RZ 2784 (#89262) 02/05/96
Computer Science/Mathematics 14 pages

Research Report

Mobile User Devices and Security Modules: Design for Trustworthiness

Andreas Pfitzmann

Technische Universität Dresden
Fakultät Informatik
Institut Theoretische Informatik
D-01062 Dresden, Germany
email: pfitza@inf.tu-dresden.de

Birgit Pfitzmann, Matthias Schunter

Universität Hildesheim
Institut für Informatik
Marienburger Platz 22
D-31141 Hildesheim, Germany
email: {pfitzb, schunter}@informatik.uni-hildesheim.de

Michael Waidner

IBM Research Division
Zürich Research Laboratory
Säumerstrasse 4
CH-8803 Rüschlikon, Switzerland
email: wmi@zurich.ibm.com

Research Division
Almaden • T.J. Watson • Tokyo • Zurich

Mobile User Devices and Security Modules: Design for Trustworthiness

Andreas Pfitzmann*, Birgit Pfitzmann†, Matthias Schunter†, Michael Waidner‡,

* Technische Universität Dresden, Institut Theoretische Informatik

† Universität Hildesheim, Institut für Informatik

‡ IBM Research Division, Zurich Research Laboratory, work mainly done at the Univ. of Karlsruhe, Institut für Rechnerentwurf und Fehlertoleranz

February 5th, 1996

Abstract: There are many plans to supply users with personal devices to improve security in such areas as electronic commerce, payment systems, and digital signature functions. Most of these plans consider only smartcards. However, far more powerful mobile user devices are a rapidly growing market; just consider mobile phones, pagers, gameboys, multifunctional watches, personal communicators, and personal digital assistants (PDAs). Moreover, most mobile user devices naturally need security functions for some of their prospective applications.

We therefore expect these two separate developments to merge to a great extent over the next decade. This opens up new architectural options for security, but also poses new threats. The present article surveys the resulting design issues. Although no device can be completely trustworthy, we believe that the combination of security functions and powerful mobile user devices is to their mutual benefit if appropriate measures are taken.

1 Devices and Applications

A *mobile user device* is any portable device that belongs to a specific user and has computing and storage capabilities. By *security module* we mean any device that is tamper-resistant, i.e., that contains measures to keep data secret and uncorrupted even under physical attacks. A mobile user device may contain one or more security modules, e.g., smartcard modules, or it may be a security module itself, similar to some point-of-sale terminals, or it may be without tamper resistance.

We consider applications where a mobile user device performs a function of legal significance for its user. Digital signatures are the most generic of these functions. More specific examples are payments, both on the Internet and in shops, that are authorized exclusively by a device without a handwritten signature. Current applications are usually sector-specific, such as authorization for mobile phone calls and access control.

Many of these applications naturally require mobile user devices. Even for applications where mobility is irrelevant, it is considered useful to equip users with at least smartcards, the simplest mobile user devices, because stationary computers are typically accessible to more than one person. However, even in this scenario with smartcards and stationary computers, one can obtain a user's digital signature on a document she did not want to sign, mainly by variants of the well-known fake-terminal attack. This means that the stationary computer is manipulated such that, instead of simply relaying messages between user and smartcard, it modifies them or intercepts and stores secrets. (The German research group *provet* demonstrated a large variety of such attacks in practice, but the results are only available in German.) Such attacks can only be prevented if the user's own mobile device has a user interface, i.e., keypad and display. This makes most other mobile user devices more suitable than smartcards for legally significant functions; see Table 1 for the features of some currently available types. Two recent examples of systems using specialized mobile user devices for payments are Mondex[1] and CAFE[2].

2 Types of Trust

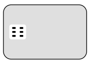
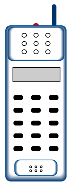
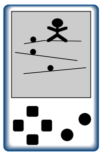
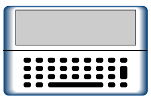
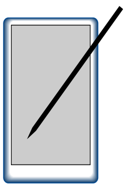
How can we build trustworthy mobile user devices for such applications of legal significance? Note that "trustworthy" is not an objective term — for widespread open applications in a free society, the individual users should be as free as possible in deciding whom they trust and under which circumstances. In particular, trustworthy means both that a device *is* good and that this is credible.

The following types of trust in a mobile user device must be distinguished. As the device *acts on behalf* of someone, we use the analogy of agents.

1. *Personal-agent trust:* For legally significant applications, a user has to be able to trust that her own device acts according to her wishes while she possesses it. For instance, it should not sign unintended statements nor unintentionally delete electronic money.

2. *Captured-agent trust:* The legitimate user may want her mobile user device to protect her even if it was lost or stolen or if she gave it away, such as for maintenance or by inserting it into a point-of-sale terminal. For instance, the finder or thief should not be able to sign statements in the legitimate user's name.

Table 1 Types of mobile user devices.

	Icon (Scale 1:8)	**Communi-cation**	**Display / other user output**	**User input**	**Slots usable for security modules**
Smartcard		serial port	none	none	none
Mobile phone		digital radio transceiver, serial port	2x16 charac-ter/ sound	12-20 keys, microphone	15x25mm or standard smartcard
Pager		radio receiver	2x16 charac-ter/ beeper	1-2 buttons	none
Gameboy		infrared trans-ceiver, serial port	LCD 160x144 dot-matrix, 4 shades of gray/ sound	4 cursor + 2 fire buttons	ROM module (up to 512kB)
Multi-functional watch		radio receiver or infrared transmit-ter	LCD dot-matrix 3x3cm / beeper	4-12 buttons	none
Organizer		infrared trans-ceiver, serial port	60x20 charac-ter LCD dis-play/beeper	QWERTY key-board	ROM modules
PDA or personal communicator		infrared trans-ceiver, serial port, digital radio transceiver	LCD 320x240 dot-matrix/ sound	pen-pad/ microphone	PC-CARD Type II, 15x25mm or standard smartcard for radio modem

3. *Undercover-agent trust:* Other parties may want a mobile user device to protect them from the legitimate user of that device. An important example is prepaid offline payment systems, where users have so-called “electronic cash” in their mobile user devices, and payments in shops can be made without connection to a bank. The shops and the bank want the mobile user device to prevent its legitimate user from spending the same bit string representing “electronic cash” in several shops.

The second and third types of trust require security modules. Note that the third type is not needed in many applications, e.g., in online payment systems or general signature applications. Whether the same security module can be used to protect both the user and other parties is discussed below.

The following design options are sorted according to the type of trust they aim at.

3 Personal-Agent Trust

Personal-agent trust comprises trustworthy design, production, shipment, and personalization of the mobile user devices and trustworthy interaction between the user and her device. We also discuss measures to prevent personal agents from being captured under this heading.

3.1 Recruiting a Good Agent

Design, production, and shipment can be made trustworthy for various end users by many standard measures known from building trustworthy devices for a specific corporate customer, with the addition of more diversity and more inspections of shipped devices.

Design and Production

Standard measures are structured design; quality control of all design and production steps, possibly with verification of small critical modules; and fault-tolerance measures such as error masking, recovery, and containment, including measures against covert channels.

To enable trust by heterogeneous end users, diversity is important. For instance, end users should have the choice between different manufacturers, and each manufacturer should be supervised by independent evaluators, such as consumer organizations. All parties benefit if the manufacturer is independent of organizations that might later be engaged in lawsuits with the end users, e.g., a bank in a payment application.

Shipment

End users have to be sure they obtain authentic devices from the production process they trust. Naturally, one should take precautions for secure delivery. However, this is much more difficult to survey comprehensively, in particular for diverse evaluators trusted by different end users, than shipment to a few corporate customers. Hence some checks on shipped devices are advisable.

To exclude fake devices from *outsiders*, all security-critical parts of a device, including the user interface, need some physical protection, at least, e.g., holograms. Otherwise an adversary can reassemble hardware and software parts of authentic devices in ways virtually impossible to detect by the users. A security module can additionally prove its authenticity by a challenge-response mechanism. For instance, the user receives a random challenge, c, and a response, r, from the manufacturer on a different channel than the device. She then verifies that the device correctly answers c by r (Figure 1). This measure assumes the adversary controls at most one of the two channels from the manufacturer to the user, and cannot establish a covert channel between the fake device, e.g., a mobile phone, and the real device; otherwise, the fake device could relay the challenge to the real device.

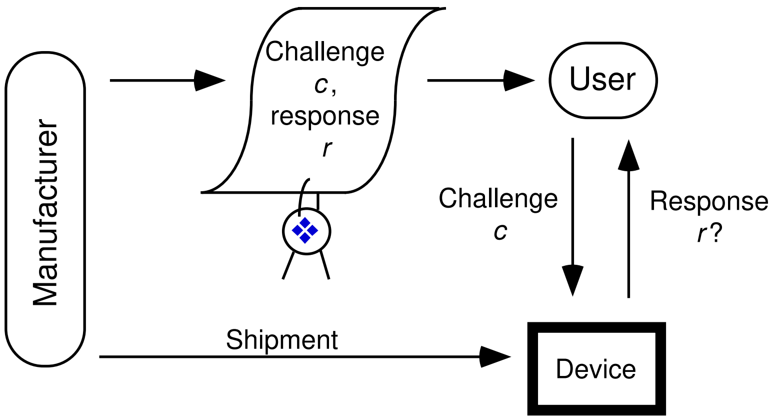


Figure 1 Challenge-response mechanism to verify correct shipment of a device.

It is also conceivable that *insiders* implement devices outside the trusted production process; they could produce devices that pass the above tests.

For software that is delivered separately, this can simply be avoided by appending a digital signature of the body that certified the evaluation of this software. Of course, the user needs a device or a party she trusts in order to check the certificate.

Certificates on the hardware design do not have the same effect, because individual users cannot examine their devices directly to compare them with the certified design. Thus we have to rely on

a) functional tests and

b) direct examination of random samples of shipped devices, e.g., by consumer organizations.

Direct examination by well-equipped organizations seems possible for the circuitry and ROM code of currently proposed smartcard-like security modules, but not necessarily for data in their rewritable memory. The main problem with functional tests is that they do not address the absence of hidden functionality, such as a 10-digit-activation code that causes a security module to output its secrets. One can at most test that all intended functionality is present and hope that it fills the available space completely. In particular, one should do this with rewritable memory of security modules, e.g., by inputting random bits that fill the expected free space completely and asking the security module to output them again (Figure 2). Of course, such tests presuppose that parts of the circuitry of the security module are correct.

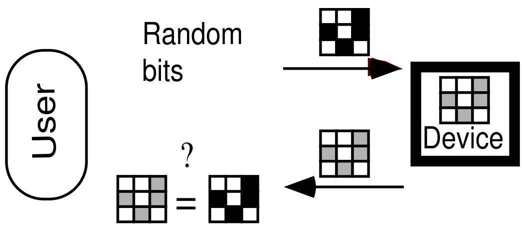


Figure 2 Verifying that a device has no hidden functionality in rewritable memory (9 bits shown).

If the mobile user devices should be indistinguishable to guarantee privacy in some applications, individual characteristics in their analogue behavior, e.g., exact waveforms, may be a special case of hidden functionality. In this case, random samples of the devices interacting with the mobile user devices should also be evaluated regularly by consumer organizations to make sure that such differences are not exploited. Establishing deliberate individual characteristics, e.g., holograms, to allow each *user* to recognize her own device (see below) is no problem as long as such characteristics are outside the contact area to other devices.

3.2 Swearing In an Agent

For the applications considered here, a mobile user device has to be personalized. In particular, it needs a secret key trusted by its user. To ensure secrecy from the start, the key should be generated in the device and never leave it. Only the corresponding public key should be output and certified at the user's request. Storage space for key generation is usually no problem: the code can be deleted afterwards. However, all key generators need an *initial random string*, from a few dozen to a few thousand bits. To ensure the secrecy and randomness of this string, it should be produced by a collective coin flipping protocol. This means that several strings, so-called shares, are combined into one string that is completely random and secret if at least one share is. One share should come from the user; highly security-conscious users will roll dice. To protect less cautious users, the manufacturer should also contribute a share. If the device contains a physical random number generator, it should generate another share. Any further parties interested in the quality of the key, e.g., the certification authority, can contribute additional shares. Collective coin flipping in a trusted device can simply be realized by exoring the shares (Figure 3).

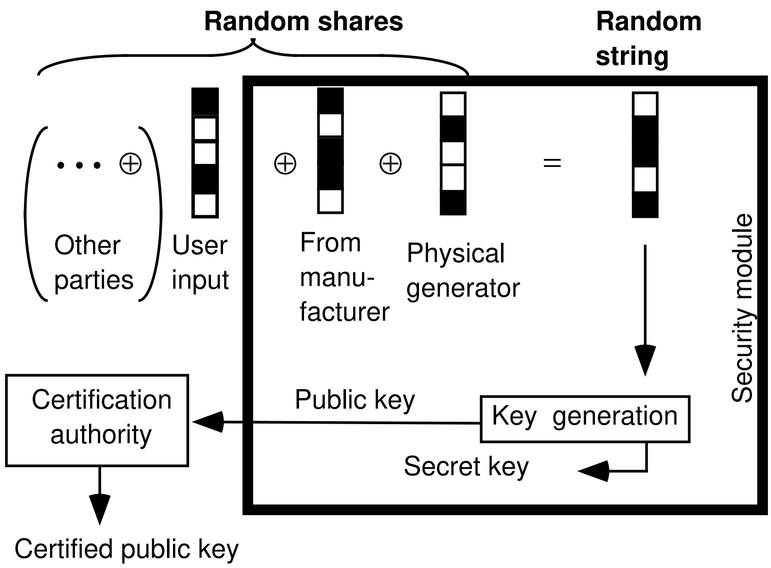


Figure 3 Key generation and certification using collective coin flipping in a security module.

3.3 Briefing and Reporting

Once a user has a personalized, trustworthy mobile user device, how can she communicate her wishes to it? Generally, the answer lies in the good design and documentation of the user interface. We do not attempt to go into ergonomics here, but we argue that some standardization is necessary for security and consider one important example.

Need for Standardization

For ordinary users, devices need a familiar and consistent user interface. Thus interfaces of legally significant applications have to be fairly constant both over different applications and over time, just as the placement of the brake pedal in cars is largely standardized.

For instance, the legal significance of handwritten signatures is not only to show who produced a document, but also to warn the signer that she now makes a legally binding statement. This is not only helpful for the signer, but also necessary for ruling that a signed document expresses the intention of the signer, in contrast to being a draft or part of a longer text. In digital applications, this warning function must be provided by the standardized user interface.

Mobile User Devices as Periscopes

A specific question, but important because of the central role of signing, is how users can be sure what they are signing if some mobile user devices do not have the screen size one would usually need to compose or display the document. Note that a user *must* browse through such a document with her trusted mobile user device, because this device can otherwise be tricked into signing something different.

Smaller documents. Obviously, application designers and users should try to keep the overall size of documents small by omitting all details that have no legal significance. Where possible, the *variable* part of a document should be reduced by using templates. The fixed part that the user need not look at may range from a reference to an external text, e.g., a section of a statute, over forms that the mobile user device obtained with a certificate from a party the user has decided to trust, e.g., a consumer organization, to forms the user herself uses repeatedly and has read once.

Good output channels. The smaller the screen, e.g., on a mobile phone or a watch, the more important high resolution is, instead of the usual small character displays, for scrolling a small window over a larger document. Additional types of output channels, such as voice, may also be helpful.

Enforced browsing. To ensure the warning function of digital signatures, the mobile user device should force the user to look at all significant parts of a document, even if the user does not like this. For instance, for a document with templates, it might show the fields one-by-one for a few seconds, and ask the user to acknowledge each field individually or even to copy the most important items correctly. For large amounts of data, e.g., a 10-page contract, this might seem infeasible. If they cannot be signed on paper instead, the device should at least show random samples, so the user can detect major modifications compared with a version she saw on a larger screen, and allow additional browsing. Nonlinear data, such as hypertext, would need special user guidance based on a standard for the scope of a signature in such data.

Additional half-trusted output channels. For additional outputs in larger format, the mobile user device should contain an interface to standard output devices, such as fax machines, printers, TV sets, or computer monitors. This solution is not perfect, but quite trustworthy because there is a large choice of such devices. It is certainly better than viewing the data via a computer of the recipient of the signature, which may be proposed instead of signing on paper if this computer is unmanned or only in a branch office. An exotic measure to improve such solutions by means of diversity is visual encryption[3], where data are output on transparencies, each of which is apparently random, whereas overlaying them yields the desired data. One can use a constant slide with a key and one printer, or share the data among several printers. This improves confidentiality, and if the data contain redundancy and some randomization, no single printer can simply alter them. Alternatively, a handscanner, easier to build into a mobile user device than a printer, could be used to check the printed output.

3.4 Keeping the Agents Free

If maintenance of a mobile user device is possible, it should have a maintenance mode that does not require the security module, so that a user never has to relinquish it.

To prevent loss and theft, the devices should be shaped such that users are encouraged to wear them on their body, instead of in a pocket or bag or even of leaving them in an office or a hotel room. For many applications, this is currently not possible with the complete mobile user device — multifunctional watches are the only type available so far. This may change if roll-up displays become reality[4]. At present, however, we usually need small *removable security modules*. There are two types:

- *Plug-in security modules* are inserted into the mobile user device for operation. Outside, most of them must be stored in special little containers.
- *External security modules* are always kept separate from the mobile user device. Hence they need a secure communication channel to it. Radio, infrared, or induction seem possible, but they restrict the places where the security module can be worn on the body.

As removable security modules only have to communicate with special mobile user devices, there is no reason to stick to ISO smartcards. Modules that are smaller, sturdier, or that contain a larger chip or several chips already exist on the market; e.g., smartcard chips on a small card, which are part of the European mobile phone standard GSM, pendants for medical emergency data, and steel “buttons” (Figure 4).

External security modules and containers for plug-in modules could be built into watches, bracelets, necklaces, earclips, lockets on key rings (although many people are not sufficiently careful with keys) or even finger rings. Several variants should be offered, so that most people find one they like.

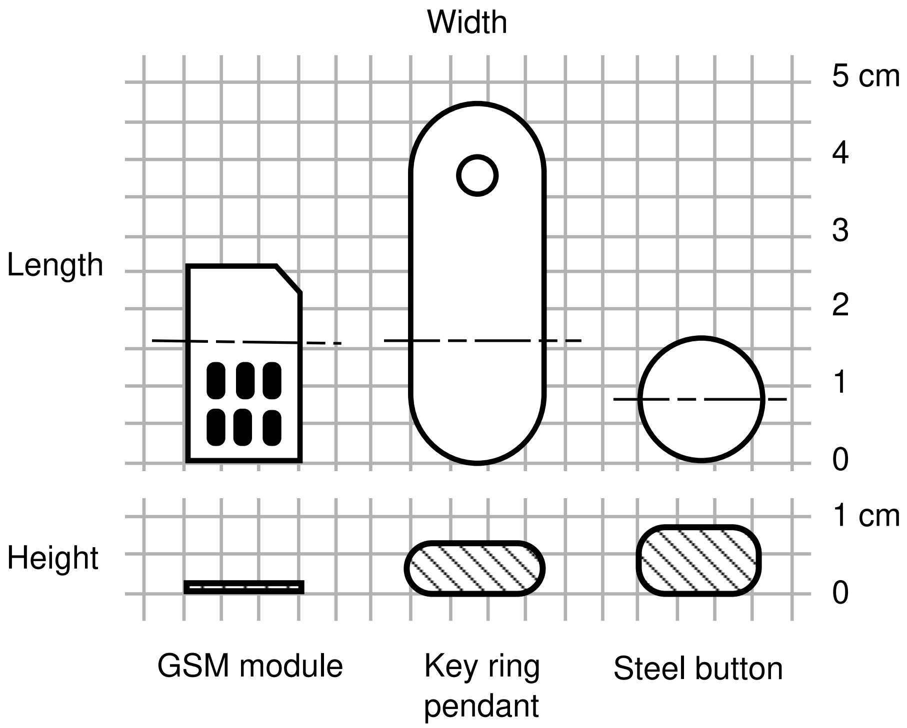


Figure 4 Different shapes of existing security modules.

4 Captured-Agent Trust

4.1 Friend or Foe

First, a device that has fallen into the wrong hands has to notice this, i.e., it needs user identification. The relative advantages of techniques like PINs (personal identification numbers), passphrases, and biometrics are well known[5]. Thus we only consider their integration into applications.

Ideal Times for User Identification

Ideally, user identification is needed for each security-critical user command, in particular, whenever

- a signed statement is to be sent that may be held against the user or
- a decision is made for private data to be released.

To prevent an adversary from keeping a stolen device operational, because tamper resistance is easier to break during operation, user identification is additionally needed at fixed time intervals. For external security modules worn directly on the body, one can try to find a cheap form of biometrics, such as temperature measurement or heart beat monitoring, that need not be able to recognize a user, but notices if the security module is removed from the body.

Managing the Risk of Observability

Most types of user identification can be observed by adversaries. PIN entry can be seen, fingerprints can be obtained from a used glass, keyboards emanate radiation while the PIN is being entered, and a parcel service currently lets recipients sign on a pen-pad and can therefore observe the dynamic signatures of a significant part of the population. Such observations can be encumbered, but not made impossible.

Hence the risk of observation should be weighed against the security gained by requiring user identification for a certain type of command, and the commands should be sorted in different risk classes. For each risk class, a different type of user identification should be used. For instance, the user identification for high-value money transfers should not be put at risk by using it for low-value payments in shops. In the case of PINs, users who can remember only one should use it for the highest risk class and not use any or only trivial ones for the other classes. Even a very simple PIN for a lower risk class is better than none — it still reduces the potential profit of an attacker who steals devices at random.

In some situations, the risk may be reduced by performing user identification before the critical command. For instance, at a gas station, one could unlock the expected amount while still sitting in the car and later merely confirm the exact amount by "OK," a trivial PIN, or a one-time PIN chosen in the car.

User-Friendliness

If the frequency and strength of user identification is a compromise between user friendliness and security, the choice should largely be left to the individual user.

For instance, users may be too lazy to use a PIN for spending "small" amounts. Each user should decide what she currently considers "small" by unlocking a total amount that can be spent before the PIN is needed again.

Users who like long PINs or passwords should not be restricted to 4 or 6 digits. This may be supported by alphanumeric PINs.

If there is a cheap type of biometrics that works with most people, one can implement it and either deactivate it on the devices of the remaining people or offer them even cheaper devices with PINs instead. Biometrics that fails when a user is ill or in other extreme situations is not suitable for the applications considered. More generally, if biometrics is used, one should start with a high tolerance so that the authentic users are never excluded, and use PINs in addition for high-risk inputs.

There may be an additional emergency PIN that can be used in case of temporary extortion. It should cause transactions that appear valid at first, but are not.

Device Identification

So far, we considered how the mobile user device identifies its user. For protection from fake-device attacks, the user also has to identify her device. The standard method is physical characteristics like holograms or implanted photographs, which can be recognized by the user. Care should be taken that this does not corrupt the user's privacy.

If the user has a second device she trusts, more secure cryptographic protocols for mutual identification of devices can be used. Typically, the second device would be a plug-in security module. On personalization, the security module receives a special device identification number, DIN, from the user. For mutual identification, security module and mobile user device first identify each other (Figure 5). If this is successful, the security module, which has no display, shows the DIN to the user via the mobile user device. If the user sees the correct DIN, she enters her PIN into the mobile user device. Now the security module can be removed. For this technique, just as for that in Figure 1, one has to ensure that an adversary

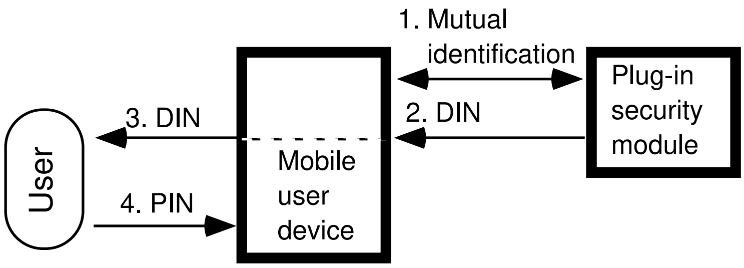


Figure 5 Figure 5. A protocol for identification of a mobile user device.

cannot establish a channel between a fake device and a real device. Countermeasures are physical protection or so-called distance bounding protocols[6].

These two techniques can be combined. For instance, a user might normally be satisfied with weak identification by a hologram, but use strong identification with a plug-in security module if the mobile user device had not been under her control for some time, e.g., during maintenance or in a hotel room.

Nevertheless, user and device identification is currently such a problem that it may be prudent to allow transactions of very high value only in the presence of a notary public.

4.2 The Agent's Armory

If a finder or thief cannot use a device in a normal way, she may try to extract its secrets. For such cases, the device must be equipped with tamper resistance.

Levels of tamper resistance are not precisely measurable. Certainly, no device is tamper-*proof*. Given typical articles[7,8], one can assume the following design rules for devices and applications employing security modules.

Design Issues

Tampering is much easier while a security module is operating. To minimize the risk after a loss in an operational state, auto shutdown should occur at least at the end of transactions and after a certain period of time of no user input. However, even a captured agent must, for example, allow three PIN entries. The information observable during this is minimized if each bit of the stored secret PIN is addressed and compared individually.

Different types of memory have different degrees of tamper resistance. In particular, reading the ROM of a smartcard-like security module seems fairly easy for a well-equipped attacker. Hence security should not rely on the secrecy of algorithms or keys stored in ROM. Secrets in erasable memory, such as EEPROM or battery-powered RAM, are more secure. If the security module has an internal power supply, secrets can even be deleted if an attack is detected. Fast deletion is crucial, for which battery-powered RAM is optimal. This is one more advantage of security modules larger than a smartcard chip.

For some applications, the user interface of a mobile user device should be integrated with the security module. Thus the whole mobile user device is a security module, but additional security modules may be connected to it. Otherwise, fake-interface attacks are possible, e.g., internal manipulation to intercept a PIN or external exchange of the lettering of the “ok” and “cancel” buttons. Some attacks of this sort are likely to happen; however, for mass fraud, they seem less profitable than observing PIN entry directly or selling fake devices. Note that a secure display may be useful even without a full secure keyboard: The user can confirm commands simply with a secure “OK” button. Even PIN input without any secure keyboard is possible[9]: The display shows a random number, and the user “corrects” it into the PIN.

Some applications require a reliable clock and a real random number generator. Thus, a universally applicable security module should provide these functions.

Risk Analysis

Even with well-designed security modules, risk analysis is necessary: Are the efforts needed to break one and the risk of detection in good relation with the advantage the attacker would gain?

The risk that someone will steal and *break* devices with security modules to defraud their individual users of low values seems relatively low, or at least lower than the risk of PIN observation. If users can move high values via mobile user devices, breaking tamper resistance may become worthwhile. Hence caution is recommended in connection with home banking of large sums and general signature applications. Violating a particular user’s privacy will usually not be worth the effort of breaking a security module. Security agencies, however, may be prepared to do so if the data to be found make backward surveillance possible.

5 Undercover-Agent Trust

For the trustworthiness of undercover agents, only design and production, personalization, and tamper-resistance have to be considered. In particular, the party protected by an undercover agent only communicates with it via other *devices*, e.g., the bank via its central computers. Thus user identification is replaced by cryptographic protocols for the mutual identification of devices.

5.1 Recruiting

The standard measures for the trustworthy design and production of undercover agents were already mentioned above, as they are also needed for personal agents.

Double-Agent Security Module?

In many applications, both the user and another party have to trust a mobile user device. We speak of double-agent trust if both trust the same security module for this, and of multi-agent trust if several other parties do so, e.g., the user’s bank, phone company, and employer.

The main argument for double- and multi-agent trust is cost. However, at least smartcard-like security modules are very cheap compared to mobile user devices. Multi-agent trust is also convenient because users do not have to change security modules. The main counterargument is that, as a fact of life, it is unlikely that many parties agree on one security module. For double-agent trust, just imagine whether banks would trust security modules where the users had the same rights to choose manufacturers and evaluators that banks usually have. Thus multi-agent trust is ok if it can be achieved, but otherwise it is pointless to insist on double-agent trust, because that only decreases the number of security modules of a user interacting with n other parties from $n + 1$ to n, and forces her to trust n other parties that do not trust each other.

Furthermore, it may be impossible to implement all required applications on just one security module. However, this can be supported if security modules can swap signed applications or even encrypted data into untrusted storage in the rest of the mobile user device.

Interaction of Opposing Agents

If an undercover agent and a personal agent are separate parts of a mobile user device, protocols for their interaction are needed.

Basically, actions are only valid if authorized, e.g., signed, by both agents. If the undercover agent can communicate with other devices only via the personal agent, the personal agent's authorization may simply consist of letting messages pass.

If privacy is desired for some applications, the undercover agent should not have covert channels with other devices (Figure 6). Thus, if messages from the undercover agent contain randomness, e.g., in the digital signatures, the personal agent must transform them. Examples of such protocols are the so-called wallet-with-observer protocols.[10]

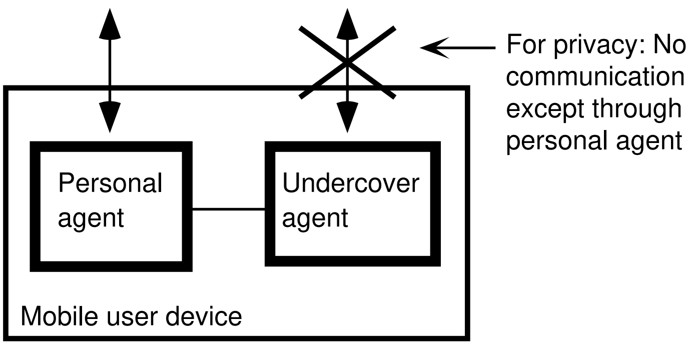


Figure 6 A mobile user device with separate security modules trusted by the user and another party.

5.2 Swearing-In

Double-agent trust in personalization can simply be achieved by letting the other party contribute an input string to collective coin flipping.

5.3 Armory

Tamper resistance for undercover agents is the usual case in the literature. Recall that the only major difference to captured agent trust is that an undercover agent has to operate continuously in a hostile environment.

As for risk analysis, defrauding large organizations, such as banks, may be worthwhile even at a very high price. Hence, systems should be designed such that breaking a certain number of mobile user devices does not break the system completely. This is a particular danger with off-line payment systems. Systems based on so-called digital coins signed individually by the bank avert this danger as far as possible.[11]

6 Conclusion

It is a great opportunity in the field of security applications that a large variety of mobile user devices are currently entering the market for other reasons. These devices offer direct user interfaces, which are an important prerequisite for security, and better performance and versatility than currently proposed specialized security devices. We surveyed how such devices can be made trustworthy, striving for completeness at least with regard to the problems to be addressed. For this, we followed the course of the life of such devices, from their design and production to shipment, personalization, normal operation, the possibility of their falling into the wrong hands, their ability to notice this, and the actions they then take. We differentiated between the point of view of the user of a device, using the notions of personal agent trust and captured agent trust, and the point of view of other parties in cases where such parties also have to trust the mobile user device, using the notion of undercover agent trust.

As no entirely tamper-resistant mobile user devices are currently available, it is reasonable to use less secure devices available from many manufacturers and to supplement them with security modules. Thus, for security applications, one should concentrate on developing protocols and hope that the acceptance is best if some people run the application in mobile phones, others in personal digital assistants, children in game boys, and only some in specialized security devices.

Acknowledgments

We are pleased to thank *Joachim Biskup*, *Hannes Federrath*, *Phil Janson*, *Torsten Polle*, *Kai Rannenberg*, *Michael Schneider*, and *Arnd Weber* for helpful discussions and comments. Birgit Pfitzmann was supported by the DFG (German Research Foundation), and Michael Waidner did part of this work at the University of Karlsruhe, Institute for Computer Design and Fault Tolerance.

References

1. R. Rolfe, "Here Comes Electronic Cash," *Credit Card Management Europe*, Jan./Feb. 1994, pp. 16-20.
2. J.-P. Boly, A. Bosselaers, R. Cramer, R. Michelsen, S. Mjølsnes, F. Muller, T. Pedersen, B. Pfitzmann, P. de Rooij, B. Schoenmakers, M. Schunter , L. Vallée, and M. Waidner, "The ESPRIT Project CAFE — High Security Digital Payment Systems," *Proc. ESORICS 94*, LNCS 875, Springer-Verlag, Berlin, 1994, pp. 217-230.
3. M. Naor and A. Shamir, "Visual Cryptography," *Proc. Eurocrypt '94*, Springer-Verlag, Berlin, 1995, pp. 1-12.
4. J. Brown, "Roll up for the flexible transistor," *New Scientist*, Vol. 143, No. 1944, 1994, p. 5.
5. D. W. Davies and W. L. Price, *Security for Computer Networks, An Introduction to Data Security in Teleprocessing and Electronic Funds Transfer*, (2nd ed.) Wiley, New York,

1989.

6. S. Brands and D. Chaum, “Distance-Bounding Protocols,” *Proc. Eurocrypt '93*, LNCS 765, Springer-Verlag, Berlin, 1994, pp. 344-359.
7. D. Kruse, “The new Siemens computer card,” *Proc. Smart Card 2000*, North-Holland, Amsterdam, 1991, pp. 3-7.
8. M. Paterson, “Secure single chip microcomputer manufacture,” *Proc. Smart Card 2000*, North-Holland, Amsterdam, 1991, pp. 29-37.
9. M. Abadi, M. Burrows, C. Kaufman, and B. Lampson, “Authentication and delegation with smart-cards,” *Science of Computer Programming*, Vol. 21, 1993, pp. 93-113.
10. D. Chaum and T. P. Pedersen, “Wallet Databases with Observers,” *Proc. Crypto '92*, LNCS 740, Springer Verlag, Berlin, 1993, pp. 89-105.
11. S. Brands, “Untraceable Off-line Cash in Wallet with Observers,” *Proc. Crypto '93*, LNCS 773, Springer-Verlag, Berlin, 1994, pp. 302-318.