Protocole de Base DIAMETER
Architecture, Entités et Commandes

EFORT
http://www.efort.com

Le protocole DIAMETER a été conçu comme une version améliorée du protocole RADIUS. D'ailleurs ces deux protocoles sont des protocoles AAA (Authentication, Authorizattion, Accounting) et sont utilisés par des procédures d'authentification, d'autorisation et de taxation (online et offline) dans le monde IP. Le protocole de base DIAMETER est spécifié dans le RFC 3588 remplacé par le RFC 6733. Contrairement à RADIUS, acronyme de Remote Authentication Dial-In User Service, son nom est un jeu de mot, DIAMETER, signifiant diamètre en anglais, qui est le double du rayon (radius en anglais).
Un des objectifs était de maximiser la compatibilité et faciliter la migration de RADIUS à DIAMETER. Par exemple, un message DIAMETER comme un message RADIUS transporte des éléments d'information appelés AVP (attribute value pair) et tous les AVPs de RADIUS ont été repris dans le protocole DIAMETER.
DIAMETER est défini à travers un protocole de base et un ensemble d'applications. Cette conception permet une extension du protocole de base pour de nouvelles applications. Le protocole de base fournit un format de commande et d'AVP, des mécanismes pour un transport fiable, la livraison des commandes et le traitement des erreurs.
Le protocole de base doit être utilisé conjointement avec une application DIAMETER. Chaque application s'appuie sur les services du protocole de base. Plus de 80 applications DIAMETER ont été définies à ce jour pour le monde des télécommunications, notamment par 3GPP pour les architectures EPS (Evolved Packet System) qui représente le réseau 4G, IMS (IP Multimedia Subsystem) qui représente une architecture de service sur IP notamment pour la téléphonie sur IP, PCC (Policy and Charging Control) permettant le contrôle et la taxation des flux IP du client sur le réseau data mobile, M2M (Machine to Machine) pour faciliter les communications machine to machine et adapter le réseau mobile à ce type de device, et GAA/GBA (Generic Authentication Architecture / Generic Bootstrapping Architecture) pour une authentification de service. Le but de ce tutoriel est de décrire le protocole de base DIAMETER.

1 DIAMETER versus RADIUS

Radius avait tout d'abord pour objet de répondre aux problèmes d'authentification pour des accès distants, par liaison téléphonique, vers les réseaux des fournisseurs d'accès ou des entreprises. C'est de là qu'il tient son nom qui signifie Remote Authentication Dial In User Service. Il est défini dans les RFCs 2865 (RADIUS Authentication) et 2866 (RADIUS Accounting). Au fil du temps, il a été enrichi et on peut envisager aujourd'hui de l'utiliser pour authentifier les postes de travail sur les réseaux locaux, qu'ils soient filaires ou sans fil.
Radius est un protocole AAA. Ces initiales résument les trois fonctions du protocole :
A = Authentication : authentifier l'identité du client ;
A = Authorization : accorder des droits au client ;
A = Accounting : enregistrer les données de comptabilité de l'usage du réseau par le client.

Plusieurs point faibles du protocole RADIUS ont été adressés dans la conception du protocole DIAMETER :

- Taille réduite de la valeur d'AVP (1 octet pour RADIUS contre 3 pour DIAMETER). La longueur de l'AVP RADIUS a une taille d'un octet, ce qui limite la valeur de l'AVP a une longueur maximum de 255 octets. La longueur de l'AVP DIAMETER a une taille de 3 octets. Ce qui signifie que la valeur d'un AVP DIAMETER a une longueur maximum de

16 millions d'octets. Ceci est très utile notamment lorsque l'AVP transporte un profil usager qui peut mesurer plusieurs kilooctets.

- Nombre réduit de messages RADIUS parallèles (Identification réduite à 1 octet contre 4 pour DIAMETER). RADIUS permet d 'identifier la transaction par un numéro encodé sur 1 octet. Le nombre de transactions pouvant être traitées en parallèle est 256. DIAMETER permet d 'identifier la transaction par un identificateur encodé sur 4 octet. Le nombre de transactions pouvant être traitées en parallèle est 4 milliards.
- Incapacité de contrôler les flux vers le serveur (RADIUS fonctionne sur UDP alors que DIAMETER opère sur TCP ou SCTP). RADIUS fonctionne sur UDP ce qui ne constitue pas un transport fiable. DIAMETER fonctionne sur TCP ou SCTP permettant de fiabiliser le transport des requêtes/réponses DIAMETER, notamment via des mécanismes de contrôle de flux et de retransmission.
- Protocole client serveur ne permettant pas au serveur d'initier une requête : RADIUS est un protocole client serveur ; seul le client peut initier une requête et le serveur retourne une réponse une fois la requête traitée. DIAMETER qui est plutôt un protocole peer to peer permet au client ou au serveur d'initier une requête. Chaque nœud DIAMETER a un role dual client et serveur.
- Impossibilité de rajouter des commandes au protocole RADIUS : RADIUS ne permet pas la définition de commandes spécifiques à un "vendor". Par contre il permet la définition d'AVP spécifiques. Le protocole DIAMETER permet la définition de commandes et d'AVP spécifiques à un "vendor". Ainsi DIAMETER doit être considéré comme un protocole de base au dessus duquel, il est possible de définir des applications avec des commandes spécifiques et/ou des AVPs spécifiques.
- Sécurité au niveau du protocole RADIUS : Avec RADIUS, le client et le serveur doivent être préconfigurés avec un secret partagé même si la sécurité au niveau IP a été activée. DIAMETER au contraire peut sécuriser la communication entre paires avec des mécanismes standard de sécurité IP tels que IPSec, TLS (Transport Layer Security), ou DTLS (Datagram TLS).
- Comportement du proxy entre client et serveur non spécifié pour RADIUS: RADIUS et DIAMETER permettent l'utilisant de nœuds additionnels sur le chemin entre client et serveur. Ces nœuds s'appellent des proxy dans l'environnement RADIUS et Agent dans l'environnement DIAMETER. L'agent DIAMETER peut avoir plusieurs rôles et un de ses rôles est Proxy. Les autres rôles possibles sont Relay, Redirect et Translation. La différence entre les deux types de nœuds est que RADIUS n'a pas spécifié le comportement du Proxy, ni toute la procédure de routage. Par contre DIAMETER a totalement spécifié le comportement de l'agent et toute sa procédure de routage.

2 Protocole de base DIAMETER et identificateur d 'application

Toute commande DIAMETER doit appartenir à une application et doit inclure l'identificateur d'application (application Id) à laquelle la commande appartient. Les commandes du protocole de base sont soit liées à des aspects authentification/autorisation, soit à des aspects taxation.
Les commandes du protocole de base relatives aux aspects autorisation et authentification ont un Application Id égal à 0. Elles sont définies dans le RFC 6733.
Les commandes du protocole de base relatives aux aspects taxation offline ont un Application Id égal à 3.. Elles sont aussi définies dans le RFC 6733.
Les commandes du protocole de base relatives aux aspects taxation online ont un Application Id égal à 4. Elles sont spécifiées dans le RFC 4006.

Une nouvelle application aura un nouvel identificateur d'application si elle ajoute de nouvelles commandes ou si elle ajoute/supprime des AVPs obligatoires aux commandes du protocole de base.
A titre d'exemple, 3GPP qui normalise le monde des télécommunications mobiles a défini une nouvelle application de taxation offline pour la data mobile, appelée Gz, mais sans rajouter de nouvelles commandes ou ajouter/supprimer des AVPs obligations aux commandes du protocole de base. Cette application se contente d'utiliser les commande de taxation offline du protocole de base et de rajouter des AVPs optionnels, au protocole de base. Son identificateur d'application est donc 3.
Il en va de même pour l'application Gy de taxation online pour la data mobile qui ne fait que rajouter des AVPs optionnels au protocole de base. Son identificateur d'application est 4. Par contre, Par contre l'application S6 de gestion de mobilité dans le réseau EPS (4G) rajoute de nombreuses commandes et des AVPs obligatoires au protocole de base. Un nouvel identificateur d'application lui a été assigné, à savoir 16777251.

3 Entités DIAMETER

- Un **nœud DIAMETER** est un hôte qui implante le protocole DIAMETER. Il joue un rôle dual client et serveur. Il est client lorsqu'il émet une requête et serveur lorsqu'il reçoit une requête. Il faut donc plutôt considérer le protocole DIAMETER comme un protocole entre peers. Dans un réseau 4G appelé EPS (Evolved Packet System), il existe de nombreux nœuds DIAMETER, tels que le MME (Mobility Management Entity) qui prend en charge la gestion de la mobilité de l'UE (User Equipment) et le HSS (Home Subscriber Server) qui fournit au MME des vecteurs d'authentification pour authentifier l'UE et le profil de l'abonné afin de permettre au MME de connaître les autorisations assignée à l'abonné lors de sa souscription.

Le protocole DIAMETER peut être utilisé en mode associé ou peer to peer (sans agent) ou en mode quasi associé (avec un agent). Le tutoriel EFORT sur le thème Réseau de Signalisation DIAMETER (http://efort.com/r_tutoriels/DIAMETER_ROAMING_EFORT.pdf) présente les avantages à introduire un agent.

- Un **agent DIAMETER** est un nœud DIAMETER qui fournit des services de relai, de proxy ou de traduction.
- Un **agent relai** est un agent DIAMETER qui accepte des requêtes, et les relaie soit à un autre agent, soit au nœud DIAMETER de destination à partir des informations présentes dans les requêtes (e.g., Destination-Realm). Cette décision de routage est réalisée grâce à la table de routage basée sur le Destination Realm (nom de domaine de la destination), qui indique le nœud suivant pour un realm de destination donné. Les agents Relai ne réalisent aucun traitement de niveau application. Les agents Relai modifient les messages DIAMETER en y insérant et en y supprimant des informations de routage, mais ne modifient aucune autre partie des messages. Les agents Relai ne maintiennent pas d 'état de session mais doivent maintenir un état de transaction. Le maintien de l'état de transaction permet de garantir qu'une requête et une réponse appartenant à une même transaction suivent le même chemin.
- Un **agent proxy** comme un agent relai route le message DIAMETER en utilisant les tables de routage de realm. Cependant, il diffère puisqu 'il peut modifier les messages DIAMETER afin d 'implanter des politiques d 'opérateur (e.g., topology hiding). Un agents Proxy peut maintenir un état de session et doit maintenir un état de transaction. Puisque la mise en œuvre de politiques requiert la compréhension du service rendu, un agent Proxy doit publier les applications qu 'il supporte et doit comprendre la sémantique des commandes DIAMETER qu'il route. Par analogie par rapport à SS7, un agent relai

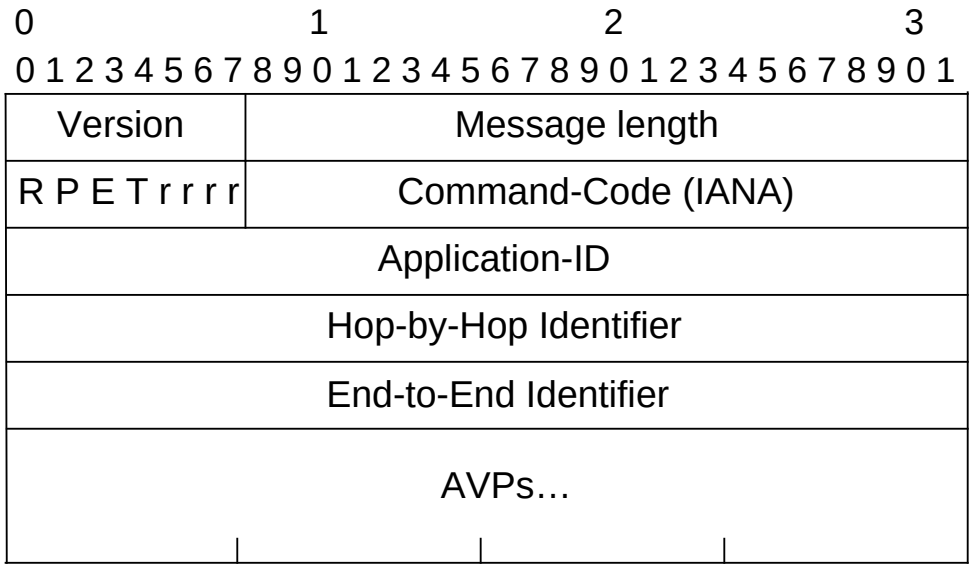
est un PTS qui route au niveau 3 en utilisant le DPC alors qu'un agent proxy est un PTS qui route au niveau SCCP et réalise des opérations GTT.

- Un **agent de redirection** fournit aussi une fonction de routage. Il sert de directory permettant généralement la traduction de Nom de domaine → Adresse du serveur. A la différence des autres types d'agent (relai et proxy) qui acheminent les messages DIAMETER, l'agent de redirection retourne un type particulier de message de réponse à l'émetteur de la requête. La réponse contient l'information de routage afin que l'émetteur puisse retransmettre son message directement au serveur destinataire. Un agent de redirection ne modifie donc pas le message DIAMETER. Il ne maintient ni un état de transaction ni un état de session.
- Un **agent de traduction** traduit les protocoles tels que DIAMETER en RADIUS ou DIAMETER en MAP (Mobile Application Part) pour des aspects d'autentification/autorisation data mobile ou DIAMETER en CAP (CAMEL Application Part) pour des aspects de taxation online ou DIAMETER en SOAP, etc.
- Les grands fournisseurs d'agent DIAMETER sont ORACLE (DSR, DIAMETER Signaling Router), F5 Networks (SDC, Signaling Delivery Controller) et Ericsson (DIAMETER Signaling Controller). L'agent est appelé DRF (DIAMETER Routing Function) dans les recommandations 3GPP.

Les nœuds DIAMETER (client/serveur) doivent supporter le protocole de base, qui inclut les aspects authentification, autorisation et taxation. Par ailleurs, ils doivent supporter chaque application Diameter nécessaire à la mise en œuvre du service du noeud.
Les agents relai et redirect DIAMETER sont par définition transparents aux applications. Ils doivent supporter le protocole de base et de manière transparente toutes les applications DIAMETER sans chercher à comprendre leur sémantique. En effet, les agents relai acheminent les requêtes et les réponses sur la base des AVPs relatifs au routage (Routing AVP) appartenant au protocole de base et présents dans toutes les commandes de toutes les applications (e.g., Destination Realm) et sur la base des entrées présentes dans la table de routage (realm-based routing table). Ils n'ont pas la nécessité de comprendre la sémantiques des commandes ou des AVPs non relatifs au routage. Les relais sont donc capables de traiter tous les types d'application DIAMETER et tous les types de commande pour offrir un service de relai.
Les proxy DIAMETER doivent supporter le protocole de base. Par ailleurs, ils doivent supporter chaque application DIAMETER nécessaire à la mise à la mise en œuvre de services proxyés. En effet, les agents proxy peuvent prendre des décisions relatives à des politiques. Par exemple, les proxy peuvent générer des messages de rejet dans le cas où des politiques sont violées. Les proxy doivent donc comprendre la sémantique des messages qui sont routés par eux et peuvent ne pas supporter toutes les applications DIAMETER.
Ainsi un agent d'un fournisseur sera proxy pour les applications que l'agent supporte et relai pour toutes les applications qu'il ne supporte pas.

4 Format des Commandes

Toute commande définie par le protocole ou par des applications doit avoir un format défini par le protocole de base et montré à la figure 1. Une commande peut être une requête ou une réponse.



R=Request bit, P=Proxiable bit, E=Error bit, T=reTransmission bit
AVP = Attribute Value Pair

Figure 1 : Format de commande DIAMETER

Une commande DIAMETER consiste en un en-tête de taille fixe (20 octets) suivi par un nombre variable d'AVPs qui représentent des éléments d'information. Le format de la commande est montré à la figure ci-dessus.

- Le champ « Version » indique le numéro de version de DIAMETER. La valeur de ce champ est positionnée à 1.
- Le champ « Message Length » indique la longueur totale du message en octets. La longueur doit avoir une valeur multiple de 4.
- Le fanion de commande spécifie 4 bits R, P, E et T et 4 bits de réserve..
  - R : Le bit R signifie « Request ». Il indique si le message est une requête ou une réponse. 1 Request; 0=Réponse.
  - P : Le bit P signifie Proxiable. Il indique si le message peut être routé par un agent proxy ou un agent relai ou un agent de redirection (P=1) ou s 'il doit être traité localement (P=0).
  - E : Le bit E signifie Error. Il indique si le message contient des erreurs protocolaires ou sémantiques. Lorsque la requête génère une erreur protocolaire, le message de réponse est retourné avec sont bit E positionné à la valeur 1 indiquant une erreur protocolaire (Catégorie de réponse 3XXX).
  - T : Le bit T signifie reTransmitted. Il indique si le message a été retransmis suite à un failover ou est utilisé pour supprimer à la réception des message dupliqués. Rappelons que toute requête DIAMETER doit être acquittée par une réponse correspondante. L'émetteur de la requête la retransmet s'il ne reçoit pas une réponse après un temporisateur.
  - r(eserved) : Les bits r sont réservés pour usager futur. Ils sont positionnés à 0 et ignorés par le récepteur.
- Command-Code (4 octets) : Est utilisé pour communiquer la commande associée au message. Chaque message DIAMETER doit contenir un code de commande afin que le récepteur sache identifier l 'action à réaliser pour chaque message.
- Application-ID (4 octets) identifie l 'application spécifique à laquelle appartient le message, tel que Mobile IP, Accounting, etc.
- Hop-by-hop identifier (4 octets) : transporte un identificateur utilisé afin d 'associer la requête et la réponse sur ce saut (hop). L 'émetteur de la réponse doit s 'assurer que la

valeur de cet identificateur est la même que celle présente dans la requête correspondante.

- End-to-end identifier (4 octets) : Est utilisé afin de détecter des messages dupliqués. L 'identificateur dans la réponse doit être identique à celui de la requête correspondante. L 'identification doit être unique pour au moins 4 minutes. Cet identificateur ainsi que l 'AVP Origin-Host (décrit plus tard) sont utilisés ensemble afin de détecter des duplications de message. Une requête dupliquée ne doit pas conduire à l 'envoi de deux réponses.

4.1 Application ID

Chaque application DIAMETER doit avoir un identificateur d'application assigné par l'IANA (Internet Assigned Numbering Authority). Pendant l'échange de capacités, les nœuds DIAMETER informent leur 'peers' des applications qu'ils supportent. Par ailleurs toutes les commandes DIAMETER contiennent un identificateur d'application, qui est utilisé par la procédure de routage des messages. En effet, un agent doit identifier s'il supporte l'application en mode relai ou proxy. Le tableau 1 liste un sous ensemble des applications DIAMETER définies par 3GPP.

Application	Application Id (Dec)	Application Id (Hex)
Diameter Common Messages	0	0x00000000
NASREQ	1	0x00000001
Diameter Base Accounting, Rf	3	0x00000003
Diameter Credit Control, Ro	4	0x00000004
Relay	0xFFFFFFFF	0xFFFFFFFF
Cx/Dx Interface Application	16777216	0x01000000
Rx Interface Application	16777236	0x01000014
Sh/Dh Interface Application	16777217	0x01000001
Re Interface Application	16777218	0x01000002
S6a/S6d Interface Application	16777251	0x01000023
S13/S13 ' Interface Application	16777252	0x01000024
S9 Interface Application	16777267	0x01000033
Gx Interface Application	16777238	0x01000016
Sy Interface Application	16777302	0x01000056
SWx Interface Application	16777265	0x01000031

Tableau 1 : Exemples d'identificateurs d'application

4.2 Hop-by-Hop Identifier

Le Hop-by-Hop Identifier est un identificateur dans la commande DIAMETER qui change à chaque saut de la commande. Ce champ permet de numéroter les messages à chaque relai ou proxy afin que la requête (commande dont le bit R =1) et la réponse associée (commande dont le bit R = 0) suivent le même chemin.
Un ID est choisi au hasard par l'entité qui initie une requête. Cet ID est sauvegardé par un agent qui est le saut suivant et remplacé par un autre ID généré par l'agent. Ce processus se répète à chaque saut tout au long du chemin. Le serveur qui répond à la requête, reprend le même ID que celui trouvé dans la requête et l'insère dans la réponse. Ainsi, les relais voient transiter les réponses avec les ID attendus. Pour poursuivre la chaîne, ils remettent l'ID de leur prédécesseur qu'ils avaient mémorisé.
Considérons l 'exemple présenté à la figure 2 :
• Un message arrive avec un ID Hop-By-Hop égal à 1.

• Le premier agent Relai/Proxy sauvegarde l 'ID présent dans la requête et le remplace par le sien (ici l' ID 3). L'agent mémore aussi l'adresse de transport du client émetteur à savoir (Port1, IP1). .
• Le second agent Relai/Proxy sauvegarde l ' ID présent dans la requête (ID = 3) et le remplace par le sien (ici l 'ID 4) et l'associe à l'adresse de transport du premier agent, à savoir, P2, I2 .
• Le serveur reprend le dernier ID Hop-By-Hop placé dans la requête et l 'insère dans la réponse (ID = 4) et le retourne à l'adresse de transport d'où provient la requête, i.e., P3, I3.

- Le second agent retrouve dans sa table des requêtes, l'ID qui correspond à la réponse qui vient d'arriver. Il remet l'ID d'origine et fait transiter le message (remplacement de l 'ID = 4 par l 'ID = 3) dans le sens arrière à P2, I2.
- Le premier agent retrouve dans sa table des requêtes, l'ID qui correspond à la réponse qui vient d'arriver. Il remet l'ID d'origine et fait transiter le message (remplacement de l 'ID = 3 par l 'ID = 1) dans le sens arrière à P1, I1.

Le client peut corréler la requête et la réponse grâce au champ End-To-End Identifier.

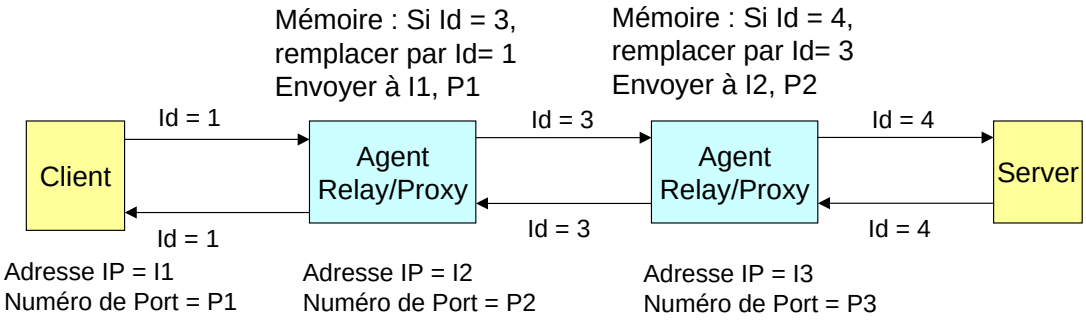


Figure 2 : Usage du Hop-by-Hop Identifier

4.3 End-To-End Identifier

Le End-To-End Identifier est un identificateur de "bout en bout" d'une longueur de 4 octets.
Le client DIAMETER va choisir aléatoirement un numéro unique pour chacune des requêtes émises.
Le serveur doit reprendre le même numéro pour retourner la réponse. Ainsi, l 'émetteur sait à quelle requête correspond la réponse.
En cas de même ID dans des commandes de réponse, les doublons devront être rejetés. Cela peut se produire lorsqu'une requête est retransmise (bit T est alors positionné à 1) et deux réponse sont retournées 'celle avec un bit 1 à 0 et celle avec un bit T à 1) avec le même End-To-End Identifier.

La figure 3 décrit un exemple d'en-tête de commande DIAMETER dont la taille est fixe et de 20 octets.. Il s'agit de la commande DIAMETER Authentication-Information-Request (AIR) de l'application S6 utilisée entre le MME et le HSS.

- Le champ « Version » est positionné à 1.
- Le champ « Message Length » indique la longueur totale du message en octets, soit dans notre exemple 376 octets. Cette valeur est toujours un multiple de 4 octets et inclut la taille de l'en-tête de 20 octets ainsi que celle de tous les AVPs.
- Le bit R est positionné à 1 puisqu 'il s 'agit d 'une requête.
- Le bit P est positionné à 1 afin de permettre à cette requête d'être routée par un agent proxy ou un agent relai ou d'être traitée par un agent de redirection.
- Le bit E est toujours positionné à 0 pour une requête.
- Le bit T est positionné à 0 car il s'agit de la première transmission de cette commande AIR.

- Les quatre autres bits du fanion sont réservés pour usager futur. Ils sont toujours positionnés à 0 et ignorés par le récepteur.
- Le champ Command-Code a la valeur 318. Il s'agit de la même valeur pour la requête AIR et la réponse AIA.
- Le champ Application-ID identifie l 'application spécifique à laquelle appartient la commande, Ici, il s 'agit de l 'application S6 dont l 'application-ID est 16777251 (valeur décimale correspondant à 0x01000023).
- Le champ Hop-by-hop Identifier est positionné à la valeur 0x30b00284. Il est utilisé afin d 'associer la requête et la réponse sur ce saut (hop).
- Le champ End-to-end Identifier est positionné à la valeur 0x0bbdcc60. Il est utilisé afin de détecter des messages dupliqués. L 'identificateur dans la réponse doit être identique à celui de la requête correspondante.

```
⊟ Diameter Protocol
    Version: 0x01
    Length: 464
  ⊟ Flags: 0xc0
      1... .... = Request: Set
      .1.. .... = Proxyable: Set
      ..0. .... = Error: Not set
      ...0 .... = T(Potentially re-transmitted message): Not set
      .... 0... = Reserved: Not set
      .... .0.. = Reserved: Not set
      .... ..0. = Reserved: Not set
      .... ...0 = Reserved: Not set
    Command Code: 318 3GPP-Authentication-Information
    ApplicationId: 3GPP S6a/S6d (16777251)
    Hop-by-Hop Identifier: 0x30b00284
    End-to-End Identifier: 0x0bbdcc60
```

Figure 3 : Exemple d'en-tête de commande DIAMETER (requête)

La figure 4 décrit l'en-tête de réponse correspondante. Les différences notables résident dans le bit R qui est positionné à 0 pour une réponse et dans la taille propre à la réponse, ici 388. Cette réponse n'indique pas d'erreur protocolaire, d'où le bit E à 0. Tous les autres champs d'en-tête de la requête et de la réponse ont des valeurs identiques.

```
⊟ Diameter Protocol
    Version: 0x01
    Length: 388
  ⊟ Flags: 0x40
      0... .... = Request: Not set
      .1.. .... = Proxyable: Set
      ..0. .... = Error: Not set
      ...0 .... = T(Potentially re-transmitted message): Not set
      .... 0... = Reserved: Not set
      .... .0.. = Reserved: Not set
      .... ..0. = Reserved: Not set
      .... ...0 = Reserved: Not set
    Command Code: 318 3GPP-Authentication-Information
    ApplicationId: 3GPP S6a/S6d (16777251)
    Hop-by-Hop Identifier: 0x30b00284
    End-to-End Identifier: 0x0bbdcc60
```

Figure 4 : Exemple d'en-tête de commande DIAMETER (réponse)

5 Commandes du protocole de base

Le protocole de base DIAMETER définit un ensemble de commandes d'authentification, d'autorisation et de taxation offline et online génériques (Figure 5). Le groupe de commandes connection management permet la mise en relation DIAMETER entre deux peers et la supervision de cette relation. Ces commandes ne sont pas routables puisqu'elles ne peuvent subir q'un seul saut..
Le groupe Session Operations permet d'établir une session DIAMETER d'authentification/d'autorisation entre un client et un serveur. Ces commandes sont de bout en bout routables via des agents.
Le groupe accounting (offline) permet l'envoi d'informations de taxation offline par un client à un server. Les commandes sont de bout en bout routables via des agents.
Le groupe accounting (online) permet la demande de crédit par un client à un serveur. Les commandes sont de bout en bout routables via des agents.

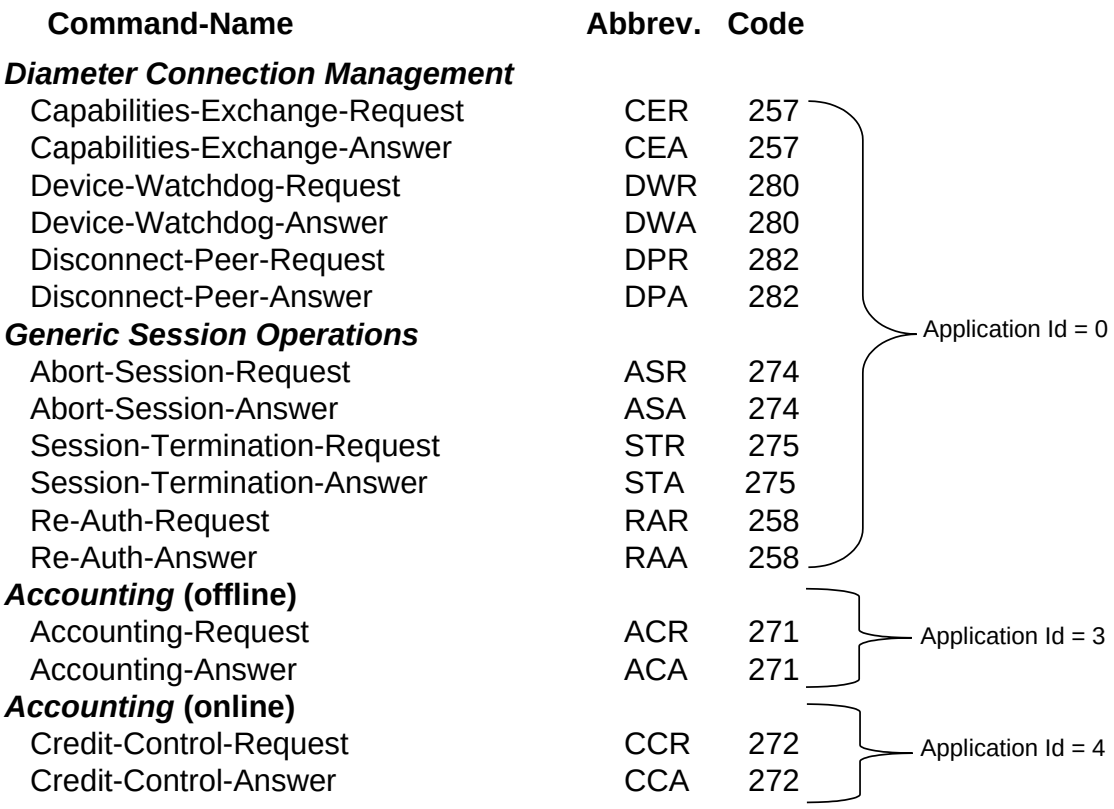


Figure 5 : Commandes du protocole de base DIAMETER

- Capabilities-Exchange-Request/Answer (CER/CEA, command code 257). Ces deux commandes doivent être utilisées dès que la connexion entre les entités DIAMETER a été établie à la couche de transport (TCP ou SCTP). Elles permettent aux deux entités paires DIAMETER d'échanger leurs capacités notamment en terme d'applications DIAMETER supportées). Si ces capacités échangées montrent une incompatibilité entre les deux entités (e.g., aucune application en commun), la mise en relation DIAMETER ne peut pas avoir lieu et la connexion est libérée à la couche transport. Les messages CER et CEA ne peuvent pas être routés par un agent DIAMETER à une entité distante DIAMETER. Ils ne sont échangés qu 'entre entités adjacentes directement reliées par une connexion TCP ou SCTP.

- Device-Watchdog-Request/Answer (DWR/DWA, command code 280). Ces commandes ont été définis afin d'améliorer la détection d'un problème au niveau d'un processus DIAMETER indépendamment des problèmes pouvant survenir au niveau de la connexion de transport. Dès que le problème est détecté, une procédure de failover est activée. La commande Device-Watchdog(Request agit comme un « ping » émis sur la connexion de transport afin de déterminer si l'entité distante est toujours accessible au niveau DIAMETER. Lorsqu'aucun trafic n'est échangé pendant un certain temps, les commandes DWR/DWA sont utilisées afin de superviser la relation DIAMETER et donc ne peuvent pas être routées par des agents DIAMETER.
- Disconnect-Peer-Request/Answer (DPR/DPA, command code 282). Afin d'améliorer le comportement des nœuds DIAMETER, si l'un des nœuds doit terminer sa relation DIAMETER avec son peer, il doit tout d'abord émettre une commande DPR afin d'informer la fin du dialogue DIAMETER. La commande DPR est acquitté par DPA. Ces commandes sont échangées entre deux entités paires DIAMETER adjacentes et ne peuvent pas être routées par un agent DIAMETER.
- Abort-Session-Request/Answer (ASR/ASA, command code 274). Lz commande Abort-Session-Request est utilisée par un serveur DIAMETER afin de demander l'abandon d'une session DIAMETER avec un équipement d'accès (NAS, Network Access Service). Ce type d'action peut être réalisé si l'autorisation auparavant accordée n'est plus valide. A la réception de cette requête le NAS répond avec Abort-Session-Answer, incluant le statut de l'opération demandée dans l'AVP Result-Code, indiquant le succès ou l'échec de l'action. Le NAS doit ensuite émettre une commande Session-Termination- Request au serveur DIAMETER qui peut être suivie par une commande Accounting-Request émise au serveur de taxation DIAMETER..
- Session-Termination-Request/Answer (STR/STA, command code 275). La commande Session-Termination-Request est utilisé par un NAS afin de notifier le serveur DIAMETER que la session indiquée n'est plus active. Le serveur DIAMETER l'acquitte avec Session-Termination-Answer.
- Re-Auth-Request/Answer (RAR/RAA, command code 258). La commande Re-Auth-Request est utilisée par le Serveur DIAMETER au NAS lorsqu'il est nécessaire de demander la re-authentification ou la re-autorisation.
- Accounting-Request/Answer (ACR/ACA, command code 271). Les commandes ACR/ACA ont la même sémantique que ceux de RADIUS. La commande Accounting-Request est utilisée par un équipement d'accès afin de rapporter des informations de taxation au serveur DIAMETER, informations acquittées avec Accounting-Answer.
- Credit-Control-Request/Answer (CCR/CCA, command code 272). La commande CCR permet d 'obtenir une autorisation de crédit pour un service donné auprès d 'un serveur de taxation online. Le crédit est retourné dans la réponse CCA..

5.1 Commandes du groupe Connection Management

La communication entre deux entités paires DIAMETER commence avec l'établissement d'une connexion de transport TCP ou SCTP (Figure 6).
L'initiateur de la communication envoie alors une commande DIAMETER capabilities-Exchange-Request (CER) à l'autre entité, qui répond avec Capabilities-Exchange-Answer (CEA).
Des exempes de capacités sont le numéro de version de protocole DIAMETER, les applications DIAMETER supportées, les mécanismes de sécurité supportés.
Si aucun message n'a été échangé pendant un certain temps, n'importe laquelle des parties peut émettre une commande Device-Watchdog-Request (DWR) qui doit être acquittée par l'autre partie à l'aide de Device-Watchdog-Answer.
N'importe laquelle des entités peut terminer la relation DIAMETER par l'envoi de Disconnect-Peer-Request (DPR) qui est acquitté par Disconnect-Peer-Answer. Puis la connexion de transport est libérée

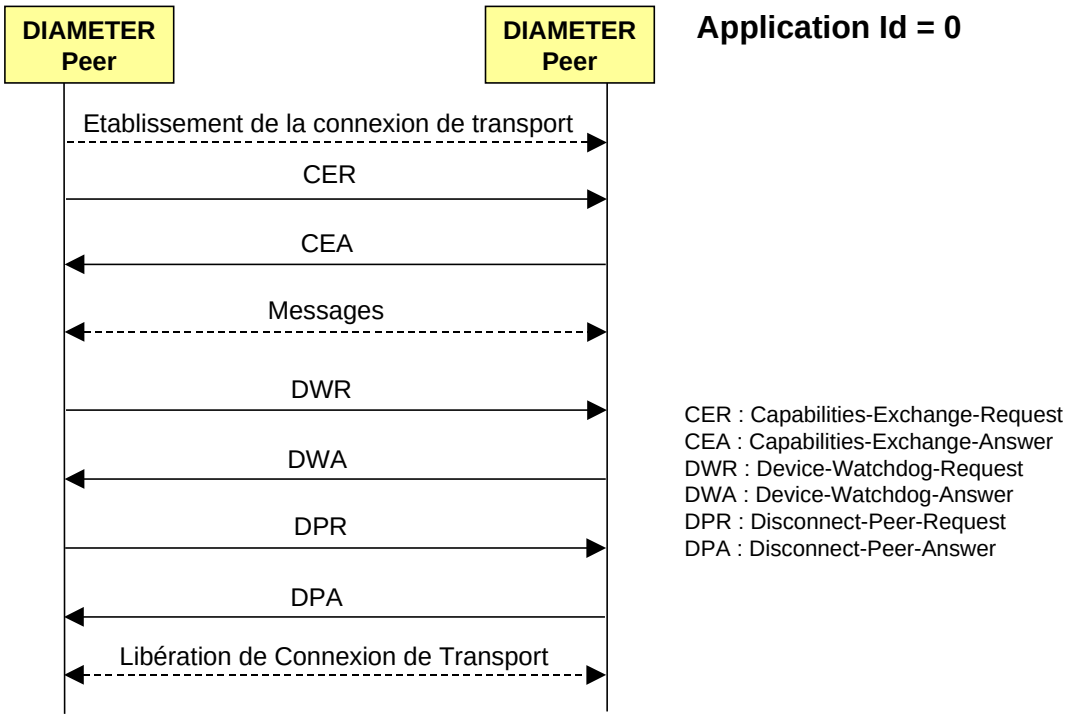


Figure 6 : Commandes du groupe Cennection Management

5.2 Commandes du groupe Session Operations

L'utilisateur souhaite accéder au réseau (e.g., accès à Internet) comme montré à la figure 7:

- Le NAS n'ayant pas les capacités d'authentifier le client, il collecte et transmet au serveur d 'authentification DIAMETER les données qu'il a collecté de l 'utilisateur (e.g., login et password). La requête émise par le NAS est AA-Request (AAR). Les commandes DIAMETER AAR et AAA ont été définies par l 'application NASREQ (Network Access Server Requirements) et ne font pas partie du protocole de base DIAMETER.
- Le serveur traite la requête AAR et retourne une réponse AA-Answer (AAA) qui inclut les informations d 'autorisation.
- On suppose le succès de l 'authentification. L 'usager accède donc au réseau. Une commande Accounting-Request (ACR) peut être envoyé pour enregistrer le début de la session.
- Le NAS a obtenu du serveur une autorisation d 'accès au réseau pour une durée limitée. Avant la fin de la durée, le NAS doit re-authentififier l 'usager. Pour ce faire, il échange de nouveau les messages AAR/AAA avec le server. Il peut aussi arriver que le serveur veuille re-authentifier l 'usager de manière non sollicitée. Dans ce cas, il émet une requête Re-Authorization Request (RAR) au NAS qui doit re-authentifier l 'usager. Le NAS acquitte le message RAR par Re-Authorization Answer (RAA), puis, afin de ré-authentifier l 'usager, échange de nouveau les commandes AAR/AAA avec le server.
- Si l 'authentification ou la ré-authentification échoue, le serveur demande l 'abandon de lasession via la commande Abort-Session-Request (ASR) qui est acquittée par Abort-Session-Answer (ASA) par le client.
- Lorsqu 'un usager indique au NAS qu 'il souhaite terminer sa session (session Internet par exemple), le NAS génère une requête Session-Termination-Request (STR) qu'il émet au serveur. Le serveur retourne une réponse Session-Termination-Answer (STA), incluant un AVP Result-Code.

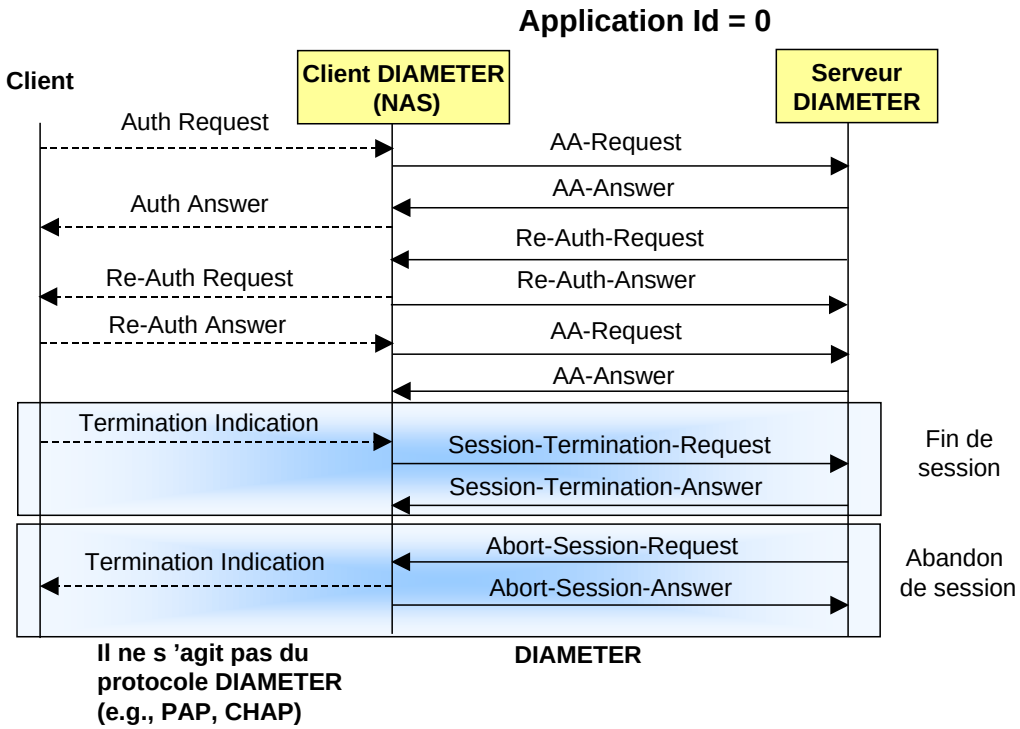


Figure 7 : Commandes du groupe Session Operations

5.3 Commande du groupe Accounting (Offline)

Le protocole de base DIAMETER fournit des services de comptabilité aux applications DIAMETER.
Lorsque une session est établie pour l 'usager, les mesures peuvent alors commencer et une commande *Accounting Request (ACR) start* est généré contenant les informations nécessaires à la taxation (Figure 8). Périodiquement pendant la session, des enregistrements intermédiaires peuvent être envoyés via la commande *Accounting Request (Interim*). Lorsque le client DIAMETER (Access Device) reçoit une demande de fin de session de l'utilisateur, une commande *Accounting Request stop* est transmise au serveur AAA du fournisseur d'accès et la session se termine implicitement.
Chaquecommande ACR est acquittée par une commande ACA (Accounting Answer).
Il est possible d'échanger des commandes ACR (Event) et ACA (Event). Dans ce cas, il n'y a pas de début de session et de fin de session. C'est le cas lors de l'envoi de SMS. Il n'y a donc qu'une commande de comptabilité émise correspondant à un événement (one-time event).
Les enregistrements de taxation sont corrélés avec l'AVP Session-Id, qui est un identificateur globalement unique et présent dans tous les commandes ACR/ACA. Par ailleurs si un service consiste en différentes sessions, chacune avec son AVP Session-Id, alors un AVP Accounting-Multi-Session-Id est utilisé pour la corrélation des enregistrements de taxation.

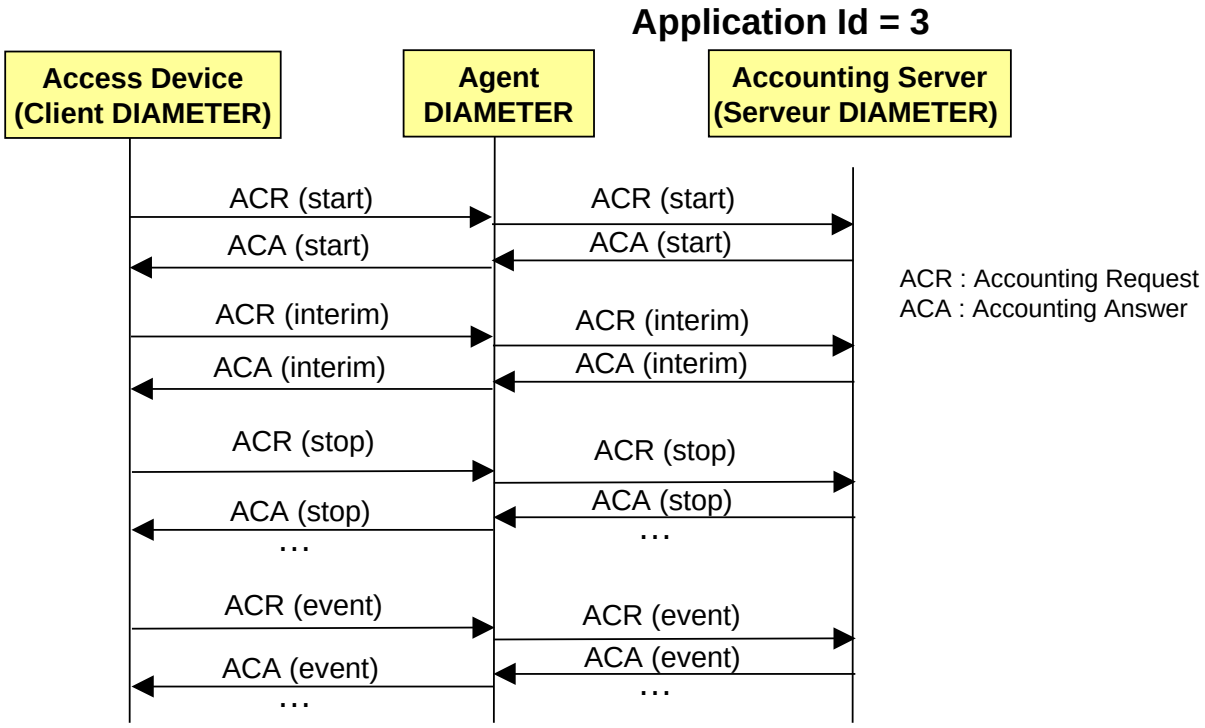


Figure 8 : Commande du groupe Accounting (Offline)

5.4 Commande du groupe Accounting (Online)

1. Le NAS (Network Access Server) reçoit une demande de session (e.g., établissement de default bearer ou contexte PDP primaire pour un APN donné si l'on considère le cas de la data mobile, le NAS étant représenté par un PCEF).
2. Afin de réaliser l'opération de réservation d'unité, Le NAS émet une requête *Credit-Control-Request* (CCR) à l'OCS avec l'AVP *CC-Request-Type* positionné à la valeur INITIAL_REQUEST.
3. L'OCS détermine le prix du service désiré à partir des informations spécifiques de service reçues dans la requête CCR et en soumettant une demande de valorisation auprès de la fonction de valorisation. Si le solde du client est suffisamment créditeur, l'OCS réserve le montant correspondant du compte du client;
4. Une fois la réservation effectuée, l'OCS retourne au NAS une réponse *Credit-Control-Answer* (CCA) avec l'AVP *CC-Request-Type* positionné à la valeur INITIAL_REQUEST afin d'autoriser l'exécution du service demandé (*Granted-Service-Unit*).
5. La livraison du service commence et les unités réservées sont au fur et à mesure décrémentées par le NAS.
6. Pendant la livraison du service, afin d'exécuter une nouvelle opération de réservation d'unités, le NAS émet une requête CCR avec l'AVP *CC-Request-Type* positionné à la valeur UPDATE_REQUEST, afin de rapporter les unités utilisées et demander de nouvelles unités. La requête CCR avec l'AVP positionné à la valeur UPDATE_REQUEST doit être émis par le NAS L'AVP *Used-Service-Unit* (USU) est présent dans la requête CCR afin de déduire les unités consommées sur le compte de l'usager et y réserver de nouvelles unités.
7. L'OCS déduit les unités consommées sur le compte. Si le solde est suffisamment créditeur, l'OCS réserve le montant correspondant sur le compte du client.
8. L'OCS retourne au NAS une réponse *Credit-Control-Answer* (CCA) avec l'AVP *CC-Request-Type* positionné à la valeur UPDATE_REQUEST afin d'autoriser la continuation du service (nouveau *Granted-Service-Unit*).
9. La livraison du service continue et les unités nouvellement réservées sont au fur et à mesure décrémentées par le NAS.

10. La session de service est terminée au niveau du NAS (e.g., Libération du default bearer ou du contexte PDP primaire d'un APN)
11. Le NAS émet une requête CCR avec l'AVP *CC-Request-Type* positionné à la valeur TERMINATION_REQUEST afin de terminer la session de contrôle de crédit et rapporter les unités utilisées.
12. L'OCS déduit le montant utilisé sur le compte du client. Les unités non utilisées ne sont pas prises en compte.
13. L'OCS acquitte la réception de la requête CCR en retournant la réponse CCA *avec l'AVP CC-Request-Type* indiquant TERMINATION_REQUEST (Eventuellement les AVPs *Cost-Information* indiquant le coût cumulé du service et *Remaining-Balance* sont inclus dans la réponse CCA). Toutes les commandes CCR/CCA INITIAL_REQUEST, UPDATE_REQUEST et TERMINATE_REQUEST partagent le même Session-Id.

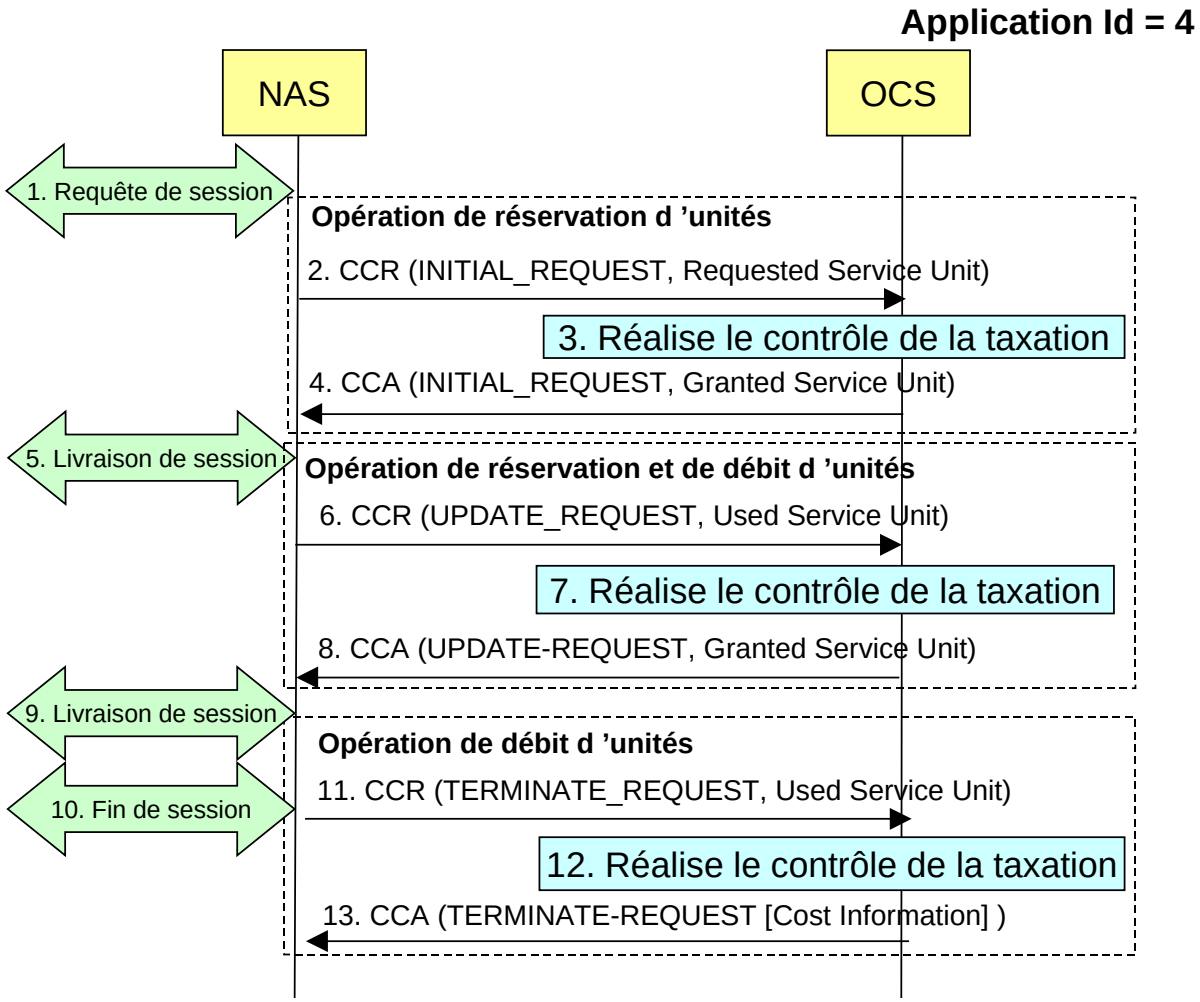


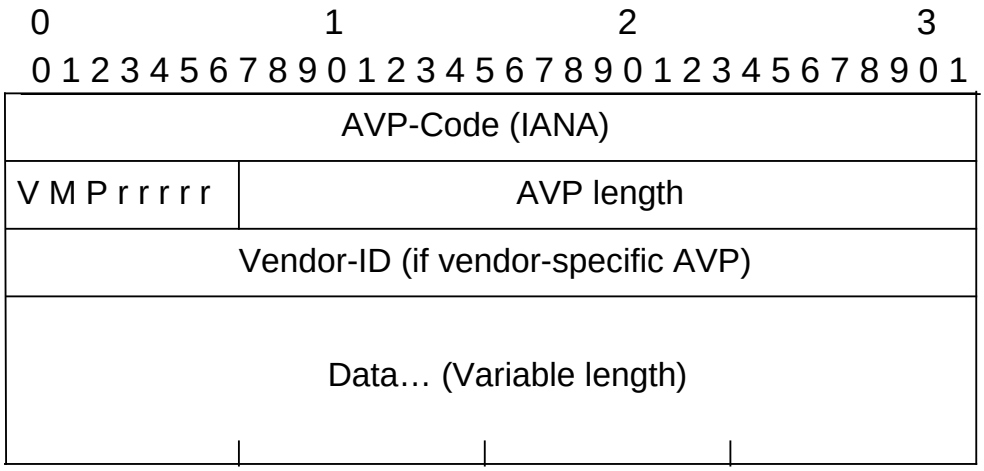
Figure 9 : Commandes du groupe Accounting (Online)

6 AVP DIAMETER

AVP (Attribute Value Pair) est l'objet le plus important du protocole DIAMETER. Il correspond à l''élément d'information dans le protocole ISUP ou au header dans le protocole SIP. Il est utilisé pour fournir toutes les données. Certains AVPs sont nécessaires à DIAMETER lui-même pour fonctionner, alors que d'autres fournissent des données liées aux applications exploitant DIAMETER. Les AVPs contenant l'information spécifique à une application peuvent être arbitrairement ajoutés aux commandes DIAMETER, aussi longtemps que les

AVPs nécessaires sont présents et que ceux qui doivent être ajoutés ne sont pas explicitement interdits par les règles du protocole.
Les AVPs transportent les informations d 'authentification, d 'autorisation, de sécurité, de comptabilité ainsi que des informations de configuration.
Le format de l 'AVP est donné à la figure 10. Il contient les champs suivants :

- AVP-Code (4 octets): identifie l 'AVP de manière unique. Les 256 premiers numéros sont réservés pour la compatibilité avec RADIUS. Les suivants sont utilisés par le protocole de base et ses extensions via des applications (numéros devant être alloués par l 'IANA).
- Fanions (8 bits):
  - Bit V, connu comme Vendor-Specific bit, indique si le champ optionnel Vendor-ID est présent dans l 'en-tête de l 'AVP. Quand positionné, le code AVP appartient à l espace d 'adressage des codes de ce constructeur.
  - Bit M : Le bit M signifie « Mandatory » bit. Il indique si le support de cet AVP est obligatoire. Si un AVP obligatoire est reçu dans une commande par un destinataire et que ce dernier ne comprend pas l'AVP, il est obligé de rejeter la commande. Par contre, si un AVP a un bit « M » est égal à 0 (AVP optionnel), le destinataire peut se contenter de l'ignorer s'il ne le comprend pas sans compromettre le traitement de la commande.
  - Bit P : Le bit P signifie « Protected ». Il indique la nécessité d 'un chiffrement pour une sécurité de bout en bout. Le protocole de base DIAMETER spécifie quels AVPs doivent être protégés.En pratique ce bit est positionné à la valeur 0.
  - Les bits rrrrr sont réservés et doivent être positionnés à la valeur 0.
- Length : Il s(agt de la longueur totale de l'AVP incluant l'en-tête et les données, mesurée en octets.
- Vendor-ID (4 octets) identifie le constructeur à l 'origine de cet AVP propriétaire. Ce champ est présent si le bit V est positionné à 1.
- Data : Longueur variable.



V=Vendor bit, M=Mandatory bit, P=Protected bit.

Figure 10 : Format d'AVP

La figure 11 montre le formation de deux AVPs, l'un défini par le protocole de base DIAMETER et l'autre spécifique à une application DIAMETER.
L 'AVP Destination-Realm est défini par le protocole de base. Il ne contient pas de paramètre Vendor-Id. Son en-tête mesure 8 octets (champs AVP Code, Flags et Length) suivi par les données. La valeur a une longueur de 33 octets. La longueur totale de l'AVP est 41 octets. Toutefois la taille de l'AVP à son émission doit être multiple de 4. L'AVP est donc

complémenté par 3 octets de bourrage pour atteindre 44 octets avant son émission. A la réception, le champs length indiquant 41 octets, le destinataire ignore les 3 derniers octets.

L 'AVP Visited-PLMN-Id est défini par 3GPP. Il contient un paramètre Vendor-Id dont la valeur est 10415 (C 'est le Vendor-Id pour 3GPP). Il s'agit d'un AVP défini par 3GPP pour ses applications DIAMETER. Son en-tête mesure 12 octets (Champs AVP Code, Flags, Vendor Id et Length) et la valeur a une longueur de 7 octets. La taille de l'AVP est donc 19 à laquelle vient se rajouter un octet de bourrage.

AVP Code: 283 Destination-Realm
AVP Flags : 0x40
0… …. = (V) Vendor-specific : Not Set
.1.. …. = (M) Mandatory: Set
..0. …. = (P) Protected: Not set
…0 …. = Reserved: Not set
…. 0… = Reserved: Not set
…. .0.. = Reserved: Not set
…. ..0. = Reserved: Not set
…. …0 = Reserved: Not set
AVP Length: 41 -- (44 padded bytes)
Value: epc.mnc001.mcc208.3gppnetwork.org

AVP Code: 1407 Visited-PLMN-Id
AVP Flags: 0xc0
1… …. = (V) Vendor-specific : Set
.1.. …. = (M) Mandatory: Set
..0. …. = (P) Protected: Not set
…0 …. = Reserved: Not set
…. 0… = Reserved: Not set
…. .0.. = Reserved: Not set
…. ..0. = Reserved: Not set
…. …0 = Reserved: Not set
AVP Vendor Id: 3GPP (10415)
AVP Length: 19 -- (20 padded bytes)
Value: 208.001

Figure 11 : Exemple d'AVPs DIAMETER

6.1 AVP Groupe

Un AVP Groupe a une valeur qui consiste en les AVPs dont il est constitué. Dans l 'exemple à la figure 12, l 'AVP Terminal-Information consiste en 2 AVPs : IMEI et Software-Version. Cet AVP a un en-tête de 12 octets car il est spécifique à un vendeur, ici, 3GPP. Par ailleurs sa longueur est de 12+44 octets = 56 octets.
Ces 44 octets représentent la taille des 2 AVPs qui forment l 'AVP Terminal-Information, à savoir IMEI (Taille = 28 octets dont 1 octet de bourrage) et Software-Version (Taille = 16 octets dot 2 octets de bourrage).

Toute requête DIAMETER doit contenir les AVPs suivants pour son routage : Orange-Host, Origin-Realm et Destination-Realm.

- AVP Origin-Host: Cet AVP est ajouté par le client à l'origine de la commande DIAMETER. Il indique son hostname.
- Origin-Realm AVP: Cet AVP contient le Realm (Nom de domaine) de l'émetteur de la commande DIAMETER.
- Destination-Realm AVP: Cet AVP contient le Realm auquel doit être routée la commande DIAMETER.

L 'AVP Destination-Host est optionnel. Le client peut ne pas connaître le serveur à contacter mais doit au moins connaître son nom de domaine (realm) ainsi que l 'application à utiliser. Le client envoie alors la requête à l 'agent relai, proxy, ou redirect. L 'agent résout alors l 'identité du serveur. Si aucun agent n 'est présent, alors le client doit connaître le serveur (Destination-Host) pour délivrer directement la requête sur la connexion de transport qu'il partage avec le serveur.
Une réponse DIAMETER ne contient que les AVPs Destination-Realm et Destination-Host. Ainsi le client connaîtra précisément le serveur en recevant la réponse. Par contre, la réponse est routée sur le même chemin que la requête grâce à l 'entête de la réponse et à l 'information hop-by-hop-identifier.

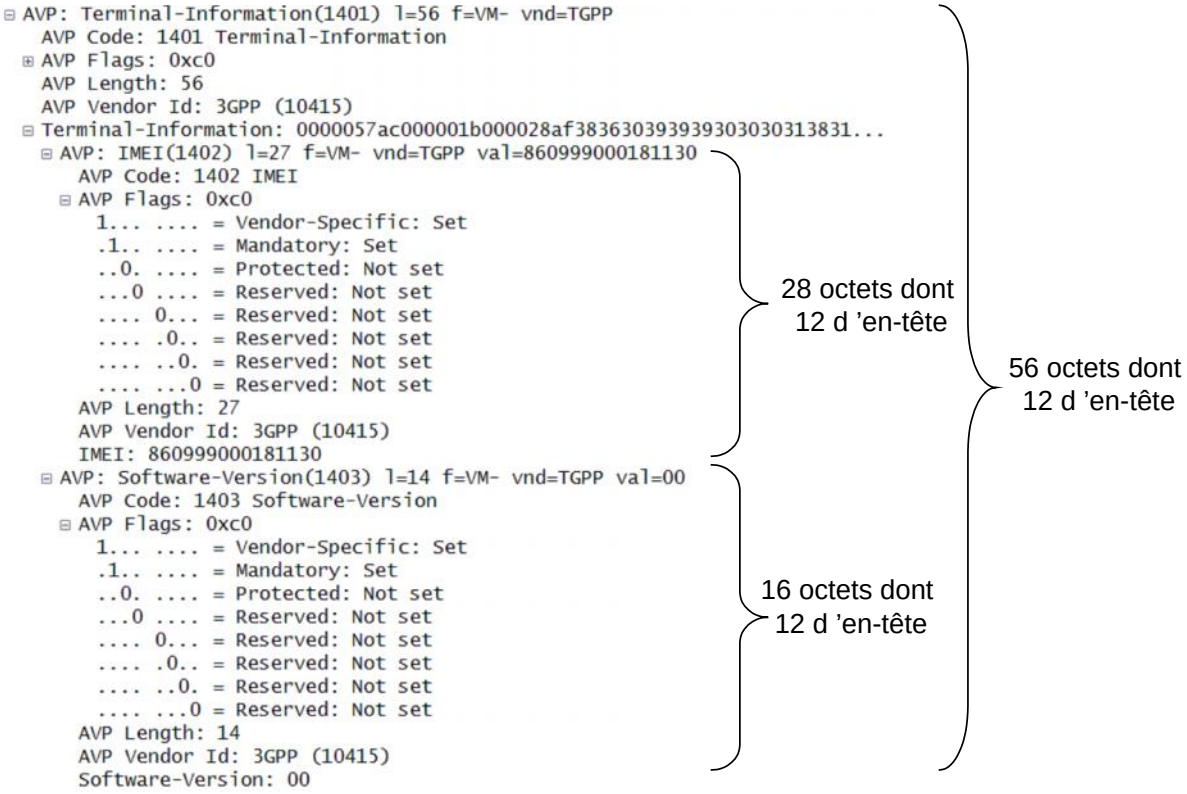


Figure 12 : Exemple d'AVP Groupe

Toute requête et tout réponse DIAMETER doivent contenir un AVP session-Id. L'AVP Session-Id (AVP Code 263) est utilisé afin d'identifier une session donnée. Toutes les commandes associées à une session doivent inclure une même valeur d'AVP Session-Id, e.g., ACR/ACA start, ACR/ACA interim et ACR/ACA stop.
Lorsqu'il est présent, cet AVP doit apparaître immédiatement derrière le header DIAMETER.
Le Session-Id doit commencer avec l'identité de l'émetteur (son hostname), suivi par deux entiers sur 32 bits chacun, suivi éventuellement d'un timestamp . Le session-Id est éternellement unique globalement. Le Session-Id est généré par le client DIAMETER qui initie la session. Exemple : hss1.orange.fr;1144207323;156

La formation EFORT « Le protocole de base DIAMETER et ses Applications » fournit toutes les clés de compréhension de l'écosystème DIAMETER défini par l'IETF et 3GPP.
http://efort.com/index.php?PageID=21&l=fr&f_id=132&imageField.x=3&imageField.y=4