WARWICK
THE UNIVERSITY OF WARWICK

A Thesis Submitted for the Degree of PhD at the University of Warwick

Permanent WRAP URL:
http://wrap.warwick.ac.uk/136651

Copyright and reuse:
This thesis is made available online and is protected by original copyright.
Please scroll down to view the document itself.
Please refer to the repository record for this item for information to help you to cite it.
Our policy information is available from the repository home page.

For more information, please contact the WRAP Team at: wrap@warwick.ac.uk

warwick.ac.uk/lib-publications

26

Data Security for Third Generation Telecommunication Systems

by

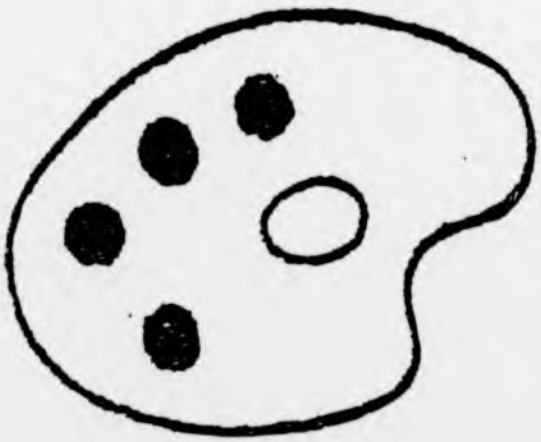
Dimitrios L. Delivasilis

A thesis submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Engineering

University of Warwick, School of Engineering

October 2003

Numerous Originals in Colour



Best Copy
Available

Shiny Paper may cause a shadow when reproduced.

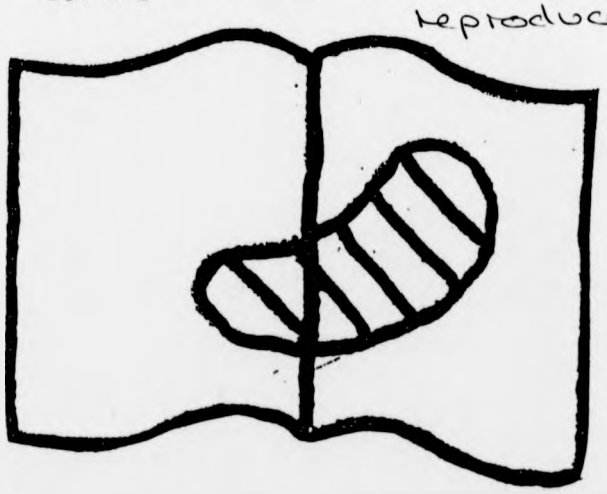


Table of Contents

List of Figures and Tables

Page
Numbering
as
Bound

Acknowledgements

I would like to thank my parents and brother for all of their support. Without them my *journey to Ithaca* would have been much longer and harder. Their continuous presence whenever was needed, enabled me to stay in focus and pursue my Ph.D.

I would also like to express my sincere gratitude to my supervisors Dr. Mark Leeson and Professor Roger Green. Mark is more than a supervisor, he is my academic mentor. He started influencing and introducing me to the *secrets* of scientific research five years ago, when I was registered for a masters degree. His experience, perceptiveness and patience with young people have guided me safely in the research area of my interest and helped me significantly in learning methods of identifying and solving new scientific problems. Roger with his vast experience in the digital signal processing provided me with the best assurance for a successful outcome of this research work. It was a real privilege of being part of his research group.

Finally, I could not forget to thank my dear friend Vasileia for tolerating me for such a long time. Without her understanding and interest my life would not have been so colorful and appealing.

Thesis Summary

This research work started as a humble, and yet ambitious attempt of enhancing the current security level in wireless data communication. Detailed analysis of the existing security protocols and wireless networks, together with their strengths and weaknesses, presented in the beginning of this thesis, indicates the need of employing human voice as an alternative security solution. Consequently the presentation of the state of the art biometric solutions and their applicability to the desired communication medium is more than necessary. However, the lack of previous involvement of human voice in an encryption method with successful results, constitute the core of this work. The majority of biometric applications aim to improve the confidence of the system to know whether a user is who he or she claims to be (authentication) and a few of them help the system in deciding his/her identity (identification). This novel method of encryption combines wireless communication (especially Third Generation mobile phones), data security, digital signal processing and smart cards. All these four different research areas have been covered theoretically in the beginning of this thesis. Thorough discussion of the fundamental principles governing each area results in identifying the strengths and weaknesses of these four core research elements and benefits the implementation stages of this work. Continuing with the thesis and proceeding to the second part, the theoretical analysis has been backed up by experimental development and testing. Every aspect of the novel research architecture has been transformed into a research algorithm. The performance of the algorithm, its cohesion and coupling has been closely investigated. The overall result of the research work accompanied with the

algorithm's strengths and weaknesses, future directions together with possible applicability of the research solution provide the epilogue of this scientific journey has been discussed.

Chapter 1: Introduction

1.1 Overview

Speech has some considerable advantages and therefore can be used as a biometric. It is not difficult to elicit, is easy to record and many types of disguise are easy to detect, even in automatic systems. Furthermore, human's linguistic behaviour is unique and manifests itself most obviously in people's speech.

Speech analysis is a popular research area, with research laboratories dedicated to speech analysis all over the world. Even though upon initial consideration it seems not to be a area of extreme difficulty, this type of analysis is of high sophistication and complexity. There are many types of sources that affect the analysis of the real voice signal. This research aims to identify the most important and finally eliminate them.

Most of the attention in the area is concentrated on voice authentication and recognition. The idea to be described in this thesis takes a human voice signal as input but looks at it from a different perspective. The signal itself is the source of data, necessary for the generation of secret keys. These keys are not going to be only random, but also extremely difficult to be produced by someone else (uniqueness).

According to the introduced idea, the user does not type the pass phrase but uses his or her own voice to speak it. As has been already mentioned, people's linguistic behaviour is unique, therefore once identified, this uniqueness can be used as an

extra security measure for the generation of random secret numbers, which can also be used as private keys for public-key cryptography.

However, the scope of the research work has not been fully defined yet. It contains another significant variable that enhances its value and broadens the horizons of its applicability. The secret keys will be generated on-the-fly, meaning that the research code has to be designed and optimised for wireless platforms (mobile phones). The necessary involvement of smart cards enables the research algorithm to reach wireless networks.

In the end of this research work, the generated results should provide an additional security measure to wireless security. In order to ensure the latter, the numerous testing phases have been designed to encompass all the real time conditions a wireless environment contains.

1.2 Research Strategy

The implementation of the research goal, identified in the previous sections, requires the thorough analysis of thee separate research areas, those of wireless communications, biometrics and cryptography. Thus, the efficient management of time and research resources was more than apparent. Equally allocation of the research effort and focus within the three areas could have been catalytic to the overall research outcome.

A prioritization in the beginning of the research work, helped in identifying the sub-targets and the difficulty of achieving them. It was clear from the beginning that emphasis should be paid to the digital signal processing phase. This can be justified by mentioning the inability of deterministic machines, such as personal computers (PCs), to simulate the functionality and behaviour of the human auditory system to an acceptable level.

Additionally, another significant research challenge was the cohesion between the three separate research areas. Part of the innovation of this thesis idea is the combination of the wireless communication, biometrics and cryptography into a single research solution. Therefore, a considerable amount of time has been invested into the efficient interoperability between those separate research elements.

Having identified the potential difficulties of the research work, a roadmap of actions was then created. It contained in detail all the steps to be followed from the beginning to the end of this thesis. Time estimations were assigned to each step, enabling the effective monitoring and evaluation of the research progress. Although this roadmap appears to have many similarities with ordinary project plans, it was decided not to be referred to with that name due to the high flexibility, in time, its research targets were designed to have.

The full specifications of the overall research challenge have to be entirely defined and justified. Their logical verification is necessary for signaling the beginning of

the designing phase. Unless all the involved parameters in the research are carefully examined for their consistency and correctness, the desired research result can definitely not be fulfilled.

In the designing phase, the backbone of the research work was created. The architecture of the software algorithm was built to ensure the desired final functionality. It had to be readable and understandable so that it could be later used as a guide for code generation and testing. This research phase had to provide a complete picture of the software, addressing the data, functional, and behavioural domains from an implementation perspective. The limitations of the wireless environment have seriously effected the entire research work. These constraints dictated the employment of fundamental techniques within the various phases of work rather than allowing more sophisticated approaches. The latter is clearly depicted in the research software design.

Continuing in the research life cycle, the development and implementation of the algorithm take the place of the design. Careful market analysis combined with previous experience helped in choosing the most appropriate software tool for this thesis work. According to information retrieved and subsequently confirmed by experience within the project, MATLAB was chosen due to its integrated technical computing environment. MATLAB combines numeric computation, advanced graphics and visualization, and a high-level programming language. It is in wide use and some of the application areas, where it dominates, including signal and image

processing, control system design, financial engineering, and medical research. The open architecture makes it easy to use MATLAB and companion products to explore data and create custom tools that provide early insights and competitive advantages.

At the beginning of the research, MATLAB version 6 was used. It was later replaced by MATLAB version 6.5. Both version have the ability of automatically translating the written MATLAB code into the more commercial C++ code. The latter simplifies significantly a future algorithm embedding into a smart card.

Finally, the testing of the developed software followed, in order to prove that the code is not injected with human fallibilities throughout the course of its production. Aiming to "demolish" the research software that was built several testing conditions were created. The objective of these conditions was to create extreme scenarios that would seriously effect the algorithm's normal operation. Due to the mostly empirical nature of this research attempt, extended testing was more than necessary, to ensure the corrective ness of the final conclusions.

1.3 Motivation

Wireless communication is an area that has experienced rapid evolution in the recent years. The need to provide its users with voice communication has been changed nowadays, containing data communication as well. The short time of this transition resulted in not entirely effective communication protocols, fully presented in the following chapters. Consequently, the current security level of wireless networks is

not adequate to protect the intact transfer of valuable information between two wireless users or a wireless user and a wired terminal. The research effort presented in this thesis, aims in providing an extra cryptographic tool that will improve the existing security level.

Additionally, the research work aims in providing a step towards an end-to-end security solution. This means that its scope is not restricted to the narrow boundaries of its specification but it is designed to serve as part of a greater security system enabling a secure end-to-end communication. The latter can be feasible with the employment of a robust key distribution scheme, allowing the wireless users to exchange securely the generated keys.

Biometrics is another research area that follows the rapid evolution of wireless communication. It is closely related with computer security because it is employed as a measure that improves the confidence of the system for a user's identity. Although the advances of biometrics solutions have been significant, their involvement in cryptographic algorithms is avoided. The investigation whether biometrics and especially voice can be employed in algorithms for secret key generation is a primary objective of this research work.

Moreover, recent cryptographic schemes are based on the random number generators. However, deterministic machines are incapable of generating real random numbers. The numbers they produce behave like random but in reality under

specific conditions they can be predicted. This is the main reason they are called pseudorandom. This weakness provides a useful trapdoor for cryptanalysts to attack the algorithm. The proposed research solution presented in this thesis describes an alternative to the pseudorandom based encryption schemes, enabling the system to defend itself better to all those cryptanalytic attacks benefiting from the lack of real randomness of the numbers.

As it has been already mentioned above, the wireless community has experienced a rapid evolution in recent years. These advances did not only affect the network technologies, but they were beneficial for the handheld devices too. The latest generation mobile phones contain the same processing power with an old personal computer. The real capabilities of those devices will be investigated, identified and finally utilised for the purposes of the research project.

1.4 Contributions to knowledge

The research presented in this thesis contributes to the body of knowledge in the following fronts as described below:

- The empirical proof of a key generation algorithm, independent of pseudo-random number generators, that is designed for handheld devices connected to wireless networks. The same algorithm with minor modifications is applicable to stationary communication terminals too.

- The analysis of human voice based on fundamental principles of digital signal processing so that its demands on processing resources is kept at a level that allows this process to be embedded on a smart card and operate on board.

- The employment of voice biometrics in the cryptography arena and more especially in symmetric key generation. It has to be emphasised that the human voice should not be transmitted over insecure networks to a powerful mainframe for digital signal analysis. All the necessary processing takes place on the handheld device, improving significantly the tolerance of the security scheme.

- The empirical proof that the elements of human voice characterising its uniqueness lie within the frequency range 1 kHz to 2 kHz. The latter constitutes the core of the security scheme presented in this thesis and has successfully resulted into a British Patent (P32647GB).

- The generation of symmetric keys for encryption purposes that are empirically proved to be related both with the uniqueness of human voice and the context of the chosen pass phrase. Thus, a single user may have as many different keys as the number of the chosen pass phrases is. The later enables the creation of a security solution that can alter the secret

keys improving the resistance of the system and minimising the impact a key compromise may have.

1.5 Publications

- D. L. Delivasilis, M. S. Leeson, R. J. Green, "**Generation of Secret Keys for Data Security Using the Human Voice**," Eight International IMA Conference on Cryptography and Coding, paper 22, Cirencester, 19-21 December 2001.
- T. S. Stergiou, D. L. Delivasilis, R. J. Green, M. S. Leeson, "**Future core network system (FCNS) – A secure signalling protocol stack for the UMTS core network**," Third International Conference on 3G Mobile Communication Technologies (3G 2002), (Session 5B), London, 8-10 May 2002.
- D. L. Delivasilis, M. S. Leeson, R. J. Green, "**A two factor user identification on Third Generation (3G) wireless terminals**," Data Security 2004, International Conference on Data Privacy and Security in a Global Society,11-13 May 2004, Skiathos, Greece.
- D. L. Delivasilis, T. Stergiou, M. S. Leeson, R. J. Green, "**3G Network Domain Security Revealed – The UMTS Initial Commercial Release**," Data Security 2004, International Conference on Data Privacy and Security in a Global Society,11-13 May 2004, Skiathos, Greece.

1.6 Patents

- **Secret Key Generation within the smart card,** British patent (P32647GB), Inventor: Dimitrios L. Delivasilis, on behalf of Wire-e plc.

1.7 Outline of the thesis

The research in this thesis is divided into eight chapters the first of which is the introduction.

- **Chapter 2** provides a detailed presentation of the fundamental principles of data security. It helps the reader to realise the security problems of digital data transmission and appreciate the need for advanced security measures.

- **Chapter 3** is introductory to digital signal processing area. It consists a useful guide for the better understanding of voice analysis part of the research algorithm.

- **Chapter 4** introduces the fundamental and state of the art principles governing the area of cryptography. It justifies the need for better key generation algorithms by presenting the existing relevant schemes. At the same time it presents the smart card area to the reader. Smart cards constitute a separate chapter due to their effect on the overall research result. Although a real smart card has never been used in this research work, their way of operation together with the limitation they impose characterize heavily all the research phases of this work.

- **Chapter 5** illustrates in detail all the research activities decided to be necessary for the efficient real life analysis of the recorded voice signal to its unique elements. Additionally, it addresses all the difficulties faced in various research phases of the voice signal analysis and explains the actions taken to overpass them

- **Chapter 6** is strongly dependant on Chapter 5 and presents in detail the sequence of processing steps that should be followed to enable the generation of the biometric signature from the analysed speech signal.

- **Chapter 7** describes thoroughly the cryptographic part of the research algorithm. It demonstrates the method according to which a biometric signature can become a 160-bit symmetric key, emphasising both on the advantages and disadvantages this may have.

- **Chapter 8** summarises the results and contributions of the thesis and discusses open questions and possible directions for future research.

Chapter 2: Data Security

2.1 Historical Review

Looking back over human history, is apparent that the form of wealth has not been constant same with the passing years. By a closer observation of the past three main forms can be extracted. In the beginning, a member of the society could be characterised as "rich" when he/she had vast expanses of land. Through the passage of time some things changed, money became the important factor which defined the "powerful" citizens. Nowadays the term "rich" can be used to describe the one who has the information.

Rifkin in his book "The Age of Access" [1] discusses exactly how this shift from ownership to access is transforming capitalism and he describes the current epoch with the words "knowledge economy". He writes that 'the repositioning of primary commerce in cyberspace and the transition to a network-based global economy are made possible by the proliferation of global electronic networks, the most important being the Internet.' [1].

The importance of information is closely related with the right timing. At one instant the information can be invaluable and the very next moment, exactly same information, can be worth nothing. This means that data should not only be distributed to many geographical places but also that this process requires the minimum time. For this reason many different types of networks have been built around the world in order to allow the fast exchange of information/data. The importance of the transferred data varies a lot; it can be as simple as an electronic

mail (e-mail) and as complicated as bank, commercial transactions or even military plans.

Prior to the major changes, which took place during the last several decades, the security of information felt to be valuable to an organisation, and was provided primarily by physical and administrative means. The existence and use of rugged filing cabinets with a combination lock for storing sensitive data provides a good example of the former. An example of the latter is personnel screening procedures used during the normal process of hiring employees.

The introduction of the microcomputer made the need for automated tools for protecting files and other information stored on the computer, a matter of prime importance. Computers stopped serving only as scientific machines and became part of the daily existence of human beings, holding vital information of both professional and personal nature. The generic name for the collection of tools designed and developed to protect data and prevent unauthorised computer users to have access is *computer security.*

Another significant change that affected security was the introduction of distributed systems accompanied with the use of networks and communications facilities for carrying data between terminal user and computer and between computer and computer. The data is exposed during the transferral process and therefore extra security measures should be taken towards in that area, providing a secure medium

for digital communication. The collection of these security measures is called *network security* and is needed to protect data during their transmission.

In a more general attempt to define security, the "security trinity" approach (Figure 2.1) can be followed, according to which its three legs, prevention, detection, and response, comprise the basis for network security. The security trinity should be the foundation for all security policies and measures that an organisation develops and deploys [2].

Figure 2.1: The Security Trinity

It is important to state at this point there is no such thing as an absolutely secure network or system. This is a real fact supported overwhelmingly by the majority of the security community [2, 3, 4]. The ultimate objective for a security architect is to provide security solutions and products with the highest possible security level.

While more and more valuable data was placed on the net, problems started to arise. People were trying to access and even use/alter data on sites where they did not have the necessary permission. These people are popularly called 'hackers' or 'adversaries' and the most sophisticated and intelligent can also be called 'cryptanalysts'. Independent of their name, they can be divided into three categories: amateurs, crackers and career criminals. The last category is the more dangerous one and they pose the greatest danger for the secure transfer of digital data. The objective of career criminals is to gather information from competitors (passive wiretapping) and sometimes try to modify or even destroy part of this information (active wiretapping) [5].

A really famous way of "destroying" data is to create a virus, a program so written that it will attach itself to a vital part of an 'operating system' and cause mischief or damage [6]. Viruses began to command worldwide attention in the end of 1987. At that time three universities suffered from virus attack on their computer systems. The 'Brain virus' struck at the University of Delaware in October of that year and in a month time the Lehigh University faced the 'Command.com' virus. In December, the Friday the 13th virus attacked the Hebrew University in Jerusalem. Viruses and many other types of security attacks will be discussed in more detail in section 2.4.

2.2 Need for Security

From the above it becomes clear that except of the need for speed, which is translated as high bandwidth, a lot of attention should be paid to security. "The

telegraph, telephone, radio, and especially the computer have put everyone on the globe within earshot-at the price of our privacy." [7]. More and more people and companies are using the biggest network in the world, the Internet, and the last thing they want to occur is for an outsider to interfere with their communications. It may be argued that the rapid evolution of electronic-commerce (e-commerce) and mobile-commerce (m-commerce) has surprised all but a few visionaries.

The number of computer crimes has increased significantly during recent years and there are fears that it will continue to do so in an even higher rate. Things become worse when taking into account that the official existing numbers for this type of crimes are unreliable since many companies try to avoid publicity when a hacker manages to get into their systems. They do not go to justice because their reputation is worth more than the damage they have suffered. Computer security law is a new field and the legal establishment has yet to reach broad agreement on many key issues. This helps the computer criminals to get away easily with crime and with a lot of profit. It is really distinctive of the fact that, according to Federal Bureau of Investigation records, only one in 20,000 computer criminals ever goes to jail [8].

In order to protect themselves, governments and big companies have understood that data security should not be considered as a luxury but as a vital factor for their organisation's efficiency and existence. Not surprisingly given the assertion, an independent survey indicated that there is expected to be an increase in the security market from today's £465million to £5.3 billion in the year 2004 [9].

2.3 Security Requirements

Computer security may be described in terms of its main six characteristics: integrity, confidentiality, availability, authentication, non-repudiation and access control. Integrity ensures that only authorised parties are able to modify computer system assets and transmitted information. A modified message should not be substituted for a legitimate one. In this context, 'modification' includes writing, changing, changing status, creating, deleting and delaying or replaying of transmitted messages. Confidentiality, in other words privacy, gives access to assets only on authorised parties and the access is classified as read only (simply revealing the existence of an object). It should be noted that in this second case, modification access should not be given by any means. The third characteristic, availability, is closely related to confidentiality. The system must provide efficient response and adequate capacity in order to support acceptable performance. Authentication ensures that the origin of a message or electronic document is correctly identified, with an assurance that the identity is not false. In other words it addresses the question of how computers can confidently associate an identity with a person. In most cases an easy solution is provided by the employment of secret words or phrases called passwords or passphrases respectively.

The relationship between integrity, confidentiality and availability is illustrated in Figure 2.2. In addition to these three main security characteristics, there are other security requirements that are worth mentioning. Authentication is one of them and it refers to the assurance of identity of person or originator of data, necessary to be provided to any twenty first century communication between two remote parties.

Another popular terminology being mentioned often lately is that of the non-repudiation, according to which the originator of communications cannot deny the transmission of data legitimately sent to other parties later.

There mainly are three different types of security for computer systems: hardware, software (program) and data. This work focuses only on data security, which happens to be the most vulnerable part of the system. Figure 2.3 shows the vulnerabilities as they apply to the three broad categories of system resources [10].

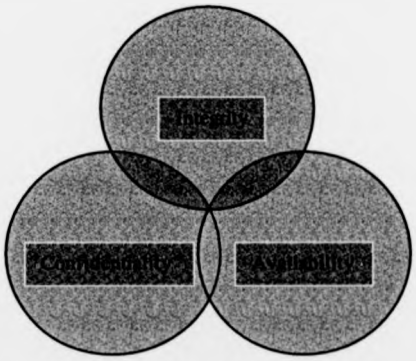


Figure 2.2: The relationship between Integrity, Confidentiality and Availability

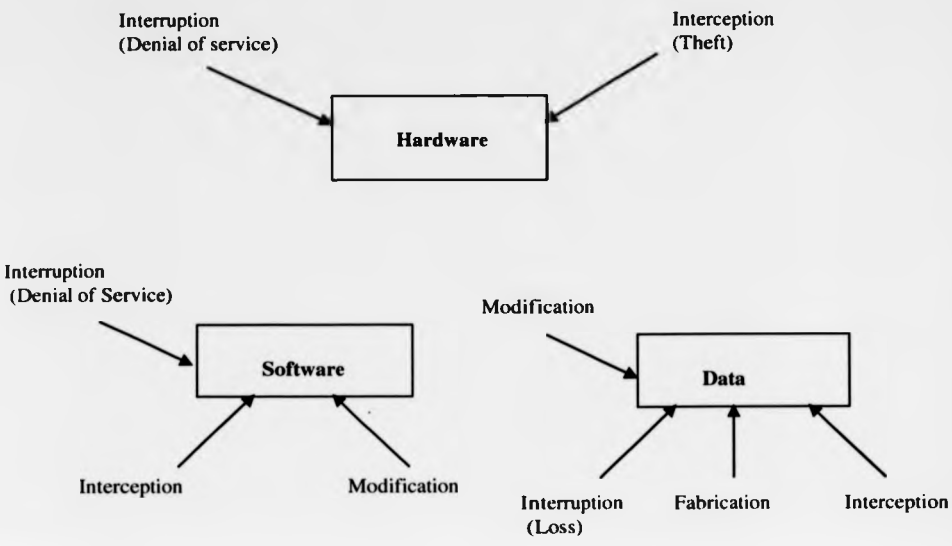


Figure 2.3: Computing System Vulnerabilities

The computer community has employed the technique of encryption for data protection. Although a detailed description of both encryption and decryption techniques can be found in Chapter 4, it is useful at this point to introduce the principle. Encryption transforms the original message or plaintext into a new form, the ciphertext. There are many variations on this theme as will be seen in Chapter 4, but the essence is often to use a long binary number, called the encryption key. This number will "lock" the message and transform it to a new form, which will be meaningless for human beings. The receiver of the message should be aware of the key or else he/she will not be able to understand what has been received [11].

2.4 Security Threats and Attacks

In general, both in computer and network security there is a flow of information from a source, such as a file or an entire region of memory, to a destination, such as another file, application or even a user. The various attacks can be best characterised by observing the behaviour of the computer system as providing the information. Figure 2.4 illustrates the four general categories of security attacks.

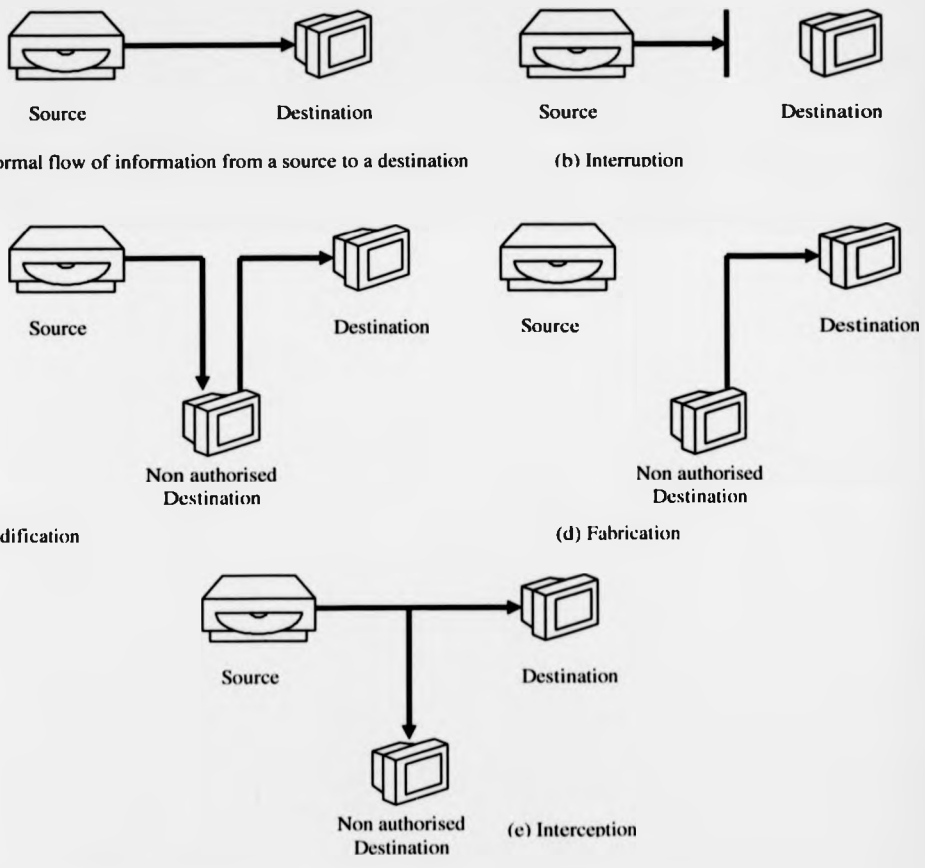


Figure 2.4: The main four categories of possible security attacks.

Figure 2.4 (b) depicts the interruption attack, according to which an asset of the system is destroyed and or becomes unavailable, thereby automatically affecting the availability of the system. Whenever the eavesdropper not only gains access to the transmitted information but also tampers with it, the attack is called 'modification' (Figure 2.4(c)). This is an attack on integrity. A simplified version of the previous way of intrusion is the so called 'interception', wherein the contents of the data become known to unauthorised parties, resulting in the severe decrement of system's confidentiality. The last section in Figure 2.4 (d) demonstrates an attack targeting authenticity, and known as 'fabrication'. Its main attribute is the insertion of counterfeit objects into the system. The result is that the recipient will assume that the received data has a valid user originator while in reality it does not.

Many authors of data security books [12-15] proceed to an extra categorisation according to which all these attacks are either passive or active. The former attacks have to do with eavesdropping and monitoring transmissions and follow the principles of the interception as described above. Theoretically all electronic transmissions (such as WWW, e-mail, telnet and so on) can be monitored. The main objective of the eavesdropper is to obtain information that is being transmitted. Traffic analysis and the release of message contents constitute the two types of passive attacks.

The 'release of message content' is related to any disclosure of the transferred information to unauthorised parties. All types of digital data transfer may contain

sensitive information and therefore there is the imperative need to protect the contents of these transmissions from possible monitoring attempts.

In an attempt to protect the transmitted data, a technique for masking the contents is employed. As indicated in Section 2.3 this technique is also known as encryption and addressed in Chapter 4. Traffic analysis is the analysis of the encrypted messages, aiming to extract useful information for the hacker. This information indicates the length of messages, their frequency, their times of origination, their surpass and their destination. This information can sometimes prove to be very useful in guessing the nature of communication that was occurring.

Passive attacks are very difficult to detect because they do not involve any alteration of data. It is however possible to prevent the success of these attacks and it is necessary to be proactive rather than reactive. An example of protection against traffic analysis attacks is the generation of fake traffic continuously even when idle from the regular communications point of view. An outsider who monitors the medium (link or free space) then always sees the link busy and cannot deduce anything from the mere fact that level of activity is high at some point in time; it is always high. Passive attacks are usually employed for gathering information that can be used later in active attacks. Figure 2.5 summarises the main passive threats a network system may be exposed to.

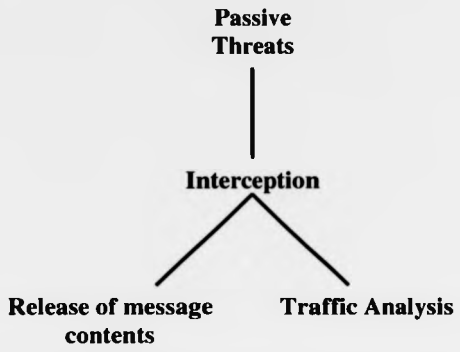


Figure 2.5: Passive Network Security Threats

Active attacks employ more overt actions within the network or system. They involve modification of transmitted data and attempts to gain unauthorised access to systems. As a result they can be easier to detect but at the same time they can be much more devastating to a network. They are subdivided into four categories: masquerade, replay, modification of messages, and denial of service as can be seen in Figure 2.6.

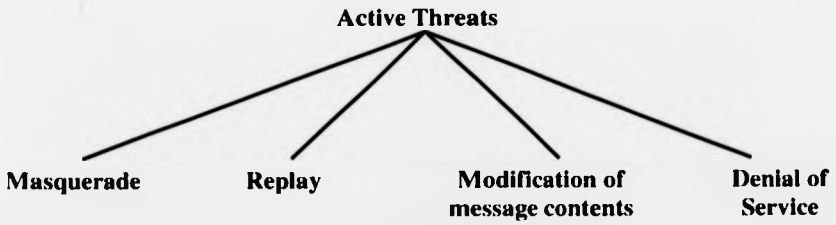


Figure 2.6: Active Network Security Threats

A *masquerade* takes place when an adversary (either a valid or invalid user) pretends to be someone else, another valid user. This attack requires the capture of authentication sequences and their later replay. It can take place in a network where different entities try to obtain more privileges by impersonating an entity that has those privileges.

An outsider executes a *replay attack* by intercepting and storing a legitimate transmission between two systems and retransmitting it at a later time. Theoretically encryption of the transmitted information cannot cause any difficulty to the adversary, thus the best defence to this attack is to employ session keys, check the time stamp on all transmissions, and use time-dependent message digests [2].

Modification of messages, as its name implies, simply means that part of the legitimate message is altered. It can also mean that messages are delayed or reordered to produce an unauthorised effect. It implies that the recipient of the altered messages would not be in a position to detect the modification and therefore assume their originality both of contents and origin.

Denial of service (DOS) attacks are designed to shut down or render inoperable a system or network. They inhibit the normal use or management of the communication facilities. In this type of attack, the ultimate goal is not to gain access, steal or modify information. The direct benefit to the attacker, in most cases, is non-existent and the real motives of the attack move to the sphere of vandalism.

Such attacks can be used to exact revenge or to punish an individual or a company for some perceived injustice. They can prevent almost any Internet server from operating and unlike other forms of hacking, DOS attacks do not require a great deal of experience or intelligence to succeed. There are many different types of DOS attacks. The most important ones are: ping of death, 'synchronise sequence number' (SYN) Flood, spamming and smurfing. SYN Flood appeared in the late 1990s and its name origin is based on the fact that it relied on the synchronisation packets that opened a TCP connection. The ping of death exploited a flaw in many vendors' implementations of ICMP (part of the IP of TCP/IP and operates at the Internet layer using the IP datagram to deliver messages [2]). SPAM is called any unwanted e-mail, a problem that all Internet users with e-mail accounts experience throughout their daily existence. The smurf attack uses forged ICMP echo request packets and directs them to IP network broadcast addresses. Its name has been taken from the source code employed to launch the attack (smurf.c).

To conclude the section on active attacks it is important to mention that there is a more serious denial of service attack called distributed denial of service (DDOS) attack. According to DDOS, the unauthorised entity should use a large number of computers on the Internet in order to launch a successful hacking operation. A strong attack might be translated into dozens or even hundreds of computers. Each of the subverted hosts waits to be 'triggered', from the attacker, before starting the DDOS attack. A list with the names of the most known attacks such as viruses, worms and Trojan horses (including the ones presented here) can be found in Appendix 1.

2.5 Wireless Communications

This section serves as an introduction to wireless communication, prior to addressing wireless security issues, vulnerabilities, strengths and limitations. In general it provides a discussion of the fundamental principles that govern non-wired transmission of information. This brief summary of the wireless arena is important because this research work utilises the wireless environment to achieve its main objectives.

Wireless communications constitute an area of digital communications that has evolved rapidly throughout the last decades. With the past and recent proliferation of handheld mobile technologies and devices, vendors have moved aggressively to extend the wired network through mobile gateways, allowing businesses and service providers to operate with confidence. Pervasive in business and increasingly preferred for personal communications, wireless appliances are well on their way to becoming ubiquitous personal accessories [16]. Today, wireless voice and data services are on the verge of attaining the economies of scale of true mass media, with all their desirable and undesirable cultural impacts. The real value for a wireless user resides with the ability to communicate at any time, over great distances, whilst in motion, at a relatively low cost. The high-speed data services provided to the mobile user constitute an extra strength.

Wireless communications is the process of communicating information in electromagnetic media over a distance through the free-space environment, rather

than traditional wired or other physical conduits [17]. The principal wireless technologies in use today are infrared and radio frequence (RF). The various wireless generations are distinguished by the bearer technologies that transmit wireless communications (table 2.1).

Table 2.1: Technologies to Generation

Generation	Architecture	Example Technologies
1st (1G)	Analog	AMPS, N-AMPS, NMT, TACS, FDMA
2nd (2G)	Digital	CDMA, TDMA, GSM
3rd (3G)	2nd Generation Digital	SMS, EDGE, GPRS, USSD, WCDMA, WATM
4th (4G)	3rd Generation Digital	Builds on CDMA, TDMA, GSM

Currently a large portion of the world is attempting adaptation to third generation technologies (3G), having spent billions of sterling for the acquisition of the 3G licenses. Even in this transitory stage from second to third generation, it is surprising that analogue cellular mobile systems are still widely deployed, and perform a vital function in largely undeveloped areas. From the Table 2.1 above, 1G systems consist of Advanced Mobile Phone System (AMPS), Narrowband Advanced Mobile Phone System (N-AMPS), Nordic Mobile Telephone (NMT), Total Access Communication System (TACS). The radio transmissions for this generation's systems have been dominated by frequency modulation (FM) (Appendix 2).

Moving to the next generation (2G) there were two significant changes towards a more efficient way of transmission and these were the Time Division Multiple Access (TDMA) and Code Division Multiple Access (CDMA). Digital handheld devices divided voice or data into selected fragments of frequency, time and code prior to the transmission. The channels' bandwidth potential could now be more effectively utilised.

Global Systems for Mobile Communications (GSM) was the first successful second generation wireless technology to be deployed and was not compatible with the previous analogue systems. GSM divides the total frequency spectrum into 200 kHz FDMA blocks. Each of these contain 8 eight TDMA time slots thus allowing up to eight users per frequency channel. Two 200kHz waveforms are necessary to establish a duplex connection. Control of the guard times between slots is more straightforward in TDMA than the filtering required for FDMA. The operating frequency spectrum for GSM technology is illustrated in Table 2.2 [18]:

Table 2.2: GSM Operating Frequency Spectrum

GSM 400	450.4 MHz to 457.6 MHz paired with 460.4 MHz to 467.6 MHz 486 MHz paired with 488.8 MHz to 496 MHz
GSM 900	880 MHz to 915 MHz paired with 925 MHz to 960 MHz
GSM 1800	1710 MHz to 1785 MHz paired with 1805 MHz to 1880 MHz
GSM 1900	1850 MHz to 1910 MHz paired with 1930 MHz to 1990 MHz

All information required to identify the user (profile) and the user's subscribed service details (access privileges) is contained within the memory of a smart card (discussed in detail in Chapter 4) known as the subscriber identity module (SIM) card. This also contains information about where the wireless device was activated and its assigned number. SIM is installed in the wireless device and its number is unique, providing a basic level of security. Cryptographic algorithms protect the valuable information stored in the SIM card.

Third Generation wireless technology (initially known as the Universal Mobile Telecommunications System, UMTS) is driven mostly by the need to satisfy the vast number of wireless users. The addition of more features and conveniences in combination with the significant improvement of those already existed, move wireless communications forward significantly from the provision of a pure telephony service. The short message service (SMS) is the most popular and important, with global acceptance, 3G technology. Actually SMS is a 2G product that transitioned very well into modern telephones. Equally important are the Enhanced Data rates for Global Evolution (EDGE), Wideband Code Division Multiple Access (WCDMA), Unstructured Supplementary Service Data (USSD), General Packet Radio Service (GPRS) and Wireless Asynchronous Transfer Mode (WATM) Appendix 3. 3G systems with data rates up to 2Mbps, offer the availability for wireless real-time video links (a quite expensive service though, costing thousands of sterlings).

2.6 Wireless Security

"Proper online security habits must become second nature to protect our privacy and the broader interests of society. These include all of the obvious things that we should do, but often don't: changing passwords; disconnecting from the Internet when it is not in use; running anti-virus software daily; changing the default password whenever a new device is purchased; and using appropriate security and encryption services. Nowhere is the development of this new security culture more important than in the wireless theatre of operations." J. M. McConnell, Vice Admiral, USN (Retired)

Former Director of the National Security Agency (NSA), 1992-1996 [15].

The employment of the air as a medium for data transfer has many advantages but at the same time introduces restrictions unknown to the wired world. One of the main advantages of wireless communication is the plethora of the different types available. This gives the ability to cover a wide range of communication needs from the simpler (telephone conversation) to the most complex (video conference based on the handheld device).

Wireless communications have made a tremendous impact on both businesses and personal life. They may thus be considered successful and offer considerable benefits. It is of greater importance at next to analyse thee drawbacks of current technology in an attempt to offer further improvement by identifying potential

vulnerabilities. Disadvantages such as health concerns, bit error rate and fading are of great interest but fall outside of this research work focus, thus will not be described here. Interested readers are referred to [19, 20].

The limited bandwidth, memory and processing capabilities of wireless devices such as mobile phones, pagers and personal digital assistants (PDAs) will here be seen through the security dimension and treated as the reasons making them 'weaker' than their wired counterparts. Wireless security, by its nature, violates fundamental security principles such as authentication and nonrepudiation, by not ensuring the identity of the user and the device, nor preventing the sender of the message from denial of service attacks. Although the fact that wireless has less physical assets to protect, at the same time, the very nature of the airwaves, makes it easier to hack. Travelling through the air gives many users ready access to the transmission medium. Given the right equipment, the wireless signal could be intercepted and/or modified.

It is safe to state that security is the largest challenge facing wireless computing. The ease of Wireless Local Area Networks' (WLAN) deployment helps in the continuing growth in popularity. Organisations are rapidly deploying wireless infrastructures based on the IEEE 802.11 standard [21]. It is the most mature wireless protocol and supports numerous WLAN technologies in the unlicensed bands of 2.4 and 5 GHz. It utilises the same Medium Access Control (MAC) for two physical layer (PHY)

specifications namely Direct-Sequence Spread Spectrum (DSSS) and Frequency-Hopping Spread Spectrum (FHSS).

The 802.11 standard uses Wired Equivalent Privacy (WEP) protocol, responsible for enhancing authentication (preventing unauthorised access) and privacy (preventing data tampering and compromise). WEP aims to ensure that WLAN systems have a level of privacy and authentication that is equivalent to that of a wired connection. It secures the confidentiality and integrity of data on 802.11 WLAN systems and provides access control through authentication.

Unfortunately, WEP contains significant flaws in the design [22, 23]. Particularly, the ones described in [22] comprise the protocol's ability to protect the network. Some other vulnerabilities in the two access control mechanisms that exist in access points built using Orinoco/Lucent 802.11 Wavelan PCMCIA cards, and a simple eavesdropping attack against the 802.11 specified shared authentication mechanism are described in [24]. The above technical and implementation weaknesses, which can be exploited by hackers, resulted in the recently issued draft specification [25] intended to enhance the security of these networks.

The supplementary draft addressed many concerns related to compromise and alteration of data and unauthorised access of the current design. Nevertheless, these will be ineffective in preventing denial of service attacks against 802.11 wireless systems. As it is stated in the same document, KERBEROS (a trusted third-party

authentication protocol, initially designed for TCP/IP networks) is invoked as an upper layer protection measure. However, KERBEROS does not prevent denial-of-service attacks [26].

The discussion of WEP protocol, emphasising mostly its vulnerabilities, can be summarised in the following statement from the Wireless Ethernet Compatibility Alliance (WECA):

> "It is important to emphasise that WEP was never intended to be a complete end-to-end security solution. It protects the wireless link between the client machine and access points. Whenever the value of the data justifies such concerns, both wired and wireless... should be supplemented with additional higher level security mechanisms such as access control, end-to-end encryption, password protection, authentication, virtual private networks, or firewalls." [27]

It has to be clear, at this point of the thesis, that the digital communication community faces many difficulties and threats in wired communication, which actually become even tougher when the communication medium is the air (wireless). Methods of encryption should stay relatively low-level to accommodate the power and speed constraints imposed by wireless devices. As chip technology advances (longer battery life and higher performance), it is evident that the level of security will also increase.

An appropriate epilogue for this section is the presentation of the specification that a secure mobile device should have namely:

- Relatively low computing power (compared to desktop PCs).
- Limits to the type of cryptographic algorithms a device can support.
- Limited storage capabilities.
- Power conservation imposed by functionality limitations.
- Fundamental restrictions on bandwidth, error rate, latency, and variability.
- Small footprint and compact I/O.
- Limited display capabilities; GUI becomes more challenging with different display form factors
- Usability and user experience issues.
- Throughput sensitivities to protocol overhead and compression.

Most existing security technologies, protocols and standards have been designed for the wired/high bandwidth environment. In many cases they are not well suited for the wireless mobile environment because they have too much overhead and exhibit tight timeouts [15].

2.7 Biometrics

Biometrics identify people by measuring some aspect of individual anatomy or physiology, some deeply ingrained skill, or other behavioural characteristics, or something that is the combination of the two. They use unique personal traits for security purposes (most probably for authentication).

Biometrics are closely related with human life and there are numerous every day examples emphasising their importance in civilised societies. For several decades, passports and national individual descriptions (IDs) carry sensitive information (for the user) such as physical description, photographs and signatures. In particular, photo ID cards' roots go back to 1930s, when they were firstly employed to allow a small number of army officers to gain access to high-security military bases during World War II.

The security problems both in computer and network environments, as described in previous sections, made clear from early days that the employment of biometrics for strengthening and improving computer-based authentication was just a matter of time. Before continuing to the difficulties a transition like this is destined to face, it is important to categorise the different biometric types. Fourteen different types of biometrics have been identified that have been used, experimentally or in practice, for authentication. These techniques fall into two categories: those that measure behaviour and those that measure physical traits [28].

The first category consists of techniques that measure physical features of human bodies that should be unique among most or all of the population and should not significantly change with the elapse of time and/or by human behaviour. The most well known biometric types fall into this category are: *fingerprints*, *hand geometry*, *eye retina scan*, *eye iris scan* and *face recognition*.

In most cases the names used for each biometric type, provide a brief definition of what feature of human body electronically "observed" to authenticate the user. Fingerprints, one of the most popular biometric techniques, employ a scanner to read the user's fingerprints. The readings taken are then compared with valid user's readings, a process known as pattern matching. This technique has been used for several decades by police departments, trying to identify criminals.

Hand geometry readings are related to user's finger size, thickness and palm geometry. Recognition Systems Incorporation have manufactured systems using this biometric approach on a commercial basis [29]. Eye iris and retina scan techniques require the employment of a camera that records the iris' distinctive texture and the retina's distinctive pattern of veins respectively. Face recognition systems aim to identify people by examining images of faces. The general interest in this area helps towards the development of applications that try to identify whether users lie (lie detector), by observing movements of face's muscles [29].

The second category, measuring behavioural traits, consists of *speaker recognition*, *written signature* and *keystroke dynamics*. These three biometric types record traits that can easily be affected by human behaviour and therefore change rapidly as time passes by. The last point distinguishes these biometrics from the ones measuring physical traits because they do not have to record the same phenomenon each time.

The nature of this research work makes speaker or voice recognition the most important of the different types, in both categories. Voice and more particularly speech, is explained in depth in the following chapter. Voice recognition systems focus on user's voice, speech behaviour and try to identify unique characteristics that distinguish him/her from the rest of population. The distinctive speech patterns, sometimes referred to as voiceprints, are collected carefully via the use of specific microphones. Recorded spoken phrases will be said more quickly or with different emphasis. Such variations should be factored out by a sophisticated biometric system. Nevertheless, such variations will inevitably crop up, affecting the efficiency and performance of the application. As a biometric system, voice recognition has many restrictions, such as background noise, mainly arising from the speaking environment. In cases where the background noise is of great level, such biometric systems are condemned to failure. Additionally in many systems, the microphone filters out most differences between live speech and high quality audio playback, thus allowing the system to be fooled by tape recordings of the user's voice.

Written signature and keystroke dynamic systems read a written signature accompanied with the dynamics of pen motion and sense a user's behaviour whilst typing at a computer keyboard respectively. Researchers have also built systems to collect information about how a user walks and use distinctive features of the user's gait for authentication. Such systems rely on special camera systems to collect the information. At present, no practical implementations have been demonstrated [30].

All biometric systems follow three distinctive steps in order to authenticate a valid user and reject an impostor [31]. The first step is to use, in most cases, sophisticated devices that in reality constitute the sensors of the system. These sensors record the necessary trait associated with the biometric. The collection of this sensitive data is a very important part of the authentication process and hence many precautions, for example control of the background environmental conditions, should be taken to guarantee a successful end product. During the second step the biometric system performs feature extraction on the digitised data, aiming to identify the distinctive features associated with the particular biometric. In other words, this step tries to extract a biometric signature that represents the trait read. Biometric signatures are eventually used in the final step to extract biometric patterns, closely associated with the user. These patterns are then compared (pattern matching) and the result of the comparison indicates whether the user is valid or not.

Biometric systems' performance can be measured in terms of *biometric accuracy*. Depending on the quality of the design of the system, there are cases where a valid user cannot have access because of a "poor" biometric signature that do not match closely with the biometric pattern (*false rejections*). Additionally there are other cases, where an impostor produces a close match between his biometric signature and the biometric pattern, and therefore gains privileges of a valid user (*false acceptance*). Both types of biometric system's malfunctions should be limited to a minimum level. Ideally a system should always accept valid users and reject invalid

ones. It is well understood that, inevitably, this is not the case. Moreover, in real life conditions, a biometric signature always varies from the original biometric pattern. This explains the presence of tolerance rates within such systems. The tolerance rate is nothing else than a percentage of variation treated as acceptable by the designer of biometric applications. Depending on the implementation details, this percentage changes to meet the application's needs or requirements.

The false acceptances and rejections are closely related with the number of people using the system. As this number increases, the chance of finding a user producing a biometric signature that matches closely with someone else's biometric pattern increases significantly. A comparison between signatures coming from the same user produces a graph as shown in Figure 2.7.

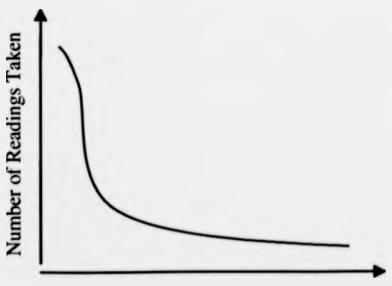


Figure 2.7: False Rejections [28]

Figure 2.7 illustrates what it was expected to be the case. The same user most of the time produces biometric signatures that vary slightly with each other. Nevertheless, rarely there may be signatures that differ significantly with each other.

The following graph, presented in Figure 2.8, is the result of the comparison between biometric signatures produced by different users. This time the graph is reversed, as if it is the mirror image of the previous one presented in Figure 2.7. Based on probabilities it is expected that most of the recordings between different people's biometric signatures, will be totally different and therefore impossible to produce a successful match. Unfortunately, there will be few cases according to which the variation between the biometric signatures will not be great, resulting in the false acceptance of a user.

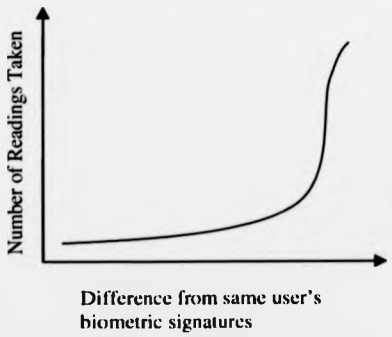


Figure 2.8: False acceptances [28]

It is vital for biometric systems' performance, to keep a balance between usability (false rejections) and security (false acceptances). The designer of such systems

should take into serious consideration the specific characteristics that define the biometric application, and set up the threshold for the matching procedure. A general example of this threshold and its catalytic effects in the performance of biometric authentication is presented in Figure 2.9. As it can be seen in the following Figure, the doted line represents the threshold setting. As the threshold increases, moving to the right hand side, the false rejections decrease and therefore the usability of the system increases but the paid price is the increase of the false acceptances lowering the level of security. On the other hand, as the threshold moves to the left hand side, the security strengthens but the usability of the systems declines.

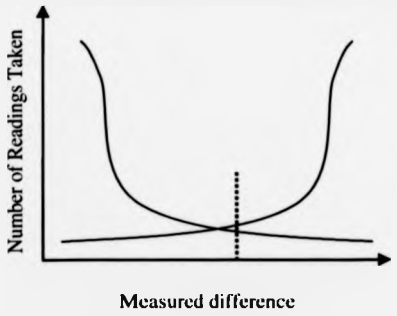


Figure 2.9: Usability versus Security [28]

Chapter 3: Theoretical Background to Voice Analysis

3.1 Content and Background

Human voice and more especially speech is in a manner a measure which defines the identity of the person. This happens because not all the people can produce the same "sound" frequencies. Speech consists of vibrations produced in the vocal tract. The vibrations themselves can be represented by speech waveforms. Human voice is contained within a specific range of frequencies, but there is a variation within that range, which makes one person's voice distinguishable from another's. This widely accepted difference presents the possibility of using voice characteristics to generate keys for encryption.

In this work, investigation commenced with the human voice. This preliminary stage aimed to ascertain how speech can be integrated into data security, especially cryptography. There are many applications commercially available which have managed to integrate human voice, with the best known being the Speech-to-Text software packages used for word processing. Another possible way of speech usage is voice recognition, currently employed by companies and organisations, in departments and zones of high security. It was soon reliably assured, from initial research, that speech has not been previously used for secret key generation. Hence the use of speech for this purpose provided a basis for a specific project.

This idea spans the areas of voice analysis and cryptography. In addition, the processing limits of mobile devices must be taken into account. Thus the work also needs to address signal processing issues, particularly in the context of devices such as smart cards. In the remaining sections of this chapter the area of voice analysis

will be introduced. The approach taken is to extract elements relevant to this particular project from a large existing body of work. The immediately following chapters will present some essentials of cryptography and smart cards in a similar fashion.

3.2 Brief History

The history of speech processing certainly does not begin with the digital signal processing engineer, nor even with the work of electrical engineers. In an interesting article surveying some of the history of speech synthesis, Flanagan [32] notes humankind's fascination with speech and voice from ancient times, and places the advent of the scientific study of speech in the Rennaisance when clever mechanical models were constructed to imitate speech. The first well-documented efforts at mechanical speech synthesis occurred in St. Petersburg and Vienna in the late eighteenth century [33]. The 1930s, a century and a half later, are often considered to be the beginning of the modern speech technology era, in large part due to two key developments at Bell Laboratories. The first was the development of pulse code modulation (PCM), the first digital representation of speech (and other waveforms) which helped to pioneer the field of digital communications [34]. The second was the demonstration of the Vocoder *(Voice Coder)* by Dudley [35], a speech synthesiser, the design of which first suggested the possibility of parametric speech representation and coding. The subsequent decades have seen an explosion of activity roughly concentrated into decades. Here some keys and relevant development are outlined: intense research on the basic acoustical aspects of speech

production and concomitant interest in electronic synthesisers in the late 1940s through the 1960s [36], which was spurred on by the invention of the spectrograph in 1946 [37]; advances in analysis and coding algorithms (linear prediction, cepstrum) in the 1960s made possible by the new digital computing machines and related work in digital signal processing; development of temporally adaptive speech coding algorithms in the 1970s; and vast interest in speech recognition research in the 1970s and 1980s and continuing into the 1990s, grounded in the development of dynamic programming techniques, hidden Markov modelling, vector quantisation, neural networks, and significant advances in processor architectures and fabrication.

3.3 Generation of speech

Human speech sounds are produced by modulating the air flow in the speech tract in order to create some kind of acoustic source. The source of power in all speech sounds is the respiratory system pushing air out of the lungs. Movements of articulators produce speech, and therefore, in some space, air is constrained to follow a smooth trajectory with occasional abrupt accelerations.

The main organs of the human body responsible for producing speech are the lungs, larynx, pharynx, nose and various parts of the mouth, which are illustrated by the cross-section shown in Fig. 3.1. Muscular force to expel air from the lungs provides the source of energy. The air flow is modulated in various ways to produce components of acoustic power in the audio frequency range. The properties of the resultant sound are modified by the rest of the vocal organs to produce speech.

The process of acoustic resonance is of prime importance in determining the properties of speech sounds. The principal resonant structure, particularly for vowels, is known as the vocal tract; it starts at the larynx and extends up through the pharynx and mouth to the lips. For some sounds the nose is also coupled in to make a more complicated resonant system. The frequencies of the resonances and the way they move with time, and to a lesser extent their intensities, are crucial in determining what is being said. The main resonant modes of the vocal tract are known as formants, and by convention they are numbered from the low-frequency end. For conciseness they are usually referred to as F_1, F_2, F_3, etc. In general F1 and F2 (normally in the range 250Hz to 3kHz) are the most significant in determining the phonetic properties of speech sounds, but some higher-frequency formants can also be important for some phonemes.

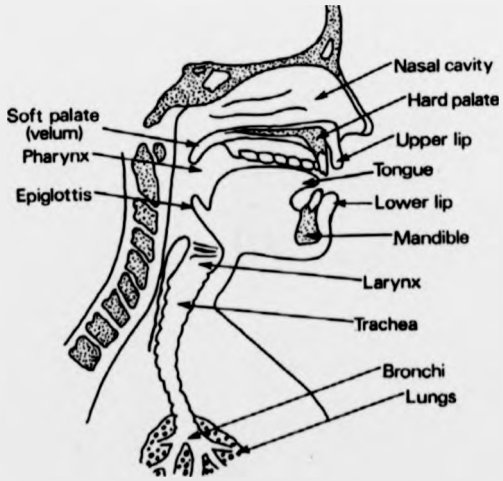


Figure 3.1: Diagrammatic cross-section of human head, showing the vocal organs [38].

There are three parameters that determine the voice source:

- The amplitude of the oscillation
- The voice source spectrum
- The fundamental period

Generally the entire procedure of speech production can be summarised and simplified in Figure 3.2.

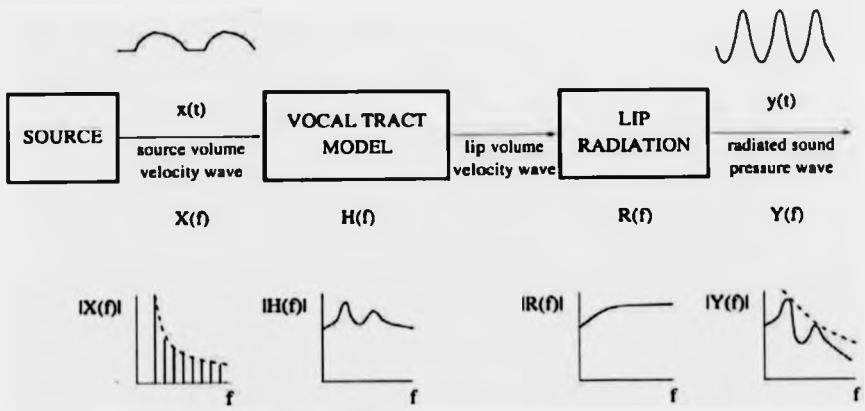


Figure 3.2: A basic model for speech production [39].

3.4 Modelling the Speech Signal

Speech waveforms are rich in information but are highly redundant in structure. Storage of acoustic data is therefore inefficient and a more compact parametric representation of the information conveyed by the signal is desirable. An ideal model should exploit the redundancy in the speech signal to give data compression whilst capturing the distinguishing features of the waveform and the underlying dynamics of the production mechanism. For speech coding and synthesis applications, the ability to regenerate the original speech waveform from the model is also necessary.

The acoustic speech waveform varies slowly with time as different sounds are produced so the frequency properties of the signal are constantly changing. A time-varying model of the waveforms is needed for which the model parameters are constantly updated at a suitable rate. Typically, a short-time analysis is used, in which the speech waveform is divided into a sequence of overlapping segments of about 20ms in duration, and a new set of model parameters calculated for each segment. Since the articulators move relatively slowly, the vocal tract resonances remain fairly constant for durations of about 10ms [40] which permits an update rate (frame rate) of 10 ms. Even the fastest transitions in plosives (sounds, such as "p", characterised by a rapid air release) can be captured relatively well by an update rate of 5ms.

Two approaches to developing a model are articulatory modelling and acoustic modelling. The first of these aims to represent the vocal tract and movement of the

articulators in as much physiological detail as possible and assumes that a similar underlying system will generate a similar output. Articulatory models have the potential for good reproduction from simple control signals and can reproduce all the perceptually relevant effects of real speech, such as co-articulation [41]. However, the dimensions of the vocal tract and a detailed analysis of the movement of the articulators are needed. Such information is difficult to obtain and often requires intrusive measurement techniques. In context, the acoustic modelling approach models the speech waveform directly in either the time or frequency domain. The models are easy to construct because only the speech waveform is required, which is easily obtained using a microphone. An exact match of the waveform or spectrum is not needed for perceptually good synthesis and events which are not perceptually relevant need not be modelled.

The most popular technique for speech modelling applications, such as speech coding and speech synthesis, is the time-domain acoustic modelling method known as linear prediction (LP) [42].

3.5 Sampling Speech

The implementation of security methods, in common with all modern speech processing, is digital. Hence, the first step must be conversion of the analogue voice signal to digital signal. The conversion requires sampling and filtering which will now be outlined.

An important parameter in speech analysis is its bandwidth or the range of frequencies it occupies. In speech processing the normal bandwidth often used is of 4-5 kHz, which is found to be a reasonable compromise between quality and sampling rate and is perfectly adequate for voiced sounds.

Another equally important and related parameter in speech processing is the sampling rate. To convert a signal from continuous time to discrete time, signal sampling is used. The sampling theorem states that if a function g(x) has a Fourier Transform (FT) G(k) then the sampling function has a Fourier Transform of [43]:

$$Gx(k) = G(k) \circ \Delta x(k)$$
$$Gx(k) = \frac{1}{x}\sum_{m} G(k - \frac{2\pi}{x})$$

Where Δx is the FT of a sequence of dirac Delta functions.

It was Nyquist who first clarified the application of sampling to communications. In 1925, in an article titled "Certain Factors Affecting Telegraph Speed" [44], he proved that the number of telegraph pulses that can be transmitted over a telegraph line per unit time are proportional to the bandwidth of the line. In 1928, in an article "Certain Topics in Telegraph Transmission Theory," [45] he proved that for complete signal reconstruction, the required frequency bandwidth is proportional to the signalling speed, and that minimum bandwidth is equal to half the number of code elements per second. Subsequently, Russian engineer V.A. Kotelnikov published a proof of the sampling theorem in 1933 [46], and in 1949 American

mathematician Claude Shannon unified many aspects of sampling and founded the larger science of information theory [47].

According to the Nyquist theorem the sampling rate must be at least twice the higher frequency present in the signal to avoid aliasing. This means that an adequate sampling frequency for voiced sounds should be of 8-10 kHz. In the speech related world instantaneous values of signal's parameters must be used because speech is time-variant. It is not possible to gather data via long term observations.

One final corollary of the sampling theorem is that complete reconstruction of a continuous time signal from discrete equally spaced samples is feasible if the highest frequency in the time signal is less than half the sampling frequency. It can be said that this theorem bridges the gap between continuous and digital time signals.

Signals must be filtered prior to sampling. Theoretically the maximum frequency that can be represented is half the sampling frequency. In practice a higher sample rate is used to allow for non-ideal filters. The signal is now represented at multiples of the sampling period, T, as S(nT) which is also written Sn. Telephone speech is sampled at 8kHz. 16kHz is generally regarded as sufficient for speech recognition and synthesis [48]. The audio standard is a sample rate of 44.1kHz (Compact Disc) or 48kHz (digital audio tape) to represent frequencies up to 20kHz.

3.6 Filters

Sampling is a necessary component in digital signal processing. However, if the sampling rate does not follow the Nyquist theorem, according to which the sampling rate should be greater than or equal to twice the highest frequency present in the signal, a phenomenon known as aliasing occurs. One way to eliminate the unwanted effects of aliasing is to apply a low pass filter (LPF) during sampling. This will result in the frequencies which are too high and inappropriate for the sampling rate being blocked and the valid ones being allowed through. Filters are used to reject undesired frequency components (noise) and allow passing signals having frequencies in a certain range. Ideal filters have the same gain, at all frequencies, in their passband and zero outside of it.

There are two different types of filters: *passive* and *active*. The first one uses capacitors, resistors and inductors, while the second one takes its name from the active elements from which it is constructed.

In this section information about band filters can be found. There are four main types of band filters: low-pass, band-pass, high-pass and band-stop. Each one of them is depicted in Figure 3.3.

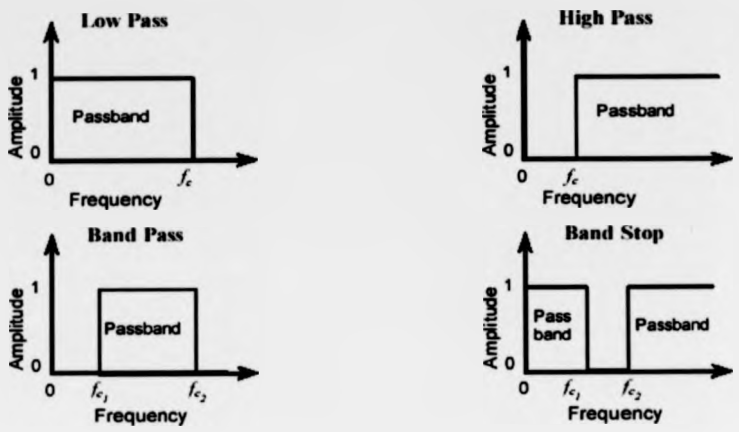


Figure 3.3 The four basic filter types [49].

Under the general category of Linear filters there are two other sub-categories: Finite Impulse Response (FIR) and Infinite Impulse Response (IIR) filters. An FIR filter produces an output, y(n), that is the weighted sum of the current and past inputs, x(n).

$$y_n = b_0 x_n + b_1 x_{n-1} + b_2 x_{n-2} + \ldots + b_q x_{n-q}$$

This is shown in Figure 3.4 with z^{-1} representing a unit delay.

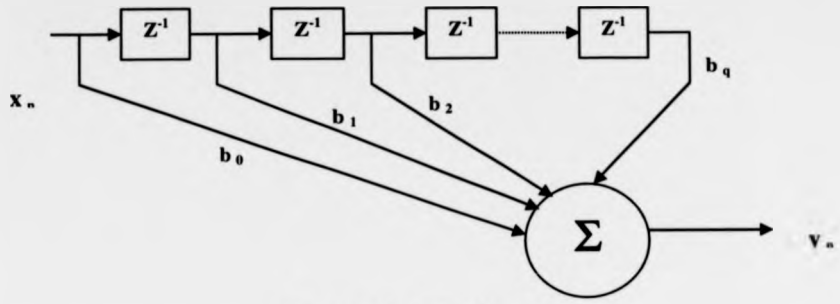


Figure 3.4: A FIR filter

FIR filters are computationally expensive to implement but need not introduce phase distortions, useful in processing high quality speech. IIR filters are often more efficient, but can not be designed to have exact linear phase.

An IIR filter produces an output, y_n, that is the weighted sum of the current and past inputs, x_n, and past outputs. The Linear Predictive model is a special case of an IIR filter, as illustrated in Figure 3.5.

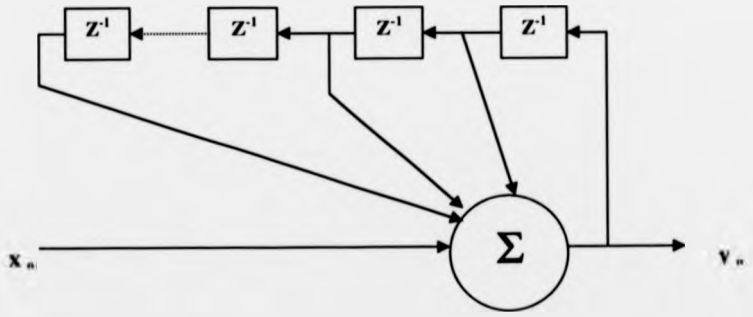


Figure 3.5: An IIR filter

Common types of IIR filter:

Type	Characteristics
Butterworth	Maximally flat amplitude
Bessel	Maximally flat group delay
Chebychev	Equiripple in pass-band or stop-band
Elliptic	Equiripple in pass-band and stop-band

According to the gain behavior in and out of the passband, filters are divided to another two categories: Butterworth and Chebyshev. A comparison diagram for both of them is illustrated in Figure 3.6.

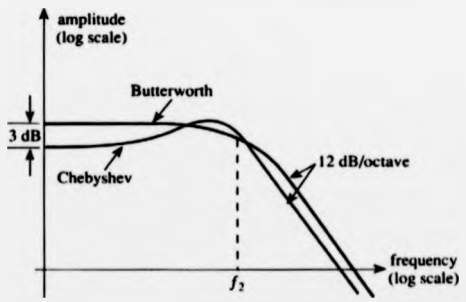


Figure 3.6: Behaviour of Chebyshev and Butterworth filters [50].

From the previous diagram it derives that in the Butterworth case the gain is simply the same within the passband (Maximally flat response), but outside of it falls off with a not fast enough rate. A Chebyshev filter has a bigger variation in the passband but transition band is narrower.

Furthermore, the 'order' of the filter is an important parameter in designing digital filters. It represents the number of previous inputs used to calculate the current output and thus determines the filter's complexity. The 'order' of a digital filter can be any positive integer.

3.7 Spectrograms

A spectrogram plots the short-term power in different frequency bands as a function of time. Originally it was implemented by recording onto magnetic tape, then passing the tape over a rapidly spinning drum contain the playback head (one loop around was the "window"), the resulting signal was fed to a band-pass filter whose centre frequency was determined by the position of a stylus on head sensitive paper. The resulting signal being used to mark the paper with the intensity of the signal at that frequency.

The system designer has to take into serious consideration two important factors before choosing the filter's bandwidth. The narrow band analysis resolves harmonics but blurs temporal detail such as burst onsets. On the other hand, wide band analysis may resolve fine temporal detail but looses fine frequency detail. In most

applications this trade off is manifested as the choice of window size. Good rule of thumb is to include a few pitch periods (20ms to 32 ms). Additional information about spectrograms and digital filters is provided in Chapters 5 and 6, together with the real life development and implementation of the research application.

Chapter 4: Elements of Cryptography and Smart Cards

4.1 Introduction

"... at the heart of our concern to protect 'privacy' lies a desire, perhaps even a need, to prevent information about us being known to others without our consent." [51]

Encryption is the transformation of text or other data into coded form, often compressed in addition, for transmission over a public network or for protection of data stored on disks. Another possible definition is the following: "Encryption is basically an indication of users' distrust of the security of the system, the owner or operator of the system, or law enforcement authorities." [52] On the other hand, Decryption is the action of applying a cipher to data that has been encrypted so as to recover the data in understandable form [6].

Unencrypted data is called plaintext; encrypted data is referred to as ciphertext. Encryption is the most effective way to achieve data security. Its algorithms use binary numbers called keys that are used to lock the data. The same keys at the receiving end are used to unlock the code allowing the recipient to read the plaintext.

Cryptography is defined as the science and study of secret writing. The word cryptography comes from Greek. Kryptos means hidden while graphia stands for writing. There are two cryptographic methods. The first one is called single key (or symmetric key) cryptography and the second one public-key cryptography or asymmetric.

The encryption algorithm determines how simple or how complex the process of transformation will be. As mentioned in Chapter 2, encryption provides confidentiality, integrity and authenticity of the information transferred from A to B. It eliminates many security risks and can be used to control access to data. However, it requires a good key management system. This system will allow the users to communicate in a secure way even if the secret/private encryption keys are lost.

4.2 Single Key System

A system can be characterised as single key one when the key used for encryption is the same as that used for decryption of data. When a user wishes to employ this specific method to encrypt a file then he/she has to find a secure way to exchange the secret key with the recipient. This is the main disadvantage of cryptosystems and is known as the key distribution problem [53].

No matter how many levels of key are used, there is always the need some keys to be transmitted in the clear. At the level at which keys are transmitted in the clear security can be achieved only by manual distribution of keys. In order to let this be done, control of physical and managerial security systems is found to be necessary [54].

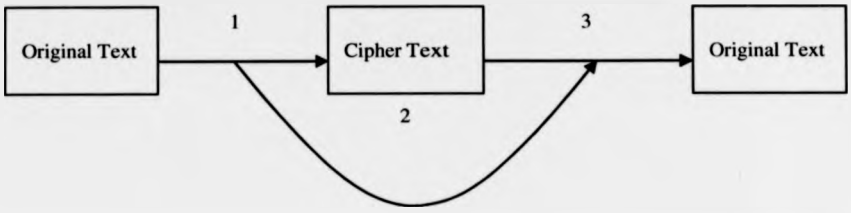


Figure 4.1 Single Key System

In Figure 4.1 a single key system is described schematically. The same procedure consisted of three steps should always be followed. In Step 1 the sender encrypts the file with the secret key and therefore the original text becomes now a cipher text. Then in step 2 the secret key used for the encryption should be given securely to the recipient. Finally the recipient, being aware of the secret key, decrypts the file by using it. This means that cipher text becomes again original text allowing the recipient to fully understand it.

Single Key standards include the popular Data Encryption Standard (DES), Blowfish, Triple-DES, International Data Encryption Algorithm (IDEA), Rivest Cipher (RC2) and RC4 [53]. The first one of those is one of the best known and commonly used technologies and was originally designed in early 1970s. It was adopted by the United States government in 1977. It was published by the American National Bureau of Standards, and is now an ANSI standard [54].

DES is a block cipher with an alphabet of 2^{64} letters, it encrypts data in 64-bit blocks and its key length is 56 bits. Most of the time this is expressed as a 64-bit number, but the eight least significant bits are parity bits that are ignored. At a fundamental level, the algorithm combines two basic techniques of encryption: confusion (a complex non-linear substitution method preventing the cryptanalyst from using the ciphertext to recover the key) and diffusion (a combination of substitution and transposition that transform the ciphertext to look like a random string of letters). The heart of DES is a Feistel cipher which is a sequence of nonlinear transformations. This was used in IBM's Lucifer cipher [53].

Single Key Systems are preferred for their high-speed. Additionally, single key is the system often chosen by vendors because of its ease of usage. It can be used as a bridge for users not having the same software because the algorithm can be included in the file.

On the other hand, sharing files with other users is a difficult task whilst someone attempts to give the secret key to another party, especially when files are being shared, then the key may possibly be intercepted by an unauthorised person. Furthermore, it may be reiterated that key management is very difficult and can be considered as a drawback.

4.3 Public Key System

In Public Key Systems there are two keys: a public and a private key. The essential difference between Single and Public Key Systems is while in the first one only one key is used for encryption and decryption, in the second there are two different keys. The public key is available to anyone and it is used for the encryption of a message. Therefore it can be published to folders or electronic libraries where no access limitation exist. The private key is kept secret and is used only by the receiver, in order to read his/her messages.

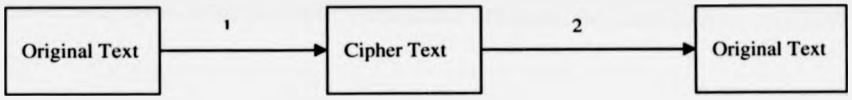


Figure 4.2 Public Key System

In Figure 4.2 a simple design of a Public Key System is illustrated. Number one represents the action of the sender to use recipient's public key and encrypt the plaintext. The recipient, in number 2, receives the cipher text and uses the secret key to decrypt it.

Public Key Cryptography was developed in 1977 by Rivest, Shamir and Adleman (RSA). In addition to RSA, public key system standards include the Elliptic Curve Cryptosystem (ECC) [55] and the Raike Public Key (RPK) [53, 56].

This technique of cryptography is more efficient than the private key cryptography. This is a fact because each user has only one key to encrypt and decrypt all the messages that he or she sends or receives. Pretty Good Privacy (PGP) is another example of public key cryptography mentioned in Chapter 2. PGP is an encryption software for electronic communications written by Philip R. Zimmerman [53].

The RSA public key cryptosystem is a method based on modular exponentiation. Its security is closely related to the level of difficulty of factoring large numbers. It is known to be not harder than factoring [57]. Public and secret keys in RSA are constructed from three positive integers n, p, and s where $p<n$ and $s<n$. The pair (p,n) is the public key and the pair (s,n) is the secret key.

To encrypt plaintext m and transform it to ciphertext c the following should be computed:

$$c = E[(p,n),m] = m^p \bmod n$$

To decrypt the ciphertext c and transform it to plaintext m the following should be computed:

$$m = D[(s, n), c] = c^s \bmod n$$

The values for n, p, and s can not be chosen randomly and only based purely on luck. Lets describe step by step the process of RSA keys generation. It consists of six steps in total.

1) Randomly select two large prime numbers q and r.
2) Let n = qr
3) An integer p should be randomly chosen, which has to be coprime to $\phi(n)$, where $\phi(n) = (q-1)(r-1)$
4) s should be computed such that $ps \bmod \phi(n) = 1$.
5) Public Key (p,n) should be published.
6) Secret Key (s,n) should be recorded [54].

Values for p and s should be high, assuring the secure encryption of any message.

The security of RSA depends on the problem of factoring large numbers. RSA is insecure if q and r are obtained and s is computed from p. This relates RSA and its security to the factorisation of large numbers. At the same time it has not been proved mathematically that this is the only possible way this public key algorithm may be broken.

In Public Key Systems the private/secret key should never need to be transmitted or even revealed to anyone. An advantage a system like this as compared to Single Key Systems is that a secure channel need not be employed. The ease of encrypting a

message by using recipient's public key, which can be retrieved by anyone, is also advantageous.

As is widely observed, it is rare that one method has only advantages compared to another, there will also be drawbacks. Public key systems are no exception and have the following disadvantages:

1) The major disadvantage of Public Key Systems is the length of private keys. People cannot memorise them and therefore it is necessary to store them securely on media. Automatically the intruder has now the opportunity to follow different approaches to compromise the key.

2) Communication between people who do not use the same software is, or at least used to be, a big issue. S/MIME, which will be briefly described in the last section of this chapter, has contributed the most on this direction.

3) Finally there is a problem with the speed. Public Key Systems are slower than Single key Systems for reasons described in Section 4.2.

4.4 Digital Envelope

In the previous two sections Single and Public Key Systems with their advantages and disadvantages, were described. Until now the impression may have been given that these two systems cannot exist at the same time. This is not always the case, and

there are some situations where both of them are used together. Single Key Cryptography, famous for its speed, provides the fastest encryption and decryption. Public Key Cryptography provides, in turn, a convenient method of transmitting the secret key. This process is known as a digital envelope.

In order to create a Digital Envelope the message is encrypted by using the secret session key. The recipient's public key is then used to encrypt the session key and is sent along with the encrypted message to the recipient. The latter uses his/her private key to decrypt the session key. The session key can now be used to decrypt the message.

This method uses the convenience of public key systems and the speed of Single Key algorithm. The key management becomes easier than Single Key Systems. At the same time the level of complexity is higher. This can be helped if the encryption software automates key management. There is also a problem sending e-mails to external users (they also need a key pair).

4.5 Characteristics of smart card technology

Smart Card is a plastic card, usually a credit-card, which contains a microprocessor and memory so that it can be interrogated by an ATM machine and keep a tally of transactions [6]. Smart cards have followed an evolution cycle that has lasted for many years. In the very beginning they could not be characterised as "smart" due to the lack of a microprocessor. There was only the ability to store a small amount of

information and therefore they were called memory cards. Depending on the importance of the stored data, the memory access was either protected or not.

Smart cards may also be divided into two categories based on the application purposes they serve. They can be designed to be used only for a single application (specific smart cards) or for a wider range of applications (multi-function smart cards).

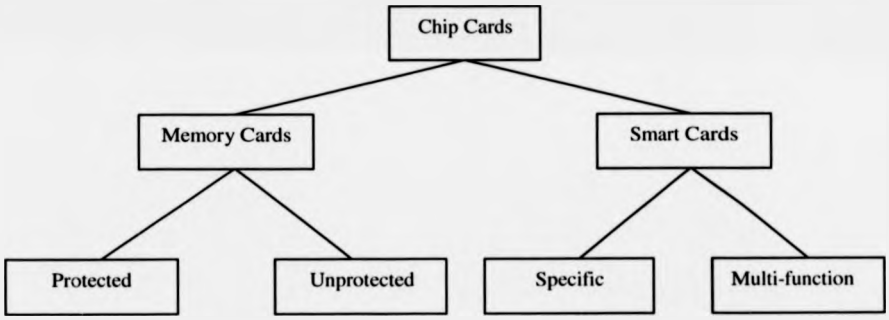


Figure 4.3: Memory and smart cards

This chapter focuses on smart cards and more specifically on their employment for cryptographic purposes. The core of the academic idea described in this thesis, has been based on the premise of utilising the smart card environment as an application platform. In order to be able to present smart card's contribution within the security

arena, it is necessary to explain first in brief what a smart card is and what it consists of.

The main characteristic of a smart card is the limited but existent processing capabilities. The chip of a smart card consists of a microprocessor, Read Only Memory (ROM), Electrical Erasable Programmable Read Only Memory (EEPROM) and Random Access Memory (RAM). Memory cards have only ROM and EEPROM and do not contain programmable logic. Depending on the need of the smart card employment there is a range of embedded CPUs, starting with an inexpensive 8-bit microprocessor and reaching up to 32-bit processors. Normally, ROM contains the card operating system and the information stored in there is written during the production. EEPROM is used for permanent storage of data, while RAM is the transient memory of the card and keeps the data as long as the card is powered [58].

Smart cards communicate with a host computer to enable the embedded processor to exchange information and commands. A successful communication link can be achieved either via the mechanical contact points of the card or via wireless transmission. Advantages and disadvantages are associated with each option and the final decision as to which communication method will be employed is application oriented. Wireless transfer of data between smart cards and host computer is economic in time and user friendly. On the other hand, the power for the processor

has to be transmitted as an electromagnetic wave, which limits the maximum power consumption of the processor and therefore the processor capabilities [59].

4.6 The security dimension of smart cards

One of the strongholds of smart cards is their high scale implementation in securing communication. They constitute the storage device for saving encryption keys, necessary for different cryptographic schemes. Usually, a symmetrical key is used to establish a secure session between a client and a server. This key, also called session key, is exchanged between the two communicating parties.

Additionally, encryption keys are necessary for securing the data withheld within the smart card. Important information such as credit card numbers, bank accounts and so on. should be kept secret at any cost, especially when it is stored in devices that can be easily stolen or lost. Smart cards employ different encryption algorithms depending on the manufacturer [60].

Moreover, the processing capabilities of smart cards can also be used to authenticate the card owner. Passwords or passphrases can be easily stolen or forgotten. This is the main reason that drove many companies to incorporate smart card technology for providing secure access to a group of people. The most commercially implemented authentication schemes follow the "two factor" approach, according to which the owner of the card has to provide a password in order to gain access to the data stored in it.

Furthermore, smart cards can serve as a highly secure storage device protecting the private keys of digital signatures. The most recent smart cards are able to execute the signing operation and even generate the signing key, inside the card. It has to be mentioned that the key used for signing the digital certificate cannot be exported by any embedded function. The latter ensures that additional copies of the private keys are very difficult to make. A characteristic example of this category smart cards is the product of RSA security, known as RSASecurID [61].

Chapter 5: Analysis of Human Voice

5.1 Introduction

This practical chapter presents the various processing phases involved in the analysis of the human voice for cryptographic purposes. As previously stated, the work aims to answer the question whether the human voice or specific characteristics of it can be employed as an alternative method to random number based secret key generation. Inevitably, a significant amount of both time and effort has been invested in the biometric part of the question. The results of this research are summarised in the sections to follow.

Speech is the most common method of communication amongst humans. Examples of everyday life demonstrate an extra attribute of the human voice, other than serving as a communication medium. Speech comprises of not only the meaning of the transmitted message but also incorporates information about the production of the human speech. This information varies significantly from user to user and is closely related to morphological characteristics of the human body. A detailed description of speech generation and its dependencies has been provided in Chapter 3. Providing a necessary training period, speech can be used as a measure helping in identifying the communicating party. The identification process is based on the individuality of the human voice, concentrating on the sound of the speech rather than in its meaning. This constitutes the core of the research work presented in this chapter.

The thorough analysis of the human voice can produce information that characterises the source of speech generation. Part of this information will be the

same among the human population because it will define some general source characteristics like gender. The focus here is on data that can not be easily reproduced and can be defined as unique for the speaker. This personal biometric signature has to be located within the voice signal and isolated. Accordingly, the speech has to go through a sequence of analytic steps as illustrated in Figure 5.1. Those processing phases, also presented in Appendix 4 as part of the overall secret key generation processing diagram, consist of the most fundamental principles of speech processing, in an attempt to lessen the processing requirements and time.

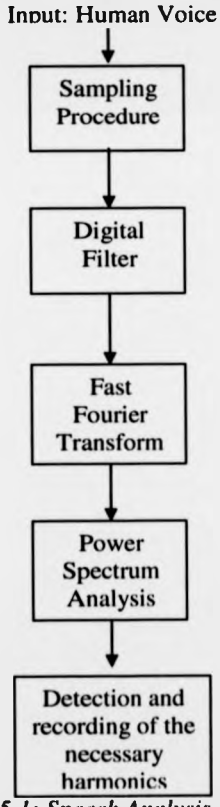


Figure 5.1: Speech Analysis Phases

5.2 Collecting the voice samples

Prior to any voice processing, samples of voice must clearly be collected. Although, there are many advanced devices that provide high fidelity voice recording, their employment was intentionally avoided in an attempt to broaden the horizons of this research work. The final outcome should be independent of the high performance standards of any expensive apparatus, other than the research algorithm's. Therefore, the device that collected the voice samples was a standard personal computer microphone. It was connected directly to the soundcard of the system and placed at a distance of twenty to thirty centimetres from the speaker's mouth.

The environmental conditions wherein all the experimental recordings occurred, were not consistent. Two different background environments were used, so that extended testing procedures could be deployed, in later stages, to assess the performance of the biometric algorithm. A quiet office environment was ideal for minimum background noise speech recordings. The signal generated and the recorded versions of it were very similar. The lack of noise interference enabled an unaltered digitisation of the analogue signal. Biometric solutions are very vulnerable to radical external condition changes. In particular, the presence of noise can create serious obstacles to the successful performance of any speech application. The second sets of recordings were intended to investigate the resilience of the research algorithm in such hostile conditions. The voice was recorded in outdoor environment, where the presence of random sound signals (like the wind) was apparent. These signals have to be isolated from the speech signal and eventually removed, leaving the human voice intact.

Twenty-five people were involved in this research work, providing two thousand voice samples. The participants were of both genders aged between twenty-one and forty five years old. Their mother tongue was English with the exception of two people who were not native speakers. Each person produced a set of forty voice recordings indoor and another set of the same number of recordings outdoors. Every speaker was instructed to repeat a certain pass phrase, unique for every user, ten times at a normal conversation speed. The same pass phrase would be repeated again at a slower and a faster speed ten times for each different pace. The collection of thirty voice recordings generated by the same human voice saying the same pass phrase at three different speed levels, enabled the extended analysis of the speech signal in relation with its time duration. Moreover, each human participant generated ten more voice print outs by repeating a different pass phrase to the initial one, which was common for all the speakers. The latter enabled the investigation of human voice characteristics of a signal conveying the same information message. A thorough discussion on the experiment's construction and its contribution to the extraction of important conclusions is presented in Chapter 8.

5.3 The digital signal

The first research task was the analysis of the human voice, generated by repeating a specific pass phrase. The most common representation of a voice signal is a two dimensional graph, where the x axis represents time in seconds and the y axis represents the amplitude of the signal. It has to be emphasised that in the time domain representation of the voice signals, time is in number of samples, with a sampling rate of 11.025 samples/sec.

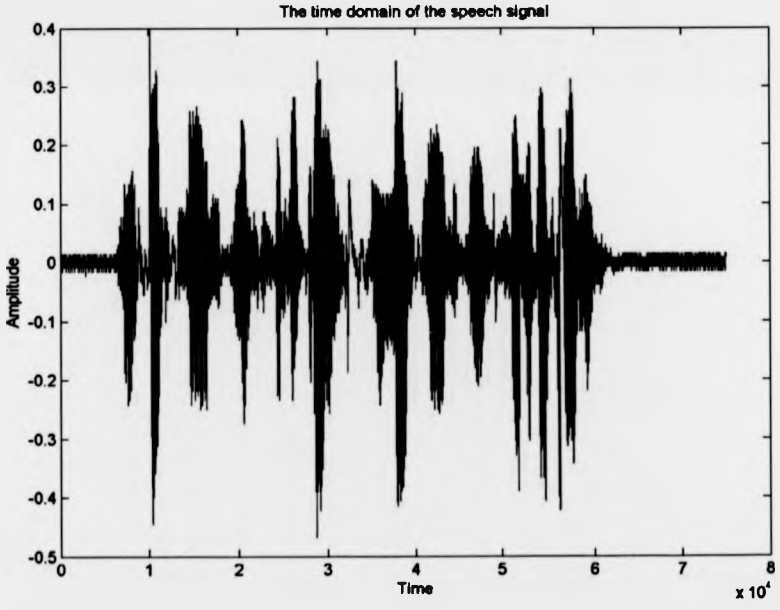


Figure 5.2: The time domain of the speech signal

The above Figure depicts a male's voice signal, as it was recorded in indoor environment. It does not require a lot of expertise in order to extract some safe

conclusions from merely observing the graph. A speech engineer can determine the number of words the pass phrase consists of, in this case ten, as they are clearly separated by short periods wherein the amplitude of the signal is approximately zero. Another interesting information is the presence of white noise throughout the entire voice recording. White noise is distinctive by its flat spectral shape wherein all frequencies have equal power. It is assumed to be generated when air passes through a constriction. It is very difficult to identify and separate white noise, as any other type of noise, from the original signal. However, in Figure 5.2, white noise can be clearly observed in time instances where the original voice signal is not present, especially during the first second and the last one and a half seconds of the recording.

There are many different methods of minimising the effect of noise in voice signals [64]. According to an oversimplified noise extraction technique, the amplitude of the voice signal is much greater than the amplitude of the noise signal. The noise signal is analysed so that its maximum amplitude (NAmax) can be defined. Therefore, all the low amplitudes of the voice signal (from zero up to NAmax) are treated as noise generated and are removed [65]. This approach cannot be applicable in this research work. It may be effective in areas where only background noise exists, but its performance degrades in areas where both noise and original voice signal are present. The alterations it causes to the recorded human voice, by removing all the low amplitude signals, condemn any attempt to identify unique speech characteristics. The most efficient method of minimising the effect background noise

has on the voice signal is by filtering. Discussion of the filtering employed in this research work is postponed until section 5.6 to contain discussion of the analysis of the human voice.

In an attempt to identify human voice characteristics, all the voice signals generated by the same individual repeating a standard pass phrase were compared. In this case, the context of the message is the same and the signals are expected to be identical because they have been generated by the same source.

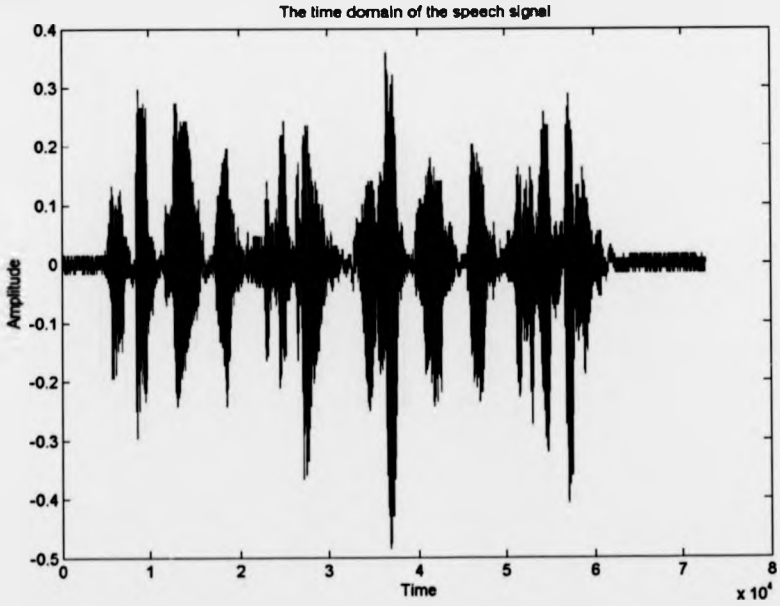


Figure 5.3: The time domain of the speech signal generated by the same individual containing the same information as the signal in Figure 5.2

Figure 5.3 presents a voice signal by the same male as in Figure 5.2, conveying the same information message. There are many similarities including the number of words the pass phrase is consisted of, the duration of the signal, the amount of time gaps between the words and some amplitude characteristics.

It is interesting to compare the previous voice signals with a third one generated by another speaker saying the same pass phrase. Until now the source of the signal was the same and therefore it was not possible to make any deductions concerning the level of dependence of the voice signal on its excitation source. Speech can be metaphorically presented as an equation with many variables, one of which is its production. During the production phase physical characteristics of the human being, such as the vocal tract, lungs, trachea and so on., heavily characterise the pitch of the generated sound. Therefore, a comparison between two speech signals conveying the same message (common pass phrase), but generated by different individuals can help in identifying those elements of speech responsible for the uniqueness of each human voice.

The following figure (5.4) presents this third voice signal conveying exactly the same information as the previous two but generated by a different excitation source. The time domain representation of the speech signal is rather poor in generating useful conclusions about signal's characteristics. This is a result of the hidden frequency information that cannot be observed in the time domain. However, even in this case the third signal has significant differences from the previous two.

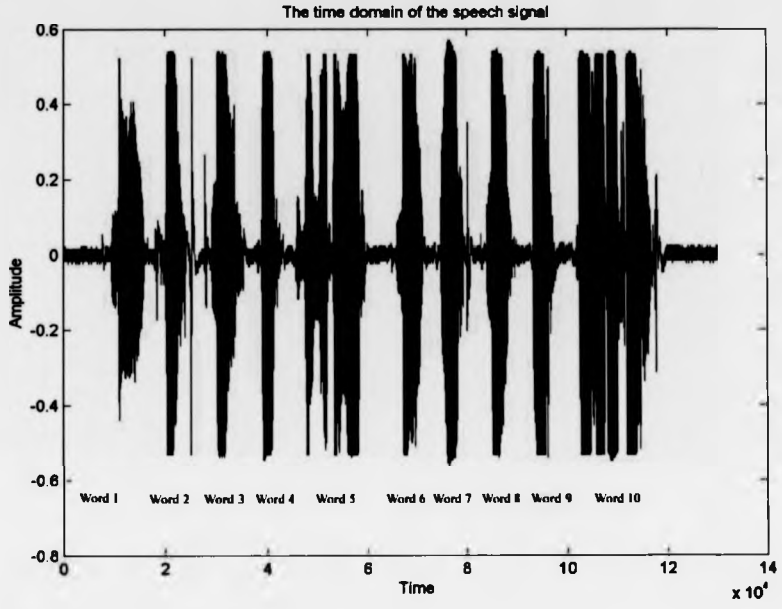


Figure 5.4: The time domain of the speech signal with the same message but different excitation source.

The second person speaks more clearly, as illustrated in Figure 5.4 by the wider separation gap between the words. Hence, the voice signal has twice the time duration of the previous two voice signals. Additionally, the amplitude of the signal does not vary as much as the previously ones. Almost all the words reach the same amplitude level and therefore the sound of the speech is expected to be unrelated with the sound generated by the first speaker.

Nevertheless, the signal in Figure 5.4 has some similarities with the signals in Figures 5.2 and 5.3. The number of words, ten, is the same in all three pass phrases. The word 5 and word 10 are longer than the others and it can also be stated that words 4 and 9 are the same because their time domain representation looks identical. At this point it has become apparent that the time domain comparison of human voice samples can produce some valuable information about the context of the pass phrase. However for an exhaustive analysis, a frequency domain representation is also necessary.

At this point it is important to define the research focus of this work. The uniqueness of human voice has been employed by many biometric systems in terms of pattern matching [66-67]. The latest development in hardware technologies has enabled the rapid evolvement of biometric applications and their employment in every day activities. However, all the voice biometric systems employ voice as a security measure that increases the overall confidence of the system whether the user is legitimate as he or she claims to be. In other words, the voice biometric system produces a binary answer (Yes or No), based on *pattern matching* techniques. Once this answer is given the behavioural biometric data is not important for the system. Additionally, the uniqueness of human voice has not been identified but its existence assumed throughout the entire length of the voice signal.

A new approach to voice biometric systems is introduced in this research work. The uniqueness of human voice is no longer regarded as a special characteristic of

speech that helps in producing specific voice patterns. On the contrary, it is considered as extra information that the speaker sends to the listener, which does not obey the dictionary and grammar rules but is specially designed and therefore only perceived from the auditory system.

Many years of research have indicated that the range of human hearing is generally considered to be 20 Hz to 20 kHz, but it is far more sensitive to sounds between 1 kHz and 4 kHz [50]. There are only about 120 levels of loudness that can be perceived from the faintest whisper to the loudest thunder. Listeners can tell that two tones are different if their frequencies differ by more than about 0.3% at 3 kHz. This increases to 3% at 100Hz. Consequently, the sensitivity of the human hearing system to sounds between 1 kHz and 4 kHz constitutes a strong indication that a number of frequencies within the speech signal may have more importance than some others. The latter enables this research to focus on a reduced part of the signal, a development very convenient for minimising the processing time of the entire biometric application. The analysis of the speech signal based on groups of frequencies will be presented in the following sections.

5.4 Spectrogram Analysis

Introduced in Chapter 3, a spectrogram is a three dimensional plot of the variations of the human voice signals with time on the horizontal axis, frequency on the vertical axis, and the darkness of the pattern being proportional to the signal energy. This device has been employed throughout this research work to increase the certainty of important digital signal processing decisions made in various phases and to verify a number of critical research results. The high processing requirements of this signal representation method appoint it as a supplementary research tool that can be used only in a laboratory environment as an alternate to the final research approach.

In the previous section the presentation in the time domain of three speech signals, a small portion of the two thousand voice signals recorded for research purposes, was aiming to outline the main characteristics of human voice. Those signals will also be used in this section to relate the time, frequency and energy of human speech in a single graph. The first human voice, the time domain appearance of which is illustrated in Figure 5.2, produced the following spectrograph. All the spectrograms presented in this thesis use seconds (sec) for the time representation and Hertz (Hz) for the frequency representation.

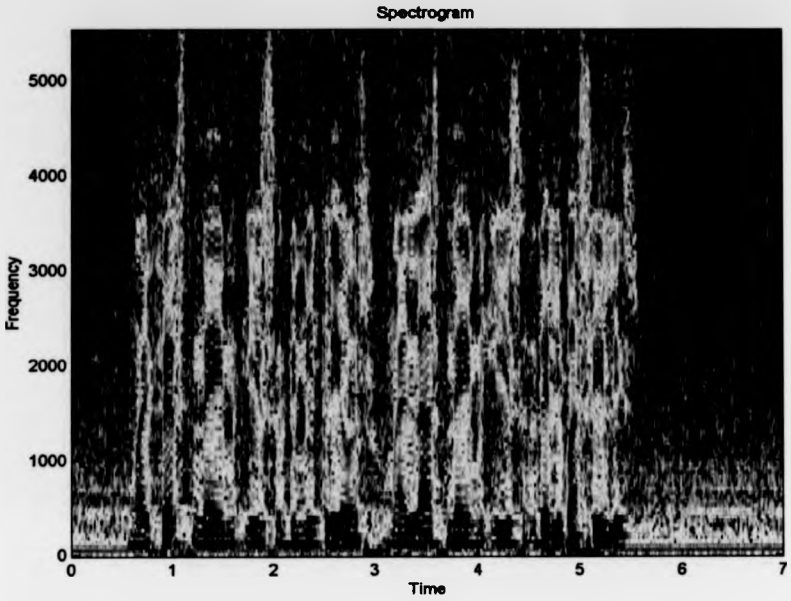


Figure 5.5: The spectrogram of the voice signal presented in Figure 5.2

There are two types of spectrograms, the wideband and the narrowband, depending on the band of frequencies chosen for analysis at a particular time instant. The former achieves a good time resolution at the expense of the frequency resolution, whereas, the latter performs better with frequency variation rather than with time. The dark horizontal bands in the spectrogram illustrate peaks in the spectrum, such as formant frequencies. The noise like excitation of unvoiced sounds appears as rectangular dark patterns with random occurrences of light spots due to sudden variation in energy. The vertical striations observed in the spectrograph represent the voiced speech, generated by the periodic glottal excitations.

The spectrograph of the human voice, the time domain of which is presented in Figure 5.3, is depicted in Figure 5.6.

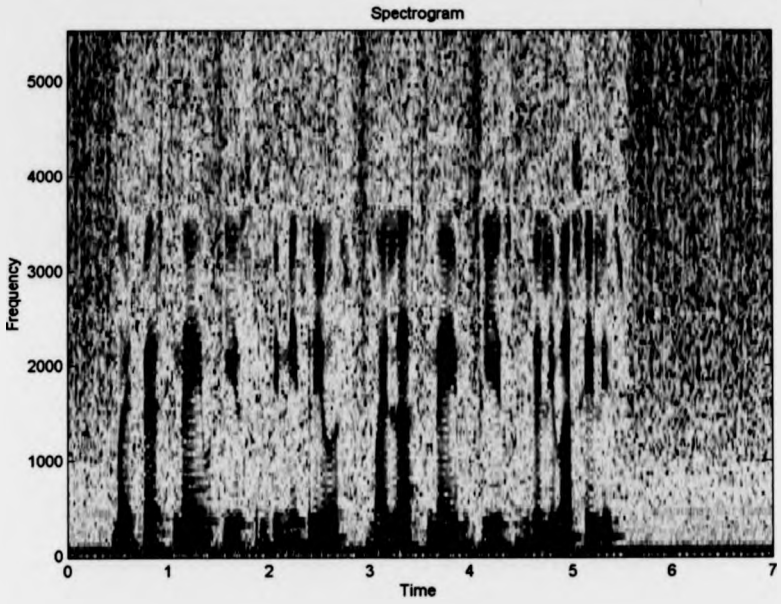


Figure 5.6: The spectrogram of the voice signal presented in Figure 5.3

In spite of the fact that the human speech analysed in Figures 5.5 and 5.6 is generated by the same person repeating the same pass phrase, the two graphs are not identical. The darker colours used in Figure 5.6 indicate that the energy of the second human speech is greater than the first one's. This is a quite common phenomenon, observed many times during the analysis of experiment's voice

samples, and occurs whenever the speaker's distance to the recording device alters, or the volume of his or her voice increases. Nevertheless the two graphs have many similarities that can assure a voice expert that these signals have common origin and transfer the same amount of information.

The latter becomes apparent, if the previous two graphs are compared with the one presented in Figure 5.7. This time, as it is already known, the human voice originated from a second person, using the same pass phrase that the first human speaker did.

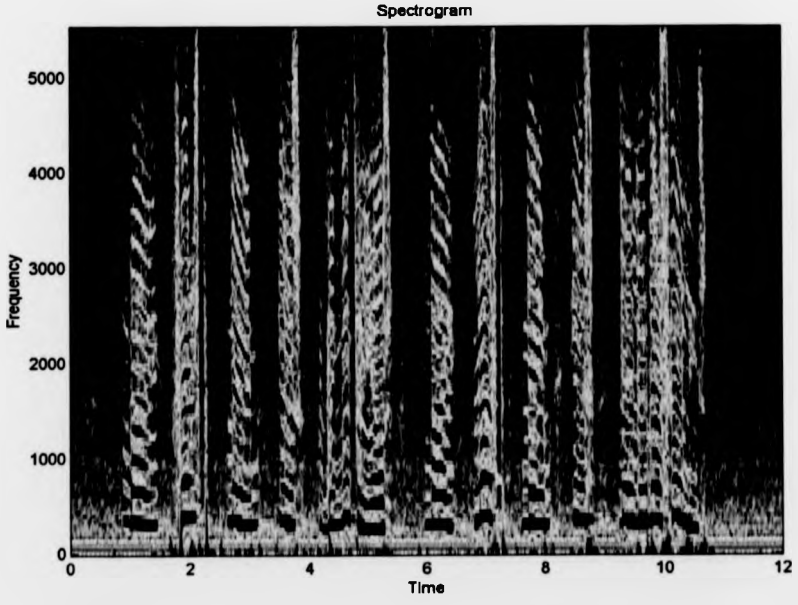


Figure 5.7: The spectrogram of the voice signal presented in Figure 5.4

The fact that the context of the pass phrase is common for all the three spectrographs is not sufficient to allow the third speech signal to masquerade as the other three. Even an inexperienced voice engineer can detect the difference, especially presented with the employment of a spectrograph. The only similarities detected between the first two speech signals and the third one is that the low frequencies, from zero to 500 Hz, contain the majority of the signal's energy and secondly, they reach high frequencies, from 4 kHz to 5kHz, at common words of the pass phrase. The latter enables a pattern to be extracted, according to which the speech signal can be simulated. Additionally, the analysis of the waveforms voice signal can reveal part of the context of the pass phrase. In the particular example presented above, there is the strong belief that there is replication of specific words within the context of the message transmitted in the form of speech communication. It is out of the scope of this research work to try to understand the exact meaning of these words. The importance is not the meaning of the various pass phrases but how their waveforms can be processed and analysed to eventually gain cryptographic value. Nevertheless, even if there was the intention of interpreting speech signal to plain English text, the entire research work would not currently be feasible in wireless communication, due to the limiting processing capabilities of handheld devices.

5.5 Investigating the uniqueness of the human voice

The previous two sections have introduced the main principles and methodologies of human voice analysis that were employed in this research work. All the voice recordings were analysed in the same manner and compared based on some test case scenarios. In order to identify the uniqueness of human voice within the speech signal four categories of speech comparison were created, as presented below:

1) Same Speaker – Same Pass Phrase
2) Same Speaker – Different Pass Phrase
3) Different Speaker – Same Pass Phrase
4) Different Speaker – Different Pass Phrase

5.5.1 Same Speaker – Same Pass Phrase

In the first category, all the voice recordings generated by a person repeating a standard pass phrase, were analysed and compared. This scenario aimed to identify the human behaviour, and record any inconsistencies in the signal waveform that may affect the research algorithm. It is important to familiarise with signal's response to external factors other than the identity of the speaker. The waveforms generated are closely related not only with the uniqueness of the speaker's voice but also with other characteristics of speech, like speed and volume. Moreover, it has been showed that speech is affected by psychological factors, explaining the employment of voice in the truth tests, used as a legal tool in court of law. The latter justifies the classification of voice as a behavioural biometric. It does not record only a physical trait but a behaviour associated with a human being. As a behavioural

characteristic, voice is more a reflection of individual's psychological makeup, although general physical traits, such as size and sex, have a major influence. The context of the pass phrase and its effect on the waveforms generated was not an issue in this category because it intentionally remained unaltered.

Theoretically it was expected that all the waveforms in this category would be identical. Unfortunately, real life recordings proved otherwise although, it was never in doubt that the signals were generated by one person. The minor changes in signals may be related to certain modifications in human's behaviour.

The most obvious inconsistency was the time duration, and it was rather unusual for two signals to have the same time response. Speech can be treated as periodic only for a short duration, because for a longer interval it varies randomly. Therefore, three extra testing scenarios were created for this first category, aiming to investigate how the speed of voice may affect its characteristics.

1.1) Same Speaker – Same Pass Phrase – Normal Speed

1.2) Same Speaker – Same Pass Phrase – Slow Speed

1.3) Same Speaker – Same Pass Phrase – Fast Speed

Every speaker involved in the experimental phase of the research, produced a set of ten recordings for each different speech pace. The following three waveforms illustrate the implication of the time factor to human voice communication.

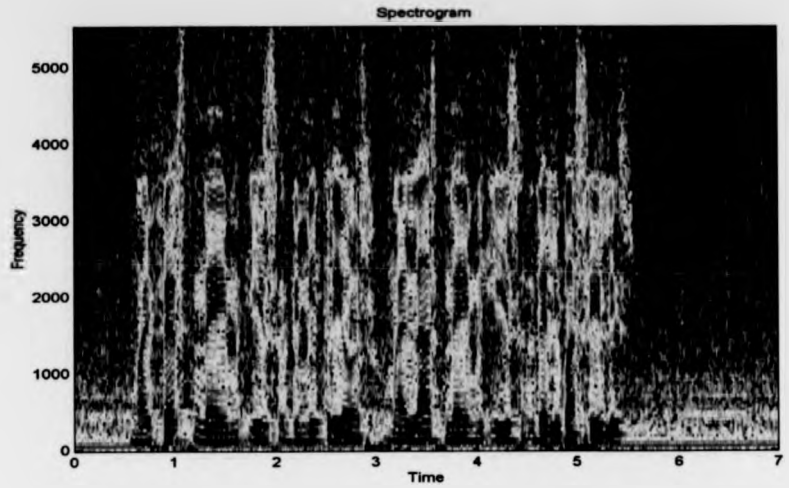


Figure 5.8: The spectrogram of the voice signal in normal speed

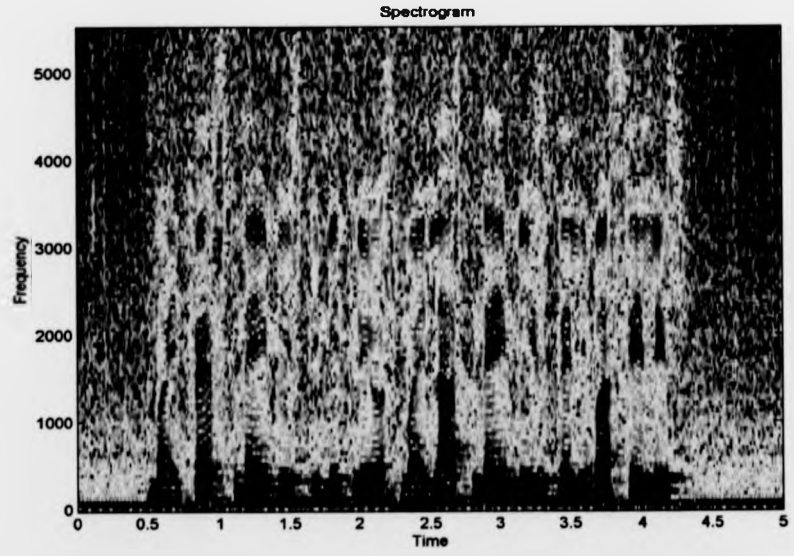


Figure 5.9: The spectrogram of the voice signal in fast speed

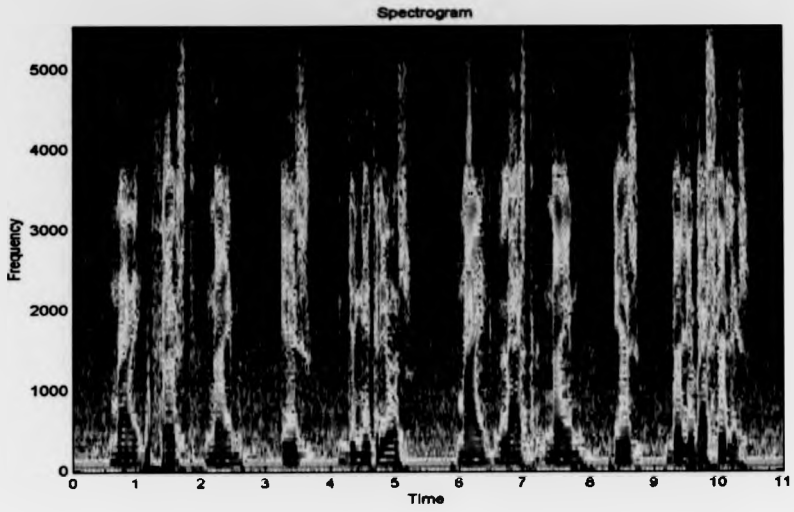


Figure 5.10: The spectrogram of the voice signal in slow speed

The Figure 5.8 is identical with the spectrogram in Figure 5.5, whereas Figures 5.9 and 5.10 consist of a faster and slower representation of the same voice signal respectively. The time variation causes a shift of the speech characteristics, frequencies, on the x-axis. Especially, when the sentence is spoken fast the signal's waveform becomes more complex due to the limited time duration. Consequently, it is expected that the performance of any speech characteristic extraction algorithm may be seriously affected.

The substantially random duration of the speech recordings did not allow the research algorithm to be time dependent. The focus had to be the frequencies of the different components comprising the speech signal. However, it has to be stated that

the periodic nature of speech over short time intervals can be employed to simplify problems occurring due to the overall non stationary behaviour over time. The voice signal is going to be segmented into small fractions, wherein it can be characterised as periodic, in order to enable the employment of techniques and methodologies that will eventually ease its analysis.

The extended analysis of all the human voice recordings, based on this first testing condition, has indicated a potential obstacle for the successful completion of this research work. This is the inability of the speaker to produce an identical analogue speech signal by repeating the same pass phrase. Therefore, the signal's digital transformation also varies, preventing the algorithm from identifying a legitimate speaker.

The latter is a common characteristic of voice biometric systems, as has been discussed in Chapter 2. The algorithm should be able to tolerate the small variations of the human voice. This can be achieved if the architect of the biometric system designs an "intelligent" routine that enables the system to cope with minor physical variations of a person's speech. During the design phase, the diagram presented in Figure 2.11 enables the architect to choose the attributes of the biometric system. The trade off between usability and security should be adjusted so that the system meets the application's requirements. A thorough discussion of the tolerance algorithm, deployed in this research work, can be found in the sections to come.

5.5.2 Same Speaker – Different Pass Phrase

This category of speech recordings investigated the relationship between the context of the spoken sentence and the generated waveform. It is expected that the output of the biometric algorithm will be closely related to the context of the pass phrase. The following Figures illustrate the speech of a person, X1, speaking three different sentences S1, S2, S3 respectively.

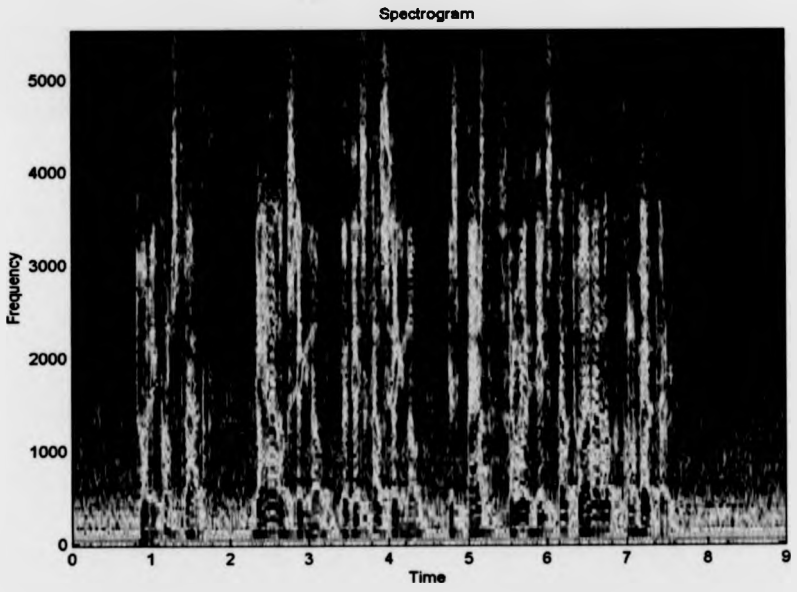


Figure 5.11: The spectrogram of the voice signal of person X1 speaking the sentence S1

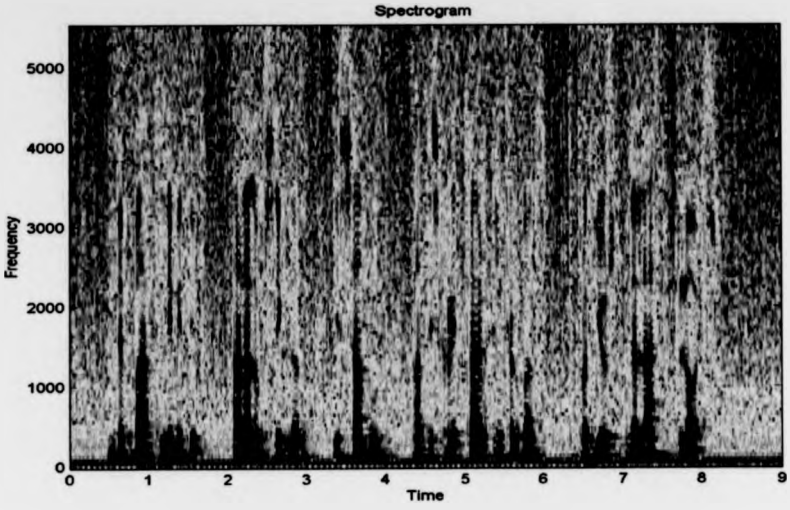


Figure 5.12: The spectrogram of the voice signal of person X1 speaking the sentence S2

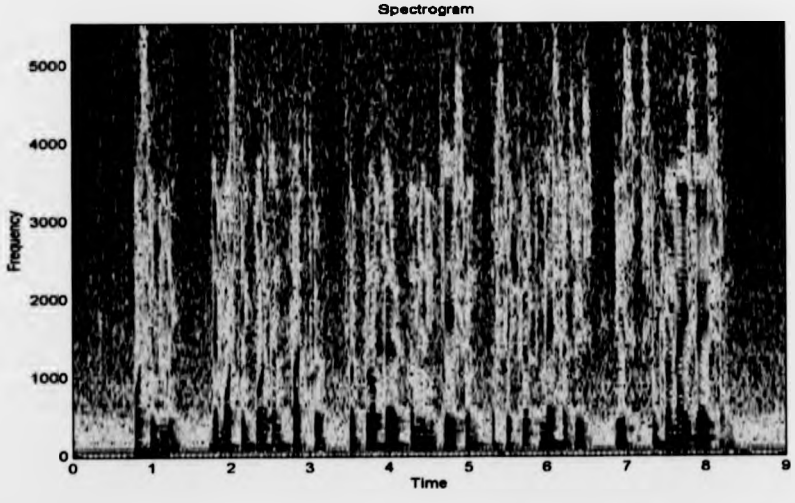


Figure 5.13: The spectrogram of the voice signal of person X1 speaking the sentence S3

The previous three spectrograms emphasise the impact the context of the pass phrase has on the generation of the voice signal, as there is no close resemblance between them. Evidently, there are some similarities, which can be reasoned due to the existence of common letters within the three spoken sentences S1, S2 and S3. The entropy of the English language, guarantees the presence of certain letters in a frequent manner, within any syntactically correct sentence. The entropy is expressed in terms of probabilities involved. The relative entropy of the source derives from the ratio of the actual to the maximum entropy with the remainder forming the redundancy.

Redundancy is the fraction of the structure of the message, which is determined not by the free choice of the sender, but rather by the accepted statistical rules governing the use of the symbols in question. It is most interesting to note that the redundancy of English is approximately fifty per cent, so that about half of the letters or words chosen in writing or speaking are controlled by the statistical structure of the language [68].

The dependency the speech has on the context information it contains, is going to be used as an advantage, enabling the speaker to alter the output of the biometric system according to his/her will. This means that the biometric system will be able to assign different biometric signatures to a single user, substantially increasing the defences of such a system as explained in Chapter 8.

5.5.3 Different Speaker – Same Pass Phrase

The purpose of existence of this third category was to investigate the possibility that two different speakers can produce the same voice characteristics by repeating a standard pass phrase. Moreover, it demonstrated the relationship between the voice signature and the uniqueness of human voice. The only substantial difference between the speech signals in the generation phase, is the excitation source. Therefore, any variation observed in the following Figures is directly associated with the uniqueness of the specific human voice sample.

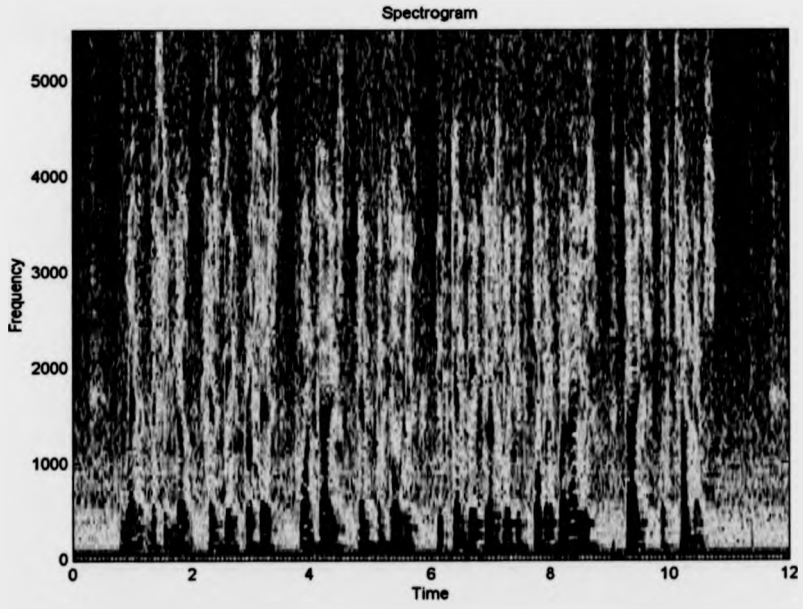


Figure 5.14: The spectrogram of the voice signal of person X1 speaking a sentence

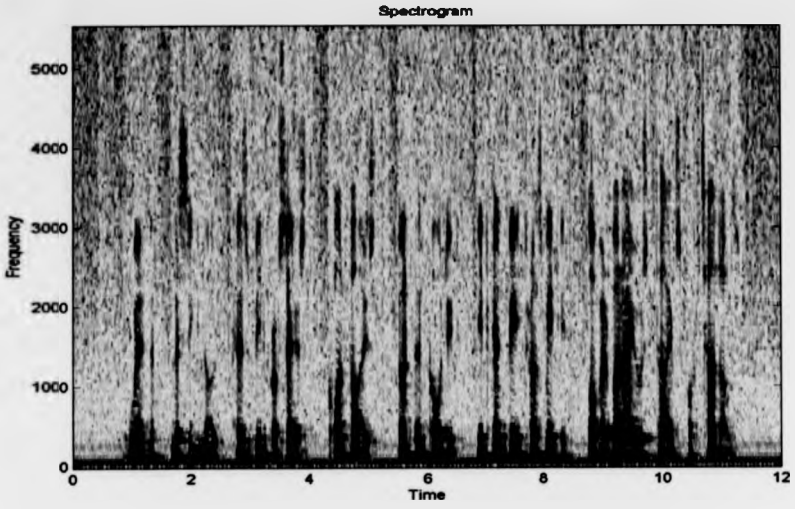


Figure 5.15: The spectrogram of the voice signal of person X2 speaking a sentence

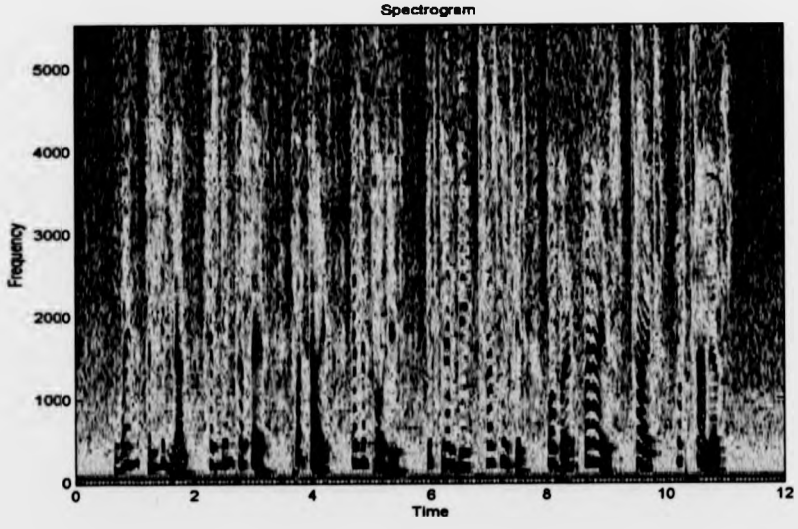


Figure 5.16: The spectrogram of the voice signal of person X3 speaking a sentence

The above three figures exemplify the impact the uniqueness of human voice can have to a signal. All the environmental conditions during the recording of the voice samples presented in this category, intentionally remained unaltered. As a result, the speech production mechanism, presented in Chapter 3, is all that is responsible for the production of the specific spectrograms. It has to be emphasised that the selection of the three voice samples was not random. These are the most similar speech signals out of a pool of two thousand samples.

There is little value in trying to identify differences between the signals presented in the Figures 5.14 – 5.16. It is a logical consequence that they vary significantly, obeying fundamental rules of digital speech processing. However, the real value is hidden in the similarities between the speech signals. In theory, the human voice generated by a person is heavily characterised by the parts of the person's body involved in the generation process. This dependency does not allow two voice signals generated by two different individuals to be exactly the same. It is also known though, that the human voice in general has certain characteristics common for any human being. The latter is dictated by limitations of the human auditory system, the range of which is generally considered to be 20 Hz to 20 kHz, as mentioned before.

In other words, the human voice, irrelevant of the origin of the source, has to follow some general rules. This means that a specific part of every speech signal is dedicated to produce the common characteristics of the human voice. The latter can

be better understood with a metaphor between speech communication and computer communication. For two computers to be able to communicate (hence exchange data packets) there has to be a common protocol in place. This "protocol" in speech communication is the common characteristics of the human voice, expected to be listened from the communicating party. These characteristics enable human beings to exchange information. According to this approach, the speech signal conveys information about the context of the spoken sentence and the source of voice generation. Nevertheless, all this information is not equally distributed within the signal. A significant amount of the signal's data is redundant and not necessary for speech recognition. Identifying similarities within two signals generated by a different source can prove the existence of considerable redundancy.

The extensive investigation of voice signals' similarities, in this category of testing, has indicated that the majority of signals' common elements can be found within the range of 20 Hz to 800 Hz. The latter can be justified by observing all the spectrograms presented in this section. Most of the speech signal's energy is "stored" in frequencies within that range. All the speech signals, recorded for the purposes of this research, have the same energy level in average, and therefore the energy measurements do not carry information that can be used for speech recognition.

The latter, in combination with the fact that the human hearing is far more sensitive to sounds between 1 kHz and 3 kHz [38, 64] have made the focus of the research to

be frequencies greater than 1 kHz. Another phase of investigation then started aiming to analyse the behaviour of frequencies lying within the specified range. An exhaustive analysis of the voice samples indicated that frequencies between 1 kHz and 3 kHz are very difficult to be produced by two different individuals even in the case the pass phrase is known. It is vital to emphatically state that there was not a complete match between two voice samples.

The unique attributes identified in frequencies between 1 kHz and 3 kHz constituted a significant breakthrough in the overall progress of the research. It enabled this range of frequencies to be isolated from the entire voice signal for processing. This approach conserves the processing power of the system, the importance of which is emphasised in several parts of this thesis.

In an attempt to investigate further the attributes of frequencies between 1 kHz and 3 kHz the previous two testing conditions, described in sections 5.5.1 and 5.5.2, have been reviewed again. This time the comparison of voice samples generated by the same person repeating the same pass phrase produced some very useful conclusions. The frequencies between 2.2 kHz and 3 kHz are very inconsistent between the numerous voice recordings. The speaker is incapable of producing the exact same set of frequencies and therefore the reproducibility of a standard biometric signature is not of an acceptable level (above eighty per cent). The spectrogram analysis empirically showed the same voice behaviour for all the speakers participated in the experiment.

Consequently, the subset of frequencies between 1 kHz and 2 kHz is the most appropriate for this research work. The employment of a band pass digital filter is necessary to remove the undesired frequencies and to minimise the collected noise signal. The analysis of the filtering process is described in the section 5.6.

5.5.4 Different Speaker – Different Pass Phrase

The last category of testing the attributes of the recorded voice samples, aimed in investigating whether two different speakers repeating different pass phrases can generate speech signals that have many common elements. The result of this special condition testing is decisive for the resistance level of the research biometric system against an impostor's attack. If the intruder is able to generate a valid biometric print out by speaking a random sentence then the security of the system is at stake, and radical protection measurements should apply immediately.

As it is accustomed to all the previous testing cases of voice samples, the presentation of three speech signals illustrate the process. The number of participants, twenty-five, in this research work can be regarded as rather limited for purposes of this fourth category. On the other hand, though, the large number of voice recordings associated with each person involved in the experiment compensates that. The following Figures illustrate spectrograms produced by speakers X1, X2 and X3 repeating phrases S1, S2 and S3 respectively.

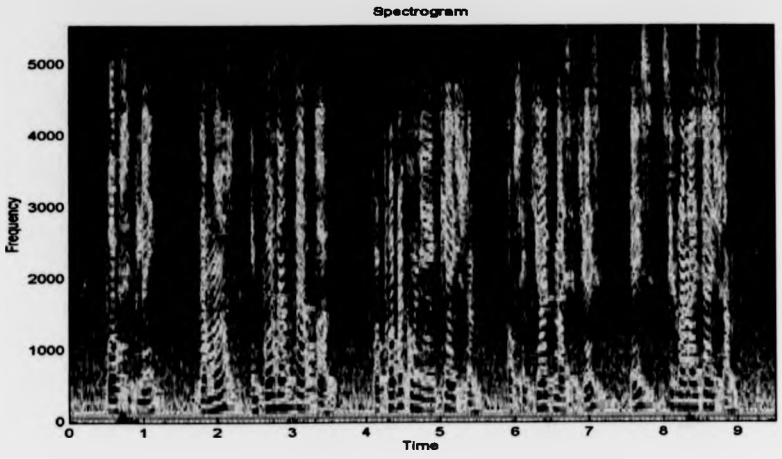


Figure 5.17: The spectrogram of the voice signal of person X1 speaking the sentence S1

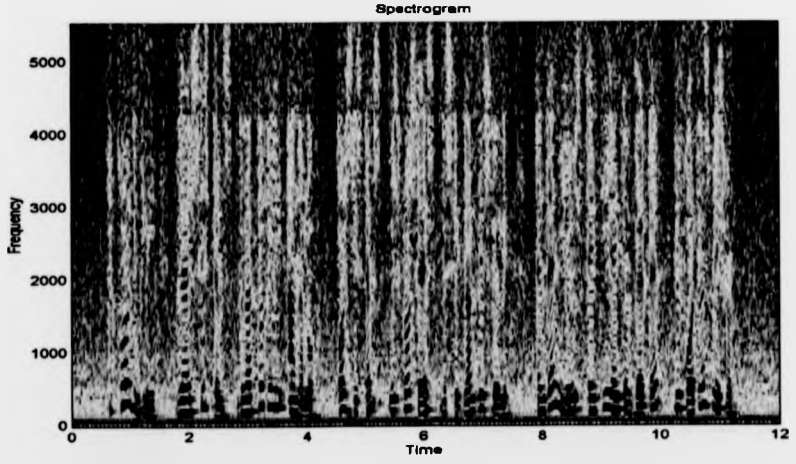


Figure 5.18: The spectrogram of the voice signal of person X2 speaking the sentence S2

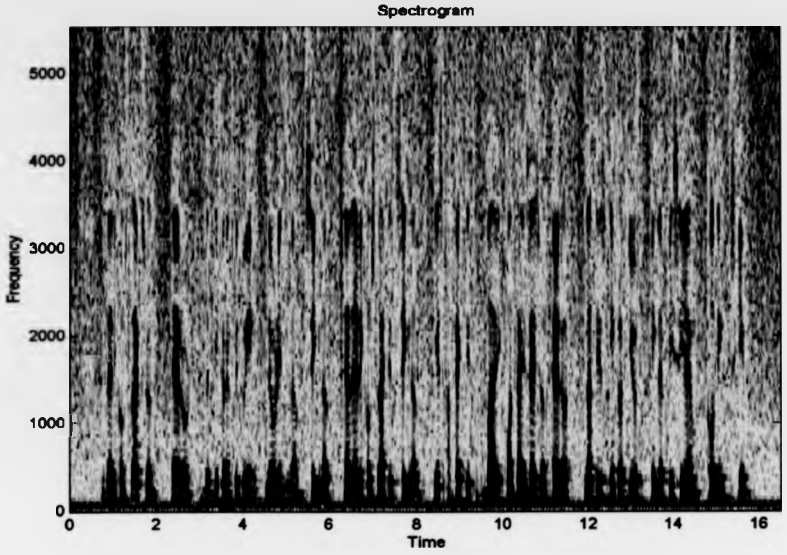


Figure 5.19: The spectrogram of the voice signal of person X3 speaking the sentence S3

As may be derived from the figures in this section, all the three voice signals consist of pass phrases with different numbers of words. The pitch of each person's voice varies significantly and this is the result of producing spectrograms where the different formant frequencies can be observed. Comparing these voice signals it can be stated that the first two look more alike than the third one. This can be explained by mentioning that speakers S1 and S2 have the same gender (females), whereas speaker S3 is a male. The analysis of the collected voice samples has showed that female speakers have formant frequencies fifteen per cent higher than males on

average, a result very close to the value of seventeen per cent, which is the theoretical expected [69].

As may be expected theoretically, the generation of a common biometric signature by two individuals repeating random phrases is highly unlikely. The dependency of the biometric output on the pitch of the human voice and the content of the spoken sentence does not allow this to happen. However, the latter statement has not been proved mathematically and is based on empirical testing. Therefore, in order to be mathematically correct it can be said that the generation of a common biometric signature by two or more individuals repeating random phrases is proved to be impossible, based on the research testing conditions and restricted only to experiment's data pool.

However, the large size of the data pool and the detailed analysis of the voice samples increase the confidence that the conclusions of this limited research work can be generalised and become applicable to speech signals recorded in an identical manner with the experiment's research data.

5.6 Filtering the speech signal

It is apparent from the research undertaken and described in the previous sections of this chapter, the employment of a digital filter is necessary. Detailed information about the types of digital filters and their specifications can be found in Chapter 3. The thorough study and analysis of digital filtering indicated that there is no real digital filter that can approximate an ideal performance [49]. However, the apparent need for digital filtering together with the limited processing capabilities of the intended application platform have resulted the selection of two different types of Infinite Impulse Response (IIR) filters. The first one is a Chebychev filter and its gain response is illustrated in Figure 5.20.

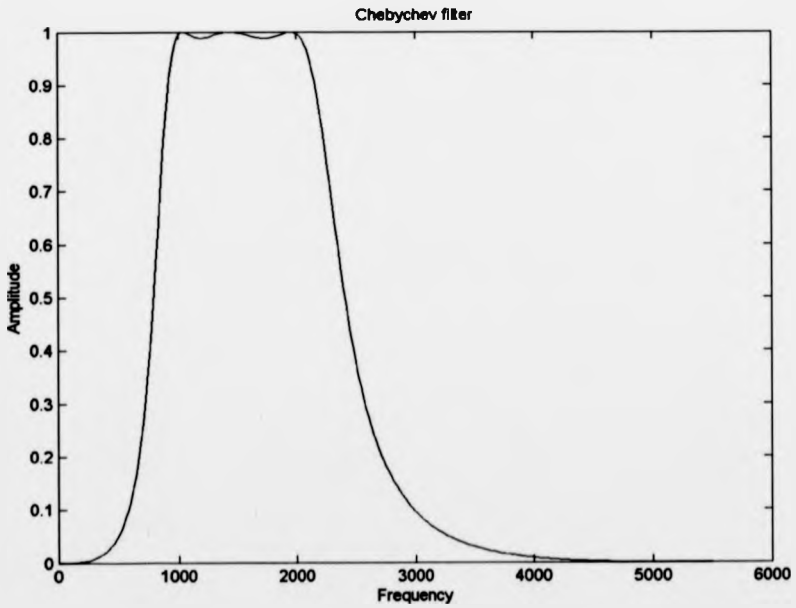


Figure 5.20: The Chebychev filter

It is a third order Chebychev filter. The 'order' of the filter, see also Chapter 3, is closely related with filter's complexity. It has been overemphasised throughout this research work that every proposed solution should be characterised by its simplicity, in an attempt to control the application process requirements. In the beginning, filters of first and second order have been implemented and tested. Nevertheless, their performance was not adequate. The third order Chebychev filter eliminated the presence of background and white noise significantly, thus chosen to be implemented in the research algorithm. Inevitably, higher order filters may perform better than the third order one. However, the decision is based on a trade off between performance and processing consumption. It can be observed from the above figure that there is a slight amplitude variation within filter's pass band; this ripple is the main disadvantage of Chebychev filters. It is compensated, however, with the rapid filter's response outside the pass band.

On the other hand, Butterworth filters have a flat gain within the range of accepted frequencies but respond slowly at cut off frequencies. The third order Butterworth filter designed and developed for this research work is presented in Figure 5.21. Its response obeys the general theoretical characteristics of the filter. It allows more non desired frequencies of the signal to pass to the system, than Chebychev filter does. Nevertheless, it has been choose as the most appropriate filter for this specific biometric system due to its performance within the pass band. The performance of any voice system is dependent on the ability of the speech signal to reproduce the

same characteristics any time it is repeated. External factors, like noise interference outlined in previous chapters, cause significant obstacles in voice reproducibility. As a logical consequence, techniques that may result in worsening the speech recognition of a signal, simply can not be tolerated. The filter's slow response in cut off frequencies is confronted with a simple routine in the algorithm that stops the system from processing with frequencies that do not belong within the range of 1 kHz to 2 kHz.

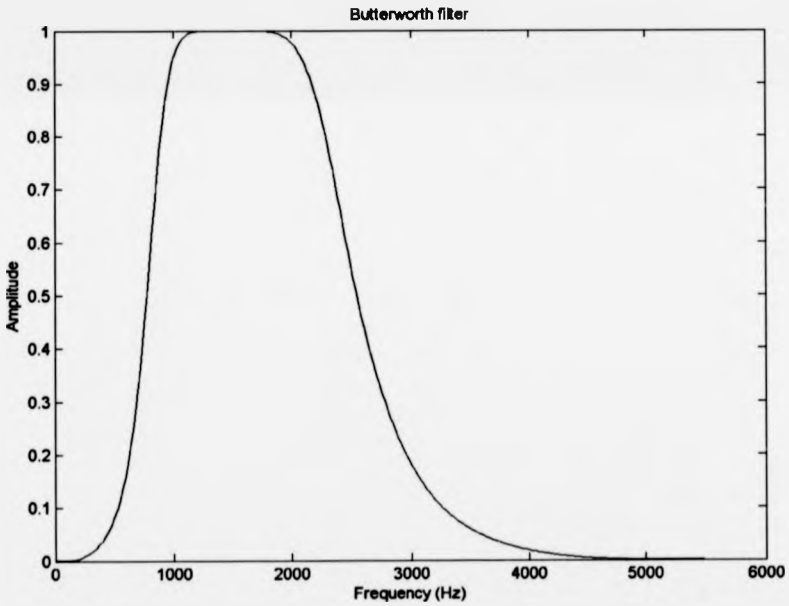


Figure 5.21: The Butterworth filter

All the speech signals are sampled at a rate of 11,025 Hz with a precision of eight bits. This sound quality has a data rate of 11,025 Hz * 8 bits = 88.2 k bits/sec. The high fidelity music systems sample fast enough (44.1 kHz) and with precision of sixteen bits, hence their data rate is 706 k bits/sec. The telecommunication systems operate with a sampling rate of 8 kHz, allowing natural sounding speech but greatly reduce music quality. The selected sampling rate (11,025 Hz) is greater than is the minimum sampling rate (4 kHz) recommended by the Nyquist theorem, for a system with a cut off frequency of 2 kHz. Thus, the aliasing of the speech signal is prevented.

The effectiveness of the digital filter implemented, can be seen by the two figures (5.22 & 5.23) below. The first represents the time domain representation of a speech signal and the second depicts the waveform of the same signal after the filtering process.

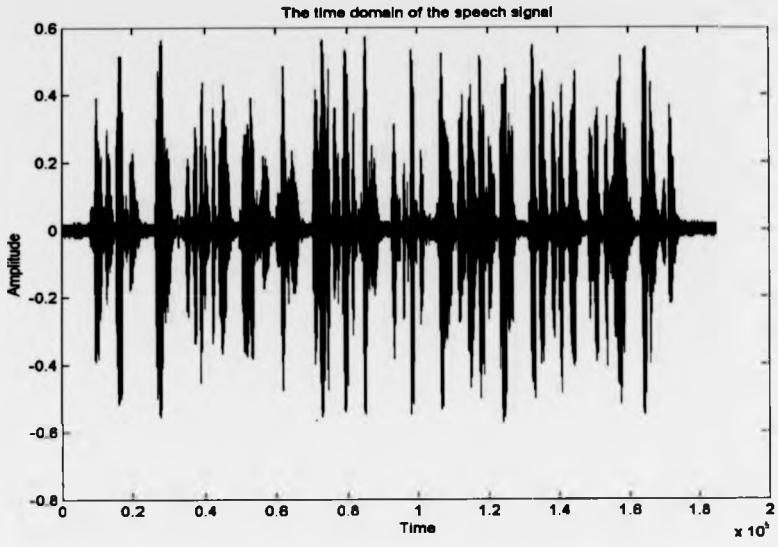


Figure 5.22: The time domain of a speech signal

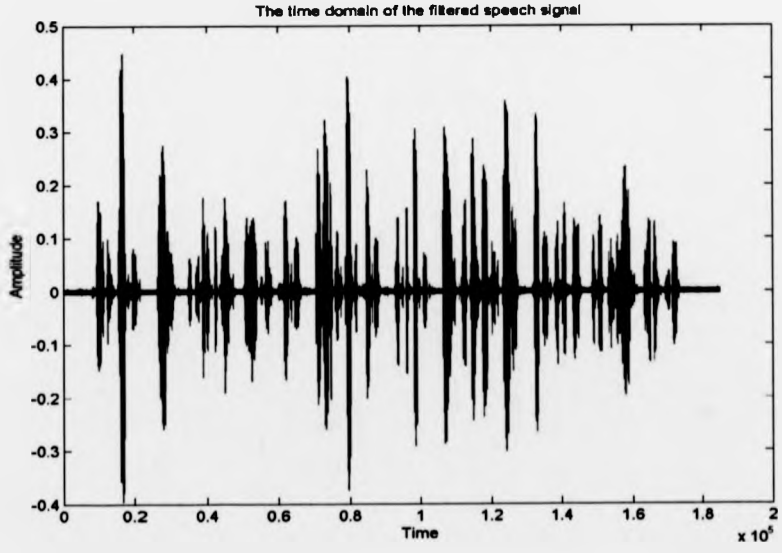


Figure 5.23: The time domain of the filtered speech signal

Background noise is present in both time domain representations of the signals but the digital filter minimises the interference of the noise significantly.

An advantage of the employment of the digital filter is that the words of the spoken sentence, in Figure 5.23, are clearly separated. This enables the detection of repeating patterns within the pass phrase, a very useful tool especially during the cryptographic stages. As has been mentioned before, the research analysis is not based on pattern matching techniques. However, alternative methodologies to human voice research, like pattern matching, have been used as a quality measurement. The results of the innovative technique described in this thesis, are compared with outcomes of well established digital signal processing and cryptographic methodologies. This comparison constitutes the best guarantee that the proposed research work is feasible and that its performance is of a high standard.

Chapter 6: Generation of Biometric Signature

6.1 Harmonic extraction

This section describes how the characteristics of the human voice, in terms of harmonics, were retrieved from the filtered speech signal. To allow the algorithm to perform this task, the speech signal must be transformed to the frequency domain. The spectrograms, presented in the previous sections, were used only in the laboratory to enable fine details of each voice recording to be studied. Unfortunately, they cannot be embedded in the algorithm because it would increase the processing requirements of the biometric system.

The Fast Fourier Transform (FFT) is a fundamental digital signal processing operation that helps in translating a function in the time domain into a function in the frequency domain. The FFT rearranges the Discrete Fourier Transform (DFT) calculations to make them computationally more efficient and is then suitable for application here.

The time domain representation of the graph in Figure 5.23 is translated by FFT and its frequency domain representation is illustrated in Figure 6.1, where the frequency is measured in Hertz (Hz).

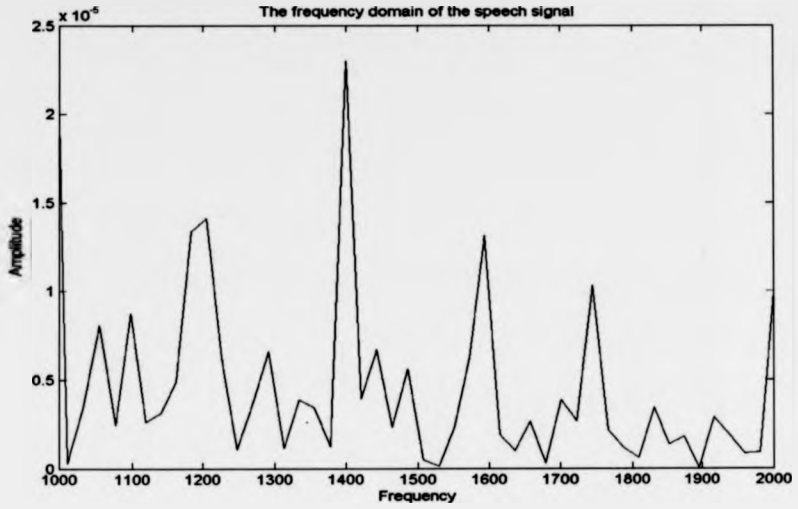


Figure 6.1: The frequency domain representation of the speech signal

The above Figure depicts the harmonics of the speech signal with frequency range between 1 kHz and 2 kHz. These harmonics provide the data the research algorithm needs to produce a 128 bit biometric signature. A procedure was implemented to read harmonics, which actually meant that the peaks were measured. At each specific peak of the harmonic, the power was calculated and stored.

It was known that the human voice was sampled with eight bit precision, where, each sample consists of eight bits. It was required that the algorithm should produce an output which has to be 128 bits. Consequently, the written procedure, the pseudo code of which is presented below, scanned the frequency domain of the speech signal and recorded the power of 16 different harmonics. Once the necessary set of

harmonics were recorded, they were assembled in a binary string placing the one after the other.

At this point the pseudo code of the procedure of the algorithm, responsible for the detection of the human voice harmonics, is presented. It covers the last box of the flow-chart produced (Figure 5.1).

```
Count = 0       // A counter used to track the number of the recorded harmonics
harmonicindicator = 0 // A variable used to indicate harmonic detection
Read sample
For (the entire length of the signal)
        Read the next_sample
        Compare sample with next_sample

        IF (next_sample < sample) & (harmonicindicator = 0)
                store sample as Amph(count)
                count = count + 1
                set harmonicindicator = 1
                sample = next_sample // Move to next sample

        ELSE IF (next_sample < sample) & (harmonicindicator = 1)
                set harmonicindicator = 1
                sample = next_sample

        ELSE
                set harmonicindicator = 0
                sample = next_sample
        END

END

IF (Count < 16)

        Calculate the number of harmonics needed for the biometric signature
```

// This is calculated by subtracting the value of Count from 16 and the result is //saved in variable NecessaryHarmonics

FOR (all the necessary harmonics)

Fill their stored values with zeros //Eight zeros for each added harmonic

END

END

The required number of harmonics, sixteen, is quite large and there is the danger that the algorithm may not find so many harmonics within 1 kHz to 2 kHz. In this case, the program pads with zeros so that the total length of the output is 128 bits. A statistical analysis of the voice recordings showed that every wav file had an average of 14.5 harmonics within the desired frequency range. The minimum of detected harmonics within a recorded sound file is 11, more than adequate for the generation of 128 bit signature.

Filling the biometric signature with zeros, so that the final length of the biometric output is reached, creates a potential security problem. Assuming that the eavesdropper attacks the biometric characteristics of the system, this means the output of the human voice analysis, he or she has to guess a string of 128 bits. The difficulty of performing successfully this task, without any knowledge of the biometric mechanism, is the same as trying to make 2^{128} combinations. Even with state of the art technology, these calculations significantly delay the attacker and make this type of attack not applicable. It is interesting enough to investigate what

happens if the intruder already knows one bit of the biometric output. This means that the length of unknown bits becomes 128-1=127 bits, and therefore the combinations to be made are 2^{127}. The difficulty of computing this biometric string is halved.

6.2 The biometric output and the key reproducibility

The proceeding sections in this chapter outlined the digital signal processing steps, employed to analyse the human voice. The biometric output is dependent on the harmonics produced by the speaker within the range of 1 kHz to 2 KHz. The performance of the algorithm is evaluated based on the testing criteria described in sections 5.5.1, 5.5.2, 5.5.3 and 5.5.4.

An important concern arose in the first testing condition, according to which a single speaker repeats a standard pass phrase. Theoretically, it was expected that this repetition would result into the regeneration of the same biometric output. In reality, though, the biometric signature was not repeated. Instead, many biometric signatures were produced indicating that the algorithm is not capable of handling the small variation of the speech signal between the various recordings. This weakness could not be tolerated because it was challenging one of the main assumptions of this research work, i.e. claim that the same speaker can produce the same voice characteristics by repeating a standard pass phrase.

The procedure responsible for harmonic extraction was reviewed thoroughly. The spectrograms of the voice recordings indicated that the frequencies within the desired range follow a repeatable pattern. However, their amplitude was not always consistent and small changes were recorded. These variations caused confusion to the algorithm because the programme was trying to find an exact match between the harmonics' amplitude, which never occur in real life conditions. Evidently, the algorithm had to be reviewed and strengthened, so that it could cope with this divergence.

A new box was created and connected with the last one of the flow chart in Figure 5.1, called "Detect and record the necessary harmonics". The name of the new procedure is "evaluate the extracted data" and its association with the existing procedures is illustrated in Figure 6.2.

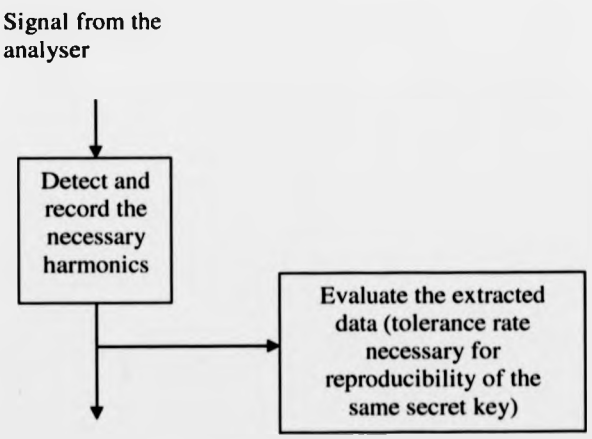


Figure 6.2: The reproducibility procedure

The new procedure performs a light weight analysis of the extracted data and implants intelligence in the algorithm. This intelligence is necessary to enable the algorithm to detect signals produced by the same speaker repeating the same sentence, even if there are minor variations. The pseudo code of the system is altered too as illustrated below.

```
Count = 0       // A counter used to track the number of the recorded harmonics
harmonicindicator = 0 // A variable used to indicate harmonic detection
previous_sample = 0 // A variable used to store the amplitude of detected harmonics
Read sample
For (the entire length of the signal)
        Read the next_sample
        Compare sample with next_sample

        IF (next_sample < sample) & (harmonicindicator = 0)
                store sample as h(count)
                store next_sample as h(count)R
                store previous_sample as h(count)L
                count = count + 1
                set harmonicindicator = 1
                previous_sample = sample // Move to next sample
                sample = next_sample

        ELSE IF (next_sample < sample) & (harmonicindicator = 1)
                set harmonicindicator = 1
                previous_sample = sample
                sample = next_sample

        ELSE
                set harmonicindicator = 0
                previous_sample = sample
                sample = next_sample
        END
END
```

As a result of this pseudocode a set of three recording variables is stored for each identified harmonic.

i.e.

```
h1L   h1   h1R   //First harmonic
h2L   h2   h2R   //Second harmonic
h3L   h3   h3R   //Third harmonic
```

The signal's behaviour close to each harmonic can now be observed and recorded. In order to achieve this, the following computations take place:

```
b1L = h1 – h1L    //Record the difference the harmonic has with
                  //the previous sample
b1R = h1 – h1R    //Record the difference the harmonic has with
                  //the next sample
```

Having finished with the analysis of the pass phrase, the system will prompt the user to enter his/her pass phrase once again. This is a standard procedure for any registration process, where the user has to enter twice his password to verify that he or she did type it correctly in the first place. Therefore the FOR loop described in the previous page is once again executed. An example of FOR loop's output is presented below:

i.e.

```
H1L   H1    H1R   //First harmonic
H2L   H2    H2R   //Second harmonic
H3L   H3    H3R   //Third harmonic
```

The behaviour of the second pass phrase will now be:

```
B1L = H1 – H1L     //Record the difference the harmonic has with
                   //the previous sample
B1R = H1 – H1R     //Record the difference the harmonic has with
                   //the next sample
```

Based on the analysis described above, the reproducibility pseudocode ends with the following FOR statement.

```
FOR (all the identified harmonics)     //Assuming that there are n identified
                                       //harmonics
    IF (hn ≠ Hn)
        Compare (bnL with BnL) and (b1R with B1R)
        // for the first comparison we have a variable called compresultL
        // for the second comparison we have a variable called compresultR
```

```
            IF ((compresultL< Tolerance %) AND (compresultR< Tolerance %))
                hn = Hn  //Same harmonic means same biometric key

            ELSE
                Prompt the user to enter the pass phrase again
            END

        END
END
```

The tolerance percentage can be set by the programmer and is closely related with the desired level of security of the application and the background environment. It is a trade off between the usability of the system and its security, Figure 2.11. The environmental noise interference in outdoor recordings distorts the signal more than it does in indoor recordings. The tolerance percentage has to be greater in the first instant than in the second.

The algorithm has to be able to detect whether the background environment, wherein the voice recording takes place, classifies as noisy or not. The latter is achieved by secretly activating the microphone few time instances before the user starts speaking. Thus, the microphone collects all the background signals except the human voice. The amplitude of the signals is analysed and depending on its value the tolerance percentage is set. There can be an unlimited number of levels of noisy

environments, but the more the system has the more storage space it requires, and the more processing power it consumes in handling the data. The system presented in this thesis, consists of three different noise environments resulting in 2, 4 and 6 per cent tolerance percentages. These values derived from an analysis of experiment's background noise signals and the impact they have on the speaker's voice. In other words these values can be modified to meet the needs of other recording environments, increasing the flexibility of the algorithm.

It is vital to mention that the pseudo code presented in section 6.1, scans the speech signal in an attempt to detect and record sixteen voice harmonics. On the other hand, the pseudo code presented in this section does not contain the same IF statement. The updated version of the procedure records a set of three values for every detected harmonic. It stores the value of the peak, harmonic, and the values of the previous and next sample. The program detects all the harmonics with frequencies between 1 kHz and 2 kHz and stores them. During the comparison of two voice recordings, the algorithm chooses eight harmonics with the closer match. Every selected harmonic consists of sixteen bit data. Eight bits are coming from the peak value sample and two sets of four least significant bits of the previous and next sample respectively (8+4+4).

6.3 Performance of the biometric algorithm

The modification of the algorithm introduced in the previous section, tackled the problem of the reproducibility of the biometric key. The system implemented was

run two thousand times, and the experiment's data pool of voice samples was thoroughly analysed. Every speech recording produced a set of graphs each one with unique research value. Most of the time, these were used manually for analytic research purposes and were not employed in the generation of the final output of the system. Nevertheless, they are of high importance because they provided the direction to achieve this thesis' target. This time consuming research phase produced results useful for the overall assessment of the written program. The evaluation of the system is based on the four categories, described in detail from section 5.5.1 to 5.5.4.

The first testing condition investigates the ability of the algorithm to reproduce the same biometric signature. Theory states that a sound, once made (even by the same individual) can never be exactly replicated in all its characteristics. Thus, the algorithm should be able to face this attribute of voice biometric systems. An acceptable performance should not fall below the 70 per cent, reproducibility, or else the system will not be applicable to a large scale deployment. Figure 6.3 summarised the results and as was expected the worst performance of the biometric algorithm occurred in outdoor recordings. The presence of intense environmental noise affected the system and the reproducibility of the same biometric key could not exceed the eighty per cent on average.

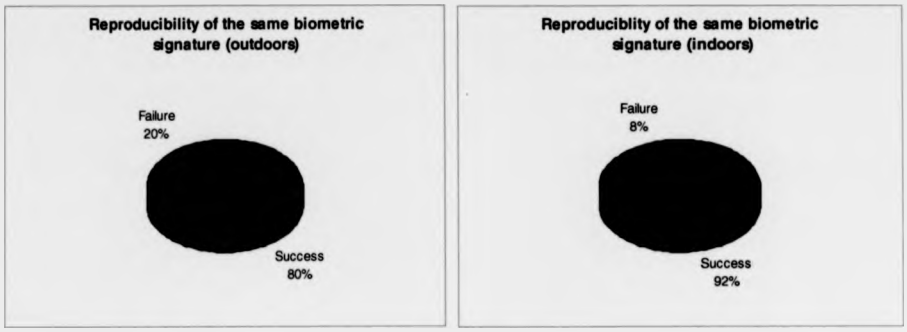


Figure 6.3: The reproducibility of the same biometric signature both outdoors and indoors

The algorithm produced better results when the human voice recorded in more friendly environmental conditions. A quiet office room proved to be adequate to increase the reproducibility of the biometric key by twelve per cent on average. The latter result indicates strongly that the research algorithm is capable of a larger scale deployment. Voice biometric systems can hardly exceed the ninety five per cent overall usability due to speech signal restrictions. Therefore, the research program satisfies the first and most difficult prerequisite.

The evaluation phase continued, testing whether the same individual, using two different pass phrases, may be in a position to produce the same voice characteristics, hence the same biometric output. As has been mentioned before, that during the implementation stages of the research algorithm it was found that the output of the biometric system was not only dependent on the pitch of the speech, but also on the context of the spoken sentence. The exhaustive analysis of the voice recordings, falling in the category investigated here, has shown that the same voice

characteristics were never extracted. The closer match had only thirty per cent (30%) common elements within the two speech signals (Figure 6.4).

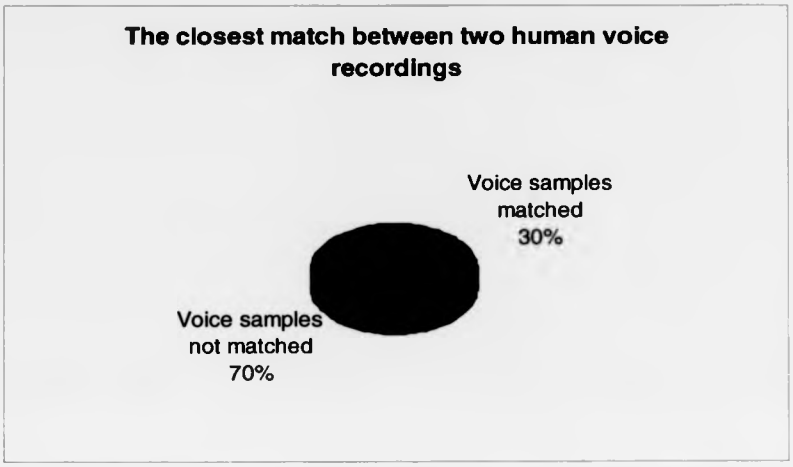


Figure 6.4: The closest match between two human voice recordings generated by the same person repeating a different pass phrase

Empirically, it has been shown that a biometric output of the research system alters every time a valid user employs a new spoken sentence. This characteristic of the system developed enables the speaker to keep a set of biometric signatures. The latter constitutes an important advantage that will be emphasised in the next chapter.

In conventional password based security systems, the strength of the application is closely associated with the secrecy of the context of the password. Many theories have been developed to guide the users and administrators of such systems on how to choose strong passwords that are unlikely to be guessed by a third party [30].

These theories, or methodologies, give advice on how these sensitive words should be stored, exchanged and what their life cycle should be. Once a password is revealed to an intruder, it provides him/her with full access to the system, dispersing any defence mechanisms.

Is the knowledge of the context of the pass phrase the same catastrophic in this system, as it is in the conventional systems described in the previous paragraph? All the voice recordings generated by different speakers repeating a standard pass phrase were collected and analysed thoroughly. The results of this analysis are presented in Figure 6.5.

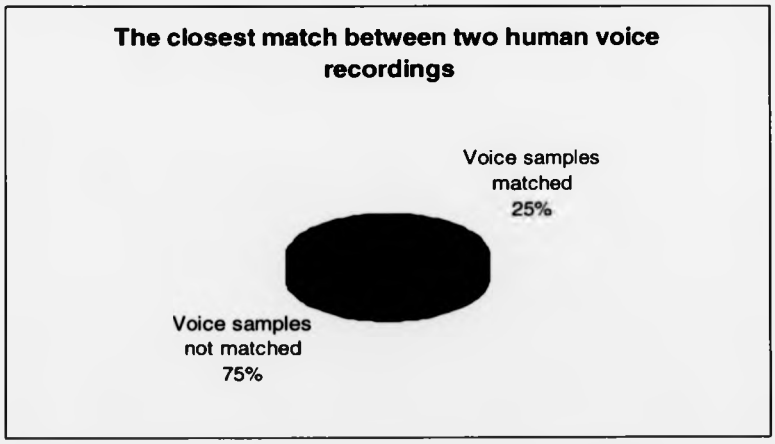


Figure 6.5: The closest match between two human voice recordings generated by different people repeating the same pass phrase

This figure demonstrates the tolerance of the biometric algorithm to attacks, where the intruder knows the context of the spoken sentence. The unauthorised party

cannot manipulate the system easily, and the final output will not be the same with a valid biometric signature. The closest match, recorded between two speech signals, reached twenty five per cent, disabling the attacker from imitating a valid user.

The last testing category aimed to investigate the probability according to which two people, repeating different sentences, may coerce the algorithm to produce the same output. All the preceded testing stages indicate that this is not likely but, the algorithm was executed numerous times, analysing human voice recordings generated by different speakers repeating different pass phrases. The algorithm did not produce the same output during this experimental phase of research. The closest match between two voice signals was of the value of fifteen per cent, as it is illustrated in Figure 6.6. It can therefore be concluded that two people, by repeating totally different pass phrases, can never generate the same speech signal.

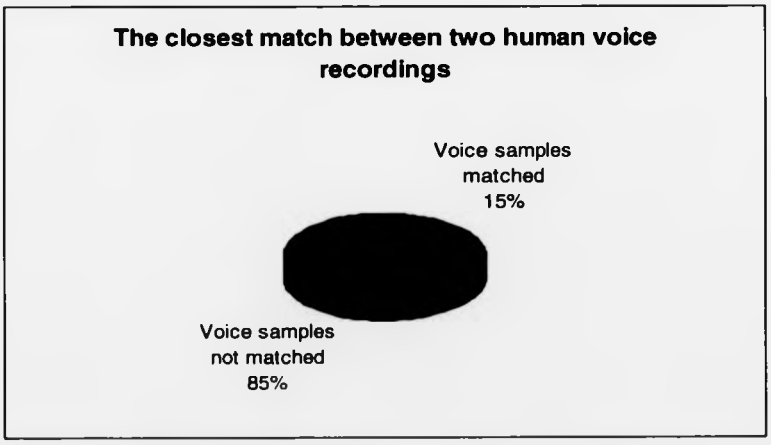


Figure 6.6: The closest match between two human voice recordings generated by different people repeating different pass phrases

The small, but not neglectable, similarity can be explained due to the redundancy of the English language. The majority of grammatically correct English sentences do contain certain letters, sometimes even words that are repeatable. This repeatable pattern creates the small number of common elements between two signals generated by different spoken sentences and people.

Chapter 7: Generation of Secret Keys

7.1 Introduction

In Chapters 5 and 6, the entire digital signal processing of the speech was presented. As was shown, the researched algorithm reliably produces 128 bit biometric information. The association of this information with the characteristics of the human voice was also emphasised. This chapter outlines the necessary steps to transform the 128 bit data into a 160 bit secret key that can be employed in symmetric key cryptography.

There are many encryption algorithms, the most important of which have been presented in Chapter 4. A common characteristic between these algorithms is the pluralism observed in secret key generation. A vast number of methodologies introduce key generation techniques, each one with its own advantages and weaknesses.

A significant part of this research work focuses on the generation of secret keys. Instead of using the widely employed pseudo random number generators, the output of the biometric system will provide the necessary seed for the key generation. This approach enables the system to produce secret keys that are closely associated with the biological characteristics of the user and, as shown previously, are very hard to copy and forge.

Once again, out of a wide range of secret key generators, one of the most fundamental mathematical formulas to strengthen the generated number was chosen. This technique is known as one way hash functions. The main reason behind this

decision is the speed of the specified function, the low resource utilisation and its simplicity.

7.2 One way hash functions

A one way hash function, H(M), is defined as a function that operates on a variable length input (M) that returns an output of fixed length (n). In equation terms it is as follows [70]:

$$h = H(M) \text{ , where h is of length n} \qquad (7.1)$$

Additionally, it has some characteristics that excuse the term "one-way". These functions are called so because of their mathematical nature, according to which:

- Assuming M is known, it is easy to compute h
- Assuming h is known, it is computationally difficult to compute M so that it satisfies the equation H(M) = h
- Assuming M is known, it is computationally difficult to find another input data, M', such that H(M) = H(M')

One way hash functions serve as digital fingerprints. They are small pieces of data that help in identifying much larger digital objects. They are an important subset of hash functions. Before presenting more detailed information about this subset it is

useful to illustrate the functionality of the entire set of hash functions. A hash function is a function h which has, as a minimum, the following two properties [70]:

1. *compression*: h maps an input x of arbitrary finite bitlength, to an output h(x) of fixed bit length n.
2. *ease of computation*: given h and an input x, h(x) is easy to compute.

In an attempt to facilitate further the above definition, three potential properties are listed for an unkeyed hash function h with inputs x, x' and outputs y, y'.

1. *preimage resistance*: for essentially all pre-specified outputs, it is computationally infeasible to find any input which hashes to that output.
2. *2^{nd}-preimage resistance*: it is computationally infeasible to find any second input which has the same output as any specified input.
3. *collision resistance*: it is computationally infeasible to find any two distinct inputs x, x' which has to the same output [70].

The preimage resistance fulfils the condition needed to classify hash functions as one way hash function. Therefore it can be stated that preimage resistance ≡ one

way. The analytical classification of cryptographic hash functions is presented in Figure 7.1.

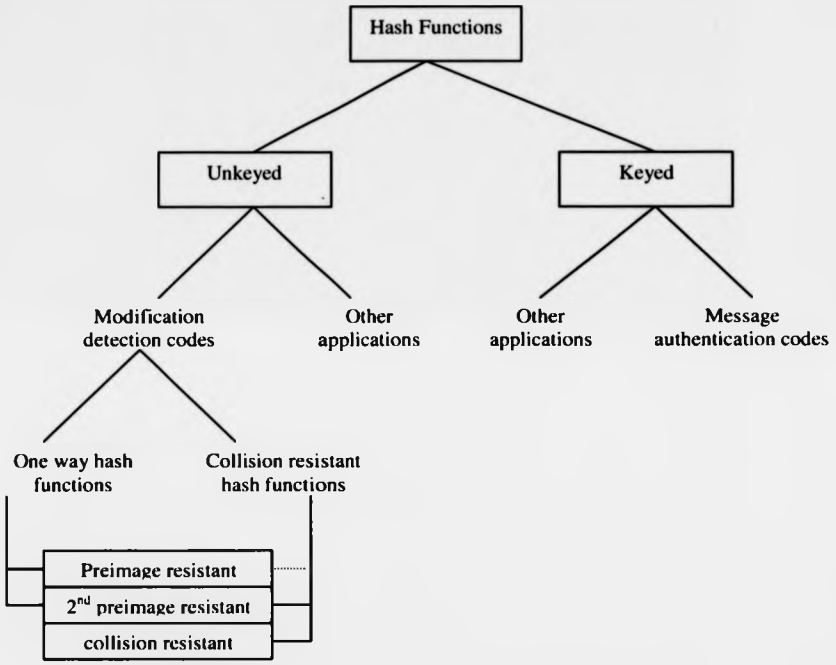


Figure 7.1: A simplified classification of cryptographic hash functions and applications [70].

From Figure 7.1, it may be seen that hash functions are divided into two main categories, namely keyed and unkeyed. The second category contains one way hash functions and therefore it is of greater importance for this thesis work. One way hash

functions together with the collision resistant hash functions constitute the modification detection codes (MDC). These, are also known as manipulation detection codes, provide a representative image of a message and aim to facilitate data integrity assurances as required by numerous applications. A hash function with 2^{nd} preimage resistance and preimage resistance as additional properties is classified as one way. If the preimage resistance is substituted by collision resistance, then the hash function becomes collision resistant. The dashed line in Figure 7.1 connecting the collision resistant hash functions with the additional property of preimage resistance indicates that the latter may be used but is not mandatory for this category of hash functions.

Iteration is a very common characteristic of unkeyed hash functions. Iterative processes are heavily involved in the design of such functions, enabling them to hash arbitrary length inputs by processing successive fixed-sized blocks of the system's input. Consequently, the input of a hash function of arbitrary finite length is divided into several fixed length blocks. Hence, the proper execution of this process requires the padding (addition of extra bits) so that an overall bitlength, which is a multiple of the block length, is attained. In highly secured systems, the padding process adds an extra block responsible for indicating the bitlength of the unpadded input.

A general model of an iterated hash function is illustrated in Figure 7.2. A high level view of the main processes involved into hashing an arbitrary length input, enable the reader to perceive the main idea behind the iterated hash functions.

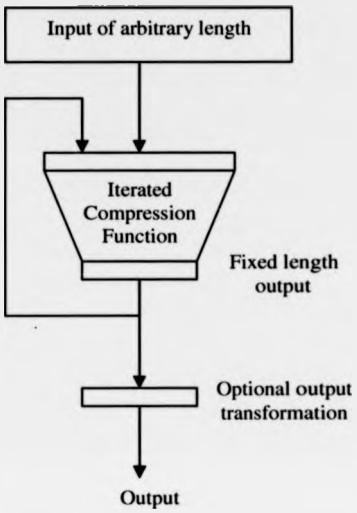


Figure 7.2: General model for an iterated hash function [70].

There are mainly three categories of iterated hash functions. The first one, consists of those functions that depend on block ciphers, modular arithmetic is employed in the second one and finally there is the category of customised hash functions. The last group are designed "from scratch", aiming to provide the algorithm with high speed and make it independent of any system subcomponents (such as modular arithmetic or block ciphers).

7.3 MD4 and SHA-1

Message Digest algorithm four (MD4) was initially designed by Ron Rivest, for software implementation on 32-bit machines. The output of the algorithm, also referred as message digest, has a length of 128 bits. The design of MD4 algorithm is based on five cornerstones which are security, direct security, speed, simplicity and compactness and favour little-endian architectures [71]. Little-endian describes computer architectures in which, bytes at lower addresses have lower significance.

According to the inventor of the algorithm, it is computationally infeasible to find two messages, the hash values of which are the same [71]. This constitutes brute force as the most efficient attacking method. The latter can be translated in 2^{64} operations in an attempt to detect distinct messages with the same hash value. This number of operations increases to 2^{128} whenever the adversary intends to find a message yielding a pre-specified value. Additionally, the security of the algorithm is not based on any assumption, and therefore its strength cannot be affected by future mathematical breakthroughs [71]. Furthermore, high speed software implementation is a distinctive characteristic of the MD4. This is achieved by intentionally avoiding large data structures and complicated programs. The optimisation of MD4 for microprocessor architectures is achieved with the employment of little-endian architectures.

Once the algorithm was released, there were parallel attempts on breaking it and exposing its vulnerabilities. Until recently, there was not any method of attack, or combination of attacks, capable of intruding the full length of MD4 algorithm.

Nevertheless, some of the cryptanalytic attempts managed to penetrate certain rounds of the algorithm. An attack targeting the first two rounds is described in [72], whereas [73] demonstrates the vulnerability of the algorithm in the last two rounds.

The last paragraph indicated the known limitations and weaknesses of the MD4 algorithm. The algorithm is briefly analysed in this section but not for its strength and resistance in cryptanalytic attacks. MD4 became the foundation for a series of more sophisticated algorithms that superseded it. Thus, the thorough analysis of its mechanism beneficial for the overall understanding of any algorithm belongs to MD4 family. Table 7.1 depicts the most important hash functions based on MD4.

Table 7.1: Summary of selected hash functions based on MD4 [70].

Name	Bitlength	Rounds x Steps per round	Relative Speed
MD4	128	3 x 16	1.00
MD5	128	4 x 16	0.68
RIPEMD-128	128	4 x 16 twice (in parallel)	0.39
SHA-1	160	4 x 20	0.28
RIPEMD-160	160	5 x 16 twice (in parallel)	0.24

The title of this section predisposes the reader for the type of one way hash function, implemented in the research work. SHA-1 stands for Secure Hash Algorithm and is the result of a mutual effort of National Institute of Standards NIST and NSA. Table

7.1 provides some basic information about the length of the generated key, the number of rounds and the relative speed of the algorithm. SHA-1 is one of the two algorithms producing the longest output, one hundred sixty bits. The size of the message digest is of great importance and played a significant role in choosing the specified algorithm. It increases the security of the algorithm by making it more tolerant against brute force attacks. Therefore, SHA-1 and RIPEMD-160 are stronger than MD5.

The U.S. federal government employs SHA-1 for a certain range of applications. It is interesting to analyse here the operations constitute the core of the algorithm. Padding is an important procedure within one way hash functions. As it was pinpointed in the last section, padding is performed by adding a one and then as many zeros as necessary to produce a bitlength which is multiple of the block length. In this case the message is transformed into 64 bits short of a multiple of 512. An extra block of 64 bits is added to indicate the bitlength of the unpadded input. The algorithm employs five 32-bit variables which are initiated as follows:

A = 0x67452301

B = 0xefcdab89

C = 0x98badcfe

D = 0x10325476

E = 0xc3d2e1f0

The message is divided into data blocks of 512 bits each and they are processed one at a time. The five variables A, B, C, D and E are replicated into variables a, b, c, d and e respectively. At this point the main loop, consisted of four rounds, begins. Each round contains a set of twenty operations and it is comprised of a nonlinear function on three of a, b, c, d and e, followed by shifting and adding similar to MD5 [53]. The sixteen 32-bit message blocks are expanded to eighty 32-bit blocks. Thus, there are 80 words that are fed to an eighty step operation. At the end of the data processing, the values of the variables a, b, c, d and e are copied to A, B, C, D and E respectively. Once finished with the first message block the same procedure, also known as compression, applies to the remaining blocks of the message. The concatenation of A, B, C, D and E, finally generates the output of the SHA-1 algorithm.

There are several differences between the SHA-1 algorithm and MD4. Starting with the length of the message digest, MD4 is weaker in brute force attacks due to its smaller output size. The compression procedure, employed in SHA-1, has an additional round, making the total number of rounds four. It has been mentioned earlier in this chapter that the MD4 algorithm is based on a three round compression procedure. Within the same procedure the secure hash algorithm expands each 16 word message to an 80 word block. In order to achieve this, the last 64 of the 80 words is the result of an XOR operation of four words from earlier positions in the expanded block. During the entire SHA-1 operation there are employed four non-zero additive constants, whereas MD4 uses only two non-zero constants. Another

major difference between the two algorithms is the use of big-endian in SHA-1 instead of the little-endian in MD4.

All the variations described briefly above, constitute SHA-1 as the most secure one way hash function. There are no known attacks against the algorithm. However, the first version of the algorithm introduced in 1993, known as SHA-0, had a certain vulnerability according to which two messages were able to have the same hashing value. This specific method of attack was able to detect two messages with this characteristic in about 2^{61} evaluations of the compression function. This number is significantly smaller than the 2^{80} evaluations needed according to the brute force attack needs, based on the birthday paradox [74]. In 1995, a second version of the algorithm, known as SHA-1, tackled this problem. Henceforth, the SHA-1 was not exposed by any cryptanalytic attack. The code that performs all the operations necessary for the implementation of the algorithm is presented in Appendix 5.

The result of this research step is the association of each biometric signature with a message digest with a standard 160 bit length. The output of the digital signal processing described in detail in Chapter 6, provides the necessary input for the one way hash algorithm. Then, a 160 bit hash value is generated and associated, in essence, with the speaker. The latter enables the user to obtain a piece of data, closely related with his or her biometric characteristics. The SHA-1 algorithm does not allow the generation of 128-bit biometric output used for the production of the message digest. Therefore, the output of the hash function is closely dependent on

the output of the biometric system; however it is impossible to compute the second by knowing the first.

The intention behind this sequential processing of data is to enable the use of the message digest as an encryption key for symmetric key cryptography. The employment of human voice as described in the previous chapters, the SHA-1 algorithm and their combination to a complete security system constitute the core of the discussion presented in the next chapter. Nevertheless, the strength and weaknesses of the 160-bit keys produced by the SHA-1 are analysed in the next section.

7.4 Strengths and weaknesses

Until now, a system has been presented that generates keys of certain length, intended to be used for symmetric encryption purposes. The first thought of a cryptanalyst, attacking a symmetric key encryption is the length of the key. Historically it has become apparent that mathematical and technological evolutions follow a rather fast pace. Consequently the ability of cryptanalytic systems to brute force attack keys is increasing exponentially [75]. Key lengths that were thought to be completely secure, in the past, have become obsolete nowadays. The minimum size for a symmetric key is 128 bits. Anything greater than this value increases significantly the security of the key itself.

However the security of the encryption algorithm does not only rest on the length of the key but is also dependent on the generation process. If the key generation procedure is not solid and leaks useful information to third parties, it will be attacked in preference to the key. There is no reason for a cryptanalyst to spend a lot of effort in guessing a symmetric key, when there is the option to control the key generation mechanism. Thus, the security of the key production mechanism is as important as the key length is.

The proposed security solution bases the generation of the symmetric keys on the SHA-1 algorithm. This algorithm needs a data input, or seed, to operate. The biometric algorithm provides the necessary information to the one way hash function. Therefore, it can be safely stated that the key generation mechanism consists of two elements, the biometric and the one-way hash algorithms.

According to the theory of one-way hash functions', whenever the input to the function is the same, the generated output has always the same value. This enables the cryptanalyst to generate a legitimate key if and only if he or she manages to feed the one-way hash function with the same message. As has been mentioned before, the implemented security system is intended to be embedded on a smart card. The environment of smart cards, described in Chapter 5, is quite restrictive for the developer of smart card applications but provides high level physical security. The data flow between the biometric system and the one-way hash function is well protected to eavesdropper's attempts. However, if the system implementation does

not take place within a smart card, extra security measures should be taken to prevent a third party tapping into in the communication between the two elements of the research scheme, to steal sensitive information.

Another important aspect of secure key management is the storage of the generated symmetric keys. There are examples of weak security systems that employ the most sophisticated key generation algorithms, produce securely the necessary keys but neglect the importance of their storage. As a result, the keys can be easily found and used from unauthorised parties with catastrophic consequences to security system. Chapter 5 has indicated that smart cards constitute a device upon which such sensitive data can be stored with adequate protection. Nevertheless, there are recorded cases according to which researchers have managed to attack successfully smart cards and access valuable data kept in them [60, 62, 63]. Most of the cryptanalytic efforts of this category, are based on hardware devices and require the presence of the smart card and its placement on a laboratory bench (refer to Chapter 4 for more details). In order to avoid running the risk of loosing encryption keys, it is better not to store the keys generated on the smart card. Instead, each time a user desires to encrypt data, a symmetric key is generated instantly. This approach is beneficial for the secrecy of the encryption key but complicates the key distribution. If the keys are generated "on the fly", just before the encryption of data, then each time a connection is established with the receiving end, the encryption key should be also transmitted. Without knowing the encryption key, the receiver of the transmitted message will be unable to decrypt the message and recover the plaintext.

Inevitably, the overhead of such a system increases significantly and causes a lot of concern.

Wireless transmission of data should efficiently utilise the limited communication bandwidth. The price of increased overhead for security purposes cannot be accepted. Any application adopting this approach becomes instantly obsolete and not applicable to wireless communication. To avoid this danger, a minor change to the existing operation of the algorithm is introduced. Via the algorithm, the user can create an encryption key which will be used to transform plaintext into ciphertext. Once this happens, the symmetric key is transmitted together with the first encrypted block of the message. However, this happens only once and should not be repeated every time the user establishes a new connection to the same receiving end. This scheme, introduces the idea of communication based on a group of symmetric keys. Lets suppose that person C wants to communicate with person B. A key is generated by C, and transmitted to the communicating party B. Every time a new communication link establishes between C and B, person C regenerates the same key to encrypt the data to be sent, but never retransmits the key again. The receiving end, B, uses the key sent to him/her in the beginning to decrypt any received data. If person C wants to establish a secure communication link with person W, then a new key should be generated by C and transmitted to W.

The new method of handling the generated encryption keys provides a viable answer to the overhead problem. It minimises the risk of losing the symmetric keys to third

parties during the transmission because this takes place only once. A disadvantage of this method occurs when a single user desires to communicate securely with a vast number of people. This implies that the user should have a huge number of symmetric keys and be able to reproduce them. There are many techniques and protocols focusing on the key distribution problem [2, 12-15]. At this point it has to be emphasised that the mechanism of distributing and managing the keys between two wireless users or a wireless user and a database, is out of the scope of this thesis. It is fully acknowledged that it is an aspect that has to be thought carefully in order to enable the proposed system to provide end to end security. Nevertheless, the main research interest is the generation of the secret keys within the restrictive environment of smart cards. It can be stated though, that there is the belief of employing hybrid systems into securely distributing the symmetric keys. Hybrid systems combine symmetric and asymmetric cryptography, utilising the advantages both techniques have into a single solution. Based on asymmetric cryptography, a user B has a pair of keys B_s and B_p, the secret and public key respectively. The public key is known to everybody who desires to communicate with person B. Application of this to the scheme above with communication parties C, B and W, is as follows. The generated symmetric key by person C, is encrypted using the public key B_p that belongs to communicating party B. Once transmitted, the receiving end decrypts the ciphertext with the employment of the secret key B_s and recovers the symmetric key that was hidden in the transmitted message. A symmetric communication between C and B can then begin.

Finally, it has to be stated that well established cryptanalytic attacks, such as known-plaintext, chosen-plaintext, differential and linear, cannot be applied to the generated keys. They become a useful research tool, only when the symmetric keys are utilised by a specific encryption algorithm. The only attack that may be proved useful is the dictionary attack. However, even in this case the cryptanalytic method should be accompanied by sophisticated human voice generation systems. Moreover, numerous recordings of a legitimate user's voice should be taken and unlimited access to the biometric part of the algorithm should be gained.

In an attempt to demonstrate the final output of the research algorithm, the voice inputs, described and analysed in the previous chapter (section 6.5.2), are fed into the research algorithm. The voice signals presented in Figures 6.11, 6.12 and 6.13 were generated by the same speaker repeating pass phrases S1, S2 and S3 respectively. The research application performs all the steps described in this thesis and generates three totally different 160-bit keys k1, k2 and k3 respectively.

k1:

00110111100010001001011100011011100100101001000111010111111111101010
10101000101001000101010011100110011011111111010000000010110010101110
01111011101101111110110

k2:

11110000001010011011101010100100001110010010011101101100010010101110

00100110100001111001100001101000110001000010100110100100100101010010

01011000011100100100101O

k3:

10100001100000100111100010000111001010000111001111111001001111001111

10100100110011100110111010010010101111100000101111000010100011111111

00101001011000101010000O

Chapter 8: Conclusion and Further Work

8.1 Introduction

In this chapter various aspects of the implemented system are analysed. Additionally, the entire research effort is going to be thoroughly discussed aiming to emphasise those elements of the system that classify it as a stand alone application. The various components of the different research stages are connected together and their final performance as one concrete unit examined.

This chapter is especially important because research investigation targets the sensitive and fragile area of digital security. Limited experience within the security arena may easily mislead an individual into assuming that the strength and tolerance of a developed system lies only on the strength and tolerance of the subsystems it consists of. History teaches that any system intended to serve the intact transmission of data, should be thoroughly analysed. The analysis should not be constrained only to the specifications and functionality of the subcomponents but also encompass the interaction and data flow between them.

8.2 Overall Description of the Security System

Behind any research work there is a motivation, necessary to define the main target and justify the importance of achieving it. The motivation in this case, has been replaced by the need to improve the current level of security in wireless communication. Wireless transmission of data is a particularly vulnerable method of communication and frequently enough sensitive information becomes prey to the desires of not so skilled cryptanalysts.

It has been identified that the weakest link in the wireless communication chain is the data package transfer between the mobile user and the antenna of the cell he or she belongs to. Therefore, a system has been developed that incorporates unique characteristics of the wireless user, in an attempt to increase the available security defences. The system requests a human voice input, generated by speaking a standard pass phrase. It collects this recording and follows a predetermined sequence of digital signal processing steps that enable the research algorithm to extract unique characteristics that are hidden within the audio signal. It has been found and empirically shown that these characteristics are contained within the range of 1 kHz and 2 kHz, benefiting the extraction algorithm, and improving its performance. The output of the human voice processing is directed to the input of a one way hash function. The chosen hash algorithm digests the biometric message and generates its hash value. The generated data is a string of 160 bits long and is ready to be utilised by a symmetric cryptographic algorithm as the secret key. The one-way nature of the algorithm prevents any reverse engineering process and does not allow a cryptanalyst to reveal the biometric output by knowing the secret key. It is important to mention that all the methodologies and techniques employed during the research work have been chosen for their low processing requirements, so that the algorithm can be embedded on a smart card and operate within the wireless device.

A functional view of the algorithm may also be presented. In the beginning, the user indicates that he or she intends in using the key generation algorithm embedded on the wireless device. The algorithm receives this request and starts its operation.

Instead of directly prompting the user to record a pass phrase of his or hers choice, the system records the environmental noise without the user's knowledge. This special recording lasts for only two seconds and collects any audio signal other than speech. The readings of this function, drive the algorithm to choose an appropriate value for the variable known as "tolerance rate". This value is critical for the successful performance of the embedded system and will be needed in later stages. The setting of the tolerance rate is closely dependent on the amplitude of the recorded audio signal. A high amplitude indicates that the environment is classified as noisy and therefore the algorithm expects a highly distorted speech signal. As mentioned in Chapter 6, the tolerance rate has three preset values 2, 4 and 6 per cent. Once the system detects a noisy environment, the value of 6 per cent is assigned to the tolerance rate. The other values may be assigned accordingly. In order to result in the three values referred above, a systematic analysis of the performance of the biometric algorithm followed closely. It is likely that these values may change if different recording devices are utilised. Thus, they are not recommended as a panacea to minimise the background noise phenomenon, they represent the laboratory values that enable the algorithm to perform as recorded.

Having set the tolerance rate, the system prompts the user to start speaking. The speaker's voice is recorded and the digital signal processing phase begins. The butterworth filter designed, removes all the undesired frequencies and keeps the voice signal with frequencies within the range of 1 kHz and 2 kHz. The initial representation of the signal, amplitude vs. time, does not provide the expected

information and the employment of the Fast Fourier Transform is mandatory. The extraction algorithm processes the remainder of the speech signal, in its new representation form (amplitude vs. frequency), and aims to detect and record the numerous peaks (harmonics). The detached algorithm for the harmonics needs to operate in more than just the peak values. This is because the speech signal is never identical when taken from different recordings. Small variations in human speech, together with the distortion of noise signals, would not allow an algorithm based solely on peaks to operate properly, severely affecting the reproducibility of the biometric output. To overcome this limitation, once a peak is identified its value is stored together with the values of the previous and next sample respectively. In essence, the algorithm holds a set of three amplitude values and an extra round of processing begins. The difference the harmonic has with the previous and the next samples is calculated and recorded on variables BnL and BnR, where n is the total number of the detected peaks in the audio signal. Every time the user attempts to reproduce the biometric signature, the new harmonics of the human voice signal are analysed and electronically stored. If there is variation between the set of harmonics of the first and second speech signals, variables BnL and BnR are recalled from the memory of the smart card together with the tolerance rate. If these variables are within the accepted tolerance rate then the biometric algorithm produces the same biometric signature having recognised that the second speech signal has been produced by the same individual repeating the same pass phrase.

information and the employment of the Fast Fourier Transform is mandatory. The extraction algorithm processes the remainder of the speech signal, in its new representation form (amplitude vs. frequency), and aims to detect and record the numerous peaks (harmonics). The detached algorithm for the harmonics needs to operate in more than just the peak values. This is because the speech signal is never identical when taken from different recordings. Small variations in human speech, together with the distortion of noise signals, would not allow an algorithm based solely on peaks to operate properly, severely affecting the reproducibility of the biometric output. To overcome this limitation, once a peak is identified its value is stored together with the values of the previous and next sample respectively. In essence, the algorithm holds a set of three amplitude values and an extra round of processing begins. The difference the harmonic has with the previous and the next samples is calculated and recorded on variables BnL and BnR, where n is the total number of the detected peaks in the audio signal. Every time the user attempts to reproduce the biometric signature, the new harmonics of the human voice signal are analysed and electronically stored. If there is variation between the set of harmonics of the first and second speech signals, variables BnL and BnR are recalled from the memory of the smart card together with the tolerance rate. If these variables are within the accepted tolerance rate then the biometric algorithm produces the same biometric signature having recognised that the second speech signal has been produced by the same individual repeating the same pass phrase.

The modification alone is important as it provides an answer to the most common and hard problem of behavioural biometric systems, that of the reproducibility of the same unique characteristics. This development improves the overall reliability of the software application implemented, guarantees the usability of the system and ensures the adjustability of the algorithm for distorted human voice signals.

The output of the biometric system is a string of data, 128 bits long, formed by a combination of data originating from the detected harmonics. Human characteristics are incorporated in the output of the biometric system. Moreover, these characteristics are responsible for the generation of the specific output. Any person who desires to regenerate the same output, has to get access to the necessary physical characteristics. This is very difficult, requiring extensive resources and valuable time for the successful invasion of the system. The detailed testing procedures, described in the previous chapters, did not demonstrate a single case where the system generated a valid biometric output for a non registered user.

The produced 128-bit biometric signature encloses the essence of this research work. It has been overemphasised, throughout this thesis, that the 128 bits are sensitive data. In case a cryptanalyst accesses this information the proposed security scheme is exposed to any security threat. Consequently, protecting the biometric signature provides the best reassurance for a tolerant security algorithm. In an attempt to minimise the risk of exposing sensitive data to unauthorised parties, the biometric signature should never be transmitted via any kind of communication network. Alternatively, a string of data that characterises the 128 bit of information, but does

contain elements that can help in reengineering the biometric signature, should be used.

The next operational step, as outlined in the designed system architecture, is the one way hash function and especially, the *secure hash algorithm 1* (SHA-1). The output of the biometric system is utilised by the hash algorithm in an attempt to expand the generated data. The addition of extra bits, thirty two in total, not only strengthens the resistance of the algorithm to certain cryptanalytic attacks, but at the same time protects the sensitivity of the biometric output. The moderate speed algorithm stops effectively any reverse engineering attempts that are intending to expose the input of SHA-1. Thus, the hashed value is generated by biometric characteristics of the user, but is unable to reveal them to third parties. In any other case, the cryptanalyst may be in a position to manipulate the key generation system by gaining access to the biometric signature.

The employment of SHA-1 algorithm is a standard procedure that does not contain research innovative elements. Its implementation is dictated by a series of steps that constitute public domain. Thus, the length of the related research work is intentionally kept to the minimum. However, the combination of the specified one-way hash algorithm with the generated biometric signature, as described in previous chapters, demonstrate a security system that uniquely produce 160-bit secret keys for encryption purposes.

8.3 Applicability of the Security System

The research on the specific technological area initiated from the need to increase the current level of security in the wireless transfer of data. The design and development of the application, according to the smart card standards, enables its usage to a wide range of communication systems.

The employment of the research algorithm requires the presence of a wireless device upon which the smart card is embedded. If the receiving site is another wireless user, an end to end security service is provided. The communicating party, which initiates the communication, speaks a pass phrase of his/her choice and the application generates a 160 bit number. This output serves as a symmetric key that can be utilised by relevant cryptographic schemes. Any data prior to its transmission should be encrypted by the generated symmetric key. A key distribution algorithm is responsible for the intact exchange of the key to the receiving end of the communication link, so that the decryption of the ciphertext can take place. Both the encryption and the key distribution algorithms were not investigated and developed during this thesis work. The undergoing research intentionally kept an equal distance for all the existing algorithms, in these two areas. Therefore, the final outcome is applicable to a wide range of encryption schemes and is not restricted to a single key distribution system. Moreover, the wireless community is in a position to deploy a software tool that generates secret keys "on the fly", without transmitting any sensitive information that will enable the adversary to penetrate the system. Another significant advantage of the end to end applicability of the research algorithm is the ability, provided to the user, to maintain a list of symmetric keys. The output binary

number is dependent, as shown in previous chapters, on the context of the pass phrase. Every time the user desires to change the encryption key, the spoken sentence should be replaced by a new one. Thus, a single user may be keeping a list of keys (one to many relationship), each one used to communicate with a different party. It is also feasible to rotate the symmetric key on a frequent basis in order to harden the task of a cryptanalyst.

Instead of a wireless user, the receiving end may not be human at all. Sophisticated network nodes are in a position to execute the research algorithm due to their artificial "intelligence". In this scenario, the wireless user follows the exact same process until the algorithm produces the secret key on board. Once this happens, the key is distributed to the node of the cell network and the communication between the two parties is encrypted. The network nodes have the ability, based on their specifications, to perform the encryption and decryption process. However, the proposed application has a significant drawback, which is no other than the high number of keys each node withholds to serve its wireless users. As this number increases, the node becomes incapable of operating properly and there is a risk of causing its temporary or even permanent cessation of operation, resulting in a network deadlock situation. Nevertheless, if it can be assumed that there is a network wherein the number of users per node is restricted to low value, the proposed application may very well provide a high level security to all the data transferred wirelessly from the user to the node serving his/her geographical area.

Another usage of the algorithm based significantly on the previous concept, is the secure communication with a database. This resides on a remote site and a person may access it by using the wireless device with the new algorithm embedded in it. The algorithm does not grant database access to the user as most conventional biometric systems do. It encrypts the data sent between the two communication points. This approach enables the database to deploy an additional security measure. Not only does it reveal the managed content to a valid account holder but it also protects its transfer via the various networks.

The first time the user contacts the database, the embedded algorithm on the wireless device generates a symmetric key. This symmetric key is exchanged with the database and securely stored. Henceforth, any data transferred between the user and the source of digital data is encrypted with the help of the symmetric key. In corporate databases, or any other systems holding valuable information, the keys generated may have a certain life time and be periodically changed. This comes in line with the approach followed in conventional password dependent systems, where the each secret word has a specific time duration during which it remains valid.

One of the innovative elements of the research work presented in this thesis is the algorithm itself, designed to be embedded on smart cards. The decision to develop programming code for such a restrictive environment did not allow the employment of a wide range of theorems and techniques that are effective but at the same time very demanding in programming resources. However, the developed research work

can migrate to more powerful processing systems. There are three applications of the implemented security system described till this point, the categorisation of which is based on the type of the communicating ends. Therefore, a wireless user can communicate with another wireless user (end-to-end), with the node of a specific cell and with a database residing in a remote location. The result of migration is to enable a person, using a standalone mainframe or personal computer, to communicate with the same three destinations as above. Although, this modification broadens the horizons of the research work it does not add substantial value to the system. All the advantages of running the algorithm on a mobile device, are not any more applicable. At the same time, the fundamental principles upon which the software program operates are significantly weaker than the methodologies already developed in this new technical environment. Moreover, the increase of the complexity level of any technical device, with an embedded central processing unit, causes a series of security concerns. This explains the higher tolerance of wireless devices towards viruses' attacks.

A simplified version of the research algorithm can also be used as a conventional biometric system. Most of biometric systems of any type, both physical and behavioural, after a series of processing phases produce a binary output which is either Yes or No. Based on the same principle, a minor modification on the algorithm can achieve the exact same functionality. The entire process, thoroughly described throughout the thesis, remains unaltered and only a small procedure is added. In fact, the modification occurs on the intended destination of the

communication link. The 160-bit key is generated and exchanged to the receiving party. The recipient associates the key with the person transmitting it and saves the processed data. Every time the same user desires to communicate with the same recipient, a new key is generated, transmitted and compared with the stored one. If they are identical then the verification process has been completed successfully and the identity of the originator of the communication link is revealed. Otherwise, the only certain information is that the transmitting end is not who he or she claims to be.

Based on the conventional biometric approach, the security system may be used to grant access to restricted areas like laboratories or managing director's offices. Instead of operating the system on a wireless device, it can be embedded on a security door lock. Any person who wants to access the secured premises should use his/her voice. The lock device will be capable of storing a number of valid keys, programmed by the security officer. Thus, if the key generated matches one on the list, the door unlocks to let the person enter the premises. Each of the locking devices can also be connected with a closed network, enabling the security officer to monitor the attendance on the protected areas.

The research algorithm presented here can also be employed as one of the counter measures to reduce the theft of mobile telephones. Once a mobile device is purchased and activated the owner can use his/her voice to generate a key. This key, with manufacturer's permission, can be stored on the device and not on the smart

card. Whenever the device is turned off and back on again, the user has to speak the secret sentence of his/her choice in order to generate the same key. Unless this happens, the mobile phone cannot be fully operational and only emergency calls can be accepted. Thus, in case of a theft the new person has to generate a key identical to the stored one, something that is highly unlikely. After three unsuccessful attempts mobile phones that support new operating systems, like SymbianOS, may even communicate with the legal owner by an SMS or e-mail.

8.4 Further Development

The investigation of a certain research goal and the organised scientific attempts to achieve it create new research dimensions that were not anticipated always and that are really interesting to follow. This section aims to presents those research directions that were not explored throughout the course of time of the Ph.D. studies. In other words, the researcher outlines the future research steps he intends to make after the completion of his/her studies. Thus, continuation of work on this project may be considered in the following areas:

1. Extension of the range of speakers and phrases tested.
2. Investigation of the effect of using languages other than English for the pass phrase.
3. Implementation of the algorithms on a on a smartcard testbed.
4. Use of the algorithm in the context of specific encryption and key distribution schemes.

5. Systematic cryptanalytic attacks on the algorithm in its smartcard wireless environment.

The above are discussed in detail below.

8.4.1 Extending the Range of Speakers Tested

The majority of the observations and conclusions made are based on empirical results and not on generated mathematical formulas. Therefore, a larger pool of data input will improve the confidence and robustness of the system. It may also provide insight to develop mathematical explanations of the findings. For the purposes of this investigation large databases have been identified (mostly located in United States of America) that contain thousands of human voice recordings [76-78]. The interested party pays a fee or a membership and gains access to the context. However, prior to any additional simulations it is important to create testing scenarios that will be used as evaluation measures of the research outcomes. Without a carefully designed testing phase, the generated results cannot be assessed and the performance of the algorithm will be questionable.

8.4.2 Effect of Language Used

Additionally, the impact of the spoken language of the pass phrase should be investigated. All the voice recordings were based on UK English language, with the majority of the users being native speakers. The entropy between different languages varies and this may have an effect on the security defences of the implemented

system. A variation in the entropy implies a variation in the redundancy of the pass phrase. The higher redundancy a spoken sentence has, the fewer letters the attacker has to guess in order to regenerate the context of the message.

8.4.3 Smart Card Implementation

HARDWARE

The algorithm may be tested by implementation on a specific smart card. The entire software program will be embedded on it and a fully operational mode will begin. The engineering background almost certainly constitutes the variation between theoretical and real values. Even the most realistic simulation packages can not fully transfer all the conditions and restrictions taking place in an out of the laboratory testing.

SOFTWARE

The implementation of all the research phases presented in the previous chapters is achieved with the assistance of MatLab software package. It is a powerful scientific tool that accelerates the development programming time by using the many built-in libraries it contains. Recent versions of MatLab have the option to automatically translate source code into C ++, increasing significantly its applicability. However, in recent years a new version of Java 2 Enterprise Edition (J2EE) is emerging, especially designed for wireless programming. It is called J2ME and stands for Java 2 Micro Edition, which is a light weight version of J2EE so that it can operate within the resource-constrained platform of wireless devices. J2ME is utilised from the SymbianOS enabled devices. SymbianOS is the most feature-rich available

operating system targeting the wireless market. A future migration of the existing source code to J2ME proliferates the potential of the research work and provides new dimensions in its applicability.

8.4.4. Specific Encryption and Key Distribution Schemes

End-to-end security is a difficult challenge for the wireless community. In order to enable the algorithm to contribute in this area, an encryption algorithm together with a key distribution algorithm should be also chosen and implemented. The main characteristics these two algorithms should have are: minimum execution time, short handshake protocol, low processing consumption and flawless handling of the generated key. A significant amount of time should be invested in analysing and comparing numerous algorithms. The detailed knowledge of core functions performed for encryption or key distribution purposes will minimise the implementation and debugging time. The application of the research system to obtain end-to-end security on ad hoc networks is of great importance due to their increasing deployment in wireless communications.

8.4.5. Cryptanalysis in the Smart Card Environment

The entire operation of the algorithm may be tested once implemented on a smart card by mounting a specific attack. The main goal is to generate a valid secret key by using an impostor's voice. A high fidelity voice reproduction system should be designed, developed, tested and deployed. It is assumed that the intruder focuses on a specific valid user and records many times his/her voice. The first testing condition

is whether the impostor is in a position to generate a valid secret key by knowing only the authorised user's voice. The task will be simplified for the attacker if he/she can guess the chosen pass phrase and then try to create the voice signal. The difficulty of the latter is closely related with the choice of the pass phrase and the entropy of the preferred language. The complexity of the pass phrase and the real information it contains are decisive for the cryptanalysis of the spoken sentence. The second testing condition is based on the assumption that the content of the pass phrase is revealed to the impostor together with some irrelevant speech recordings of the user, but the voice signal should be generated. Thus, the capabilities of the developed reproduction system will be extensively tested. The duration to mount an attack of this type and the amount of data, voice recordings, necessary for its successful completion, are also two vital variables for the evaluation of the cryptanalytic method.

APPENDIX 1

The most known attacks.

Viruses	Worm
Trojan Horses	Trap Doors
Logic Bombs	Port Scanning
Spoofs	Ping of Death
SYN Flooding	SPAM
Smurf Attack	IP Address Spoofing
Session Highjacking	Sequence Number Spoofing
Sequence Number Spoofing	Man in the Middle Attack (MIM)
Replay Attack	Redirects
Social Engineering	Password Cracking
Denial of Service	War Dialing
Web site Defacement	Sniffing

APPENDIX 2

Analog Cellular systems outside the US TACS Variants, http://www.iit.edu/~diazrob/cell/tacsvar.html.

Advanced Mobile Phone System (AMPS) was the first analog cellular system developed by AT&T. Although largely used in the United States, the most widely deployed wireless system, AMPS has worldwide use also, with systems operating in over seventy two countries. Today, more than half the cellular phones in the world operate according to AMPS standards, which, since 1988, have been maintained and developed by the Telecommunications Industry Association (TIA). "AMPS allocates frequency ranges, within the 800 and 900 Megahertz (MHz) spectrum, to cellular telephone. Each service provider can use half of the 824-849 MHz range for receiving signals from cellular phones and half the 869 through 894 MHz range for transmitting to cellular phones. The bands are divided into 30 kHz subbands, called channels."

AMPS uses a 3 kHz (standard landline telephone line bandwidth) voice channel modulated onto 30 kHz FM carriers (one frequency for transmit, another to receive). The total of 50 MHz of bandwidth is divided between two operators, each of which uses half of its bandwidth for the forward channel (from base station to mobile) and half for the reverse channel. The B band (or block) is assigned to the local telephone company ("wire-line carrier"), and the A band is assigned to a non-wire line carrier. The division of the spectrum into sub-band channels is achieved by using frequency division multiple access (FDMA). The

two channels supporting a single conversation are separated widely to avoid confusion on the part of the terminal equipment. On average, the AMPS cell site has a radius of approximately one mile. Based on FDMA transmission and data duplexing methods, AMPS does not handle data well, with transmission generally limited to 6,800 bps.

H. Lawrence, T. Schaffnit, S. Kellog, "The comprehensive Guide to Wireless Technologies: Cellular, PCS, Paging, SMR and Satellite," APDG Publishing, pp. 79-81, 1999.

Narrowband Advanced Mobile Phone Service (N-AMPS) is an analog cellular system that was commercially launched in late 1991 by Motorola. Although similar to AMPS, N-AMPS uses analog FM radio for voice transmissions and features increased performance. N-AMPS acquired its name and differs from AMPS in that it uses "narrow" 10 kHz bandwidths for radio channels, one-third the size of AMPS channels. More of the narrow channels can be installed in each cell site and therefore serve more customers without the need to add additional cell sites. System capacity is improved by splitting a 30 kHz channel into three 10 kHz channels, thereby tripling AMPS capacity. Some of the control commands (signalling frequencies) are shifted to the subaudio, below the audio bandwidth for speech, 300 to 3,000 Hz, to facilitate simultaneous voice and data transmissions. Motorola equipment is necessary and only a small number of U.S. carriers deploy N-AMPS.

H. Lawrence, T. Schaffnit, S. Kellog, "The comprehensive Guide to Wireless Technologies: Cellular, PCS, Paging, SMR and Satellite," APDG Publishing, pp. 79-81, 1999.

"Nordic Mobile Telephone (NMT) was developed and placed into service in the 1980s by the telecommunications administrations of Sweden, Norway, Finland and Denmark to create a compatible mobile telephone system in the Nordic countries." NMT consists of two systems, the NMT 450, low capacity system, and the NMT 900, high capacity system. Nokia, the leading supplier, with more than three million customers served in almost 60 networks, made the NMT 450 commercially available in late 1981. Although NMT 450 had very good initial success the original design had limited capacity, which spurred the development of the NMT 900 and its subsequent introduction in 1986. "The NMT 450, operates in the 450 MHz range, has excellent signal propagation and especially suitable for sparsely populated areas supported by few cell sites – such as Eastern Europe – where distances to base stations can be several tens of miles. NMT 900 operates in the 900 MHz range, and is appropriate for more densely populated areas." Few nations outside of the Scandinavian countries use the NMT 450, whereas the NMT 900 has services available in over 40 nations, including certain Asian countries. The NMT system standard includes services such as caller ID, short message service (SMS), and voice mail indication. The NMT system will eventually be displaced by GSM.

T.S. Rappaport, "Wireless Communications: Principles and Practice," Prentice Hall PTR, 1995.

"The Total Access Communication System (TACS) was developed for use in the United Kingdom in the 900 kHz band. Its primary differences include changes to the radio channel frequencies, smaller radio channel bandwidths, and data i.e. original analog cellular signalling rates." Improving the efficiency of the AMPS cellular system radio channels produced the TACS system. There are multiple other variants to TACS, with each being derived from the basic U.S. AMPS cellular system. The frequency ranges of most TACS systems are 890 MHz to 915 MHz for the uplink and 935 MHz to 960 MHz for the downlink. The TACS system was initially allocated 25 MHz although 10 MHz of the 25 MHz was reserved for future pan-European systems in the UK. TACS has found acceptance in only a few nations, and it is not considered a long-term technology solution. The TACS is being replaced by new digital cellular systems such as GSM.

APPENDIX 3

AT&T Takes WIreless to the EDGE, by Stuart J. Johnston, PC World, July 18, 2000, http://www.pcworld.com/news/article.asp?aid= 17701, July 12, 2001.

Enhanced Data rates for Global Evolution (EDGE) is a new technology designed to provide operators a way to provide 3G wireless services using their existing hardware and spectrum. This is done by using a modulation scheme that is more efficient than that currently used in the GSM Standard, which is the Gaussian pre-filtered minimum shift keying. For every 1-bit-per pulse rate that the GSM Standard provides, Edge's *8-phase shift key* (8-PSK) will carry 3 bits of information. Therefore, Edge has the potential to increase the efficiency of GSM three-fold. EDGE is implemented over existing TDMA and GSM networks. Its design allows an operator to provide 3G services without purchasing 3G licenses and obtaining additional spectrum. This may be especially appealing in the United States where additional spectrum availability is in question. The primary two companies who will be selling EDGE technology are Ericsson, who developed EDGE, and Nokia. Both companies should have products ready for distribution beginning in 2001 through 2002. However, the development schedule will depend a lot on operators and their desire to implement EDGE either by itself or in combination with *Wideband CODE Division Multiple Access* (WCDMA). Another potential challenge for EDGE may lie in security issues similar to other wireless techniques. Since this technology is designed to work over existing TDMA/GSM infrastructures there is the possibility that a

wireless device could be configured to use more timeslots than it is allotted or to intercept information destined for another device. Since one of the goals of EDGE is to provide additional bandwidth there is every indication that wireless devices will be used for different types of applications, which may even include checking mail, including attachments, or accessing corporate intranets. Protection of entire packets, or at least the data within, therefore becomes a large issue since it is being broadcast over the airwaves for anyone within a cell to receive. At this time few details about the inner workings of EDGE are publicly available so it remains to be seen what kind of data protection will be used. The future for EDGE is anything but certain.

http://www.ericsson.com/3g/how/wcdma.shtml, July 4, 2001.

Wideband Code Division Multiple Access (WCDMA) is a wideband radio interface technology that provides far higher data rates than other 2G and even 3G wireless bearers available today. WCDMA technology supplies up to 2Mbps, and a highly efficient use of the total available radio spectrum. Compared to other technologies, WCDMA networks also may enable a faster and more cost efficient rollout. Base station deployment needs can be reduced by up to 30 percent and are available in a multitude of environments. Nokia describes that deploying WCDMA hardware with an existing GSM/EDGE base is relatively easy. Ericsson is another prominent developer of WCDMA based technologies since research began in the 1980s. The company delivered the world's first experimental WCDMA system to NTT DoCoMo in Japan in 1997. WCDMA

offers the concept of "capacity borrowing" from less-loaded adjacent cells to increase the instantaneous traffic handling capacity of an overloaded cell.

Sicap USSD Gateway, Product Description, Sicap Ltd., http://www.sicap.com/mobile_ussdgateway.cfm, July 4, 2001.

"Unstructured Supplementary Service Data, instead of the store and forward functionality, as with SMS, USSD is session-oriented, which indicates that when a user accesses a USSD service, the radio connection stays open until it is released by the user, application, or time-out. This provides faster response times for interactive applications." Due to that nature of USSD, the user can access these services while roaming; assuming the network they are currently using has the right infrastructure to support USSD messages. As with SMS, security could be a problem depending on what services are provided with USSD. As a security mechanism it is suggested that users supply a PIN along with the USSD communication, however this is not necessarily a good thing because, as with SMS the transmission medium is easily accessible by any who have the right tools, and decoding of the data potentially could be easy. If used for mobile banking applications or other financial services then, as with SMS, there are enough rewards to make compromise attractive. Sufficient encryption will need to be utilized to protect transactions and detect spoofing of transactions. Also similar to SMS, USSD can support the use of a SIM Toolkit and Smart Card for operators to develop and manage applications, which can uniquely identify subscribers and encrypt data using such algorithms as triple DES.

L. Huovinen, "Authentication and Security in GPRS Environment: An Overview," Department of Computer Science and Engineering, Helsinki University of Technology, http://www.hut.fi/-lhouvine/netsec98/gprs_access.html, June 21, 2001.

General Packet Radio Service or GPRS is a new wireless technology being introduced as an intermediate step between 2G and 3G wireless networks. GPRS is expected to allow data transfer rates many times this rate with a theoretical limit of 171 Kbps. Security of information with GPRS is probably one of the most challenging issues facing GPRS. Since GPRS, by design, connects the wireless GPRS backbone to external packet networks, and most often this is the Internet, GPRS data can now be subject to many of the attacks against traditional network assets connected to the Internet. Data for applications such as user names and passwords are a potential target as is denial of service attacks. Use of GPRS potentially allows the use of FTP or Telnet, user names and passwords could be sent in clear text from the GGSN to the destination server. The use of IPSec or VPNs can help provide security by encrypting information that is sent over the Internet. During the course of a GPRS transaction authentication is encrypted using the A3 encryption algorithm. When authenticated the subscriber can choose to have data encrypted as well. If this is chosen then encryption between the SGSN and the mobile device is performed "using the *GPRS encryption algorithm* (GEA), which is a stream cipher similar to A5." However, as the GEA is kept secret, evaluation of the algorithm is difficult if not impossible. With a maximum key length of 64 bits there is also cause for concern since this is far shorter than most generally used key lengths. Even if the

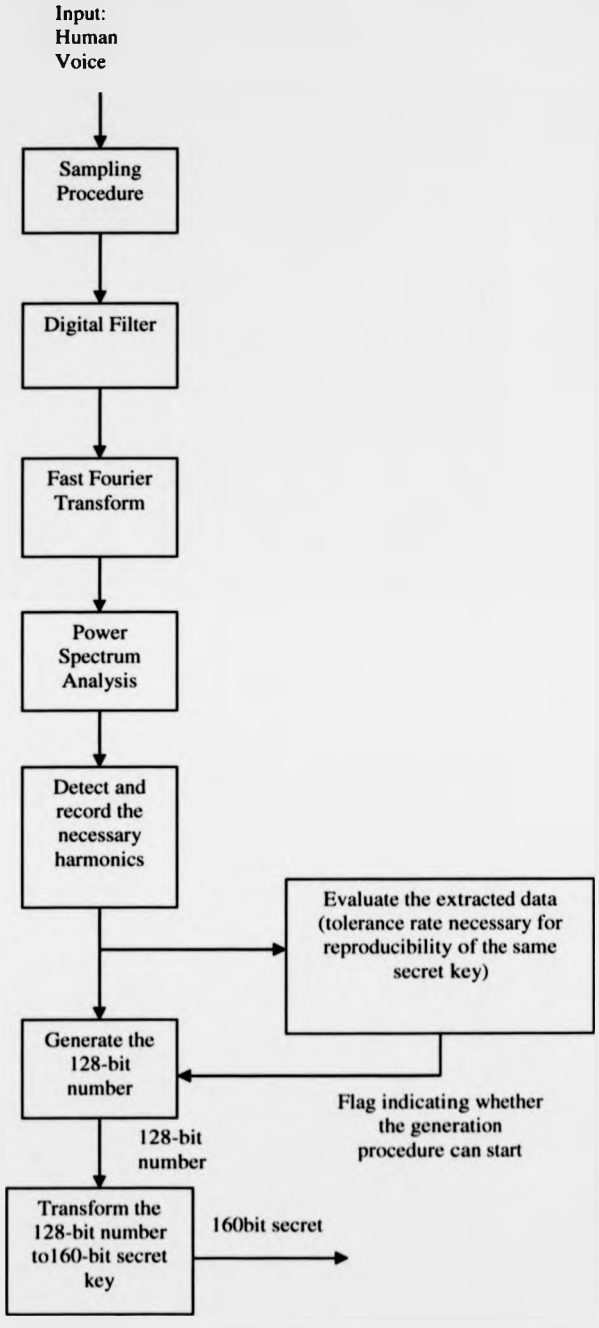
encryption is not compromised, theft of a subscriber's private key from the SIM card (smartcard) can be a problem. This type of attack requires physical access to the SIM card for some time as well as the technical skills to do it. However, once stolen, the thief can use the GPRS network as if they were the targeted *user* and incur large bills on their behalf, or potentially worse, use the stolen access as a stepping stone to attacks on other networks. The practices of mail bombing or spamming are also a significant threat to the GPRS user. If while using GPRS they are charged based on the number of bytes transmitted and received, then the consequences of large volumes of electronic mail could pose a significant threat, delaying the arrival of important mail and increasing the subscriber's bill significantly.

A. Hac, "Multimedia Applications Support for Wireless ATM Networks," Prentice- Hall, 2000.

Wireless Asynchronous Transfer Mode (WATM or Wireless ATM) is being investigated by a number of universities and research labs with the goal of using Wireless ATM and developing a standard specification to govern its implementation and use. "Wireless ATM (Asynchronous Transfer Mode) ATM is an emerging technology for high-speed networking. ATM is designed to support both real-time and on-real-time traffic with different delay, loss, and throughput requirements. In addition, ATM has the major ad- vantage of being scaling." ATM has been used for many years in traditional wired networks. The technology is flexible in that it can provide large amounts of bandwidth for

multimedia presentations when needed while experiencing minimal collisions and supporting high quality of service (QoS). At the same time it can also be used for video conferencing, which tends to be more variable in the pattern of data that is sent, and for normal network data, which tends to be less bandwidth intensive and does not necessarily mind minor delays or some retransmitted packets. At this writing, there are still many issues to be worked out with Wireless ATM. Although it seems to be in the sights of the wireless industry, it appears that WATM has a long way to go before it is commercially viable, not the least of which is formal specification, and adoption by the industry and international standards bodies.

APPENDIX 4


Input:
Human
Voice
Sampling Procedure
Digital Filter
Fast Fourier Transform
Power Spectrum Analysis
Detect and record the necessary harmonics
Evaluate the extracted data (tolerance rate necessary for reproducibility of the same secret key)
Generate the 128-bit number
Flag indicating whether the generation procedure can start
128-bit number
Transform the 128-bit number to160-bit secret key
160bit secret

APPENDIX 5

http://www.cr0.net:8040/code/crypto/sha1.php

SHA-1 Source Code

This optimized SHA-1 implementation conforms to FIPS 180-1.

sha1.h

```
#ifndef _SHA1_H
#define _SHA1_H

#define uint8  unsigned char
#define uint32 unsigned long int

struct sha1_context
{
    uint32 total[2];
    uint32 state[5];
    uint8 buffer[64];
};

void sha1_starts( struct sha1_context *ctx );
void sha1_update( struct sha1_context *ctx, uint8 *input, uint32
length );
void sha1_finish( struct sha1_context *ctx, uint8 digest[20] );

#endif /* sha1.h */
```

sha1.c

```
/*
 * FIPS 180-1 compliant SHA-1 implementation,
 * by Christophe Devine <devine@cr0.net>;
 * this program is licensed under the GPL.
 */

#include <string.h>
#include "sha1.h"

#define GET_UINT32(n,b,i)                       \
{                                               \
    (n) = ( (b)[(i)    ] << 24 )                \
        | ( (b)[(i) + 1] << 16 )                \
        | ( (b)[(i) + 2] <<  8 )                \
        | ( (b)[(i) + 3]       );               \
}

#define PUT_UINT32(n,b,i)                       \
{                                               \
    (b)[(i)    ] = (uint8) ( (n) >> 24 );       \
    (b)[(i) + 1] = (uint8) ( (n) >> 16 );       \
    (b)[(i) + 2] = (uint8) ( (n) >>  8 );       \
```

```
    (b)[(i) + 3] = (uint8) ( (n)       );       \
}

void sha1_starts( struct sha1_context *ctx )
{
    ctx->total[0] = 0;
    ctx->total[1] = 0;
    ctx->state[0] = 0x67452301;
    ctx->state[1] = 0xEFCDAB89;
    ctx->state[2] = 0x98BADCFE;
    ctx->state[3] = 0x10325476;
    ctx->state[4] = 0xC3D2E1F0;
}

void sha1_process( struct sha1_context *ctx, uint8 data[64] )
{
    uint32 temp, A, B, C, D, E, W[16];

    GET_UINT32( W[0],  data,  0 );
    GET_UINT32( W[1],  data,  4 );
    GET_UINT32( W[2],  data,  8 );
    GET_UINT32( W[3],  data, 12 );
    GET_UINT32( W[4],  data, 16 );
    GET_UINT32( W[5],  data, 20 );
    GET_UINT32( W[6],  data, 24 );
    GET_UINT32( W[7],  data, 28 );
    GET_UINT32( W[8],  data, 32 );
    GET_UINT32( W[9],  data, 36 );
    GET_UINT32( W[10], data, 40 );
    GET_UINT32( W[11], data, 44 );
    GET_UINT32( W[12], data, 48 );
    GET_UINT32( W[13], data, 52 );
    GET_UINT32( W[14], data, 56 );
    GET_UINT32( W[15], data, 60 );

#define S(x,n) ((x << n) | ((x & 0xFFFFFFFF) >> (32 - n)))

#define R(t)                                                \
(                                                           \
    temp = W[(t -  3) & 0x0F] ^ W[(t - 8) & 0x0F] ^         \
           W[(t - 14) & 0x0F] ^ W[ t      & 0x0F],          \
    ( W[t & 0x0F] = S(temp,1) )                             \
)

#define P(a,b,c,d,e,x)                                      \
{                                                           \
    e += S(a,5) + F(b,c,d) + K + x; b = S(b,30);            \
}

    A = ctx->state[0];
    B = ctx->state[1];
    C = ctx->state[2];
    D = ctx->state[3];
    E = ctx->state[4];

#define F(x,y,z) (z ^ (x & (y ^ z)))
#define K 0x5A827999

    P( A, B, C, D, E, W[0]  );
    P( E, A, B, C, D, W[1]  );
```

```
    P( D, E, A, B, C, W[2]  );
    P( C, D, E, A, B, W[3]  );
    P( B, C, D, E, A, W[4]  );
    P( A, B, C, D, E, W[5]  );
    P( E, A, B, C, D, W[6]  );
    P( D, E, A, B, C, W[7]  );
    P( C, D, E, A, B, W[8]  );
    P( B, C, D, E, A, W[9]  );
    P( A, B, C, D, E, W[10] );
    P( E, A, B, C, D, W[11] );
    P( D, E, A, B, C, W[12] );
    P( C, D, E, A, B, W[13] );
    P( B, C, D, E, A, W[14] );
    P( A, B, C, D, E, W[15] );
    P( E, A, B, C, D, R(16) );
    P( D, E, A, B, C, R(17) );
    P( C, D, E, A, B, R(18) );
    P( B, C, D, E, A, R(19) );

#undef K
#undef F

#define F(x,y,z) (x ^ y ^ z)
#define K 0x6ED9EBA1

    P( A, B, C, D, E, R(20) );
    P( E, A, B, C, D, R(21) );
    P( D, E, A, B, C, R(22) );
    P( C, D, E, A, B, R(23) );
    P( B, C, D, E, A, R(24) );
    P( A, B, C, D, E, R(25) );
    P( E, A, B, C, D, R(26) );
    P( D, E, A, B, C, R(27) );
    P( C, D, E, A, B, R(28) );
    P( B, C, D, E, A, R(29) );
    P( A, B, C, D, E, R(30) );
    P( E, A, B, C, D, R(31) );
    P( D, E, A, B, C, R(32) );
    P( C, D, E, A, B, R(33) );
    P( B, C, D, E, A, R(34) );
    P( A, B, C, D, E, R(35) );
    P( E, A, B, C, D, R(36) );
    P( D, E, A, B, C, R(37) );
    P( C, D, E, A, B, R(38) );
    P( B, C, D, E, A, R(39) );

#undef K
#undef F

#define F(x,y,z) ((x & y) | (z & (x | y)))
#define K 0x8F1BBCDC

    P( A, B, C, D, E, R(40) );
    P( E, A, B, C, D, R(41) );
    P( D, E, A, B, C, R(42) );
    P( C, D, E, A, B, R(43) );
    P( B, C, D, E, A, R(44) );
    P( A, B, C, D, E, R(45) );
    P( E, A, B, C, D, R(46) );
    P( D, E, A, B, C, R(47) );
```

```
    P( C, D, E, A, B, R(48) );
    P( B, C, D, E, A, R(49) );
    P( A, B, C, D, E, R(50) );
    P( E, A, B, C, D, R(51) );
    P( D, E, A, B, C, R(52) );
    P( C, D, E, A, B, R(53) );
    P( B, C, D, E, A, R(54) );
    P( A, B, C, D, E, R(55) );
    P( E, A, B, C, D, R(56) );
    P( D, E, A, B, C, R(57) );
    P( C, D, E, A, B, R(58) );
    P( B, C, D, E, A, R(59) );

#undef K
#undef F

#define F(x,y,z) (x ^ y ^ z)
#define K 0xCA62C1D6

    P( A, B, C, D, E, R(60) );
    P( E, A, B, C, D, R(61) );
    P( D, E, A, B, C, R(62) );
    P( C, D, E, A, B, R(63) );
    P( B, C, D, E, A, R(64) );
    P( A, B, C, D, E, R(65) );
    P( E, A, B, C, D, R(66) );
    P( D, E, A, B, C, R(67) );
    P( C, D, E, A, B, R(68) );
    P( B, C, D, E, A, R(69) );
    P( A, B, C, D, E, R(70) );
    P( E, A, B, C, D, R(71) );
    P( D, E, A, B, C, R(72) );
    P( C, D, E, A, B, R(73) );
    P( B, C, D, E, A, R(74) );
    P( A, B, C, D, E, R(75) );
    P( E, A, B, C, D, R(76) );
    P( D, E, A, B, C, R(77) );
    P( C, D, E, A, B, R(78) );
    P( B, C, D, E, A, R(79) );

#undef K
#undef F

    ctx->state[0] += A;
    ctx->state[1] += B;
    ctx->state[2] += C;
    ctx->state[3] += D;
    ctx->state[4] += E;
}

void sha1_update( struct sha1_context *ctx, uint8 *input, uint32
length )
{
    uint32 left, fill;

    if( ! length ) return;

    left = ( ctx->total[0] >> 3 ) & 0x3F;
    fill = 64 - left;
```

```
    ctx->total[0] += length <<  3;
    ctx->total[1] += length >> 29;

    ctx->total[0] &= 0xFFFFFFFF;
    ctx->total[1] += ctx->total[0] < ( length << 3 );

    if( left && length >= fill )
    {
        memcpy( (void *) (ctx->buffer + left), (void *) input,
fill );
        sha1_process( ctx, ctx->buffer );
        length -= fill;
        input  += fill;
        left = 0;
    }

    while( length >= 64 )
    {
        sha1_process( ctx, input );
        length -= 64;
        input  += 64;
    }

    if( length )
    {
        memcpy( (void *) (ctx->buffer + left), (void *) input,
length );
    }
}

static uint8 sha1_padding[64] =
{
 0x80, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
};

void sha1_finish( struct sha1_context *ctx, uint8 digest[20] )
{
    uint32 last, padn;
    uint8 msglen[8];

    PUT_UINT32( ctx->total[1], msglen, 0 );
    PUT_UINT32( ctx->total[0], msglen, 4 );

    last = ( ctx->total[0] >> 3 ) & 0x3F;
    padn = ( last < 56 ) ? ( 56 - last ) : ( 120 - last );

    sha1_update( ctx, sha1_padding, padn );
    sha1_update( ctx, msglen, 8 );

    PUT_UINT32( ctx->state[0], digest,  0 );
    PUT_UINT32( ctx->state[1], digest,  4 );
    PUT_UINT32( ctx->state[2], digest,  8 );
    PUT_UINT32( ctx->state[3], digest, 12 );
    PUT_UINT32( ctx->state[4], digest, 16 );
}

#ifdef TEST
```

```
#include <stdlib.h>
#include <stdio.h>

/*
 * those are the standard FIPS 180-1 test vectors
 */

static char *msg[] =
{
    "abc",
    "abcdbcdecdefdefgefghfghighijhijkijkljklmklmnlmnomnopnopq",
    NULL
};

static char *val[] =
{
    "a9993e364706816aba3e25717850c26c9cd0d89d",
    "84983e441c3bd26ebaae4aa1f95129e5e54670f1",
    "34aa973cd4c4daa4f61eeb2bdbad27316534016f"
};

int main( int argc, char *argv[] )
{
    FILE *f;
    int i, j;
    char output[41];
    struct sha1_context ctx;
    unsigned char sha1sum[20], buffer[1000];

    if( argc < 2 )
    {
        for( i = 0; i < 3; i++ )
        {
            sha1_starts( &ctx );

            if( i < 2 )
            {
                sha1_update( &ctx, (uint8 *) msg[i], strlen(
msg[i] ) );
            }
            else
            {
                memset( buffer, 'a', 1000 );

                for( j = 0; j < 1000; j++ )
                {
                    sha1_update( &ctx, (uint8 *) buffer, 1000 );
                }
            }

            sha1_finish( &ctx, sha1sum );

            for( j = 0; j < 20; j++ )
            {
                sprintf( output + j * 2, "%02x", sha1sum[j] );
            }

            printf( "test %d ", i + 1 );
```

```
            if( ! memcmp( output, val[i], 40 ) )
            {
                printf( "passed\n" );
            }
            else
            {
                printf( "failed\n" );
                return( 1 );
            }
        }
    }
    else
    {
        if( ! ( f = fopen( argv[1], "rb" ) ) )
        {
            perror( "fopen" );
            return( 1 );
        }

        sha1_starts( &ctx );

        while( ( i = fread( buffer, 1, sizeof( buffer ), f ) ) >
0 )
        {
            sha1_update( &ctx, buffer, i );
        }

        sha1_finish( &ctx, sha1sum );

        for( j = 0; j < 20; j++ )
        {
            printf( "%02x", sha1sum[j] );
        }

        printf( "  %s\n", argv[1] );
    }

    return( 0 );
}

#endif
```

REFERENCES

1. Jeremy Rifkin, "The Age of Access," Penguin Books, 2000.
2. John E. Canavan, "Fundamentals of Network Security," Artech House, 2001.
3. Steven Levy, "Crypto," Viking Penguin, 2001.
4. Ross Anderson, "Security Engineering," Wiley Computer Publishing, 2001.
5. R. A. Elbra, "Computer Security Handbook," NCC Blackwell, 1992.
6. I. R. Sinclair, "Dictionary of Personal Computing," Collins, 1991.
7. Ken C. Pohlmann, "Principles of Digital Audio," Fourth Edition, Mc Graw Hill, 2000.
8. Federal Bureau of Investigation (FBI), http://www.fbi.gov
9. IT Consultant Magazine, Penton Media Europe Limited, 2001.
10. C. P. Pfleeger, "Security in Computing," Third Edition, Prentice Hall, 2002.
11. M. Smith, "Commonsense Computer Security," Second Edition, McGraw-Hill, 1994.
12. W. Stallings, "Cryptography and Network Security, Principles and Practice," Prentice-Hall, 1999.
13. W. Stallings, "Network Security Essentials, Applications and Standards," Prentice-Hall, 2000.
14. S. Kent, "Encryption-Based Protection for Interactive User/Computer Communication," Proceedings of the Fifth Data Communications Symposium, September 1977.

15. R. K. Nichols, P. C. Lekkas, "Wireless Security, Models Threats, and Solutions," McGraw-Hill Telecom, 2002.

16. G. Christensen, "Wireless Infrastructure Technologies," Faulkner Information Services, 2000.

17. J. N. Pelton, "Wireless and Satellite Telecommunications," Prentice Hall, 1995.

18. "GSM Frequencies", http://www.gsmworld.com/technology/spectrum_gsm.html. July 4, 2001.

19. United States General Accounting Office, "Research Regulatory Efforts on Mobile Phone Health Issues," http://www.gao.gov/new.items/d01545.pdf

20. P. Stetz, "The Cell Phone Handbook," Aegis Pub Group, 1999.

21. "LAN MAN Standards of the IEEE Computer Society. Wireless LAN medium access control (MAC) and physical layer (PHY) specification. IEEE Standard 802.11, 1997 Edition," 1997.

22. N. Borisov, I. Goldberg, D. Wagner, "Intercepting Mobile Communications: The Insecurity of 802.11," http://www.isacc.cs.berkeley.edu/isaac/wep-draft.pdf

23. J. Walker, "Unsafe at any key size: An analysis of the WEP encapsulation", Tech. Rep. 03628E, IEEE 802.11 committee, March 2000. http://grouper.ieee.org/groups/802/11/Documents/DocumentHolder/0-362.zi%p

24. W. A. Arbaugh, N. Shankar, Y. C. Justin Wan, "Your 802.11 Wireless Network Has No Clothes," http://www.cs.umd.edu/~waa/wireless.pdf

25. Draft Supplement to Standard for Telecommunications And Information Exchange Between Systems-LAN/MAN Specific Requirements – Part 11: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Specification For Enhanced Security.

26. S. M. Bellovin, M. Merritt, "Limitations of the Kerberos Protocol," Winter 1991 USENIX Conference Proceedings, USENIX Association, 1991, pp. 253-267.

27. The Wireless Ethernet Compatibility Alliance (WECA), http://www.weca.net

28. A. K. Jain, Ruud Bolle, Sharath Pankati, "Biometrics: Personal Identification in Networked Society," Kluwer Academic Publishers, 1998.

29. Recognition Systems Incorporation, http://www.recogsys.com

30. R. E. Smith, "Authentication, From Passwords to Public Keys," Addison-Wesley, 2002.

31. J. Chirillo, S. Blaul, "Implementing Biometric Security." Wiley & Sons, 2003.

32. J. L. Flanagan, "Speech Analysis Synthesis and Perception", Springer, 1972.

33. A.M. Elgendy, L.C.W. Pols, "Mechanical versus perceptual constraints as determinants of articulatory strategy", Proceedings of the Institute of Phonetic Sciences of the University of Amsterdam, pp. 57-63, 2001.

34. W. N. Waggener, "Pulse Code Modulation Systems Design," Artech House, 1998.

35. H. Dudley, "The VOCODER", Bell Labs Rec, 1939.

36. G. Fant, "Acoustic Theory of SpeechAcoustic Theory of Speech Production", Mouton, 's-Gravenhage, The Netherlands, 1960.

37. R. K. Potter, "Introduction to Technical Discussions of Sound Portrayal," J. Acoust. Soc. Am. 18, 1946.

38. J. R. Deller, JR, J. G. Proakis, J. H. L. Hansen, "Discrete-time Processing of speech signals," Macmillan Publishing Company, 1993.

39. L. R. Rabiner, R. W. Schafer, "Digital Processing of Speech Signals," Prentice Hall signal processing series, 1978.

40. R. Linggard, "Electronic Synthesis of Speech," Cambridge University Press, pp. 29- 37, 1985.

41. T. Baer, "Observation of Vocal Fold Vibration: Measurement of Excised Larynges," *Vocal Fold Physiology,* University of Tokyo Press, pp. 119-136, 1981.

42. P. Strobach, Linear Prediction Theory: A Mathematical Basis for Adaptive Systems," Springer-Verlag, 1990.

43. Z. Govindarajulu, "Elements of Sampling Theory and Methods," Prentice Hall, 1999.

44. H. Nyquist, "Certain Factors Affecting Telegraph Speed," Bell System Technical Journal, Vol.3 pp.324, 1924.

45. H. Nyquist, "Certain Topics in Telegraph Transmission Theory," *A.I.E.E. Trans.*, v. 47, p. 617, 1928.

46. V. A. Kotel'nikov, "On the transmission capacity of 'ether' and wire in electrocommunications," In IZD. Red. Upr. Svyazi RKKA (Moscow), 1933.

47. C. E. Shannon, "Communication Theory of Secrecy Systems," Bell System Technical Journal, pp 656-715, 1949.

48. J. M. Griffiths, "ISDN Explained: worldwide network and applications technology," Wiley & Sons, 1998.

49. http://www.es.oersted.dtu.dk/~kah/31650/Documents/DAQ/DAQ-fund5.pdf

50. S. W. Smith, "The Scientist and Engineer's Guide to Digital Signal Processing," California Technical Publishing, 1997.

51. R. Wacks, "Review of 'Privacy and Press Freedom'," Blackstone Press Limited, 1995.

52. L. Rose, S. Rogers, J. Cuthbertson, "NETLAW: Your rights in the Online World," McGraw-Hill Osborne Media, 1995.

53. B. Schneier, "Applied Cryptography," Second Edition, John Wiley & Sons, 1996.

54. Notes from the MSc course in Data Communication Systems, Brunel University.

55. M. Rosing, "Implementing Elliptic Curve Cryptography," Manning Publications, 1999.

56. D. E. R. Denning, "Cryptography and Data Security," Addison-Wesley Publishing Company, 1982.

57. http://www.rsasecurity.com

58. W. Rankl, W. Effing, "Smart Card Handbook," John Wiley and Sons, 1998.

59. B. Chambers, "Octopus – The Hong Kong Contactless Smart Card Project (Contactless smart card in mass transit application, operations started October 97)." Proceedings of the CardTech/SecurTech, 1999.

60. R. J. Anderson and M. G. Kuhn, "Low cost attacks on tamper resistant devices," Proceedings of International Workshop on Security Protocols 1997 (Paris, France) (M. Lomas et.al., ed.), Lecture Notes in Computer Science, vol. 1361, Springer-Verlag, 1997, pp. 125-136.

61. RSA Security, RSASecurID, http://www.rsasecurity.com/products/securid/index.html

62. R. J. Anderson and M. G. Kuhn, "Tamper resistance-a cautionary note," Proceedings of Second USENIX Workshop on Electronic Commerce (Oakland, California), pp. 1-11, 1996.

63. O. Kommerling and M. G. Kuhn, "Design principles for tamper-resistant smartcard processors," Proceedings of USENIX Workshop on Smartcard Technology, pp. 9-20, 1999.

64. S. V. Vaseghi, "Advanced Signal Processing and Noise Reduction," Wiley & Sons, 2nd edition, 2000.

65. E. R. Davies, "Electronics, Noise and Signal Recovery," Academic Press, 1997.

66. Nuance Incorporation, http://www.nuance.com

67. IBM software-Speech Recognition, http://www-3.ibm.com/software/speech

68. Claude E. Shannon, Warren Weaver, "The Mathematical Theory of Communication," University of Illinois Press, 1949, 1963.

69. Frank Fallside, William A. Woods, "Computer Speech Processing," Prentice-Hall international (UK) Ltd, 1985.

70. A. Menezes, P. van Oorschot, S. Vanstone, "Handbook of Applied Cryptography," CRC Press, 1996.

71. R. L. Rivest, "The MD4 Message Digest Algorithm," Advances in Cryptology- CRYPTO '90 Proceedings, Springer-Verlag, 1991.

72. E. Biham, "On the applicability of differential cryptanalysis to hash functions," Lecture at EIES Workshop on Cryptographic Hash Functions, 1992.

73. B. der Boer, A. Bosselaers, "An Attack on the last two rounds of MD4," Advances in Cryptology- CRYPTO '91 Proceedings, Springer–Verlag, 1992.

74. F. Chabaud, A. Joux, "Differential collisions in SHA-0," Advances in Cryptology-CRYPTO'98 Proceedings, Springer-Verlag, 1999.

75. C.A. Deavours, L. Kruh, "Machine Cryptography and Modern Cryptanalysis," Artech House, 1985.

76. SpeedDat-E, Voice Database, http://www.fee.vutbr.cz/SPEECHDAT-E

77. American Speech-Language-Hearing Association, http://www.asha.org/index.cfm

78. Natural Language Software Registry, http://registry.dfki.de