Équipe d'ingénierie de l'Internet (IETF)
Request for Comments : 6733
RFC rendues obsolètes : 3588, 5719
Catégorie : En cours de normalisation
ISSN : 2070-1721
Traduction Claude Brière de L'Isle

V. Fajardo, éd., Telcordia Technologies
J. Arkko, Ericsson Research
J. Loughney, Nokia Research Center
G. Zorn, éd., Network Zen
octobre 2012

Protocole de base Diameter

Résumé
Le protocole de base Diameter est destiné à fournir un cadre d'authentification, autorisation, et comptabilité (AAA) pour des applications comme l'accès réseau ou la mobilité IP dans des situations aussi bien locales que d'itinérance. Le présent document spécifie le format de message, le transport, le rapport d'erreurs, la comptabilité et les services de sécurité utilisés par toutes les applications Diameter. Le protocole de base Diameter tel que définit dans le présent document rend obsolète les RFC 3588 et RFC 5719, et il doit être pris en charge par toutes les nouvelles mises en œuvre Diameter.

Statut de ce mémoire
Ceci est un document de l'Internet en cours de normalisation.

Le présent document a été produit par l'équipe d'ingénierie de l'Internet (IETF). Il représente le consensus de la communauté de l'IETF. Il a subi une révision publique et sa publication a été approuvée par le groupe de pilotage de l'ingénierie de l'Internet (IESG). Tous les documents approuvés par l'IESG ne sont pas candidats à devenir une norme de l'Internet ; voir la Section 2 de la RFC5741.

Les informations sur le statut actuel du présent document, tout errata, et comment fournir des réactions sur lui peuvent être obtenues à http://www.rfc-editor.org/info/rfc6733

Table des Matières

1. Introduction

Les protocoles d'authentification, autorisation et comptabilité (AAA, *Authentication, Authorization, and Accounting*) comme TACACS [RFC1492] et RADIUS [RFC2865] ont été initialement déployés pour fournir un accès PPP numéroté [RFC1661] et de serveur terminal. Avec le temps, la prise en charge de AAA s'est révélée nécessaire sur de nombreuses technologies d'accès nouvelles, l'échelle et la complexité des réseaux AAA s'est accrue, et AAA était aussi utilisé sur de nouvelles applications (comme la voix sur IP). Cela a conduit à de nouvelles demandes à l'égard des protocoles AAA.

Les exigences d'accès réseau pour les protocoles AAA sont résumées dans Aboba, et al. [RFC2989]. Elles incluent :

Reprise sur défaillance : la [RFC2865] ne définit pas le mécanisme de reprise sur défaillance et par conséquent, le comportement de reprise sur défaillance diffère selon la mise en œuvres. Afin de fournir un comportement bien défini de reprise sur défaillance, Diameter prend en charge les accusés de réception de niveau application et définit des algorithmes de reprise sur défaillance et l'automate à états associé.

Sécurité de niveau transmission : RADIUS [RFC2865] définit un schéma d'authentification et d'intégrité de niveau application qui n'est obligatoire qu'avec les paquets de réponse. Bien que la [RFC2869] définisse un mécanisme supplémentaire d'authentification et d'intégrité, son utilisation n'est exigée que durant les sessions du protocole d'authentification extensible (EAP) [RFC3748]. Alors que la dissimulation d'attribut est prise en charge, la [RFC2865] ne fournit pas la prise en charge de la confidentialité par paquet. En comptabilité, la [RFC2866] suppose que la protection contre la répétition est fournie par le serveur de facturation de l'extrémité arrière plutôt qu'au sein du protocole lui-même. Bien que la [RFC3162] définisse l'utilisation de IPsec avec RADIUS, la prise en charge de IPsec n'est pas exigée. Pour fournir une prise en charge universelle de la sécurité de niveau transmission, et permettre des déploiements AAA intra- et inter-domaines, Diameter fournit la prise en charge de TLS/TCP et DTLS/SCTP. La sécurité est discutée à la Section 13.

Transport fiable : RADIUS fonctionne sur UDP, et ne définit pas de comportement de retransmission ; par suite, la fiabilité varie selon la mise en œuvre. Comme décrit dans la [RFC2975], ceci est un problème majeur en comptabilité, où la

perte de paquet peut se traduire directement en perte de revenu. Afin de fournir un comportement de transport bien défini, Diameter fonctionne avec des mécanismes de transport fiable (TCP, protocole de transmission de contrôle de flux (SCTP)) comme définit dans la [RFC3539].

Prise en charge d'agent : RADIUS ne fournit pas une prise en charge explicite des agents, incluant les mandataires, les agents de redirection, et les relais. Comme le comportement attendu n'est pas défini, il varie selon la mise en œuvre. Diameter définit explicitement le comportement des agents ; ceci est décrit au paragraphe 2.8.

Messages initiés par le serveur : Bien que les messages initiés par le serveur soient définis dans RADIUS [RFC5176], leur prise en charge est facultative. Cela rend difficile la mise en œuvre de caractéristiques comme la déconnexion non sollicitée ou la réauthentification/réautorisation à la demande à travers un déploiement hétérogène. Pour régler ce problème, la prise en charge des messages initiés par le serveur est obligatoire dans Diameter.

Prise en charge des transitions : Bien que Diameter ne partage pas d'unités de données de protocole (PDU) communes avec RADIUS, un effort considérable a été apporté à la recherche de la rétro compatibilité avec RADIUS afin que les deux protocoles puissent être déployés dans le même réseau. Initialement, on pensait que Diameter serait déployé dans de nouveaux appareils réseau, ainsi que dans des passerelles permettant la communication entre les appareils RADIUS traditionnels et les agents Diameter. Cette capacité permet que la prise en charge de Diameter soit ajoutée aux réseaux traditionnels, par l'ajout d'une passerelle ou d'un serveur parlant à la fois RADIUS et Diameter.

En plus des exigences ci-dessus, Diameter prend aussi en charge ce qui suit :

Négociation de capacité : RADIUS ne prend pas en charge les messages d'erreur, la négociation de capacité, ou un fanion obligatoire/non obligatoire pour les attributs. Comme les clients et serveurs RADIUS ne connaissent pas leurs capacités réciproques, ils peuvent n'être pas capables de négocier avec succès un service mutuellement acceptable ou, dans certains cas, être même informés du service mis en œuvre. Diameter inclut la prise en charge du traitement des erreurs (Section 7) de la négociation de capacité (paragraphe 5.3) et des paires d'attributs valeurs (AVP, *Attribute-Value Pair*) obligatoires/non obligatoires (paragraphe 4.1).

Découverte et configuration d'homologue : les mises en œuvre RADIUS exigent normalement que le nom ou l'adresse des clients et serveurs soit configurée manuellement, avec les secrets partagés correspondants. Il en résulte une grosse charge administrative et cela crée la tentation de réutiliser le secret partagé RADIUS, d'où peuvent résulter des vulnérabilités majeures pour la sécurité si l'authentificateur de demande n'est pas mondialement et temporellement unique comme exigé par la [RFC2865]. Avec le DNS, Diameter permet une découverte dynamique des homologues (voir le paragraphe 5.2). La déduction dynamique des clés de session est permise via la sécurité de niveau transmission.

Avec le temps, les capacités des appareils serveur d'accès réseau (NAS, *Network Access Server*) ont substantiellement augmenté. Par suite, bien que Diameter soit un protocole considérablement plus sophistiqué que RADIUS, il reste possible de le mettre en œuvre dans des appareils incorporés.

1.1 Protocole Diameter

Le protocole de base Diameter fournit les facilités suivantes :
- o capacité d'échanger des messages et livrer des AVP,
- o capacités de négociation,
- o notification des erreurs,
- o extensibilité, requise dans la [RFC2989], par l'ajout de nouvelles applications, commandes, et AVP,
- o services de base nécessaires pour les applications, comme le traitement des sessions d'utilisateur ou de la comptabilité.

Toutes les données livrées par le protocole sont sous la forme d'AVP. Certaines de ces valeurs d'AVP sont utilisées par le protocole Diameter lui-même, tandis que d'autres livrent des données associées à des applications particulières qui emploient Diameter. Les AVP peuvent être arbitrairement ajoutées aux messages Diameter, la seule restriction étant que la spécification du format de code de commande (CCF, *Command Code Format*) (paragraphe 3.2) soit satisfaite. Les AVP sont utilisées par le protocole Diameter de base pour la prise en charge des caractéristiques exigées suivantes :
- o Transporter les informations d'authentification de l'utilisateur, afin de permettre au serveur Diameter d'authentifier l'usager,
- o Transporter les informations d'autorisation spécifiques du service, entre clients et serveurs, en permettant aux homologues de décider si une demande d'accès d'un utilisateur devrait être accordée,
- o Échanger des informations d'utilisation de ressources, qui peuvent être utilisés pour la comptabilité, la planification des capacités, etc.,
- o Acheminement, relais, mandatement, et redirection des messages Diameter à travers une hiérarchie de serveurs.

Le protocole de base Diameter satisfait les exigences minimales pour un protocole AAA, comme spécifié par la [RFC2989]. Le protocole de base peut être utilisé par lui-même pour les seuls besoins de comptabilité, ou il peut être utilisé avec une application Diameter, comme IPv4 mobile [RFC4004], ou l'accès réseau [RFC4005]. Il est aussi possible que le protocole de base soit étendu pour être utilisé dans de nouvelles applications, via l'ajout de nouvelles commandes ou AVP. Diameter se concentrait initialement sur les applications d'accès réseau et de comptabilité. Un protocole AAA vraiment générique utilisé par de nombreuses applications peut fournir des fonctionnalités que n'a pas Diameter. Donc, il est impératif que les concepteurs de nouvelles applications comprennent leurs exigences avant d'utiliser Diameter. Voir au paragraphe 1.3.4 d'autres informations sur les applications Diameter.

Tout nœud peut initier une demande. Dans ce sens, Diameter est un protocole d'homologue à homologue. Dans le présent document, un client Diameter est un appareil au bord du réseau qui effectue le contrôle d'accès, comme un serveur d'accès réseau (NAS, *Network Access Server*) ou un agent étranger (FA, *Foreign Agent*). Un client Diameter génère des messages Diameter pour demander des services d'authentification, d'autorisation et de comptabilité pour l'usager. Un agent Diameter est un nœud qui ne fournit pas de services d'authentification ou d'autorisation de l'utilisateur ; les agents incluent des mandataires, des agents de redirection, et des agents de relais. Un serveur Diameter effectue l'authentification et/ou l'autorisation de l'usager. Un nœud Diameter peut agir comme agent pour certaines demandes alors qu'il agit comme serveur pour d'autres.

Le protocole Diameter prend aussi en charge les messages initiés par le serveur, comme une demande d'interrompre le service pour un usager particulier.

1.1.1 Description de l'ensemble du document

La spécification Diameter consiste en une version mise à jour de la spécification du protocole de base (le présent document) et du profil de transport [RFC3539]. Le présent document rend obsolète les deux RFC 3588 et RFC 5719. Un résumé des mises à jour du protocole de base incluses dans le présent document se trouve au paragraphe 1.1.3.

Le présent document définit la spécification du protocole de base pour AAA, qui inclut la prise en charge de la comptabilité. Il y a aussi une myriade de documents d'applications qui décrivent les applications qui utilisent cette spécification de base pour l'authentification, l'autorisation, et la comptabilité. Ces documents d'application spécifient comment utiliser le protocole Diameter dans le contexte de leur application.

Le document de profil de transport [RFC3539] discute les questions de couche transport qui se posent avec les protocoles AAA et fait des recommandations sur la façon de répondre à ces problèmes. Le présent document définit aussi l'algorithme Diameter de reprise sur défaillance et l'automate à états.

La [RFC5729] "Précisions sur l'acheminement des demandes Diameter sur la base du nom d'utilisateur et du domaine" définit le comportement spécifique pour acheminer les demandes sur la base du contenu de l'AVP User-Name.

1.1.2 Conventions utilisées dans le document

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

1.1.3 Changements par rapport à la RFC 3588

Le présent document rend obsolète la RFC 3588 mais est entièrement rétro compatible avec ce document. Les changements introduits dans ce document se concentrent sur la correction des problèmes qui sont apparus durant la mise en œuvre de Diameter (RFC 3588). Une vue d'ensemble des changements majeurs est donnée ci-dessous.

- L'utilisation de l'AVP Inband-Security est déconseillée pour la négociation de la sécurité de la couche transport (TLS, *Transport Layer Security*) [RFC5246]. Il a été généralement considéré que l'amorçage de TLS via l'AVP Inband-Security crée certains risques pour la sécurité parce qu'elle ne protège pas complètement les informations portées dans la demande/réponse d'échange de capacités (CER/CEA, *Capabilities-Exchange-Request/Capabilities-Exchange-Answer*). La présente version de Diameter adopte l'approche commune de définition d'un accès sécurisé bien connu que les homologues devraient utiliser lors des communications via TLS/TCP et DTLS/SCTP. Cette nouvelle approche augmente la négociation de sécurité existante dans la bande, mais elle ne la remplace pas complètement. La vieille méthode est conservée pour des raisons de rétro compatibilité.

- Est déconseillé l'échange des messages CER/CEA dans l'état Ouvert. Cette caractéristique était impliquée dans le tableau d'automate à états de l'homologue de la RFC 3588, mais elle n'était pas clairement définie ailleurs dans le document. Avec les progrès du travail sur ce document, il est devenu clair que la multiplicité des significations et des

utilisations des AVP Application-Id dans les messages CER/CEA (et des messages eux-mêmes) est vue comme un abus des règles d'extensibilité de Diameter et exigeait donc une simplification. L'échange de capacités dans l'état Ouvert a été réintroduit dans une spécification séparée [RFC6737], qui définit clairement de nouvelles commandes pour ce dispositif.

- Exigences de sécurité simplifiées. L'utilisation d'un transport sécurisé pour échanger les messages Diameter reste obligatoire. Cependant, TLS/TCP et DTLS/SCTP sont devenues les principales méthodes de sécurisation de Diameter avec IPsec comme solution secondaire. Voir les détails à la Section 13. La prise en charge du cadre de sécurité de bout en bout (AVP E2E-Sequence et bit 'P' dans l'en-tête d'AVP) a aussi été déconseillée.

- Changement de l'extensibilité de Diameter. Cela inclut des corrections à la description de l'extensibilité de Diameter (paragraphe 1.3 et autres) pour mieux aider les concepteurs d'applications Diameter ; de plus, la nouvelle spécification assouplit la politique à l'égard de l'allocation des codes de commandes pour les usages spécifiques de fabricant.

- Précisions sur l'utilisation de l'identifiant d'application. On a précisé le bon usage des informations de l'identifiant d'application, qui peuvent se trouver dans plusieurs endroits au sein d'un message Diameter. Cela inclut de corréler les identifiants d'application qui se trouvent dans les en-têtes de message et les AVP. Ces changements spécifient aussi clairement la bonne valeur d'identifiant d'application à utiliser pour les messages spécifiques du protocole de base (ASR/ASA, STR/STA) ainsi qu'ils précisent le contenu et l'utilisation de Vendor-Specific-Application-Id.

- Précisions sur les corrections d'acheminement. Le présent document spécifie plus clairement quelles informations (AVP et identifiants d'application) peuvent être utilisés pour prendre des décisions générales d'acheminement. Une règle pour l'établissement des priorités des critères de redirection d'acheminement lorsque plusieurs entrées de chemin se trouvent via des redirections a aussi été ajoutée (voir au paragraphe 6.13).

- Découverte simplifiée de l'homologue Diameter. Le processus de découverte Diameter prend maintenant en charge seulement des schémas de découverte largement utilisés ; le reste a été déconseillé (voir les détails au paragraphe 5.2).

De nombreuses corrections diverses ont été introduites dans ce document qui ne peuvent pas être considérées comme significatives, mais qui ont néanmoins leur importance. Des exemples sont la suppression de types obsolètes, des corrections à l'automate à états, des précisions sur le processus d'élection, la validation des messages, des corrections aux valeurs des AVP Failed-AVP et Result-Code, etc. Tous les champs d'errata sur la RFC 3588 avant la publication du présent document ont été traités. Une liste complète de ces changements n'est pas établie ici pour des raisons pratiques.

1.2 Terminologie

AAA *(Authentication, Authorization, et Accounting)* : authentification, autorisation, et comptabilité

ABNF *(Augmented Backus-Naur Form)* : format Backus-Naur augmenté [RFC5234]. Métalangage qui a ses propres règles et syntaxe formelle. Il se fonde sur le format Backus-Naur et est utilisé pour définir les échanges de messages dans un protocole de communications bidirectionnel.

Comptabilité : acte de collecte des informations sur l'utilisation de ressources à des fins de planification des capacités, d'examen, de facturation, ou d'allocation de coûts.

Enregistrement comptable : un enregistrement comptable représente un résumé de la consommation de ressources d'un utilisateur sur la session entière. Les serveurs de comptabilité qui créent l'enregistrement comptable peuvent le faire en traitant des événements comptables intermédiaires ou des événements comptables à partir de plusieurs appareils qui desservent le même usager.

Authentification : acte de vérification de l'identité d'une entité (sujet).

Autorisation : acte de détermination de si une entité demandeuse (sujet) sera autorisée à accéder à une ressource (objet).

Paire attribut-valeur (AVP, *Attribute-Value Pair*) : le protocole Diameter consiste en un en-tête suivi par une ou plusieurs paires d'attribut-valeur (les AVP). Une AVP inclut un en-tête et est utilisée pour encapsuler des données spécifiques du protocole (par exemple, des informations d'acheminement) ainsi que des informations d'authentification, autorisation, ou comptabilité.

Format de code de commande (CCF) : forme modifiée de l'ABNF utilisée pour définir les commandes Diameter (voir au paragraphe 3.2).

Agent Diameter : c'est un nœud Diameter qui fournit des services de relais, de mandataire, de redirection, ou de traduction.

Client Diameter : c'est un nœud Diameter qui prend en charge les applications de client Diameter ainsi que le protocole de base. Les clients Diameter sont souvent mis en œuvre dans des appareils situés à la bordure d'un réseau et fournissent des services de contrôle d'accès pour ce réseau. Des exemples typiques de clients Diameter incluent le serveur d'accès réseau (NAS) et l'agent étranger IP mobile (FA).

Nœud Diameter : un nœud Diameter est un processus hôte qui met en œuvre le protocole Diameter et agit comme un client, un agent, ou un serveur.

Homologue Diameter : deux nœuds Diameter partageant une connexion de transport directe TCP ou SCTP sont appelés des homologues Diameter.

Serveur Diameter : un serveur Diameter est un nœud Diameter qui traite les demandes d'authentification, d'autorisation et de comptabilité pour un domaine particulier. Par sa nature même, un serveur Diameter doit prendre en charge les applications de serveur Diameter en plus du protocole de base.

Vers l'aval : l'aval est utilisé pour identifier la direction d'un message Diameter particulier du serveur de rattachement vers le client Diameter.

Domaine de rattachement *(Home Realm)* : un domaine de rattachement est le domaine administratif avec lequel l'usager maintient une relation comptable.

Serveur de rattachement : serveur Diameter qui dessert le domaine de rattachement.

Comptabilité intermédiaire *(Interim Accounting)* : un message de comptabilité intermédiaire donne une photographie de l'usage durant une session d'un utilisateur. Normalement, il est mis en œuvre pour fournir une comptabilité partielle d'une session d'utilisateur au cas où un appareil se réamorce ou si un autre problème réseau empêche la livraison d'un message de résumé de session ou d'enregistrement de session.

Domaine local : c'est le domaine administratif qui fournit des services à un utilisateur. Un domaine administratif peut agir comme domaine local pour certains usagers tout en étant un domaine de rattachement pour d'autres.

Multi session : une multi session représente une liaison logique de plusieurs sessions. Les multi sessions sont suivies en utilisant le Acct-Multi-Session-Id. Un exemple de multi session serait un faisceau multi liaison PPP. Chaque branche du faisceau serait une session alors que le faisceau entier sera une multi session.

Identifiant d'accès réseau (NAI, *Network Access Identifier*) [RFC4282] : il est utilisé dans le protocole Diameter pour extraire l'identité et le domaine d'un utilisateur. L'identité est utilisée pour identifier l'usager durant l'authentification et/ou l'autorisation tandis que le domaine est utilisé pour les besoins de l'acheminement des messages.

Agent mandataire ou mandataire : en plus de transmettre les demandes et les réponses, les mandataires prennent des décisions de politique relatives à l'usage des ressource et à l'approvisionnement. Normalement, ceci se fait en suivant l'état des appareils de NAS. Bien que normalement les mandataires ne répondent pas aux demandes du client avant d'avoir reçu une réponse du serveur, ils peuvent générer des messages de rejet dans des cas où des politiques sont violées. Par suite, les mandataires ont besoin de comprendre la sémantique des messages qui passent à travers eux, et ils peuvent ne pas prendre en charge toutes les applications Diameter.

Domaine : chaîne dans le NAI qui suit immédiatement le caractère '@'. Les noms de domaine de NAI doivent être uniques et sont portés sur l'espace de noms administré par le DNS. Diameter utilise le domaine pour déterminer si les messages peuvent être satisfaits localement ou si ils doivent être acheminés ou redirigés. Dans RADIUS, les noms de domaine ne sont pas nécessairement portés par l'espace de noms du DNS mais peuvent en être indépendants.

Comptabilité en temps réel : elle implique le traitement des informations sur l'utilisation des ressources dans une fenêtre temporelle définie. Normalement, les contraintes de temps sont imposées afin de limiter le risque financier. L'application de contrôle de crédit Diameter [RFC4006] est un exemple d'application qui définit une fonction de comptabilité en temps réel.

Agent de relais ou relais : les relais transmettent les demandes et réponses sur la base des AVP relatives à l'acheminement et des entrées des tableaux d'acheminement. Comme les relais ne prennent pas de décisions de politique, ils n'examinent ni n'altèrent pas les AVP qui ne sont pas d'acheminement. Par suite, les relais ne génèrent jamais de messages, n'ont pas besoin de comprendre la sémantique des messages ou des AVP qui ne sont pas d'acheminement, et sont capables de traiter tout type d'application ou message Diameter. Comme les relais prennent des décisions sur la base des

informations contenues dans les AVP d'acheminement et les tableaux de transmission des domaines, ils ne conservent pas l'état sur l'usage des ressource de NAS ou sur les sessions en cous.

Agent de redirection : plutôt que de transmettre les demandes et réponses entre clients et serveurs, les agents de redirection mettent en contact les clients avec les serveurs et leur permettent de communiquer directement. Comme les agents de redirection ne se tiennent pas dans le chemin de transmission, ils n'altèrent aucune AVP transitant entre client et serveur. Les agents de redirection ne génèrent pas de messages et sont capables de traiter tout type de message, bien qu'il puissent être configurés à ne rediriger que les messages de certains types, tout en agissant comme agents de relais ou mandataires pour d'autres types. Comme avec les agents de relais, les agents de redirection ne conservent pas l'état par rapport aux sessions ou ressources de NAS.

Session : une session est une progression d'événements en rapports dédiés à une activité particulière. Les documents d'application Diameter donnent des lignes directrices sur le moment où une session commence et finit. Tous les paquets Diameter qui ont le même identifiant de session sont considérés comme faisant partie de la même session.

Agent à états pleins : un agent à états pleins est celui qui conserve les informations d'état de session, en gardant trace de toutes les sessions actives autorisées. Chaque session autorisée est liée à un service particulier, et son état est considéré comme actif jusqu'à une notification contraire ou jusqu'à expiration.

Sous session : une sous session représente un service distinct (par exemple, caractéristiques de qualité de service ou des données) fourni à une certaine session. Ces services peuvent se produire concurremment (par exemple, voix et transfert de données simultanés durant la même session) ou à la suite. Ces changements dans les sessions sont retracés avec l'AVP Accounting-Sub-Session-Id.

État de transaction : le protocole Diameter exige que les agents conservent l'état de transaction, qui est utilisé pour les besoins de la reprise sur défaillance. L'état de transaction implique qu'à la transmission d'une demande, l'identifiant bond par bond soit sauvegardé ; le champ est remplacé par un identifiant localement unique, qui est restauré à sa valeur d'origine lorsque la réponse correspondante est reçue. L'état de la demande est libéré à réception de la réponse. Un agent sans état est celui qui ne conserve que l'état de transaction.

Agent de traduction : un agent de traduction (TLA sur la Figure 4) est un nœud Diameter à états pleins qui effectue la traduction de protocole entre Diameter et un autre protocole AAA, comme RADIUS.

Vers l'amont : vers l'amont est utilisé pour identifier la direction d'un message Diameter particulier du client Diameter vers le serveur de rattachement.

Usager : entité ou appareil qui demande ou utilise une ressource, pour la prise en charge de laquelle un client Diameter a généré une demande.

1.3 Approche de l'extensibilité

Le protocole Diameter est conçu pour être extensible, en utilisant plusieurs mécanismes, parmi lesquels :
- o Définir de nouvelles valeurs d'AVP
- o Créer de nouvelles AVP
- o Créer de nouvelles commandes
- o Créer de nouvelles applications

Du point de vue de l'extensibilité, les applications Diameter d'authentification, d'autorisation et de comptabilité sont traitées de la même façon.

Note : Les concepteurs de protocoles devraient essayer de réutiliser les fonctionnalités existantes, à savoir les valeurs d'AVP, les AVP, les commandes, et les applications Diameter. La réutilisation simplifie la normalisation et la mise en œuvre. Pour éviter de possibles problèmes d'interopérabilité, il est important de s'assurer que la sémantique des caractéristiques réutilisées est bien comprise. Étant donné que Diameter peut aussi porter des attributs RADIUS comme AVP Diameter, ces considérations de réutilisation s'appliquent aussi aux attributs RADIUS existants qui peuvent être utiles dans une application Diameter.

1.3.1 Définition de nouvelles valeurs d'AVP

Pour allouer une nouvelle valeur d'AVP pour les AVP définies dans le protocole de base Diameter, l'IETF doit approuver une nouvelle RFC qui décrit la valeur de l'AVP. Les considérations relatives à l'IANA pour ces valeurs d'AVP sont discutées au paragraphe 11.3.

L'allocation de valeurs d'AVP pour les autres AVP sont guidées par les considérations relatives à l'IANA du document qui définit ces AVP. Normalement, l'allocation de nouvelles valeurs pour une AVP définie dans une RFC va exiger une revue de l'IETF [RFC5226], tandis que les valeurs des AVP spécifiques de fabricant peuvent être allouées par le fabricant.

1.3.2 Création de nouvelles AVP

La définition d'une nouvelle AVP DOIT utiliser un des types de données figurant aux paragraphes 4.2 ou 4.3. Si un type de données dérivé approprié est déjà défini, il DEVRAIT être utilisé au lieu d'un type de données de base pour encourager la réutilisabilité et un bonne pratique de conception.

Dans le cas où un groupement logique des AVP est nécessaire, et si plusieurs "groupes" sont possibles dans une certaine commande, il est recommandé qu'une AVP Grouped soit utilisée (voir au paragraphe 4.4).

La création de nouvelles AVP peut arriver de diverses façons. L'approche recommandée est de définir une nouvelle AVP d'utilisation générale dans une RFC sur la voie de la normalisation approuvée par l'IETF. Cependant, comme décrit au paragraphe 11.1.1, il y a d'autres mécanismes.

1.3.3 Création de nouvelles commandes

Un nouveau code de commande DOIT être alloué lorsque des AVP exigées (celles indiquées comme {AVP} dans la définition du CCF) sont ajoutées à, supprimées de, ou redéfinies dans (par exemple, en changeant une AVP obligatoire en une facultative) une commande existante.

De plus, si les caractéristiques de transport d'une commande sont changées (par exemple, par rapport au nombre d'allers-retours exigés) un nouveau code de commande DOIT être enregistré.

Un changement au CCF d'une commande, comme décrit ci-dessus, DOIT résulter en la définition d'un nouveau code de commande. Ceci conduit ensuite au besoin de définir une nouvelle application Diameter pour toute application qui va utiliser la nouvelle commande.

Les considérations relatives à l'IANA pour les codes de commandes sont discutées au paragraphe 3.1.

1.3.4 Création de nouvelles applications Diameter

Chaque spécification d'application Diameter DOIT avoir un identifiant d'application alloué par l'IANA (voir au paragraphe 2.4). L'espace d'identifiants d'application géré n'est pas hiérarchisé, et il n'y a pas de relation entre les différentes applications Diameter par rapport à leur identifiant d'application. À ce titre, il n'y a pas de prise en charge des numéros de version fournie par les identifiants d'application eux-mêmes ; chaque application Diameter est une application autonome. Si l'application a une relation avec d'autres applications Diameter, elle n'est pas connue de Diameter.

Avant de décrire les règles de création de nouvelles applications Diameter, il est important d'exposer la sémantique des occurrences d'AVP comme déclarée dans le CCF et le fanion bit M (paragraphe 4.1) pour une AVP. Il n'y a pas de relation imposée entre les deux ; ils sont établis de façon indépendante.

- Le CCF indique quelles AVP sont placées dans une commande Diameter par l'envoyeur de cette commande. Souvent, comme il y a plusieurs modes d'interactions de protocole, beaucoup de ces AVP sont indiquées comme facultatives.

- Le bit M permet à l'envoyeur d'indiquer au receveur si il comprend ou non la sémantique d'une AVP et si son contenu est obligatoire. Si le bit M est établi par l'envoyeur et si le receveur ne comprend pas l'AVP ou les valeurs portées dans cette AVP, une erreur est générée (voir la Section 7).

Il appartient au concepteur de protocole de décider de développer une nouvelle application Diameter plutôt que d'étendre Diameter d'une autre façon. Cependant, une nouvelle application Diameter DOIT être créée lorsque un ou plusieurs des critères suivants sont réunis :

Établissement du bit M : une AVP avec le bit M dans la colonne DOIT du tableau des fanions d'AVP est ajoutée à la commande/application existante. Une AVP avec le bit M dans la colonne PEUT du tableau des fanions d'AVP est ajoutée à une commande/application existante.

Note : l'établissement du bit M pour une certaine AVP relève d'une application et chaque commande dans cette application qui comporte cette AVP. C'est-à-dire que si une AVP apparaît dans deux commandes pour l'application Foo et si les

réglages du bit M sont différents dans chaque commande, il devrait alors y avoir deux tableaux de fanions d'AVP qui décrivent quand établir le bit M.

Commandes : une nouvelle commande est utilisée au sein de l'application existante parce que soit une commande supplémentaire est ajoutée, une commande existante a été modifiée de sorte qu'un nouveau code de commande doit être enregistré, soit qu'une commande a été supprimée.

Bits de fanion d'AVP : si une application existante change la signification/sémantique de ses fanions d'AVP ou ajoute de nouveaux bits de fanion, une nouvelle application Diameter DOIT alors être créée.

Si la définition du CCF d'une commande le permet, une mise en œuvre peut ajouter des AVP facultatives arbitraires avec le bit M à zéro (incluant des AVP spécifiques de fabricant) à cette commande sans avoir besoin de définir une nouvelle application. Prière de se reporter au paragraphe 11.1.1 pour les détails.

2. Vue d'ensemble du protocole

Le protocole Diameter de base s'occupe d'établir des connexions avec les homologues, de la négociation des capacités, de la façon dont les messages sont envoyés et acheminés jusqu'aux homologues, et de la façon dont les connexions sont finalement supprimées. Le protocole de base définit aussi certaines règles qui s'appliquent à tous les échanges de messages entre les nœuds Diameter.

La communication entre les homologues Diameter commence par l'envoi d'un message par un homologue à un autre homologue Diameter. L'ensemble des AVP incluses dans le message est déterminé par une application Diameter particulière. Une AVP qui est incluse pour référencer une session d'utilisateur est le Session-Id *(identifiant de session)*.

La demande initiale pour l'authentification et/ou l'autorisation d'un usager va inclure l'AVP Session-Id. Le Session-Id est alors dans tous les messages suivants pour identifier la session de l'usager (voir plus d'informations à la Section 8). Les parties communicantes peuvent accepter la demande ou la rejeter en retournant un message de réponse avec l'AVP Result-Code *(code de résultat)* réglé à indiquer qu'une erreur s'est produite. Le comportement spécifique du serveur ou client Diameter à réception d'une demande dépend de l'application Diameter employée.

L'état de session (associé à un Session-Id) DOIT être libéré à réception de Session-Termination-Request *(demande de terminaison de session)*, de Session-Termination-Answer *(réponse de terminaison de session)*, à l'expiration du temps de service autorisé dans l'AVP Session-Timeout *(temporisation de session)*, et conformément aux règles établies dans une application Diameter particulière.

Le protocole Diameter de base peut être utilisé par lui-même pour des applications de comptabilité. Pour l'authentification et l'autorisation, il est toujours étendu pour une application particulière.

Les clients Diameter DOIVENT prendre en charge le protocole de base, qui inclut la comptabilité. De plus, ils DOIVENT pleinement prendre en charge chaque application Diameter qui est nécessaire pour mettre en œuvre le service du client, par exemple, les exigences des serveurs d'accès réseau (NASREQ, *Network Access Server Requirements*) [RFC2881] et/ou d'IPv4 mobile. Un client Diameter DOIT être désigné comme "client Diameter X" où X est l'application qu'il prend en charge et non comme un "client Diameter".

Les serveurs Diameter DOIVENT prendre en charge le protocole de base, qui inclut la comptabilité. De plus, ils DOIVENT pleinement prendre en charge chaque application Diameter qui est nécessaire pour mettre en œuvre le service prévu, par exemple, NASREQ et/ou IPv4 mobile. Un serveur Diameter DOIT être désigné comme "serveur Diameter X" où X est l'application qu'il prend en charge, et non comme un "serveur Diameter".

Les agents Diameter de relais et redirection sont transparents aux applications Diameter, mais ils DOIVENT prendre en charge le protocole de base Diameter, qui inclut la comptabilité, et toutes les applications Diameter.

Les mandataires Diameter DOIVENT prendre en charge le protocole de base, qui inclut la comptabilité. De plus, ils DOIVENT pleinement prendre en charge chaque application Diameter qui est nécessaire pour mettre en œuvre des services mandatés, par exemple, NASREQ et/ou IPv4 mobile. Un mandataire Diameter DOIT être désigné comme "mandataire Diameter X" où X est l'application qu'il prend en charge, et non comme un "mandataire Diameter".

2.1 Transport

Le profil de transport Diameter est défini dans la [RFC3539].

Le protocole Diameter de base fonctionne sur l'accès 3868 pour TCP [RFC0793] et SCTP [RFC4960]. Pour TLS [RFC5246] et la sécurité de couche transport des datagrammes (DTLS, *Datagram Transport Layer Security*) [RFC6347], un nœud Diameter qui initie une connexion avant tout échange de messages DOIT fonctionner sur l'accès 5658. On suppose que TLS fonctionne par dessus TCP quand il est utilisé, et DTLS fonctionne par dessus SCTP lorsque utilisé.

Si l'homologue Diameter ne prend pas en charge la réception des connexions TLS/TCP et DTLS/SCTP sur l'accès 5658 (c'est-à-dire, si l'homologue se conforme seulement à la RFC 3588) l'initiateur PEUT alors revenir à l'utilisation de TCP ou SCTP sur l'accès 3868. Noter que ce schéma n'est conservé que pour les besoins de la rétro compatibilité et qu'il y a des faiblesses de sécurité inhérentes lorsque les messages CER/CEA initiaux sont envoyés sans protection (voir au paragraphe 5.6).

Les clients Diameter DOIVENT prendre en charge TCP ou SCTP ; les agents et serveurs DEVRAIT les prendre tous les deux en charge.

Un nœud Diameter PEUT initier des connexions à partir d'un accès de source autre que celui qu'il déclare accepter pour les connexions entrantes, et il DOIT toujours être prêt à recevoir des connexions sur l'accès 3868 pour TCP ou SCTP et sur l'accès 5658 pour les connexions TLS/TCP et DTLS/SCTP. Lorsque est utilisée la découverte de l'homologue fondée sur le DNS (paragraphe 5.2) les numéros d'accès reçus des enregistrements SRV prennent le pas sur les accès par défaut (3868 et 5658).

Une certaine instance Diameter de l'automate à états de l'homologue NE DOIT PAS utiliser plus d'une connexion de transport pour communiquer avec un certain homologue, sauf si plusieurs instances existent sur l'homologue, auquel cas une connexion séparée par procès est permise.

Lorsque il n'existe pas de connexion de transport avec un homologue, une tentative de connexion DEVRAIT être faite périodiquement. Ce comportement est traité via le temporisateur Tc (voir les détails à la Section 12) dont la valeur recommandée est de 30 secondes. Il y a certaines exceptions à cette règle, comme lorsque un homologue a terminé la connexion de transport en déclarant qu'il ne souhaite plus communiquer.

Lors de la connexion à un homologue et si zéro, un, ou plusieurs transports sont spécifiés, TLS DEVRAIT être essayé d'abord, suivi par DTLS, puis par TCP, et finalement par SCTP. Voir au paragraphe 5.2 plus d'informations sur la découverte de l'homologue.

Les mises en œuvre Diameter DEVRAIENT être capables d'interpréter les messages ICMP "accès de protocole injoignable" comme une indication explicite que le serveur n'est pas joignable, sous réserve de la politique de sécurité sur la confiance à accorder à de tels messages. On peut trouver plus d'indications sur le traitement des erreurs ICMP dans les [RFC5927] et [RFC5461]. Les mises en œuvre Diameter DEVRAIENT aussi être capables d'interpréter un rétablissement provenant du transport et des fins de temporisation de tentatives de connexion. Si Diameter reçoit des données de la couche inférieure qui ne peuvent pas être analysées ou identifiées comme erreur Diameter faite par l'homologue, le flux est compromis et ne peut pas être récupéré. La connexion de transport DOIT être fermée en utilisant un appel RESET (envoi d'un bit TCP RST) ou un message SCTP ABORT (la fermeture en douceur est compromise).

2.1.1 Lignes directrices SCTP

Les messages Diameter DEVRAIENT être transposés en un flux SCTP d'une façon qui évite le blocage de tête de ligne (HOL, *head-of-the-line*). Parmi les différentes façons d'effectuer la transposition qui satisfont à cette exigence, il est RECOMMANDÉ qu'un nœud Diameter envoie chaque message Diameter (demande ou réponse) sur un flux zéro avec le fanion Non ordonné établi. Cependant, les nœuds Diameter PEUVENT choisir et mettre en œuvre d'autres solutions de remplacement pour éviter le blocage HOL, comme d'utiliser plusieurs flux avec le fanion Non ordonné à zéro (comme le demandait à l'origine la RFC 3588). Du côté réception, une entité Diameter DOIT être prête à recevoir des messages Diameter sur tout flux, et elle est libre de retourner des réponses sur un flux différent. De cette façon, les deux côtés gèrent les flux disponibles dans la direction d'envoi, indépendamment des flux choisis par l'autre côté pour envoyer un certain message Diameter. Ces messages peuvent être déclassés et appartenir à des sessions Diameter différentes.

La livraison en désordre pose des problèmes particuliers durant l'établissement et la terminaison d'une connexion. Lorsque une connexion est établie, le côté qui répond envoie un message CEA et passe à l'état R-Ouvert comme spécifié au paragraphe 5.6. Si un message d'application est envoyé peu après le CEA et est livré déclassé, le côté initiateur, toujours dans l'état Wait-I-CEA, va éliminer le message d'application et clore la connexion. Pour éviter cette condition, le côté receveur NE DEVRAIT PAS utiliser de méthodes de livraison sans ordre jusqu'à ce que le premier message ait été reçu de

l'initiateur, prouvant qu'il est passé à l'état I-Ouvert. Pour déclancher un tel message, le côté receveur pourrait envoyer un DWR immédiatement après l'envoi d'un CEA. À réception du DWA correspondant, le côté receveur devrait commencer d'utiliser des méthodes de livraison sans ordre pour contrer le blocage HOL.

Une autre condition de compétition peut survenir quand des messages DPR et DPA sont utilisés. DPR et DPA sont tous deux de petite taille ; donc, ils peuvent être livrés à l'homologue plus vite que les messages d'application quand un mécanisme de livraison sans ordre est utilisé. Donc, il est possible qu'un échange DPR/DPA s'achève alors que des messages d'application sont encore en transit, d'où résulte la perte de ces messages. Une mise en œuvre pourrait atténuer cette condition de compétition, par exemple, en utilisant des temporisateurs, et en attendant un bref instant que les messages en cours de niveau application arrivent avant de procéder à la déconnexion de la connexion de transport. Éventuellement, les messages perdus sont traités par le mécanisme de retransmission décrit au paragraphe 5.5.4.

Un agent Diameter DEVRAIT utiliser des identifiants de protocole de charge utile (PPID, *payload protocol identifier*) dédiés pour les tronçons DATA SCTP en texte source et chiffrés au lieu d'utiliser seulement l'identifiant de protocole de charge utile non spécifié (valeur 0). À cette fin, deux valeurs de PPID sont allouées : la valeur de PPID de 46 est pour les messages Diameter de tronçons DATA SCTP en clair, et la valeur de PPID 47 est pour les messages Diameter dans les tronçons DATA DTLS/SCTP protégés.

2.2 Sécurisation des messages Diameter

Les connexions entre homologues Diameter DEVRAIENT être protégées par TLS/TCP et DTLS/SCTP. Toutes les mises en œuvre du protocole de base Diameter DOIVENT prendre en charge l'utilisation de TLS/TCP et DTLS/SCTP. Si désiré, des mécanismes de sécurité de remplacement qui sont indépendants de Diameter, comme IPsec [RFC4301], peuvent être déployé pour sécuriser les connexions entre homologues. Le protocole Diameter NE DOIT PAS être utilisé sans TLS, DTLS, ou IPsec.

2.3 Conformité à l'application Diameter

Les identifiants d'application sont annoncés durant la phase d'échange de capacités (voir au paragraphe 5.3). L'annonce de la prise en charge d'une application implique que l'envoyeur prenne en charge la fonction spécifiée dans la spécification d'application Diameter pertinente.

Les mises en œuvre PEUVENT ajouter des AVP facultatives arbitraires avec le bit M à zéro (incluant des AVP spécifiques de fabricant) à une commande définie dans une application, mais seulement si la spécification de la syntaxe de CCF de la commande le permet. Prière de se reporter pour les détails au paragraphe 11.1.1.

2.4 Identifiants d'application

Chaque application Diameter DOIT avoir un identifiant d'application alloué par l'IANA. Le protocole de base n'exige pas un identifiant d'application car sa prise en charge est obligatoire. Durant l'échange de capacités, les nœuds Diameter informent leurs homologues des applications prises en charge en local. De plus, tous les messages Diameter contiennent un identifiant d'application, qui est utilisé dans le processus de transmission du message.

Les valeurs d'identifiant d'application suivantes sont définies :

0 : messages Diameter commun,
3 : comptabilité Diameter de base,
0xffffffff : Relais

Les agents de relais et de redirection DOIVENT annoncer l'identifiant d'application de relais, tandis que tous les autres nœuds Diameter DOIVENT annoncer les applications prises en charge en local. Le receveur d'un message d'échange de capacités qui annonce un service de relais DOIT supposer que l'envoyeur prend en charge toutes les applications actuelles et futures.

Les agents de relais et de redirection Diameter sont chargés de trouver un serveur vers l'amont qui prend en charge l'application d'un message particulier. Si aucun n'est trouvé, un message d'erreur est retourné avec l'AVP Result-Code réglée à DIAMETER_UNABLE_TO_DELIVER *(Diameter est incapable de le livrer)*.

2.5 Connexions et sessions

Ce paragraphe tente de faire comprendre au lecteur la différence entre "connexion" et "session", qui sont des termes

largement utilisés dans le présent document.

Une connexion se réfère à une connexion de niveau transport entre deux homologues et qui est utilisée pour envoyer et recevoir des messages Diameter. Une session est un concept logique de la couche application qui existe entre le client Diameter et le serveur Diameter ; elle est identifiée via l'AVP Session-Id.

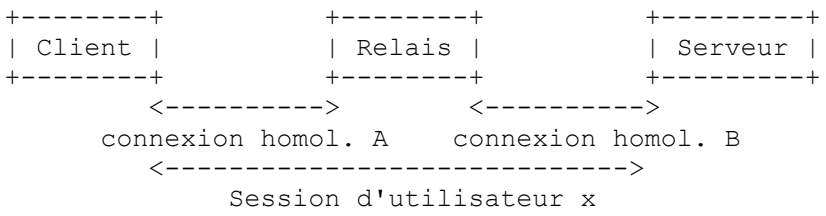


Figure 1 : Connexions et sessions Diameter

Dans l'exemple de la Figure 1, la connexion d'homologue A est établie entre le client et le relais. La connexion d'homologue B est établie entre le relais et le serveur. La session d'utilisateur X s'étend à partir du client via le relais jusqu'au serveur. Chaque "usager" d'un service cause l'envoi d'une demande d'authentification, avec un identifiant de session unique. Une fois acceptée par le serveur, le client et le serveur sont tous deux informés de la session.

Il est important de noter qu'il n'y a pas de relation entre une connexion et une session, et que les messages Diameter pour plusieurs sessions sont tous multiplexés à travers une seule connexion. Aussi, on notera que les messages Diameter qui appartiennent à la session, aussi bien spécifiques d'application que ceux qui sont définis dans le présent document comme ASR/ASA, RAR/RAA, et STR/STA, DOIVENT porter l'identifiant d'application de l'application. Les messages Diameter qui relèvent de l'établissement et la maintenance de la connexion d'homologue comme CER/CEA, DWR/DWA, et DPR/DPA DOIVENT porter un identifiant d'application de zéro (0).

2.6 Tableau d'homologues

Le tableau des homologues Diameter est utilisé pour la transmission des messages et est référencé par le tableau d'acheminements. Une entrée de tableau d'homologue contient les champs suivants :

Identité d'hôte : suivant les conventions décrites pour le format de données d'AVP dérivée DiameterIdentity au paragraphe 4.3.1, ce champ contient le contenu de l'AVP Origin-Host (paragraphe 6.3) trouvé dans le message CER ou CEA.

StatusT : c'est l'état de l'entrée de l'homologue, et il DOIT correspondre à une des valeurs citées au paragraphe 5.6.

Statique ou dynamique : spécifie si une entrée d'un homologue a été configurée statiquement ou découverte de façon dynamique.

Heure d'expiration : spécifie l'heure à laquelle les entrées de tableau d'homologue découvert de façon dynamique vont être rafraîchies, ou vont expirer. Si des certificats de clé publique sont utilisés pour la sécurité de Diameter (par exemple, avec TLS) cette valeur NE DOIT PAS être supérieure à celle des heures d'expiration des certificats qui s'y rapportent.

TLS/TCP et DTLS/SCTP activé : spécifie si TLS/TCP et DTLS/SCTP sont utilisés lors des communications avec l'homologue.

Informations de sécurité supplémentaires, lorsque nécessaire (par exemple, clés, certificats).

2.7 Tableau d'acheminement

Toutes les recherches d'acheminement fondées sur le domaine sont effectuées par rapport à ce qui est généralement appelé le tableau d'acheminement (voir la Section 12). Chaque entrée du tableau d'acheminement contient les champs suivants :

Nom de domaine : c'est le champ qui DOIT être utilisé comme clé principale dans les recherches au sein du tableau d'acheminement. Noter que certaines mises en œuvre effectuent leurs recherches sur la base de la plus longue correspondance sur la droite du domaine plutôt que d'exiger une correspondance exacte.

Identifiant d'application : une application est identifiée par un identifiant d'application. Une entrée de chemin peut avoir une destination différente sur la base de l'identifiant d'application dans l'en-tête du message. Ce champ DOIT être utilisé comme champ de clé secondaire pour les recherches dans les tableaux d'acheminement.

Action locale : le champ Action locale est utilisé pour identifier comment un message devrait être traité. Les actions suivantes sont prises en charge :

1. LOCAL - messages Diameter qui peuvent être satisfaits localement et n'ont pas besoin d'être acheminés à une autre entité Diameter.
2. RELAIS – tous les messages Diameter qui entrent dans cette catégorie DOIVENT être acheminés à une entité Diameter de prochain bond qui est indiquée par l'identifiant décrit ci dessous. L'acheminement est fait sans modifier d'AVP qui n'est pas d'acheminement. Voir au paragraphe 6.1.9 les lignes directrices du relais.
3. PROXY – tous les messages Diameter qui entrent dans cette catégorie DOIVENT être acheminés à une prochaine entité Diameter qui est indiquée par l'identifiant décrit ci dessous .Le serveur local PEUT appliquer ses politiques locales au message en incluant de nouvelles AVP au message avant de l'acheminer. Voir au paragraphe 6.1.9 les lignes directrices sur le mandatement.
4. REDIRECT – les messages Diameter qui entrent dans cette catégorie DOIVENT avoir l'identité du ou des serveurs Diameter de rattachement ajoutée, et retournée à l'envoyeur du message. Voir au paragraphe 6.1.8 les lignes directrices sur la redirection.

Identifiant de serveur : l'identité d'un ou plusieurs serveurs auxquels le message est à acheminer. Cette identité DOIT aussi être présente dans le champ Identité d'hôte du tableau d'homologue (paragraphe 2.6). Lorsque l'action locale est réglée à RELAIS ou PROXY, ce champ contient l'identité du ou des serveurs auxquels le message DOIT être acheminé. Lorsque le champ Action locale est réglé à REDIRECT, ce champ contient l'identité de un ou plusieurs serveurs auxquels le message DOIT être redirigé.

Statique ou dynamique : spécifie si une entrée de chemin a été configurée de façon statique ou découverte de façon dynamique.

Heure d'expiration : spécifie l'heure à laquelle expire une entrée de tableau de chemin découverte dynamiquement. Si des certificats de clé publique sont utilisés pour la sécurité de Diameter (par exemple, avec TLS) cette valeur NE DOIT PAS être supérieure à l'heure d'expiration dans les certificats concernés.

Il est important de noter que les agents Diameter DOIVENT prendre en charge au moins un des modes de fonctionnement LOCAL, RELAIS, PROXY, ou REDIRECT. Les agents n'ont pas besoin de prendre en charge tous les modes de fonctionnement pour être conformes à la spécification du protocole, mais ils DOIVENT suivre les lignes directrices de la conformité au protocole de la Section 2. Les agents de relais et mandataires NE DOIVENT PAS réorganiser les AVP.

Le tableau d'acheminement PEUT inclure une entrée par défaut qui DOIT être utilisée pour toutes les demandes qui ne correspondent à aucune des autres entrées. Le tableau d'acheminement PEUT consister en seulement une telle entrée.

Lorsque une demande est acheminée, le serveur cible DOIT avoir annoncé l'identifiant d'application (voir au paragraphe 2.4) pour le message concerné ou s'être annoncé lui-même comme agent de relais ou mandataire. Autrement, une erreur est retournée avec l'AVP Result-Code réglée à DIAMETER_UNABLE_TO_DELIVER.

2.8 Rôle des agents Diameter

En plus des clients et serveurs, le protocole Diameter introduit des agents de relais, mandataires, de redirection, et de traduction, qui sont tous définis au paragraphe 1.2. Les agents Diameter sont utiles pour plusieurs raisons :

- Ils peuvent distribuer l'administration des systèmes sur un groupement configurable, incluant la maintenance des associations de sécurité.
- Ils peuvent être utilisés pour concentrer les demandes provenant d'un certain nombre d'équipements de NAS colocalisés ou répartis réglés à un ensemble de groupes d'utilisateurs semblables.
- Ils peuvent effectuer un traitement à valeur ajoutée sur les demandes ou réponses.
- Ils peuvent être utilisés pour l'équilibrage de charge.
- Un réseau complexe aura plusieurs sources d'authentification dont ils peuvent trier les demandes et les transmettre aux bonnes cibles.

Le protocole Diameter exige que les agents conservent l'état de transaction qui est utilisé pour les besoins de reprise sur défaillance. L'état de transaction implique qu'à la transmission d'une demande, son identifiant bond par bond soit sauvegardé ; le champ est remplacé par un identifiant localement unique, qui est restauré à sa valeur d'origine lorsque la réponse correspondante est reçue. L'état de la demande est libéré à réception de la réponse. Un agent sans état est celui qui conserve seulement l'état de transaction.

L'AVP Proxy-Info permet aux agents sans état d'ajouter l'état local à une demande Diameter, avec la garantie que le même état sera présent dans la réponse. Cependant, les procédures de reprise sur défaillance du protocole exigent que les agents

conservent une copie des demandes en cours.

Un agent à états pleins est celui qui conserve les informations d'état de session en gardant trace de toutes les sessions actives autorisées. Chaque session autorisée est liée à un service particulier, et son état est considéré actif jusqu'à ce que soit l'agent soit notifié qu'il en est autrement, soit que la session arrive à expiration. Chaque session autorisée a une heure d'expiration, qui est communiquée par les serveurs Diameter via l'AVP Session-Timeout.

Conserver l'état de session peut être utile dans certaines applications, comme :
- o les traductions de protocole (par exemple, RADIUS <-> Diameter)
- o limiter les ressources autorisées à un usager particulier,
- o un examen par usager ou par transaction.

Un agent Diameter PEUT agir à états pleins pour certaines demandes et être sans état pour d'autres. Une mise en œuvre Diameter PEUT agir comme un type d'agent pour certaines demandes et comme un autre type d'agent pour d'autres.

2.8.1 Agents de relais

Les agents de relais sont des agents Diameter qui acceptent les demandes et acheminent les messages aux autres nœuds Diameter sur la base des informations trouvées dans les messages (par exemple, la valeur de l'AVP Destination-Realm, voir au paragraphe 6.6). Cette décision d'acheminement est effectuée en utilisant une liste de domaines acceptés et d'homologues connus. Ceci est appelé le tableau d'acheminement, comme défini au paragraphe 2.7.

Les relais peuvent, par exemple, être utilisés pour agréger les demandes provenant de plusieurs serveurs d'accès réseau (NAS) au sein d'une zone géographique commune (point de présence, POP). L'utilisation de relais est avantageuse car elle élimine le besoin que les NAS soient configurés avec les informations de sécurité nécessaires qui seraient autrement exigées pour communiquer avec les serveurs Diameter dans d'autres domaines. De même, cela réduit la charge de configuration pour les serveurs Diameter qui serait autrement nécessaire lorsque des NAS sont ajoutés, changés, ou supprimés.

Les relais modifient les messages Diameter en insérant et supprimant des informations d'acheminement, mais ils ne modifient aucune autre portion d'un message. Les relais NE DEVRAIENT PAS conserver l'état de session mais DOIVENT conserver l'état de transaction.

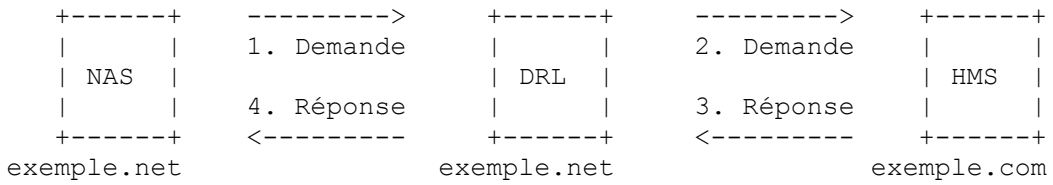


Figure 2 : Relais des messages Diameter

L'exemple de la Figure 2 décrit une demande issue d'un NAS, qui est un appareil d'accès, pour l'usager bob@exemple.com. Avant de produire la demande, le NAS effectue une recherche de chemin Diameter, en utilisant "exemple.com" comme clé, et détermine que le message est à relayer à un DRL, qui est un relais Diameter. Le DRL effectue la même recherche de chemin que le NAS, et relaye le message au HMS, qui est le serveur de rattachement de exemple.com. Le HMS identifie que la demande peut être prise en charge localement (via le domaine) traite la demande d'authentification et/ou autorisation, et répond. Sa réponse est acheminée en retour au NAS en utilisant l'état de transaction sauvegardé.

Comme les relais n'effectuent aucun traitement de niveau application, ils fournissent des services de relais pour toutes les applications Diameter ; donc, ils DOIVENT annoncer l'identifiant d'application de relais.

2.8.2 Agents mandataires

Comme les relais, les agents mandataires acheminent les messages Diameter en utilisant le tableau d'acheminement Diameter. Cependant, ils en diffèrent car ils modifient les messages pour mettre en œuvre l'application de politique. Ceci exige des mandataires qu'ils conservent l'état de leurs homologues vers l'aval (par exemple, les appareils d'accès) pour appliquer l'utilisation des ressources, assurer le contrôle d'admission, et assurer l'approvisionnement.

Les mandataires peuvent, par exemple, être utilisés dans des centres de contrôle d'appel ou chez des FAI d'accès qui fournissent des connexions hors source ; ils peuvent surveiller le nombre et le type des accès utilisés et prendre des décisions d'allocation et d'admission selon leur configuration.

Comme l'application des politiques exige de comprendre le service fourni, les mandataires DOIVENT seulement annoncer les applications Diameter qu'ils prennent en charge.

2.8.3 Agents de redirection

Les agents de redirection sont utiles dans les scénarios où la configuration d'acheminement Diameter doit être centralisée. Un exemple est un agent de redirection qui fournit des services à tous les membres d'un consortium, mais ne souhaite pas s'encombrer à relayer tous les messages entre les domaines. Ce scénario est avantageux car il n'exige pas que le consortium assure les mises à jour d'acheminement pour ses membres lorsque des changements sont apportés à l'infrastructure d'un membre.

Comme les agents de redirection ne relayent pas les messages, et retournent seulement une réponse avec les informations nécessaires pour que les agents Diameter communiquent directement, ils ne modifient pas les messages. Comme les agents de redirection ne reçoivent pas de messages de réponse, ils ne peuvent pas conserver l'état de session.

L'exemple de la Figure 3 décrit une demande issue de l'appareil d'accès, NAS, pour l'usager bob@exemple.com. Le message est transmis par le NAS à son relais, DRL, qui n'a pas d'entrée d'acheminent dans son tableau d'acheminement Diameter pour exemple.com. Le DRL a un chemin par défaut configuré pour DRD, qui est un agent de redirection qui retourne une notification de redirection à DRL, ainsi que les informations de contact de HMS. À réception de la notification de redirection, le DRL établit une connexion de transport avec le HMS, si il n'en existe pas déjà une, et lui transmet la demande.

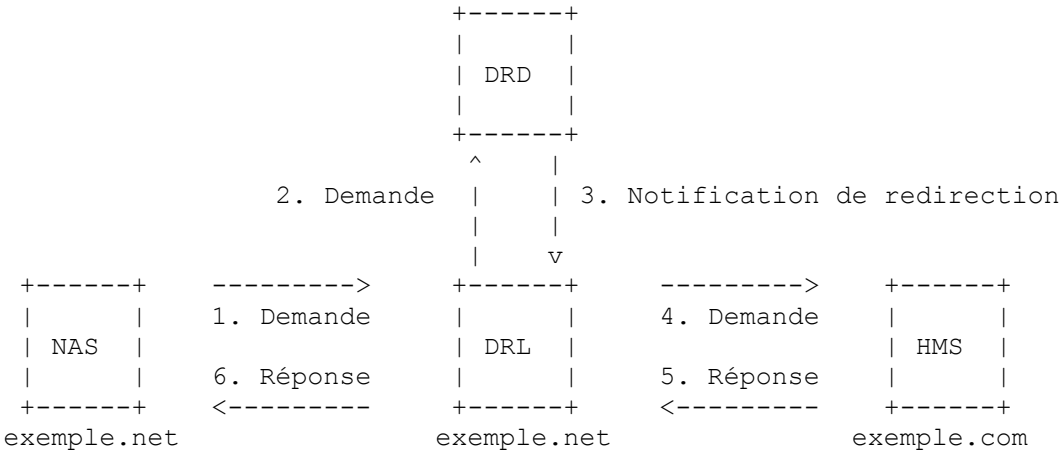


Figure 3 : Redirection d'un message Diameter

Comme les agents de redirection n'effectuent aucun traitement de niveau application, ils fournissent des services de relais pour toutes les applications Diameter ; donc, ils DOIVENT annoncer l'identifiant d'application de relais.

2.8.4 Agents de traduction

Un agent de traduction est un appareil qui assure la traduction entre deux protocoles (par exemple, RADIUS<->Diameter, TACACS+<->Diameter). Les agents de traduction vont vraisemblablement être utilisés comme serveurs d'agrégation pour communiquer avec une infrastructure Diameter, tout en permettant aux systèmes incorporés de migrer à un rythme plus lent.

Comme le protocole Diameter introduit le concept de sessions autorisés sur le long terme, les agents de traduction DOIVENT être à états de session pleins et DOIVENT conserver l'état de transaction. La traduction des messages ne peut se produire que si l'agent reconnaît l'application d'une certaine demande, et donc les agents de traduction DOIVENT seulement annoncer les applications qu'ils prennent en charge localement.

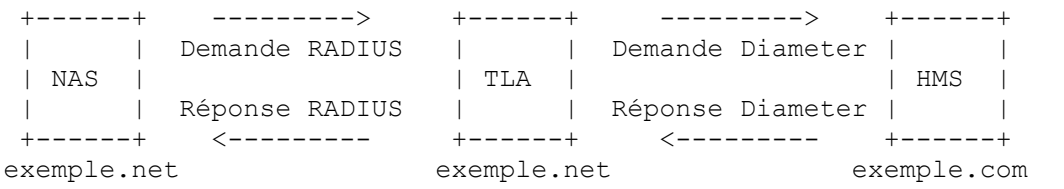


Figure 4: Traduction de RADIUS en Diameter

2.9 Autorisation de chemin Diameter

Comme noté au paragraphe 2.2, Diameter assure la sécurité de niveau transmission pour chaque connexion en utilisant TLS/TCP et DTLS/SCTP. Donc, chaque connexion peut être authentifiée et peut être protégée en intégrité et contre la répétition.

En plus d'authentifier chaque connexion, la session entière DOIT aussi être autorisée. Avant d'initier une connexion, un homologue Diameter DOIT vérifier que ses homologues sont autorisés à agir dans leurs rôles. Par exemple, un homologue Diameter peut être authentique, mais cela ne signifie pas qu'il soit autorisé à agir comme serveur Diameter annonçant un ensemble d'applications Diameter.

Avant d'activer une connexion, des vérifications d'autorisation sont effectuées sur chaque connexion le long du chemin. La négociation des capacités Diameter (CER/CEA) DOIT aussi être conduite, afin de déterminer quelles applications Diameter sont prises en charge par chaque homologue. Les sessions Diameter DOIVENT être acheminées à travers les seuls nœuds autorisés qui ont annoncé la prise en charge de l'application Diameter demandée par la session.

Comme noté au paragraphe 6.1.9, un agent de relais ou mandataire DOIT ajouter une AVP Route-Record à toutes les demandes transmises. L'AVP contient l'identité de l'homologue duquel la demande a été reçue.

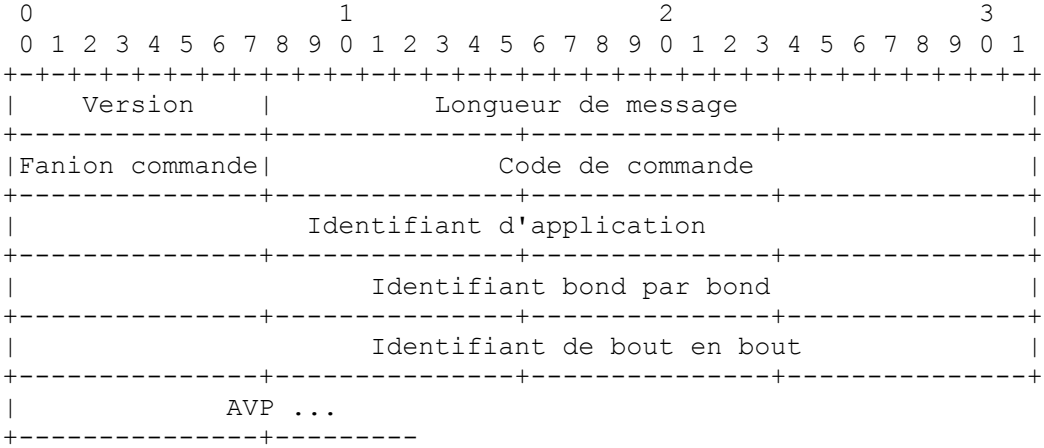
Avant d'autoriser une session, le serveur Diameter de rattachement DOIT vérifier les AVP Route-Record pour s'assurer que le chemin traversé par la demande est acceptable. Par exemple, les administrateurs au sein du domaine de rattachement peuvent ne pas souhaiter honorer les demandes qui ont été acheminées à travers des domaines qui ne sont pas de confiance. En autorisant une demande, le serveur Diameter de rattachement indique implicitement qu'il veut s'engager dans une transaction d'affaire comme spécifié par toute relation contractuelle entre le serveur et le bond précédent. Un message d'erreur DIAMETER_AUTORISATION_REJECTED (voir au paragraphe 7.1.5) est envoyé si le chemin traversé par la demande est inacceptable.

Un domaine de rattachement peut aussi souhaiter vérifier que chaque message de demande de comptabilité correspond à une réponse Diameter autorisant la session. Les demandes de comptabilité sans réponses d'autorisation correspondantes DEVRAIENT être soumises à un examen complémentaire, comme le devraient être les demandes de comptabilité qui indiquent une différence entre le service demandé et celui fourni.

Transmettre une réponse d'autorisation est considéré comme la preuve d'une volonté de prendre un risque financier à l'égard de la session. Un domaine local peut souhaiter limiter ce risque, par exemple, en établissant des limites de crédit pour les domaines intermédiaires et en refusant d'accepter des réponses qui violeraient ces limites. En produisant une demande de comptabilité correspondant à la réponse d'autorisation, le domaine local indique implicitement son accord pour fournir le service indiqué dans la réponse d'autorisation. Si le service ne peut pas être fourni par le domaine local, un message d'erreur DIAMETER_UNABLE_TO_COMPLY DOIT alors être envoyé au sein de la demande de comptabilité ; un client Diameter qui reçoit une réponse d'autorisation pour un service qu'il ne peut pas effectuer NE DOIT PAS substituer un autre service et envoyer ensuite à la place des demandes de comptabilité pour le service de remplacement.

3. En-tête Diameter

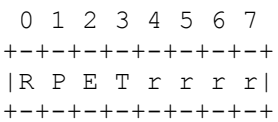
Ci-dessous est présenté un sommaire du format d'en-tête Diameter. Les champs sont transmis dans l'ordre des octets du réseau.



Version : ce champ Version DOIT être réglé à 1 pour indiquer Diameter Version 1.

Longueur de message : le champ Longueur de message fait trois octets et indique la longueur du message Diameter incluant les champs d'en-tête et les AVP avec leur bourrage. Donc, le champ Longueur de message est toujours un multiple de 4.

Fanions de commande : le champ Fanions de commandes fait un octet. Les bits suivants sont alloués :



R(equest) : S'il est à un, le message est une demande. S'il est à zéro, le message est une réponse.

P(roxiable) : S'il est à un, le message PEUT être mandaté, relayé, ou redirigé. S'il est à zéro, le message DOIT être traité en local.

E(rreur) : S'il est à un, le message contient une erreur de protocole, et le message ne va pas se conformer au CCF décrit pour cette commande. Les messages avec le bit 'E' établi sont couramment appelés des messages d'erreur. Ce bit NE DOIT PAS être établi dans les messages de demande (voir au paragraphe 7.2).

T(message potentiellement retransmis) : ce fanion est établi après une procédure de reprise sur défaillance de la liaison, pour aider à la suppression des demandes dupliquées. Il est établi lors du renvoi de demandes non encore acquittées, comme indication d'un possible dupliqué dû à une défaillance de la liaison. Ce bit DOIT être à zéro lors de l'envoi d'une demande pour la première fois ; autrement, l'envoyeur DOIT régler ce fanion à un. Les agents Diameter sont seulement concernés par le nombre de demandes qu'ils envoient sur la base d'une seule demande reçue ; les retransmissions par d'autres entités n'ont pas besoin d'être retracées. Les agents Diameter qui reçoivent une demande avec le fanion T établi, DOIVENT garder le fanion T établi dans la demande transmise. Ce fanion NE DOIT PAS être établi si un message de réponse d'erreur (par exemple, une erreur de protocole) a été reçu pour le message antérieur. Il ne peut être établi que dans les cas où aucune réponse n'a été reçue du serveur pour une demande, et la demande a été envoyée à nouveau. Ce fanion NE DOIT PAS être établi dans un message de réponse.

r(éservé) : ces bits fanions sont réservés pour une utilisation future ; ils DOIVENT être réglés à zéro et ignorés par le receveur.

Code de commande : le champ Code de commande fait trois octets et est utilisé afin de communiquer la commande associée au message. L'espace d'adresse de 24 bits est géré par l'IANA (voir au paragraphe 3.1). Les valeurs de code de commande 16 777 214 et 16 777 215 (valeurs hexadécimales FFFFFE-FFFFFF) sont réservées aux utilisations expérimentales (voir au paragraphe 11.2).

Identifiant d'application : l'identifiant d'application fait quatre octets et est utilisé pour identifier l'application à laquelle le message est applicable. L'application peut être une application d'authentification, une application de comptabilité, ou une application spécifique du fabricant. La valeur du champ Identifiant d'application dans l'en-tête DOIT être la même que celle des AVP Identifiant d'application pertinentes contenues dans le message.

Identifiant bond par bond : l'identifiant bond par bond est un champ d'entier non signé de 32 bits (dans l'ordre des octets du réseau) qui aide à faire correspondre les demandes et les réponses. L'envoyeur DOIT s'assurer que l'identifiant bond par bond dans une demande est unique sur une connexion à tout moment, et il PEUT tenter de s'assurer que le nombre est unique à travers les réamorçages. L'envoyeur d'un message de réponse DOIT s'assurer que le champ Identifiant bond par bond contient la même valeur que celle trouvée dans la demande correspondante. L'identifiant bond par bond est normalement un nombre à accroissement monotone, dont la valeur de début a été générée au hasard. Un message de réponse qui est reçu avec un identifiant bond par bond inconnu DOIT être éliminé.

Identifiant de bout en bout : l'identifiant de bout en bout est un champ d'entier non signé de 32 bits (dans l'ordre des octets du réseau) qui est utilisé pour détecter les messages dupliqués. Au réamorçage, la mise en œuvre PEUT régler les 12 bits de poids fort à contenir les 12 bits de moindre poids de l'heure courante, et les 20 bits de moindre poids à une valeur aléatoire. Les envoyeurs de messages de demande DOIVENT insérer un identifiant univoque sur chaque message. L'identifiant DOIT rester localement unique pendant au moins 4 minutes, même à travers des réamorçages. Le générateur d'un message de réponse DOIT s'assurer que le champ Identifiant de bout en bout contient la même valeur que celle trouvée dans la demande correspondante. L'identifiant de bout en bout NE DOIT PAS être modifié par les agents Diameter de toute sorte. La combinaison de l'AVP Origin-Host (paragraphe 6.3) et de ce champ est utilisée pour détecter les dupliqués. Les demandes dupliquées DEVRAIENT causer la transmission de la même réponse (modulo le champ Identifiant bond par bond et toutes AVP d'acheminement qui peuvent être présentes) et elles NE DOIVENT PAS

affecter d'état qui a été établi lorsque la demande originale a été traitée. Les messages de réponse dupliqués qui sont pour la consommation locale (voir au paragraphe 6.2) DEVRAIENT être éliminés en silence.

AVP : les AVP sont une méthode d'encapsulation des informations relatives au message Diameter. Voir plus d'informations sur les AVP à la Section 4.

3.1 Codes de commandes

Un code de commande est alloué à chaque paire de commandes demande/réponse, et le sous type (c'est-à-dire, demande ou réponse) est identifié via le bit 'R' dans le champ Fanions de commande de l'en-tête Diameter.

Chaque message Diameter DOIT contenir un code de commande dans le champ Code de commande de son en-tête, qui est utilisé pour déterminer l'action prise pour un message particulier. Les codes de commande suivants sont définis dans le protocole de base Diameter :

Nom de commande		**Abrév.**	**Code**	**Paragraphe**
Abort-Session-Request	*(demande d'interruption de session)*	ASR	274	8.5.1
Abort-Session-Answer	*(réponse d'interruption de session)*	ASA	274	8.5.2
Accounting-Request	*(demande de comptabilité)*	ACR	271	9.7.1
Accounting-Answer	*(réponse de comptabilité)*	ACA	271	9.7.2
Capabilities-Exchange-Request	*(demande d'échange de capacités)*	CER	257	5.3.1
Capabilities-Exchange-Answer	*(réponse d'échange de capacités)*	CEA	257	5.3.2
Device-Watchdog-Request	*(demande de chien de garde d'appareil)*	DWR	280	5.5.1
Device-Watchdog-Answer	*(réponse de chien de garde d'appareil)*	DWA	280	5.5.2
Disconnect-Peer-Request	*(demande de déconnexion de l'homologue)*	DPR	282	5.4.1
Disconnect-Peer-Answer	*(réponse de déconnexion de l'homologue)*	DPA	282	5.4.2
Re-Auth-Request	*(demande de réautorisation)*	RAR	258	8.3.1
Re-Auth-Answer	*(réponse de réautorisation)*	RAA	258	8.3.2
Session-Termination-Request	*(demande de terminaison de session)*	STR	275	8.4.1
Session-Termination-Answer	*(réponse de terminaison de session)*	STA	275	8.4.2

3.2 Spécification du format de code de commande

Chaque code de commande défini DOIT inclure une spécification de format de code de commande Format (CCF) correspondant, qui est utilisé pour définir les AVP qui DOIVENT ou PEUVENT être présentes lors de l'envoi du message. L'ABNF qui suit spécifie le CCF utilisé dans la définition:

définition de commande = "<" nom de commande ">" "::=" message diameter

nom de commande = nom diameter

nom diameter = ALPHA *(ALPHA / CHIFFRE / "-")

message diameter = en-tête *fixé *exigé *facultatif

en-tête = "<En-tête diameter:" identifiant de commande [bit r] [bit p] [bit e] [identifiant d'application]">"

identifiant d'application = 1*identifiant d'application

identifiant de commande = 1*identifiant d'application ; code de commande alloué à la commande.

bit r = ", REQ" ; si présent, le bit 'R' dans les fanions de commande est établi, indiquant que le message est une demande par opposition à une réponse.

bit p = ", PXY" ; si présent, le bit 'P' dans les fanions de commande est établi, indiquant que le message peut être mandaté

bit e = ", ERR" ; si présent, le bit 'E' dans les fanions de commande est établi, indiquant que le message de réponse contient une AVP Code de résultat dans la classe "erreur de protocole".

fixé = [qual] "<" avp-spec ">" ; définit la position fixée d'une AVP.

exigé = [qual] "{" avp-spec "}" ; l'AVP DOIT être présente et peut apparaître n'importe où dans le message.

facultatif = [qual] "[" avp-name "]" ; avp-name dans la règle 'facultatif' ne peut pas s'évaluer comme nom d'AVP inclus dans une règle fixé ou exigé. L'AVP peut apparaître n'importe où dans le message.

Note : "[" et "]" ont une signification légèrement différente de celle de l'ABNF. Ces crochets ne peuvent pas être utilisés pour exprimer des règles facultatives fixées (comme un ICV facultatif à la fin). Pour cela, la convention est '0*1fixé'.

qual = [min] "*" [max] ; voir les conventions d'ABNF, RFC 5234, Section 4. L'absence de tout qualificatif dépend de si il précède une règle fixé, exigé, ou facultatif. Si une règle fixé ou exigé n'a pas de qualificatif, exactement une telle AVP DOIT alors être présente. Si une règle facultatif n'a pas de qualificatif, 0 ou 1 telle AVP peut alors être présente. Si une règle facultatif a un qualificatif la valeur de min DOIT alors être 0 si elle est présente.

min = 1*CHIFFRE ; nombre minimum de fois que l'élément peut être présent. S'il est absent, la valeur par défaut est 0 pour les règles fixé et facultatif et 1 pour les règles exigé. La valeur DOIT être au moins 1 pour les règles exigé.

max = 1*CHIFFRE ; nombre maximum de fois que l'élément peut être présent. S'il est absent, la valeur par défaut est l'infini. Une valeur de 0 implique que l'AVP NE DOIT PAS être présente.

avp-spec = nom diameter ; avp-spec doit être un nom d'AVP, défini dans les spécifications Diameter de base ou d'extension.

avp-name = avp-spec / "AVP" ; la chaîne "AVP" signifie *tout* Nom d'AVP arbitraire, non autrement cité dans cette définition de code de commande. L'inclusion de cette chaîne est recommandée pour toutes les CCF pour permettre l'extensibilité.

Voici une définition d'un code de commande fictif :

```
Demande-Exemple ::= < En-tête Diameter : 9999999, REQ, PXY >
                        { Nom d'utilisateur }
                     1* { Hôte d'origine }
                      * [ AVP ]
```

3.3 Conventions de dénomination des commandes Diameter

Les noms de commandes Diameter incluent normalement un ou plusieurs mots anglais suivis par le verbe "Request" *(demander)* ou "Answer" *(répondre)*. Chaque mot anglais est délimité par un tiret. Un acronyme de trois lettre pour la demande et la réponse est normalement aussi fourni.

Un exemple est un ensemble de messages utilisés pour terminer une session. Le nom de la commande est Session-Terminate-Request et Session-Terminate-Answer, tandis que les acronymes sont respectivement STR et STA.

La demande et la réponse pour une certaine commande partagent toutes deux le même code de commande. La demande est identifiée par le bit R(equest) dans l'en-tête Diameter réglé à un (1), pour demander qu'une action particulière soit effectuée, comme d'autoriser un usager ou terminer une session. Une fois que le receveur a achevé la demande, il produit la réponse correspondante, qui inclut un code de résultat qui communique un de ce qui suit :

- o la demande a réussi
- o la demande a échoué
- o une demande supplémentaire doit être envoyée pour fournir les informations que l'homologue demande avant de retourner une réponse de succès ou d'échec
- o le receveur n'a pas pu traiter la demande, mais fournit des informations sur un homologue Diameter qui est capable de satisfaire la demande, qu'on appelle une redirection.

Des informations supplémentaires, codées dans les AVP, peuvent aussi être incluses dans des messages de réponse.

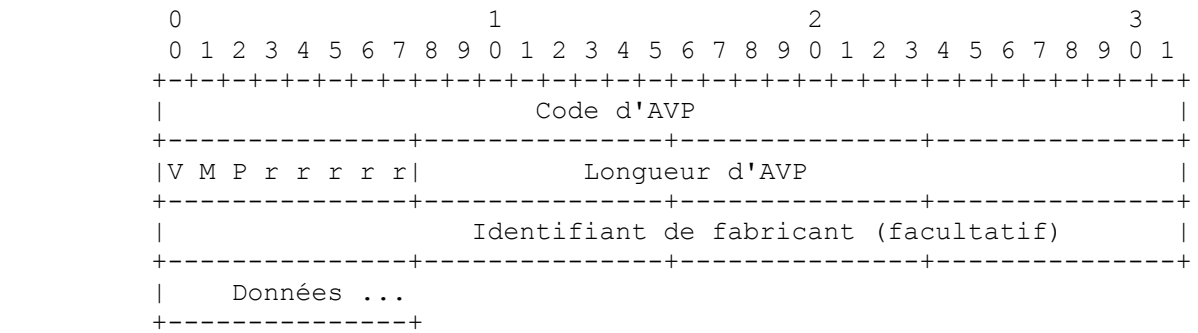
4. AVP Diameter

Les AVP Diameter portent des informations spécifiques d'authentification, de comptabilité, d'autorisation, et d'acheminement ainsi que des détails de configuration pour la demande et la réponse.

Chaque AVP de type OctetString DOIT être bourrée pour s'aligner sur une limite de 32 bits, tandis que les autres types d'AVP s'alignent naturellement. Un certain nombre d'octets de valeur zéro sont ajoutés à la fin du champ Données de l'AVP jusqu'à atteindre une limite de mot. La longueur du bourrage n'est pas reflétée dans le champ Longueur de l'AVP.

4.1 En-tête d'AVP

Les champs dans l'en-tête d'AVP DOIVENT être envoyés dans l'ordre des octets du réseau. Le format de l'en-tête est :



Code d'AVP : le Code d'AVP, combiné avec le champ Identifiant de fabricant, identifie l'attribut de façon univoque. Les numéros d'AVP de 1 à 255 sont réservés pour la réutilisation des attributs RADIUS, sans établir le champ Identifiant de fabricant. Les numéros d'AVP 256 et au dessus sont utilisés pour Diameter, et sont alloués par l'IANA (voir au paragraphe 11.1.1).

Fanions d'AVP : le champ Fanions d'AVP informe le receveur sur la façon dont chaque attribut doit être traité. Les nouvelles applications Diameter NE DEVRAIENT PAS définir de bits fanions d'AVP supplémentaires. Cependant, noter que de nouvelles applications Diameter PEUVENT définir des bits supplémentaires au sein de l'en-tête d'AVP, et un bit non reconnu DEVRAIT être considéré comme une erreur. L'envoyeur de l'AVP DOIT régler les bits 'r' (réservé) à 0 et le receveur DEVRAIT ignorer tous les bits 'r' (réservé). Le bit 'P' a été réservé à un futur usage de sécurité de bout en bout. Au moment de la rédaction du présent document, il n'y a pas de mécanisme de sécurité de bout en bout spécifié ; donc le bit 'P' DEVRAIT être réglé à 0.

Le bit 'M', appelé le bit Obligatoire, indique si le receveur de l'AVP DOIT analyser et comprendre la sémantique de l'AVP incluant son contenu. L'entité receveuse DOIT retourner un message d'erreur approprié si elle reçoit une AVP qui a le bit M établi mais qu'elle ne comprend pas. Une exception s'applique quand l'AVP est incorporée dans une AVP Grouped. Voir les détails au paragraphe 4.4. Les relais et agents de redirection Diameter NE DOIVENT PAS rejeter les messages qui ont des AVP non reconnues. Le bit 'M' DOIT être réglé conformément aux règles définies dans la spécification de l'application qui introduit ou réutilise cette AVP. Dans une certaine application, le réglage du bit M pour une AVP est défini soit pour tous les types de commandes soit pour chaque type de commande. Les AVP avec le bit 'M' à zéro ne sont que pour information ; un receveur qui reçoit un message avec une telle AVP qui n'est pas prise en charge, ou dont la valeur n'est pas prise en charge PEUT simplement ignorer l'AVP.

Le bit 'V', qui est le bit spécifique du fabricant, indique si le champ facultatif Identifiant de fabricant est présent dans l'en-tête de l'AVP. Lorsque il est établi, le code d'AVP appartient à l'espace d'adresse de code spécifique de fabricant.

Longueur d'AVP : le champ Longueur d'AVP fait trois octets, et indique le nombre d'octets dans cette AVP incluant le champ Code d'AVP, le champ Longueur d'AVP, le champ Fanions d'AVP, le champ Identifiant de fabricant (si il est présent), et le champ Données d'AVP. Si un message est reçu avec une longueur d'attribut invalide, le message DOIT être rejeté.

4.1.1 Éléments d'en-tête facultatifs

L'en-tête d'AVP contient un champ facultatif. Ce champ n'est présent que si le bit fanion conservé est établi.

Identifiant de fabricant *(Vendor-ID)* : le champ Identifiant de fabricant est présent si le bit 'V' est établi dans le champ

Fanions d'AVP. Le champ facultatif Identifiant de fabricant de quatre octets contient la valeur allouée par l'IANA de "SMI Network Management Private Enterprise Codes" [ENTERPRISE], codée dans l'ordre des octets du réseau. Tout fabricant ou toute organisation de normalisation qui est aussi traitée comme fabricant dans l'espace géré par l'IANA "SMI Network Management Private Enterprise Codes" qui souhaite mettre en œuvre une AVP Diameter spécifique de fabricant DOIT utiliser son propre identifiant de fabricant avec son espace d'adresses d'AVP à gestion privée, ce qui garantit qu'il n'y aura pas de collision avec d'autres AVP spécifiques de fabricant ou avec de futures AVP de l'IETF.

Une valeur d'identifiant de fabricant de zéro (0) correspond aux valeurs d'AVP adoptées par l'IETF, gérées par l'IANA. Comme l'absence du champ Identifiant de fabricant implique que l'AVP en question n'est pas spécifique de fabricant, la mise en œuvre NE DOIT PAS utiliser la valeur zéro (0) pour le champ Vendor-ID.

4.2 Formats de base de données d'AVP

Le champ Données fait zéro, un ou plusieurs octets et contient des informations spécifiques de l'attribut. Le format et la longueur du champ Données sont déterminés par les champs Code d'AVP et Longueur d'AVP. Le format du champ Données DOIT être un des types de données de base ou un type de données déduit des types de données de base suivants. Dans le cas où un nouveau format d'AVP Données de base serait nécessaire, une nouvelle version de la présente RFC DEVRA être créée.

OctetString : les données contiennent des données arbitraires de longueur variable. Sauf mention contraire, le champ Longueur d'AVP DOIT être réglé à au moins 8 (12 si le bit 'V' est activé). Les valeurs d'AVP de ce type qui ne sont pas des multiples de 4 octets sont suivies par le bourrage nécessaire afin que la prochaine AVP (si il en est) commence sur une limite de 32 bits.

Integer32 : valeur signée de 32 bits, dans l'ordre des octets du réseau. Le champ Longueur d'AVP DOIT être réglé à 12 (16 si le bit 'V' est activé).

Integer64 : valeur signée de 64 bits, dans l'ordre des octets du réseau. Le champ Longueur d'AVP DOIT être réglé à 16 (20 si le bit 'V' est activé).

Unsigned32 : valeur non signée de 32 bis, dans l'ordre des octets du réseau. Le champ Longueur d'AVP DOIT être réglé à 12 (16 si le bit 'V' est activé).

Unsigned64 : valeur non signée de 64 bits, dans l'ordre des octets du réseau. Le champ Longueur d'AVP DOIT être réglé à 16 (20 si le bit 'V' est activé).

Float32 : cela représente des valeurs de précision seule à virgule flottante, comme décrit dans [FLOATPOINT]. La valeur de 32 bits est transmise dans l'ordre des octets du réseau. Le champ Longueur d'AVP DOIT être réglé à 12 (16 si le bit 'V' est activé).

Float64 : cela représente des valeurs de double précision à virgule flottante, comme décrit dans [FLOATPOINT]. La valeur de 64 bits est transmise dans l'ordre des octets du réseau. Le champ Longueur d'AVP DOIT être réglé à 16 (20 si le bit 'V' est activé).

Grouped : le champ Données est spécifié comme une séquence d'AVP. Ces AVP sont enchaînées – incluant leurs en-têtes et bourrages – dans l'ordre dans lequel elles sont spécifiées et le résultat est encapsulé dans le champ Données. Le champ Longueur d'AVP est réglé à 8 (12 si le bit 'V' est activé) plus la longueur totale de toutes les AVP incluses, y compris leurs en-têtes et bourrages. Donc, le champ Longueur d'AVP d'une AVP de type Grouped est toujours un multiple de 4.

4.3 Formats dérivés de données d'AVP

En plus d'utiliser les formats d'AVP Données de base, les applications peuvent définir des formats de données dérivés des formats d'AVP Données de base. Une application qui définit de nouveaux formats d'AVP Données dérivés DOIT les inclure dans une section intitulée "Formats de données d'AVP dérivés", utilisant le même format que celui des définitions ci-dessous. Chaque nouvelle définition DOIT être définie ou citée avec une référence à la RFC qui définit le format.

4.3.1 Formats courants d'AVP Données dérivés

Les formats d'AVP Données dérivés suivants sont couramment utilisés.

Address : le format Address est dérivé du format d'AVP de base OctetString *(chaîne d'octets)*. C'est une union discriminée

qui représente, par exemple, une adresse de 32 bits (IPv4) [RFC0791] ou de 128 bits (IPv6) [RFC4291] dont l'octet de poids fort est en premier. Les deux premiers octets de l'AVP Address représentent le type d'adresse, qui contient une famille d'adresses, définie dans [IANAADFAM]. AddressType est utilisé pour différencier le contenu et le format des octets restants.

Time : le format Time est dérivé du format d'AVP de base OctetString. La chaîne DOIT contenir quatre octets, dans le même format que les quatre premiers octets du format d'horodatage NTP. Le format d'horodatage NTP est défini à la Section 3 de la [RFC5905]. Cela représente le nombre de secondes depuis le 1er janvier 1900 à 0 h par rapport au temps universel coordonné (UTC). À 6 h 28 m 16 s UTC, le 7 février 2036, la valeur de l'heure va déborder. Le protocole simple de l'heure du réseau (SNTP, *Simple Network Time Protocol*) [RFC5905] décrit une procédure pour étendre l'heure à 2104. Cette procédure DOIT être prise en charge par tous les nœuds Diameter.

UTF8String : le format UTF8String est dérivé du format d'AVP de base OctetString. C'est une chaîne lisible par l'homme représentée à l'aide du jeu de caractères de la norme internationale ISO/CEI IS 10646-1, codée comme chaîne d'octets en utilisant le format de transformation UTF-8 [RFC3629]. Comme des codets supplémentaires sont ajoutés de temps en temps par des amendements à la norme 10646, les mises en œuvre DOIVENT être prêtes à rencontrer tout codet de 0x00000001 à 0x7fffffff. Les séquences d'octet qui ne correspondent pas au codage valide d'un codet dans le jeu de caractères UTF-8 ou sont hors de cette gamme sont interdits. L'utilisation de codes de commandes DEVRAIT être évitée. Lorsque il est nécessaire de représenter une nouvelle ligne, la séquence de code de commande CR LF DEVRAIT être utilisée. L'utilisation d'espaces blanches en tête ou en queue DEVRAIT être évitée. Pour des codets non directement pris en charge par le matériel ou logiciel d'interface d'utilisateur, un moyen de remplacement d'entrée et d'affichage, comme l'hexadécimal, PEUT être fourni. Pour les informations codées en US-ASCII à 7 bits , le jeu de caractères UTF-8 est identique au jeu de caractères US-ASCII. L'UTF-8 peut exiger plusieurs octets pour représenter un seul caractère/codet ; donc, la longueur d'une UTF8String en octets peut être différente du nombre de caractères codés. Noter que le champ Longueur d'AVP d'une UTF8String est mesuré en octets et non en caractères.

DiameterIdentity : le format DiameterIdentity est dérivé du format d'AVP de base OctetString. DiameterIdentity = FQDN/Realm. La valeur DiameterIdentity est utilisée pour identifier de façon univoque un nœud Diameter pour les besoins de détection des connexions dupliquées et des boucles d'acheminement, et pour qu'un domaine détermine si les messages peuvent être satisfaits en local ou si ils doivent être acheminés ou redirigés. Lorsque une valeur DiameterIdentity est utilisée pour identifier un nœud Diameter, le contenu de la chaîne DOIT être le nom de domaine pleinement qualifié (FQDN, *Fully Qualified Domain Name*) du nœud Diameter. Si plusieurs nœuds Diameter fonctionnent sur le même hôte, chaque nœud Diameter DOIT avoir allouée une DiameterIdentity univoque. Si un nœud Diameter peut être identifié par plusieurs FQDN, un seul FQDN devrait être retenu au départ et utilisé comme seule DiameterIdentity pour ce nœud, quelle que soit la connexion sur laquelle il est envoyé. Dans le présent document, on note que DiameterIdentity est en forme ASCII afin d'être compatible avec l'infrastructure existante du DNS. Voir à l'Appendice D les interactions entre le protocole Diameter et les noms de domaines internationalisés (IDN).

DiameterURI : DiameterURI DOIT suivre les règles de la syntaxe des identifiants de ressource universels [RFC3986] spécifiée ci-dessous :

"aaa://" FQDN [accès] [transport] [protocole] ; pas de sécurité du transport
"aaas://" FQDN [accès] [transport] [protocole] ; utilisation de la sécurité du transport
FQDN = < nom de domaine pleinement qualifié >
accès = ":" 1*CHIFFRE ; un des accès utilisés pour écouter les connexions entrantes. S'il est absent, l'accès Diameter par défaut (3868) est supposé si aucune sécurité de transport n'est utilisée et l'accès 5658 lorsque la sécurité du transport est utilisée (TLS/TCP et DTLS/SCTP).
transport = ";transport=" transport-protocol ; un des transports utilisés pour écouter les connexions entrantes. S'il est absent, le protocole par défaut est supposé être TCP. UDP NE DOIT PAS être utilisé lorsque le champ aaa-protocol est réglé à diameter.
transport-protocol = ("tcp" / "sctp" / "udp")
protocol = ";protocol=" aaa-protocol ; s'il est absent, le protocole AAA par défaut est Diameter.
aaa-protocol = ("diameter" / "radius" / "tacacs+")

Voici des exemples d'identités valides d'hôte Diameter :

aaa://host.exemple.com;transport=tcp
aaa://host.exemple.com:6666;transport=tcp
aaa://host.exemple.com;protocol=diameter
aaa://host.exemple.com:6666;protocol=diameter
aaa://host.exemple.com:6666;transport=tcp;protocol=diameter
aaa://host.exemple.com:1813;transport=udp;protocol=radius

Enumerated : le format Enumerated est dérivé du format d'AVP de base Integer32. La définition contient une liste de valeurs valides et leur interprétation et est décrite dans l'application Diameter qui introduit l'AVP.

IPFilterRule : le format IPFilterRule est dérivé du format d'AVP de base OctetString et utilise le jeu de caractères ASCII. La syntaxe de la règle est un sous ensemble modifié de ipfw(8) de FreeBSD. Les paquets peuvent être filtrés sur la base des informations suivantes qui y sont associées :

Direction (entrante ou sortante)
Adresse IP de source et destination (éventuellement masquée)
Protocole
Accès de source et destination (listes ou gammes)
Fanions TCP
Fanion de fragment IP
Options IP
Types ICMP

Les règles pour la direction appropriée sont évaluées dans l'ordre, la première règle qui correspond termine l'évaluation. Chaque paquet est évalué une fois. Si aucune règle ne correspond, le paquet est éliminé si la dernière règle évaluée était une permission, et passé si la dernière règle était un refus.

Les filtres IPFilterRule DOIVENT respecter le format : action dir proto de source à destination [options]

action : permit – permet les paquets qui satisfont la règle
deny – élimine les paquets qui satisfont la règle.

dir : "in" est en provenance du terminal, "out" est vers le terminal.

proto : un protocole IP spécifié par un numéro. le mot clé "ip" signifie que tout protocole va correspondre.

source et destination : <address/mask> [accès]

<address/mask> peut être spécifié comme : ipno : un numéro IPv4 ou IPv6 en forme quadratique séparée par des points ou en forme canonique IPv6. Seul ce numéro IP exact va satisfaire la règle.

ipno/bits : numéro IP comme ci-dessus avec une largeur de gabarit de la forme 192.0.2.10/24. Dans ce cas, tous les numéros IP de 192.0.2.0 à 192.0.2.255 vont correspondre. La largeur binaire DOIT être valide pour la version IP, et le numéro IP NE DOIT PAS avoir de bits établis au delà du gabarit. Pour qu'une correspondance se produise, doit être présente dans le paquet la même version IP que celle utilisée dans la description de l'adresse IP. Pour vérifier une version IP particulière, la partie bits peut être réglée à zéro. Le mot clé "any" est 0.0.0.0/0 ou l'équivalent IPv6. Le mot clé "assigned" est l'adresse ou l'ensemble d'adresses allouées au terminal. Pour IPv4, une première règle typique est souvent "deny in ip! assigned".

Le sens de la correspondance peut être inversé en faisant précéder une adresse du signe ne pas modifier (!), ce qui cause à la place la correspondance de toutes les autres adresses. Cela n'affecte pas la sélection des numéros d'accès.

Avec les protocoles TCP, UDP, et SCTP, des accès facultatifs peuvent être spécifiés comme : {port/port-port}[,ports[,...]]

La notation '-' spécifie une gamme d'accès (incluant les frontières).

Les paquets fragmentés qui ont un décalage différent de zéro (c'est-à-dire, qui ne sont pas le premier fragment) ne vont jamais satisfaire une règle qui a une ou plusieurs spécifications d'accès. Voir l'option frag pour les détails sur la correspondances des paquets fragmentés.

options : frag. Elle correspond si le paquet est un fragment et que ce n'est pas le premier fragment du datagramme. frag ne peut pas être utilisé en conjonction avec tcpflags ou des spécifications d'accès TCP/UDP.

ipoptions spec : correspond si l'en-tête IP contient la liste séparée par des virgules des options spécifiées dans spec. Les options IP prises en charge sont : ssrr (chemin de source strict), lsrr (chemin de source lâche), rr (enregistrer le chemin du paquet), et ts (horodatage). L'absence d'une option particulière peut être notée avec un '!'.

tcpoptions spec : correspond si l'en-tête IP contient la liste séparée par des virgules des options spécifiées dans spec. Les options IP prises en charge sont : mss (taille maximum de segment), window (annonce de fenêtre tcp), sack (accusé de réception sélectif), ts (horodatage de la rfc1323), et cc (compte de connexion t/tcp de la rfc1644). L'absence d'une option particulière peut être notée avec un '!'.

established : seulement les paquets TCP. Correspondent les paquets qui ont le bit RST ou ACK établi.

setup : seulement les paquets TCP. Correspondent les paquets qui ont le bit SYN établi mais pas de bit ACK.

tcpflags spec : seulement les paquets TCP. Correspond si l'en-tête TCP contient la liste séparée par des virgules des fanions spécifiés dans spec. Les fanions TCP pris en charge sont : fin, syn, rst, psh, ack, et urg. L'absence d'un fanion particulier peut être notée avec un '!'. Une règle qui contient une spécification tcpflags ne peut jamais correspondre à un paquet fragmenté qui a un décalage différent de zéro. Voir les détails de l'option frag sur la correspondance des paquets fragmentés.

icmptypes types : seulement les paquets ICMP. Correspond si le type ICMP est dans la liste des types. La liste peut être spécifiée comme toute combinaison de gammes ou de types individuels séparés par des virgules. Les valeurs numériques et les valeurs symboliques mentionnées ci-dessous peuvent être utilisées. Les types ICMP pris en charge sont : echo reply *(réponse d'écho)* (0), destination unreachable *(destination injoignable)* (3), source quench *(assourdissement de source)* (4), redirect *(redirection)* (5), echo request *(demande d'écho)* (8), router advertisement *(annonce de routeur)* (9), router solicitation *(sollicitation de routeur)* (10), time-to-live exceeded *(durée de vie dépassée)* (11), IP header bad *(mauvais en-tête IP)* (12), timestamp request *(demande d'horodatage)* (13), timestamp reply *(réponse d'horodatage)* (14), information request *(demande d'informations)* (15), information reply *(réponse d'information)* (16), address mask request *(demande de gabarit d'adresse)* (17), et address mask reply (18).

Il y a une sorte de paquet que l'appareil d'accès DOIT toujours éliminer, qui est un fragment IP avec un décalage de fragment de un. C'est un paquet valide, mais il n'a qu'une utilité, celle d'essayer de circonvenir les pare-feu.

Un appareil d'accès qui n'est pas capable d'interpréter ou appliquer une règle 'deny' DOIT terminer la session. Un appareil d'accès qui n'est pas capable d'interpréter ou appliquer une règle 'permit' PEUT appliquer une règle plus restrictive. Un appareil d'accès PEUT appliquer des règles de refus de lui-même avant les règles fournies, par exemple pour protéger l'infrastructure du possesseur de l'appareil d'accès.

4.4 Valeurs d'AVP Grouped

Le protocole Diameter permet des valeurs d'AVP de type 'Grouped'. Cela implique que le champ Data soit en fait une séquence d'AVP. Il est possible d'inclure une AVP de type Grouped au sein d'un type Grouped, c'est-à-dire de les incorporer l'une dans l'autre. Les AVP au sein d'une AVP de type Grouped ont les mêmes exigences de bourrage que les AVP non groupées, comme défini au paragraphe 4.4.

L'espace de numérotation de code AVP de toutes les AVP incluses dans une AVP Grouped est le même que pour les AVP non groupées. Les receveurs d'une AVP Grouped qui n'a pas le bit 'M' (obligatoire) établi (à 1) et une ou plusieurs des AVP encapsulées au sein du groupe ont le bit 'M' établi PEUVENT simplement l'ignorer si l'AVP Grouped elle-même n'est pas reconnue. La règle s'applique même si l'AVP encapsulée avec son bit 'M' établi est à son tout encapsulée au sein d'autres sous groupes, c'est-à-dire, d'autres AVP Grouped incorporées au sein de l'AVP Grouped.

Chaque définition d'AVP Grouped DOIT inclure une grammaire correspondante, utilisant l'ABNF [RFC5234] (avec modifications) comme défini ci-dessous :

grouped-avp-def = "<" nom ">" "::=" avp

name-fmt = ALPHA *(ALPHA / CHIFFRE / "-")

nom = name-fmt ; Le nom doit être celui d'une AVP, définie dans les spécifications Diameter de base ou étendues.

avp = en-tête *fixé *exigé *facultatif

en-tête = "<" "En-tête d'AVP:" avpcode [fabricant] ">"

avpcode = 1*CHIFFRE ; Code d'AVP alloué à l'AVP Grouped.

fabricant = 1*CHIFFRE ; identifiant de fabricant alloué à l'AVP Grouped. Absent, la valeur par défaut zéro est utilisée.

4.4.1 Exemple d'AVP avec le type de données Grouped

Exemple-AVP (code AVP 999999) est de type Grouped et est utilisé pour préciser comment fonctionnent les valeurs d'AVP Grouped. Le champ Grouped Data a la grammaire CCF suivante :

```
Exemple-AVP  ::= < En-tête d'AVP : 999999 >
                  { Origin-Host }
                 1*{ Session-Id }
                  *[ AVP ]
```

Voici un exemple d'AVP avec Grouped Data :
L'AVP Origin-Host (paragraphe 6.3) est requis. Dans ce cas : Origin-Host = "exemple.com".
Une ou plusieurs identifiants de session doivent suivre. Ici, il y en a deux :
Session-Id = "grump.exemple.com:33041;23432;893;0AF3B81"
Session-Id = "grump.exemple.com:33054;23561;2358;0AF3B82"

Les AVP facultatives incluses sont :

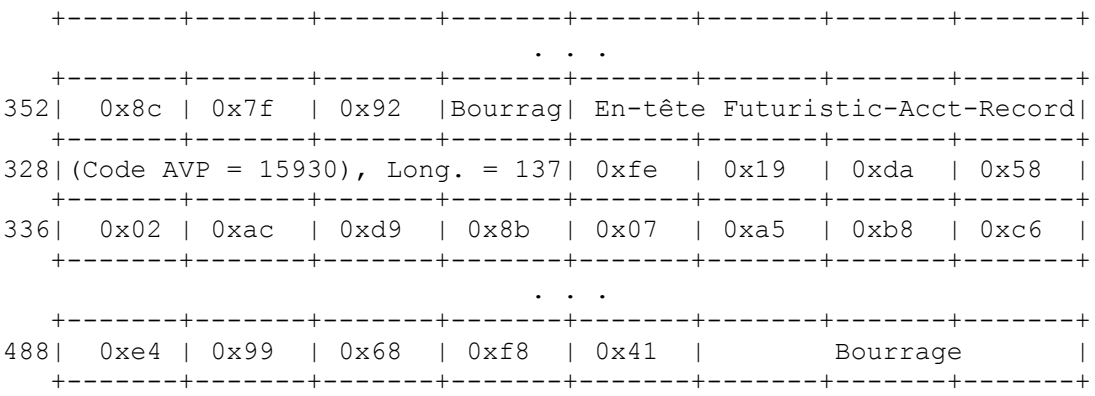
Recovery-Policy = <binary>
2163bc1d0ad82371f6bc09484133c3f09ad74a0dd5346d54195a7cf0b352cabc881839a4fdcfbc1769e2677a4c1fb499284c5f7
0b48f58503a45c5c2d6943f82d5930f2b7c1da640f476f0e9c9572a50db8ea6e51e1c2c7bdf8bb43dc995144b8dbe297ac73949
3946803e1cee3e15d9b765008a1b2acf4ac777c80041d72c01e691cf751dbf86e85f509f3988e5875dc90511926841f00f0e29a
6d1ddc1a842289d440268681e052b30fb638045f7779c1d873c784f054f688f5001559ecff64865ef975f3e60d2fd7966b8c7f92

Futuristic-Acct-Record = <binary>
fe19da5802acd98b07a5b86cb4d5d03f0314ab9ef1ad0b67111ff3b90a057fe29620bf3585fd2dd9fcc38ce62f6cc208c6163c00
8f4258d1bc88b817694a74ccad3ec69269461b14b2e7a4c111fb239e33714da207983f58c41d018d56fe938f3cbf089aac12a91
2a2f0d1923a9390e5f789cb2e5067d3427475e49968f841

Les données pour les AVP facultatives sont représentés en hexadécimal car le format de ces AVP n'est pas connu au moment de la définition du groupe Exemple-AVP ni (probablement) au moment où l'instance exemple de cette AVP est interprétée – sauf par les mises en œuvre Diameter qui prennent en charge le même ensemble d'AVP. L'exemple de codage illustre comment est utilisé le bourrage et comment les champs de longueur sont calculés. Aussi, on notera que les AVP peuvent être présentes dans la valeur d'AVP Grouped que le receveur ne peut pas interpréter (ici, les AVP Recover-Policy et Futuristic-Acct-Record). La longueur de Exemple-AVP est la somme de toutes les longueurs des AVP membres, incluant leur bourrage, plus la taille de l'en-tête Exemple-AVP.

Cette AVP serait codée comme suit :

```
      0       1       2       3       4       5       6       7
   +-------+-------+-------+-------+-------+-------+-------+-------+
0  |  En-tête AVP Exemple (Code d'AVP = 999999), Longueur = 496    |
   +-------+-------+-------+-------+-------+-------+-------+-------+
8  |  En-tête AVP Origin-Host (Code d'AVP = 264), Longueur = 19    |
   +-------+-------+-------+-------+-------+-------+-------+-------+
16 |  'e'  |  'x'  |  'e'  |  'm'  |  'p'  |  'l'  |  'e'  |  '.'  |
   +-------+-------+-------+-------+-------+-------+-------+-------+
24 |  'c'  |  'o'  |  'm'  |Bourrag|  En-tête AVP Session-Id  r    |
   +-------+-------+-------+-------+-------+-------+-------+-------+
32 |(Code AVP = 263), Longueur  49 |  'g'  |  'r'  |  'u'  |  'm'  |
   +-------+-------+-------+-------+-------+-------+-------+-------+
                              . . .
   +-------+-------+-------+-------+-------+-------+-------+-------+
72 |  'F'  |  '3'  |  'B'  |  '8'  |  '1'  |        Bourrage       |
   +-------+-------+-------+-------+-------+-------+-------+-------+
80 |  En-tête AVP Session-Id (Code d'AVP = 263), Longueur = 50     |
   +-------+-------+-------+-------+-------+-------+-------+-------+
88 |  'g'  |  'r'  |  'u'  |  'm'  |  'p'  |  '.'  |  'e'  |  'x'  |
   +-------+-------+-------+-------+-------+-------+-------+-------+
                              . . .
   +-------+-------+-------+-------+-------+-------+-------+-------+
120|  '5'  |  '8'  |  ';'  |  '0'  |  'A'  |  'F'  |  '3'  |  'B'  |
   +-------+-------+-------+-------+-------+-------+-------+-------+
128|  '8'  |  '2'  |    Bourrage   |  En-tête AVP Recovery-Policy  |
   +-------+-------+-------+-------+-------+-------+-------+-------+
136|  Code = 8341), Longueur = 223 | 0x21  | 0x63  | 0xbc  | 0x1d  |
   +-------+-------+-------+-------+-------+-------+-------+-------+
144|  0x0a | 0xd8  | 0x23  | 0x71  | 0xf6  | 0xbc  | 0x09  | 0x48  |
```



4.5 AVP du protocole Diameter de base

Le tableau qui suit décrit les AVP Diameter définies dans le protocole de base, leurs valeurs de code d'AVP, leur type, et les valeurs de fanion possibles. Pour des contraintes de place, la forme abrégée DiamIdent est utilisée pour représenter DiameterIdentity.

				Règles de fanion d'AVP	
Nom d'attribut	**Code d'AVP**	**§**	**Type de données**	**DOIT**	**NE DOIT PAS**
Acct-Interim-Interval	85	9.8.2	Unsigned32	M	V
Accounting-Realtime-Required	483	9.8.7	Enumerated	M	V
Acct-Multi-Session-Id	50	9.8.5	UTF8String	M	V
Accounting-Record-Number	485	9.8.3	Unsigned32	M	V
Accounting-Record-Type	480	9.8.1	Enumerated	M	V
Acct-Session-Id	44	9.8.4	OctetString	M	V
Accounting-Sub-Session-Id	287	9.8.6	Unsigned64	M	V
Acct-Application-Id	259	6.9	Unsigned32	M	V
Auth-Application-Id	258	6.8	Unsigned32	M	V
Auth-Request-Type	274	8.7	Enumerated	M	V
Autorisation-Lifetime	291	8.9	Unsigned32	M	V
Auth-Grace-Period	276	8.10	Unsigned32	M	V
Auth-Session-State	277	8.11	Enumerated	M	V
Re-Auth-Request-Type	285	8.12	Enumerated	M	V
Class	25	8.20	OctetString	M	V
Destination-Host	293	6.5	DiamIdent	M	V
Destination-Realm	283	6.6	DiamIdent	M	V
Disconnect-Cause	273	5.4.3	Enumerated	M	V
Error-Message	281	7.3	UTF8String		V, M
Error-Reporting-Host	294	7.4	DiamIdent		V, M
Event-Timestamp	55	8.21	Time	M	V
Experimental-Result	297	7.6	Grouped	M	V
Experimental-Result-Code	298	7.7	Unsigned32	M	V
Failed-AVP	279	7.5	Grouped	M	V
Firmware-Revision	267	5.3.4	Unsigned32		V, M
Host-IP-Address	257	5.3.5	Address	M	V
Inband-Security-Id	299	6.10	Unsigned32	M	V
Multi-Round-Time-Out	272	8.19	Unsigned32	M	V
Origin-Host	264	6.3	DiamIdent	M	V
Origin-Realm	296	6.4	DiamIdent	M	V
Origin-State-Id	278	8.16	Unsigned32	M	V
Product-Name	269	5.3.7	UTF8String		V, M
Proxy-Host	280	6.7.3	DiamIdent	M	V
Proxy-Info	284	6.7.2	Grouped	M	V
Proxy-State	33	6.7.4	OctetString	M	V
Redirect-Host	292	6.12	DiamURI	M	V
Redirect-Host- Usage	261	6.13	Enumerated	M	V
Redirect-Max-Cache-Time	262	6.14	Unsigned32	M	V
Result-Code	268	7.1	Unsigned32	M	V
Route-Record	282	6.7.1	DiamIdent	M	V

Session-Id	263	8.8	UTF8String	M	V
Session-Timeout	27	8.13	Unsigned32	M	V
Session-Binding	270	8.17	Unsigned32	M	V
Session-Server-Failoverur	271	8.18	Enumerated	M	V
Supported-Vendor-Id	265	5.3.6	Unsigned32	M	V
Termination-Cause	295	8.15	Enumerated	M	V
User-Name	1	8.14	UTF8String	M	V
Vendor-Id	266	5.3.3	Unsigned32	M	V
Vendor-Specific-Application-Id	260	6.11	Grouped	M	V

(Note du traducteur : le V dans la dernière colonne signifie que le fanion V ne doit pas être établi dans cette AVP, voir au paragraphe 4.1)

5. Homologues Diameter

Cette section décrit comment les nœuds Diameter établissent les connexions et communiquent avec leurs homologues.

5.1 Connexions d'homologues

Les connexions entre les homologues Diameter sont établies en utilisant leur DiameterIdentity valide. Un nœud Diameter qui initie une connexion à un homologue DOIT connaître la DiameterIdentity de l'homologue. Les méthodes pour découvrir un homologue Diameter se trouvent au paragraphe 5.2.

Bien qu'un nœud Diameter puisse avoir de nombreux homologues possibles avec lesquels il est capable de communiquer, il peut n'être pas très rentable d'avoir une connexion établie avec chacun d'entre eux. Au minimum, un nœud Diameter DEVRAIT avoir une connexion établie avec deux homologues par domaine, appelés les homologues primaire et secondaire. Bien sûr, un nœud PEUT avoir des connexions supplémentaires, si il l'estime nécessaire. Normalement, tous les messages pour un domaine sont envoyés à l'homologue primaire mais, dans le cas où les procédures de reprise sur défaillance sont invoquées, toutes les demandes en cours sont envoyées à l'homologue secondaire. Cependant, les mises en œuvre sont libres d'équilibrer la charge des demandes entre un ensemble d'homologues.

Noter qu'un certain homologue PEUT agir comme primaire pour un domaine tout en agissant comme secondaire pour un autre.

Lorsque un homologue est réputé suspect, ce qui peut se produire pour diverses raisons, incluant de ne pas recevoir un DWA dans un certain délai, aucune nouvelle demande ne devrait être transmise à l'homologue, mais les procédures de reprise sur défaillance sont invoquées. Lorsque un homologue actif est déplacé dans ce mode, des connexions supplémentaires DEVRAIENT être établies pour s'assurer qu'il existe le nombre nécessaire de connexions actives.

Il y a deux façons de retirer un homologue de la liste des homologues suspects :
1. l'homologue n'est plus joignable, causant la fermeture de la connexion de transport. L'homologue est passé à l'état Fermé ;
2. trois messages de chien de garde sont échangés avec le délai d'aller-retour accepté, et la connexion vers l'homologue est considérée comme stabilisée.

Lorsque l'homologue à supprimer est le primaire ou le secondaire, un homologue de remplacement DEVRAIT être substitué à celui qui est supprimé et assumer le rôle de primaire ou secondaire.

5.2 Découverte d'homologue Diameter

Permettre la découverte dynamique des agents Diameter rend possible un déploiement plus simple et plus robuste des services Diameter. Afin de promouvoir une mise en œuvre interopérable de la découverte de l'homologue Diameter, les mécanismes suivants (configuration manuelle et DNS) sont décrits. Ils se fondent sur les standard existants de l'IETF. Les deux mécanismes DOIVENT être pris en charge par toutes les mises en œuvre Diameter; l'un et l'autre PEUVENT être utilisés.

Il y a deux cas où la découverte de l'homologue Diameter peut être effectuée. La première est quand un client Diameter a besoin de découvrir un agent Diameter de premier bond. Le second cas est quand un agent Diameter a besoin de découvrit un autre agent pour la suite du traitement d'une opération Diameter. Dans les deux cas, l'ordre de recherche suivant est recommandé :

1. La mise en œuvre Diameter consulte sa liste de localisations d'agents Diameter configurée statiquement (manuellement). Cela sera utilisé si ils existent et répondent.

2. La mise en œuvre Diameter effectue une interrogation NAPTR pour un serveur dans un domaine particulier. La mise en œuvre Diameter doit savoir à l'avance dans quel domaine chercher un agent Diameter. Cela peut être déduit, par exemple, du 'domaine' dans un NAI sur lequel la mise en œuvre Diameter a eu besoin d'effectuer une opération Diameter. L'usage de NAPTR dans Diameter suit l'application DDDS S-NAPTR [RFC3958] dans laquelle le champ SERVICE inclut des étiquettes pour l'application désirée et le protocole d'application pris en charge. L'étiquette de service d'application pour une application Diameter est 'aaa' et les étiquettes de protocole d'application pris en charge sont 'diameter.tcp', 'diameter.sctp', 'diameter.dtls', ou 'diameter.tls.tcp' [RFC6408]. Le client peut suivre le processus de résolution défini par l'application DDDS S-NAPTR [RFC3958] pour trouver l'enregistrement SRV, A, ou AAAA correspondant d'un homologue convenable. Les suffixes de domaine dans le champ de remplacement NAPTR DEVRAIENT correspondre au domaine de l'interrogation d'origine. On trouvera un exemple à l'Appendice B.

3. Si on ne trouve aucun enregistrement NAPTR, le demandeur interroge directement un des enregistrements SRV suivants : pour Diameter sur TCP, "_diameter._tcp.realm" ; pour Diameter sur TLS, "_diameters._tcp.realm" ; pour Diameter sur SCTP, "_diameter._sctp.realm" ; pour Diameter sur DTLS, "_diameters._sctp.realm". Si des enregistrements SRV sont trouvés, le demandeur peut alors effectuer une interrogation d'enregistrement d'adresse (RR A et/ou RR AAAA) pour le nom d'hôte cible spécifié dans les enregistrements SRV suivant la règle donnée dans la [RFC2782]. Si aucun enregistrement SRV n'est trouvé, le demandeur abandonne.

Si le serveur utilise un certificat de site, le nom de domaine dans l'interrogation NAPTR et le nom de domaine dans le champ de remplacement DOIVENT tous deux être valides sur la base du certificat de site traité par le serveur dans l'échange TLS/TCP et DTLS/SCTP ou du protocole d'échange de clés (IKE, *Internet Key Exchange*). De même, le nom de domaine dans l'interrogation SRV et le nom de domaine dans la cible dans l'enregistrement SRV DOIVENT tous deux être valides sur la base du même certificat de site. Autrement, un attaquant pourrait modifier les enregistrements du DNS pour qu'ils contiennent des valeurs de remplacement dans un domaine différent, et le client ne pourrait pas valider si c'est le comportement désiré ou le résultat d'une attaque.

Aussi, l'homologue Diameter DOIT s'assurer que les homologues découverts sont autorisés à agir dans ce rôle. L'authentification via IKE ou TLS/TCP et DTLS/SCTP, ou la validation des RR du DNS via DNSSEC n'est pas suffisante pour cela. Par exemple, un serveur de la Toile peut avoir obtenu un certificat TLS/TCP et DTLS/SCTP valide, et des RR sécurisés peuvent être inclus dans le DNS, mais cela n'implique pas qu'il soit autorisé à agir comme serveur Diameter.

L'autorisation peut se réaliser, par exemple, par la configuration d'une autorité de certification (CA, *Certification Authority*) de serveur Diameter. La CA de serveur produit un certificat au serveur Diameter, qui comporte un identifiant d'objet (OID, *Object Identifier*) pour indiquer que le sujet est un serveur Diameter dans l'extension Extended Key Usage [RFC5280]. Ce certificat est alors utilisé durant la négociation de sécurité TLS/TCP, DTLS/SCTP, ou IKE. On notera cependant qu'au moment de la rédaction du présent mémoire, il n'existe aucune autorité de certification de serveur Diameter.

Un homologue découvert de façon dynamique cause la création d'une entrée dans le tableau d'homologue (voir au paragraphe 2.6). Noter que les entrées créées via le DNS DOIVENT arriver à expiration (ou être rafraîchies) dans la durée de vie (TTL, *Time to Live*) du DNS. Si un homologue est découvert en dehors du domaine local, une entrée de tableau d'acheminement (voir au paragraphe 2.7) est créée pour le domaine de l'homologue. L'expiration de l'entrée de tableau d'acheminement DOIT correspondre à la valeur d'expiration de l'homologue.

5.3 Échange de capacités

Lorsque deux homologues Diameter établissent une connexion de transport, ils DOIVENT échanger des messages d'échange de capacités, comme spécifié dans l'automate à états d'homologue (voir au paragraphe 5.6). Ce message permet la découverte de l'identité d'un homologue et de ses capacités (numéro de version du protocole, identifiants des applications Diameter prises en charge, mécanismes de sécurité, etc.).

Le receveur produit seulement des commandes à ses homologues qui ont annoncé la prise en charge de l'application Diameter que définit la commande. Un nœud Diameter DOIT mettre en antémémoire les identifiants d'application pris en charge afin de s'assurer que des commandes et/ou des AVP non reconnues ne sont pas envoyées inutilement à un homologue.

Un receveur d'un message Demande d'échange de capacités (CER, *Capabilities-Exchange-Request*) qui n'a aucune application en commun avec l'envoyeur DOIT retourner une Capabilities-Exchange-Answer (CEA) avec l'AVP Result-Code réglée à DIAMETER_NO_COMMON_APPLICATION et DEVRAIT déconnecter la connexion de couche transport. Noter que recevoir une CER ou CEA d'un homologue qui s'annonce comme un relais (voir au paragraphe 2.4) DOIT être

interprété comme avoir des applications communes avec l'homologue.

Le receveur de la Capabilities-Exchange-Request (CER) DOIT déterminer les applications communes en calculant l'intersection de son propre ensemble d'identifiants d'application pris en charge avec toutes les AVP Application-Id (Auth-Application-Id, Acct-Application-Id, et Vendor-Specific-Application-Id) présentes dans la CER. La valeur de l'AVP Vendor-Id dans le Vendor-Specific-Application-Id NE DOIT PAS être utilisée durant le calcul. L'envoyeur de la Capabilities-Exchange-Answer (CEA) DEVRAIT inclure toutes ses applications prises en charge comme une indication au receveur concernant toutes ses capacités d'application.

Les mises en œuvre Diameter DEVRAIT d'abord tenter d'établir une connexion TLS/TCP et DTLS/SCTP avant l'échange de CER/CEA. Cela protège les informations de capacité des deux homologues. Pour prendre en charge les mises en œuvre Diameter plus anciennes qui ne se conforment pas pleinement au présent document, la sécurité du transport PEUT quand même être négociée via une AVP Inband-Security. Dans ce cas, le receveur d'un message Capabilities-Exchange-Request (CER) qui n'a aucun mécanisme de sécurité en commun avec l'envoyeur DOIT retourner une Capabilities-Exchange-Answer (CEA) avec l'AVP Result-Code réglée à DIAMETER_NO_COMMON_SECURITY et DEVRAIT déconnecter la connexion de couche transport.

Les CER reçues d'homologues inconnus PEUVENT être éliminées en silence, ou une CEA PEUT être produite avec l'AVP Result-Code réglée à DIAMETER_UNKNOWN_PEER. Dans les deux cas, la connexion de transport est close. Si la politique locale permet de recevoir des CER d'hôtes inconnus, une CEA réussie PEUT être retournée. Si il est répondu à une CER d'un homologue inconnu par une CEA de succès, la durée de vie de l'entrée de l'homologue est égale à la durée de vie de la connexion de transport. En cas de défaillance du transport, toutes les transactions en cours destinées à l'homologue inconnu peuvent être éliminées.

Les messages CER et CEA NE DOIVENT PAS être mandatés, redirigés, ou relayés.

Comme les messages CER/CEA ne peuvent pas être mandatés, il est quand même possible qu'un agent amont reçoive un message pour lequel il n'a pas d'homologue disponible pour traiter l'application qui correspond au code de commande. Dans de telles instances, le bit 'E' est établi dans le message de réponse (Section 7) avec l'AVP Result-Code réglée à DIAMETER_UNABLE_TO_DELIVER pour informer l'agent aval de la nécessité d'une action (par exemple, réacheminer la demande à un homologue de remplacement).

À l'exception du message Capabilities-Exchange-Request, un message de type Request qui comporte les AVP Auth-Application-Id ou Acct-Application-Id, ou un message avec un code de commande spécifique de l'application PEUT seulement être transmis à un hôte qui a explicitement annoncé la prise en charge de l'application (ou a annoncé l'identifiant d'application relais).

5.3.1 Capabilities-Exchange-Request

Le message Capabilities-Exchange-Request (CER), indiqué par le code de commande réglé à 257 et le bit 'R' des fanions de commandes établi, est envoyé pour échanger les capacités locales. Si on détecte une défaillance du transport, ce message NE DOIT PAS être envoyé à un homologue de remplacement.

Lorsque Diameter fonctionne sur SCTP [RFC4960] ou DTLS/SCTP [RFC6083], qui permettent aux connexions de s'étendre sur plusieurs interfaces et plusieurs adresses IP, le message Capabilities-Exchange-Request DOIT contenir une AVP Host-IP-Address pour chaque adresse IP potentielle qui PEUT être utilisé localement lors de la transmission de messages Diameter.

Format de message :

```
<CER> ::= < En-tête Diameter : 257, REQ >
                { Origin-Host }
                { Origin-Realm }
               1* { Host-IP-Address }
                { Vendor-Id }
                { Product-Name }
                [ Origin-State-Id ]
               * [ Supported-Vendor-Id ]
               * [ Auth-Application-Id ]
               * [ Inband-Security-Id ]
               * [ Acct-Application-Id ]
               * [ Vendor-Specific-Application-Id ]
                 [ Firmware-Revision ]
```

```
            * [ AVP ]
```

5.3.2 Capabilities-Exchange-Answer

Le message Capabilities-Exchange-Answer (CEA), indiqué par le code de commande 257 et le bit 'R' des fanions de commande à zéro, est envoyé en réponse à un message CER.

Lorsque Diameter fonctionne sur SCTP [RFC4960] ou DTLS/SCTP [RFC6083], ce qui permet aux connexions de s'étendre sur plusieurs interfaces, et donc plusieurs adresses IP, le message Capabilities-Exchange-Answer DOIT contenir une AVP Host-IP-Address pour chaque adresse IP potentielle qui PEUT être utilisée localement lors de la transmission de messages Diameter.

Format de message :

```
<CEA> ::= < En-tête Diameter : 257 >
              { Result-Code }
              { Origin-Host }
              { Origin-Realm }
            1* { Host-IP-Address }
              { Vendor-Id }
              { Product-Name }
              [ Origin-State-Id ]
              [ Error-Message ]
              [ Failed-AVP ]
            * [ Supported-Vendor-Id ]
            * [ Auth-Application-Id ]
            * [ Inband-Security-Id ]
            * [ Acct-Application-Id ]
            * [ Vendor-Specific-Application-Id ]
              [ Firmware-Revision ]
            * [ AVP ]
```

5.3.3 AVP Vendor-Id

L'AVP Vendor-Id (code d'AVP : 266) est du type Unsigned32 et contient la valeur allouée par l'IANA de "SMI Network Management Private Enterprise Codes" [ENTERPRISE] au fabricant de logiciel Diameter. Il est envisagé que la combinaison des AVP Vendor-Id, Product-Name (paragraphe 5.3.7), et Firmware-Revision (paragraphe 5.3.4) puisse fournir d'utiles informations de débogage.

Une valeur de Vendor-Id de zéro dans le message CER ou CEA est réservée et indique que ce champ est ignoré.

5.3.4 AVP Firmware-Revision

L'AVP Firmware-Revision (code d'AVP : 267) est du type Unsigned32 et est utilisée pour informer un homologue Diameter de la révision du logiciel de l'appareil producteur.

Pour les appareils qui n'ont pas de révision du logiciel (ordinateurs d'utilité générale qui fonctionnent avec des modules de logiciel Diameter, par exemple) la révision du module de logiciel Diameter peut être rapportée à la place.

5.3.5 AVP Host-IP-Address

L'AVP Host-IP-Address (code d'AVP : 257) est du type Address et est utilisée pour informer un homologue Diameter de l'adresse IP de l'envoyeur. Toutes les adresses de source qu'un nœud Diameter s'attend à utiliser avec SCTP [RFC4960] ou DTLS/SCTP [RFC6083] DOIVENT être annoncées dans les messages CER et CEA en incluant une AVP Host-IP-Address pour chaque adresse.

5.3.6 AVP Supported-Vendor-Id

L'AVP Supported-Vendor-Id AVP (code d'AVP : 265) est du type Unsigned32 et contient la valeur allouée par l'IANA de "SMI Network Management Private Enterprise Codes" [ENTERPRISE] à un fabricant autre que celui de l'appareil mais incluant le fabricant de l'application. Elle est utilisée dans les messages CER et CEA afin d'informer l'homologue que l'envoyeur prend en charge (un sous ensemble) des AVP Vendor-Specific définies par le fabricant identifié dans cette AVP.

La valeur de cette AVP NE DOIT PAS être réglée à zéro. Plusieurs instances de cette AVP contenant la même valeur NE DEVRAIENT PAS être envoyées.

5.3.7 AVP Product-Name

L'AVP Product-Name (code d'AVP : 269) est du type UTF8String et contient le nom alloué par le fabricant pour le produit. L'AVP Product-Name DEVRAIT rester constante à travers les révisions du logiciel pour le même produit.

5.4 Déconnexion des connexions avec les homologues

Lorsque un nœud Diameter déconnecte une de ses connexions de transport, son homologue ne peut pas connaître la raison de la déconnexion et va très vraisemblablement supposer qu'un problème de connexité s'est produit ou que l'homologue s'est réamorcé. Dans ce cas, l'homologue peut périodiquement tenter de se reconnecter, comme indiqué au paragraphe 2.1. Si la déconnexion résulte d'une panne de ressource interne ou simplement de ce que le nœud en question n'a pas l'intention de transmettre de messages Diameter à l'homologue dans un futur proche, une demande de connexion périodique ne sera pas la bienvenue. L'AVP Disconnexion-Reason contient la raison pour laquelle le nœud Diameter a produit le message Disconnect-Peer-Request.

Le message Disconnect-Peer-Request est utilisé par un nœud Diameter pour informer son homologue de son intention de déconnecter la couche transport et que l'homologue ne devrait pas se reconnecter sauf si il a une raison valide de le faire (par exemple, un message à transmettre). À réception du message, le message Disconnect-Peer-Answer est retourné, qui DEVRAIT contenir une erreur si des messages ont été récemment transmis, et sont probablement en cours, ce qui autrement causerait une condition de conflit.

Le receveur du message Disconnect-Peer-Answer initie la déconnexion du transport. L'envoyeur du message Disconnect-Peer-Answer devrait être capable de détecter la clôture du transport et de supprimer la connexion.

5.4.1 Disconnect-Peer-Request

Le message Disconnect-Peer-Request (DPR), indiqué par le code de commande réglé à 282 et le bit 'R' des fanions de commande établi à 1, est envoyé à un homologue pour l'informer de son intention de clore la connexion de transport. Si on détecte une défaillance du transport, ce message NE DOIT PAS être envoyé à un homologue de remplacement.

Format de message :

```
<DPR>  ::= < En-tête Diameter : 282, REQ >
             { Origin-Host }
             { Origin-Realm }
             { Disconnect-Cause }
           * [ AVP ]
```

5.4.2 Disconnect-Peer-Answer

Le message Disconnect-Peer-Answer (DPA), indiqué par le code de commande réglé à 282 et le bit 'R' des fanions de commande réglé à zéro, est envoyé en réponse au message Disconnect-Peer-Request. À réception de ce message, la connexion de transport est clôturée.

Format de message :

```
<DPA>  ::= < En-tête Diameter : 282 >
             { Result-Code }
             { Origin-Host }
             { Origin-Realm }
             [ Error-Message ]
             [ Failed-AVP ]
           * [ AVP ]
```

5.4.3 AVP Disconnect-Cause

L'AVP Disconnect-Cause AVP (code d'AVP : 273) est du type Enumerated. Un nœud Diameter DOIT inclure cette AVP dans le message Disconnect-Peer-Request pour informer l'homologue de la raison de son intention de clore la connexion de transport. Les valeurs suivantes sont acceptées :

REBOOTING : 0. Un réamorçage programmé est imminent. Un receveur d'un DPR avec ce code de résultat PEUT tenter la reconnexion.

BUSY : 1. Les ressources internes de l'homologue sont restreintes, et il a déterminé que la connexion de transport doit être fermée. Le receveur d'un DPR avec le code de résultat ci-dessus NE DEVRAIT PAS tenter la reconnexion.

DO_NOT_WANT_TO_TALK_TO_YOU : 2. L'homologue a déterminé qu'il ne voit pas le besoin d'une connexion de transport, car il n'attend aucun échange de messages dans un avenir proche. Un receveur d'un DPR avec le code de résultat ci-dessus NE DEVRAIT PAS tenter de reconnexion.

5.5 Détection d'une défaillance du transport

Étant donnée la nature du protocole Diameter, il est recommandé que les défaillances de transport soient détectées aussitôt que possible. Détecter de telles défaillances minimise l'occurrence de messages envoyés à des agents indisponibles, résultant en délais inutiles, et donnera de meilleures performances de reprise sur défaillance. Les messages Device-Watchdog-Request et Device-Watchdog-Answer, définis dans cette section, sont utilisés pour détecter activement les défaillances de transport.

5.5.1 Device-Watchdog-Request

Le message Device-Watchdog-Request (DWR), indiqué par le code de commande 280 et le bit 'R' des fanions de commande établi à 1, est envoyé à un homologue lorsque aucun trafic n'a été échangé entre deux homologues (voir au paragraphe 5.5.3). À la détection d'une défaillance de transport, ce message NE DOIT PAS être envoyé à un homologue de remplacement.

Format de message :

```
<DWR>  ::= < En-tête Diameter : 280, REQ >
             { Origin-Host }
             { Origin-Realm }
             [ Origin-State-Id ]
            * [ AVP ]
```

5.5.2 Device-Watchdog-Answer

Le message Device-Watchdog-Answer (DWA), indiqué par le code de commande réglé à 280 et le bit 'R' des fanions de commande à zéro, est envoyé comme réponse au message Device-Watchdog-Request.

Format de message :

```
<DWA>  ::= < En-tête Diameter : 280 >
             { Result-Code }
             { Origin-Host }
             { Origin-Realm }
             [ Error-Message ]
             [ Failed-AVP ]
             [ Origin-State-Id ]
            * [ AVP ]
```

5.5.3 Algorithme de défaillance de transport

L'algorithme de défaillance de transport est défini dans la [RFC3539]. Toutes les mises en œuvre Diameter DOIVENT prendre en charge l'algorithme défini dans cette spécification afin de se conformer au protocole de base Diameter.

5.5.4 Reprise sur défaillance et procédures de reprise sur défaillance

Lorsque une défaillance de transport est détectée avec un homologue, il est nécessaire que tous les messages de demande en cours soient transmis à un agent de remplacement, si possible. C'est ce qu'on appelle généralement la "reprise sur défaillance".

Pour qu'un nœud Diameter effectue les procédures de reprise sur défaillance, il est nécessaire que le nœud conserve une file d'attente des messages en cours pour un certain homologue. Lorsque un message de réponse est reçu, la demande correspondante est retirée de la file d'attente. Le champ Identifiant bond par bond est utilisé pour confronter la réponse à la

demande de la file d'attente.

Lorsque une défaillance de transport est détectée, si possible, tous les messages de la file d'attente sont envoyés à un agent de remplacement avec le fanion T établi. À l'amorçage d'un client ou agent Diameter, le fanion T est aussi établi sur tous les enregistrements restants dans une mémorisation non volatile qui attendent encore d'être transmis. Un exemple de cas où il n'est pas possible de transmettre le message à un serveur de remplacement est quand le message a une destination fixée, et où l'homologue indisponible est la destination finale du message (voir l'AVP Destination-Host). Une telle erreur exige que l'agent retourne un message de réponse avec le bit 'E' établi et l'AVP Result-Code réglée à DIAMETER_UNABLE_TO_DELIVER *(Diameter incapable de livrer)*.

Il est important de noter que plusieurs demandes ou réponses identiques PEUVENT être reçues par suite d'une reprise sur défaillance. Le champ Identifiant de bout en bout dans l'en-tête Diameter avec l'AVP Origin-Host DOIT être utilisé pour identifier les messages dupliqués.

Comme décrit au paragraphe 2.1, une demande de connexion devrait être périodiquement tentée avec l'homologue défaillant afin de rétablir la connexion de transport. Une fois qu'une connexion a pu être établie, les messages peuvent à nouveau être transmis à l'homologue. C'est ce qu'on appelle généralement une "reprise sur défaillance".

5.6 Automate à état d'homologue

Cette section traite d'un automate à états finis qui DOIT être respecté par toutes les mises en œuvre Diameter. Chaque nœud Diameter DOIT suivre l'automate à états décrit ci-dessous lors des communications avec chaque homologue. Plusieurs actions sont séparées par des virgules, et peuvent continuer sur les lignes suivantes, selon les exigences d'espace. De même, un état et l'état suivant peuvent aussi s'étendre sur plusieurs lignes, selon que l'espace l'exige.

Cet automate à états est étroitement couplé avec l'automate à états décrit dans la [RFC3539], qui est utilisé pour ouvrir, clore, reprendre sur défaillance, sonder, et rouvrir les connexions de transport. En particulier, on note que la [RFC3539] exige l'utilisation de messages de chien de garde pour sonder les connexions. Pour Diameter, les messages DWR et DWA sont à utiliser.

Le préfixe I- est utilisé pour représenter la connexion initiatrice (qui connecte) tandis que le préfixe R- est utilisé pour représenter la connexion qui répond (qui écoute). L'absence d'un préfixe indique que l'événement ou action est le même sans considération de la connexion sur laquelle l'événement se produit.

Les états stables dans lesquels peut être un automate à états sont Closed *(fermé)*, I-Open, et R-Open ; tous les autres états sont des états intermédiaires. Noter que I-Open et R-Open sont équivalents sauf pour le fait que la connexion de transport initiatrice ou répondante est utilisée pour la communication.

Un message CER est toujours envoyé sur la connexion initiatrice immédiatement après l'achèvement réussi de la demande de connexion. Dans le cas d'une élection, une des deux connexions sera fermée. La connexion répondante va survivre si l'hôte d'origine de l'entité Diameter locale est plus forte que celui de l'homologue ; la connexion initiatrice survivra si le Origin-Host de l'homologue est plus élevé. Tous les messages suivants sont envoyés sur la connexion survivante. Noter que le résultat d'une élection sur un homologue est garanti d'être l'inverse du résultat sur l'autre.

Pour l'utilisation de TLS/TCP et DTLS/SCTP, une prise de contact TLS/TCP et DTLS/SCTP DEVRAIT commencer lorsque les deux extrémités sont dans l'état fermé avant tout échange de message Diameter. La connexion TLS/TCP et DTLS/SCTP DEVRAIT être établie avant d'envoyer aucun message CER ou CEA pour sécuriser et protéger les informations de capacité des deux homologues. La connexion TLS/TCP et DTLS/SCTP DEVRAIT être déconnectée lorsque l'automate à états passe à l'état fermé. Lorsque il y a connexion à des répondants qui ne se conforment pas au présent document (c'est-à-dire, à des mises en œuvre Diameter plus anciennes qui ne sont pas prêtes à recevoir des connexions TLS/TCP et DTLS/ SCTP dans l'état fermé) la tentative de connexion initiale TLS/TCP et DTLS/ SCTP va échouer. L'initiateur PEUT alors tenter de se connecter via TCP ou SCTP et initier la prise de contact TLS/TCP et DTLS/SCTP lorsque les deux extrémités sont dans l'état ouvert. Si la prise de contact réussit, tous les messages ultérieurs seront via TLS/TCP et DTLS/ SCTP. Si la prise de contact échoue, les deux extrémités passent à l'état fermé.

L'automate à états ne contraint que le comportement d'une mise en œuvre Diameter telle que vue par les homologues Diameter à travers les événements sur le réseau.

Toute mise en œuvre qui produit des résultats équivalents est considérée comme conforme.

État	Événement	Action	Prochain état
Closed	Start R-Conn-CER	I-Snd-Conn-Req R-Accept, Process-CER, R-Snd-CEA	Wait-Conn-Ack R-Open
Wait-Conn-Ack	I-Rcv-Conn-Ack I-Rcv-Conn-Nack R-Conn-CER Timeout	I-Snd-CER Cleanup R-Accept, Process-CER Error	Wait-I-CEA Closed Wait-Conn-Ack/Elect Closed
Wait-I-CEA	I-Rcv-CEA R-Conn-CER I-Peer-Disc I-Rcv-Non-CEA Timeout	Process-CEA R-Accept, Process-CER, Elect I-Disc Error Error	I-Open Wait-Returns Closed Closed Closed
Wait-Conn-Ack/Elect	I-Rcv-Conn-Ack I-Rcv-Conn-Nack R-Peer-Disc R-Conn-CER Timeout	I-Snd-CER, Elect R-Snd-CEA R-Disc R-Reject Error	Wait-Returns R-Open Wait-Conn-Ack Wait-Conn-Ack/Elect Closed
Wait-Returns	Win-Election I-Peer-Disc I-Rcv-CEA R-Peer-Disc R-Conn-CER Timeout	I-Disc, R-Snd-CEA I-Disc, R-Snd-CEA R-Disc R-Disc R-Reject Error	R-Open R-Open I-Open Wait-I-CEA Wait-Returns Closed
R-Open	Send-Message R-Rcv-Message R-Rcv-DWR R-Rcv-DWA R-Conn-CER Stop R-Rcv-DPR R-Peer-Disc	R-Snd-Message Process Process-DWR, R-Snd-DWA Process-DWA R-Reject R-Snd-DPR R-Snd-DPA R-Disc	R-Open R-Open R-Open R-Open R-Open Closing Closing Closed
I-Open	Send-Message I-Rcv-Message I-Rcv-DWR I-Rcv-DWA R-Conn-CER Stop I-Rcv-DPR I-Peer-Disc	I-Snd-Message Process Process-DWR, I-Snd-DWA Process-DWA R-Reject I-Snd-DPR I-Snd-DPA I-Disc	I-Open I-Open I-Open I-Open I-Open Closing Closing Closed
Closing	I-Rcv-DPA R-Rcv-DPA Timeout I-Peer-Disc R-Peer-Disc	I-Disc R-Disc Error I-Disc R-Disc	Closed Closed Closed Closed Closed

5.6.1 Connexions entrantes

Lorsque une demande de connexion est reçue d'un homologue Diameter, il n'est pas, en général, possible de connaître l'identité de cet homologue jusqu'à ce qu'un CER soit reçu de lui. C'est parce que l'hôte et l'accès déterminent l'identité d'un homologue Diameter ; l'accès de source d'une connexion entrante est arbitraire. À réception d'un CER, l'identité de l'homologue qui se connecte peut être déterminée de façon univoque par le Origin-Host.

Pour cette raison, un homologue Diameter doit employer une logique différente de l'automate à états pour recevoir les demandes de connexion, les accepter, et attendre le CER. Une fois le CER arrivé sur une nouvelle connexion, le Origin-Host qui identifie l'homologue est utilisé pour localiser l'automate à états associé à cet homologue, et la nouvelle connexion et le CER sont passés à l'automate à états comme un événement R-Conn-CER.

La logique qui traite les connexions entrantes DEVRAIT clore et éliminer la connexion si un message autre qu'un CER arrive ou si se produit la fin de temporisation déterminée par la mise en œuvre avant la réception du CER.

Parce que le traitement des connexions entrantes jusque et y compris la réception d'un CER exige une logique, différente de celle de tout automate à états individuel associé à un homologue particulier, il est décrit séparément dans cette section plutôt que dans l'automate à états ci-dessus.

5.6.2 Événements

Les transitions et actions dans l'automate sont causées par des événements. Dans cette section, on ignorera les préfixes I- et R-, car l'événement réel serait identique, mais il va se produire sur une des deux connexions possibles.

Start : l'application Diameter a signalé qu'une connexion devrait être initiée avec l'homologue.
R-Conn-CER : un accusé de réception est reçu déclarant que la connexion de transport a été établie, et le CER associé est arrivé.
Rcv-Conn-Ack : un accusé de réception positif est reçu qui confirme l'établissement de la connexion de transport.
Rcv-Conn-Nack : un accusé de réception négatif a été reçu déclarant que la connexion de transport n'a pas été établie.
Timeout : un temporisateur défini par l'application est arrivé à expiration pendant l'attente d'un certain événement.Rcv-CER : un message CER a été reçu de l'homologue.
Rcv-CEA : un message CEA a été reçu de l'homologue.
Rcv-Non-CEA : un message, autre qu'un CEA, a été reçu de l'homologue.
Peer-Disc : une indication de déconnexion a été reçue de l'homologue.
Rcv-DPR : un message DPR a été reçu de l'homologue.
Rcv-DPA : un message DPA a été reçu de l'homologue.
Win-Election : une élection a eu lieu, et le nœud local l'a gagnée.
Send-Message : un message va être envoyé.
Rcv-Message : un message autre que CER, CEA, DPR, DPA, DWR, ou DWA a été reçu.
Stop : l'application Diameter a signalé qu'une connexion devrait être terminée (par exemple, sur fermeture du système).

5.6.3 Actions

Les actions dans l'automate sont causées par des événements et indiquent normalement la transmission de paquets et/ou une action à entreprendre sur la connexion. Dans ce paragraphe, on ignorera les préfixes I- et R-, car l'action réelle sera identique, mais elle ne va se produire que sur une des deux connexions possibles.

Snd-Conn-Req : une connexion de transport est initiée avec l'homologue.
Accept : la connexion entrante associée au R-Conn-CER est acceptée comme connexion répondante.
Reject : la connexion entrante associée au R-Conn-CER est déconnectée.
Process-CER : le CER associé au R-Conn-CER est traité.
Snd-CER : un message CER est envoyé à l'homologue.
Snd-CEA : un message CEA est envoyé à l'homologue.
Cleanup : si nécessaire, la connexion est close, et toutes les ressources locales sont libérées.
Error : la connexion de couche transport est déconnectée, soit poliment, soit interrompue, en réponse à une condition d'erreur. Les ressources locales sont libérées.
Process-CEA : un CEA reçu est traité.
Snd-DPR : un message DPR est envoyé à l'homologue.
Snd-DPA : un message DPA est envoyé à l'homologue.
Disc : La connexion de couche transport est déconnectée, et les ressources locales sont libérées.
Elect : une élection a lieu (voir au paragraphe 5.6.4 plus d'informations).
Snd-Message : un message est envoyé.
Snd-DWR : un message DWR est envoyé.
Snd-DWA : un message DWA est envoyé.
Process-DWR : le message DWR est servi.
Process-DWA : le message DWA est servi.
Process : un message est servi.

5.6.4 Processus d'élection

L'élection est effectuée sur le répondant. Le répondant compare le Origin-Host reçu dans le CER avec son propre Origin-Host comme deux flux d'octets. Si le Origin-Host local surpasse lexicographiquement le Origin-Host reçu, un événement Win-Election est produit localement. Les identités Diameter sont en forme ASCII ; donc, la comparaison lexicale est cohérente avec l'insensibilité à la casse du DNS, où les octets qui rentrent dans la gamme ASCII de 'a' à 'z' DOIVENT se comparer également à leur contrepartie en majuscule entre 'A' et 'Z'. Voir à l'Appendice D les interactions entre le protocole

Diameter et les noms de domaine internationalisés (IDN, *Internationalized Domain Name*).

Le vainqueur de l'élection DOIT clore la connexion qu'il a initiée. Historiquement, conserver le côté répondant d'une connexion était plus efficace que de conserver le côté initiateur. Cependant, les pratiques courantes rendent cette distinction sans objet.

6. Traitement des messages Diameter

Cette section décrit comment les demandes et réponses Diameter sont créées et traitées.

6.1 Généralités sur l'acheminement des demandes Diameter

Une demande est envoyée vers sa destination finale en utilisant une des trois combinaisons suivantes des AVP Destination-Realm et Destination-Host :

- Une demande qui n'est pas capable d'être mandatée (comme un CER) NE DOIT PAS contenir les AVP Destination-Realm ou Destination-Host.
- Une demande qui a besoin d'être envoyée à un serveur de rattachement qui dessert un domaine spécifique, mais pas à un serveur spécifique (comme la première demande d'une série d'allers-retours) DOIT contenir une AVP Destination-Realm mais NE DOIT PAS contenir une AVP Destination-Host. Pour les clients Diameter, la valeur de l'AVP Destination-Realm PEUT être extraite de l'AVP User-Name, ou d'autres méthodes.
- Autrement, une demande qui a besoin d'être envoyée à un serveur de rattachement spécifique parmi ceux qui desservent un certain domaine DOIT contenir les deux AVP Destination-Realm et Destination-Host.

L'AVP Destination-Host est utilisée comme décrit ci-dessus lorsque la destination de la demande est fixée, ce qui inclut :

- Les demandes d'authentification qui s'étendent sur plusieurs allers-retours.
- Un message Diameter qui utilise un mécanisme de sécurité qui se sert d'une clé de session préétablie partagée entre la source et la destination finale du message.
- Les messages initiés par le serveur qui DOIVENT être reçus par un client Diameter spécifique (par exemple, un appareil d'accès) comme le message Abort-Session-Request, qui est utilisé pour demander que la session d'un utilisateur particulier soit terminée.

Noter qu'un agent peut seulement transmettre une demande à un hôte décrit dans l'AVP Destination-Host si l'hôte en question est inclus dans son tableau d'homologues (voir au paragraphe 2.6). Autrement, la demande est acheminée seulement sur la base du domaine de destination (voir au paragraphe 6.1.6).

Lorsque un message est reçu, il est traité dans l'ordre suivant :

- Si le message est destiné à l'hôte local, on suit les procédures du paragraphe 6.1.4.
- Si le message est destiné à un homologue Diameter avec qui l'hôte local est capable de communiquer directement, on suit les procédures du paragraphe 6.1.5. C'est ce qu'on appelle "transmission de demande".
- Les procédures du paragraphe 6.1.6 sont suivies, qu'on appelle "acheminement de demande".
- Si aucune des procédures qui précèdent ne réussit, une réponse est retournée avec le code de résultat réglé à DIAMETER_UNABLE_TO_DELIVER, avec le bit 'E' établi *(à 1)*.

Pour que l'acheminement des messages Diameter fonctionne au sein d'un domaine administratif, tous les nœuds Diameter au sein du domaine DOIVENT être homologues.

La vue d'ensemble de ce paragraphe 6.1 est destinée à fournir des lignes directrices générales aux développeurs Diameter. Les mises en œuvre ont toute liberté pour utiliser des méthodes différentes de celles décrites ici pour autant qu'elles se conforment aux exigences spécifiées aux paragraphes 6.1.1 à 6.1.9. Voir à la Section 7 les détails du traitement d'erreur.

6.1.1 Générer une demande

Lors de la création d'une demande, en plus de toutes les autres procédures décrites dans la définition d'application pour cette demande spécifique, les procédures suivantes DOIVENT être respectées :

- le code de commande est réglé à la valeur appropriée ;
- le bit 'R' est établi ;
- l'identifiant de bout en bout est réglé à une valeur localement unique ;
- les AVP Origin-Host et Origin-Realm DOIVENT être réglées aux valeurs appropriées, utilisées pour identifier la source du message ; et
- les AVP Destination-Host et Destination-Realm DOIVENT être réglées aux valeurs appropriées, comme décrit au paragraphe 6.1.

6.1.2 Envoi d'une demande

Lors de l'envoi d'une demande, générée localement ou par suite d'une opération de transmission ou d'acheminement, les procédures suivantes DEVRAIENT être respectées :

- L'identifiant bond par bond DEVRAIT être réglé à une valeur localement unique.
- Le message DEVRAIT être sauvegardé dans la liste de demandes en cours.

Les autres actions à effectuer sur le message selon le rôle particulier tenu par l'agent sont décrites dans les paragraphes qui suivent.

6.1.3 Réception des demandes

Un agent de relais ou mandataire DOIT vérifier qu'il n'y a pas de transmission en boucle lors de la réception des demandes. Une boucle est détectée si le serveur trouve sa propre identité dans une AVP Route-Record. Lorsque cela se produit, l'agent DOIT répondre par l'AVP Result-Code réglée à DIAMETER_LOOP_DETECTED *(boucle Diameter détectée)*.

6.1.4 Traitement des demandes locales

Une demande est réputée être de consommation locale lorsque une des conditions suivantes se produit :

- L'AVP Destination-Host contient l'identité de l'hôte local ;
- L'AVP Destination-Host n'est pas présente, l'AVP Destination-Realm contient un domaine que le serveur est configuré à traiter en local, et l'application Diameter est prise en charge en local ; ou
- Les AVP Destination-Host et Destination-Realm sont toutes deux absentes.

Lorsque une demande est traitée en local, les règles du paragraphe 6.2 devraient être utilisées pour générer la réponse correspondante.

6.1.5 Transmission de la demande

La transmission de demande est faite en utilisant le tableau d'homologue Diameter. Le tableau d'homologue Diameter contient tous les homologues avec lesquels le nœud local est capable de communiquer directement.
Lorsque une demande est reçue, et que l'hôte codé dans l'AVP Destination-Host est un de ceux qui sont présents dans le tableau d'homologues, le message DEVRAIT être transmis à l'homologue.

6.1.6 Acheminement de demande

L'acheminement des messages de demande Diameter est fait via les domaines et les identifiants d'application. Un message Diameter qui peut être transmis par des agents Diameter (mandataires, agents de redirection, ou agents de relais) DOIT inclure le domaine cible dans l'AVP Destination-Realm. L'acheminement des demandes DEVRAIT s'appuyer sur l'AVP Destination-Realm et l'identifiant d'application présent dans l'en-tête du message de demande pour aider la décision d'acheminement. Le domaine PEUT être restitué à partir de l'AVP User-Name, qui est sous la forme d'un identifiant d'accès réseau (NAI, *Network Access Identifier*). La portion domaine du NAI est insérée dans l'AVP Destination-Realm.
Les agents Diameter PEUVENT avoir une liste des domaines et applications acceptés localement, et ils PEUVENT avoir une liste des domaines et applications acceptés en externe. Lorsque une demande reçue comporte un domaine et/ou application qui n'est pas accepté localement, le message est acheminé à l'homologue configuré dans le tableau d'acheminement (voir au paragraphe 2.7).
Les noms des domaines et les identifiants d'application sont le critère d'acheminement minimum pris en charge ; des informations supplémentaires peuvent être nécessaires pour la prise en charge de la sémantique de redirection.

6.1.7 Évitement prédictif de boucle

Avant de transmettre ou acheminer une demande, les agents Diameter, en plus d'effectuer le traitement décrit au paragraphe 6.1.3, DEVRAIENT vérifier la présence de l'identité de l'homologue du chemin candidat dans toutes les AVP Route-Record. Si l'agent détecte la présence de l'identité d'homologue d'un chemin candidat dans une AVP Route-Record, l'agent DOIT ignorer un tel chemin pour le message de demande Diameter et tenter s'il en existe des chemins de remplacement. Si tous les chemins candidats sont éliminés par les critères ci-dessus, l'agent DEVRAIT retourner un message DIAMETER_UNABLE_TO_DELIVER.

6.1.8 Demandes de redirection

Lorsque un agent de redirection reçoit une demande dont l'entrée d'acheminement est réglée à REDIRECT, il DOIT répondre par un message dont le bit 'E' est établi, tout en maintenant l'identifiant bond par bond dans l'en-tête, et en incluant l'AVP Result-Code à DIAMETER_REDIRECT_INDICATION. Chacun des serveurs associés à l'entrée d'acheminement est entré dans une AVP Redirect-Host séparée.

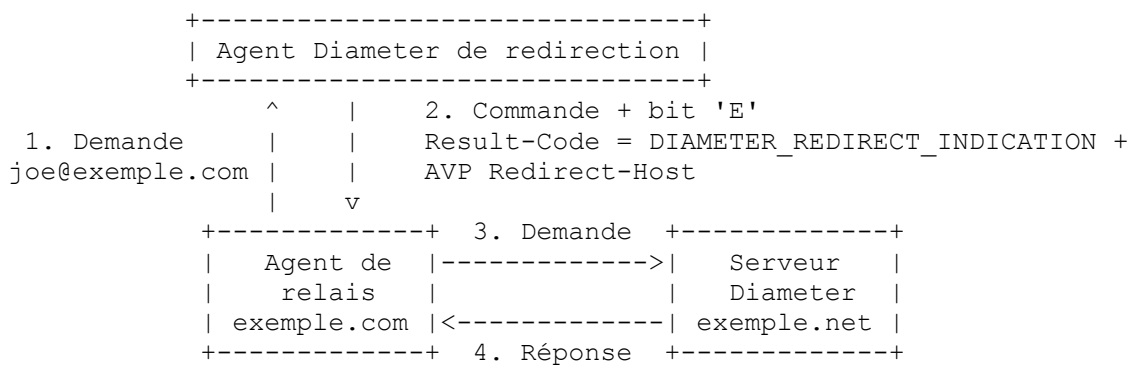


Figure 5 : Agent de redirection Diameter

Le receveur d'un message de réponse avec le bit 'E' établi et l'AVP Result-Code réglée à DIAMETER_REDIRECT_INDICATION utilise l'identifiant bond par bond dans l'en-tête Diameter pour identifier la demande dans la file d'attente de messages en instance (voir au paragraphe 5.5.4) qui doit être redirigée. Si il n'existe pas de connexion de transport avec le nouvel homologue, il en est créé un, et la demande lui est directement envoyée.

Plusieurs AVP Redirect-Host sont permises. Le receveur du message de réponse avec le bit 'E' établi choisit exactement un de ces hôtes comme destination du message redirigé.

Lorsque l'AVP Redirect-Host-Usage incluse dans le message de réponse a une valeur différente de zéro, il est créé une entrée d'acheminement pour les indications de redirection et elle est mise en antémémoire par le receveur. L'usage de la redirection pour une telle entrée de chemin est réglé par la valeur de l'AVP Redirect-Host-Usage et la durée de vie de l'entrée de chemin dans l'antémémoire est réglée par la valeur de l'AVP Redirect-Max-Cache-Time.

Il est possible que plusieurs indications de redirection puissent créer plusieurs entrées de chemin en antémémoire qui ne diffèrent que par leur usage de redirection et l'homologue auquel transmettre les messages. Par exemple, deux (2) entrées de chemin qui sont créées par deux (2) indications de redirection résultent en deux (2) chemins en antémémoire pour le même domaine et identifiant d'application. Cependant, l'une a un usage de redirection de ALL_SESSION, où les demandes qui correspondent vont être transmises à un homologue ; l'autre a un usage de redirection de ALL_REALM, où les demandes sont transmises à un autre homologue. Donc, une demande entrante qui correspond au domaine et identifiant d'application des deux chemins va avoir besoin d'outils de résolution supplémentaires. Dans ce cas, une règle de préséance d'acheminement DOIT être utilisée sur la valeur d'usage de redirection pour résoudre le conflit. On trouvera la règle de préséance au paragraphe 6.13.

6.1.9 Relais et mandatement des demandes

Un agent de relais ou mandataire DOIT ajouter une AVP Route-Record à toutes les demandes transmises. L'AVP contient l'identité de l'homologue d'où la demande a été reçue.

L'identifiant bond par bond dans la demande est sauvegardé et remplacé par une valeur localement unique. La source de la demande est aussi sauvegardée, ce qui inclut l'adresse IP, l'accès, et le protocole.

Un agent de relais ou mandataire PEUT inclure l'AVP Proxy-Info dans les demandes si il exige l'accès à des informations d'état local lorsque la réponse correspondante est reçue. L'AVP Proxy-Info a des implications de sécurité car les informations d'état sont distribuées aux autres entités. À ce titre, il est RECOMMANDÉ que le contenu de l'AVP Proxy-Info soit protégé par un mécanisme de chiffrement, par exemple, en utilisant un résumé de message chiffré comme HMAC-SHA1 [RFC2104]. Un tel mécanisme, exige cependant la gestion de clés, quoique seulement en local au serveur Diameter. Une pleine description de la gestion des clés utilisées pour protéger l'AVP Proxy-Info sort du domaine d'application du présent document. Voici une liste de recommandations courantes :

- o les clés devraient être générées de façon sûre en suivant les recommandations d'aléa de la [RFC4086] ;
- o les clés et les algorithmes de protection cryptographique devraient faire au moins 128 bits ;
- o les clés ne devraient pas être utilisées pour d'autre objet que la génération et vérification des instances d'AVP Proxy-Info

- o les clés devraient être changées régulièrement ;
- o les clés devraient être changées si le format d'AVP ou les algorithmes de protection cryptographique changent.

Le message est ensuite transmis au prochain bond, comme identifié sur le tableau d'acheminement.

La Figure 6 montre un exemple d'acheminement de message en utilisant les procédures mentionnées ci-dessus.

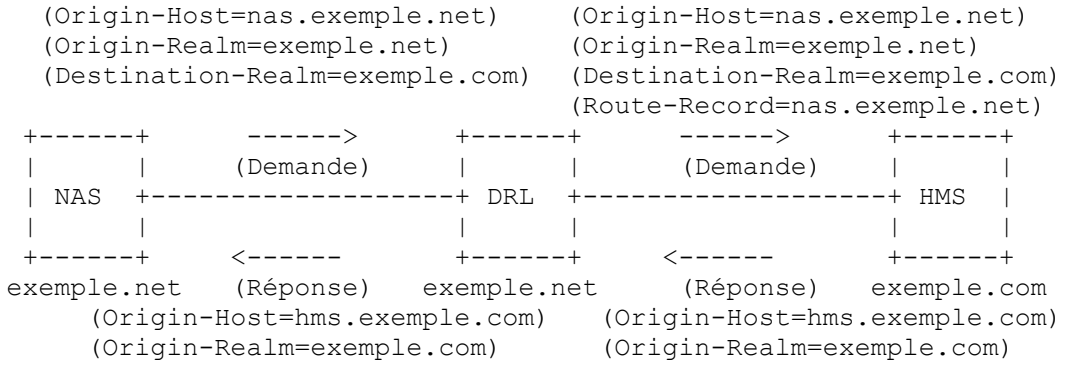


Figure 6 : Acheminement des messages Diameter

Les agents de relais et mandataires ne sont pas obligés d'effectuer une inspection complète des messages entrants. Au minimum, la validation de l'en-tête de message et des AVP d'acheminement pertinentes doit être faite lors du relais de messages. Les agents mandataires peuvent facultativement effectuer une validation de message plus en profondeur pour les applications qui les intéressent.

6.2 Traitement de la réponse Diameter

Lorsque une demande est traitée en local, les procédures suivantes DOIVENT être appliquées pour créer la réponse associée, en plus de toutes procédures supplémentaires qui PEUVENT être exposées dans l'application Diameter qui définit la commande :

- o Le même identifiant bond par bond dans la demande est utilisé dans la réponse.
- o L'identité de l'hôte local est codée dans l'AVP Origin-Host.
- o Les AVP Destination-Host et Destination-Realm NE DOIVENT PAS être présentes dans le message de réponse.
- o L'AVP Result-Code est ajoutée avec sa valeur qui indique le succès ou l'échec.
- o Si l'identifiant de session est présent dans la demande, il DOIT être inclus dans la réponse.
- o Toutes les AVP Proxy-Info de la demande DOIVENT être ajoutées au message de réponse, dans le même ordre que dans la demande.
- o Le bit 'P' est réglé à la même valeur que celui de la demande.
- o Le même identifiant de bout en bout que dans la demande est utilisé dans la réponse.

Noter que les messages d'erreur (voir à la Section 7) sont aussi soumis aux règles de traitement ci-dessus.

6.2.1 Traitement des réponses reçues

Un client ou mandataire Diameter DOIT confronter l'identifiant bond par bond d'une réponse reçue à la liste des demandes en cours. Le message correspondant devrait être retiré de la liste des demandes en cours. Il DEVRAIT ignorer les réponses reçues qui ne correspondent pas à un identifiant bond par bond connu.

6.2.2 Relais et mandatement des réponses

Si la réponse est pour une demande qui a été mandatée ou relayée, l'agent DOIT restaurer la valeur d'origine du champ Identifiant bond par bond de l'en-tête Diameter.

Si la dernière AVP Proxy-Info dans le message est ciblée sur le serveur Diameter local, l'AVP DOIT être retirée avant la transmission de la réponse.

Si un agent de relais ou mandataire reçoit une réponse avec une AVP Result-Code indiquant un échec, il NE DOIT PAS modifier le contenu de l'AVP. Toute autre erreur locale supplémentaire détectée DEVRAIT être enregistrée mais non reflétée dans l'AVP Result-Code. Si l'agent reçoit un message de réponse avec une AVP Result-Code indiquant la réussite, et si il souhaite modifier l'AVP pour indiquer une erreur, il DOIT modifier l'AVP Result-Code pour qu'elle contienne l'erreur appropriée dans le message destiné à l'appareil d'accès ainsi que pour qu'elle inclue l'AVP Error-Reporting-Host ; il DOIT

aussi produire un message STR au nom de l'appareil d'accès pour le serveur Diameter.

L'agent DOIT alors envoyer la réponse à l'hôte dont il a reçu la demande d'origine.

6.3 AVP Origin-Host

L'AVP Origin-Host (code d'AVP : 264) est du type DiameterIdentity, et elle DOIT être présente dans tous les messages Diameter. Cette AVP identifie le point d'extrémité qui a généré le message Diameter. Les agents de relais NE DOIVENT PAS modifier cette AVP.

La valeur de l'AVP Origin-Host est garantie d'être unique au sein d'un seul hôte.

Noter que l'AVP Origin-Host peut se résoudre en plus d'une adresse car l'homologue Diameter peut prendre en charge plus d'une adresse.

Cette AVP DEVRAIT être placée aussi près de l'en-tête Diameter que possible.

6.4 AVP Origin-Realm

L'AVP Origin-Realm (code d'AVP : 296) est du type DiameterIdentity. Cette AVP contient le domaine du générateur de tout message Diameter et DOIT être présente dans tous les messages.

Cette AVP DEVRAIT être placée aussi près que possible de l'en-tête Diameter.

6.5 AVP Destination-Host

L'AVP Destination-Host (code d'AVP : 293) est du type DiameterIdentity. Cette AVP DOIT être présente dans tous les messages non sollicités initiés par l'agent, PEUT être présente dans les messages de demande, et NE DOIT PAS être présente dans les messages de réponse.

L'absence de l'AVP Destination-Host va causer l'envoi d'un message à tout serveur Diameter qui prend en charge l'application au sein du domaine spécifié dans l'AVP Destination-Realm.

Cette AVP DEVRAIT être placée aussi près que possible de l'en-tête Diameter.

6.6 AVP Destination-Realm

L'AVP Destination-Realm (code d'AVP : 283) est du type DiameterIdentity et contient le domaine auquel le message sera acheminé. L'AVP Destination-Realm NE DOIT PAS être présente dans les messages de réponse. Le client Diameter insère la portion domaine de l'AVP User-Name. Les serveurs Diameter qui initient un message de demande utilisent la valeur de l'AVP Origin-Realm d'un message précédent reçu de l'hôte cible prévu (sauf si elle est connue a priori). Lorsque présente, l'AVP Destination-Realm est utilisée pour prendre les décisions d'acheminement du message.

Le CCF pour un message de demande qui inclut l'AVP Destination-Realm DEVRAIT mentionner les AVP Destination-Realm comme AVP obligatoires (une AVP indiquée par {AVP}) ; autrement, le message est par nature un message non acheminable.

Cette AVP DEVRAIT être placée aussi près que possible de l'en-tête Diameter.

6.7 AVP d'acheminement

Les AVP définies dans ce paragraphe sont des AVP Diameter utilisées pour les besoin de l'acheminement. Ces AVP changent lorsque les messages Diameter sont traités par les agents.

6.7.1 AVP Route-Record

L'AVP Route-Record (code d'AVP : 282) est du type DiameterIdentity. L'identité ajoutée dans cette AVP DOIT être la même que celle reçue dans l'AVP Origin-Host du message d'échange de capacités.

6.7.2 AVP Proxy-Info

L'AVP Proxy-Info (code d'AVP : 284) est du type Grouped. Cette AVP contient l'identité et les informations d'état local du nœud Diameter qui la crée et l'ajoute à un message. Le champ Données groupées a la grammaire de CCF suivante :

```
Proxy-Info ::= < En-tête d'AVP : 284 >
                 { Proxy-Host }
                 { Proxy-State }
               * [ AVP ]
```

6.7.3 AVP Proxy-Host

L'AVP Proxy-Host (code d'AVP : 280) est du type DiameterIdentity. Cette AVP contient l'identité de l'hôte qui ajoute l'AVP Proxy-Info.

6.7.4 AVP Proxy-State

L'AVP Proxy-State (code d'AVP : 33) est du type OctetString. Elle contient les informations d'état qui seraient autrement mémorisées chez l'entité Diameter qui l'a créée. À ce titre, cette AVP DOIT être traitée comme des données opaques par les autres entités Diameter.

6.8 AVP Auth-Application-Id

L'AVP Auth-Application-Id (code d'AVP : 258) est du type Unsigned32 et est utilisée afin d'annoncer la prise en charge de la portion authentification et autorisation d'une application (voir au paragraphe 2.4). Si elle est présente dans un message autre que CER et CEA, la valeur de l'AVP Auth-Application-Id DOIT correspondre à l'identifiant d'application présent dans l'en-tête du message Diameter.

6.9 AVP Acct-Application-Id

L'AVP Acct-Application-Id (code d'AVP : 259) est du type Unsigned32 et est utilisée afin d'annoncer la prise en charge de la portion comptabilité d'une application (voir au paragraphe 2.4). Si elle est présente dans un message autre que CER et CEA, la valeur de l'AVP Acct-Application-Id DOIT correspondre à l'identifiant d'application présent dans l'en-tête du message Diameter.

6.10 AVP Inband-Security-Id

L'AVP Inband-Security-Id (code d'AVP : 299) est du type Unsigned32 et est utilisée afin d'annoncer la prise en charge de la portion sécurité de l'application. L'utilisation de cette AVP dans les messages CER et CEA N'EST PAS RECOMMANDÉE. À la place, la découverte des capacités de sécurité d'une entité Diameter peut être effectuée soit par configuration statique soit via la découverte d'homologue Diameter comme décrit au paragraphe 5.2.

Les valeurs suivantes sont prises en charge :

NO_INBAND_SECURITY : 0. Cet homologue ne prend pas en charge TLS/TCP et DTLS/SCTP. C'est la valeur par défaut si l'AVP est omise.

TLS : 1. Ce nœud prend en charge la sécurité TLS/TCP [RFC5246] et DTLS/SCTP [RFC6083].

6.11 AVP Vendor-Specific-Application-Id

L'AVP Vendor-Specific-Application-Id (code d'AVP : 260) est du type Grouped et est utilisée pour annoncer la prise en charge d'une application Diameter spécifique d'un fabricant. Exactement une instance de l'AVP Auth-Application-Id ou Acct-Application-Id DOIT être présente. L'identifiant d'application porté par l'AVP Auth-Application-Id ou Acct-Application-Id DOIT se conformer à l'allocation d'identifiant d'application spécifique de fabricant décrite au paragraphe 11.3. Elle DOIT aussi correspondre à l'identifiant d'application présent dans l'en-tête Diameter excepté quand elle est utilisée dans un message CER ou CEA.

L'AVP Vendor-Id est une AVP informative qui relève du fabricant qui peut avoir les droits d'auteur de l'application Diameter spécifique de fabricant. Elle NE DOIT PAS être utilisée comme moyen de définir un espace d'identifiant d'application spécifique de fabricant complètement séparé.

L'AVP Vendor-Specific-Application-Id DEVRAIT être placée aussi près que possible de l'en-tête Diameter.

Format d'AVP :
<Vendor-Specific-Application-Id> ::= < En-tête d'AVP : 260 >
{ Vendor-Id }
[Auth-Application-Id]
[Acct-Application-Id]

Une AVP Vendor-Specific-Application-Id DOIT contenir exactement une de Auth-Application-Id ou Acct-Application-Id. Si un Vendor-Specific-Application-Id est reçu sans une de ces deux AVP, le receveur DEVRAIT alors produire une réponse avec un Result-Code réglé à DIAMETER_MISSING_AVP. La réponse DEVRAIT aussi inclure une Failed-AVP, qui DOIT contenir un exemple d'une AVP Auth-Application-Id et une AVP Acct-Application-Id.

Si une Vendor-Specific-Application-Id est reçue qui contient à la fois Auth-Application-Id et Acct-Application-Id, le receveur DOIT alors produire une réponse avec le Result-Code réglé à DIAMETER_AVP_OCCURS_TOO_MANY_TIMES *(l'AVP Diameter est trop répétée)*. La réponse DOIT aussi inclure une Failed-AVP, qui DOIT contenir les AVP Auth-Application-Id et Acct-Application-Id reçues.

6.12 AVP Redirect-Host

L'AVP Redirect-Host (code d'AVP : 292) est du type DiameterURI. Une ou plusieurs instances de cette AVP DOIVENT être présentes si le bit 'E' du message de réponse est établi et si l'AVP Result-Code est réglée à DIAMETER_REDIRECT_INDICATION *(indication de redirection Diameter)*.

À réception de cela, le nœud Diameter receveur DEVRAIT transmettre la demande directement à un des hôtes identifiés dans ces AVP. Le serveur contenu dans l'AVP Redirect-Host choisi DEVRAIT être utilisé pour tous les messages satisfaisant aux critères établis par l'AVP Redirect-Host-Usage.

6.13 AVP Redirect-Host-Usage

L'AVP Redirect-Host-Usage (code d'AVP : 261) est du type Enumerated. Cette AVP PEUT être présente dans les messages de réponse dont le bit 'E' est établi et dont l'AVP Result-Code est réglée à DIAMETER_REDIRECT_INDICATION.

Lorsque présente, cette AVP donne des indications sur la façon dont l'entrée d'acheminement résultant du Redirect-Host va être utilisée. Les valeurs suivantes sont acceptées :

DONT_CACHE : 0. L'hôte spécifié dans l'AVP Redirect-Host NE DEVRAIT PAS être mis en antémémoire. C'est la valeur par défaut.

ALL_SESSION : 1. Tous les messages de la même session, comme défini par la même valeur de l'AVP Session-ID DEVRAIENT être envoyés à l'hôte spécifié dans l'AVP Redirect-Host.

ALL_REALM : 2. Tous les messages destinés au domaine demandé DEVRAIENT être envoyés à l'hôte spécifié dans l'AVP Redirect-Host.

REALM_AND_APPLICATION : 3. Tous les messages pour l'application demandée au domaine spécifié DEVRAIENT être envoyés à l'hôte spécifié dans l'AVP Redirect-Host.

ALL_APPLICATION : 4. Tous les messages pour l'application demandée DEVRAIENT être envoyés à l'hôte spécifié dans l'AVP Redirect-Host.

ALL_HOST : 5. Tous les messages qui vont être envoyés à l'hôte qui a généré le Redirect-Host DEVRAIENT être envoyés à l'hôte spécifié dans l'AVP Redirect-Host.

ALL_USER : 6. Tous les messages pour l'usager demandé DEVRAIENT être envoyés à l'hôte spécifié dans l'AVP Redirect-Host.

Lorsque plusieurs chemins sont créés en antémémoire par des indications de redirection et qu'ils ne diffèrent que par l'usage de redirection et les homologues auxquels transmettre les demandes (voir au paragraphe 6.1.8) une règle de préséance DOIT être appliquée aux valeurs d'usage de redirection des chemins en antémémoire durant l'acheminement normal pour résoudre les conflits qui peuvent se produire. La régime de préséance est l'ordre qui dicte quel usage de redirection devrait

être retenu avant tout autre lorsque ils apparaissent. L'ordre est le suivant :

1. ALL_SESSION *(toutes les sessions)*
2. ALL_USER *(tous les utilisateurs)*
3. REALM_AND_APPLICATION *(domaine et application)*
4. ALL_REALM *(tous les domaines)*
5. ALL_APPLICATION *(toutes les applications)*
6. ALL_HOST *(tous les hôtes)*

6.14 AVP Redirect-Max-Cache-Time

L'AVP Redirect-Max-Cache-Time (code d'AVP : 262) est du type Unsigned32. Cette AVP DOIT être présente dans les messages de réponse dont le bit 'E' est à 1, dont l'AVP Result-Code est réglé à DIAMETER_REDIRECT_INDICATION, et dont l'AVP Redirect-Host-Usage est réglé à une valeur non zéro.

Cette AVP contient le nombre maximum de secondes pendant lequel les entrées de tableau d'homologue et de chemin, créées par suite du Redirect-Host, DEVRAIENT être conservées en antémémoire. Noter qu'une fois qu'un hôte n'est plus joignable, toutes les entrées associées de tableaux d'homologues et d'acheminement en antémémoire DOIVENT être supprimées.

7. Traitement des erreurs

Il y a deux différents types d'erreurs dans Diameter ; les erreurs de protocole et les erreurs d'application. Une erreur de protocole est celle qui survient au niveau du protocole de base et PEUT exiger une attention par bond (par exemple, une erreur d'acheminement de message). Les erreurs d'application, d'un autre côté, surviennent généralement à cause d'un problème avec une fonction spécifiée dans une application Diameter (par exemple, l'authentification de l'utilisateur, une AVP manquante).

Les valeurs d'AVP Result-Code qui sont utilisées pour rapporter les erreurs de protocole DOIVENT être seulement présentes dans les messages de réponse dont le bit 'E' est établi. Lorsque un message de demande est reçu et cause une erreur de protocole, un message de réponse est retourné avec le bit 'E' établi, et l'AVP Result-Code est réglée à la valeur d'erreur de protocole appropriée. Comme la réponse est renvoyée à l'origine de la demande, chaque agent mandataire ou de relais PEUT agir sur le message.

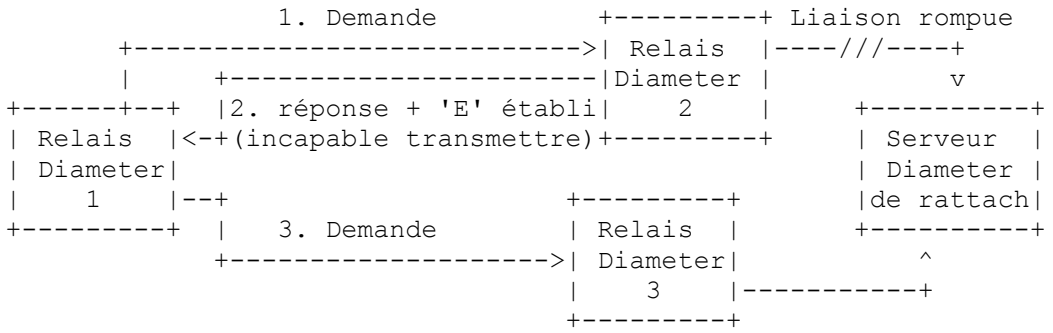


Figure 7 : Exemple d'erreur de protocole qui cause un message de réponse

La Figure 7 donne un exemple de message transmis en amont par un relais Diameter. Lorsque le message est reçu par Relais 2, et qu'il détecte qu'il ne peut pas transmettre la demande au serveur de rattachement, un message de réponse est retourné avec le bit 'E' établi et l'AVP Result-Code réglée à DIAMETER_UNABLE_TO_DELIVER. Étant donné que cette erreur rentre dans la catégorie des erreurs de protocole, Relais 1 va prendre une action particulière, et selon l'erreur, tenter d'acheminer le message par son Relais 3 de remplacement.

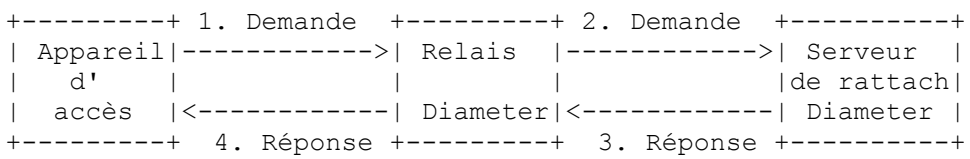


Figure 8 : Exemple de message de réponse d'erreur d'application

La Figure 8 donne un exemple de message Diameter qui a causé une erreur d'application. Lorsque une erreur d'application se produit, l'entité Diameter qui rapporte l'erreur met le bit 'R' à zéro dans les fanions de commandes et ajoute l'AVP Result-Code avec la valeur appropriée. Les erreurs d'application n'exigent aucun engagement de l'agent mandataire ou relais ; donc, le message va être retransmis au générateur de la demande.

Dans le cas où le message de réponse lui-même contient des erreurs, toute session qui s'y rapporte DEVRAIT être terminée par l'envoi d'un message STR ou ASR. L'AVP Termination-Cause dans le STR PEUT être remplie avec la valeur appropriée pour indiquer la cause de l'erreur. Une application PEUT aussi envoyer une demande spécifique de l'application au lieu d'un message STR ou ASR pour signaler l'erreur dans le cas où aucun état n'est conservé ou pour permettre certaines formes de récupération d'erreur avec l'entité Diameter correspondante.

Certaines erreurs d'application d'AVP Result-Code exigent que des AVP supplémentaires soient présentes dans la réponse. Dans ce cas, le nœud Diameter qui établit l'AVP Result-Code pour indiquer l'erreur DOIT ajouter les AVP. Des exemples suivent :

- Une demande avec une AVP non reconnue reçue avec le bit 'M' (bit "obligatoire") établi cause l'envoi d'une réponse avec l'AVP Result-Code réglée à DIAMETER_AVP_UNSUPPORTED et l'AVP Failed-AVP contenant l'AVP en cause.

- Une demande avec une AVP reçue avec une valeur non reconnue cause le retour d'une réponse avec l'AVP Result-Code réglée à DIAMETER_INVALID_AVP_VALUE, avec l'AVP Failed-AVP contenant l'AVP qui a causé l'erreur.

- Une commande reçue où manquent des AVP qui sont définies comme exigées dans le CCF de la commande ; les exemples sont les AVP indiquées comme {AVP}. Le receveur produit une réponse avec le Result-Code réglé à DIAMETER_MISSING_AVP et crée une AVP avec le code d'AVP et les autres champs réglés comme attendu dans l'AVP manquante. L'AVP créée est alors ajoutée à l'AVP Failed-AVP.

L'AVP Result-Code décrit l'erreur que le nœud Diameter a rencontrée lors du traitement. Si il y a plusieurs erreurs, le nœud Diameter DOIT rapporter seulement la première erreur rencontrée (détectée éventuellement dans un ordre dépendant de la mise en œuvre). Les erreurs spécifiques qui peuvent être décrites par cette AVP sont traitées dans les paragraphes suivants.

7.1 AVP Result-Code

L'AVP Result-Code (code d'AVP : 268) est du type Unsigned32 et indique si une demande particulière a été achevée avec succès ou si une erreur s'est produite. Tous les messages de réponse Diameter dans les spécifications d'application Diameter définies par l'IETF DOIVENT inclure une AVP Result-Code. Une AVP Result-Code d'échec (qui contient une valeur non 2xxx autre que DIAMETER_REDIRECT_INDICATION) DOIT inclure l'AVP Error-Reporting-Host si l'hôte qui établit l'AVP Result-Code est différent de l'identité codée dans l'AVP Origin-Host.

Le champ de données de code de résultat contient un espace d'adresse de 32 bits géré par l'IANA qui représente les erreurs (voir au paragraphe 11.3.2). Diameter fournit les classes d'erreurs suivantes, toutes identifiées par le chiffre des milliers en notation décimale :

- 1xxx (information)
- 2xxx (succès)
- 3xxx (erreurs de protocole)
- 4xxx (échecs provisoires)
- 5xxx (échecs permanents)

Une classe non reconnue (dont le premier chiffre n'est pas défini dans ce paragraphe) DOIT être traitée comme un échec permanent.

7.1.1 Information

Les erreurs qui entrent dans cette catégorie sont utilisées pour informer le demandeur qu'une demande pourrait n'être pas

satisfaite, et qu'une action supplémentaire est requise de sa part avant que l'accès soit accordé.

DIAMETER_MULTI_ROUND_AUTH : 1001. Cette erreur pour information est retournée par un serveur Diameter pour informer l'appareil d'accès que le mécanisme d'authentification utilisé exige plusieurs allers-retours, et qu'une autre demande doit être produite afin que l'accès soit accordé.

7.1.2 Succès

Les erreurs qui entrent dans cette catégorie de succès sont utilisées pour informer un homologue qu'une demande a été achevée avec succès.

DIAMETER_SUCCESS : 2001. La demande s'est achevée par un succès.

DIAMETER_LIMITED_SUCCESS : 2002. Lorsque ce résultat est retourné, la demande est achevée avec succès, mais un traitement supplémentaire est nécessaire de la part de l'application afin de fournir le service à l'usager.

7.1.3 Erreurs de protocole

Les erreurs qui entrent dans la catégorie des erreurs de protocole DEVRAIENT être traitées bond par bond, et les mandataires Diameter PEUVENT tenter de corriger l'erreur, si c'est possible. Noter que ces erreurs ne DOIVENT être utilisées que dans les messages de réponse dont le bit 'E' est établi.

DIAMETER_COMMAND_UNSUPPORTED : 3001. Ce code d'erreur est utilisée lorsque une entité Diameter reçoit un message avec un code de commande qu'elle ne prend pas en charge.

DIAMETER_UNABLE_TO_DELIVER : 3002. Cette erreur est produite lorsque Diameter ne peut pas livrer le message à la destination, soit parce que aucun hôte prenant en charge l'application requise n'était disponible dans le domaine pour traiter la demande, soit parce que l'AVP Destination-Host a été donnée sans l'AVP Destination-Realm associée.

DIAMETER_REALM_NOT_SERVED : 3003. Le domaine prévu pour la demande n'est pas reconnu.

DIAMETER_TOO_BUSY : 3004. Lorsque ce code est retourné, un nœud Diameter DEVRAIT tenter d'envoyer le message à un homologue de remplacement. Cette erreur ne DOIT être utilisée que lorsque un serveur spécifique est demandé, et qu'il ne peut pas fournir le service demandé.

DIAMETER_LOOP_DETECTED : 3005. Un agent a détecté une boucle en essayant d'envoyer le message au receveur prévu. Le message PEUT être envoyé à un homologue de remplacement, si il en est un disponible, mais l'homologue qui rapporte l'erreur a identifié un problème de configuration.

DIAMETER_REDIRECT_INDICATION : 3006. Un agent de redirection a déterminé que la demande ne pouvait pas être satisfaite localement, et l'initiateur de la demande DEVRAIT diriger la demande directement sur le serveur, dont les informations de contact ont été ajoutées à la réponse. Dans ce cas, l'AVP Redirect-Host DOIT être présente.

DIAMETER_APPLICATION_UNSUPPORTED : 3007. Une demande a été envoyée pour une application qui n'est pas prise en charge.

DIAMETER_INVALID_HDR_BITS : 3008. Une demande est reçue dont les bits dans l'en-tête Diameter sont réglés à une combinaison invalide ou à une valeur non cohérente avec la définition du code de commande.

DIAMETER_INVALID_AVP_BITS : 3009. Une demande est reçue qui inclut une AVP dont les bits fanions sont réglés à une valeur non reconnue ou qui est incohérente avec la définition de l'AVP.

DIAMETER_UNKNOWN_PEER : 3010. Une CER a été reçue d'un homologue inconnu.

7.1.4 Défaillances temporaires

Les erreurs qui entrent dans la catégorie des erreurs temporaires sont utilisées pour informer un homologue que la demande n'a pas pu être satisfaite au moment où elle a été reçue, mais qu'il PEUT être capable de satisfaire la demande à l'avenir. Noter que ces erreurs DOIVENT être utilisées dans les messages de réponse dont le bit 'E' est à zéro.

DIAMETER_AUTHENTIFICATION_REJECTED : 4001. Le processus d'authentification a échoué pour l'usager, très probablement à cause d'un mot de passe invalide utilisé par l'usager. De nouveaux essais ne DOIVENT être tentés

qu'après avoir invité l'usager à formuler un nouveau mot de passe.

DIAMETER_OUT_OF_SPACE : 4002. Un nœud Diameter a reçu la demande comptable mais a été incapable de l'affecter à une mémorisation stable à cause d'un manque temporaire d'espace.

ELECTION_LOST : 4003. L'homologue a déterminé qu'il a perdu le processus d'élection et a donc déconnecté la connexion de transport.

7.1.5 Défaillances permanentes

Les erreurs qui entrent dans la catégorie des défaillances permanentes sont utilisées pour informer l'homologue que la demande a échoué et qu'elle ne devrait pas être tentée à nouveau. Noter que ces erreurs DEVRAIENT être utilisées dans des messages de réponse dont le bit 'E' est à zéro. Dans les conditions d'erreur où il n'est pas possible ou efficace de composer une grammaire de réponse spécifique de l'application, les messages de réponse avec le bit 'E' établi et qui se conforment à la grammaire décrite au paragraphe 7.2 PEUVENT aussi être utilisés pour les erreurs permanentes.

DIAMETER_AVP_UNSUPPORTED : 5001. L'homologue a reçu un message qui contenait une AVP qui n'est pas reconnue ou supportée et a été marquée du bit 'M' (obligatoire). Un message Diameter avec cette erreur DOIT contenir une ou plusieurs AVP Failed-AVP contenant les AVP qui ont causé la défaillance.

DIAMETER_UNKNOWN_SESSION_ID : 5002. La demande contenait un identifiant de session inconnu.

DIAMETER_AUTORISATION_REJECTED : 5003. Une demande a été reçue pour laquelle l'usager n'a pas pu être autorisé. Cette erreur peut survenir si le service demandé n'est pas permis à l'usager.

DIAMETER_INVALID_AVP_VALUE : 5004. La demande contenue dans une AVP avait une valeur invalide dans sa portion de données. Un message Diameter qui indique cette erreur DOIT inclure les AVP en cause dans une AVP Failed-AVP.

DIAMETER_MISSING_AVP : 5005. La demande ne contenait pas une AVP exigée par la définition du code de commande. Si cette valeur est envoyée dans l'AVP Result-Code, une AVP Failed-AVP DEVRAIT être incluse dans le message. L'AVP Failed-AVP DOIT contenir un exemple de l'AVP manquante complète avec l'identifiant de fabricant si applicable. Le champ Valeur de l'AVP manquante devrait être de la longueur minimum correcte et contenir des zéros.

DIAMETER_RESOURCES_EXCEEDED : 5006. Une demande a été reçue qui ne peut pas être autorisée parce que l'usager a déjà dépensé les ressources allouées. Un exemple de cette condition d'erreur est lorsque un usager qui est restreint à un accès PPP commuté tente d'établir une seconde connexion PPP.

DIAMETER_CONTRADICTING_AVPS : 5007. Le serveur Diameter de rattachement a détecté que les AVP dans la demande se contredisent, et il refuse de fournir le service à l'usager. L'AVP Failed-AVP DOIT être présente, contenant les AVP qui se contredisent.

DIAMETER_AVP_NOT_ALLOWED : 5008. Un message a été reçu avec une AVP qui NE DOIT PAS être présente. L'AVP Failed-AVP DOIT être incluse et contenir une copie de l'AVP en cause.

DIAMETER_AVP_OCCURS_TOO_MANY_TIMES : 5009. Un message reçu incluait une AVP qui apparaît plus souvent que permis dans la définition du message. L'AVP Failed-AVP DOIT être incluse et contenir une copie de la première instance de l'AVP en cause qui excédait le nombre maximum d'occurrences.

DIAMETER_NO_COMMON_APPLICATION : 5010. Cette erreur est retournée par un nœud Diameter qui reçoit un CER par lequel aucune application n'est commune entre l'homologue qui envoie le CER et celui qui le reçoit.

DIAMETER_UNSUPPORTED_VERSION : 5011. Cette erreur est retournée lorsque le numéro de version de la demande reçue n'est pas pris en charge.

DIAMETER_UNABLE_TO_COMPLY : 5012. Cette erreur est retournée lorsque une demande est rejetée pour des raisons non spécifiées.

DIAMETER_INVALID_BIT_IN_HEADER : 5013. Cette erreur est retournée lorsque un bit réservé dans l'en-tête Diameter est établi à un (1) ou lorsque les bits dans l'en-tête Diameter sont réglés de façon incorrecte.

DIAMETER_INVALID_AVP_LENGTH : 5014. La demande contient une AVP avec une longueur invalide. Un message

Diameter indiquant cette erreur DOIT inclure les AVP en cause dans une AVP Failed-AVP. Si la valeur de longueur de l'AVP erronée excède la longueur du message ou fait moins que la longueur minimum d'en-tête d'AVP, il est suffisant d'inclure l'en-tête de l'AVP en cause et une charge utile remplie de zéros de la longueur minimum requise pour le type de données de charge utile. Si l'AVP est de type Grouped, l'en-tête d'AVP Grouped avec une charge utile vide sera suffisant pour indiquer l'AVP en cause. Si l'en-tête de l'AVP en cause ne peut pas être complètement décodé lorsque la longueur de l'AVP est inférieure à la longueur minimum d'en-tête d'AVP, il est suffisant d'inclure l'en-tête de l'AVP en cause qui est formulé en bourrant l'en-tête de l'AVP incomplète avec des zéros jusqu'à la longueur minimum d'en-tête d'AVP.

DIAMETER_INVALID_MESSAGE_LENGTH : 5015. Cette erreur est retournée lorsque une demande est reçue avec une longueur de message invalide.

DIAMETER_INVALID_AVP_BIT_COMBO : 5016. La demande contenait une AVP avec laquelle il n'est pas permis d'avoir cette valeur dans le champ Fanions d'AVP. Un message Diameter qui indique cette erreur DOIT inclure les AVP en cause dans une AVP Failed-AVP.

DIAMETER_NO_COMMON_SECURITY : 5017. Cette erreur est retournée lorsque un message CER est reçu, et qu'il n'y a pas de mécanisme de sécurité commun entre les homologues. Un message de réponse d'échange de capacités (CEA, *Capabilities-Exchange-Answer*) DOIT être retourné avec l'AVP Result-Code réglée à DIAMETER_NO_COMMON_SECURITY.

7.2 Bit Erreur

Le bit fanion 'E' (bit d'erreur) dans l'en-tête Diameter est établi lorsque la demande a causé une erreur en rapport avec le protocole (voir au paragraphe 7.1.3). Un message qui a le bit 'E' NE DOIT PAS être envoyé en réponse à un message de réponse. Noter qu'un message avec le bit 'E' est encore soumis aux règles de traitement définies au paragraphe 6.2. Lorsque il est établi, le message de réponse ne va pas se conformer à la spécification de CCF pour la commande ; à la place, il va se conformer au CCF suivant :

Format de message :

```
<answer-message> ::= < En-tête Diameter : code, ERR [, PXY] >
                    0*1< Session-Id >
                       { Origin-Host }
                       { Origin-Realm }
                       { Result-Code }
                      [ Origin-State-Id ]
                      [ Error-Message ]
                      [ Error-Reporting-Host ]
                      [ Failed-AVP ]
                      [ Experimental-Result ]
                     * [ Proxy-Info ]
                     * [ AVP ]
```

Noter que le code utilisé dans l'en-tête est le même que celui qu'on trouve dans le message de demande, mais avec le bit 'R' à zéro et le bit 'E' établi. Le bit 'P' dans l'en-tête est réglé à la même valeur que celle du message de demande.

7.3 AVP Error-Message

L'AVP Error-Message (code d'AVP : 281) est du type UTF8String. Elle PEUT accompagner une AVP Result-Code comme message d'erreur lisible par l'homme. L'AVP Error-Message n'est pas destinée à être utile dans un environnement où les messages d'erreur sont traités automatiquement. On NE DEVRAIT PAS s'attendre à ce que le contenu de cette AVP soit analysé par les entités du réseau.

7.4 AVP Error-Reporting-Host

L'AVP Error-Reporting-Host (code d'AVP : 294) est du type DiameterIdentity. Cette AVP ne contient l'identité de l'hôte Diameter qui a envoyé l'AVP Result-Code à une valeur autre que 2001 (Succès) que si l'hôte qui règle le code de résultat est différent de celui qui est codé dans l'AVP Origin-Host. Cette AVP est destinée à être utilisée à des fins de résolution de problèmes, et elle DOIT être établie lorsque l'AVP Result-Code indique une défaillance.

7.5 AVP Failed-AVP

L'AVP Failed-AVP (code d'AVP : 279) est du type Grouped et fournit des informations de débogage dans les cas où une demande est rejetée ou non complètement traitée à cause d'informations erronées dans une AVP spécifique. La valeur de l'AVP Result-Code va fournir des informations sur la raison de l'AVP Failed-AVP. Un message de réponse Diameter DEVRAIT contenir une instance de l'AVP Failed-AVP qui correspond à l'erreur indiquée par l'AVP Result-Code. Pour des besoins pratiques, cette AVP Failed-AVP va normalement se référer à la première erreur de traitement d'AVP que rencontre un nœud Diameter.

Les raisons possibles de cette AVP sont la présence d'une AVP construite de façon inappropriée, une AVP non prise en charge ou non reconnue, une valeur d'AVP invalide, l'omission d'une AVP obligatoire, la présence d'une AVP explicitement exclue (voir les tableaux de la Section 10) ou la présence de deux occurrences ou plus d'une AVP qui est restreinte à 0, 1, ou 0-1 occurrences.

Un message Diameter DEVRAIT contenir une AVP Failed-AVP, contenant l'AVP entière qui n'a pas pu être traitée avec succès. Si la raison de la défaillance est l'omission d'une AVP obligatoire, une AVP avec le code d'AVP manquante, l'identifiant de fabricant manquant, et une charge utile remplie de zéros de la longueur minimum requise pour l'AVP omise sera ajoutée. Si la raison de la défaillance est une longueur d'AVP invalide où la longueur rapportée est inférieure à la longueur minimum d'en-tête d'AVP ou supérieure à la longueur rapportée du message, une copie de l'en-tête d'AVP en cause et une charge utile remplie de zéros de la longueur minimum requise DEVRAIT être ajoutée.

Dans le cas où l'AVP en cause est incorporée dans une AVP Grouped, l'AVP Failed-AVP PEUT contenir l'AVP groupée, qui à son tour contient la seule AVP en cause. La même méthode PEUT être employée si l'AVP groupée elle-même est incorporée dans une autre AVP déjà groupée et ainsi de suite. Dans ce cas, l'AVP Failed-AVP PEUT contenir la hiérarchie d'AVP groupées jusqu'à la seule AVP en cause. Cela permet au receveur de détecter la localisation de l'AVP en cause lorsque elle est incorporée dans un groupe.

Format d'AVP :
<Failed-AVP> ::= < En-tête d'AVP : 279 > 1* {AVP}

7.6 AVP Experimental-Result

L'AVP Experimental-Result (code d'AVP : 297) est du type Grouped, et indique si une demande particulière spécifique d'un fabricant s'est achevée avec succès ou si une erreur s'est produite. Cette AVP a la structure suivante :

Format d'AVP :
Experimental-Result ::= < En-tête d'AVP : 297 > { Vendor-Id } { Experimental-Result-Code }

L'AVP Vendor-Id (voir au paragraphe 5.3.3) dans cette AVP groupée identifie le fabricant responsable de l'allocation du code de résultat qui suit. Tous les messages de réponse Diameter définis dans des applications spécifiques de fabricant DOIVENT inclure une AVP Result-Code ou une AVP Experimental-Result.

7.7 AVP Experimental-Result-Code

L'AVP Experimental-AVP-Result-Code (code d'AVP : 298) est du type Unsigned32 et contient une valeur allouée par le fabricant qui représente le résultat du traitement de la demande.

Il est recommandé que les codes de résultat spécifiques de fabricant suivent les mêmes conventions que données pour les AVP Result-Code concernant les différents types de codes de résultat et le traitement des erreurs (pour les valeurs autres que 2xxx).

8. Sessions d'utilisateur Diameter

En général, Diameter peut fournir deux différents types de services aux applications. Le premier implique l'authentification et l'autorisation, et il peut facultativement utiliser la comptabilité. Le second utilise seulement la comptabilité.

Lorsque un service utilise la portion authentification et/ou autorisation d'une application, et qu'un usager demande l'accès au réseau, le client Diameter produit une demande d'authentification à son serveur local. La demande d'authentification est définie dans une application Diameter spécifique du service (par exemple, NASREQ). La demande contient une AVP Session-Id, qui est utilisée dans les messages suivants (par exemple, les autorisations, la comptabilité, etc.) relatifs à la

session de l'usager. L'AVP Session-Id est un moyen pour que clients et serveurs corrèlent un message Diameter avec une session d'utilisateur.

Lorsque un serveur Diameter autorise un usager à mettre en œuvre des ressources du réseau pour une durée finie, et qu"il veut étendre l'autorisation via une demande future, il DOIT ajouter l'AVP Authorization-Lifetime au message de réponse. L'AVP Authorization-Lifetime définit le nombre maximum de secondes pendant lequel un usager PEUT utiliser les ressources avant qu'une autre demande d'autorisation soit attendue par le serveur. L'AVP Auth-Grace-Period contient le nombre de secondes suivant l'expiration de la durée de vie d'autorisation, après lequel le serveur va libérer toutes les informations d'état qui se rapportent à la session de l'usager. Noter que si le domaine qui dessert le domaine de rattachement de l'usager attend un payement pour les services, l'AVP Authorization-Lifetime, combiné avec l'AVP Auth-Grace-Period, implique la longueur maximum de la session pour laquelle le domaine de rattachement accepte d'être fiscalement responsable. Les services fournis après l'expiration des AVP Authorization-Lifetime et Auth-Grace-Period sont de la responsabilité de l'appareil d'accès. Bien sûr, le coût réel des services rendus sort clairement du domaine d'application de ce protocole.

Un appareil d'accès qui ne s'attend pas à envoyer une demande de réautorisation ou de terminaison de session au serveur PEUT inclure l'AVP Auth-Session-State avec la valeur réglée à NO_STATE_MAINTAINED comme indication au serveur. Si le serveur accepte l'indication, il accepte que comme aucun message de terminaison de session ne va être reçu une fois le service à l'usager terminé, il ne peut pas maintenir l'état pour la session. Si le message de réponse du serveur contient une valeur différente dans l'AVP Auth-Session-State (ou la valeur par défaut si l'AVP est absente) l'appareil d'accès DOIT suivre les directives du serveur. Noter que la valeur NO_STATE_MAINTAINED NE DOIT PAS être établie dans les demandes et réponses suivantes de réautorisation.

Le protocole de base n'inclut aucun message de demande d'autorisation, car ceux-ci sont largement spécifiques des applications et sont définis dans un document d'application Diameter. Cependant, le protocole de base définit bien un ensemble de messages qui sont utilisés pour terminer les sessions d'utilisateur. Ils sont utilisés pour permettre aux serveurs qui conservent les informations d'état de libérer les ressources.

Lorsque un service utilise seulement la portion comptabilité du protocole Diameter, même en combinaison avec une application, l'identifiant de session est quand même utilisé pour identifier les sessions d'utilisateur. Cependant, les messages de terminaison de session ne sont pas utilisés, car une session est signalée comme étant terminée en produisant un message d'arrêt de comptabilité.

Diameter peut aussi être utilisé pour des services qui ne peuvent pas être facilement catégorisés comme authentification, autorisation, ou comptabilité (par exemple, certaines interfaces du système multimédia Internet de projet de partenariat de troisième génération (3GPP IMS, *Third Generation Partnership Project Internet Multimedia System*)). Dans ce cas, l'automate à états finis défini dans les paragraphes qui suivent peut n'être pas applicable. Donc, l'application elle-même PEUT devoir définir son propre automate à états finis. Cependant, de tels automates à états finis spécifiques d'application DEVRAIENT suivre le cadre général d'automate à états décrit dans le présent document comme l'utilisation des AVP Session-Id et des messages STR/STA, ASR/ASA pour les sessions à états pleins.

8.1 Automate à états de session d'autorisation

Ce paragraphe contient un ensemble d'automates à états finis, qui représentent le cycle de vie des sessions Diameter et qui DOIVENT être respectés par toutes les mises en œuvre Diameter qui utilisent la portion authentification et/ou autorisation d'une application Diameter. Le terme "spécifique du service" ci-dessous se réfère à un message défini dans une application Diameter (par exemple, IPv4 mobile, NASREQ).

Il y a quatre automates à états de session d'autorisation différents qui sont pris en charge dans le protocole de base Diameter. Les deux premiers décrivent une session dans laquelle le serveur conserve l'état de session, indiqué par la valeur de l'AVP Auth-Session-State (ou son absence). L'un décrit la session du point de vue du client, l'autre du point de vue du serveur. Les deux seconds automates à états sont utilisés lorsque le serveur ne conserve pas l'état de session. Là encore, l'un décrit la session du point de vue du client, et l'autre du point de vue du serveur.

Lorsque une session passe à l'état Repos, toutes les ressources qui étaient alloués à cette session doivent être libérées. Tout événement non mentionné dans l'automate à états DOIT être considéré comme une condition d'erreur, et une réponse, si applicable, DOIT être retournée au générateur du message.

Au cas où une application ne prend pas en charge la réauthentification, lorsque les deux sessions de client et de serveur conservent l'état (par exemple, Envoi RAR, En cours, Reçoit RAA) les transitions d'état relatives à la réauthentification à l'initiative du serveur PEUVENT être ignorées.

Dans le tableau d'état, l'événement "Échec à envoyer X" signifie que l'agent Diameter est incapable d'envoyer la commande X à la destination désirée. Cela peut être dû à ce que l'homologue est hors service ou à ce que l'homologue renvoie une notification de défaillance temporaire ou une erreur temporaire de protocole DIAMETER_TOO_BUSY ou DIAMETER_LOOP_DETECTED dans l'AVP Result-Code de la commande de réponse correspondante. L'événement 'X envoyé avec succès' est le complément de 'Échec de l'envoi de X'.

L'automate à états suivant est appliqué par un client lorsque l'état est conservé au serveur :

Client, états pleins

État	Événement	Action	Nouvel état
Repos	Le client ou appareil demande l'accès	Envoi demande autorisation spécifique du service	En cours
Repos	ASR reçu pour session inconnue	Envoie ASA avec code de résultat = UNKNOWN_SESSION_ID	Repos
Repos	RAR reçue pour session inconnue	Envoie RAA avec code de résultat = UNKNOWN_SESSION_ID	Repos
En cours	Réponse d'autorisation spécifique du service réussie reçue avec valeur par défaut de Auth-Session-State	Accorde l'accès	Ouvert
En cours	Réponse d'autorisation spécifique du service réussie reçue, mais service non fourni	STR envoyée	Déconnecté
En cours	Erreur de traitement de réponse d'autorisation spécifique du service réussie	STR envoyée	Déconnecté
En cours	Réponse d'échec d'autorisation spécifique du service reçue	Nettoyage	Repos
Ouvert	L'appareil d'usager ou de client demande l'accès au service	Envoi demande autor. spécifique du service	Ouvert
Ouvert	Réponse d'autorisation spécifique du service réussie reçue	Fournit le service	Ouvert
Ouvert	Réponse d'échec d'autorisation spécifique du service reçue.	Usager/appareil déconnecté.	Repos
Ouvert	RAR reçue et le client va effectuer la réauthentification suivante	Envoie RAA avec code de résultat = SUCCESS	Ouvert
Ouvert	RAR reçue et le client ne va pas effectuer la réauthentification suivante	Envoie RAA avec code de résultat != SUCCESS, usager/appareil déconnecté.	Repos
Ouvert	Le temporisateur de session expire sur l'appareil d'accès	Envoie STR	Déconnecté
Ouvert	ASR reçue, le client se conforme à la demande de fin de session	Envoie ASA avec code de résultat = SUCCESS, Envoie STR.	Déconnecté
Ouvert	ASR reçue, le client ne va pas se conformer à la demande de fin de session	Envoie ASA avec code de résultat != SUCCESS	Ouvert
Ouvert	Durée de vie d'autorisation + Auth-Grace-Period expire sur l'appareil d'accès	Envoie STR	Déconnecté
Déconnecté	ASR reçue	Envoie ASA	Déconnecté
Déconnecté	STA reçue	Usager/appareil déconnecté.	Repos

L'automate à états suivant est respecté par un serveur lorsque il conserve l'état pour la session :

Serveur, états pleins

État	Événement	Action	Nouvel état
Repos	Demande d'autorisation spécifique du service reçue, et l'usager est autorisé	Envoie réponse de succès spécifique du service	Ouvert
Repos	Demande d'autorisation spécifique du service reçue, et l'usager n'est pas autorisé	Envoie réponse d'échec spécifique du service	Repos
Ouvert[2]	Demande d'autorisation spécifique du service reçue, et l'usager est autorisé	Envoie réponse de succès spécifique du service	Ouvert
Ouvert	Demande d'autorisation spécifique du service reçue, et l'usager n'est pas autorisé	Envoie réponse d'échec spécifique du service, nettoyage	Repos
Ouvert	Le serveur de rattachement veut confirmer l'authentification et/ou autorisation de l'usager	Envoie RAR	En cours
En cours	RAA reçue avec code de résultat d'échec	Nettoyage	Repos
En cours	RAA reçue avec code de résultat = SUCCESS	Mise à jour de la session	Ouvert
Ouvert	Le serveur de rattachement veut terminer le service	Envoie ASR	Déconnecté
Ouvert	Authorisation-Lifetime (et Auth-Grace-Period)	Nettoyage	Repos

	expire sur le serveur de rattachement		
Ouvert	Session-Timeout expire sur le serveur de rattachement	Nettoyage	Repos
Déconnecté	Échec de l'envoi de ASR	Attente, renvoi de ASR	Déconnecté
Déconnecté	ASR bien envoyé et ASA reçu avec Result-Code	Nettoyage	Repos
Non déconnecté	ASA reçue	Aucune	Pas de changement
Any	STR reçue	Envoie STA, nettoyage	Repos

L'automate à états suivant est observé par un client lorsque l'état n'est pas conservé sur le serveur :

Client, sans état

État	Événement	Action	Nouvel état
Repos	Le client ou appareil demande l'accès	Envoi demande d'autorisation spécifique du service	En cours
En cours	Réponse d'autorisation spécifique du service réussie reçue avec Auth-Session-State réglé à NO_STATE_MAINTAINED	Accorde l'accès	Ouvert
En cours	Réponse d'échec d'autorisation spécifique du service reçue	Nettoyage	Repos
Ouvert	Session-Timeout expire sur l'appareil d'accès	Usager/appareil déconnecté.	Repos
Ouvert	Le service à l'usager est terminé	Usager/appareil déconnecté.	Repos

L'automate à états suivant est observé par un serveur quand il ne conserve pas l'état pour la session :

Serveur, sans état

État	Événement	Action	Nouvel état
Repos	Demande d'autorisation spécifique du service reçue, et traitée avec succès	Envoie réponse spécifique du service	Repos

8.2 Automate à états de session de comptabilité

Les automates à états suivants DOIVENT être pris en charge pour les applications qui ont une portion comptabilité ou qui n'exigent que des services de comptabilité. Le premier automate à états doit être observé par les clients.

Voir au paragraphe 9.7 les codes de commande pour la comptabilité et au paragraphe 9.8 pour les AVP de comptabilité.

Le côté serveur dans l'automate à états de comptabilité dépend dans certains cas de l'application particulière. Le protocole de base Diameter définit un automate à états par défaut qui DOIT être suivi par toutes les applications qui n'ont pas spécifié d'autres automates à états. C'est le second automate à états de ce paragraphe.

L'automate à états côté serveur par défaut exige la réception d'enregistrements de comptabilité dans n'importe quel ordre et à tout moment, et il n'impose aucune exigence sur le traitement de ces enregistrements. Les mises en œuvre de Diameter peuvent effectuer des tâches de vérification, mise en ordre, corrélation, détection de fraude, et autres sur la base de ces enregistrements. Les AVP peuvent devoir être inspectées au titre de ces tâches. Les tâches peuvent se faire immédiatement après la réception de l'enregistrement ou dans une phase postérieure. Cependant, comme ces tâches dépendent normalement de l'application ou même de la politique, elles ne sont pas normalisées par la spécification Diameter. Les applications PEUVENT définir des exigences sur le moment où accepter des enregistrements de comptabilité sur la base de la valeur utilisée de l'AVP Accounting-Realtime-Required, de vérification de limites de crédit, et ainsi de suite.

Cependant, le protocole de base Diameter définit un automate à états côté serveur facultatif qui PEUT être suivi par les applications qui exigent de garder trace de l'état de session au serveur de comptabilité. Noter qu'un tel traçage est incompatible avec la capacité de résoudre le problème de la connexité de longue durée. Donc, l'utilisation de cet automate à états n'est recommandée que dans les applications où la valeur de l'AVP Accounting-Realtime-Required est DELIVER_AND_GRANT ; donc, les problèmes de connexité comptable sont obligés de causer la déconnexion de l'usager desservi. Autrement, les enregistrements produits par le client peuvent être perdus par le serveur, qui ne les accepte plus après le rétablissement de la connexité. Cet automate à états est le troisième automate à états de ce paragraphe. L'automate à états est supervisé par un temporisateur de supervision de session Ts, dont la valeur devrait être raisonnablement supérieure à la valeur de Acct_Interim_Interval. Ts PEUT être réglé à deux fois la valeur de Acct_Interim_Interval afin d'éviter que la session comptable passe dans le serveur Diameter à l'état Repos en cas de brève défaillance temporaire du réseau.

Tout événement non mentionné dans les automates à états DOIT être considéré comme condition d'erreur, et une réponse correspondante, si applicable, DOIT être retournée au générateur du message.

Dans le tableau d'états, l'événement "échec d'envoi" signifie que le client Diameter est dans l'incapacité de communiquer avec la destination désirée. Cela peut être dû à un arrêt de l'homologue, ou au renvoi par l'homologue d'une défaillance temporaire ou d'une notification d'erreur de protocole temporaire DIAMETER_OUT_OF_SPACE, DIAMETER_TOO_BUSY, ou DIAMETER_LOOP_DETECTED dans l'AVP Result-Code de la commande Réponse de comptabilité.

L'événement "échec de réponse" signifie que le client Diameter a reçu une notification de défaillance non temporaire dans la commande Réponse de comptabilité.

Noter que l'action "Déconnecter usager/appareil" DOIT aussi avoir un effet sur le tableau d'états de session d'autorisation, par exemple, causer l'envoi du message STR, si l'application en cause a les deux portions authentification/autorisation et comptabilité.

Les états PendingS, PendingI, PendingL, PendingE, et PendingB signifient les états en instance d'attente d'une réponse à des demandes de comptabilité relatives respectivement à des états Start, Interim, Stop, Event, ou des enregistrements en mémoire tampon *(Buffered)*.

Client, comptabilité

État	Événement	Action	Nouvel état
Repos	Le client ou appareil demande l'accès	Envoi demande début de comptabilité	PendingS
Repos	Le client ou appareil demande un service unique	Envoi demande d'événement de comptabilité	PendingE
Repos	Enregistrements en mémorisation	Envoi d'enregistrement	PendingB
PendingS	Réponse de début de comptabilité réussie reçue		Ouvert
PendingS	Échec d'envoi et espace de mémoire tampon disponible et temps réel non égal à DELIVER_AND_GRANT	Mémorise le début d'enregistrement	Ouvert
PendingS	Échec d'envoi et pas d'espace de mémoire tampon disponible et temps réel égal à GRANT_AND_LOSE		Ouvert
PendingS	Échec d'envoi et pas d'espace de mémoire tampon disponible et temps réel non égal à GRANT_AND_LOSE	Déconnecte usager/appareil	Repos
PendingS	Réponse d'échec de début de comptabilité reçue et temps réel égal à GRANT_AND_LOSE		Ouvert
PendingS	Réponse d'échec de début de comptabilité reçue et temps réel non égal à GRANT_AND_LOSE	Déconnecte usager/appareil	Repos
PendingS	Service d'usager terminé	Mémorise le début d'enregistrement	PendingS
Ouvert	Intervalle intermédiaire écoulé	Envoi de l'enregistrement intermédiaire de comptabilité	PendingI
Ouvert	Service d'usager terminé	Envoi demande d'arrêt de comptabilité	PendingL
PendingI	Réponse de comptabilité intermédiaire réussie reçue		Ouvert
PendingI	Échec d'envoi (espace de mémoire tampon disponible ou le vieil enregistrement peut être écrasé) et temps réel non égal à DELIVER_AND_GRANT	Mémorise l'enregistrement intermédiaire	Ouvert
PendingI	Échec d'envoi et pas d'espace de mémoire tampon disponible et temps réel égal à GRANT_AND_LOSE		Ouvert
PendingI	Échec d'envoi et pas d'espace de mémoire tampon disponible et temps réel non égal à GRANT_AND_LOSE	Déconnecte usager/appareil	Repos
PendingI	Réponse d'échec de comptabilité intermédiaire reçue et temps réel égal à GRANT_AND_LOSE		Ouvert
PendingI	Réponse d'échec de comptabilité intermédiaire reçue et temps réel non égal à GRANT_AND_LOSE	Déconnecte usager/appareil	Repos
PendingI	Service d'usager terminé	Mémorise l'enregistrement d'arrêt	PendingI
PendingE	Réponse d'événement de comptabilité réussi reçue		Repos
PendingE	Échec d'envoi et espace de mémoire tampon disponible	Mémorise l'enregistrement d'événement	Repos
PendingE	Échec d'envoi et pas d'espace de mémoire tampon disponible		Repos
PendingE	Réponse d'échec d'événement de comptabilité reçue		Repos
PendingB	Réponse de comptabilité réussie reçue	Supprime l'enregistrement	Repos
PendingB	Échec d'envoi		Repos

PendingB	Réponse d'échec de comptabilité reçue	Supprime l'enregistrement	Repos
PendingL	Réponse d'arrêt de comptabilité réussi reçue		Repos
PendingL	Échec d'envoi et espace de mémoire tampon disponible	Mémorise l'enregistrement d'arrêt	Repos
PendingL	Échec d'envoi et pas d'espace de mémoire tampon disponible		Repos
PendingL	Réponse d'échec d'arrêt de comptabilité reçue		Repos

Serveur, comptabilité sans état

État	Événement	Action	Nouvel état
Repos	Demande de début de comptabilité reçue et traitée avec succès	Envoie réponse de début de comptabilité	Repos
Repos	Demande d'événement de comptabilité reçue et traitée avec succès	Envoie réponse d'événement de comptabilité	Repos
Repos	Enregistrement intermédiaire reçu et traité avec succès.	Envoie réponse de comptabilité intermédiaire	Repos
Repos	Demande d'arrêt de comptabilité reçu et traitée avec succès	Envoi réponse d'arrêt de comptabilité	Repos
Repos	Demande de comptabilité reçue; pas d'espace pour mémoriser les enregistrements	Envoi réponse de comptabilité ; code de résultat = OUT_OF_SPACE	Repos

Serveur, comptabilité à états pleins

État	Événement	Action	Nouvel état
Repos	Demande de début de comptabilité reçue et traitée avec succès	Envoie réponse de début de comptabilité ; lancer Ts	Ouvert
Repos	Demande d'événement de comptabilité reçue et traitée avec succès.	Envoie réponse d'événement de comptabilité	Repos
Repos	Demande de comptabilité reçue; pas d'espace pour mémoriser les enregistrements	Envoi réponse de comptabilité ; code de résultat = OUT_OF_SPACE	Repos
Ouvert	Enregistrement intermédiaire reçu et traité avec succès.	Envoie réponse de comptabilité intermédiaire ; redémarrer Ts	Ouvert
Ouvert	Demande d'arrêt de comptabilité reçu et traitée avec succès	Envoi réponse d'arrêt de comptabilité ; arrêter Ts	Repos
Ouvert	Demande de comptabilité reçue; pas d'espace pour mémoriser les enregistrements	Envoi réponse de comptabilité ; code de résultat = OUT_OF_SPACE ; arrêter Ts	Repos
Ouvert	Expiration du temporisateur de supervision de session Ts	Arrêter Ts	Repos

8.3 Réautorisation à l'initiative du serveur

Un serveur Diameter peut initier un service de réauthentification et/ou réautorisation pour une session particulière en produisant une Re-Auth-Request (RAR) *(demande de réautorisation)*.

Par exemple, pour des services prépayés, le serveur Diameter qui a à l'origine autorisé une session peut avoir besoin d'une confirmation que l'usager utilise toujours les services.

Un appareil d'accès qui reçoit un message RAR avec l'identifiant de session égal à une session actuellement active DOIT initier une réauthentification à l'égard de l'usager, si le service prend en charge ce dispositif particulier. Chaque application Diameter DOIT déclarer si la réauthentification à l'initiative du serveur est prise en charge, car certaines applications ne permettent pas que l'appareil d'accès invite l'usager à se réauthentifier.

8.3.1 Re-Auth-Request

Le message Re-Auth-Request (RAR), indiqué par le code de commande réglé à 258 et le bit 'R' des fanions de message établi, peut être envoyé par tout serveur à l'appareil d'accès qui reçoit le service de session, pour demander que l'usager soit réauthentifié et/ou réautorisé.

Format de message :

```
<RAR>  ::= < En-tête Diameter : 258, REQ, PXY >
           < Session-Id >
           { Origin-Host }
           { Origin-Realm }
           { Destination-Realm }
```

```
        { Destination-Host }
        { Auth-Application-Id }
        { Re-Auth-Request-Type }
        [ User-Name ]
        [ Origin-State-Id ]
      * [ Proxy-Info ]
      * [ Route-Record ]
      * [ AVP ]
```

8.3.2 Re-Auth-Answer

Le message Re-Auth-Answer (RAA), indiqué par le code de commande réglé à 258 et le bit 'R' des fanions de message à zéro, est envoyé en réponse au RAR. L'AVP Result-Code DOIT être présente, et elle indique la disposition de la demande.

Un message RAA réussi DOIT être suivi par un message spécifique d'application d'authentification et/ou autorisation.

Format de message :

```
<RAA>  ::= < En-tête Diameter : 258, PXY >
             < Session-Id >
             { Result-Code }
             { Origin-Host }
             { Origin-Realm }
             [ User-Name ]
             [ Origin-State-Id ]
             [ Error-Message ]
             [ Error-Reporting-Host ]
             [ Failed-AVP ]
           * [ Redirect-Host ]
             [ Redirect-Host-Usage ]
             [ Redirect-Max-Cache-Time ]
           * [ Proxy-Info ]
           * [ AVP ]
```

8.4 Terminaison de session

Il est nécessaire qu'un serveur Diameter qui a autorisé une session, pour laquelle il conserve l'état, soit notifié de l'arrêt d'activité d'une session, pour les besoins du traçage ainsi que pour permettre aux agents à états pleins de libérer toutes les ressources qu'ils ont pu fournir à la session de l'usager. Pour les sessions dont l'état n'est pas conservé, cette section n'est pas utilisée.

Lorsque une session d'utilisateur qui exigeait une autorisation Diameter se termine, l'appareil d'accès qui a fourni le service DOIT produire un message Session-Termination-Request (STR) au serveur Diameter qui a autorisé le service, pour lui notifier que la session n'est plus active. Un STR DOIT être produit lorsque une session d'utilisateur se termine quelle que soit la raison, incluant la déconnexion de l'usager, l'expiration du temporisateur de session, une action administrative, la terminaison à réception d'une demande d'interruption de session (Abort-Session-Request) (voir ci-dessous), une fermeture régulière de l'appareil d'accès, etc.

L'appareil d'accès DOIT aussi produire un STR pour une session qui a été autorisée mais n'a en fait jamais commencé. Ceci peut se produire, par exemple, du fait d'un manque soudain de ressource dans l'appareil d'accès, ou parce que l'appareil d'accès ne veut pas fournir le type de service demandé dans l'autorisation, ou parce que l'appareil d'accès ne prend pas en charge une AVP obligatoire retournée dans l'autorisation, etc.

Il est aussi possible qu'une session qui était autorisée n'ai jamais commencé à cause d'une action d'un mandataire. Par exemple, un mandataire peut modifier une réponse d'autorisation, convertir le résultat de succès en échec, avant de transmettre le message à l'appareil d'accès. Si la réponse ne contenait pas une AVP Auth-Session-State avec la valeur NO_STATE_MAINTAINED, un mandataire qui cause le non démarrage d'une session autorisée DOIT produire un STR au serveur Diameter qui a autorisé la session, car l'appareil d'accès n'a pas de moyen de savoir que la session a été autorisée.

Un serveur Diameter qui reçoit un message STR DOIT libérer les ressources (par exemple, l'état de session) associées au Session-Id spécifié dans le STR et retourner une Session-Termination-Answer.

Un serveur Diameter DOIT aussi libérer les ressources lorsque le temporisateur de session arrive à expiration, ou lorsque

les AVP Autorisation-Lifetime et Auth-Grace-Period expirent sans que soit reçue une demande de réautorisation, sans considération de l'éventuelle réception d'un STR pour cette session. L'appareil d'accès n'est pas supposé fournir de service au delà de l'expiration de ces temporisateurs ; donc, l'expiration de l'un ou l'autre de ces temporisateurs implique que l'appareil d'accès peut subir une fermeture inattendue.

8.4.1 Session-Termination-Request

Le message Session-Termination-Request (STR), indiqué par le code de commande 275 et le bit 'R' des fanions de commande établi, est envoyé par un client Diameter ou par un mandataire Diameter pour informer le serveur Diameter qu'une session authentifiée et/ou autorisée se termine.

Format de message :

```
<STR>  ::= < En-tête Diameter : 275, REQ, PXY >
             < Session-Id >
             { Origin-Host }
             { Origin-Realm }
             { Destination-Realm }
             { Auth-Application-Id }
             { Termination-Cause }
             [ User-Name ]
             [ Destination-Host ]
           * [ Class ]
             [ Origin-State-Id ]
           * [ Proxy-Info ]
           * [ Route-Record ]
           * [ AVP ]
```

8.4.2 Session-Termination-Answer

Le message Session-Termination-Answer (STA), indiqué par le code de commande 275 et le bit 'R' des fanions de message à zéro, est envoyé par le serveur Diameter pour accuser réception de la notification de la terminaison de la session. L'AVP Result-Code DOIT être présente, et elle PEUT contenir l'indication qu'une erreur s'est produite pendant le service de la STR.

À l'envoi ou la réception de la STA, le serveur Diameter DOIT libérer toutes les ressources pour la session indiquée par l'AVP Session-Id. Tout serveur intermédiaire dans la chaîne des mandataires PEUT aussi libérer toutes les ressources, si nécessaire.

Format de message :

```
<STA> ::= < En-tête Diameter : 275, PXY >
            < Session-Id >
            { Result-Code }
            { Origin-Host }
            { Origin-Realm }
            [ User-Name ]
          * [ Class ]
            [ Error-Message ]
            [ Error-Reporting-Host ]
            [ Failed-AVP ]
            [ Origin-State-Id ]
          * [ Redirect-Host ]
            [ Redirect-Host-Usage ]
            [ Redirect-Max-Cache-Time ]
          * [ Proxy-Info ]
          * [ AVP ]
```

8.5 Interruption d'une session

Un serveur Diameter peut demander que l'appareil d'accès arrête de fournir le service pour une certaine session en produisant une demande d'interruption de session (ASR, Abort-Session-Request).

Par exemple, le serveur Diameter qui a autorisé à l'origine la session peut être obligé de causer l'arrêt de la session pour manque de crédit ou d'autres raisons qui n'ont pas été prévues lorsque la session a été autorisée.

Un appareil d'accès qui reçoit une ASR avec l'identifiant de session égal à une session actuellement active PEUT arrêter la session. Il appartient à la mise en œuvre et/ou la configuration de décider si l'appareil d'accès arrête ou non la session. Par exemple, un appareil d'accès peut honorer les ASR provenant seulement de certains agents. Dans tous les cas, l'appareil d'accès DOIT répondre par un message Abort-Session-Answer, incluant une AVP Result-Code pour indique l'action prise.

8.5.1 Abort-Session-Request

Le message Abort-Session-Request (ASR), indiqué par le code de commande 274 et le bit 'R' des fanions de message établi, peut être envoyé par tout serveur Diameter ou tout mandataire Diameter à l'appareil d'accès qui fournit un service de session, pour demander que la session identifiée par le Session-Id soit arrêtée.

Format de message :

```
<ASR>  ::= < En-tête Diameter : 274, REQ, PXY >
             < Session-Id >
             { Origin-Host }
             { Origin-Realm }
             { Destination-Realm }
             { Destination-Host }
             { Auth-Application-Id }
             [ User-Name ]
             [ Origin-State-Id ]
           * [ Proxy-Info ]
           * [ Route-Record ]
           * [ AVP ]
```

8.5.2 Abort-Session-Answer

Le message Abort-Session-Answer (ASA), indiqué par le code de commande 274 et le bit 'R' des fanions de message à zéro, est envoyé en réponse à la ASR. L'AVP Result-Code DOIT être présente et indiquer la disposition de la demande.

Si la session identifiée par le Session-Id dans l'ASR s'est terminée avec succès, le code de résultat est réglé à DIAMETER_SUCCESS. Si la session n'est pas actuellement active, le code de résultat est réglé à DIAMETER_UNKNOWN_SESSION_ID. Si l'appareil d'accès n'arrête pas la session pour une autre raison, le code de résultat est réglé à DIAMETER_UNABLE_TO_COMPLY.

Format de message :

```
<ASA>  ::= < En-tête Diameter : 274, PXY >
             < Session-Id >
             { Result-Code }
             { Origin-Host }
             { Origin-Realm }
             [ User-Name ]
             [ Origin-State-Id ]
             [ Error-Message ]
             [ Error-Reporting-Host ]
             [ Failed-AVP ]
           * [ Redirect-Host ]
             [ Redirect-Host-Usage ]
             [ Redirect-Max-Cache-Time ]
           * [ Proxy-Info ]
           * [ AVP ]
```

8.6 Déduction d'une terminaison de session de Origin-State-Id

L'AVP Origin-State-Id est utilisée pour permettre la détection de sessions terminées pour lesquelles aucun STR n'aurait été produit, dû à une fermeture imprévue d'un appareil d'accès.

Un client Diameter ou appareil d'accès incrémente la valeur de Origin-State-Id chaque fois qu'il est démarré ou allumé. La

nouvelle Origin-State-Id est alors envoyée dans un message CER/CEA dès la connexion au serveur. Le serveur Diameter qui reçoit la nouvelle Origin-State-Id peut déterminer si le client Diameter envoyeur a fermé de façon abrupte en comparant si la vieille valeur de la Origin-State-Id qu'il a conservé pour ce client spécifique est moins que la nouvelle valeur et si il a des sessions non terminées originaires de ce client.

Un appareil d'accès peut aussi inclure le Origin-State-Id dans des messages de demande autres que le CER si il y a des relais ou des mandataires entre l'appareil d'accès et le serveur. Dans ce cas, cependant, le serveur ne peut pas découvrir que l'appareil d'accès a été redémarré tant qu'il n'a pas reçu de lui une nouvelle demande. Donc, ce mécanisme est plus opportuniste selon les mandataires et relais.

Le serveur Diameter peut supposer que toutes les sessions qui ont été actives avant la détection du redémarrage d'un client ont été terminées. Le serveur Diameter PEUT nettoyer tout l'état de session associé à de telles sessions perdues, et il PEUT aussi produire des STR pour toutes ces sessions perdues qui étaient autorisées sur les serveurs vers l'amont, pour permettre que l'état de session soit globalement nettoyé.

8.7 AVP Auth-Request-Type

L' AVP Auth-Request-Type (code d'AVP : 274) est du type Enumerated et est incluse dans des demandes d'authentification spécifiques d'application pour informer les homologues si un usager doit être seulement authentifié, autorisé, ou les deux. Noter que toute valeur autre que les deux PEUT causer des problèmes d'interopérabilité avec RADIUS. Les valeurs suivantes sont définies :

AUTHENTICATE_ONLY : 1. La demande envoyée est seulement pour authentification, et elle DOIT contenir les AVP d'authentification pertinentes spécifiques de l'application nécessaires au serveur Diameter pour authentifier l'usager.

AUTHORIZE_ONLY : 2. La demande envoyée est seulement pour autorisation, et elle DOIT contenir les AVP d'autorisation spécifiques de l'application qui sont nécessaires pour identifier le service demandé/offert.

AUTHORIZE_AUTHENTICATE : 3. La demande concerne à la fois l'authentification et l'autorisation. Elle DOIT inclure les informations d'authentification pertinentes spécifiques de l'application et les informations d'autorisation nécessaires pour identifier le service demandé/offert.

8.8 AVP Session-Id

L' AVP Session-Id (code d'AVP : 263) est du type UTF8String et est utilisée pour identifier une session spécifique (voir la Section 8). Tous les messages relevant d'une session spécifique DOIVENT n'inclure qu'une seule AVP Session-Id, et la même valeur DOIT être utilisée tout au long de la vie de la session. Lorsque elle est présente, l'AVP Session-Id DEVRAIT apparaître immédiatement après l'en-tête Diameter (voir la Section 3).

L'AVP Session-Id DOIT être globalement et éternellement unique, car elle est destinée à identifier de façon univoque une session d'utilisateur sans référence à d'autres informations et il peut être nécessaire de corréler l'historique des informations d'authentification avec les informations de comptabilité. Le Session-Id inclut une portion obligatoire et une portion définie par la mise en œuvre ; un format recommandé pour la portion définie par la mise en œuvre est décrit ci-dessous.

Le Session-Id DOIT commencer par l'identité de l'envoyeur codée dans le type DiameterIdentity (voir au paragraphe 4.3.1). Le reste du Session-Id est délimité par un caractère ";" , et il PEUT être toute séquence dont le client peut garantir qu'il sera éternellement univoque ; cependant, le format suivant est recommandé, (les crochets angulaires [] indiquent un élément facultatif) : <DiameterIdentity>;<32 bits de poids fort>;<32 bits de moindre poids>[;<valeur facultative>]
<32 bits de poids fort> et <32 bits de moindre poids> sont les représentations décimales de 32 bits de poids fort et de moindre poids d'une valeur de 64 bits à accroissement monotone. La valeur de 64 bits est rendue en deux parties pour simplifier le formatage par les processeurs à 32 bits. Au démarrage, les 32 bits de poids fort de la valeur de 64 bits PEUVENT être initialisés à l'heure en format NTP [RFC5905], et les 32 bits de moindre poids PEUVENT être initialisés à zéro. Cela va pour des raisons pratiques éliminer la possibilité de chevauchement des identifiants de session après un réamorçage, en supposant que le procès de réamorçage prend plus d'une seconde. Autrement, une mise en œuvre PEUT garder trace de la valeur croissante dans une mémoire non volatile. <valeur facultative> est spécifique de la mise en œuvre, mais peut inclure l'identifiant d'appareil d'un modem, une adresse de couche 2, un horodatage, etc.

Exemple, dans lequel il n'y a pas de valeur facultative : accesspoint7.exemple.com;1876543210;523

Exemple, dans lequel figure une valeur facultative : accesspoint7.exemple.com;1876543210;523;mobile@200.1.1.88

L'identifiant de session est créé par l'application Diameter qui initie la session, ce qui dans la plupart des cas, est fait par le client. Noter qu'un Session-Id PEUT être utilisé pour les commandes d'authentification, d'autorisation et de comptabilité d'une application.

8.9 AVP Authorization-Lifetime

L'AVP Autorisation-Lifetime (code d'AVP : 291) est du type Unsigned32 et contient le nombre maximum de secondes de service à fournir à l'usager avant qu'il soit réauthentifié et/ou réautorisé. Il faut faire attention quand la valeur de Autorisation-Lifetime est déterminée, car une valeur non zéro faible pourrait créer un trafic Diameter significatif , qui pourrait congestionner à la fois le réseau et les agents.

Une valeur de zéro (0) signifie qu'une réauthentification immédiate est nécessaire pour l'appareil d'accès. L'absence de cette AVP, ou une valeur toute de uns (ce qui veut dire que tous les bits dans le champ de 32 bits sont réglés à un) signifie qu'aucune réauthentification n'est attendue.

Si cette AVP et l'AVP Session-Timeout sont toutes deux présentes dans un message, la valeur de cette dernière NE DOIT PAS être inférieure à celle de l'AVP Autorisation-Lifetime.

Une AVP Autorisation-Lifetime PEUT être présente dans des messages de réautorisation, et elle contient le nombre de secondes pendant lequel l'usager est autorisé à recevoir le service à partir du moment où le message de réponse de réautorisation est reçu par l'appareil d'accès.

Cette AVP PEUT être fournie par le client comme conseil sur la durée de vie maximum qu'il veut accepter. Le serveur DOIT retourner une valeur égale ou inférieure à celle fournie par le client.

8.10 AVP Auth-Grace-Period

L'AVP Auth-Grace-Period (code d'AVP : 276) est du type Unsigned32 et contient le nombre de secondes pendant lequel le serveur Diameter va attendre à la suite de l'expiration de l'AVP Autorisation-Lifetime avant de nettoyer les ressources pour la session.

8.11 Auth-Session-State

L'AVP Auth-Session-State (code d'AVP : 277) est du type Enumerated et spécifie si l'état est conservé pour une certaine session. Le client PEUT inclure cette AVP dans les demandes comme conseil au serveur, mais la valeur dans le message de réponse du serveur est obligatoire. Les valeurs suivantes sont prises en charge :

STATE_MAINTAINED : 0. Cette valeur est utilisée pour spécifier que l'état de la session est conservé, et que l'appareil d'accès DOIT produire un message de terminaison de session lorsque le service à l'usager est terminé. C'est la valeur par défaut.

NO_STATE_MAINTAINED : 1. Cette valeur est utilisée pour spécifier qu'aucun message de terminaison de session ne sera envoyé par l'appareil d'accès à l'expiration de la durée de vie d'autorisation.

8.12 Re-Auth-Request-Type

L'AVP Re-Auth-Request-Type (code d'AVP : 285) est du type Enumerated et est incluse dans les réponses d'autorisation spécifiques de l'application pour informer le client de l'action attendue à l'expiration de la durée de vie d'autorisation.

Si le message de réponse contient une AVP Autorisation-Lifetime avec une valeur positive, l'AVP Re-Auth-Request-Type DOIT être présente dans un message de réponse. Les valeurs suivantes sont définies :

AUTHORIZE_ONLY : 0. Une réautorisation seulement d'autorisation est attendue à l'expiration de la durée de vie d'autorisation. C'est la valeur par défaut si l'AVP n'est pas présente dans les messages de réponse qui incluent la durée de vie d'autorisation.

AUTHORIZE_AUTHENTICATE : 1. Une réautorisation d'authentification et d'autorisation est attendue à l'expiration de la durée de vie d'autorisation.

8.13 AVP Session-Timeout

L'AVP Session-Timeout (code d'AVP : 27) [RFC2865] est du type Unsigned32 et contient le nombre maximum de secondes de service à fournir à l'usager avant la terminaison de la session. Lorsque les deux AVP Session-Timeout et Autorisation-Lifetime sont présentes dans un message de réponse, la première DOIT être égale ou supérieure à la valeur de la deuxième.

Une session qui se termine sur un appareil d'accès à cause de l'expiration de Session-Timeout DOIT causer l'envoi d'un STR, sauf si l'appareil d'accès et le serveur de rattachement s'étaient préalablement mis tous deux d'accord pour qu'aucun message de terminaison de session ne soit envoyé (voir la Section 8).

Une AVP Session-Timeout PEUT être présente dans un message de réponse de réautorisation, et elle contient le nombre de secondes restantes depuis le début de la réautorisation.

Une valeur de zéro, ou l'absence de cette AVP, signifie que cette session a un nombre illimité de secondes avant sa terminaison.

Cette AVP PEUT être fournie par le client comme conseil sur la temporisation maximum qu'il veut accepter. Cependant, le serveur PEUT retourner une valeur qui est égale ou inférieure à celle fournie par le client.

8.14 AVP User-Name

L'AVP User-Name (code d'AVP : 1) [RFC2865] est du type UTF8String, qui contient le nom de l'usager, dans un format cohérent avec la spécification du NAI [RFC4282].

8.15 AVP Termination-Cause

L'AVP Termination-Cause (code d'AVP : 295) est du type Enumerated, et est utilisée pour indiquer la raison pour laquelle une session s'est terminée sur l'appareil d'accès. Les valeurs actuellement allouées pour cette AVP se trouvent dans le registre IANA des valeurs d'AVP Termination-Cause [IANATCV].

8.16 AVP Origin-State-Id

L'AVP Origin-State-Id (code d'AVP : 278) de type Unsigned32, est une valeur d'accroissement monotone qui est augmentée chaque fois qu'une entité Diameter redémarre avec perte de l'état antérieur, par exemple, sur un réamorçage. L'AVP Origin-State-Id PEUT être incluse dans tout message Diameter, y compris CER.

Une entité Diameter qui produit cette AVP DOIT créer une valeur supérieure pour cette AVP chaque fois que son état est restauré. Une entité Diameter PEUT régler Origin-State-Id à l'heure de démarrage, ou elle PEUT utiliser un compteur incrémentaire conservé dans une mémoire non volatile à travers les redémarrages.

L'AVP Origin-State-Id, si elle est présente, DOIT refléter l'état de l'entité indiquée par Origin-Host. Si un mandataire modifie Origin-Host, il DOIT retirer le Origin-State-Id ou le modifier de façon appropriée. Normalement, Origin-State-Id est utilisée par un appareil d'accès qui démarre toujours sans session active; c'est-à-dire que toute session active avant le redémarrage aura été perdue. En incluant Origin-State-Id dans un message, cela permet aux autres entités Diameter de déduire que les sessions associées à un Origin-State-Id inférieur ne sont pus actives. Si un appareil d'accès ne veut pas qu'une telle déduction soit faite, il DOIT ne pas inclure Origin-State-Id dans le message ou régler sa valeur à 0.

8.17 AVP Session-Binding

L'AVP Session-Binding (code d'AVP : 270) est du type Unsigned32, et elle PEUT être présente dans les messages de réponse d'autorisation spécifiques d'application. Si elle est présente, cette AVP PEUT informer le client Diameter que tous les futurs messages de réauthentification spécifiques d'application et Session-Termination-Request pour cette session DOIVENT être envoyés au même serveur d'autorisation.

Ce champ est un gabarit binaire, et les bits suivants ont été définis :

RE_AUTH : 1. Lorsque il est à 1, les futurs messages de réauthentification pour cette session NE DOIVENT PAS inclure l'AVP Destination-Host. Lorsque il est à zéro, valeur par défaut, l'AVP Destination-Host DOIT être présente dans tous les messages de réauthentification pour cette session.

STR : 2. Lorsque il est à 1, le message STR pour cette session NE DOIT PAS inclure l'AVP Destination-Host. Lorsque il est

à zéro, valeur par défaut, l'AVP Destination-Host AVP DOIT être présente dans le message STR pour cette session.

ACCOUNTING : 4. Lorsque il est à 1, tous les messages de comptabilité pour cette session NE DOIVENT PAS inclure l'AVP Destination-Host. Lorsque il est à zéro, valeur par défaut, l'AVP Destination-Host, si elle est connue, DOIT être présente dans tous les messages de comptabilité pour cette session.

8.18 AVP Session-Server-Failover

L'AVP Session-Server-Failover (code d'AVP : 271) est du type Enumerated et PEUT être présente dans les messages de réponse d'autorisation spécifiques d'application qui n'incluent pas l'AVP Session-Binding ou incluent l'AVP Session-Binding avec un des bits réglé à zéro. Si elle est présente, cette AVP PEUT informer le client Diameter que si un message de réautorisation ou STR échoue à cause d'un problème de livraison, le client Diameter DEVRAIT produire un message suivant sans l'AVP Destination-Host. Lorsque absente, la valeur par défaut est REFUSE_SERVICE. Les valeurs suivantes sont prises en charge :

REFUSE_SERVICE : 0. Si la livraison du message de réautorisation ou de STR échoue, terminer le service avec l'usager et ne pas faire d'autre tentative.

TRY_AGAIN : 1. Si la livraison du message de réautorisation ou de STR échoue, renvoyer le message en échec sans l'AVP Destination-Host.

ALLOW_SERVICE : 2. Si la livraison du message de réautorisation échoue, supposer que la réautorisation a réussi. Si la livraison du message STR échoue, terminer la session.

TRY_AGAIN_ALLOW_SERVICE : 3. Si la livraison du message de réautorisation et/ou de STR échoue, renvoyer le message en échec sans l'AVP Destination-Host. Si la seconde livraison échoue pour la réautorisation, supposer que la réautorisation a réussi. Si la seconde livraison échoue pour STR, terminer la session.

8.19 AVP Multi-Round-Time-Out

L'AVP Multi-Round-Time-Out (code d'AVP : 272) est du type Unsigned32 et DEVRAIT être présente dans les messages de réponse d'autorisation spécifiques d'application dont l'AVP Result-Code est réglée à DIAMETER_MULTI_ROUND_AUTH. Cette AVP contient le nombre maximum de secondes pendant lequel l'appareil d'accès DOIT attendre que l'usager réponde à une demande d'authentification.

8.20 AVP Class

L'AVP Class (code d'AVP : 25) est du type OctetString et est utilisée par les serveurs Diameter pour retourner les informations d'état à l'appareil d'accès. Lorsque une ou plusieurs AVP Class sont présentes dans les messages de réponse d'autorisation spécifiques d'application, elles DOIVENT être présentes dans les messages de réautorisation, de terminaison de session et de comptabilité suivants. Les AVP Class trouvées dans un message de réponse de réautorisation prennent le pas sur celles qui se trouvent dans tout message de réponse d'autorisation antérieur. Les mises en œuvre de serveur Diameter NE DEVRAIENT PAS retourner les AVP Class qui exigent plus de 4096 octets de mémorisation sur le client Diameter. Un client Diameter qui reçoit des AVP Classe dont la taille excède la mémorisation locale disponible DOIT terminer la session.

8.21 AVP Event-Timestamp

L'AVP Event-Timestamp (code d'AVP : 55) est du type Time et PEUT être incluse dans des messages de demande/réponse de comptabilité pour enregistrer l'heure à laquelle l'événement rapporté s'est produit, en secondes depuis le 1er janvier 1900 00:00 UTC.

9. Comptabilité

Ce protocole de comptabilité se fonde sur un modèle dirigé par un serveur avec des capacités de livraison en temps réel des informations comptables. Plusieurs méthodes de résilience aux fautes [RFC2975] ont été construites dans le protocole afin de minimiser la perte des données comptables dans diverses situations de faute et sous différentes hypothèses sur les capacités des appareils utilisés.

9.1 Modèle dirigé par le serveur

Le modèle dirigé par le serveur signifie que l'appareil qui génère les données comptables obtient les informations soit du serveur d'autorisation (si il est contacté) soit du serveur de comptabilité concernant la façon dont les données comptables devront être transmises. Ces informations incluent les exigences de traitement en temps utile des enregistrements comptables.

Comme exposé dans la [RFC2975], le transfert en temps réel des enregistrements comptables est une exigence, comme le besoin d'effectuer des vérifications de limite de crédit et de détection de fraude. Noter que la comptabilité par lots n'est pas une exigence, et n'est donc pas prise en charge par Diameter. Si la comptabilité par lots devaient être exigée à l'avenir, une nouvelle application Diameter devrait être créée, ou elle pourrait être traitée en utilisant un autre protocole. Noter cependant que même si à la couche Diameter, les demandes comptables sont traitées une par une, les protocoles de transport utilisés avec Diameter mettent normalement en lots plusieurs demandes dans le même paquet dans des conditions de fort trafic. Ceci peut être suffisant pour de nombreuses applications.

Le serveur (la chaîne de serveurs) d'autorisation dirige le choix de la stratégie de transfert appropriée, sur la base de sa connaissance de l'usager et des relations de partenariat d'itinérance. Le serveur (ou les agents) utilisent les AVP Acct-Interim-Interval et Accounting-Realtime-Required pour contrôler le fonctionnement de l'homologue Diameter opérant comme client. L'AVP Acct-Interim-Interval, lorsque présente, ordonne au nœud Diameter qui agit comme client de produire des enregistrements comptables en continu même durant une session. L'AVP Accounting-Realtime-Required est utilisée pour contrôler le comportement du client lorsque le transfert des enregistrements de comptabilité provenant du client Diameter est retardé ou ne réussit pas.

Le serveur de comptabilité Diameter PEUT outrepasser l'intervalle intermédiaire ou les exigences de temps réel en incluant l'AVP Acct-Interim-Interval ou Accounting-Realtime-Required dans le message Accounting-Answer. Lorsque une de ces AVP est présente, la dernière valeur reçue DEVRAIT être utilisée dans les activités comptables à venir pour la même session.

9.2 Messages du protocole

Un nœud Diameter qui reçoit un message d'authentification et/ou autorisation réussie du serveur Diameter DEVRAIT collecter les informations de comptabilité pour la session. Le message Accounting-Request est utilisé pour transmettre les informations de comptabilité au serveur Diameter, qui DOIT répondre par le message Accounting-Answer pour en confirmer la réception. Le message Accounting-Answer inclut l'AVP Result-Code, qui PEUT indiquer qu'une erreur était présente dans le message de comptabilité. La valeur de l'AVP Accounting-Realtime-Required reçue antérieurement pour la session en question peut indiquer que la session de l'usager doit être terminée quand un message Accounting-Request rejeté a été reçu.

9.3 Extension et exigences de l'application de comptabilité

Chaque application Diameter (par exemple, NASREQ, IP mobile) DEVRAIT définir les AVP spécifiques du service qui DOIVENT être présentes dans le message Accounting-Request dans une section intitulée "AVP de comptabilité". L'application DOIT supposer que les AVP décrites dans le présent document seront présentes dans tous les messages de comptabilité, de sorte que seules leurs AVP respectives spécifiques du service doivent être définies dans cette section.

Les applications ont la faculté d'utiliser un ou deux des modèles d'extension d'application de comptabilité suivants :

Service de comptabilité partagé
Le message de comptabilité va porter l'identifiant d'application de l'application de comptabilité Diameter de base (voir au paragraphe 2.4). Les messages de comptabilité peuvent être acheminées aux nœuds Diameter autres que de l'application Diameter correspondante. Ces nœuds peuvent être des serveurs de comptabilité centralisés qui fournissent des services comptables pour plusieurs applications Diameter différentes. Ces nœuds DOIVENT annoncer l'identifiant d'application de comptabilité Diameter de base durant l'échange de capacités.

Service de comptabilité couplé
Le message de comptabilité va porter l'identifiant d'application de l'application qui l'utilise. L'application elle-même va traiter les enregistrements de comptabilité reçus ou les transmettre à un serveur de comptabilité. Aucune annonce d'application de comptabilité n'est exigée durant l'échange de capacités, et les messages de comptabilité vont être acheminés de la même façon que tous les autres messages d'application.

Dans les cas où une application ne définit pas son propre service de comptabilité, il est préférable d'utiliser le modèle de comptabilité partagé.

9.4 Résilience aux fautes

Les mécanismes du protocole de base Diameter sont utilisés pour surmonter les petites pertes de message et les défaillances réseau de nature temporaire.

Les homologues Diameter qui agissent comme clients DOIVENT mettre en œuvre la reprise sur défaillance pour se protéger des défaillances de serveur et de certaines défaillances du réseau. Les homologues Diameter qui agissent comme agents ou les systèmes de traitement hors ligne en rapport DOIVENT détecter les enregistrements de comptabilité dupliqués causés par l'envoi du même enregistrement à plusieurs serveurs et la duplication de messages dans le transit. Cette détection DOIT se fonder sur l'inspection des paires d'AVP Session-Id et Accounting-Record-Number. L'Appendice C expose les besoins de la détection des dupliqués et les problèmes de mise en œuvre.

Les clients Diameter PEUVENT avoir une mémoire non volatile pour la mémorisation sûre des enregistrements de comptabilité à travers les réamorçages ou des défaillances étendues du réseau, des partitions de réseau, et des défaillances de serveur. Si une telle mémoire est disponible, le client DEVRAIT y mémoriser les nouveaux enregistrements de comptabilité aussitôt que les enregistrements sont créés et jusqu'à ce qu'un accusé de réception positif ait été reçu du serveur Diameter. Sur un réamorçage, le client DOIT commencer à envoyer les enregistrements qui sont dans la mémoire non volatile au serveur de comptabilité avec les modifications appropriées en cause de terminaison, longueur de session, et autres informations pertinentes dans les enregistrements.

Une autre application de ce protocole peut inclure les AVP pour contrôler le nombre maximum d'enregistrements de comptabilité qui peuvent être mémorisés chez le client Diameter sans les remiser dans la mémoire non volatile ou les transférer au serveur Diameter.

Le client NE DEVRAIT PAS retirer des données de comptabilité des zones de sa mémoire avant que la réponse correcte Accounting-Answer ait été reçue. Le client PEUT retirer les plus anciennes données de comptabilité, non livrées, ou encore non acquittées si il se trouve à court de ressources comme de mémoire. C'est une question qui dépend de la mise en œuvre que le client accepte de nouvelles sessions dans ces conditions.

9.5 Enregistrements comptables

Dans tous les enregistrements de comptabilité, l'AVP Session-Id DOIT être présente ; l'AVP User-Name DOIT être présente si elle est disponible au client Diameter.

Différents types d'enregistrements de comptabilité sont envoyés selon le type réel du service comptable et les directives du serveur d'autorisation pour la comptabilité intermédiaire. Si le service comptabilisé est un événement unique, ce qui signifie que le début et la fin de l'événement sont simultanés, l'AVP Accounting-Record-Type DOIT alors être présente et réglée à la valeur EVENT_RECORD.

Si le service comptabilisé est une longueur mesurable, l'AVP DOIT alors utiliser les valeurs START_RECORD, STOP_RECORD, et éventuellement, INTERIM_RECORD. Si le serveur d'autorisation n'a pas donné de directives pour activer la comptabilité intermédiaire pour la session, deux enregistrements de comptabilité DOIVENT être générés pour chaque service de type session. Lorsque la demande initiale de comptabilité est envoyée pour une certaine session, l'AVP Accounting-Record-Type DOIT être réglée à la valeur START_RECORD. Lorsque la dernière demande de comptabilité est envoyée, la valeur DOIT être STOP_RECORD.

Si le serveur d'autorisation a donné des directives pour que la comptabilité intermédiaire soit activée, le client Diameter DOIT produire des enregistrements supplémentaires entre le START_RECORD et le STOP_RECORD, marqués INTERIM_RECORD. La production de ces enregistrements est aussi provoquée par Acct-Interim-Interval comme toute réauthentification ou réautorisation de la session. Le client Diameter DOIT écraser tous les enregistrements de comptabilité intermédiaires qui sont mémorisés en local pour livraison, si un nouvel enregistrement est généré pour la même session. Cela assure qu'un seul enregistrement intérimaire en cours peut exister sur un appareil d'accès pour une certaine session.

Une valeur particulière de Accounting-Sub-Session-Id ne DOIT apparaître que dans une séquence d'enregistrements de comptabilité provenant d'un client Diameter, sauf pour les besoins de la retransmission. La séquence qui est envoyée DOIT être un enregistrement avec l'AVP Accounting-Record-Type réglée à la valeur EVENT_RECORD ou plusieurs enregistrements commençant par celui qui a la valeur START_RECORD, suivi par zéro, un ou plusieurs INTERIM_RECORD et un seul STOP_RECORD. Une spécification d'application Diameter particulière DOIT définir le type de séquences qui DOIT être utilisé.

9.6 Corrélation des enregistrements de comptabilité

Si une application utilise des messages de comptabilité, elle peut corréler les enregistrements de comptabilité avec une session spécifique d'application en utilisant l'identifiant de session de la session particulière de l'application dans les messages de comptabilité. Les messages de comptabilité PEUVENT aussi utiliser un identifiant de session différent de celui des sessions de l'application, mais dans ce cas, les autres informations relatives à la session devront effectuer la corrélation.

Dans les cas où une application exige plusieurs sous sessions de comptabilité, un identifiant de sous session Accounting-Sub-AVP est utilisé pour différencier chaque sous session. L'identifiant de session va rester constant pour toutes les sous sessions et sera utilisé pour corréler toutes les sous sessions à une session d'application particulière. Noter que recevoir un STOP_RECORD sans AVP Accounting-Sub-Session-Id lorsque des sous sessions ont été utilisées à l'origine dans les messages START_RECORD implique que toutes les sous sessions sont terminées.

Il y a aussi des cas où une application a besoin de corréler plusieurs sessions d'application dans un seul enregistrement comptable ; l'enregistrement comptable peut s'étendre sur plusieurs applications et sessions Diameter différentes utilisées par le même usager à un moment donné. Dans ce cas, l'AVP Acct-Multi-Session-Id est utilisé. L'AVP Acct-Multi-Session-Id DEVRAIT être signalé par le serveur à l'appareil d'accès (normalement, durant l'autorisation) lorsque il détermine qu'une demande appartient à une session existante. L'appareil d'accès DOIT alors inclure l'AVP Acct-Multi-Session-Id dans tous les messages de comptabilité suivants.

L'AVP Acct-Multi-Session-Id PEUT inclure la valeur de l'identifiant de session original. Son contenu est spécifique de la mise en œuvre, mais il DOIT être globalement unique parmi tous les autres Acct-Multi-Session-Id et NE DOIT PAS changer durant la vie d'une session.

Un document d'application Diameter DOIT définir le concept exact de session qui est l'objet de la comptabilité et il PEUT définir le concept de multisession. Par exemple, l'application NASREQ DIAMETER traite une seule connexion PPP à un serveur d'accès réseau comme une session et un ensemble de sessions PPP multi liaisons comme une multi session.

9.7 Codes de commandes de comptabilité

Cette section définit les valeurs des codes de commande qui DOIVENT être prises en charge par toutes les mises en œuvre Diameter qui fournissent des services de comptabilité.

9.7.1 Accounting-Request

La commande Accounting-Request (ACR) indiquée par le champ Code de commande réglé à 271 et le bit 'R' des fanions de commande établi, est envoyé par un nœud Diameter, agissant comme client, afin d'échanger des informations comptables avec un homologue.

En plus des AVP mentionnées ci-dessous, les messages Accounting-Request DEVRAIENT inclure les AVP de comptabilité spécifiques du service.

Format de message :

```
<ACR> ::= < En-tête Diameter : 271, REQ, PXY >
          < Session-Id >
          { Origin-Host }
          { Origin-Realm }
          { Destination-Realm }
          { Accounting-Record-Type }
          { Accounting-Record-Number }
          [ Acct-Application-Id ]
          [ Vendor-Specific-Application-Id ]
          [ User-Name ]
          [ Destination-Host ]
          [ Accounting-Sub-Session-Id ]
          [ Acct-Session-Id ]
          [ Acct-Multi-Session-Id ]
          [ Acct-Interim-Interval ]
          [ Accounting-Realtime-Required ]
          [ Origin-State-Id ]
          [ Event-Timestamp ]
```

* [Proxy-Info]
* [Route-Record]
* [AVP]

9.7.2 Accounting-Answer

La commande Accounting-Answer (ACA) indiquée par le champ Code de commande réglé à 271 et le bit 'R' des fanions de commande à zéro, est utilisée pour accuser réception d'une commande Accounting-Request. La commande Accounting-Answer contient le même identifiant de session que la demande correspondante.

Seul le serveur Diameter cible, connu comme serveur Diameter de rattachement, DEVRAIT répondre avec la commande Accounting-Answer.

En plus des AVP mentionnées ci-dessous, les messages Accounting-Answer DEVRAIENT inclure les AVP de comptabilité spécifiques du service.

Format de message :

```
<ACA> ::= < En-tête Diameter : 271, PXY >
          < Session-Id >
          { Result-Code }
          { Origin-Host }
          { Origin-Realm }
          { Accounting-Record-Type }
          { Accounting-Record-Number }
          [ Acct-Application-Id ]
          [ Vendor-Specific-Application-Id ]
          [ User-Name ]              [ Accounting-Sub-Session-Id ]
          [ Acct-Session-Id ]
          [ Acct-Multi-Session-Id ]
          [ Error-Message ]
          [ Error-Reporting-Host ]
          [ Failed-AVP ]
          [ Acct-Interim-Interval ]
          [ Accounting-Realtime-Required ]
          [ Origin-State-Id ]
          [ Event-Timestamp ]
        * [ Proxy-Info ]
        * [ AVP ]
```

9.8 AVP de comptabilité

Cette section contient les AVP qui décrivent les informations d'usage comptable relatives à une session spécifique.

9.8.1 AVP Accounting-Record-Type

L'AVP Accounting-Record-Type (code d'AVP : 480) est du type Enumerated et contient le type d'enregistrement comptable envoyé. Les valeurs suivantes sont actuellement définies pour l'AVP Accounting-Record-Type :

EVENT_RECORD : 1. Un enregistrement d'événement comptable est utilisé pour indiquer qu'un événement unique s'est produit (ce qui signifie que le début et la fin de l'événement sont simultanés). Cet enregistrement contient toutes les informations relatives au service, et est le seul enregistrement du service.

START_RECORD : 2. Les enregistrements de début de comptabilité, de comptabilité intermédiaire, et d'arrêt de comptabilité sont utilisés pour indiquer qu'un service de longueur mesurable a été fourni. Un enregistrement de début de comptabilité est utilisé pour initier une session comptable et contient les informations comptables relatives à l'initiation de la session.

INTERIM_RECORD : 3. Un enregistrement de comptabilité intermédiaire contient des informations cumulatives de comptabilité pour une session comptable existante. Les enregistrements de comptabilité intermédiaires DEVRAIENT être envoyés chaque fois qu'une réauthentification ou réautorisation se produit. De plus, des déclanchement d'enregistrement intermédiaires supplémentaires PEUVENT être définis par des applications

Diameter spécifiques. Le choix d'utiliser ou non des enregistrements INTERIM_RECORD est fait avec l'AVP Acct-Interim-Interval.

STOP_RECORD : 4. Un enregistrement d'arrêt de comptabilité est envoyé pour terminer une session de comptabilité et contient des informations comptables cumulatives relatives à la session existante.

9.8.2 AVP Acct-Interim-Interval

L'AVP Acct-Interim-Interval (code d'AVP : 85) est du type Unsigned32 et est envoyée du serveur d'autorisation Diameter de rattachement au client Diameter. Le client utilise les informations de cette AVP pour décider comment et quand produire les enregistrements de comptabilité. Avec les différentes valeurs de cette AVP, les sessions de service peuvent résulter en un, deux, ou deux + N enregistrements de comptabilité, selon les besoins de l'organisation de rattachement. Le comportement de production d'enregistrements de comptabilité suivant est dirigé par l'inclusion de cette AVP :

1. L'omission de l'AVP Acct-Interim-Interval ou son inclusion avec le champ Valeur réglé à 0 signifie que EVENT_RECORD, START_RECORD, et STOP_RECORD sont produits, comme approprié pour le service.

2. L'inclusion de l'AVP avec le champ Valeur réglé à une autre valeur que zéro signifie que les enregistrements INTERIM_RECORD DOIVENT être produits entre les enregistrements START_RECORD et STOP_RECORD. Le champ Valeur de cette AVP est l'intervalle nominal en secondes entre ces enregistrements. Le nœud Diameter qui génère les informations de comptabilité, connu comme "le client", DOIT produire le premier enregistrement INTERIM_RECORD à peu près au moment où cet intervalle nominal s'est écoulé depuis le START_RECORD, le prochain à nouveau lorsque l'intervalle s'est encore écoulé, et ainsi de suite jusqu'à la fin de la session et la production d'un enregistrement STOP_RECORD.

Le client DOIT s'assurer les instants de production des enregistrements intermédiaires sont rendus aléatoires afin que de grosses tempêtes de messages de comptabilité ne soient pas créées avec les enregistrements ou avec un instant commun de début de service.

9.8.3 AVP Accounting-Record-Number

L'AVP Accounting-Record-Number (code d'AVP : 485) est du type Unsigned32 et identifie cet enregistrement au sein d'une session. Comme les AVP Session-Id sont globalement uniques, la combinaison des AVP Session-Id et Accounting-Record-Number est aussi globalement unique et peut être utilisée pour confronter les enregistrements de comptabilité avec les confirmations. Un moyen aisé de produire des nombres uniques est de régler la valeur à 0 pour les enregistrements de type EVENT_RECORD et START_RECORD et de régler la valeur à 1 pour le premier INTERIM_RECORD, à 2 pour le second, et ainsi de suite jusqu'à ce que la valeur pour STOP_RECORD soit un de plus que le dernier INTERIM_RECORD.

9.8.4 AVP Acct-Session-Id

L'AVP Acct-Session-Id (code d'AVP : 44) est du type OctetString n'est utilisée que quand se produit une traduction RADIUS/Diameter. Cette AVP contient le contenu de l'attribut RADIUS Acct-Session-Id.

9.8.5 AVP Acct-Multi-Session-Id

L'AVP Acct-Multi-Session-Id (code d'AVP : 50) est du type UTF8String, suivant le format spécifié au paragraphe 8.8. L'AVP Acct-Multi-Session-Id est utilisée pour lier plusieurs sessions de comptabilité en rapports, où chaque session va avoir un identifiant de session mais la même AVP Acct-Multi-Session-Id. Cette AVP PEUT être retournée par le serveur Diameter dans une réponse d'autorisation, et elle DOIT être utilisée dans tous les messages de comptabilité pour cette session.

9.8.6 AVP Accounting-Sub-Session-Id

L'AVP Accounting-Sub-Session-Id (code d'AVP : 287) est du type Unsigned64 et contient l'identifiant de sous session de comptabilité. La combinaison de l'identifiant de session et de cette AVP DOIT être unique par sous session, et la valeur de cette AVP DOIT être à accroissement monotone de un pour toute nouvelle sous session. L'absence de cette AVP implique qu'aucune sous session n'est utilisée, à l'exception d'une demande de comptabilité dont le Accounting-Record-Type est réglé à STOP_RECORD. Un message STOP_RECORD sans AVP Accounting-Sub-Session-Id présente va signaler la terminaison de toutes les sous sessions pour un certain identifiant de session.

9.8.7 AVP Accounting-Realtime-Required

L'AVP Accounting-Realtime-Required (code d'AVP : 483) est du type Enumerated et est envoyée du serveur d'autorisation Diameter de rattachement au client Diameter ou dans la réponse de comptabilité provenant du serveur de comptabilité. Le client utilise les informations de cette AVP pour décider que faire si l'envoi des enregistrements de comptabilité au serveur de comptabilité a été temporairement empêché à cause, par exemple, d'un problème réseau.

DELIVER_AND_GRANT : 1. L'AVP avec le champ Valeur réglé à DELIVER_AND_GRANT signifie que le service DOIT être accordé seulement si il y a une connexion avec un serveur de comptabilité. Noter que l'ensemble des serveurs de comptabilité de remplacement est traité comme un seul serveur dans ce sens. Avoir à déplacer le flux d'enregistrements comptables sur un serveur de sauvegarde n'est pas une raison pour arrêter le service à l'usager.

GRANT_AND_STORE : 2. L'AVP avec le champ Valeur réglé à GRANT_AND_STORE signifie que le service DEVRAIT être accordé si il y a une connexion, ou tant que les enregistrements peuvent être mémorisés comme décrit au paragraphe 9.4. C'est le comportement par défaut si l'AVP n'est pas incluse dans la réponse du serveur d'autorisation.

GRANT_AND_LOSE : 3. L'AVP avec le champ Valeur réglé à GRANT_AND_LOSE signifie que le service DEVRAIT être accordé même si les enregistrements ne peuvent pas être livrés ou mémorisés.

10. Tableaux d'occurrence des AVP

Les tableaux suivants présentent les AVP définies dans le présent document et spécifient dans quels messages Diameter elles PEUVENT ou NE PEUVENT PAS être présentes. Les AVP qui ne surviennent qu'à l'intérieur d'AVP Grouped ne sont pas montrées dans ces tableaux.

Les tableaux utilisent les symboles suivants :
0 : L'AVP NE DOIT PAS être présente dans le message.
0+ : Zéro, une ou plusieurs instances de l'AVP PEUVENT être présentes dans le message.
0-1 : Zéro, ou une instance de l'AVP PEUT être présente dans le message. C'est une erreur qu'il y ait plus d'une instance de l'AVP.
1 : Une instance de l'AVP DOIT être présente dans le message.
1+ : Au moins une instance de l'AVP DOIT être présente dans le message.

10.1 Tableau des AVP du protocole de commandes de base

Le tableau de ce paragraphe se limite aux codes de commandes non de comptabilité définis dans cette spécification.

Code de commande												
Nom d'attribut	CER	CEA	DPR	DPA	DWR	DWA	RAR	RAA	ASR	ASA	STR	STA
Acct-Interim-Interval	0	0	0	0	0	0	0-1	0	0	0	0	0
Accounting-Realtime-Required	0	0	0	0	0	0	0-1	0	0	0	0	0
Acct-Application-Id	0+	0+	0	0	0	0	0	0	0	0	0	0
Auth-Application-Id	0+	0+	0	0	0	0	1	0	1	0	1	0
Auth-Grace-Period	0	0	0	0	0	0	0	0	0	0	0	0
Auth-Request-Type	0	0	0	0	0	0	0	0	0	0	0	0
Auth-Session-State	0	0	0	0	0	0	0	0	0	0	0	0
Autorisation-Lifetime	0	0	0	0	0	0	0	0	0	0	0	0
Class	0	0	0	0	0	0	0	0	0	0	0+	0+
Destination-Host	0	0	0	0	0	0	1	0	1	0	0-1	0
Destination-Realm	0	0	0	0	0	0	1	0	1	0	1	0
Disconnect-Cause	0	0	1	0	0	0	0	0	0	0	0	0
Error-Message	0	0-1	0	0-1	0	0-1	0	0-1	0	0-1	0	0-1
Error-Reporting-Host	0	0	0	0	0	0	0	0-1	0	0-1	0	0-1
Failed-AVP	0	0-1	0	0-1	0	0-1	0	0-1	0	0-1	0	0-1
Firmware-Revision	|0-1	0-1	0	0	0	0	0	0	0	0	0	0
Host-IP-Address	1+	1+	0	0	0	0	0	0	0	0	0	0
Inband-Security-Id	0	0	0	0	0	0	0	0	0	0	0	0
Multi-Round-Time-Out	0	0	0	0	0	0	0	0	0	0	0	0
Origin-Host	1	1	1	1	1	1	1	1	1	1	1	1

Origin-Realm	1	1	1	1	1	1	1	1	1	1	1	1
Origin-State-Id	0-1	0-1	0	0	0-1	0-1	0-1	0-1	0-1	0-1	0-1	0-1
Product-Name	1	1	0	0	0	0	0	0	0	0	0	0
Proxy-Info	0	0	0	0	0	0	0+	0+	0+	0+	0+	0+
Redirect-Host	0	0	0	0	0	0	0	0+	0	0+	0	0+
Redirect-Host-Usage	0	0	0	0	0	0	0	0-1	0	0-1	0	0-1
Redirect-Max-Cache-Time	0	0	0	0	0	0	0	0-1	0	0-1	0	0-1
Result-Code	0	1	0	1	0	1	0	1	0	1	0	1
Re-Auth-Request-Type	0	0	0	0	0	0	1	0	0	0	0	0
Route-Record	0	0	0	0	0	0	0+	0	0+	0	0+	0
Session-Binding	0	0	0	0	0	0	0	0	0	0	0	0
Session-Id	0	0	0	0	0	0	1	1	1	1	1	1
Session-Server-failback	0	0	0	0	0	0	0	0	0	0	0	0
Session-Timeout	0	0	0	0	0	0	0	0	0	0	0	0
Supported-Vendor-Id	0+	0+	|0	0	0	0	0	0	0	0	0	0
Termination-Cause	0	0	0	0	0	0	0	0	0	0	1	0
User-Name	0	0	0	0	0	0	0-1	0-1	0-1	0-1	0-1	0-1
Vendor-Id	1	1	0	0	0	0	0	0	0	0	0	0
Vendor-Specific-Appli.-Id	0+	0+	0	0	0	0	0	0	0	0	0	0

10.2 Tableau des AVP de comptabilité

Le tableau de ce paragraphe est utilisé pour représenter quelles AVP définies dans ce document doivent être présentes dans les messages de comptabilité. Ces exigences d'occurrence des AVP sont des lignes directrices, qui peuvent être étendues, et/ou outrepassées par des exigences spécifiques de l'application dans les documents d'applications Diameter.

Code de commande		
Nom d'attribut	**ACR**	**ACA**
Acct-Interim-Interval	0-1	0-1
Acct-Multi-Session-Id	0-1	0-1
Accounting-Record-Number	1	1
Accounting-Record-Type	1	1
Acct-Session-Id	0-1	0-1
Accounting-Sub-Session-Id	0-1	0-1
Accounting-Realtime-Required	0-1	0-1
Acct-Application-Id	0-1	0-1
Auth-Application-Id	0	0
Class	0+	0+
Destination-Host	0-1	0
Destination-Realm	1	0
Error-Reporting-Host	0	0+
Event-Timestamp	0-1	0-1
Failed-AVP	0	0-1
Origin-Host	1	1
Origin-Realm	1	1
Proxy-Info	0+	0+
Route-Record	0+	0
Result-Code	0	1
Session-Id	1	1
Termination-Cause	0	0
User-Name	0-1	0-1
Vendor-Specific-Application-Id	0-1	0-1

11 Considérations relatives à l'IANA

Cette section donne des directives à l'Autorité d'allocation des numéros de l'Internet (IANA, *Internet Assigned Numbers Authority*) concernant l'enregistrement des valeurs relatives au protocole Diameter, conformément à la [RFC5226]. Les registres et allocations existants de l'IANA mis en place par la RFC 3588 restent les mêmes sauf explicitement mis à jour ou déconseillés dans cette section.

11.1 En-tête d'AVP

Comme définit dans la Section 4, l'en-tête d'AVP contient trois champs qui requièrent la gestion d'espace de noms de l'IANA : le code d'AVP, l'identifiant de fabricant, et les champs de fanions.

11.1.1 Codes d'AVP

Il y a plusieurs espaces de noms. Les fabricants peuvent avoir leur propre espèce de noms de code d'AVP qui sera identifié par leur identifiant de fabricant (aussi appelé numéro d'entreprise) et ils contrôlent l'allocation de leurs codes d'AVP spécifiques du fabricant au sein de leur propre espace de noms. L'absence d'un identifiant de fabricant ou une valeur de Vendor-ID de zéro (0) identifie l'espace de noms de code d'AVP de l'IETF, qui est sous le contrôle de l'IANA. Le code d'AVP et parfois de possibles valeurs dans une AVP sont contrôlés et tenus par l'IANA. Le code d'AVP 0 n'est pas utilisé. Les codes d'AVP de 1 à 255 sont gérés séparément comme types d'attributs RADIUS. Lorsque une AVP spécifique de fabricant est mise en œuvre par plus d'un fabricant, l'allocation d'AVP globales devrait être plutôt encouragée.

Les AVP peuvent être allouées suivant revue d'expert (par un expert désigné) avec spécification exigée [RFC5226]. Une allocation de bloc (livraison de plus de trois AVP à un certain moment pour un objet donné) exige la revue par l'IETF [RFC5226].

11.1.2 Fanions d'AVP

Le paragraphe 4.1 décrit les fanions d'AVP existants. Les bits restants peuvent seulement être alloués via une action de normalisation [RFC5226].

11.2 En-tête Diameter

11.2.1 Codes de commandes

Pour l'en-tête Diameter, l'allocation d'espace de noms de code de commande a changé. Les nouvelles règles d'allocation sont les suivantes :

Les valeurs de code de commande 256 à 8 388 607 (0x100 à 0x7fffff) sont pour des commandes permanentes, normalisées, allouées par revue de l'IETF[RFC5226].

Les valeurs 8 388 608 à 16 777 213 (0x800000 à 0xfffffd) sont réservées pour les codes de commande spécifiques de fabricant, pour être allouées sur la base du premier arrivé, premier servi par l'IANA [RFC5226]. La demande à l'IANA pour un code de commande spécifique de fabricant DEVRAIT inclure une référence à une spécification publiquement disponible qui documente la commande en détail suffisant pour aider à l'interopérabilité entre des mises en œuvre indépendantes. Si la spécification ne peut pas être rendue publiquement disponible, la demande de code de commande spécifique de fabricant DOIT inclure les informations de contact des personnes et/ou entités responsables des droits d'auteur et de la maintenance de la commande.

Les valeurs 16 777 214 et 16 777 215 (valeurs hexadécimales 0xfffffe - 0xffffff) sont réservées pour des commandes expérimentales. Comme ces codes sont seulement à des fins expérimentales et d'essai, aucune garantie n'est donnée d'interopérabilité entre homologues Diameter qui utilisent des commandes expérimentales.

11.2.2 Fanions de commandes

La Section 3 décrit les champs existants de fanions de commandes. Les bits restants ne peuvent être alloués que via une action de normalisation [RFC5226].

11.3 Valeurs d'AVP

Pour les valeurs d'AVP, l'allocation de la valeur d'AVP Experimental-Result-Code a été ajoutée; voir au paragraphe 11.3.1. L'ancienne règle d'allocation de valeur d'AVP, consensus de l'IETF, a été mise à jour en revue de l'IETF selon la [RFC5226], et les AVP affectées sont mentionnées pour mémoire.

11.3.1 AVP Experimental-Result-Code

Les valeurs pour cette AVP sont purement locales pour le fabricant indiqué, et aucun registre IANA n'est tenu pour elles.

11.3.2 Valeurs d'AVP Result-Code

De nouvelles valeurs sont disponibles pour être allouées via revue par l'IETF [RFC5226].

11.3.3 Valeurs d'AVP Accounting-Record-Type

De nouvelles valeurs sont disponibles pour être allouées via revue par l'IETF [RFC5226].

11.3.4 Valeurs d'AVP Termination-Cause

De nouvelles valeurs sont disponibles pour être allouées via revue par l'IETF [RFC5226].

11.3.5 Valeurs d'AVP Redirect-Host-Usage

De nouvelles valeurs sont disponibles pour être allouées via revue par l'IETF [RFC5226].

11.3.6 Valeurs d'AVP Session-Server-Failover

De nouvelles valeurs sont disponibles pour être allouées via revue par l'IETF [RFC5226].

11.3.7 Valeurs d'AVP Session-Binding

De nouvelles valeurs sont disponibles pour être allouées via revue par l'IETF [RFC5226].

11.3.8 Valeurs d'AVP Disconnect-Cause

De nouvelles valeurs sont disponibles pour être allouées via revue par l'IETF [RFC5226].

11.3.9 Valeurs d'AVP Auth-Request-Type

De nouvelles valeurs sont disponibles pour être allouées via revue par l'IETF [RFC5226].

11.3.10 Valeurs d'AVP Auth-Session-State

De nouvelles valeurs sont disponibles pour être allouées via revue par l'IETF [RFC5226].

11.3.11 Valeurs d'AVP Re-Auth-Request-Type

De nouvelles valeurs sont disponibles pour être allouées via revue par l'IETF [RFC5226].

11.3.12 Valeurs d'AVP Accounting-Realtime-Required

De nouvelles valeurs sont disponibles pour être allouées via revue par l'IETF [RFC5226].

11.3.13 AVP Inband-Security-Id (code 299)

L'utilisation de cette AVP a été déconseillée.

11.4 Nom de service et enregistrement de numéro d'accès _diameters

L'IANA a enregistré le nom de service "_diameters" et a alloué les numéros d'accès pour TLS/TCP et DTLS/SCTP conformément aux lignes directrices données dans la [RFC6335].

Nom de service : _diameters
Protocoles de transport : TCP, SCTP
Syndic : IESG <iesg@ietf.org>
Contact : Président de l'IETF <chair@ietf.org>

Description : Diameter sur TLS/TCP et DTLS/SCTP
Référence : RFC 6733
Numéro d'accès : 5868, de la gamme d'usager

11.5 Identifiants de protocole de charge utile SCTP

Deux identifiant s de protocole de charge utile SCTP ont été enregistrés dans le registre des identifiants de protocole de charge utile SCTP :

Valeur	Identifiant de protocole de charge utile SCTP
46	Diameter dans un tronçon DATA SCTP
47	Diameter dans un tronçon DATA DTLS/SCTP

11.6 Paramètres S-NAPTR

L'étiquette suivante a été enregistrée dans le registre des étiquettes de protocole d'application S-NAPTR :

Étiquette	Protocole
diameter.dtls.sctp	DTLS/SCTP

12. Paramètres Diameter configurables en relation avec le protocole

Cette section contient les paramètres configurables qui se trouvent dans le présent document :

Homologue Diameter
Une entité Diameter PEUT communiquer avec des homologues qui sont statiquement configurés. Un homologue Diameter configuré statiquement va exiger que soit fournie l'adresse IP ou le nom de domaine pleinement qualifié (FQDN) qui sera alors utilisé pour la résolution à travers le DNS.

Tableau d'acheminements
Un serveur Diameter mandataire achemine les messages sur la base de la portion domaine d'un identifiant d'accès réseau (NAI). Le serveur DOIT avoir un tableau des noms de domaines, et l'adresse de l'homologue auquel le message doit être transmis. Le tableau d'acheminements PEUT aussi inclure un "chemin par défaut", qui est normalement utilisé pour tous les messages qui ne peuvent pas être traités en local.

Temporisateur Tc
Le temporisateur Tc contrôle la fréquence à laquelle la connexion de transport tente de joindre un homologue avec lequel aucune connexion de transport active n'existe. La valeur recommandée est 30 secondes.

13. Considérations sur la sécurité

Les messages du protocole de base Diameter DEVRAIENT être sécurisés en utilisant TLS [RFC5246] ou DTLS/SCTP [RFC6083]. Des mécanismes de sécurité supplémentaires comme IPsec [RFC4301] PEUVENT aussi être déployés pour sécuriser les connexions entre homologues. Cependant, toutes les mises en œuvre du protocole de base Diameter DOIVENT prendre en charge l'utilisation de TLS/TCP et DTLS/SCTP, et le protocole Diameter NE DOIT PAS être utilisé sans TLS, DTLS, ou IPsec.

Si une connexion Diameter doit être protégée via TLS/TCP et DTLS/SCTP ou IPsec, alors TLS/TCP et DTLS/SCTP ou IPsec/IKE DEVRAIENT commencer avant tout échange de message Diameter. Tous les paramètres de sécurité pour TLS/TCP et DTLS/SCTP ou IPsec sont configurés indépendamment du protocole Diameter. Tous les messages Diameter seront envoyés à travers une connexion TLS/TCP et DTLS/SCTP ou IPsec après la réussite de l'établissement.

Pour que les connexions TLS/TCP et DTLS/SCTP soient établies dans l'état ouvert, l'échange CER/CEA DOIT inclure une AVP Inband-Security-ID avec une valeur de TLS/TCP et DTLS/SCTP. La prise de contact TLS/TCP et DTLS/SCTP commencera lorsque les deux extrémités auront réussi à atteindre l'état ouvert, après l'achèvement de l'échange CER/CEA. Si la prise de contact TLS/TCP et DTLS/SCTP est réussie, tous les messages suivants seront envoyés via TLS/TCP et DTLS/SCTP. Si la prise de contact échoue, les deux extrémités DOIVENT passer à l'état fermé. Voir au paragraphe 13.1 pour les détails.

13.1 Utilisation TLS/TCP et DTLS/SCTP

Les nœuds Diameter qui utilisent TLS/TCP et DTLS/SCTP pour la sécurité DOIVENT s'authentifier mutuellement au titre de l'établissement de la session TLS/TCP et DTLS/SCTP. Afin d'assurer l'authentification mutuelle, le nœud Diameter qui agit comme serveur TLS/TCP et DTLS/SCTP DOIT demander un certificat au nœud Diameter qui agit comme client TLS/TCP et DTLS/SCTP, et le nœud Diameter qui agit comme client TLS/TCP et DTLS/SCTP DOIT être prêt à fournir un certificat à la demande.

Les nœuds Diameter DOIVENT être capables de négocier les suites de chiffrement TLS/TCP et DTLS/SCTP suivantes :

TLS_RSA_WITH_RC4_128_MD5
TLS_RSA_WITH_RC4_128_SHA
TLS_RSA_WITH_3DES_EDE_CBC_SHA

Les nœuds Diameter DEVRAIENT être capables de négocier la suite de chiffrement TLS/TCP et DTLS/SCTP suivante :

TLS_RSA_WITH_AES_128_CBC_SHA

Noter qu'il est possible que la prise en charge de la suite de chiffrement TLS_RSA_WITH_AES_128_CBC_SHA soit EXIGÉE à l'avenir. Les nœuds Diameter PEUVENT négocier d'autres suites de chiffrement TLS/TCP et DTLS/ SCTP.

Si des certificats de clé publique sont utilisés pour la sécurité de Diameter (par exemple, avec TLS) la valeur des heures d'expiration dans les tableaux d'acheminement et d'homologues NE DOIT PAS être supérieure à l'heure d'expiration dans les certificats qui s'y rapportent.

13.2 Considérations d'homologue à homologue

Comme avec tout protocole d'homologue à homologue, une configuration appropriée du modèle de confiance au sein d'un homologue Diameter est essentielle pour la sécurité. Lorsque des certificats sont utilisés, il est nécessaire de configurer les autorités de certification (CA) racine de confiance pour l'homologue Diameter. Ces CA racines seront vraisemblablement uniques pour l'usage de Diameter et distinctes des CA racines qui peuvent être de confiance pour d'autres objets tels que la navigation sur la Toile. En général, on s'attend à ce que ces CA racines soient configurées de façon à refléter les relations d'affaire entre l'organisation qui héberge l'homologue Diameter et les autres organisations. Par suite, un homologue Diameter ne va normalement pas être configuré à permettre la connexité avec un homologue arbitraire. Avec l'authentification de certificat, les homologues Diameter peuvent n'être pas connus à l'avance et donc la découverte de l'homologue peut être requise.

13.3 Considérations sur les AVP

Les AVP Diameter contiennent souvent des données sensibles à la sécurité ; par exemple, des mots de passe d'utilisateur et des données de localisation, des adresses réseau et des clés cryptographiques. Les AVP suivantes définies dans ce document sont considérées comme sensibles pour la sécurité :

- Acct-Interim-Interval
- Accounting-Realtime-Required
- Acct-Multi-Session-Id
- Accounting-Record-Number
- Accounting-Record-Type
- Accounting-Session-Id
- Accounting-Sub-Session-Id
- Class
- Session-Id
- Session-Binding
- Session-Server-Failover
- User-Name

Les messages Diameter qui contiennent ces AVP ou toute autre considérée comme étant sensible pour la sécurité DOIVENT n'être envoyés que protégés via TLS ou IPsec mutuellement authentifié. De plus, ces messages NE DOIVENT PAS être envoyés via des nœuds intermédiaires sauf si il y a la sécurité de bout en bout entre le générateur et le receveur ou si le générateur a une configuration locale de confiance qui indique que la sécurité de bout en bout n'est pas nécessaire. Par exemple, la sécurité de bout en bout peut n'être pas requise dans le cas où un nœud intermédiaire est connu pour fonctionner au titre du même domaine administratif que les points d'extrémité de sorte que la capacité de réussir à compromettre l'intermédiaire impliquerait une forte probabilité d'être capable de compromettre aussi les points d'extrémité. Noter qu'aucun mécanisme de sécurité de bout en bout n'est spécifié dans le présent document.

14. Références

14.1 Références normatives

[FLOATPOINT] Institute of Electrical et Electronics Engineers, "IEEE Standard for Binary Floating-Point Arithmetic, ANSI/IEEE Standard 754-1985", août 1985.

[IANAADFAM] IANA, "Address Family Numbers", < http://www.iana.org/assignments/address-family-numbers >.

[RFC0791] J. Postel, éd., "Protocole Internet - Spécification du protocole du programme Internet", STD 5, septembre 1981.

[RFC0793] J. Postel (éd.), "Protocole de commande de transmission – Spécification du protocole du programme Internet DARPA", STD 7, septembre 1981.

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

[RFC3492] A. Costello, "Punycode : Codage Bootstring d'Unicode pour les noms de domaine internationalisés dans les applications (IDNA)", mars 2003. *(P.S.)*

[RFC3539] B. Aboba, J. Wood, "Profil de transport d'authentification, d'autorisation et de comptabilité (AAA)", juin 2003. *(P.S.)*

[RFC3629] F. Yergeau, "UTF-8, un format de transformation de la norme ISO 10646", STD 63, novembre 2003.

[RFC3958] L. Daigle, A. Newton, "Localisation de service d'application fondée sur le domaine avec les enregistrements de ressource de SRV et le service de recherche dynamique de délégation (DDDS)", janvier 2005. *(P.S.)*

[RFC3986] T. Berners-Lee, R. Fielding et L. Masinter, "Identifiant de ressource uniforme (URI) : Syntaxe générique", STD 66, janvier 2005.

[RFC4004] P. Calhoun et autres, "Application IPv4 mobile Diameter", août 2005. *(P.S.)*

[RFC4004] P. Calhoun et autres, "Application IPv4 mobile Diameter", août 2005. *(P.S.)*

[RFC4006] H. Hakala et autres, "Application Diameter de contrôle de crédit", août 2005. *(P.S.)*

[RFC4086] D. Eastlake 3rd, J. Schiller, S. Crocker, "Exigences d'aléa pour la sécurité", juin 2005. (*Remplace* RFC1750) (BCP0106)

[RFC4282] B. Aboba et autres, "L'identifiant d'accès réseau", décembre 2005. (*Remplace* RFC2486) *(P.S.)*

[RFC4291] R. Hinden, S. Deering, "Architecture d'adressage IP version 6", février 2006. (*MàJ par* 5952 et 6052) *(D.S.)*

[RFC4960] R. Stewart, éd., "Protocole de transmission de commandes de flux", septembre 2007. (*Remplace* RFC2960, RFC3309) *(P.S.)*

[RFC5226] T. Narten et H. Alvestrand, "Lignes directrices pour la rédaction d'une section Considérations relatives à l'IANA dans les RFC", BCP 26, mai 2008. (*Remplace* RFC2434)

[RFC5234] D. Crocker, éd., P. Overell, "BNF augmenté pour les spécifications de syntaxe : ABNF", janvier 2008. (STD0068)

[RFC5246] T. Dierks, E. Rescorla, "Version 1.2 du protocole de sécurité de la couche Transport (TLS)", août 2008. (*Remplace* RFC3268, RFC4346, RFC4366) (*MàJ* RFC4492) (*MàJ par* RFC5746, RFC5878)*(P.S.)*

[RFC5280] D. Cooper et autres, "Profil de certificat d'infrastructure de clé publique X.509 et de liste de révocation de certificat (CRL) pour l'Internet", mai 2008. (*Remplace les* RFC3280, RFC4325, RFC4630) *(P.S.)*

[RFC5729] J. Korhonen, éd., M. Jones, L. Morand, T. Tsou, "Précisions sur l'acheminement des demandes Diameter sur la base du nom d'utilisateur et du domaine", décembre 2009, *(P. S.)*

[RFC5890] J. Klensin, "Noms de domaine internationalisés pour les applications (IDNA) : Définitions et cadre documentaire", août 2010. *(Remplace RFC3490) (P.S.)*

[RFC5891] J. Klensin, "Noms de domaine internationalisés pour les applications (IDNA) : Le protocole", août 2010. (*Remplace* RFC3490, RFC3491) (*MàJ* RFC3492) *(P.S.)*

[RFC6083] M. Tyexen, R. Seggelmann et E. Rescorla, "Sécurité de la couche Transport de datagrammes (DTLS) pour le protocle de transmission de contrôle de flux (SCTP)", janvier 2011. *(P.S.)*

[RFC6347] E. Rescorla, N. Modadugu, "Sécurité de la couche transport de datagrammes, version 1.2", janvier 2012. *(Remplace la* RFC4347) *(P.S.)*

[RFC6408] M. Jones, J. Korhonen, L. Morand, "Utilisation du pointeur d'autorité de dénomination directe Diameter (S-NAPTR)", novembre 2011. (*MàJ la* RFC3588) *(P.S.)*

14.2 Références pour information

[ENTERPRISE] IANA, "SMI Network Management Private Enterprise Codes", <http://www.iana.org/assignments/enterprise-numbers>.

[IANATCV] IANA, "Termination-Cause AVP Values (code 295)", <http://www.iana.org/assignments/aaa-parameters/aaa-parameters.xml#aaa-parameters-16>.

[RFC1492] C. Finseth, "Un protocole de contrôle d'accès , parfois appelé TACACS", juillet 1993. *(Information)*

[RFC1661] W. Simpson, éditeur, "Protocole point à point (PPP)", STD 51, juillet 1994. *(MàJ par la* RFC2153)

[RFC2104] H. Krawczyk, M. Bellare et R. Canetti, "HMAC : Hachage de clés pour l'authentification de message", février 1997.

[RFC2782] A. Gulbrandsen, P. Vixie et L. Esibov, "Enregistrement de ressource DNS pour la spécification de la localisation des services (DNS SRV)", février 2000.

[RFC2865] C. Rigney et autres, "Service d'authentification à distance de l'utilisateur appelant (RADIUS)", juin 2000. (*MàJ par* RFC2868, RFC3575, RFC5080) *(D.S.)*

[RFC2866] C. Rigney, "Comptabilité RADIUS", juin 2000. (*MàJ par* RFC2867, RFC5080) *(Information)*

[RFC2869] C. Rigney, W. Willats, P. Calhoun, "Extensions à RADIUS", juin 2000. (*MàJ par* RFC3579, RFC5080) *(Information)*

[RFC2881] D. Mitton, M. Beadles, "Exigences pour la prochaine génération de serveur d'accès réseau (NASREQNG) – modèle de NAS", juillet 2000. *(Information)*

[RFC2975] B. Aboba, J. Arkko, D. Harrington, "Introduction à la gestion comptable", octobre 2000. *(Information)*

[RFC2989] B. Aboba et autres, "Critères d'évaluation des protocoles AAA pour l'accès réseau",novembre 2000. *(Info.)*

[RFC3162] B. Aboba, G. Zorn, D. Mitton, "RADIUS et IPv6", août 2001. *(P.S.)*

[RFC3748] B. Aboba et autres, "Protocole extensible d'authentification (EAP)", juin 2004. *(P.S., MàJ par* RFC5247)

[RFC4301] S. Kent et K. Seo, "Architecture de sécurité pour le protocole Internet", décembre 2005. *(P.S.) (Remplace la* RFC2401)

[RFC4690] J. Klensin et autres, "Révisons et recommandations pour les noms de domaines internationalisés (IDN)", septembre 2006. *(Information)*

[RFC5176] M. Chiba et autres, "Extensions d'autorisation dynamique au service d'authentification distante d'utilisateur appelant (RADIUS)", janvier 2008. (*Remplace* RFC3576) *(Information)*

[RFC5461] F. Gont, "Réaction de TCP aux erreurs logicielles", février 2009. *(Information)*

[RFC5905] D. Mills, J. Martin, J. Burbank, W. Kasch, "Protocole de l'heure du réseau version 4 (NTPv4) : Spécification du protocole et des algorithmes ", juin 2010. (*Remplace* RFC1305, RFC4330). *(P. S.)*

[RFC5927] F. Gont, "Attaques ICMP contre TCP", juillet 2010. *(Information)*

[RFC6335] M. Cotton et autres, "Procédures de l'autorité d'allocation des numéros de l'Internet (IANA) pour la gestion du registre des numéros d'accès aux noms de service et protocoles de transport", août 2011. *(MàJ les* RFC2780, RFC2782, RFC3828, RFC4340, RFC4960, RFC5595) (BCP0165)

[RFC6737] K. Jiao, G. Zorn, "Application de mise à jour de capacités Diameter", octobre 2012. *(P.S.)*

Appendice A. Remerciements

A.1. Pour le présent document

Les auteurs tiennent à remercier les personnes suivantes qui ont fourni des propositions et contributions au document :
À Vishnu Ram et Satendra Gera pour leurs contributions sur la mise à jour des capacités, l'évitement de boucle prédictive, ainsi que de nombreuses autres propositions techniques. À Tolga Asveren pour ses conseils et contributions sur presque toutes les solutions proposées incorporées dans le document. À Timothy Smith pour son aide sur la mise à jour de capacités et autres sujets. À Tony Zhang qui a fourni des remèdes aux créneaux sur la composition de Failed-AVP ainsi que sur de nombreux autres problèmes et sujets. À Jan Nordqvist qui a formulé clairement l'usage des identifiants d'application. À Anders Kristensen qui a fourni des opinions techniques nécessaires. À David Frascone qui a assuré une relecture précieuse du document. À Mark Jones qui a fourni un texte précisant les codes de commandes de fabricant et d'autres indicateurs spécifiques de fabricant. À Victor Pascual et Sebastien Decugis pour de nouveaux textes et recommandations sur SCTP/DTLS. À Jouni Korhonen pour avoir pris en charge les tâches d'édition et résolu les derniers morceaux des versions 27 à 29.

Des remerciements particuliers à l'équipe de conception de l'extensibilité Diameter qui a aidé à résoudre la délicate question des AVP obligatoires et de la sémantique ABNF. Les membres de cette équipe étaient Avi Lior, Jari Arkko, Glen Zorn, Lionel Morand, Mark Jones, Tolga Asveren, Jouni Korhonen, et Glenn McGregor. Des remerciements particuliers aussi à ceux qui ont fourni de précieux commentaires et apports en particulier pour résoudre des questions controversées : Glen Zorn, Yoshihiro Ohba, Marco Stura, Stephen Farrel, Pete Resnick, Peter Saint-Andre, Robert Sparks, Krishna Prasad, Sean Turner, Barry Leiba, et Pasi Eronen.

Finalement, nous tenons à remercier les auteurs d'origine de ce document : Pat Calhoun, John Loughney, Jari Arkko, Erik Guttman, et Glen Zorn. Leurs précieuses connaissances et leur expérience nous ont donné un protocole AAA robuste et souple dont beaucoup ont reconnu la grande valeur en l'adoptant. Nous saluons leur soutien et leur prise en charge de la poursuite de l'amélioration du protocole Diameter. Un grand merci également à tous ceux qui, à côté des auteurs ont assisté et contribué à la version originale de ce document. Leurs efforts ont contribué de façon significative au succès de Diameter.

A.2 La RFC 3588

Les auteurs tiennent à remercier Nenad Trifunovic, Tony Johansson et Pankaj Patel de leur participation au groupe de lecture de documents pré IETF. Allison Mankin, Jonathan Wood, et Bernard Aboba ont fourni une assistance précieuse par leur travail sur les questions de transport et cela a aussi été le cas avec Steven Bellovin dans le domaine de la sécurité.

Paul Funk et David Mitton ont permis de corriger l'automate à états d'homologues, et nous les remercions chaleureusement d'avoir donné de leur temps pour cette tâche. Du texte de ce document a aussi été fourni par Paul Funk, Mark Eklund, Mark Jones, et Dave Spence. Jacques Caron a fourni un grand nombre de commentaires suite à une relecture attentive de la spécification.

Les auteurs remercient aussi les personnes suivantes de leur contribution au développement du protocole Diameter : Allan C. Rubens, Haseeb Akhtar, William Bulley, Stephen Farrell, David Frascone, Daniel C. Fox, Lol Grant, Ignacio Goyret, Nancy Greene, Peter Heitman, Fredrik Johansson, Mark Jones, Martin Julien, Bob Kopacz, Paul Krumviede, Fergal Ladley, Ryan Moats, Victor Muslin, Kenneth Peirce, John Schnizlein, Sumit Vakil, John R. Vollbrecht, et Jeff Weisberg.

Finalement, Pat Calhoun tient à remercier Sun Microsystems car la plus grande partie du temps qu'il a consacré à ce document l'a été pendant qu'il en était salarié.

Appendice B. Exemple de S-NAPTR

Considérons un client qui souhaite résoudre "aaa: ex1.exemple.com". Le client effectue une interrogation NAPTR sur ce domaine, et les enregistrements NAPTR suivants sont retournés :

;;	ordre	préf.	fanions	service	regexp	de remplacement
IN NAPTR	50	50	"s"	"aaa:diameter.tls.tcp"	""	_diameter._tls.ex1.exemple.com
IN NAPTR	100	50	"s"	"aaa:diameter.tcp"	""	_aaa._tcp.ex1.exemple.com
IN NAPTR	150	50	"s"	"aaa:diameter.sctp"	""	_diameter._sctp.ex1.exemple.com

Cela indique que le serveur prend en charge TLS, TCP, et SCTP dans cet ordre. Si le client prend en charge TLS, TLS sera utilisé, ciblé sur un hôte déterminé par une recherche de SRV de _diameter._tls.ex1.exemple.com. Cette recherche va retourner :

;;	Priorité	Poids	Accès	Cible
IN SRV	0	1	5060	server1.ex1.exemple.com
IN SRV	0	2	5060	server2.ex1.exemple.com

Dans un autre exemple, un client souhaite résoudre "aaa: ex2.exemple.com". Le client effectue une interrogation NAPTR pour ce domaine, et les enregistrements NAPTR suivants sont retournés :

;;	ordre	préf.	fanions	service	regexp	de remplacement
IN NAPTR	150	50	"a"	"aaa:diameter.tls.tcp"	""	server1.ex2.exemple.com
IN NAPTR	150	50	"a"	"aaa:diameter.tls.tcp"	""	server2.ex2.exemple.com

Cela indique que le serveur prend en charge TCP disponible dans les noms d'hôte retournés.

Appendice C. Détection des dupliqués

Comme décrit dans la Section 9.4, la détection des enregistrements comptables dupliqués se fonde sur les identifiants de session. Des dupliqués peuvent apparaître pour diverses raisons :

- Reprise sur défaillance sur un serveur de remplacement. Lorsque sont exigées des performances proches du temps réel, les seuils de reprise sur défaillance doivent être bas. Cela peut conduire à une probable augmentation du nombre de dupliqués. La reprise sur défaillance se produit chez le client ou chez les agents Diameter.

- La défaillance d'un client ou agent après l'envoi d'un enregistrement issu de la mémoire non volatile, mais avant la réception d'un ACK de couche application et la suppression de l'enregistrement à envoyer. Il va en résulter la retransmission de l'enregistrement peu après que le client ou agent réamorce.

- Les dupliqués reçus de passerelles RADIUS. Comme le comportement de retransmission de RADIUS n'est pas défini dans la [RFC2865], la probabilité de duplication va varier selon la mise en œuvre.

- Problèmes de mise en œuvre et mauvaise configuration.

Le fanion T est utilisé comme indication d'un événement de retransmission de couche application, par exemple, dû à la reprise sur défaillance sur un serveur de remplacement. Il est défini pour les seuls messages de demande envoyés par les clients ou agents Diameter. Par exemple, après un réamorçage, un client ne peut pas savoir si il a déjà essayé d'envoyer les enregistrements de comptabilité dans sa mémoire non volatile avant que se produise le réamorçage. Les serveurs Diameter PEUVENT utiliser le fanion T comme aide lors du traitement des demandes et la détection de messages dupliqués. Cependant, les serveurs qui font cela DOIVENT s'assurer que les dupliqués sont trouvés même lorsque la première demande transmise arrive au serveur après la demande retransmise. Cela ne peut être utilisé que dans les cas où aucune réponse n'a été reçue du serveur pour une demande et la demande est envoyée à nouveau (par exemple, à cause d'une reprise sur défaillance sur un homologue de remplacement, à cause de la récupération d'un homologue principal ou parce qu'un client envoie à nouveau un enregistrement mémorisé dans une mémoire non volatile après le réamorçage du client ou agent).

Dans certains cas, le serveur de comptabilité Diameter peut retarder la détection de dupliqués et le traitement de

l'enregistrement comptable jusqu'à ce qu'une phase de post traitement ait lieu. Au moment où les enregistrements vont probablement être triés selon les User-Name inclus l'élimination des dupliqués est alors facile. Dans d'autres situations, il peut être nécessaire d'effectuer une détection des dupliqués en temps réel, comme lorsque des limites de crédit sont imposées ou qu'est désirée une détection de fraude en temps réel.

En général, seule la génération de dupliqués dus à la reprise sur défaillance ou au ré envoi d'enregistrements de la mémoire non volatile peut être détectée de façon fiable par les clients ou agents Diameter. Dans ces cas, les clients ou agents Diameter peuvent marquer le message comme possible dupliqué en établissant le fanion T. Comme le serveur Diameter est chargé de la détection des dupliqués, il peut choisir d'utiliser ou non le fanion T, afin d'optimiser la détection des dupliqués. Comme le fanion T n'affecte pas l'interopérabilité, et qu'il peut n'être pas nécessaire pour certains serveurs, la génération du fanion T est EXIGÉE pour les clients et agents Diameter, mais elle PEUT être mise en œuvre par les serveurs Diameter.

Par exemple, on peut généralement supposer que les dupliqués apparaissent dans une fenêtre de temps de plus longue partition de réseau enregistrée ou de faute d'appareil, peut-être d'un jour. De sorte que seuls les enregistrements qui sont dans cette fenêtre doivent être examinés dans la direction vers l'amont. Ensuite, les techniques de hachage ou autres schémas, tels que l'utilisation du fanion T dans les messages reçus, peuvent être utilisées pour éliminer le besoin d'effectuer une recherche complète même dans cet ensemble sauf quelques cas rares.

Voici un exemple de la façon dont le fanion T peut être utilisé par le serveur pour détecter les demandes dupliquées.

Un serveur Diameter PEUT vérifier le fanion T du message reçu pour déterminer si l'enregistrement est un dupliqué possible. Si le fanion T est établi dans le message de demande, le serveur cherche si il y a un dupliqué dans une fenêtre temporelle de duplication configurable vers l'avant et vers l'arrière. Cela limite la recherche dans la base de données aux enregistrements dont le fanion T est établi. Dans un réseau bien géré, les partitions de réseau et les fautes d'appareils seront vraisemblablement des événements rares, de sorte que cette approche représente une optimisation substantielle du processus de détection des dupliqués. Durant la reprise sur défaillance, il est possible que l'enregistrement original soit reçu après celui marqué du fanion T, dû aux différences de délais de réseau rencontrés le long du chemin par les transmissions originale et dupliquée. La probabilité que cela arrive augmente avec la diminution de l'intervalle de reprise sur défaillance. Afin d'être capable de détecter les dupliqués qui sont déclassés, le serveur Diameter devrait utiliser des fenêtres temporelles vers l'avant et vers l'arrière lorsque il effectue la recherche de dupliqués avec les demandes marqués du fanion T. Par exemple, afin de donner le temps à l'enregistrement original de sortir du réseau et être enregistré par le serveur de comptabilité, le serveur Diameter peut retarder le traitement des enregistrements qui ont le fanion T établi pendant un délai de TIME_WAIT + RECORD_PROCESSING_TIME après la clôture de la connexion de transport originale. Après la fin de ce délai, il peut confronter les enregistrements marqués du fanion T avec la base de données en ayant une relative assurance que les enregistrements originaux, s'ils ont été envoyés, ont été reçus et enregistrés.

Appendice D. Noms de domaines internationalisés

Pour être compatibles avec l'infrastructure DNS existante et simplifier la comparaison hôte et nom de domaine, les identités Diameter (les FQDN) sont représentées en forme ASCII. Cela permet au protocole Diameter de rester en ligne avec la stratégie du DNS de transparence aux effets des noms de domaine internationalisés (IDN) en suivant les recommandations de la [RFC4690] et de la [RFC5890]. Les applications qui prennent en charge les IDN sortent du domaine d'application du protocole Diameter mais l'interaction avec lui DEVRAIT utiliser le cadre de représentation et conversion décrit dans les [RFC5890], [RFC5891], et [RFC3492].

Adresse des auteurs

Victor Fajardo (éditeur)
Telcordia Technologies
One Telcordia Drive, 1S-222
Piscataway, NJ 08854
USA
tél. : +1-908-421-1845
mél : vf0213@gmail.com

Jari Arkko
Ericsson Research
02420 Jorvas
Finlande
tél. : +358 40 5079256
jari.arkko@ericsson.com

John Loughney
Nokia Research Center
955 Page Mill Road
Palo Alto, CA 94304
USA
tél. : +1-650-283-8068
john.loughney@nokia.com

Glen Zorn (éditeur)
Network Zen
227/358 Thanon Sanphawut
Bang Na, Bangkok 10260
Thailande
tél. : +66 (0) 87-0404617
mél : glenzorn@gmail.com