FACULDADE SENAC DE CIÊNCIAS

EXATAS E TECNOLOGIA

Segurança de Rede e de Sistemas

José Adauto Ribeiro

XML Signature e os Desafios da Utilização de Assinatura Digital

São Paulo
2004

JOSÉ ADAUTO RIBEIRO

XML Signature e os Desafios da Utilização de Assinatura Digital

Trabalho de conclusão do curso “Segurança de Redes e Aplicação”, da Faculdade SENAC de Ciências Exatas e Tecnologia.

Orientador: Prof. Dr. Volnys Borges Bernal

São Paulo
2004

Ribeiro, José Adauto

XML Signature e os Desafios da Utilização de Assinatura Digital / José Adauto Ribeiro – São Paulo, 2004.
65f.

Trabalho de Conclusão de Curso – Faculdade SENAC de Ciências Exatas e Tecnologia

Orientador: Prof. Dr. Volnys Borges Bernal

1. Assinatura digital; 2. XML; 3. Política de assinatura; 4. Estampilha temporal

Aluno: José Adauto Ribeiro
Título: XML Signature e os Desafios da
Utilização de Assinatura Digital

Faculdade SENAC de Ciências Exatas e Tecnologia

A banca examinadora dos Trabalhos de Conclusão de Curso, em sessão pública realizada em dd/mm/2004, considerou o candidato:

( ) aprovado ( ) reprovado

1) Examinador: Luis Gustavo Gasparini Kiatake

2) Examinador: Murilo Rivau Fernandes

3) Presidente: Dr. Adilson Eduardo Guelfi

Agradecimento

Agradeço a paciência e o incentivo da minha esposa, Maria Aparecida Camargo Ribeiro (Mana), e da minha filha, Estefânia, sem o que talvez não tivesse concluído este trabalho.

Agradeço também ao meu orientador, Professor Dr. Volnys Borges Bernal, que também me incentivou e muito me ajudou a vencer este importante estágio.

Agradeço ainda à banca, que muito agregou com suas sugestões e correções.

Resumo

Com a crescente utilização da Internet e o intercâmbio de informações entre empresas, bem como a utilização crescente de documentos eletrônicos, aumentam a importância e a necessidade de utilização da assinatura digital para dar maior segurança e confiabilidade a tais facilidades. Por outro lado, o padrão de formatação de documentos/mensagens em XML teve e ainda tem um crescimento muito grande em vários segmentos, o que levou à definição de padrão para assinatura digital seguindo essa formatação: chamada de “XML Signature”. Independente do formato, no entanto, a assinatura digital, em seu estágio atual, traz ainda algumas deficiências que precisam ser superadas. Neste trabalho é feita uma breve descrição do processo que envolve uma assinatura digital, sua formatação em XML e é apresentada a proposta do ETSI, que poderá vir a ser uma solução para superar essas deficiências.

Palavras-chave: Assinatura digital, XML, política de assinatura, estampilha temporal.

Abstract

With the increasing use of the Internet and the interchange of information among companies, as well as the increasing use of electronic documents, it's growing the importance and the necessity of using digital signature to give greater security and trustworthiness to such facilities. On the other hand, the standard of documents/messages formatting in XML had and still has a very great growth in some segments, that led to the definition of standard for digital signature following this formatting: called "XML Signature". Independent of the format, however, the digital signature, in its current state, still brings some deficiencies that they need to be surpassed. In this work a brief description of the process that involves a signature digital is made, its XML formatting and it is presented the proposal of the ETSI, that may come to be a solution to surpass these deficiencies.

Keywords: digital signature, XML, signature policy, time-stamping.

Sumário

1. INTRODUÇÃO 1

2. ASSINATURA DIGITAL 2

2.1. O que é e como funciona 2

2.2. O que não é garantido 8

2.3. Padrões estabelecidos 9

3. O PADRÃO XML SIGNATURE 10

3.1. Histórico do XML 10

3.2. Estrutura de documento em XML 10

3.3. Vantagens do uso de XML 12

3.4. Algumas especificações relacionadas ao padrão XML 15

3.5. XML Signature 16

4. DESAFIOS PARA UTILIZAÇÃO 30

4.1. Política de assinatura digital 30

4.2. Realização de uma assinatura com data no passado 32

4.3. A longevidade de um documento assinado digitalmente 33

5. A PROPOSTA DO ETSI 36

5.1. Formato XAdES-BES 37

5.2. Formato XAdES-EPES 42

5.3. Formato XAdES-T 45

5.4. Formato XAdES-C 46

6. RESPOSTA AOS DESAFIOS PROPOSTOS 52

6.1. Implementação da especificação XAdES 53

7. CONCLUSÃO 57

8. REFERÊNCIAS 61

GLOSSÁRIO 64

1. INTRODUÇÃO

Este trabalho trata dos aspectos envolvidos na utilização da tecnologia de assinatura digital, em especial à padronização relativamente recente de assinatura digital no formato XML (‘XML Signature”); aborda os maiores desafios envolvidos com a assinatura digital e as iniciativas em andamento para superá-los.

Está dividido da seguinte forma: no capítulo 2 é abordada de forma geral a tecnologia da assinatura digital, como a mesma é gerada e aspectos envolvidos na sua validação.

No capítulo 3 são abordados o padrão XML e o padrão proposto para assinatura digital em XML (XML Signature).

No capítulo 4 são abordados os pontos que constituem os maiores desafios para que a assinatura digital venha a se tornar uma tecnologia mais confiável e que possa vir a substituir a assinatura tradicional (manuscrita), tanto em termos dos vários contextos em que se aplica quanto em termos de longevidade em sua verificação.

No capítulo 5 é abordada a proposta de padrão publicada pelo ETSI (ETSI, 2004), que procura exatamente resolver as questões apontadas no capítulo anterior. Neste trabalho será abordado apenas o formato XML justamente porque se aposta que essa é a tendência na utilização desse serviço.

No capítulo 6 é enfatizada a idéia de que a proposta do ETSI pode ser a solução para os desafios atuais da assinatura digital.

Ao final, é apresentada uma conclusão, que faz uma abordagem rápida sobre a legislação sobre assinatura digital no Brasil.

2. ASSINATURA DIGITAL

2.1. O que é e como funciona

A definição abaixo é dada no “site” do ITI (Instituto Nacional de Tecnologia da Informação), que é a Autoridade Certificadora Raiz - **AC Raiz -** da Infra-Estrutura de Chaves Públicas Brasileira - **ICP-Brasil**.

“A assinatura digital é uma modalidade de assinatura eletrônica, resultado de uma operação matemática que utiliza algoritmos de criptografia assimétrica e permite aferir, com segurança, a origem e a integridade do documento”.

“A assinatura digital fica de tal modo vinculada ao documento eletrônico ‘subscrito’ que, ante a menor alteração neste, a assinatura se torna inválida. A técnica permite não só verificar a autoria do documento, como estabelece também uma ‘imutabilidade lógica’ de seu conteúdo, pois qualquer alteração do documento, como, por exemplo, a inserção de mais um espaço entre duas palavras, invalida a assinatura.”

“Necessário distinguir assinatura digital da assinatura digitalizada. A assinatura digitalizada é a reprodução da assinatura autógrafa como imagem por um equipamento tipo scanner. Ela não garante a autoria e integridade do documento eletrônico, porquanto não existe uma associação inequívoca entre o subscritor e o texto digitalizado, uma vez que ela pode ser facilmente copiada e inserida em outro documento”.

A seguir, conforme colocado de forma bastante didática por Nakov (2002), é descrito como funciona um processo de criação e verificação de uma assinatura digital.

2.1.1. Criando assinaturas digitais

A criptografia de chave pública permite um método confiável de assinatura digital e verificação da assinatura baseada no par de chaves pública e privada. Uma pessoa pode assinar uma dada mensagem digital (arquivo, documento, e-mail e assim por diante) com sua chave privada.

Do ponto de vista técnico, a assinatura digital de uma mensagem é executada em dois passos:

Figura 01 – Criação da assinatura digital
Fonte: adaptado de Nakov, 2002

Passo 1: Cálculo do resumo da mensagem

Neste primeiro passo do processo, um valor de "hash" da mensagem (freqüentemente chamado de "message digest", ou resumo da mensagem) é calculado pela aplicação de um algoritmo criptográfico de hashing (por exemplo, MD2, MD4, MD5, SHA1, ou outro). O valor de hash calculado de uma mensagem é uma seqüência de bits, usualmente com um tamanho fixo, extraído de alguma maneira da mensagem.

Todos algoritmos confiáveis para o cálculo do valor do hash aplicam transformações matemáticas tais que, quando apenas um simples bit da mensagem de entrada é alterado, um valor de hash completamente diferente é obtido. Devido a este comportamento, estes algoritmos são muito estáveis em ataques cripto-analíticos; ou seja, é computacionalmente inviável, a partir de um valor de hash

encontrar a mensagem que originou tal valor. Esta inviabilidade para a recuperação da mensagem de entrada deve-se ao fato que o valor de hash de uma mensagem pode ter o tamanho muitas vezes menor do que a mensagem de entrada. Há consenso atualmente que os recursos computacionais necessários para encontrar uma mensagem, dado seu valor de hash, são tão grandes que inviabilizam a tarefa.

É também importante saber que, teoricamente, é possível para duas mensagens inteiramente diferentes ter o mesmo valor de hash calculado por um algoritmo de hash, mas a probabilidade de isto acontecer é tão pequena que, na prática, é desconsiderada.

Passo 2: Cálculo da assinatura digital

No segundo passo da assinatura digital da mensagem, a informação obtida no primeiro passo, o valor de resumo da mensagem, é cifrada com a chave privada da pessoa que assina a mensagem e assim um valor resumo da mensagem cifrado, também chamado de assinatura digital, é obtido. Para este propósito, um algoritmo criptográfico-matemático de encriptação para o cálculo da assinatura digital de um dado resumo de mensagem é utilizado. Os algoritmos usados mais freqüentemente são RSA (baseado na teoria dos números), DSA (baseado na teoria dos logaritmos discretos) e o ECDSA (baseado na teoria das curvas elípticas). Freqüentemente, a assinatura digital obtida é juntada à mensagem em um formato especial para ser verificada posteriormente se for necessário.

2.1.2. Verificando assinaturas digitais

A tecnologia de assinatura digital permite ao receptor de uma dada mensagem assinada verificar sua origem real e sua integridade. O processo de verificação da assinatura digital é proposto para determinar se uma dada mensagem

foi assinada pela chave privada que corresponde a uma dada chave pública. A verificação da assinatura digital não pode determinar se uma dada mensagem foi assinada por uma determinada entidade. Se há necessidade de verificar se uma entidade assinou uma dada mensagem, precisa-se obter a sua chave pública de uma maneira segura (por exemplo, um disquete ou um CD) ou com a ajuda da Infraestrutura de Chaves Públicas utilizando um certificado digital. Sem ter um modo seguro de obter a chave pública real de uma dada entidade, não há a possibilidade de verificar se uma dada mensagem é realmente assinada pela mesma.

Do ponto de vista técnico, a verificação da assinatura digital é executada em três passos:

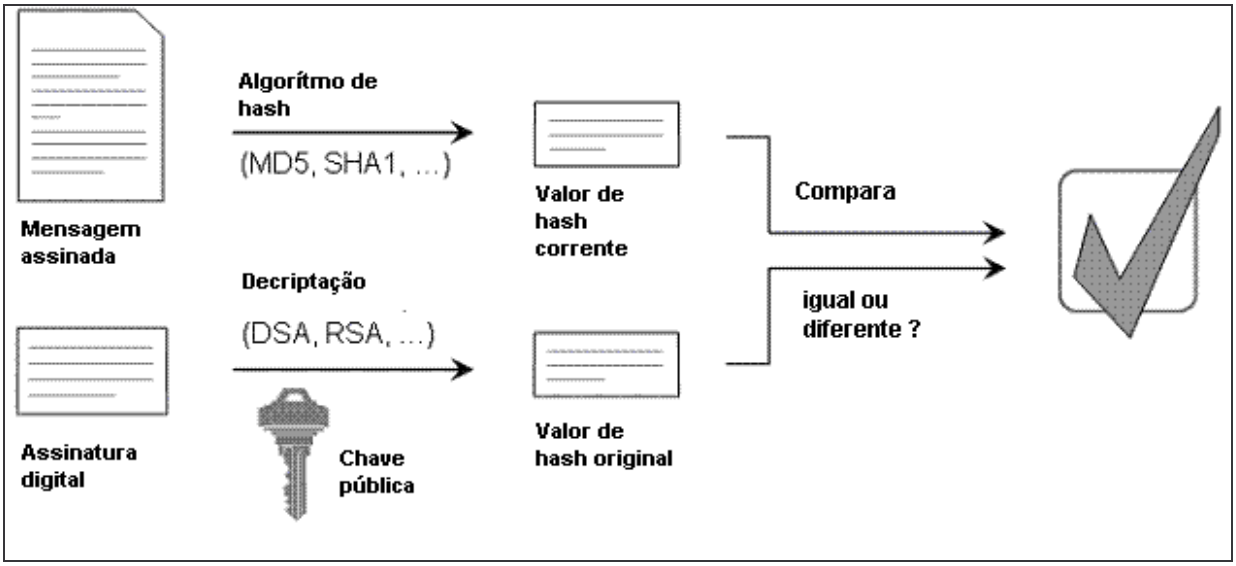

Figura 02 – Verificação da assinatura digital
Fonte: adaptado de Nakov, 2002

Passo 1: Cálculo do valor corrente do hash

No primeiro passo, um valor de hash da mensagem assinada é calculado. Para esse cálculo, o mesmo algoritmo de hash é usado tal como foi usado

durante o processo de geração da assinatura. O valor obtido é chamado de “valor de hash corrente” porque ele é calculado a partir do estado atual da mensagem.

Passo 2: Calcular o valor original do hash

No segundo passo, a assinatura digital é decifrada com o mesmo algoritmo utilizado durante a geração da assinatura. A decifração é feita com a chave pública associada à chave privada utilizada durante a assinatura da mensagem. Como resultado, é obtido o valor original do hash que foi calculado da mensagem original durante o primeiro passo da criação da assinatura digital (o valor original do resumo da mensagem – valor de hash).

Passo 3: Comparar os valores corrente e original do hash

No terceiro passo é comparado o valor corrente do hash obtido no primeiro passo com o valor original do hash obtido no segundo passo. Se os dois valores forem idênticos, a verificação é bem sucedida e prova que a mensagem foi assinada com a chave privada que corresponde à chave pública usada na verificação. Se forem diferentes, isto significa que a assinatura digital é inválida e a verificação falha.

2.1.3. Razões para assinaturas inválidas

No processo criptográfico de verificação, há pelo menos três possíveis razões para resultar em assinatura digital inválida:

- Se a assinatura digital é adulterada (ela não é verdadeira) e é decifrada com a chave pública verdadeira, o valor original obtido não será o valor de “hash” original da mensagem original, mas algum outro valor;

- Se a mensagem foi alterada (adulterada) após a assinatura, o valor de “hash” corrente calculado dessa mensagem adulterada será diferente do valor de “hash” original porque as duas mensagens diferentes correspondem a valores de “hash” diferentes;
- Se a chave pública utilizada não corresponde à chave privada usada para efetuar a assinatura digital, o valor de “hash” obtido por decriptação da assinatura não será igual ao valor de “hash” corrente obtido a partir da mensagem.

Se a verificação falhar, indica que a assinatura que está sendo verificada não foi obtida assinando a mensagem que está sendo verificada com a chave privada que corresponde à chave pública usada para a verificação. A verificação mal sucedida não significa necessariamente que uma tentativa de adulteração da assinatura digital foi detectada. Às vezes, a verificação pode falhar porque uma chave pública inválida é usada. Tal situação poderia ser obtida quando a mensagem não é emitida pela entidade que se esperou emití-la ou quando o sistema de verificação da assinatura tem uma chave pública incorreta para esta entidade. É mesmo possível para uma entidade possuir diversas chaves públicas válidas diferentes junto com certificados válidos para cada uma delas e do sistema ter tentado verificar uma mensagem recebida desta entidade com alguma destas chaves públicas, mas não com a correta (a que corresponde à chave privada usada na criação da mensagem assinada).

Para que tais problemas sejam evitados, o mais usual é que, quando um documento assinado é gerado, o certificado do signatário seja anexado junto a este documento e a assinatura digital correspondente. Assim, durante a verificação, a chave pública contida no certificado recebido é usada para a verificação da

assinatura. Se a verificação for bem sucedida, considera-se que o documento é assinado pela entidade que possui o certificado. Naturalmente, antes do uso de qualquer certificado, é necessário validá-lo.

2.2. O que não é garantido

Conforme colocado na própria RFC-3275 (IETF, 2002), a força de uma particular assinatura depende de todas as ligações na corrente da segurança. Isto inclui os algoritmos de assinatura e de “hashing” utilizados, a força da geração de chave aleatória e o tamanho da chave, a segurança da chave e do mecanismo de autenticação e da distribuição de certificado, da política de validação da cadeia de certificados, da proteção do processo criptográfico contra a observação hostil, falsificação, etc.

Cuidado deve ser tomado por aplicações ao executar os vários algoritmos que podem ser especificados em uma assinatura e em processar todo o conteúdo que possa ser fornecido a tais algoritmos como parâmetros. Os algoritmos especificados serão executados geralmente através de uma biblioteca segura, mas parâmetros maliciosos podem causar a demanda inaceitável de processamento ou de memória. Certamente mais cuidado pode ser assegurado com algoritmos definidos pela aplicação.

A segurança de um sistema como um todo depende também da segurança e da integridade de seus procedimentos operacionais, seu pessoal e do regulamento administrativo para esses procedimentos.

2.3. Padrões de assinatura digital

Atualmente um dos padrões mais utilizados para assinatura digital é o CMS (IETF; 1999, 2002, 2004), também conhecido por PKCS#7, desenvolvido originalmente pela empresa RSA Security Inc..

Outro padrão que tende a ser bastante utilizado, em especial por aplicativos Web, é o “XML Signature”, cuja especificação é mantida pela organização World Wide Web Consortium (W3C) e Internet Engineering Task Force – a última especificação é dada pela RFC-3275 (IETF, 2002).

Neste trabalho estará sendo focado o padrão “XML Signature”, a partir do qual estará sendo analisada a proposta feita pelo ETSI no aprimoramento desse padrão.

3. O PADRÃO XML SIGNATURE

3.1. Histórico do XML

O padrão XML[1](W3C, 2004), estabelecido por uma recomendação publicada pela World Wide Web Consortium (W3C) em 1998, em trabalho conjunto com o Internet of Engineering Task Force (IETF), tornou-se de fato um dos padrões para documentos estruturados mais utilizados, em especial para aplicações Web.

Desde a sua primeira publicação em 1998 até fevereiro de 2004, quando foi publicada a Terceira Edição da Recomendação, a versão da especificação XML era a 1.0. Em fevereiro de 2004, a W3C publicou também a sua Recomendação para a versão 1.1 da XML. A maior parte das diferenças entre essas duas versões diz respeito ao uso de novos conjuntos de caracteres, principalmente em função das atualizações na especificação do padrão Unicode.

A XML pode ser considerada um subconjunto da especificação SGML (Standard Generalized Markup Language) (ISO, 1986). A XML foi concebida visando facilitar a implementação de aplicativos e o uso de documentos estruturados na Web, bem como interoperar com sistemas que interpretam HTML (W3C, 1999) e SGML.

3.2. Estrutura de documento em XML

A estrutura de um documento XML é dada com o uso de marcação, que consiste em elementos. Por sua vez, um elemento XML consiste em uma *tag* de início e uma *tag* de fim.

[1] Quando estivermos falando da linguagem XML estará sendo utilizado no feminino; quando estivermos falando do padrão XML, estará sendo utilizado no masculino.

Uma *tag* de início começa com o símbolo "**<**" e termina com "**>**". *Tags* de fim começam com "**</**" e terminam com "**>**". Uma outra notação que aparece em muitos documentos XML é a que representa um elemento vazio. Ele pode ser representado de duas formas: uma tag de início seguida imediatamente de uma tag de fim (exemplo: `<middlename></middlename>`) ou simplesmente com a chamada tag vazia (exemplo: `<middlename/>`).

Nome de *tag*: a especificação XML é muito específica com relação a nomes de *tags*. O nome de uma *tag* pode ser iniciado com uma letra, um sublinhado ou um sinal de dois-pontos. Os caracteres seguintes podem ser letras, dígitos, sublinhados, hífens, pontos e dois-pontos.

Elemento raiz: cada elemento XML bem formado precisa conter um elemento que contém todos os outros elementos. Esse elemento é denominado elemento raiz.

Na Figura 03, o elemento "DOCUMENTO" é um elemento raiz.

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/css" href="greeting.css"?>
<DOCUMENTO>
    <CUMPRIMENTO>

        Bom dia, senhores !

    </CUMPRIMENTO>
    <MENSSAGEM>
        Bem vindo ao mundo da XML.
    </MESSAGEM>
</DOCUMENTO>
```

Figura 03 – Elemento Raiz (Bem Formado)

Atributos: atributos em XML são muito semelhantes a atributos em HTML – eles são pares de nome e valor que permitem especificar dados adicionais em *tags*, sempre na *tag* inicial.

Na Figura 04, a *tag* inicial de CLIENTE tem o atributo STATUS para acrescentar uma informação a respeito do elemento descrito:

```
<?xml version="1.0"?>
<DOCUMENTO
      <CLIENTE STATUS="Credito Excelente">
            <NOME>
                  <PRENOME>Jose Augusto</PRENOME>
                  <SOBRENOME>Oliveira</SOBRENOME>
            </NOME>
.
.
.
```

Figura 04 – Elemento com Atributo

Não é objetivo deste tópico descrever todas as regras para codificação de um documento XML, que são amplamente esclarecidas em (XML, 2004), mas apenas mostrar algumas características que são mencionadas no desenvolver de alguns capítulos à frente.

3.3. Vantagens do Uso de XML

As principais vantagens do uso da XML, conforme listadas em (Sousa, 2002, p. 45), são:

- ✓ A XML é extensível. A possibilidade de criar *tags* de um modo arbitrário (respeitando sempre as regras de aninhamento), permite adaptar a estrutura de um documento XML a praticamente qualquer situação específica.
- ✓ Os documentos XML são autodescritivos. São, portanto, relativamente fáceis de interpretar, manipular e interrogar. Esta característica pode também revolucionar o modo como as pesquisas são efetuadas na Internet, permitindo o aparecimento de interfaces

de pesquisa que realizem as pesquisas tendo em conta o significado (contexto) dos dados, em vez de se basearem unicamente na associação de palavras-chave.

- ✓ Apesar da sua simplicidade, a XML permite criar estruturas bastante complexas (árvores ou grafos de profundidade arbitrária e, eventualmente, cíclicos e recursivos).
- ✓ A XML é extremamente flexível, possibilitando a representação, quer de dados estruturados, quer de dados semiestruturados.
- ✓ Recorrendo à definição de um esquema, é possível efetuar a validação de documentos. Esta característica revela-se de extrema importância para aplicações em que a verificação estrutural de dados seja vital, como é o caso das bases de dados convencionais.
- ✓ O conteúdo de um documento XML pode ser facilmente manipulado pelas aplicações de software (recorrendo às APIs existentes), o que torna possível atingir níveis de automação bastante elevados.
- ✓ Uma vez que a XML tem uma natureza metalingüística, as organizações podem utilizá-la para desenvolver padrões específicos (novas linguagens baseadas na XML), definindo esquemas comuns, de modo a trocarem, eficientemente, dados entre si. Estes esquemas podem ser disponibilizados publicamente na Internet. O objetivo é utilizar a XML como a “língua comum” para a troca de dados entre os sistemas de informação organizacionais.
- ✓ A XML é um padrão aberto. Os documentos XML são independentes das aplicações, dos sistemas operacionais, etc. Esta

característica pode vir a revolucionar a integração de sistemas heterogêneos.

- ✓ Uma vez que o conteúdo de um documento XML está separado da sua apresentação, é possível obter múltiplas perspectivas sobre um mesmo documento XML (recorrendo à XSL (W3C, 2003)).
- ✓ Um documento XML pode tornar-se uma verdadeira base de dados. Note-se que, embora não exista ainda uma norma, já foram apresentadas várias propostas de linguagens de interrogação para a XML.
- ✓ Manutenção: um documento no formato XML é de fácil manutenção, pois os dados vêm separados das especificações de estilo e *links*.

O padrão XML torna mais fácil a troca de informações entre aplicações dentro de uma mesma organização ou entre organizações diferentes. É igualmente útil como meio de integração de diversas fontes de informação e apresentação de interface uniforme para esses dados.

Atualmente o padrão XML é suportado por todos os principais browsers de mercado, tais como Internet Explorer, Netscape Communicator, Mozila e Opera.

Desde a sua publicação, muitas outras especificações baseadas no padrão XML têm sido publicadas e estão cada vez mais ampliando o uso desse formato, tanto no ambiente Internet (Web) quanto no intercâmbio de informações entre aplicativos diferentes (principalmente entre plataformas com sistema operacional diferente).

3.4. Algumas especificações relacionadas ao padrão XML

A seguir estão listadas algumas das tecnologias que foram especificadas e que são utilizadas no tratamento de documentos no padrão XML.

DTD: O "Document Type Definition" é parte da especificação original XML 1.0 e permite ao desenvolvedor especificar quais elementos e atributos podem ser usados em um documento XML particular. Indica também como deve ser a estrutura e os aninhamentos permitidos. É também chamado de modelo de conteúdo ou esquema de um documento XML.

XSLT: "eXtensible Stylesheet Language Transformation" é uma linguagem que permite que documentos XML sejam transformados de um esquema para outro, ou mesmo para padrões diferente, como páginas HTML ou arquivos PDF.

XPath: "XML Path Language" é a linguagem para endereçamento e consulta de conteúdo dos documentos XML.

XLink: "XML Linking Language" descreve hiperlinks em documentos XML. Estende os conceitos de hiperlinks da HTML.

XPointer: "XML Pointer Language" é um padrão complementar do XLink e descreve mecanismos para endereçamento de partes particulares de um documento XML.

XML Schema: "XML Schema" é um esforço atual do W3C para dar mais flexibilidade e poder na descrição das estruturas dos documentos. Incluindo, inclusive, definições de tipos de dados.

XHTML: "Extensible HyperText Markup Language" é a reformulação da HTML 4.0 baseada na XML. Poderá se tornar, logo, o padrão de fato da INTERNET.

WML: "Wireless Markup Language" é usada em sistemas WAP (telefone) para permitir um ambiente de INTERNET móvel e é inteiramente baseada na XML.

SVG: Scalable Vector Graphics é uma aplicação XML usada para descrever gráficos vetoriais 2D, textos e imagens "raster".

XML Encryption Syntax and Processing: especifica o processo para encriptação de dados e a representação do resultado em XML.

XML Signature Syntax and Processing: especifica a sintaxe em XML e as regras de processamento para criar e representar assinaturas digitais.

DOM: "Document Object Model" descreve como um analisador gramatical ("parser") retorna a informação contida em um documento XML. Os elementos do documento XML são descritos como nós de uma árvore que pode ser percorrida pelo programador.

SAX: "Simple API for XML" fornece outro modelo de programação usado pelos "parsers". Ele é baseado em eventos ao invés de uma árvore de nós que pode ser percorrida.

3.5. XML Signature

Tendo em vista a grande utilização do padrão XML, em especial no ambiente Web, foi definido pelo consórcio W3C, em trabalho que no início contou com a participação também da IETF, a especificação **XML Signature Syntax and Processing**, correspondente à RFC-3275.

Complementarmente à especificação da XML Signature, foi publicada a especificação **XML Encryption Syntax and Processing** como suporte para o serviço de sigilo de informações colocadas em documentos no padrão XML.

A seguir, baseado principalmente em Gokul (2002), é dada uma descrição dos principais aspectos que envolvem uma assinatura digital em XML.

A Assinatura Digital em XML segue os padrões de formatação do XML e serve como um carimbo quando adicionado a um documento XML, assegurando a autenticidade e originalidade do documento.

O receptor de um documento XML assinado (o qual pode conter informações confidenciais de negócios) pode verificar os elementos da assinatura para assegurar que a origem do documento bem como sua integridade.

Uma típica assinatura digital em XML (grafada por XML Dsig) parece algo como a figura abaixo:

```
<Signature xmlns="http://www.w3.org/2000/09/xmldsig#">
   <SignedInfo Id="MyXMLDSigExample">
      <CanonicalizationMethod
         Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
      <SignatureMethod Algorthm="http://www.w3.org/2000/09/xmldsig#dsa-sha1" />
      <Reference URI="http://www.informit.com/wssecurity/art02.xml">
         <DigestMethod Algorthm="http://www.w3.org/2000/09/xmldsig#sha1" />
         <DigestValue>j61wx3rvEP00v23Rup4NbeVu8nk=</DigestValue>
      </Reference>
   </SignedInfo>
   <SignatureValue>AM0E~PE*</SignetureValue>
</Signature>
```

Figura 05 – Exemplo de uma Assinatura Digital em XML
Fonte: Gokul, 2002

3.5.1. Terminologia da assinatura digital em XML

Antes de olhar os elementos de uma assinatura em XML, é interessante entender alguns conceitos básicos e a terminologia que são mais específicos do mundo da assinatura digital em XML. Segue uma breve explicação de cada um deles.

3.5.1.1. Recursos Assinados

Além de assinar documentos em formato XML, XML Dsig pode também ser utilizado para assinar outros tipos de dados, tais como:

- ✓ Dados codificados em ascii em diversos tipos de formatos (como, por exemplo, um arquivo html contendo qualquer tipo de informação);
- ✓ Dados em código binário (imagens em formato jpg, por exemplo)
- ✓ Dados em formato XML.

É assim apropriado usar o termo *recurso* para representar a entidade que é assinada por XML DSig. Assim, os *recursos* assinados usando XML DSig podem ser chamados ***recursos assinados.***

3.5.1.2. Digests

Para criar uma assinatura digital para um recurso, um pequeno e único código gerado a partir de um algoritmo de "hash" (chamado de "hash" ou "digest" ou, algumas vezes, em português, de resumo) é transformado usando a chave privada do remetente. Este "digest" (chamado de *digest value* ou *valor resumo*) tanto quanto o algoritmo usado para a transformação (chamado "*digest method"*) torna-se parte integral da assinatura digital em XML gerada para o documento.

O valor obtido para o "digest" é muito sensível para qualquer alteração feita no recurso ao qual se refere. Assim, ele torna-se vital para assegurar que o documento não foi modificado no transito até o receptor.

Cada recurso assinado, atestado pela assinatura, precisa ter o seu correspondente *digest* (valor calculado pelo algoritmo de hash).

3.5.1.3. Processo de Normalização

É possível que dois documentos similares em XML que contenham idênticos dados possam diferir em termos de representação textual em função dos espaços em branco, salto de linhas, representação de elemento, etc. Normalização é o processo de eliminar os efeitos dessas pequenas diferenças, tal que o código hash gerado não é afetado pelas variações textuais.

Vários algoritmos padrões estão atualmente disponíveis para normalização de XML. A sintaxe da assinatura XML inclui um elemento <CanonicalizationMethod> para indicar o algoritmo usado para o documento em questão.

Os documentos representados nas figuras 06 e 07, em termos de processamento dos elementos XML, são iguais, apesar das pequenas diferenças visuais (ordem da apresentação das propriedades, comentário e espaço em branco após valor de elemento), portanto para que os mesmos tenham o mesmo valor de hash precisam ser normalizados. O resultado da normalização é representado pela Figura 08.

```
<?xml version="1.0"?>
    <mensagens>
        <nota data='30/10/2001' ID='mensagem001' >
                    <para>  Mauro  </para>
                    <de>  Jane  </de>
                    <assunto>Lembrete</assunto>
                    <mensagem>Não esqueça de mim neste fim de semana!   </mensagem>
        </nota>
        <nota data=31/10/2001 ID='mensagem002' >
                    <para>Jane   </para>
                    <de>  Mauro</de>
                    <assunto>  Re: Lembrete  </assunto>
                    <mensagem>Não esquecerei! </mensagem>
        </nota>
    </mensagens>
```

Figura 06 – Documento 01 Original

```
<?xml version="1.0"?>
<mensagens>
        <nota ID="mensagem001" data='30/10/2001' >
                <para>Mauro  </para>
                <de>  Jane </de>
                <assunto>Lembrete </assunto>
                <mensagem>Não esqueça de mim neste fim de semana!</mensagem>
        </nota> <!—isto pode fazer a diferença -->
        <nota ID="mensagem002" data='31/10/2001' >
                <para>Jane</para>
                <de>Mauro  </de>
                <assunto>Re: Lembrete</assunto>
                <mensagem >Não esquecerei ! </mensagem>
        </nota>
</mensagens>
```

Figura 07 – Documento 01 Alterado

```
<?xml version=”1.0”?>
<mensagens>
<nota ID=”mensagem001” data=30/10/2001>
<para>Mauro</para>
<de>Jane</de>
<assunto>Lembrete</assunto>
<mensagem>Não esqueça de mim neste fim de semana!</mensagem>
</nota>
<nota ID=”mensagem002” data=31/10/2001>
<para>Jane</para>
<de>Mauro</de>
<assunto>Re: Lembrete</assunto>
<mensagem>Não esquecerei!</mensagem>
</nota>
</mensagens>
```

Figura 08 – Documento 01 Normalizado

3.5.2. Os elementos da assinatura digital em XML

Para ter um melhor entendimento dos diferentes elementos e nós que constituem uma assinatura digital em XML, observe a Figura 09, a qual contém apenas os elementos sem qualquer dado.

3.5.2.1. Elemento <Signature>

O elemento Signature é o elemento raiz de todo padrão de Assinatura Digital em XML. Ele contém os três principais elementos apresentados abaixo:

- `SignedInfo` (<SignedInfo>...</SignedInfo>)
- `SignatureValue` (<SignatureValue >...</SignatureValue >)
- `KeyInfo` (<KeyInfo >...</KeyInfo >)

Figura 09 – Representação dos Elementos da Assinatura Digital em XML
Fonte: Gokul, 2002

3.5.2.2. Elemento <SignedInfo>

O elemento obrigatório `SignedInfo` é o que contém todas as informações requeridas que dizem respeito ao recurso assinado. Isto inclui o seguinte:

a. O algoritmo de normalização aplicado no recurso assinado

`<CanonicalizationMethod`

`Algorithm`="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>

Note que o mesmo algoritmo pode ser utilizado para normalizar múltiplos recursos especificados nos elementos `<Reference>`.

b. O algoritmo que é usado para a geração da assinatura digital e validação:

`<SignatureMethod`

`Algorithm`="http://www.w3.org/2000/09/xmldsig#dsa-sha1"/>

Este algoritmo é uma combinação de um algoritmo de hash e um algoritmo de criptografia assimétrica. Ele é aplicado sobre o recurso normalizado indicado por `SignedInfo` para calcular a assinatura digital. Embora a metodologia de assinatura esteja dentro do elemento <SignedInfo>, a assinatura real propriamente (dentro do elemento <SignatureValue>) está fora desse elemento.

A especificação XML DSig indica o algoritmo DSA com SHA-1 como requerido e o algoritmo RSA com SHA-1 como recomendado para gerar assinatura digital.

c. Um ou mais elementos <Reference>.

3.5.2.3. Elemento <Reference>

Cada recurso assinado (o documento XML, neste caso) que a assinatura autentica tem que estar dentro de um elemento <Reference>. Uma assinatura pode ser aplicada para assinar múltiplos recursos, por isso é comum encontrar diversos nós <Reference>...</Reference> dentro de um elemento `Signature`.

Cada nó <Reference> contém o seguinte:

a. Uma referência através do parâmetro URI para o recurso assinado (opcional).

<Reference URI="http://www.informit.com/wssecurity/art02.xml">

O parâmetro URI é opcional porque haverá muitos casos nos quais as assinaturas digitais são enviadas juntas com os documentos, assim sendo este parâmetro pode não ser necessário.

b. Uma lista ordenada de algoritmos de transformação aplicados ao documento original antes do cálculo do hash, através do elemento <Transforms> (opcional),

<Transforms>

<Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>

</Transforms >

Por segurança adicional, o signatário pode aplicar uma ou mais transformações sobre o dado original antes de assiná-lo. Essas transformações, bem como a ordem na qual elas são aplicadas, precisam ser comunicados para o receptor de tal forma que o processo inverso para recompor o documento possa ser aplicado.

c. O algoritmo de hash que foi aplicado no recurso assinado para gerar o hash (obrigatório).

<DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />

d. O valor real do hash, o qual for gerado pela aplicação do algoritmo de hash sobre o recurso assinado (obrigatório)

<DigestValue>j6lwx3rvEPO0v23Rup4NbeVu8nk=</DigestValue>

3.5.2.4. Elemento <SignatureValue>

Este elemento obrigatório contém o valor da assinatura digital, calculado pela aplicação do algoritmo de assinatura sobre o elemento indicado por *SignedInfo*.

<SignatureValue>AM0E~PE*</SignatureValue>

3.5.2.5. Elemento <KeyInfo>

O elemento opcional `KeyInfo` contém referências para a chave pública do remetente, a qual pode então ser usada pelo receptor para validar a assinatura digital e os recursos. É opcional porque é desnecessário anexar a chave pública a cada documento assinado que o remetente está transmitindo para o mesmo receptor.

O elemento `KeyInfo` tipicamente contém as chaves públicas, os nomes das chaves, certificados e outras informações de gerenciamento de chaves públicas.

O elemento `KeyInfo` contém:

a. Um valor alfanumérico que representa um par de chaves:

<KeyName>ABC Key</KeyName >

b. A própria chave como um outro elemento:

<KeyValue>1awerfvzxcvzxv1343x5fcds2f1r423</KeyValue >

Dependendo do tipo de chave (RSA, DSA, etc.), o elemento KeyInfo pode definir ele próprio, conforme exemplo abaixo:

```
<KeyInfo>
   <KeyValue>
     <RSAKeyValue>
            <Modules>y32K..4NE=</Modules >
        <Exponent >Aw==</Exponent>
     </RSAKeyValue >
   </KeyValue >
</KeyInfo >
```

Figura 10 – Exemplo de elemento KeyInfo

3.5.2.6. Tipos de Assinaturas Digitais em XML

Há diversos modos de juntar o recurso assinado com a assinatura gerada para o recurso, conforme ilustra a Figura 11.

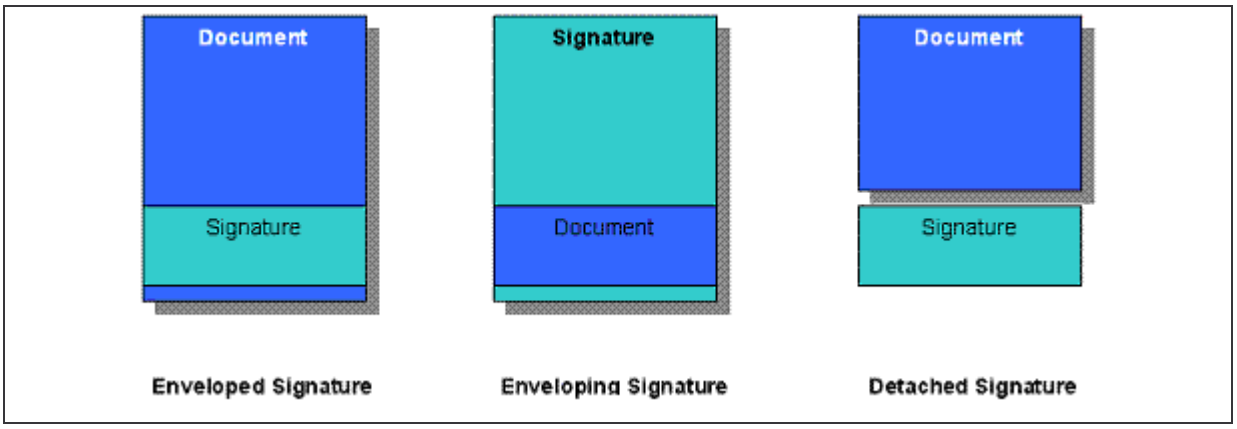

Figura 11 – Tipos de Assinatura Digital em XML
Fonte: Gokul, 2002

a. Enveloped Signatures

O objeto assinado é o próprio documento XML ou um elemento dentro dele, excluindo-se, no cálculo da assinatura, o elemento `Signature`. Desta forma, a assinatura gerada é embutida dentro do próprio documento XML assinado.

Este tipo é aplicável somente se o recurso assinado é um documento XML. Como parte do processo da criação e da verificação da assinatura, o próprio elemento Signature é excluído do mesmo antes do cálculo da assinatura.

b. Enveloping Signatures

O objeto assinado é um elemento `Object` dentro do elemento `Signature` do próprio documento XML.

Tal como o anterior, este tipo é relevante apenas para documentos XML.

c. Detached Signatures

O objeto assinado e o documento contendo a assinatura estão separados. Isto é mais adequado para assinar recursos que não sejam XML (por exemplo, arquivos binários e html).

Em tais casos, o elemento <Reference> usualmente contém a URI do recurso assinado, o qual é autenticado.

3.5.3. Usando assinaturas digitais em XML

O processo de geração da assinatura segue os seguintes passos:

a. Identificar os recursos que precisam ser assinados;
b. Aplicar os algoritmos de transformação, se for o caso (isto inclui os algoritmos de criptografia);
c. Aplicar o algoritmo de hash sobre o recurso e calcular o seu valor;
d. Repetir o processo acima para cada recurso que precisa ser assinado. Inserir os detalhes no elemento <Reference>;
e. Agrupar todos elementos <Reference> dentro do elemento <SignedInfo>. Indicar os métodos de normalização e assinatura;
f. Normalizar o conteúdo do elemento <SignedInfo>, aplicar os algoritmos de assinatura e gerar a assinatura digital em XML;
g. Colocar a assinatura gerada dentro do elemento <SignatureValue>;
h. Adicionar informações chave relevantes, se existir, e produzir o elemento <Signature>.
i. Enviar o documento XML gerado para o receptor.

O processo de verificação da assinatura segue os seguintes passos:

a. Verificar a assinatura contida no elemento <SignatureValue>. Fazer isto calculando o hash do elemento <SignedInfo>, e aplicando o algoritmo de assinatura com a chave pública do signatário. Este processo pode identificar se a chave é válida ou não.

b. Se esse processo for bem sucedido, aplicar o algoritmo de hash para os recursos requeridos e recalcular o hash para os mesmos. Comparar esses valores com aqueles dentro das correspondentes tags do elemento <Reference>. Para cada documento/elemento apontado pelo elemento Reference dentro do elemento SignedInfo deve-se aplicar os algoritmos de transformação indicados no elemento Transforms, na ordem inversa ao que foi aplicado na criação da assinatura

Primeiramente, assinatura em um documento transformado não garante nenhuma informação eliminada pelo algoritmo de transformação: somente o que é assinado é seguro. Note que o uso do algoritmo de normalização de XML [Xml-c14n] assegura de que todas as entidades internas e os namespaces de XML estejam expandidos dentro do conteúdo que está sendo assinado. Todas as entidades são substituídas com suas definições e a forma normalizada representa explicitamente o namespace que um elemento herdaria de outra maneira. As aplicações que não normalizam o conteúdo de XML (especial o elemento SignedInfo) não devem usar entidades internas e devem representar o namespace explicitamente dentro do conteúdo que está sendo assinado desde que não podem confiar na normalização para fazer isto para elas. Também, os usuários preocupados com a integridade das definições do tipo do elemento associado com a instância do XML que está sendo assinado podem desejar assinar também aquelas definições (isto é, o esquema, o DTD, ou a descrição da língua natural associada com o namespace/identificador). Em segundo lugar, um envelope que contém a informação assinada não é garantido pela assinatura. Por exemplo, quando um envelope cifrado contém uma assinatura, a assinatura não protege a autenticidade

ou a integridade do cabeçalho não assinado do envelope nem sua mensagem cifrada, garante somente o conteúdo realmente assinado.

Na figura 12 podemos visualizar um documento XML que foi gerado com uma assinatura digital embutida ao mesmo. Pode-se reparar que neste caso está incluso o certificado digital com a chave pública do signatário (item <X509Data>).

```
<Signature xmlns=”http://www.w3.org/2000/09/xmldsig#”>
  <SignedInfo>
    <CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-
20010315"></CanonicalizationMethod>
    <SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#dsa-sha1"></SignatureMethod>
    <Reference URI="#Res0">
      <Transforms>
        <Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"></Transform>
      </Transforms>
      <DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"></DigestMethod>
      <DigestValue>psLjBQtzI7t7wiozRfLoNdaIb08=</DigestValue>
    </Reference>
  </SignedInfo>
  <SignatureValue>
    ZimJc+beK3HLLlpPcxh0kvBIPYB6YQs+as6SqXID4pY0tyF6qWQ3KQ==
  </SignatureValue>
  <KeyInfo>
    <KeyValue>
      <DSAKeyValue>
        <P>
          /X9TgR11EilS30qcLuzk5/YRt1I870QAwx4/gLZRJmlFXUAiUftZPY1Y+r/F9bow9s
          ubVWzXgTuAHTRv8mZgt2uZUKWkn5/oBHsQIsJPu6nX/rfGG/g7V+fGqKYVDwT7g/bT
          xR7DAjVUE1oWkTL2dfOuK2HXKu/yIgMZndFIAcc=
        </P>
        <Q>l2BQjxUjC8yykrmCouuEC/BYHPU=</Q>
        <G>
          9+GghdabPd7LvKtcNrhXuXmUr7v6OuqC+VdMCz0HgmdRWVeOutRZT+ZxBxCBgLRJFn
          Ej6EwoFhO3zwkyjMim4TwWeotUfI0o4KOuHiuzpnWRbqN/C/ohNWLx+2J6ASQ7zKTx
          vqhRkImog9/hWuWfBpKLZl6Ae1UlZAFMO/7PSSo=
        </G>
        <Y>
          HPnTW4UAvR4MIYOqym1zdLFlzNREQpKCxOMbiSvD04es7RONiBECT6OtlMWAZ9k/9N
          gOK/HmxObSC5inAjXN2omNZzBburQuJAmRs/G2nwRp49yLRNHWq1922jCYoZ6iFmNA
          iraKSIOYTS1LnfGEEgl7zNc/XDsArRtZ+LBdrZo=
        </Y>
      </DSAKeyValue>
    </KeyValue>
    <X509Data>
      <X509IssuerSerial>
        <X509IssuerName>CN=Nick Chase,OU=InformIT,O=Pearson,L=New Port
Richey,ST=Florida,C=US</X509IssuerName>
        <X509SerialNumber>1089899916</X509SerialNumber>
      </X509IssuerSerial>
      <X509SubjectName>CN=Nick Chase,OU=InformIT,O=Pearson,L=New Port
Richey,ST=Florida,C=US</X509SubjectName>
      <X509Certificate>
MIIDGjCCAtgCBED2jYwwCwYHKoZIzjgEAwUAMHMxCzAJBgNVBAYTAlVTMRAwDgYDVQQIEwdGbG9y
aWRhMRgwFgYDVQQHEw9OZXcgUG9ydCBSaWNoZXkxEDAOBgNVBAoTB1BlYXJzb24xETAPBgNVBAsT
CEluZm9ybUlUMRMwEQYDVQQDEwpOaWNrIENoYXNlMB4XDTA0MDcxNTEzNTgzNloXDTA0MTAxMzEz
NTgzNlowczELMAkGA1UEBhMCVVMxEDAOBgNVBAgTB0Zsb3JpZGExGDAWBgNVBAcTD05ldyBQb3J0
IFJpY2hleTEQMA4GA1UEChMHUGVhcnNvbjERMA8GA1UECxMISW5mb3JtSVQxEzARBgNVBAMTCk5p
Y2sgQ2hhc2UwggG3MIIBLAYHKoZIzjgEATCCAR8CgYEA/X9TgR11EilS30qcLuzk5/YRt1I870QA
wx4/gLZRJmlFXUAiUftZPY1Y+r/F9bow9subVWzXgTuAHTRv8mZgt2uZUKWkn5/oBHsQIsJPu6nX
/rfGG/g7V+fGqKYVDwT7g/bTxR7DAjVUE1oWkTL2dfOuK2HXKu/yIgMZndFIAccCFQCXYFCPFSML
zLKSuYKi64QL8Fgc9QKBgQD34aCF1ps93su8q1w2uFe5eZSvu/o66oL5V0wLPQeCZ1FZV4661FlP
5nEHEIGAtEkWcSPoTCgWE7fPCTKMyKbhPBZ6i1R8jSjgo64eK7OmdZFuo38L+iE1YvH7YnoBJDvM
pPG+qFGQiaiD3+Fa5Z8GkotmXoB7VSVkAUw7/s9JKgOBhAACgYAc+dNbhQC9Hgwhg6rKbXN0sWXM
1ERCkoLE4xuJK8PTh6ztE42IEQJPo62UxYBn2T/02A4r8ebE5tILmKcCNc3aiY1nMFu6tC4kCZGz
8bafBGnj3ItE0darX3baMJihnqIWY0CKtopIg5hNLUud8YQSCXvM1z9cOwCtG1n4sF2tmjALBgcq
hkjOOAQDBQADLwAwLAIUUGLeAge6Ldui6tYQxPS4L3WMXkoCFCKEwGg3Ah8jEBum+rbhmrvo3xp/
      </X509Certificate>
    </X509Data>
  </KeyInfo>
  <dsig:Object xmlns="" xmlns:dsig="http://www.w3.org/2000/09/xmldsig#" Id="Res0"><theroot>

<thedata>data!</thedata>
</theroot></dsig:Object>
</Signature>
```

Figura 12 – Exemplo de Documento XML com Assinatura Digital
Fonte: Chase, 2004

4. Desafios à Utilização da Assinatura Digital

A utilização da assinatura digital em documentos eletrônicos, substituindo o equivalente em papel, enfrenta ainda alguns desafios. Tratamos aqui de três dos principais, que são:

- Política de assinatura digital;
- Realização de uma assinatura com a data no passado; e
- Longevidade de um documento assinado digitalmente.

4.1. Política de assinatura digital

No mundo do documento em papel, uma assinatura pode estar inserida em alguns contextos, como por exemplo:

- As partes interessadas no contrato representado pelo documento;
- Testemunhal ou contra-assinatura, assinatura colocada como um testemunho que a outra parte assinou;
- Notarial, uma assinatura para dar um caráter oficial (legal) de reconhecimento do documento sendo assinado.

Pode haver, ainda, múltiplas assinaturas de cada uma das partes interessadas e haver uma determinada ordem para que as mesmas ocorram, bem como haver a necessidade de apresentação anterior de determinado documento que comprove que determinado signatário tem realmente autoridade para assinar aquele tipo de documento.

Como transportar para o mundo da assinatura digital todas essas nuances embutidas na assinatura de um documento no mundo do papel?

Isto deverá, tanto quanto possível, ser tratado na implementação da Política de Assinatura.

Política de assinatura é um conjunto de regras para a criação e validação de uma assinatura eletrônica, sob a qual a assinatura pode ser determinada se é válida. Uma política de assinatura, portanto, necessita estabelecer as condições sob as quais as partes envolvidas em um negócio concordam em aceitar assinatura eletrônica e as regras para sua criação e verificação. É geralmente aceita que uma política de assinatura consiste de duas partes: uma política de criação da assinatura e uma política de verificação da assinatura.

Uma política de assinatura pode potencialmente servir a dois propósitos de negócio:

- Um direcionamento dos procedimentos usados por uma organização ou entidade na criação, verificação e uso de assinaturas eletrônicas no seu próprio interesse (isto é, ser confiável para as demais); e
- Um direcionamento das condições sobre as quais uma assinatura eletrônica será aceita como válida por aquela organização (isto é, para ser aceita pelas demais).

Uma política de assinatura pode ser relacionada a uma validação de uma assinatura simples ou a múltiplas assinaturas em um único documento, por exemplo, um contrato. Por outro lado, ela pode ser potencialmente muito complexa, gerenciando assinaturas que são requeridas em múltiplos estágios de uma transação comercial, por exemplo, transações de comércio internacional envolvendo controles de exportação/importação. Essas políticas podem ser distinguidas de outras chamando-as de política de assinatura transacional ou política de assinatura contratual.

Abaixo alguns exemplos de circunstâncias que cercam a criação de uma assinatura para que ela seja válida:

- exigências que a assinatura seja criada por escrito, ou sob a mão do signatário;
- que a assinatura seja autenticada ou testemunhada;
- que a transação seja registrada.

  Estas se aplicam mais geralmente em:

  - Testamentos;
  - Casos de família;
  - Transações que envolvem terra;
  - Proteção do consumidor e/ou serviços financeiros;
  - Contratos para a garantia ou fiança;
  - Exigências sob as leis de companhia aberta.

4.2. Realização de assinatura com uma data no passado

Um dos principais problemas associado à assinatura digital é a possibilidade de, ao gerar a assinatura digital, se especificar uma data no passado, pois essa data não é obtida de uma entidade externa confiável.

Isto pode causar enormes problemas, pois se uma chave privada tem sua confidencialidade comprometida, mesmo revogada posteriormente, a mesma poderá ser utilizada indevidamente para diversos fins utilizando-se uma data no passado anterior à data da revogação.

Isto pode acontecer, pois na especificação em uso (CMS/XML Signature), não há uma obrigatoriedade de a data e hora serem fornecidas por uma entidade externa confiável e se ter essa informação de forma segura.

4.3. Longevidade de um documento assinado digitalmente

À medida que o uso de documentos assinados digitalmente vai crescendo, aumenta a necessidade que os mesmos possam ser guardados por períodos de tempo mais longos e possam ser verificados/validados a qualquer momento.

Ao tratar das assinaturas digitais de longo prazo, todos os dados usados na verificação (a saber, caminho de certificação e informação de revogação) de tais assinaturas devem ser armazenados e ter uma estampilha temporal associada para dar maior garantia da sua validade no tempo. Considerações similares aplicam-se aos certificados de atributo se aparecerem dentro da assinatura. Em alguns sistemas, pode ser conveniente adicionar estes dados à assinatura digital (como propriedades não assinadas) para finalidades de arquivamento. Alternativamente, outros sistemas podem considerar conveniente arquivá-los em outra parte. Nesses casos, cada assinatura eletrônica deve incorporar referências a todos estes dados dentro da assinatura, reduzindo também o tamanho da assinatura digital armazenada.

Desta forma, uma propriedade importante para a longa duração das assinaturas digitais é que a mesma, uma vez validada, precisa continuar a ser válida por meses ou anos mais tarde.

Um signatário, um verificador ou ambos podem ser solicitados para prover, a pedido, prova de que a assinatura digital foi criada ou verificada durante o período de validade de todos os certificados que fazem parte do caminho de certificação. Neste caso, o signatário, o verificador ou ambos também serão solicitados a mostrar provas de que todos os certificados de usuários e da

Autoridade Certificadora utilizados não estavam revogados quando a assinatura foi criada ou verificada.

Seria inaceitável considerar uma assinatura como inválida mesmo se as chaves ou certificados fossem mais tarde comprometidos. Assim, para prover evidência de longo prazo da validade da assinatura, há a necessidade de se ser capaz de demonstrar que a chave de assinatura era válida em um momento bem próximo ao que a assinatura foi criada.

A Estampilha Temporal (“time-stamping”) fornecida pela Autoridade de Estampilha Temporal (AET) pode suprir tal evidência. Uma Estampilha Temporal é obtida enviando um valor de "hash" de um determinado dado para a AET. A Estampilha Temporal retornada é um objeto de dados assinado que contém: um valor de "hash", a identidade da AET e a Estampilha Temporal. Isto prova que o dado existia antes daquela data e hora indicada. O padrão de estampilha temporal seguido é o definido pela RFC-3161 (IETF, 2001).

Obtendo uma Estampilha Temporal de uma assinatura eletrônica antes da revogação da chave privada do signatário e antes do fim da validade do certificado, provê evidência de que a assinatura foi criada enquanto o certificado era válido e antes que ele fosse revogado.

Se um receptor quer manter uma assinatura eletrônica válida, terá que assegurar que também obteve uma Estampilha Temporal válida para a assinatura, antes que a chave (e qualquer chave envolvida na validação) seja revogada. O melhor é que a Estampilha Temporal seja obtida o quanto antes logo depois da assinatura.

É importante notar que assinaturas podem ser geradas "off-line" e a Estampilha Temporal ser obtida mais tarde por qualquer um, por exemplo, pelo

signatário ou qualquer receptor interessado na validade da assinatura. A Estampilha Temporal pode assim ser provida pelo signatário junto com o objeto de dados assinado, ou obtido pelo receptor após a recepção do objeto assinado.

5. A PROPOSTA DO ETSI

O ETSI (European Telecommunications Standards Institute) é um instituto independente, sem fins lucrativos, cuja responsabilidade é elaborar padrões de tecnologia da informação e telecomunicação para a comunidade européia. Seus objetivos são oficialmente reconhecidos pela Comissão Européia e pela "European Fair Trade Association" (EFTA – Associação Européia de Comércio).

A proposta do ETSI é compatível com a Diretiva "1999/93/EC" do Parlamento Europeu e da Assembléia da União Européia de 13 de dezembro de 1999, que estabeleceu uma estrutura padrão para o uso de assinaturas eletrônicas.

Em relação às especificações sobre assinatura digitais, o objetivo do ETSI é estabelecer uma padronização que permita a implementação de sistemas para assinaturas digitais que incorporem recursos para a sua verificação com segurança por longos períodos de tempo e guarde evidências de sua validade mesmo que o signatário ou o verificador tente negá-la mais tarde.

Em fevereiro de 2002, o ETSI publicou a sua proposta para assinatura digital usando XML, documento "**ETSI-TS-101-903-XML Advanced Electronic Signatures (XAdES)**", versão 1.1.1, o qual estende as especificações da W3C, acrescentando diversos elementos para qualificar uma assinatura digital no formato XML. Em abril de 2004 foi publicada a versão 1.2.2 dessa especificação, na qual foram revistos os formatos estabelecidos como padrões.

Ainda em abril de 2002, o ETSI publicou também o documento "**ETSI-TR-102-038-XML format for signatures policies**", no qual são abordados os vários aspectos de uma política de assinatura que precisam ser levadas em consideração para dar maior credibilidade e abrangência à assinatura digital. Com esta proposta,

pretende-se cobrir os aspectos abordados dentro da Política de Assinatura que serviu de base para o documento ETSI-TS-101-903.

Conforme (ETSI, 2004), para estar em conformidade com a especificação 1.2.2 do padrão XAdES, um dos formatos abaixo tem que ser adotado:

- Assinatura Eletrônica Básica (XAdES-BES)
- Assinatura Eletrônica com Política Explícita (XAdES-EPES)
- Assinaturas Eletrônicas com Validação de Dados, que pode ser**:**
  - Assinatura Eletrônica com Validação de Tempo (XAdES-T); ou
  - Assinatura Eletrônica com Validação Completa (XAdES-C)

5.1. Formato XAdES-BES

O XAdES-BES (Figura 13) é o mais simples dos formatos previsto nesta especificação.

Na notação XML abaixo, o prefixo `ds:` antes do nome do elemento XML indica que o mesmo faz parte da especificação XMLDSig da W3C (W3C, 2002). Quando não tiver esse prefixo, significa que é definição da especificação do ETSI. O caracter “?” após o nome do elemento significa que pode ter zero ou uma ocorrência do mesmo; “+” significa uma ou mais ocorrências; e “*” significa zero ou mais ocorrências.

Para este formato é mandatório proteger o(s) certificado(s) que assina(m) o conteúdo. Desta forma, deve conter ao menos um dos seguintes elementos:

a. A propriedade assinada `SigningCertificate`. Esta propriedade deve conter a referência e o valor de resumo do certificado que

assina o conteúdo. Pode conter referências e valores de resumo de outros certificados (que pode formar uma cadeia até o certificado raiz).

b. O elemento `ds:KeyInfo`. Se o elemento `SigningCertificate` estiver presente na assinatura, nenhuma limitação aplica-se a este elemento. Se o elemento `SigningCertificate` não estiver presente na assinatura, então as seguintes limitações aplicam-se:
   i. O elemento de `ds:KeyInfo` deve incluir um `ds:X509Data` contendo o certificado que assina o conteúdo;
   ii. O elemento `ds:KeyInfo` também pode conter outros certificados, formando uma cadeia que pode atingir o certificado raiz;
   iii. O elemento de `ds:SignedInfo` deve conter um elemento `ds:Reference` que referencia `ds:KeyInfo`, de modo que o último seja incluído na computação do valor da assinatura. Desta maneira, o certificado que assina o conteúdo é garantido pela assinatura.

Este formato pode também conter as seguintes propriedades:

a. A propriedade assinada `SigningTime`: contém a data e hora em que o signatário indica estar fazendo a assinatura;
b. A propriedade assinada `DataObjectFormat`: este elemento, quando presente, indica o tipo do objeto que foi assinado (texto, imagem, etc.);
c. A propriedade assinada `CommitmentTypeIndication`: elemento que pode ser utilizado para indicar o tipo de compromisso que está

sendo assumido na assinatura efetuada (por exemplo: prova de reconhecimento que recebeu o documento; prova de reconhecimento que criou, aprovou e enviou o documento assinado; etc.);

d. A propriedade assinada `SignerRole`: elemento para indicar o cargo ou posição do signatário na empresa/organização; pode ser através de um nome do cargo e/ou de um certificado específico que indica essa posição na empresa/organização (“Attribute Certificate”);
e. A propriedade assinada `SignatureProductionPlace`: elemento para indicar o local onde (para todos efeitos) o signatário está assinando aquele documento (pode conter: cidade, estado, país e código postal);
f. Uma ou mais propriedades `IndividualDataObjectsTimeStamp` assinadas ou uma propriedade `AllDataObjectTimeStamp` assinada;
g. Uma ou mais propriedades não assinadas `CounterSignature`: este elemento é utilizado para dar suporte à assinaturas de documentos onde são necessários múltiplas assinaturas e no qual a ordem em que as mesmas são aplicadas é importante.

Os elementos abaixo listados, que podem ou não estar presentes, são também garantidos através de uma Estampilha Temporal.

- **`AllDataObjectsTimeStamp`:** este elemento contém a Estampilha Temporal calculada antes da geração da assinatura, sobre a seqüência formada por todos os elementos `ds:Reference` dentro do `ds:SignedInfo`, referenciando

tudo que o signatário quer assinar, exceto o elemento `SignedProperties`. É um elemento assinado. Sua finalidade é provar que todas as assinaturas para os objetos apontados pelos elementos `ds:Reference` foram feitas depois da data e hora contida nessa estampilha.

- **`IndividualDataObjectsTimeStamp:`** este elemento contém a Estampilha Temporal, calculada antes da geração da assinatura, sobre a seqüência formada por alguns elementos `ds:Reference` dentro do `ds:SignedInfo`. Esta seqüência não pode conter um elemento `ds:Reference` considerado no elemento `SignedProperties`. É um elemento assinado que qualifica o objeto assinado. Sua finalidade é provar que todas as assinaturas para os objetos referenciados foram feitas antes da data e hora contida neste elemento.

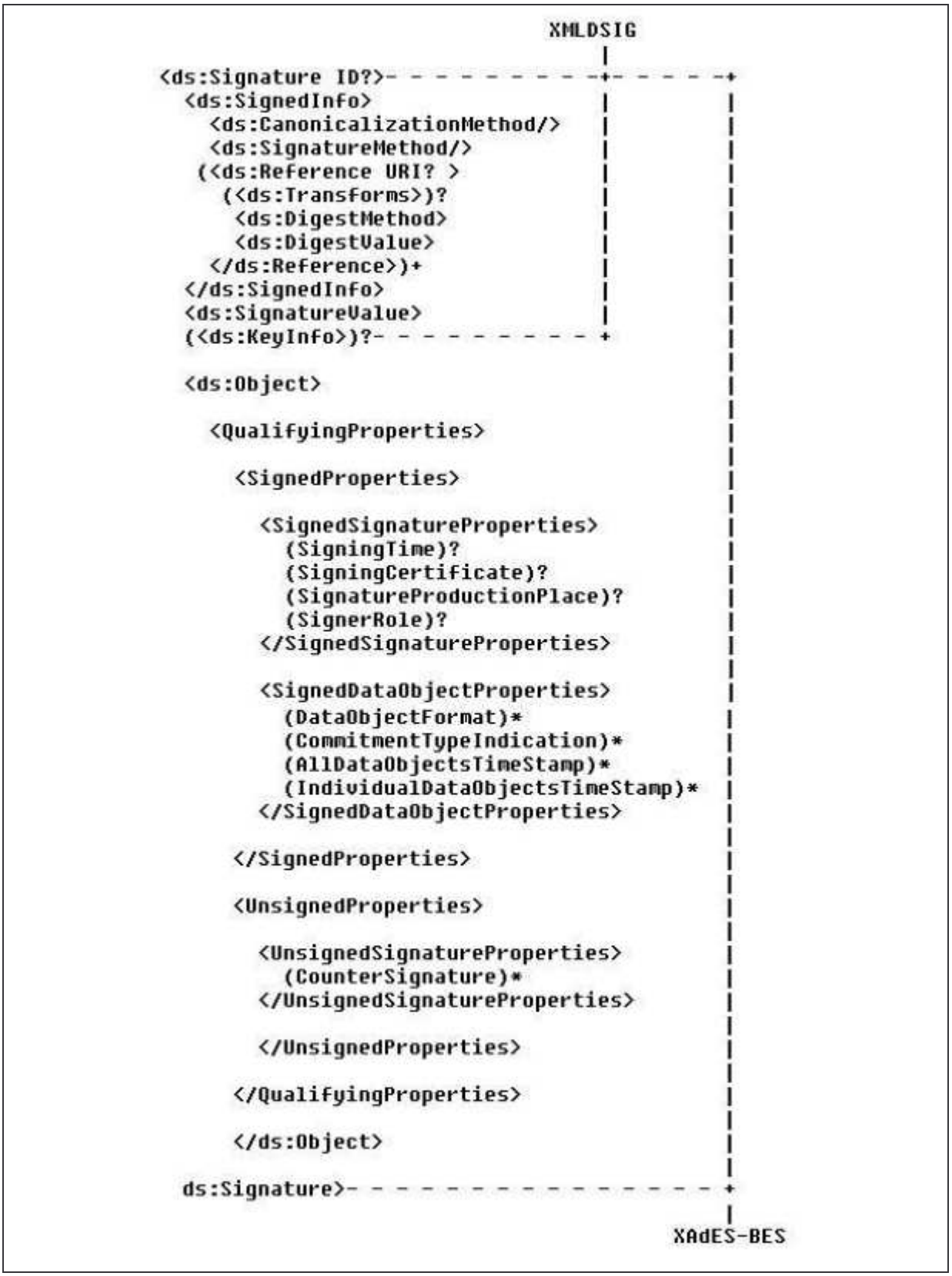

Figura 13 - Formato XAdES – Assinatura Eletrônica Básica
Fonte: ETSI/XAdES, 2004

5.2. Formato XAdES-EPES

No formato XAdES-EPES (Figura 15), para incluir as regras da Política de Assinatura adotada pelas partes que assinarão e farão verificação do documento eletrônico, foi definido o elemento `SignaturePolicyIdentifier`, sob o qual foram definidos elementos para conter informações da Política de Assinatura adotada, ou indicar que a mesma é implícita, e que deve ser atendida tanto para a criação quanto para a verificação das assinaturas do documento assinado com esse padrão. A Política de Assinatura utilizada de forma explícita deve estar acessível de forma programática, pois será necessário tanto para gerar a assinatura quanto para verificá-la.

O Schema XML que define esse elemento está apresentado na figura 14.

```
<xsd:element name="SignaturePolicyIdentifier"
 type="SignaturePolicyIdentifierType"/>
  <xsd:complexType name="SignaturePolicyIdentifierType">
    <xsd:choice>
      <xsd:element name="SignaturePolicyId" type="SignaturePolicyIdType"/>
      <xsd:element name="SignaturePolicyImplied"/>
    </xsd:choice>
  </xsd:complexType>
  <xsd:complexType name="SignaturePolicyIdType">
    <xsd:sequence>
      <xsd:element name="SigPolicyId" type="ObjectIdentifierType"/>
      <xsd:element ref="ds:Transforms" minOccurs="0"/>
      <xsd:element name="SigPolicyHash" type="DigestAlgAndValueType"/>
      <xsd:element name="SigPolicyQualifiers"
       type="SigPolicyQualifiersListType" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="SigPolicyQualifiersListType">
    <xsd:sequence>
      <xsd:element name="SigPolicyQualifier" type="AnyType"
       maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>
```

Figura 14 – Schema do elemento SignaturePolicyIdentifier
Fonte: ETSI/XAdES, 2004

O elemento **SignaturePolicyId** irá aparecer quando a Política de Assinatura a ser utilizada for explícita e deve identificá-la de forma única, sem ambigüidade.

O elemento **SigPolicyId** contém um identificador que indica de forma única uma versão específica da política de assinatura utilizada.

O elemento **SigPolicyHash** contém o identificador do algoritmo de hash e o valor de hash da política de assinatura.

O elemento **SigPolicyQualifier** pode conter informação adicional qualificando o identificador da política de assinatura.

O elemento opcional **ds:Transforms** pode conter as transformações efetuadas no documento da política de assinatura antes do cálculo do seu valor de hash.

Alternativamente, o elemento vazio **SignaturePolicyImplied** irá aparecer quando a política de assinatura a ser utilizada estiver implícita em função do tipo de objeto sendo assinado ou alguma outra informação externa.

Como se pode ver na Figura 15, o formato XAdES-EPES difere-se do formato XAdES-BES apenas pela inclusão do elemento `SignaturePolicyIdentifier` e o tratamento associado ao mesmo.

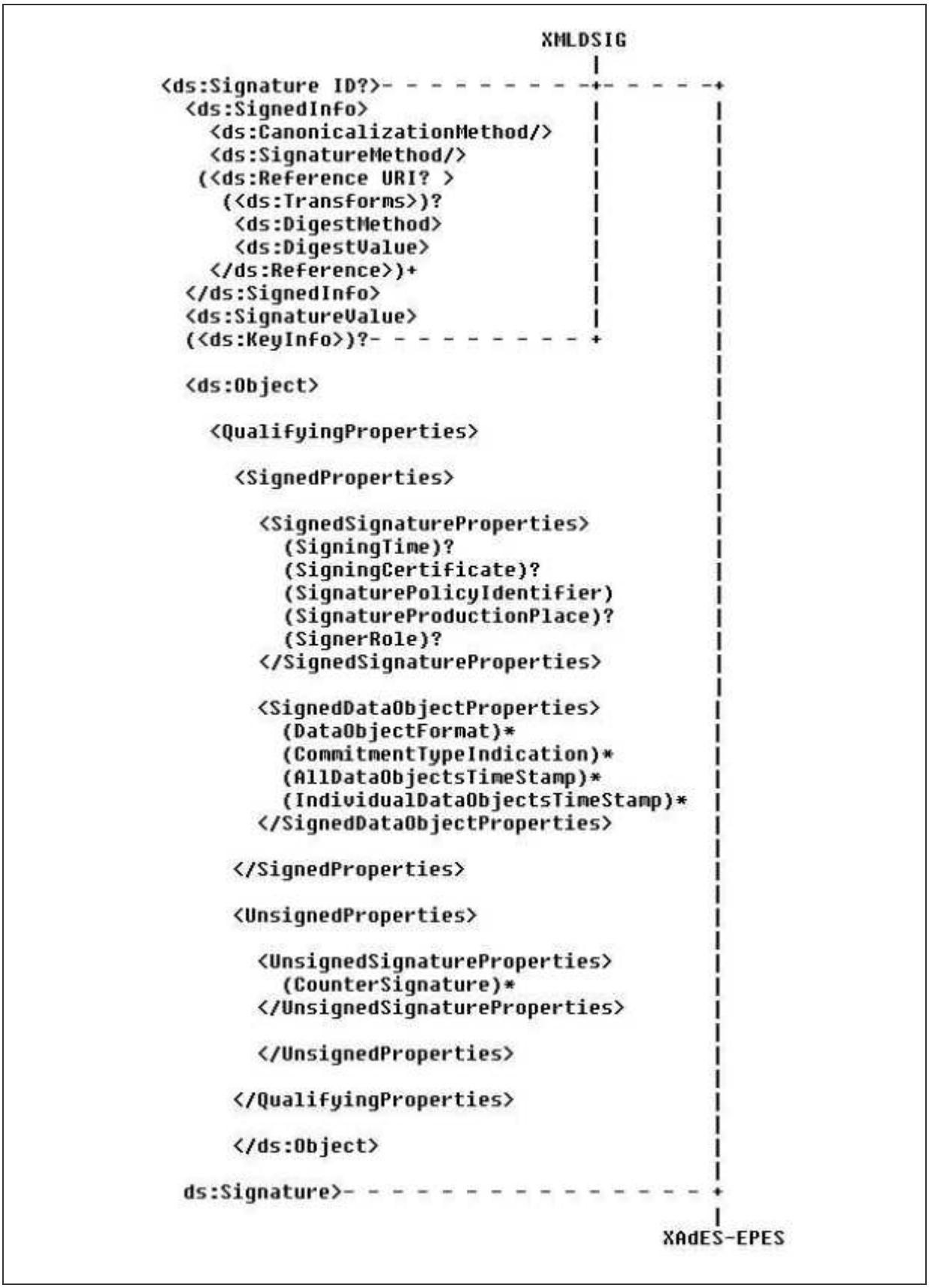

Figura 15 - Formato XAdES - Politica Explicita de Assinatura
Fonte: ETSI/XAdES, 2004

5.3. Formato XAdES-T

O formato XAdES-T pode ser formado a partir do XAdES-BES ou XAdES-EPES com a inclusão do elemento `SignatureTimeStamp` (veja Figura 16).

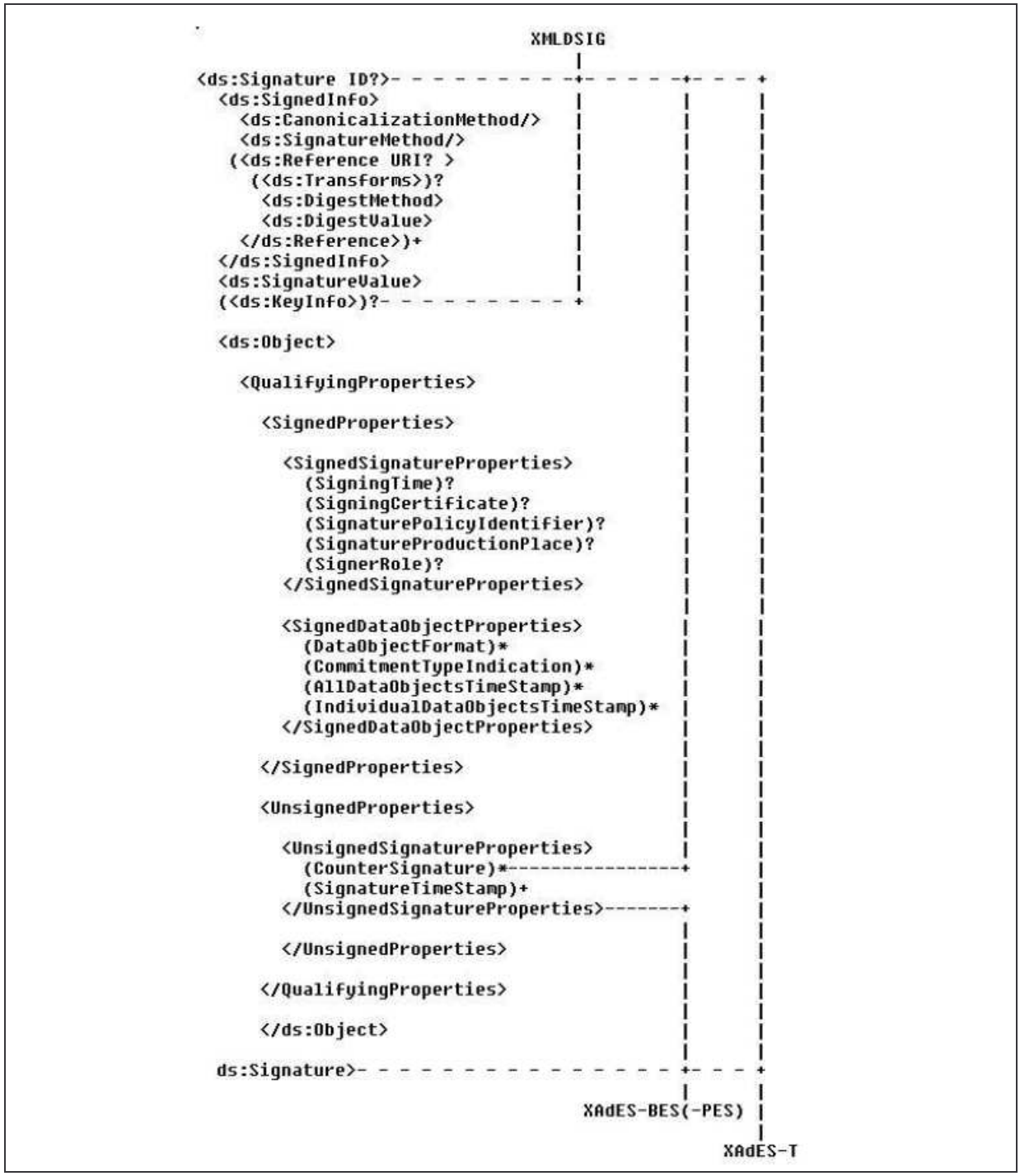

Figura 16 - Formato XAdES – Com Validação de Tempo
Fonte: ETSI/XAdES, 2004

O elemento **`SignatureTimeStamp`** encapsula a Estampilha Temporal sobre o elemento `ds:SignatureValue`. Portanto, este elemento é o fator fundamental para se garantir que o certificado está sendo utilizado dentro do seu prazo de validade e não permite que o mesmo seja utilizado com uma data/hora retroativa (não mais do que permitir a política de assinatura adotada).

O elemento `SignatureTimeStamp` é uma propriedade não assinada qualificando a assinatura. O formato XAdES-T pode conter diversos elementos `SignatureTimeStamp`, obtidos de diferentes Autoridades de Estampilha Temporal.

A política de assinatura pode, por exemplo, especificar a diferença de tempo máxima aceitável que é permitida entre o tempo indicado no elemento `SigningTime` e o tempo indicado no elemento `SignatureTimeStamp`. Se esta diferença é excedida então a assinatura deve ser considerada inválida.

5.4. Formato XAdES-C

O formato XAdES-C (veja Figura 17) é formado a partir do XAdES-T com a inclusão dos seguintes elementos: `CompleteCertificateRefs`, `CompleteRevocationRefs`, `AttributeCertificateRefs` e `AttributeRevocationRefs`.

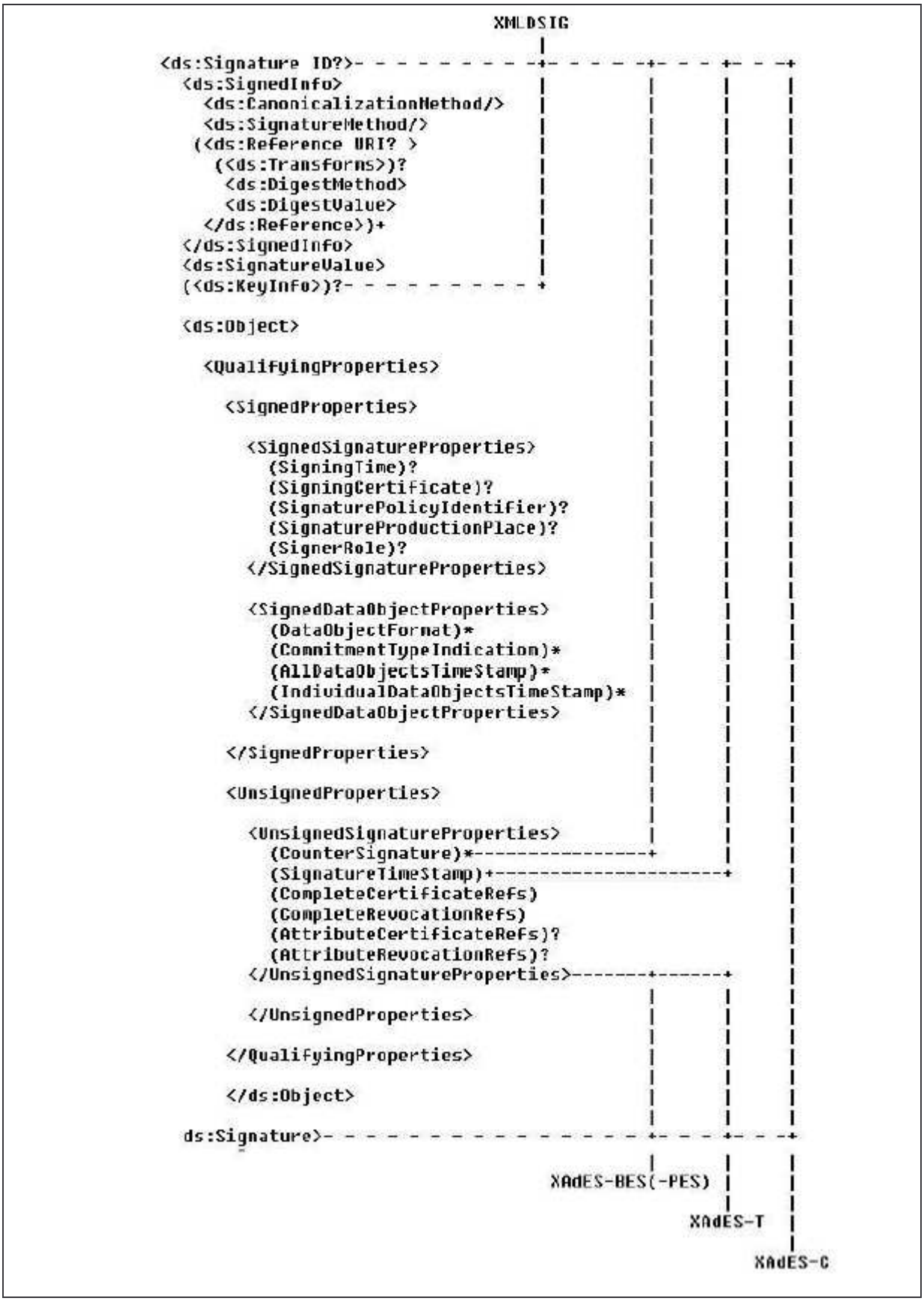

Figura 17 - Formato XAdES Com Validação Completa
Fonte: ETSI/XAdES, 2004

O formato XAdES-C incorpora os seguintes elementos adicionais em relação ao formato XAdES-T, que são requeridos para a validação:

a. **`CompleteCertificateRefs:`** Seqüência das referências ao conjunto completo dos certificados das Autoridades Certificadoras que foram usados para validar a assinatura digital até (mas não incluindo) o certificado do signatário; é um elemento não assinado.
b. **`CompleteRevocationRefs:`** Conjunto completo das referências aos dados da lista de revogação e/ou “OCSP Responses” que foram usados na validação do signatário e de certificados de Autoridades Certificadoras; é um elemento não assinado.
c. **`AttributeCertificateRefs:`** este elemento e o seguinte são necessários quando um Certificado de Atributo (“Attribute Certificate” – (IETF, 2002)) está presente na assinatura; neste caso, esse elemento contém as referências a todos os certificados de “Attribute Authorities” que foram utilizados para validar os Certificados de Atributos.
d. **`AttributeRevocationRefs:`** esse elemento contém referência ao conjunto completo da Lista de Certificados de Atributos Revogados e/ou “OCSP Responses” que foram utilizados na validação dos Certificados de Atributos presentes na assinatura.

Os elementos acima descritos são suficientes para dar a essa especificação, quando utilizando o formato XAdES-C, a garantia da aplicação de uma política de assinatura, que não foi utilizada uma data indevida no passado e que o certificado utilizado não estava revogado no momento da assinatura.

Os elementos abaixo descritos são colocados como uma extensão aos formatos XAdES, não sendo necessário a sua implementação para estar em conformidade com essa especificação. Estes elementos estão relacionados à questão do arquivamento de documentos com assinatura digital por longos períodos de tempo.

- **SigAndRefsTimeStamp**: quando uma resposta de OCSP (Online Certificate Status Protocol) é usada, é necessário obter a Estampilha Temporal, em particular para a resposta no caso de chave de um provedor que possa ser comprometida. Desde que a informação contida na resposta OCSP é específica do usuário e em um momento específico, uma estampilha temporal individual é necessária para toda assinatura recebida. A Estampilha Temporal será solicitada sobre o hash dos seguintes elementos: `ds:Signature` (todas assinaturas presentes), `SignatureTimeStamp`, `CompleteCertificateRefs` (todas as referências aos certificados utilizados), `CompleteRevocationRefs` (todas as referências à Lista de Certificados Revogados e/ou "OCSP responses"), `AttributeCertificateRefs` (se presente) e `AttributeRevocationRefs` (se presente). Pelo mesmo custo de criptografia, isto proverá um mecanismo de integridade sobre a assinatura eletrônica. Qualquer modificação pode ser imediatamente detectada. Pode-se notar que outros meios de proteção/detecção da integridade da assinatura eletrônica existem e podem ser utilizados. Este é um elemento não assinado que qualifica a assinatura.

- **`RefsOnlyTimeStamp`**: Obter Estampilha Temporal para cada Assinatura Eletrônica com validação completa de dados tal como definido no elemento `SigAndRefsTimeStamp` pode não ser eficiente, particularmente quando o mesmo conjunto de certificados de uma Autoridade Certificadora e informações de

Lista de Certificados Revogados são utilizados para validar muitas assinaturas. A emissão de Estampilha Temporal de certificados e LCRs comumente utilizados pode ser feita centralizadamente, por exemplo, dentro de uma companhia ou por um provedor de serviço. Este método reduz a quantidade de Estampilha Temporal para o verificador emitir ou verificar. O valor de "hash" enviado para a AET será calculado sobre a concatenação dos elementos `CompleteCertificateRefs` e `CompleteRevocationRefs`. O elemento `RefsOnlyTimeStamp` é um qualificador não assinado. Este elemento não estará presente se o elemento `SigAndRefsTimeStamp` estiver, e vice-versa.

- **`CertificateValues:`** Ao tratar das assinaturas eletrônicas de longo prazo, todos os dados usados na verificação (incluindo a cadeia de certificação) devem ser convenientemente arquivados. A princípio, o elemento `CertificateValues` contém o todos os certificados que foram usados para validar a assinatura, incluindo o certificado do signatário. Entretanto, não é necessário incluir os certificados que já estiverem no elemento de `ds:KeyInfo` da assinatura.

- **`RevocationValues:`** todos os dados da Lista de Certificados Revogados e/ou OCSP Responses serão colocados neste elemento, incluindo a LCR e/ou OCSP do certificado contido nas Estampilhas Temporais (se já não estiverem presentes).

- **`ArchiveTimeStamp:`** Os avanços na computação aumentam a probabilidade de ser viável quebrar algoritmos e comprometer chaves atualmente consideradas seguras. Existe assim uma exigência de ser possível proteger assinaturas eletrônicas contra essa possibilidade. Passado um período de tempo, fraquezas podem ocorrer em algoritmos criptográficos usados para criar uma assinatura eletrônica. Antes que tais fraquezas se tornem comum, um verificador

tomaria medidas extras para manter a validade da assinatura eletrônica. Diversas técnicas poderiam ser utilizadas para atingir este objetivo, dependendo da natureza da criptografia enfraquecida. A fim de simplificar o problema, uma técnica simples, chamada “Archive validation data”, cobrindo todos os casos é apresentada na especificação proposta. “Archive validation data” consiste na obtenção de Estampilha Temporal para os dados completos de validação e dos certificados e da lista de revogação junto com a assinatura eletrônica. O recurso “Archive validation data” é necessário se a função de “hash” e os algoritmos de criptografia que foram usados para criar a assinatura não mais são seguros. O possibilidade de a chave da Autoridade Certificadora ser comprometida é significativamente menor que a do usuário, porque é esperado que as Autoridades Certificadoras utilizem criptografia forte e tenham melhor proteção da chave. É esperado que novos algoritmos (ou mesmos os atuais com chaves de tamanhos maiores) irão ser utilizados. Em tais casos, a seqüência de estampilha temporal protegerá contra a fraude. Cada Estampilha Temporal precisa ser obtida antes ou do comprometimento da chave de assinatura ou da quebra do algoritmo usado pela Autoridade de Estampilha Temporal. O elemento `ArchiveTimeStamp` é uma propriedade não assinada que qualifica a assinatura.

6. RESPOSTA AOS DESAFIOS PROPOSTOS

Para atender aos desafios colocados em relação à assinatura digital, no que diz respeito à política de assinatura e à realização de assinatura com uma data no passado, a proposta do ETSI, se utilizada adequadamente, pode ser a resposta procurada.

O formato XAdES-T, incorporando o XAdES-EPES, cobre os requisitos para conter informações sobre a Política de Assinatura e garante, com as Estampilhas Temporais, que as informações utilizadas para a Assinatura Digital (tais como certificados) eram válidas até aquele momento e não se está utilizando uma data/hora retroativa indevida.

Para a questão da longevidade de um documento assinado digitalmente, deve-se utilizar o formato XAdES-C, podendo-se até incluir os elementos definidos como extensão desse formato (`SigAndRefsTimeStamp`, `RefsOnlyTimeStamp`, `CertificateValues`, `RevocationValues` e `ArchiveTimeStamp`), conforme descrito no item anterior, dando-se ainda maior garantia para os documentos arquivados e que poderiam ser revalidados quando necessário.

Desta forma, a especificação proposta pelo ETSI procura cobrir todos aspectos que podem tornar a assinatura digital confiável tanto para a utilização de curto prazo quanto para a que exige maior longevidade. Mas, a mesma ainda é apenas uma especificação. Como estarão as implementações para essa especificação? No próximo item é feito um breve posicionamento a respeito.

6.1. Implementação da especificação XAdES

Com o objetivo de estimular e ajudar na implementação da especificação XAdES, o ETSI, em 2003 e 2004, patrocinou dois eventos, nos quais foram reunidos desenvolvedores que estavam implementando sistemas que utilizam a especificação proposta, visando dar apoio para criar implementações interoperáveis e obter uma realimentação dos mesmos para o processo de manutenção e futuras versões dessa especificação.

O primeiro evento, “ETSI XAdES PLUGTESTS Interoperability” (2003), ocorreu no período de 3 a 7 de novembro de 2003, em Sophia Antipolis, França. Os participantes desse primeiro evento foram:

- Baltimore Technologies – dos Estados Unidos
- Institute for Applied Information Processing and Communications (IAIK), Graz University of Technology – da Alemanha
- Kopint–Datorg Rt. (Kopdat) – da Hungria
- Microsoft – dos Estados Unidos
- AS Sertifitseerimiskeskus (SK) – da Estônia
- Universitat Politecnica de Catalunya (UPC) – da Espanha

Nesse evento foi possível detectar algumas modificações que seriam necessárias na especificação vigente (de fevereiro de 2002), por exemplo, para tornar factível as implementações referentes às verificações de “time-stamps” (estampilha temporal).

Dois desenvolvedores apresentaram maior compatibilidade com a especificação, apesar de nenhum dos dois estar totalmente compatível: Baltimore e IAK.

Também os desenvolvedores presentes tiveram um retorno positivo no sentido de esclarecer certos detalhes e ter melhor entendimento das especificações, que permitirá uma implementação mais aderente.

Um segundo evento desse mesmo tipo foi realizado no período de 24 a 28 de maio de 2004 em Sophia Antipolis, França, chamado de “Security PlugTests Event” (2004). O objetivo deste segundo evento incluía também testes com a tecnologia de Infraestrutura de Chave Pública.

Neste segundo evento, em relação à implementação da especificação XAdES, foram testadas dos seguintes desenvolvedores:

- CDC Mercure
- SEB IT Partner – da Estônia
- Microsoft – dos Estados Unidos
- BeTrusted – dos Estados Unidos
- Universitat Politecnica de Catalunya (UPC) – da Espanha
- Institute for Applied Information Processing and Communications (IAIK), Graz University of Technology – da Alemanha

O fato de este evento ter sido realizado muito próximo à divulgação da versão 1.2.2 da especificação, que ocorreu em abril de 2004, não permitiu que as implementações demonstradas já estivessem aderentes a essa versão. Desta forma, todas elas ainda refletiam a especificação 1.1.1, porém as que já tinham participado do primeiro evento puderem mostrar que tiveram progressos na implementação eliminando algumas falhas anteriormente verificadas.

Nesta data, dezembro de 2004, encontram-se disponíveis as seguintes implementações do padrão XML de Assinatura Digital:

a) Incorporando a especificação XAdES:

- IAIK XML Security Library (IXSIL), versão 2.0 em beta teste, do "Institute for Applied Information Processing and Communication" (IAIK, http://jce.iaik.tugraz.at/). É o único software comercial até dezembro de 2004 que implementa a especificação XadES do ETSI.
- DigiDoc Systems, versão 1.1.06, é uma biblioteca gratuita para desenvolvimento de sistemas utilizando XML Signature, que atende ao padrão XAdES, disponível em www.openxades.org, cujo projeto é mantido por duas entidades certificadoras, uma da Estônia (AS Sertifitseerimiskeskus) e outra da Finlândia (Väestörekisterikeskus);

b) XML Signature sem incorporar a especificação XAdES:

Abaixo são listados alguns dos muitos produtos que implementam assinatura digital em XML, que, no entanto, até 12 de dezembro de 2004, quando foi concluída esta pesquisa, ainda não incorporavam as recomendações do ETSI (padrão XAdES):

- XML Security Suíte, da IBM;
- VeriSign's Trust Services Integration Kit, da VeriSign
- Signed and Store Solution, da BeTrusted, pertencente ao grupo Bank One, dos Estados Unidos;
- Microsoft tem bibliotecas para desenvolvimento de XML Signature na arquitetura dotNET e anterior (como VB 6);

- XML Security Library, de Aleksey Sanin e outros, biblioteca gratuita, disponível no endereço http://www.aleksey.com.
- Tamino XML Server, da Software AG.
- Exchanger XML Editor, da Cladonia Ltd., da Irlanda.
- InfoMosaic Secure XML Digital Signature, da InfoMosaic Corporation, dos Estados Unidos.
- Java Web Services Developers Pack, da Sun Microsystems – pacote de softwares livres que inclui APIs para o desenvolvimento de aplicativos que tratem Assinaturas Digitais em XML.
- eXSign Software Development Kit, da Security Technology Competence Centre, da Eslovênia.

7. CONCLUSÃO

Com a crescente utilização da Internet e do intercâmbio de informações entre empresas e a preocupação com a segurança da informação, faz-se antever que a assinatura digital tende a ser um recurso cada vez mais utilizado no comércio eletrônico.

Duas grandes preocupações em relação ao uso da assinatura digital dizem respeito aos diversos contextos em que uma assinatura digital pode ser utilizada (quando se pensa em tê-la como substituta da assinatura manuscrita) e, também, a uma forma de garantir que a assinatura digital não está sendo gerada com uma data no passado e que o certificado digital utilizado para gerá-la era válido naquele momento.

Tendo em vista que o uso crescente da Internet no comércio eletrônico e que o formato XML está se tornando um padrão de fato para o intercâmbio de informações entre aplicações, em especial de plataformas diferentes, o padrão XML Signature poderá vir muito utilizado no serviço de assinatura digital. Desta forma, o padrão XAdES proposto pelo ETSI sobre a XML Signature poderá vir a ser um padrão de referência para atender a todos os requisitos de segurança e garantia de longevidade para um documento assinado eletronicamente.

Com as diretrizes traçadas pelo Comitê Europeu para o comércio eletrônico (European Parliament and The Council of European Union, 1999), espera-se que haja um investimento maior das empresas desses países na adoção da assinatura digital nos vários segmentos, tendo à frente a padronização do ETSI, que atende aos requisitos das diretrizes traçadas e dá maior segurança aos processos, podendo atender inclusive aos desafios atuais para a assinatura digital.

No Brasil, atualmente, a legislação sobre o assunto tem como base principal a Medida Provisória nº 2200-2, de 24 de agosto de 2001, que “institui a Infra-Estrutura de Chaves Públicas Brasileira - ICP-Brasil, transforma o Instituto Nacional de Tecnologia da Informação em autarquia, e dá outras providências”.

O parágrafo primeiro do artigo 10 dessa Medida Provisória estabelece o seguinte: “As declarações constantes dos documentos em forma eletrônica produzidos com a utilização de processo de certificação disponibilizado pela ICP-Brasil presumem-se verdadeiros em relação aos signatários, na forma do art. 131 da Lei nº 3.071, de 01 de janeiro de 1916 - Código Civil.”

Há um esforço também no sentido de se aprovar uma legislação mais específica sobre o comércio eletrônico, impulsionando o uso da assinatura digital. Atualmente, no Congresso Nacional, está em andamento o Projeto de Lei nº 4.906, de 21 de junho de 2001, de autoria do Senador Lúcio Alcântara, que “dispõe sobre o comércio eletrônico”.

Apensos a este Projeto de Lei estão outros relacionados ao mesmo assunto, que são:

a) Projeto de Lei nº 1483/1999, do Deputado Hélio de Oliveira Santos, que “institui a fatura eletrônica e a assinatura digital nas transações de "comércio" eletrônico”;
b) Projeto de Lei nº 6965/2002, do Deputado José Carlos Coutinho, que “confere valor jurídico à digitalização de documentos, e dá outras providências”;
c) Projeto de Lei nº 7093/2002, do Deputado José Ivan de Carvalho Paixão, que “dispõe sobre correspondência comercial e dá outras providências”.

Além disso, temos atualmente estabelecidas algumas regulamentações específicas como, por exemplo, no caso de Contratos de Câmbio, cuja utilização no formato eletrônico, com assinatura digital, foi autorizada na Carta Circular nº 3.134, de 27 de abril de 2004, do Banco Central. Neste documento ficou estabelecido que o padrão seria PKCS#7, utilizando chaves no padrão PKCS#1 e algoritmo de hashing SHA-1. Conforme o item IV dessa Carta, "a data-hora da efetivação deve ser controlada pela contratante" e ficará registrada no atributo "signingTime" do campo "authenticatedAttributes" da estrutura "SignedInfo".

Com relação às "Time Stamp Authorities", ou Autoridades de Estampilha Digital, ou Autoridades Datadoras (como já empregadas por algumas entidades), no Brasil, o órgão oficial por gerar a Hora Legal e disseminá-la pelos meios de comunicação é o Observatório Nacional, conforme Decreto nº 4.264, de 10 de junho de 2002. Uma das formas de disseminação dessa informação é através do fornecimento do serviço de "Carimbo de Tempo", referido neste trabalho como Estampilha Temporal.

Vê-se assim que, apesar da tecnologia disponível, no Brasil não há ainda uma determinação oficial quanto ao uso desse recurso para certificação da data-hora de determinados eventos do processo de assinatura digital.

Apesar de insipiente, o uso de assinatura digital agregado a documentos deve crescer ano a ano. Um projeto que deve incentivar em muito o uso de assinatura digital é o da Secretaria da Receita Federal (SRF), que implantou o e-CPF e o e-CNPJ (certificados digitais em cartões inteligentes). O e-CPF é voltado para a pessoa física e o e-CNPJ para as empresas. Uma série de serviços está sendo aos poucos disponibilizada para os portadores desse certificado digital no "site" da Receita Federal. Ainda em janeiro de 2005, foi firmado acordo entre o ITI, a

FEBRABAN e a SRF para incentivar o uso desses certificados nas transações bancárias. Além disso, alguns grandes bancos estão sendo homologados como Autoridade de Registro (AR), o que deve aumentar a base de usuários com certificados digitais.

Assim, termos uma solução para os desafios aqui colocados é muito importante e tem que ser agregado o quanto antes, dando maior segurança e credibilidade a todo processo.

8. REFERÊNCIAS

BANCO CENTRAL DO BRASIL, **Carta Circular nº 3134**, de 27 de abril de 2004.

CHASE, Nicholas. **XML Digital Signatures,** 18 de agosto de 2004. Disponível em http://www.informit.com/guides/content.asp?g=xml&seqNum=142. Acesso em: 12 dez. 2004.

ETSI, **Security Plugtests Event – PKI/XAdES**, 29 de junho de 2004. Disponível em: <http://www.etsi.org>. Acesso em: 12 dez. 2004.

____, **XML Advanced Electronic Signatures (XAdES)**, abril de 2004. Disponível em: <http://www.etsi.org>. Acesso em: 12 dez. 2004.

____, **ETSI XADES PlugTests – Final Report**, novembro de 2003. Disponível em: <http://www.etsi.org>. Acesso em: 12 dez. 2004.

____, **Signature policy for extended business model**, março de 2003. Disponível em: <http://www.etsi.org>. Acesso em: 12 dez. 2004.

____, **XML format for signature policy**, abril de 2002. Disponível em: <http://www.etsi.org>. Acesso em: 12 dez. 2004.

____, **XML Advanced Electronic Signatures (XAdES)**, fevereiro de 2002. Disponível em: <http://www.etsi.org>. Acesso em: 12 dez. 2004.

EUROPEAN PARLIAMENT AND THE COUNCIL OF EUROPEAN UNION, **Directive 1999/93/EC**, 13 de dezembro de 1999.

GOKUL, Seshadri. **XML Digital Signatures**, 30 de agosto de 2002. Disponível em: <http://www.informit.com/articles/article.asp?p=29032>. Acesso em: 12 dez. 2004.

IETF, **RFC-3161**, Internet X.509 Public Key Infrastructure Time-Stamp Protocol (TSP), de agosto de 2001. Disponível em: <http://www.faqs.org/rfcs/rfc3161.html>. Acesso em: 12 dez. 2004. – 3369 2630

____, **RFC-3852**, Cryptographic Message Syntax, de julho de 2004. Disponível em: <http://www.faqs.org/rfcs/rfc3852.html>. Acesso em: 12 dez. 2004.

____, **RFC-3369,** Cryptographic Message Syntax, de agosto de 2002. Disponível em: <http://www.faqs.org/rfcs/rfc3369.html>. Acesso em: 12 dez. 2004.

____, **RFC-3281,** An Internet Attribute Certificate Profile for Authorization, de abril de 2002. Disponível em: <http://www.faqs.org/rfcs/rfc3281.html>. Acesso em: 12 dez. 2004.

____, **RFC-3275,** XML-Signature Syntax and Processing, de março de 2002. Disponível em: <http://www.faqs.org/rfcs/rfc3275.html>. Acesso em: 12 dez. 2004.

____, **RFC-2360,** Cryptographic Message Syntax, de junho de 1999. Disponível em: <http://www.faqs.org/rfcs/rfc2630.html>. Acesso em: 12 dez. 2004.

ISO, ISO-8879 Standard Generalized Markup Language (SGML), 1986.

NAKOV, Svetlin. **How Digital Signatures Work: Digitally Signing Messages**. Disponível em: <http://www.developer.com/java/ent/article.php/3092771>. Acesso em: 12 dez. 2004.

SOUSA, Artur Afonso. **Base de Dados, Web e XML**. Lisboa: FCA Editora de Informática, 2002.

W3C, **Extensible Markup Language (XML) 1.0,** 04 de fevereiro de 2004. Disponível em: <http://www.w3.org/TR/2004/REC-xml-20040204/>. Acesso em: 12 dez. 2004.

___, **Extensible Stylesheet Language (XSL) Version 1.1**, 17 de dezembro de 2003. Disponível em: <http://www.w3.org/TR/2003/WD-xsl11-20031217/>. Acesso em: 12 dez. 2004.

___, **XML Signature Syntax and Processing**, 12 de fevereiro de 2002. Disponível em: <http://www.w3.org/TR/2002/REC-xmldsig-core-20020212/>. Acesso em: 12 dez. 2004.

___, **HTML 4.01 Specification**, 24 de dezembro de 1999. Disponível em: <http://www.w3.org/TR/html401/>. Acesso em: 12 dez. 2004.

GLOSSÁRIO

AET: acrônimo para Autoridade de Estampilha Temporal, tradução utilizada para o acrônimo, em inglês, TSA, de Time Stamp Authority. Entidade responsável pela geração de um protocolo digital garantindo a data e hora real em que um certo evento ocorreu;.

API: acrônimo para Application Program Interface, trata-se de um mecanismo de programação que permite a interação de um programa com o sistema; a sua forma varia de acordo com a linguagem de programação utilizada.

CMS: acrônimo para "Cryptographic Message Syntax Standard", padrão de criptografia desenvolvido originalmente pela empresa RSA Security Inc..

DSA: acrônimo para Digital Signature Algorithm.

ECDSA: acrônimo para Elliptic Curve Digital Signature Algorithm.

ETSI: acrônimo para European Telecommunications Standards Institute, é um instituto sem fins lucrativos cuja responsabilidade é elaborar padrões de tecnologia da informação e telecomunicação para a comunidade européia.

FEBRABAN: acrônimo para Federação Brasileira dos Bancos

HASH: valor com um determinado número de dígitos calculado por um algoritmo específico a partir de um valor de entrada; o algoritmo deve ser tal que o valor resultante seja diferente para cada valor de entrada.

ITI: acrônimo para Instituto Nacional de Tecnologia da Informação

MD-2, MD-4 e MD-5: são algoritmos para cálculo de hash desenvolvidos por Ronald L. Rivest em 1989, 1990 e 1991, respectivamente.

OCSP: acrônimo para "Online Certificate Status Protocol", protocolo utilizado para se verificar em tempo real o status de um certificado na Lista de Certificados Revogados.

RSA: acrônimo para “Rivest, Shamir e Adleman”, nomes dos matemáticos responsáveis pela criação do algoritmo de criptografia assimétrica que ficou conhecido pela sigla RSA.

SGML: acrônimo para Standard Generalized Markup Language.

SHA: acrônimo para Secure Hash Algorthm, algoritmo para cálculo de hash.

SRF: acrônimo para Secretaria da Receita Federal

TSA: acrônimo para Time Stamp Authority (veja AET).

URI: acrônimo para Uniform Resource Identifiers.

W3C: acrônimo para World Wide Web Consortium, entidade que tem como um dos seus principais objetivos estabelecer padrões que permitam maior interoperabilidade através da Internet; conta atualmente com mais de 350 membros, entre empresas e universidades de todo mundo.

XAdES: acrônimo para XML Advanced Electronic Signature.

XML: acrônimo para “Extensible Markup Language”.

XML Dsig: abreviatura para XML Digital Signature.