Groupe de travail Réseau
Request for Comments : 4130
Catégorie : Sur la voie de la normalisation
Traduction Claude Brière de L'Isle

D. Moberg, Cyclone Commerce
R. Drummond, Drummond Group Inc.

juillet 2005

Échange de données d'affaires sécurisé d'homologue à homologue fondé sur MIME en utilisant HTTP : déclaration d'applicabilité 2 (AS2)

Résumé
Le présent document fournit une déclaration d'applicabilité (paragraphe 3.2 de la RFC 2026) qui décrit comment échanger en toute sécurité des données d'affaires structurées en utilisant le protocole de transfert HTTP, au lieu de SMTP ; la déclaration d'applicabilité pour SMTP se trouve dans la RFC 3335. Les données d'affaires structurées peuvent être en XML, dans le format X12 d'échange de données électroniques (EDI, *Electronic Data Interchange*) du comité national de normalisation américain (ANSI, *American National Standards Committee*) ou dans le format d'échange de données électroniques des Nations Unies pour l'administration, le commerce et le transport (UN/EDIFACT, *UN Electronic Data Interchange for Administration, Commerce, and Transport*) ou d'autres formats de données structurés. Les données sont mises en paquet en utilisant les structures MIME standard. L'authentification et la confidentialité des données sont obtenues en utilisant la syntaxe de message cryptographique (CMS, *Cryptographic Message Syntax*) avec les parties de corps de sécurité S/MIME. Les accusés de réception authentifiés utilisent les réponses multiparties/signées de notification de disposition de message (MDN, *Message Disposition Notification*) au message HTTP d'origine. La présente déclaration d'applicabilité est désignée de façon informelle par "AS2" parce que c'est la seconde déclaration d'applicabilité, produite après "AS1", RFC3335.

Table des Matières

1. Introduction

1.1 RFC Applicables

Les travaux antérieurs sur les EDI Internet se concentraient sur la spécification des types de contenu MIME pour les données d'EDI [RFC1767] et étendaient ce travail pour prendre en charge le transport sûr des EC/EDI sur SMTP [RFC3335]. Le présent document reprend la RFC 1767 pour spécifier un ensemble complet de caractéristiques de sécurité des données, spécifiquement la confidentialité des données, l'intégrité/authenticité des données, la non répudiation de l'origine, et la non répudiation de la réception sur HTTP. Le présent document reconnaît aussi les RFC contemporaines et tente de "réinventer" aussi peu que possible. Bien que le présent document se concentre sur les données d'EDI, tous les autres types de données descriptibles dans un format MIME sont aussi pris en charge.

Les EDI Internet fondés sur MIME peuvent être réalisés en utilisant et se conformant aux RFC suivantes :
- RFC 2616 Protocole de transfert Hypertexte
- RFC 1767 Type de contenu d'EDI
- RFC 3023 Types de supports XML
- RFC 1847 Sécurité multiparties pour MIME
- RFC 3462 Multipart/Report
- RFC 2045 à 2049 MIME
- RFC 3798 Notification de disposition de message
- RFC 3851, 3852 Spécification de S/MIME v3.1

L'intention du présent document est de décrire clairement et précisément comment ces RFC sont utilisées ensemble, et ce qui est exigé des agents d'utilisateur pour se conformer à ce document.

Les mots clés "DOIT", "NE DOIT PAS", "EXIGE", "DEVRA", "NE DEVRA PAS", "DEVRAIT", "NE DEVRAIT PAS", "RECOMMANDE", "PEUT", et "FACULTATIF" en majuscules dans ce document sont à interpréter comme décrit dans le BCP 14, [RFC2119].

1.2 Termes

AS2 *(Applicability Statement 2)* Déclaration d'applicabilité 2 (ce document) ; voir le paragraphe 3.2 de la [RFC2026].

EDI *(Electronic Data Interchange)* échange électronique de données

EC *(Business-to-Business Electronic Commerce)* commerce électronique d'affaires

B2B *(Business to Business)* d'affaires

Récépissé : message fonctionnel qui est envoyé d'un receveur à un envoyeur pour accuser réception d'un échange EDI/EC. Ce message peut être par nature synchrone ou asynchrone.

Récépissé signé : récépissé avec une signature numérique.

Récépissé synchrone : récépissé retourné à l'envoyeur durant la même session HTTP que le message original de l'envoyeur.

Récépissé asynchrone : récépissé retourné à l'envoyeur sur une session de communication différente de la session du message original de l'envoyeur.

Notification de disposition de message (MDN, *Message Disposition Notification*) : format de messagerie Internet utilisé pour porter un récépissé. Ce terme est utilisé de façon interchangeable avec récépissé. Une MDN est un récépissé.

Non répudiation de récépissé (NRR) : "événement légal" qui survient lorsque l'envoyeur d'origine d'un échange EDI/EC signé a vérifié le récépissé signé revenant du receveur. Le récépissé contient des données qui identifient le message d'origine pour lequel il est un récépissé, incluant l'identifiant de message et un hachage cryptographique (MIC). L'envoyeur d'origine doit conserver des enregistrements convenables fournissant la preuve du contenu du message, son identifiant de message, et sa valeur de hachage. L'envoyeur d'origine vérifie que la valeur du hachage conservée est la même que le résumé du message d'origine, comme rapporté dans le récépissé signé. NRR n'est pas considéré comme un message technique, mais est plutôt vu comme le résultat de la possession d'une preuve pertinente.

S/MIME : format et protocole pour ajouter une signature cryptographique et/ou des services de chiffrement aux messages MIME Internet.

Syntaxe de message cryptographique (CMS, *Cryptographic Message Syntax*) : syntaxe d'encapsulation utilisée pour signer numériquement, résumer, authentifier, ou chiffrer des messages arbitraires.

SHA-1 : algorithme sûr de hachage unidirectionnel utilisé en conjonction avec la signature numérique. C'est l'algorithme recommandé pour AS2.

MD5 : algorithme sûr de hachage unidirectionnel utilisé en conjonction avec la signature numérique. Cet algorithme est permis dans AS2.

MIC : vérification d'intégrité de message (MIC, *message integrity check*) aussi appelé résumé de message, c'est le résultat résumé de l'algorithme de hachage utilisé par la signature numérique. La signature numérique est calculée sur la MIC.

Agent d'utilisateur (UA, *User Agent*) : application qui traite la demande AS2.

2. Vue d'ensemble

2.1 Fonctionnement global

L'opération HTTP POST [RFC2616] est utilisée pour envoyer des données d'EDI, XML, ou autres données d'affaires mises en paquet de façon appropriée. L'URI de demande *(Request-URI)* (paragraphe 9.5 de la [RFC2616]) identifie un processus de dépaquetage et traitement des données du message et pour générer une réponse au client qui contienne une notification de disposition de message (MDN) signée ou non signée. La MDN est retournée soit dans le corps du message de réponse HTTP, soit par une nouvelle opération HTTP POST à un URL pour l'envoyeur d'origine.

Cette échange transactionnel demande/réponse peut fournir un transport sûr, fiable, et authentifié pour les données d'EDI ou autres données d'affaires en utilisant HTTP comme protocole de transfert.

Les protocoles et structures de sécurité utilisés prennent aussi en charge des enregistrements qui peuvent servir à l'analyse a posteriori de ces document de transmission, accusé de réception, et authentification des données.

2.2 Objet des lignes directrices de sécurité pour les EDI MIME

L'objet de ces spécifications est d'assurer l'interopérabilité entre les agents d'utilisateur EC B2B, impliquant certaines ou toutes les caractéristiques de sécurité couramment attendues. Le présent document N'EST PAS limité à la stricte utilisation des EDI ; il s'applique à toute application de commerce électronique pour laquelle des données d'affaires doivent être échangées sur l'Internet en toute sécurité.

2.3 Définitions

2.3.1 Boucle de transmission sécurisée

Le présent document se concentre sur les formats et protocoles pour échanger en toute sécurité des contenus d'EDI/EC dans l'environnement HTTP de l'Internet.

Dans la "boucle de transmission sûre" pour EDI/EC, une organisation envoie un échange signé et chiffré EDI/EC à une autre organisation et demande un récépissé signé, et plus tard, l'organisation receveuse renvoie le récépissé signé à l'organisation envoyeuse. En d'autres termes, on a ceci :

- L'organisation qui envoie des données EDI/EC signe et chiffre les données en utilisant S/MIME. De plus, le message va demander qu'un récépissé signé soit retourné à l'envoyeur. Pour prendre en charge la NRR, l'envoyeur d'origine conserve des enregistrements du message, de l'identifiant de message, et de la valeur du résumé (MIC).
- L'organisation receveuse déchiffre le message et vérifie la signature, d'où résulte la vérification de l'intégrité des données et de l'authenticité de l'envoyeur.
- L'organisation receveuse retourne alors un récépissé signé en utilisant le corps de réponse HTTP ou une opération HTTP POST séparée à l'organisation envoyeuse sous la forme d'une notification de disposition de message signée. Ce récépissé signé va contenir le hachage du message reçu, permettant à l'envoyeur d'origine d'avoir la preuve que le message reçu a été authentifié et/ou déchiffré correctement par le receveur.

Les fonctionnalités décrites vont, si elles sont mises en œuvre, satisfaire toutes les exigences de sécurité et mettre en œuvre la non répudiation du récépissé pour l'échange. Cette spécification laisse cependant une pleine souplesse aux utilisateurs pour décider du degré de déploiement de ces caractéristiques de sécurité voulu avec leurs partenaires commerciaux.

2.3.2 Définition des récépissés

Le terme utilisé à la fois pour l'activité fonctionnelle et le message pour reconnaître la livraison d'un échange EDI/EC est "récépissé" ou "récépissé signé". Le premier est utilisé si la reconnaissance est celle d'un échange résultant en un récépissé qui N'EST PAS signé. Le second est utilisé si la reconnaissance est pour un échange résultant en un récépissé qui EST signé.

Le terme de non répudiation de récépissé (NRR) est souvent utilisé en combinaison avec les récépissés. NRR se réfère à un événement légal qui ne se produit que lorsque l'envoyeur original d'un échange a vérifié le récépissé signé revenant du receveur du message, et a vérifié que la valeur de MIC retournée dans la MDN correspond à la valeur antérieurement calculée pour le message d'origine.

La NRR est le mieux établi lorsque le message original et le récépissé font tous deus usage de signatures numériques. Voir dans la Section "Considérations sur la sécurité" certains avertissements concernant la NRR.

Pour des informations sur le traitement et le format des récépissés dans AS2, voir la Section 7.

2.4 Hypothèses

2.4.1 Hypothèses sur le processus EDI/EC

- L'objet chiffré est un échange EDI/EC. Cette spécification suppose qu'un échange EDI/EC normal est l'objet de niveau inférieur qui sera l'objet des services de sécurité. Précisément, dans EDI ANSI X12, cela signifie que tout, entre et inclus, les segments ISA et IEA, est sûr. Dans EDIFACT, cela signifie que tout, entre et inclus, les segments UNA/UNB et UNZ, est sûr. En d'autres termes, les échanges EDI/EC incluant les segments d'enveloppe restent intacts et illisibles durant un transport pleinement sécurisé.

- Les en-têtes d'enveloppe EDI sont chiffrés. Conformément à l'hypothèse ci-dessus, les en-têtes d'enveloppe EDI NE sont PAS visibles dans le paquetage MIME. Afin d'optimiser l'acheminement à partir des réseaux d'EDI commerciaux existants (appelés réseaux à valeur ajoutée (VAN, *Value Added Network*) vers l'Internet, il serait utile de rendre visibles

certaines informations d'enveloppe. Cette spécification ne fournit cependant aucun soutien à cette optimisation.

- Considérations sur la sécurité de X12.58 et de UN/EDIFACT : les organismes de normalisation les plus courants des EDI, ANSI X12 et EDIFACT, ont défini des dispositions internes pour la sécurité. X12.58 est le mécanisme de sécurité pour ANSI X12, et AUTACK assure la sécurité pour EDIFACT. Cette spécification N'IMPOSE PAS l'utilisation ou la non utilisation de ces normes de sécurité. Elles sont toutes deux pleinement compatibles, bien qu'éventuellement redondantes, avec la présente spécification.

2.4.2 Hypothèses de souplesse

- Données chiffrées ou non chiffrées : cette spécification permet un échange de messages EDI/EC dans lesquels les données EDI/EC peuvent être non protégées ou protégées au moyen du chiffrement.

- Données signées ou non signées : cette spécification permet un échange de messages EDI/EC avec ou sans signature numérique de la transmission d'EDI d'origine.

- Utilisation facultative de récépissé : cette spécification permet la transmission de messages EDI/EC avec ou sans demande de réception de récépissé. Une notification de récépissé signé est demandée ; cependant, une valeur de MIC est EXIGÉE au titre du récépissé retourné, sauf lorsque une condition d'erreur sévère empêche le calcul de la valeur de résumé. Dans ce cas exceptionnel, un récépissé signé devrait être retourné avec un message d'erreur qui explique effectivement pourquoi la MIC est absente.

- Utilisation de récépissés synchrones ou asynchrones : en plus d'une demande de récépissé, cette spécification permet la spécification du type de récépissé qui devrait être retourné. Elle prend en charge les récépissés synchrones ou asynchrones dans le format MDN spécifié à la Section 7 de ce document.

- Formatage de sécurité : cette spécification s'appuie sur les lignes directrices établies dans les [RFC3851], [RFC3852] "Spécification de message S/MIME version 3.1 ; syntaxe de message cryptographique".

- Fonction de hachage, choix de résumé de message : quand une signature est utilisée, il est RECOMMANDÉ que l'algorithme de hachage SHA-1 soit utilisé pour tous les messages sortants, et que MD5 et SHA-1 soient tous deux pris en charge pour les messages entrants.

- Résumé des permutations : en résumé, les douze permutations de sécurité suivantes sont possibles dans toute relation commerciale :
  1. L"envoyeur envoie des données non chiffrées et NE demande PAS de récépissé.
  2. L'envoyeur envoie des données non chiffrées et demande un récépissé non signé. Le receveur renvoie le récépissé non signé.
  3. L'envoyeur envoie des données non chiffrées et demande un récépissé signé. Le receveur renvoie le récépissé signé.
  4. L'envoyeur envoie des données chiffrées et NE demande PAS de récépissé.
  5. L'envoyeur envoie des données chiffrées et demande un récépissé non signé. Le receveur renvoie le récépissé non signé.
  6. L'envoyeur envoie des données chiffrées et demande un récépissé signé. Le receveur renvoie le récépissé signé.
  7. L'envoyeur envoie des données signées et NE demande PAS de récépissé, signé ou non signé.
  8. L'envoyeur envoie des données signées et demande un récépissé non signé. Le receveur renvoie le récépissé non signé.
  9. L'envoyeur envoie des données signées et demande un récépissé signé. Le receveur renvoie le récépissé signé.
  10. L'envoyeur envoie des données chiffrées et signées et NE demande PAS de récépissé signé ou non signé.
  11. L'envoyeur envoie des données chiffrées et signées et demande un récépissé non signé. Le receveur renvoie le récépissé non signé.
  12. L'envoyeur envoie des données chiffrées et signées et demande un récépissé signé. Le receveur renvoie le récépissé signé

Les utilisateurs peuvent choisir une des douze possibilités, mais seule la dernière (12), quand un récépissé signé est demandé, offre la pleine suite de caractéristiques de sécurité décrite au paragraphe 2.3.1, "Boucle de transmission sûre".

De plus, les récépissés présentés ci-dessus peuvent être synchrones ou asynchrones selon le type demandé. L'utilisation de récépissés synchrones ou asynchrones ne change pas la nature de la boucle de transmission sûre pour la NRR.

3. RFC référencées et leurs contributions

3.1 HTTP v1.1 [RFC2616]

Ce document spécifie comment les données sont transférées avec HTTP.

3.2 MIME Sécurité multiparties [RFC1847]

Ce document définit la sécurité multiparties pour MIME:multiparties/chiffré et multiparties/signé.

3.3 Multiparties/rapport [RFC3462]

Cette RFC définit l'utilisation du type de contenu multiparties/rapport, sur laquelle s'appuie la MDN de la [RFC3798].

3.4 Contenu EDI [RFC1767]

Cette RFC définit l'utilisation du type de contenu "application" pour ANSI X12 (application/EDI-X12), EDIFACT (application/EDIFACT), et les EDI mutuellement définies (application/EDI-Consent).

3.5 MIME [RFC2045] [RFC2046] et [RFC2049]

Ce sont les normes de base de MIME, sur lesquelles se fondent toutes les RFC relatives à MIME, y compris celle-ci. Les contributions clés incluent la définition de "type de contenu", "sous type", et "multiparties", ainsi que les lignes directrices du codage, qui établissent l'US-ASCII à 7 bits comme jeu de caractères canonique à utiliser dans la messagerie Internet.

3.6 Notification de disposition de message [RFC3798]

Cette RFC de l'Internet définit comment une MDN est demandée, le format et la syntaxe de la MDN. La MDN est la base sur laquelle les récépissés et les récépissés signés sont définis dans la présente spécification.

3.7 Spécification de message S/MIME v3.1 et syntaxe de message cryptographique (CMS) [RFC3851] et [RFC3852]

Ces spécifications décrivent comment S/MIME porte les objets de CMS.

3.8 Types de supports XML [RFC3023]

Cette RFC définit l'utilisation du type de contenu "application" pour XML (application/xml).

4. Structure d'un message AS2

4.1 Introduction

La structure de base d'un message AS2 consiste en un format MIME à l'intérieur d'un message HTTP avec quelques en-têtes AS2 spécifiques supplémentaires. Les structures sont décrites de façon hiérarchique dans les termes selon lesquels les RFC sont appliquées pour former la structure spécifique. Les détails de la façon de coder conformément à toutes les RFC impliquées sont donnés dans les RFC référencées. Toute différence entre les mises en œuvre AS2 et les RFC sont mentionnées dans les paragraphes ci-dessous.

4.2 Structure d'un message Internet EDI MIME

Pas de chiffrement, pas de signature
-RFC2616/2045
-RFC1767/RFC3023 (application/EDIxxxx ou /xml)

Pas de chiffrement, signature
-RFC2616/2045

-RFC1847 (multipart/signé)
-RFC1767/RFC3023 (application/EDIxxxx ou /xml)
-RFC3851 (application/pkcs7-signature)

Chiffrement, pas de signature
-RFC2616/2045
-RFC3851 (application/pkcs7-mime)
-RFC1767/RFC3023 (application/EDIxxxx ou /xml)(chiffré)

Chiffrement, signature
-RFC2616/2045
-RFC3851 (application/pkcs7-mime)
-RFC1847 (multipart/signé)(chiffré)
-RFC1767/RFC3023 (application/EDIxxxx ou /xml)(chiffré)
-RFC3851 (application/pkcs7-signature)(chiffré)

MDN sur HTTP, pas de signature
-RFC2616/2045
-RFC3798 (message/disposition-notification)

MDN sur HTTP, signature
-RFC2616/2045
-RFC1847 (multipart/signé)
-RFC3798 (message/disposition-notification)
-RFC3851 (application/pkcs7-signature)

MDN sur SMTP, pas de signature
MDN sur SMTP, signature
Voir la norme des EDI sur SMTP [RFC3335].

Bien que tous les types de contenu MIME DEVRAIENT être pris en charge, les types de contenu MIME suivants DOIVENT être pris en charge :
Content-type : multipart/signé
Content-Type : multipart/rapport
Content-type : message/disposition-notification
Content-Type : application/PKCS7-signature
Content-Type : application/PKCS7-mime
Content-Type : application/EDI-X12
Content-Type : application/EDIFACT
Content-Type : application/edi-consent
Content-Type : application/XML

5. Considérations sur HTTP

5.1 Envoi d'EDI dans des demandes HTTP POST

La ligne de demande va avoir la forme : "POST Request-URI HTTP/1.1", avec des espaces et suivie par un CRLF. L'URI de demande est normalement échangé hors bande, au titre de l'établissement d'un accord bilatéral entre partenaires commerciaux. Les applications DEVRAIENT être prêtes à traiter une réponse initiale contenant une déclaration indiquant le besoin d'authentification des types usuels utilisés pour autoriser l'accès à l'URI de demande ([RFC2616], paragraphe 10.4.2 et ailleurs).

La ligne de demande est suivie par les en-têtes d'entité qui spécifient la longueur du contenu ([RFC2616], paragraphe 14.14) et le type de contenu ([RFC2616], paragraphe 14.18). L'en-tête de demande Host ([RFC2616], Section 9 et paragraphe 14.23) est aussi inclus.

Quand on utilise la sécurité de couche Transport [RFC2246] ou SSLv3, l'URI de demande DEVRAIT indiquer la valeur de schéma appropriée, HTTPS. Généralement, seul un corps de message multipart/signé sera envoyé en utilisant TLS, car les corps de message chiffrés vont être redondants. Cependant, les corps de message chiffrés ne sont pas interdits.

Le système AS2 receveur PEUT se déconnecter du système AS2 envoyeur avant d'achever la réception de l'entité entière si

il détermine que l'entité envoyée est trop grosse pour être traitée.

Pour HTTP version 1.1, les connexions TCP persistantes sont par défaut, ([RFC2616] paragraphes 8.1.2, 8.2, et 19.7.1). Un certain nombre d'autres différences existent parce que HTTP ne se conforme pas à MIME [RFC2045] comme utilisé dans le transport SMTP. Les différences pertinentes sont résumées ci-dessous.

5.2. En-têtes et opérations MIME non utilisées

5.2.1 Codage de transfert de contenu non utilisé dans le transport HTTP

HTTP peut traiter les données binaires et il n'est donc pas besoin d'utiliser le codage de transfert de contenu de MIME [RFC2045]. Cette différence est expliquée au paragraphe 19.4.5 de la [RFC2616]. Cependant, une valeur de codage de transfert de contenu de "binary" ou "8-bit" est permise mais pas exigée. L'absence de cet en-tête NE DOIT PAS résulter en un échec de transaction. Le codage de transfert de contenu de parties de corps MIME au sein du corps de message AS2 est aussi permis.

5.2.2 Corps de message

Au paragraphe 3.7.2 de la [RFC2616], il est explicitement noté que les multiparties DOIVENT avoir des épilogues nuls.

Au paragraphe 5.4.1 de la [RFC3335], des options de traitement de grands fichiers sont discutées pour le transport SMTP. Pour HTTP, les grands fichiers DEVRAIENT être traités correctement par la couche TCP. Cependant, les paragraphes 3.5 et 3.6 de la [RFC2616] discutent certaines options pour compresser ou tronçonner des entités à transférer. Le paragraphe 8.1.2.2 de la [RFC2616] discute d'une option de traitement en parallèle qui est utile pour segmenter de grosses quantités de données.

5.3 Modification des en-tête ou paramètres MIME ou autres utilisés

5.3.1 Longueur de contenu

L'utilisation de l'en-tête Longueur de contenu DOIT suivre les lignes directrices de la [RFC2616], spécifiquement des paragraphes 4.4 et 14.13.

5.3.2 Receveur final et original

Les valeurs de receveur final et d'origine DEVRAIENT être la même valeur. Ces valeurs NE DOIVENT PAS être des alias ou des listes de diffusion.

5.3.3 Identifiant de message et identifiant de message original

Message-Id et Original-Message-Id sont formatés comme défini au paragraphe 3.6.4 de la [RFC2822] :

"<" id-gauche "@" id-droite ">"

La longueur de l'identifiant de message a un maximum de 998 caractères. Pour une rétro compatibilité maximum, la longueur de l'identifiant de message DEVRAIT être de 255 caractères ou moins. L'identifiant de message DEVRAIT être unique au monde, et "id-droite" DEVRAIT être quelque chose d'unique pour l'environnement de l'hôte envoyeur (par exemple, un nom d'hôte).

Lors de l'envoi d'un message, toujours inclure les crochets angulaires. Les crochets angulaires ne font pas partie de la valeur d'identifiant de message. Pour une rétro compatibilité maximum, lors de la réception d'un message, ne pas vérifier les crochets angulaires. Lors de la création de l'en-tête "Original-Message-Id" dans une MDN, utiliser toujours la syntaxe exacte comme reçue dans le message d'origine ; ne pas supprimer ni ajouter de crochets angulaires.

5.3.4 En-tête Host

Le champ d'en-tête de demande "host" DOIT être inclus dans la demande POST faite quand on envoie des données d'affaires. Ce champ est destiné à permettre à une adresse IP de serveur de servir plusieurs noms d'hôtes, et éventuellement de conserver les adresses IP. Voir les paragraphes 14.23 et 19.5.1 de la [RFC2616].

5.4 Codes d'état de réponse HTTP

Les codes d'état retournent l'état des opérations HTTP. Par exemple, le code d'état 401, joint à l'en-tête WWW-Authenticate, est utilisé pour inviter le client à répéter la demande avec un en-tête Authorization. D'autres codes d'état explicites sont documentés au paragraphe 6.1.1, et dans la Section 10 de la [RFC2616].

Pour les erreurs dans l'URI de demande, 400 ("Mauvaise demande"), 404 ("Introuvable") et des codes similaires sont des codes d'état appropriés. Ces codes et leur sémantique sont spécifiés par la [RFC2616]. Un examen attentif de ces codes et de leur sémantique devrait être fait avant de mettre en œuvre toute fonction de répétition. On NE DEVRAIT PAS faire de répétition si l'erreur n'est pas temporaire ou si les réessais sont explicitement déconseillés.

5.5 Récupération d'erreur HTTP

Si le client HTTP échoue à lire les données de la réponse du serveur HTTP, l'opération POST avec un contenu identique, incluant le même identifiant de message, DEVRAIT être répétée, si la condition est temporaire.

L'identifiant de message sur une opération POST ne peut être réutilisé que si tout le contenu (incluant la date d'origine) est identique.

Les détails du processus de réessai (y compris les intervalles de pause, le nombre d'essais à tenter, et les temporisations entre les essais) dépendent de la mise en œuvre. Ces réglages sont choisis au titre de l'accord avec le partenaire commercial.

Les serveurs DEVRAIT être prêts à recevoir un POST avec un identifiant de message répété. Le corps de réponse MIME précédemment envoyé DEVRAIT être renvoyé, y compris la MDN et les autres parties MIME.

6. En-têtes HTTP supplémentaires spécifiques de AS2

Les en-têtes qui suivent sont à inclure dans tous les messages AS2 et toutes les MDN AS2, sauf pour les MDN asynchrones qui sont envoyées avec SMTP et suivent la sémantique AS1 [RFC3335].

6.1 En-tête de version AS2

Pour faciliter la rétro compatibilité, AS2 comporte un en-tête "version" :

AS2-Version: 1.0 – Utilisé dans toutes les mises en œuvre de la présente spécification. 1.x sera interprété comme 1.0 par toutes les mises en œuvre qui ont l'en-tête "AS2 Version: 1.0". C'est-à-dire, seul le chiffre de plus fort poids est utilisé comme identifiant de version pour celles qui ne mettent pas en œuvre la fonctionnalité supplémentaire non spécifiée par AS2. "AS2-Version: 1.0 à 1.9" PEUT être utilisé. Toutes les mises en œuvre DOIVENT interpréter "1.0 à 1.9" comme mettant en œuvre la présente spécification. Cependant, une mise en œuvre PEUT étendre la présente spécification avec des fonctions supplémentaires en spécifiant des versions 1.1 à 1.9. Si ce mécanisme est utilisé, la fonction supplémentaire DOIT être complètement transparente aux mises en œuvre qui ont la désignation "AS2-Version: 1.0".

AS2-Version: 1.1 – Désigne les mises en œuvre qui prennent en charge la compression définie par la [RFC3274].

Les systèmes receveurs NE DOIVENT PAS échouer à cause de l'absence de l'en-tête AS2-Version. Son absence indiquerait que le message provient d'une mise en œuvre fondée sur une version antérieure à la présente spécification.

6.2 Identifiant de système AS2

Pour aider le système receveur à identifier le système envoyeur, on utilise les en-têtes AS2-From et AS2-To.

AS2-From: < nom AS2 >
AS2-To: < nom AS2 >

Ces en-têtes AS2 contiennent des valeurs textuelles, comme décrit ci-dessous, qui identifient l'envoyeur/receveur d'un échange de données. Leur valeur peut être spécifique d'une entreprise, comme les numéros du système de numérotation universel des données (DUNS, *Data Universal Numbering System*) ou elle peut être simplement une chaîne d'identification sur laquelle les partenaires commerciaux se sont mis d'accord.

AS2-text = "!" / %d35-91 / %d93-126 ; caractères ASCII imprimables sauf guillemets (%d34) ou barre oblique inverse

(%d92)

AS2-qtext = AS2-text / SP ; ne permet d'espace que dans le texte entre guillemets

AS2-quoted-pair = "\" DQUOTE / ; \" ou "\" "\" ; \\

AS2-quoted-name = DQUOTE 1*128(AS2-qtext / AS2-quoted-pair) DQUOTE

AS2-atomic-name = 1*128AS2-text

AS2-name = AS2-atomic-name / AS2-quoted-name

La valeur de l'en-tête AS2-From et la valeur de l'en-tête AS2-To DOIVENT être un nom AS2, DOIVENT être constituées de 1 à 128 caractères ASCII imprimables, et NE DOIVENT PAS revenir à la ligne. La valeur de chacun de ces en-têtes est sensible à la casse. Les définitions de chaînes données ci-dessus sont en format ABNF [RFC2234].

Le "AS2-quoted-name" DEVRAIT n'être utilisé que si le "AS2-name" ne se conforme pas à "AS2-atomic-name".

Les champs AS2-To et AS2-From DOIVENT être présents dans tous les messages AS2 et les MDN AS2 asynchrones ou synchrones, sauf pour les MDN asynchrones qui sont envoyées avec SMTP.

Le nom AS2 pour l'en-tête AS2-To dans une réponse ou MDN DOIT correspondre au nom AS2 de l'en-tête AS2-From dans le message de demande correspondant. De même, le nom AS2 pour l'en-tête AS2-From dans une réponse ou MDN DOIT correspondre au nom AS2 de l'en-tête AS2-To dans le message de demande AS2 correspondant.

Le système envoyeur peut choisir de limiter les valeurs textuelles possibles de AS2-To/AS2-From mais NE DOIT PAS les excéder. Le système receveur DOIT ne faire aucune restriction sur les valeurs textuelles et DEVRAIT traiter toutes les mises en œuvre possibles. Cependant, les développeurs doivent être conscients que les produits AS2 plus anciens peuvent ne pas suivre cette convention. Des accords entre partenaires commerciaux devraient être conclus pour s'assurer que les produits plus anciens prennent en charge les identifiants de système qu'ils utilisent.

Il n'y a pas de réponse obligée à une demande de client qui contient des valeurs invalides ou inconnues d'en-tête AS2-From ou AS2-To. Le système AS2 receveur PEUT retourner une MDN non signée avec une explication de l'erreur, si le système envoyeur avait demandé une MDN.

7. Structure et traitement d'un message de MDN

7.1 Introduction

Pour prendre en charge la non répudiation de récépissé, un récépissé signé, sur la base de la signature numérique d'une notification de disposition de message, est à mettre en œuvre par l'UA d'un partenaire commercial receveur. La notification de disposition de message, spécifiée par la [RFC3798], est signée numériquement par un partenaire commercial receveur au titre d'un message multipart/signé MIME.

La prise en charge des récépissés signés suivante est EXIGÉE :
1. Capacité de créer un multipart/rapport; où le type de rapport = disposition-notification.
2. Capacité de calculer une vérification d'intégrité de message (MIC) sur le message reçu. La valeur de MIC calculée sera retournée à l'envoyeur du message au sein du récépissé signé.
3. Capacité de créer un contenu multipart/signé avec la notification de disposition de message comme première partie de corps, et la signature comme seconde partie de corps.
4. Capacité de retourner le récépissé signé au partenaire commercial envoyeur.
5. Capacité de retourner un récépissé synchrone ou asynchrone lorsque l'envoyeur le demande.

Le récépissé signé est utilisé pour notifier au partenaire commercial envoyeur qui a demandé le récépissé signé que :
1. Le partenaire commercial receveur accuse réception de l'échange EC envoyé.
2. Si le message envoyé était signé, le partenaire commercial receveur a alors authentifié l'envoyeur de l'échange EC.
3. Si le message envoyé était signé, le partenaire commercial receveur a vérifié l'intégrité de l'échange EC envoyé.

Sans considérer si l'échange EDI/EC était ou non envoyé en format S/MIME, l'UA du partenaire commercial receveur DOIT fournir le traitement de base suivant :
1. Si l'échange EDI/EC envoyé est chiffré, la clé chiffrée symétrique et la valeur d'initialisation (si applicable) sont

déchiffrées en utilisant la clé privée du receveur.
2. La clé de chiffrement symétrique déchiffrée est alors utilisée pour déchiffrer l'échange EDI/EC.
3. Le partenaire commercial receveur authentifie les signatures dans un message en utilisant la clé publique de l'envoyeur. L'algorithme d'authentification effectue ce qui suit :
   a. La vérification d'intégrité de message (MIC ou résumé de message) est déchiffrée en utilisant la clé publique de l'envoyeur.
   b. Une MIC sur le contenu signé (en-tête MIME et objet EDI codé, conformément à la [RFC1767]) du message reçu est calculée en utilisant la même fonction de hachage unidirectionnelle qu'a utilisé le partenaire commercial envoyeur.
   c. La MIC extraite du message qui a été envoyé et la MIC calculée en utilisant la même fonction de hachage unidirectionnelle qu'a utilisé le partenaire commercial envoyeur sont comparées pour égalité.
4. Le partenaire commercial receveur formate la MDN et règle la MIC calculée dans le champ d'extension "Received-content-MIC".
5. Le partenaire commercial receveur crée un message MIME multipart/signé MIME conformément à la [RFC1847].
6. La MDN est la première partie du message multipart/signé, et la signature numérique est crée sur cette MDN, incluant les en-têtes MIME.
7. La seconde partie du message multipart/signé contient la signature numérique. L'option "protocol" spécifiée dans la seconde partie du multipart/signé est comme suit : S/MIME: protocol = "application/pkcs-7-signature"
8. Les informations de signature sont formatées conformément aux spécifications S/MIME.

L'échange EC et l'en-tête de contenu EDI MIME de la [RFC1767] peuvent en fait faire partie d'un type MIME multi-parties. Quand l'échange EDI fait partie d'un type de contenu MIME multiparties, la MIC DOIT être calculée sur le contenu multi-parties entier, incluant les en-têtes MIME.

La MDN signée, quand elle est reçue par l'envoyeur de l'échange EDI, peut être utilisée comme suit par l'envoyeur :
- Comme reconnaissance que l'échange EDI envoyé a été livré et reconnu par le partenaire commercial receveur. Le receveur fait cela en retournant l'identifiant du message d'origine du message envoyé dans la portion MDN du récépissé signé.
- Comme reconnaissance que l'intégrité de l'échange EDI a été vérifiée par le partenaire commercial receveur. Le receveur fait cela en retournant la MIC calculée de l'échange EC reçu (et des en-têtes MIME de la RFC1767) dans le champ "Received-content-MIC" de la MDN signée.
- Comme reconnaissance que le partenaire commercial receveur a authentifié l'envoyeur de l'échange EDI.
- Comme non répudiation du récépissé quand la MDN signée est bien vérifiée par l'envoyeur avec la clé publique du partenaire commercial receveur et que la valeur de MIC retournée dans la MDN est la même que le résumé du message original.

7.2 MDN synchrones et asynchrones

La MDN AS2 existe sous deux formes : synchrones et asynchrones.

La MDN AS2 synchrone est envoyée comme réponse HTTP à un POST HTTP ou une réponse HTTPS à un POST HTTPS. Cette forme de MDN AS2 est appelée synchrone parce que la MDN AS2 est retournée au générateur du POST sur la même connexion TCP/IP.

La MDN AS2 asynchrone est envoyée sur une connexion TCP/IP HTTP, HTTPS, ou SMTP séparée. Logiquement, la MDN AS2 asynchrone est une réponse à un message AS2. Cependant, à la couche de protocole de transfert, en supposant qu'aucun traitement HTTP en parallèle n'est utilisé, la MDN AS2 asynchrone est livrée sur une connexion TCP/IP unique, distincte de celle utilisée pour livrer le message AS2 d'origine. Quand on traite une demande asynchrone, la réponse HTTP DOIT être renvoyée avant que la MDN soit traitée et envoyée sur la connexion séparée.

Quand une MDN AS2 asynchrone est demandée par l'envoyeur d'un message AS2, la réponse synchrone HTTP ou HTTPS retournée à l'envoyeur avant de terminer la connexion DOIT être une réponse de couche de transfert qui indique la réussite ou l'échec du transfert de données. Le format d'une telle réponse synchrone PEUT être le même que celui de la réponse retournée quand aucune MDN AS2 n'est demandée.
Le diagramme qui suit illustre les variétés synchrone et asynchrone de la livraison de MDN AS2 en utilisant HTTP :

MDN AS2 synchrone

```
[Peer1] -----( connecte )----> [Peer2]
[Peer1] -----( envoie )-------> [Peer2]   [Demande HTTP [Message AS2]]
[Peer1] <----( reçoit )--------- [Peer2]   [Réponse HTTP [MDN AS2]]
```

MDN AS2 asynchrone

[Peer1] ----(connecte)----> [Peer2]
[Peer1] -----(envoie)------> [Peer2] [Demande HTTP [Message AS2]]
[Peer1] <---(reçoit)--------- [Peer2] [Réponse HTTP]

[Peer1]*<---(connecte)----- [Peer2]
[Peer1] <--- (envoie)-------- [Peer2] [Demande HTTP [MDN AS2]]
[Peer1] -----(reçoit)--------> [Peer2] [Réponse HTTP]

* Note : une MDN AS2 peut être dirigée sur un hôte différent de celui de l'envoyeur du message AS2. Elle peut utiliser un protocole de transfert différent de celui utilisé pour envoyer le message AS2 d'origine.

L'avantage de la MDN synchrone est qu'elle peut fournir à l'envoyeur du message AS2 une confirmation vérifiable de la livraison du message dans un flux logique synchrone. Cependant, si le message est relativement grand, le temps requis pour traiter ce message et retourner une MDN AS2 à l'envoyeur sur la même connexion TCP/IP peut excéder le délai maximum configuré permis pour une connexion IP.

L'avantage de la MDN asynchrone est qu'elle fournit un retour rapide d'une réponse de couche transfert du receveur, confirmant la réception des données, n'exigeant donc pas qu'une connexion TCP/IP reste nécessairement ouverte pendant très longtemps. Cependant, cette conception exige que la MDN AS2 asynchrone contienne assez d'informations pour identifier de façon univoque le message original afin que, quand il est reçu par le générateur du message AS2, le statut du message AS2 d'origine puisse être correctement mis à jour sur la base du contenu de la MDN AS2.

Les MDN synchrones ou asynchrones HTTP ou HTTPS sont traitées conformément aux exigences de la présente spécification.

Cependant, les MDN SMTP sont formatées en accord avec les exigences de la [RFC3335].

7.3 Demande d'un récépissé signé

Les notifications de disposition de message sont demandées conformément à la [RFC3798]. Une demande que l'agent d'utilisateur receveur produise une notification de disposition de message est faite en plaçant l'en-tête suivant dans le message à envoyer :

MDN-request-header = "Disposition-notification-to" ":" adresse de messagerie

L'exemple suivant est pour demander une MDN :

Disposition-notification-to: xxx@exemple.com

Cette syntaxe est un résidu de l'utilisation des MDN avec le transfert SMTP. Parce que la présente spécification ajuste la fonctionnalité de SMTP à HTTP tout en conservant autant que possible des fonctionnalités de la [RFC3335], l'adresse de messagerie DOIT être présente. Le champ "adresse de messagerie" est spécifié comme une adresse localpart@domaine [addr-spec] de la [RFC2822]. Cependant, l'adresse n'est pas utilisée pour identifier où retourner la MDN. Les applications receveuses DOIVENT ignorer la valeur et NE DOIVENT PAS se plaindre de violations de la syntaxe d'adresse de la RFC2822.

Quand il demande des récépissés fondés sur la MDN, le générateur fournit des en-têtes d'extension supplémentaires qui précèdent le corps de message. Ces en-têtes "étiquettes" sont comme suit :

Un en-tête Message-ID est ajouté pour prendre en charge la réconciliation de message, afin qu'une valeur de "Original-Message-Id" puisse être retournée dans la partie de corps de la MDN. D'autres en-têtes, en particulier "Subject" et "Date", DEVRAIENT être fournis ; les valeurs de ces en-têtes sont souvent mentionnées dans la section lisible par l'homme de la MDN pour aider à identifier le message original.

Les MDN seront retournées dans la réponse HTTP quand c'est demandé, sauf si un retour asynchrone est demandé.

Pour demander une notification de disposition de message asynchrone, l'en-tête suivant est placé dans le message envoyé :

Receipt-Delivery-Option: URL de retour

Voici un exemple de demande que la MDN soit asynchrone :

Receipt-Delivery-Option: http://www.exemple.com/chemin

La syntaxe de "Receipt-delivery-option" permet que l'URL de retour utilise des schémas autres que HTTP utilisant la méthode POST.

La chaîne "receipt-delivery-option: return-url" indique l'URL à utiliser pour une MDN asynchrone. Cet en-tête N'EST PAS présent si le récépissé doit être synchrone. La valeur de "mail" dans "Disposition-notification-to" n'est pas utilisée dans la présente spécification parce qu'elle a été limitée aux adresses de la RFC 2822 ; l'en-tête d'extension "Receipt-delivery-option" a été introduit pour fournir un URL pour la MDN retournée par plusieurs options de transfert.

La valeur de "receipt-delivery-option" DOIT être un URL indiquant la destination du transport de livraison pour la réception.

Exemple de demande de MDN asynchrone via un transport HTTP :

Receipt-delivery-option: http://www.exemple.com

Exemple de demande de MDN asynchrone via un transport HTTP/S :

Receipt-delivery-option: https://www.exemple.com

Exemple de demande de MDN asynchrone via un transport SMTP :

Receipt-delivery-option: mailto:as2@exemple.com

Pour plus d'informations sur la demande de MDN SMTP, se reporter à la [RFC3335].

Finalement, l'en-tête "Disposition-notification-options" identifie les caractéristiques de la notification de disposition de message comme dans la [RFC3798]. Les plus importantes de ces options sont pour indiquer les options de signature pour la MDN, comme dans l'exemple suivant :

Disposition-notification-options:
signed-receipt-protocol=optional,pkcs7-signature;
signed-receipt-micalg=optional,sha1,md5

Pour les options de signature, on prend en considération la syntaxe de disposition-notification-options :

Disposition-notification-options = "Disposition-Notification-Options" ":" disposition-notification-parameters

où
disposition-notification-parameters = paramètre *(";" paramètre)

où
paramètre = attribut "=" importance ", " 1#value"

où
importance = "requis" | "facultatif"

De sorte que la chaîne "Disposition-notification-options" pourrait être :

signed-receipt-protocol=facultatif,<symbole de protocole>;
signed-receipt-micalg=facultatif,<micalg 1>,<micalg 2>,...;

La valeur actuellement utilisée pour <symbole de protocole> est "pkcs7-signature" pour le format de signature S/MIME détachée.

Les valeurs actuellement prises en charge pour l'algorithme de MIC <micalg> sont :

Algorithme	Valeur utilisée
SHA-1	sha1
MD5	md5

La sémantique des paramètres "signed-receipt-protocol" et "signed-receipt-micalg" est la suivante :

1. Le paramètre "signed-receipt-protocol" est utilisé pour demander un récépissé signé au partenaire commercial receveur. Le paramètre "signed-receipt-protocol" spécifie aussi le format sous lequel le récépissé signé DEVRAIT être retourné au demandeur. Le paramètre "signed-receipt-micalg" est une liste d'algorithmes de MIC préférés par le demandeur à utiliser pour la signature du récépissé retourné. La liste des algorithmes de MIC DEVRAIT être respectée par le receveur de gauche à droite. Les paramètres d'option "signed-receipt-protocol" et "signed-receipt-micalg" sont tous deux EXIGÉS dans une demande de récépissé signé. L'absence de "Receipt-Delivery-Option" indique qu'un récépissé est de nature synchrone. La présence de "Receipt-Delivery-Option: return-url" indique qu'un récépissé asynchrone est demandé et DEVRAIT être envoyé à "return-url".

2. L'attribut "Importance" de "facultatif" est défini au paragraphe 2.2 de la [RFC3798], et a la signification suivante : les paramètres avec une importance de "facultatif" permettent à un UA qui ne comprend par le paramètre d'option particulier de générer quand même une MDN en réponse à une demande de MDN. Un UA qui ne comprend pas le paramètre "signed-receipt-protocol" ou "signed-receipt-micalg" ne va évidemment pas retourner un récépissé signé. L'importance de "facultatif" est utilisée pour les paramètres de récépissé signé parce qu'il est RECOMMANDÉ qu'une MDN soit retournée au partenaire commercial demandeur même si le receveur n'a pas pu le signer. La MDN retournée va contenir des informations sur la disposition du message et la raison pour laquelle la MDN n'a pas pu être signée. Voir le champ "Disposition" au paragraphe 7.5 pour plus d'informations. Dans une relation d'EDI commerciale, si un récépissé signé est attendu et n'est pas retourné, il appartient aux partenaires commerciaux de juger de la validité de la transaction. En général, si un récépissé signé est demandé dans la relation commerciale et s'il n'est pas reçu, la transaction ne sera probablement pas considérée comme valide.

7.3.1 Considérations sur les récépissés signés

La méthode utilisée pour demander un récépissé ou un récépissé signé est définie dans la RFC 3798, "Format extensible de message pour notifications de disposition de message".

Les "règles" sont les suivantes :
1. Quand un récépissé est demandé, spécifiant explicitement que le récépissé soit signé, le récépissé DOIT alors être retourné avec une signature.
2. Quand un récépissé est demandé, spécifiant explicitement que le récépissé soit signé, mais que le receveur ne peut pas prendre en charge le format de protocole demandé ou les algorithmes de MIC demandés, un récépissé signé ou non signé DEVRAIT être retourné.
3. Quand une signature n'est pas explicitement demandée, ou si le paramètre de demande de récépissé signé n'est pas reconnu par l'UA, aucun récépissé, un récépissé non signé, ou un récépissé signé, PEUT être retourné par le receveur.

Note : pour les EDI Internet, il est RECOMMANDÉ que quand une signature n'est pas explicitement demandée, ou si des paramètres ne sont pas reconnus, l'UA renvoie, au minimum, un récépissé non signé. Si, cependant, un récépissé signé a toujours été retourné selon la politique, qu'il soit demandé ou non, tout faux récépissé non signé peut être répudié.

Quand est faite une demande de récépissé signé, mais qu'il y a une erreur de traitement dans le contenu du message, un récépissé signé DOIT quand même être retourné. La demande de récépissé signé DEVRA quand même être honorée, bien que la transaction elle-même puisse n'être pas valide. La raison pour laquelle le contenu n'a pas pu être traité DOIT être mise dans le champ "disposition".

Quand est faite une demande de récépissé signé, le champ "Received-content-MIC" DOIT toujours être retourné au demandeur (sauf quand la corruption empêche le calcul du résumé en accord avec la spécification suivante). Le champ "Received-content-MIC" DOIT être calculé comme suit :
- Pour tout message signé, la MIC à retourner est calculée sur l'en-tête et le contenu MIME des RFC1767/RFC3023. La canonisation sur les en-têtes MIME DOIT être effectuée avant le calcul de la MIC, car l'envoyeur qui demande le récépissé signé a aussi EXIGÉ la canonisation.
- Pour les messages chiffrés non signés, la MIC à retourner est calculée sur l'en-tête et contenu MIME déchiffré des RFC1767/RFC3023. Le contenu après déchiffrement DOIT être canonisé avant le calcul de la MIC.
- Pour les messages non signés non chiffrés, la MIC DOIT être calculée sur le contenu du message sans les en-têtes MIME ou tout autre en-tête de la RFC2822, car ceux-ci sont parfois altérés ou réordonnés par les agents de transport de messagerie (MTA, *Mail Transport Agent*).

7.4 Format et valeurs de notification de disposition de message

Ce paragraphe définit le format de la notification de disposition de message AS2 (AS2-MDN, *AS2 Message Disposition Notification*).

7.4.1 Formats généraux de MDN AS2

La AS2-MDN suit la spécification de MDN [RFC3798] sauf comme noté dans ce paragraphe. Les définitions d'ABNF modifié dans ce document utilisent le caractère barre verticale , '|', pour noter une construction logique "OU". Cet usage suit la [RFC2616]. Les entités HTTP désignées ci-dessous ne sont pas autrement définies dans ce document. Se reporter à la [RFC2616] pour les définitions complètes des entités HTTP. Le format de AS2-MDN est :

AS2-MDN = AS2-sync-MDN | AS2-async-http-MDN | AS2-async-smtp-MDN

AS2-sync-MDN = Ligne d'état *((en-tête général | en-tête de réponse | en-tête d'entité) CRLF)
CRLF
corps AS2-MDN

Ligne d'état = HTTP-Version SP Code d'état SP Phrase de cause CRLF

AS2-async-http-MDN = Ligne de demande
*((en-tête général | en-tête de réponse | en-tête d'entité) CRLF)
CRLF
corps AS2-MDN

Ligne de demande = Méthode SP URI de demande SP Version HTTP CRLF

AS2-async-smtp-MDN = *((en-tête général | en-tête de réponse | en-tête d'entité) CRLF)
CRLF
corps AS2-MDN

corps AS2-MDN = corps AS2-MDN-signé | corps AS2-MDN-non-signé

7.4.2 Construction de MDN AS2

Le corps AS2-MDN est formaté comme un multipart/rapport MIME avec un type de rapport de "disposition-notification". Quand le message est non signé, les en-têtes d'entité de couche transfert ("les plus externes") de la AS2-MDN contiennent l'en-tête content-type qui spécifie un type de contenu de "multipart/report" et des paramètres indiquant le type de rapport, et la valeur de la limite multiparties la plus externe.

Quand la AS2-MDN est signée, les en-têtes d'entité de couche transfert ("les plus externes") de la AS2-MDN contiennent un en-tête content-type qui spécifie un type de contenu de "multipart/signed" et des paramètres qui indiquent l'algorithme utilisé pour calculer le résumé de message, le protocole de formatage de signature (par exemple, pkcs7-signature) et la valeur de la limite multiparties la plus externe. La première partie du message MIME multiparties/signé est un multiparties/rapport MIME incorporé de type "disposition-notification". La seconde partie du message multiparties/signé contient un message MIME application/pkcs7-signature.

La première partie du multiparties/rapport MIME est une portion "lisible par l'homme" qui contient une description générale de la disposition du message. La seconde partie du multiparties/rapport MIME est une portion "lisible par la machine" qui est définie comme :

AS2-disposition-notification-content =
[reporting-ua-field CRLF]
[mdn-gateway-field CRLF]
final-recipient-field CRLF
[original-message-id-field CRLF]
AS2-disposition-field CRLF
*(failure-field CRLF)
*(error-field CRLF)
*(warning-field CRLF)

*(extension-field CRLF)
[AS2-received-content-MIC-field CRLF]

7.4.3 Champs de MDN AS2

Les règles pour construire le contenu AS2-disposition-notification sont identiques à celles de disposition-notification-content fournies à la Section 7 de la [RFC3798], sauf que le champ disposition de la RFC3798 a été remplacé par le champ AS2-disposition et que le champ AS2-received-content-MIC a été ajouté. Les différences entre le champ Disposition de la RFC3798 et le champ AS2-disposition sont décrites ci-dessous. Lorsque il y a des différences entre ce document et la RFC3798, les noms des entités ont été changés en ajoutant devant "AS2-". Les entités qui ne diffèrent pas de celles de la RFC3798 ne sont pas nécessairement redéfinies dans le présent document ; se référer à la Section 7 de la RFC3798, "Grammaire collectée", pour la grammaire originale.

AS2-disposition-field = "Disposition" ":" disposition-mode ";" AS2-disposition-type ['/' AS2-disposition-modifier]

disposition-mode = action-mode "/" sending-mode

action-mode = "manual-action" | "automatic-action"

sending-mode = "MDN-sent-manually" | "MDN-sent-automatically"

AS2-disposition-type = "processed" | "failed"

AS2-disposition-modifier = ("error" | "warning") | AS2-disposition-modifier-extension

AS2-disposition-modifier-extension =
"error: authentication-failed" |
"error: decompression-failed" |
"error: decryption-failed" |
"error: insufficient-message-security" |
"error: integrity-check-failed" |
"error: unexpected-processing-error" |
"warning: " AS2-MDN-warning-description |
"failure: " AS2-MDN-failure-description

AS2-MDN-warning-description = *(TEXT)

AS2-MDN-failure-description = *(TEXT)

AS2-received-content-MIC-field = "Received-content-MIC" ":" encoded-message-digest "," digest-alg-id CRLF

encoded-message-digest = 1*('A'-'Z' | 'a'-'z' | '0'-'9' | '/' | '+' | '=') (i.e. base64(message-digest))

digest-alg-id = "sha1" | "md5"

"Insufficient-message-security" et "decompression-failed" sont de nouveaux codes d'erreur qui ne sont pas mentionnés dans l'AS1 de la RFC 3335, et peuvent n'être pas compatibles avec les mises en œuvre antérieures de AS2.

Le champ d'extension "Received-content-MIC" est établi quand l'intégrité du message reçu est vérifiée. La MIC est le résumé de message codé en base64 calculé sur le message reçu avec une fonction de hachage. Ce champ est exigé pour les récépissés signés mais facultatif pour les récépissés non signés. Voir au paragraphe 7.3.1 les détails définissant le contenu spécifique sur lequel le résumé de message doit être calculé.

Pour les messages signés, l'algorithme utilisé pour calculer la MIC DOIT être le même que celui utilisé sur le message qui a été signé. Si le message n'est pas signé, l'algorithme SHA-1 DEVRAIT alors être utilisé. Ce champ n'est établi que lorsque le contenu du message est traité avec succès. Ce champ est utilisé en conjonction avec la signature du receveur sur la MDN afin que l'envoyeur puisse vérifier la non répudiation du récépissé.

Les noms de champs AS2-MDN (par exemple, "Disposition:", "Final-Recipient:") sont insensibles à la casse (cf. paragraphe 3.1.1 de la [RFC3798]). Les valeurs AS2-MDN action-modes, sending-modes, AS2-disposition-types, et AS2-disposition-modifier, qui sont définies ci-dessus, et les valeurs fournies par l'utilisateur *(TEXT) sont aussi insensibles à la casse. Les mises en œuvre de AS2 NE DOIVENT PAS faire d'hypothèses sur les valeurs fournies pour AS2-MDN-warning-

description ou AS2-MDN-failure-description, ou pour les valeurs de toute erreur (facultative) avertissement, ou champ d'échec.

7.4.4 Notes supplémentaires de programmation de MDN AS2

- À la différence de SMTP, pour les transactions HTTP, Original-Recipient et Final-Recipient NE DEVRAIENT pas être différents. La valeur dans Original-Message-ID DEVRAIT correspondre à la valeur originale de l'en-tête Message-ID.
- Se référer à la RFC 3798 pour le formatage de la MDN, sauf pour les différences spécifiques mentionnées ci-dessus.
- Se référer aux RFC3462 et RFC3798 pour le formatage des en-têtes d'entité de type de contenu pour la MDN.
- Utiliser un mode d'action de "automatic-action" quand la disposition décrite par le type de disposition résultait d'une action automatique plutôt que d'une instruction explicite de l'utilisateur pour ce message.
- Utiliser un mode d'action de "manual-action" quand la disposition décrite par le type de disposition résultait d'une instruction explicite de l'utilisateur plutôt que d'une sorte d'action effectuée automatiquement.
- Utiliser un mode d'envoi de "MDN-sent-automatically" quand la MDN est envoyée parce que l'UA avait été préalablement configuré à faire ainsi.
- Utiliser un mode d'envoi de "MDN-sent-manually" quand l'utilisateur donne explicitement la permission que cette MDN particulière soit envoyée.
- Le mode d'envoi de "MDN-sent-manually" n'a de sens QUE avec "manual-action", pas avec "automatic-action".
- Le type de disposition "failed" NE DOIT PAS être utilisé pour une situation dans laquelle il y a des problèmes de traitement du message autres que d'interpréter la demande d'une MDN. Le type de disposition "processed" ou autre avec le modificateur de disposition approprié est à utiliser dans de telles situations.

7.5 Mode, type, et modificateur de disposition

7.5.1 Vue d'ensemble du mode de disposition

Ce paragraphe fait un bref survol de la façon d'utiliser "processed", "error", "failure", et "warning".

7.5.2 Indication d'état de traitement réussi

Lorsque la demande d'un récépissé ou récépissé signé, et le contenu du message reçu sont traités avec succès par l'UA EDI receveur, un récépissé ou une MDN DEVRAIT être retourné avec le type de disposition réglé à "processed". Quand la MDN est envoyée automatiquement par l'UA EDI, et qu'il n'y a pas de façon explicite pour l'utilisateur de contrôler l'envoi de la MDN, la première partie de "disposition-mode" DEVRAIT alors être réglée à "automatic-action". Quand la MDN est envoyée sous le contrôle configurable de l'utilisateur, la première partie de "disposition-mode" DEVRAIT alors être réglée à "manual-action". Comme une demande de récépissé signé devrait toujours être honorée, l'utilisateur NE DOIT PAS pouvoir configurer l'UA à ne pas envoyer de récépissé signé quand l'envoyeur en demande un.

La seconde partie du mode de disposition est réglée à "MDN-sent-manually" si l'utilisateur a donné la permission explicite que la MDN soit envoyée. Là encore, l'utilisateur NE DOIT PAS pouvoir refuser explicitement d'envoyer un récépissé signé quand l'envoyeur en demande un. La seconde partie du "disposition-mode" est réglée à "MDN-sent-automatically" chaque fois que l'UA EDI envoie la MDN automatiquement, sans considérer si l'envoi était fait sous le contrôle d'un utilisateur, d'un administrateur, ou d'un logiciel.

Parce qu'un contenu d'EDI est généralement traité automatiquement par l'UA d'EDI, une demande de récépissé ou de récépissé signé va généralement retourner ce qui suit dans le champ "disposition" :

```
Disposition: automatic-action/MDN-sent-automatically; processed
```

Noter que la présente spécification ne restreint pas l'utilisation du mode de disposition aux seules actions automatiques. Les actions manuelles sont valides pour autant qu'on se souvienne qu'une demande de récépissé signé DOIT être honorée.

7.5.3. Échec de traitement du contenu

La demande d'un récépissé signé exige l'utilisation de deux "disposition-notification-options", qui spécifient le format de protocole du récépissé signé retourné, et l'algorithme de MIC utilisé pour calculer la MIC sur le contenu du message. Les valeurs du "disposition-field" qui devraient être utilisées si le contenu du message est rejeté ou ignoré (par exemple, si l'UA d'EDI détermine qu'un récépissé signé ne peut pas être retourné parce qu'il ne prend pas en charge le format de protocole demandé, l'UA d'EDI choisit de ne pas traiter le contenu du message lui-même) DOIVENT être spécifiées dans le "disposition-field" de la MDN comme suit :

Disposition: "disposition-mode"; failed/Failure: unsupported format

Le type de disposition AS2 "failed" DOIT être utilisé quand une défaillance se produit qui empêche la génération appropriée d'une MDN. Par exemple, ce type de disposition s'appliquerait si l'envoyeur du message demandait l'application d'un algorithme de vérification d'intégrité de message (MIC) non pris en charge.

L'extension de modificateur de disposition AS2 "failure:" DEVRAIT être utilisée avec une description définie par la mise en œuvre de la défaillance. Plus d'informations sur la défaillance peuvent être contenues dans un champ "failure-field".

La syntaxe du type de disposition "failed" est générale, permettant l'envoi de toute information textuelle avec le type de disposition "failed". Les mises en œuvre DOIVENT prendre en charge tous les caractères textuels imprimables après le type de disposition " Failure". Pour l'utilisation dans les EDI Internet, les valeurs de "failed" suivantes sont prédéfinies et DOIVENT être acceptées :

"Failure: format non pris en charge"
"Failure: algorithmes de MIC non pris en charge"

7.5.4 Échec de traitement autre que de contenu

Quand des erreurs se produisent dans le traitement du message reçu (autres que de contenu) le champ "disposition" DOIT être réglé à la valeur "processed" pour le type de disposition et à la valeur "error" pour le modificateur de disposition.

L'erreur AS2-disposition-modifier avec le type de disposition "processed" DOIT être utilisée pour indiquer qu'une erreur d'une certaine sorte s'est produite qui a empêché le bon traitement du message. D'autres informations peuvent être contenues dans un champ "erreur".

Une erreur AS2-disposition-modifier-extension DEVRAIT être utilisée pour combiner l'indication d'une erreur avec une description prédéfinie d'une erreur spécifique bien connue. D'autres informations sur l'erreur peuvent être contenues dans un champ d'erreur.

Pour l'usage des EDI Internet, les valeurs d'erreur AS2-disposition-modifier suivantes sont définies :

- "Error: decryption-failed" : le receveur n'a pas pu déchiffrer le contenu du message.
- "Error: authentication-failed" : le receveur n'a pas pu authentifier l'envoyeur.
- "Error: integrity-check-failed" : le receveur n'a pas pu vérifier l'intégrité du contenu.
- "Error: unexpected-processing-error" : fourre-tout pour toute erreur de traitement supplémentaire.

Exemple de ce à quoi pourrait ressembler un "disposition-field" quand des erreurs autres que celles de traitement de contenu sont détectées :

Disposition: "disposition-mode"; processed/Error: decryption-failed

7.5.5 Avertissements de traitement

Il arrive dans les EDI des situations où, même si un partenaire commercial ne peut pas être authentifié correctement, les partenaires commerciaux s'accordent quand même pour continuer de traiter les transactions d'EDI. La réconciliation de transaction est faite ultérieurement entre les partenaires commerciaux. Dans le traitement de contenu de situations d'avertissements comme décrites plus haut, le "disposition-field" DOIT être réglé à la valeur de type de disposition de "processed", et le "warning" à la valeur "disposition-modifier".

L'avertissement AS2-disposition-modifier DOIT être utilisé avec le type de disposition "processed" pour indiquer que le message a bien été traité mais qu'une condition exceptionnelle s'est produite. Plus d'informations peuvent être contenues dans un champ "avertissement".

Un "warning:" AS2-disposition-modifier-extension DEVRAIT être utilisé pour combiner l'indication d'un avertissement avec une description de l'avertissement définie par la mise en œuvre. Plus d'informations sur l'avertissement peuvent être

contenues dans un champ d'avertissement.

Pour l'usage des EDI Internet, la valeur d'extension de modificateur de disposition "warning" suivante est définie :

"Warning: authentication-failed, processing continued"

Voici un exemple de ce à quoi pourrait ressembler le champ "disposition" quand est détecté un avertissement autre que celui pour le traitement de contenu :

Disposition: "disposition-mode"; processed/Warning: authentication-failed, processing continued

7.5.6 Rétro compatibilité avec type, modificateur, et extension de disposition

Voici un ensemble d'exemples représentant des constructions typiques du champ Disposition qui sont utilisées par des mises en œuvre AS2. Ceci N'EST PAS une liste exhaustive des constructions possibles. Cependant, les mises en œuvre de AS2 DOIVENT accepter que les constructions de ce type soient rétro compatibles avec les versions AS2 antérieures.

Disposition: automatic-action/MDN-sent-automatically; processed

Disposition: automatic-action/MDN-sent-automatically; processed/error: authentication-failed

Disposition: automatic-action/MDN-sent-automatically; processed/warning: duplicate-document

Disposition: automatic-action/MDN-sent-automatically; failed/failure: sender-equals-receiver

L'ensemble d'exemples suivants représente des constructions admissibles du champ Disposition qui combinent les constructions historiques ci-dessus avec les champs facultatifs d'erreur, d'avertissement et de défaillance de la RFC 3798. Les mises en œuvre AS2 PEUVENT produire ces constructions. Cependant, les serveurs AS2 ne sont pas obligés pour l'instant de reconnaître ou traiter les champs facultatifs d'erreur, d'avertissement, ou de défaillance. Noter que l'utilisation de plusieurs champs d'erreur dans le second exemple ci-dessous assure une indication de plusieurs conditions d'erreur.

Disposition: automatic-action/MDN-sent-automatically; processed

Disposition: automatic-action/MDN-sent-automatically; processed/error: decryption-failed
Erreur : la signature ne s'est pas déchiffrée en un bloc PKCS n° 1 type 2 valide.
Erreur : la longueur de la clé déchiffrée n'est pas égale à la longueur d'octets du module.

Disposition: automatic-action/MDN-sent-automatically; processed/warning: duplicate-document
Avertissement : un message identique existe déjà au serveur de destination.

Disposition: automatic-action/MDN-sent-automatically; failed/failure: sender-equals-receiver
Défaillance : le nom AS2-To est identique au nom AS2-From.

L'ensemble d'exemples suivant représente des constructions admissibles du champ Disposition qui emploient de purs modificateurs de disposition de la RFC 3798 avec des champs facultatifs d'erreur, avertissement, et défaillance. Ces exemples ne sont donnés qu'à des fins d'information. Il n'est pas garanti que ces constructions soient rétro compatibles avec les mises en œuvre AS2 antérieures à la 1.1.

Disposition: automatic-action/MDN-sent-automatically; processed

Disposition: automatic-action/MDN-sent-automatically; processed/error
Erreur : échec d'authentification
Erreur : la signature ne se déchiffre pas en une bloc PKCS n° 1 type-2 valide.
Erreur : la longueur de la clé déchiffrée n'est pas égale à la longueur d'octets du module.

Disposition: automatic-action/MDN-sent-automatically; processed/warning
Avertissement : document dupliqué

Disposition: automatic-action/MDN-sent-automatically; failed
Défaillance : envoyeur égal au receveur

7.6 Considérations de réponse de récépissé dans une commande POST HTTP

Les détails de la réponse à la commande POST varient selon qu'un récépissé a été demandé.

En l'absence d'un en-tête étendu demandant un récépissé, et d'erreurs de traitement d'accès à l'URI de demande spécifié, la ligne d'état dans la réponse à la demande POST DEVRAIT être dans la gamme des 200. Les codes d'état dans la gamme des 200 DEVRAIENT aussi être utilisés quand une entité est retournée (un récépissé signé dans un type de contenu multiparties/signé ou un récépissé non signé dans un multiparties/rapport). Même quand la disposition des données était une condition d'erreur à l'authentification, au déchiffrement ou autre de niveau supérieur, le code d'état HTTP DEVRAIT indiquer le succès au niveau HTTP.

L'application HTTP côté serveur peut répondre avec un multipart/report non sollicité comme corps de message que le client HTTP peut n'avoir pas sollicité, mais le client peut l'éliminer. Les applications DEVRAIENT éviter d'émettre des réponses de récépissé non sollicitées parce que des limitations de bande passante ou de traitement peuvent conduire les administrateurs à suspendre la demande d'accusés de réception.

Les notifications de disposition de message, quand elles sont utilisées dans le contexte de réponse HTTP, seront parallèles à la MDN SMTP. Par exemple, le champ disposition est un élément exigé dans la seconde partie lisible par la machine d'un multipart/report pour une MDN. La valeur du champ Receveur final (paragraphe 3.1 de la [RFC3798]) DEVRAIT être déduite des en-têtes d'entité de la demande.

Dans une MDN, la première partie du multipart/report (la partie lisible par l'homme) DEVRAIT inclure des éléments comme le sujet, la date, et autres informations quand ces champs sont présents dans les champs d'en-tête d'entité qui suivent la demande POST. Une application DOIT rapporter l'identifiant de message de la demande dans la seconde partie du multipart/report (la partie lisible par la machine). Aussi, une MDN DEVRAIT avoir son propre en-tête HTTP Message-ID univoque. La réponse HTTP DEVRAIT normalement omettre la troisième partie facultative du multipart/report (utilisée pour retourner le message d'origine ou ses en-têtes dans le contexte SMTP).

8. Traitement de certificat de clé publique

À court terme, l'échange de clés publiques et la certification de ces clés DOIVENT être traités au titre du processus d'établissement d'un partenariat commercial. L'UA et/ou l'interface d'application d'EDI doivent tenir une base de données des clés publiques utilisées pour le chiffrement ou les signatures, en plus de la transposition entre l'identifiant du partenaire commercial d'EDI et l'adresse de messagerie électronique de la [RFC2822] et l'URL/URI HTTP. Les procédures pour établir un partenariat commercial et configurer le système sûr de messagerie d'EDI peut varier selon les partenaires commerciaux et les paquetages de logiciels.

Les certificats X.509 sont EXIGÉS. Il est RECOMMANDÉ que les partenaires commerciaux s'auto certifient l'un l'autre si une autorité de certification mutuellement acceptée n'est pas utilisée. La présente déclaration d'applicabilité N'EXIGE PAS l'utilisation d'une autorité de certification. L'utilisation d'une autorité de certification est donc FACULTATIVE. Les certificats peuvent être auto signés.

Il est RECOMMANDÉ que lorsque des partenaires commerciaux utilisent S/MIME, ils échangent aussi les certificats de clé publique, en considérant l'avis fourni dans la [RFC3850].

Les formats de message utiles pour l'échange de certificats se trouvent dans les [RFC3851] et [RFC3852].

À long terme, des normes supplémentaires pourront être développées pour simplifier le processus d'établissement d'un partenariat commercial incluant l'authentification par un tiers des partenaires commerciaux ainsi que des attributs de la relation commerciale.

9. Considérations sur la sécurité

Le présent document est tout entier concerné par le transport sûr de données d'affaires, et il considère les questions de confidentialité et d'authentification des données.

Extrait de la [RFC3851] :
Le chiffrement à 40 bits est considéré comme faible par la plupart des cryptographes. L'utilisation d'une cryptographie faible dans S/MIME offre peu de sécurité réelle pour l'envoi du texte source. Cependant, d'autres caractéristiques de S/MIME, comme la spécification du triple DES et la capacité d'annoncer des capacités cryptographiques plus fortes aux

parties avec lesquelles on communique, permet aux envoyeurs de créer des messages qui utilisent un chiffrement fort. Utiliser une cryptographie faible n'est jamais recommandé sauf si la seule solution de remplacement est pas de chiffrement du tout. Lorsque faisable, les agents d'envoi et de réception DEVRAIENT informer les envoyeurs et receveurs de la force relative du chiffrement des messages.

Extrait de la [RFC3850] :
Lors du traitement des certificats, il y a de nombreuses situations où le traitement peut échouer. Comme le traitement peut être fait par un agent d'utilisateur, une passerelle de sécurité, ou autre programme, il n'y a pas une seule façon de traiter de telles défaillances. Simplement parce que la liste des méthodes pour traiter les défaillances n'a pas été établie, le lecteur ne devrait pas supposer qu'elles ne sont pas importantes. C'est le contraire qui est vrai : si un certificat n'est pas d'une validité démontrable et associée au message, le logiciel de traitement devrait prendre des mesures immédiates et perceptibles pour en informer l'utilisateur final.

Parmi les nombreuses situations dans lesquelles la vérification de signature et du certificat peut échouer on notera les suivantes :
- o aucune chaîne de certificats ne conduit à une CA de confiance,
- o pas de capacité de vérifier la liste de révocation de certificats (CRL, *Certificate Revocation List*) pour un certificat,
- o une CRL invalide a été reçue,
- o la CRL vérifiée est arrivée à expiration,
- o le certificat est arrivé à expiration,
- o le certificat a été révoqué.

Il y a certainement d'autres instances où un certificat peut être invalide, et il est de la responsabilité du logiciel de traitement de les vérifier toutes attentivement, et de décider que faire si la vérification échoue. Voir dans la RFC 3280 des informations supplémentaires sur la validation du chemin de certification.

Les considérations de sécurité suivantes s'ajoutent à celles des [RFC3851] et [RFC3850].

9.1 Avertissements de NRR

Cette spécification cherche à fournir plusieurs mécanismes qui puissent être combinés en accord avec les politiques locales pour satisfaire une large gamme de besoins de sécurité comme déterminé par les analyses de menaces et de risques des homologues d'affaires. Il est exigé que tous ces mécanismes soient mis en œuvre par le logiciel AS2 afin qu'il ait les capacités qui promeuvent une forte interopérabilité, quelles que soient les politiques adoptées.

Un groupe de forts mécanismes (la boucle de transmission sûre) peut fournir un bon support pour satisfaire les besoins fondés sur la preuve de la non répudiation du récépissé (NRR) par l'envoyeur d'origine et par un tiers, fournie avec toutes les preuves avancées. Cependant, la présente spécification ne définit pas elle-même la non répudiation de récépissé ni n'énumère ses propriétés essentielles parce que la non répudiation de récépissé est une analyse professionnelle et/ou une exigence légale, et n'est pas définie de façon pertinente par une déclaration d'applicabilité technique.

Certains analystes observent que la non répudiation de récépissé présuppose que la non répudiation de l'envoyeur du message d'origine est obtenue, et de plus que la non répudiation devrait être mise en œuvre au moyen d'une signature numérique sur le message d'origine. Pour satisfaire à une stricte preuve de NRR, l'authentification et l'intégrité DOIVENT être fournies par certains mécanismes, et le mécanisme RECOMMANDÉ est la signature numérique sur le message d'origine et sur le message de récépissé.

Étant donné que la présente spécification a choisi plusieurs mécanismes qui peuvent être combinés de plusieurs façons, il est important de réaliser que si une signature numérique est omise sur le message d'origine, afin de satisfaire l'analyse précédente des exigences de NRR, un mécanisme d'authentification DOIT accompagner la demande d'un récépissé signé et sa valeur incluse de Received-content-MIC. Cette authentification peut venir de l'utilisation de SSL côté client, de l'authentification via IPsec, ou de l'authentification HTTP (tout en utilisant SSL). Dans tous les cas, les enregistrements du contenu du message, ses bases de sécurité, et la valeur de résumé doivent être conservés pour le processus de NRR.

Donc, si la NRR est un des buts de la politique qui est adoptée, en utilisant les mécanismes de boucle de transmission sûre mentionnés plus haut et en conservant les enregistrements appropriés d'authentification sur le site de l'envoyeur du message d'origine, les exigences de forte preuve proposées pour la NRR peuvent être satisfaites.

D'autres façons de procéder peuvent manquer à satisfaire les ensembles de preuve les plus contraignants exigés pour obtenir la NRR, mais peuvent néanmoins faire partie d'un accord commercial et, à ce titre, sont assez bons pour les parties impliquées. Cependant, si des MDN sont retournées non signées, les exigences de preuves pour la NRR sont faibles ; une certaine authentification de l'identité du receveur est nécessaire.

9.2 Remarques sur HTTPS

Les types de certificat suivants DOIVENT être pris en charge pour les certificats SSL du côté serveur :
- o avec URL dans l'attribut de nom commun de nom distinctif
- o sans URL dans l'attribut de nom commun de nom distinctif
- o auto signé (auto produit)
- o certifié par une autorité de certification

L'URL, qui correspond à l'identité du serveur source, DEVRAIT être porté dans le certificat. Cependant, il n'est pas exigé de vérification du DNS ou des recherches inverses pour garantir la précision de l'URL ou de la valeur du serveur.

Comme les certificats du côté serveur sont échangés, et qu'aussi la confiance est établie durant la configuration de la relation entre les partenaires commerciaux, les vérifications au moment du démarrage ne sont pas exigées par les mises en œuvre de la présente spécification.

La chaîne de certification complète DOIT être incluse dans tous les certificats. Toutes les vérifications de certificat DOIVENT s'enchaîner à la racine ou à une ancre de confiance acceptée. De plus, le hachage du certificat DEVRAIT correspondre au hachage recalculé par le receveur.

9.3 Remarques sur la répétition

Parce que normalement les documents d'affaires contiennent les identifiants de transaction, les répétitions (comme les renvois non encore acquittés par des messages d'accusé de réception) sont éliminées au titre du processus normal de détection des dupliqués. La détection des dupliqués par les identifiants de message ou par des identifiants de transaction d'affaire est recommandée.

10. Considérations relatives à l'IANA

La RFC 3335 enregistrait deux paramètres Disposition-Notification-Options :
Parameter-name: signed-receipt-protocol
Parameter-name: signed-receipt-micalg
qui sont aussi utilisés par cette spécification (voir le paragraphe 7.3).

La RFC 3335 enregistrait aussi le nom de champ MDN Extension :
Nom de champ Extension : Received-content-MIC
qui est aussi utilisé par cette spécification (voir le paragraphe 7.4.3).

Leur enregistrement n'est donc pas nécessaire.

10.1 Enregistrement

Cette spécification définit une extension au protocole de notification de disposition de message (MDN) pour un modificateur de disposition dans le champ Disposition d'un corps de type de contenu "message/disposition-notification".

10.1.1 "Avertissement" de modificateur de disposition

Parameter-name: warning
Sémantique : voir les paragraphes 7.4.3 et 7.5.5 de ce document.

11. Remerciements

Carl Hage, Karen Rosenfeld, Chuck Fenton, et de nombreux autres ont fourni de précieuses suggestions qui ont amélioré la présente déclaration d'applicabilité. Les auteurs tiennent aussi à remercier les fabricants qui ont participé aux essais d'interopérabilité d'AS2 du Drummond Group Inc.. Leurs contributions ont conduit à une grande amélioration de la clarté de ce document.

12. Références

12.1 Références normatives

[RFC1767] D. Crocker,"Encapsulation MIME d'objets EDI", mars 1995. *(P.S.)*

[RFC1847] J. Galvin, S. Murphy, S. Crocker, N. Freed, "Sécurité multiparties pour MIME : multipartie/signée et multipartie/chiffrée", octobre 1995. *(P.S.)*

[RFC2026] S. Bradner, "Le processus de normalisation de l'Internet -- Révision 3", (BCP0009) octobre 1996. *(Remplace* RFC1602, RFC1871) (*MàJ par* RFC3667, 3668, 3932, 3979, 3978, 5378, 6410, 8179)

[RFC2045] N. Freed et N. Borenstein, "Extensions de messagerie Internet multi-objets (MIME) Partie 1 : Format des corps de message Internet", novembre 1996. *(D. S., MàJ par* 2184, 2231, 5335.)

[RFC2046] N. Freed et N. Borenstein, "Extensions de messagerie Internet multi-objets (MIME) Partie 2 : Types de support", novembre 1996. *(D. S., MàJ par* 2646, 3798, 5147, 6657, 8098)

[RFC2049] N. Freed, N. Borenstein, "Extensions multi-objets de la messagerie Internet (MIME) Partie cinq : critères de conformité et exemples", novembre 1996. *(Remplace* RFC1521, RFC1522, RFC1590) *(D.S.)*

[RFC2119] S. Bradner, "Mots clés à utiliser dans les RFC pour indiquer les niveaux d'exigence", BCP 14, mars 1997. *(MàJ par* RFC8174)

[RFC2234] D. Crocker et P. Overell, "BNF augmenté pour les spécifications de syntaxe : ABNF", novembre 1997. (*Obsolète, voir* RFC5234)

[RFC2616] R. Fielding et autres, "Protocole de transfert hypertexte -- HTTP/1.1", juin 1999. *(D.S., MàJ par* 2817, 6585)

[RFC2822] P. Resnick, "Format de message Internet", avril 2001. *(Remplace la* RFC0822, STD 11, *Remplacée par* RFC5322)

[RFC3023] M. Murata, S. St.Laurent et D. Kohn, "Types de support XML", janvier 2001. (*Obsolète, voir* RFC7303)

[RFC3335] T. Harding, R. Drummond, C. Shih, "Échange de données d'affaire sécurisées d'homologue à homologue fondé sur MIME sur l'Internet", septembre 2002. *(P.S.)*

[RFC3462] G. Vaudreuil, "Type de contenu Multipart/Report pour les rapports des messages administratifs du système de messagerie", janvier 2003. (*Remplacée par* RFC6522, STD 73*)*

[RFC3798] T. Hansen et G. Vaudreuil, éd., "Notification de disposition de message", mai 2004. (*MàJ par* RFC5337, RFC6533) *(D.S.; Rendue obsolète par* RFC8098)

[RFC3850] B. Ramsdell, éd., "Traitement de certificat d'extensions multi-objets/sécurisées de messagerie Internet (S/MIME) version 3.1", juillet 2004. *(P.S.) (Remplacée par* RFC5750)

[RFC3851] B. Ramsdell, "Spécification du message d'extensions de messagerie Internet multi-objets/sécurisé (S/MIME) version 3.1", juillet 2004. *(Obsolète, voir* RFC5751)

[RFC3852] R. Housley, "Syntaxe de message cryptographique (CMS)", juillet 2004. *(Obsolète, voir la* RFC5652)

12.1 Références pour information

[RFC2246] T. Dierks et C. Allen, "Protocole TLS version 1.0", janvier 1999. *(P.S. ; MàJ par* RFC7919)

Appendice A : Exemples de messages

Note : tous les exemples sont fournis à de seules fins d'illustration, et ne sont pas considérés faire partie de la spécification du protocole. Si un exemple entre en conflit avec les définitions du protocole spécifiées ci-dessus ou dans d'autres RFC référencées, c'est l'exemple qui est erroné.

A.1 Message signé demandant un récépissé signé synchrone

```
POST /receive HTTP/1.0
Host: 10.234.160.12:80
User-Agent: AS2 Company Server
Date: Wed, 31 Jul 2002 13:34:50 GMT
From: mrAS2@example.com
AS2-Version: 1.1
AS2-From: "\"  as2Name  \""
AS2-To: 0123456780000
Subject: Test Case
Message-Id: <200207310834482A70BF63@\"~~foo~~\">
Disposition-Notification-To: mrAS2@example.com
Disposition-Notification-Options: signed-receipt-protocol=optional,
  pkcs7-signature; signed-receipt-micalg=optional,sha1
Content-Type: multipart/signed; boundary="as2BouNdary1as2";
  protocol="application/pkcs7-signature"; micalg=sha1
Content-Length: 2464

--as2BouNdary1as2
Content-Type: application/edi-x12
Content-Disposition: Attachment; filename=rfc1767.dat
  [ISA ...EDI transaction data...IEA...]

--as2BouNdary1as2
Content-Type: application/pkcs7-signature

  [on omet les données binaires pkcs7 de signature]
--as2BouNdary1as2--
```

A.2 MDN pour le message A.1

```
HTTP/1.0 200 OK
AS2-From: 0123456780000
AS2-To: "\"  as2Name  \""
AS2-Version: 1.1
Message-ID: <709700825.1028122454671.JavaMail@ediXchange>
Content-Type: multipart/signed; micalg=sha1;
    protocol="application/pkcs7-signature";
    boundary="----=_Part_57_648441049.1028122454671"
Connection: Close

Content-Length: 1980

------=_Part_57_648441049.1028122454671

& Content-Type: multipart/report;
& Report-Type=disposition-notification;
&    boundary="----=_Part_56_1672293592.1028122454656"
&
&------=_Part_56_1672293592.1028122454656
&Content-Type: text/plain
&Content-Transfer-Encoding: 7bit
&
&MDN for -
```

```
& Message ID: <200207310834482A70BF63@\"~~foo~~\">
&  From: "\"  as2Name  \""
&  To: "0123456780000"
&  Received on: 2002-07-31 at 09:34:14 (EDT)
& Status: processed
& Comment: Ceci n'est pas une garantie que le message ait
& été complètement traité ou compris par le traduteur receveur
&
&------=_Part_56_1672293592.1028122454656
&Content-Type: message/disposition-notification
&Content-Transfer-Encoding: 7bit
&
&Reporting-UA: AS2 Server
&Original-Recipient: rfc822; 0123456780000
&Final-Recipient: rfc822; 0123456780000
&Original-Message-ID: <200207310834482A70BF63@\"~~foo~~\">
&Received-content-MIC: 7v7F++fQaNB1sVLFtMRp+dF+eG4=, sha1
&Disposition: automatic-action/MDN-sent-automatically;
&  processed
&
&------=_Part_56_1672293592.1028122454656--

------=_Part_57_648441049.1028122454671
Content-Type: application/pkcs7-signature; name=smime.p7s
Content-Transfer-Encoding: base64
Content-Disposition: attachment; filename=smime.p7s

MIAGCSqGSIb3DQEHAqCAMIACAQExCzAJBgUrDgMCGgUAMIAGCSqGSIb3DQ
cp24hMJNbxDKHnlB9jTiQzLwSwo+/90Pc87x+Sc6EpFSUYWGAAAAAAAA
------=_Part_57_648441049.1028122454671--
```

Notes :

1. Les lignes précédées de "&" sont ce sur quoi la signature est calculée.
2. Pour les détails de la façon de préparer le multipart/signed avec protocol = "application/pkcs7-signature", voir la "Spécification de message S/MIME, services de sécurité PKCS pour MIME".)
3. Noter que la première partie textuelle de corps du multipart/report peut être utilisée pour inclure une explication plus détaillée des conditions d'erreur rapportées par les en-tête de disposition. La première partie de corps du multipart/report, quand elle est utilisée de cette façon, permet à une personne de mieux diagnostiquer un problème en détails.
4. Comme spécifié par la [RFC3462], retourner l'original ou des portions du message d'origine dans la troisième partie de corps du multipart/report n'est pas exigé. C'est une partie de corps facultative. Cependant, il est RECOMMANDÉ que cette partie de corps soit omise ou laissée en blanc.

A.3 Message signé et chiffré demandant un récépissé signé asynchrone

```
Message-ID: <#as2_company#01#a4260as2_companyout#>
Date: Thu, 19 Dec 2002 15:04:18 GMT
From: me@example.com
Subject: Async MDN request
Mime-Version: 1.0
Content-Type: application/pkcs7-mime;
  smime-type=enveloped-data; name=smime.p7m
Content-Transfer-Encoding: binary
Content-Disposition: attachment; filename=smime.p7m
Recipient-Address: 10.240.1.2//
Disposition-Notification-To:
  http://10.240.1.2:8201/exchange/as2_company
Disposition-Notification-Options: signed-receipt-protocol=optional,
 pkcs7-signature; signed-receipt-micalg=optional,sha1
Receipt-Delivery-Option:
  http://10.240.1.2:8201/exchange/as2_company
```

```
AS2-From: as2_company
AS2-To: "AS2 Test"
AS2-Version: 1.1
Host: 10.240.1.2:8101
Connection: close
Content-Length: 3428
```

[les données chiffrées binaires sont omises]

A.4 MDN asynchrone pour le message A.3

```
POST / HTTP/1.1
Host: 10.240.1.2:8201
Connection: close, TE
TE: trailers, deflate, gzip, compress
User-Agent: RPT-HTTPClient/0.3-3I (Windows 2000)
Date: Thu, 19 Dec 2002 15:03:38 GMT
Message-ID: <AS2-20021219_030338@as2_company.dgi_th>
AS2-Version: 1.1
Mime-Version: 1.0
Recipient-Address:
http://10.240.1.2:8201/exchange/as2_company
AS2-To: as2_company
AS2-From: "AS2 Test"
Subject: Votre réponse à la MDN demandée
From: as2debug@example.com
Accept-Encoding: deflate, gzip, x-gzip, compress, x-compress
Content-Type: multipart/signed; micalg=sha1;
  protocol="application/pkcs7-signature";
  boundary="----=_Part_337_6452266.1040310218750"
Content-Length: 3103

------=_Part_337_6452266.1040310218750
Content-Type: multipart/report;
  report-type=disposition-notification;
  boundary="----=_Part_336_6069110.1040310218718"

------=_Part_336_6069110.1040310218718
Content-Type: text/plain; charset=us-ascii
Content-Transfer-Encoding: 7bit
```

Le message <x12.edi> envoyé au receveur <AS2 Test> le mardi 19 décembre 2002 15:04:18 GMT avec Subject <demande de MDN async> a été reçu. L'échange EDI a été déchiffré avec succès, et son intégrité vérifiée. De plus, l'envoyeur du message, Sender <as2_company> localisé à http://10.240.1.2:8201/exchange/as2_company a été authentifié comme origine du message. Il n'y a cependant aucune garantie que l'échange d'EDI soit syntaxiquement correct, ou qu'il ait été reçu par l'application/traducteur EDI.

```
------=_Part_336_6069110.1040310218718
Content-Type: message/disposition-notification
Content-Transfer-Encoding: 7bit

Reporting-UA: AS2@test:8101
Original-Recipient: rfc822; "AS2 Test"
Final-Recipient: rfc822; "AS2 Test"
Original-Message-ID: <#as2_company#01#a4260as2_companyout#>
Disposition: automatic-action/MDN-sent-automatically; processed
Received-Content-MIC: Hes6my+vIxIYxmvsA+MNpEOTPAc=, sha1

------=_Part_336_6069110.1040310218718--

------=_Part_337_6452266.1040310218750
Content-Type: application/pkcs7-signature; name=smime.p7s
```

```
Content-Transfer-Encoding: base64
Content-Disposition: attachment; filename=smime.p7s

BhbWjEfbyXoTAS/H0zpnEqLqbaBh29y2v82b8bdeGw8pipBQWmf53hIcqHGM
4ZBF3CHw5Wrf1JIE+8TwOzdbal30zeChw88WfRfD7c/j1fIA8sxsujvf2d9j
UxCUga8BVdVB9kH0Geexytyt0KvWQXfaEEcgZGUAAAAAAAA=

------=_Part_337_6452266.1040310218750-
```

Adresse des auteurs

Dale Moberg
Cyclone Commerce
8388 E. Hartford Drive, Suite 100
Scottsdale, AZ 85255 USA
mél : dmoberg@cyclonecommerce.com

Rik Drummond
Drummond Group Inc.
4700 Bryant Irvin Court, Suite 303
Fort Worth, TX 76107 USA
mél : rvd2@drummondgroup.com

Déclaration complète de droits de reproduction

Remerciement
Le financement de la fonction d'édition des RFC est actuellement fourni par la Internet Society.