

## Usability and document authentication issues

José Kahan (W3C)  
W3C / ERCIM,  
INRIA Rhône-Alpes,  
ZIRST, 655 av. de l'Europe, Montbonnot,  
FR-38334 ST ISMIER CEDEX, France  
`jose.kahan@w3.org`

January 25, 2006

### 1 Introduction

Most of today's support for authentication in browsers has focused on transport security and one-way or mutual online authentication of peers. Transport security concerns the protection of the confidentiality and/or integrity of data during transit between peers. Peer authentication concerns the verification of the identity of one peer by another peer.

Browsers currently support transport security by establishing a secure channel to a server using the TLS/SSL protocol by means of server associated public-key certificates. These certificate also provide for authentication of servers. Authentication of users is done either thru passwords, IP address or indirect single-sign on protocols.

Both transport security and peer authentication are measures that are effective during a session. However, they don't provide any protection for the data itself. In other words, a document may be modified in an unauthorized way before or after it has been transmitted. Transport security and peer authentication don't guarantee the origin and integrity of a document.

Document authentication, provided thru digital signatures, can counter these problems. The properties of a digital signature ensure the integrity of a signed document. Moreover, a digital signature allows to know who has

**position paper for the W3C Workshop on Transparency and Usability for Web Authentication, 15/16 March 2006 - NYC, USA**

---

signed something. If the author of a document signs it, we have proof of the origin of a document. Finally, the properties of a digital signature can persist over time, and not expire once a document has been transmitted.

Recent work in the XForms Working Group and adoption of this recommendation by browsers show that the next generation of browsers will support native generation and processing of XML Signatures (XML-SIG). This will bring forth new authentication and usability problems related to signed documents.

My position is that web authentication should not only concern network security, but also document authentication. Moreover, many of the usability requirements for document authentication will match those for network security. Usability of authentication should take into account this, on one hand, to speed up integration of document authentication, and, on the other hand, to avoid having confusing interfaces later on.

The following sections give a brief introduction to XML-SIG documents, present some use-cases, and describe potential usability issues.

## **2 Document authentication, digital signatures, and XML-SIG**

Document authentication concerns both the verification of the integrity of a document as well as the authentication of the issuer of this document. It is important that any unauthorized change to a document be detected. Likewise, it is important that we are sure who has written a document, to avoid forgeries. Both of these goals can be achieved by digital signatures.

A digital signature is a security mechanism that, thru a cryptographic function, guarantees the integrity of a document as well as the authentication of its source. We'll now describe the use of a asymmetric key (or public-key) based digital signature algorithm.

Typically, a user (Alice) has a cryptographic key pair, a private key, known only to her, and a public-key that can be known by everyone else. When Alice wants to sign a document, she uses an asymmetric signature creation algorithm, giving to it as parameters the document and her private key. The digital signature is the result of this operation. When another user (Bob) wants to verify Alice's signature of the document, he uses an asymmetric signature verification algorithm, feeding to it both the document,

## **position paper for the W3C Workshop on Transparency and Usability for Web Authentication, 15/16 March 2006 - NYC, USA**

---

Alice's signature of it, and Alice's public-key. The signature can only be successfully verified if the document or Alice's signature hasn't been altered, and if the key used to sign it is the complementary one of the one used to verify the signature.

The integrity of a signed document is guaranteed in that any modification of this document will be detected because the verification of Alice's signature will fail. If Alice's public-key is used successfully to verify a signature, the authentication of Alice as the signed is implied. Only Alice knows the related private-key.

XML-SIG is a W3C Recommendation that defines an XML syntax for creating and representing digital signatures. XML-Signatures can be applied to any kind of digital document, including XML documents. For XML documents, the signature can be applied to a whole document or to part of it. There are several tool kits, both free and commercial, that have successfully implemented XML-SIG.

A public-key is often published in the form of a public-key certificate, which binds the key with data about both the user and the key. The certificate itself is signed by a certificate authority. XKMS is a recent W3C Recommendation that allows applications to register public-keys, as well as to find, validate, and parse public-key certificates.

### **3 Some use cases for signed web documents**

Signed documents can offer several advantages on an every-day life use:

- **e-commerce.** When Bob purchases an item thru an on-line vendor, he can request that the vendor signs the order. Bob then has a document that binds the vendor to the order and that he can show to other parties in problems arise. Compare this to today's situation where Bob would receive an email copy of the order, or be proposed to print-out the order, or have the order be stored only on the vendor's site. Following the same train of thought, the vendor could ask that Bob signs the order too, to have a proof of the purchase.

**position paper for the W3C Workshop on Transparency and Usability for Web Authentication, 15/16 March 2006 - NYC, USA**

---

- **Official documents and mirrors.** Data on the web today can be quickly mirrored, thus blurring the actual source of the information. If someone wants to check the truthfulness of a document, one needs to track down the original site and then search for the document there. If documents are signed, a user only needs to find the valid public-key certificate that will allow to verify the signature, and thus check the origin of the document. This is interesting in that it provides for mirroring without distortion of information.
- **Compound documents.** A compound document is one that may combine different external data files, such as CSS style sheets and images. A similar case happens when a document is split over many files, such as an index. A digital signature applied over a compound document allows to bind together the document with its external files.

## 4 Usability-related issues

For simplicity, we limit this discussion to issues related to browsing previously signed documents. Issues related to user signing of a document are left for discussion elsewhere.

When a user browses a signed document, the browser should propose the following steps for each of the signatures found in the document:

- Indicate the user when a browsed document has been signed.
- Show the user the details of the signature: who signed the document and other attributes binded to the signature.
- Prompt the user for action for verifying a signature?
- Indicate whether a signed document has a valid or invalid signature.
- Show the user what is covered by this signature (which elements, whole document, document and other related content such as images and style sheets). Basically, having a common understanding between the signer and the verifier of a signature of what data was actually signed.

Note that many of these issues are similar to those related to user authentication and one may wonder if both could be solved with a same standard interface.

**position paper for the W3C Workshop on Transparency and Usability for Web Authentication, 15/16 March 2006 - NYC, USA**

The last issue is similar to a P3P issue, where a user needs to see which parts of a policy are associated to which parts of a document (for example, see <http://www.privacybird.com/>). This issue is also similar when signing a document on the client-side. The signature verifier has to have the same understanding of what the client has signed.