Groupe de travail Réseau
Request for Comments : 4538
Catégorie : Sur la voie de la normalisation
Traduction Claude Brière de L'Isle

J. Rosenberg, Cisco Systems, Inc.

juin 2006

Autorisation de demande par identification de dialogue dans le protocole d'initialisation de session (SIP)

Résumé
La présente spécification définit le champ d'en-tête Target-Dialog *(dialogue cible)* pour le protocole d'initialisation de session (SIP, *Session Initiation Protocol*) et l'étiquette d'option correspondante, tdialog. Ce champ d'en-tête est utilisé dans les demandes qui créent des dialogues SIP. Il indique au receveur que l'envoyeur a connaissance d'un dialogue existant avec le receveur, soit parce que l'envoyeur est sur l'autre côté de ce dialogue, soit parce que il a accès aux identifiants de dialogue. Le receveur peut alors autoriser la demande sur la base de cette connaissance.

Table des matières

1. Introduction

Le protocole d'initialisation de session (SIP) [RFC3261] définit le concept de dialogue comme une relation persistante entre une paire d'agents d'utilisateur. Les dialogues fournissent un contexte, incluant des numéros de séquence, un acheminement par un mandataire, et des identifiants de dialogue. Les dialogues sont établis par la transmission de demandes SIP avec des méthodes particulières. Précisément, les demandes INVITE, REFER [RFC3515], et SUBSCRIBE [RFC3265] créent toutes des dialogues.

Quand un agent d'utilisateur reçoit une demande qui crée un dialogue, il doit décider si il autorise cette demande. Pour

certaines demandes, l'autorisation est une fonction de l'identité de l'envoyeur, de la méthode de demande, et ainsi de suite. Cependant, il y a de nombreuses situations où la décision d'autorisation d'un agent d'utilisateur dépend de si l'envoyeur de la demande est actuellement dans un dialogue avec cet agent d'utilisateur, ou de si l'envoyeur de la demande a connaissance d'un dialogue que l'agent d'utilisateur a avec une autre entité.

Un tel exemple est celui du transfert d'appel, réalisé avec REFER. Si les agents d'utilisateur A et B sont dans un dialogue INVITE, et si l'agent d'utilisateur A souhaite transférer l'agent d'utilisateur B à l'agent d'utilisateur C, l'agent d'utilisateur A doit envoyer une demande REFER à l'agent d'utilisateur B, demandant à l'agent d'utilisateur B d'envoyer une demande INVITE à l'agent d'utilisateur C. L'agent d'utilisateur B a besoin d'autoriser cette demande REFER. La propre décision d'autorisation est que l'agent d'utilisateur B devrait accepter la demande si elle vient d'un utilisateur avec lequel B a actuellement une relation de dialogue INVITE. Les mises en œuvre actuelles traitent cela en envoyant le REFER sur le même dialogue que celui en place entre les agents d'utilisateur A et B. Cependant, cette approche pose de nombreux problèmes [RFC5057]. Ces problèmes incluent des difficultés à déterminer le cycle de vie du dialogue et ses usages, ainsi qu'à déterminer quels messages sont associés à chaque usage d'application. Une meilleure approche est plutôt que l'agent d'utilisateur A envoie la demande REFER à l'agent d'utilisateur B en dehors du dialogue. Dans ce cas, il faut un moyen pour que l'agent d'utilisateur B autorise le REFER.

Un autre exemple est le cadre d'interaction d'application [RFC5629]. Dans ce cadre, les serveurs mandataires sur le chemin d'une demande SIP INVITE peuvent placer des composants d'interface d'utilisateur sur l'agent d'utilisateur qui a généré ou reçu la demande. Pour ce faire, le serveur mandataire doit envoyer une demande REFER à l'agent d'utilisateur, ciblée sur l'URI d'agent d'utilisateur d'acheminement mondial (GRUU, *Globally Routable User Agent URI*) [RFC5627], demandant à l'agent d'utilisateur d'aller chercher une ressource HTTP contenant le composant d'interface d'utilisateur. Dans un tel cas, il y a besoin d'un moyen pour que l'agent d'utilisateur autorise le REFER. Le cadre d'interaction d'application recommande que la demande soit autorisée si elle a été envoyée d'une entité qui se trouve sur le chemin du dialogue original. Cela peut être fait en incluant les identifiants de dialogue dans le REFER, ce qui prouve que l'agent d'utilisateur qui a envoyé le REFER a connaissance de ces identifiants de dialogue (cela doit bien sûr être sécurisé contre l'espionnage par les mécanismes de sips).

Un autre exemple est celui de deux agents d'utilisateur qui partagent un dialogue INVITE, et un élément sur le chemin de la demande INVITE souhaite retracer l'état de l'INVITE. Dans ce cas, il envoie une demande SUBSCRIBE au GRUU de l'agent d'utilisateur, demandant un abonnement au paquetage d'événements du dialogue. Si la demande SUBSCRIBE vient d'un élément sur le chemin de la demande INVITE, elle devrait être autorisée.

1.1 Terminologie

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].

2. Vue d'ensemble du fonctionnement

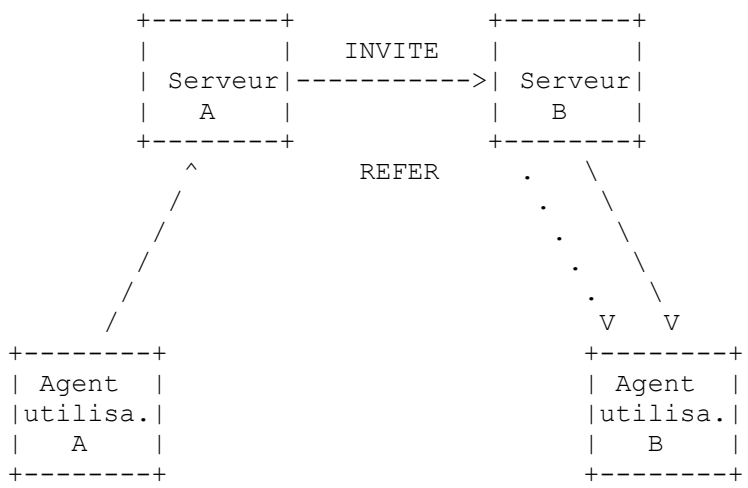

Figure 1

La Figure 1 montre le modèle de fonctionnement de base. L'agent d'utilisateur A envoie un INVITE à l'agent d'utilisateur B, traversant deux serveurs, le serveur A et le serveur B. Les deux serveurs agissent comme mandataires pour cette transaction. L'usager B envoie une réponse 200 OK à l'INVITE. Ce 200 OK inclut un champ d'en-tête Supported qui indique la prise en charge de cette spécification (par la présence de l'étiquette d'option tdialog). La réponse 200 OK établit un dialogue entre les deux agents d'utilisateur.

Ensuite, une entité qui était présente le long du chemin de la demande (le serveur A, par exemple) souhaite envoyer une demande de formation d'un dialogue (comme un REFER) à l'agent d'utilisateur A ou B (l'usager B par exemple). Ainsi, l'entité agit comme un agent d'utilisateur et envoie la demande à l'agent d'utilisateur B. Cette demande est adressée à l'URI de l'agent d'utilisateur B, que le serveur A a appris en inspectant le champ d'en-tête Contact dans le 200 OK de la demande INVITE. Si cet URI a la propriété GRUU [RFC4086] (qui peut être utilisée par tout élément sur l'Internet, comme le serveur A, pour atteindre l'instance spécifique d'agent d'utilisateur qui a généré ce 200 OK à l'INVITE) alors le mécanisme va fonctionner à travers des frontières de NAT.

Le demande générée par le serveur A va contenir un champ d'en-tête Target-Dialog. Ce champ d'en-tête contient les identifiants de dialogue pour le dialogue INVITE entre les agents d'utilisateur A et B, composés du Call-ID, d'une étiquette locale, et de l'étiquette distante. Le serveur A a appris d'inclure le champ d'en-tête Target-Dialog dans la demande REFER parce que il sait que l'agent d'utilisateur B le prend en charge.

Quand la demande arrive chez l'agent d'utilisateur B, il doit prendre une décision d'autorisation. Parce que le dialogue INVITE a été établi en utilisant un URI sips, et parce que les identifiants de dialogue sont cryptographiquement aléatoires [RFC3261], aucune entité sauf l'agent d'utilisateur A ou les mandataires sur le chemin de la demande INVITE initiale ne peut connaître les identifiants de dialogue. Donc, parce que la demande contient ces identifiants de dialogue, l'agent d'utilisateur B peut être certain que la demande vient de l'agent d'utilisateur A, des deux mandataires, ou d'une entité à laquelle l'agent d'utilisateur ou les mandataires ont donné les identifiants de dialogue. À ce titre, il autorise la demande et et effectue les actions demandées.

3. Comportement du client d'agent d'utilisateur

Un client d'agent d'utilisateur (UAC, *User Agent Client*) DEVRAIT inclure un champ d'en-tête Target-Dialog dans une demande si les conditions suivantes sont toutes vraies :

1. La demande est à envoyer en dehors de tout dialogue existant.

2. Le client d'agent d'utilisateur estime que la demande peut n'être pas autorisée par le serveur d'agent d'utilisateur sauf si le client d'agent d'utilisateur peut prouver qu'il connaît les identifiants de dialogue pour un autre dialogue. On appelle ce dialogue le dialogue cible.

3. La demande ne contient par ailleurs pas d'informations qui indiquent que l'UAC connaît ces identifiants de dialogue.

4. Le client d'agent d'utilisateur sait que le serveur d'agent d'utilisateur prend en charge le champ d'en-tête Target-Dialog. Il peut savoir cela si il a vu une demande ou une réponse provenant du serveur d'agent d'utilisateur au sein du dialogue cible qui contenait un champ d'en-tête Supported qui incluait l'étiquette d'option tdialog.

Si la quatrième condition n'est pas satisfaite, l'UAC NE DEVRAIT PAS utiliser cette spécification. À la place, si il est actuellement dans un dialogue avec le serveur d'agent d'utilisateur (UAS, *User Agent Server*) il DEVRAIT tenter d'envoyer la demande au sein du dialogue cible existant.

Voici des exemples de cas d'utilisation dans lesquels ces conditions sont satisfaites :

- Une demande REFER est envoyée en accord avec les principes de la [RFC5629]. Cette demande REFER est envoyée en dehors d'un dialogue et ne contient aucune autre information qui indique la connaissance du dialogue cible. La [RFC5629] exige aussi que le REFER ne soit envoyé que si l'UA indique la prise en charge de la spécification de dialogue cible.

- L'usager A est dans des appels séparés avec les usagers B et C. L'usager A décide de commencer un appel à trois, et donc se transforme en point de concentration [RFC4353]. L'usager B voudrait savoir qui sont les autres participants à la conférence. Il envoie donc une demande SUBSCRIBE à l'usager A (qui agit maintenant comme point de concentration)

pour avoir le paquetage d'événement de conférence [RFC4575]. Il est envoyé en dehors du dialogue existant entre l'usager B et le point de concentration, et il va être autorisé par A si l'usager B peut prouver qu'il connaît les identifiants de dialogue pour son dialogue existant avec le point de concentration. Donc, le champ d'en-tête Target-Dialog va être inclus dans le SUBSCRIBE.

Voici des exemples de cas d'utilisation dans lesquels ces conditions ne sont pas satisfaites :

- Un serveur agissant comme mandataire est un participant à un dialogue INVITE qui établit une session. Le serveur voudrait utiliser le paquetage d'événement de langage de balisage de clavier (KPML, *Keypad Markup Language*) [RFC4730] pour découvrir les fonctions des touches du clavier de l'agent d'utilisateur d'origine. Pour faire cela, il envoie une demande SUBSCRIBE. Cependant, le champ d'en-tête Event de cette demande SUBSCRIBE contient des paramètres d'événement qui indiquent le dialogue cible de l'abonnement. À ce titre, la demande peut être autorisée sans informations supplémentaires.

- Un serveur agissant comme mandataire participant à un dialogue INVITE qui établit une session. Le serveur voudrait utiliser le paquetage d'événement de dialogue [RFC4235] pour découvrir les dialogues chez l'agent d'utilisateur générateur. Pour ce faire, il envoie une demande SUBSCRIBE. Cependant, le champ d'en-tête Event de cette demande SUBSCRIBE contient des paramètres d'événement qui indiquent le dialogue cible de l'abonnement. À ce titre, la demande peut être autorisée sans informations supplémentaires.

Les spécifications qui entendent utiliser le champ d'en-tête Target-Dialog DEVRAIENT discuter les conditions spécifiques dans lesquelles il est à inclure.

En supposant qu'il soit à inclure, la valeur de la production callid dans le champ d'en-tête Target-Dialog DOIT être égale au Call-ID du dialogue cible. Le paramètre de champ d'en-tête "remote-tag" DOIT être présent et DOIT contenir l'étiquette qui va être vue comme étiquette distante du point de vue du receveur de la nouvelle demande. Le paramètre de champ d'en-tête "local-tag" DOIT être présent et DOIT contenir l'étiquette qui va être vue comme étiquette locale du point de vue du receveur de la nouvelle demande.

La demande envoyée par l'UAC DEVRAIT inclure un champ d'en-tête Require comportant l'étiquette d'option tdialog. Cette demande ne devrait en principe jamais échouer avec une réponse 420 (Mauvaise extension) parce que l'UAC n'aurait pas envoyé la demande si il pensait que l'UAS ne prend pas l'extension en charge. Si un champ d'en-tête Require n'était pas inclus, et si l'UAS ne prenait pas en charge l'extension, il rejetterait normalement la demande parce que non autorisée, probablement avec un 403. Cependant, sans le champ d'en-tête Require, l'UAC ne serait pas capable de différencier entre :
- un 403 arrivé parce que l'UAS ne comprend en fait pas le champ d'en-tête Target-Dialog (et dans ce cas le client devrait envoyer la demande dans le dialogue cible si il le peut)
- un 403 arrivé parce que l'UAS comprend le champ d'en-tête Target-Dialog, mais a choisi de ne pas autoriser la demande en dépit du fait que l'UAC a prouvé sa connaissance du dialogue cible (et dans ce cas le client ne devrait pas renvoyer la demande dans le dialogue cible, même si il le pourrait).

4. Comportement du serveur d'agent d'utilisateur

Si un serveur d'agent d'utilisateur reçoit une demande créant un dialogue et souhaite autoriser la demande, et si cette autorisation dépend de si l'envoyeur a ou non connaissance d'un dialogue existant avec l'UAS, et si des informations en dehors du champ d'en-tête Target-Dialog ne fournissent pas la preuve de cette connaissance, l'UAS DEVRAIT vérifier dans la demande l'existence du champ d'en-tête Target-Dialog. Si ce champ d'en-tête n'est pas présent, l'UAS PEUT quand même autoriser la demande par d'autres moyens.

Si le champ d'en-tête est présent, et si la valeur de la production callid, de la "remote-tag", et de la "local-tag" correspondent au Call-ID, remote tag, et local tag d'un dialogue existant, et si le dialogue auquel elles correspondent a été établi en utilisant un URI sips, l'UAS DEVRAIT autoriser la demande si il autoriserait une entité sur le chemin de la demande qui a créé ce dialogue, ou une entité de confiance pour une entité sur le chemin de la demande qui a créé ce dialogue.

Si les identifiants de dialogue correspondent, mais qu'il correspondent à un dialogue qui n'a pas été créé avec un URI sips, l'UAS PEUT autoriser la demande si il autoriserait une entité sur le chemin de la demande qui a créé ce dialogue, ou toute entité de confiance pour une entité sur le chemin de la demande qui a créé ce dialogue. Cependant, dans ce cas, un espion sur le chemin du dialogue original aurait accès aux identifiants de dialogue, et donc l'autorisation est facultative.

Si les identifiants de dialogue ne correspondent pas, ou si ils ne contiennent pas à la fois un paramètre "remote-tag" et "local-tag", le champ d'en-tête DOIT être ignoré, et l'autorisation PEUT être déterminée par d'autres moyens.

5. Comportement de mandataire

Le comportement de mandataire n'est pas affecté par la présente spécification.

6. Considérations d'extensibilité

La présente spécification dépend de ce que le client d'agent d'utilisateur sait, avant l'envoi d'une demande à un serveur d'agent d'utilisateur, si le serveur d'agent d'utilisateur prend ou non en charge le champ d'en-tête Target-Dialog. Comme exposé à la Section 3, l'UAC peut le savoir parce que il a vu une demande ou une réponse envoyée par cet UAS dans le dialogue cible qui contenait le champ d'en-tête Supported dont la valeur incluait l'étiquette d'option tdialog.

À cause de cette exigence, il est particulièrement important que les agents d'utilisateur conformes à la présente spécification incluent un champ d'en-tête Supported dans tous les dialogues qui forment des demandes et des réponses. L'inclusion des champs d'en-tête Supported dans les demandes est au niveau DEVRAIT selon la RFC 3261. La présente spécification n'altère pas cette exigence. Cependant, les mises en œuvre devraient réaliser que, sauf si l'étiquette d'option tdialog est placée dans le champ d'en-tête Supported des demandes et des réponses, cette extension ne sera probablement pas utilisée, et à la place, la demande sera probablement envoyée à nouveau dans le dialogue cible existant (en supposant que l'envoyeur est l'UA de l'autre côté du dialogue cible). À ce titre, les conditions dans lesquelles le DEVRAIT ne va pas être suivi vont être les rares cas dans lesquels l'UA ne veut pas activer l'usage de cette extension.

7. Définition du champ d'en-tête

La grammaire pour le champ d'en-tête Target-Dialog est définie comme suit :

```
Target-Dialog = "Target-Dialog" HCOLON callid *(SEMI td-param)          ; callid de la RFC 3261
td-param = remote-param / local-param / generic-param
remote-param = "remote-tag" EQUAL token
local-param  = "local-tag" EQUAL token                                  ; token et generic-param de la RFC 3261
```

Les Figures 3 et 4 sont une extension des Tableaux 2 et 3 de la [RFC3261] pour le champ d'en-tête Target-Dialog. La colonne "INF" est pour la méthode INFO [RFC2976], "PRA" est pour la méthode PRACK [RFC3262], "UPD" est pour la méthode UPDATE [RFC3311], "SUB" est pour la méthode SUBSCRIBE [RFC3265], "NOT" est pour la méthode NOTIFY [RFC3265], "MSG" est pour la méthode MESSAGE [RFC3428], "REF" est pour la méthode REFER [RFC3515], et "PUB" est pour la méthode PUBLISH [RFC3903].

Champ d'en-tête	**où**	**Mandataire**	**ACK**	**BYE**	**CAN**	**INV**	**OPT**	**REG**	**PUB**
Target-Dialog	R	-	-	-	-	o	-	-	-

Figure 3 : Méthodes permises pour Target-Dialog

Champ d'en-tête	**où**	**Mandataire**	**PRA**	**UPD**	**SUB**	**NOT**	**INF**	**MSG**	**REF**
Target-Dialog	R	-	-	-	o	-	-	-	o

Figure 4 : Méthodes permises pour Target-Dialog

8. Considérations sur la sécurité

Le champ d'en-tête Target-Dialog est utilisé pour autoriser les demandes sur la base du fait que l'envoyeur de la demande a accès à des informations auxquelles seules certaines entités ont accès. Afin qu'une telle décision d'autorisation soit sûre, deux conditions doivent être satisfaites. D'abord, qu'aucun espion ne puisse avoir accès à ces informations. Cela exige que

le dialogue SIP original soit établi en utilisant un URI sips, qui fournit TLS sur chaque bond. Avec un URI sips, seuls les agents d'utilisateur et mandataires sur le chemin de la demande seront capables de connaître les identifiants de dialogue. La seconde condition est que les identifiants de dialogue soient suffisamment aléatoires cryptographiquement pour qu'ils ne puissent pas être devinés. La RFC 3261 exige l'unicité mondiale pour le Call-ID et 32 bits d'aléa cryptographique pour chaque étiquette (il y a deux étiquettes pour un dialogue). Étant donnée la courte durée d'un dialogue normal (peut-être pas plus d'un jour) cette quantité d'aléa paraît adéquate pour empêcher une attaque par supposition. Cependant, il est important de noter que la présente spécification exige un véritable aléa cryptographique comme établi dans la [RFC4086]. Des identifiants pseudo aléatoires réduisent la probabilité de collision, mais parce que on peut les deviner, ils ne sont pas suffisants pour empêcher un attaquant d'observer une séquence d'identifiants, et de deviner le suivant, et ensuite d'utiliser la présente spécification pour lancer une attaque.

9. Relations avec In-Reply-To

La RFC 3261 définit le champ d'en-tête In-Reply-To. Il fournit une liste des identifiants d'appel pour les appels auxquels la demande actuelle fait référence ou qu'elle retourne. Il était destiné à servir à un objet similaire à celui de Reply-To dans la messagerie électronique : faciliter la construction de "trames" de conversations à une interface d'utilisateur. Target-Dialog est similaire, en ce qu'il fait aussi référence à une session antérieure. Du fait de ces similarités, il est important de comprendre leurs différences, car ces deux champs d'en-tête ne sont pas des substituts l'un de l'autre.

D'abord, In-Reply-To est destiné à un humain ou un sous programme d'interface d'utilisateur, pour lui fournir un contexte lui permettant de décider de l'objet d'un appel et si il va le prendre. Target-Dialog, est par ailleurs destiné à l'agent d'utilisateur lui-même, pour faciliter l'autorisation des demandes de session dans des cas particuliers où l'autorisation n'est pas fonction de l'utilisateur, mais plutôt des protocoles sous-jacents. Un UA va autoriser un appel contenant un Target-Dialog sur la base d'une valeur correcte du champ d'en-tête Target-Dialog.

Ensuite, Target-Dialog fait référence à un dialogue spécifique qui doit être actuellement en cours. In-Reply-To fait référence à une précédente tentative d'appel, qui n'a très probablement pas résulté en un dialogue. C'est pourquoi In-Reply-To utilise un identifiant d'appel, et Target-Dialog utilise un ensemble d'identifiants de dialogue.

Finalement, In-Reply-To implique une cause et un effet. Quand In-Reply-To est présent, il signifie que la demande est envoyée à cause de la demande précédente qui a été délivrée. Target-Dialog n'implique pas de cause et d'effet, simplement la connaissance de l'objet de l'autorisation.

10. Exemple de flux d'appel

Dans cet exemple, l'agent d'utilisateur A et l'agent d'utilisateur B établissent un dialogue initié par INVITE à travers les serveur-A et le serveur-B, qui agissent chacun comme mandataires pour le INVITE. Le serveur-B voudrait utiliser le cadre d'interaction d'application [RFC5629] pour demander que l'agent d'utilisateur A aille chercher un composant d'interface d'utilisateur HTML. Pour ce faire, il envoie une demande REFER à l'URI de A. Ce flux est montré dans la Figure 5. Les conventions de la [RFC4475] sont utilisées pour décrire la représentation des lignes de long message.

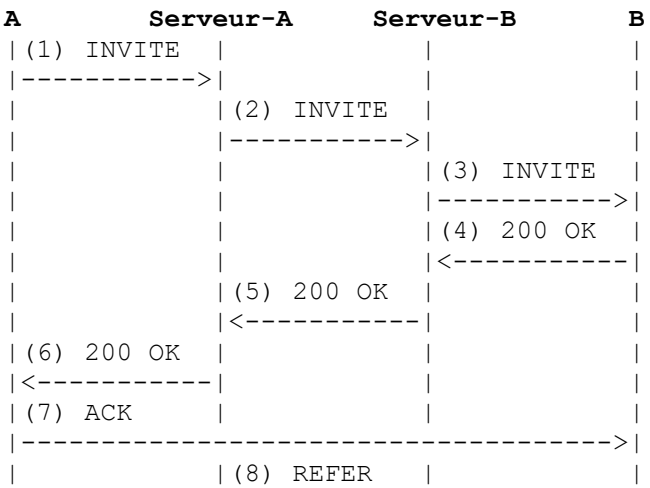

Figure 5

D'abord, l'appelant envoie un INVITE, comme le montre le message 1.

INVITE sips:B@exemple.com SIP/2.0
Via: SIP/2.0/TLS host.exemple.com;branch=z9hG4bK9zz8
From: Caller <sip:A@exemple.com>;tag=kkaz-
To: Callee <sip:B@exemple.org>
Call-ID: fa77as7dad8-sd98ajzz@host.exemple.com
CSeq: 1 INVITE
Max-Forwards: 70
Supported: tdialog
Allow: INVITE, OPTIONS, BYE, CANCEL, ACK, REFER
Accept: application/sdp, text/html
<allOneLine>
Contact: <sips:A@exemple.com;gruu;opaque=urn:uuid:f81d4fae-7dec-11d0-a765-
00a0c91e6bf6;grid=99a>;schemes="http,sip,sips"
</allOneLine>
Content-Length: ...
Content-Type: application/sdp

--SDP n'est pas montré--

Le INVITE indique que l'appelant prend en charge GRUU (noter sa présence dans le champ d'en-tête Contact de l'INVITE) et le champ d'en-tête Target-Dialog. Cet INVITE est transmis à l'appelé (messages 2-3) qui génère une réponse 200 OK. Celle-ci est retournée à l'appelant (message 4-5). Le message 5 pourrait ressembler à

SIP/2.0 200 OK
Via: SIP/2.0/TLS host.exemple.com;branch=z9hG4bK9zz8
From: Caller <sip:A@exemple.com>;tag=kkaz-
To: Callee <sip:B@exemple.org>;tag=6544
Call-ID: fa77as7dad8-sd98ajzz@host.exemple.com
CSeq: 1 INVITE
Contact: <sips:B@pc.exemple.org>
Content-Length: ...
Content-Type: application/sdp

--SDP n'est pas montré--

Dans ce cas, l'appelé ne prend pas en charge GRUU ni le champ d'en-tête Target-Dialog. L'appelant génère un ACK (message 7). Le serveur-B décide alors d'envoyer un REFER à l'usager A :

<allOneLine>
REFER sips:A@exemple.com;gruu;opaque=urn:uuid:f81d4fae-7dec-11d0-a765-00a0c91e6bf6;grid=99a SIP/2.0
</allOneLine>
Via: SIP/2.0/TLS serverB.exemple.org;branch=z9hG4bK9zz10
From: Server B <sip:serverB.exemple.org>;tag=mreysh
<allOneLine>
To: Caller <sips:A@exemple.com;gruu;opaque=urn:uuid:f81d4fae-7dec-11d0-a765-00a0c91e6bf6;grid=99a>
</allOneLine>
Target-Dialog: fa77as7dad8-sd98ajzz@host.exemple.com
;local-tag=kkaz-
;remote-tag=6544

Refer-To: http://serverB.exemple.org/ui-component.html
Call-ID: 86d65asfklzll8f7asdr@host.exemple.com
CSeq: 1 REFER
Max-Forwards: 70
Require: tdialog
Allow: INVITE, OPTIONS, BYE, CANCEL, ACK, NOTIFY
Contact: <sips:serverB.exemple.org>
Content-Length: 0

Ce REFER va être livré au serveur-A parce que il a été envoyé au GRUU. À partir de là, il est transmis à l'agent d'utilisateur A (message 9) et autorisé à cause de la présence du champ d'en-tête Target-Dialog.

11. Considérations relatives à l'IANA

La présente spécification enregistre un nouveau champ d'en-tête SIP, une nouvelle étiquette d'option en accord avec les processus de la [RFC3261], et deux nouveaux paramètres de champ d'en-tête selon les processus de la [RFC3968].

11.1 Champ d'en-tête

Numéro de RFC : RFC 4538
Nom de champ d'en-tête : Target-Dialog
Forme compacte : aucune

11.2 Paramètres de champ d'en-tête

Ce paragraphe enregistre deux paramètres de champ d'en-tête selon les processus de la [RFC3968].

11.2.1 local-tag

Champ d'en-tête : Target-Dialog
Paramètre de champ d'en-tête : local-tag
Valeurs pré définies : aucune
RFC : RFC 4538

11.2.2 remote-tag

Champ d'en-tête : Target-Dialog
Paramètre de champ d'en-tête : remote-tag
Valeurs pré définies : aucune
RFC : RFC 4538

11.3 Étiquette d'option SIP

La présente spécification enregistre une nouvelle étiquette d'option SIP selon les lignes directrices du paragraphe 27.1 de la RFC 3261.

Nom : tdialog
Description : cette étiquette d'option est utilisée pour identifier l'extension de champ d'en-tête dialogue cible. Quand utilisée dans un champ d'en-tête Require, elle implique que le receveur doit prendre en charge le champ d'en-tête Target-Dialog. Quand utilisée dans un champ d'en-tête Supported, elle implique que l'envoyeur du message la prend en charge.

12. Remerciements

Cette spécification se fonde sur un champ d'en-tête proposé par Robert Sparks dans le projet d'usage de dialogue

[RFC5057]. John Elwell a fourni des commentaires utiles.

13. Références

13.1 Références normatives

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

[RFC2976] S. Donovan, "Méthode INFO pour SIP", octobre 2000. *(P.S., Remplacée par la* RFC6086)

[RFC3261] J. Rosenberg et autres, "SIP : Protocole d'initialisation de session", juin 2002. (*Mise à jour par* 3265, 3853, 4320, 4916, 5393, 6665, 8217, 8760)

[RFC3265] A.B. Roach, "Notification d'événement spécifique du protocole d'initialisation de session (SIP)", juin 2002. (*MàJ par* RFC6446) (*Remplacée par la* RFC6665)

[RFC3262] J. Rosenberg et H. Schulzrinne, "Fiabilité des réponses provisoires dans le protocole d'initialisation de session (SIP)", juin 2002. *(P.S.)*

[RFC3311] J. Rosenberg, "Méthode UPDATE du protocole d'initialisation de session (SIP) ", octobre 2002.

[RFC3428] B. Campbell et autres, "Extension de messagerie instantanée pour le protocole d'initialisation de session (SIP)", décembre 2002.

[RFC3515] R. Sparks, "Méthode Refer du protocole d'initialisation de session (SIP)", avril 2003. *(MàJ par* RFC8217)

[RFC3903] A. Niemi, "Extension au protocole d'initialisation de session (SIP) pour la publication d'état d'événement", octobre 2004.

[RFC3968] G. Camarillo, "Registre des paramètres de champ d'en-tête de l'IANA pour le protocole d'initialisation de session (SIP)", décembre 2004. (BCP0098)

13.2 Références pour information

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

[RFC4235] J. Rosenberg et autres, "Paquetage d'événement de dialogue initié par INVITE pour le protocole d'initialisation de session (SIP)", novembre 2005. *(P.S.)*

[RFC4353] J. Rosenberg, "Cadre pour les conférences avec le protocole d'initialisation de session (SIP)", février 2006. *(Information)*

[RFC4475] R. Sparks et autres, "Messages d'essais de résistance du protocole d'initialisation de session (SIP)", mai 2006. *(Info.)*

[RFC4575] J. Rosenberg et autres, "Paquetage d'événement du protocole d'initialisation de session (SIP) pour l'état Conference", août 2006. *(P.S.)*

[RFC4730] E. Burger, M. Dolly, "Paquetage d'événement du protocole d'initialisation de session (SIP) pour stimulus par langage de balisage à pression de touche (KPML)", novembre 2006. *(P.S.)*

[RFC5057] R. Sparks, "Usages de dialogues multiples dans le protocole d'initialisation de session", novembre 2007. *(Information)*

[RFC5627 J. Rosenberg, "Obtention et utilisation des URI d'agent d'utilisateur mondialement acheminable (GRUU) dans le protocole d'initialisation de session (SIP)", octobre 2009. *(P. S.)*

[RFC5629] J. Rosenberg, "Cadre de l'interaction d'application dans le protocole d'initialisation de session (SIP)", octobre 2009. *(P.S.)*

Adresse de l'auteur

Jonathan Rosenberg
Cisco Systems
600 Lanidex Plaza
Parsippany, NJ 07054
US

téléphone : +1 973 952-5000
mél : jdrosen@cisco.com
URI: http://www.jdrosen.net

Déclaration complète de droits de reproduction

Remerciement
Le financement de la fonction d'édition des RFC est fourni par l'activité de soutien administatif (IASA) de l'IETF.