بنك الجزائر
BANQUE D'ALGERIE

cma

GUIDE DE SIGNATURE ELECTRONIQUE DE MESSAGES MX

Système ARTS

pour la Banque d'Algérie

Numéro de version: 001

© 2025, CMA Small Systems AB

Aucune partie de cette publication ne peut être reproduite ou transmise sous quelque forme ou à quelque fin que ce soit sans l'autorisation expresse de CMA Small Systems AB.

Nom de project:	Mise à niveau ARTS - RTGS de la Banque d'Algérie
Nom du document:	Guide de signature électronique de messages MX

CONTENU

1 Historique des révisions

Version	Date	Auteurs	Commentaires
001	17.06.2025	CMA Small Systems AB	Version initiale

1 Introduction

Interface de messagerie de l'API REST. Cette option peut être utilisée par les clients sans installation de logiciel CMA dans leur infrastructure. Le canal de connexion HTTP est protégé par le cryptage TLS/SSL - HTTPS. Dans la mise en œuvre du RTGS, l'authentification client sera effectuée avec un certificat pour garantir que le nœud central accepte les demandes de services Web uniquement d'un système (ou de systèmes) disposant d'un certificat délivré par l'autorité de certification désignée. La signature numérique du contenu des messages sortants et la vérification des signatures numériques des messages entrants doivent être mises en œuvre dans l'application client (telle que le système d'information de Participant).

Ce document décrit l'utilisation globale des signatures numériques lors de l'échange de messages ISO20022 avec RTGS.

La sécurité des Systèmes utilisant les échanges via Réseau Privé (VPN) répond à plusieurs objectifs :

- Confidentialité – les informations ne sont divulguées qu'à la personne autorisée à l'endroit autorisé.
- Intégrité – les informations peuvent être considérées comme complètes, exactes et valides.

La confidentialité et l'intégrité sont assurées par la sécurisation de la transmission, de la livraison et du stockage des messages, par la validation des messages et par l'utilisation de facilité de cryptographie.

Tous les messages transmis entre le site du Participant et le site du Système Central sont cryptés. Seules les personnes autorisées peuvent utiliser un logiciel destiné à communiquer avec le système.

Tous les messages en entrée sont validés pour garantir qu'ils sont conformes à la syntaxe des messages du système. Seuls les messages conformes à la syntaxe sont acceptés.

L'authentification du message garantit l'identité de l'expéditeur et l'intégrité du texte du message.

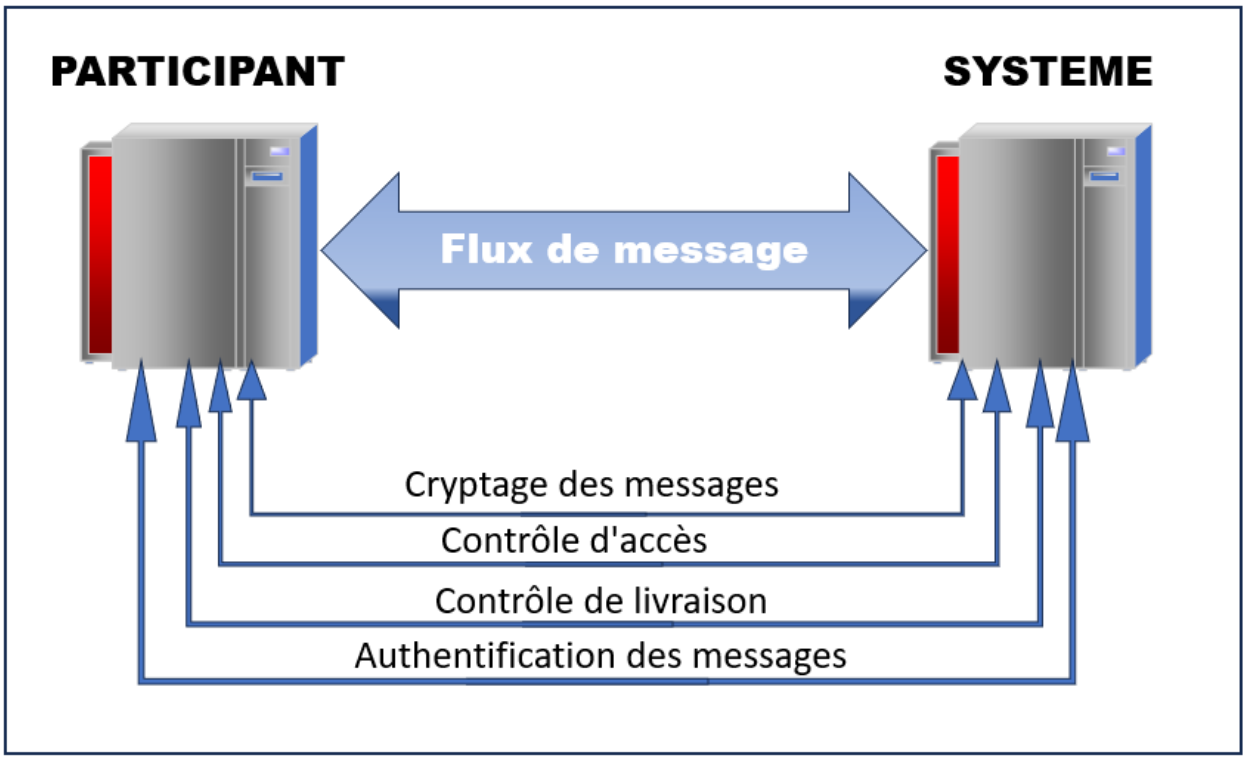

Figure 1.1 Principes de sécurité

2 Définitions et abréviations

ACK	Accusé de réception
API	Interface de programmation d'applications
BAH	En-tête d'application métier (partie du message MX)
LCR	Liste de révocation de certificats
VPN	Réseau privé virtuel
HTTP	Hypertext Transfer Protocol est un protocole d'application pour les systèmes d'information hypermédia distribués et collaboratifs
HTTPS	Hypertext Transfer Protocol over TLS/SSL est un protocole d'application pour les systèmes d'information hypermédia distribués et collaboratifs avec cryptage des données
IETF	Groupe de travail sur l'ingénierie Internet
ISO	Organisation internationale de normalisation
LDAP	Le Lightweight Directory Access Protocol est un protocole d'application standard ouvert, indépendant du fournisseur, permettant d'accéder et de maintenir des services d'informations d'annuaire distribués sur un réseau IP (Internet Protocol)
MX	Message XML au format ISO20022
NAK	Accusé de réception négatif
Participant	Participant au RTGS

REST	Representational State Transfer (REST) est un style architectural logiciel qui définit un ensemble de contraintes à utiliser pour créer des services Web.
RSC	Communication sécurisée à distance
SSL	Le protocole Secure Sockets Layer est un protocole cryptographique conçu pour assurer la sécurité des communications sur un réseau informatique
STP	Traitement de bout en bout
TLS	Le protocole Transport Layer Security est un protocole cryptographique conçu pour assurer la sécurité des communications sur un réseau informatique
W3C	World Wide Web Consortium

3 Schémas de signature de messages

L'infrastructure PKI de la banque centrale sera utilisée pour émettre et gérer les certificats. Il incombe à la partie expéditrice de générer la signature numérique du bloc financier du message et de placer cette signature numérique dans le BAH (head.001).

Les sections ci-dessous représentent les schémas de signature pour différentes interfaces lorsque le message est envoyé à travers le canal de transfert VPN.

En cas d'échec de validation de la signature du contenu métier, le système retournera un code erreur http 400.

Exemple d'erreur:

```
HTTP/1.1 400 Bad Request
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers
Content-Type: application/json
X-Content-Type-Options: nosniff
X-XSS-Protection: 0
Cache-Control: no-cache, no-store, max-age=0, must-revalidate
Pragma: no-cache
Expires: 0
Strict-Transport-Security: max-age=31536000 ; includeSubDomains
X-Frame-Options: DENY
Content-Length: 254
{
   "timestamp" : "2024-04-12T10:02:38.285+00:00",
   "status" : 400,
   "error" : "Bad Request",
   "message" : "Signature check fail, Signature of MX Document is missing",
   "path" : "/input/0eecaf02-2301-4638-bb96-b67973c57943",
   "errorCode" : "SC312"
}
```

Le paramètre « errorCode » contient toujours la valeur « SC312 ». Les détails d'erreur sont indiqués dans le paramètre « message ».

Pour chaque utilisateur enregistré dans le système, tout Participant utilisant le Réseau Privé doit disposer d'une paire de clés numériques. Une partie de la paire de clé dite secrète (appelée clé privée) doit être conservée dans un lieu de stockage sûr. L'autre partie est appelée clé publique. Celle-ci doit être accessible en lecture aux autres participants.

Lors d'un échange le premier participant applique la signature numérique à l'aide de sa propre clé privée et l'autre participant vérifie cette signature à l'aide de la clé publique du premier.

4 REST

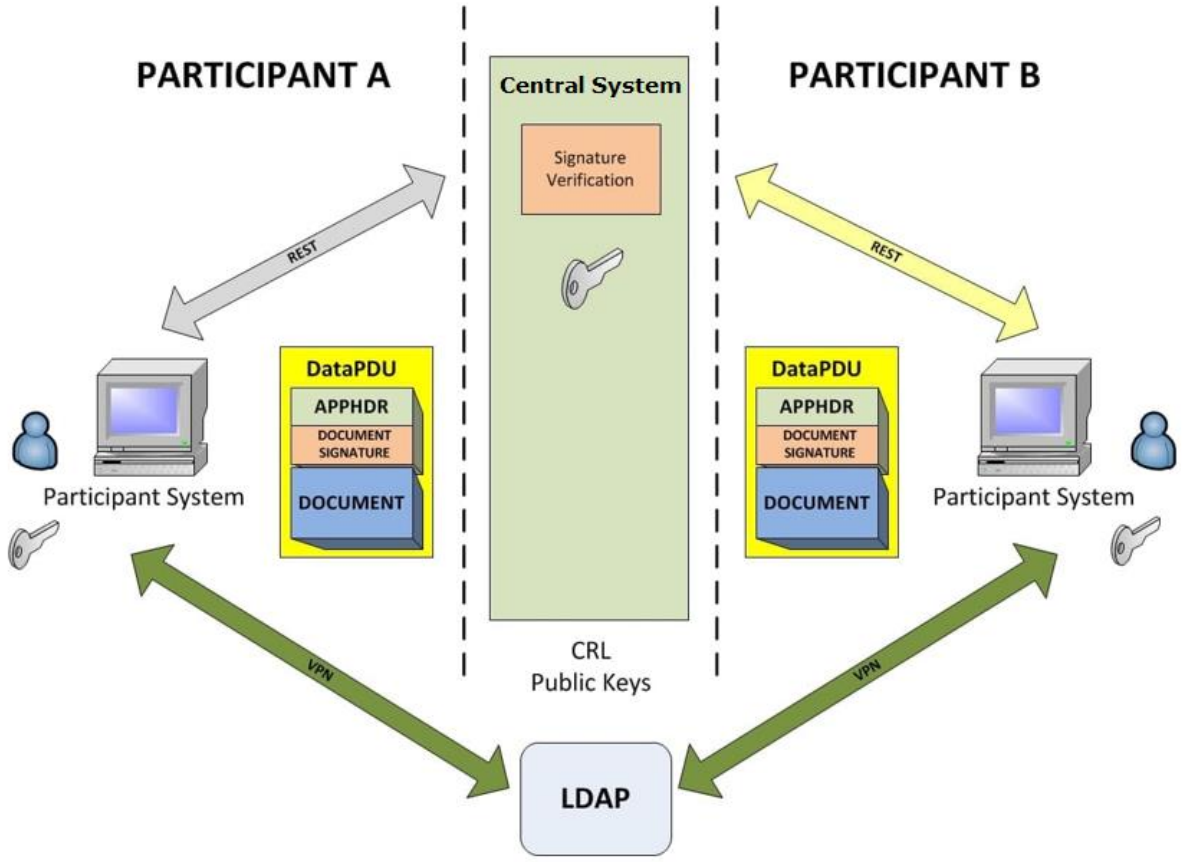

Figure 4.1 Processus de signature - Interface REST

1. Le Système d'Information Bancaire (Participant A) génère le bloc **Document** (bloc d'informations financières).

2. Le Système d'Information Bancaire (Participant A) applique la signature métier. C'est la responsabilité de l'émetteur de générer la signature numérique du bloc financier et de placer cette signature numérique dans l'en-tête financier **AppHdr** (head.001).

   Il n'est pas nécessaire d'ajouter une signature de transport supplémentaire ou d'appliquer un cryptage puisque tous les messages transmis entre le site du participant et le site du système sont cryptés par SSL au niveau HTTP (c'est-à-dire HTTPS).

3. Le message est envoyé au système pour traitement.

4. Le message du participant A est reçu et traité dans le système.

   - Lorsque le système reçoit le message, la signature métier d'en-tête est vérifiée et stockée avec le message dans la base de données du système.

- En cas d'échec de la validation de la signature, le code erreur http 400 est envoyé en réponse.

5. Le système génère le message au participant destinataire (Participant B) au format MX (avec signature métier).

   - La signature du Participant A (signature initiale) est remplacée par la signature numérique du système (le contenu de la balise « Document » n'est pas modifiée en cas de génération de copie du message).

6. Le message destiné au participant B est envoyé par le système.

7. Le message au format MX (avec signature métier) est transmis au Système d'Information Bancaire du participant B.

8. Le Système d'Information Bancaire du participant B vérifie la signature métier à l'aide de la clé publique du Système Centrtal et de la CRL - Certificate Revocation List (Liste de révocation de certificat) obtenues auprès du LDAP de l'institution centrale par connexion au réseau privé (ou préalablement téléchargée et placée dans un magasin de clés local).

5 Signature numérique et vérification des signatures des messages MX

Dans le cas des messages MX, la signature numérique de la couche métier du message MX est utilisée pour authentifier l'expéditeur du message et pour garantir l'intégrité du contenu financier. Cette signature Métier doit être conforme au standard XAdES: la signature XAdES-BES avec certains accords/conventions sont définis ci-dessous.

Alors que la norme de signature XML IETF/W3C (généralement appelée XML-DSig) constitue un cadre général pour la signature numérique de documents, XAdES spécifie des profils précis de XML-DSig offrant certaines garanties. XAdES-BES (pour « Basic Electronic Signature ») fournit l'authentification de base et une protection d'intégrité, essentielles pour les signatures électroniques avancées dans les systèmes de paiement.

Dans XAdES, la signature doit être appliquée de la manière habituelle de XML-DSig sur le document à signer et sur l'ensemble des propriétés signées (élément SignedProperties). L'information obligatoire dans l'élément SignedProperties est l'heure de signature, correspondant à l'heure à laquelle le signataire indique avoir effectué le processus de signature. Ci-dessous, une description détaillée des règles d'utilisation de XAdES adoptées dans le système.

5.1 Règles d'utilisation de XAdES

Les signatures XAdES-BES utilisées dans le système doivent être conformes aux règles d'utilisation suivantes :

1. Utilisation du bloc « Object » (ds:Object)

   L'élément ds:Object ne doit avoir qu'une seule valeur dans QualifyingProperties/SignedProperties/SignedSignatureProperties : l'heure de signature, spécifiant l'heure à laquelle le signataire déclare avoir effectué le processus de signature.

   L'attribut Id de l'élément KeyInfo est obligatoire et la valeur de l'attribut ID doit être un trait de soulignement (« _ ») suivi d'un identifiant universellement unique (UUID), basé sur le temps ou aléatoire.

2. Utilisation du bloc KeyInfo

   La norme XAdES permet à deux méthodes différentes de se conformer à l'exigence XAdES-BES. Dans le système, il a été décidé d'utiliser celui qui inclut le numéro de série du certificat du signataire dans l'élément KeyInfo :

- L'élément KeyInfo doit être présent et doit inclure les balises ds:X509Data/ds:X509IssuerSerial/ds:X509IssuerName et

ds:X509Data/ds:X509IssuerSerial/ds:X509SerialNumber contenant le nom de l'émetteur et le numéro de série du certificat du signature respectivement.

- L'attribut 'Id' de l'élément <KeyInfo> est obligatoire et la valeur de l'attribut 'Id' doit être un trait de soulignement (« _ ») suivi d'un identifiant universellement unique (UUID), basé sur le temps ou aléatoire.

- L'élément <SignedInfo> doit référencer l'élément <KeyInfo> à l'aide de l'attribut 'Id'. L'utilisation de l'élément alternatif ds:Object/ QualifyingProperties/ SignedProperties/ SignedSignatureProperties/ SigningCertificate n'est pas autorisée.

- L'attribut 'Id' de l'élément <Signature> est obligatoire et la valeur de l'attribut 'Id' doit être un trait de soulignement (« _ ») suivi d'un identifiant universellement unique (UUID), qui est soit basé sur le temps, soit aléatoire.

- <QualifyingProperties> doit avoir un attribut 'Target' faisant référence à 'Id' de l'élément <Signature>.

5.2 Références dans SignedInfo

Il doit y avoir 3 références dans <SignedInfo> :

1. Référence à <KeyInfo> (qui contient une référence sans ambiguïté au certificat du signataire).

   La référence doit avoir l'attribut URI faisant référence à l'ID de l'élément <KeyInfo>.

2. Référence à <SignedProperties> sous <ds:Object>. La référence doit avoir l'attribut 'URI' faisant référence l'attribut 'Id' de l'élément <SignedProperties>.

3. Référence spécifique à l'application sans URI - cette référence pointe vers le document métier - élément <Document>.

   Les signatures XML standard autorisent une référence spéciale : une référence sans attribut URI. Il est défini par la spécification de signature XML que l'application réceptrice doit être capable d'identifier l'objet utilisé dans ce cas. Dans cette spécification et ce système, la référence sans URI fait référence à l'élément <Document> avec l'intégralité de son contenu.

5.3 Exemple de signature

Ci-dessous un exemple de signature acceptable :

```
<Sgntr>
    <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
Id="_201591cc-9272-480b-9465-a91924096872">
        <ds:SignedInfo>
```

```
            <ds:CanonicalizationMethod
Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"/>
            <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-
more#rsa-sha256"/>
            <ds:Reference URI="#_d15ad124-9b27-4230-ba6b-bdaaf15d2d99">
                <ds:Transforms>
                    <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-
exc-c14n#"/>
                </ds:Transforms>
                <ds:DigestMethod
Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>

<ds:DigestValue>F+Tw5BGdS82rsn8gk9N+NLD3nirXiPrQv3uhNF9d4hs=</ds:DigestValue>
            </ds:Reference>
            <ds:Reference
Type="http://uri.etsi.org/01903/v1.3.2#SignedProperties" URI="#_a697c49c-
5d4b-4ca7-8eff-4c5017c39216-signedprops">
                <ds:Transforms>
                    <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-
exc-c14n#"/>
                </ds:Transforms>
                <ds:DigestMethod
Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>

<ds:DigestValue>x0J1//wYP35qelDk9+HVmn4kp98xiJoXLqgMFgbXp9c=</ds:DigestValue>
            </ds:Reference>
            <ds:Reference>
                <ds:Transforms>
                    <ds:Transform Algorithm="http://www.w3.org/2001/10/xml-
exc-c14n#"/>
                </ds:Transforms>
                <ds:DigestMethod
Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>

<ds:DigestValue>+Vd0LNeuGtAsI4r3JXtiwwjwPWTZyk2VWdVKAH5iwkk=</ds:DigestValue>
            </ds:Reference>
        </ds:SignedInfo>

<ds:SignatureValue>rsaivpqdIFViKMZFLjN/O5eB/jA24q9ioMyiCLv5/ROF3bnzSXPko2cFKI
nx4Ul3YWy3M7XXmJ5oUb9q0/RjU/ddYETORZiPH01EmXnR28fBRdKKHUIiFLFb9fjezE3SCI9kI/7
49WRaTGJ8qIXnd5Np0cJyXbG4SM4ptIT0oKVZ0Z+Y0CpX+3+mp4aSz0Rwi2nkuoGRyGUiA0YAWzXR
0SkJQb4hpN6MU7CL5VnPajWFYPSw7Rxj0+KS1u7zYLLyqzl8VCXNKDIHofMaBB9bI/zJ7cOyhQRVg
gW1xfinHSCcmkdStakXN3HHwJ0DGTPcEDq1reCeg3CMR97p6kGwgg==</ds:SignatureValue>
        <ds:KeyInfo Id="_d15ad124-9b27-4230-ba6b-bdaaf15d2d99">
            <ds:X509Data>
                <ds:X509IssuerSerial>
                    <ds:X509IssuerName>CN=CMA-TEST-DESK, DC=test-ca, DC=cma,
DC=ru</ds:X509IssuerName>

<ds:X509SerialNumber>468315657143224659708344587689388376988845567</ds:X509Se
rialNumber>
                </ds:X509IssuerSerial>
            </ds:X509Data>
        </ds:KeyInfo>
        <ds:Object>
```

```
            <xades:QualifyingProperties
xmlns:xades="http://uri.etsi.org/01903/v1.3.2#" Target="#_201591cc-9272-480b-
9465-a91924096872">
                <xades:SignedProperties Id="_a697c49c-5d4b-4ca7-8eff-
4c5017c39216-signedprops">
                    <xades:SignedSignatureProperties>
                        <xades:SigningTime>2024-12-
28T14:52:32+07:00</xades:SigningTime>
                    </xades:SignedSignatureProperties>
                </xades:SignedProperties>
            </xades:QualifyingProperties>
        </ds:Object>
    </ds:Signature>
</Sgntr>
```

- La signature commence par l'élément Signature dans l'espace de noms http://www.w3.org/2000/09/xmldsig#.
- L'élément SignedInfo correspond aux informations réellement signées. Il référence les données signées et précise les algorithmes qui sont utilisés.
- Les éléments SignatureMethod et CanonicalizationMethod sont utilisés par l'élément SignatureValue et sont inclus dans SignedInfo pour les protéger contre la falsification.
- Un ou plusieurs éléments Reference spécifient la ressource signée par référence URI; et toutes les transformations à appliquer à la ressource avant la signature. La transformation peut être une expression XPath qui sélectionne un sous-ensemble défini de l'arborescence du document.
- DigestMethod spécifie l'algorithme de hachage avant l'application du hachage.
- DigestValue contient le résultat de l'application de l'algorithme de hachage à la ressource transformée.
- L'élément SignatureValue contient le résultat de la signature codée en Base64 - la signature générée avec les paramètres spécifiés dans l'élément SignatureMethod - de l'élément SignedInfo après application de l'algorithme spécifié par CanonicalizationMethod.
- L'élément KeyInfo permet au vérificateur d'identifier le certificat du signataire nécessaire à la validation de la signature. KeyInfo ne doit contenir que le nom de l'émetteur et le numéro de série du certificat, mais pas le certificat complet - ceci afin d'économiser du trafic réseau et de l'espace disque/base de données.

Des informations supplémentaires concernant le fichier XSD pour la section de signature sont disponibles sur les sites Web:

https://www.w3.org/

https://www.w3.org/TR/xmldsig-core/

https://uri.etsi.org/01903/v1.3.2/XAdES.xsd

5.4 Exemple complet de document MX avec signature

La signature XAdES décrite ci-dessus doit être stockée dans l'élément Sgntr de l'élément AppHdr du document MX.

Ci-dessous un exemple complet de document avec signature. Le bloc de signature numérique est surligné en vert. Les données commerciales signées (contenu de l'élément Document) sont surlignées en bleu. Il s'agit des données protégées par la signature.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<DataPDU xmlns="urn:cma:stp:xsd:stp.1.0">
    <Body>
        <AppHdr xmlns="urn:iso:std:iso:20022:tech:xsd:head.001.001.01">
            <Fr>
                <FIId>
                    <FinInstnId>
                        <BICFI>AAAAYYZZ</BICFI>
                    </FinInstnId>
                </FIId>
            </Fr>
            <To>
                <FIId>
                    <FinInstnId>
                        <BICFI>SYSTEMZZ</BICFI>
                    </FinInstnId>
                </FIId>
            </To>
            <BizMsgIdr>AAAAYYZZAXXX18021710012000000123O</BizMsgIdr>
            <MsgDefIdr>pacs.008.001.08</MsgDefIdr>
            <BizSvc>IPS</BizSvc>
            <CreDt>2019-09-13T18:18:00Z</CreDt>
            <Sgntr>
                <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
Id="_201591cc-9272-480b-9465-a91924096872">
                    <ds:SignedInfo>
                        <ds:CanonicalizationMethod
Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"/>
                        <ds:SignatureMethod
Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"/>
                        <ds:Reference URI="#_d15ad124-9b27-4230-ba6b-
bdaaf15d2d99">
                            <ds:Transforms>
                                <ds:Transform
Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"/>
                            </ds:Transforms>
                            <ds:DigestMethod
Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>

<ds:DigestValue>F+Tw5BGdS82rsn8gk9N+NLD3nirXiPrQv3uhNF9d4hs=</ds:DigestValue>
```

```
                        </ds:Reference>
                        <ds:Reference
Type="http://uri.etsi.org/01903/v1.3.2#SignedProperties" URI="#_a697c49c-
5d4b-4ca7-8eff-4c5017c39216-signedprops">
                            <ds:Transforms>
                                <ds:Transform
Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"/>
                            </ds:Transforms>
                            <ds:DigestMethod
Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>

<ds:DigestValue>x0J1//wYP35qelDk9+HVmn4kp98xiJoXLqgMFgbXp9c=</ds:DigestValue>
                        </ds:Reference>
                        <ds:Reference>
                            <ds:Transforms>
                                <ds:Transform
Algorithm="http://www.w3.org/2001/10/xml-exc-c14n#"/>
                            </ds:Transforms>
                            <ds:DigestMethod
Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>

<ds:DigestValue>+Vd0LNeuGtAsI4r3JXtiwwjwPWTZyk2VWdVKAH5iwkk=</ds:DigestValue>
                        </ds:Reference>
                    </ds:SignedInfo>

<ds:SignatureValue>rsaivpqdIFViKMZFLjN/O5eB/jA24q9ioMyiCLv5/ROF3bnzSXPko2cFKI
nx4Ul3YWy3M7XXmJ5oUb9q0/RjU/ddYETORZiPH01EmXnR28fBRdKKHUIiFLFb9fjezE3SCI9kI/7
49WRaTGJ8qIXnd5Np0cJyXbG4SM4ptIT0oKVZ0Z+Y0CpX+3+mp4aSz0Rwi2nkuoGRyGUiA0YAWzXR
0SkJQb4hpN6MU7CL5VnPajWFYPSw7Rxj0+KS1u7zYLLyqzl8VCXNKDIHofMaBB9bI/zJ7cOyhQRVg
gW1xfinHSCcmkdStakXN3HHwJ0DGTPcEDq1reCeg3CMR97p6kGwgg==</ds:SignatureValue>
                    <ds:KeyInfo Id="_d15ad124-9b27-4230-ba6b-bdaaf15d2d99">
                        <ds:X509Data>
                            <ds:X509IssuerSerial>
                                <ds:X509IssuerName>CN=CMA-TEST-DESK, DC=test-
ca, DC=cma, DC=ru</ds:X509IssuerName>

<ds:X509SerialNumber>468315657143224659708344587689388376988845567</ds:X509Se
rialNumber>
                            </ds:X509IssuerSerial>
                        </ds:X509Data>
                    </ds:KeyInfo>
                    <ds:Object>
                        <xades:QualifyingProperties
xmlns:xades="http://uri.etsi.org/01903/v1.3.2#" Target="#_201591cc-9272-480b-
9465-a91924096872">
                            <xades:SignedProperties Id="_a697c49c-5d4b-4ca7-
8eff-4c5017c39216-signedprops">
                                <xades:SignedSignatureProperties>
                                    <xades:SigningTime>2024-12-
28T14:52:32+07:00</xades:SigningTime>
                                </xades:SignedSignatureProperties>
                            </xades:SignedProperties>
                        </xades:QualifyingProperties>
                    </ds:Object>
                </ds:Signature>
            </Sgntr>
        </AppHdr>
```

```
<Document xmlns="urn:iso:std:iso:20022:tech:xsd:pacs.008.001.08">
    <FIToFICstmrCdtTrf>
        <GrpHdr>
            <MsgId>AAAAYYZZAXXX18021710012000000123O</MsgId>
            <CreDtTm>2019-09-13T18:18:00</CreDtTm>
            <NbOfTxs>1</NbOfTxs>
            <SttlmInf>
                <SttlmMtd>CLRG</SttlmMtd>
            </SttlmInf>
        </GrpHdr>
        <CdtTrfTxInf>
            <PmtId>
                <EndToEndId>NOTPROVIDED</EndToEndId>
                <TxId>AAAAYYZZAXXX18021710012000000123O</TxId>
            </PmtId>
            <PmtTpInf>
                <ClrChanl>RTNS</ClrChanl>
                <LclInstrm>
                    <Prtry>CSCT</Prtry>
                </LclInstrm>
                <CtgyPurp>
                    <Prtry>001</Prtry>
                </CtgyPurp>
            </PmtTpInf>
            <IntrBkSttlmAmt Ccy="USD">71.12</IntrBkSttlmAmt>
            <IntrBkSttlmDt>2019-09-14</IntrBkSttlmDt>
            <ChrgBr>SLEV</ChrgBr>
            <InstgAgt>
                <FinInstnId>
                    <BICFI>AAAAYYZZ</BICFI>
                </FinInstnId>
            </InstgAgt>
            <InstdAgt>
                <FinInstnId>
                    <BICFI>BBBBYYZZ</BICFI>
                </FinInstnId>
            </InstdAgt>
            <Dbtr>
                <Nm>John Johnson</Nm>
            </Dbtr>
            <DbtrAcct>
                <Id>
                    <IBAN>12345678998076</IBAN>
                </Id>
            </DbtrAcct>
            <DbtrAgt>
                <FinInstnId>
                    <BICFI>AAAAYYZZ</BICFI>
                    <Othr>
                        <Id>200004</Id>
                        <SchmeNm>
                            <Prtry>1700085041</Prtry>
                        </SchmeNm>
                    </Othr>
                </FinInstnId>
            </DbtrAgt>
            <DbtrAgtAcct>
```

```
                        <Id>
                            <IBAN>89980761234567</IBAN>
                        </Id>
                    </DbtrAgtAcct>
                    <CdtrAgt>
                        <FinInstnId>
                            <BICFI>BBBBYYZZ</BICFI>
                            <Othr>
                                <Id>210027</Id>
                                <SchmeNm>
                                    <Prtry>1400108191</Prtry>
                                </SchmeNm>
                            </Othr>
                        </FinInstnId>
                    </CdtrAgt>
                    <CdtrAgtAcct>
                        <Id>
                            <IBAN>98765432198765</IBAN>
                        </Id>
                    </CdtrAgtAcct>
                    <Cdtr>
                        <Nm>Omega Jones</Nm>
                    </Cdtr>
                    <CdtrAcct>
                        <Id>
                            <IBAN>54637281908745</IBAN>
                        </Id>
                    </CdtrAcct>
                    <InstrForNxtAgt>
                        <InstrInf>/BNF/Details</InstrInf>
                    </InstrForNxtAgt>
                    <Purp>
                        <Prtry>5814</Prtry>
                    </Purp>
                    <RgltryRptg>
                        <Dtls>
                            <Inf>SOMEINFORMATIONABOUTPAYMENT-1</Inf>
                            <Inf>SOMEINFORMATIONABOUTPAYMENT-2</Inf>
                            <Inf>SOMEINFORMATIONABOUTPAYMENT-3</Inf>
                        </Dtls>
                    </RgltryRptg>
                    <RmtInf>
                        <Ustrd>INFORMATION</Ustrd>
                        <Ustrd>EXTRA INFO</Ustrd>
                    </RmtInf>
                </CdtTrfTxInf>
            </FIToFICstmrCdtTrf>
        </Document>
    </Body>
</DataPDU>
```

5.5 Vérification de signature

Les étapes obligatoires de la verification comprennent la validation des références, la vérification du digest contenu dans chaque référence de SignedInfo et la validation de la signature cryptographique de la signature calculée sur SignedInfo.

Validation des références

1. Canonicaliser l'élément SignedInfo sur la base de la méthode de CanonicalizationMethod de SignedInfo.

2. Pour chaque référence dans SignedInfo :

   - Obtenir l'objet de données pour lequel il faut calculer le digest.

   - Calculer le digest de l'objet de données résultant à l'aide de la méthode DigestMethod spécifiée dans la spécification de la référence.

   - Comparer la valeur de digest générée à DigestValue dans la référence SignedInfo; en cas de non-concordance, la validation échoue.

Validation de la signature

1. Obtenir les informations relatives à la clé à partir de KeyInfo.

2. Obtenir la forme canonique de SignatureMethod à l'aide de CanonicalizationMethod et utiliser le résultat (ainsi que KeyInfo obtenue précédemment) pour confirmer la valeur de la signature sur l'élément SignedInfo.

Pour en savoir plus sur la vérification de signature, veuillez vous rendre sur la page suivante: https://www.w3.org/TR/xmldsig-core1/#sec-CoreValidation

6 Module d'exemples de signature

Le module d'exemples de signature est une application Spring Boot utilisée pour signer et vérifier des exemples et des tests. Ce module est fourni avec ce document sous forme d'un archive zip. Pour les instructions d'utilisation consultez le fichier README.md qui se trouve dans le répertoire racine.

En plus d'être une source d'exemples, il peut être utilisé pour signer ou vérifier des messages.

6.1 Signature de messages

Pour un exemple de code source pour la signature de messages, voir le fichier SignerService.java dans src/main/java/se/highex/examples/signatures/mx/service/.

Pour signer un message à l'aide d'une paire de clés dans un keystore, exécutez la commande suivante

```
java -jar signatures-mx-xades-1.6.jar \
--se.highex.example.action=sign \
--se.highex.example.keystoreFile=/cma/keystore.pfx \
--se.highex.example.keystorePass=123456 \
--se.highex.example.keyAlias=te-da551a69-6d1d-4948-b10a-9b1e637de589 \
--se.highex.example.documentToSign=/cma/example.xml
```

Par défaut, la sortie des opérations de signature est verbeuse et les journaux contiennent des données de pré-digestion au format hexagonal (également appelées « tampons de signature » ou « sign buffers »).

6.2 Vérification de signature

Pour un exemple de code source pour la vérification des signatures, voir le fichier VerifierService.java dans src/main/java/se/highex/examples/signatures/mx/service/.

Pour vérifier la signature d'un message en utilisant un certificat, exécutez la commande suivante :

```
java -jar signatures-mx-xades-1.6.jar
--se.highex.example.action=verify \
--se.highex.example.certFile=/cma/certificate.pem \
--se.highex.example.documentToVerify=/cma/example-signed.xml
```

6.3 Déréférenceur d'URI personnalisé

Les signatures XML standard autorisent une référence spéciale : une référence sans attribut URI. Il est défini par la spécification de signature XML que l'application réceptrice doit être capable d'identifier l'objet qui est utilisé dans ce cas. Dans cette

spécification et dans le système, la référence sans URI fait référence à l'élément Document (et à l'intégralité de son contenu). Par conséquent, un déréférenceur personnalisé est requis pour savoir comment obtenir toutes les données référencées à partir des URI dans tous les types de références, y compris une référence sans attribut URI. Ce déréférenceur personnalisé implémente javax.xml.crypto.URIDereferencer et s'appelle NoUriDereferencer.

Pour un exemple du code source du déréférenceur, voir le fichier NoUriDereferencer.javan dans src/main/java/se/highex/examples/signatures/mx/util/.

6.4 Débogage des erreurs de vérification de signature numérique

Lors de l'ajout de la prise en charge des signatures numériques MX dans votre application sur la base des exemples ci-dessus, vous pourriez rencontrer un problème lorsque vous réussissez à créer une signature numérique, mais que votre propre code de vérification (ou celui d'un tiers) ne parvient pas à la vérifier (c'est-à-dire que la signature n'est pas valide).

Cela peut se produire si le document a été modifié entre la création de la signature et sa vérification; par exemple des ajouts d'espaces supplémentaires, des caractères de fin ligne, des données ajoutées dans le document, ou si le document a été reformaté d'une manière ou d'une autre.

Par défaut, la sortie des opérations de signature contient des journaux de débogage et des données de pré-digestion au format hexagonal (également appelées « sign buffers »).

Si votre signature échoue à la vérification dans le système central, activez la sortie de débogage (debug output), signez un document et transmettez-le avec la sortie de débogage à l'assistance ou au fournisseur du système central pour qu'il procède à des recherches plus approfondies. En comparant vos « données transformées » et les valeurs de digest avec les valeurs obtenues lors de la vérification, ils pourraient être en mesure de trouver la partie du document qui a été modifiée et d'aider à résoudre le problème.

Vous pouvez configurer les informations de débogage dans la sortie en utilisant le paramètre logging.level.se.highex dans le fichier application.properties qui se trouve dans src/main/resources/.