J. Lopez, et al., "Designing Software Tools for the Use of Secure Electronic Forms", *3rd ACIS Int. Conf. on Software Engineering, Artificial Intelligence Networking and Parallel/Distributed Computing (SNPD02)*, pp. 157-163, 2002.
NICS Lab. Publications: https://www.nics.uma.es/publications

Designing Software Tools for the Use of Secure Electronic Forms*

Javier López, Antonio Maña, José A. Montenegro, Juan J. Ortega, José M. Troya

Computer Science Department, E.T.S.Ingeniería Infromática,
University of Málaga (Spain)

{jlm, amg, monte, junajose, troya} @ lcc.uma.es

Abstract

Interaction of organizations and their clients by using the Internet can produce meaningful benefits in the accessibility, efficiency and availability of documents, regardless of time and location. However, some types of problems hinder a higher degree of communication. This paper presents some of the results of a Research Project that focuses on the influence of typical open networks risks in electronic interactions and on the need of creating software tools to manage electronic versions of the paper-based forms, as this is the traditional way of interaction through the Web.

1. Introduction

Actual growth of the Internet and the World Wide Web creates the adequate environment for the development of a multiplicity of new services in different scenarios. However, there are still some technological and legal difficulties that avoid a higher rate of communication among organizations.

Inside the scope of a research project we have studied what we consider two of the hardest technological problems. Firstly, it becomes evident that any type of digital transaction is influenced by typical open networks risks. Agents involved need to work in a trustful environment, which must satisfy the required security levels. Thus, privacy and authentication of digital information will be guaranteed to senders and receivers [1]. Secondly, there is a lack of software tools that help to create, distribute and manage in an easy and flexible way the electronic versions of paper-based forms, which is the usual way of interaction through the Web. Clearly, these tools must incorporate authenticity and integrity mechanisms that mimic those ones existing in the traditional paper-based documents [2].

Considering these two problems we present in this paper some of the results of a research project, and show how the integration of the approaches produce a solution that enhances many of actual developments. These results are presented in the following way:

(i) The design and development of a (real) hierarchical *Public Key Infrastructure* (PKI), which we consider the most convenient type of infrastructure for the operation of administrative procedures that involve digital signatures;

(ii) The design of a language for the description of *electronic forms*, as well as the development of a set of tools that allows the use of secure and digitally signed forms. These tools are integrated with the PKI of point (i).

The two problems addressed are related. It is convenient to produce electronic documents and, particularly, electronic forms, that allow the integration of security properties into Internet applications. However, it is has not been the intention of the research project to provide a global solution for this type of applications. We know there is a need for a much broader solution that fulfills other technical requirements that were not considered in our research. For this reason, the schemes designed and developed in our work must be considered as modules that can be adapted and integrated into other broader solutions.

The structure of the rest of the paper is as follows: Section 2 shows the main design

* This work has been partially funded Project FEDER 1FD97-0850

features of the PKI that has been developed in the research project as the basis of digital signatures procedures and that fulfill the requirements of hierarchical organizations. Section 3 outlines the new language, *Form Description Language* (FDL), which has been entirely designed to describe electronic forms and to provide authenticity and integrity whenever they are transmitted. Also, this section describes a tool developed for management of electronic forms, and finally, Section 4 present the conclusions.

2. Design and Development of a Public Key Infrastructure

It is well known that digital signatures schemes are based on the use of public-key cryptosystems [3]. The reasons are that these schemes offer the same functionality than hand-written signatures, and also a high protection against fraud. However, the global use of any of those cryptosystems needs a reliable and efficient mean to manage and distribute public keys, by using digital certificates. Such functionality is provided by a PKI, which is formed by a diversity of Certification Authorities. A PKI becomes essential because, without its use, public key cryptography is marginally more useful than traditional symmetric one [4].

Several PKI models have been considered and studied in our project [5]. The main goal of this study has been to find the model that fits better to the needs of big organizations. We have not only studied theoretic PKI models, but also have tested the certification capabilities provided by commercial electronic mail programs (e-mail agents), which use *PKCS7* certification standard [6,7].

Although addition of certification capabilities in commercial electronic mail programs is a very helpful feature, a detailed analysis shows that these schemes result not satisfactory for e-government applications. Some design features that may compromise the security of the systems have been detected. We summarize some of the most important ones:

- The private key of every user is stored in a local database of the e-mail agent. Therefore, private keys belonging to different users are not completely “isolated” when they share the same computer system (which is common in many organizations). This drawback does not allow the appropriate use of a very important security service, the non-repudiation service [8].
- The certificates needed for a verification of documents that have been digitally signed must be obtained by users from sources that are external to the electronic mail programs. Therefore, it is very possible that users do not verify them properly (as they are not forced to do it). Moreover, use of *Certificate Revocation Lists* (CRLs) is constrained.
- Certification Authorities operate as “islands of trust”, and not as real organized certification structures (what can be easily checked in the configuration of the commercial applications). Because of the hierarchical structure of many organizations, a hierarchical PKI is needed in order to accomplish with full certification needs of communities with thousands of users.

These considerations has taken us to develop our own PKI, that has the following features:

- Adapted to the multi-hierarchical Internet structure because this is the operational environment.

- Provides secure means to identify users and distribute their public keys.

- Uses an architecture of CAs that satisfies the needs of near-certification so the trust can be based on whatever criteria is used in real life.

-Eliminates problems of revocation procedures, particularly those associated with the use of Certificate Revocation Lists.

The main element in the hierarchy is the *Keys Service Unit* (KSU), which integrates certification and management functions. We use a scheme with various KSUs operating over disjoint groups of users, conforming a predefined hierarchy.

KSU hierarchy is parallel to the hierarchy of Internet domains. KSUs are associated to the corresponding e-mail offices. Every KSU is managed by a CA (Figure 1).

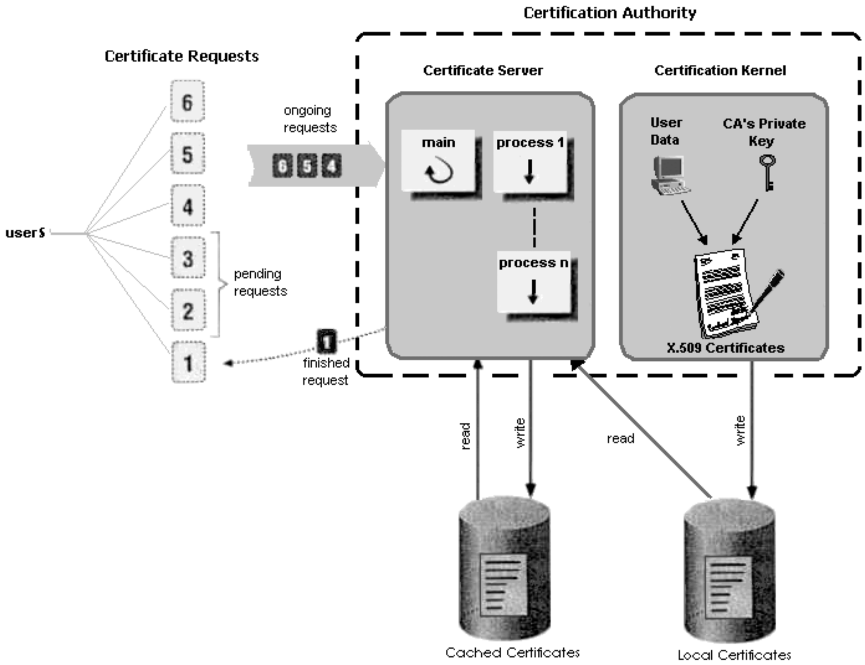


Figure 1. Composition of a Keys Service Unit

Additionally, and it can be seen in the figure, it contains a portion of the certificates database to store the certified keys of its users. Another component is the *key server*, which receives requests and delivers the certificates. The key server manages a certificate proxy that keeps some of the recently received external certificates. The certified keys are managed solely by the corresponding CA; therefore, key updating and revocation are local operations that do not affect the rest of the system.

3. Design of a Form Description Language and Development of Tools

The use of highly structured forms has been the traditional method of interaction through the Internet [9]. Moreover, the use of hand-written signatures in this type of documents has provided the necessary legal bindings for most of scenarios. Our previous study of common applications has showed us that if paper-based forms have to be substituted by electronic forms, then these ones must have the following characteristics: integrity (non-modification by external entities), non-repudiation (agreements must be undeniable), and auditability.

Additionally, global management of electronic forms must include the following features:

- every form must be autonomous from the rest;

- distributed access to unfilled forms, new forms releases, etc., must be provided;

- forms structure must be easy to amend;

- integration of digital signatures capabilities inside forms is mandatory;

- possibility of management of structured and unstructured forms;

- abstraction of forms as extensible objects that encapsulate information.

By taking these features as a starting point, we have tried to design an appropriate language for the description of forms. We have studied XML [10, 11]. This is not a language, but a meta-language that defines a set of rules to create languages. XML does not define specific labels, they are created and adapted for applications. In fact, there are several attempts to standardize different sets of labels for different areas such as education, transport, libraries, etc. However, the final solutions are not clear yet. We also have realized that HTML standard is being worsening as each browser includes its own labels to cover user needs [12]. HTML needs to be updated, and this is the reason why next version of HTML will match XML. These reasons recommended us to try to design and develop our own language. Its name is FDL (Formal Description Language), and it is based on XML. To be more precise, it is based on XFDL [13].

The use of our own specific language, with its own tools, and completely adapted to XML, introduces many advantages in comparison with traditional use of HTML. Figure 2 shows the UML diagram for the basic structure of an FDL file.

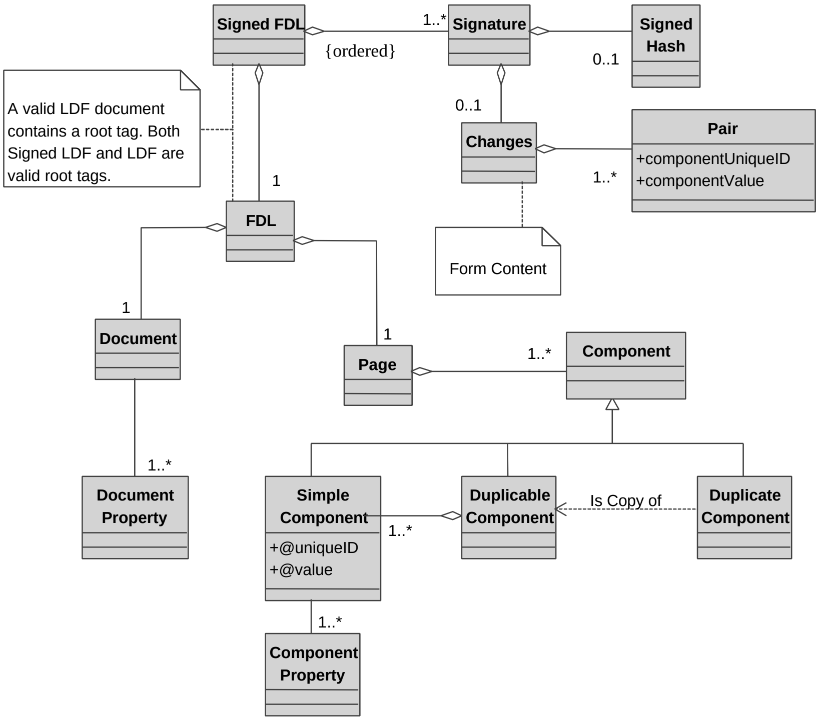


Figure 2. Basic structure of an FDL file

The most important advantages are briefly summarized:

- Regarding forms status:
  - It is easy to add new components not included in HTML. These new components are very useful to avoid invalid user inputs into the forms; therefore, a more dynamic management is done.
  - Automatic data validation is done without programming specific code for that operation. The reason is that the specification of the form itself includes the valid ranges.
  - Definition of the structure of the fields where signatures are contained simplify the (automatic) process of signature verification.
  - The particular design of our language, together with the standards where it relies on, facilitate creation of parsers that automatically translate forms to any other language (i.e, XML)
  - Development of specific mechanisms allow duplication of fields, application of mathematic expressions, and the inclusion/exclusion of components into the form, in real-time execution. This feature provides an advantage when compared to HTML, which must make use of JavaScript to get the same functionality.
- Regarding forms management:
  - We include the possibility of forms visualization by using a traditional browser (for on-line operations). Also, an independent application (for any off-line process, like electronic mail) can be used.

- Management of signed forms is easier. The signer can store in his/her own hard disk a copy of a partially filled document. This can be later opened using a browser in order to fill it completely.

- One or several persons can sign forms, and these ones can be encrypted using unconstrained implementations of algorithms.

- Oppositely to HTML, there are some extensions defined to distinguish different parts and formats in the same document. For this reason, data and signatures are integrated into the form.

- The position of components in the screen is very precise, and is adapted to official organizations requirements.

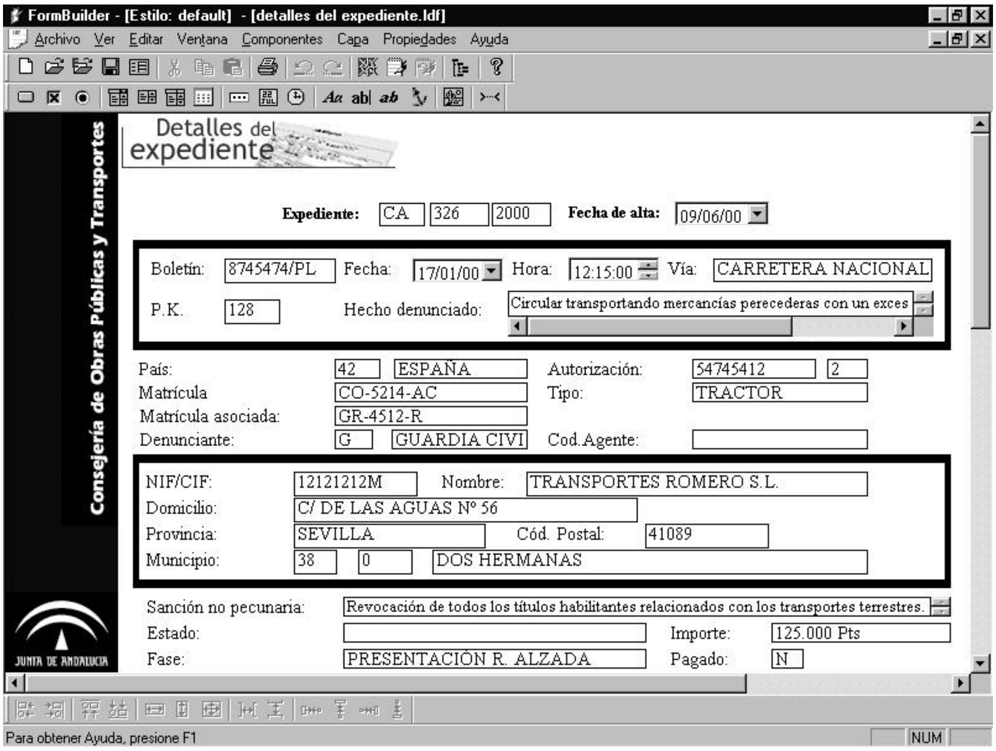
- Oppositely to HTML, the status of the form is preserved; thus, it is not necessary to manage forms in a single and large page, or to store them and pass the status from page to page. It is not even necessary to use cookies to store personal and private data. FDL has been designed to organize any form in several pages while having data in memory continuously.

- Regarding communication:

    - A proprietary format facilitates that the context of the signature is not lost, so ambiguity of data in the form never occurs. Moreover, the document is audited (persons involved, date of the agreement, etc.) on its own.

    - Oppositely to HTML, FDL provides a data structure, and separates application, presentation and logic levels.

These capabilities have been integrated into Netscape browser, adding it a plug-in that captures the electronic form and interpret it as a MIME type. At the same time, the PKI that we have developed has been integrated into Netscape security module, substituting the operation of those CAs that are included in the original product. Therefore, every digital signature operation is based on the use of the hierarchy of CAs inside the PKI. At the same time, verification of signed forms is easily done regardless the number of users in the community. It is scheduled to include all those features in other commercial browsers. More precisely, and as a first step, into Microsoft Explorer.


FormBuilder - [Estilo: default] - [detalles del expediente.ldf]
Archivo Ver Editar Ventana Componentes Capa Propiedades Ayuda
Aa abl ab
Consejería de Obras Públicas y Transportes
JUNTA DE ANDALUCIA
Para obtener Ayuda, presione F1
NUM

Detalles del expediente

Expediente: CA 326 2000 Fecha de alta: 09/06/00

Boletín: 8745474/PL Fecha: 17/01/00 Hora: 12:15:00 Vía: CARRETERA NACIONAL

P.K. 128 Hecho denunciado: Circular transportando mercancías perecederas con un exces

País: 42 ESPAÑA Autorización: 54745412 2

Matrícula CO-5214-AC Tipo: TRACTOR

Matrícula asociada: GR-4512-R

Denunciante: G GUARDIA CIVI Cod.Agente:

NIF/CIF: 12121212M Nombre: TRANSPORTES ROMERO S.L.

Domicilio: C/ DE LAS AGUAS Nº 56

Provincia: SEVILLA Cód. Postal: 41089

Municipio: 38 0 DOS HERMANAS

Sanción no pecunaria: Revocación de todos los títulos habilitantes relacionados con los transportes terrestres.

Estado: Importe: 125.000 Pts

Fase: PRESENTACIÓN R. ALZADA Pagado: N

Figure 3: Visual Toolkit Form

3.1 Visual Tool for the automatic generation of forms

During last decade markup languages have suffered a great impulse, mainly because of the use of web applications. However, only recently visual tools for the generation and development of advanced graphic applications had appeared in the market.

The reason is that advantages introduced by a markup language tends to decrease the design of the final product. Although they are programmed by using plane code, with no need of compilation, the introduction of numerous features incite to make errors and impedes an easy learning.

Our language introduces many features to enhance graphic design of forms, and many of them may be unknown to users. Therefore, we have developed a WYSIWYG form editor that facilitates the creation of forms and allows the correct and exact introduction of components into the form. FDL learning time is eliminated.

Figure 3 shows the process for the creation of forms, as well as menus and other elements of the tool developed, named *FormBuilder*.

3.2 Tool for the management of electronic forms

Another tool that has been developed during the project is SAFE. This tool facilitates management of forms but also of persons involved in their administration. SAFE is a framework that allows any organization an easy and quick adoption of electronic forms. Figure 4 shows the elements of the framework.

SAFE is mainly formed by three components:

- *Form Server*: This component is divided into two:
  (i) A public part that, through a Web server, provides forms skeleton to users,
  (ii) A private part for management of forms and the verification of their signatures (interacting with the PKI). Management is done by a supervisor, that creates managers during the installation phase, as can be seen in the figure.
- *Common Interface*: This module defines the common interface that must be adopted by the organization to interact with the Form Server.
- *Implementation Interface*: It is the instantiation of the Common Interface. It allows the organization to adapt to the system.

4. Conclusions

The paper presents the results of a research project that studies two joint problems: (a) the need of using security for communications over open networks, and (b) the use of electronic versions of the paper-based forms.

We have presented the main features of the PKI specifically developed and the reasons for its design. Regarding the electronic forms we have designed a language for their description. The most important characteristics of the language and the objectives of its creation have been summarized in order to show why XML or HTML are not appropriate for our work.

Modular design and development of those tools facilitates that the outcome of the work is integrated into broader systems. These new solutions also help in establishing the basis for future design and development of schemes oriented to electronic forms signature in communications among organizations.

References

[1] European Commission, “Directive 1999/93 of the European Parliament and the Council on a Community Framework for Electronic Signatures”, December 1999.

[2] European Commission, “Directive on Certain Legal Aspects of Electronic Commerce in the Internal Market”, February 2000.

[3] W. Diffie, M. Hellman, "New Directions in Cryptography", IEEE Transactions on Information Theory. IT-22, n. 6. 1976, pp. 644-654.

[4] W. Burr, “Public Key Infrastructure Technical Specifications (version 2.3). Part C: Concepts of Operations”, Public Key Infrastructure Working Group, National Institute of Standards and Technology, November 1996.

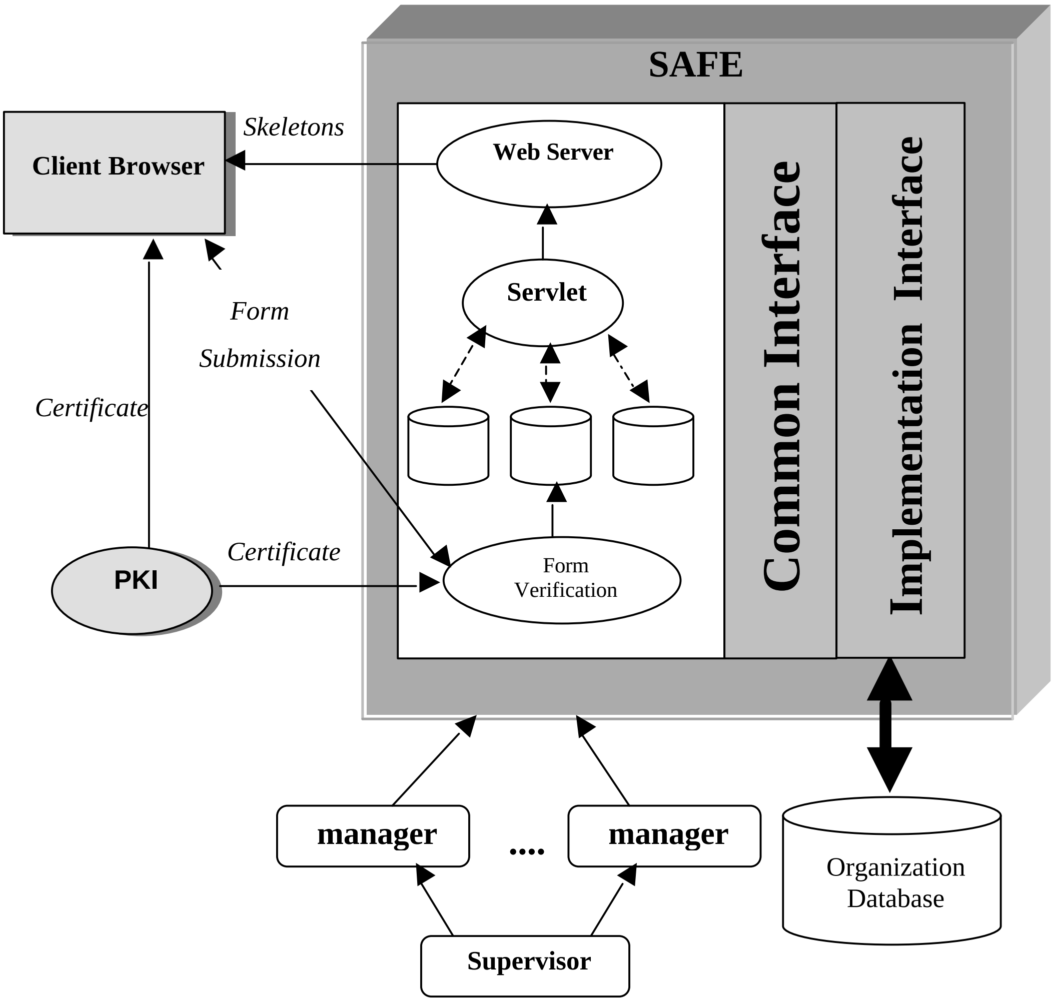


Figure 4: SAFE Global System

[5] W. Ford, M. Baum, "Secure Electronic Commerce", Prentice-Hall, 2000.

[6] “PKCS#7: Cryptographic Message Syntax Standard”, RSA Laboratories Technical Note, November 1993.

[7] B. Kaliski, K. Kingdon, “Extensions and Revisions to PKCS#7”, RSA Laboratories Technical Note, May 1997.

[8] Jianying Zhou, “Non-Repudiation in Electronic Commerce”, Artech House, 2001.

[9] “Improving Electronic Document Management”, Guidelines For Australian Government Agencies, Australian Office of Government Information Technology, 1996.

[10] “Extensible Markup Language (XML)” http://www.w3.org/XML

[11] “Canonical XML, Version 1.0”, W3C Working Draft, September 2000. http://www.w3.org/TR/2000/WD-xml-c14n-20000907

[12] HTML 4.01 Specification, W3C Recommendation, December 1999. http://www.w3.org/TR/html4/

[13] B. Blair, J. Boyer, “XFDL: Creating Electronic Commerce Transaction Records using XML”, Computer Networks, n. 31, pp. 1611-1622, 1999.