Security and Trust in the Italian Legal Digital Signature Framework

Stefano Zanero

Dip. di Elettronica e Informazione
Politecnico di Milano
via Ponzio 34/5
20133 Milano, Italy
zanero@elet.polimi.it

Abstract. The early adoption of a national, legal digital signature framework in Italy has brought forth a series of problems and vulnerabilities. In this paper we describe each of them, showing how in each case the issue does not lie in the algorithms and technologies adopted, but either in faulty implementations, bad design choices, or legal and methodological issues. We also show which countermeasures would be appropriate to reduce the risks. We show the reflex of these vulnerabilities on the trust-based framework which gives legal value to digital signatures. We think that this study can help to avoid similar mistakes, now that under EU directives a similar architecture is planned or under development in most EU countries.

1 Introduction

In 1997, the concept of "law-strong" digital signature (we will call it ILDS, Italian Legal Digital Signature, in the rest of this paper, for the sake of clarity and to distinguish it from the more general concept of "digital signatures") was introduced in Italy [1]. The decree was revolutionary for its time, giving to electronically signed documents (prepared with prescribed methods) the same value of signed paper documents, "trusting" them to show the will of the signatory.

Just as it happened with the early introduction of a law on privacy protection (see, for a commentary, [2]), the early adoption of a legal digital signature scheme has brought forth a series of issues and vulnerabilities, which we helped to identify in at least one well-known case.

A popular misconception, during the ensuing debate, was that such vulnerabilities were defects of the digital signature technology itself. In this paper, we describe these issues, and we show that in each case the problem does not lie in the choice of algorithms and technologies, but either in faulty software implementations, in bad design choices, or in legal and methodological issues. We also show that there exist countermeasures which can be adopted, in order to make the process more secure.

The remainder of the paper is organized as follows: in Section 2 we describe the original Italian law on digital signature, and its recent modifications. In

Section 3 we describe four cases of failures and bugs, in particular the bug we discovered, and show how appropriate countermeasures could avoid such problems, or at least diminished their potential impact. Finally, in Section 4 we draw some conclusions about what can be learned from the vulnerabilities of ILDS.

2 The Italian law on digital signatures

In [1] the underlying concept is that if the procedure (*protocol*) used to generate the ILDS signature is secure, the electronic document can be trusted to show the actual will of the signer, giving to the electronic signature the same meaning and force as the traditional handwritten one. Really, the law recognized that a chain of trusted processes could build a trusted proof, such as the handwritten signature traditionally is.

Truly, in the original legal framework the digital signature was valued more than a traditional one: the law declared that it could not be "unrecognized". In the Italian civil framework, a written but unsigned document has proof value only if the author recognizes it, a signed document has value unless the author unrecognizes the signature, while an authenticated signature, which has been stamped by a public notary, cannot be unrecognized. This is because the notary authenticates the signatory securely. The law thus made the digital signatures similar in strength to an authenticated signature, even if the Certification Authority itself is not a public notary and does not follow notary procedures. This is a strange juridical status, see for a discussion [3]. Since the algorithms used to generate the signature are backed up by strong mathematical principles, *a digital signature cannot be claimed to be forged*, unless proof is given that, for instance, the private key was lost before the signature was placed (for a more detailed discussion, see [4]).

Technical regulations were subsequently released to describe a PKI architecture which could enforce such level of trust. The ILDS system, described in [5] is based on standard X.509 certificates, and on the usage of tamper-proof hardware devices "that can be programmed only at the origin" for generating the keys (i.e. requiring the use of smart cards). Thus, the only way for user to claim the loss of his private key is to declare that his smart card has been lost or stolen, in which case the certificate must be immediately revoked.

The original version of the law required ILDS certificates to be created and signed by trusted certification authorities, and the regulatory agency AIPA[1] was entrusted as the keeper of the CA registry. Requirements for these trusted CAs were:

- to be "S.p.A.", which is a particular societary form, and to have a minimum capital of about 6.500.000 EUR

[1] Agenzia per l'Informatica nella Pubblica Amministrazione: Regulatory Agency for IT in the Government, now called CNIPA, Consiglio Nazionale per l'Informatica nella Pubblica Amministrazione, National Council for IT in the Government

- particular requirements of "honorability" for their administrators (e.g. they must never have been filed for bankruptcy, have been forbidden from taking public offices as a result of a penal process, and so on)
- their technicians must show due diligence and competence to meet with technical regulations
- their IT processes must respect international standards for quality assurance

A new regulation [6], introduced to incorporate the recent EU directive on digital signatures [7], has introduced different "levels of trust" for different types of electronic signatures. It is beyond the scope of this article to fully discuss the official EU terminology and its differences with the Italian one, but the definitions can be summed up as follows:

Electronic Signature: a set of data in electronic format, which is attached to or logically linked with some other electronic data and which is used for the signatory's authentication

Advanced Electronic Signature: an electronic signature which is unambiguously linked with the signatory, can identify the signatory, is created with a method or device which can be controlled solely by the signatory, and is attached to the signed data so that any changes in the data can be noticed

Qualified Electronic Signature: an advanced signature based on a qualified certificate and made with a secure signature creation device; this is the type of signature which has similar value to traditional handwritten signatures

Qualified certificate: a certificate with particular requirements of security

Digital Signature: a qualified electronic signature created with asymmetric cryptography algorithms; this definition has been added in order to make the new law backward-compatible with the old ILDS definition

Entering the public registry of CAs which are certified to create qualified certificates is now called "accreditation". In this paper we will be mainly talking about the requirements for "digital signatures", because part of it was developed under the old ILDS framework. We refer the reader to [8, 9] for a more complete discussion on the legislative evolution of the italian law on digital and electronic signatures, and to [10] for a commentary on international trends in law.

An important point which follows from the history of the development of the ILDS scheme is that each authorized CA implemented its own application for the creation and the verification of digital signatures. This has created a number of problems for interoperability, in addition to the vulnerabilities we address in this article.

There is a wide debate on the validity of the different types of signature. For our discussion, it suffices to say that a "digital signature" can be used to sign contracts or documents destined to government agencies and branches, with full effect. It still cannot be used, for instance, in order to buy a house (since the Italian law requires the presence of buyer and seller in front of a public notary) or to sign a petition for a referendum consultation (for this will require you to be recognized by a public officer). In either case, you could use the ILDS certificate,

providing that the notary or the officer have adequate means to let you sign the documents with it.

There are also some fields where a scheme of legal digital signature is probably not worth the effort: for example, small business to consumer e-commerce transactions have been done and will be done without using strong digital signatures [11].

3 The failures of the Digital Signature

3.1 Word macros and fields

The first serious vulnerability we found in a ILDS application deals with the use of Word macro scripting and dynamic fields.

The vulnerability consists of a flaw in the application design, which we originally discovered in the digital signature software DiKe, developed by InfoCamere (as we said earlier, each CA implemented and released its own toolkit for generating signatures). A quick survey showed afterwards that most of the applications from other CAs were similarly affected.

The principle of the vulnerability is simple. If a Microsoft Office document containing a dynamic field (e.g. a self-updating time and date field) is signed by the means of DiKe, and then verified at a later time, the application shows it in an integrated viewer along with the updated field, without either detecting the variation or alerting the user that a dynamic field is present. This class of vulnerabilities was described also in [12].

This can end up in rather anomalous results, as can be seen in Figure 1. We can see multiple repetition of the documents' date and time. The first is a dynamic field, which is displayed differently each time the document is opened by DiKe, without warning the user in any way. If opened in Word, the changing field would at least show up.

The vendors, alerted by us with a short vulnerability advisory [13], tried to minimize, pointing out that:

- the defect had little impact because another regulation [14] required government agencies and offices to use document formats which cannot contain macros or executable code
- DiKe, like any other ILDS software, digitally signs (and verifies) not the textual content of a document, but an hash of the file containing the document. The execution of the macro does not alter the file contents, but just its representation, thus DiKe correctly reports that the document integrity has not been broken
- Office macros cannot be deactivated from the document viewer APIs used by third party developers. Microsoft, after our advisory and the discussion which ensued, acknowledged this design flaw (which affects all the versions of Office) and released on the Italian market an add-in to deal with it [15].

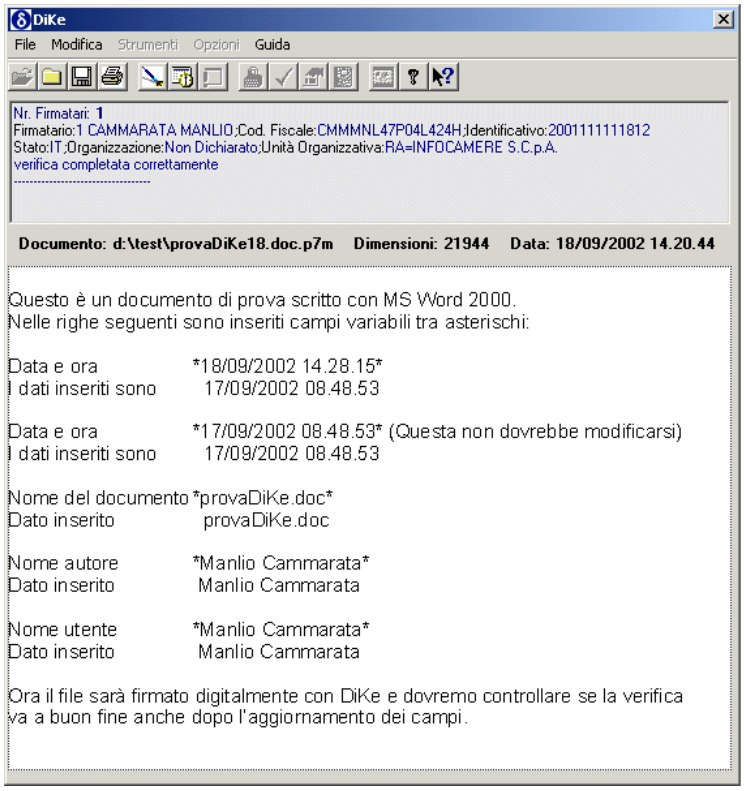

Fig. 1. The behavior of dynamic fields in DiKe. DiKe displays the field on the fourth line, which is a changing date, without alerting the user in any way

In our opinion, this does not reduce the impact of the problem. There is no doubt that DiKe implements correctly the cryptographic algorithms: the sequence of bits of the file containing the document does not change, and thus the algorithm cannot detect any change. But still, the result is not what the end user - or the law - would expect.

One possible solution to this particular issue is to disqualify files that contain macros. This is the simplest solution, and following our advisory (and, according to [16], because of it), it was adopted as part of the new regulations on digital documents and signatures [17], at article 3. Giving a handle to turn off the updating of fields, as Microsoft did, is helpful in this particular case, but does not solve a more general problem.

In Italian (and particularly in law terminology), the word “document” does not have the same meaning as “file”. A legal document is “the representation of acts and facts of judiciary relevance” [18], so an electronic document is not the “file”, but rather the representation of the contents, i.e. what the reader can

see, or even better, what the author actually wanted to show. A digital signature software, in order to be compliant with the spirit and the letter of the Italian law, should then verify that the *representation* is still authentic, not the file itself. In addition, in [5], article 10 states further that the signature applications must represent to the signer "in an unambiguous manner" what he is going to sign. In other words, the signature framework should preserve the proper representation of the will of the signer, building a chain of trusted transformations that grants a correct, non-repudiable and non forgeable end-to-end transmission of this act of will.

A document containing dynamic fields is just one of the examples of a large class of possible problems. For example, another issue could rise from the large amount of metadata that is attached to some file formats, notably Office platform files [19]. Usually, the user is unaware of the metadata contents, and a number of curious cases (such as involuntary disclosure of deleted and corrected parts of documents on press releases) happened because of this. Are these data part of the signed document? If so, how is the signer supposed to know what is embedded into these hidden tags?

A solution could be that the signature application could automatically generate an image or PDF copy of the document, and let the signer sign this copy. This could create a problem of royalties and patents for the application developers, but could be a viable short-term solution. But going in depth, any decoding system used to represent the document to the signer, i.e. any viewer for any file format, should be validated and incorporated into a secure ILDS application. This is evidently impractical.

Choosing a standard format for data, such as XML, combined with "XML Signature" proposed standards [20], could be the ideal solution for this type of problems. Possible problems have also already been identified [21]. We think that this is an interesting topic for further research.

3.2 The PostECom failure

Firma&Cifra is the application released by PostECom, another accredited CA. This application contains a vulnerability [22], reported by an anonymous researcher, which makes it totally insecure. Exactly as in the case of the previous bug, the problem does not reside in the cryptographic algorithms.

The bug leverages the fact that in ILDS signatures, as in most standards, the digital certificate used to generate the signature is appended to the signature itself, using a PKCS#7 envelope [23]. The certificate should be verified first (using the trusted certificates of the authorized CNIPA CAs already stored), and then added to a cache list of verified certificates. If a root certificate is inserted in the PKCS#7 envelope, it should be discarded, because the storage of Firma&Cifra is pre-loaded with the approved root certificates of the accredited CAs, or at least the user should be warned before trusting it. Firma&Cifra instead does not discard it, but imports it automatically to the certificate storage area, and then uses it to verify other certificates.

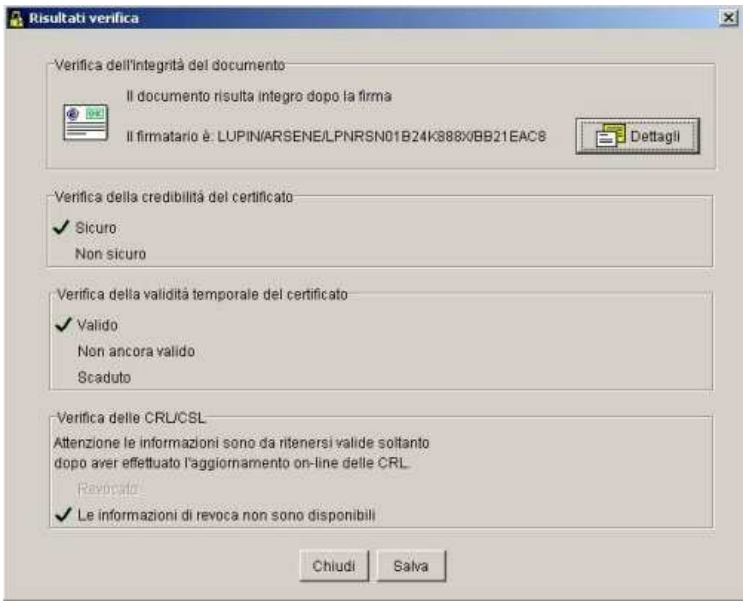

Fig. 2. The certificate of Arsène Lupin in Firma&Cifra. Please note that there is no hint that it has not been signed by a recognized root CA.

This incredible error leads to astonishing results, such as those described in Figure 2: here we see that the software has gladly accepted as authentic the certificate in name of "Arsène Lupin", which is obviously fake.

We must note that also in this case the response from the software vendor has been less than prompt and vaguely worrying, describing this as a feature, and not a bug. This shows a worrying trend in the attention of accredited CAs about the security of their own products. Fortunately, after a while, an update correcting this bug was released.

3.3 Procedural issues with token distribution

A number of procedural issues have also come up. As it may be guessed, one of the most important and delicate points in the certificate roll-out procedure is the proper identification of the subject on whose name a digital certificate is going to be created, and how the smart card with the private key is handed over to this person. This is the fundamental building block of the trust preservation chain which the ILDS framework strives to build.

In many documented cases which were reported to us, a CA (whose name will be omitted) created a number of certificates for the clients of various professional accountants, which were given (along with their PIN) to the accountants themselves, instead of the clients. The accountants were in most cases blissfully

unaware of the full possibilities of these smart cards, which they used only to sign and deposit balances and filings on behalf of the customers. In most cases they also kept all the smartcards, with their PIN sticked on the top, in a nice binder on an office shelf.

This kind of problems is not unheard-of in commercial, private CAs. However, when the digital signature becomes, by force of law, totally equivalent to an handwritten signature, the procedure must be considerably strengthened. The PIN and the smart card should be given only to the person on whose name the certificate has been created, verifying that he or she actually knows the full extent of the law concerning its use.

Since the Italian authorities now require some filings to be done only in electronic form, with the use of ILDS, and since in Italy almost always filings are done by accountants on behalf of their customers, we also feel that proper education on the real value of the smart card and the PIN would be necessary, in order to avoid that, after receiving their devices, less-than-knowledgeable people will give them to a potentially untrusted third party.

A possible solution, requiring technical and legal modifications, would be to create *limited trust* digital signature certificates, with predefined usage limits (e.g. "This certificate can be used to sign only financial filings"), which can then be delegated to third parties according to their responsibilities.

While the law does not explicitly allow this (since the electronic document, signed with a qualified digital signature, is substantially equivalent to a written and undersigned paper document), a regulation (in [18], article 27-bis, third paragraph) says that "The qualified certificate may contain, on request of the owner... [eventual] delegations of powers; b) usage restrictions for certificate, as defined by article 28-bis, paragraph 3; c) limitations on the values of acts and contracts for which the certificate can be used [...]". Article 28-bis, paragraph 3, introduced in D.Lgs. 10/02, explicitly states that "The C.A. can indicate in a qualified certificate usage limitations for said certificate, or a limitation on the value of the contracts the certificate can be used for", provided that third parties can check these limitations. But note that neither of the texts explicitly states that, if said limits are exceeded, the signed document is invalid. Article 43 of the cited [17] states more strongly that the C.A. "must, on request, insert into the certificate any usage limitations".

3.4 Insecurity of the host system

In [24] a further attack on digital signature systems is presented. It basically uses a vulnerability in the Java class loader to trojanize the digital signature application on the host PC, in order to sign documents without the user's acknowledgment. The work underlines how to exploit a shortcoming of the JVM to make an user unwittingly sign a document, but from a security point of view the issue is far more general, and quite well known: an insecure system (one on which a trojan can be present, as it is assumed in this work) cannot generally be used for generating trusted, secure digital signatures, and this was already

demonstrated a number of times (e.g. in [25]). The vulnerability in the class loading and certification scheme is one of the many possible attack paths that are opened if the basic assumption that the host system is secure fails. This discovery once again received great attention from the italian press as the "first practical realization worldwide of an attack against the digital signature devices", which is evidently not the case.

These demonstrations, however, mark a point: how can a digital signature be trusted to the full extent required by ILDS if it has been generated on a computer whose security is not granted? And if digital signatures are to be used by average computer users, how can we ensure or assume the security of their host systems?

If this assumption does not hold any more, we need to completely rethink the transmission paths between applications and signature devices, and we need to insert somehow "trusted checkpoints" on which the user can check what he is really going to sign. For instance, a new card reader device could display on a small embedded screen the fingerprint of the document that is being signed, and ask for further confirmation by the user (for instance by pressing a button on the device). In the same line of thought, devices that require the user to enter the PIN directly on a number keyboard attached to the smart card reader could reduce the risk of exposure to trojanized applications and similars. A more radical approach is the proposal of a dedicated, trusted device, similar to a small PDA, for signature purposes [26].

All these ideas, however, are not very practical, since standard readers already show interoperability problems, and such extensions could magnify them unbearably. In addition, creating nonstandard devices would significantly heighten the costs. However, no solutions based simply on software can prevent a trojan from capturing and replaying the PIN or altering the data flow.

A viable solution is proposed in [27]. The authors rely on the presence of a Trusted Platform Module, as proposed in the TCPA alliance Trusted Computing Platform specifications, and on the Intelligent Adjunct solution proposed in [28].

4 Conclusions

In this paper we have briefly presented two technological issues with two different ILDS applications, and we have shown that in each case the issue does not lie in the cryptographic algorithms: in one case, the abnormal behavior is a matter of a bad design choice, while in the other case the culprit is a faulty implementation of the certificate checking process. We have also shown that certifying the representation of a document, as opposed to the file containing the document, is not a trivial problem, and more research is required in order to properly solve it.

In addition, we have reported an example of methodological issue dealing with certificate distribution and user education. We have also generalized an attack recently reported on a particular architecture as being one facet of the many, well-known problems in the generation of trusted signatures on an un-

trusted machine. In these two cases, the cryptographic algorithms are not even challenged: they are completely bypassed by other issues. Solutions exist, but they have not been applied in the commercially developed signing devices for the ILDS.

In conclusion, this case study shows once more that the sound security of the cryptographic algorithms is just one of the issues to be solved in order to properly implement a Public Key Infrastructure, or indeed any secure system. As an old maxim of cryptographers has it, "If you think cryptography can solve your problem, then you don't understand your problem and you don't understand cryptography"[29].

The Italian law on digital signature focused on the robustness of the algorithms as a sufficient proof of trust and security of the ILDS. However, as we have shown, mathematical proofs of correctness and security do not always translate seamlessly to the real world, where the design, the implementation, and above all people behavior constitute the true, weak component of any security architecture.

5 Acknowledgments

This work was partially supported by the Italian FIRB-Perf project. We also thank dr. Pierluigi Perri of the University of Milano and prof. Andrea Monti of the University of Chieti for their helpful comments on legal issues. The images are a courtesy of the InterLex archive [30, 31].

References

1. D.P.R. 10-11-1997, n. 513, "Regolamento contenente i criteri e le modalitá per la formazione, l'archiviazione e la trasmissione di documenti con strumenti informatici e telematici a norma dell'articolo 15, comma 2, della legge 15 marzo 1997, n. 59". Gazzetta Ufficiale n. 60 (13 mar.), in Italian (1998)
2. Perri, P., Zanero, S.: Lessons learned from the italian law on privacy. Computer Law and Security Report **20** (2004)
3. Monti, A.: Il documento informatico nei rapporti di diritto privato. InterLex website, in Italian (1997)
4. Borruso, R., Buonomo, G., Corasaniti, G., D'Aietti, G.: Profili penali dell'informatica. Giuffré (1994)
5. D.P.C.M. 08-02-1999, "Regole tecniche per la formazione, la trasmissione, la conservazione, la duplicazione, la riproduzione e la validazione, anche temporale, dei documenti informatici". Gazzetta Ufficiale n. 87 (15 apr.), in Italian (1999)
6. D.P.R. 07/04/2003, n. 137, "Regolamento recante disposizioni di coordinamento in materia di firme elettroniche a norma dell'articolo 13 del decreto legislativo 23 gennaio 2002, n. 10". in Italian (2003)
7. 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 L013 (19 jan.) (2000)
8. Cammarata, M., Maccarone, E.: La firma digitale sicura. Il documento informatico nell'ordinamento italiano. Giuffré, Milan (2003)

9. Dumortier, J.: Legal status of qualified electronic signatures in europe. In Paulus, S., Pohlmann, N., Reimer, H., eds.: ISSE 2004-Securing Electronic Business Processes, Vieweg (2004) 281–289
10. Brazell, L.: Electronic signatures: law and regulation. Sweet & Maxwell, London (2004)
11. Winn, J.K.: The emperor's new clothes: The shocking truth about digital signatures and internet commerce. Idaho Law Review Symposium on Uniform Electronic Transaction Act (2001)
12. Kain, K., Smith, S., Asokan, R.: Digital signatures and electronic documents: A cautionary tale. In: Advanced Communications and Multimedia Security, IFIP TC6/TC11 6th Joint Working Conference on Communications and Multimedia Security. Volume 228 of IFIP Conference Proceedings., Kluwer Academic (2002) 293–308
13. Zanero, S.: Sconfinati campi di cavoli amari. Vulnerability Advisory, in Italian (2002)
14. Autoritá per l'informatica nella pubblica amministrazione: Deliberazione n. 51/2000, "regole tecniche in materia di formazione e conservazione di documenti informatici delle pubbliche amministrazioni ai sensi dellart. 18, comma 3, del decreto del presidente della repubblica 10 novembre 1997, n. 513". In Italian (2000)
15. Firma digitale sicura in Microsoft Word. Press Release, in Italian (2003)
16. Cammarata, M.: Regole tecniche per bachi legali. InterLex website, in Italian (2003)
17. D.P.C.M. 13 gennaio 2004, "Regole tecniche per la formazione, la trasmissione, la conservazione, la duplicazione, la riproduzione e la validazione, anche temporale, dei documenti informatici". Gazzetta Ufficiale n. 98 (27 apr.), in Italian (2004)
18. D.P.R. 28-12-2000, n. 445, "Testo unico delle disposizioni legislative e regolamentari in materia di documentazione amministrativa". Gazzetta Ufficiale n. 42 (20 feb), in Italian (2001)
19. How to minimize metadata in Word 2003. Microsoft Knowledge Base (2004)
20. XML signature requirements. Request For Comments 2807 (2000)
21. Jsang, A., Povey, D., Ho, A.: What you see is not always what you sign. In: The proceedings of the Australian UNIX User Group. (2002)
22. Anonymous: Security Advisory, in Italian (2003)
23. Pkcs #7: RSA cryptographic message syntax standard. RSA Laboratories (1993) version 1.5.
24. Bruschi, D., Fabris, D., Glave, V., Rosti, E.: How to unwittingly sign non-repudiable documents with Java applications. In: 9th Annual Computer Security Applications Conference. (2003)
25. Spalka, A., Cremers, A.B., Langweg, H.: The fairy tale of what you see is what you sign: Trojan horse attacks on software for digital signature. In: Proceedings of the IFIPWG9.6/11.7 Working Conference, Security and Control of IT in Society-II (SCITS-II). (2001)
26. Weber, A.: See what you sign: Secure implementations of digital signatures. In: Proceedings of the 5th International Conference on Intelligence and Services in Networks, LNCS 1430, Springer-Verlag (1998) 509–520
27. Spalka, A., Cremers, A.B., Langweg, H.: Protecting the creation of digital signatures with trusted computing platform technology against attacks by trojan horse programs. In: Proceedings of the 16th International Conference on Information Security: Trusted Information. (2001) 403–419

28. Balacheff, B., Chan, D., Chen, L., Pearson, S., Proudler, G.: Securing intelligent adjuncts using trusted computing platform technology. In: Proceedings of the 4th Working Conference on Smart Card Research and Advanced Applications, Kluwer Academic Publishers (2001) 177–195
29. Schneier, B.: A hacker looks at cryptography. In: Black Hat Conference. (1999)
30. Gelpi, A.: La firma è sicura, il documento no. InterLex website, in Italian (2002)
31. Cammarata, M.: Il certificato di Arsène Lupin. InterLex website, in Italian (2003)