2000

Full Bindingness and Confidentiality: Requirements for Secure Computers, and Design Options

A. Weber
University of Freiburg, aweber@iig.uni-freiburg.de

search

Full Bindingness and Confidentiality

Requirements for Secure Computers, and Design Options

A.WEBER

University of Freiburg, Germany

Aweber@iig.uni-freiburg.de

***Abstract*-As electronic commerce will increase, players will increasingly wish to have signed documents and strong encryption. According to current knowledge, it must be expected that the costs of attacking, through Trojan horses, crypto implementations on computers with mainstream operating systems, are low. This threatens legal certainty, either because such Trojan horses will emerge, or because their effect will be claimed. Also, they may attack encryption tools. It is discussed that it should be possible to build secure systems in a way that impersonation and eavesdropping is completely implausible. Some requirements for such solutions are discussed. It is concluded that on different market segments, different computers will be used, varying essentially in functionality, screen size, means of input, and degree of tamper resistance.**

I. INTRODUCTION

Components such as software and smartcards for encrypting or for digitally signing documents are typically used in computers which cannot reliably be protected against viri or Trojan horses from the Internet. Even companies who manufacture such components have, in the past, been victimised. Regarding encryption, it must be expected that either the secrets will be easvesdropped when somewhere in the clear, or the passwords, or that "dual use" encryption software will install itself unnoticed by the users. Regarding signature solutions, it must be anticipated that other information will be visualised than signed, which threatens both the signing and the relying party. Alternatively, signers might in case of a conflict simply claim to have become a victim. These threats are discussed in Section II. Subsequently, existing countermeasures are discussed (III). In Section IV, secure computers are discussed as a remedy. They can be designed to protect critical applications from untrustworthy code even in the hands of laypersons. User requirements and design options for these are discussed. It is concluded that, today, secure Linux PCs and PDA-phones are the most promising objectives of migration (Section V). Without such secure computers, electronic commerce could be threatened by attacks or just rumours and claimed attacks, which may severely deteriorate usage of electronic commerce.

The scenarios imagined for this paper are transactions the value of which is definitely beyond what players wish to bear as a damage. Today, they typically demand, as a business rule, a signed paper document. In electronic commerce, with high value, they won't wish to rely on good will case of disputes. Values worth the effort will be concerned if, e.g., somebody does remote banking owning accounts worth Euro 100,000, or if a self-employed person makes a Euro 10,000 transaction, or if a small company makes a Euro 1,000,000 contract. With these types of transactions the threat for the players is significant, while the number of future players can be anticipated to be in the range of at least hundreds of thousands, thus allowing for mass production of the means of protection discussed. Of course, large companies which can bear larger risks will also use the tools if cost-efficient.

II. TROJAN HORSES

A. Direct Trojan Horse Attacks

1. General

This paper[1] discusses Trojan horses being created with the intention of damaging somebody in electronic commerce over open networks. "A *Trojan horse* is a computer program that appears to the user to perform a legitimate function but in fact carries out some illicit function that the user of the program did not intend." ([12] p. 75). In the context of electronic commerce, Trojan Horses can be used to steal secrets, such as business data, keys or passwords, or to misrepresent data to be signed or verified. It has been demonstrated how easy it is to develop a login-window being a Trojan horse capturing passwords [37]. Alternatively, a Trojan horse could look like an update of a program, a useful plug-in, a macro, a tool apparently providing free access to something, etc. A Trojan horse attack may be restricted to a few recipients or even target one single victim only. Individuals could be specifically deceived at different times and in different places, so that they would have considerable difficulty in making the damages believable. Also, the Trojan horse could delete itself after the attack was made.

Trojan horse incidents have been reported in the security literature [23, 13]. Examples include a modification of Telnet which captured all passwords ([23], see also [34]). An example of a Trojan horse which caused financial damage was one which produced excessive phone bills to the benefit of insiders with a foreign telecom provider [26]. In [28] it was reported about a virus which infected a homebanking transaction using a Java-applet. In March 1998, some German pupils developed a Trojan horse attacking password files in implementations of Internet access software of the provider T-Online.

The Melissa virus demonstrated that users can easily be lead to neglect any security precautions such as not to run code from untrustworthy sources. Reportedly, it displayed a

[1] The author wishes to thank Gérard Lacoste, James Riordan, Josef Siman, Dimitri Tchekoff, and Michael Waidner for valuable suggestions. This work has been partially supported by EU-project SEMPER, but it represents the author's view. Some of the findings presented are based on surveys made for SEMPER.

message: “Here is that document you asked for ... don't show anyone else ;-) ” 1.2 mio victims have been reported [3, 31]. Also the ExploreZip Worm of 1999 fooled users. It gave the impression that an individual the recipient knows sent compressed files, and used a Winzip Icon to make users run the malicious executable [21].

Note that Trojan horses can also be put into hardware. Reportedly, a hacker had manipulated CAD-files of Motorola cellular phones. “When Shimomura tracked Kevin Mitnick and captured him ... the files that ... he discovered cached away by Mitnick included ... CAD files... What it means is that if I can take a CAD file from Motorola’s cellular ASIC that will be manufactured in quantity millions, alter the CAD file, put a hardware Trojan horse in, put them back, and have them compile into hardware, which is then embedded in manufactured devices that go out in quantity millions, I’ve created a basic insecurity in those devices in which we trust everything.” [11]

2. Encryption

Recently the US government discussed to “covertly gain access to personal computers” for obtaining confidential information [7]. The FBI is planned to be granted $80 mio. for research on methods for decrypting messages [14]. The Australian Security Intelligence Organisation reportedly wrote: “The opportunity may present itself... to alter software located in premises used by subjects of intensive investigation... The software (or more rarely the hardware) may relate to communication, data storage, encoding, encryption or publishing devices... (S)ome modifications ... may create an intelligent memory, a permanent set of commands not specified in the program written by the manufacturer... The advent of widespread use of strong encryption ... will necessitate ... methods ... to acquire keys or passwords.” [3] It has been discussed in crypto groups that one could produce backdoored versions of crypto applications, such as SSL or PGP, and replace the real ones on people’s computers (“dual use” crypto). Reportedly, the European Union’s confidential positions in the GATT negotiations in 1995 has been communicated to the US, through a Trojan horse in routers of the European Parliament [32].

3. Digital Signatures

It must be expected that criminals will aim at attacking digital signature solutions through Trojan horses. Already in 1994, in the discussions preceding the creation of the German Digital Signature Law, Roßnagel et al. demonstrated that a misbehaving employee can change files another employee wishes to sign without the latter noticing the change [29]. This was not a Trojan horse attack through software on the network, but rather a physical attack through a human being in the room, but it cannot be excluded that such an attack can be done remotely and cheaply.

The significance of the threat is also visible on the Europay France homepage. They define, using the well-known word ‘virus’: “‘(N)on-WYSIWYG‘ viruses: refers to viruses which are capable of displaying on-screen different information from that used in the actual payment under way... (They) are programs in their own right which can be transmitted by means of an infected diskette, through a local network, an FTP download, a message received with an attachment, or simply by viewing an Internet Web page (via Java applets or Active X). Some viruses (although still rare) act as logic bombs, lying dormant in a system, awaiting specific trigger events before acting. Possible trigger events include the detection of an X509-standard file which might constitute a customer certificate, the presence of a non-encrypted card number along the computer's internal bus, or the entry of PIN used to unlock a credit card number.” [10]

Also in the US, already in 1995 a White Paper by the APSON group (Advanced Payment Systems for Open Networks) arrived at the conclusion that “protected devices” are required for secure payments: “Protected devices encompass more than the card/token, but include reader/keyboard (I/O) as well. This is to prevent Trojan horse attacks for input and interfacing the protected device.” ([2] p. 7)

In 1998, Juenemann, Security Architect of Novell, discussed Trojan horses attacking signature solutions [17]. The issue also emerged in debates organised by the US Federal Trade Commission about consumer protection in electronic commerce, where Ellison (Intel) and Winn wrote: “It would be more difficult for malfeasors to access the key used in the consumer's authentication procedure if the systems for controlling access stored the key on a separate token such as a smartcard or required a biometric identifier. Even such more sophisticated access controls may be defeated by attacks such as virus software running on the consumer's computer but not under the consumer's control and without the consumer's knowledge.” They conclude: “There are no systems which remain trustworthy when exposed to normal consumer Internet use and software acquisition.” [46]

4. Significance

Also researchers from IBM point out that Trojan horses are a significant threat, such as Waidner or Gordon and Chess. “The most fundamental problem in security in general is to find a trustworthy computing base. In electronic commerce ... this is particularly important for end users’ devices because anything a user can do online, a successful attacker against these devices can do, too.” [40] Or, regarding business use: “(A) tailored Trojan horse attack could be devastating to a business”. [13]

Such attacks could not only be made by criminals through the network, but also by network managers, maintenance or repair staff. Also insiders could perform attacks, which would be very difficult to detect. In cash dispenser systems, dishonest bank employees repeatedly withdrew money from customers’ bank accounts. The first victims had a very difficult time to make themselves believed [16, 45, 1].

In general, Trojan horse attacks are made more difficult if smartcards are used to store keys and perform the crypto processes. For the attack to work, the card needs to be inserted. Nevertheless, it must be assumed that there are ways to attack smartcard-based solutions. Confidential information could be communicated to an eavesdropper as long as the text is in the clear elsewhere. In signature applications, what is displayed could be different from what the smartcard actually signs. Either the document would be altered and the fake one displayed for only a very short period of time, so that it appears like a flicker. Or the Trojan horse would have the smartcard sign a second document, unnoticed by the user, possibly later. In interviews experts estimated that it might be a few days’ or

a few weeks' work to develop a Trojan horse which attacks a smartcard-based solution on normal PCs on the Internet [42].

The relevance of Trojan horses can also be seen from the fact that banks and certification authorities request from customers to keep their systems free of any malicious code. Verisign prescribes "each certificate applicant shall securely generate his, her, or its own private key, using a trustworthy system, and take necessary precautions to prevent its compromise, loss, disclosure, modification, or unauthorized use" [36]. In their 'Frequently Asked Questions', they demand to "(t)ake measures to protect your computer from viruses, because a virus may be able to attack a private key" [39]. Also German certification authorities, such as TC Trust and Deutsche Telekom request that users make sure that they protect signature components against the influence of Trojan horses.[2]

B. *Indirect Trojan Horse Attacks*

Whereas so far we have been thinking of a criminal attacking a signature or encryption application, in a way that either other data are signed than visualised, or encrypted data are sent in the clear to the eavesdropper, there is also the threat that passwords be eavesdropped and abused in a different context, such as a homebanking application or a physical cash dispenser. It must be assumed that in particular users operating many access systems will re-use passwords and can thus be attacked in other areas.

C. *Claimed Trojan Horse Attacks*

A different threat is that a signer may claim to have been impersonated without actually having been so. This is a risk for the relying party. It may come into effect, e.g., at court when a judge rules that the conditions imposed by a certification authority onto a signer are unfair as impersonation cannot be prevented when using normal computers on the Internet.

D. *The Risk*

To best knowledge, little damage has yet been produced with Trojan horses attacking e-commerce solutions. It must be anticipated, however, that with the increase in e-commerce, attacks will increase.

Regarding digital signatures, once they will be in widespread use, attacks will probably be aimed at, e.g., by organised crime, much like attacks on debit cards, or European Sky-TV and phone cards only emerged after massive deployment had taken place and thus attacks became economically attractive. The risk of a signing party claiming, in case of a dispute, the effect of a Trojan horse is difficult to estimate. But it must be assumed it exists. Also, it must be expected that hackers will try to demonstrate any weaknesses to the public. In any case, as certification authorities demand that users protect themselves against malicious code, the risk seems worth to be addressed.

Regarding encryption, it must be expected that economic espionage will be aimed through other channels, now that business secrets are increasingly strongly encrypted when transmitted.

Though one might be tempted to conclude from current low damage that these risks are small, this can change quickly, as the denial of service attack early in 2000 has shown. Or, as Neumann put it: Risk analysis is risky.

III. Existing Countermeasures

A. *Software Scanners*

Typical advice given to users is (1) to refrain from running untrustworthy code and (2) to use software scanners. Regarding the first, this is in contrary to common practice of computer use of Internet users. Both business and private users regularly have to obtain new software in order to be able to use their computer as anticipated. Such software may contain security holes, and it is very difficult to check that its origin is really trustworthy. Furthermore, normal computer use increasingly requires the download of plug-ins and applets which makes this task even more difficult [24]. While it may already be regarded unfair to request from a user to refrain from using such code, popular operating systems have a tendency to increasingly execute code automatically, so that in normal operation the user won't even notice that untrusted code is running [22].

Therefore the advice is given to use software scanners such as Anti-Virus-software. Such software, however, cannot necessarily detect a new Trojan horse. The Melissa virus demonstrated that virus-protection can take days until it takes effect [8]. In 1999, viri caused large security and computer companies to temporarily shut down their email connections. Thus, demanding from users to keep their machines secure using scanners means demanding something being close to impossible. The demand also faces the risk to be regarded unfair at court.

Thompson [36] described how difficult it is to detect Trojan horses. Scanners also cannot easily detect self-deleting Trojan horses [13]. Last but not least, any attempt by secret services to obtain business secrets through malicious code may not necessarily be detected.

B. *Code-signing*

Possibly, future mainstream operating systems could use code-signing mechanisms extensively. It is doubtful, however, whether this will produce sufficient protection for the following reasons:

1. The operating systems may not be designed well enough to offer reliable protection.
2. Many programmers will be allowed to produce signed code, so criminal programmers will obtain a chance to distribute malicious code. Foreign entities doing espionage will certainly have means to get code signed.
3. Should certification of programmers or software companies be handled very restrictively in order to combat the aforementioned problem, there will be many programs which users will wish to run which are simply not signed.

[2] TCTrust requests "Es ist sicherzustellen, daß sich auf den verwendeten Geräten keine Viren oder schädigende Software befinden, die zu einer Preisgabe der Identifikationsdaten oder der geheimen Schlüssel führen können, oder den Signier- oder Signaturprüfvorgang verfälschen können." [33] Deutsche Telekom recommend: "Verhindern Sie eine Beeinflussung der Signaturkomponenten durch Computerviren, trojanische Pferde..." [6] p.7

4. There is also the risk of certification authorities not working properly or of fake certification authorities.
5. Furthermore, there is the possibility to change certificates unnoticed by the user [30].
6. Signed code containing a Trojan horse may delete its signature after the attack has been made [18].
7. Using signed programs, it might be possible to create malicious code, such as macros, which would possibly not be detected as unsigned code.

Of course, one cannot exclude that future popular large operating systems for laypersons will be highly secure, but, judging from past experience, one has to anticipate the opposite [24].

C. Special User I/O

In the realm of digital signatures, special user input and output has been discussed to become a remedy. Software viewers have the potential to be very flexible for displaying different types of documents through some sort of bitmap. The German Digital Signature Law requests secure visualisation: "Technical components with safeguards are required for the generation and storage of signature keys and for the generation and verification of digital signatures which reliably reveal forged digital signatures and manipulated signed data and provide protection against unauthorised use of private signature keys." (German Law for Digital Signatures, § 14, Section 1) Also the EU Directive says that users, when creating an "advanced electronic signature", should do this "using means that the signatory can maintain under his sole control" (Article 2). They request in Annex III that the data "can be reliably protected by the legitimate holder against the use of others", and that "secure signature creation devices must not alter the data to be signed or prevent such data from being presented to the signatory prior to the signature process".[7]

It is being debated whether such secure visualisation is needed. The ordinance accompanying the German law states that such secure components will need only be used "as required"[3]. The EU discusses in its introduction: "Whereas Annex III covers requirements for secure signature-creation devices to ensure the functionality of advanced electronic signatures; whereas it does not cover the entire system environment in which such devices operate..." (item 15 of introduction) Nilsson et al. interpret this section: "This raises the question on whether or not the presentation of the data to be signed must be presented by the 'secure signature creation device' and whether the presentation shall be done securely. The answer in the Directive is left open" [25]. We must expect that the issue will arise in future court cases. Most likely the ambiguity stems from the fact that no secure solutions exist yet.

Viewers have started to emerge on the German market. One has been certified according to ITSEC E2 under the assumption that the machine is free of "untrustworthy software." [5]. Debis-ITSEC refer to a recommendation that the users of the Utimaco Sign&Crypt solution, which complies to the German digital signature law, are advised "not to install untrustworthy software" [5]. Such software viewers for normal operating systems will most likely not qualify for E4. So the use of such software viewers will make attacks more difficult, but still does not provide a very robust solution, as users are assumed to keep their machines secure.

Special hardware viewers have the potential to provide a much better protection, such as the Bull *Safepad*, a smartcard reader with a PIN-pad and a 32-characters display. Such a special signing device can probably be well protected against malicious code. Three questions, however, emerge:

1. Why would private users be willing to buy special devices? Or, alternatively, will banks or certification authorities be willing to subsidise them?
2. Why would business users be willing to pay for them if normal business documents cannot be viewn or edited on them? If, alternatively, one would equip them with sufficient means, wouldn't they become as expensive as a full-fledged palmtop computer or PDA?
3. As such devices have certain costs, it would make sense that holders use them with different computers. Would they be willing to carry them around for using them in the office, at home, while travelling or perhaps even at the point of sale?

Also plans by companies such as Brokat or Sonera to use mobile phones as secure input and output show that players see the risk needs to be addressed. Here, the task is to design subscriber cards and mobile phones sufficiently secure. Again, it is unclear how to handle normal business documents.

With both mobile phones and special smartcard readers, there is a possibility that they will be designed for only a few applications approved by the issuer, in order to control the security of such devices. This in turn will lead to inflexibility on the user side.

D. Insurance

An alternative to developing highly secure devices might be to insure the risk. Whereas, regarding signatures, this is a possibility investigated by insurance companies[4], we believe research for technical remedies is needed for three reasons:

1. Economically weak parties may need protection as there will be no usable proofs, much like with physical theft. The insurance company will have little means to tell a genuine victim from a criminal pretender.[5] Thus, a genuine victim faces the risk of not being able to successfully claim compensation, as s(he) has no proofs. This may be a moderate threat in low value transactions, and also one which very large companies can bear. Private persons, however, when doing valuable transactions, as well as small companies and self-employed persons when participating in medium-value electronic commerce etc. (above Euro 10,000 or 100,000), face a substantial threat.

[3] "nach Bedarf", see § 16 of SigV

[4] See, e.g., the E-Certify Indemnity Policy by Hiscox [15], though it does not seem they insure relying parties trusting in signatures made by impostors.

[5] Similarly, sometimes graphologic experts are in a difficult position in disputes of signatures on travellers' cheques. In [39] it was reported about a case of lost traveller's cheques in which the customer did not get the money back as the expert ruled that the second signature was the victim's. With digital signatures, it will be impossible to base a decision on whether a signature is genuine or fake on the disputed signature.

2. Once digital signature technology will be in wide-spread use and becomes attacked, damages may quickly increase and insurance companies will demand technical countermeasures in order to avoid unbearable insurance premiums.
3. Regarding confidentiality, insurance may not help anyway.

IV. Secure Computers

A. *Secure System Design*

Summarising we can say that without secure devices with secure user input and output, legal certainty and confidentiality are at risk. This is of concern for private persons, small companies, and even large organisations if the attack is substantial. What is needed are secure computers which are designed in a way that [27]:

1. Trojan horses have no chance;
2. Users unambiguously see what they handle;
3. Users are securely identified; and
4. Lost or stolen devices cannot be abused.

In general, such devices will have to be open for different applications, as chosen by the user. This means that if a large provider, such as a telecom company or a certification authority decides which applications will be allowed, this will be insufficient in many environments. For instance, a competing telecom company may not be able to recommend the use of code of their choice. Or it will not be possible that a buyer selects an encryption program of his or her choice. Note that, e.g., companies or groups of companies will wish to install code issued by themselves. Substantial scope for altering the code is also required because program updates will be necessary. Small systems with a closed design, allowing, e.g., only for display of a few characters in a signature solution, can probably be designed well, but this does not solve the problem in general, and faces the risk that it does not sell well. The issue of running untrustworthy code will certainly become more important once PDA-phones exploit larger bandwidth.

Thus, in general an open design is needed. This means that a secure operating system is beneficial [12]. Loscocco et al. already pointed out that unfortunately providers of security components do not sufficiently acknowledge their solutions' dependence on the operating system's security: "If security practitioners were to more openly acknowledge their security solution's operating system dependencies and state these dependencies as requirements for future operating systems, then the increased demand for secure operating systems would lead to new research and development in the area and ultimately to commercially viable secure systems. In turn, the availability of secure operating systems would enable security practitioners to concentrate on security services that belong in their particular components rather than dooming them to try to address the total security problem with no hope of success." [20]

So what would be needed is a solid base. Users should be put into a position to make their own choice which application thy trust. For example, they may wish that code signed by their own certification authority is trusted. Thus, a secure operating system will be a crucial element for having secure devices. Such a system can allow the user to run trustworthy and non-trustworthy applications on the same machine, as long as they are well separated. Extensive penetration testing of them is an essential requirement which can be reflected by an appropriate certification level.

A secure operating system and a trustworthy source of the applications is not necessarily enough. It must also be ensured that the hardware does not contain any Trojan horses. This is of particular relevance if high-value business secrets are to be encrypted. This may mean, e.g., that European businesses use chips designed, evaluated and produced in an environment they can trust.

B. *Components*

1. *Secure Cryptotools*

As discussed elsewhere in the literature, secure cryptographic components are necessary. This means, e.g., sufficient key length, preparation for the use of a second algorithm in case one is broken, etc.

2. *Secure User Input and Output*

Regarding output, several possibilities exist. The minimum would be a display as used for WAP-phones. Next would be displays of PDA-size. Large texts would either have to be scrolled through, or a magnifying lens would be used (see Fig. 1), a secure printout made, or a device the size of a laptop computer be used.

For digital signatures, it is possible to split the presentation of the document, i.e. have the bulk of a contract on a normal computer, and only display essential fields securely (see Fig. 2 and [43] for an example). Unambiguous fonts and layout will be an essential element of any such visualis ation.

Fig. 1. Phone with magnifying screen (Kopin)

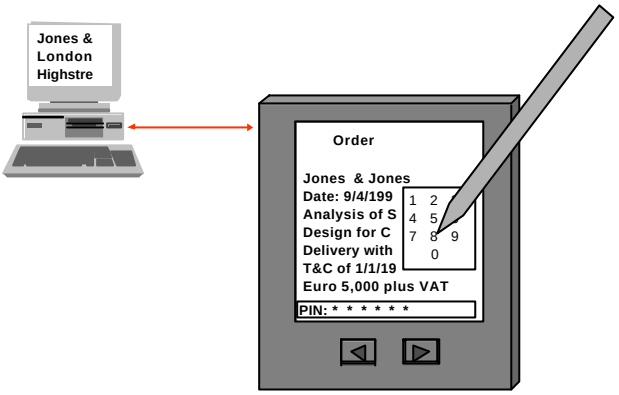

Fig. 2. Split user interface

Regarding input, the usage of passwords and passphrases has been discussed versus the usage of biometric means. While characters can be eavesdropped, biometrics may be fooled and the legitimate owner be rejected, or a criminal accepted. This discussion shall not be pursued here. Note that what should be avoided is to have a PIN-pad which a user only uses for entering a PIN, as this may lead to worn keys, making PIN-guessing easier.

With normal cable connectors, there are reliable ways to make sure with which other devices the secure device communicates. Using contactless interfaces, experience has shown that it is much more difficult to define these. Already in the ambitious St. Moritz trial made by the Swiss bank UBS in 1989, it turned out that the radio interface used to transmit electronic cash sometimes led to unwanted results. For instance, users accidentally touching their pockets found that they unwillingly paid a near-by bus. Also, gates in ski-resorts opened or remained closed unexpectedly because the gate had communicated with a different device than the user believed it had. In project CAFE, using infrared for communicating anonymous cash, pointing was possible. Here, the issue of replay arose, with avoiding double payment. These problems need smart solutions with any contactless interface (see Stajano and Anderson [32] who recommend *contact*).

3. *Tamper Resistant Components*

Different levels of tamper resistance and tamper evidence are possible. One option is to use smartcards. This can mean one or possibly two smartcards of standard bank card size, or one or several of smaller size, as they are used with GSM mobile phones. Alternatively, other form factors can be used. One option is steel buttons (Dallas Semiconductor), another one PCMCIA-cards, and a third is even larger PCI-boards as produced by IBM for use, e.g., in banking. Larger components may have the advantage of storing more data unencrypted, yet securely. Also, they may have sophisticated intrusion detection mechanism and zeroisation circuitry (Fig. 3). Of course, the design needs to be evaluated against different attacks, such as power consumption analysis.

For protection against the threat of the device being stolen, manipulated and returned, the whole device may have to have protection making tampering evident. This can well be combined with holograms for visualising the authenticity of a device, and with challenge-and-response procedures for recognising one's own device.

A securely designed device, combined with appropriate tamper resistance, also has the potential to substitute for special tamper resistant PIN-pads, as they are used in cash dispensers by some banks. This would allow for the production of cheap home-ATMs.

4. *Other Components*

The devices may have to be equipped with suitable storage and back-up media. The usage of solar cells could provide for a high availability. Note that this may be in contrast to the traditional requirement of high speed, which governed the design of security chips, e.g., for use at the point of sale.

C. *Combinations*

1. *Types of Devices*

Future devices will be differentiated in terms of storage, tamper resistance, etc. This will mean that devices will have scope and costs. Note that the inclusion of user input and output into a smartcard of normal thickness (.76 mms) is hard to achieve. Already in 1989 Toshiba built a "Supersmartcard", which, however, had only a very small display, and a difficult to operate keyboard.

Regarding the costs, based on interviews with experts, it appears that in large volumes, the price of the components as mentioned above should not hinder diffusion. I.e. for private use, with a normal smartcard, it should be possible to develop devices with small user input and output for costs in the range of Euro 50. This assumes the production of around 1 mio devices. These costs could even be smaller if the components were integrated into a mobile phone, PDA or palmtop computer. For higher values or maximum "peace of mind", components for re-designed PDAs with higher tamper resistance could probably be manufactured for costs in the range of Euro 100 - 200. It may even be that such a cost level can be achieved with volumes of 100,000 only. Such volumes should not be too difficult to achieve, given the number of self-employed persons, people trading stock, representatives of SMEs, etc. The estimates are mentioned because it has been believed that the marginal costs of such secure devices would rather be in the range of Euro 1,000. I was not able to learn about components which, if manufactured in large quantities, would justify such prices. It will require, however, some effort to make more precise estimates, and to discover market segments in which there is a match of willingness to pay with the production costs for a device capable of handling a given document size and having a certain level of tamper resistance.

2. *Types of Applications*

Secure computers could very well be used with all sorts of business applications, on the Internet or at the Point of Sale, such as signed offers, orders, payments, receipts, etc. Note that with them, if combined with Mixes and eCash, untraceable sales and banking become possible. Other fields of applications are the storage of passwords, electronic tickets and other valuable information. Last but not least applications for use of secure devices will emerge in health or transport.

Assuming one wishes to use a mobile device for securing business processes, the question arises whether it should be combined with a mobile phone. In first interviews made, respondents expressed two possible views. One opinion expressed is that signing with secure hardware should work "via my Palm Pilot or mobile phone. No new device, one should build upon what exists." Another opinion is, however, that secure hardware for signing "should be separate. Otherwise I'd have the problem: 'Give it to me' and the other one makes a phone call. Then the whole mechanism is in hands in which it should not to be. Or it gets left somewhere." [44] How many companies will agree or disagree with the storage of business secrets in mobile phones or PDAs?

V. Conclusion

Given the possibility of an open, highly secure design, combined with a high level of tamper resistance and a proper user identification, there is an option to manufacture devices which in practical terms can be used to achieve *full bindingness and confidentiality*. Of course, in a theoretical sense, there will be no 100% security. There is always a possibility that somebody has broken the system. But it seems the statement "there is no 100% security" has been abused too often for selling components with known weaknesses when being used in insecure computers. With the technologies mentioned it should be possible to make attacks "completely implausible", as one expert put it. This means that if somebody has a liability limit of, e.g., Euro 2,000, using a smartcard to store secrets, a secure system with user I/O as sketched above should leave no risk that this person gets impersonated if the device is lost or stolen. Similarly, with higher levels of tamper resistance and even more sophisticated user identification, it should be possible to make it completely implausible that a transaction worth Euro 1 mio. gets attacked. With a secure design of hard- and software, it should also be possible to encrypt business information in a way that it is practically impossible to decrypt it.

Two migration paths currently appear feasible. One is to turn the current move to Linux into a move towards a secure Linux, usable by laypersons. The other one is to turn mobile phones into secure computers. Somewhat stylised, the two paths have certain pros and cons, as visualised in Table 1.

Whatever option chosen, there will be some costs for tamper resistance and the crypto applications. Actually these might be offset by an increase in performance of a secure system and the smaller resources needed. Thus, peace of mind should become achievable for all users of signature and encryption solutions.

TABLE 1
STYLISED CHARACTERISTICS OF SECURE COMPUTERS

	Secure PDA-Phone	Secure PC
Convenient interface for business use/large documents		x
Convenient encryption of business secrets		x
Suitable for supplementing PCs with traditional OS, for signatures only	x	
Suitable for highest tamper resistance or secure PIN-pads	x [a]	
Personal control	[b]	x
Easy portability	x	
Suitability for mobile transactions, e.g., at point of sale	x	
Suitable for chips free of Trojan horses	x [c]	

[a] Large parts of phone can be made tamper resistant.
[b] With a PDA-phone, one could take the keys out when handing it over.
[c] Likely it is cheaper to control a few phone chips

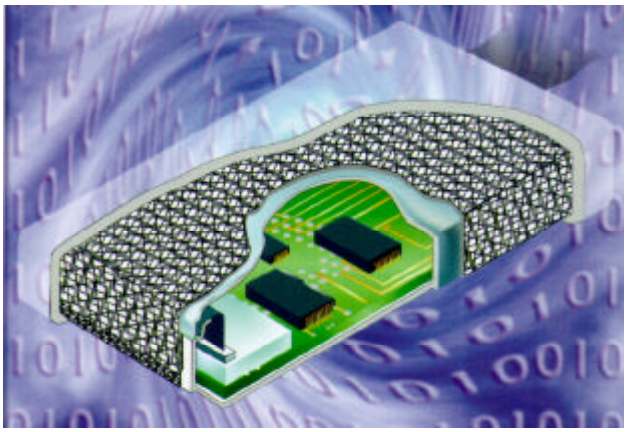

Fig. 3. Membrane for intrusion detection (Gore)

search

REFERENCES

[1] Anderson, R.: Why Cryptosystems Fail. 1st Conference on Computer & Communication Security 1993 (ACM)

[2] APSON-Group (Advanced Payment Systems for Open Networks): White Paper. (1995)

[3] Australian Security Intelligence Organisation, cf. <http://www.newswire.com.au/9911/asio.htm>

[4] CERT Advisory CA-99-04-Melissa-Macro-Virus; 27.3.1999

[5] Debis IT Security Services: Certification Report SafeGuard Sign&Crypt, Utimaco Software AG, No. 04007, 1999, http://www.itsec-debis.de/debiszert.html#Zertifizierungsreports

[6] Deutsche Telekom: Informationen zur Teilnahme am Public Key Service. 1999

[7] Directive of the European Parliament and of the Council of 13 December 1999 on a Community framework for electronic signatures. http://europa.eu.int/eur-lex/en/dat/2000/l_013/l_01320090119 en00120020.pdf

[8] Edupage of 12 April 1999

[9] Edupage of 20 Aug. 1999

[10] Europay France of 10.3.1998 http://www.europayfrance.fr/us/commerce/secur.htm#2

[11] Gage, J.: Comment, on April 3 1997. http://www.sun.com/Keynote_McNealy.html

[12] Gasser, M.: Building a Secure Computer System. New York 1988

[13] Gordon, S.; Chess, D.: Where There's Smoke, There's Mirrors: The Truth about Trojan Horses on the Internet. Paper presented at the Virus Bulletin Conference in Munich, Germany, 1998, http://www.ibm.com

[14] Heise News-Ticker: USA: Krypto-Exporte ohne Beschränkungen möglich. http://www.heise.de/newsticker/data/jk-17.09.99-000/

[15] Hiscox Syndicates: E-Certify Indemnity Policy. 1999 http://www.e-certify.com/Lloyds-insurance.pdf

[16] Jack Committee, Report (Review Committee on Banking Services Law, Chairman Robert Jack). London 1989

[17] Juenemann, R. 16.6.98 : http://lawonline.jp.pima.gov/interim/digsig5a.htm

[18] Kabay, M.: Infosec: The Year in Review. (1998) http://www.ncsa.com

[19] Knight, W.: Crypto could force government hacking. http://www. zdnet.co.uk/news/1999/46/ns-11678.html of 23 Nov 1999

[20] Loscocco, P.; Smalley, S.; Muckelbauer, P.; Taylor, R.; Turner, S.; Farrell, J. (NSA): The Inevitability of Failure: The Flawed Assumption of Security in Modern Computing Environments. Paper presented at 21st National Information Systems Security Conference, 1998. http://www.cs.utah.edu/flux/fluke/html/inevit-abs.html

[21] Network Associates: http://www.avertlabs.com/public/datafiles/valerts/vinfo/va10185.asp 6.7.1999

[22] New Scientist: http://www.newscientist.com/cgi-bin/pageserver.cgi?/ns/980425/nwindows.html 20.9.1999

[23] Neumann, P.: Computer Related Risks. Reading et al. (1995)

[24] Neurotec Hochtechnology GmbH: OCOCAT-S. Object Code and Optimizing Compiler Analyzing Tool. http://www.bsi.de 1998

[25] Nilsson, H.; van Eecke, P.; Medina, M.; Pinkas, D.; Pope, N.: "Final Draft of the EESSI Expert Team Report", Brussels 1999

[26] Overill, R.E.: Computer crime - an historical survey. (1998). http://www.kcl.ac.uk/orgs/icsa/ crime.htm

[27] Pfitzmann, A.; Pfitzmann, B.; Schunter, M.; Waidner, M.: Trusting Mobile User Devices and Security Modules. In: Computer 1997, pp. 61-68

[28] Posegga, J.: Die Sicherheitsaspekte von Java. Informatik Spektrum 1998, pp. 16-22

[29] Roßnagel, A., et al.: Die Simulationsstudie Rechtspflege. Eine neue Methode zur Technikgestaltung für Telekooperation. Berlin (1994)

[30] Schneier, B.: http://www.counterpane.com/crypto-gram-9904. html#certificates of 15.4.1999

[31] Schneier, B.: The Trojan Horse Race. CACM vol 42, Sep. 1999, p.128

[32] Stajano, F.; Anderson, R.: The Resurrecting Duckling: Security Issues for Ad-hoc Wireless Networks. In: Christianson, B.; Crispo, B.; Roe, M. (eds.): Security Protocols. LNCS, Berlin Heidelberg 1999

[33] Sunday Times 1996: http://www.sunday-times.co.uk/news/pages/Sunday-Times/stifgnnws01015.html

[34] Symantec 1999 http://www.symantec.com/avcenter/warn/backorifice.html

[35] TCTrust: Sorgfalts- und Mitwirkungspflichten des Zertifikatinhabers. (Hamburg) 1998

[36] Thompson, K.: Reflections on Trusting Trust. In: CACM, 1984, pp. 761-763

[37] Tygar, J.D.; Whitten, A.: WWW Electronic Commerce and Java Trojan Horses. In: The Second USENIX Workshop on Electronic Commerce Proceedings, Berkeley 1996, pp. 243-250

[38] Verisign Certification Practice Statement. July 21, 1999. https://www.verisign.com/repository/CPS1.2/CPSCH4.HTM#_toc361807021

[39] Verisign: Repository Frequently Asked Questions. https://www.verisign.com/repository/PrivateKey_FAQ/ July 21, 1999

[40] Waidner, M.: Future Directions in Secure Electronic Commerce. In: Lacoste, G.; Pfitzmann, B.; Steiner, M,; Waidner, M. (eds.): SEMPER Final Report. LNCS Berlin et al. Forthcoming

[41] Weber, A.; Carter, B.; Pfitzmann, B.; Schunter, M.; Stanford, Ch.; Waidner, M.: Secure International Payment and Information Transfer. Frankfurt 1995

[42] Weber, A.: “See What You Sign. Secure Implementations of Digital Signatures.” in: Trigila, S.; Mullery, A.; Campolargo, M.; Vanderstraeten, H.; Mampaey, M. (eds.): Intelligence in Services and Networks: Technology for Ubiquitous Telecom Services. IS&N'98. LNCS 1430. Berlin et al. 1998, pp. 509-520

[43] Weber, A.: Sind rechtsverbindliche digitale Signaturen möglich? In: Röhm, A.; Fox, D.; Grimm, R.; Schoder, D. (eds.): Sicherheit und Electronic Commerce. Proceedings des VIS-Workshop "Sicherheit und Electronic Commerce", Essen 1998, Braunschweig, Wiesbaden 1999, pp. 205-218

[44] Weber, A.: Spontaneous Secure Transactions, see <http:/www.iig.uni-freiburg.de/~aweber/>

[45] Whybrow, M.: ATM Security. Ghost in the machine. In: Banking Technology 1991, pp. 39-43

[46] Winn, J.; Ellison, C.: U.S. Perspectives on Consumer Protection in the Global Electronic Marketplace – Comment P994312 to the Federal Trade Commission http://www.ftc.gov/bcp/icpw/comments/revwin~1.htm March 26, 1999