Retour d'expérience sur la ToIP et la messagerie instantanée

Philippe Sultan
INRIA - MIRIAD
Domaine de Voluceau, Rocquencourt, B.P. 105, 78153 Le Chesnay Cedex France
philippe.sultanimageinria.fr

Résumé

La téléphonie sur IP (ou ToIP[1]) et la messagerie instantanée (ou IM[2]) prennent une place de plus en plus importante dans le système d'information, mettant à disposition des utilisateurs et des administrateurs de nombreux outils d'exploitation (clients VoIP – IM, serveurs, IP-PBX, etc.).

L'INRIA mène depuis plusieurs mois des actions visant à l'expérimentation et au déploiement des architectures de communication nouvelles que sont la ToIP et la messagerie instantanée.

Nous présentons ici les protocoles SIP et XMPP (Jabber), sur lesquels reposent nos architectures ToIP et IM, elles-mêmes détaillées par la présentation des composants logiciels utilisés (Asterisk, OpenSER, Jabberd2), et des travaux menés dans le cadre du projet SIP.edu.

Les perspectives d'évolution des architectures présentées et des protocoles SIP et XMPP concluent cet article.

Mots clefs

SIP, XMPP, Jabber, Asterisk, OpenSER, Jabberd2, VoIP, ToIP.

1 Introduction

La ToIP est apparue au milieu des années 90, suite à la publication de la recommandation H.323 [1] (origine ITU) et du standard SIP [2] - Session Initiation Protocol - (origine IETF). Une compétition naturelle entre standards s'est alors développée, consécutive à la simultanéité de la publication des spécifications.

Aujourd'hui, le protocole SIP semble s'être imposé sur ses concurrents, tant chez les opérateurs de téléphonie que dans le monde de l'entreprise, voire même chez les particuliers. Prenons-en pour preuve le choix de SIP comme protocole de communication de l'architecture IMS (IP Multimedia Subsystem) dans le cadre de la convergence fixe-mobile, les IP-PBX actuels qui offrent ajourd'hui pratiquement tous une interface SIP, et le fait que Microsoft ait supprimé son client Netmeeting (H.323) de son OS grand public et introduit SIP dans sa suite Office Live Communication Server.

Le protocole XMPP – Extensible Messaging and Presence Protocol, aussi appelé Jabber[3], est issu de l'IETF (RFC 3920 [3], 3921 [4], 3922 [5], 3923 [6]) dans le courant de l'année 2004[4]. La spécification XMPP définit une architecture et un protocole de communication pour un service de messagerie instantanée et de gestion de présence. Les spécifications actuelles et futures sont désignées par le terme XEP (XMPP Extension Proposal), et rédigées dans le cadre de la XSF[5] (XMPP Standards Foundation).

Il est intéressant de constater actuellement l'extension du protocole XMPP à la gestion des sessions multimédia (voix/video) au travers du futur standard Jingle [7], alors que la suite protocolaire SIMPLE (SIP for Instant Messaging and Presence Leveraging Extensions [8]) fait de SIP un protocole utilisable pour l'IM. Les deux standards couvrant des domaines qui tendent à se confondre, on peut se demander si une nouvelle « guerre des protocoles » aura lieu.

Aujourd'hui, de nombreuses implémentations logicielles des protocoles SIP et XMPP, portant sur l'ensemble des composants de chaque architecture (client SIP/XMPP, serveur XMPP, Proxy/Registrar/B2BUA SIP, ..) sont disponibles, souvent sous la forme de logiciels libres. Nous présentons ici les architectures SIP et XMPP déployées à partir d'implémentations libres (Asterisk, OpenSER et Jabberd2) dans le cadre du projet SIP.edu et des travaux de mise en oeuvre d'un service d'IM pour l'INRIA.

2 SIP : le standard pour la ToIP

2.1 Le protocole SIP

Initialement envisagé pour gérer les sessions interactives multimedia, le protocole SIP fait toujours l'objet de groupes de travail à l'IETF[6].

[1]ToIP est l'acronyme anglais de Telephony over IP.

[2]IM sont les initiales du terme anglais Instant Messaging.

[3]Pour une explication des quelques différences entre XMPP et Jabber, voir http://www.saint-andre.com/jabber/xmpp+jabber.html.

[4]Les travaux de développement et de spécification avaient débuté à la fin des années 1990.

[5]La XSF est un organisme de standardisation, descendant direct de la JSF (Jabber Software Foundation).

[6]Ces groupes de travail traitent plus généralement des applications temps réel :

SIP est un protocole de signalisation, et n'est de ce fait pas chargé de transporter les flux média, généralement véhiculés par le protocole RTP [9]. Cependant, complété par le protocole SDP (Session Description Protocol), il offre un mécanisme de négociation de session multimedia (ex. codecs voix/vidéo). En fait, toute application nécessitant un mécanisme de gestion de session peut être l'objet d'un support SIP, et l'on peut ainsi envisager d'utiliser SIP pour l'établissement de sessions de jeux vidéo.

Si à l'origine la plupart des implémentations SIP ont été développées sur UDP, on trouve aujourd'hui de plus en plus de « piles logicielles » SIP/TCP, et plus rarement SIP/SCTP. Une raison majeure de préférer le transport de SIP sur TCP est qu'avec l'extension du protocole, les messages SIP deviennent trop volumineux pour ne pas être fragmentés au niveau IP s'il sont transportés dans UDP, et peuvent être refusés par certains firewalls ou routeurs.

2.1.1 Identifiant et localisation

URI SIP

L'identifiant d'un utilisateur est généralement une URI[7] SIP de la forme sip:user@domain.

« user » identifie un utilisateur par exemple par un login ou un numéro de téléphone, « domain » correspond à un nom de domaine DNS ou à une adresse de machine.

Beaucoup d'autres types d'URIs sont autorisés, les URIs SIP se conformant à la syntaxe de la RFC 2396 [10].

Localisation DNS, enregistrement DNS SRV

Pour retransmettre une requête à destination d'une URI sip:user@domain, il faut faire correspondre le domaine DNS donné dans l'URI à une adresse IP de serveur, généralement un proxy SIP. L'architecture DNS joue ici un rôle majeur, puisque la correspondance est réalisée par un enregistrement DNS de type SRV.

Pour une zone DNS particulière, un enregistrement DNS SRV[8] associe une machine (un enregistrement DNS de type A plus précisément) à un service, de la même manière que l'enregistrement MX pour SMTP.

Exemple :

```
_sip._udp.inria.fr.     172800  IN      SRV     0
0 5060 softswitch.inria.fr.
```

L'entrée précédente dans le serveur DNS du domaine « inria.fr » indique que le proxy SIP desservant le domaine « inria.fr » en UDP est accessible à l'adresse « softswitch.inria.fr » sur le port 5060.

http://www.ietf.org/html.charters/wg-dir.html#Real-time%20Applications%20and%20Infrastructure%20Area.

[7]Uniform Resource Identifier

[8]Plus d'information sur ce type d'enregistrement : http://www.voip-info.org/wiki-DNS+SRV

2.1.2 Format des messages

Le format des messages SIP (codés en ASCII) est repris sur celui utilisé par HTTP. Le protocole est constitué d'un ensemble de requêtes (appelées "méthodes" dans la terminologie SIP) auxquelles correspond un ensemble de réponses numérotées par un code de retour.

Les méthodes principalement utilisées sont :

- REGISTER pour s'enregistrer sur un serveur ;
- INVITE pour établir une session ;
- ACK pour confirmer l'établissement de la session ;
- BYE pour terminer une session en cours.

La flexibilité caractérise le protocole SIP et autorise les nombreux groupes de travail impliqués à étendre le protocole vers de nouvelles applications. Ainsi, de nouvelles méthodes viennent s'ajouter régulièrement, par exemple, pour la gestion de présence (SUBSCRIBE, NOTIFY).

Du côté des réponses, on retrouve des codes similaires à ceux d'HTTP :

- 100 Trying ;
- 180 Ringing ;
- 200 OK ;
- 404 Not Found ;
- 486 Busy ;
- etc.

2.2 Architecture

Une architecture SIP classique repose sur quatre éléments fonctionnels : User Agent (Client et Server), Registrar, Proxy et B2BUA (Back to Back User Agent).

2.2.1 User Agent

La notion de client et de serveur en SIP n'est pas aussi tranchée qu'en HTTP ou XMPP. Les fonctionnalités UAS (User Agent Server) et UAC (User Agent Client) sont incluses dans tout terminal SIP de type téléphone physique ou logiciel.

La fonctionnalité UAS permet de traiter les requêtes d'établissement et de fermeture de session et de répondre à partir des codes de retour disponibles. La fonctionnalité UAC permet d'émettre les requêtes d'établissement et de fermeture de session, et de traiter les réponses reçues de la part des autres éléments fonctionnels. Un User Agent SIP classique supporte les fonctionnalités UAS et UAC.

2.2.2 Registrar

Un serveur registrar traite les requêtes d'enregistrement REGISTER émises par les terminaux SIP (User Agent). La possibilité offerte aux terminaux SIP de s'enregistrer sur un serveur permet de les localiser. Le serveur registrar stocke l'adresse IP du terminal enregistré, qui sera retournée en réponse à une requête de recherche provenant d'une autre entité SIP.

L'enregistrement d'un terminal SIP nécessite que ce dernier s'authentifie auprès du serveur registrar. Le mécanisme d'authentification repose sur la présentation d'un challenge[9] au client, qui renvoie une réponse au serveur. Ce mécanisme est celui utilisé dans le cadre de l'authentification HTTP-Digest [11], dont l'avantage est d'éviter la transmission du mot de passe en clair ou encodé (en base64 par exemple), en n'envoyant que le résultat d'un calcul MD5 (chiffrement irréversible) fonction du mot de passe et du challenge présenté par le serveur.

2.2.3 Proxy

La fonction principale du serveur proxy est de localiser l'URI SIP du destinataire. Un serveur proxy dessert généralement un domaine DNS donné, ce qui permet d'adresser des URIs SIP de la forme sip:user@domain plutôt que sip:user@proxy.domain.

L'interrogation du service DNS visant à déterminer l'adresse IP du serveur proxy d'un domaine s'effectue via des requêtes DNS SRV. Pour déterminer l'adresse IP d'un User Agent donné correspondant au domaine administré, un serveur proxy peut interroger le registrar associé, qui maintient une correspondance URI - adresse IP.

Une fois la session établie, le flux média RTP ne traverse pas les Proxies intermédiaires, qui n'interviennent que pour l'ouverture de session. De même, en fonction de la configuration du proxy (en-tête Record Route), la clôture de session peut se faire directement entre SIP User Agents.

A titre d'exemple de proxy et de registrar, on peut citer les logiciels libres OpenSER et SER.

2.2.4 B2BUA

Un SIP B2BUA (Back to Back User Agent) dispose, comme un proxy, d'une fonction de relayage des requêtes d'établissement de session. Il peut de surcroît établir des sessions, les modifier et les clore. Un SIP B2BUA est un relais intelligent, capable par exemple de maintenir des tables d'état des appels.

A titre d'exemple de SIP B2BUA, on peut citer le logiciel libre Asterisk.

[9]Dans un contexte d'authentification client/serveur, un challenge est une chaîne de caractères pseudo-aléatoire envoyée par un serveur pour être concaténée avec les éléments classiques de l'authentification (mot de passe, nom d'utilisateur) par le client. L'ensemble de la chaîne ainsi constituée peut être chiffré irréversiblement par une fonction MD5, produisant ainsi une version protégée des éléments d'authentification qui sera envoyée au serveur pour vérification avant autorisation.

3 Le projet SIP.edu

3.1 Objectif

Initié par l'organisation Internet2, le projet SIP.edu [12] propose une architecture SIP cible destinée à être déployée sur le site d'une quelconque institution académique. Cette architecture vise à rendre accessibles les postes téléphoniques (IP ou non) d'une institution membre depuis un terminal SIP connecté à l'Internet, via une adresse email.

Pour contacter un poste téléphonique d'une institution membre depuis un terminal (ex. softphone[10] SIP), on composera non pas un numéro de téléphone, mais l'adresse email du correspondant désiré. Les éléments du coeur de l'architecture cible (proxy SIP, annuaire, passerelle RTC[11]) assurent le relais vers le poste téléphonique correspondant.

Il est important de noter que l'architecture proposée permet simplement « de se rendre joignable » par SIP. Nous verrons par la suite une possibilité d'extension à une solution de ToIP pour nomades permettant de passer des appels téléphoniques depuis l'Internet, vers des numéros de téléphones classiques.

De nombreuses institutions, essentiellement étrangères, ont mis en oeuvre l'architecture cible SIP.edu. Les outils opensource disponibles, qui implémentent les fonctionnalités requises par l'architecture, sont de grande qualité, et sont souvent à la base des déploiements constatés[12].

3.2 Architecture

L'architecture cible SIP.edu repose sur trois composants fonctionnels :

- un proxy SIP desservant un domaine DNS ;
- un annuaire permettant de traduire les adresses email en numéros de téléphones ;
- une passerelle SIP – RTC permettant de router les appels vers les postes téléphoniques classiques.

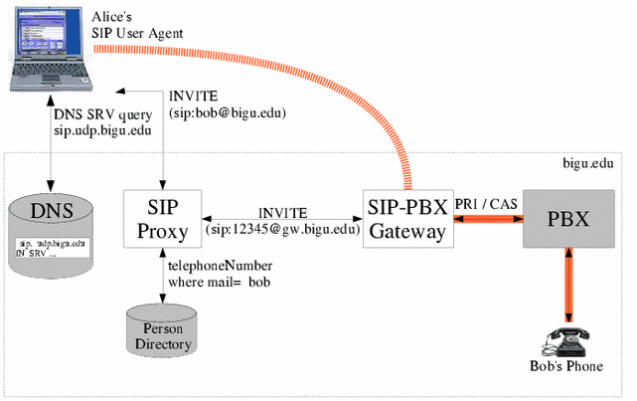
Exemple d'appel :

Dans cet exemple, Alice désire téléphoner à Bob, un chercheur de l'université BigU, à partir d'un softphone SIP. Bob n'est pas joignable en ToIP, seul le poste téléphonique analogique de son bureau à l'université BigU est accessible.

[10]Softphone : une application logicielle de téléphonie.

[11]RTC : Réseau Téléphonique Commuté.

[12]La liste des exemples de déploiement souligne la tendance, tant les références à OpenSER, SER ou Asterisk sont nombreuses : http://mit.edu/sip/sip.edu/deployments.shtml.



Voici les étapes menant à l'établissement d'une communication téléphonique entre Alice et Bob.

- Alice tape l'URI SIP de Bob, construite sur son adresse email, dans l'interface de son softphone SIP : sip:bob@bigu.edu ;
- Une requête DNS de type DNS SRV permet au softphone d'Alice de déterminer l'adresse du proxy SIP en charge du domaine « bigu.edu » ;
- La requête d'établissement de session INVITE à destination de sip:bob@bigu.edu est envoyée au proxy SIP ;
- Le proxy SIP consulte l'annuaire du site pour retrouver le numéro de téléphone de l'utilisateur dont le champ email est 'bob' ;
- La requête INVITE est relayée vers la passerelle SIP / RTC, après remplacement de l'URI initiale par le numéro de téléphone de Bob.

Une fois l'appel établi, le trafic voix circule directement entre le softphone d'Alice et la passerelle SIP / RTC.

3.3 SIP.edu à l'INRIA

Les éléments logiciels et matériels utilisés pour déployer l'architecture cible SIP.edu sont des logiciels libres, à l'exception de la passerelle SIP / RTC (routeur Cisco 3620). Nous les présentons ici :

- proxy SIP : OpenSER

OpenSER est une implémentation libre d'un proxy SIP. Très flexible, bien que difficile à appréhender, il permet notamment d'intégrer des scripts déclenchés sur certains événemements. Cette fonction est utilisée pour interroger l'annuaire LDAP de l'INRIA sur réception d'une requête SIP INVITE d'ouverture de session.

- Annuaire : OpenLDAP

OpenLDAP est une implémentation libre d'un serveur LDAP. Les informations contenues dans l'annuaire sont publiquement accessibles (par l'intermédiaire d'un serveur web). Si le choix de masquer les numéros de téléphones dans les échanges SIP est arrêté, le proxy OpenSER devra être configuré en conséquence.

- Passerelle SIP / RTC : routeur Cisco 3620

N'importe quel équipement muni d'interfaces SIP et téléphonique conviendra. La pile SIP de base venant avec le système d'exploitation IOS a été activée. Dans nombre de déploiements de l'architecture SIP.edu, un serveur Asterisk est utilisé comme passerelle.

On peut noter qu'aucun contrôle d'accès n'est effectué lors de l'établissement d'un appel, rendant ainsi accessible l'ensemble du personnel de l'INRIA via une adresse de la forme sip:prenom.nom@inria.fr. Nous décrivons au paragraphe suivant une extension de l'architecture cible, offrant aux utilisateurs de l'INRIA seuls la possibilité d'appeler n'importe quel numéro de téléphone via la passerelle SIP / RTC.

Pour plus d'informations concernant l'architecture SIP.edu à l'INRIA (fichiers de configuration, schémas, ...), de nombreuses notes de déploiement ont été consignées [13].

3.4 Extension à une solution de ToIP pour nomades

L'architecture cible proposée par le projet SIP.edu permet à toute personne connectée sur l'Internet, et disposant d'un softphone SIP, de contacter les membres de l'INRIA.

La possibilité de composer une URI SIP de la forme sip: 0123456789@inria.fr est un service offert aux utilisateurs nomades de l'INRIA (par exemple en mission à l'étranger), élargissant ainsi l'accès téléphonique vers tout numéro de téléphone joignable depuis l'autocommutateur de l'INRIA.

Un mécanisme de sécurisation doit protéger cet accès. Le protocole RADIUS [14], largement éprouvé dans le cadre d'architectures AAA[13], répond au besoin de contrôle d'accès. C'est le proxy SIP OpenSER qui, en tant que client RADIUS, relaie les demandes d'authentification vers un serveur RADIUS (FreeRADIUS).

4 XMPP : le standard pour la messagerie instantanée

De nombreux services de messagerie instantanée sont disponibles depuis plusieurs années, Yahoo Messenger, MSN, AIM étant les exemples les plus connus. Chacun de ces services repose sur une architecture fermée et est par conséquent incompatible avec les autres services.

XMPP définit une architecture et un protocole permettant d'offrir un service de messagerie instantanée et de son service corollaire, la gestion de présence.

[13]Authentication, Authorization, Accounting, que l'on peut traduire par Authentification, autorisation, comptabilisation

.

4.1 Le protocole XMPP

Tous les services d'IM connus ne sont pas nécessairement cloisonnés. Google a ainsi construit le sien (GoogleTalk – Gmail) autour du standard XMPP, fournissant de ce fait une interface vers d'autres services basés sur ce même standard. On pourra ainsi trouver dans la liste de contacts d'un client GoogleTalk des contacts dont l'identifiant est suffixé par des domaines DNS divers (jabber.org, jabber.fr, inria.fr, ..), ceux-ci étant aussi gérés par un serveur XMPP différent.

L'architecture XMPP est semblable à celle de la messagerie électronique SMTP. Un client communique avec le serveur auquel il est rattaché, lui même lié à un noeud représentant généralement un domaine DNS.

Contrairement à SIP, qui ne distingue pas de client et de serveur parmi les éléments fonctionnels de son architecture (pour rappel, un poste SIP est tout à la fois un UAS et un UAC), le standard XMPP définit un protocole de communication de client à serveur (c2s), et un protocole d'échanges de serveur à serveur (s2s), qui traite des communications inter-domaines DNS. Ceci implique de distinguer les éléments « client » et « serveur » dans l'architecture.

De plus, le protocole XMPP est transporté sur TCP exclusivement, véhiculant donc un flux d'information continu, par ailleurs facilement sécurisable par TLS.

4.1.1 Identifiant et localisation

Identifiant

Les utilisateurs sont identifiés par un JID, ou Jabber Identifier[14]. Le format général d'un JID est le suivant :

```
node@domain/resource
```

« node » désigne un utilisateur, « domain » le domaine DNS de rattachement. Ce domaine peut aussi être remplacé par une adresse DNS de machine.

« resource » est une chaîne de caractères optionnelle permettant de distinguer différentes sessions pour un même utilisateur. Par exemple l'architecture XMPP de Google distingue pour chaque utilisateur les sessions établies par le biais de l'interface web Gmail, de celles établies via le client GoogleTalk. Cette distinction est nécessaire car la session GoogleTalk est enrichie de la fonctionnalité VoIP, inutilisable via l'interface Gmail.

Localisation DNS, enregistrement DNS SRV

A un serveur XMPP correspond généralement un domaine DNS. Pour désigner le serveur XMPP responsable d'un domaine DNS donné, un enregistrement de type DNS SRV doit exister dans le serveur DNS.

Exemple :

```
_xmpp-server._tcp.inria.fr. 172800 IN   SRV     5
0 5269 softswitch.inria.fr.
```

L'entrée précédente dans le serveur DNS du domaine « inria.fr » indique que le serveur XMPP desservant le domaine « inria.fr » en TCP est accessible à l'adresse « softswitch.inria.fr » sur le port 5269 (le port standard pour les communications de serveur à serveur).

4.1.2 Sessions XMPP, format des messages

XML est le format de base des échanges entre les composants de l'architecture XMPP.

Ouverture et fermeture de session

L'ouverture d'une session par un client s'effectue par la connexion au serveur (port 5222/TCP, ou 5223 si SSL), suivi de l'envoi d'une balise XML d'ouverture de flux <stream>. Un flux unidirectionnel est alors ouvert entre le client et le serveur, qui doit à son tour ouvrir un flux suivant la même procédure pour qu'une connexion bidirectionnelle soit établie.

Une procédure de chiffrement par TLS (STARTTLS) peut à ce stade être lancée en fonction des configurations, avant l'authentification du client par SASL [15]. SASL est un ensemble de mécanismes d'authentification[15], et offre une procédure suffisamment flexible pour pouvoir intégrer toute base d'authentification au serveur XMPP. Pour sécuriser les échanges, on imposera dans le cas du mécanisme SASL-PLAIN de chiffrer le flux TCP au préalable entre le client et le serveur.

Pour clore une session XMPP, le client et le serveur s'échangent les balises de fermeture des flux unidirectionnels établis : </stream>.

Envoi/réception de messages

Les messages sont transmis sous la forme de « strophes »[16] XML. Une strophe est une unité d'information structurée par une représentation XML et un espace de nommage. De plus, une strophe est un élément XML fils direct de l'élément racine <stream/>. Les clients et les serveurs XMPP s'échangent trois types de strophes :

- <message> : pour les échanges de messages instantanés (« chat ») ;
- <presence> : pour la diffusion de l'état ;
- <iq>[17] : un mécanisme de requête/réponse générique.

[14]La terminologie Jabber persiste ici, il n'est pas envisagé d'identifier les utilisateurs par un quelconque XID!

[15]La liste des mécanismes d'authentification est disponible sur ce site : ftp://ftp.isi.edu/in-notes/iana/assignments/sasl- mechanisms/. Retenons que les mécanismes DIGEST-MD5 (RFC 2831) et PLAIN (RFC 4616) sont les plus répandus.

[16]Le terme utilisé en Anglais est « stanza », qui se traduit par « strophe » en Français.

[17]IQ : Info/Query.

Ces messages forment le coeur du protocole XMPP.

4.2 Application : « Chat » entre utilisateurs

Une fois la connexion avec le serveur établie, le client exécute deux actions :

- le téléchargement de sa liste de contacts[18], stockée sur le serveur ;
- la publication de son état, qui sera relayé par le serveur XMPP aux éléments de la liste de contacts.

Lorsque deux utilisateurs s'échangent des messages instantanés, ils ne font qu'envoyer des strophes XML <message> à destination de l'utilisateur distant. Tout comme pour l'envoi d'un courrier électronique, le message considéré n'est pas transmis directement à l'utilisateur distant. Il transite par les serveurs XMPP auxquels sont rattachés les utilisateurs en correspondance.

Exemple:

```
<message    to='romeo@example.net'
from='juliet@example.com'
type='chat'    xml:lang='en'>  <body>Wherefore
art thou, Romeo?</body> </message>
```

Dans cet exemple, juliet@example.com demande dans la langue de Shakespeare à romeo@example.net où il se trouve. Ce message aura transité dans l'ordre par les serveurs XMPP desservant les zones example.com et example.net, avant de parvenir au destinataire.

5 XMPP à l'INRIA

Le service d'IM de l'INRIA a été mis en oeuvre à titre expérimental depuis le mois de février 2007. Il est aujourd'hui utilisé quotidiennement par quelques dizaines d'utilisateurs.

L'idée est de permettre aux utilisateurs de l'INRIA de communiquer directement en « chat » ou dans des salles de conversation « à la IRC ». Les utilisateurs choisissent leur client XMPP, qu'il trouvent sans difficulté sur l'Internet. Beaucoup ont une préférence pour les clients multi-protocoles comme Gajim ou Miranda, qui supportent XMPP, MSN, Yahoo Messenger et AIM.

Le serveur installé à l'INRIA est le logiciel jabberd2. Parmi les logiciels libres serveurs XMPP, citons Openfire (disponible aussi en version commerciale), jabberd1.4 et ejabberd.

5.1 Intégration dans le système d'information

5.1.1 Authentification

L'intégration du service d'IM avec la base d'authentification de l'INRIA est un point très important. Les utilisateurs du service s'authentifient par un login basé sur l'adresse email, le mot de passe fourni étant vérifié sur une base d'authentification centrale hégergeant environ 5000 comptes.

Jabberd2 dispose de la fonctionnalité client LDAP nécessaire à l'interrogation du serveur d'authentification.

5.1.2 Support multi-domaine

L'ensemble des utilisateurs de l'INRIA peut utiliser le service en s'identifiant par le format générique de l'adresse email : prénom.nom@inria.fr.

De plus, les membres des différents Centres de Recherche (CR) de l'INRIA peuvent être identifiés par une adresse au format prénom.nom@domaine_du_cr, sous réserve que leur CR dispose d'un domaine DNS propre. Dans les deux cas, le serveur d'authentification central LDAP est sollicité pour identifier les utilisateurs.

5.2 Composants externes

5.2.1 Passerelles multi-protocoles

Nous n'avons pas installé de passerelle sur notre serveur XMPP qui permettrait par exemple à un utilisateur d'accéder depuis un client XMPP à ses contacts MSN (sous réserve de disposer d'un compte MSN).

En effet, si des demandes dans ce sens ont éte exprimées, les utilisateurs ont rapidement répondu à leur propre besoin en installant des clients IM multi-protocoles.

5.2.2 Serveur Asterisk

Un serveur Asterisk est disponible pour les utilisateurs du service XMPP de l'INRIA. L'idée d'intégrer un tel serveur fait suite à la publication des travaux de la XSF visant à standardiser le transport multimédia voix et vidéo sur XMPP, via Jingle.

Le module XMPP du serveur Asterisk supportant le standard Jingle[19], nous avons ajouté une fonctionnalité ToIP au service d'IM existant. L'idée est d'offrir la possibilité de composer un numéro de téléphone classique E.164 [16] depuis un client XMPP.

Cependant, les implémentations de Jingle dans les clients logiciels sont encore rares, et nous n'avons jusqu'à maintenant pu que valider par des tests la connexion ToIP

[18]Aussi appelée « buddy list » ou « roster » dans le jargon des utilisateurs de messagerie instantanée.

[19]Le standard n'étant pas achevé, la compatibilité Jingle du module XMPP d'Asterisk a été comparée avec les implémentations disponibles lors des tests effectués sur Windows : celles des clients GoogleTalk et Jabbin.

entre un client XMPP et le serveur Asterisk, à l'aide des clients GoogleTalk et Jabbin.

6 SIP / XMMP, domaines d'application

Pour conclure cet article, nous revenons sur les différences majeures entre les architectures et protocoles basées sur SIP et XMPP, leurs conséquences sur les services fournis aux utilisateurs, ainsi que les domaines d'application des deux normes.

6.1 Service de ToIP pour nomades et d'IM : SIP ou XMPP?

6.1.1 Authentification, protocole de transport

Les services offerts aux utilisateurs de l'INRIA sont généralement accessibles après authentification. La ToIP et l'IM n'échappent pas à la règle, et comme nous l'avons vu précédemment, le mécanisme d'authentification de SIP ne permet pas de réutiliser simplement une base externe dans laquelle les mots de passe seraient chiffrés.

La raison principale de la sécurisation de l'authentificaiton par un mécanisme challenge/response tient au choix d'UDP comme protocole de transport historique. En effet, contrairement au couple formé par TLS / TCP, aucun protocole de sécurisation n'est associé à UDP, interdisant de fait le transport d'informations sensibles sans protection. On rappelle aussi que la gestion de la fragmentation IP de datagrammes UDP est une source de difficulté de connexion.

Inversement, XMPP s'appuie sur un transport TCP. Comme on l'a vu précédemment, le processus d'authentification repose sur un mécanisme flexible (SASL), et peut être protégé par sécurisation du canal de transmission à l'aide de TLS. La flexibilité offerte par SASL permet d'authentifier les utilisateurs à partir d'une base d'un quelconque type, et le logiciel Jabberd2 offre par ailleurs la possibilité d'adapter l'authentification à l'aide de scripts Perl.

6.1.2 ToIP pour nomades

Le choix de SIP pour assurer un service de ToIP pour nomades tel que décrit précédemment est naturel. Dans le coeur du réseau, les logiciels utilisés (OpenSER, Asterisk) sont fiables, flexibles et répondent aux besoins d'authentification et de traçabilité (identification de l'appelant et de l'appelé, durée des communications, ...). De plus, l'accessibilité du service pour les clients SIP situés derrière des routeurs NAT peut être assurée soit par des compléments logiciels (Mediaproxy, RTPProxy pour OpenSER), soit par un service complémentaire (ex. : tunnel VPN IPsec).

Les logiciels clients SIP utilisés (Xlite, SJPhone) sont de très bonne qualité, et disposent souvent d'une interface vidéo.

Du côté des clients XMPP, comme on l'a vu, le support voix/vidéo nécessite une implémentation logicielle du standard encore inachevé : Jingle. Si un certain nombre de projets de logiciels clients opensource orientent leur travaux vers Jingle, peu sont encore disponibles dans des versions stables.

De nouvelles implémentations prometteuses apparaissent, souvent avec un support multi-protocole (SIP, XMPP, MSN, ...). Citons parmi elles SIP Communicator [article 132 de JRES 2007], basée sur l'API JAIN-SIP[20] et développée à l'université Louis Pasteur de Strasbourg.

6.1.3 IM

Le choix de XMPP comme protocole de base du service de messagerie instantanée pour l'INRIA est plus judicieux. L'exploitation de la base d'authentification nationale est immédiate, de même que la mise en place du support multi-domaine.

SIP, via la suite protocolaire SIMPLE offre une architecture adaptée à l'IM mais ne prévoit pas, par exemple, de service natif de stockage des listes de contacts des utilisateurs, déléguée à un service externe comme XCAP [17].

7 Conclusion

SIP est à la base de nombreux services de ToIP proposés par les opérateurs. Ainsi, l'opérateur Free met à disposition de ses clients un service SIP permettant l'enregistrement sur un registrar, afin d'être joignable sur un softphone tout en pouvant y passer des appels. Dans le coeur du réseau, SIP est présent en tant que protocole de base de l'architecture IMS (IP Multimedia Subsystem).

SIP est aussi présent dans le domaine de la téléphonie d'entreprise. La plupart des offres d'IP-PBX actuels s'accompagnent d'une interface SIP en complément de l'interface propriétaire du constructeur.

XMPP traitait initialement du domaine de l'IM, et ne s'inscrit donc pas dans le paysage des protocoles de télécommunication, où l'on trouve exclusivement SIP et encore H.323 ou MGCP pour les offres de Centrex. Bien que XMPP soit moins répandu que SIP, il est intéressant de voir l'intérêt que lui portent quelques « gros » acteurs de l'Internet tels que Google ou IBM, qui appuient certains de leurs services autour d'XMPP. De plus, à la suite des travaux présentés dans cet article, il nous a semblé clair que XMPP était plus adapté à fournir un service de messagerie instantanée pour l'INRIA, et plus généralement pour une entreprise.

L'un des axes de travail de standardisation du protocole XMPP traite de la ToIP, par le développement du protocole Jingle. Concernant les implémentations

[20]JAIN : Java API for Integrated Networks. JAIN-SIP est une API développée dans le cadre des travaux du groupe de standardisation américain NIST (National Institute of Standards and Technology).

logicielles, Google et son client GoogleTalk supportent (en partie) le futur standard Jingle, et de nombreux logiciels opensource travaillent à l'implémenter (Jabbin, Psi, Coccinella, ...). On rappelle aussi que l'IP-PBX multi-protocole Asterisk implémente un module XMPP ainsi qu'un module Jingle dans sa dernière version.

SIP et XMPP s'étendent vers des domaines qui tendent à se superposer, ce qui les fera peut-être entrer en concurrence. En effet, un fournisseur de service de ToIP + IM a désormais pratiquement le choix entre deux standards, et il aura bientôt le choix entre de nombreuses implémentations logicielles.

Concernant l'INRIA, les perspectives d'évolution portent surtout sur le service de messagerie instantanée, notamment en développant un client basé sur le standard XMPP, et en poursuivant le développement du module XMPP du logiciel Asterisk, en vue de l'intégrer au service de messagerie instantanée.

Bibliographie

[1] Packet-based multimedia communications systems, H.323, http://www.itu.int/rec/T-REC-H.323/en.

[2] SIP, RFC 3261, http://www.ietf.org/rfc/rfc3261.txt.

[3] XMPP : core, RFC 3920, http://www.ietf.org/rfc/rfc3920.txt.

[4] XMPP : Instant Messaging and Presence, RFC 3921, http://www.ietf.org/rfc/rfc3921txt.

[5] Mapping XMPP to Common Presence and Instant Messaging (CPIM), RFC 3922, http://www.ietf.org/rfc/rfc3922.txt.

[6] End-to-End Signing and Object Encryption for XMPP, RFC 3923, http://www.ietf.org/rfc/rfc3923.txt.

[7] Jingle, XEP 166, http://www.xmpp.org/extensions/xep-0166.html.

[8] SIP Extension for Instant Messaging, RFC 3428, http://www.ietf.org/rfc/rfc3428.txt.

[9] Real Time Protocol, RFC 3550, http://www.ietf.org/rfc/rfc3550.txt.

[10] URI : Generic Syntax, RFC 2396, http://www.ietf.org/rfc/rfc2396.txt.

[11] HTTP Authentication: Basic and Digest Access Authentication, RFC 2617, http://www.ietf.org/rfc/rfc2617.txt.

[12] Philippe Sultan, Authentification SIP et présentation du projet SIP.edu. *MISC n°31*, mai-juin 2007.

[13] SIP.edu Deployment Notes , http://mit.edu/sip/sip.edu/deployments.shtml.

[14] Remote Authentication Dial In User Service (RADIUS), RFC 2865, http://www.ietf.org/rfc/rfc2865.txt.

[15] Simple Authentication and Security Layer (SASL), RFC 2222, http://www.ietf.org/rfc/rfc2222.txt.

[16] The international public telecommunication numbering plan, E.164, http://www.itu.int/rec/T-REC-E.164/en.

[17] The Extensible Markup Language (XML) Configuration Access Protocol (XCAP), RFC 4825, http://www.ietf.org/rfc/rfc4825.txt.