Groupe de travail Réseau
Request for Comments : 5626
RFC mises à jour : 3261, 3327
Catégorie : Sur la voie de la normalisation
Traduction Claude Brière de l'Isle

C. Jennings, éd., Cisco Systems
R. Mahy, éditeur
F. Audet, éditeur, Skype Labs

octobre 2009

Gestion des connexions initiés par le client dans le protocole d'initialisation de session (SIP)

Résumé
Le protocole d'initialisation de session (SIP, *Session Initiation Protocol*P) permet à des serveurs mandataires d'initier des connexions TCP ou d'envoyer des datagrammes UDP asynchrones aux agents d'utilisateur afin de livrer des demandes. Cependant, dans un grand nombre de déploiements réels, de nombreuses considérations pratiques, comme l'existence de pare-feu et de traducteurs d'adresse réseau (NAT, *Network Address Translator*) ou l'utilisation de TLS avec des certificats fournis par le serveur, empêchent les serveurs de se connecter de cette façon aux agents d'utilisateur. La présente spécification définit des comportements pour les agents d'utilisateur, registraires, et serveurs mandataires qui permettent que les demandes soient livrées sur les connexions existantes établies par l'agent d'utilisateur. Elle définit aussi des comportements de maintien en vie nécessaires pour garder ouverts les liens de NAT et spécifie l'usage de plusieurs connexions de l'agent d'utilisateur à son registraire.

Statut de ce mémoire
Ceci est un document de l'Internet sur la voie de la normalisation.

Le présent document a été produit par l'équipe d'ingénierie de l'Internet (IETF). Il représente le consensus de la communauté de l'IETF. Il a subi une révision publique et sa publication a été approuvée par le groupe de pilotage de l'ingénierie de l'Internet (IESG). Tous les documents approuvés par l'IESG ne sont pas candidats à devenir une norme de l'Internet ; voir la Section 2 de la RFC5741.

Les informations sur le statut actuel du présent document, tout errata, et comment fournir des réactions sur lui peuvent être obtenues à http://www.rfc-editor.org/info/rfc5626

Table des matières

1. Introduction....2
2. Conventions et terminologie....3
  2.1 Définitions....3
3. Vue d'ensemble....3
  3.1 Résumé du mécanisme....4
  3.2 Un seul registraire et UA....4
  3.3 Plusieurs connexions provenant d'un agent d'utilisateur....5
  3.4 Mandataires de bordure....6
  3.5 Technique de maintien en vie....7

4. Procédures d'agent d'utilisateur....7
4.1 Création d'identifiant d'instance....7
4.2 Enregistrements....8
4.3 Envoi de demandes non REGISTER....10
4.4 Maintien en vie et détection de défaillance de flux....10
4.5 Récupération de flux....12
5. Procédures de mandataire de bordure....13
5.1 Traitement des demandes Register....13
5.2 Génération de jetons de flux....13
5.3 Transmission de demandes non REGISTER....13
5.3.1 Traitement des demandes entrantes....13
5.4 Traitement de maintien en vie par le mandataire de bordure....14
6. Procédures de registraire....14
7. Procédures de mandataire d'autorité : tranmission des demandes....15
8. Traitement du maintien en vie STUN....16
8.1 Utilisation avec SigComp....17
9. Exemple de flux de messages....17
9.1 Abonnement au paquetage de configuration....17
9.2 Enregistrement....18
9.3 Appel entrant et défaillance du mandataire....20
9.4 Réenregistrement....22
9.5 Appel sortant....22
10. Grammaire....23
11. Considérations relatives à l'IANA....23
11.1 Champ d'en-tête Flow-Timer....23
11.2 Paramètre "reg-id" de champ d'en-tête Contact....24
11.3 Paramètres d'URI SIP/SIPS....24
11.4 Étiquette d'option SIP....24
11.5 Code de réponse 430 (Flux défaillant)....24
11.6 Code de réponse 439 (Le premier bond ne prend pas en charge Outbound)....24
11.7 Étiquette de caractéristique de support....25
12. Considérations sur la sécurité....25
13. Notes sur le fonctionnement des transports....26
14. Exigences....26
15. Remerciements....26
16. Références....26
16.1 Références normatives....26
16.2 Références pour information....27
Appendice A. Temps de retard d'enregistrement de flux par défaut....28
Appendice B. ABNF....28
Adresse des auteurs....28

1. Introduction

Il y a de nombreux environnements de déploiements de SIP [RFC3261] dans lesquels l'agent d'utilisateur (UA) peut former une connexion à un registraire ou mandataire mais dans lesquels les connexions dans la direction inverse vers l'UA ne sont pas possibles. Cela peut arriver pour plusieurs raisons, mais la lus probable est un NAT ou un pare-feu entre l'UA SIP et le mandataire. Beaucoup de ces appareils vont seulement permettre des connexions sortantes. La présente spécification permet à un agent d'utilisateur SIP derrière un tel pare-feu ou NAT de recevoir le trafic entrant associé aux enregistrements ou dialogues qu'il initie.

La plupart des téléphones IP et des ordinateurs personnels obtiennent leur configuration réseau dynamiquement via un protocole comme le protocole de configuration dynamique d'hôte (DHCP, *Dynamic Host Configuration Protocol*) [RFC2131]. Ces systèmes n'ont normalement pas de nom utile dans le système des noms de domaines (DNS, *Domain Name System*) [RFC1035], et ils n'ont presque jamais de nom à long terme stable du DNS qui soit approprié pour être utilisé dans le subjectAltName d'un certificat, comme exigé par la [RFC3261]. Cependant, ces systèmes peuvent quand même agir comme client de sécurité de la couche transport (TLS, *Transport Layer Security*) [RFC5246] et former des connexions sortantes à un mandataire ou registraire qui les authentifie avec un certificat de serveur. Le serveur peut authentifier l'UA en utilisant un secret partagé dans un défi de résumé (comme défini dans la Section 22 de la RFC 3261) sur cette connexion TLS. La présente spécification permet à un agent d'utilisateur SIP qui doit initier la connexion TLS de recevoir le trafic entrant associé aux enregistrements ou dialogues qu'il initie.

L'idée clé de cette spécification est que quand un UA envoie une demande REGISTER ou une demande de formation de

dialogue, le mandataire peut ensuite utiliser ce même "flux" de réseau – que ce soit un flux bidirectionnel de datagrammes UDP, une connexion TCP, ou un concept analogue dans un autre protocole de transport – pour transmettre toute demande entrante qui a besoin d'aller à cet UA dans le contexte de l'enregistrement ou dialogue.

Pour qu'un UA reçoive les demandes entrantes, l'UA doit se connecter à un serveur. Comme le serveur ne peut pas se connecter à l'UA, l'UA doit s'assurer qu'un flux est toujours actif. Cela exige que l'UA détecte quand un flux est défaillant. Comme une telle détection prend du temps et laisse une fenêtre d'opportunité pour les demandes entrantes manquées, ce mécanisme permet à l'UA de s'enregistrer sur plusieurs flux en même temps. La présente spécification définit aussi deux schémas de maintien en vie. Le mécanisme de maintien en vie est utilisé pour garder les liens de NAT frais, et pour permettre à l'UA de détecter quand un flux est défaillant.

2. Conventions et 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].

2.1 Définitions

Mandataire d'autorité : mandataire qui traite des demandes non REGISTER pour une adresse d'entretien (AOR, *Address-of-Record* (AOR) spécifique, effectue la recherche logique de serveur de localisation décrite dans la [RFC3261], et transmet ces demandes à des URI de contact spécifiques. (Dans la [RFC3261], le rôle qui est d'autorité pour les demandes REGISTER pour une AOR spécifique est un serveur d'enregistrement.)

Mandataire de bordure : un mandataire de bordure est tout mandataire qui est situé topologiquement entre l'agent d'utilisateur qui s'enregistre et le mandataire d'autorité. Le "premier" mandataire de bordure se réfère au premier mandataire de bordure rencontré quand un UA envoie une demande.

Flux : un flux est une association de couche transport entre deux hôtes, qui est représentée par l'adresse réseau et le numéro d'accès des deux extrémités et par le protocole de transport. Pour TCP, un flux est équivalent à une connexion TCP. Pour UDP un flux est un flux bidirectionnel de datagrammes entre une seule paire d'adresses et accès IP des deux homologues. Avec TCP, un flux a souvent une correspondance biunivoque avec un seul descripteur de fichier dans le système d'exploitation.

Jeton de flux : identifiant univoque d'un flux qui peut être inclus dans un identifiant de ressource universel (URI, *Uniform Resource Identifier)* SIP [RFC3986].

reg-id *(identifiant d'enregistrement)* : ceci se réfère à la valeur d'une nouvelle valeur de paramètre de champ d'en-tête pour le champ d'en-tête Contact. Quand un UA s'enregistre plusieurs fois, chaque fois pour un flux différent, chaque enregistrement concurrent obtient une valeur unique de reg-id.

instance-id *(identifiant d'instance)* : la présente spécification utilise le terme de "instance-id" pour se référer à la valeur de l'étiquette de caractéristique de support "sip.instance" qui apparaît comme un paramètre de champ d'en-tête Contact "+sip.instance". C'est un nom de ressource universel (URN, *Uniform Resource Name*) qui identifie de façon univoque cette instance d'UA spécifique.

Paramètre "ob" : c'est un paramètre d'URI SIP qui a une signification différente selon le contexte. Dans une valeur du champ d'en-tête Path, il est utilisé par le premier mandataire de bordure pour indiquer qu'un jeton de flux a été ajouté à l'URI. Dans une valeur de champ d'en-tête Contact ou Route, il indique que l'UA voudrait que d'autres demandes dans le même dialogue soient acheminées sur le même flux.

Ensemble de mandataires sortants : ensemble d'URI SIP qui représente chacun des mandataires sortants (souvent des mandataires de bordure) avec lesquels l'UA va tenter de maintenir un flux direct. Le premier URI dans l'ensemble est souvent appelé le mandataire sortant principal et le second le mandataire sortant secondaire. Il n'y a pas de différence entre les URI de cet ensemble, et la terminologie principal/secondaire n'implique pas qu'un est préféré à l'autre.

3. Vue d'ensemble

Les mécanismes définis dans le présent document sont utiles dans plusieurs scénarios discutés plus loin, incluant le simple

registraire et mandataire colocalisés, un agent d'utilisateur désirant plusieurs connexions à une ressource (pour la redondance, par exemple) et un système qui utilise des mandataires de bordure.

Toute cette section est non normative.

3.1 Résumé du mécanisme

Chaque UA a un unique instance-id qui reste le même pour cet UA même si l'UA réamorce ou est réalimenté. Chaque UA peut s'enregistrer plusieurs fois sur différents flux pour la même AOR SIP pour réaliser une haute fiabilité. Chaque enregistrement inclut l'instance-id pour l'UA et une étiquette reg-id qui est différente pour chaque flux. Le registraire peut utiliser le instance-id pour reconnaître que deux enregistrements différents correspondent au même UA. Le registraire peut utiliser l'étiquette reg-id pour reconnaître si un UA a créé un nouveau flux ou refraîchit ou remplace un vieux, éventuellement après un réamorçage ou une défaillance du réseau.

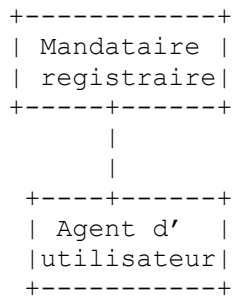
Quand un mandataire va acheminer un message à un UA pour lequel il a un lien, il peut utiliser un des flux sur lesquels un enregistrement réussi a été réalisé. Un échec de livraison d'une demande sur un flux particulier peut être réessayé sur un autre flux. Les mandataires peuvent déterminer quels flux vont au même UA en comparant le instance-id. Les mandataires peuvent dire qu'un flux remplace un flux précédemment abandonné en regardant son reg-id.

Quand il envoie une demande de formation de dialogue, un UA peut aussi demander à son premier mandataire de bordure d'acheminer les demandes suivantes de ce dialogue sur le même flux. Ceci est nécessaire, que l'UA soit enregistré ou non.

Les UA utilisent un simple message périodique comme mécanisme de maintien en vie de leur flux au mandataire ou registraire. Pour les transports en mode connexion comme TCP, ceci se fonde sur des séquences de retour chariot et saut à la ligne (CRLF) tandis que pour les transports qui ne sont pas en mode connexion, ceci est réalisé en utilisant un profil d'usage spécifique de SIP d'utilitaires de traversée de session pour les NAT (STUN, *Session Traversal Utilities for NAT*) [RFC5389].

3.2 Un seul registraire et UA

Dans la topologie montrée ci-dessous, un seul serveur agit à la fois comme registraire et mandataire.



Les agents d'utilisateur qui forment seulement un flux continuent de s'enregistrer normalement mais incluent le instance-id comme décrit au paragraphe 4.1. L'UA inclut aussi un paramètre de champ d'en-tête Contact "reg-id" qui est utilisé pour permettre au registraire de détecter et éviter de garder des contacts invalides quand un UA réamorce ou se reconnecte après que sa vieille connexion a eu une défaillance pour une raison quelconque.

Pour être clair, voici un exemple. L'UA de Bob crée un nouveau flux TCP vers le registraire et envoie la demande REGISTER suivante :

```
REGISTER sip:exemple.com SIP/2.0
Via: SIP/2.0/TCP 192.0.2.2;branch=z9hG4bK-bad0ce-11-1036
Max-Forwards: 70
From: Bob <sip:bob@exemple.com>;tag=d879h76
To: Bob <sip:bob@exemple.com>
Call-ID: 8921348ju72je840.204
CSeq: 1 REGISTER
Supported: path, outbound
Contact: <sip:line1@192.0.2.2;transport=tcp>; reg-id=1;
 ;+sip.instance="<urn:uuid:00000000-0000-1000-8000-000A95A0E128>"
Content-Length: 0
```

Le registraire défie cet enregistrement pour authentifier Bob. Quand le registraire ajoute une entrée pour ce contact sous l'AOR pour Bob, le registraire garde aussi trace de la connexion sur laquelle il a reçu cet enregistrement.

Le registraire sauvegarde l'instance-id ("urn:uuid:00000000-0000-1000-8000-000A95A0E128") et reg-id ("1") avec le reste du champ d'en-tête Contact. Si le instance-id et reg-id sont les mêmes que dans un enregistrement précédent pour la même AOR, le registraire remplace l'ancien URI Contact et informations de flux. Cela permet à un UA qui a réamorcé de remplacer son précédent enregistrement pour chaque flux avec un impact minimal sur la charge globale du système.

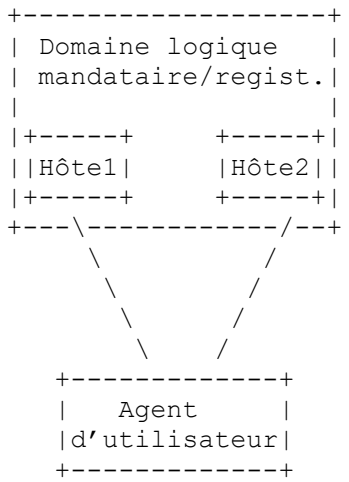
Quand Alice envoie une demande à Bob, son mandataire d'autorité choisit l'ensemble cible. Le mandataire transmet la demande aux éléments dans l'ensemble cible sur la base de la politique du mandataire. Le mandataire examine l'ensemble cible et utilise le instance-id pour comprendre si deux cibles finissent par acheminer sur le même UA. Quand le mandataire va transmettre une demande à une cible donnée, il cherche et trouve les flux sur lesquels il a reçu l'enregistrement. Le mandataire transmet alors la demande sur un flux existant, au lieu de résoudre l'URI de contact en utilisant les procédures de la [RFC3263] et d'essayer de former un nouveau flux pour ce contact.

Comme décrit au paragraphe suivant, si le mandataire a plusieurs flux qui vont tous à cet UA, le mandataire peut choisir un de ces liens d'enregistrement pour cette AOR qui a le même instance-id que l'UA choisi.

3.3 Plusieurs connexions provenant d'un agent d'utilisateur

Il y a diverses façons de déployer SIP pour construire un système fiable et adaptable. Ce paragraphe discute une de ces conceptions possibles avec les mécanismes de la présente spécification. D'autres conceptions sont aussi possibles.

Dans l'exemple de système ci-dessous, le mandataire/registraire logique sortant pour le domaine fonctionne sur deux hôtes qui partagent l'état approprié et peuvent tous deux fournir la fonction de registraire et mandataire sortant pour le domaine. L'UA va former des connexions aux deux hôtes physiques qui peuvent effectuer la fonction de mandataire/registraire d'autorité pour le domaine. La fiabilité est réalisée en faisant que l'UA forme deux connexions TCP au domaine.



L'UA est configuré avec plusieurs URI d'enregistrement de mandataire sortant. Ces URI sont configurés dans l'UA parce que le mécanisme normal va configurer l'adresse de mandataire et l'AOR dans l'UA. Si l'AOR est alice@exemple.com, l'ensemble de mandataires sortants pourrait ressembler à quelque chose comme "sip:principal.exemple.com" et "sip: secondaire.exemple.com". Noter que chaque URI dans l'ensemble de mandataires sortants pourrait se résoudre en plusieurs hôtes physiques différents. Le domaine administratif qui a créé ces URI devrait s'assurer que les deux URI se résolvent en des hôtes distincts. Ces URI sont traités en accord avec les règles normales de traitement de SIP, de sorte que des mécanismes comme le SRV DNS [RFC2782] peuvent être utilisés pour faire l'équilibrage de charge à travers un groupe de mandataires. L'approche du présent document n'empêche pas de futures extensions, comme le cadre de configuration d'UA SIP [RFC6080], d'ajouter d'autres façon pour qu'un agent d'utilisateur découvre son ensemble de mandataires sortants.

Le domaine aussi a besoin de s'assurer qu'une demande pour l'UA envoyée de Hôte1 ou Hôte2 est envoyée à travers le flux approprié à l'UA. Le domaine pourrait choisir d'utiliser l'approche de l'en-tête Path (comme décrit au paragraphe suivant) pour mémoriser ces informations d'acheminement interne sur Hôte1 ou Hôte2.

Quand un seul serveur a une défaillance, tous les UA qui ont un flux qui les traverse vont détecter une défaillance de flux et vont essayer de se reconnecter. Cela peut causer de lourdes charges sur le serveur. Quand un grand nombre d'hôtes se reconnectent presque simultanément, c'est appelé le problème de l'avalanche de redémarrages, et est discuté au paragraphe 4.5. Les multiples flux à de nombreux serveurs aident à réduire la charge causée par l'avalanche de redémarrages. Si un UA a plusieurs flux, et qu'un des serveurs a une défaillance, l'UA retarde d'une durée recommandée

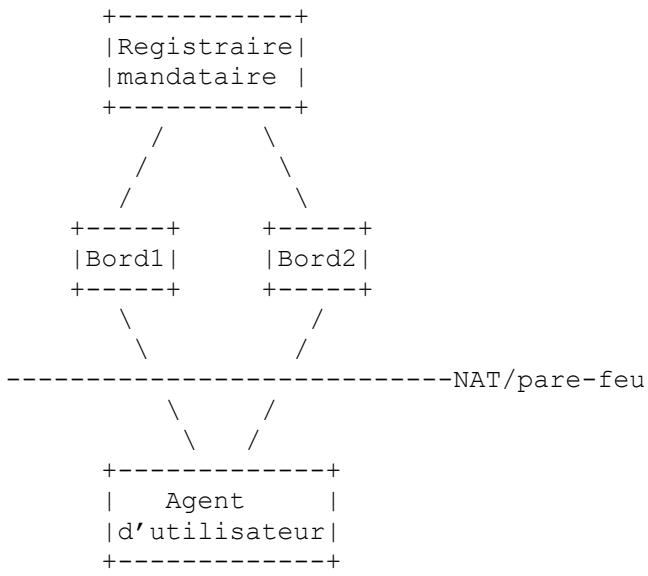
avant d'essayer de former une nouvelle connexion pour remplacer le flux au serveur défaillant. En étalant le temps utilisé pour tous les UA pour se reconnecter à un serveur, la charge sur le groupe de serveur est réduite.

L'adaptabilité est réalisée en utilisant le SRV DNS [RFC2782] pour équilibrer la charge de la connexion principale à travers un ensemble de machines qui peuvent desservir la connexion principale, et aussi en utilisant le SRV DNS pour équilibrer la charge à travers un ensemble distinct de machines qui peuvent desservir la connexion secondaire. Le déploiement exige que le DNS soit configuré avec une entrée qui se résolve en tous les hôtes principaux et une autre entrée qui se résolve en tous les hôtes secondaires. Bien que cela introduise une configuration DNS supplémentaire, l'approche fonctionne et n'exige pas d'extensions SIP supplémentaires à la [RFC3263].

Un autre motif pour maintenir plusieurs flux entre l'UA et son registraire est relative aux UA multi rattachements. De tels UA peuvent bénéficier de connexions multiples provenant d'interfaces différentes pour protéger contre la défaillance d'une liaison d'accès individuelle.

3.4 Mandataires de bordure

Certains déploiements SIP utilisent des mandataires de bordure afin que l'UA envoie le REGISTER à un mandataire de bordure qui transmet ensuite le REGISTER au registraire. Ce pourrait être un NAT ou un pare-feu entre l'UA et le mandataire de bordure.



Le mandataire de bordure inclut un en-tête Path [RFC3327] afin que quand le mandataire/registraire transmet ensuite une demande à cet UA, la demande soit acheminée à travers le mandataire de bordure.

Ces systèmes peuvent utiliser effectivement le même mécanisme que décrit dans les paragraphes précédents mais ont besoin d'utiliser l'en-tête Path. Quand le mandataire de bordure reçoit un enregistrement, il a besoin de créer une valeur d'identifiant unique pour ce flux (et non un flux suivant avec les mêmes adresses) et de mettre cet identifiant dans l'URI d'en-tête Path. Cet identifiant a deux objets. D'abord, il permet au mandataire de bordure de transposer de futures demandes sur le flux correct. Ensuite, parce que l'identifiant va seulement être retourné si l'utilisateur réussit à s'authentifier avec le registraire, il permet au mandataire de bordure de vérifier indirectement les informations d'authentication de l'utilisateur via le registraire. L'identifiant est placé dans la portion utilisateur d'un chemin lâche dans l'en-tête Path. Si l'enregistrement réussit, le mandataire de bordure a besoin de transposer les futures demandes (qui sont acheminées à la valeur de l'identifiant à partir de l'en-tête Path) au flux associé.

Le terme de mandataire de bordure est souvent utilisé pour se référer à des déploiements où le mandataire de bordure est dans le même domaine administratif que le registraire. Cependant, dans la présente spécification, on utilise le terme pour se référer à tout mandataire entre l'UA et le registraire. Par exemple, le mandataire de bordure peut être dans une entreprise qui exige son utilisation, et le registraire pourrait être d'un fournisseur de service sans relation avec l'entreprise. Sans considération de si ils sont ou non dans le même domaine administratif, la présente spécification exige que les registraires et mandataires de bordure prennent en charge le mécanisme d'en-tête Path de la [RFC3327].

3.5 Technique de maintien en vie

Le présent document décrit deux mécanismes de maintien en vie : un CRLF de maintien en vie et un maintien en vie

STUN. Chacun de ces mécanismes utilise un "ping" de maintien en vie de client à serveur et un message "pong" correspondant de serveur à client. Cette séquence de ping-pong permet au client, et facultativement au serveur, de dire si son flux est encore actif et utile pour le trafic SIP. Le serveur répond aux pings en envoyant des pongs. Si le client ne reçoit pas de pong en réponse à son ping (permettant une retransmission pour STUN comme décrit au paragraphe 4.4.2) il déclare le flux mort et ouvre un nouveau flux à sa place.

Le présent document suggère aussi des valeurs de temporisateur pour ces mécanismes de maintien en vie de client. Ces valeurs de temporisateur ont été choisies pour garder ouverts la plupart des liens de NAT et pare-feu, pour détecter les serveurs qui ne répondent pas dans les 2 minutes, et pour atténuer le problème de l'avalanche de redémarrages. Cependant, le client peut choisir des valeurs différentes de temporisateurs pour convenir à ses besoins, par exemple pour optimiser la vie de la batterie. Dans certains environnements, le serveur peut aussi garder trace du temps depuis qu'un ping a été reçu sur un flux pour deviner la probabilité que le flux soit encore utile pour livrer les messages SIP.

Quand l'UA détecte qu'un flux est défaillant ou que la définition de flux a changé, l'UA a besoin de se réenregistrer et va utiliser le mécanisme de retard décrit au paragraphe 4.5 pour fournir l'allègement d'encombrement quand un grand nombre d'agents réamorcent simultanément.

Un mécanisme de maintien en vie a besoin de garder rafraîchis les liens de NAT ; pour les connexions, il doit aussi détecter la défaillance d'une connexion, et pour les transports sans connexion, il doit détecter les défaillances de flux y compris les changements de la transposition publique de NAT. Pour les transports en mode connexion comme TCP [RFC0793] et SCTP [RFC4960], la présente spécification décrit une approche de maintien en vie fondée sur l'envoi de CRLF. Pour les transport sans connexion, comme UDP [RFC0768], la présente spécification décrit l'utilisation de STUN [RFC5389] sur le même flux que le trafic SIP pour effectuer le maintien en vie.

Les UA et mandataires sont aussi libres d'utiliser des maintiens en vie natifs du transport ; cependant, l'application peut ne pas être capable de régler ces temporisateurs connexion par connexion, et le serveur ne peut certainement pas faire d'hypothèses sur les valeurs utilisées. L'utilisation de maintiens en vie natifs du transport sort du domaine d'application de ce document.

3.5.1 Technique de maintien en vie par CRLF

Cette approche peut seulement être utilisée avec des transports en mode connexion comme TCP ou SCTP. Le client envoie périodiquement un double-CRLF (le "ping") puis attend de recevoir un seul CRLF (le "pong"). Si le client ne reçoit pas un "pong" dans un délai approprié, il considère que le flux est défaillant.

Note : l'envoi d'un CRLF sur un transport en mode connexion est rétro compatible (à cause des exigences du paragraphe 7.5 de la [RFC3261]) mais seules les mises en œuvre qui prennent en charge la présente spécification vont répondre à un "ping" par un "pong".

3.5.2 Technique de maintien en vie STUN

Cette approche peut seulement être utilisée pour les transports sans connexion, comme UDP.

Pour les transports sans connexion, une définition de flux pourrait changer parce qu'un appareil de NAT dans le chemin du réseau réamorce et que l'adresse IP publique résultant ou la transposition d'accès pour l'UA change. Pour détecter cela, des demandes STUN sont envoyées sur le même flux qu'utilisé pour le trafic SIP. Le mandataire ou registraire agit comme un serveur limité d'utilitaire de traversée de session pour NAT (STUN, *Session Traversal Utilities for NAT*) [RFC5389] sur l'accès de signalisation SIP.

Note : le mécanisme STUN est très robuste et permet la détection d'un changement d'adresse et accès IP. De nombreuses autres options ont été considérées, mais le groupe de travail SIP a choisi l'approche fondée sur STUN. Les approches qui utilisent des demandes SIP ont été abandonnées parce que beaucoup pensent que de bonnes performances et la pleine rétro compatibilité en utilisant cette méthode sont mutuellement exclusives.

4. Procédures d'agent d'utilisateur

4.1 Création d'identifiant d'instance

Chaque UA DOIT avoir un identifiant d'instance de nom de ressource universelle (URN, *Uniform Resource Name*) [RFC2141] qui identifie de façon univoque l'appareil. L'usage d'un URN fournit un nom persistant et unique pour l'instance d'UA. Il fournit aussi une façon simple de garantir l'unicité au sein de l'AOR. Cet URN DOIT être persistant à travers les remises sous tension de l'appareil. L'identifiant d'instance NE DOIT PAS changer lorsque l'appareil se déplace d'un réseau à un autre.

Un UA DEVRAIT créer un URN d'identifiant mondialement unique (UUID, *Universally Unique Identifier*) [RFC4122] comme identifiant d'instance. L'URN UUID permet un calcul non centralisé d'un URN fondé sur l'heure, des noms uniques (comme des adresses MAC) ou un générateur de nombres aléatoires.

Note : un appareil comme un "téléphone logiciel", quand il est installé pour la première fois, peut générer un UUID [RFC4122] et le sauvegarder ensuite dans une mémorisation permanente pour toutes les utilisations futures. Pour un appareil comme un "téléphone matériel", qui va seulement avoir un seul UA SIP présent, l'UUID peut inclure l'adresse MAC et être généré à tout moment parce que il est garanti qu'aucun autre UUID n'est généré en même temps sur cet appareil physique. Cela signifie que la valeur du composant horaire de l'UUID peut être choisie arbitrairement comme tout instant après que l'appareil a été fabriqué. Un temps de 0 (comme montré dans l'exemple du paragraphe 3.2) est parfaitement légal pour autant que l'appareil ne connaisse pas d'autres UUID qui aient été générés à cet instant sur cet appareil.

Si un schéma d'URN autre que UUID est utilisé, l'UA DOIT seulement utiliser des URN pour lesquels une RFC (du flux de l'IETF) définit comment l'URN spécifique doit être construit et utilisé dans le paramètre de champ d'en-tête Contact "+sip.instance" pour le comportement sortant.

Pour porter son identifiant d'instance dans les deux demandes et réponses, l'UA inclut une étiquette de caractéristique de support de "sip.instance" comme caractéristique d'UA [RFC3840]. Cette étiquette de caractéristique de support est codée dans le champ d'en-tête Contact comme le paramètre de champ d'en-tête Contact "+sip.instance". Un cas où un UA pourrait préférer omettre l'étiquette de caractéristique de support "sip.instance" est quand il fait une demande anonyme ou qu'un autre souci de confidentialité exige que l'UA ne révèle pas son identité.

Note : la [RFC3840] définit des règles d'égalité pour les paramètres de capacité de l'appelé, et conformément à cette spécification, l'étiquette de caractéristique de support "sip.instance" va être comparée par une comparaison de chaîne insensible à la casse. Cela signifie que l'URN va être encapsulé entre des crochets angulaires ("<" et ">") quand il est placé dans la valeur de chaîne entre guillemets du paramètre du champ d'en-tête Contact "+sip.instance". Les règles de correspondance sensibles à la casse s'appliquent seulement aux usages génériques définis dans les spécifications des capacités de l'appelé [RFC3840] et des préférences de l'appelant [RFC3841]. Quand l'identifiant d'instance est utilisé dans cette spécification, il est "extrait" de la valeur de l'étiquette de caractéristique de support "sip.instance". Donc, les comparaisons pour égalité sont effectuées en utilisant les règles d'égalité d'URN qui sont spécifiques du schéma de l'URN. Si l'élément qui effectue les comparaisons ne comprend pas le schéma d'URN, il effectue les comparaisons en utilisant les règles d'égalité lexicales définies dans la [RFC2141]. L'égalité lexicale pourrait résulter en ce que deux URN soient considérés non égaux quand ils sont en fait égaux. Dans cet usage spécifique des URN, le seul élément qui fournit l'URN est l'instance d'UA SIP identifiée par cet URN. Par suite, l'instance d'UA doit fournir des URN lexicalement équivalents dans chaque enregistrement qu'il génère. Ceci est probablement le comportement normal dans tous les cas ; les clients ne vont probablement pas modifier la valeur de l'identifiant d'instance afin qu'il reste fonctionnellement équivalent aux enregistrements précédents (mais lexicographiquement différents).

4.2 Enregistrements

4.2.1 Enregistrements initiaux

Au moment de la configuration, les UA obtiennent un ou plusieurs URI SIP représentant l'ensemble de mandataires sortants par défaut. La présente spécification suppose que l'ensemble est déterminé via un mécanisme de configuration, et de futures spécifications pourront définir des mécanismes supplémentaires comme d'utiliser le DNS pour découvrir cet ensemble. Comment l'UA est configuré sort du domaine d'application de cette spécification. Cependant, un UA DOIT prendre en charge des ensembles d'au moins deux URI de mandataire sortant et DEVRAIT prendre en charge des ensembles de jusqu'à quatre URI.

Pour chaque URI de mandataire sortant de l'ensemble, le client d'agent d'utilisateur (UAC) DEVRAIT envoyer une demande REGISTER utilisant cet URI comme mandataire sortant par défaut. (Autrement, l'UA pourrait limiter le nombre de flux formés pour préserver la batterie, par exemple). Si l'ensemble a plus d'un URI, l'UAC DOIT envoyer une demande REGISTER à au moins deux des mandataires sortants par défaut de l'ensemble. Les UA qui prennent en charge cette spécification DOIVENT inclure l'étiquette d'option "outbound" dans un champ d'en-tête Supported dans une demande REGISTER. Chacune de ces demandes REGISTER va utiliser un identifiant d'appel univoque. La formation de l'ensemble de chemins pour la demande sort du domaine d'application de ce document, mais résulte normalement en l'envoi du REGISTER de telle façon que le champ d'en-tête Route du sommet contienne un chemin lâche à l'URI de mandataire sortant.

Les demandes REGISTER autres que celles décrites au paragraphe 4.2.3, DOIVENT inclure une étiquette de caractéristique de support d'identifiant d'instance comme spécifié au paragraphe 4.1.

Un UAC conforme à cette spécification DOIT inclure dans le champ d'en-tête Contact, un paramètre "reg-id" distinct des autres paramètres "reg-id" utilisés dans les autres enregistrements qui utilisent le même paramètre de champ d'en-tête Contact "+sip.instance" et AOR. Chacun de ces enregistrements va former un nouveau flux de l'UA au mandataire. La séquence des valeurs de reg-id n'a pas à être à la suite mais DOIT être exactement la même séquence de valeurs de reg-id chaque fois que l'instance d'UA se met sous tension ou se réamorce, afin que les valeurs de reg-id coincident avec les valeurs de reg-id précédement utilisées. C'est pour que le registraire puisse remplacer les anciens enregistrements.

Note : l'UAC peut décider selon la situation si il demande un comportement de sortie en incluant ou omettant le paramètre de champ d'en-tête Contact "reg-id". Par exemple, imaginons que l'ensemble de mandataires sortants contienne deux mandataires dans des domaines différents, EP1 et EP2. Si un enregistrement de style sortant a réussi pour un flux à travers EP1, l'UA pourrait décider d'inclure "outbound" dans son champ d'en-tête Require quand il s'enregistre à EP2, afin d'assurer la cohérence. De façon similaire, si l'enregistrement à travers EP1 ne prend pas en charge "outbound", l'UA ne pourrait pas s'enregistrer du tout auprès de EP2.

L'UAC DOIT prendre en charge le mécanisme d'en-tête Path [RFC3327], et indiquer sa prise en charge en incluant l'étiquette d'option "path" dans une valeur de champ d'en-tête Supported dans ses demandes REGISTER. À part examiner facultativement le vecteur Path dans la réponse, c'est tout ce qui est exigé de l'UAC pour prendre en charge Path.

L'UAC cherche dans les réponses d'enregistrement réussi la présence d'une étiquette d'option outbound dans une valeur de champ d'en-tête Require. La présence de cette étiquette d'option indique que le registraire est conforme à cette spécification, et que tous les mandataires de bordure qui ont besoin de participer sont aussi conformes. Si le registraire ne prenait pas en charge "outbound", l'UA a potentiellement enregistré un contact non acheminable. Il est de la responsabilité de l'UA de supprimer tous les contacts inappropriés.

Si un enregistrement sortant réussit, comme indiqué par la présence de l'étiquette d'option "outbound" dans le champ d'en-tête Require d'une réponse d'enregistrement réussi, l'UA commence à envoyer des maintiens en vie comme décrit au paragraphe 4.4.

Note : l'UA doit honorer les réponses 503 (Service indisponible) aux enregistrements comme décrit dans les [RFC3261] et [RFC3263]. En particulier, les mises en œuvre devraient noter que quand elles reçoivent une réponse 503 (Service indisponible) avec un champ d'en-tête Retry-After, l'UA est supposé attendre la durée indiquée et réessayer l'enregistrement. Une valeur de champ d'en-tête Retry-After de 0 est valide et indique que l'UA est supposé réessayer la demande REGISTER immédiatement. Les mises en œuvre doivent s'assurer que quand elles réessayent la demande REGISTER, elles revisitent les résultats de résolution du DNS afin que l'UA puisse choisir un autre hôte que celui choisi la fois précédente que l'URI a été résolu.

Si l'UA qui s'enregistre reçoit une réponse 439 (Le premier bond ne prend pas en charge Outbound) à une demande REGISTER, il PEUT tenter à nouveau l'enregistrement sans utiliser le mécanisme outbound (sous réserve de la politique locale du client). Si le client a un ou plusieurs mandataires sortants de remplacement disponibles, il PEUT tenter à nouveau l'enregistrement à travers un de ces mandataires sortants. Voir au paragraphe 11.6 plus d'informations sur le code de réponse 439.

4.2.2 Demandes REGISTER suivantes

Les enregistrements pour rafraîchir un lien et pour supprimer un lien utilisent les mêmes valeurs de instance-id et reg-id que l'enregistrement initial correspondant où le lien était ajouté. Les enregistrements qui rafraîchissent simplement un lien existant sont envoyés sur le même flux que l'enregistrement original où le lien a été ajouté.

Si un réenregistrement est rejeté avec une réponse d'erreur récupérable, par exemple un 503 (Service indisponible) contenant un en-tête Retry-After, l'UAC NE DEVRAIT PAS supprimer le flux correspondant si le flux utilise un transport en mode connexion comme TCP. Tant que des "pong" sont reçus en réponse aux "ping", le flux DEVRAIT être gardé actif jusqu'à ce qu'une réponse d'erreur non récupérable soit reçue. Cela empêche des fermetures et ouvertures inutiles de connexions.

4.2.3 Enregistrements de tiers

Dans un enregistrement initial ou un réenregistrement, un UA NE DOIT PAS inclure de paramètre de champ d'en-tête "reg-id" dans le champ d'en-tête Contact si l'UA qui s'enregistre n'est pas la même instance que l'UA référencé par le champ d'en-tête Contact cible. (Cette pratique est utilisée occasionnellement pour installer la politique de transmission dans les registraires.)

Un UAC NE DOIT PAS non plus inclure une étiquette de caractéristique instance-id ou un paramètre de champ d'en-tête

Contact "reg-id" dans une demande de désenregistrement de tous les contacts (une seule valeur de champ d'en-tête Contact avec la valeur de "*").

4.3 Envoi de demandes non REGISTER

Quand un UAC va envoyer une demande, il effectue d'abord le traitement normal pour choisir l'URI de prochain bond. L'UA peut utiliser diverses techniques pour calculer l'ensemble de chemins et donc l'URI de prochain bond. La discussion de ces techniques sort du domaine d'application de ce document. Les UA qui prennent en charge la présente spécification DEVRAIENT inclure l'étiquette d'option "outbound" dans un champ d'en-tête Supported dans une demande qui n'est pas une demande REGISTER.

L'UAC effectue la résolution DNS normale sur l'URI de prochain bond (comme décrit dans la [RFC3263]) pour trouver un protocole, une adresse IP, et un accès. Pour les protocoles qui n'utilisent pas TLS, si l'UAC a un flux existant à cette adresse IP, et un accès avec le protocole correct, l'UAC DOIT alors utiliser la connexion existante. Pour les protocoles TLS, il DOIT aussi y avoir correspondance entre la production d'hôte dans le prochain bond et un des URI contenus dans le subjectAltName du certificat de l'homologue. Si l'UAC ne peut pas utiliser un des flux existants, il DEVRAIT alors former un nouveau flux en envoyant un datagramme ou en ouvrant une nouvelle connexion au prochain bond, comme approprié pour le protocole de transport.

Normalement, un UAC qui utilise les procédures de ce document et envoie une demande formant dialogue va vouloir que toutes les demandes suivantes dans le dialogue arrivent sur le même flux. Si l'UAC utilise un URI d'UA mondialement acheminable (GRUU, *Globally Routable UA URI*) [RFC5627] qui a été instancié en utilisant une valeur de champ d'en-tête Contact qui incluait un paramètre "ob", l'UAC envoie la demande sur le flux utilisé pour l'enregistrement, et les demandes suivantes vont arriver sur ce même flux. Si l'UAC n'utilise pas un tel GRUU, l'UAC ajoute alors un paramètre "ob" à sa valeur de champ d'en-tête Contact. Cela va faire que toutes les demandes suivantes dans le dialogue vont arriver sur le flux instancié par la demande formant dialogue. Ce cas est normal quand la demande est envoyée avant l'enregistrement, comme dans le dialogue d'abonnement initial pour le cadre de configuration [RFC6080].

Note : Si l'UAC veut qu'un flux UDP fonctionne à travers des NAT ou pare-feu, il a quand même besoin de mettre le paramètre "rport" [RFC3581] dans sa valeur de champ d'en-tête Via, et d'envoyer à partir de l'accès sur lequel il est prêt à recevoir. Des informations générales sur la traversée de NAT dans SIP sont décrites dans la [RFC6314].

4.4 Maintien en vie et détection de défaillance de flux

Les maintiens en vie sont utilisés pour rafraîchir les liens de NAT/pare-feu et détecter les défaillances de flux. Les flux peuvent défaillir pour de nombreuses raisons incluant le réamorçage des NAT et les pannes des mandataires de bordure.

Comme décrit au paragraphe 4.2, un UA qui s'enregistre va commencer par envoyer des maintiens en vie après une réponse d'enregistrement appropriée. Un UA qui ne s'enregistre pas (par exemple, une passerelle du RTPC derrière un pare-feu) peut aussi envoyer des maintiens en vie dans certaines circonstances.

Dans des circonstances spécifiques, un UAC pourrait être admis à envoyer des maintiens en vie STUN même si les procédures du paragraphe 4.2 ne sont pas achevées, pourvu qu'il y ait une indication explicite que le nœud cible de premier bond SIP prend en charge les maintiens en vie STUN. Par exemple, cela s'applique à un UA qui ne s'enregistre pas ou à un cas où l'enregistrement d'UA a réussi, mais la réponse n'incluait pas d'étiquette d'option "outbound" dans le champ d'en-tête Require.

Note : un UA peut "toujours" envoyer un double CRLF (un "ping") sur des transports en mode connexion comme c'est déjà permis par le paragraphe 7.5 de la [RFC3261]. Cependant un UA qui ne s'est pas enregistré en utilisant un enregistrement sortant ne peut pas attendre un CRLF en réponse (un "pong") sauf si l'UA a une indication explicite que les maintiens en vie CRLF sont supportés comme décrit dans ce paragraphe. De même, un UA qui n'a pas réussi à s'enregistrer avec les procédures "outbound" a besoin d'une indication explicite que le nœud cible de premier bond SIP supporte les maintiens en vie STUN avant de pouvoir envoyer des messages STUN.

Une option de configuration qui indique la prise en charge des maintiens en vie pour une cible spécifique est considérée donner une indication explicite. Si ces conditions sont satisfaites, l'UA envoie ses maintiens en vie selon les mêmes lignes directrices qu'utilisées quand les UA s'enregistrent ; ces lignes directrices sont décrites ci-dessous.

L'UA a besoin de détecter quand un flux spécifique a une défaillance. L'UA essaye activement de détecter une défaillance en envoyant périodiquement des messages de maintien en vie en utilisant une des techniques décrites aux paragraphes 4.4.1 ou 4.4.2. Si un flux avec un enregistrement a échoué, l'UA suit les procédures du paragraphe 4.2 pour former un nouveau flux pour remplacer celui qui est défaillant.

Quand une réponse d'enregistrement réussi contient le champ d'en-tête Flow-Timer, la valeur de ce champ d'en-tête est le nombre de secondes que le serveur est prêt à attendre sans voir de maintien en vie avant qu'il puisse considérer que le flux correspondant est mort. Noter que le serveur pourrait attendre une durée plus longue que le Flow-Timer afin d'avoir une période de grâce pour tenir compte du délai de transport. L'UA DOIT envoyer des maintiens en vie au moins aussi souvent que ce nombre de secondes. Si l'UA utilise la fréquence de maintien en vie recommandée par le serveur, il DEVRAIT envoyer ses maintiens en vie de telle façon que l'intervalle entre chaque maintien en vie soit aléatoirement distribué entre 80 % et 100 % du temps fourni par le serveur. Par exemple, si le serveur suggère 120 secondes, l'UA devrait envoyer chaque maintien en vie avec une fréquence différente entre 95 et 120 secondes.

Si aucun champ d'en-tête Flow-Timer n'était présent dans une réponse Register pour ce flux, l'UA peut envoyer des maintiens en vie à sa discrétion. Les paragraphes ci-dessous donnent des valeurs par défaut RECOMMANDÉES pour ces maintiens en vie.

Le client doit effectuer la résolution DNS SIP normale de la [RFC3263] sur l'URI provenant de l'ensemble de mandataires sortants pour prendre un transport. Une fois qu'un transport est choisi, l'UA choisit l'approche de maintien en vie qui est recommandée pour ce transport.

Le paragraphe 4.4.1 décrit un mécanisme de maintien en vie pour les transports en mode connexion comme TCP ou SCTP. Le paragraphe 4.4.2 décrit un mécanisme de maintien en vie pour les transports sans connexion comme UDP. La prise en charge d'autres transports comme DCCP [RFC4340] fera l'objet d'études futures.

4.4.1 Maintien en vie avec CRLF

Cette approche DOIT seulement être utilisée avec des transports en mode connexion comme TCP ou SCTP ; elle NE DOIT PAS être utilisée avec des transports sans connexion comme UDP.

Un agent d'utilisateur qui forme des flux vérifie si l'URI configuré auquel l'UA se connecte se résout en un transport en mode connexion (par exemple, TCP et TLS sur TCP).

Pour ce mécanisme, le "ping" de client est une séquence de double CRLF, et le "pong" de serveur est un seul CRLF, comme défini dans l'ABNF ci-dessous :

```
CRLF = CR LF
double-CRLF = CR LF CR LF
CR = %x0D
LF = %x0A
```

Le "ping" et le "pong" doivent être envoyés entre des messages SIP et ne peuvent pas être envoyés au milieu d'un message SIP. Si l'envoi est sur TLS, les CRLF sont envoyés à l'intérieur du canal protégé par TLS. Si l'envoi est sur un flux de données compressé par SigComp [RFC3320], les CRLF de maintien en vie sont envoyés à l'intérieur du flux compressé. Le double CRLF est considéré comme un seul message SigComp. Le mécanisme spécifique pour représenter ces caractères est une affaire spécifique de la mise en œuvre à traiter par le compresseur SigComp à l'extrémité d'envoi.

Si un pong n'est pas reçu dans les 10 secondes après l'envoi d'un ping (ou immédiatement après le traitement d'un message entrant reçu quand ces 10 secondes expirent) alors le client DOIT traiter le flux comme défaillant. Les clients DOIVENT prendre en charge le CRLF de maintien en vie.

Note : cette valeur de 10 secondes de temporisation a été choisie comme assez longue pour permettre qu'un serveur envoie une réponse même si le serveur est temporairement occupé avec une activité administrative. En même temps, elle a été choisie pour être assez petite pour qu'un UA enregistré à deux serveurs redondants avec un temps d'activation de matériel non remarquable puisse quand même facilement fournir de très hauts niveaux de fiabilité globale. Bien que certains protocoles Internet soient conçus pour des temps d'aller-retour de plus de 10 secondes, SIP pour les communications en temps réel n'est pas réellement utilisable dans ce type d'environnements car les utilisateurs abandonnent souvent les appels sans attendre plus de quelques secondes.

Quand un champ d'en-tête Flow-Timer n'est pas fourni dans la plus récente réponse d'enregistrement de succès, le choix de la fréquence appropriée de maintien en vie est principalement un compromis entre l'usage de la batterie et la disponibilité. L'UA DOIT choisir un nombre aléatoire entre une limite supérieurs fixe ou configurable et une limite inférieure, où la limite inférieure est 20 % de moins que la limite supérieure. La limite supérieure fixe ou la limite supérieure par défaut configurable DEVRAIT être de 120 secondes (95 secondes pour la limite inférieure) lorsque la puissance de la batterie n'est pas un souci et 840 secondes (672 secondes pour la limite inférieure) lorsque la puissance de la batterie pose problème. Le nombre aléatoire sera différent pour chaque "ping" de maintien en vie.

Note sur le choix des valeurs de temps : la limite supérieure de 120 secondes a été choisie suivant l'idée que pour une

bonne expérience d'utilisateur, les défaillances vont normalement être détectée dans ce délai et qu'une nouvelle connexion va être établie. La limite supérieure de 14 minutes pour les appareils alimentés par batterie a été choisie sur la base des NAT avec des temporisations TCP de 15 minutes. Les opérateurs qui souhaitent changer la relation entre charge sur les serveurs et temps attendu pendant lequel un utilisateur pourrait ne pas recevoir de communication entrante vont probablement ajuster cette durée. La limite inférieure de 95 seconde a été choisie afin que la gigue introduite résulte en une charge relativement égale sur les serveurs après 30 minutes.

4.4.2 Maintien en vie avec STUN

Cette approche DOIT seulement être utilisée avec les transports sans connexion, comme UDP ; elle NE DOIT PAS être utilisée pour les transports en mode connexion comme TCP et SCTP.

Un agent d'utilisateur qui forme des flux vérifie si l'URI configuré auquel l'UA se connecte se résout en l'utilisation du transport UDP. L'UA peut effectuer périodiquement des vérifications de maintien en vie en envoyant des demandes de lien STUN [RFC5389] sur le flux comme décrit à la Section 8. Les clients DOIVENT prendre en charge les maintiens en vie fondés sur STUN.

Quand un champ d'en-tête Flow-Timer n'est pas inclus dans une réponse d'enregistrement réussi, le temps entre chaque demande de maintien en vie DEVRAIT être un nombre aléatoire entre 24 et 29 secondes.

Note sur le choix des valeurs de temps : la limite supérieure de 29 secondes a été choisie, parce que de nombreux NAT ont des temporisations UDP jusqu'à 30 secondes. la limite inférieure de 24 secondes a été choisie afin qu'après 10 minutes la gigue introduite par les différents temporisateurs rende les demandes de maintien en vie non synchronisées pour étaler équitablement la charge sur les serveurs. Noter que les courtes temporisations de NAT avec UDP ont un impact négatif sur la vie de la batterie.

Si une réponse STUN Erreur de lien est reçue, ou si aucune réponse de lien n'est reçue après 7 retransmissions (16 fois le temporisateur STUN "RTO", où RTO est une estimation du délai d'aller-retour) l'UA considère que le flux est défaillant. Si le XOR-MAPPED-ADDRESS dans la réponse de lien STUN change, l'UA DOIT traiter cet événement comme une défaillance sur le flux.

4.5 Récupération de flux

Quand un flux utilisé pour l'enregistrement (par un URI particulier dans l'ensemble de mandataires sortants) est défaillant, l'UA doit former un nouveau flux pour remplacer le vieux flux et remplacer tous les enregistrements qui ont été envoyés précédemment sur ce flux. Chaque nouvel enregistrement DOIT avoir la même valeur de reg-id que l'enregistrement qu'il remplace. Ceci est fait de la même façon que de former un nouveau flux comme décrit au paragraphe 4.2 ; cependant, si il y a un échec dans la formation de ce flux, l'UA doit attendre un certain temps avant de réessayer de former un flux à ce prochain bond particulier.

Le temps d'attente dépend de si la tentative précédente d'établissement d'un flux a réussi. Pour les besoins de cette section, un flux est considéré réussi si l'enregistrement sortant a réussi, et si les maintiens en vie sont utilisés sur ce flux, au moins une réponse suivante de maintien en vie a été reçue.

Le nombre de secondes à attendre est calculé de la façon suivante : si tous les flux à chaque URI dans l'ensemble de mandataires sortants ont échoué, le temps de base est réglé à une valeur inférieure (30 secondes par défaut) ; autrement, dans le cas où au moins un des flux n'est pas défaillant, le temps de base est réglé à une valeur supérieure (avec un défaut de 90 secondes). La limite supérieure de temps d'attente (W) est calculée en prenant deux à la puissance du nombre de défaillances consécutives d'enregistrement pour cet URI, et en le multipliant par le temps de base, jusqu'à un temps maximum configurable (avec une valeur par défaut de 1800 secondes).

W = min (temps-max, (temps de base * (2 ^ défaillances consécutives)))

Ces temps PEUVENT être configurables à l'UA. Les trois temps sont :
- o temps max avec une valeur par défaut de 1800 secondes
- o temps de base (si toutes ont échoué) avec une valeur par défaut de 30 secondes
- o temps de base (si toutes n'ont pas échoué) avec une valeur par défaut de 90 secondes

Par exemple, si le temps de base est 30 secondes, et si il y a eu trois défaillances, la limite supérieure de temps d'attente est min(1800, 30*(2^3)) ou 240 secondes. La durée réelle d'attente de l'UA avant de réessayer l'enregistrement (le délai de ré-essai) est calculé en choisissant un délai aléatoire uniforme entre 50 et 100 % de la limite supérieure de temps d'attente. L'UA DOIT attendre au moins la valeur du délai de ré-essai avant d'essayer un autre enregistrement pour former un nouveau flux pour cet URI (une réponse 503 à une tentative d'enregistrement non aboutie antérieure avec une valeur de

champ d'en-tête Retry-After peut causer une plus longue attente de l'UA).

Pour être explicitement clair sur les conditions de limites, quand l'UA s'amorce, il essaye immédiatement de s'enregistrer. Si cela échoue et qu'aucun enregistrement ne réussit sur d'autres flux, le premier essai se produit quelque part entre 30 et 60 secondes après l'échec de la première demande d'enregistrement. Si le nombre d'échecs consécutifs est assez grand pour que le maximum de 1800 secondes soit atteint, l'UA va continuer d'essayer indéfiniment avec un délai aléatoire de 15 à 30 minutes entre chaque tentative.

5. Procédures de mandataire de bordure

5.1 Traitement des demandes Register

Quand un mandataire de bordure reçoit une demande d'enregistrement avec un paramètre de champ d'en-tête "reg-id" dans le champ d'en-tête Contact, il doit déterminer si il (le mandataire de bordure) va devoir être visité pour des demandes suivantes envoyées à l'agent d'utilisateur identifié dans le champ d'en-tête Contact, ou non. Si le mandataire de bordure est le premier bond, comme indiqué par le champ d'en-tête Via, il DOIT insérer son URI dans une valeur du champ d'en-tête Path comme décrit dans la [RFC3327]. Si il n'est pas le premier bond, il pourrait quand même décider de s'ajouter à l'en-tête Path sur la base de la politique locale. De plus, si le mandataire de bordure est le premier nœud SIP après l'UAC, le mandataire de bordure DOIT soit mémoriser un "jeton de flux" (contenant des informations sur le flux provenant du bond précédent) dans son URI Path, soit rejeter la demande. Le jeton de flux DOIT être un identifiant unique pour ce flux de réseau. Le jeton de flux PEUT être placé dans la partie utilisateur de l'URI. De plus, le premier nœud DOIT inclure un paramètre d'URI "ob" dans sa valeur du champ d'en-tête Path. Si le mandataire de bordure n'est pas le premier nœud SIP après l' UAC, il NE DOIT PAS placer un paramètre d'URI "ob" dans une valeur du champ d'en-tête Path. Le mandataire de bordure peut déterminer si il est le premier bond en examinant le champ d'en-tête Via.

5.2 Génération de jetons de flux

Une façon triviale mais impraticable de satisfaire l'exigence de jeton de flux du paragraphe 5.1 implique de mémoriser une transposition entre un compteur incrémentaire et les informations de connexion ; cependant, cela exigerait que le mandataire de bordure garde une quantité d'état ingérable. Quand cet état pourrait être supprimé n'est pas clair, et l'approche aurait des problèmes si le mandataire tombe en panne et perd la valeur du compteur. Un exemple sans état est donné ci-dessous. Un mandataire peut utiliser tout algorithme qu'il veut tant que le jeton de flux est unique pour un flux, le flux peut être récupéré du jeton, et le jeton ne peut pas être modifié par des attaquants.

Exemple d'algorithme : quand le mandataire s'amorce, il choisit une clé de 20 octets crypto aléatoire appelée K que seul le mandataire de bordure connaît. Un dispositif d'octets, appelé S, est formé contenant les informations suivantes sur le flux sur lequel la demande a été reçue : une énumération indiquant le protocole, l'adresse et l'accès IP locaux, l'adresse et l'accès IP distants. Le HMAC de S est calculé en utilisant la clé K et l'algorithme HMAC-SHA1-80, comme défini dans la [RFC2104]. L'enchaînement de HMAC et de S est codé en base64, comme défini dans la [RFC4648], et utilisé comme identifiant de flux. Quand on utilise des adresses IPv4, il va en résulter un identifiant de 32 octets.

5.3 Transmission de demandes non REGISTER

Quand un mandataire de bordure reçoit une demande, il applique les procédures d'acheminement normales avec les ajouts suivants. Si le mandataire de bordure reçoit une demande où le mandataire de bordure est l'hôte dans la valeur de champ d'en-tête Route supérieur, et si la valeur de champ d'en-tête Route contient un jeton de flux, le mandataire suit les procédures de ce paragraphe. Autrement, le mandataire de bordure saute les procédures de ce paragraphe, se supprime du champ d'en-tête Route, et continue de traiter la demande.

Le mandataire décode le jeton de flux et compare le flux dans le jeton de flux avec la source de la demande pour déterminer si c'est une demande "entrante" ou "sortante".

Si le flux dans le jeton de flux identifié par la valeur du champ d'en-tête Route supérieur correspond à l'adresse et accès de source IP de la demande, la demande est une demande "sortante" ; autrement, c'est une demande "entrante".

5.3.1 Traitement des demandes entrantes

Si la valeur d'en-tête Route contient un paramètre d'URI "ob", l'en-tête Route a probablement été copié de l'en-tête Path dans un enregistrement. Si la valeur d'en-tête Route contient un paramètre d'URI "ob", et si la demande est une nouvelle demande de formation de dialogue, le mandataire doit ajuster l'ensemble de chemins pour s'assurer que les demandes suivantes dans le dialogue peuvent être livrées sur un flux valide à l'instance d'UA identifiée par le jeton de flux.

Note : une approche simple pour satisfaire cette exigence est que le mandataire ajoute une valeur de champ d'en-tête Record-Route qui contient le jeton de flux, en copiant l'URI dans l'en-tête Route moins le paramètre "ob".

Ensuite, que le champ d'en-tête Route ait contenu un paramètre d'URI "ob" ou non, le mandataire supprime la valeur du champ d'en-tête Route et transmet la demande sur le "flux logique" identifié par le jeton de flux, qui est connu pour livrer les données à l'instance d'UA cible spécifique. Si le jeton de flux a été altéré, le mandataire DEVRAIT envoyer une réponse 403 (Interdit). Si le flux n'existe plus, le mandataire DEVRAIT envoyer une réponse 430 (Flux défaillant) à la demande.

Les mandataires qui ont utilisé l'exemple d'algorithme décrit au paragraphe 5.2 pour former un jeton de flux suivent les procédures ci-dessous pour déterminer le flux correct. Pour décoder le jeton de flux, prendre l'identifiant de flux dans la portion utilisateur de l'URI et le décoder de base64, puis vérifier que le HMAC est correct en recalculant le HMAC et en vérifiant qu'il correspond. Si le HMAC n'est pas correct, la demande a été altérée.

5.3.2 Traitement des demandes sortantes

Pour que les demandes de mi-dialogue fonctionnent avec des UA sortants, les demandes doivent être transmises sur un flux valide à l'instance d'UA appropriée. Si le mandataire de bordure reçoit une demande de formation de dialogue sortante, le mandataire de bordure peut utiliser la présence du paramètre d'URI "ob" dans l'URI de contact de l'UAC (ou le champ d'en-tête Route sommital) pour déterminer si le mandataire de bordure a besoin d'aider à l'acheminement de la demande de mi-dialogue.

Note de mise en œuvre : des procédures spécifiques chez le mandataire de bordure pour assurer que les demandes de mi-dialogue sont acheminées sur un flux existant ne font pas partie de cette spécification. Cependant, une approche comme de faire que le mandataire de bordure ajoute un en-tête Record-Route avec un jeton de flux est une façon de s'assurer que les demandes de mi-dialogue sont acheminées sur le flux correct.

5.4 Traitement de maintien en vie par le mandataire de bordure

Tous les mandataires de bordure conformes à la présente spécification DOIVENT mettre en œuvre la prise en charge des maintiens en vie STUN de NAT sur leur accès SIP UDP comme décrit à la Section 8.

Quand un serveur reçoit une séquence de double CRLF entre des messages SIP sur un transport en mode connexion tel que TCP ou SCTP, il DOIT immédiatement répondre avec un seul CRLF sur la même connexion.

Le dernier mandataire à transmettre une réponse d'enregistrement réussi à un UA PEUT inclure un champ d'en-tête Flow-Timer si la réponse contient l'étiquette d'option "outbound" dans une valeur de champ d'en-tête Require de la réponse. La raison pour laquelle un mandataire enverrait un Flow-Timer est qu'il souhaite détecter des défaillances de flux de façon proactive et prendre une action appropriée (par exemple, des alarmes d'enregistrement, fournir un traitement de remplacement si des demandes entrantes sont reçues pour l'UA, etc.). Le serveur DOIT attendre un temps supérieur à Flow-Timer afin d'avoir une période de grâce pour tenir compte du délai de transport.

6. Procédures de registraire

La présente spécification met à jour la définition d'un lien de la Section 10 de la [RFC3261], et du paragraphes 5.3 de la [RFC3327].

Les registraires qui mettent en œuvre la présente spécification DOIVENT prendre en charge le mécanisme d'en-tête Path [RFC3327].

Quand il reçoit une demande REGISTER, le registraire DOIT vérifier d'après son champ d'en-tête Via si le registraire est le premier bond ou non. Si le registraire n'est pas le premier bond, il DOIT examiner l'en-tête Path de la demande. Si le champ d'en-tête Path manque ou si il existe mais si le premier URI n'a pas de paramètre d'URI "ob", alors le traitement sortant NE DOIT PAS être appliqué à l'enregistrement. Dans ce cas, le traitement suivant s'applique : si la demande REGISTER contient le reg-id et l'étiquette d'option sortante dans un champ d'en-tête Supported, alors le registraire DOIT répondre à la demande REGISTER avec une réponse 439 (Le premier bond ne prend pas en charge Outbound) ; autrement, le registraire DOIT ignorer le paramètre "reg-id" de l'en-tête Contact. Voir au paragraphe 11.6 plus d'informations sur le code de réponse 439.

Une valeur de champ d'en-tête Contact avec une étiquette de caractéristique de support instance-id mais pas de paramètre de champ d'en-tête "reg-id" est valide (cette combinaison va résulter en la création d'une GRUU, comme décrit dans la [RFC5627]) mais une avec un reg-id et sans instance-id n'est pas valide. Si le registraire traite une valeur de champ d'en-

tête Contact avec un reg-id mais pas de instance-id, il ignore simplement le paramètre reg-id.

Un enregistrement contenant un paramètre de champ d'en-tête "reg-id" et une valeur d'expiration non zéro est utilisé pour enregistrer une seule instance d'UA sur un seul flux, et peut aussi désenregistrer tout champ d'en-tête Contact avec une valeur d'expiration de zéro. Donc, si le champ d'en-tête Contact contient plus d'une valeur de champ d'en-tête avec une expiration non zéro et si une de ces valeurs de champ d'en-tête contient un paramètre de champ d'en-tête Contact "reg-id", l'enregistrement entier DEVRAIT être rejeté avec une réponse 400 (Mauvaise demande). La justification de la recommandation de rejet plutôt que de le rendre obligatoire est qu'il est permis au receveur par la [RFC3261] d'étouffer (ne pas répondre) les messages excessivement mal formés ou malveillants.

Si l'en-tête Contact ne contenait pas de paramètre de champ d'en-tête Contact "reg-id" ou si ce paramètre a été ignoré (comme décrit ci-dessus) le registraire NE DOIT PAS inclure l'étiquette d'option outbound dans le champ d'en-tête Require de sa réponse.

Le registraire DOIT être prêt à recevoir, simultanément pour la même AOR, des enregistrements qui utilisent un instance-id et un reg-id et des enregistrements qui ne le font pas. Le registraire PEUT être configuré avec une politique locale à rejeter tout enregistrement qui n'inclut pas de instance-id et reg-id, ou avec des valeurs de champ d'en-tête Path qui ne contiennent pas le paramètre d'URI "ob". Si le champ d'en-tête Contact ne contient pas de paramètre de champ d'en-tête Contact "+sip.instance", le registraire traite la demande en utilisant les règles de lien de Contact de la [RFC3261].

Quand un paramètre de champ d'en-tête Contact "+sip.instance" et un paramètre de champ d'en-tête Contact "reg-id" sont présents dans un champ d'en-tête Contact d'une demande REGISTER (après la validation de l'en-tête Contact comme décrit ci-dessus) le lien correspondant est entre une AOR et la combinaison de l'identifiant d'instance (provenant du paramètre d'en-tête Contact "+sip.instance") et la valeur du paramètre de champ d'en-tête Contact "reg-id". Le registraire DOIT mémoriser dans le lien l'URI de contact, tous les paramètres de champ d'en-tête Contact, et toutes les valeurs de champ d'en-tête Path. (Même si lURI de contact n'est pas utilisé pour les comparaisons de liens, il est quand même nécessaire pour que le mandataire d'autorité forme l'ensemble cible.) Pourvu que l'UAC ait inclus une étiquette d'option "outbound" (définie au paragraphe 11.4) dans une valeur de champ d'en-tête Supported dans la demande REGISTER, le registraire DOIT inclure l'étiquette d'option "outbound" dans une valeur de champ d'en-tête Require dans sa réponse à cette demande REGISTER.

Si l'UAC a un flux direct avec le registraire, le registraire DOIT mémoriser assez d'informations pour identifier de façon univoque le flux de réseau sur lequel la demande est arrivée. Pour les systèmes d'exploitation courants avec TCP, cela va normalement être juste la bride pour le descripteur de fichier où la bride va devenir invalide si la session TCP a été close. Pour les systèmes d'exploitation courants avec UDP cela va normalement être le descripteur de fichier pour la prise locale qui a reçu la demande, l'interface locale, et l'adresse et numéro d'accès IP du côté distant qui a envoyé la demande. Le registraire PEUT mémoriser cette information en s'ajoutant lui-même au champ d'en-tête Path avec un jeton de flux approprié.

Si le registraire reçoit un réenregistrement pour une combinaison spécifique d'AOR, et de valeurs d'instance-id et reg-id, le registraire DOIT mettre à jour toute information qui identifie de façon unique le flux de réseau sur lequel la demande est arrivée si cette information a changé, et DEVRAIT mettre à jour l'heure de la dernière mise à jour du lien.

Pour être conforme à la présente spécification, les registraires qui peuvent recevoir des demandes SIP directement d'un UAC sans mandataire de bordure intermédiaire DOIVENT mettre en œuvre les mêmes mécanismes de maintien en vie que les mandataires de bordure (paragraphe 5.4). Les registraires avec un flux direct à un UA PEUVENT inclure un en-tête Flow-Timer dans une réponse d'enregistrement de classe 2xx qui inclut l'étiquette d'option "outbound" dans l'en-tête Require.

7. Procédures de mandataire d'autorité : tranmission des demandes

Quand un mandataire utilise le service de localisation pour chercher un lien d'enregistrement et ensuite mandate une demande à un contact particulier, il choisit normalement un contact à utiliser, avec quelques règles supplémentaires :

- Le mandataire NE DOIT PAS remplir l'ensemble cible avec plus d'un contact avec la même AOR et le même identifiant d'instance à la fois.

- Si une demande pour une AOR et instance-id particulière échoue avec une réponse 430 (Flux défaillant) le mandataire DEVRAIT remplacer la branche défaillante par une autre cible (si une est disponible) avec la même AOR et instance-id, mais un reg-id différent.

- Si le mandataire reçoit d'une branche une réponse finale autre qu'une réponse 408 (Fin de temporisation de demande)

ou 430 (Flux défaillant) le mandataire NE DOIT PAS transmettre la même demande à une autre cible représentant la même AOR et instance-id. L'instance ciblée a déjà fourni sa réponse.

Le mandataire utilise la cible de prochain bond du message et la valeur de tout vecteur de champ d'en-tête Path mémorisé dans le lien d'enregistrement pour décider comment transmettre et remplir l'en-tête Route dans la demande. Si le mandataire est co-localisé avec le registraire et a mémorisé des informations sur le flux à l'UA qui a créé le lien, alors le mandataire DOIT envoyer la demande sur le même "flux logique" sauvegardé avec le lien, car ce flux est connu pour livrer les données au flux de réseau spécifique de l'instance d'UA cible qui a été sauvegardé avec le lien.

Note de mise en œuvre : normalement cela signifie que pour TCP, la demande est envoyée sur la même prise TCP qui a reçu la demande REGISTER. Pour UDP, la demande est envoyée de la même adresse IP locale et accès sur lequel l'enregistrement a été reçu, à la même adresse et accès IP d'où le REGISTER a été reçu.

Si un mandataire ou registraire reçoit des informations du réseau qui indiquent qu'aucun futur message ne va être livré sur un flux spécifique, alors le mandataire DOIT invalider tous les liens dans l'ensemble cible qui utilise ce flux (sans considération de l'AOR). Des exemples de cela sont une prise TCP qui clôt ou reçoit une erreur ICMP "destination inaccessible" sur un flux UDP. De même, si un mandataire clôt un descripteur de fichier, il DOIT invalider tous les liens dans l'ensemble cible avec le flux qui utilise ce descripteur de fichier.

8. Traitement du maintien en vie STUN

Cette Section décrit les changements à la couche de transport SIP qui permettent aux demandes de liens SIP et STUN [RFC5389] d'être mêlées sur le même flux. Cela constitue un nouvel usage de STUN. Les messages STUN sont utilisés pour vérifier que la connexité est encore disponible sur un flux UDP, et pour fournir des maintiens en vie périodiques. Ces maintiens en vie STUN sont toujours envoyés au prochain bond SIP. Les messages STUN ne sont pas livrés de bout en bout.

Les seuls messages STUN exigés par cet usage sont les demandes de lien, les réponses de lien, et les réponses d'erreur de lien. L'UAC envoie les demandes de lien sur le même flux UDP qu'utilisé pour l'envoi des messages SIP. Ces demandes de lien n'exigent aucun attribut STUN. Les réponses de lien correspondantes n'exigent aucun attribut STUN à part XOR-MAPPED-ADDRESS. L'UAS, mandataire, ou registraire répond à une demande de lien valide avec une réponse de lien qui DOIT inclure l'attribut XOR-MAPPED-ADDRESS.

Si un serveur conforme à cette section reçoit des demandes SIP sur une interface et accès UDP, il DOIT aussi fournir une version limitée de serveur STUN sur la même interface et accès UDP.

Note : Il est facile de distinguer les paquets STUN et SIP envoyés sur UDP, parce que le premier octet d'une méthode de lien STUN a une valeur de 0 ou 1, tandis que le premier octet d'un message SIP n'est jamais 0 ou 1.

Parce que l'envoi et la réception de données binaires STUN sur les mêmes accès qu'utilisés pour SIP est un changement significatif et non rétro compatible avec la RFC 3261, cette section exige un certain nombre de vérifications avant d'envoyer des messages STUN à un nœud SIP. Si un nœud SIP envoie des demandes STUN (par exemple, à cause d'une configuration incorrecte) en dépit de ces avertissements, le nœud pourrait être mis sur une liste noire pour le trafic UDP.

Un nœud SIP NE DOIT PAS envoyer de demandes STUN sur un flux si il n'a pas une indication explicite que le serveur cible de prochain bond SIP prend en charge la présente spécification. Les UAC NE DOIVENT PAS utiliser une option de configuration ambiguë telle que "travailler à travers des NAT ?" ou "Faire des maintiens en vie ?" pour impliquer la prise en charge de STUN du prochain bond. Un UAC PEUT utiliser la présence d'un paramètre d'URI "ob" dans l'en-tête Path dans une réponse d'enregistrement comme l'indication que son premier mandataire de bordure prend en charge les maintiens en vie définis dans le présent document.

Note : normalement, un nœud SIP envoie d'abord une demande SIP et attend de recevoir une réponse de classe 2xx sur un flux à une nouvelle destination cible, avant d'envoyer des messages STUN. Quand il est programmé pour le prochain rafraîchissement de NAT, le nœud SIP envoie des demandes STUN à la cible.

Une fois qu'un flux est établi, les défaillances d'une demande STUN (incluant ses retransmissions) sont considérées comme une défaillance du flux sous-jacent. Pour SIP sur des flux UDP, si la XOR-MAPPED-ADDRESS retournée sur le flux change, cela indique que la connexité sous-jacente a changé, et est considérée être une défaillance de flux. L'usage de maintien en vie STUN SIP n'exige pas de rétro compatibilité avec la [RFC3489].

8.1 Utilisation avec SigComp

Quand STUN est utilisé avec des messages SIP compressés par SigComp [RFC3320] sur le même flux, les messages STUN sont simplement envoyés non compressés, "en dehors" de SigComp. Ceci est pris en charge par le multiplexage des messages STUN avec les messages SigComp en vérifiant les bits de poids fort du message. Ces bits sont toujours un pour SigComp, ou zéro pour STUN.

Note : tous les messages SigComp contiennent un préfixe (les cinq bits de poids fort du premier octet sont réglés à un) qui ne se produit pas dans les messages de texte codés en UTF-8 [RFC3629] de sorte que pour les applications qui utilisent ce codage (ou le codage ASCII) il est possible de multiplexer les messages d'application non compressés et les messages SigComp sur le même accès UDP. Les deux bits de poids fort de chaque méthode de lien STUN sont tous deux à zéro. Cela, combiné avec le mouchard magique, aide à différencier les paquets STUN des autres protocoles quand STUN est multiplexé avec d'autres protocoles sur le même accès.

9. Exemple de flux de messages

Voici un exemple de flux de messages qui illustre la plupart des concepts discutés dans la présente spécification. Dans de nombreux cas, les en-têtes Via, Longueur de contenu, et Max-Forwards sont omis par souci de concision et de lisibilité.

Dans ces exemples, "EP1" et "EP2" sont les mandataires sortants, et "Proxy" est le mandataire d'autorité.

La section est subdivisée en flux d'appels indépendants ; cependant, ils sont structurés en ordre séquentiel d'une séquence hypothétique de flux d'appels.

9.1 Abonnement au paquetage de configuration

Si l'ensemble de mandataires sortants est déjà configuré sur l'UA de Bob, ce paragraphe peut alors être sauté. Autrement, si l'ensemble de mandataires sortant est appris par le paquetage de configuration, l'UA de Bob envoie une demande SUBSCRIBE pour le paquetage de configuration du profil d'UA [RFC6080]. Cette demande est une interrogation (Expires est zéro). Après la réception de la demande NOTIFY, l'UA de Bob va chercher la configuration externe en utilisant HTTPS (non montré) et obtient un fichier de configuration qui contient les ensembles de mandataires sortants "sip:ep1.exemple.com;lr" et "sip:ep2.exemple.com;lr".

Dans cet exemple, le serveur DNS est configuré de façon que sip: exemple.com se résolve en EP1 et EP2.

Exemple de message n° 1 :

SUBSCRIBE sip:00000000-0000-1000-8000-AABBCCDDEEFF@exemple.com SIP/2.0
Via: SIP/2.0/TCP 192.0.2.2;branch=z9hG4bKnlsdkdj2
Max-Forwards: 70
From: <anonymous@exemple.com>;tag=23324
To: <sip:00000000-0000-1000-8000-AABBCCDDEEFF@exemple.com>
Call-ID: nSz1TWN54x7My0GvpEBj
CSeq: 1 SUBSCRIBE
Event: ua-profile ;profile-type=device ;vendor="exemple.com";model="uPhone";version="1.1"
Expires: 0
Supported: path, outbound
Accept: message/external-body, application/x-uPhone-config
Contact: <sip:192.0.2.2;transport=tcp;ob>

```
 ;+sip.instance="<urn:uuid:00000000-0000-1000-8000-AABBCCDDEEFF>"
Content-Length: 0
```

Dans le message n° 2, EP1 ajoute l'en-tête Record-Route suivant :

```
Record-Route: <sip:GopIKSsn0oGLPXRdV9BAXpT3coNuiGKV@ep1.exemple.com;lr>
```

Dans le message n° 5, le serveur de configuration envoie un NOTIFY avec un URL externe pour que Bob aille chercher sa configuration. Le NOTIFY a un en-tête Subscription-State qui termine l'abonnement.

Message n° 5

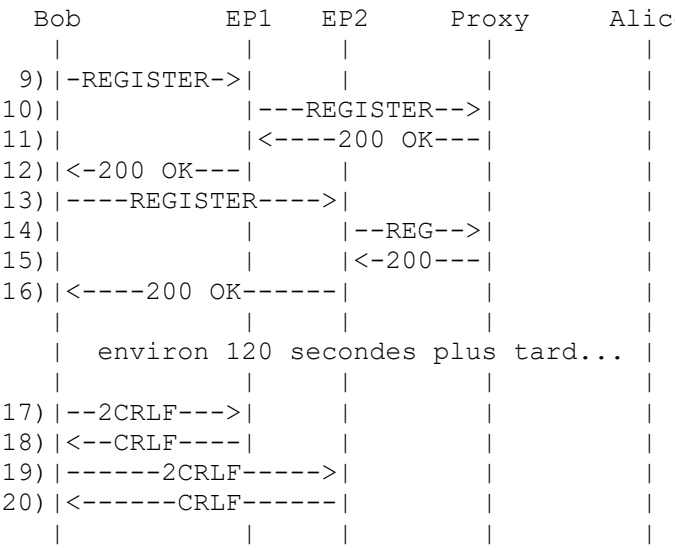
```
NOTIFY sip:192.0.2.2;transport=tcp;ob SIP/2.0
Via: SIP/2.0/TCP 192.0.2.5;branch=z9hG4bKn81dd2
Max-Forwards: 70
To: <anonymous@exemple.com>;tag=23324
From: <sip:00000000-0000-1000-8000-AABBCCDDEEFF@exemple.com>;tag=0983
Call-ID: nSz1TWN54x7My0GvpEBj
CSeq: 1 NOTIFY
Route: <sip:GopIKSsn0oGLPXRdV9BAXpT3coNuiGKV@ep1.exemple.com;lr>
Subscription-State: terminated;reason=timeout
Event: ua-profile
Content-Type: message/external-body; access-type="URL"
 ;expiration="Thu, 01 Jan 2009 09:00:00 UTC"
 ;URL="http://exemple.com/uPhone.cfg"
 ;size=9999;hash=10AB568E91245681AC1B
Content-Length: 0
```

EP1 reçoit cette demande NOTIFY, supprime l'en-tête Route, extrait le jeton de flux, calcule le flux correct, et transmet la demande (message n° 6) sur ce flux à Bob.

L'UA de Bob va chercher le fichier de configuration et apprend l'ensemble de mandataires sortants.

9.2 Enregistrement

Maintenant que l'UA de Bob est configuré avec l'ensemble de mandataires sortants par configuration ou en utilisant les procédures de cadre de configuration du paragraphe précédent, l'UA de Bob envoie des demandes REGISTER à travers chaque mandataire de bordure de l'ensemble. Une fois que les enregistrements ont réussi, l'UA de Bob commence à envoyer des CRLF de maintien en vie environ toutes les 2 minutes.



Dans le message n° 9, l'UA de Bob envoie son premier enregistrement à travers le premier mandataire de bordure dans l'ensemble de mandataires sortants en incluant un chemin lâche. L'UA inclut un instance-id et un reg-id dans sa valeur de champ d'en-tête Contact. Noter les étiquettes d'option dans l'en-tête Supported.

Message n° 9

```
REGISTER sip:exemple.com SIP/2.0
Via: SIP/2.0/TCP 192.0.2.2;branch=z9hG4bKnashds7
Max-Forwards: 70
From: Bob <sip:bob@exemple.com>;tag=7F94778B653B
To: Bob <sip:bob@exemple.com>
Call-ID: 16CB75F21C70
CSeq: 1 REGISTER
Supported: path, outbound
Route: <sip:ep1.exemple.com;lr>
Contact: <sip:bob@192.0.2.2;transport=tcp>;reg-id=1
 ;+sip.instance="<urn:uuid:00000000-0000-1000-8000-AABBCCDDEEFF>"
Content-Length: 0
```

Le message n° 10 est similaire. EP1 supprime la valeur de champ d'en-tête Route, décrémente Max-Forwards, et ajoute sa valeur de champ d'en-tête Via. Comme EP1 est le premier mandataire de bordure, il ajoute un en-tête Path avec un jeton de flux et inclut le paramètre "ob".

```
Path: <sip:VskztcQ/S8p4WPbOnHbuyh5iJvJIW3ib@ep1.exemple.com;lr;ob>
```

Comme la réponse au REGISTER (message n° 11) contient l'étiquette d'option sortante dans le champ d'en-tête Require, l'UA de Bob va savoir que le registraire a utilisé les règles de lien sortantes. La réponse contient aussi les contacts actuellement actifs, et le Path pour l'enregistrement en cours.

Message n° 11

```
SIP/2.0 200 OK
Via: SIP/2.0/TCP 192.0.2.15;branch=z9hG4bKnuiqisi
Via: SIP/2.0/TCP 192.0.2.2;branch=z9hG4bKnashds7
From: Bob <sip:bob@exemple.com>;tag=7F94778B653B
To: Bob <sip:bob@exemple.com>;tag=6AF99445E44A
Call-ID: 16CB75F21C70
CSeq: 1 REGISTER
Supported: path, outbound
Require: outbound
Contact: <sip:bob@192.0.2.2;transport=tcp>;reg-id=1;expires=3600
 ;+sip.instance="<urn:uuid:00000000-0000-1000-8000-AABBCCDDEEFF>"
Path: <sip:VskztcQ/S8p4WPbOnHbuyh5iJvJIW3ib@ep1.exemple.com;lr;ob>
Content-Length: 0
```

Le second enregistrement à travers EP2 (message n° 13) est similaire sauf que le Call-ID a changé, le reg-id est 2, et l'en-tête Route passe à travers EP2.

Message n° 13

```
REGISTER sip:exemple.com SIP/2.0
Via: SIP/2.0/TCP 192.0.2.2;branch=z9hG4bKnqr9bym
Max-Forwards: 70
From: Bob <sip:bob@exemple.com>;tag=755285EABDE2
To: Bob <sip:bob@exemple.com>
Call-ID: E05133BD26DD
CSeq: 1 REGISTER
Supported: path, outbound
Route: <sip:ep2.exemple.com;lr>
Contact: <sip:bob@192.0.2.2;transport=tcp>;reg-id=2
 ;+sip.instance="<urn:uuid:00000000-0000-1000-8000-AABBCCDDEEFF>"
Content-Length: 0
```

De même dans le message n° 14, EP2 ajoute un en-tête Path avec jeton de flux et paramètre "ob".

```
Path: <sip:wazHDLdIMtUg6r0I/oRZ15zx3zHE1w1Z@ep2.exemple.com;lr;ob>
```

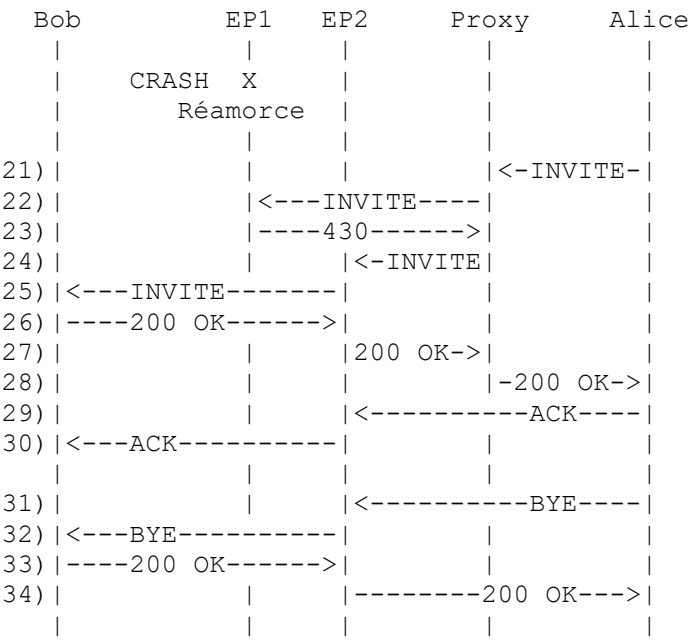
Le message n° 16 dit à l'UA de Bob que l'enregistrement sortant a réussi, et montre les deux contacts. Noter que seul le Path correspondant à l'enregistrement en cours est retourné.

Message n° 16

SIP/2.0 200 OK
Via: SIP/2.0/TCP 192.0.2.2;branch=z9hG4bKnqr9bym
From: Bob <sip:bob@exemple.com>;tag=755285EABDE2
To: Bob <sip:bob@exemple.com>;tag=49A9AD0B3F6A
Call-ID: E05133BD26DD
Supported: path, outbound
Require: outbound
CSeq: 1 REGISTER
Contact: <sip:bob@192.0.2.2;transport=tcp>;reg-id=1;expires=3600
 ;+sip.instance="<urn:uuid:00000000-0000-1000-8000-AABBCCDDEEFF>"
Contact: <sip:bob@192.0.2.2;transport=tcp>;reg-id=2;expires=3600
 ;+sip.instance="<urn:uuid:00000000-0000-1000-8000-AABBCCDDEEFF>"
Path: <sip:wazHDLdIMtUg6r0I/oRZ15zx3zHE1w1Z@ep2.exemple.com;lr;ob>
Content-Length: 0

9.3 Appel entrant et défaillance du mandataire

Dans cet exemple, après l'enregistrement, EP1 a une panne et réamorce. Avant que l'UA de Bob remarque que son flux à EP1 ne répond plus, Alice appelle Bob. Le mandataire d'autorité de Bob essaye d'abord le flux à EP1, mais EP1 n'a plus de flux pour Bob, donc il répond avec un 430 (Flux défaillant). Le mandataire supprime l'enregistrement périmé et essaye le prochain lien pour la même instance.



Message n° 21

INVITE sip:bob@exemple.com SIP/2.0
To: Bob <sip:bob@exemple.com>
From: Alice <sip:alice@a.exemple>;tag=02935
Call-ID: klmvCxVWGp6MxJp2T2mb
CSeq: 1 INVITE

Le mandataire de Bob réécrit l'URI de demande à l'URI de contact utilisé dans l'enregistrement de Bob, et place le chemin pour un des enregistrements vers l'instance d'UA de Bob dans un champ d'en-tête Route. Ce chemin passe par EP1.

Message n° 22

INVITE sip:bob@192.0.2.2;transport=tcp SIP/2.0
To: Bob <sip:bob@exemple.com>
From: Alice <sip:alice@a.exemple>;tag=02935
Call-ID: klmvCxVWGp6MxJp2T2mb

CSeq: 1 INVITE
Route: <sip:VskztcQ/S8p4WPbOnHbuyh5iJvJIW3ib@ep1.exemple.com;lr;ob>

Comme EP1 vient de réamorcer, il n'a pas le flux décrit dans le jeton de flux. Il retourne une réponse 430 (Flux défaillant).

Message n° 23

SIP/2.0 430 Flux défaillant
To: Bob <sip:bob@exemple.com>
From: Alice <sip:alice@a.exemple>;tag=02935
Call-ID: klmvCxVWGp6MxJp2T2mb
CSeq: 1 INVITE

Le mandataire supprime le lien pour ce chemin et essaye à nouveau de transmettre le INVITE, cette fois avec le chemin à travers EP2.

Message n° 24

INVITE sip:bob@192.0.2.2;transport=tcp SIP/2.0
To: Bob <sip:bob@exemple.com>
From: Alice <sip:alice@a.exemple>;tag=02935
Call-ID: klmvCxVWGp6MxJp2T2mb
CSeq: 1 INVITE
Route: <sip:wazHDLdIMtUg6r0I/oRZ15zx3zHE1w1Z@ep2.exemple.com;lr;ob>

Dans le message n° 25, EP2 a besoin d'ajouter une valeur de champ d'en-tête Record-Route, afin que tous les messages suivants dans le dialogue provenant de l'UA d'Alice arrivent à l'UA de Bob. EP2 peut déterminer qu'il a besoin du Record-Route car la demande est une demande de formation de dialogue et l'en-tête Route contenait un jeton de flux et un paramètre "ob". Ces informations de Record-Route sont repassées à l'UA d'Alice dans les réponses (messages n° 26, 27, et 28).

Message n° 25

INVITE sip:bob@192.0.2.2;transport=tcp SIP/2.0
To: Bob <sip:bob@exemple.com>
From: Alice <sip:alice@a.exemple>;tag=02935
Call-ID: klmvCxVWGp6MxJp2T2mb
CSeq: 1 INVITE
Record-Route:
<sip:wazHDLdIMtUg6r0I/oRZ15zx3zHE1w1Z@ep2.exemple.com;lr>

Message n° 26

SIP/2.0 200 OK
To: Bob <sip:bob@exemple.com>;étiquette=skduk2
From: Alice <sip:alice@a.exemple>;tag=02935
Call-ID: klmvCxVWGp6MxJp2T2mb
CSeq: 1 INVITE
Record-Route:
<sip:wazHDLdIMtUg6r0I/oRZ15zx3zHE1w1Z@ep2.exemple.com;lr>

À ce point, les deux UA ont l'ensemble de chemins correct pour le dialogue. Toutes les demandes suivantes dans ce dialogue vont s'acheminer correctement. Par exemple, la demande ACK dans le message n° 29 est envoyée de l'UA d'Alice directement à EP2. La demande BYE dans le message n° 31 utilise le même ensemble de chemins.

Message n° 29

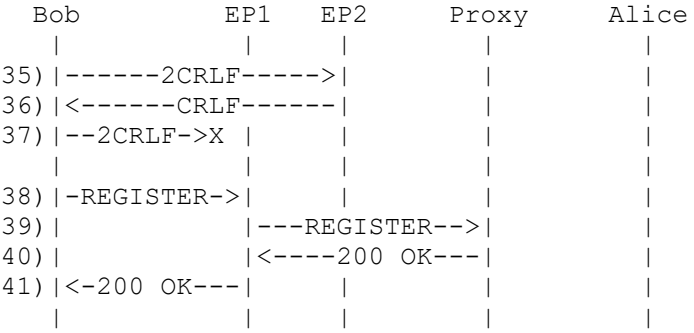
ACK sip:bob@192.0.2.2;transport=tcp SIP/2.0
To: Bob <sip:bob@exemple.com>;tag=skduk2
From: Alice <sip:alice@a.exemple>;tag=02935
Call-ID: klmvCxVWGp6MxJp2T2mb
CSeq: 1 ACK
Route: <sip:wazHDLdIMtUg6r0I/oRZ15zx3zHE1w1Z@ep2.exemple.com;lr>

Message n° 31

BYE sip:bob@192.0.2.2;transport=tcp SIP/2.0
To: Bob <sip:bob@exemple.com>;tag=skduk2
From: Alice <sip:alice@a.exemple>;tag=02935
Call-ID: klmvCxVWGp6MxJp2T2mb
CSeq: 2 BYE
Route: <sip:wazHDLdIMtUg6r0I/oRZ15zx3zHE1w1Z@ep2.exemple.com;lr>

9.4 Réenregistrement

Un peu plus tard, l'UA de Bob envoie des maintiens en vie à ses deux mandataires de bordure, mais il découvre que le flux avec EP1 est défaillant. L'UA de Bob se réenregistre à travers EP1 en utilisant le même reg-id et Call-ID qu'il avait précédemment utilisé.



Message n° 38

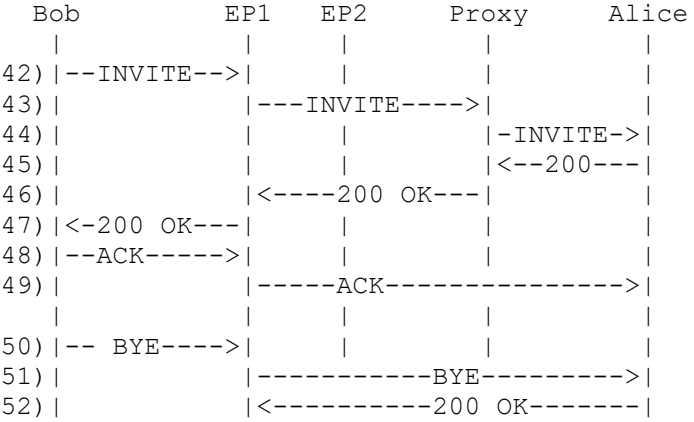
REGISTER sip:exemple.com SIP/2.0
From: Bob <sip:bob@exemple.com>;tag=7F94778B653B
To: Bob <sip:bob@exemple.com>
Call-ID: 16CB75F21C70
CSeq: 2 REGISTER
Supported: path, outbound
Route: <sip:ep1.exemple.com;lr>
Contact: <sip:bob@192.0.2.2;transport=tcp>;reg-id=1
;+sip.instance="<urn:uuid:00000000-0000-1000-8000-AABBCCDDEEFF>"

Dans le message n° 39, EP1 insère un en-tête Path avec un nouveau jeton de flux :

Path: <sip:3yJEbr1GYZK9cPYk5Snocez6DzO7w+AX@ep1.exemple.com;lr;ob>

9.5 Appel sortant

Finalement, Bob fait un appel sortant à Alice. L'UA de Bob inclut un paramètre "ob" dans son URI de contact dans le message n° 42. EP1 ajoute un Record-Route avec un jeton de flux dans le message n° 43. L'ensemble de chemins est retourné à Bob dans la réponse (messages n° 45, 46, et 47) et Bob ou Alice peut envoyer des demandes dans le dialogue.



Message n° 42

INVITE sip:alice@a.exemple SIP/2.0
From: Bob <sip:bob@exemple.com>;tag=ldw22z
To: Alice <sip:alice@a.exemple>
Call-ID: 95KGsk2V/Eis9LcpBYy3
CSeq: 1 INVITE
Route: <sip:ep1.exemple.com;lr>
Contact: <sip:bob@192.0.2.2;transport=tcp;ob>

Dans le message n° 43, EP1 ajoute l'en-tête Record-Route suivant :

Record-Route: <sip:3yJEbr1GYZK9cPYk5Snocez6DzO7w+AX@ep1.exemple.com;lr>

Quand EP1 reçoit le BYE (message n° 50) de l'UA de Bob, il peut dire que la demande est une demande "sortante" (car la source de la demande correspond au flux dans le jeton de flux) et simplement supprimer sa valeur de champ d'en-tête Route et transmettre la demande à l'UA d'Alice.

Message n° 50

BYE sip:alice@a.exemple SIP/2.0
From: Bob <sip:bob@exemple.com>;tag=ldw22z
To: Alice <sip:alice@a.exemple>;tag=plqus8
Call-ID: 95KGsk2V/Eis9LcpBYy3
CSeq: 2 BYE
Route: <sip:3yJEbr1GYZK9cPYk5Snocez6DzO7w+AX@ep1.exemple.com;lr>
Contact: <sip:bob@192.0.2.2;transport=tcp;ob>

10. Grammaire

La présente spécification définit un nouveau champ d'en-tête "Flow-Timer", et de nouveaux paramètres de champ d'en-tête Contact, "reg-id" et "+sip.instance". La grammaire inclut les définitions provenant de la [RFC3261]. Flow-Timer est un en-tête d'extension provenant de l'en-tête de message de l'ABNF de la [RFC3261].

L'ABNF [RFC5234] est :

Flow-Timer = "Flow-Timer" HCOLON 1*DIGIT

contact-params =/ c-p-reg / c-p-instance

c-p-reg = "reg-id" EQUAL 1*DIGIT ; 1 à (2^31 - 1)

c-p-instance = "+sip.instance" EQUAL DQUOTE "<" instance-val ">" DQUOTE

instance-val = 1*uric ; défini dans la RFC 3261

La valeur de reg-id NE DOIT PAS être 0 et DOIT être moins que 2^31.

11. Considérations relatives à l'IANA

11.1 Champ d'en-tête Flow-Timer

La présente spécification définit un nouveau champ d'en-tête SIP "Flow-Timer" dont la syntaxe est définie à la Section 10.

Nom d'en-tête	Forme compacte	Référence
Flow-Timer	-	[RFC5626]

11.2 Paramètre "reg-id" de champ d'en-tête Contact

La présente spécification définit un nouveau paramètre de champ d'en-tête Contact appelé reg-id dans le sous registre "Paramètres de champ d'en-tête et valeurs de paramètres" suivant le registre créé par la [RFC3968]. Sa syntaxe est définie à la Section 10. L'information exigée est :

Champ d'en-tête	Nom du paramètre	Valeurs prédéfinies	Référence
Contact	reg-id	non	[RFC5626]

11.3 Paramètres d'URI SIP/SIPS

La présente spécification augmente le sous registre "Paramètres d'URI SIP/SIPS" du registre créé par la [RFC3969]. L'information exigée est :

Nom du paramètre	Valeurs prédéfinies	Référence
ob	non	[RFC5626]

11.4 Étiquette d'option SIP

La présente spécification enregistre une nouvelle étiquette d'option SIP, selon les lignes directrices du paragraphe 27.1 de la [RFC3261].

Nom : outbound

Description : cette étiquette d'option est utilisée pour identifier les UA et registraires qui prennent en charge les extensions pour les connexions initiées par le client. Un UA place cette option dans un en-tête Supported pour communiquer sa prise en charge de l'extension. Un registraire place cette étiquette d'option dans un en-tête Require pour indiquer à l'agent d'utilisateur qui s'enregistre que le registraire a utilisé des enregistrements selon les règles de lien définies dans la présente extension.

11.5 Code de réponse 430 (Flux défaillant)

Le présent document enregistre un nouveau code de réponse SIP (430 Flux défaillant) selon les lignes directrices du paragraphe 27.4 de la [RFC3261]. Ce code de réponse est utilisé par un mandataire de bordure pour indiquer au mandataire d'autorité qu'un flux spécifique à une instance d'UA est défaillant. D'autres flux à la même instance pourraient quand même réussir. Le mandataire d'autorité DEVRAIT tenter de transmettre à une autre cible (flux) avec le même identifiant d'instance et AOR. Les points d'extrémité ne devraient jamais recevoir une réponse 430. Si un point d'extrémité reçoit une réponse 430, il devrait la traiter comme un 400 (Mauvaise demande) selon les procédures normales du paragraphe 8.1.3.2 de la [RFC3261]. Ce code de réponse est défini par les information suivantes, qui ont été ajoutées au sous registre de méthodes et codes de réponse sous le registre des paramètres SIP.

Code de réponse	Référence
4xx Échec de demande	
430 Défaillance de flux	[RFC5626]

11.6 Code de réponse 439 (Le premier bond ne prend pas en charge Outbound)

Le présent document enregistre un nouveau code de réponse SIP (439 Le premier bond ne prend pas en charge Outbound) selon les lignes directrices du paragraphe 27.4 de la [RFC3261]. Ce code de réponse est utilisé par un registraire pour indiquer qu'il prend en charge la caractéristique "outbound" décrite dans cette spécification, mais que le premier mandataire sortant que l'utilisateur à travers lequel tente de s'enregistrer ne le fait pas. Noter que ce code de réponse est seulement approprié dans le cas où l'agent d'utilisateur enregistrant annonce la prise en charge du traitement sortant en incluant l'étiquette d'option "outbound" dans un champ d'en-tête Supported. Les mandataires NE DOIVENT PAS envoyer une réponse 439 à une demande qui ne contient pas un paramètre "reg-id" et une étiquette d'option "outbound" dans un champ d'en-tête Supported. Ce code de réponse est défini par les informations suivantes, qui ont été ajoutées au sous registre de méthodes et codes de réponse sous le registre des paramètres SIP.

Code de réponse	Référence
4xx Échec de demande	
439 Le premier bond ne prend pas en charge Outbound	[RFC5626]

11.7 Étiquette de caractéristique de support

Ce paragraphe enregistre une nouvelle étiquette de caractéristique de support, selon les procédures définies dans la [RFC2506]. L'étiquette est placée dans l'arborescence sip, qui est définie dans la [RFC3840].

Nom d'étiquette de caractéristique de support : sip.instance

Identifiant ASN.1 : 23

Résumé de la caractéristique de support indiquée par cette étiquette : cette étiquette de caractéristique contient une chaîne contenant un URN qui indique un identifiant univoque associé à l'instance d'UA enregistrant le contact.

Valeurs appropriées à utiliser avec cette étiquette de caractéristique : Chaîne (relation d'égalité).

L'étiquette de caractéristique est principalement destinée à être utilisée dans les applications, protocoles, services, ou mécanismes de négociation suivants : cette étiquette de caractéristique est la plus utile dans une application de communications, pour décrire les capacités d'un appareil, comme un téléphone ou un PDA.

Exemples d'utilisation typique : acheminement d'un appel à un appareil spécifique.

Normes ou documents en rapport : RFC 5626

Considérations de sécurité : cette étiquette de caractéristique de support peut être utilisée de façons qui affectent les comportements de l'application. Par exemple, l'extension de préférences de l'appelant SIP [RFC3841] permet que les décisions d'acheminement d'appel soient fondées sur les valeurs de ces paramètres. Donc, si un attaquant peut modifier les valeurs de cette étiquette, il pourrait être capable d'affecter le comportement des applications. Par suite, les applications qui utilisent cette étiquette de caractéristique de support DEVRAIENT fournir un moyen de s'assurer de son intégrité. De même, cette étiquette de caractéristique devrait seulement être crue comme valide quand elle vient de l'utilisateur ou agent d'utilisateur décrit par l'étiquette. Par suite, les protocoles pour transporter cette étiquette de caractéristique DEVRAIENT fournir un mécanisme pour garantir son authenticité.

12. Considérations sur la sécurité

Un des problèmes de sécurité clé de ce travail est de s'assurer qu'un attaquant ne peut pas capturer les sessions d'un utilisateur valide et causer l'envoi à l'attaquant de tous les appels destinés à cet utilisateur. Noter que l'intention n'est pas d'empêcher les attaques actives existantes sur le trafic SIP UDP et TCP, mais de s'assurer qu'aucune nouvelle attaque n'est ajoutée en introduisant le mécanisme "outbound".

Le cas simple est quand il n'y a pas de mandataire de bordure. Dans ce cas, la seule fois qu'une entrée peut être ajoutée à l'acheminement pour une AOR est quand l'enregistrement réussit. SIP protège déjà contre les attaquants capables de réussir à s'enregistrer, et ce schéma s'appuie sur cette sécurité. Certaines mises en œuvre ont examiné l'idée de juste sauvegarder l'identifiant d'instance sans le relier à l'AOR avec lequel il s'enregistre. Cette idée ne va pas fonctionner parce que l'UA d'un attaquant peut se faire passer pour l'identifiant d'instance d'un utilisateur valide et capturer les appels de cet utilisateur.

Le cas plus complexe implique un ou plusieurs mandataires de bordure. Quand un UA envoie une demande REGISTER à travers un mandataire de bordure au registraire, le mandataire de bordure insère une valeur du champ d'en-tête Path. Si l'enregistrement est authentifié avec succès, le registraire mémorise la valeur du champ d'en-tête Path. Plus tard, quand le registraire transmet une demande destinée à l'UA, il copie la valeur mémorisée du champ d'en-tête Path dans le champ d'en-tête Route de la demande et transmet la demande au mandataire de bordure.

La seule fois où un mandataire de bordure va acheminer sur un flux particulier est quand il a reçu un en-tête Route qui a les informations d'identifiant de flux qu'il a créé. Une demande entrante aurait obtenu ces informations du registraire. Le registraire va seulement sauvegarder ces informations pour une AOR donnée si l'enregistrement pour l'AOR a réussi ; et l'enregistrement va seulement réussir si l'UA peut être correctement authentifié. Même si un attaquant a falsifié des informations dans l'en-tête Path envoyé au registraire, l'attaquant ne sera pas capable de faire que le registraire accepte ces informations pour une AOR qui n'appartient pas à l'attaquant. Le registraire ne va pas passer ces mauvaises informations à d'autres, et d'autres ne seront pas trompés à contacter l'attaquant.

Les considérations sur la sécurité discutées dans les [RFC3261] et [RFC3327] sont aussi pertinentes pour ce document. Pour les considérations sur la sécurité de la génération des jetons de flux, voir aussi le paragraphe 5.2. Une discussion sur la façon d'empêcher le problème de l'avalanche de redémarrages est au paragraphe 4.5.

Le présent document ne change pas les mécanismes de sécurité de mise en œuvre obligatoire dans SIP. Les agents d'utilisateur sont déjà obligés de mettre en œuvre l'authentification par résumé lorsque la prise en charge de TLS est recommandée ; les serveurs mandataires sont déjà obligés de mettre en œuvre Digest et TLS.

13. Notes sur le fonctionnement des transports

Cette Section n'est pas normative.

La [RFC3261] exige des mandataires, registraires, et agents d'utilisateur qu'ils mettent en œuvre TCP et UDP mais les déploiements peuvent choisir quels protocoles de transport ils veulent utiliser. Les déploiements doivent être prudents dans le choix du transport à utiliser. De nombreuses caractéristiques et extensions de SIP, comme de grands corps de notification de présence, résultent en des demandes SIP qui peuvent être trop grandes pour être raisonnablement transportées sur UDP. La [RFC3261] déclare que quand une demande est trop grande pour UDP, l'appareil qui envoie la demande tente de passer à TCP. Il est important de noter que quand on utilise "outbound", cela ne va fonctionner que si l'UA a formé les deux flux sortants UDP et TCP. La présente spécification permet à l'UA de faire ainsi, mais dans la plupart des cas, il va probablement paraître plus naturel à l'UA de former seulement une connexion TCP sortante, plutôt que de former les deux flux UDP et TCP. Une des raisons clés pour que de nombreux déploiements choisissent de ne pas utiliser TCP est la difficulté de construction de mandataires qui puissent maintenir un très grand nombre de connexions TCP actives. De nombreux déploiements utilisent aujourd'hui SIP d'une façon telle que les messages sont assez petits pour qu'ils passent sur UDP mais ils ne peuvent pas tirer parti de toutes les fonctions offertes par SIP. Les déploiements qui utilisent seulement des connexions UDP sortantes vont échouer avec des messages SIP un peu grands.

14. Exigences

La présente spécification a été développée pour satisfaire les exigences suivantes :

1. être capable de détecter qu'un UA prend en charge ces mécanismes,
2. prendre en charge les UA derrière des NAT,
3. prendre en charge TLS à un UA sans nom DNS ou adresse IP stable,
4. détecter la défaillance d'une connexion et être capable de la corriger,
5. prendre en charge le réamorçage simultané de nombreux UA,
6. prendre en charge le réamorçage ou la réinitialisation d'un NAT,
7. minimiser la charge initiale de démarrage sur un mandataire,
8. prendre en charge des architectures avec des mandataires de bordure.

15. Remerciements

Francois Audet a agit comme berger de ce document, lisant des centaines de commentaires et incorporant de nombreuses corrections grammaticales ainsi qu'en stimulant les éditeurs pour "avancer le travail". Jonathan Rosenberg, Erkki Koivusalo, et Byron Campen ont fourni de nombreux commentaires et textes utiles. Dave Oran a apporté l'idée d'utiliser d'abord l'enregistrement le plus récent dans le mandataire. Alan Hawrylyshen est un des co-auteurs du document qui formait le texte initial de la présente spécification. De plus, de nombreux concepts ont leur origine dans une réutilisation de connexion de la réunion de l'IETF 60 qui incluait les auteurs, Jon Peterson, Jonathan Rosenberg, Alan Hawrylyshen, et Paul Kyzivat. L'équipe de conception TCP qui consistait en Chris Boulton, Scott Lawrence, Rajnish Jain, Vijay K. Gurbani, et Ganesh Jayadevan a fourni des apports et du texte. Nils Ohlmeier a fourni de nombreuses correections et l'expérience de mise en œuvre initiale. De plus, merci de leurs utiles commentaires aux personnes suivantes : Francois Audet, Flemming Andreasen, Mike Hammer, Dan Wing, Srivatsa Srinivasan, Dale Worely, Juha Heinanen, Eric Rescorla, Lyndsay Campbell, Christer Holmberg, Kevin Johns, Jeroen van Bemmel, Derek MacDonald, Dean Willis, et Robert Sparks.

16. Références

16.1 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)

[RFC2141] R. Moats, "Syntaxe des URN", mai 1997. *(Obsolète, voir* RFC8141)

[RFC2506] K. Holtman, A. Mutz, T. Hardie, "Procédure d'enregistrement d'étiquette de caractéristique de support", mars

1999. (BCP0031)

[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)

[RFC3263] J. Rosenberg, H. Schulzrinne, "Protocole d'initialisation de session (SIP) : Localisation des serveurs SIP", juin 2002. (*Remplace* RFC2543) *(P.S. ; MàJ par* RFC7984, RFC8898)

[RFC3327] D. Willis, B. Hoeneisen, "Champ d'en-tête d'extension du protocole d'initialisation de session (SIP) pour enregistrer des contacts non adjacents", décembre 2002. *(P.S.)*

[RFC3581] J. Rosenberg, H. Schulzrinne, "Extension au protocole d'initialisation de session (SIP) pour l'acheminement à réponse symétrique", août 2003. *(P.S.)*

[RFC3629] F. Yergeau, "UTF-8, un format de transformation de la norme ISO 10646", STD 63, novembre 2003.

[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

[RFC3841] J. Rosenberg, H. Schulzrinne, P. Kyzivat, "Préférences de l'appelant pour le protocole d'initialisation de session (SIP)", août 2004. *(P.S.)*

[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)

[RFC3969] G. Camarillo, "Registre des paramètres d'identifiant de ressource uniforme (URI) de l'IANA pour le protocole d'initialisation de session (SIP)", décembre 2004. (BCP0099 *; Mis à jour par* RFC 5727)

[RFC4122] P. Leach et autres, "Espace de noms d'URN d'identifiant univoque universel (UUID)", juillet 2005. *(P.S.)*

[RFC5234] D. Crocker, P. Overell, "BNF augmenté pour les spécifications de syntaxe : ABNF", janvier 2008. (STD0068)

[RFC5389] J. Rosenberg et autres, "Utilitaires de traversée de session pour les NAT (STUN)", octobre 2008. (*P.S. ; Remplace* RFC3489 *; remplacée par* RFC8489)

16.2 Références pour information

[RFC0768] J. Postel, "Protocole de datagramme d'utilisateur (UDP)", (STD 6), 28 août 1980.

[RFC0793] J. Postel (éd.), "Protocole de commande de transmission – Spécification du protocole du programme Internet DARPA", STD 7, septembre 1981. (*Remplacée par* RFC9293)

[RFC1035] P. Mockapetris, "Noms de domaines – Mise en œuvre et spécification", STD 13, novembre 1987. (*MàJ par* RFC1101, 1183, 1348, 1876, 1982, 1995, 1996, 2065, 2136, 2181, 2137, 2308, 2535, 2673, 2845, 3425, 3658, 4033, 4034, 4035, 4343, 5936, 5966, 6604, 7766, 8482, 8767)

[RFC2104] H. Krawczyk, M. Bellare et R. Canetti, "HMAC : Hachage de clés pour l'authentification de message", février 1997.

[RFC2131] R. Droms, "Protocole de configuration dynamique d'hôte", mars 1997. *(DS) (Mà J par* RFC3396, RFC4361, RFC5494, et RFC6849)

[RFC2782] A. Gulbrandsen, P. Vixie et L. Esibov, "Enregistrement de ressource DNS pour la spécification de la localisation des services (DNS SRV)", février 2000.

[RFC3320] R. Price, et autres, "Compression de signalisation (SigComp)", janvier 2003. (*MàJ par* RFC4896) *(P.S.)*

[RFC3489] J. Rosenberg et autres, "STUN - Simple traversée par le protocole de datagramme d'utilisateur (UDP) des traducteurs d'adresse réseau (NAT)", mars 2003. (*Obsolète, voir* RFC5389) *(P.S.)*

[RFC3986] T. Berners-Lee, R. Fielding et L. Masinter, "Identifiant de ressource uniforme (URI) : Syntaxe générique", STD 66, janvier 2005. *(P.S. ; MàJ par* RFC8820)

[RFC4340] E. Kohler et autres, "Protocole de contrôle d'encombrement de datagrammes (DCCP)", mars 2006. *(P.S.) (MàJ par* 6773)

[RFC4648] S. Josefsson, "Codages de données Base16, Base32 et Base64", octobre 2006. (*Remplace* RFC3548) *(P.S.)*

[RFC4960] R. Stewart, éd., "Protocole de transmission de commandes de flux (SCTP)", septembre 2007. (*Remplace* RFC2960, RFC3309 ; *P.S. ; Remplacée par* RFC9260)

[RFC5246] T. Dierks, E. Rescorla, "Version 1.2 du protocole de sécurité de la couche Transport (TLS)", août 2008. *(P.S.* ; *remplace* RFC3268, 4346, 4366 ; *MàJ* RFC4492 ; *rendue obsolète par la* RFC8446)

[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.)*

[RFC6080] D. Petrie, S. Channabasappa, éd.. "Cadre pour la livraison de profil d'agent d'utilisateur du protocole d'initialisation de session", mars 2011. *(P. S.)*

[RFC6314] C. Boulton et autres, "Pratiques de la traversée de NAT pour client-serveur SIP", juillet 2011. *(Information)*

Appendice A. Temps de retard d'enregistrement de flux par défaut

Le temps de base utilisé pour les temps de retard de réenregistrement de flux décrits au paragraphe 4.5 est configurable. Si la valeur de base-time-all-fail est réglée à la valeur par défaut de 30 secondes et si la valeur de base-time-not-failed est réglée à la valeur par défaut de 90 secondes, le tableau suivant montre la durée résultante pendant laquelle l'UA va attendre pour réessayer l'enregistrement.

Nombre d'échecs d'enreg.	Tous flux inutilisables	> 1 flux non en échec
0	0 s	0 s
1	30-60 s	90-180 s
2	1-2 min	3-6 min
3	2-4 min	6-12 min
4	4-8 min	12-24 min
5	8-16 min	15-30 min
6 ou plus	5-30 min	15-30 min

Appendice B. ABNF

Cet appendice contient l'ABNF défini plus tôt dans ce document.

```
CRLF = CR LF
double-CRLF = CR LF CR LF
CR = %x0D
LF = %x0A
Flow-Timer = "Flow-Timer" HCOLON 1*DIGIT
contact-params =/ c-p-reg / c-p-instance
c-p-reg = "reg-id" EQUAL 1*DIGIT                    ; 1 à (2^31 - 1)
c-p-instance = "+sip.instance" EQUAL DQUOTE "<" instance-val ">" DQUOTE
instance-val = 1*uric                               ; défini dans la RFC 3261
```

Adresse des auteurs

Cullen Jennings (éditeur)
Cisco Systems
170 West Tasman Drive
Mailstop SJC-21/2
San Jose, CA 95134
USA
téléphone : +1 408 902-3341
mél : fluffy@cisco.com

Rohan Mahy (éditeur)
mél : rohan@ekabal.com

François Audet (éditeur)
Skype Labs
mél : francois.audet@skypelabs.com