

Groupe de travail Réseau  
**Request for Comments : 3311**  
 Catégorie : En cours de normalisation

J. Rosenberg, dynamicsoft  
 septembre 2002  
 Traduction Claude Brière de L'Isle

## Méthode UPDATE du protocole d'initialisation de session (SIP)

### Statut de ce mémoire

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

### Notice de copyright

Copyright (C) The Internet Society (2002). Tous droits réservés

### Résumé

La présente spécification définit la nouvelle méthode UPDATE pour le protocole d'initialisation de session (SIP). UPDATE permet à un client de mettre à jour les paramètres d'une session (tels que l'ensemble des flux des supports et leurs codecs) mais n'a pas d'impact sur l'état d'un dialogue. Dans ce sens, elle est comme un re-INVITE, mais à la différence de re-INVITE, elle peut être envoyée avant que le INVITE initial ait été achevé. Cela la rend très utile pour mettre à jour les paramètres de session au sein des dialogues précoces.

## Table des Matières

1. Introduction.....	1
2. Terminologie.....	2
3. Vue d'ensemble du fonctionnement.....	2
4. Détermination de la prise en charge de cette extension.....	2
5. Traitement de UPDATE.....	2
5.1 Envoi d'un UPDATE.....	2
5.2 Réception d'un UPDATE.....	3
5.3 Traitement de la réponse à UPDATE.....	4
6. Comportement du mandataire.....	4
7. Définition de la méthode UPDATE.....	4
8. Exemple de flux d'appel.....	5
9. Considérations pour la sécurité.....	6
10. Considérations relatives à l'IANA.....	6
11. Notice concernant les droits de propriété intellectuelle.....	6
12. Références normatives.....	7
13. Remerciements.....	7
14. Adresse de l'auteur.....	7
15. Déclaration complète de droits de reproduction.....	7

## 1. Introduction

Le protocole d'initialisation de session (SIP, *Session Initiation Protocol*) [RFC3261] définit la méthode INVITE pour l'initialisation et la modification des sessions. Cependant, cette méthode affecte en réalité deux importants états. Elle impacte la session (les flux de support qu'établit SIP) et aussi le dialogue (l'état que définit SIP lui-même). Bien que cela soit raisonnable dans de nombreux cas, il y a des scénarios importants dans lesquels ce couplage cause des complications.

La principale difficulté se présente lorsque des aspects de la session doivent être modifiés avant qu'il ait été répondu à l'INVITE initial. Un exemple de cette situation est le "support précoce", une condition dans laquelle la session est établie, pour les besoins du transport des progrès de l'appel, mais avant que l'INVITE lui-même soit accepté. Il est important que l'appelant ou l'appelé soient capables de modifier les caractéristiques de cette session (en mettant le support précoce en garde, par exemple) avant de répondre à l'appel. Cependant, une re-INVITE ne peut pas être utilisée à cette fin, parce que la re-INVITE a un impact sur l'état du dialogue, en plus de celui de la session.

Il en résulte qu'il faut trouver une solution pour permettre à l'appelant ou à l'appelé de fournir des informations de session

à jour avant que soit générée une réponse finale à la demande INVITE initiale. La méthode UPDATE, définie ici, satisfait à ce besoin. Elle peut être envoyée par un agent d'utilisateur (UA, *user agent*) au sein d'un dialogue (précoce ou confirmé) pour mettre à jour les paramètres de session sans impact sur l'état du dialogue lui-même.

## 2. Terminologie

Dans le présent document, les mots clés "DOIT", "NE DOIT PAS", "EXIGE", "DEVRA", "NE DEVRA PAS", "DEVRAIT", "NE DEVRAIT PAS", "RECOMMANDÉ", "PEUT", et "FACULTATIF" sont à interpréter comme décrit dans le BCP 14, [RFC 2119] et indiquent les niveaux d'exigence pour les mises en œuvre conformes.

## 3. Vue d'ensemble du fonctionnement

Le fonctionnement de cette extension est très simple. L'appelant commence par une transaction INVITE, qui se fait normalement. Une fois qu'un dialogue est établi, soit précoce, soit confirmé, l'appelant peut générer une méthode UPDATE qui contient une offre du protocole de description de session (SDP, *session description protocol*) [RFC3264] afin de mettre la session à jour. La réponse à la méthode UPDATE contient la réponse. De même, une fois qu'un dialogue est établi, l'appelant peut envoyer un UPDATE avec une offre, et l'appelant place sa réponse dans les 2xx au UPDATE. Le champ d'en-tête Allow est utilisé pour indiquer la prise en charge de la méthode UPDATE. Il y a des contraintes supplémentaires sur la façon dont UPDATE peut être utilisé, fondées sur les restrictions du modèle offre/réponse.

## 4. Détermination de la prise en charge de cette extension

L'initialisation d'une session fonctionne comme spécifié dans la [RFC3261]. Cependant, un UAC conforme à la présente spécification DEVRAIT aussi inclure un champ d'en-tête Allow dans la demande INVITE, comportant la méthode UPDATE, pour indiquer sa capacité à recevoir une demande UPDATE.

Lorsque un UAS conforme à la présente spécification reçoit une demande INVITE pour un nouveau dialogue, et génère une réponse provisoire fiable qui contient SDP, cette réponse DEVRAIT contenir un champ d'en-tête Allow qui cite la méthode UPDATE. Cela informe l'appelant de la capacité de l'appelé à recevoir une demande UPDATE à tout moment. Une réponse provisoire non fiable PEUT contenir un champ d'en-tête Allow citant la méthode UPDATE, et une réponse 2xx DEVRAIT contenir un champ d'en-tête Allow qui cite la méthode UPDATE.

Les réponses sont traitées normalement selon la [RFC3261], et dans le cas de réponses provisoires fiables, conformément à la [RFC3262]. Il est important de noter qu'une réponse provisoire fiable va toujours créer un dialogue précoce chez l'UAC. La création de ce dialogue est nécessaire afin de recevoir les demandes UPDATE de l'appelé.

Si la réponse contient un champ d'en-tête Allow avec la valeur "UPDATE", l'UAC va savoir que l'appelé prend en charge UPDATE, et qu'il est permis à l'UAC de suivre les procédures du paragraphe 5.1.

## 5. Traitement de UPDATE

### 5.1 Envoi d'un UPDATE

La demande UPDATE est construite comme le serait toute autre demande au sein d'un dialogue existant, comme décrit au paragraphe 12.2.1 de la [RFC3261]. Elle PEUT être envoyée pour un dialogue aussi bien précoce que confirmé, et PEUT être envoyée par l'appelant ou l'appelé. Bien que UPDATE puisse être utilisé sur des dialogues confirmés, il est RECOMMANDÉ qu'une re-INVITE soit plutôt utilisée. Cela parce qu'on doit répondre immédiatement à une UPDATE, excluant la possibilité d'une approbation de l'utilisateur. Une telle approbation va fréquemment être nécessaire, et est possible avec une re-INVITE.

L'UAC PEUT ajouter des en-têtes facultatifs pour la demande UPDATE, comme défini aux Tableaux 1 et 2.

UPDATE est une demande de rafraîchissement de la cible. Comme spécifié dans la [RFC3261], cela signifie qu'elle peut mettre à jour la cible distante d'un dialogue. Si un UA utilise une demande ou réponse UPDATE pour modifier la cible distante alors qu'une transaction INVITE est en cours, et si c'est un UAS pour cette transaction INVITE, il DOIT placer la même valeur dans le champ d'en-tête Contact de la réponse 2xx à l'INVITE que celle qu'il a placée dans la demande ou

réponse UPDATE.

Les règles pour l'inclusion des offres et des réponses dans les messages SIP telles que définies au paragraphe 13.2.1 de la [RFC3261] s'appliquent toujours. Ces règles existent pour garantir une vue cohérente de l'état de session. Cela signifie que pour l'appelant :

- o Si le UPDATE est envoyé avant l'achèvement de la transaction INVITE initiale, et si l'INVITE initiale contenait une offre, le UPDATE peut contenir une offre si l'appelant a généré une réponse dans une réponse provisoire fiable, et si l'appelant a reçu une réponse à toute autre offre qu'il a envoyée dans un PRACK ou UPDATE, et a généré des réponses pour toute offre qu'il a reçue dans un UPDATE provenant de l'appelé.
- o Si le UPDATE est envoyé avant l'achèvement de la transaction INVITE initiale, et si l'INVITE initiale ne contenait pas une offre, le UPDATE peut contenir une offre si l'appelé a généré une offre dans une réponse provisoire fiable, et si l'UAC a généré une réponse dans le PRACK correspondant. Bien sûr, il ne peut pas envoyer un UPDATE si il n'a pas reçu de réponse aux autres offres qu'il a envoyé dans un PRACK ou UPDATE, ou n'a généré de réponse pour aucune des autres offres qu'il a reçues dans un UPDATE provenant de l'appelé.
- o Si le UPDATE est envoyé après l'achèvement de la transaction INVITE initiale, il ne peut pas contenir une offre si l'appelant a généré ou reçu des offres dans un re-INVITE ou UPDATE auquel il n'a pas été répondu.

et pour l'appelé :

- o Si le UPDATE est envoyé avant l'achèvement de la transaction INVITE, et si l'INVITE initiale contenait une offre, le UPDATE ne peut pas être envoyé avec une offre sauf si l'appelé a généré une réponse dans une réponse provisoire fiable, a reçu un PRACK pour cette réponse provisoire fiable, n'a pas reçu de demande (PRACK ou UPDATE) avec des offres auxquelles il n'a pas répondu, et n'a envoyé aucune demande UPDATE contenant des offres qui n'ont pas eu de réponse.
- o Si le UPDATE est envoyé avant l'achèvement de la transaction INVITE, et si l'INVITE initiale ne contenait pas d'offre, le UPDATE ne peut pas être envoyé avec une offre sauf si l'appelé a envoyé une offre dans une réponse provisoire fiable, a reçu une réponse dans un PRACK, et n'a reçu aucune demande UPDATE avec des offres auxquelles il n'a pas été répondu, et n'a envoyé aucune demande UPDATE contenant des offres qui n'ont pas eu de réponse.
- o Si le UPDATE est envoyé après l'achèvement de la transaction INVITE initiale, il ne peut pas être envoyé avec une offre si l'appelé a généré ou reçu des offres dans un re-INVITE ou UPDATE qui n'ont pas eu de réponse.

## 5.2 Réception d'un UPDATE

Le UPDATE est traité comme toute autre demande de rafraîchissement de cible à mi-dialogue, comme décrit au paragraphe 12.2.2 de la [RFC3261]. Si la demande est généralement acceptable, le traitement continue comme décrit ci-dessous. Ce traitement est presque identique à celui du paragraphe 14.2 de la [RFC3261], mais généralisé pour le cas de UPDATE.

Un UAS qui reçoit un UPDATE avant qu'il ait généré une réponse finale à un UPDATE précédent dans le même dialogue DOIT retourner une réponse 500 au nouvel UPDATE, et DOIT inclure un champ d'en-tête Retry-After avec une valeur choisie au hasard entre 0 et 10 secondes.

Si un UPDATE est reçu contenant une offre, et si l'UAS a généré une offre (dans un UPDATE, PRACK ou INVITE) à laquelle il n'a pas encore reçu de réponse, l'UAS DOIT rejeter le UPDATE avec une réponse 491. De même, si un UPDATE est reçu contenant une offre, et si l'UAS a reçu une offre (dans un UPDATE, PRACK, ou INVITE) à laquelle il n'a pas encore généré une réponse, l'UAS DOIT rejeter le UPDATE avec une réponse 500, et DOIT inclure un champ d'en-tête Retry-After avec une valeur choisie au hasard entre 0 et 10 secondes.

Si un UA reçoit un UPDATE pour un dialogue existant, il DOIT vérifier tout identifiant de version dans la description de session ou, si il n'y a pas d'identifiant de version, le contenu de la description de session pour voir si il a changé. Si la description de session a changé, l'UAS DOIT ajuster les paramètres de session en conséquence et générer une réponse dans les 2xx. Cependant, à la différence de re-INVITE, le UPDATE DOIT recevoir rapidement une réponse, et donc l'utilisateur ne peut généralement pas être invité à approuver les changements de session. Si l'UAS ne peut pas changer les paramètres de session sans inviter l'utilisateur, il DEVRAIT rejeter la demande avec une réponse 504. Si la nouvelle description de session n'est pas acceptable, l'UAS peut la rejeter en retournant une réponse 488 (Non acceptable ici) pour le UPDATE. Cette réponse DEVRAIT inclure un champ d'en-tête Warning.

### 5.3 Traitement de la réponse à UPDATE

Le traitement de la réponse UPDATE à l'UAC suit les règles du paragraphe 12.2.1.2 de la [RFC3261] pour une demande de rafraîchissement de cible. Une fois que ce traitement est achevé, il continue comme spécifié ci-dessous. Ce traitement est presque identique à celui du paragraphe 14.1 de la [RFC3261], mais généralisé pour UPDATE.

Si un UA reçoit une réponse finale non 2xx à un UPDATE, les paramètres de session DOIVENT rester inchangés, comme si aucun UPDATE n'avait été produit. Noter que, comme établi au paragraphe 12.2.1 de la [RFC3261], si la réponse finale non 2xx est un 481 (Cet appel/transaction n'existe pas) ou un 408 (Fin de temporisation de demande) ou si aucune réponse n'est reçue pour le UPDATE (c'est-à-dire, si une fin de temporisation est retournée par la transaction de client UPDATE) l'UAC va terminer le dialogue.

Si un UAC reçoit une réponse 491 à un UPDATE, il DEVRAIT lancer un temporisateur d'une valeur T choisie comme suit :

1. Si l'UAC est le propriétaire du Call-ID de l'identifiant de dialogue (ce qui signifie qu'il a généré la valeur) T a une valeur choisie au hasard entre 2,1 et 4 secondes en unités de 10 ms.
2. Si l'UAC n'est pas le propriétaire du Call-ID de l'identifiant de dialogue, T a une valeur choisie au hasard entre 0 et 2 secondes en unités de 10 ms.

Lorsque le temporisateur arrive à expiration, l'UAC DEVRAIT tenter une fois de plus le UPDATE, si il désire toujours que cette modification de session ait lieu. Par exemple, si l'appel a déjà été raccroché par un BYE, le UPDATE n'aurait pas lieu.

## 6. Comportement du mandataire

Le traitement d'une demande UPDATE par le mandataire est identique à celui de toute autre demande non INVITE.

## 7. Définition de la méthode UPDATE

La sémantique de la méthode UPDATE est décrite en détails ci-dessous. Cette extension ajoute une autre valeur au BNF Method décrit dans la [RFC3261] :

UPDATEm = %x55.50.44.41.54.45 ; UPDATE en majuscules  
 Method = INVITEm / ACKm / OPTIONSm / BYEm / CANCELm / REGISTERm / UPDATEm / extension-method

Le Tableau 1 étend le Tableau 2 de la [RFC3261] pour la méthode UPDATE.

Le Tableau 2 met à jour le Tableau 3 de la [RFC3261] pour la méthode UPDATE.

Champ d'en-tête	où	mandataire	UPDATE
Accept	R		o
Accept	2xx		o
Accept	415		c
Accept-Encoding	R		o
Accept-Encoding	2xx		o
Accept-Encoding	415		c
Accept-Language	R		o
Accept-Language	2xx		o
Accept-Language	415		c
Alert-Info			-
Allow	R		o
Allow	2xx		o
Allow	r		o
Allow	405		m
Allow-Events	(1)		-
Authentication-Info	2xx		o
Authorization	R		o
Call-ID	c	r	m
Call-Info		ar	o
Contact	R		m

Contact	1xx		o
Contact	2xx		m
Contact	3xx	d	o
Contact	485		o
Content-Disposition			o
Content-Encoding			o
Content-Language			o
Content-Length		ar	t
Content-Type			*
Cseq	c	r	m
Date		a	o
Error-Info	300-699	a	o
Event	(1)		-
Expires			-
From	c	r	m
In-Reply-To			-
Max-Forwards	R	amr	m
Min-Expires			-
MIME-Version			o
Organization		ar	o

(1) défini dans la [RFC3265].

**Tableau 1 : Résumé des champs d'en-tête, A--O**

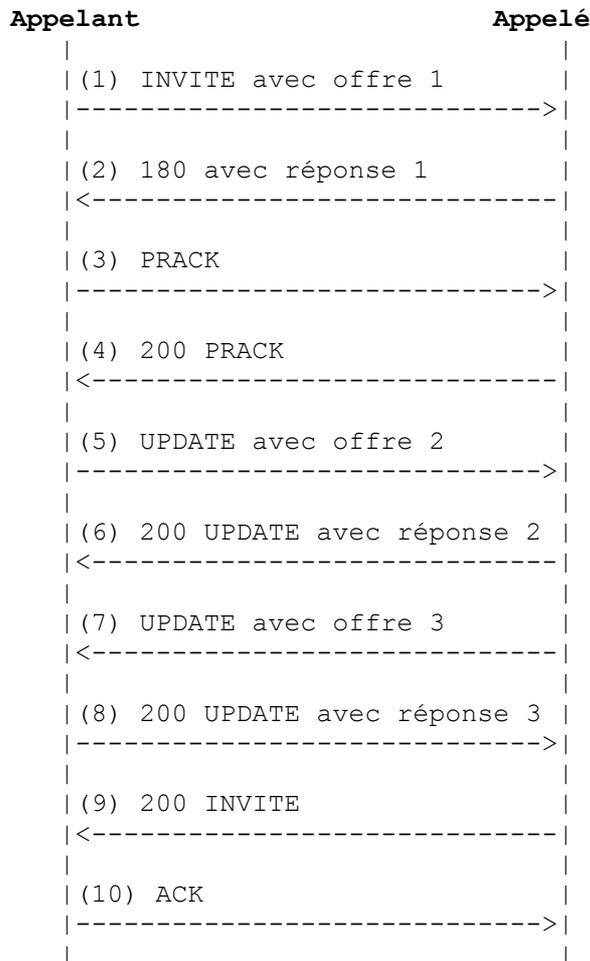
Champ d'en-tête	où	mandataire	UPDATE
Priority			-
Proxy-Authenticate	407	ar	m
Proxy-Authenticate	401	ar	o
Proxy-Authorization	R	dr	o
Proxy-Require	R	ar	o
RAck	R		-
Record-Route	R	ar	o
Record-Route	2xx,18x	mr	o
Reply-To			-
Require		ar	c
Retry-After	404,413,480,486	o	
	500,503		o
	600,603		o
Route	R	adr	c
Rseq	-		-
Server	r		o
Subject	-		-
Subscription-State	(1)		-
Supported	R		o
Supported	2xx		o
Timestamp			o
To	c	r	m
Unsupported	420		m
User-Agent			o
Via	R	amr	m
Via	rc	dr	m
Warning	r		o
WWW-Authenticate	401	ar	m
WWW-Authenticate	407	ar	o

**Tableau 2 : résumé des champs d'en-tête, P--Z.**

## 8. Exemple de flux d'appel

Cette section présente un exemple de flux d'appel utilisant la méthode UPDATE. Le flux est montré à la Figure 1. L'appelant envoie un INVITE initial (1) qui contient une offre. L'appelant génère une réponse 180 (2) avec une réponse à cette offre. Avec l'achèvement d'un échange offre/réponse, la session est établie, bien que le dialogue soit encore dans l'état

précoce. L'appelant génère un PRACK (3) pour accuser réception du 180, et il est répondu au PRACK par un 200 OK (4). L'appelant décide de mettre à jour certains aspects de la session – pour la mettre en garde, par exemple. Donc, il génère une demande UPDATE (5) avec une nouvelle offre. Il est répondu à cette offre dans la réponse 200 au UPDATE (6). Peu après, l'appelé décide de mettre à jour un aspect de la session, donc, il génère une demande UPDATE (7) avec une offre, et la réponse est envoyée dans la réponse 200 (8). Finalement, l'appelé répond à l'appel, d'où résulte une réponse 200 OK à l'INVITE (9), et ensuite un ACK (10). Ni le 200 OK à l'INVITE, ni le ACK, ne va contenir de SDP.



**Figure 1 : Flux d'appel UPDATE**

## 9. Considérations pour la sécurité

Les considérations de sécurité pour UPDATE sont identiques à celles pour re-INVITE. Il est important que l'intégrité du UPDATE soit protégée et qu'il soit authentifié comme venant de la même source que l'entité de l'autre extrémité du dialogue. La [RFC3261] discute des mécanismes de sécurité pour réaliser ces fonctions.

## 10. Considérations relatives à l'IANA

Selon le paragraphe 27.4 de la [RFC3261], cette spécification sert d'enregistrement pour la méthode de mande SIP UPDATE. Les informations à ajouter au registre sont :

RFC 3311 : Cette spécification sert de RFC d'enregistrement de la méthode de demande UPDATE.

Nom de la méthode : UPDATE

Phrase de cause : Non applicable.

## 11. Notice concernant les droits de propriété intellectuelle

Des droits de propriété intellectuelle ont été notifiés à l'IETF en ce qui concerne tout ou partie de la spécification contenue

dans le présent document. Pour plus d'informations, consulter la liste en ligne des revendications de droits.

## 12. Références normatives

- [RFC2119] S. Bradner, "[Mots clés à utiliser](#) dans les RFC pour indiquer les niveaux d'exigence", BCP 14, mars 1997.
- [RFC3261] J. Rosenberg et autres, "SIP : [Protocole d'initialisation de session](#)", juin 2002. (*Mise à jour par [RFC3265](#), [RFC3853](#), [RFC4320](#), [RFC4916](#), [RFC5393](#), [RFC6665](#)* )
- [RFC3262] J. Rosenberg et H. Schulzrinne, "[Fiabilité des réponses provisoires](#) dans le protocole d'initialisation de session (SIP)", juin 2002. (*P.S.*)
- [RFC3264] J. Rosenberg et H. Schulzrinne, "[Modèle d'offre/réponse](#) avec le protocole de description de session (SDP)", juin 2002.
- [RFC3265] A.B. Roach, "[Notification d'événement spécifique](#) du protocole d'initialisation de session (SIP)", juin 2002. (*MàJ par [RFC6446](#)*) (*Remplacée par la [RFC6665](#)*)

## 13. Remerciements

L'auteur tient à remercier Jo Hornsby, Markus Isomaki, Rohan Mahy, et Bob Penfield de leurs commentaires.

## 14. Adresse de l'auteur

Jonathan Rosenberg  
dynamicsoft  
72 Eagle Rock Avenue  
First Floor  
East Hanover, NJ 07936  
USA  
mél : [jdrosen@dynamicsoft.com](mailto:jdrosen@dynamicsoft.com)

## 15. Déclaration complète de droits de reproduction

Copyright (c) The Internet Society (2002). Tous droits réservés.

Le présent document et ses traductions peuvent être copiés et fournis aux tiers, et les travaux dérivés qui les commentent ou les expliquent ou aident à leur mise en œuvre peuvent être préparés, copiés, publiés et distribués, en tout ou partie, sans restriction d'aucune sorte, pourvu que la déclaration de copyright ci-dessus et le présent et paragraphe soient inclus dans toutes telles copies et travaux dérivés. Cependant, le présent document lui-même ne peut être modifié d'aucune façon, en particulier en retirant la notice de droits de reproduction ou les références à la Internet Society ou aux autres organisations Internet, excepté autant qu'il est nécessaire pour le besoin du développement des normes Internet, auquel cas les procédures de droits de reproduction définies dans les procédures des normes Internet doivent être suivies, ou pour les besoins de la traduction dans d'autres langues que l'anglais.

Les permissions limitées accordées ci-dessus sont perpétuelles et ne seront pas révoquées par la Internet Society, ses successeurs ou ayant droits.

Le présent document et les informations y contenues sont fournies 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 ci encloses ne violent aucun droit ou aucune garantie implicite de commercialisation ou d'aptitude à un objet particulier.

## Remerciement

Le financement de la fonction d'édition des RFC est actuellement fourni par l'Internet Society.