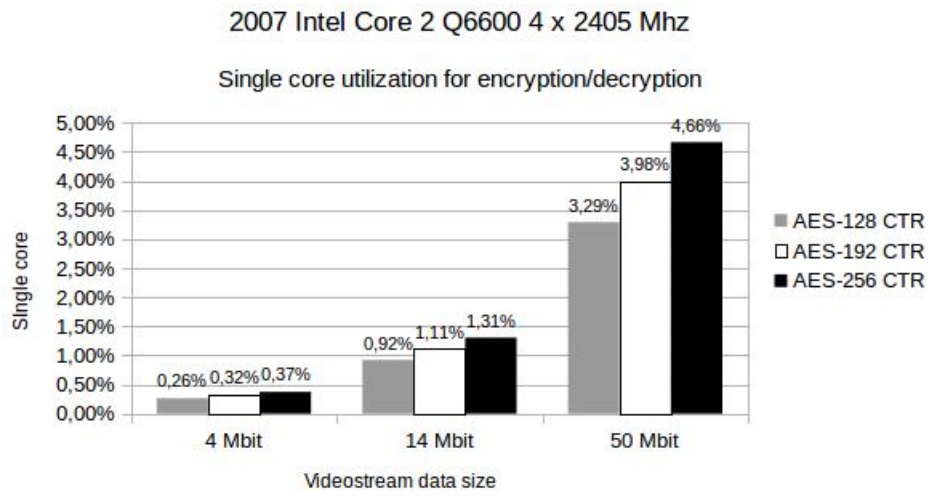

Figure 6: AES with CTR block cipher mode

The embedded Snapdragon, Figure 7, show an interesting characteristic in the higher utilization for the 192-bits key than the 256-bits key, which is odd. Theoretically, it should be placed somewhere between the values of the 128 and 256-bits keys. The difference between the 128 and 192-bits keys is expected to be 20% given the extra rounds of the 192-bits key. There are either an unknown specific hardware related cause or a discrepancy in the test suite of SUPERCOP. The oddity is though persistent through all of the reported data lengths from eBACS[1] and the value of the 256-bit key is within expected range compared to the 128-bits key. Even though it is possible, and still have margin, encrypting of a full HD 1080p video transmission in a 2-way setup will demand 24% of a single core.

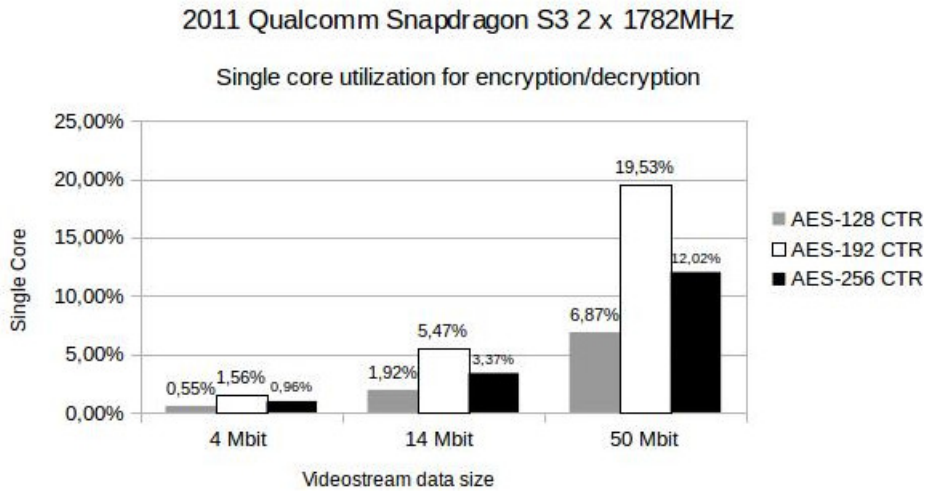

Figure 7: AES with CTR block cipher mode

The performance on the AMD A10-4655m, Figure 8, is very high. In fact, it is so high that it becomes irrelevant from the perspective of impacting video transmissions. Utilization only

[1] http://bench.cr.yp.to/

reaches 0.84% when simultaneously encrypting and decrypting a 2-way video transmission that is a full HD 1080p video transmission using a 256-bit key. The high performance could be explained by the AVX instructions present in this CPU. It most likely also explains the higher performance of the 256-bits key than the theoretically faster 128-bits key.

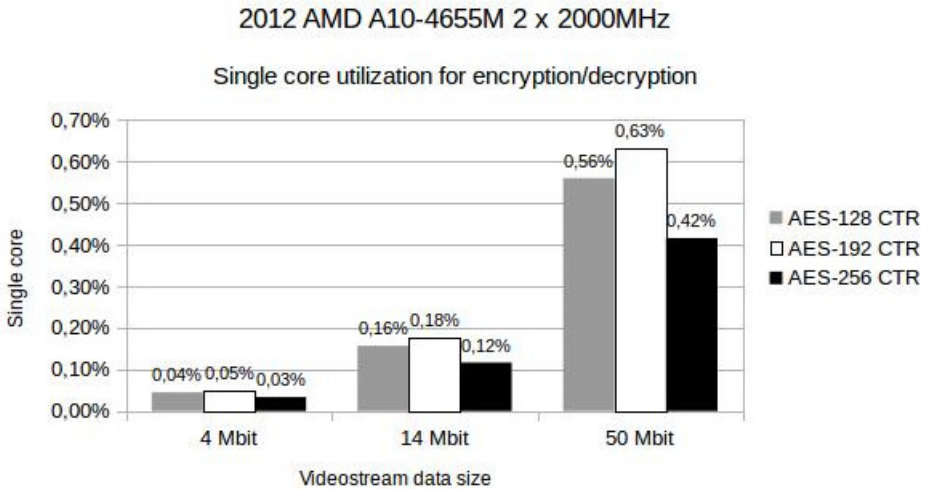

Figure 8: AES with CTR block cipher mode

4 Conclusion

The original research question was: ”What performance, clocks per byte, are there for video transmission adhering to NIST PUB 800-21 symmetric encryption specifications for use in e-health applications [7]?”

The performance for a wide set of hardware related to a Persona that does not have the most recent technology available has been presented. For each hardware the corresponding CPU/core utilization of AES in counter mode has been visualized in relation to the key size and three different video resolutions. The video coding is well known and has been used in e-health teleconsultations. To that point we can conclude that the question has been answered.

The impact on system performance of encryption with AES is minimal to irrelevant, which means that AES is suitable to use when encrypting video transmissions in an e-health application. Minimal in this sense still amounts to as high as 24%, of a single core, for the embedded Snapdragon. From the perspective of encryption that still would be considered manageable, but it does start to tax the available resources which could be noticeable. In that case the proper way would be to use a lower resolution more suitable to the ubiquitous environment. This would also reduce strain from the encryption as well as from the video encoding itself.

The focus of the thesis has been on e-health, but the results can be transferable to other areas where encryption with AES is considered or needed.

4.1 Discussion

I have concluded that even though AES encryption and decryption does not come without cost in terms of utilization, it is so low that it can work for a wide range of devices without impacting data transmissions such as video streaming. That is in the respect of not considering other utilization costs most likely involved, such as encoding of a video, which can be very taxing. Even the five-year-old embedded Snapdragon processor was only marginally affected and could easily cope with simultaneous encryption and decryption. However it also showed, that testing on specific hardware gives a much clearer picture, which was expected. Different hardware will have different constructions that can be hard to theoretically derive from its specification, which can be seen in the difference between AES-192 and AES-256 in the Snapdragon. Security issues, in general, seems to be attracting more focus amongst hardware manufacturers. Qualcomm has on its 2014 Snapdragon 805 implemented hardware support for AES in a module that is FIPS 140-2 validated.

AES has a high throughput, with a high security, which is available to the public and recommended by NIST in an e-health context. Therefore, I suggest using the AES-256, not

only because of its high security but also for its limited impact on hardware resources. This means that there are resources available for other tasks involved in the process.

The observant reader will have noticed that the Persona John has a limited network connection which will imply that in his case the resolution of the video stream has to be lowered as the network connection will present a bottleneck for higher resolutions regardless of encryption.

Having personal relations in the medical field and the general ambition to create more with less, I see many potential benefits of e-health that should enhance the quality of life. I rest assured that the full benefit has not yet been explored and I am equally convinced that the shortcomings of e-health have many pitfalls as there always are when treading new ground. The benefits and shortcomings have to be evaluated by experts in the medical field. My contribution is from a technical perspective.

The original hypothesis, which invoked my curiosity to discover, has been proven wrong. The cost for confidentiality is far from a compromise, but a cheap cost in terms of the security it presents.

4.2 Future Work

There are still security issues to address since encryption only provides confidentiality especially in the context of a typical user scenario. There are a few loose ends that I believe could benefit from further studies, such as: What impact do hashes, message authentication codes and asymmetric key handling have on performance? How are network congestions in a particular area, especially in remote locations with low bandwidth availability, dealt with? There are studies performed both on 3G and 4G mobile networks, but to be sure real tests need to conclude local limitations and to what extent it impacts video transmission. What encoding has the least net cost in bandwidth and can robustly withstand lost packages and jitter related issues in a network?

References

[1] N. Majedi, M. Naeem, and A. Anpalagan, "Telecommunication integration in e-healthcare: technologies, applications and challenges," *Transactions on Emerging Telecommunications Technologies*, vol. 27, pp. 775–789, jun 2016.

[2] C. LeRouge, M. Garfield, and A. Hevner, "Quality attributes in telemedicine video conferencing," in *Proceedings of the 35th Annual Hawaii International Conference on System Sciences*, pp. 2050–2059, IEEE Comput. Soc, 2002.

[3] B. C. Meyer, C. a. Clarke, T. M. Troke, and L. S. Friedman, "Essential Telemedicine Elements (Tele-Ments) for Connecting the Academic Health Center and Remote Community Providers to Enhance Patient Care," *Academic Medicine*, vol. 87, no. 8, pp. 1032–1040, 2012.

[4] M. T. Krishna, R. C. Knibb, and A. P. Huissoon, "Is there a role for telemedicine in adult allergy services?," *Clinical & Experimental Allergy*, vol. 46, no. 5, pp. 668–677, 2016.

[5] World Health Organization Global Observatory for eHealth, "Telemedicine: Opportunities and developments in Member States," *Observatory*, vol. 2, p. 96, 2010.

[6] N. Ferguson and B. Schneier, *Cryptography Engineering: Design Principles and Practical Applications*. Wiley Pub., Inc, 2010.

[7] National Institute of Standards and Technology, *Special Publication (NIST SP) - 800-21 2nd ed: Guideline for Implementing Cryptography in the Federal Government [Second Edition]*. Gaithersburg, MD, USA: National Institute of Standards and Technology, 2005.

[8] J. Pruitt and J. Grundin, "Personas : Practice and Theory," in *Proceedings of the 2003 conference on Designing for user experiences*, pp. 1–15, 2003.

[9] M. L. Mat Kiah, S. H. Al-Bakri, A. A. Zaidan, B. B. Zaidan, and M. Hussain, "Design and Develop a Video Conferencing Framework for Real-Time Telemedicine Applications Using Secure Group-Based Communication Architecture," *Journal of Medical Systems*, vol. 38, p. 133, oct 2014.

[10] V. V. A. Thampi and R. K. Gopal, "A review on different encryption algorithms for a wellness tracking system," *2015 Global Conference on Communication Technologies (GCCT)*, no. Gcct, pp. 817–822, 2015.

[11] S. Singh and R. Maini, "Comparison of data encryption algorithms," *International Journal of Computer Science and Communication*, vol. 2, no. 1, pp. 125–127, 2011.

[12] N. B. F. D. Silva, D. F. Pigatto, P. S. Martins, and K. R. L. J. C. Branco, "Case Studies of Performance Evaluation of Cryptographic Algorithms for an Embedded System

and a General Purpose Computer," *Journal of Network and Computer Applications*, vol. 60, pp. 1–14, 2015.

[13] S. D. Rihan and S. E. F. Osman, "A Performance Comparison of Encryption Algorithms AES and DES," *International Journal of Engineering Research & Technology (IJERT)*, vol. 4, no. 12, pp. 151–154, 2015.

[14] P. Patil and P. Narayankar, "A Comprehensive Evaluation of Cryptographic Algorithms: DES, 3DES, AES, RSA and Blowfish," *Procedia Computer Science*, vol. 78, no. December 2015, pp. 617–624, 2016.

[15] D. Kahn, "The Codebreakers: The story of secret writing," *Scribner, New York*, 1996.

[16] C. Severance, "Alan Turing and Bletchley Park," *Computer*, vol. 45, pp. 6–8, jun 2012.

[17] G. Strawn, "Alan Turing," *IT Professional*, vol. 16, pp. 5–7, jan 2014.

[18] J. Katz and Y. Lindell, "Introduction to Modern Cryptography," *Ucsd Cse*, pp. 1–498, 2007.

[19] W. Stallings, *Cryptography and network security : principles and practice*. New York: Prentice Hall, 5th ed., 2011.

[20] E. M. Strehle and N. Shabde, "One hundred years of telemedicine: does this new technology have a place in paediatrics?," *Archives of disease in childhood*, vol. 91, pp. 956–959, jul 2006.

[21] National Institute of Standards and Technology, *Special Publication (NIST SP) - 800-57 Pt1 Rev 4: Recommendation for Key Management, Part 1: General*. Gaithersburg, MD, USA: National Institute of Standards and Technology, jan 2016.

[22] M. A. Jimale, S. Of, D. For, P. Fulfillment, O. F. The, D. Of, M. Of, C. Science, F. Of, and I. Technology, *Securing Mobile Communication Using Public Key Infrastructure For Multimedia Messaging Services (MMS)*. PhD thesis, UNIVERSITY MALAYA, 2008.

[23] A. Luthra, G. Sullivan, and T. Wiegand, "Introduction to the special issue on the H.264/AVC video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, pp. 557–559, jul 2003.

[24] L. J. Caffery and A. C. Smith, "Investigating the quality of video consultations performed using fourth generation (4G) mobile telecommunications.," *Journal of telemedicine and telecare*, vol. 21, no. 6, pp. 348–354, 2015.

[25] G. J. Sullivan, "The H.264/AVC advanced video coding standard: overview and introduction to the fidelity range extensions," *Proceedings of SPIE*, vol. 5558, no. 5558, pp. 454–474, 2004.

[26] National Institute of Standards and Technology, *NISTIR 7614: The ABC's of Standards Activities*. No. August, Gaithersburg, MD, USA: National Institute of Standards and Technology, 2009.

[27] National Institute of Standards and Technology, *Special Publication (NIST SP) - 800-66 Rev 1: An Introductory Resource Guide for Implementing the Health Insurance Portability and Accountability Act (HIPAA) Security Rule*. No. October, Gaithersburg, MD, USA: National Institute of Standards and Technology, 2008.

[28] C. L. Bennett, D. Larson, J. L. Weiland, N. Jarosik, G. Hinshaw, N. Odegard, K. M. Smith, R. S. Hill, B. Gold, M. Halpern, E. Komatsu, M. R. Nolta, L. Page, D. N. Spergel, E. Wollack, J. Dunkley, A. Kogut, M. Limon, S. S. Meyer, G. S. Tucker, and E. L. Wright, "Nine-Year Wilkinson Microwave Anisotropy Probe (Wmap) Observations: Final Maps and Results," *The Astrophysical Journal Supplement Series*, vol. 208, no. 2, p. 20, 2013.

[29] J. Daemen and V. Rijmen, *The Design of Rijndael*. Information Security and Cryptography, Berlin, Heidelberg: Springer Berlin Heidelberg, 2002.

[30] National Institute of Standards and Technology, *FIPS PUB 197: Announcing the advanced encryption standard (AES)*. Gaithersburg, MD, USA: National Institute of Standards and Technology, 2001.

[31] D. J. Bernstein and P. Schwabe, "New AES Software Speed Records," *Progress in Cryptology - Indocrypt 2008*, vol. 5365, pp. 322–336, 2008.

[32] National Institute of Standards and Technology, *Special Publication (NIST SP) - 800-67 Rev 1: Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher*. Gaithersburg, MD, USA: National Institute for Standards and Technology, 2012.

[33] National Institute of Standards and Technology, *Special Publication 800-38A 2001 ED: Recommendation for Block Cipher Modes of Operation Methods and Techniques*. No. December, Gaithersburg, MD, USA: National Institute of Standards and Technology, 2001.

[34] H. Lipmaa, P. Rogaway, and D. Wagner, "Comments to NIST Concerning AES-modes of Operations: CTR-mode Encryption," *Symmetric Key Block Cipher Modes of Operation Workshop*, no. 1, pp. 2–5, 2000.

[35] D. J. Bernstein and T. Lange(editors)., "ebacs: Ecrypt benchmarking of cryptographic systems. http://bench.cr.yp.to." 13 May 2016.