

# XML SIGNATURE

Eric Déchaux  
DESS IDM – Évry Val d'Essonne  
[eric.thierry@dechaux.nom.fr](mailto:eric.thierry@dechaux.nom.fr)  
Présentation Groupe PIN – 22 mars 2005

## Définition

- La mission de ce groupe de travail est de développer une **syntaxe** XML utilisée afin de **représenter** des signatures pour des **contenus numériques** ainsi que des **procédures** pour les calculer et les vérifier. Les signatures doivent garantir :
  - l'intégrité des données,
  - l'authentification et/ou
  - la non répudiation.

## Présentation

- I. Principes de conception
- II. Prérequis
- III. Syntaxe
- IV. Extensions
- V. Implantations
- VI. Derniers mots
- VII. Références

3

## Présentation

- I. Principes de conception
- II. Prérequis
- III. Syntaxe
- IV. Extensions
- V. Implantations
- VI. Derniers mots
- VII. Références

4

## I. Principes de conception

1. Principes et étendue
2. Modèle de données
3. Format
4. Cryptographie et traitements

5

## 1. Principes et étendue

- Règle 1
  - Permettre de signer un contenu **numérique** et, en particulier, un contenu **XML**.
- Règle 2
  - Générée à partir d'un **condensat** sur la forme **canonique** du « **manifest** ».

6

## 2. Modèle de données

- Règle 1
  - **Basée** sur le modèle de données de **RDF** sans suivre sa syntaxe de sérialisation.
- Règle 2
  - Applicable sur une **partie ou la totalité** d'un document XML.

7

## 3. Format

- Règle 1
  - C'est un **élément** XML.
- Règle 2
  - L'ajout d'une signature à un document :
    - L'élément racine **doit rester** la racine,
    - Elle doit être placée à **l'endroit permis** par le modèle de contenu.

8

## 4. Cryptographie et traitement

- Règle 1
  - **N'importe quel** algorithme de signature, d'authentification, ou d'approbation de clé doit être **utilisable**.
  
- Règle 2
  - **Une méthode obligatoire** à implanter doit être spécifiée pour réaliser la canonisation, le condensat et la signature.

9

## Présentation

- I. Principes de conception
- II. Prérequis
- III. Syntaxe
- IV. Extensions
- V. Implantations
- VI. Derniers mots
- VII. Références

10

## II. Prérequis

- Règle 2 des principes et étendue :
  - « générées à partir d'un **condensat** sur la forme **canonique** [...]».
- 1. Le condensat
- 2. La forme canonique

11

## 1. Le condensat - 1

- **Résultat** d'une fonction appelée fonction de **hachage**.
- Convertit un ensemble en un plus petit.
- Propriétés suivantes :
  - $H(x) \neq H(y)$  implique  $x \neq y$
  - $H(x) = H(y)$  implique  $x = y$

12

## 1. Le condensat - 2

- Intérêt :
  - Signature sur des données plus petites.
  - Vérifier les données signées.
  - Pas réversible.
- MD5 et SHA-1 et dérivés

13

## 2. La forme canonique - 1

- Deux éléments XML :
  - `<elem id='0' xml:lang='fr'>`
  - `<elem xml:lang='fr' id='0'>`
- Éléments équivalents mais différents.
- Condensats différents :
  - `dd00bd0a32e382e508112ac869f2ddad`
  - `d0a3f919ca76b518705fc461331fb5a0`
- Signatures différentes.

14

## 2. La forme canonique - 2

- Canonisation : méthode de standardisation d'un arbre XML.
- Deux méthodes :
  - XML Canonicalization,
  - XML Exclusive Canonicalization.
- Même objectif, la seconde méthode ne fait pas hériter les nœuds fils des informations des nœuds pères.

15

## Présentation

- I. Principes de conception
- II. Prérequis
- III. Syntaxe
- IV. Extensions
- V. Implantations
- VI. Derniers mots
- VII. Références

16



## III. Syntaxe

1. Structure
2. Formes de signature
3. Règles de traitement
4. Génération
5. Validation

17

## 1. Structure - 1

```
→ <Signature>  
    <SignedInfo>  
        <CanonicalizationMethod />  
        <SignatureMethod />  
        <Reference />  
    </SignedInfo>  
    <SignatureValue />  
    <KeyInfo />  
    <Object />  
→ </Signature>
```

18

## 1. Structure - 2

```
<Signature>  
  → <SignedInfo>  
    <CanonicalizationMethod />  
    <SignatureMethod />  
    <Reference />  
  → </SignedInfo>  
    <SignatureValue />  
    <KeyInfo />  
    <Object />  
</Signature>
```


19

## 1. Structure - 3

```
<Signature>  
  <SignedInfo>  
    → <CanonicalizationMethod />  
    <SignatureMethod />  
    <Reference />  
  </SignedInfo>  
  <SignatureValue />  
  <KeyInfo />  
  <Object />  
</Signature>
```


20

## 1. Structure - 4

```
<Signature>  
  <SignedInfo>  
    <CanonicalizationMethod />  
     <SignatureMethod />  
    <Reference />  
  </SignedInfo>  
  <SignatureValue />  
  <KeyInfo />  
  <Object />  
</Signature>
```


21

## 1. Structure - 5

```
<Signature>  
  <SignedInfo>  
    <CanonicalizationMethod />  
    <SignatureMethod />  
     <Reference />  
  </SignedInfo>  
  <SignatureValue />  
  <KeyInfo />  
  <Object />  
</Signature>
```


22

## 1. Structure - 6

```
<Signature>
  <SignedInfo>
    <Reference>
       <Transforms />
      <DigestMethod />
      <DigestValue />
    </Reference>
  </SignedInfo>
  <SignatureValue />
  <KeyInfo />
  <Object />
</Signature>
```


23

## 1. Structure - 7

```
<Signature>
  <SignedInfo>
    <Reference>
      <Transforms />
       <DigestMethod />
      <DigestValue />
    </Reference>
  </SignedInfo>
  <SignatureValue />
  <KeyInfo />
  <Object />
</Signature>
```


24

## 1. Structure - 8

```
<Signature>  
  <SignedInfo>  
    <Reference>  
      <Transforms />  
      <DigestMethod />  
       <DigestValue />  
    </Reference>  
  </SignedInfo>  
  <SignatureValue />  
  <KeyInfo />  
  <Object />  
</Signature>
```

25

## 1. Structure - 9

```
<Signature>  
  <SignedInfo>  
    <CanonicalizationMethod />  
    <SignatureMethod />  
    <Reference />  
  </SignedInfo>  
   <SignatureValue />  
  <KeyInfo />  
  <Object />  
</Signature>
```

26

## 1. Structure - 10

```
<Signature>  
  <SignedInfo>  
    <CanonicalizationMethod />  
    <SignatureMethod />  
    <Reference />  
  </SignedInfo>  
  <SignatureValue />  
  <KeyInfo />  
  <Object />  
</Signature>
```

27

## 1. Structure - 11

```
<Signature>  
  <SignedInfo>  
    <CanonicalizationMethod />  
    <SignatureMethod />  
    <Reference />  
  </SignedInfo>  
  <SignatureValue />  
  <KeyInfo />  
  <Object />  
</Signature>
```

28

## 2. Formes de signature - 1

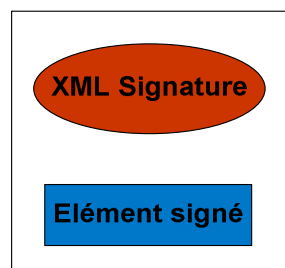
- Trois formes de signature
  - Signature Détachée
  - Signature Enveloppée
  - Signature Enveloppante

29

## 2. Formes de signature - 2

- La signature détachée

```
<Signature>  
...  
  <Reference  
    URI= « http://ex.com/doc.txt » />  
...  
</Signature>
```



30

## 2. Formes de signature - 3

- La Signature enveloppante

<Signature>

...

<Reference URI= « #data » />

<Object id =«data» >

<document>

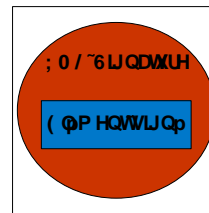
data ...

</document>

</Object>

...

</Signature>



31

## 2. Formes de signature - 4

- La Signature enveloppée

<document>

data ...

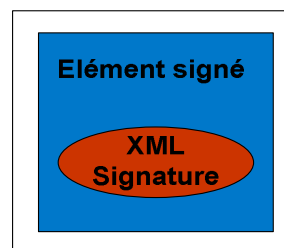
<Signature>

...

<Reference URI= « / » />

</Signature>

</document>



32



### 3. Règles de traitement

- Émetteur
  - Création du message,
  - canonisation,
  - signature.
- Récepteur
  - Réception du message,
  - canonisation,
  - vérification de la signature.

33

### 4. Génération - 1

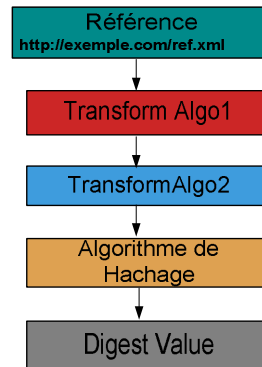
- Deux étapes
  - Génération de la référence
  - Génération de la signature

34

## 4. Génération - 2

### ● Génération de la référence

```
<Reference URI='http://exemple.com/ref.xml'>  
  <Transforms>  
    <Transform Algorithm='Algo1' />  
    <Transform Algorithm='Algo2' />  
  </Transforms>  
  <DigestMethod Algorithm='SHA1' />  
  <DigestValue>...<DigestValue />  
</Reference>
```

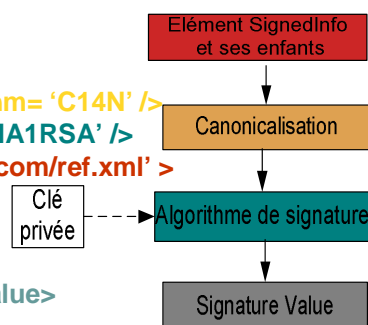


35

## 4. Génération - 3

### ● Génération de la signature

```
<Signature>  
  <SignedInfo>  
    <CanonicalizationMethod Algorithm='C14N' />  
    <SignatureMethod Algorithm='SHA1RSA' />  
    <Reference URI='http://exemple.com/ref.xml'>  
      .....  
    </Reference>  
  </SignedInfo>  
  <SignatureValue /> ... </SignatureValue>  
</Signature>
```



36

## 5. Validation - 1

### ● Validation de la référence

<Reference URI= 'http://exemple.com/ref.xml'>

<Transforms>

<Transform Algorithm= 'Algo1' />

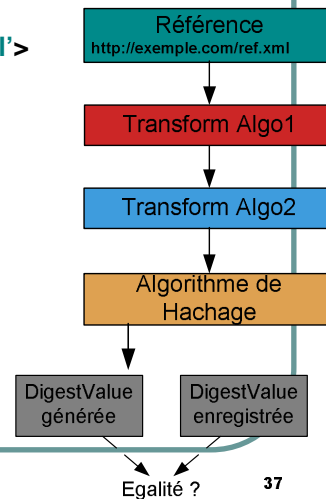
<Transform Algorithm= 'Algo2' />

</Transforms>

<DigestMethod Algorithm = 'SHA1' />

<DigestValue> .... </DigestValue>

</Reference>



37

## 5. Validation - 2

### ● Validation de la signature

<Signature>

<SignedInfo>

<CanonicalizationMethod Algorithm= 'C14N' />

<SignatureMethod Algorithm= 'SHA1RSA' />

<Reference URI = 'http://exemple.com/ref.xml' >

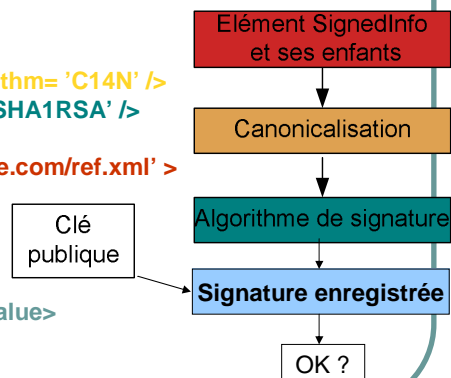
.....

</Reference>

</SignedInfo>

<SignatureValue> ... </SignatureValue>

</Signature>



38

## Présentation

- I. Principes de conception
- II. Prérequis
- III. Syntaxe
- IV. Extensions
- V. Implantations
- VI. Derniers mots
- VII. Références

39

## IV. Extentions

- Proposées par l'IETF.
- Toujours à l'état de draft.
- Ajout d'algorithmes :
  - DigestMethod
  - SignatureMethod
  - TransformMethod
  - EncryptionMethod
  - KeyInfo

40

## Présentation

- I. Principes de conception
- II. Prérequis
- III. Syntaxe
- IV. Extensions
- V. Implantations
- VI. Derniers mots
- VII. Références

41

## V. Implantations

- Libres
  - Apache XML Security Java
  - IBM alphaWorks XML Security Suite
  - C XML Security Library
- Commerciales
  - Baltimore Technologies KeyTools
  - Webgetail Communications Java Crypto and Security Implementation
  - Microsoft SDK .Net

42

## Présentation

- I. Principes de conception
- II. Prérequis
- III. Syntaxe
- IV. Extensions
- V. Implantations
- VI. Derniers mots
- VII. Références

43

## VI. Derniers mots - 1

- Tout à fait capable de rentrer dans le cadre de la pérennisation des données.
- Quelques limites liées au format des fichiers signés.
- Très intéressant utilisées avec des documents XML (contenu) et XSLT (présentation).

44

## IV. Derniers mots - 2

- Exemple d'utilisation

- XKMS, protocole de haut niveau de gestion de clés publiques dans le cadre d'une architecture PKI.
- Côté client, récupération de clés.
- Côté serveur, dialogue avec les autorités de certification.

45

## Présentation

- I. Principes de conception
- II. Prérequis
- III. Syntaxe
- IV. Extensions
- V. Implantations
- VI. Derniers mots
- VII. Références

46

## VII. Références

- W3C

- XML-Signature Requirements draft
- XML-Signature Syntax and Processing
- XML-Signature XPath Filter 2.0
- Canonical XML Version 1.0
- Exclusive XML Canonicalization Version 1.0

- IETF

- Additional XML Security URIs draft #9

47

Merci de votre attention

Questions



## 1. Principes et étendue - 2

### ● Règle 3

- Le « **manifest** » supporte des références :
  - À des ressources sur **Internet**,
  - sur les **condensats** du contenu de la référence ou sa **forme canonique** et
  - optionnellement au **type de contenu** de la ressource.

49

## 1. Principes et étendue - 3

### ● Règle 4

- Le contenu du « **manifest** » est lié avec une **clé** grâce à une **transformation à sens unique**.
- Seules la **syntaxe** et les règles de traitements nécessaires à la **validité** d'une signature sont spécifiées.

50

## 1. Principes et étendue - 4

- Règle 5

- **Conforme** aux spécifications suivantes :

- XML Namespaces
- XLink
- XML Pointers

51

## 3. Format - 2

- Règle 3

- Une utilisation importante des signatures XML sera les **signatures détachées**.
- Les signatures peuvent être **intégrées** au, ou **encapsuler** le, contenu signé.

52

## IV. Extensions - 2

- DigestMethod
  - MD5
  - SHA-224, SHA-384
- SignatureMethod (Authentication)
  - HMAC-MD5
  - HMAC-SHA
  - HMAC-RIPEMD160

53

## IV. Extensions - 3

- SignatureMethod (Clé publique)
  - RSA-MD5
  - RSA-SHA256
  - RSA-SHA384
  - RSA-SHA512
  - RSA-RIPEMD160
  - ECDSA-SHA
  - ESIGN-SHA1

54

## IV. Extensions - 4

- Transform
  - XPointer
- EncryptionMethod
  - ARCFOUR
  - Camellia Block
  - Camellia Key Wrap
  - PSEC-KEM
- KeyInfo
  - PKCS #7 et CRLs