Groupe de travail Réseau
Request for Comments : 5360
Catégorie : Sur la voie de la normalisation
Traduction Claude Brière de L'Isle

J. Rosenberg, Cisco Systems
G. Camarillo, éditeur, Ericsson
D. Willis
octobre 2008

Cadre pour la communication fondée sur le consentement dans le protocole d'initialisation de session (SIP)

Statut du présent mémoire

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

Résumé

SIP prend en charge les communications pour plusieurs services, incluant l'audio en temps réel, la vidéo, le texte, la messagerie instantanée, et la présence. Dans sa forme actuelle, il permet des invitations aux sessions, des messages instantanés, et autres demandes à livrer d'une partie à une autre sans exiger de consentement explicite de la part du receveur. Sans un tel consentement, il est possible que SIP soit utilisé dans une intention malveillante, incluant des attaques d'amplification et de déni de service (DoS, *Denial of Service*). Le présent document identifie un cadre pour les communications fondées sur le consentement dans SIP.

Table des matières

1. Introduction

Le protocole d'initialisation de session (SIP, *Session Initiation Protocol*) [RFC3261] prend en charge les communications pour plusieurs services, incluant l'audio en temps réel, la vidéo, le texte, la messagerie instantanée, et la présence. Cette communication est établie par la transmission de diverses demandes SIP (comme INVITE et MESSAGE [RFC3428]) d'un initiateur au receveur avec lequel la communication est désirée. Bien que le receveur d'une telle demande SIP puisse rejeter la demande, et donc refuser la session, un réseau de serveurs mandataires de SIP va livrer une demande SIP à ses receveurs sans leur consentement explicite.

La réception de ces demandes sans consentement explicite peut causer un certain nombre de problèmes. Cela inclut les attaques d'amplification et de déni de service (DoS, *Denial of Service*). Ces problèmes sont décrits plus en détail dans un document d'accompagnement sur les exigences [RFC4453].

La présente spécification définit un cadre de base pour ajouter à SIP la communication fondée sur le consentement.

2. Définitions 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].

URI de receveur : URI de demande d'une demande sortante envoyée par une entité (par exemple, un agent d'utilisateur ou un mandataire). L'envoi d'une telle demande peut avoir été le résultat d'une opération de traduction.

Relais : tout serveur SIP, qu'il soit un mandataire, un agent d'utilisateur de bout en bout (B2BUA, *Back-to-Back User Agent*) ou un mélange des deux, qui reçoit une demande, traduit son URI de demande en un ou plusieurs URI de prochain bond (c'est-à-dire, des URI de receveur) et livre la demande à ces URI.

URI cible : URI de demande d'une demande entrante arrivant à un relais qui va effectuer une opération de traduction.

Logique de traduction : logique qui définit une opération de traduction à un relais. Cette logique inclut les URI cible et receveur de la traduction.

Opération de traduction : opération par laquelle un relais traduit l'URI de demande d'une demande entrante (c'est-à-dire, l'URI cible) en un ou plusieurs URI (c'est-à-dire, des URI de receveur) qui sont utilisés comme URI de demande d'une ou plusieurs demandes sortantes.

3. Relais et traductions

Les relais jouent un rôle clé dans ce cadre. Un relais est défini comme tout serveur SIP, qu'il soit un mandataire, un agent d'utilisateur de bout en bout) ou un hybride des deux, qui reçoit une demande, traduit son URI de demande en un ou plusieurs URI de prochain bond, et livre la demande à ces URI. L'URI de demande de la demande entrante est appelé un "URI cible"' et les URI de destination des demandes sortantes sont appelés des "URI de receveur", comme montré à la Figure 1.

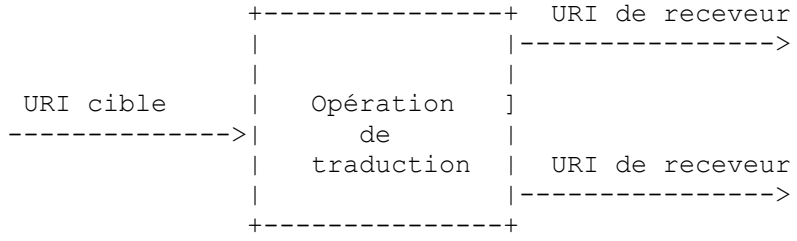


Figure 1 : Opération de traduction

Donc, un aspect essentiel d'un relais est celui de la traduction. Quand un relais reçoit une demande, il traduit l'URI de

demande (URI cible) en un ou plusieurs URI supplémentaires (URI de receveur). Par cette opération de traduction, le relais peut créer des demandes sortantes à un ou plusieurs URI de receveur supplémentaires, créant ainsi le problème du consentement.

Le problème du consentement est créé par deux types de traductions : les traductions fondées sur des données locales et les traductions qui impliquent des amplifications.

Les opérations de traduction fondées sur la politique locale ou des données locales (comme des enregistrements) sont le véhicule par lequel une demande est livrée directement à un point d'extrémité, quand il ne serait autrement pas possible de le faire. En d'autres termes, si un envoyeur de pourriels a l'adresse d'un utilisateur, "sip:utilisateur@exemple.com", il ne peut pas livrer une demande MESSAGE à l'UA (agent d'utilisateur) de cet utilisateur sans avoir accès aux données d'enregistrement qui transposent "sip:utilisateur@exemple.com" en l'agent d'utilisateur sur lequel cet utilisateur est présent. Donc, c'est l'usage de ces données d'enregistrement, et plus généralement, de la logique de traduction, qui est supposée être autorisée afin d'empêcher les communications non désirées. Bien sûr, si l'envoyeur de pourriels connaît l'adresse de l'agent d'utilisateur, il va être capable de lui livrer directement les demandes.

Les opérations de traduction qui résultent en plus d'un URI de receveur sont une source d'amplification. Les serveurs qui ne font pas de traductions, comme un serveur mandataire sortant, ne causent pas d'amplification. Par ailleurs, les serveurs qui effectuent des traductions (par exemple, les mandataires entrants d'autorité responsables d'un domaine SIP) peuvent causer une amplification si l'utilisateur peut être joint à plusieurs points d'extrémité (résultant ainsi en plusieurs URI de receveur).

La Figure 2 montre un relais qui effectue des traductions. Le client d'agent d'utilisateur dans la figure envoie une demande SIP à un URI représentant une ressource dans le domaine "exemple.com" (sip:resource@exemple.com). Cette demande peut passer à travers un mandataire local de sortie (non montré) mais arrive finalement à un serveur d'autorité pour le domaine "exemple.com". Ce serveur, qui agit comme relais, effectue une opération de traduction, traduisant l'URI cible en un ou plusieurs URI de receveur, qui peuvent (mais n'y sont pas obligés) appartenir au domaine "exemple.com". Ce relais peut être, par exemple, un serveur mandataire ou un service de liste d'URI [RFC5363].

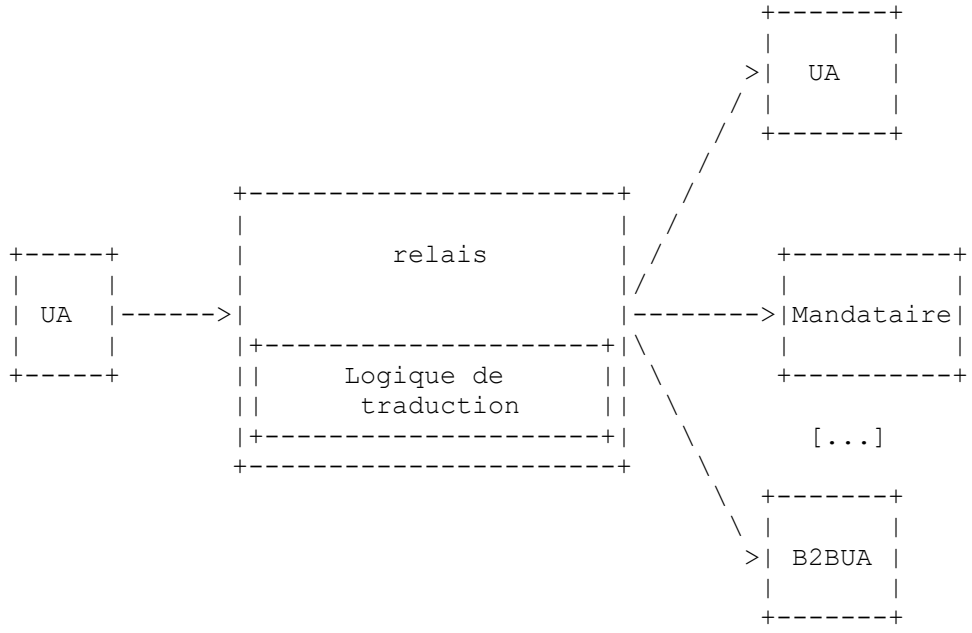


Figure 2 : Relais effectuant une traduction

Ce cadre permet à des receveurs potentiels d'une traduction d'accepter d'être les receveurs réels en donnant au relais qui effectue la traduction la permission de leur envoyer du trafic.

4. Architecture

La Figure 3 montre les éléments de l'architecture de ce cadre. La manipulation de la logique de traduction de relais cause

normalement l'envoi par le relais d'une demande de permission, qui à son tour cause l'accord ou le refus du receveur d'accorder au relais la permission de la traduction. Le paragraphe 4.1 décrit le rôle des permissions à un relais. Le paragraphe 4.2 discute les actions prises par un relais quand sa logique de traduction est manipulée par un client. Le paragraphe 4.3 discute des serveurs de mémorisation-transmission et de leurs fonctions. Le paragraphe 4.4 décrit comment de potentiels receveurs peuvent accorder aux relais les permissions de les ajouter à la logique de traduction du relais. Le paragraphe 4.5 discute des entités qui doivent mettre en œuvre ce cadre.

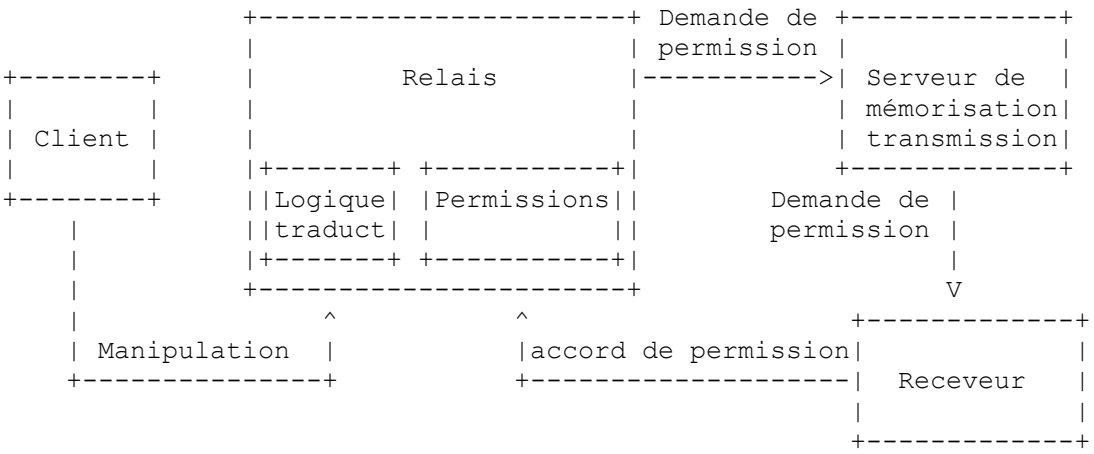


Figure 3 : Architecture de référence

4.1 Permissions à un relais

Les relais qui mettent en œuvre ce cadre obtiennent et mémorisent les permissions associées à leur logique de traduction. Ces permissions indiquent si un receveur particulier a ou non accepté de recevoir le trafic à un moment donné. Les receveurs qui n'ont pas donné au relais la permission de leur envoyer du trafic sont simplement ignorés par le relais quand il effectue une traduction.

En principe, les permissions sont valides tant que le contexte dans lequel elles ont été accordées est valide ou jusqu'à ce qu'elles soient révoquées. Par exemple, les permissions obtenues par un service SIP de liste d'URI qui distribue des demandes MESSAGE à un ensemble de receveurs va être valide tant que le service SIP de liste d'URI existe ou jusqu'à ce que les permissions soient révoquées.

De plus, si un receveur est supprimé de la logique de traduction d'un relais, le relais DEVRAIT supprimer les permissions relatives à ce receveur. Par exemple, si l'enregistrement d'un URI de contact expire ou est autrement terminé, le registraire supprime les permissions relatives à cette adresse de contact.

Il est aussi RECOMMANDÉ que les relais demandent aux receveurs de rafraîchir périodiquement leurs permissions. Si un receveur échoue à rafraîchir ses permissions pendant une certaine période, le relais DEVRAIT supprimer les permissions relatives à ce receveur.

Ce cadre ne donne aucune directive sur les valeurs des intervalles de rafraîchissement parce que des applications différentes peuvent avoir des exigences différentes sur le réglage de ces valeurs. Par exemple, un relais traitant des receveurs qui ne mettent pas en œuvre ce cadre peut choisir d'utiliser de plus longs intervalles entre les rafraîchissements. Le processus de rafraîchissement chez de tels receveurs doit être effectué manuellement par ses utilisateurs (car les receveurs ne mettent pas en œuvre ce cadre) et avoir des intervalles de rafraîchissement trop courts peut devenir une charge trop lourde pour ces utilisateurs.

4.2 Consentement à des manipulations sur une logique de traduction de relais

Ce cadre vise à assurer que tout relais particulier effectue seulement des traductions vers les destinations qui ont donné au relais la permission d'effectuer une telle traduction. Par conséquent, quand la logique de traduction d'un relais est manipulée (par exemple, un nouvel URI de receveur est ajouté) le relais obtient la permission du nouveau receveur afin d'installer la nouvelle logique de traduction. Les relais demandent aux receveurs la permission d'utiliser des demandes MESSAGE [RFC3428].

Par exemple, le relais qui héberge le service de liste d'URI à "sip:friends@exemple.com" effectue une traduction de cet URI cible à un ensemble d'URI de receveur. Quand un client (par exemple, l'administrateur de ce service de liste d'URI) ajoute "bob@exemple.org" comme nouvel URI de receveur, le relais envoie une demande MESSAGE à "sip:bob@exemple.org" demandant si il est ou non d'accord pour effectuer la traduction de "sip:friends@exemple.com" en "sip:bob@exemple.org". La demande MESSAGE porte dans son corps de message un document de permission qui décrit la traduction pour laquelle les permissions sont demandées et une partie lisible par l'homme qui décrit aussi la traduction. Si la réponse est positive, la nouvelle logique de traduction est installée au relais. C'est-à-dire, le nouvel URI de receveur est ajouté.

La partie lisible par l'homme est incluse afin que les agents d'utilisateur qui ne comprennent pas les documents de permission puissent quand même traiter la demande et l'afficher de façon correcte à l'utilisateur.

Le mécanisme à utiliser pour manipuler la logique de traduction d'un relais particulier dépend du relais. Deux mécanismes existants pour manipuler la logique de traduction sont le protocole d'accès à la configuration XML (XCAP, *XML Configuration Access Protocol*) [RFC4825] et les transactions REGISTER.

La Section 5 utilise un service de liste d'URI dont la logique de traduction est manipulée avec XCAP comme exemple de traduction, afin de spécifier ce cadre. Le paragraphe 5.10 explique comment appliquer ce cadre aux enregistrements, qui sont un type différent de traduction.

Dans tous les cas, les relais qui mettent en œuvre ce cadre DEVRAIENT avoir un moyen d'indiquer qu'un URI de receveur particulier est dans les états spécifiés dans la [RFC5362] (c'est-à-dire, en instance, en attente, erreur, refusé, ou accordé).

4.3 Serveurs de transmission différée

Quand une demande MESSAGE avec un document de permission arrive à l'URI de receveur auquel il a été envoyé par le relais, l'utilisateur receveur peut accorder ou refuser la permission nécessaire pour effectuer la traduction. Cependant, l'utilisateur receveur peut n'être pas disponible quand la demande MESSAGE arrive, ou il peut avoir exprimé des préférences pour bloquer toutes les demandes entrantes pendant un certain temps. Dans ce cas, un serveur de mémorisation transmission peut agir comme substitut de l'utilisateur et mettre en mémoire tampon les demandes MESSAGE entrantes, qui sont ensuite livrées à l'utilisateur quand il redevient disponible.

Il y a plusieurs mécanismes pour mettre en œuvre les services de mémorisation-transmission de message (par exemple, avec un message instantané à la passerelle de messagerie). Un de ces mécanismes peut être utilisé entre un agent d'utilisateur et son serveur de serveur de mémorisation-transmission pour autant qu'ils s'accordent sur le mécanisme à utiliser. Donc, ce cadre ne prend aucune disposition sur l'interface entre les agents d'utilisateur et leurs serveurs de mémorisation-transmission.

Noter que le même service de mémorisation-transmission de message peut traiter toutes les demandes MESSAGE entrantes pour un utilisateur quand ils sont hors ligne, et pas seulement les demandes MESSAGE avec un document de permission dans leur corps.

Bien que les serveurs de mémorisation-transmission effectuent une fonction utile et qu'on s'attende à ce qu'ils soient déployés dans la plupart des domaines, certains domaines ne vont pas les déployer à priori. Cependant, les agents d'utilisateur et les relais dans les domaines sans serveur de mémorisation-transmission peuvent quand même utiliser ce cadre de consentement.

Quand un relais demande des permissions à partir d'un agent d'utilisateur hors ligne qui n'a pas de serveur de mémorisation-transmission associé, le relais va obtenir une réponse d'erreur indiquant que sa demande MESSAGE ne pourra pas être livrée. Le client qui a tenté d'ajouter l'utilisateur hors ligne à la logique de traduction du relais va avoir notification de l'erreur (par exemple, en utilisant le paquetage d'événements Ajouts en cours [RFC5362]). Ce client PEUT tenter d'ajouter plus tard le même utilisateur, espérant qu'il soit en ligne. Le client peut découvrir si un utilisateur est ou non en ligne en utilisant un service de présence, par exemple.

4.4 Les receveurs accordent les permissions

Les documents de permission générés par un relais incluent des URI qui peuvent être utilisés par le receveur du document pour accorder ou refuser au relais la permission décrite dans le document. Les relais incluent toujours des URI SIP et peuvent inclure des URI HTTP [RFC2616] à cette fin. Par conséquent, les receveurs fournissent aux relais les permissions

en utilisant des demandes SIP PUBLISH ou des demandes HTTP GET.

4.5 Entités qui mettent en œuvre ce cadre

Le but de ce cadre est d'empêcher les relais d'exécuter des traductions vers des receveurs non consentants. Donc, tous les relais DOIVENT mettre en œuvre ce cadre afin d'éviter d'être utilisés pour effectuer des attaques (par exemple, des attaques d'amplification).

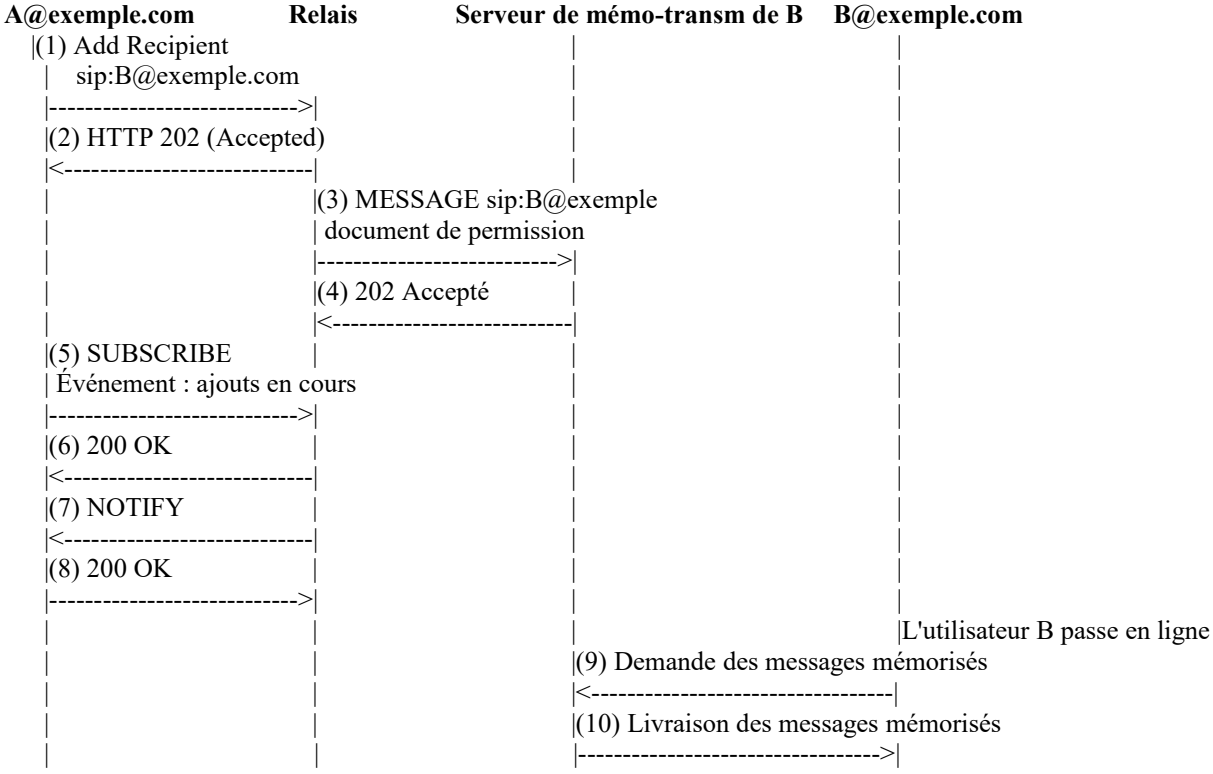
Ce cadre a été conçu en pensant à la rétro compatibilité afin que les agents d'utilisateur traditionnels (c'est-à-dire, les agents d'utilisateur qui ne mettent pas en œuvre ce cadre) puissent agir comme clients et receveurs avec un niveau acceptable de fonctionnalité. Cependant, il est RECOMMANDÉ que les agents d'utilisateur mettent en œuvre ce cadre, qui inclut de prendre en charge le paquetage d'événements Ajouts en cours spécifié dans la [RFC5362], le format des documents de permission spécifié dans la [RFC5361], et les champs d'en-tête et le code de réponse spécifiés dans le présent document, afin de réaliser une pleine fonctionnalité.

La seule exigence que ce cadre fait peser sur les serveurs de mémorisation-transmission est qu'ils doivent être capables de livrer des messages chiffrés et protégés en intégrité à leurs agents d'utilisateur, comme exposé à la Section 7. Cependant, cette exigence n'est pas spécifique de ce cadre mais est une exigence générale pour les serveurs de mémorisation-transmission.

5. Fonctionnement du cadre

Cette section spécifie ce cadre de consentement en utilisant un exemple du prototype de flux d'appel. Les éléments décrits dans la Section 4 (c'est-à-dire, les relais, traductions, et serveurs de mémorisation-transmission) jouent un rôle essentiel dans ce flux d'appels.

La Figure 4 montre le processus complet pour ajouter un URI de receveur ("sip:B@exemple.com") à la logique de traduction d'un relais. L'utilisateur A tente d'ajouter "sip:B@exemple.com" comme nouvel URI de receveur à la logique de traduction du relais (1). L'utilisateur A utilise XCAP [RFC4825] et le format de langage de balisage extensible (XML, *Extensible Markup Language*) pour représenter les listes de ressources [RFC4826] pour effectuer cet ajout. Comme le relais n'a pas la permission de "sip:B@exemple.com" d'effectuer les traductions vers cet URI, le relais place "sip:B@exemple.com" dans l'état En instance, comme spécifié dans la [RFC5362].



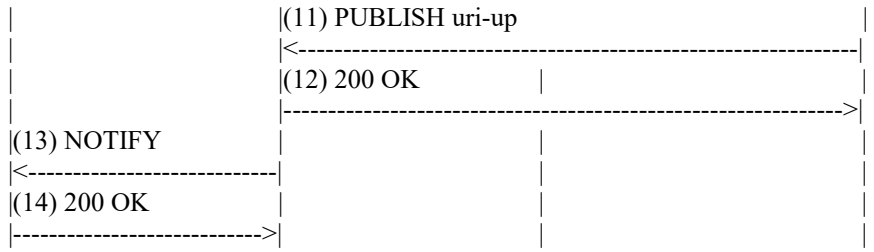


Figure 4 : Prototype de flux d'appels

5.1 Évitement d'amplification

Une fois que "sip:B@exemple.com" est dans l'état en instance, le relais doit demander à l'utilisateur B la permission en envoyant une demande MESSAGE à "sip:B@exemple.com". Cependant, le relais doit s'assurer qu'il n'est pas utilisé comme amplificateur pour lancer des attaques d'amplification.

Dans une telle attaque, l'attaquant va ajouter un grand nombre d'URI de receveur à la logique de traduction d'un relais. Le relais va alors envoyer une demande MESSAGE à chacun de ces URI de receveur. La bande passante générée par le relais va être bien plus grande que celle utilisée par l'attaquant pour ajouter ces URI de receveur à la logique de traduction du relais.

Ce cadre utilise un mécanisme d'autorisation fondé sur le crédit pour éviter l'attaque qu'on vient de décrire. Il exige que les utilisateurs qui ajoutent de nouveaux URI de receveur à une traduction génèrent une quantité de bande passante comparable à celle que le relais va générer quand il envoie des demandes MESSAGE vers ces URI de receveur. Quand XCAP est utilisé, cette exigence est satisfaite en ne permettant pas aux clients d'ajouter plus d'un URI par transaction HTTP. Quand une transaction REGISTER est utilisée, cette exigence est satisfaite en ne permettant pas aux clients d'enregistrer plus d'un contact par transaction REGISTER.

5.1.1 Comportement du relais

Les relais qui mettent en œuvre ce cadre NE DOIVENT PAS permettre aux clients d'ajouter plus d'un URI de receveur par transaction. Si un client qui utilise XCAP tente d'ajouter plus d'un URI de receveur dans une seule transaction HTTP, le serveur XCAP DEVRAIT retourner une réponse HTTP 409 (Conflit). Le serveur XCAP DEVRAIT décrire la raison du refus dans un corps XML utilisant l'élément <constraint-failure>, comme décrit dans la [RFC4825]. Si un client tente d'enregistrer plus d'un contact dans une seule transaction REGISTER, le registraire DEVRAIT retourner une réponse SIP 403 et expliquer la raison du refus dans sa phrase de cause (par exemple, un contact maximum par enregistrement).

5.2 Abonnement à l'état de permission

Les clients ont besoin d'un moyen d'être informés de l'état de l'opération demandée. Autrement, les utilisateurs pourraient attendre qu'une opération réussisse alors qu'en fait elle a déjà échoué. En particulier, si la cible de la demande de consentement n'était pas accessible et n'avait pas de serveur de mémorisation-transmission associé, le client a besoin de savoir si il peut réessayer la demande plus tard. Le paquetage d'événements SIP Ajouts en cours [RFC5362] est un moyen pour fournir cette information aux clients.

Les clients peuvent utiliser le paquetage d'événements SIP Ajouts en cours pour être informés de l'état des opérations qu'ils ont demandé. C'est-à-dire, le client va être informé quand une opération (par exemple, l'ajout d'un URI de receveur à la logique de traduction d'un relais) est autorisée (et donc exécutée) ou rejetée. Les clients utilisent l'URI cible de la traduction SIP manipulée pour s'abonner au paquetage d'événements "Ajouts en cours".

Dans notre exemple, après réception de la réponse du relais (2), l'utilisateur A s'abonne au paquetage d'événements "Ajouts en cours" auprès du relais (5). Cet abonnement tient l'utilisateur A informé de l'état des permissions (par exemple, accordé ou refusé) que le relais va obtenir.

5.2.1 Comportement du relais

Les relais DEVRAIENT prendre en charge le paquetage d'événements SIP "Ajouts en cours" spécifié dans la [RFC5362].

5.3 Demande de permission

Un relais demande aux receveurs potentiels la permission de les ajouter à sa logique de traduction en utilisant des demandes MESSAGE. Dans notre exemple, à réception de la demande d'ajout de l'utilisateur B à la logique de traduction du relais (1), le relais génère un demande MESSAGE (3) à "sip:B@exemple.com". Cette demande MESSAGE porte un document de permission, qui décrit la traduction qui a besoin d'être autorisée et porte un ensemble d'URI à utiliser par le receveur pour accorder ou refuser au relais la permission d'effectuer cette traduction. Comme l'utilisateur B n'est pas en ligne, la demande MESSAGE va être mise en mémoire tampon par le serveur de mémorisation-transmission de l'utilisateur B. L'utilisateur B va plus tard venir en ligne et autoriser la traduction en utilisant un de ces URI, comme décrit au paragraphe 5.6. La demande MESSAGE porte aussi une partie de corps qui contient les mêmes informations que le document de permission mais dans un format lisible par l'homme.

Quand l'utilisateur B utilise un des URI du document de permission pour accorder ou refuser les permissions, le relais doit s'assurer que c'est bien l'utilisateur B qui utilise cet URI, et non un attaquant. Le relais peut utiliser une des méthodes décrites au paragraphe 5.6 pour authentifier le document de permission.

5.3.1 Comportement du relais

Les relais qui mettent en œuvre ce cadre DOIVENT obtenir les permissions des receveurs potentiels avant de les ajouter à leur logique de traduction. Les relais demandent les permissions aux receveurs potentiels en utilisant des demandes MESSAGE.

Le paragraphe 5.6 décrit les méthodes qu'un relais peut utiliser pour authentifier les receveurs qui donnent au relais la permission d'effectuer une traduction particulière. Ces méthodes sont l'identité SIP [RFC4474], P-Asserted-Identity [RFC3325], un essai d'acheminement de retour, ou un résumé SIP. Les relais qui utilisent la méthode consistant en un essai d'acheminement de retour doivent envoyer leurs demandes MESSAGE à un URI SIPS, comme spécifié au paragraphe 5.6.

Les demandes MESSAGE envoyées pour demander les permissions DOIVENT inclure un document de permission et DEVRAIENT inclure une partie lisible par l'homme dans leur corps. La partie lisible par l'homme contient les mêmes informations que le document de permission (mais dans un format lisible par l'homme) incluant les URI pour accorder et refuser les permissions. Les agents d'utilisateur qui ne comprennent pas les documents de permission peuvent quand même traiter la demande et l'afficher d'une façon correcte à l'utilisateur, comme ils afficheraient n'importe quel autre message instantané. De cette façon, même si l'agent d'utilisateur ne met pas en œuvre ce cadre, l'utilisateur (humain) va être capable de cliquer manuellement sur l'URI correct afin d'accorder ou refuser les permissions. Ce qui suit est un exemple d'une demande MESSAGE qui porte une partie lisible par l'homme et un document de permission, qui suit le format spécifié dans la [RFC5361], dans son corps. Tous les champs d'en-tête ne sont pas montrés pour des raisons de simplicité.

```
MESSAGE sip:bob@exemple.org SIP/2.0
From: <sip:alices-friends@exemple.com>;tag=12345678
To: <sip:bob@exemple.org>
Content-Type: multipart/mixed;boundary="boundary1"

--boundary1
Content-Type: text/plain

Si vous consentez à recevoir le trafic envoyé à <sip:alices-friends@exemple.com>, veuillez utiliser un des URI
suivants : <sips:grant-1awdch5Fasddfce34@exemple.com> ou <https://exemple.com/grant-1awdch5Fasddfce34>.
Autrement, utilisez un des URI suivants : <sips:deny-23rCsdfgvdT5sdfgye@exemple.com> ou
<https://exemple.com/deny-23rCsdfgvdT5sdfgye>.

--boundary1
Content-Type: application/auth-policy+xml

<?xml version="1.0" encoding="UTF-8"?>
   <cp:ruleset
      xmlns="urn:ietf:params:xml:ns:consent-rules"
      xmlns:cp="urn:ietf:params:xml:ns:common-policy"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
```

```
    <cp:rule id="f1">
   <cp:conditions>
     <cp:identity>
       <cp:many/>
     </cp:identity>
     <receveur>
       <cp:one id="sip:bob@exemple.org"/>
     </receveur>
     <target>
       <cp:one id="sip:alices-friends@exemple.com"/>
     </target>
   </cp:conditions>
   <cp:actions>
     <trans-handling
       perm-uri="sips:grant-1awdch5Fasddfce34@exemple.com">
       grant</trans-handling>
     <trans-handling
       perm-uri="https://exemple.com/grant-1awdch5Fasddfce34">
       grant</trans-handling>
     <trans-handling
       perm-uri="sips:deny-23rCsdfgvdT5sdfgye@exemple.com">
       deny</trans-handling>
     <trans-handling
       perm-uri="https://exemple.com/deny-23rCsdfgvdT5sdfgye">
       deny</trans-handling>
   </cp:actions>
   <cp:transformations/>
  </cp:rule>
  </cp:ruleset>
--boundary1--
```

5.4 Structure du document de permission

Un document de permission est la représentation (par exemple, codée en XML) d'une permission. Un document de permission contient plusieurs éléments de données :

Identité de l'envoyeur : un URI représentant l'identité de l'envoyeur pour lequel les permissions sont accordées.

Identité du receveur original : un URI représentant l'identité du receveur original, qui est utilisé comme entrée pour l'opération de traduction. C'est aussi appelé un URI de cible.

Identité du receveur final : un URI représentant le résultat de la traduction. La permission accorde à l'envoyeur la capacité d'envoyer des demandes à l'URI de cible et pour le relais qui reçoit ces demandes, de les transmettre à cet URI. C'est aussi appelé un URI de receveur.

URI pour accorder la permission : les URI que les receveurs peuvent utiliser pour accorder au relais la permission d'effectuer la traduction décrite dans le document. Les relais DOIVENT prendre en charge l'utilisation des URI SIP et SIPS dans les documents de permission et PEUVENT prendre en charge l'utilisation des URI HTTP et HTTPS.

URI pour refuser la permission : les URI que les receveurs peuvent utiliser pour refuser au relais la permission d'effectuer la traduction décrite dans le document. Les relais DOIVENT prendre en charge l'utilisation des URI SIP et SIPS dans les documents de permission et PEUVENT prendre en charge l'utilisation des URI HTTP et HTTPS.

Les documents de permission peuvent contenir des caractères génériques *(wildcards)*. Par exemple, un document de permission peut demander une permission pour tout relais de transmettre des demandes venant d'un envoyeur particulier à un receveur particulier receveur. Un tel document de permission s'appliquerait à tout URI cible. C'est-à-dire, le champ contenant l'identité du receveur original correspondrait à tout URI. Cependant, l'URI de receveur NE DOIT PAS être remplacé par un caractère générique.

Les entités qui mettent en œuvre ce cadre DOIVENT prendre en charge le format des documents de permission défini dans

la [RFC5361] et PEUVENT prendre en charge d'autres formats.

Dans notre exemple, le document de permission dans la demande MESSAGE (3) envoyée par le relais contient les valeurs suivantes :
Identité de l'envoyeur : tout envoyeur
Identité du receveur original : sip:friends@exemple.com
Identité du receveur final : sip:B@exemple.com
URI pour accorder la permission : sips:grant-1awdch5Fasddfce34@exemple.com
URI pour accorder la permission : https://exemple.com/grant-1awdch5Fasddfce34
URI pour refuser la permission : sips:deny-23rCsdfgvdT5sdfgye@exemple.com
URI pour refuser la permission : https://exemple.com/deny-23rCsdfgvdT5sdfgye

On s'attend à ce que le champ Sender contienne souvent un caractère générique. Cependant, des scénarios impliquant des listes d'URI contenues dans la demande, comme celle décrite au paragraphe 5.9, peuvent exiger des documents de permission qui s'appliquent à un envoyeur spécifique. Dans les cas où l'identité de l'envoyeur importe, les relais DOIVENT authentifier les envoyeurs.

5.5 Notification de permission demandée

À réception de la demande MESSAGE (3), le serveur de mémorisation-transmission de l'utilisateur B la mémorise parce que l'utilisateur B est hors ligne à ce moment. Quand l'utilisateur B revient en ligne, il va chercher toutes les demandes que son serveur de mémorisation-transmission a mémorisées (9).

5.6 Accord de permission

Un receveur donne à un relais la permission d'exécuter la traduction décrite dans un document de permission en envoyant une demande SIP PUBLISH ou HTTP GET à un des URI pour accorder les permissions contenues dans le document. De même, un receveur refuse à un relais la permission d'exécuter la traduction décrite dans un document de permission en envoyant une demande SIP PUBLISH ou HTTP GET à un des URI pour refuser les permissions contenues dans le document. Les demande d'accorder ou refuser les permissions contiennent un corps vide.

Dans notre exemple, l'utilisateur B obtient le document de permission (10) qui a été reçu plus tôt par son serveur de mémorisation-transmission dans la demande MESSAGE (3). L'utilisateur B autorise la traduction décrite dans le document de permission reçu en envoyant une demande PUBLISH (11) à l'URI SIP pour accorder les permissions contenues dans le document de permission.

5.6.1 Comportement du relais

Les relais DOIVENT s'assurer que la demande SIP PUBLISH ou HTTP GET reçue a été générée par le receveur de la traduction et non par un attaquant. Les relais peuvent utiliser quatre méthodes pour authentifier ces demandes : identité SIP, P-Asserted-Identity [RFC3325], un essai d'acheminement de retour, ou un résumé SIP. Alors que les essais d'acheminement de retour peuvent être utilisés pour authentifier les demandes SIP PUBLISH et HTTP GET, l'identité SIP, P-Asserted-Identity, et le résumé SIP peuvent seulement être utilisées pour authentifier les demandes SIP PUBLISH. Le résumé SIP peut seulement être utilisé pour authentifier les receveurs qui partagent un secret avec le relais (par exemple, les receveurs qui sont dans le même domaine que le relais).

5.6.1.1 Identité SIP

Le mécanisme d'identité SIP [RFC4474] peut être utilisé pour authentifier l'envoyeur d'une demande PUBLISH. Le relais DOIT vérifier que l'origine de la demande PUBLISH est le propriétaire de l'URI de receveur dans le document de permission. Autrement, la demande PUBLISH DEVRAIT avoir une réponse 401 (Non autorisé) et son traitement NE DOIT PAS être poursuivi.

5.6.1.2 P-Asserted-Identity

Le mécanisme P-Asserted-Identity [RFC3325] peut aussi être utilisé pour authentifier l'envoyeur d'une demande PUBLISH. Cependant, comme discuté dans la [RFC3325], ce mécanisme est destiné à être utilisé seulement au sein de réseaux de serveurs SIP de confiance. C'est-à-dire, l'utilisation de ce mécanisme n'est applicable qu'à l'intérieur d'un

domaine administratif avec des politiques ayant fait l'objet d'un accord préalable.

Le relais DOIT vérifier que le générateur de la demande PUBLISH est le propriétaire de l'URI de receveur dans le document de permission. Autrement, la demande PUBLISH DEVRAIT avoir une réponse 401 (Non autorisé) et son traitement NE DOIT PAS être poursuivi.

5.6.1.3 Acheminement de retour

L'identité SIP fournit un bon mécanisme d'authentification pour les demandes PUBLISH entrantes. Néanmoins, l'identité SIP n'est pas encore largement disponible sur l'Internet public. C'est pourquoi un mécanisme d'authentification qui puisse être utilisé dès maintenant est nécessaire.

Les essais d'acheminement de retour ne fournissent pas le même niveau de sécurité que l'identité SIP, mais ils fournissent un niveau de sécurité meilleur que rien dans les architectures où le mécanisme d'identité SIP n'est pas disponible (par exemple, dans l'Internet actuel). Le relais génère un URI imprévisible (c'est-à-dire, avec une partie utilisateur cryptographiquement aléatoire) et le place dans le document de permission dans la demande MESSAGE (3). Le receveur doit envoyer une demande SIP PUBLISH ou HTTP GET à cet URI. Toute demande entrante envoyée à cet URI DEVRAIT être considérée comme authentifiée par le relais.

Noter que la méthode de l'acheminement de retour est la seule qui permette l'utilisation des URI HTTP dans les documents de permission. Les autres méthodes exigent l'utilisation d'URI SIP.

Les relais qui utilisent un essai d'acheminement de retour pour effectuer cette authentification DOIVENT envoyer la demande MESSAGE avec le document de permission à un URI SIPS. Cela assure que des attaquants ne vont pas obtenir l'accès à l'URI (imprévisible). Donc, le seul utilisateur capable d'utiliser l'URI (imprévisible) est le receveur de la demande MESSAGE. De même, les documents de permission envoyés par les relais en utilisant un essai d'acheminement de retour DOIVENT contenir seulement des URI sûrs (c'est-à-dire, SIPS et HTTPS) pour accorder et refuser les permissions. Une partie de ces URI (par exemple, la partie utilisateur d'un URI SIPS) DOIT être cryptographiquement aléatoire avec au moins 32 bits d'aléa.

Les relais peuvent passer des essais d'acheminement de retour à l'identité SIP en demandant simplement l'utilisation de l'identité SIP pour les demandes PUBLISH entrantes. C'est-à-dire, un tel relais va rejeter les demandes PUBLISH qui n'utilisent pas l'identité SIP.

5.6.1.4 Résumé SIP

Le mécanisme de résumé SIP peut être utilisé pour authentifier l'envoyeur d'une demande PUBLISH pour autant que cet envoyeur partage un secret avec le relais. Le relais DOIT vérifier que le générateur de la demande PUBLISH est le propriétaire de l'URI de receveur dans le document de permission. Autrement, la demande PUBLISH DEVRAIT avoir une réponse 401 (Non autorisé) et son traitement NE DOIT PAS être poursuivi.

5.7 Notification d'accord de permission

À réception de la demande PUBLISH (11), le relais envoie une demande NOTIFY (13) pour informer l'utilisateur A que la permission pour la traduction a été reçue et que la logique de traduction au relais a été mise à jour. C'est-à-dire, que "sip:B@exemple.com" a été ajouté comme URI de receveur.

5.8 Révocation de permission

À tout moment, si un receveur veut révoquer une permission, il utilise l'URI qu'il a reçu dans le document de permission pour refuser les permissions qu'il avait précédemment accordées. Si un receveur perd cet URI pour une raison quelconque, il doit attendre de recevoir une nouvelle demande produite par la traduction. Une telle demande va contenir un champ d'en-tête Trigger-Consent avec un URI. Ce champ d'en-tête Trigger-Consent va avoir un paramètre de champ d'en-tête target-uri qui identifie l'URI de cible de la traduction. Le receveur doit envoyer une demande PUBLISH avec un corps vide à l'URI contenu dans le champ d'en-tête Trigger-Consent afin de recevoir une demande MESSAGE du relais. Une telle demande MESSAGE va contenir un document de permission avec un URI pour révoquer la permission accordée précédemment.

La Figure 5 montre un exemple de la façon dont un utilisateur qui a perdu l'URI pour révoquer les permissions à un relais

peut obtenir un nouvel URI en utilisant le champ d'en-tête Trigger-Consent d'une demande entrante. L'utilisateur rejette une demande INVITE entrante (1) qui contient un champ d'en-tête Trigger-Consent. En utilisant l'URI de ce champ d'en-tête, l'utilisateur envoie une demande PUBLISH (4) au relais. À réception de la demande PUBLISH (4), le relais génère une demande MESSAGE (6) à l'utilisateur. Finalement, l'utilisateur révoque les permissions en envoyant une demande PUBLISH (8) au relais.

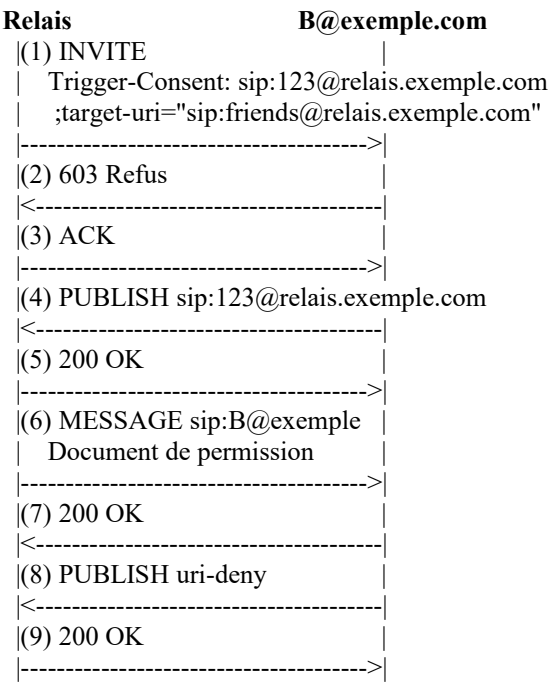


Figure 5 : Révocation de permission

5.9 Listes d'URI contenues dans la demande

Dans les scénarios décrits jusqu'à présent, un utilisateur ajoute des URI de receveur à la logique de traduction d'un relais. Cependant, le relais n'effectue pas traductions vers ces URI de receveur tant que les permissions ne sont pas obtenues.

Les services de liste d'URI qui utilisent des listes d'URI contenues dans la demande sont un cas particulier parce que le choix des URI de receveur est effectué en même temps que la tentative de communication. Un utilisateur place un ensemble d'URI de receveur dans une demande et l'envoie à un relais afin que le relais envoie une demande similaire à tous ces URI de receveur.

Les relais qui mettent en œuvre ce cadre de consentement et fournissent des services de liste d'URI contenues dans la demande se comportent d'un manière légèrement différente de celle des relais décrits jusqu'à présent. Ce type de relais tient aussi une liste des URI de receveur pour lesquels des permissions ont été reçues. Les clients manipulent aussi cette liste en utilisant un mécanisme de manipulation (par exemple, XCAP). Néanmoins, cette liste ne représente pas les URI de receveur de chaque traduction effectuée par le relais. Cette liste représente juste tous les URI de receveur pour lesquels des permissions ont été reçues -- c'est-à-dire, l'ensemble des URI qui vont être acceptés si une demande contenant une liste d'URI arrive au relais. Cet ensemble d'URI est un sur-ensemble des URI de receveur de toute traduction particulière que le relais effectue.

5.9.1 Comportement du relais

À réception d'une liste d'URI contenue dans une demande, le relais vérifie si il a ou non les permissions pour tous les URI contenus dans la liste d'URI entrante. Si il les a, le relais effectue la traduction. Si il manque des permissions pour un ou plusieurs URI, le relais NE DOIT PAS effectuer la traduction et DEVRAIT retourner une réponse d'erreur.

Un relais qui reçoit une liste d'URI contenue dans une demande avec un URI pour lequel le relais n'a pas de permission DEVRAIT retourner une réponse 470 (Consentement nécessaire). Le relais DEVRAIT ajouter un champ d'en-tête Permission manquante avec les URI pour lesquels le relais n'a pas de permissions.

La Figure 6 montre un relais qui reçoit une demande (1) contenant des URI pour lesquels le relais n'a pas de permission (le INVITE porte les URI de receveur dans son corps de message). Le relais rejette la demande avec une réponse 470 (Consentement nécessaire) (2). Cette réponse contient un champ d'en-tête Permission manquante avec les URI pour lesquels il n'a pas de permission.

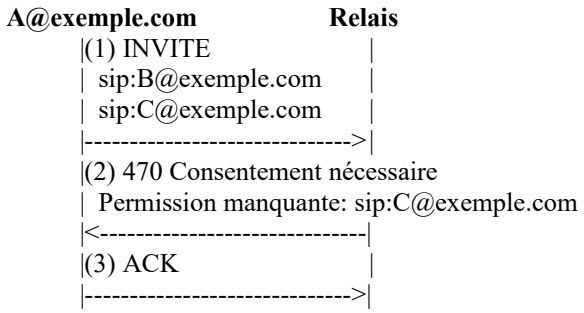


Figure 6 : INVITE avec une liste d'URI dans son corps

5.9.2 Définition du code de réponse 470

Une réponse 470 (Consentement nécessaire) indique que la demande qui a déclenché la réponse contenait une liste d'URI avec au moins un URI pour lequel le relais n'avait pas de permission. Un serveur d'agent d'utilisateur qui génère une réponse 470 (Consentement nécessaire) DEVRAIT y inclure un champ d'en-tête Permission manquante. Ce champ d'en-tête porte le ou les URI pour lesquels le relais n'a pas de permission.

Un client d'agent d'utilisateur qui reçoit une réponse 470 (Consentement nécessaire) sans champ d'en-tête Permission manquante a besoin d'un mécanisme de remplacement (par exemple, XCAP) pour découvrir pour quel ou quels URI il n'y avait pas de permissions.

Un client qui reçoit une réponse 470 (Consentement nécessaire) utilise un mécanisme de manipulation (par exemple, XCAP) pour ajouter ces URI à la liste des URI du relais. Le relais va obtenir de la façon usuelle les permissions pour ces URI.

5.9.3 Définition du champ d'en-tête Permission manquante

Les champs d'en-tête Permission manquante portent les URI pour lesquels un relais n'avait pas de permissions. Voici la syntaxe en forme Backus-Naur augmenté (ABNF) [RFC5234] pour le champ d'en-tête Permission manquante. Certains de ses éléments sont définis dans la [RFC3261].

```
Permission manquante  =  "Permission manquante" HCOLON per-miss-spec *( COMMA per-miss-spec )
    per-miss-spec  =  ( name-addr / addr-spec ) *( SEMI generic-param )
```

Voici un exemple d'un champ d'en-tête Permission manquante :

```
   Permission manquante: sip:C@exemple.com
```

5.10 Enregistrements

Même si l'exemple utilisé pour spécifier ce cadre avait été un service de liste d'URI, ce cadre s'applique à tout type de traduction (c'est-à-dire, pas seulement aux services de liste d'URI). Les enregistrements sont un type différent de traduction qui mérite discussion.

Les enregistrements sont un type de traduction spécial. L'utilisateur qui s'enregistre a une relation de confiance avec le registraire dans son domaine de rattachement. Ce n'est pas le cas quand un utilisateur donne des permissions de n'importe quel type à un relais dans un domaine différent.

Traditionnellement, les transactions REGISTER ont effectué deux opérations en même temps : établir une traduction et autoriser l'utilisation de cette traduction. Par exemple, un utilisateur qui enregistre son URI de contact actuel donne la

permission au registraire de transmettre le trafic envoyé à l'adresse de rattachement (AoR, *Address of Record*) de l'utilisateur à l'URI de contact enregistré. Cela fonctionne bien quand l'entité qui enregistre est la même que celle qui va recevoir le trafic un peu plus tard (par exemple, l'entité qui reçoit le trafic sur la même connexion utilisée pour l'enregistrement comme décrit dans la [RFC5626]). Cependant, ce schéma crée un potentiel d'attaques en rapport avec les enregistrements de tiers.

Un attaquant lie, via un enregistrement, son AoR avec l'URI de contact d'une victime. La victime va maintenant recevoir le trafic non sollicité qui était à l'origine adressé à l'attaquant.

Le processus d'autorisation d'un enregistrement est montré à la Figure 7. L'utilisateur A effectue un enregistrement de tiers (1) et reçoit une réponse 202 (Accepté) (2).

Comme le relais n'a pas la permission de "sip:a@ws123.exemple.com" pour effectuer des traductions sur cet URI de receveur, le relais place "sip:a@ws123.exemple.com" dans l'état "en instance". Une fois que "sip:a@ws123.exemple.com" est dans l'état "Permission en cours", le registraire doit demander à "sip:a@ws123.exemple.com" sa permission en envoyant une demande MESSAGE (3).

Après la réception de la réponse du relais (2), l'utilisateur A s'abonne au paquetage d'événements Ajouts en cours chez le registraire (5). Cet abonnement tient l'utilisateur informé de l'état des permissions (par exemple, accordé ou refusé) que le registraire va obtenir. Le reste du processus est similaire à celui décrit à la Section 5.

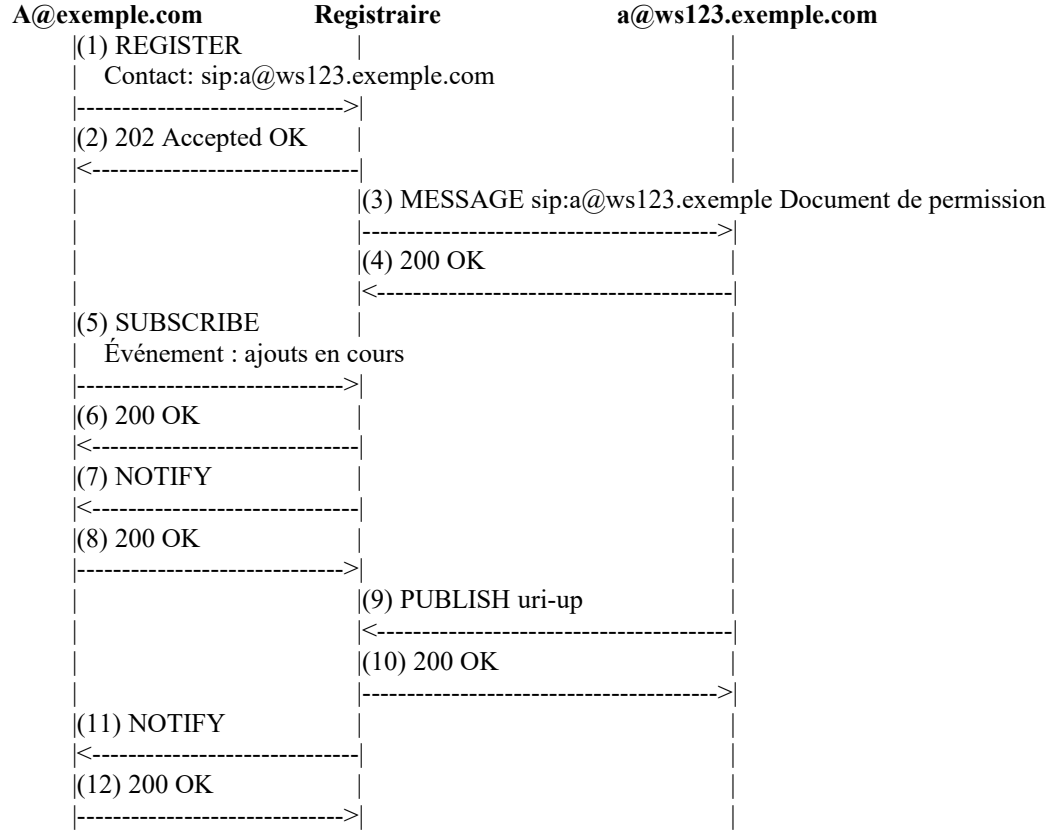


Figure 7 : Enregistrement

Les documents de permission générés par les registraires sont normalement très généraux. Par exemple, dans un tel document un registraire peut demander à un receveur la permission de transmettre toute demande de tout envoyeur à l'URI du receveur. C'est le type de granularité que ce cadre a l'intention de fournir pour les enregistrements. Les utilisateurs qui veulent définir comment les demandes entrantes sont traitées avec une granularité plus fine (par exemple, les demandes provenant de l'utilisateur A sont seulement acceptées entre 9:00 et 11:00) devraient utiliser d'autres mécanismes comme le langage de traitement d'appel (CPL, *Call Processing Language*) [RFC3880].

Noter que, comme indiqué précédemment, les agents d'utilisateur qui utilisent la même connexion pour s'enregistrer et pour recevoir le trafic provenant du registraire, comme décrit dans la [RFC5626], n'ont pas besoin d'utiliser le mécanisme décrit dans ce paragraphe.

Un agent d'utilisateur enregistré par un tiers peut être incapable d'utiliser les mécanismes d'identité SIP, P-Asserted-Identity, ou résumé SIP pour prouver au registraire que l'agent d'utilisateur est le propriétaire de l'URI enregistré (par exemple, sip:utilisateur@192.0.2.1) qui est l'URI de receveur de la traduction. Dans ce cas, l'acheminement de retour DOIT être utilisé.

5.11 Relais générant du trafic vers les receveurs

Les relais qui génèrent du trafic vers les receveurs doivent s'assurer que ces receveurs peuvent révoquer à tout moment les permissions qu'ils ont données. Trigger-Consent aide à le faire.

5.11.1 Comportement du relais

Un relais qui exécute une traduction impliquant d'envoyer une demande à un URI dont les permissions ont été obtenues précédemment DEVRAIT ajouter un champ d'en-tête Trigger-Consent à la demande. L'URI dans le champ d'en-tête Trigger-Consent DOIT avoir un paramètre de champ d'en-tête target-uri qui identifie l'URI cible de la traduction.

À réception d'une demande PUBLISH adressée à l'URI qu'un relais a précédemment placé dans un champ d'en-tête Trigger-Consent, le relais DEVRAIT envoyer une demande MESSAGE à l'URI de receveur correspondant avec un document de permission. Donc, le relais doit être capable de corréler l'URI qu'il place dans le champ d'en-tête Trigger-Consent avec l'URI de receveur de la traduction.

5.11.2 Définition du champ d'en-tête Trigger-Consent

Voici la syntaxe en forme Backus-Naur augmenté (ABNF) [RFC5234] du champ d'en-tête Trigger-Consent. Certains de ses éléments sont définis dans la [RFC3261].

```
Trigger-Consent     =  "Trigger-Consent" HCOLON trigger-cons-spec *( COMMA trigger-cons-spec )
   trigger-cons-spec  = ( SIP-URI / SIPS-URI ) *( SEMI trigger-param )
   trigger-param      = target-uri / generic-param
   target-uri         = "target-uri" EQUAL LDQUOT *( qdtext / quoted-pair ) RDQUOT
```

Le paramètre de champ d'en-tête target-uri DOIT contenir un URI.

Voici un exemple d'un champ d'en-tête Trigger-Consent :

```
   Trigger-Consent: sip:123@relais.exemple.com
                ;target-uri="sip:friends@relais.exemple.com"
```

6. Considérations relatives à l'IANA

Selon les paragraphes qui suivent, l'IANA a enregistré un code de réponse SIP, deux champs d'en-tête SIP, et un paramètre de champ d'en-tête SIP.

6.1 Enregistrement du code de réponse 470

L'IANA a ajouté le nouveau code de réponse suivant au sous registre Méthodes et Codes de réponse du registre des paramètres SIP.

Numéro de code de réponse : 470
Phrase de cause par défaut : Consentement nécessaire
Référence : [RFC5360]

6.2 Enregistrement du champ d'en-tête Trigger-Consent

L'IANA a ajouté le nouveau champ d'en-tête SIP suivant au sous registre des champs d'en-tête dans le registre des

paramètres de SIP :

Nom d'en-tête : Trigger-Consent
Forme compacte : aucune
Référence : [RFC5360]

6.3 Enregistrement du champ d'en-tête Permission manquante

L'IANA a ajouté le nouveau champ d'en-tête SIP suivant au sous registre des champs d'en-tête du registre des paramètres SIP :

Nom d'en-tête : Permission manquante
Forme compacte : aucune
Référence : [RFC5360]

6.4 Enregistrement du paramètre de champ d'en-tête target-uri

L'IANA a enregistré le paramètre "target-uri" du champ d'en-tête Trigger-Consent dans le sous registre des Paramètres de champ d'en-tête et des valeurs de paramètres dans le registre des paramètres SIP :

Champ d'en-tête	Nom de paramètre	Valeurs prédéfinies	Référence
Trigger-Consent	target-uri	Non	[RFC5360]

7. Considérations sur la sécurité

La sécurité a été discutée tout au long de ce document. Cependant, certaines questions méritent une attention particulière.

Les relais mettent généralement en œuvre plusieurs mécanismes de sécurité relatifs à l'authentification et l'autorisation du client. Les clients sont normalement authentifiés avant qu'ils puissent manipuler la logique de traduction d'un relais. De plus, les clients sont normalement aussi authentifiés et parfois doivent effectuer des tâches de prévention des pourriels [RFC5039] quand ils envoient du trafic à un relais. Il est important que les relais mettent en œuvre ces types de mécanismes de sécurité. Cependant, ils sortent du domaine d'application de ce cadre. Même avec ces mécanismes, il est toujours besoin que les relais mettent en œuvre ce cadre parce que l'utilisation de ces mécanismes n'empêche pas les clients autorisés d'ajouter des receveurs à une traduction sans leur consentement. Par conséquent, les relais qui effectuent des traductions DOIVENT mettre en œuvre ce cadre.

Noter que, comme indiqué précédemment, les agents d'utilisateur qui utilisent la même connexion pour enregistrer et recevoir du trafic provenant du registraire, comme décrit dans la [RFC5626], n'ont pas besoin d'utiliser ce cadre. Donc, un registraire qui n'accepte pas d'enregistrements de tiers n'a pas besoin de mettre en œuvre ce cadre.

Comme mentionné au paragraphe 5.6.1.3, quand les essais d'acheminement de retour sont utilisés pour authentifier les receveurs qui accordent ou refusent des permissions, les URI utilisés pour accorder ou refuser les permissions ont besoin d'être protégés contre les attaquants. Les URI SIPS fournissent un bon outil pour satisfaire cette exigence, comme décrit dans la [RFC5361]. Quand les serveurs de mémorisation-transmission sont utilisés, l'interface entre un agent d'utilisateur et son serveur de mémorisation-transmission est fréquemment non fondé sur SIP. Dans ce cas, SIPS ne peut pas être utilisé pour sécuriser ces URI. Les mises en œuvre de serveurs de mémorisation-transmission DOIVENT fournir un mécanisme pour livrer des messages chiffrés et protégés en intégrité à leurs agents d'utilisateur.

Les information fournies par le paquetage d'événements Ajouts en cours peuvent être sensibles. Pour cette raison, comme décrit dans la [RFC5362], les relais doivent utiliser un moyen fort pour l'authentification et la confidentialité des informations. Les URI SIPS sont un bon mécanisme pour satisfaire cette exigence.

Les documents de permission peuvent révéler des informations sensibles. Des attaquants peuvent tenter de les modifier afin que les clients accordent ou refusent des permissions différentes de celles qu'ils pensent accorder ou refuser. Pour cette raison, il est RECOMMANDÉ que les relais utilisent de forts moyens pour la protection de l'intégrité et de la confidentialité des informations quand ils envoient des documents de permission aux clients.

Le mécanisme utilisé pour porter des informations aux clients DEVRAIT assurer l'intégrité et la confidentialité des

informations. Afin de réaliser cela, un mécanisme de chiffrement SIP de bout en bout, comme S/MIME, comme décrit dans la [RFC3261], DEVRAIT être utilisé.

Si des moyen forts de sécurité de bout en bout (comme ci-dessus) ne sont pas disponibles, il est RECOMMANDÉ que la sécurité bond par bond fondée sur TLS et les URI SIPS, comme décrit dans la [RFC3261], soit utilisée.

8. Remerciements

Henning Schulzrinne, Jon Peterson, et Cullen Jennings ont fourni des idées utiles sur ce document. Ben Campbell, AC Mahendran, Keith Drage, et Mary Barnes on effectué une relecture attentive de ce document.

9. Références

9.1 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)

[RFC2616] R. Fielding et autres, "Protocole de transfert hypertexte -- HTTP/1.1", juin 1999. *(D.S., MàJ par* 2817, 6585)

[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)

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

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

[RFC5361] G. Camarillo, "Format de document pour une demande de consentement", octobre 2008. *(P.S.)*

[RFC5362] G. Camarillo, "Paquetage d'événements Ajouts en instance du protocole d'initialisation de session (SIP)", octobre 2008. *(P.S.)*

[RFC5363] G. Camarillo, A.B. Roach, "Cadre et considérations sur la sécurité pour les services URI-List du protocole d'initialisation de session (SIP)", octobre 2008. *(P.S.)*

9.2 Références pour information

[RFC3325] C. Jennings, J. Peterson et M. Watson, "Extensions privées au protocole d'initialisation de session (SIP) pour l'assertion d'identité au sein de réseaux de confiance", novembre 2002. *(Information ; MàJ par* RFC8217)

[RFC3880] J. Lennox, X. Wu, H. Schulzrinne, "Langage de traitement d'appel (CPL) : un langage pour le contrôle d'usager des services de téléphonie Internet", octobre 2004.

[RFC4453] J. Rosenberg et autres, "Exigences pour les communications fondées sur le consentement dans le protocole d'initialisation de session (SIP)", avril 2006. *(Information)*

[RFC4474] J. Peterson et C. Jennings, "Améliorations de la gestion d'identité authentifiée dans le protocole d'initialisation de session (SIP)", août 2006. *(P.S. ; Remplacée par* RFC8224)

[RFC4825] J. Rosenberg, "Protocole d'accès de configuration (XCAP) du langage de balisage extensible (XML)", mai 2007. *(P.S.)*

[RFC4826] J. Rosenberg, "Formats du langage de balisage extensible (XML) pour représenter des listes de ressources",

mai 2007. *(P.S.)*

[RFC5039] J. Rosenberg, C. Jennings, "Le protocole d'initialisation de session (SIP) et les pourriels", janvier 2008. *(Information)*

[RFC5626] C. Jennings, R. Mahy, F. Audet, éd., "Gestion des connexions initiées par le client dans le protocole d'initialisation de session (SIP)", octobre 2009. *(MàJ* RFC3261, RFC3327*) (P. S.)*

Adresse des auteurs

Jonathan Rosenberg
Cisco
Iselin, NJ 08830
USA
mél : jdrosen@cisco.com
URI : http://www.jdrosen.net

Gonzalo Camarillo (éditeur)
Ericsson
Hirsalantie 11
Jorvas 02420
Finland
mél : Gonzalo.Camarillo@ericsson.com

Dean Willis
3100 Independence Pkwy #311-164
Plano, TX 75075
USA
mél : dean.willis@softarmor.com

Déclaration complète de droits de reproduction