

Groupe de travail Réseau
Request for Comments : 5196
 Catégorie : Sur la voie de la normalisation
 Traduction Claude Brière de L'Isle

M. Lonnfors, Nokia
 K. Kiss, Nokia
 septembre 2008

Extension de capacité d'agent d'utilisateur du protocole d'initialisation de session au format de données d'informations de présence (PIDF)

Statut du présent mémoire

Le présent document spécifie un protocole Internet sur la voie de la normalisation pour la communauté de l'Internet, et appelle à des discussions et suggestions pour son amélioration. Prière de se référer à l'édition en cours des "Normes officielles des protocoles de l'Internet" (STD 1) pour connaître l'état de la normalisation et le statut de ce protocole. La distribution du présent mémoire n'est soumise à aucune restriction.

Résumé

Le format de données d'informations de présence (PIDF, *Presence Information Data Format*) définit un format de données de présence commun pour les protocoles de présence conformes au profil commun pour présence (CPP, *Common Profile for Presence*). Le présent mémoire définit une extension PIDF pour représenter les capacités d'agent d'utilisateur SIP.

Table des matières

1. Introduction.....	1
1.1 Motifs.....	2
1.2 Domaine d'application.....	2
2. Conventions.....	2
3. Extension pour indiquer les capacités d'agent d'utilisateur SIP dans les documents PIDF.....	2
3.1 Vue d'ensemble du fonctionnement.....	2
3.2 Capacités de service.....	3
3.3 Capacités d'appareil.....	8
4 Lignes directrices d'utilisation.....	9
4.1 Utilisation des éléments <supported> et <notsupported>.....	9
5. Exemples.....	9
6. Définition des schémas XML.....	10
7. Considérations relatives à l'IANA.....	16
7.1 Enregistrement de sous espace de noms d'URN pour 'urn:ietf:params:xml:ns:pidf:caps'.....	16
7.2 Enregistrement de schéma pour "urn:ietf:params:xml:schema:pidf:caps".....	16
8. Considérations sur la sécurité.....	16
9. Remerciements.....	17
10. Références.....	17
10.1 Références normatives.....	17
10.2 Références pour information.....	17
Adresse des auteurs.....	18
Déclaration complète de droits de reproduction.....	18

1. Introduction

Le profil commun de présence (CPP, *Common Profile for Presence*) [RFC3859] et le profil commun pour la messagerie instantanée (CPIM, *Common Profile for Instant Messaging*) [RFC3860] définissent des opérations et formats communs que tous les services de présence et de messagerie instantanée doivent accepter afin qu'une interopérabilité de base soit possible. Le format de base réel pour la présence est défini dans le format de document d'informations de présence (PIDF, *Presence Information Document Format*) [RFC3863]. Le PIDF a été conçu pour réduire le besoin de passerelles et pour permettre la sécurité de bout en bout des informations de présence. On a pris une approche très minimaliste pour prendre en charge de telles opérations. Afin de rendre le PIDF utilisable par différentes applications de présence, ces applications doivent généralement étendre le PIDF de base avec des mécanismes XML standard comme définis dans PIDF [RFC3863].

Le but du présent mémoire est d'introduire un mécanisme d'extension spécifique de SIP pour le PIDF qui porte les mêmes étiquettes de caractéristique de supports SIP que décrites dans la [RFC3840]. Avec cette extension, les applications de présence fondées sur SIP peuvent avoir des informations de présence plus riches et plus utilisables comparées au PIDF de base.

1.1 Motifs

Le PIDF [RFC3863] définit un élément <contact> qui peut apparaître une fois dans chaque élément <tuple>. Le contenu de l'élément <contact> code les CONTACT ADDRESS et CONTACT MEANS comme défini dans la [RFC2778]. L'élément <contact> est défini comme étant un URI de n'importe quel schéma. Dans certaines mises en œuvre, le schéma d'URI peut identifier de façon univoque le service que le couple veut décrire (par exemple, le schéma d'URI im: représente généralement le service de messagerie instantanée). Cependant, ce peut n'être pas le cas dans toutes les mises en œuvre. Par exemple dans SIP, un schéma d'URI SIP peut représenter différents types de services. Un schéma d'URI SIP peut être utilisé pour contacter des services vocaux, des services de vidéo, ou des services de messagerie. Si ce n'est pas connu par d'autres moyens, il pourrait être difficile pour les applications qui traitent les informations de présence contenant seulement des adresses de contact d'URI SIP de savoir quels services particuliers le couple a l'intention de décrire. Aussi, les observateurs qui reçoivent des informations de présence vont probablement tirer parti de l'obtention d'informations plus descriptives sur les moyens ou services particuliers de communication qui sont pris en charge par la présence.

L'extension de capacités d'agent d'utilisateur [RFC3840] définissent un ensemble d'extensions qui permettent aux agents d'utilisateur d'exprimer leurs préférences sur le traitement des demandes chez les serveurs SIP. Les mêmes informations peuvent aussi avoir une certaine valeur pour les observateurs afin qu'ils puissent prendre des décisions plus rationnelles sur la façon dont une présence devrait être contactée si un document de présence contenait ces informations.

1.2 Domaine d'application

Le présent document définit une extension PIDF, qui permet aux mises en œuvre de présence SIP de représenter les capacités d'agent d'utilisateur [RFC3840] dans les informations de présence.

Cette extension ne remplace pas les mécanismes de négociation de supports définis pour SIP (par exemple, SDP [RFC4566]). L'objet de cette extension est qu'une présence donne aux observateurs des conseils sur les préférences de la présence, ses désirs, et ses capacités de communiquer avant que les observateurs initient la communication avec la présence.

2. Conventions

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

Le présent mémoire utilise le vocabulaire défini dans les [RFC2778] et [RFC3863].

3. Extension pour indiquer les capacités d'agent d'utilisateur SIP dans les documents PIDF

Cette section présente les éléments d'extension, les attributs, leurs valeurs, et leur signification. Elle décrit aussi comment cette extension peut être encore étendue.

Cette extension est destinée à être utilisée dans le PIDF [RFC3863] et cet usage particulier est décrit ici. Cette extension peut aussi être utilisée avec d'autres documents XML si c'est approprié.

3.1 Vue d'ensemble du fonctionnement

Le présent document définit comment les caractéristiques présentées dans la [RFC3840] peuvent être fournies au titre des informations de présence. De plus, le présent mémoire inclut la définition de l'étiquette de caractéristique "type" [RFC2913], de l'étiquette de caractéristique de type de support "message" [RFC4569], et de l'étiquette de caractéristique "language" [RFC4646]. L'ajout de ces caractéristiques au PIDF signifie de les transposer en une structure de format XML.

Le modèle de données de présence [RFC4479] définit des informations de présence consistant en trois types d'éléments de données : personne, service, et appareil. Le présent mémoire suit ce modèle de sorte qu'une extension XML est définie pour décrire les capacités d'appareil et une autre pour décrire les capacités de service.

Les URI d'espace de noms pour les éléments définis par ce document sont des URN qui utilisent l'identifiant d'espace de noms "ietf" défini dans la [RFC2648] et étendu par la [RFC3688].

Quand ces espaces de noms d'extension sont agrégés avec le document PIDF, le document combiné DOIT suivre les mêmes règles générales de formatage que spécifié au paragraphe 4.1 de la [RFC3863].

3.2 Capacités de service

Les éléments qui appartiennent aux capacités de service sont utilisés pour décrire les caractéristiques dynamiques d'un service. Ces capacités sont encloses dans l'élément <servcaps> qui DEVRAIT être situé dans le document PIDF comme un élément fils de l'élément <tuple> de l'espace de noms urn:ietf:params:xml:ns:pidf [RFC3863].

L'identifiant d'espace de noms pour ces éléments est : urn:ietf:params:xml:ns:pidf:caps

3.2.1 Élément <servcaps>

L'élément racine des capacités de service est <servcaps>. L'élément racine doit toujours être présent. Cet élément peut contenir les éléments fils suivants : <audio>, <application>, <data>, <control>, <video>, <text>, <message>, <type>, <automata>, <class>, <duplex>, <description>, <event-packages>, <priority>, <methods>, <extensions>, <schemes>, <actor>, <isfocus>, et <languages> suivis par un nombre quelconque d'éléments d'extension facultatifs provenant d'autres espaces de noms.

Un élément <servcaps> peut contenir un nombre quelconque d'attributs d'extension facultatifs provenant d'autres espaces de noms.

3.2.2 Élément <audio>

L'élément <audio> indique que le service prend en charge l'audio comme type de support d'écoulement, comme défini dans la [RFC3840].

L'élément <audio> est un type booléen et n'a aucun attribut. La valeur "vrai" indique que le service prend en charge le type de support audio, et la valeur "faux" indique que le service ne prend pas en charge le type de support audio.

3.2.3 Élément <application>

L'élément <application> indique que le service prend en charge les applications comme type de support d'écoulement de flux comme défini dans la [RFC3840].

L'élément <application> est un type booléen et n'a pas d'attribut. La valeur "vrai" indique que le service prend en charge le type de support "application", et la valeur "faux" indique que le service ne prend pas en charge de type de supports "application".

3.2.4 Élément <data>

L'élément <data> indique que le service prend en charge des données comme type de support d'écoulement de flux comme défini dans la [RFC3840].

L'élément <data> est un type booléen et n'a pas d'attribut. La valeur "vrai" indique que le service prend en charge le type de support "data", et la valeur "faux" indique que le service ne prend pas en charge le type de supports "data".

3.2.5 Élément <control>

L'élément <control> indique que le service prend en charge "control" comme type de support d'écoulement de flux comme défini dans la [RFC3840].

L'élément <control> est un type booléen et n'a pas d'attribut. La valeur "vrai" indique que le service prend en charge le type de supports "control", et la valeur "faux" indique que le service ne prend pas en charge le type de supports "control".

3.2.6 Élément <video>

L'élément <video> indique que le service prend en charge la vidéo comme type de support d'écoulement de flux comme défini dans la [RFC3840].

L'élément <video> est un type booléen et n'a pas d'attribut. La valeur "vrai" indique que le service prend en charge le type de support "vidéo", et la valeur "faux" indique que le service ne prend pas en charge le type de support "vidéo".

3.2.7 Élément <text>

L'élément <text> indique que le service prend en charge le texte comme type de support d'écoulement de flux comme défini dans la [RFC3840].

L'élément <text> est un type booléen et n'a aucun attribut. La valeur "vrai" indique que le service prend en charge le type de support texte, et la valeur "faux" indique que le service ne prend pas en charge le type de support texte.

3.2.8 Élément <message>

L'élément <message> indique que le service prend en charge la messagerie comme type de support d'écoulement de flux comme défini dans la [RFC4569].

L'élément <message> est un type booléen et n'a aucun attribut. La valeur "vrai" indique que le service prend en charge le type de support "message", et la valeur "faux" indique que le service ne prend pas en charge le type de support "message".

3.2.9 Élément <type>

L'élément <type> indique un type de contenu de support MIME (c'est-à-dire, qui apparaît dans un en-tête "Content-type:" des données formatées selon MIME correspondantes) comme défini dans la [RFC2913].

L'élément <type> est du type chaîne et n'a aucun attribut. Il DOIT être une chaîne de la forme "type/sous type", où "type" et "sous type" sont définis par la spécification MIME [RFC2045]. Seules des lettres minuscules DEVRAIENT être utilisées.

3.2.10 Élément <automata>

L'élément <automata> indique si le service représente un automate (comme un serveur de messagerie vocale, un serveur de conférence, ou un appareil enregistreur) ou un humain, comme défini dans la [RFC3840].

L'élément <automata> est a type booléen et n'a aucun attribut. La valeur "vrai" indique que le service représente un automate, et la valeur "faux" indique qu'il représente un humain.

3.2.11 Élément <class>

L'élément <class> indique le réglage, professionnel ou personnel, dans lequel le service de communication est utilisé comme défini dans la [RFC3840].

L'élément <class> peut contenir deux éléments : <supported> (*pris en charge*) et <notsupported> (*non pris en charge*). Les classes qui sont prises en charge par le service peuvent être énumérées sous l'élément <supported>, et les classes qui ne sont pas prises en charge par le service peuvent être énumérées sous l'élément <notsupported>.

Les éléments <supported> et <notsupported> peuvent contenir les éléments <business> (*professionnel*) et <personal> suivis par un nombre quelconque l'éléments d'extension facultatifs provenant d'autres espaces de noms. La sémantique de professionnel et personnel est définie dans la [RFC3840] comme :

- o <business> : le service est utilisé pour des communications professionnelles.
- o <personal> : le service est utilisé pour des communications personnelles.

Toute valeur enregistrée par l'IANA pour l'arborescence d'enregistrement d'étiquette de caractéristique de support SIP comme étiquette de caractéristique de support sip.class peut être utilisée comme valeur d'un élément d'extension. Si la valeur appropriée n'est pas enregistrée, elle DEVRAIT être enregistrée comme défini dans la [RFC3840].

3.2.12 Élément <duplex>

L'élément <duplex> dit si un service de communications peut simultanément envoyer et recevoir des supports ("plein"), alterner entre envoi et réception ("demi"), seulement recevoir ("réception seule"), ou seulement envoyer ("envoi seul") comme défini dans la [RFC3840].

L'élément <duplex> peut contenir deux éléments : <supported> et <notsupported>. Les modes duplex qui sont pris en charge par le service peuvent être énumérés sous l'élément <supported>, et les modes duplex qui ne sont pas pris en charge par le service peuvent être énumérés sous l'élément <notsupported>.

Les éléments <supported> et <notsupported> peuvent contenir les éléments <plein>, <demi>, <réception seule>, et <envoi seul> suivis par un nombre quelconque d'éléments d'extension facultatifs provenant d'autres espaces de noms. La sémantique de ces éléments est définie dans la [RFC3840] comme :

- o <plein> : le service peut simultanément envoyer et recevoir des supports ;
- o <demi> : le service peut alterner entre envoyer et recevoir des supports ;
- o <réception seule> : le service peut seulement recevoir des supports ;
- o <envoi seul> : le service peut seulement envoyer des supports.

Toute valeur enregistrée par l'IANA pour l'arborescence d'enregistrement d'étiquette de caractéristique de support SIP comme étiquette de caractéristique de support sip.duplex peut être utilisée comme valeur d'un élément d'extension. Si la valeur appropriée n'est pas enregistrée, elle DEVRAIT être enregistré comme défini dans la [RFC3840].

3.2.13 Élément <description>

L'élément <description> fournit une description textuelle du service comme défini dans la [RFC3840].

L'élément <description> est du type chaîne et n'a aucun attribut.

L'élément <description> DEVRAIT être étiqueté avec l'attribut "xml:lang" pour indiquer son langage et son écriture. La spécification permet plusieurs occurrences de ces éléments de sorte que la présentité peut porter des éléments <description> dans plusieurs écritures et langages. Si aucun attribut "xml:lang" n'est fourni, la valeur par défaut est "i-default" comme défini dans la [RFC2277].

3.2.14 Élément <event-packages>

L'élément <event-packages> fait la liste des paquetages d'événement pris en charge par un service.

L'élément <event-packages> peut contenir deux éléments: <supported> et <notsupported>. Les paquetages d'événement qui sont pris en charge par le service peuvent être énumérés sous l'élément <supported>, et les paquetages d'événement qui ne sont pas pris en charge par le service peuvent être énumérés sous l'élément <notsupported>.

Les éléments <supported> et <notsupported> peuvent contenir n'importe quelles valeurs provenant du registre d'espace de noms de type d'événement SIP de l'IANA suivies par un nombre quelconque d'éléments d'extension facultatifs provenant d'autres espaces de noms. Au moment de la rédaction du présent mémoire, le registre d'espace de noms de type d'événement SIP de l'IANA inclut les paquetages suivants : <conference>, <dialog>, <kpml>, <message-summary>, <poc-settings>, <presence>, <reg>, <refer>, <Siemens-RTP-Stats>, <spirits-INDPs>, <spirits-user-prof>, et <winfo>.

3.2.15 Élément <priority>

L'élément <priority> indique les priorités d'appel que le service accepte de traiter comme défini dans la [RFC3840].

L'élément <priority> peut contenir deux éléments : <supported> et <notsupported>. Les valeurs de priorité qui sont prises en charge par le service peuvent être énumérées sous l'élément <supported>, et les valeurs de priorité qui ne sont pas prises en charge par le service peuvent être énumérées sous l'élément <notsupported>.

Les éléments <supported> et <notsupported> peuvent contenir un nombre quelconque d'éléments <lowerthan>, <higherthan>, <equals>, et <range> suivis par un nombre quelconque d'éléments d'extension facultatifs provenant d'autres espaces de noms.

3.2.15.1 Élément <lowerthan> Element

L'élément <lowerthan> a un seul attribut appelé "maxvalue". L'attribut "maxvalue" est utilisé pour donner la plus haute valeur de priorité que le service veut accepter. Toutes les valeurs égales et inférieures à cette valeur sont prises en charge.

3.2.15.2 Élément <higherthan>

L'élément <higherthan> a un seul attribut appelé "minvalue". L'attribut "minvalue" est utilisé pour donner la plus basse valeur de priorité que le service veut prendre en charge. Toutes les valeurs égales et supérieures à cette valeur sont prises en charge.

3.2.15.3 Élément <equals>

L'élément <equals> est utilisé pour indiquer la valeur exacte de priorité que le service veut traiter. L'élément <equals> a un seul attribut appelé "valeur". L'attribut "valeur" est utilisé pour indiquer la valeur exacte de priorité prise en charge.

3.2.15.4 Élément <range>

L'élément <range> est utilisé pour indiquer la gamme de priorités que le service veut traiter. L'élément <range> a deux attributs appelés "minvalue" et "maxvalue". La valeur de l'attribut "minvalue" indique la plus faible valeur de priorité prise en charge par le service, et la valeur de l'attribut "maxvalue" indique la plus forte valeur de priorité prise en charge par le service.

3.2.16 Élément <methods>

L'élément <methods> indique les méthodes SIP prises en charge par un service. Dans ce cas, "supported" signifie que le service peut recevoir des demandes avec cette méthode. Dans ce sens, il a la même connotation que le champ d'en-tête Allow comme défini dans la [RFC3840].

L'élément <methods> peut contenir deux éléments: <supported> et <notsupported>. Les méthodes qui sont prises en charge par le service peuvent être énumérées sous l'élément <supported>, et les méthodes qui ne sont pas prises en charge par le service peuvent être énumérées sous l'élément <notsupported>.

Les éléments <supported> et <notsupported> peuvent contenir toutes valeurs provenant du tableau des méthodes du registre des paramètres SIP de l'IANA suivies par un nombre quelconque d'éléments d'extension facultatifs provenant d'autres espaces de noms. Au moment de la rédaction du présent mémoire, le registre des paramètres SIP de l'IANA inclut les méthodes suivantes : <ACK>, <BYE>, <CANCEL>, <INFO>, <INVITE>, <MESSAGE>, <NOTIFY>, <OPTIONS>, <PRACK>, <PUBLISH>, <REFER>, <REGISTER>, <SUBSCRIBE>, et <UPDATE>.

3.2.17 Élément <extensions>

L'élément <extensions> est une liste des extensions SIP (dont chacune est définie par une étiquette d'option enregistrée par l'IANA) qui sont comprises par le service. Comprises, dans ce contexte, signifie que l'étiquette d'option va être incluse dans un champ d'en-tête Supported dans une demande, comme défini dans la [RFC3840].

L'élément <extensions> peut contenir deux éléments : <supported> et <notsupported>. Les extensions qui sont prises en charge par le service peuvent être énumérées sous l'élément <supported>, et les extensions qui ne sont pas prises en charge par le service peuvent être énumérées sous l'élément <notsupported>.

Les éléments <supported> et <notsupported> peuvent contenir toutes valeurs provenant du tableau des étiquettes d'option du registre des paramètres SIP de l'IANA suivies par un nombre quelconque d'éléments d'extension facultatifs provenant d'autres espaces de noms. Au moment de la rédaction du présent mémoire, le registre des paramètres SIP de l'IANA inclut les étiquettes d'option suivantes : <rel100>, <early-session>, <eventlist>, <from-change>, <gruu>, <histinfo>, <join>, <norefersub>, <path>, <precondition>, <pref>, <privacy>, <recipient-list-invite>, <recipient-list-subscribe>, <replaces>, <resource-priority>, <sdp-anat>, <sec-agree>, <tdialog>, et <timer>.

3.2.18 Élément <schemes>

L'élément <schemes> fournit l'ensemble des schémas d'URI qui sont pris en charge par un service. "Pris en charge" implique, par exemple, que le service va savoir comment traiter un URI de ce schéma dans le champ d'en-tête Contact d'une réponse redirigée, comme défini dans la [RFC3840].

L'élément <schemes> peut contenir deux éléments : <supported> et <notsupported>. Les schémas qui sont pris en charge par le service peuvent être énumérés sous l'élément <supported>, et les schémas qui ne sont pas pris en charge par le service peuvent être énumérés sous l'élément <notsupported>.

Les éléments <supported> et <notsupported> peuvent contenir un nombre quelconque d'éléments <s>, qui peuvent être utilisés pour décrire les schémas individuels pris en charge par le service.

3.2.18.1 Élément <s>

L'élément <s> est du type chaîne et est utilisé pour décrire un schéma individuel pris en charge par le service. Les valeurs qui peuvent être utilisées ici sont des noms de schéma qui sont enregistrés dans le registre de schémas d'URI de l'IANA.

3.2.19 Élément <actor>

L'élément <actor> indique le type d'entité disponible à cet URI, comme défini dans la [RFC3840].

L'élément <actor> peut contenir deux éléments: <supported> et <notsupported>. Les types "actor" qui sont pris en charge par le service peuvent être énumérés sous l'élément <supported>, et les types "actor" qui ne sont pas pris en charge par le service peut être énumérés sous l'élément <notsupported>.

Les éléments <supported> et <notsupported> peuvent contenir les éléments <principal>, <attendant>, <msg-taker>, et <information> suivis par un nombre quelconque d'éléments d'extension facultatifs provenant d'autres espaces de noms.

La sémantique de ces éléments est définie dans la [RFC3840] comme :

- o <principal> : le service assure la communication avec le principal associé au service. Ce sera souvent une personne spécifique, mais ce peut être un automate (par exemple, pour appeler un portail vocal).
- o <attendant>: le service assure la communication avec un automate ou une personne qui va agir en intermédiaire en contactant le principal associé au service, ou un substitut de celui-ci.
- o <msg-taker>: le service assure la communication avec un automate ou une personne qui va prendre les messages et les livrer au principal.
- o <information>: le service assure la communication avec un automate ou une personne qui va fournir des informations sur le principal.

Toute valeur enregistrée par l'IANA pour l'arborescence d'enregistrement d'étiquette de caractéristique de support SIP comme étiquette de caractéristique de support sip.actor peut être utilisée comme valeur d'un élément d'extension. Si la valeur appropriée n'est pas enregistrée, elle DEVRAIT être enregistrée comme défini dans la [RFC3840].

3.2.20 Élément <isfocus>

L'élément <isfocus> indique que le service est un serveur de conférence, aussi appelé un point de concentration, comme défini dans la [RFC3840].

L'élément <isfocus> est de type booléen et n'a aucun attribut. La valeur "vrai" indique que le service est un serveur de conférence, et la valeur "faux" indique que le service ne prend pas en charge les conférences.

3.2.21 Élément <languages>

L'élément <languages> indique la capacité à afficher des langages particuliers, comme défini dans la [RFC4646].

L'élément <languages> peut contenir deux éléments: <supported> et <notsupported>. Les langues qui sont prises en charge par le service peuvent être énumérées sous l'élément <supported>, et les langues qui ne sont pas prises en charge par le service peuvent être énumérées sous l'élément <notsupported>.

Les éléments <supported> et <notsupported> peuvent contenir un nombre quelconque d'éléments <l> qui peuvent être utilisés pour décrire les langages individuels pris en charge par le service.

3.2.21.1 Élément <I>

L'élément <I> est de type chaîne et est utilisé pour décrire une langue individuelle prise en charge par le service. Les valeurs qui peuvent être utilisées ici sont des sous étiquettes de langue qui sont enregistrées dans le registre des sous étiquettes de langues de l'IANA conformément à la [RFC4646].

3.3 Capacités d'appareil

Les éléments qui relèvent des capacités d'appareils sont utilisés pour décrire les caractéristiques dynamiques d'un appareil. Ces capacités sont encloses dans l'élément <devcaps>, qui DEVRAIT être situé dans le document PIDF comme élément fils de l'élément <device> de l'espace de noms urn:ietf:params:xml:ns:pidf:data-model [RFC4479].

L'identifiant d'espace de noms pour ces éléments est urn:ietf:params:xml:ns:pidf:caps

3.3.1 Élément <devcaps>

L'élément racine des capacités d'appareil est <devcaps>. L'élément racine doit toujours être présent. Cet élément peut contenir les éléments fils suivants : <mobility> et <description> suivis par un nombre quelconque d'éléments d'extension facultatifs provenant d'autres espaces de noms.

Un élément <devcaps> peut contenir un nombre quelconque d'attributs d'extension facultatifs provenant d'autres espaces de noms.

3.3.2 Élément <mobility>

L'élément <mobility> indique si l'appareil est fixe (ce qui signifie qu'il est associé à un point de contact fixe avec le réseau) ou mobile (ce qui signifie qu'il n'est pas associé à un point de contact fixe). Noter que les téléphones sans cordon sont fixes, non mobiles, sur la base de cette définition de la [RFC3840].

L'élément <mobility> peut contenir deux éléments : <supported> et <notsupported>. Les modes de mobilité qui sont pris en charge par l'appareil peuvent être énumérés sous l'élément <supported> et les modes de mobilité qui ne sont pas pris en charge par l'appareil peuvent être énumérés sous l'élément <notsupported>.

Les éléments <supported> et <notsupported> peuvent contenir des éléments <fixed> et <mobile> suivis par un nombre quelconque d'éléments d'extension facultatifs provenant d'autres espaces de noms.

La sémantique de ces éléments est définie dans la [RFC3840] comme :

- o <fixed> : l'appareil est stationnaire.
- o <mobile> : l'appareil peut bouger avec l'utilisateur.

Toute valeur enregistrée par l'IANA dans l'arborescence d'enregistrement d'étiquette de caractéristique de support SIP comme étiquette de caractéristique de support sip.mobility peut être utilisée comme valeur d'un élément d'extension. Si la valeur appropriée n'est pas enregistrée, elle DEVRAIT être enregistrée comme défini dans la [RFC3840].

3.3.3 Élément <description>

L'élément <description> donne une description textuelle de l'appareil comme défini dans la [RFC3840].

L'élément <description> est du type chaîne et n'a aucun attribut.

L'élément <description> DEVRAIT être étiqueté avec l'attribut "xml:lang" pour indiquer son langage et son écriture. La spécification permet de multiples occurrences de cet élément de sorte que la présentité peut porter des éléments <description> dans plusieurs écritures et langages. Si aucun attribut "xml:lang" n'est fourni, la valeur par défaut est "i-default", comme défini dans la [RFC2277].

4 Lignes directrices d'utilisation

L'extension Capacités d'agent d'utilisateur [RFC3840] recommande qu'un UA fournisse des informations complètes dans son prédicat contact. Cependant, il se peut que la présentité ne veuille pas publier des informations de présence qui seraient cohérentes avec l'appareil ou les capacités de service réelles (par exemple, la présentité peut ne pas vouloir indiquer qu'elle prend en charge la voix alors que le service est en fait capable de la prendre en charge). Les règles d'autorisation ou les politiques du serveur de présence peuvent limiter ou modifier les informations de présence publiées par la présentité. Aussi, combiner des informations de présence provenant de multiples sources peut résulter en la perte ou la discordance des informations.

Il est RECOMMANDÉ que les agents d'utilisateur de présence (PUA, *Presence User Agent*) qui utilisent cette extension fournissent des informations de présence aussi complètes qu'ils le peuvent. Si le PUA publie des informations sensibles en utilisant cette extension, il DEVRAIT obtenir la permission de la présentité. Les PUA peuvent indiquer les capacités explicitement prises en charge en utilisant l'élément <supported>, et les capacités qui sont explicitement non prises en charge en utilisant l'élément <notsupported>.

Il n'est pas obligatoire que les informations de présence soient cohérentes avec les capacités réelles du service ou de l'appareil. Cependant, il est de l'intérêt de la présentité d'éviter de publier de fausses informations de présence et de fournir des informations précises pour aider à minimiser les échecs d'invitations de communication. Autrement, les observateurs pourraient conclure que la communication ne peut pas être établie avec la présentité, alors qu'en réalité elle serait possible ; ou les observateurs pourraient conclure que certaines capacités de communication sont disponibles, mais en réalité une tentative d'établissement de communication va échouer en utilisant ces capacités. En tous cas, les observateurs ne devraient pas s'attendre à ce que les informations de présence représentées par cette extension soient pleinement alignées avec le service réel de la présentité ou les capacités de l'appareil. Comme expliqué au paragraphe 1.2, la présence de cette extension ne remplace pas l'utilisation de la signalisation SIP pour la négociation de capacités.

4.1 Utilisation des éléments <supported> et <notsupported>

Les PUA ne devraient ajouter des informations sous les éléments <supported> et <notsupported> que quand ils estiment que cela peut affecter la prise de décision du côté de l'observateur, c'est-à-dire, des informations qui devraient être pertinentes et valables pour l'observateur. Faire la liste de toutes les informations possibles sous <supported> et <notsupported> est rarement nécessaire.

Par exemple, si le PUA veut annoncer un service de messages qui prend en charge la méthode MESSAGE, il devrait l'ajouter sous l'élément <supported> dans l'élément <methods>. Même si le service ne prend pas d'autre méthode, il est peu probable que faire la liste de toutes les méthodes non prises en charge sous l'élément <notsupported> ait une quelconque valeur pour l'observateur.

En cas de conflit d'informations, c'est-à-dire, si le même élément fils apparaît sous les éléments <supported> et <notsupported> avec la même valeur, l'observateur peut en toute sécurité supposer que les capacités de la liste sont prises en charge sans considération de l'inclusion de la capacité sous l'élément <notsupported>.

5. Exemples

```
<?xml version="1.0" encoding="UTF-8"?>
<presence xmlns="urn:ietf:params:xml:ns:pidf"
  xmlns:caps="urn:ietf:params:xml:ns:pidf:caps"
  xmlns:mod="urn:ietf:params:xml:ns:pidf:data-model"
  entity="pres:someone@example.com">
  <tuple id="joi9877866786ua9">
    <status>
      <basic>open</basic>
    </status>
    <caps:servcaps>
      <caps:audio>true</caps:audio>
      <caps:description xml:lang="fr">
        Exemple de service
      </caps:description>
      <caps:description xml:lang="hu">
```

```

    Pe'lda szolga'ltata's
  </caps:description>
  <caps:duplex>
    <caps:supported>
      <caps:full/>
    </caps:supported>
  </caps:duplex>
  <caps:message>true</caps:message>
  <caps:methods>
    <caps:supported>
      <caps:ACK/>
      <caps:BYE/>
      <caps:INVITE/>
      <caps:MESSAGE/>
    </caps:supported>
  </caps:methods>
  <caps:priority>
    <caps:supported>
      <caps:lowerthan maxvalue="10"/>
    </caps:supported>
  </caps:priority>
  <caps:schemes>
    <caps:supported>
      <caps:s>sip</caps:s>
    </caps:supported>
  </caps:schemes>
  <caps:video>>false</caps:video>
</caps:svcaps>
<contact>sip:someone@example.com</contact>
</tuple>
<mod:device id="hgt67">
  <caps:devcaps>
    <caps:mobility>
      <caps:supported>
        <caps:mobile/>
      </caps:supported>
    </caps:mobility>
  </caps:devcaps>
  <mod:deviceID
    >urn:uuid:d27459b7-8213-4395-aa77-ed859a3e5b3a</mod:deviceID>
  </mod:device>
</presence>

```

6. Définition des schémas XML

Cette section donne les définitions de schémas XML pour les extensions définies dans le présent document. L'identifiant d'espace de noms pour ce schéma est urn:ietf:params:xml:ns:pidf:caps.

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:tns="urn:ietf:params:xml:ns:pidf:caps"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  targetNamespace="urn:ietf:params:xml:ns:pidf:caps"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">

<!-- Cette importation amène l'attribut de langage XML xml:lang-->

<xs:import namespace="http://www.w3.org/XML/1998/namespace"
  schemaLocation="http://www.w3.org/2001/xml.xsd"/>

```

```

<!-- RACINE -->
<xs:element name="servcaps" type="tns:servcapstype"/>
<xs:complexType name="servcapstype">
  <xs:sequence>
    <xs:element name="actor" type="tns:actortype" minOccurs="0"/>
    <xs:element name="application" type="tns:applicationtype"
      minOccurs="0"/>
    <xs:element name="audio" type="tns:audiotype" minOccurs="0"/>
    <xs:element name="automata" type="tns:automatatype" minOccurs="0"/>
    <xs:element name="class" type="tns:classtype" minOccurs="0"/>
    <xs:element name="control" type="tns:controltype" minOccurs="0"/>
    <xs:element name="data" type="tns:datatype" minOccurs="0"/>
    <xs:element name="description" type="tns:descriptiontype"
      minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="duplex" type="tns:duplextype" minOccurs="0"/>
    <xs:element name="event-packages" type="tns:event-packagestype" minOccurs="0"/>
    <xs:element name="extensions" type="tns:extensionstype" minOccurs="0"/>
    <xs:element name="isfocus" type="tns:isfocustype" minOccurs="0"/>
    <xs:element name="message" type="tns:messagetype" minOccurs="0"/>
    <xs:element name="methods" type="tns:methodstype" minOccurs="0"/>
    <xs:element name="languages" type="tns:languagestype" minOccurs="0"/>
    <xs:element name="priority" type="tns:prioritytype" minOccurs="0"/>
    <xs:element name="schemes" type="tns:schemestype" minOccurs="0"/>
    <xs:element name="text" type="tns:texttype" minOccurs="0"/>
    <xs:element name="type" type="tns:typetype" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="video" type="tns:videotype" minOccurs="0"/>
    <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:anyAttribute namespace="##any" processContents="lax"/>
</xs:complexType>

<xs:element name="devcaps" type="tns:devcaps"/>
<xs:complexType name="devcaps">
  <xs:sequence>
    <xs:element name="description" type="tns:descriptiontype" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="mobility" type="tns:mobilitytype" minOccurs="0"/>
    <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:anyAttribute namespace="##any" processContents="lax"/>
</xs:complexType>

<!-- AUDIO -->
<xs:simpleType name="audiotype">
  <xs:restriction base="xs:boolean"/>
</xs:simpleType>

<!-- APPLICATION -->
<xs:simpleType name="applicationtype">
  <xs:restriction base="xs:boolean"/>
</xs:simpleType>

<!-- DATA -->
<xs:simpleType name="datatype">
  <xs:restriction base="xs:boolean"/>
</xs:simpleType>

<!-- CONTROL -->
<xs:simpleType name="controltype">
  <xs:restriction base="xs:boolean"/>
</xs:simpleType>

<!-- VIDEO -->

```

```

<xs:simpleType name="videotype">
  <xs:restriction base="xs:boolean"/>
</xs:simpleType>

<!-- TEXT -->
<xs:simpleType name="texttype">
  <xs:restriction base="xs:boolean"/>
</xs:simpleType>

<!-- MESSAGE -->
<xs:simpleType name="messagetype">
  <xs:restriction base="xs:boolean"/>
</xs:simpleType>

<!-- TYPE -->
<xs:simpleType name="typetype">
  <xs:restriction base="xs:string"/>
</xs:simpleType>

<!-- AUTOMATA -->
<xs:simpleType name="automatatype">
  <xs:restriction base="xs:boolean"/>
</xs:simpleType>

<!-- CLASS -->
<xs:complexType name="classtype">
  <xs:sequence>
    <xs:element name="supported" type="tns:classtypes" minOccurs="0"/>
    <xs:element name="notsupported" type="tns:classtypes" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="classtypes">
  <xs:sequence>
    <xs:element name="business" type="xs:string" minOccurs="0"/>
    <xs:element name="personal" type="xs:string" minOccurs="0"/>
    <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<!-- DUPLEX -->
<xs:complexType name="duplextype">
  <xs:sequence>
    <xs:element name="supported" type="tns:duplextypes" minOccurs="0"/>
    <xs:element name="notsupported" type="tns:duplextypes" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="duplextypes">
  <xs:sequence>
    <xs:element name="full" type="xs:string" minOccurs="0"/>
    <xs:element name="half" type="xs:string" minOccurs="0"/>
    <xs:element name="receive-only" type="xs:string" minOccurs="0"/>
    <xs:element name="send-only" type="xs:string" minOccurs="0"/>
    <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<!-- DESCRIPTION -->
<xs:complexType name="descriptiontype">
  <xs:simpleContent>
    <xs:extension base="xs:string">
      <xs:attribute ref="xml:lang"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

```

```

</xs:simpleContent>
</xs:complexType>

<!-- EVENT-PACKAGES -->
<xs:complexType name="event-packagestype">
  <xs:sequence>
    <xs:element name="supported" type="tns:eventtypes" minOccurs="0"/>
    <xs:element name="notsupported" type="tns:eventtypes" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="eventtypes">
  <xs:sequence>
    <xs:element name="conference" type="xs:string" minOccurs="0"/>
    <xs:element name="dialog" type="xs:string" minOccurs="0"/>
    <xs:element name="kpml" type="xs:string" minOccurs="0"/>
    <xs:element name="message-summary" type="xs:string" minOccurs="0"/>
    <xs:element name="poc-settings" type="xs:string" minOccurs="0"/>
    <xs:element name="presence" type="xs:string" minOccurs="0"/>
    <xs:element name="reg" type="xs:string" minOccurs="0"/>
    <xs:element name="refer" type="xs:string" minOccurs="0"/>
    <xs:element name="Siemens-RTP-Stats" type="xs:string" minOccurs="0"/>
    <xs:element name="spirits-INDPs" type="xs:string" minOccurs="0"/>
    <xs:element name="spirits-user-prof" type="xs:string" minOccurs="0"/>
    <xs:element name="winfo" type="xs:string" minOccurs="0"/>
    <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<!-- PRIORITY -->
<xs:complexType name="prioritytype">
  <xs:sequence>
    <xs:element name="supported" type="tns:prioritytypes" minOccurs="0"/>
    <xs:element name="notsupported" type="tns:prioritytypes" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="prioritytypes">
  <xs:sequence>
    <xs:element name="equals" type="tns:equalstype" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="higherthan" type="tns:higherthantype" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="lowerthan" type="tns:lowerthantype" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="range" type="tns:rangetype" minOccurs="0" maxOccurs="unbounded"/>
    <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="lowerthantype">
  <xs:attribute name="maxvalue" type="xs:integer" use="required"/>
</xs:complexType>
<xs:complexType name="higherthantype">
  <xs:attribute name="minvalue" type="xs:integer" use="required"/>
</xs:complexType>
<xs:complexType name="equalstype">
  <xs:attribute name="value" type="xs:integer" use="required"/>
</xs:complexType>
<xs:complexType name="rangetype">
  <xs:attribute name="minvalue" type="xs:integer" use="required"/>
  <xs:attribute name="maxvalue" type="xs:integer" use="required"/>
</xs:complexType>

<!-- METHODS -->
<xs:complexType name="methodstype">
  <xs:sequence>
    <xs:element name="supported" type="tns:methodtypes" minOccurs="0"/>

```

```

<xs:element name="notsupported" type="tns:methodtypes" minOccurs="0"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="methodtypes">
<xs:sequence>
<xs:element name="ACK" type="xs:string" minOccurs="0"/>
<xs:element name="BYE" type="xs:string" minOccurs="0"/>
<xs:element name="CANCEL" type="xs:string" minOccurs="0"/>
<xs:element name="INFO" type="xs:string" minOccurs="0"/>
<xs:element name="INVITE" type="xs:string" minOccurs="0"/>
<xs:element name="MESSAGE" type="xs:string" minOccurs="0"/>
<xs:element name="NOTIFY" type="xs:string" minOccurs="0"/>
<xs:element name="OPTIONS" type="xs:string" minOccurs="0"/>
<xs:element name="PRACK" type="xs:string" minOccurs="0"/>
<xs:element name="PUBLISH" type="xs:string" minOccurs="0"/>
<xs:element name="REFER" type="xs:string" minOccurs="0"/>
<xs:element name="REGISTER" type="xs:string" minOccurs="0"/>
<xs:element name="SUBSCRIBE" type="xs:string" minOccurs="0"/>
<xs:element name="UPDATE" type="xs:string" minOccurs="0"/>
<xs:any namespace="##other" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>

<!-- EXTENSIONS -->
<xs:complexType name="extensionstype">
<xs:sequence>
<xs:element name="supported" type="tns:extensiontypes" minOccurs="0"/>
<xs:element name="notsupported" type="tns:extensiontypes" minOccurs="0"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="extensiontypes">
<xs:sequence>
<xs:element name="rel100" type="xs:string" minOccurs="0"/>
<xs:element name="early-session" type="xs:string" minOccurs="0"/>
<xs:element name="eventlist" type="xs:string" minOccurs="0"/>
<xs:element name="from-change" type="xs:string" minOccurs="0"/>
<xs:element name="gruu" type="xs:string" minOccurs="0"/>
<xs:element name="hist-info" type="xs:string" minOccurs="0"/>
<xs:element name="join" type="xs:string" minOccurs="0"/>
<xs:element name="norefersub" type="xs:string" minOccurs="0"/>
<xs:element name="path" type="xs:string" minOccurs="0"/>
<xs:element name="precondition" type="xs:string" minOccurs="0"/>
<xs:element name="pref" type="xs:string" minOccurs="0"/>
<xs:element name="privacy" type="xs:string" minOccurs="0"/>
<xs:element name="recipient-list-invite" type="xs:string" minOccurs="0"/>
<xs:element name="recipient-list-subscribe" type="xs:string" minOccurs="0"/>
<xs:element name="replaces" type="xs:string" minOccurs="0"/>
<xs:element name="resource-priority" type="xs:string" minOccurs="0"/>
<xs:element name="sdp-anat" type="xs:string" minOccurs="0"/>
<xs:element name="sec-agree" type="xs:string" minOccurs="0"/>
<xs:element name="tdialog" type="xs:string" minOccurs="0"/>
<xs:element name="timer" type="xs:string" minOccurs="0"/>
<xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>

<!-- SCHEMES -->
<xs:complexType name="schemestype">
<xs:sequence>
<xs:element name="supported" minOccurs="0">
<xs:complexType>

```

```

    <xs:sequence>
      <xs:element name="s" type="xs:string" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="notsupported" minOccurs="0">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="s" type="xs:string" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>

<!-- ACTOR -->
<xs:complexType name="actortype">
  <xs:sequence>
    <xs:element name="supported" type="tns:actortypes" minOccurs="0"/>
    <xs:element name="notsupported" type="tns:actortypes" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="actortypes">
  <xs:sequence>
    <xs:element name="attendant" type="xs:string" minOccurs="0"/>
    <xs:element name="information" type="xs:string" minOccurs="0"/>
    <xs:element name="msg-taker" type="xs:string" minOccurs="0"/>
    <xs:element name="principal" type="xs:string" minOccurs="0"/>
    <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<!-- ISFOCUS -->
<xs:simpleType name="isfocustype">
  <xs:restriction base="xs:boolean"/>
</xs:simpleType>

<!-- LANGUAGES -->
<xs:complexType name="languagestype">
  <xs:sequence>
    <xs:element name="supported" minOccurs="0">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="l" type="xs:string" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="notsupported" minOccurs="0">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="l" type="xs:string" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
</xs:complexType>

<!-- MOBILITY -->
<xs:complexType name="mobilitytype">
  <xs:sequence>
    <xs:element name="supported" type="tns:mobilitytypes" minOccurs="0"/>
    <xs:element name="notsupported" type="tns:mobilitytypes" minOccurs="0"/>
  </xs:sequence>

```

```

</xs:sequence>
</xs:complexType>
<xs:complexType name="mobilitytypes">
  <xs:sequence>
    <xs:element name="fixed" type="xs:string" minOccurs="0"/>
    <xs:element name="mobile" type="xs:string" minOccurs="0"/>
    <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
</xs:schema>

```

7. Considérations relatives à l'IANA

L'IANA a enregistré un nouvel URN d'espace de noms XML et un schéma comme défini dans la [RFC3688].

7.1 Enregistrement de sous espace de noms d'URN pour 'urn:ietf:params:xml:ns:pidf:caps'

URI : urn:ietf:params:xml:ns:pidf:caps

Description : c'est l'espace de noms XML pour les éléments XML définis par la RFC 5196 pour décrire les capacités de service et d'appareil dans le type de contenu application/pidf+xml.

Contact d'enregistreur : IETF, groupe de travail SIMPLE à <simple@ietf.org>, Mikko Lonnfors à <mikko.lonnfors@nokia.com>

XML :

DÉBUT

```

<?xml version="1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML Basic 1.0//EN"
"http://www.w3.org/TR/xhtml-basic/xhtml-basic10.dtd">
<html> xmlns="http://www.w3.org/1999/xhtml"
<head>
  <meta http-equiv="content-type" content="text/html;charset=iso-8859-1"/>
  <title>Namespace for PIDF user agent capability extension</title>
</head>
<body>
  <h1>Namespace for PIDF service capability extension</h1>
  <h2>urn:ietf:params:xml:ns:pidf:caps</h2>
  <p>
    See <a href="http://www.rfc-editor.org/rfc/rfc5196.txt">RFC 5196</a>.
  </p>
</body>
</html>
FIN

```

7.2 Enregistrement de schéma pour "urn:ietf:params:xml:schema:pidf:caps"

URI : urn:ietf:params:xml:schema:pidf:caps

Contact d'enregistreur : IESG

XML : voir la Section 6

8. Considérations sur la sécurité

Toutes les considérations sur la sécurité spécifiées dans les [RFC3859] et [RFC3863] s'appliquent au présent document. Comparé à PIDF [RFC3863], ce format de document de présence peut révéler des informations supplémentaires sur le service de l'utilisateur et les capacités de l'appareil. Donc, le PUA DEVRAIT toujours obtenir la permission de la présence quand il publie des informations sensibles en utilisant cette extension.

9. Remerciements

Les auteurs du présent document tiennent à remercier les personnes suivantes de leurs contributions et précieux commentaires : Paul Kyzivat, Jonathan Rosenberg, Markus Isomaki, Eva Leppanen, Miguel Garcia, Jari Urpalainen, et Hisham Khartabil.

10. Références

10.1 Références normatives

- [RFC2045] N. Freed et N. Borenstein, "[Extensions de messagerie Internet](#) multi-objets (MIME) Partie 1 : Format des corps de message Internet", novembre 1996. (*D. S.*, *MàJ par* [2184](#), [2231](#), [5335](#).)
- [RFC2119] S. Bradner, "[Mots clés à utiliser](#) dans les RFC pour indiquer les niveaux d'exigence", BCP 14, mars 1997. (*MàJ par* [RFC8174](#))
- [RFC2277] H. Alvestrand, "Politique de l'IETF en matière de [jeux de caractères et de langages](#)", BCP 18, janvier 1998.
- [RFC2913] G. Klyne, "[Types de contenu MIME](#) dans les expressions de caractéristiques de support", septembre 2000. (*P.S.*)
- [RFC3688] M. Mealling, "[Registre XML de l'IETF](#)", BCP 81, janvier 2004.
- [RFC3840] J. Rosenberg, H. Schulzrinne et P. Kyzivat, "[Indication des capacités d'agent d'utilisateur](#) dans le protocole d'initialisation de session (SIP)", août 2004
- [RFC3859] J. Peterson, "[Profil commun pour les services de présence](#) (CPP)", août 2004. (*P.S.*)
- [RFC3863] H. Sugano et autres, "[Format des données d'information de présence](#) (PIDF)", août 2004.
- [RFC4479] J. Rosenberg, "[Modèle de données pour Presence](#)", juillet 2006. (*P.S.*)
- [RFC4646] A. Phillips, M. Davis, "[Étiquettes d'identification des langues](#)", [BCP0047](#) septembre 2006. (*Remplacée par* [RFC5646](#))

10.2 Références pour information

- [RFC2648] R. Moats, "Espace de nom d'URN pour les documents de l'IETF", août 1999. (*Information*)
- [RFC2778] M. Day, J. Rosenberg et H. Sugano, "[Modèle pour Presence et la messagerie instantanée](#)", février 2000.
- [RFC3860] J. Peterson, "[Profil commun pour la messagerie instantanée](#) (CPIM)", août 2004. (*P.S.*)
- [RFC4566] M. Handley, V. Jacobson et C. Perkins, "SDP : [Protocole de description de session](#)", juillet 2006. (*P.S.* ; *remplacée par* [RFC8866](#))
- [RFC4569] G. Camarillo, "Enregistrement par l'autorité d'allocation des numéros de l'Internet (IANA) de l'étiquette de caractéristique de support de message", juillet 2006. (*Information*)

Adresse des auteurs

Mikko Lonnfors
Nokia
P.O. Box 321
Helsinki
Finland
téléphone : +358 71 8008000
mél : mikko.lonnfors@nokia.com

Krisztian Kiss
Nokia
313 Fairchild Dr
Mountain View, CA 94043
US
téléphone : +1 650 391 5969
mél : krisztian.kiss@nokia.com

Déclaration complète de droits de reproduction

Copyright (C) The IETF Trust (2008).

Le présent document est soumis aux droits, licences et restrictions contenus dans le BCP 78, et à www.rfc-editor.org, et sauf pour ce qui est mentionné ci-après, les auteurs conservent tous leurs droits.

Le présent document et les informations contenues sont fournis sur une base "EN L'ÉTAT" et le contributeur, l'organisation qu'il ou elle représente ou qui le/la finance (s'il en est), la INTERNET SOCIETY et la INTERNET ENGINEERING TASK FORCE déclinent toutes garanties, exprimées ou implicites, y compris mais non limitées à toute garantie que l'utilisation des informations encloses ne viole aucun droit ou aucune garantie implicite de commercialisation ou d'aptitude à un objet particulier.

Propriété intellectuelle

L'IETF ne prend pas position sur la validité et la portée de tout droit de propriété intellectuelle ou autres droits qui pourraient être revendiqués au titre de la mise en œuvre ou l'utilisation de la technologie décrite dans le présent document ou sur la mesure dans laquelle toute licence sur de tels droits pourrait être ou n'être pas disponible ; pas plus qu'elle ne prétend avoir accompli aucun effort pour identifier de tels droits. Les informations sur les procédures de l'ISOC au sujet des droits dans les documents de l'ISOC figurent dans les BCP 78 et BCP 79.

Des copies des dépôts d'IPR faites au secrétariat de l'IETF et toutes assurances de disponibilité de licences, ou le résultat de tentatives faites pour obtenir une licence ou permission générale d'utilisation de tels droits de propriété par ceux qui mettent en œuvre ou utilisent la présente spécification peuvent être obtenues sur le répertoire en ligne des IPR de l'IETF à <http://www.ietf.org/ipr>.

L'IETF invite toute partie intéressée à porter son attention sur tous copyrights, licences ou applications de licence, ou autres droits de propriété qui pourraient couvrir les technologies qui peuvent être nécessaires pour mettre en œuvre la présente norme. Prière d'adresser les informations à l'IETF à ietf-ipr@ietf.org.