THE FLORIDA STATE UNIVERSITY

COLLEGE OF ARTS AND SCIENCES

LEGALLY RESILIENT SIGNATURES:

A MIDDLE-AGE APPROACH TO A DIGITAL AGE PROBLEM

By

MATTHEW E. RICE

A Thesis submitted to the
Department of Computer Science
in partial fulfillment of the
requirements for the degree of
Masters of Science

Degree Awarded:
Spring Semester, 2005

The members of the Committee approve the Thesis of Matthew E. Rice defended on April 11, 2005.

Mike Burmester
Professor Directing Thesis

Breno de Mederios
Committee Member

Lois Hawkes
Committee Member

Alec Yasinsac
Committee Member

The Office of Graduate Studies has verified and approved the above named committee members.

To Paul, Sue, and Nathaniel,
my loving and supportive family.

ACKNOWLEDGEMENTS

I would first like to thank Dr. Mike Burmester, my major professor, for his continued guidance and support on this thesis. He challenged me to think through the problems on my own, but also offered assistance when I would go astray. The knowledge and experience that I have gained from him is immense, and I could not have accomplished this thesis without him.

Next I would thank my committee members Dr. Breno de Mederios, Dr. Lois Hawkes, and Dr. Alec Yasinsac. Dr. Breno has an inspiration with much of the cryptography and implementation of this thesis. He is one of the most brilliant professors that I have ever had the privilege to study under. Dr. Hawkes has been very supportive from the start and has a gift for understanding the student's needs. Dr. Yasinsac was a great source of writing and grammatical information on this thesis.

Also I thank my Pi Kappa Phi brother Ben Bush. He took personal time to sit down with me and discuss some of the more important legal aspects that my thesis dealt with. He was just a phone call away when I needed to verify anything. Thank you very much Ben.

I also want to acknowledge one of my colleagues, Eric Hokanson. He offered great insight to this problem, when he and I worked on it in the fall. In addition, he pushed me to do my best work on this thesis, and gave some very significant feedback.

Last, but certainly not least, I thank the 2005 Advanced Cryptography class that was taught by Dr. Burmester. Todd McDonald and Willard Thompson, both Ph.D. candidates, were there whenever I needed advice. Also, Todd helped coin the *legally resilient* terminology that is used throughout this thesis. To the others: Tiffany Wood, Shelly Rush, and Karolina Maneve, thank you for all your constructive comments, especially near the end of the term.

TABLE OF CONTENTS

LIST OF FIGURES

ABSTRACT

Digital signatures are essential to Internet applications and more generally electronic commerce. However, they are subject to various attacks that lend them impractical for some legal applications. In this thesis we consider technologies, which can be used for digital applications in a legally resilient way.

We introduce two signature schemes, which combine current biometric research along with digital signature schemes. In order to show the limitation of the human signature, we first introduce a scheme that has an inherent flaw we have deemed the 'fax-copy' attack. By addressing the ability to photocopy a human signature, we strengthen our scheme by allowing a human signature to be encompassed by the digital signature. The second scheme, which we entitle the 'Signet signature scheme', withstands the 'fax-copy' attack and binds a human signature to a digital signature in a legally resilient way.

CHAPTER 1

INTRODUCTION

With the advent of asymmetric or public key cryptography in 1976 by Whit Diffie and Martin Hellman [DH76], the concept of public key cryptography and in particular digital signatures was first envisioned. Two years later, Ron Rivest, Adi Shamir, and Len Adleman invented the RSA cryptosystem. With RSA both digital signatures as well as encryption can be applied to digital documents [RSA78]. Over the years many other signature schemes have been proposed. One scheme in particular, the Digital Signature Standard, was adopted as a standard by National Institute of Standards and Technology (NIST) [FIPS94]. Although there have been many schemes visited and revisited, the signing and verification processes within all of them are very similar.

Today digital signatures are successfully used in e-commerce and Internet applications, such as email. The success and integration of digital signatures in e-commerce and Internet applications is due in part of the success of the Internet. The Internet has provided a medium in which communication between parties is almost instantaneous. Individuals no longer have to leave their houses to file their taxes, to purchase items such as books, or to even transfer funds and write checks. The security of e-commerce is based on the trust that digital signatures are computationally difficult to forge.

However, there are some applications in which digital signatures are not considered adequate. These applications are generally legal in nature and have long life spans. In this thesis we propose a technology, which attempts to reduce the gap between human and digital signatures by using a dual signature scheme. A dual signature scheme is one in which two signatures, digital and handwritten, are combined in a legally resilient way that provides robustness. The *Signet Signature* is robust against *fax-copy* attacks and more general attacks which digital signatures are prone to versus handwritten signatures. When used in a secure way, the mechanism (see chapters 4, 5, and 6) combines the strengths of both the social and the digital realms. This is a first attempt at a very complex and socially relevant problem.

1.1 A Problem Informally Defined

Applications that are bounded by time or have a short life span work rather well with utilizing digital signature security. However, some documents are deemed to have a substantial worth and do not expire as quickly, as is the case with legal wills. Due to the temptation to forge for financial gain and the indefinite life span, wills do not currently have a digital signature process. Legal traditions such as physical inspection of the document and consciousness and willingfullness of the signer are aspects not currently addressed with digital signature schemes. Also, legal attacks may be introduced if there is too much reliance on heuristic security (security based on a hard problem). To attack the validity of a document, legal professionals will have to simply point out Moore's Law which states that every 18 months, computational power doubles [Int05]. With this law and the recent break of SHA-1[*] (a very common hashing algorithm) a digital document's authenticity may be called into question.

1.1.1 Problem Formally Defined

Although currently digital signatures are inadequate for some legal documents (i.e. wills), we speculate if there is a way to strengthen digital signatures so that they may be used in a legal and binding way.

1.2 A Solution

In an age where convenience drives technology, legal wills are excluded from being digitally signed and stored. We propose a solution that seeks to do just that. Through the use of our Signet Signature scheme, individuals will not only be able to digitally sign their wills but physically sign them as well. If a will is ever questioned, the digital signature can be verified and the characteristics of the handwritten signature be analyzed by appropriate individuals (graphologists).

In chapter 2 we overview previous work, ideas, and laws that have encouraged our model. Chapter 3 we overview both handwritten and digital signatures, their

[*] SHA-1 has been broken, but not cracked, by a University of China research team of Xiaoyun Wang, Yiqun Lisa Yin, and Hongbo Yu [Sch05]

respective strengths and weaknesses, and the requirements that make them legal. Chapter 4 discusses two ways in which a human signature may be linked to a digital signature. Then in chapter 5 and 6 we provide the model for our solution and a proof of security. Next in chapter 7 we offer some ways to further strengthen and expand our proposed model. Finally, we conclude in chapter 8.

CHAPTER 2

PREVIOUS WORK

Biometric authentication has been of great interest as of late. The idea is to use unique properties from individuals, which essentially turns people into keys. Examples include fingerprints, retinal scans, and voice recognition. One form of biometrics that is of interest to our scheme is a handwritten signature. Handwritten signature features are classified in to two groups: global and local. The global features are related to the overall signature, typically the signing speed and trajectory of pen strokes. Local features correspond to specific points gathered along the signature, distance from selected points and the change in curves. Due to the variance in signing speed, handwritten signatures tend to deviate from one another, even though they belong to the same individual. By using an algorithm called dynamic time warping algorithm with variant Euclidian distance [MC97, PP90, OKM00] and Hidden Markov models [VODA98] are used in such cases to align two signatures. One recent verification scheme claims that an error rate of 1.4% was achieved in detecting genuine signatures from skilled forgeries. The scheme uses a combination of a pressure sensitive tablet and pen [KY03]. Yet another approach aims at producing a system that recognizes signatures at a low computational cost [MLR99]. A combination of successful signature recognition and low computational cost may prove quite useful for our scheme.

Currently, each state in the U.S. has considered electronic signatures and has either passed or is introducing legislation concerning their use. Although they all tend to agree that electronic signatures should be as binding and legal as their traditional ancestors, handwritten signatures, conflicts concerning the methods and techniques exist [Ask01]. The purpose of enacting the Uniform Electronic Transactions Act (UETA) was to give states a uniform set of rules that would govern electronic commerce and its subsequent transactions. UETA sets three fundamental goals [NCCUSL99]:

1. A record of signature will not be denied legal effect and enforceability because an electronic record was used in its formation.

2. An electronic record will satisfy any law that requires a writing.

3. Any signature requirement in the law will be met if there is an electronic signature.

There is a belief that digital evidence systems and legislation need to be carefully looked at and revised accordingly. In a paper entitled *New Approaches to Digital Evidence* [Mau04], Ueli Maurer points out various shortcomings with the current legal system and it allowing for digital evidence, including digital signatures. He details requirements for contract signing systems as follows [Mau04]:

- Practicality—procedures must be practical and efficient.

- Unambiguity—resulting evidence should be unambiguous.

- Security—if a user has not agreed to document *d*, the risk of convincing evidence for this claim should be negligible.

- Low cost—regarding infrastructure, technology, and processes.

- Low trust requirements—need for trusted entities should be minimal.

- Precise and simple legislation—legislation should be unambiguous and simple.

- Smooth integration—into existing technical and legal infrastructure.

- Wide usability and acceptance—easy to use.

Another convincing argument that originally led us to propose our model was one that Bruce Schneier wrote on digital signatures and PKI schemes and their weaknesses when used in current electronic commerce, and other valid scenarios [ES00]. His main argument is that due to the computational requirements of digital signatures, the link between digital signing and physically signing is severed. He argues that the digital signatures themselves are not necessarily useless, but that the gap between assuming a user has seen a document and initiated the signing computer is much too large [Sch00].

> "The signer computes a digital signature of message m by computing m^e mod n. This is complete nonsense. I have digitally signed thousands of electronic documents and I have never computed m^e mod n in my entire life. My computer makes that calculation. I am not signing anything; my computer is [Sch00]."

Schneier continues to argue that due to the proliferation in malicious code such as viruses and Trojan horses, the ability to guarantee a signer has seen and willingfully signed a document is greatly diminished. "Imagine Alice in court, answering questions about a document she signed. I never saw it she says [Sch00]."

One open architecture company, CIC, has been researching ways to incorporate an online signature scheme into digital documents. They use a variety of electronic signature types including handwritten signatures, voiceprints, fingerprints, or PIN/PWD. They utilize cryptographic hashing algorithms (e.g. SHA-1) to provide some security, and their current technology keeps a permanent record to provide audit trail [CIC].

CHAPTER 3

SIGNATURES

3.1 A Brief History of Signatures

For centuries, human societies have been using some form of authentication, dating all the way back to the originators of writing, the Sumerians [Fil97]. Historically, a signature can be any mark made with the intention of authenticating or acknowledging the marked document [ABA96, ZD00]. Some societies choose intricate works of art known as seals to bind human intention to documents; other societies choose handwritten signatures for the same purpose [LIG00]. Whatever the medium used the overall goal is the same: to bind a human to an object, usually some form of documentation.

Many of our current laws and legal traditions are heavily influenced by Western history. As such, our legal system uses signatures for three vital reasons: message authentication, message or data integrity, and non-repudiation. Message authentication is primarily concerned with identity assurance [For94]. Message or data integrity is the assurance that the message or data has not been modified since the signature was appended. Finally, non-repudiation deals with providing evidence to a third-party (like a judge or jury) that a transaction was agreed upon by the parties in question; thereby protecting against false denials of participation [Fil97].

3.2 Digital Signatures—An Overview

In general, digital signatures work as follows. Suppose Alice wants to sign a document *m* to be verified by Bob, who shares a hash function and digital signature scheme with Alice. A cryptographic hashing algorithm is applied to the message *m*, which outputs a message digest *m'* that has a fixed length. A common family of cryptographic hashing algorithms is the SHA family, which varies between 160 bits for SHA-1, 256 bits for SHA-256, and 512 bits for SHA-512 [Kam04]. It is important to note that a cryptographic hashing algorithm must have three properties to ensure security [MOV97]:

1. One-way preimage resistant: given a message digest y, the equation $y = h(x)$ cannot be solved efficiently for x.

2. 2nd-preimage resistant: given a message x and the digest $y = h(x)$, the equation $y = h(x)$ cannot be solved efficiently for a 2nd-preimage $x' = x$, with $y = h(x)$.

3. Collision resistant: one cannot find efficiently a pair of distinct messages x, x' for which $h(x) = h(x')$.

Next Alice either generates or is given a pair of keys, one being her secret decryption key $\mathrm{SK}_{\mathrm{Alice}}$ and the other being her public encryption key $\mathrm{PK}_{\mathrm{Alice}}$. Note that it should be hard to compute $\mathrm{SK}_{\mathrm{Alice}}$ given $\mathrm{PK}_{\mathrm{Alice}}$. This ensures Alice's signature cannot be forged. The public key $\mathrm{PK}_{\mathrm{Alice}}$ of Alice is managed by a Public Key Infrastructure (see 3.3).

To sign a digital document m, Alice uses as input to the signature algorithm her private key $\mathrm{SK}_{\mathrm{Alice}}$ and message digest $m'= h(m)$. The output, which is the message signature $Sign_{\mathrm{Alice}}(m')$, is generally appended to the original digital message m. To verify the signature $Sign_{\mathrm{Alice}}(m')$ Bob separates the message bit string m from the signature bit string $Sign_{\mathrm{Alice}}(m')$ and hashes the message m using the same hashing algorithm h, to get $m'=h(m')$[**]. Next, Bob will input the message digest m' and Alice's public key $\mathrm{PK}_{\mathrm{Alice}}$, along with the digital signature $Sign_{\mathrm{Alice}}(m')$ into a verification algorithm. The algorithm will determine whether $Sign_{\mathrm{Alice}}(m')$ is valid or not.

[**] Note that at this point, if the message has not been modified then both Alice's and Bob's message digests, will be the same.

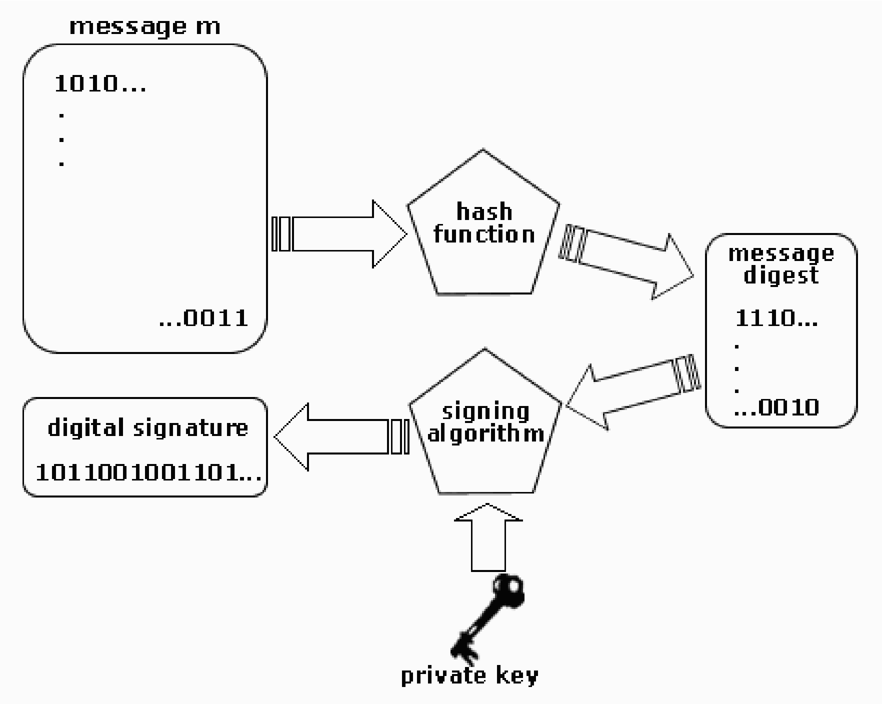

Figure 1. Digital signing process

3.3 Strengths and Weaknesses of Handwritten Signatures

Handwritten signatures do not require the infrastructure that digital signature schemes use to provide authenticity. However, they do have an important drawback, especially in today's world. That drawback is the ease of replication or duplication. Within our current society, transactions requiring handwritten signatures are done on a daily basis. Along with photocopiers, scanners, faxes, and other image copying devices, it is extremely easy to forge a document and append a photocopy of another person's signature to it.

Signature experts (graphologists) provide a trained professional opinion to third parties regarding signature forgeries. Although signatures are rarely signed the same way twice, they do adhere within certain boundaries unique to each individual [Gau04]. It is this fact that provides a basis for signature experts to identify a forgery versus an authentic signature. The uniqueness of handwritten signatures results from complex signals sent from the brain to the muscle fibers within the hands and fingers. Different pressure points and timing changes are developed and are difficult to replicate or forge dynamically [Gau04].

The act of signing a name to a document not only utilizes the unique biometrics of individuals, but also a *visual conscience*. To further explain, when Alice wants to sign a contract, say an apartment lease, there are fail-safes that are in place to protect all parties involved. For example, Alice may be required to initialize important paragraphs or pages, providing evidence to a third-party that she did in fact read the document.

3.4 Strengths and Weaknesses of Digital Signatures

Unlike traditional handwritten signatures, digital signatures cannot be in any feasibly way forged but require the computational power of computers. This unforgeability can be viewed as both a strength and a weakness. Because computers are required to compute the digital signature the human element is abstracted from the signing process. No longer does a person physically sign his or her name to a digital document, but rather has a computer perform the signing act on their behalf using their signing key. The security of the digital signature schemes lie within the series of procedures working together perfectly. When done correctly, digital signatures can provide non-forgeable bit strings that can be used to seal a digital document. The human recognition of bit strings is more difficult than the human recognition of handwritten signatures. This difficulty coupled with the abstraction provided by the signing computer, hinders the ability to properly bind any human element to a digital signature; thus creating a gap between the digital signature and the human person.

We live in an age where malicious code such as worms and Trojan horses exist. So, how can one be sure that Alice herself initiated the signing process? Following this idea further, we are not even guaranteed that Alice even saw the document at all. Any signing computer that is compromised could theoretically sign any document with the person having little to no knowledge of the act. At the same time, the document would be valid and conform to the standards that are currently in place [Sch00]. Both message authentication and non-repudiation are called into question in such instances. In an attempt to solve this problem digital signature schemes utilize key management and trust infrastructures. Public key infrastructures (PKI's) consist of a certification authority (CA), a registration authority (RA), a certificate repository (CR), a certificate registration list (CRL), public and private keys of users, and lastly digital certificates. The CA is an important trusted party who signs and issues certificates for users and the RA may help the CA in some of its tasks. Digital certificates store the information from the CA, RA, and the keys in one structure, which in turn are stored in CRL's and the filed away in a CR [Cis03].

To strengthen the PKI models and to reduce the risk of security vulnerabilities, key distribution must be made perfectly clear. How will the private keys be distributed to the correct individuals? How will access to the public keys be made available? How are the public keys linked to their owners? All these questions have to be addressed accordingly. Impersonation attacks are a possibility in public key infrastructure schemes; again the anonymity that is given by the computer is primarily at fault for allowing these attacks. For example, Eve can register herself for a set of keys and pretend that she is Alice. If the certificate authority or manager of the keys does not properly check Eve's true identity, then Eve will be assumed as Alice in the system.

In addition, digital signatures do not age well. The security of a digital signature scheme relies on the hashing algorithm properties (see 3.2) and the key size being large enough to prevent brute-force attacks. However, using Moore's Law as a reference we already know that processing power doubles about every 18 months. Keeping this in mind, for a key to be secure from brute-force attacks in the year 2025, it is estimated that a key size of 20,000 bits[***] is needed [BFS91]. So documents signed and sealed today, may be broken and forgeable in twenty years. This is a great cost for important documents such as wills.

3.5 Legality of Signatures

3.5.1 Requirements that Bind a Signature to a Document

Traditionally there are requirements that properly bind a signature to a legal document. Failure to adhere to the following can result in the nullification of a document:

[***] Ron Rivest never specified in the actual proceedings, but later in a general forum discussion the number of bits.

1. The signer produces a document *m* [contract authenticity].

2. The signer signs the document in a conscience and willing act [willingfullness],

    a. The signer is of legal age,
    b. The signer is deemed mentally capable [conscience],
    c. The signer is aware of the content contained within the document [conscience].

3. The document is verified and key terms are initialized [contract authenticity, conscience and willing],

4. The document is signed by the correct intended individuals [signature authentication, contract authenticity],

5. The signer has one or more witnesses that would also sign the document [contract and signature authenticity].

6. A notary notarizes the document for authenticity [contract and signature authenticity].

The first four requirements are the more general requirements for documents that require a handwritten signature. Due to the generality of the first four requirements, they traditionally are first to be challenged in legal actions. Wills are considered the most rigorous legal documents because they require not only the signer, but also two witnesses to sign the document all within the presence of one another. In addition, a notary must notarize the document. Due to their nature, wills incorporate all the aspects listed above and can be considered a subset of any legally binding document. Therefore any document that adheres to the format of a will will also be considered legally binding.

3.5.2 Legally Resilient – An Informal Definition

We want to focus on strengthening the link between Alice and her signing key SK_{Alice}. With this in mind, we formally define *legally resilient* to mean any signature scheme that not only upholds message authentication, message integrity, and non-repudiation but also includes a physical element of the signer in the signing process.

3.5.3 Legal requirements of digital signatures

When trying to make a digital signature scheme stronger, one must rely on analyzing earlier models and analyzing their weaknesses. Also, when developing a scheme for such strong legal actions and contracts such as wills, minute details cannot be overlooked; failure to do so can offer devastating loopholes. Bearing this in mind, we offer a detailed list of requirements to gain legal resilience.

[Legal Aspects]

1. Contract authenticity—was the document signed the version that was meant to be signed by Alice?

2. Willingfullness—did Alice see the document and/or sign the document in a conscience and willing manner?

3. Coercion—was Alice told what to write or say?

4. Pre/Post Amendments—was an amendment made before or after the original document was signed?

5. Signature authentication—is the signature on the document Alice's authentic signature?

[Cryptographic Aspects]

6. Key protection—was the signing key (private key) of Alice compromised in any way.

7. Secure hashing security—does the hashing algorithm used have any weaknesses?

8. Computer security—how secure is the signing computer? Could a virus or malicious code compromise the security? Can a document containing malicious code transmit its code into the signing computer?

9. Certificate Authority creditability—is the CA trusted? How did it become trusted? Are Alice's certificates hers or could someone impersonate Alice and be given valid public/private key pairs?

10. Security against adaptive attacks—using an adaptive chosen message attack, could a history be built up and a valid signature gained and used?

CHAPTER 4

LINKING HANDWRITTEN AND DIGITAL SIGNATURES

4.1 Putting the human factor back in

The weaknesses of traditional signatures are primarily due to their ease of replication in today's digital age. However, this is not the case with digital signatures. The strength of digital signatures lies within the fact that they are not directly forgeable. By this we mean that if the certificate authority and trust scheme has been set up correctly and all identities have been certified, then an adversary Eve cannot genuinely sign a document with Alice's secret key, and thus impersonate Alice. However, removing the human element by the use of computers, viruses and other malicious code attacks significantly weakens digital signature models. So the real question is how can the human element be tied back into the digital signing process?

We believe that the solution resides in the combination of the strengths of handwritten signatures with the strengths of digital signatures. Handwritten signatures have human recognition, are biometrically difficult to forge, and are bound to a human by muscle conditioning. Digital signatures are computationally difficult to forge and act as a seal to the digital data contained within. By combining handwritten signatures with digital signatures we believe that the weaknesses of each can be handled and properly dealt with. The following are two solutions exploring methods that attempt to solve this dilemma.

4.2 Special Hardware

4.2.1 Smart Cards

Within both models (see 4.2.2, 4.2.3) specialized hardware will be implemented. Smart cards, similar to those currently in use across college campuses, would contain the person's secret signing key, a small picture of the handwritten signature (preferably a space saving format such as gif), along with a biometric fuzzy signature all three of which would have to reside on a protected area of the card, limiting the chance to tamper with or defraud the information.

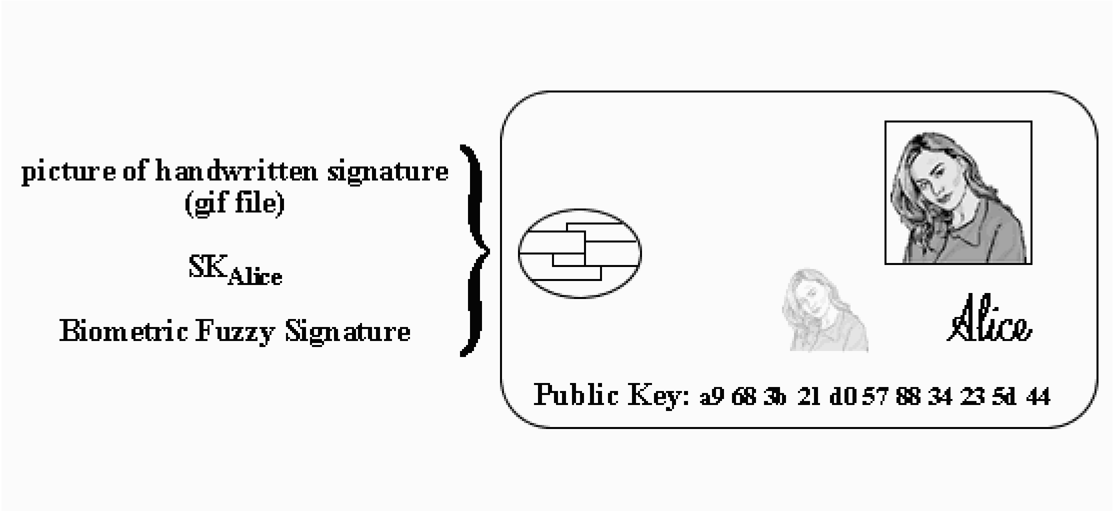
picture of handwritten signature
(gif file)
SK_{Alice}
Biometric Fuzzy Signature
Alice
Public Key: a9 68 3b 21 d0 57 88 34 23 5d 44

Figure 2. Example of smart card

4.2.2 Tamper-resistant Signing Pads

A secure signing pad (SP) that is similar to one used in department stores and notaries will be attached to the signing computer. The SP will have access to two specialized algorithms; one algorithm will be a biometric signature recognition algorithm, which is currently being researched and refined. [KY03] The other algorithm will be the signing algorithm that will be used to digitally sign the document. It is necessary for the SP to have these two algorithms so that the SP's can be as generalized as possible, therefore actually having the ability to be produced.

4.2.3 Trusted Display and Input Computer

Along with the smart cards and signing pad, a trusted display screen and input computer are needed. Both of which would be very limited on the hardware and software installed. This is necessary to reduce the possibility of malicious code attacks from entering the system. If malicious code is allowed to enter the system, then the trust is compromised. The trusted display would simply be an output device displaying the document in need of being signed. The input computer being limited on hardware would simply take as input a PDF file and output it to the trusted display. The software would be limited to include Adobe Reader, a sufficient software program in reading the inputted PDF documents. Choosing PDF versions for documents make the most sense when dealing in the applications that these schemes hope to be used for. For one, PDF's have a level of cryptographic security that other programs like Microsoft's Word documents do not have [Cal01]. Secondly, and most importantly, PDF documents unlike Word documents rarely contain malicious code (only two have been known to exist) [Pdf]. Furthermore, the malicious code in PDF's is not executed when Adobe Reader is used [Sha01, Cal01]. So, even if a document being signed does have malicious code, it will not weaken the signing methods.

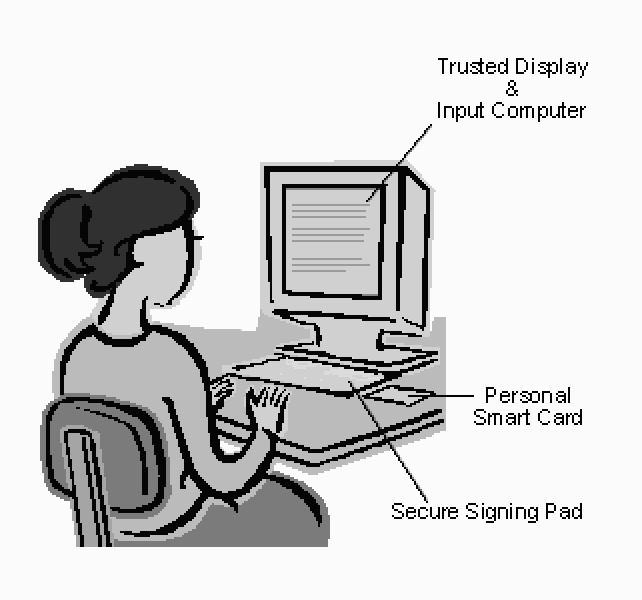

Figure 3. Overview of system

4.2.4 Trust Assumptions

We do assume that our three devices (the secure signing pad, the trusted display and input computer, and the smart card) all are tamper-resistant or employ some tamper-resistant technology. However, just employing tamper-resistant technology is not enough. We realize that there are attacks that can be used if careful analysis is not done.

First, an attacker can encompass the trusted hardware with a similar looking untrusted display. If the attacker pre-loads a document *m'* that they want signed, then there is a possibility that the valid signer would not know they are signing the wrong document *m'*, as shown in Figure 4.

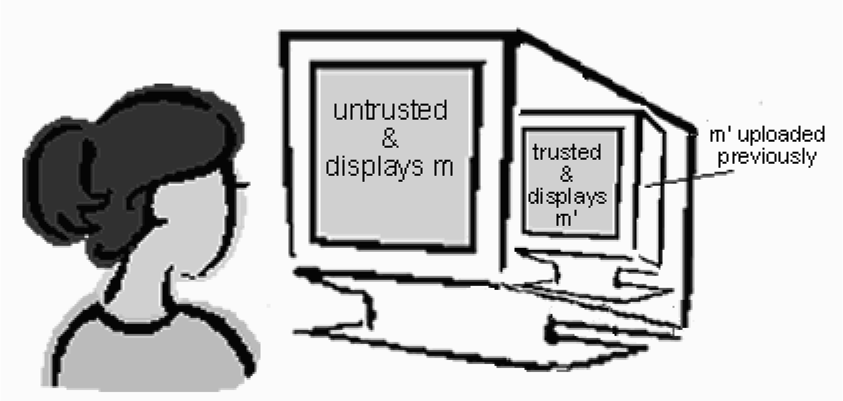

Figure 4. Trusted display within an untrusted display. Alice would believe that she is signing message m, when indeed she would be signing message m'

This can be remedied in a similar manner to how ATM's are secured, by shutting down if fraudulent activities are discovered [ATM]. In addition, if the signing pad and trusted display were manufactured into one device, then the difficulty of putting an untrusted display with the same signing pad would inevitably increase, see Figure 5. Tablet PC's and personal digital assistants (PDA's) are some examples where the display and the signing pads have been merged.

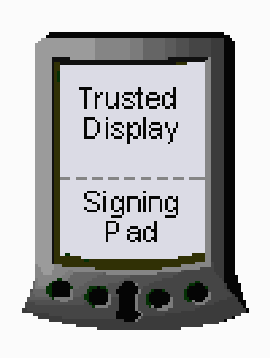

Figure 5. Trusted display merged with signing pad

Secondly, a persistent attacker with enough resources may be able to forge the various hardware pieces, and successfully deceive the signer into using an untrusted device. This would be extremely hazardous, especially with information capturing devices already available. At a bare minimum, we want to ensure that the SP device would authenticate itself to the smart card. By forcing the SP device to authenticate itself to the smart card, the signer can be assured that the device is indeed trusted. To further the authentication process, the signing computer and trusted display will also authenticate themselves to each other. If one is ever absent of the other, the signing process will not work.

All the hardware can benefit by using a type of case-intrusion detection that can employ a physical tamper-resistance [Scea]. A simple example would be proprietary emblems that seal the casing similar to warranty seals that are currently in some electronic devices. It may be important that the seals be visible so that a valid signer can detect a device that has been compromised.

4.3 Two Methods to Bind

4.3.1 A First Attempt to Bind a Digital Signature to the Signer

To acquire a legally resilient digital signature Alice inputs her document *m* (i.e. a pdf file) into the signing computer. The device has a screen that will display the document *m*, allowing Alice to verify that the document *m* is indeed the correct document to be signed. Along with the document, Alice also supplies her own personal smart card (SC), which contains the three specific data segments mentioned above (gif file, fuzzy signature, and SK_{Alice}). The signing pad (SP) has a smart card reader which Alice inserts her SC into the secure signing pad and authenticates herself by signing the SP. The SP would take Alice's dynamic hand signature and input it into the biometric algorithm that it has access to. The biometric algorithm will output a bit string which will then be compared to her fuzzy signature bit string that resides in some protected read-only memory of the SC. It is important that the handwriting signature algorithm within the SP correspond to the algorithm originally used when Alice purchased her SC, thereby reducing the error rate that could potentially arise from different signature recognition algorithms being used. If Alice fails to correctly authenticate herself within a standard error ε, then the system would destroy her SC or simply lock her out; either way the system would prevent Alice from digitally signing any documents.

Once Alice is authenticated, Alice gives the command to the signing computer to attach her digital signature. The SP would use Alice's signing key SK_{Alice} (also residing in some protected read-only area of the SC's memory) and use that in the digital signature algorithm to create a message signature as described earlier (see 3.1). Once the digital signature is created it is appended to the message, much in the same way current schemes do. Next, the pictorial representation of Alice's handwritten signature (also residing on the SC, protected) is appended; encapsulating the entire message that was just signed, see Figure 2.

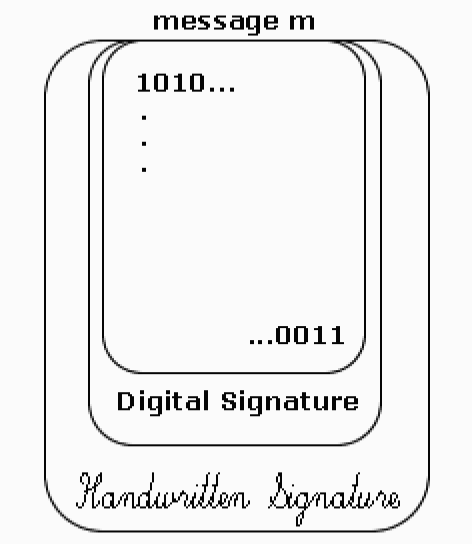

Figure 6. Handwritten signature encapsulating digital signature

By attaching a picture of Alice's handwritten signature to the document, a great advantage is gained that currently does not exist in other schemes. If document *m* is ever called into question, not only can the document be verified to contain Alice's digital signature using the verification algorithms, but physical verification can be done as well. Anyone having access to the document *m* could simply print out a hard copy format and see that Alice's signature resides on the document. This would allow for third-party individuals to actually see a signature rather than a long, alien looking string representing the digital signature.

Even though this seems like a good attempt to bind a human signature to a digital signature, it does have a major flaw. That flaw being something we have named the 'fax-attack'. Because the handwritten signature encapsulates the digital signature and document, anyone having access to Alice's handwritten signature would be able to impersonate her, allowing for all the current arguments to again rise. With the advent of the photocopier and fax machine, it is much simpler to impersonate individuals and their documents. A great example of this would be faxing a document to another individual. If a previous history of CEO Bob's signature has been obtained, then a simple document can be created with a cut and paste of Bob's signature. After faxing the document, the receiver would generally never know the difference unless of course they double-checked it with Bob himself. However, what if Bob was not available to question, much like the case of legal estate wills. When a will is called into question the creator of the will has passed on, so the third-party cannot question the originators intentions.

4.3.2 A Second Attempt to Bind a Digital Signature to the Signer: Signet Signatures

Similarly to the previous model, Alice inserts her document *m* and verifies that it is the document that she intends to digitally sign. Again, Alice authenticates herself to the SP by inserting her SC and signing her name. The algorithm that resides on the SP will once again take Alice's dynamic signature as input and output a bit string to be compared to fuzzy signature that resides on Alice's SC. If the bit strings fail to equal each other, then Alice is rejected from the system and prevented access to the digital signing process. If the bit strings do equal each other then Alice has properly identified herself and the signing computer will begin the process of digitally signing document *m*.

Unlike the first model, the digital signature is not immediately appended to the document. Instead, Alice's handwritten signature picture would first be inserted to the document *m* as figure 7 illustrates.

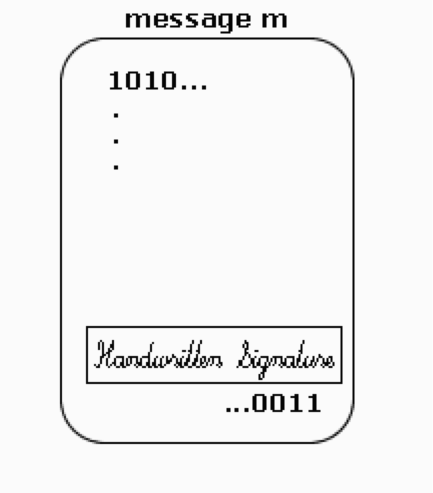

Figure 7. Handwritten signature inserted into document before digital signature is applied

Once the picture of Alice's handwritten signature has been appended, the document can be digitally signed. The message, which now includes a picture of Alice's handwritten signature, will be hashed and then inputted into the signature algorithm. The output would be the digital signature, which then is appended to the document, thus encapsulating and essentially sealing the document.

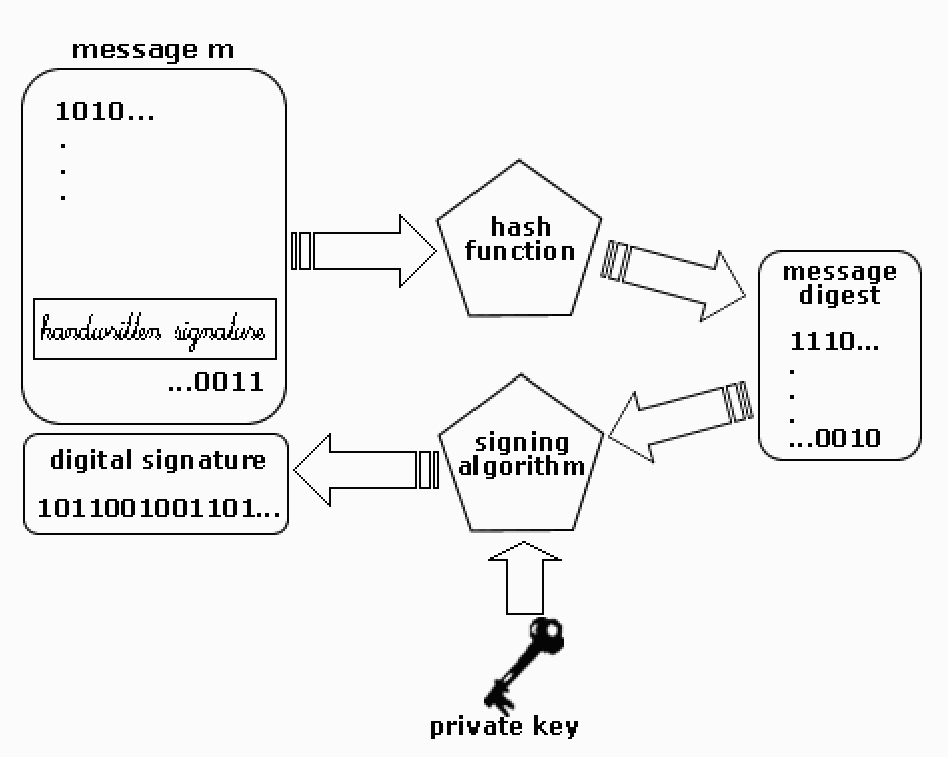

Figure 8. Digital signature encapsulating an appended handwritten signature and document

This method addresses the fax-copy attack that Method 1 has. No longer can an attacker utilize Alice's publicly attainable handwritten signature to impersonate her. If the digital signature is verified to be Alice's and there is a picture of Alice's handwritten signature, we can be assured that Alice was aware of the document and she initiated the signing process. The Signet Signature Model combines the strengths of both handwritten signatures and digital signatures, much in the way that royalty sealed documents with wax and signet rings in the Middle-Ages to prevent tampering. Once again hard copies could be printed out allowing for human comparison much in the same way as the first model. In addition, the digital signature prevents tampering to the document. As with any digital signature scheme, if any part of the document changes, which now includes the data of the handwritten signature, the entire document will fail the verification process.

CHAPTER 5

A SECURITY MODEL FOR LEGALLY RESILIENT SIGNATURES

We consider a model that utilizes a witnessing oracle, $\mathcal{O}_{\text{witness}}$. Given any document *m* and a digital signature scheme $\mathcal{O}_{\text{witness}}$ witnesses the following processes:

1. Alice produces document *m* in some secure format (i.e. pdf).
2. Alice is cognizant of the content of the document.
3. The version of the document presented is indeed the correct version.
4. Alice signs the document using her handwritten signature.
5. Alice's public key is PK_{Alice}.
6. The document is digitally signed using Alice's key.

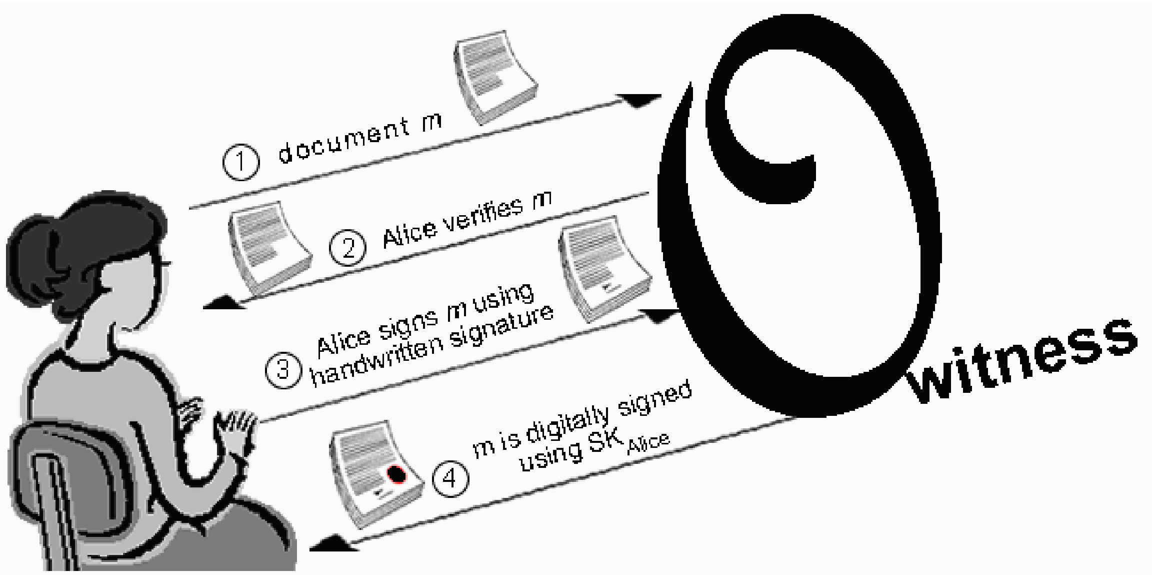
① document m
② Alice verifies m
③ Alice signs m using handwritten signature
④ m is digitally signed using SK Alice
witness

Figure 9. Oracle model for legally resilient signatures

We say that the signing method produces a *legally resilient signature* if for any document *m* that Alice produces,

$$\mathcal{O}_{\text{witness}}\,(\text{Alice}, m, \text{Alice}_{\text{cognizant of } m}, \text{Alice}_{\text{hand signing process}}, \text{PK}_{\text{Alice}}) = 1$$

That is, a *legally resilient signature* models a legally binding handwritten signature in which Alice has both physically and digitally signed a document in a conscience and willing manner. We say a signature scheme is *legally resilient* if it combines a handwritten signature with a digital signature in such a way that it is validated by $\mathcal{O}_{\text{witness}}$

CHAPTER 6

A SECURITY PROOF

In the previous section we proposed a security model for signature schemes. We shall use this model to design a signature scheme, the *Signet Signature*, which is legally resilient. Signet signatures are bases on a tamper-resistant black box tablet, which can input and display documents and have the ability to read smart cards, much in the same way that ATM's provide tamper-resistance.

6.1 An overview of Signet Signatures

During the initialization phase, Alice will apply to some trusted party (i.e. CA) for her own smart card, in which she will present proper identification (i.e. photo id, social security card) identifying her as Alice. In addition, Alice will sign on a signature pad her handwritten signature. A biometric algorithm that is contained within the signature pad will compute Alice's fuzzy signature, which will then be imprinted on a secure read-only area of her smart card. Along with the fuzzy signature a small gif file picture of Alice's signature is imprinted onto the card. Lastly, the signing key SK_{Alice} will be imprinted onto the smart card.

When Alice wants to sign a document she will present to the tamper-resistant tablet her smart card and the document *m*. When Alice inserts the smart card, an authentication process will occur. Alice will be asked to sign her name on the tablet. Using the same biometric algorithm that was used to initialize the smart card, Alice's signature will verify against the bit string that resides on the read-only memory of the smart card. Upon receiving the document *m*, the tablet will verify that the document *m* is in a secure format. Alice will then verify the document *m* by reading paragraphs of the document and signing her signature, which will be imprinted into the document. When the entire document has been verified, Alice will sign the document once more at the end. The overall process and the individual processes are all timed; only upon completion of the processes within the allotted time may Alice give the command to have the document digitally signed. To digitally sign, the tablet will be provided access to the smart card, which contains the signing key of Alice SK_{Alice}.

> **Theorem 1.** The signet tablet is a legally resilient signature scheme and produces legally resilient signatures.

6.2 The Proof of Theorem 1

To prove that Signet tablet signatures are legally resilient we must show that an enemy Eve cannot validly sign a new message. Assuming that the tablet is tamper-resistant and contains the necessary secure hardware and software, we claim that our scheme is just as secure as using the $\mathcal{O}_{\text{witness}}$ model outlined previously.

1. The signer cannot introduce malicious code because the file, hardware, and software are all secured against such attacks,

2. The signer has read the document *m* and is conscience and willing to sign the document *m*. Eve cannot claim that she never saw the document. This follows from the fact that the signer has inputted a document and verified to the tablet that *m* is indeed the correct version,

3. The signer is indeed the owner of the smart card. Eve cannot steal Alice's smart card and sign with it. This follows from:

    a. The only acceptable input to the signing device is a dynamic handwritten signature and a smart card,
    b. Having the tablet ask for the signer to sign multiple times (signing paragraphs and the end of the document),
    c. Having the smart card maintain the fuzzy signature on a protected read-only portion of memory,
    d. Having the tablet check every time a signature is inputted that it matches with the fuzzy signature.
    e. Having the signer locked out if more than a threshold *t* of incorrect signatures is inputted.

4. Eve's identity is known to the system, she may not impersonate Alice. This follows from ensuring that the initialization steps were followed, and the tablet has access to verify that the signer's public key is indeed PK_{Signer}.

5. Eve is limited on time between processes because the tablet has a timeout mechanism. If a threshold *t* time is exceeded between processes, then the signer is locked out.

6. The tablet only signs using the signing key SK_{Signer} of the signer. This follows from the fact that the tablet does not have any access to another signing key other than the one that resides on the read-only memory of the inputted smart card.

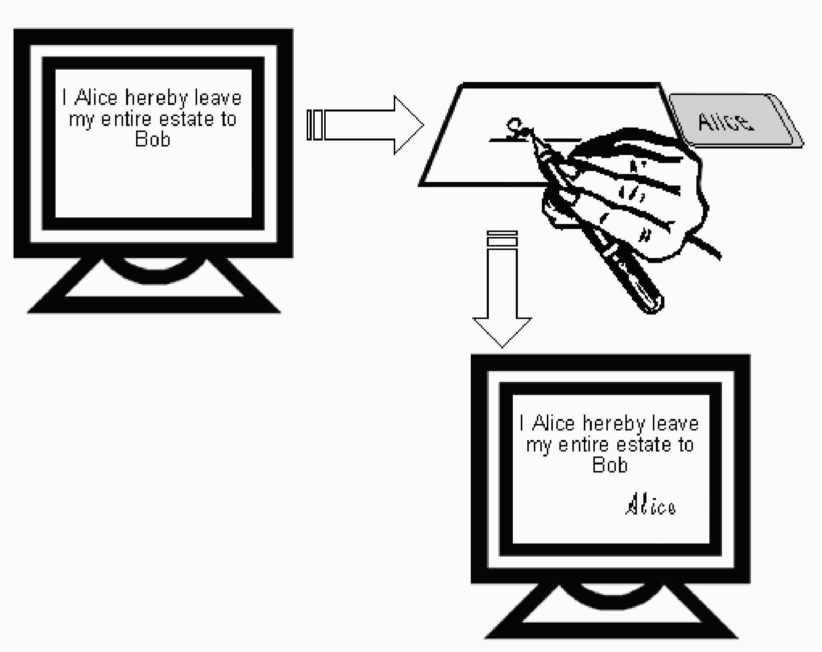
I Alice hereby leave my entire estate to Bob
Alice
I Alice hereby leave my entire estate to Bob
Alice

Figure 10. Alice verifying that she read the paragraph that is displayed by signing her name. This concept follows from initializing paragraphs in current legal documents such as leases

CHAPTER 7

FURTHER STRENGTHENING AND EXPANDING SIGNET SIGNATURES

We do realize that the weak point of the tablet lies within the implementation of the signing pad. If the signing pad holds too much information (i.e. a copy of the handwritten signature) than replay attacks are possible. It is our intention that the black box tablet contains no more memory than is absolutely necessary to properly implement the protocol.

7.1 Strengthening the Digital Signature Component

To further expand and secure the Signet signature scheme, the digital signature chosen should be secure against adaptive chosen message attacks. There are several such schemes (e.g. Goldwasser-Micali-Rivest [GMR1988], Cramer-Shoup [CS00], Merkle [MOV97]). Here we describe one such scheme, the Merkle one-time signature scheme [MOV97]. The signature of document *m* is generated by:

1. Computing c, the binary representation of the number of 0's in *m*.
2. Form w = concatenating m with $c = (a_1a_2\ldots a_t)$.
3. Determine coordinate positions $i_1 < i_2 < \ldots < i_u$ in w such that $a_{ij} = 1, 1 \leq j \leq u$.
4. Let $s_j = k_{ij}, 1 \leq j \leq u$.
5. Signature for m is $(s_1, s_2, \ldots, s_u)$.

7.1.1 Forward Security

The Signet Signature scheme, as described, is no stronger than its digital signature component. That is to say, if at some point in time it becomes feasible to break the digital signature scheme used, then it becomes feasible to forge the Signet Signature (by using the fax-attack). In an attempt to elevate this attack we can implore the use of Merkel Hash Tree authentication.

Once the document *m* has been digitally signed, it can be inserted into a structure called a Merkel Hash Tree (MHT). We signify the Merkle one-time signature on *m* as *Sign*(*m*). The signature on vertex R_0 is of a trusted third party (TTP). The MHT signature on *Sign*(*m*) will contain a path that can be later extracted to validate the authenticity of the document, or node. If the path of the MHT signature leads to the public root node (R_0), then the document is considered valid; otherwise the document is considered invalid [MOV97]. It is important to note that the hashing algorithm used must adhere to the properties described earlier (see 3.1). MHT's have a nice property; the storage of *n* documents requires only h^2 and can be traversed at a cost of *h*, where *h* is the height of the tree [MOV97].

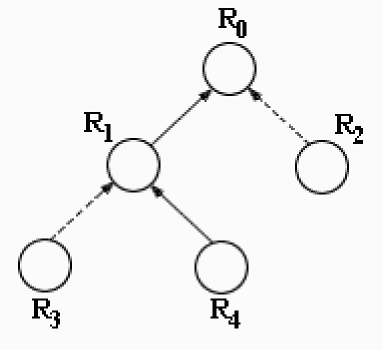

Figure 11. Authentication tree for Merkle one-time signature scheme

We can use the authentication provided by MHT's to expand and provide an amendment scheme. We denote a legally amended document as m_i'. A new signature can be generated on m_i' and inserted into the MHT as a new node. The authenticity can be validated the same way, by hashing until the root node is reached. If two valid documents are challenged, the document residing further in the authentication tree would be given precedence.

However, even implementing the MHT has one critical weakness. At some point in time it may become feasible to break the hashing algorithm used, as such is the case currently with the SHA-1 hashing algorithm. If this were to happen, then it is conceivable that a forged document *m'* may be inserted into the MHT and validly hash to the public node. To address this issue, the MHT may be best implemented within a trusted repository (much in the same way that deeds to houses and birth certificates are stored). The storage space would still be minimal (only h^2), and the security would be gained by using the trusted repository.

7.1.2 Time Stamps

In addition, using a timestamp imprinted within the document at the time of digitally signing could provide useful information to a third-party. For example, if a person is deemed mentally incapable at a certain point in time, documents legally amended may be subject to the timestamp within. This would further add strength by imploring the review and judgment of the human third-party.

7.1.3 Hybrid Signatures

The solution we propose to our legally resilient signature is to use "dual signatures", in which a handwritten signature is coupled with a digital signature in a legally resilient way. This is a first approach to resolving our problem, which in many respects is a fundamental requirement for our social functioning. What is truly needed is a hybrid signature that binds both aspects (social and digital) in one secure entity.

7.1.4 Scope

We realize that the scope of the problem may need to be narrowed a bit. It may never be feasible to include all types of wills, including multi-million dollar documents. Maybe the solution resides in determining an acceptable amount of terms and conditions that the will contains. Thereby reducing the risk of attack, making it cost ineffective to an enemy.

In addition, current wills require at least two witnessing signatures as stated previously. However, our Signet Signature scheme does not directly address this issue. We propose that if a signature can be deemed legally resilient for one party, then it is conceivable the signature scheme may be extended to include multiple parties.

CHAPTER 8

CONCLUSIONS

We have proposed two schemes in which both merge a traditional handwritten signature into currently implemented digital signature schemes. The first method was shown to have a major flaw in its design, yielding it susceptible to current *fax-attacks*. The second model in which we have entitled the *Signet Signature* which pulls its inspiration from the Middle-Age Era when Kings or other high ranking officials had to contact and authenticate themselves via parchment paper. Much in the same way as signet rings behave; the digital signature seals the data thus providing data integrity. With a picture of Alice's handwritten signature now part of the data, traditional legal actions can be taken to provide a stronger basis for authentication and non-repudiation. The *Signet Signature Model* reduces a person's ability to forge or perform any illegal act regarding documents, which require a legal signature.

We realize that our solution is not a perfect solution to the problem proposed. In an ideal solution, a hybrid signature would be used to securely bind both social and digital aspects together. We also realize that users could abuse the mechanism described, the black box technologies should make this more difficult. In closing, this is a first approach to resolving our problem. A problem that is not only challenging but also fundamental and open-ended.

REFERENCES

[ABA96] *Digital Signature Guidelines: Legal Infrastructure for Certification Authorities and Secure Electronic Commerce*, American Bar Association, Chicago, Illinois, 1996.

[Ask01] R. Askew, *Electronic Signatures and the E-Sign* Act, International Legal Technology Association, 2001.

[ATM] http://www.atmmarketplace.com, accessed April 6, 2005.

[BFS91] T. Beth, M. Frish, G.J. Simmons, *(Eds) Report: Public Key Cryptography, State of the Art and Future Directions*, EISS Workshop, Overwolfail, Germany, July 1991.

[Cal01] H. Calabia, *New virus spreads using Acrobat files*, IDG News Service, Buenos Aires Bureau, written 2001, http://www.belcart.com/it_news/eng/2001_08_07_Virus_PDF.htm, accessed February 28, 2005.

[CIC] CIC | Enterprise Solutions, http://www.penop.com/enterprise/legal, accessed January 25, 2005.

[Cis03] *Security+ Guide to Network Security Fundamentals*, Cisco Learning Institute, Course Technology, 1 edition, 2003.

[CS00] R. Cramer, V. Shoup, *Signature Schemes Based on the Strong RSA Assumption*, ACM Transactions on Information and System Security (ACM TISSEC), 3(3) pp. 161-185, 2000.

[DH76] W. Diffie, M. E. Hellman, *New Directions in Cryptography*, IEEE Transactions on Information Theory, IT-22(6), pp. 644-654, November 1976.

[ES00] C. Ellison, B. Schneier, *Ten Risks of PKI: What You're not Being Told about Public Key Infrastructure*, Computer Security Journal, Vol XVI, number 1, 2000.

[Fil97] D. Fillingham, *A Comparison of Digital and Handwritten Signatures*, Paper for MIT 6.805/STS085: Ethics and Law on the Electronic Frontier, 1997.

[FIPS94] *Digital Signature Standard (DSS)*, Federal Information Processing Standards Publication 186, 1994.

[For94] W. Ford, *Computer Communications Security, Principles, Stand Protocols and Techniques*, pp 109, Prentice Hall, Englewood Cliffs, NJ, 1994.

[GMR1988] S. Goldwasser, S. Micali, R. Rivest, *A digital signature scheme secure against adaptive chosen message attacks*, SIAM J. Computing, 17(2), pp 281-308, April 1988.

[Int05] *Moore's Law 40th Anniversary*, http://www.intel.com/technology/silicon/mooreslaw/, accessed April 6, 2005.

[Kam04] A. Kaminsky, *Cryptographic One-way Hash Functions*, Rochester Institute of Technology, http://www.cs.rit.edu/~ark/lectures/onewayhash/onewayhash.shtml written February 17, 2004, accessed April 2, 2005

[KY03] A. Kholmatov, B. Yanikoglu, *Biometric Authentication Using Online Signatures*, Sabanci University, Tuzla, Istanbul, Turkey, 2003.

[LIG00] B. Lutterbeck, K. Ishii, R. Gehring, *Governing Legal Identities Lessons from the History of Seals and Signatures*, Technical University of Berlin, Department of Computer Science, 2000.

[Mau04] U. Maurer, *New Approaches to Digital Evidence*, Proceedings of the IEEE, Vol. 92, No. 6, pp. 933-947, June 2004.

[MLR99] J.C. Martinez, J. Lopez, F.J. Luna Rosas, *A low-cost system for signature recognition*, Circuits and Systems, 42nd Midwest Symposium, Vol. 1, pp 101-104, 1999.

[MOV97] A. Menezes, P. van Oorschot, S. Vanstone, *Handbook of Applied Cryptography*, pp 323-324, CRC Press LLC, Boca Raton, Florida, 1997.

[MC97] R. Martens, L. Claesen, *Dynamic programming optimization for on-line signature verification*, ICDAR97, 1997.

[NCCUSL99] National Conference of Commissioners on Uniform State Laws – Introductions Adoptions of Uniform acts, adopted 1999, http://www.law.upenn.edu/bll/ulc/fnact99/1990s/ueta99.htm, accessed March 2, 2005.

[OKM00] T. Ohishi, Y. Komiya, T. Matsumoto, *On-line signature verification using pen-position, pen-pressure and pen-inclination trajectories*, ICPR, Vol IV, pp 547-550, 2000.

[PP90] M. Parizeau, R. Plamondon, *A comparative analysis of regional correlation, dynamic time warping and skeletal tree matching for signatures*, IEEE Trans. Pattern Analysis and Machine Intelligence 12, pp 710-717, 1990.

[Pdf] PDFZone.com, http://www.pdfzone.com/news/101739.html, accessed January 25, 2005.

[RSA78] R. Rivest, A. Shamir, L. Adleman, *A Method for Obtaining Digital Signatures and Public-Key Cryptosystems*, Communications of the ACM, 21(2), pp. 120-126, 1978.

[Scea] Warranty Information, Sony Electronic Arts of America, http://www.us.playstation.com/support.aspx?id=warranties, accessed April 5, 2005.

[Sch00] B. Schneier, *Why Digital Signatures Are Not Signatures*, written 2000, revised 2004, http://www.hipaadvisory.com/tech/whynot.htm, accessed September 25, 2004.

[Sch05] B. Schneier, *Cryptanalysis of SHA-1*, written 2005, http://www.schneier.com/blog/archives/2005/02/cryptanalysis_o.html, accessed April 1, 2005.

[Sha01] S. Shankland, *New virus travels in PDF files*, CNET News.com, http://news.com.com/2100-1001-271267.html, published August 7, 2001, accessed January 25, 2005.

[VODA98] J. Van Oosterhout, H. Dolfing, E. Aarts, *On-line signature verification with hidden markov models*, ICPR, 1998.

[ZD00] Jianying Zhou, Robert Deng, *On the Validity of Digital Signatures*, ACM SIGCOMM Computer Communications Review, Vol. 30, issue 2, April 2000.

BIOGRAPHICAL SKETCH

Matthew E. Rice was born in Bennington, Vermont on September 17, 1981. He spent most of his childhood enjoying sports such as baseball, soccer, bowling, downhill skiing, and golf. In the early 1990's he battled and defeated a rare cancer. After graduating from Mount Anthony Union High School in 1999, he attended Elon University in North Carolina. While receiving his Bachelors of Arts in Computer Science, Matthew was a founding father of the Epsilon Alpha chapter of the Pi Kappa Phi fraternity. In addition, he was a member of the Academic of Computing Machinery at Elon University. Matthew decided to continue his education by attending graduate school at Florida State University. In 2003, he began working on his Masters degree in Computer Science. Upon graduating from Florida State University in 2005, he earned the Information Security Specialist Certificate. In his spare time, Matthew enjoys the Florida weather by improving his golf game.