

Securing the Process of Insurance Application

Vincent Wolff-Marting, André Köhler, Volker Gruhn
Chair of Applied Telematics and e-Business, Department of Computer Science,
University of Leipzig, Klostergasse 3, 04109 Leipzig, Germany
{Wolff-Marting, Gruhn, Koehler}@ebus.informatik.uni-leipzig.de

Abstract—Legislation in many countries treats electronic signatures similar to autographic ones. Completely electronic processes thereby are theoretically possible but customers today have neither knowledge nor means to issue electronic signatures. This study describes a process that will produce some reliable signature without the need of preparation on side of the customer. As an example the insurance application processes that usually still relies on paper forms is inspected. A threat model for a generic process is described and countermeasures like cryptography, biometric features, tamper-resistant devices, timestamps, signature databases and others are discussed. A show case combining the useful measures to a reliable process is presented as well. The proposed process can be implemented with commercially available components.

Index Terms—biometrics, electronic signature, insurance, smartcard.

I. INTRODUCTION

The major load of insurance applications is still submitted via paper forms. Frequently the forms are generated with the salesman's notebook at the point of sell, then printed, signed, mailed to the insurance company and digitized again. The original application is destroyed afterwards while an microfilm or a low-resolution-scan is kept as a proof. Depending on the character of the insurance, the customer's signature might be mandatory by law or desirable to prevent legal proceedings later on. Electronic signatures can substitute autographic ones [1], [2], [3], [4] but currently most customers lack the means such as a cryptographic key to issue appropriate electronic signatures [5].

Commercial products try to use the personal signature as a biometric feature to replace the missing key. Biometric features such as facial human characteristics, fingerprints or personal signatures are being used for various security-related purposes. It is fairly agreed that such features can provide reliable means for the authentication of a human being as long as the biometric data are taken directly from the person being authenticated [6] [7]. Trying to enhance security of electronic signatures by the integration of biometric data seems questionable as those data can not be considered secret [6].

This paper describes how electronic and personal signatures can be combined in a process to ensure a reliable business transaction without need for prior preparations on the customer side. To verify the effectivity of the proposed process, a threat model will be developed.

This paper is organized as follows: After an overview of the related work in section II, a sample process is presented (Section III-A) and the threats for that process are analysed

in a threat model (Section. III-B). Section III-C focuses on the Opponents. It is analysed, which incentives they might have and what resources they might use to attack the process. Measures to prevent an successful attack are presented and analysed in section IV. Finally it is described in a show case how the measures can be combined to a secure process in section IV-J, before section V sums up the results.

II. RELATED WORK

Authentication of humans is a crucial condition for all huge class of business transactions and correspondingly the topic has been comprehensively researched.

Since electronic signatures are accepted by legislation (for example [1], [2], [3], [4]) as a surrogate for manual signatures, there is an ongoing discussion, how reliable technologies can be tailored for the mass-market. [8] proposes the implementation of signing components to mobile phones to increase the penetration of the market especially concerning non technically oriented people. The economic aspects of this idea have been analysed by [9]. This idea could certainly solve many challenges discussed in this study, but it has not been implemented by phone manufacturers or telephone carriers yet. [10] summarises the benefits, expectations, hopes and fears related to the use of electronic signatures in general.

Biometric features as natural means of authentication and their transformation to and verification by computer systems have been extensive studied by various researchers. [6] gives a interesting overview over existing identification schemes, practical applications and techniques. They strongly doubt the usefulness of biometric samples as part of electronic signatures. Some good background information on automated verification of handwritten signatures is provided by [11], [12], [13], [14], [15] and [16]. In contrast [17] and [18] examine strength and weaknesses of authentication by means of passwords.

Tamper resistant hardware, usually in form of so called smart cards, brings a useful layer of physical protection to security relevant processes like the one discussed in this paper. [19] outlines the evolution of that hardware, [20] and [21] explain known attacks and possible improvements of tamper resistant processors.

III. THE PROCESS AND ITS THREATS

A. A Generic Insurance Application Process

The insurance application process analysed in this study represents a generic process that can be mapped to the processes

of most insurance companies with one important distinction: paper forms previously required by legislation or preferred as a proof are substituted by electronic documents. The process is applicable whenever an application form is filled out by the customer in the presence of a salesperson or representative.

In the generic process the application form is generated and completed electronically using the salespersons notebook. It will be signed electronically (details can be found in section IV) and sent to some sales organisation which extracts relevant data and forwards it to the insurance company. There are plenty of different sales channels for insurance products of which some include one or more intermediate sales organisations while others omit that chain link. In this example one organisation is included, however little changes will be required to adjust the model to different distribution types. This study does not cover direct marketing through internet or other distribution types omitting the sales representative: his contribution is mandatory since he will have to provide hard- and software to the process that the customer can not be expected to own.

An insurance application form contains several information about the insurant, the property or person insured and the risks covered by the insurance. The application is expected to bear a signature of the customer, sometimes there are special clauses – for example concerning data protection, cancellation or rescission – that have to be signed separately.

Usually a duplicate of that form will remain with the customer, additional duplicates might remain with the representative and the sales organisation.

B. Threat Model

This section discusses threats for a simple generic insurance application process. The process contains the electronic transmission of the application data and the biometric data of the applicant's personal signature.

According to Schneier central questions of a threat model are "what [is] the system [...] designed to protect, from whom, and for how long." [22] At this point possible targets for an attack shall be identified while appropriate countermeasures will be described in section IV. The duration of the protection needed varies with the type of insurance. Life assurances for example have policy period of decades so one might want to last the protection at least that long. On the other hand, fraud affecting the application form is most likely to occur at the begin of the insurance period, so a protection lasting only a few years might be enough to counter most attacks and will probably still be better than no protection at all.

First of all it is possible to manipulate just a single copy of the application form. Such an imitation should be detected easily unless the signature is counterfeited as well. It might be possible to circumvent a proper verification of the signatures: When embedded into sophisticated file formats such as the Portable Document Format (PDF) [23] or the Tagged Image File Format (TIFF) [24], electronic signatures usually consist of some visual annotation for human beholders and some hidden binary data composing the actual signature. The visual annotation can be simulated simply by inserting a picture to the

document. The signature verification software will probably ignore such a picture. If there are other genuine signatures embedded in the same file, then the software might report "all signatures valid" – even if those signatures cover other parts of the document. A quite similar vulnerability of a popular cryptographic software has been reported recently [25]. To manipulate the application process, an attacker might just change the document as desired and resign signatures that are in his sphere of influence after having replaced the all other signatures by simple pictures. The attack is based on the fact, that a human beholder is unable to check the validity of an electronic signature without the aid of a computer while the computer itself lacks information about the structure of the signed document. This issue can be solved by introducing a firm structure into the document that prescribes which parts of the document have to be covered by a signature.

If the manipulation of the document is accompanied by the forgery of signatures, then the exposure is difficult or maybe even impossible. When the attacker knows the private key of the signature or has broken the underlying mathematical algorithms then the adulteration of the signature will be as good as an original one. This is an important difference to an imitated personal signature. The usage of reliable algorithms and scrutinized implementations should prevent or at least complicate such an attack. Both hash algorithms and encryption can be the target.

Attacking the hash algorithm means trying to find a second document that can be reduced to the same hash as the original. Any signature issued for one of them will be valid for both. Recent research [26] confirms that threat. Being prepared to change the algorithm used within a reasonable amount of time seems to be a useful countermeasure [27].

Successfully attacking the encryption algorithm provides the attacker with the means to decrypt the raw biometric data of the Signature. Possessing both the cryptographic key and the raw biometric data, the attacker can issue fake signatures at will. It will be shown that there are other ways to obtain those elements so attacking the encryption can be classified as preparative.

Besides manipulating signed application forms it might be possible to temper unsigned message streams. The fraud will become obvious as soon as the message streams are compared to the signed form. The comparison has to be part of the regular process. Afterwards comparisons will not be possible since usually only the forms will be archived but not the whole message streams. So it can not be distinguished at which point of the process the differences occurred for the first time. Relying solely on signed data throughout the whole process might be the best solution. Existing systems often separate the data actually used for processing from the signed data. While that separation is not per se necessary it simplifies the integration of an electronic signature into existing processes. Software tools gathering the insurance details and forward it to the next step of the process chain do not sign their data. They simply transmit it to a separate signature software, usually via the printer interface. Thereby existing tools can be used without much configuration effort and the signature software stays application-independent. But thereby the signed

document loses its context as well. Retrieving the originally structured data from that document is a non-trivial task. Consequently comparing it to an alternative representation of the (presumed) same data can be difficult as well.

An attack might be lead against the signature verification software. This requires profound knowledge about the software and access to the targeted system. The access does not need to be physically, specially prepared documents might provide the required control, for example via a buffer overflow as described in [28] (see also [7]). Subject to the existence of a suitable vulnerability the attacker might route even unsigned documents through the process without being noticed.

Finally a denial-of-service-attack can be used to interfere the signature verification (see also [7]). Somehow the process has to be flooded by a huge amount of applications. They might consist of randomly generated or withheld real applications. Alternatively the verification system can be deceived to reject legitimate applications. Natural maxima due to seasonal factors or legislative reforms might flank the attack. Rejected applications require special treatment such as a manual revision. A persisting denial-of-service might erode the diligence so that invalidated applications might slip through unnoticed. A proper fall back process will reduce the impact of a denial-of-service-attack. On the other hand, the scenario seems rather implausible, so investing into a fall back process might not be cost-effective.

Malfunctions, transmission errors, etc. will, as well as deliberate manipulations, render a signature invalid. The authenticity of signed papers will not suffer from initial tearing, small holes or dirt, while a single twisted bit will ruin an electronically signed document.

C. Opponents

Literary any participant of the application process could see some incentive in changing some parts of the insurance application form, so anybody has to be considered as a possible opponent. The customer could be motivated to manipulate his own application to benefit from insurance coverage without paying an appropriate premium, additional proceeds on savings or to cover already incurred losses. Naturally the insurance certificate (policy) rather than an application form will determine extent and manner of the coverage. But depending on national law and special circumstances the application form can be crucial as well. Moreover the policy might be counterfeit as well. Discussing details in an international context would exceed the extend of this paper.

There are important distinctions between electronically signed forms and paper forms as the following example shows:

- 1) The insurance company possesses an application paper form while the customer has a carbon copy. Both are signed by the applicant. It would be easy for the applicant to obtain a blank form, fill in whatever he likes and sign it, while the company would need to forge it costly. If the content of application form and carbon copy show differences then the changes have been made by the customer most likely.
- 2) Both parties hold a digital application form. The documents are electronically signed with embedded biometric

data of the applicant. The signatures have been generated with hard- and software provided by the company, cryptographic keys are under control of the company. This time, the credibility of the customer is highly improved.

There are simplifications in that example. Even today, most companies do not archive the raw document but a scanned picture or a micro copy, which provides little authenticity. The second part assumes the customer can breach the encryption. Still he has to arrange a context where the insurance policy can not be held against him.

Such a context might be hiding relevant factors such as properties of the insured object, perils, chronic illness, etc. Deceit and fraudulent concealment of such information regularly voids or reduces the coverage – again details depend on national legislation. A malicious insurant could generate two versions of the application. The first one contains all relevant factors while the second one does not. After acquiring insurance with the second one it is possible to produce the first one as soon as discrepancies arise. In that case the policy is not relevant, as it can not prove whether or not the insurant provided the facts in question.

As means for an attack the customer owns a copy of his application. He possesses the biometric data of his own signature but he needs a measuring device. They are sold over the counter. Usually he lacks the cryptographic key. Consequently as long as the used algorithms are reliable, the implementation is solid and the keys are kept safe, successful attacks by the customer are not likely. Manipulating an application without forging a signature has no use for the customer, as the signature will be checked as soon as he tries to take advantage of the manipulation.

Intermediaries have, contrary to customers, keen insights into the process, but their influence is limited to the time before the conclusion of the contract. They can be motivated to tamper to obtain commission claims or meet some sales target. They might furthermore assist a customer in adultery, for example predate an application before the occurrence of loss or injury.

An attack might target on getting the biometric data of the applicant, to sign some more or other forms. This can be done without breaking a valid signature by rigging the measuring system, so that it will not encrypt the data at all. The malicious intermediary can use that data to commit little changes to real applications, hopping to exploit that gap multiple times over a long time. He can as well try to get a huge amount at once and hope to withdraw fast enough. The first approach contains the risk of getting discovered prematurely. The second approach can be discovered by data mining. The sudden success of the intermediary in dispute should trigger closer investigation.

Every manipulation with impact on the commission requires noticeable changes of the contract. The customer will most probably complain about any changes to his disadvantage, especially an increased premium. Insurances never ordered will raise protests as well.

It seems possible, to forge and predate an application for an already dead applicant using recorded biometric data. Depending on the character of the contract, this will be in favour of the insurance company, if a loss event is impossible

in result of the death. It will be in favour of the heirs if it is a life assurance with interim cover. The primal application, that the data are taken from, must be withheld, otherwise the attack can be easily tracked. Moreover, this type of fraud can not be used frequently without getting suspicious.

All pieces of soft- and hardware that are used during the signing process will be permanently under the control of the intermediate. He has got the opportunity to analyse everything carefully and to prepare his fraud unhindered.

It seems quite unlikely for the insurance company to forge signatures. But besides the company as a whole some single employee might, driven by a personal motive such as sales targets, corruption or just malevolence, commit some fraud. Additionally any plausible accusation that a company manipulates applications can do lasting damage. Such rumour will discourage potential customers. If it is possible for the company to alter signed applications at least theoretically, then its position in a lawsuit will degrade.

There are divers ways to attack the process from within the company. Altering unsigned streams of data as described before will be possible. In the long run it will be to the disadvantage of the company, as the company will be bound the genuine application form. Manipulations will only temporarily affect the company's IT-System, so they are only useful to insiders with short-term interests who accept or even compass harm for the company. The same is true for manipulations at the signature verification system of the company.

The company and employees respectively can try to manipulate signed application forms. The company controls both soft- and hardware that are used for the signing process. It also possesses the cryptographic keys. To verify biometric data they have to be deciphered, therefore chances are that the company can misuse the raw biometric data as well. The resulting difficulties will be discussed in chapter IV.

Counterfeit insurance application forms can be used to impose single contract terms or whole contracts on the customer. He might overlook single terms when he expects a similar policy, while he will probably not expect or pay for a unexpected insurance. Even if the fake application is taken as genuine, the customer has the right to withdraw from the contract in most countries.

Finally outsiders could want to attack the application process. It has to be distinguished between attackers who accomplice with some of the already described factions and attackers who follow their own interests.

Accomplices can be technicians or developer with keen knowledge about hard- or software of the signature verification process. They can provide their accessories with the means for an attack.

Others could deliberately try to damage the insurance company, for example to change the companies stock exchange quotation.

IV. COUNTERMEASURES

Having discussed threats and adversaries in the previous section, this section will concentrate on techniques and procedures to protect the sample process. The measures with their

strengths and weaknesses will be introduced individually at first. In a next step it will be discussed how an integration of multiple measures can lead to a more comprehensive protection.

A. Plain Cryptography

Plain cryptography composes the base for most of the following approaches. But not all risks can be solved by cryptography, it takes more to get a secure process [7].

Cryptography will preserve integrity, if a message is sent between two nodes via an insecure channel [29]. Conferred to the insurance application process the customer is the sender and the insurance company is the receiver. Intermediaries constitute a – possible – insecure channel. This special scenario (see Chapter III-A) allows intermediates to read but not to change the message. So only integrity and not confidentiality is expected. This challenge can be solved by electronic signatures. But it is demanded, that the customer neither needs a cryptographic key nor soft- or hardware to accomplish cryptographic calculations. Intermediary and company can use these things. In that case cryptography can only preserve the integrity of the application form between intermediary and company.

Signatures can as well serve a secure storage of messages – effectively a transport from history to the future [29]. Concerning the application process that means, if an application was true at the time of signing, it will stay so in future. This protection can be circumvent: cryptography will not stop the key owner from signing additional documents with different content. So this feature will stop any attacker but the key owner.

B. Strong Cryptography

It is mandatory that a signature really serves its purpose. "With enough effort, any cryptographic system can be attacked successfully" [29]. To prevent this, reliable techniques, so called strong cryptography, has to be used.

Prognoses are that a technique will be safe for the next decades, if a successful attack needs at least 2^{128} steps [29]. That value must not be confused with the actual length of a given key. It is based on theoretic considerations concerning the future development of soft- and hardware. The evaluation of current techniques rely on currently known attacks. The prognoses might be wrong. Dependable long time storage can nevertheless be reached. The documents to be saved have to be resigned frequently with one of the most current technique. Old signatures have to be included so they are protected as well. There is no need for the original signer to conduct the re-signing, an independent third party will be sufficient [30].

There are several national recommendations about currently suitable algorithms [31], [32], [33]. It has to be expected that the recommendations will change continuously over the time so it is advisable to prepare for changing the used algorithms whenever it is useful.

C. Structured Data

Electronic signatures can only protect the document it refers to. As explained in chapter III-B, some products do not use the signed document but an unsigned data stream for further processing. The data to be signed is captured with a special printer driver, somehow an image of the real document. The signed data is not structured, which means, it can be difficult to extract a certain piece of information from the document. Structured documents in contrast follow a defined layout and provide easy access to the content.

It should be avoided to route the same information over two parallel streams for it bears the danger of inconsistency. As the signed but unstructured stream can not be analysed automatically, any discrepancy can remain hidden for a long time. To prevent that problem, solely signed data should be used throughout the process. In principle, any electronic data format can be used, since signature algorithms can process any bit string regardless its meaning. Some formats as the Portable Document Format (PDF) [23] or the Extensible Markup Language (XML) [34] are designed to hold both structured data and electronic signatures, even biometric data [35].

With structured documents it is possible to define, which part of the document is covered by a signature [23]. That is especially useful for documents containing more than one signature. Later it can be analysed automatically whether the required signatures are present at the right place. With non structured documents, automated processes can only verify if there are signatures at all. Not even an optical review can verify that everything is in place, since it might be deceived by a picture of a valid signature.

D. Biometrics

The previous sections showed, how cryptography can protect the communication between insurance intermediary and company. The personal signature shall help to expand the secured path to the customer. The biometric data of the personal signature comprise a factor that is not controlled by the intermediate.

Biometric characteristics are intrinsically tied to a human. They can hardly be stolen or eavesdropped, as long as they are verified directly after being sampled from a living person [6]. If they are sampled, digitalised and transferred, then they are no longer reliable. The receiver can not prove the time or genuine purpose of the measuring [36]. Therefore it is most important to install further protection techniques.

It must be possible to test the consistency that is, to detect subsequent changes of the signed document, without decryption of the biometric data. Every legitimate holder of a document copy may want to perform that test any time. Validating the biometric data on the other hand is quite special. That will require a valid reference signature and expertise about signatures – be it the artificial knowledge of a software exerciser [16] or the natural knowledge of a forensic appraiser. The raw (that is unencrypted) data of at least two signatures is needed for a comparison – but the raw data can be misused. While testing the consistency is harmless and useful for all

stakeholders, testing the biometrics can only be done by few and is potentially dangerous. Therefore it might be useful to use different cryptographic techniques or keys for both tasks and escrow the only copy of the critical key at a trusted site. Depending on local legislation and accepted customs a notary or someone alike will qualify as such a third party.

E. Comparing Symmetric and Asymmetric Encryption

This section addresses the danger of manipulation by the key holder. A lot of security policies follow the principle of the least privilege: a subject shall not have more privileges than it needs for its work [7]. It is especially important for electronic systems. The secret key of the DVD-Security is embedded in every DVD-Player. By decompiling a single software player and extracting that key, hackers had broken the whole safety system. Similar problems arose with some Pay-TV-Systems and some North American local traffic payment systems. It is a general problem of systems that store a global secret in a user device [7]. Unfortunately exactly that is done with many tools for biometric signatures.

Generally there are different encryption techniques that can be used for a biometric signature:

- symmetric encryption of the biometric data,
- asymmetric encryption, using the public key for encryption and the private key for decryption,
- asymmetric encryption, using the private key for encryption and the public key for decryption.

Symmetric encryption provides most possibilities for an attack. Both signing and verification software must contain the shared secret. If only one part is analysed successfully then the whole process is compromised.

Asymmetric encryption seems to offer two variants, but this is not exactly true in this case. Technically there is no distinction between both keys: One can decrypt what the other encrypts and vice versa. The main difference lies in the publication of one key and the hiding of the other.

The examined scenario needs two different key pairs. The first one will be used for the encryption and decryption of the biometric data, the second one for the encryption and decryption of the hash value. The key for the encryption of the biometric data has to be embedded into the sampling device. A huge number of sampling devices will be in use, so this key can not be considered secret. That key can not be used to change existing documents. It could be used to circumvent the usual process of signing a document, so if someone manages to capture raw biometric data, he can use the key to encrypt it and append it to an arbitrary message (replay attack). It could be used for a "chosen plaintext attack" as well, but a modern cypher will be resistant against that [29], [37]. Since that key can not be efficiently kept secret, it should be regarded as public without weakening the security of the process.

The counterpart will be embedded in the verification system. As stated in section IV-D, verification shall only be done by few trusted people, as the decoded raw biometric data can be misused. So this key has to be kept private.

The opposite is true about the encryption of the hash value. Verifying the genuineness of a document is useful for every

holder of a copy, so the key to decrypt the hash value has to be public. The key for encrypting it on the other hand should be secret, since anyone who has that key and some raw biometric data can use it to sign phoney documents. Even without biometric data that key can be used to fabricate rudimentary imitation. The imitation would stand until the biometric data is verified which might be long enough for some kind of deception. Unfortunately that key needs to be distributed with the sampling device so keeping it secret is not trivial. Section IV-F addresses that challenge.

F. Countersignature of a Representative

So far it has been assumed that only a single pair of keys will be used for all soft- and hardware. Alternatively each sales representative and intermediary can be equipped with a personal, private key for the encryption of the hash value. Thereby the origin of every application form can be tracked. If there are circulating different versions of an application with different content, precisely one person has countersigned each version. He can be interrogated and called to account if appropriate. The attacker can no longer hide in a anonymous crowd of people who could have done it.

Generation and distribution of the private key have to be done in a way which guaranties that no one but the legitimate key holder gets a copy [33]. Who ever verifies the integrity of an application form needs the public key of the representative. The distribution can be done by a public key infrastructure (PKI, also known as certification-service-provider) [29]. It should provide at least the following information:

- the public key for a given signature,
- validity period of a key, especially
- the information whether the key has been valid at the time of signing and
- some information identifying the key holder.

There are existing both national and international laws and recommendations concerning public key infrastructures [4], [2]. The infrastructure can be provided by the insurance company, it can be provided by a third party as well. Some legislation lay out different compliance levels for a PKI implicating different legal effects of the resulting signatures. Which level will be suitable for a given company can not be discussed in the scope of this document. Generally, the expense of the infrastructure should somehow balance the reduced risk of loss due to a fake signature.

Some sampling devices [38] merge a serial number with the biometric data. That serial number shall serve a similar purpose as a countersignature. If the company keeps account of those numbers, than it can identify the originator of each signature. There are some restrictions: sampling devices are not as tightly bound to a person as private keys should be. Broken devices have do be repaired, meanwhile the intermediate will use a substitute device. Employees of a brokerage office may share one device. These are challenges that could be met by organisational measure. But technically the serial is bound to a signature, not to a application form. If an attacker succeeds in transferring the biometric data together with the serial from one document to another, than the serial will point

to the genuine origin of the data and not to the attacker. Finally it might be possible to separate the serial number from the biometric data. Further research on the devices in question would be needed to verify or falsify that. A countersignature on the other hand utilises public-key cryptography that has been explored soundly [39].

G. Tamper-resistant Devices

Tamper-resistant devices that provide means for a reliable storage of cryptographic keys are denoted "secure tokens" [29], or secure-signature-creation devices [2]. They are designed to withstand a wide range of attacks both local and remote, both invasive and non invasive [21]. Such devices consist of a non-volatile memory and a central processing unit capable of cryptographic calculations. Before usage the device has to be unlocked by a password, a personal identification number (PIN) or through a biometric attribute of the user measured by the device. A tamper-resistant hardware class commonly used for electronic signatures are the smart cards [19], [40], [20].

Those devices can be used by the representative to countersign an application as proposed in section IV-F. They can also be useful if no personal keys are issued as shall be shown exemplarily.

Figure 1 illustrates a simplified generic signing process. This process can easily be implemented with components of the shelf. The usage of a tamper-resistant device yields little advantage, solely the cryptographic key is protected. Application form and biometric data are kept unprotected at the central memory of the computer. They can be manipulated by a malicious user or a prepared program (trojan horse). The secure-signature-creation device will sign an altered form unobjected. So this implementation is overall not very secure.

The architecture visualised by figure 2 provides improved protection. Major parts of the process are conducted within the tamper-resistant device. The biometric data will not be revealed to the outside. The device displays the document to the customer and nothing but the approved document will be signed. It is certainly possible to manipulate the device as well, but it is much more difficult in comparison to the manipulation of a personal computer [21]. Specially designed hardware is needed to assemble this architecture. Both sampling device and display need to be controlled by the tamper-resistant device via a secure canal. The device has to be capable of displaying complex documents.

The author holds the opinion that though theoretically interesting, it will not be cost-effective to implement such an architecture in the given context. It is probably more easy to issue private keys to the representatives and hold them liable for their signatures. Than manipulations will have negative impact for them and the design described in figure 1 will provide sufficient protection.

H. Timestamp

Some products [38] claim to gain security by embedding timestamps into the signatures. This section shall explain the mode of operation and impact of that feature. A timestamp is

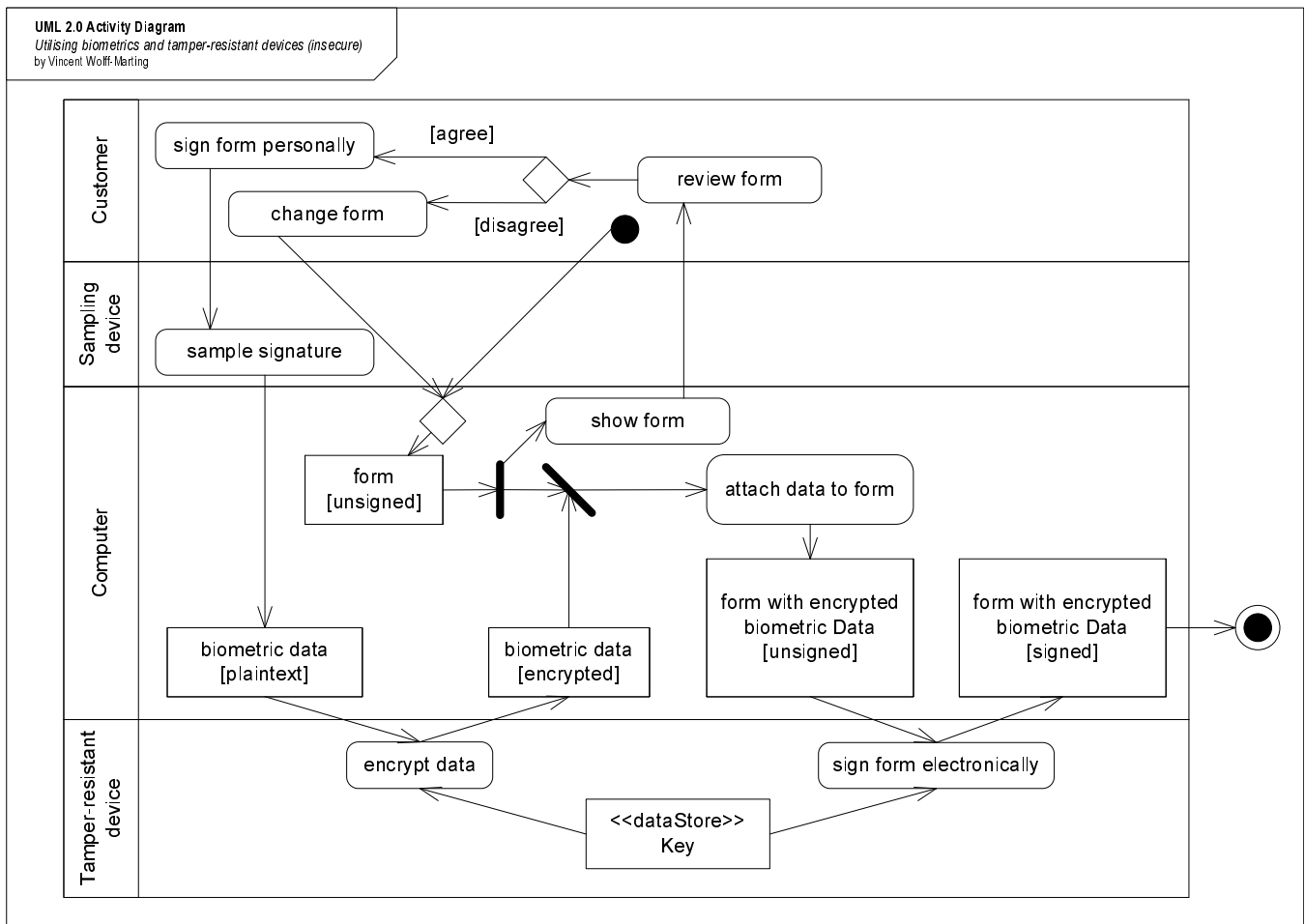


Fig. 1. Utilising biometrics and tamper-resistant devices (insecure design).

a string of the current time and date appended to an electronic document. It is important, whenever the chronological sequence of a process shall be traceable. To prevent the alteration of a set timestamp it can be signed electronically, furthermore a mechanism to prevent manipulations of the underlying clock is needed [29].

Trusted third parties can be entrusted with the task of time stamping and signing documents. Afterwards all parties involved can prove the existence of the document at a given time. That special case of an electronic signature has partly been codified in Germany [30].

A timestamp set by signature-creation devices renders additional plausibility checks possible. A timestamp reveals application forms antedated with intent to defraud. Such a defraud will become less promising as the processing time of an electronic form is significantly faster compared to a paper form so an application form out of date will be suspicious to begin with. A considerable old timestamp can be a sign of a reused signature from an older application form. There the timestamp will help identifying the original application bearing the misused signature. If a lot of signatures are generated within a very short period of time or if they are dated at unusual times then they are possibly set by the representative without knowledge of the potential customer

and further investigation might be useful. If multiple signatures are set at exactly the same time then it is either a huge technical fault or a defraud.

If a timestamp is embedded to a signature it should be routinely checked during the verification process. Therefore it must not be encrypted with the biometric data – then the verification would require an decryption of the biometric data which shall be avoided (compare Section IV-D).

I. Signature Database

The biometric data can only be verified against an other sample of that data from the same person. If biometrics are used for frequent authentication of a relatively closed user group then reference samples are stored in a database [41]. Use cases like banks inspecting signatures on cheques and remittance slips or access control systems usually rely on such databases. Insurance companies in contrary do not get new documents signed by existing customers on a regular base. It has to be assumed, that the forms are signed by applicants whose signature is unknown to the company. Even if the applicant is a regular customer, existing signatures can be several year old. Natural variances of the signature increase the risk of false rejection.

In section IV-D the risk adherent with the verification of

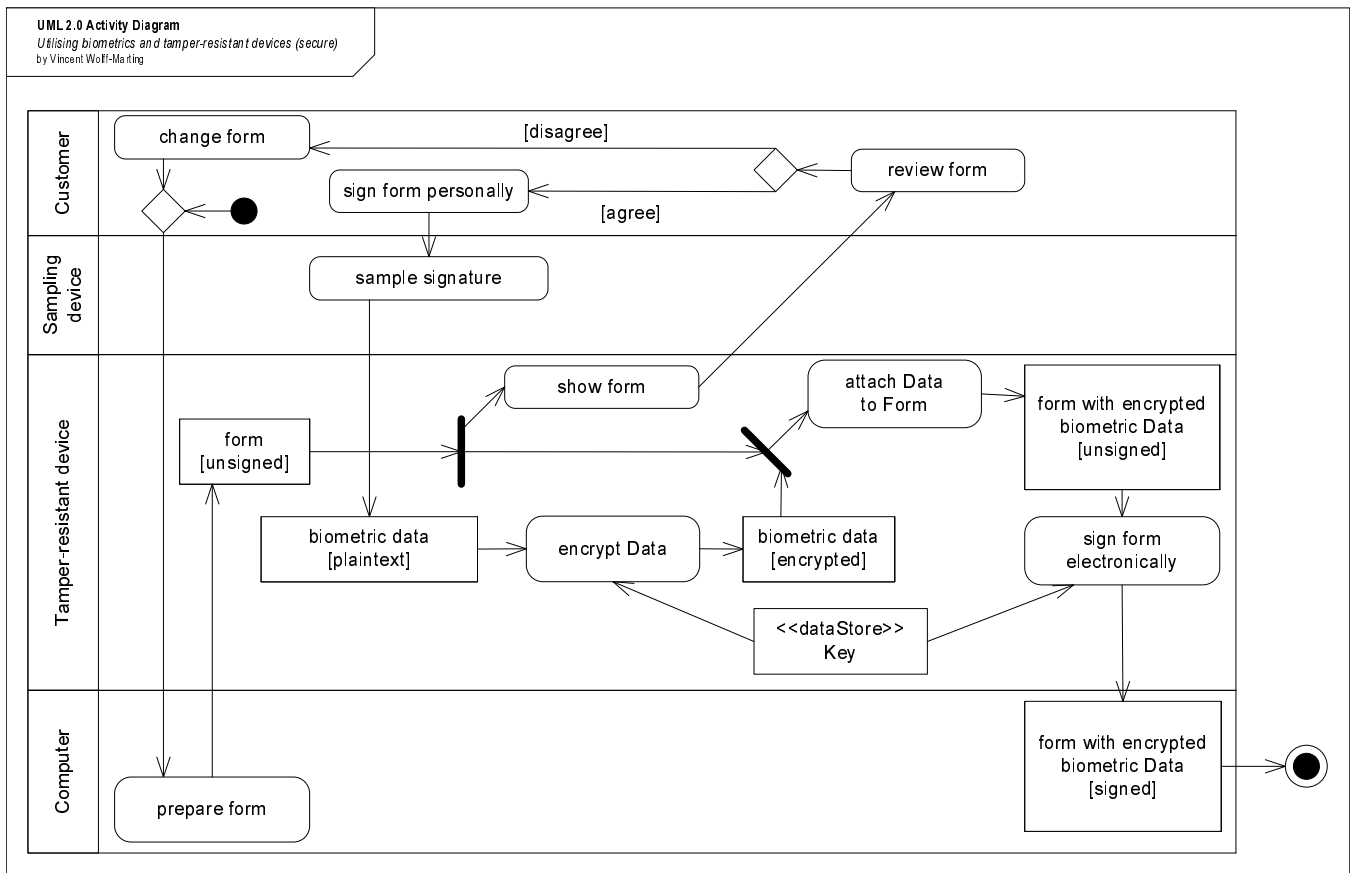


Fig. 2. Utilising biometrics and tamper-resistant devices (secure design).

biometric data have been shown: The data has to be available unencrypted and can at least theoretically be misused. In that context a signature database bears a risk. The risk can be minimised by restrictive access control, detailed documentation of database requests and similar measures. But as long as the database can be used for few applications with a high false rejection rate, it seems not efficient to invest into both database and safety measures.

J. Combining Countermeasures for a Comprehensive Protection

Figure 3 illustrates a show case process including the identified protection measures. The model has been reduced to the steps relevant for a secure communication. Review of the application form by the customer, generation of a copy for the customer as well as succeeding steps like effecting the policy have been excluded.

The personal signature of the customer will be sampled and encrypted asymmetrically. The key needed for the encryption is considered public (see section IV-E), so this part does not need to be protected by a tamper-resisted hardware. The biometric data will be kept unprotected at the central memory of the computer. While that is not ideal it can not be avoided using commercially available products (see section IV-G). Indeed there are some devices encrypting the data for the transmission to the dedicated software [38]. Eventually such

devices can be customised to accomplish the final encryption as well.

There are three ways of time stamping an application, which are not shown in figure 3. The least secure way is to use the internal clock of the computer. Any attacker can change it at will. A time stamp from the sampling device would be more secure. That time stamp is regularly encrypted together with the biometric data, but it should rather be appended as plain text to the jet unsigned application form. Thereby it stays readable to whom it may concern while being protected by the signature. Since that has to be done by the computer, there is some risk of intervention by an attacker. The most secure way would be a time stamp issued by the tamper-resistant device, but again, commercially available products do not support that option jet.

The actual signing of the application and the biometric data with the private key of the representative is conducted in the protected environment of the tamper-resistant device. The representative has to authenticate himself first. If he fails to (e.g. if enters a wrong PIN three consecutive times), the device can disable itself permanently or for a fixed amount of time and the process will be cancelled. Such a restrictive approach might be deprecate by representatives, as they might loose a lucrative customer in case they fail to authenticate by accident. They might improve tactics to circumvent that danger – eventually by writing their PIN on the device – and thereby ruin the effort to secure the process [29]. To

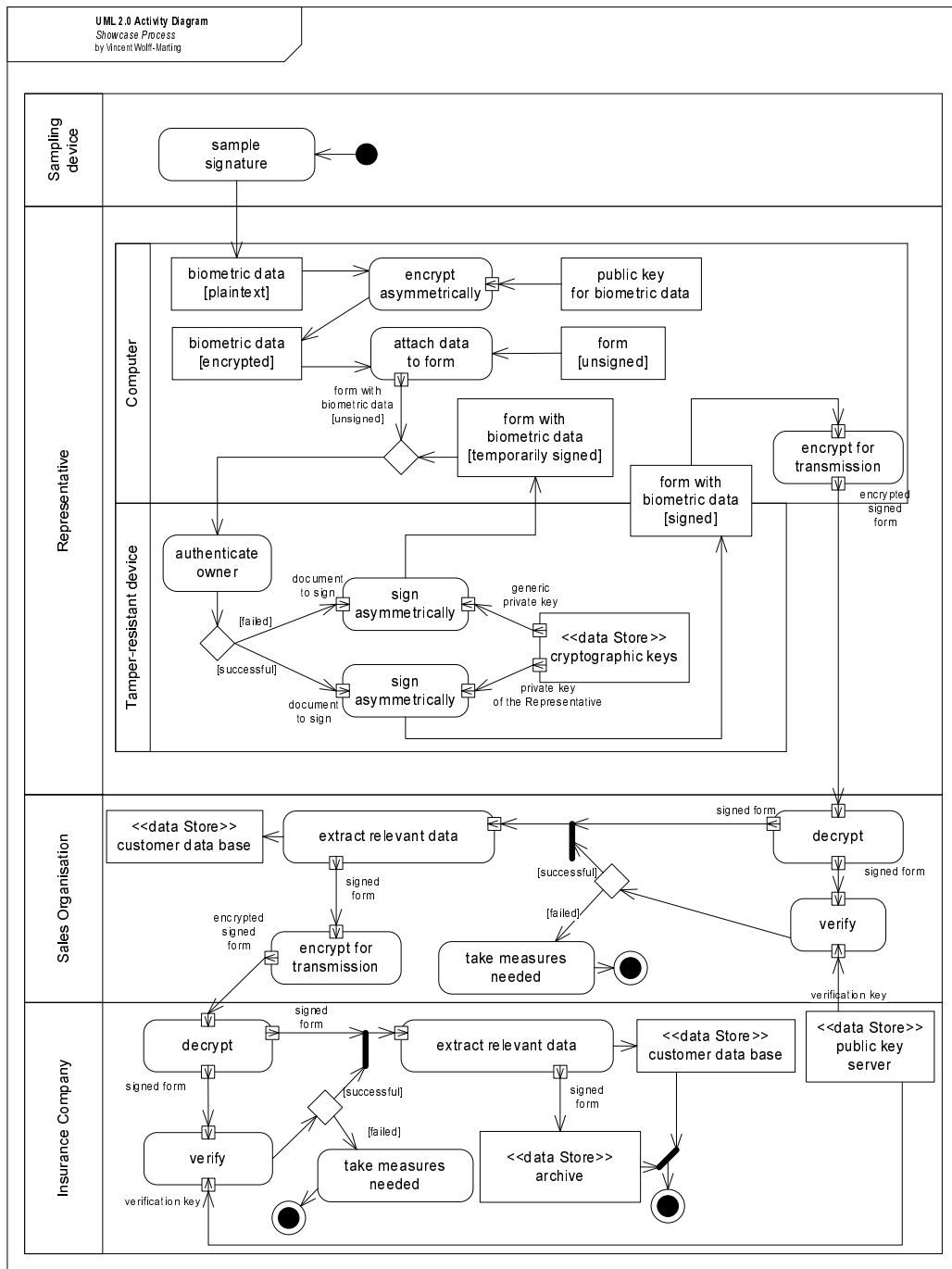


Fig. 3. Show Case of the secure Process

prevent annoyance of customer and representative the device might instead sign the form with a temporary key. Then the representative has to approve the form with a valid signature later. The tamper-resistant device can verify the integrity of the temporary signature before appending the new one. That way the delayed signature does not comprise additional threat to the process.

The signed application form will be sent to the sales organisation. It should be encrypted for the transfer to protect the privacy, especially if the transfer is via a public channel like the internet. The representative might also want to save

the signed form locally but that does not influence the process. Upon receipt the sales organisation will verify the signature with the public key of the representative. Secondly it should be confirmed that all required signatures are in place. The authenticity of biometric features can not be verified at this point. The public keys of the representatives are usually provided by a central server (see section IV-F). Figure 3 displays that server as part of the insurance company, it can be operated by the sales organisation or an independent third party as well. Even multiple server at different locations can be an option.

In case a signatures can not be verified, the reason for the failure needs to be investigated. Possible reasons are transmission errors, accidentally transmission of an unfinished application or an attempt to defraud. Depending on the assumed reason, appropriate measures must be taken, for example asking the customer to sign the application again or propose him an insurance based on the invalid application.

If the application is signed validly then the sales organisation can extract all data needed for further processing. It has to be reemphasised, that the organisation thereby relies only on signed and verified data, no unsigned secondary data stream does exist.

Finally the sales organisation has to encrypt the application again and send it to the insurance company. Here the process described before is mostly repeated.

Only in case of controversy about the insurance application the biometric data need to be verified. The required keys should be under sole control of a trusted third party. Neither insurance company nor sales organisation should have direct access to that key. The contested signature has to be decrypted by the third party and forwarded to an forensic appraiser or an software exerciser.

The process described addresses the threats identified in section III-B. Especially the countersignature of the representative minimises the possibilities to issue manipulated application forms. For all kind of changes to the form need to be finalised through a signature of the representative. The representative most likely will not consciously sign a counterfeited document. Counterfeiting the representatives signature will be almost impossible as long as the underlying cryptographic algorithms are reliable and soundly implemented. Pictures used to feign signatures will be detected as each signature within the structured document will be explicitly verified.

Attacks utilising real biometric data can not be eliminated totally; Customers can always sample their own signature and it might be possible, to obtain a signature in blank by some confidence trick. These attacks are beyond control of the insurance company and can not be prohibited efficiently. But still the sole biometric data will not be enough to tamper an application form.

Unsigned data streams are not provided which eliminates another vulnerability. There is still a risk of manipulation when the application data has been extracted and inserted to a operational database. That issue exists with paper applications as well and it can be solved by access control and audit trails. It exceeds the scope of this study and can therefore not be examined further.

Purposeful or accidental malfunctions of the verification system remain a challenge. The threat by malicious soft- or hardware suppliers however is not limited to signature verification systems. Recent research on aspects of that challenge have been made by [42] for example. Certification (such as [43]) might add some creditability to a verification system. Fidelity bonds can cover remaining dangers.

Denial-of-service attacks can not be generally averted [7], as it will always be possible to produce more application forms than the company can handle in the same time. In such a situation the company should try to find and possibly

eliminate the source of the attack. If the company is forced to accept applications without proper verification, the verification should be conduct as soon as the verification system is working faultless again.

Finally the representatives computer might become compromised. The attacker can obtain the biometric data of a signature via a prepared computer, but as shown before, those data alone are not sufficient to forge an application. The compromised computer could as well manipulate an application prior to the signing. Even if the manipulated document is completely displayed by the tamper-resisted device, the changes might be overlooked by the signer. The architecture illustrated by figure 2 (Section IV-G) would prevent that kind of attack but it requires specially designed hardware. It seems more advisable to secure the representatives computers than to invest into special hardware.

V. CONCLUSIONS

It has been shown that it is possible to set up a process that generating documents electronically signed by customers who need no special devices or knowledge. Compared to conventional semi-electronic processes that destroy the paper forms after archiving a digitised copy, this process provides a noticeable increase in security. The process can be set up using commercially available components. Admittedly they have to be configured carefully. This study focuses on the insurance application process. The results can probably be transferred to other scenarios as well, as long as some representative or salesperson is involved at the point of sell. It remains to be determined, whether the process described here meets the demands of a given local legislation regarding signatures.

It is worth mentioning that a so called secure hardware does not automatically increase the protection. It has rather to be carefully adopted into the process. Furthermore, the use of a signature database as it is mandatory in the banking business will not at all improve the security in this context. On the contrary, the data in such a database might be misused so it has to be advised against it. The biggest gain in protection is provided by the countersignature of the representative. That measure is still advantageous, even if the representative himself is an adversary because it simplifies penalisation.

REFERENCES

- [1] UNCITRAL, Ed., *Model Law on Electronic Commerce with Guide To Enactment*. New York: United Nations Publication, 1996, 1998. [Online]. Available: <http://www.uncitral.org/pdf/english/texts/electcom/05-89450.Ebook.pdf>
- [2] the European Parliament and the Council of the European Union, "Directive 1999/93/EC of the European Parliament and of the Council of 13 December 1999 on a Community framework for electronic signatures," *Official Journal of the European Communities*, vol. L 12, pp. 12–20, 2000-01-19 2000. [Online]. Available: <http://europa.eu.int/eur-lex/pri/en/oj/dat/2000/l12/l1201320000119en00120020.pdf>
- [3] 106th Congress, "Electronic Signatures in Global and National Commerce Act," http://www.occ.treas.gov/efiles/disk2/resources/elect_bank/con-_pub.l.106-229_e-sign_act.pdf, 7 2000. [Online]. Available: http://www.occ.treas.gov/efiles/disk2/resources/elect_bank/con-_pub.l.106-229_e-sign_act.pdf
- [4] UNCITRAL, Ed., *Model Law on Electronic Signatures*. New York, Wien: United Nations Publication, 2002. [Online]. Available: <http://www.uncitral.org/pdf/english/texts/electcom/ml-elecsig-e.pdf>

- [5] H. Rossnagel and D. Royer, "Investing in security Solutions - Can Qualified Electronic Signatures be Profitable for Mobile Operators?" in *Proceedings of the Eleventh Americas Conference on Information Systems (AMCIS 2005)*, Omaha, Nebraska, 2005. [Online]. Available: <http://www.wiiv.de/publikationen/InvestinginsecuritySolutionsC1315.pdf>
- [6] "A study on PKI and biometrics," Abruf: 2005-12-12 2005. [Online]. Available: http://www.fidis.net/fileadmin/fidis/deliverables/fidis-wp3-del3.2.study_on_PKI_and_biometrics.pdf
- [7] B. Schneier, *Secrets and Lies: Digital Security in a Networked World*. New York: John Wiley & Sons, 2004.
- [8] L. Fritsch, J. Ranke, and H. Rossnagel, "Qualified mobile electronic signatures: Possible, but worth a try?" in *Information Security Solutions Europe (ISSE) 2003 Conference*, Vienna, 2003. [Online]. Available: <http://www.wiiv.de/publikationen/Qualifiedmobileelectronicsigna609.pdf>
- [9] H. Rossnagel and D. Royer, "Profitability of Mobile Qualified Electronic Signatures," in *The Ninth Pacific Asia Conference on Information Systems*, Bangkok, 2005. [Online]. Available: <http://www.is-frankfurt.de/publikationenNeu/ProfitabilityofMobileQualified1320.pdf>
- [10] M. A. Broderick, V. R. Gibson, and P. Tarasewich, "Electronic signatures: they're legal, now what?" *Internet Research: Electronic Networking Applications and Policy*, vol. 11, no. 5, pp. 423–434, October 2001. [Online]. Available: <http://www.ccs.neu.edu/home/tarase/BrodGibTaraseESig.pdf>
- [11] H. N. M. and C.-N. Liu, "Automatic Signature Verification Based on Accelerometry," *IBM Journal of Research and Development*, vol. 21, pp. 245–253, Issue 3 1977. [Online]. Available: <http://www.research.ibm.com/journal/rd/213/ibmrd2103E.pdf>
- [12] R. S. Kashi, W. Turin, and W. L. Nelson, "On-line handwritten signature verification using stroke direction coding," *Optical Engineering*, vol. 35, pp. 2526–2533, Sept. 1996.
- [13] T. Wessels and C. Omlin, "A Hybrid System for Signature Verification," in *IEEE-INNS-ENNS International Joint Conference on Neural Networks (IJCNN'00)*, vol. 5, 2000, p. 5509. [Online]. Available: <http://csdl.computer.org/dl/proceedings/ijcnn/2000/0619/05/06195509.pdf>
- [14] E. J. R. Justino, A. El Yacoubi, F. Bortolozzi, and R. Sabourin, "An Off-Line Signature Verification System Using HMM and Graphometric Features," in *Fourth IAPR International Workshop on Document Analysis Systems, (DAS 2000)*, Rio de Janeiro, 2000, pp. 211–222. [Online]. Available: <http://www.livia.etsmtl.ca/publications/2000/JustinoDAS.pdf>
- [15] M. Tistarelli, J. Bigun, and E. Grosso, *Advanced Studies in Biometrics*, ser. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Verlag, 2005, no. 3161.
- [16] L. Oliveira, E. Justino, C. Freitas, and R. Sabourin, "The Graphology Applied to Signature Verification," in *12th Conference of the International Graphonomics Society (IGS2005)*, Salerno, 2005, pp. 178–182. [Online]. Available: <http://www.livia.etsmtl.ca/publications/2005/graphology.pdf>
- [17] L. Zhuang, F. Zhou, and J. D. Tygar, "Keyboard Acoustic Emanations Revisited," in *CCS '05: Proceedings of the 12th ACM conference on Computer and communications security*. New York: ACM Press, 2005, pp. 373–382. [Online]. Available: http://portal.acm.org/ft_gateway.cfm?id=1102169&type=pdf&coll=GUIDE&dl=GUIDE&CFID=695892&CFTOKEN=49637104
- [18] J. Yan, Alan, Ross, and Alasdair, "Password memorability and security: Empirical results," *IEEE Security and Privacy*, vol. 02, no. 5, pp. 25–31, 09 2004. [Online]. Available: <http://csdl.computer.org/dl/mags/sp/2004/05/j5025.pdf>
- [19] K. M. Shelfer and J. D. Procaccino, "Smart card evolution," *Communications of the ACM*, vol. 45, no. 7, pp. 83–88, 2002.
- [20] S. W. Moore, R. D. Mullins, P. A. Cunningham, R. J. Anderson, and G. S. Taylor, "Improving smart card security using self-timed circuits," in *8th International Symposium on Advanced Research in Asynchronous Circuits and Systems (ASYNC 2002)*, 9-11 April 2002, Manchester, UK. IEEE Computer Society, 2002, pp. 211–219. [Online]. Available: <http://csdl.computer.org/dl/proceedings/async/2002/1540/00/15400211.pdf>
- [21] R. Anderson, M. Bond, J. Clulow, and S. Skorobogatov, "Cryptographic processors-a survey," in *Proceedings of the IEEE*, vol. 94, no. 2. IEEE, 02 2006, pp. 357–369. [Online]. Available: <http://www.cl.cam.ac.uk/~mkb23/research/Survey.pdf>
- [22] B. Schneier, "Why Cryptography Is Harder Than It Looks," 1997. [Online]. Available: <http://www.schneier.com/essay-037.html>
- [23] "PDF Reference," 2004. [Online]. Available: <http://partners.adobe.com/public/developer/en/pdf/PDFReference16.pdf>
- [24] "TIFF Specification," 1995. [Online]. Available: <http://partners.adobe.com/public/developer/en/tiff/TIFF6.pdf>
- [25] W. Koch, "GnuPG does not detect injection of unsigned data," 2006. [Online]. Available: <http://lists.gnupg.org/pipermail/gnupg-announce/2006q1/000216.html>
- [26] X. Wang, Y. L. Yin, and H. Yu, "Finding Collisions in the Full SHA-1," 2005. [Online]. Available: <http://www.infosec.sdu.edu.cn/paper/sha1-crypto-auth-new-2-yao.pdf>
- [27] B. Schneier, "NIST Hash Workshop Liveblogging," Oktober 2005. [Online]. Available: http://www.schneier.com/blog/archives/2005/10/nist_hash_works_3.html
- [28] "Security Advisory: Acrobat and Adobe Reader plug-in buffer overflow," 2005. [Online]. Available: <http://www.adobe.com/support/techdocs/321644.html>
- [29] N. Ferguson and B. Schneier, *Practical Cryptography*. Indianapolis: Wiley Publishing, 2003.
- [30] "Law Governing Framework Conditions for Electronic Signatures and Amending Other Regulations (unofficial version for industry consultation)," of 22. May 2004. [Online]. Available: <http://www.bundesnetzagentur.de/media/archive/1850.pdf>
- [31] "Computer Security Resource Center," 2005. [Online]. Available: <http://csrc.nist.gov>
- [32] "Notification in accordance with the Electronic Signatures Act and the Electronic Signatures Ordinance (overview of suitable algorithms)," Mainz. [Online]. Available: <http://www.bundesnetzagentur.de/media/archive/5171.pdf>
- [33] "IT Baseline Protection Manual," Cologne, 2004. [Online]. Available: <http://www.bsi.bund.de/english/gshb/index.htm>
- [34] "Extensible Markup Language (XML) 1.1," Februar 2004. [Online]. Available: <http://www.w3.org/TR/2004/REC-xml11-20040204/>
- [35] T. Aichelen, E. Day, P. H. Griffin, P. Grme, J. Larmouth, M. Martin, B. Scott, A. Triglia, P. Thorpe, R. Randall, J. Messing, C. Thompson, J. Aerts, and M. Nguyen, "XML Common Biometric Format," August 2003. [Online]. Available: <http://www.oasis-open.org/specs/index.php>
- [36] U. Waldmann, D. Scheuermann, and C. Eckert, "Protected Transmission of Biometric User Authentication Data for OncardMatching," in *SAC '04: Proceedings of the 2004 ACM Symposium on Applied Computing*. New York: ACM Press, 2004, pp. 425–430. [Online]. Available: <http://portal.acm.org/citation.cfm?id=967990>
- [37] B. Schneier, *Applied Cryptography*, 2nd ed. New York, Chichester, Brisbane, Toronto, Singapore: John Wiley & Sons, 1996.
- [38] "Stepover," 2006. [Online]. Available: <http://www.stepover.de>
- [39] A. J. Menezes, P. C. v. Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*. CRC Press, 1996. [Online]. Available: www.cacr.math.uwaterloo.ca/hac
- [40] W. Rankl and W. Effing, *Smart Card Handbook*, 3rd ed. Chichester: John Wiley & Sons, 2003.
- [41] "Softpro," 2006. [Online]. Available: <http://www.softpro.de>
- [42] D. A. Wheeler, "Countering Trusting Trust through Diverse Double-Compiling," in *21st Annual Computer Security Applications Conference*, Tucson, Arizona, 2005. [Online]. Available: <http://www.acsa-admin.org/2005/papers/47.pdf>
- [43] "Common Criteria Protection Profile for Biometric Verification Mechanisms," Bonn, Essen, Dezember 2004. [Online]. Available: http://www.bsi.de/cc/pplist/Biometric_PP_final103.pdf