|  |  |
| --- | --- |
| FT_horiz_bicolor  **research & development**  **FT/OLNC/OLN/CNC/GPC/2011/???/EDM** |  |
| Mobile Packet Core  **Policy & Charging Control**  **Manuel utilisateur des simulateurs**  Version 1.0  Date: janvier 2013  Authors:  Eric D'HEM (FT/OLNC/OLN/CNC/GPC/ATP) | | | |
| Present document contains information proprietary to France Telecom. Accepting this document means for its recipient he or she recognises the confidential nature of its content and his or her engagement not to reproduce it, not to transmit it to a third party, not to reveal its content and not to use it for commercial purposes without previous FT-Division R&D written consent. | | | |

|  |  |
| --- | --- |
| **CONTRIBUTion :** | |
| Nom : | Entité : |
| Maël THOUVENOT | FT/OLNC/OLN/CNC/GPC/ATP |
| Judikael GOURMELON | FT/OLNC/OLN/CNC/GPC/ATP |
| **Vincent ROUDIL** | FT/OLNC/OLN/CNC/GPC/ATP |
| **FT / Division R&D contacts :** | |
| [mael.thouvenot@orange-ftgroup.com](mailto:mael.thouvenot@orange-ftgroup.com)  [judikael.gourmelon@orange-ftgroup.com](mailto:judikael.gourmelon@orange-ftgroup.com)  [vincent.roudil@orange-ftgroup.com](mailto:vincent.roudil@orange-ftgroup.com) | |

|  |  |  |  |
| --- | --- | --- | --- |
| **VERIFICATION :** | | | |
|  | ***Nom*** | **Entité** | **Date** |
| Approbation Technique : | Maël Thouvenot  Vincent Roudil  Judikaël Gourmelon | CORE/M21/ATP |  |
| Contrôle livraison : | Jean-Philippe Le Brenn | CNC/GPC |  |
|  |  | | |
| **FT R&D / Orange Labs contacts :** | | | |
| [mael.thouvenot@orange-ftgroup.com](mailto:mael.thouvenot@orange-ftgroup.com)  [judikael.gourmelon@orange-ftgroup.com](mailto:judikael.gourmelon@orange-ftgroup.com)  [vincent.roudil@orange-ftgroup.com](mailto:vincent.roudil@orange-ftgroup.com)  [jeanphilippe.lebrenn@orange.com](mailto:jeanphilippe.lebrenn@orange.com) | | | |

Historique du document:

| **Version :** | **Date :** | **Modifications :** | **Auteurs :** |
| --- | --- | --- | --- |
| V0.1 | 09/02/2012 | Création | Eric D’HEM |
| V0.2 | 29/02/2012 | OCS V1.3.0 et PCRF V1.0.0. | Eric D’HEM |
| V0.3 | 04/06/2012 | OCS V1.4.0 et PCRF V1.1.0. | Eric D’HEM |
| V0.4 | 06/02/2013 | PCRF V1.2.0. | Eric D’HEM |
| V0.5 | 07/02/2013 | OCS V1.5.0. | Eric D’HEM |

Historique des livraisons du simulateur d'OCS :

| **Version :** | **Date :** | **Modifications :** | **Versions MTS à utiliser** |
| --- | --- | --- | --- |
|  | 01/10/2010 | Gestion de la consommation d'un seul service, en temps uniquement, d'un abonné identifié par sa SIP URI. | V4.9.0 et suivantes |
| V0.1.0 | 16/09/2011 | Gestion de la consommation en volume. |  |
| V0.2.0 | 16/09/2011 | Gestion du code 4012 et de l'AVP Validity-Time. |  |
| V0.3.0 | 16/09/2011 | Gestion de l'AVP Final-Unit-Action. |  |
| V0.4.0 | 26/09/2011 | Gestion du code 4011. |  |
| V0.5.0 | 30/09/2011 | Gestion de plusieurs services et de l'AVP Rating-Group. |  |
| V0.5.1 | 04/10/2011 | Ecriture du script de façon modulaire. Séparation claire du traitement respectif des CCR-I, CCR-U, CCR-T. |  |
| V1.0.0 | 07/11/2011 | Gestion des sessions, du RAR et de l'ASR. |  |
| V1.1.0 | 18/11/2011 | Gestion de plusieurs USU dans le CCR.  Gestion partielle de l'AVP Tarif-Time-Change. |  |
| V1.2.0 | 20/12/2011 | Gestion de plusieurs type d’identifiant de souscription. |  |
| V1.3.0 | 06/03/2012 | Gestion de services décrits avec plusieurs types d'unité. |  |
| V1.4.0 | 04/06/2012 | Gestion d'un CCR-I sans Service-Identifier. | V5.5.0 et suivantes |
| V1.5.0 | 01/02/2013 | Gestion des AVP 868, 869, 1226, 871, 881 et 1264 dans le CCA.  Possibilité de recharger le crédit de l'abonné avant d'envoyer un RAR.  Correction d'anomalies :  - Result-Code au niveau MSCC.  - Envoi du RAR et de l'ASR par la connexion en cours de la session. | V5.7.1 et suivantes  Modification du dictionnaire Diameter (changecontrol.xml) |

NB : La version 2.0 sera celle qui gère la réception de plusieurs MSCC dans un CCR.

Historique des livraisons du simulateur du PCRF :

| **Version :** | **Date :** | **Modifications :** | **Versions MTS à utiliser** |
| --- | --- | --- | --- |
| V.1.0.0 | 29/02/2012 | Gestion des sessions, du RAR et de l'ASR.  Gestion des AVP Event-Trigger et Charging-Information dans le CCA. | V5.4.0 et suivantes  Modification du dictionnaire Diameter (dictionary.xml) |
| V1.1.0 | 27/06/2012 | Gestion de l'AVP Charging-Rule-Name dans le CCA pour une règle prédéfinie. | V5.5.0 et suivantes |
| V1.2.0 | 04/02/2013 | Gestion de des AVP Monitoring-Key et Usage-Monitoring-Level dans le CCA.  Correction de l'anomalie : envoi du RAR et de l'ASR par la connexion en cours de la session. | V5.7.1 et suivantes |

NB : La version 2.0 sera celle qui gère plusieurs règles prédéfinies et/ou dynamiques dans le CCA.

Table des matières

[1. Introduction 4](#_Toc326314074)

[1.1 Architecture Policy & Charging Control 4](#_Toc326314075)

[1.2 Grandes lignes 4](#_Toc326314076)

[1.3 Abbreviations 4](#_Toc326314077)

[1.4 Reference 5](#_Toc326314078)

[2. Manuel utilisateur 6](#_Toc326314079)

[2.1 Comment démarrer le simulateur 6](#_Toc326314080)

[2.1.1 En mode graphique 6](#_Toc326314081)

[2.1.2 En mode commande 6](#_Toc326314082)

[2.2 Comment configurer le simulateur 6](#_Toc326314083)

[2.2.1 Configuration générale 6](#_Toc326314084)

[2.2.1.1 Configuration locale 6](#_Toc326314085)

[2.2.1.2 Configuration modulaire 7](#_Toc326314086)

[2.2.2 Configuration pour la charge 7](#_Toc326314087)

[2.2.2.1 Parallèliser 7](#_Toc326314088)

[2.2.2.2 La mémoire 7](#_Toc326314089)

[2.2.3 Configuration Diameter 7](#_Toc326314090)

[2.2.3.1 Le routage des messages 7](#_Toc326314091)

[2.2.3.2 No capable peer 8](#_Toc326314092)

[2.2.3.3 DIAMETER\_APPLICATION\_UNSUPPORTED 8](#_Toc326314093)

[2.2.3.4 Configuration des dictionnaires Diameter 8](#_Toc326314094)

[2.2.4 Configuration de l'utilisateur 9](#_Toc326314095)

[2.3 Comment remplir le fichier de données 9](#_Toc326314096)

# Introduction

## Architecture Policy & Charging Control

La figure ci-dessous présente les fonctions principales définies dans le document de spécification technique 3GPP Policy & Charging Control Architecture (3GPP TS 23.203 ‎[6]).



OCFS Online Charging System

OCS Online Charging System

PCEF Policy and Charging Enforcement Function

PCRF Policy and Charging Rules Function

SPR Subscriber Profile Repository

AF Application Function

BBERF Bearer Binding and Event Reporting Function

En vert, les fonctions faisant l'objet des simulateurs dont le présent document est le manuel utilisateur.

## Grandes lignes

Ce document complète le manuel officiel de l'outil cité en référence. Il décrit comment configurer MTS, comment le lancer et plus spécifiquement, comment décrire les données concernant les abonnés et les services.

## Abbreviations

|  |  |
| --- | --- |
| **3GPP** | 3rd Generation Partnership Project |
| **AVP** | Attribute Value Pair |
| **CCA** | Credit Control Answer |
| **CTF** | Charging Trigger Function |
| **DCCA** | Diameter Credit Control Application |
| **MSCC** | Multiple Service Credit Control |
| **MTS** | Multi-protocol Test Suite (former IMSLoader tool, now Open Source) |
| **OCS** | Online Charging System |
| **OFCS** | Offline Charging System |
| **PCC** | Policy and Charging Control |
| **PCEF** | Policy and Charging Enforcement Function |
| **PCRF** | Policy and Charging Rule Function |
| **QCT** | Quota Consumption Time |
| **QHT** | Quota Holding Time |
| **RAT** | Radio Access Technology |
| **RG** | Rating Group |
| **SID** | Service-Identifier |
| **SPR** | Service Profile Repository |
| **TTC** | Tariff-Time-Change |

## Reference

1. IETF RFC 3588 : "Diameter Base Protocol".
2. IETF RFC 4006 : "Diameter Credit Control Application".
3. 3GPP TS 32.299 : "Diameter charging applications. Gy reference point".
4. MTS user manual.
5. 3GPP TS 32.296 : "Online Charging System (OCS): Applications and interfaces".
6. 3GPP TS 23.203 : "Policy and charging control architecture".

# Manuel utilisateur

## Comment démarrer le simulateur

MTS fonctionne sous Windows et sous Linux. Les commandes de ce document sont décrites pour Linux. Sauf contre-indication explicite, elles sont également applicables pour Windows en adaptant la syntaxe (notamment, l'extension des ".sh" devient ".bat").

Le simulateur peut se démarrer en mode graphique (*startGui*) ou en mode commande (*startCmd*). Il est lancé à partir du répertoire <installDir>/bin car le shell en appelle d'autres du même repertoire.

Dans l'environnement Linux :

* Il est lancé avec *sudo* pour permettre l'ouverture de fichiers ou l'écoute des ports systèmes qui nécessitent les droits de root.
* Ne pas lancer en arrière plan sinon le mot de passe de *sudo* est compris comme une commande. Rappel : utiliser la séquence "$ Ctrl+z $ bg" pour mettre la dernière commande en arrière plan).

### En mode graphique

Le mode graphique est documenté dans <installDir>/doc/MTS\_manual\_gui.htm.

### En mode commande

Le mode commande est documenté dans <installDir>/doc/MTS\_manual\_cmd.htm.

$ sudo ./startCmd.sh

Usage: startCmd <testFile>|<masterFile>

-seq[uential]|-load|<testcaseName>

-testplan

[-param[eter]:<paramName>+<paramValue>]

[-config[uration]:<configName>+<configValue>]

[-level[Log]:ERROR=0|WARN=1|INFO=2|DEBUG=3]

[-stor[ageLog]:disable=0|file=1]

[-gen[Report]:false|true]

[-show[Report]:false|true]

**testFile** : nom du fichier contenant la balise racine de la grammaire <test>.

**masterFile** : ne sert qu'en mode maître/esclave.

**-seq**[uential] : tous les testcases dont l'attribut "state" sont à "true" sont lancés séquentiellement.

**-load** : tous les testcases dont l'attribut "state" sont à "true" sont lancés simultanément.

**testcaseName** : nom de l'unique testcase à lancer

**-stor**[ageLog]:disable=0|file=1 : indique si des logs sont à générer ou pas.

**-level**[Log]:ERROR=0|WARN=1|INFO=2|DEBUG=3 : indique le niveau des logs à générer (quand ils sont demandés).

$ sudo ./startCmd.sh ../../PCC/Scripts/OCS/test.xml OCS\_Server -level:DEBUG -storageLog:file

Une fois lancé en mode commande :

* Taper 'S' équivaut à cliquer sur le bouton "Stop" du testcase en mode graphique.
* Taper 'K' sert à quitter l'outil brutalement.

## Comment configurer le simulateur

### Configuration générale

#### Configuration locale

La configuration de l'outil se trouve dans le repertoire <installDir>/conf. Il est possible de particulariser cette configuration en utilisant un fichier de propriétés local rangé dans le répertoire du script. MTS cherche d'abord une valeur locale d'une propriété avant de chercher celle de l'outil.

Pour particulariser la configuration :

* lancer MTS en mode graphique,
* lancer la commande **Test/Edit configuration**.

#### Configuration modulaire

La configuration d'MTS est modulaire. Chaque protocole supporté a son jeu de propriétés. Celles concernant la configuration générale de l'outil se trouvent dans le fichier <installDir>/conf/tester.properties.

Dans le cadre des simulateurs PCC, les propriétés à retenir sont :

* gui.EDITOR\_PATH : positionné à "usr/bin/gedit" sans les guillemets. Elle associe l'éditeur Gedit pour lire les scripts XML quand on presse les boutons "Edit" en mode graphique.
* operations.CSV\_SEPARATOR : positionne la valeur du caractère de séparation des fichiers CSV.
* logs.MAXIMUM\_LEVEL et logs.STORAGE\_LOCATION positionnent respectivement le niveau de logs (ERROR, WARN, INFO, DEBUG) et la façon de les présenter : DISABLE (pas de logs), FILE (en fichiers), MEMORY (en mode graphique uniquement).

### Configuration pour la charge

#### Parallèliser

Chaque testcase jouant un scenario en mode serveur traite un message. Pour augmenter le nombre de messages traités simultanément, il suffit de lancer en parallèle plusieurs testcase du même scenario serveur.

<test>

<for index="[i]" from="1" to="4">

<testcase name="PCRF[i]" description="diameter" state="true">

<scenario name="DIAMETER">PCRF.xml</scenario>

</testcase>

</for>

</test>

Dans l'exemple ci-dessus, un message entrant dans MTS dont le protocole est DIAMETER sera traité par le premier scenario PCRF qui se libère.

#### La mémoire

L'erreur *java.lang.OutOfMemoryError* est levée lorsque la JVM ne peut plus allouer de mémoire pour un objet. Le ramasse-miette (*garbage collector*) ne peut plus en libérer.

Une possibilité est de fixer la taille maximale de la mémoire avec l'option –**Xmx**n au lancement de la JVM où n indique la mémoire initiale disponible. La valeur de n est fixée à l'installation de MTS mais peut se modifier par la suite en éditant le fichier <installDir>/bin/memory.

### Configuration Diameter

La configuration de l'outil se trouve dans le fichier <installDir>/conf/diameter.properties.

Dans le cadre des simulateurs PCC, les propriétés à retenir sont :

* listenpoint.LOCAL\_PORT : port d'écoute par défaut.
* capability.AUTO\_CER\_CEA\_ENABLE : positionné à "true", dispense le script de gérer l'échange de capacité.
* DEFAULT\_SERVER\_URL : destination du CER quand MTS est à l'origine de l'échange de capacités.

#### Le routage des messages

Le routage des messages dans MTS est documenté dans le document <installDir>/doc/MTS\_user\_manual\_core.htm (§6.6.4 Routing messages to scenarios).

Dans le cadre des simulateurs PCC, le routage a été fait avec les valeurs suivantes :

* route.MESSAGE\_ID : sa valeur ne sert pas car les messages ne sont pas envoyés avec l'attribut "destScenario".
* route.TRANSACTION\_ID : header.hopByHop. Cette valeur ne sert qu'en mode client. Elle ne sert donc pas pour les simulateurs d'OCS et PCRF qui sont en mode serveur.
* route.SCENARIO\_NAME : message.protocol. Par cette valeur, les messages entrants dans MTS sont routés vers le scenario dont le nom correspond au protocole. Le scenario qui déroule le script du simulateur (OCS, PCRF) a été nommé DIAMETER pour que lui soient routés tous les messages Diameter entrants dans MTS.

#### No capable peer

Dans la configuration Diameter de MTS, nous avons les propriétés suivantes :

* listenpoint.LOCAL\_HOST : quand cette valeur est vide, MTS prend l'adresse IP de l'une des interfaces.
* DEFAULT\_SERVER\_URL

Le CER est construit ainsi :

***Origin-Host*** *= listenpoint.LOCAL\_HOST*

***Origin-Realm*** *= node.REALM*

***Host-IP-Address*** *= listenpoint.LOCAL\_HOST*

***Vendor-Id*** *= node.VENDOR\_ID*

***Product-Name*** *= node.PRODUCT\_NAME*

***Supported-Vendor-Id*** *= les capability.SUPPORTED\_VENDORID.n*

***Auth-Application-Id*** *= les capability.AUTH\_APPLICATION.n*

***Acct-Application-Id*** *= les capability.ACCT\_APPLICATION.n*

***Vendor-Specific-Application-Id*** *= les capability.VENDOR\_ACCT\_APPLI.n et les capability.VENDOR\_AUTH\_APPLI.n*

***Firmware-Revision*** *= node.FIRMWARE\_REVISION*

Il est envoyé sur la valeur de l'attribut "remoteURL" de la balise <sendMessageDiameter> du premier envoi de requète. Si l'attribut "remoteURL" n'est pas renseigné, la propriété DEFAULT\_SERVER\_URL est utilisée.

Le CEA est construit ainsi :

***Result-Code*** *= 2001*

***Origin-Host*** *=* *listenpoint.LOCAL\_HOST*

***Origin-Realm*** *= node.REALM*

***Host-IP-Address*** *= listenpoint.LOCAL\_HOST*

***Vendor-Id*** *= node.VENDOR\_ID*

***Product-Name*** *= node.PRODUCT\_NAME*

***Supported-Vendor-Id*** *= les capability.SUPPORTED\_VENDORID.n qui étaient dans le CER*

***Auth-Application-Id*** *= les capability.AUTH\_APPLICATION.n qui étaient dans le CER*

***Acct-Application-Id*** *= les capability.ACCT\_APPLICATION.n qui étaient dans le CER*

***Vendor-Specific-Application-Id*** *= les capability.VENDOR\_ACCT\_APPLI.n et les capability.VENDOR\_AUTH\_APPLI.n qui étaient dans le CER*

***Firmware-Revision*** *= node.FIRMWARE\_REVISION*

Il est envoyé sur le *Host-IP-Address* du CER.

2 raisons peuvent être à l'origine d'un échec dans l'échange de capacités :

* Quand MTS ne reconnaît pas son correspondant. Par exemple quand le DEFAULT\_SERVER\_URL de l'envoyeur ne correspond pas l'Origin-Host du receveur.
* Quand les 3 intersections entre ce qui proposé par l'envoyeur dans le CER et ce qui est supporté également par le receveur sont vides. Les 3 intersections sont la liste des valeurs communes des AVP 258 Auth-Application-Id, 259 Acct-Application-Id et 260 Vendor-Specific-Application-Id.

#### DIAMETER\_APPLICATION\_UNSUPPORTED

Lors de l'échange de capacités, MTS établit 3 listes d'applications négociées :

* Celles concernant les authentifications et autorisations d'accès. Elles ont été négociées avec l'AVP 258 Auth-Application-Id.
* Celles concernant la gestion de comptes. Elles ont été négociées avec l'AVP 259 Acct-Application-Id.
* Celles qui sont spécifiques à un vendeur. Elles ont été négociées avec l'AVP 260 Vendor-Specific-Application-Id.

Si une application, (DDCA par exemple), a été négociée en tant que spécifique à un vendeur. Si la pile Diameter reçoit une commande (CCR par exemple) avec un AVP 258 ou 259 au lieu de 260, elle retournera l'erreur DIAMETER\_APPLICATION\_UNSUPPORTED avant même de router le message vers un scenario.

#### Configuration des dictionnaires Diameter

MTS, à l'instar de Wireshark, permet d'encoder et de décoder les valeurs numériques qui correspondent aux commandes, aux vendeurs, aux AVP et aux valeurs énumérées. L'encodage permet d'écrire des scripts lisibles en replaçant le code par une chaîne de caractères plus parlante. Le décodage permet d'écrire des logs lisibles en traduisant le code reçu en texte plus explicite.

Toujours à l'instar de Wireshark, il est possible d'adapter le contenu du dictionnaire en ajoutant, modifiant ou supprimant des correspondances. Cette modification se fait pour tout l'outil; il faut donc être très prudent quant aux adaptations qui doivent être faites. Actuellement, MTS ne conserve qu'une seule correspondance par AVP, la dernière chargée est conservée. Il est donc impossible de traduire un AVP de deux façons différentes dans un même script même en spécifiant le code du vendeur spécifique. Le Mantis (outil Devoteam de remontée de problèmes) 2733 a été ouvert le 09 février 2012. Une propriété de configuration (dictionary.PATH) a été ajoutée dans MTS V5.6.0.

Comment Wireshark et MTS construisent leur dictionnaire :

* La grammaire à respectée est décrite dans le fichier <installDir>/conf/diameter/**dictionary.dtd**.
* Le fichier racine contenant le dictionnaire est <installDir>/conf/diameter/**dictionary.xml**.
* Il est possible d'y inclure des fichiers séparés en respectant la syntaxe XML par l'utilisation des déclarations de références d'entités (DOCTYPE, ENTIY).

### Configuration de l'utilisateur

L'utilisateur peut rendre configurable un paramètre défini dans le script :

<parameter name="[sessionsHandled]" description="To handle Session-Id" operation="set" value="true" editable="true"/>

Dans la ligne ci-dessus, la valeur du paramètre sessionsHandled est positionnée à true à l'ouverture de la liste de tests. L'attribut editable positionné à true permet de modifier cette valeur avant de lancer le ou les tests.

La modification en mode graphique se fait par le menu "Test/Edit parameter".

Liste des paramètres configurables définis pour l'OCS :

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Value** |
| [PCEF-Realm] |  | pcef.imsloader.com |
| [OCS-Realm] |  | ocs.imsloader.com |
| [sessionsHandled] | To handle Session-Id | true |
| [TQTbyOCS] | Time-Quota-Threshold AVP sent by OCS. | true |
| [VQTbyOCS] | Volume-Quota-Threshold AVP sent by OCS. | true |
| [UQTbyOCS] | Unit-Quota-Threshold AVP sent by OCS. | true |
| [QHTbyOCS] | Quota-Holding-Time AVP sent by OCS. | true |
| [QCTbyOCS] | Quota-Consumption-Time AVP sent by OCS. | true |
| [TriggerByOCS] | Trigger AVP sent by OCS. | true |

Liste des paramètres configurables définis pour le PCRF :

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Value** |
| [GxApplicationId] | 3GPP TS 29.212 Gx application identifier | 16777238 |
| [sessionsHandled] | To handle Session-Id | true |

## Comment remplir le fichier de données

Au lancement, le simulateur charge dans des variables les informations du fichier Subscriber.csv.

L'OCS modifie dynamiquement le crédit dans ses variables mais pas dans le fichier. Donc arrêter le script, c'est perdre les valeurs dynamiques.

Pour l'OCS, chaque ligne de ce fichier est un service d'un abonné. Un abonné ayant plusieurs services se trouvent sur plusieurs lignes.

Pour le PCRF, chaque ligne de ce fichier est un abonné. Il n'y a pas de notion de service.

Les données sont décrites dans un fichier CSV.

**Attention** : ne pas mettre de point-virgule dans les valeurs, c'est le caractère de séparation. Le Mantis 3735 a été ouvert.

Les colonnes sont:

**userIdE164** : valeur de l'AVP Subscription-Id-Data quand l'AVP Subscription-Id-Type vaut 0 (END\_USER\_E164).

**userIdIMSI** : valeur de l'AVP Subscription-Id-Data quand l'AVP Subscription-Id-Type vaut 1 (END\_USER\_IMSI).

**userIdSIP** : valeur de l'AVP Subscription-Id-Data quand l'AVP Subscription-Id-Type vaut 2 (END\_USER\_SIP\_URI).

**userIdNAI** : valeur de l'AVP Subscription-Id-Data quand l'AVP Subscription-Id-Type vaut 3 (END\_USER\_NAI).

**userIdPrivate** : valeur de l'AVP Subscription-Id-Data quand l'AVP Subscription-Id-Type vaut 4 (END\_USER\_PRIVATE).

Le type est reçu dans le CCR, la recherche de l'abonné se fait en conséquence.

**serviceId** : valeur de l'AVP Service-Identifier.

**ratingGroup** : valeur de l'AVP Rating-Group.

**serviceControl** : indique si on doit retourner un 4011. 0 pour pas de contrôle et retour d'un 4011, 1 pour contrôle nécessaire, donc pas de 4011.

**grantedQuotaLevel** : indique si le quota est accordé seulement pour le service (valeur 1) ou pour tout le Rating-Group (valeur différente de 1). Quand le quota est accordé pour tout le Rating-Group, il n'y a pas d'AVP Service-Identifier dans le CCA. Tant que le quota n'est pas épuisé, il n'y a pas de CCR-U même pour un autre service du même Rating-Group.

**unitTypes** : valeur décimale dont les coefficients du développement dyadique donne la gestion (1) ou non (0) des types d'unité. Exemple pour la valeur 5 = 1\*1+0\*2+1\*4; les coefficients sont donc (1,0,1,0,0,0), les types d'unité 0 (TIME) et 2 (TOTAL\_OCTETS) sont gérés, les types d'unité 1 (MONEY), 3 (INPUT\_OCTETS), 4 (OUTPUT\_OCTETS) et 5 (SERVICE\_SPECIFIC\_UNITS) sont ignorés. La mise à jour du crédit ne se fera que pour les types d'unité gérés par le service. Les autres types d'unité présents dans l'AVP Used-Service-Unit seront ignorés. Dès qu'un des types d'unité est en fin de crédit, l'AVP Final-Unit-Indication est envoyé dans le CCA.

**initialCreditTime** : Crédit en secondes de l'abonné pour le service.

**grantedSlideTime** : Quota en secondes accordé à l'abonné pour le service.

**initialCreditMoney** : Crédit en unités monétaires de l'abonné pour le service.

**grantedSlideMoney** : Quota en unités monétaires accordé à l'abonné pour le service.

**initialCreditTotalOctets** : Crédit en octets (quelque soit la direction) de l'abonné pour le service.

**grantedSlideTotalOctets** : Quota en octets (quelque soit la direction) accordé à l'abonné pour le service.

**initialCreditInputOctets** : Crédit en octets (entrants) de l'abonné pour le service.

**grantedSlideInputOctets** : Quota en octets (entrants) accordé à l'abonné pour le service.

**initialCreditOutputOctets** : Crédit en octets (sortants) de l'abonné pour le service.

**grantedSlideOutputOctets** : Quota en octets (sortants) accordé à l'abonné pour le service.

**initialCreditServiceSpecificUnits** : Crédit en unités spécifiques de l'abonné pour le service.

**grantedSlideServiceSpecificUnits** : Quota en unités spécifiques accordé à l'abonné pour le service.

**validityTime** : valeur de l'AVP Validity-Time. Quand les sessions ne sont pas gérées, cet AVP est envoyé systèmatiquement dans le MSCC du CCA sauf pour un CCA TERMINATE. Quand les sessions sont gérées, cet AVP est envoyé uniquement dans le CCA du premier CCR contenant l'identifiant de service.

**finalUnitAction** : contient la valeur de l'AVP Final-Unit-Action : 0 (TERMINATE) ou 1 (REDIRECT). La valeur 2 (RESTRICT\_ACCESS) n'est pas implémentée : pas de gestion de l'AVP Restriction-Filter-Rule. La valeur 1 est implémentée : gestion de AVP groupé Redirect-Server.

**redirectAddressType** : valeur de l'AVP Redirect-Address-Type sous-AVP de Redirect-Server.

**redirectServerAddress** : valeur de l'AVP Redirect-Server-Address sous-AVP de Redirect-Server.

**timeQuotaThreshold** : seuil en secondes retourné dans l'AVP Time-Quota-Threshold du CCA de 'OCS.

**volumeQuotaThreshold** : seuil en octets retourné dans l'AVP Volume-Quota-Threshold du CCA de 'OCS.

**unitQuotaThreshold** : seuil en unités spécifiques retourné dans l'AVP Unit-Quota-Threshold du CCA de 'OCS.

**quotaHoldingTime** : valeur en secondes retournée dans l'AVP Quota-Holding-Time du CCA de 'OCS.

**quotaConsumptionTime** : valeur en secondes retournée dans l'AVP Quota-Consumption-Time du CCA de 'OCS.

**triggerTypes** : valeur décimale dont les coefficients du développement dyadique donne la présence (1) ou non (0) des triggers dans l'AVP 1264 Trigger. Exemple pour la valeur 8 = 0\*1+0\*2+0\*4+1\*8; les coefficients sont donc (0,0,0,1,0,…,0), seul l'événement 3 (CHANGE\_IN\_LOCATION) est présent, les autres événements ne le sont pas. La liste des événements déterminée par cette valeur est retournée dans l'AVP Trigger du CCA de 'OCS.

Chacun de ces 6 AVP peuvent être retournés ou non dans le CCA de l'OCS selon la configuration des paramètres [TQTbyOCS], [VQTbyOCS], [UQTbyOCS], [QHTbyOCS], [QCTbyOCS], [TriggerByOCS] de MTS.

**RARdelay** : délai en secondes. 0 pour pas d'envoi.

**refillCredit** : multiplicateur, nombre de quotas accordés à ajouter à la valeur courante du crédit de l'abonné.. La mise à jour du crédit ne se fera que pour les types d'unité gérés par le service. Exemple : si unitTypes vaut 1 (Time), grantedSlideTime vaut 60 secondes et refillCredit vaut 3, 180 secondes seront rechargées avant d'envoyer le RAR.

**ASRdelay** : délai en secondes. 0 pour pas d'envoi.

Quand on clique sur le bouton "Run" du testcase, le scénario fait la liste, parmi les sessions ouvertes, des RAR/ASR à envoyer en respectant le délai en secondes.

**tariffTimeChange** : au format "jj/mm/aaa hh:mm:ss:sss" contient l'horodatage du prochain changement de tarif. Si la date est passée, l'AVP Tariff-Time-Change ne sera pas envoyé. La valeur maximale est "07/02/2036 06:28:15:000".

Les colonnes suivantes ont été ajoutées pour la Gx :

**eventTriggers** : valeur décimale dont les coefficients du développement dyadique donne l'activation (1) ou non (0) des Event-Trigger. Exemple pour la valeur 5 = 1\*1+0\*2+1\*4; les coefficients sont donc (1,0,1,0,0), les Event-Trigger 0 et 2 sont présents, les Event-Trigger 1, 3 et 4 sont absents. Le CCA contiendra 2 fois l'AVP Event-Trigger, l'un avec la valeur 0 (SGSN\_CHANGE), l'autre avec la valeur 2 (RAT\_CHANGE).

**primaryEventChargingFunctionName** : valeur à mettre dans l'AVP Primary-Event-Charging-Function-Name.

**secondaryEventChargingFunctionName** : valeur à mettre dans l'AVP Secondary-Event-Charging-Function-Name.

**primaryChargingCollectionFunctionName** : valeur à mettre dans l'AVP Primary -Charging-Collection-Function-Name.

**secondaryChargingCollectionFunctionName** : valeur à mettre dans l'AVP Secondary-Charging-Collection-Function-Name.

Ces 4 AVP constituent l'AVP groupé Charging-Information. Si aucune de ces diameterURI ne sont renseignées, l'AVP Charging-Information ne sera pas envoyé.

**monitoringKey** : valeur de l'AVP Monitoring-Key sous-AVP de Usage-Monitoring-Information. Si cette valeur est vide, l'AVP Usage-Monitoring-Information ne sera pas envoyé.

**usageMonitoringLevel** : valeur de l'AVP Usage-Monitoring-Level sous-AVP de Usage-Monitoring-Information.