13. Limiting Liability in Electronic Commerce

Digital signatures are supposed to ensure non-repudiation. However, depending on the environment, attacks might obtain signatures in an underhanded way. This implies conflicting interests: The key holder does not want to be held liable for signatures he has not made. The relying party wants to rely on a digital signature and be able to enforce it in a law court. This chapter tries to find an acceptable compromise. It introduces the necessity for limiting liability for digitally signed transactions, presents the Commitment Service and shows how it can be used as Liability-Cover Service to limit the key holder’s overall liability for digitally signed messages. The main principle is a separation between deniable signatures and undeniable commitments.

13.1 Introduction

13.1.1 Necessity to Limit Liability

Digital signatures (Diffie and Hellman 1976; Menezes, van Oorschot, and Vanstone 1997) are currently not established to be used as equivalents to “handwritten” signatures. A uniform legal framework has not yet been built to recognize digital signatures in the same manner as handwritten signatures (compare Chapters 3 and 14). The EU Directive on “Electronic Signatures” (European Union 2000) is a considerable step in that direction, at least for electronic commerce within the European member states and others that will apply the Directive. However, none of the existing approaches is able to solve all problems which could and should be solved. The main reason is that they do not cover the entire system environment at the key holder’s side.

In spite of sufficiently secure existing signing algorithms, the technical possibility of obtaining signatures in an underhanded way, aided by such mechanisms as Trojan horse attacks, cannot be ignored. Such attacks might result in an unpredictably high damage for the key holder, especially if he cannot prove that an attack has happened. Hundreds or thousands of transactions can be made within a short time by an attack.

To protect the key holder as well as the relying party, we recommend to implement digital signature laws based on the emerging EU Directive (see
also Chapter 14) only in combination with Commitment Service developed by SEMPER as presented in this chapter.

**Trojan-Horse Attacks.** Trojan horses would not affect the signing algorithms themselves. They could manipulate the information to be signed without being noticed by the key holder. Even when smartcards are used for storing keys and performing the signing algorithm, passphrases might be captured. Even for very sophisticated secure smartcard solutions, Trojan horse attacks might choose another malicious strategy: they could pop up a window which looks exactly like the window the key holder expects and which contains the information he or she intends to sign, but then inject—unnoticed by the key holder—completely different information which would actually be signed. For software-only equipment, attacks can be performed even more easily by capturing passphrases or even the signing key itself.

Trojan horses, no matter for which configuration or environment they have been designed, might typically be part of a program different from the one to be attacked (e.g., an electronic-commerce program). They might be located on the same machine or network and thus be executed by the machine which runs the program to be attacked. The latter itself might even have been designed absolutely correct, or at least, might have been designed correctly under the assumption that the operation system does exactly what its specification says and nothing else. Most programs today are designed based on this assumption, and especially for software-only solutions, there is no real chance to design a program which is able to prevent other programs from affecting it. This means: A software solution for digital signatures might be completely correct and nevertheless cannot prevent malicious Trojan horse attacks. This situation will persist as long as no really provably secure operating systems exists which can prevent applications from affecting each other. A good protection, in terms of attack detection, can be provided by so-called “secure devices” which are tamper-resistant, store the secret key and perform the signature without leaving the device, and have a display to show the user exactly what he signs.

**Other Attacks.** There are other kinds of attacks which have to be mentioned here, too. For those attacks, as well as for the malicious Trojan horse attacks, we want to reduce the key holder's potential damage. The Commitment Service presented in this chapter will be the vehicle to achieve this.

Let us start with some attacks which can easily be avoided by the user himself, and increase the seriousness along the considered attacks.

- This “attack” is performed by the user himself. It can be seen as any disruptive element which leads to an unintended signature of the user. It might be caused by the user himself.
- The attacker uses the user’s passphrase which he finds written near the user’s computer.
- The attacker continues the user’s session and makes a signed transaction while the user left his computer for a moment.