EVALUATION DES SOLUTIONS DE SECURITE DU PROTOCOLE SIP

Ce chapitre présente l'architecture Session Initiation Protocol (SIP) ainsi qu'une analyse des solutions de sécurité associées. Nous avons choisi ce protocole car il s'impose dans de nombreuses infrastructures de ToIP pour assurer le service de téléphonie. Au travers de la problématique de l'authentification qui est au cœur de nos travaux, il est mis en évidence les risques pour les usagers de ne pas bénéficier de mécanismes d'authentification adéquats.

3. Evaluation des solutions de sécurité du protocole SIP

3.1. La place du protocole SIP dans la téléphonie sur IP

Comme cela a été précisé dans le chapitre précédent, pour permettre l'établissement d'un appel en téléphonie sur IP le réseau transporte deux types d'informations : la signalisation d'appel et le média (la voix pour un appel téléphonique). Ces informations sont générées par des logiciels hébergés sur des terminaux ou sur des serveurs.

La ToIP devient ainsi un service parmi tant d'autres dans les SI. La voix et la signalisation sont véhiculées par des paquets IP. Le travail de cette thèse a pris le parti de se focaliser sur le protocole de téléphonie le plus populaire [BER08], en l'occurrence SIP (Session Initiation Session) [RFC3261] pour étudier les opportunités de renforcer la sécurité de la ToIP.

SIP spécifie les propriétés, les caractéristiques, et le mode de fonctionnement de la signalisation. Il décrit également l'interaction avec le protocole de la couche Transport d'Internet. Il décrit également comment le flux de voix est émis et reçu. Le but de ce chapitre est de décrire les grands principes de SIP et de présenter les solutions de sécurité envisagées dans [RFC3261]. L'idée de ce chapitre n'est pas de décrire la totalité du protocole SIP mais de mettre en avant les principes qui seront nécessaires pour les contributions de la thèse.

3.2. Architecture et protocoles dans un environnement SIP

3.2.1. *L'architecture globale*

SIP est issu de l'IETF[16] (Internet Engineering Task Force) au travers d'un RFC[17]. Les premiers travaux datent de 1995 et ont abouti à une première version de SIP avec la parution de [RFC2543] en 1999. Une deuxième version de SIP a été éditée en 2002 pour corriger certains défauts de jeunesse. Cette dernière version est toujours en vigueur au travers de [RFC3261].

SIP permet comme son nom l'indique d'initier, mais également de modifier et de terminer des sessions[18] voix mais aussi multimédias. La session voix est l'équivalent de notre « appel téléphonique ». SIP se situe au niveau applicatif. Pour fonctionner, SIP a donc besoin d'autres standards ou protocoles. A ce titre, SIP est souvent décrit comme un protocole « chapeau » puisqu'il s'appuie sur d'autres briques protocolaires comme UDP [RFC768] ou TCP [RFC793] pour la couche Transport. La figure 5 présente la pile protocolaire SIP pour la signalisation et le média.

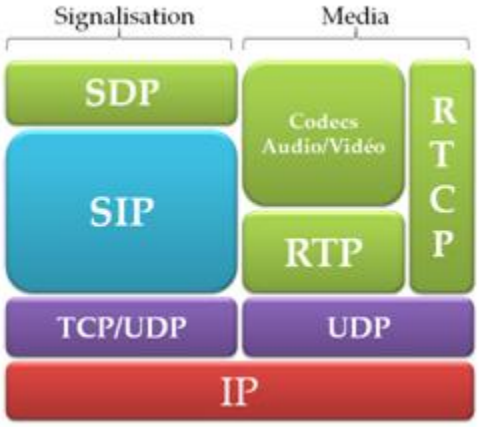

Figure 5. La pile protocolaire de SIP

Le fonctionnement de SIP s'appuie sur une architecture générique appelé « trapézoïde SIP » comme l'illustre la figure 6. Il existe deux grandes catégories d'acteurs dans cet environnement :

- les clients appelés « User Agent » (UA) qui initient et reçoivent les appels ;
- les serveurs qui relaient ou traitent les messages SIP émis par les UA ou les autres serveurs.

---

[16] IETF : L'Internet Engineering Task Force est un groupe informel et international qui participe à l'élaboration de standards pour Internet.

[17] RFC : Les Requests For Comments (RFC) sont une série numérotée de documents issus de l'IETF décrivant les aspects techniques d'Internet.

[18] Une session SIP recouvre plusieurs types d'échange. Une session peut être, par exemple, une conversation téléphonique, une visioconférence, une prise de contrôle d'un PC à distance, un échange de données ou encore un échange de messages instantanés.

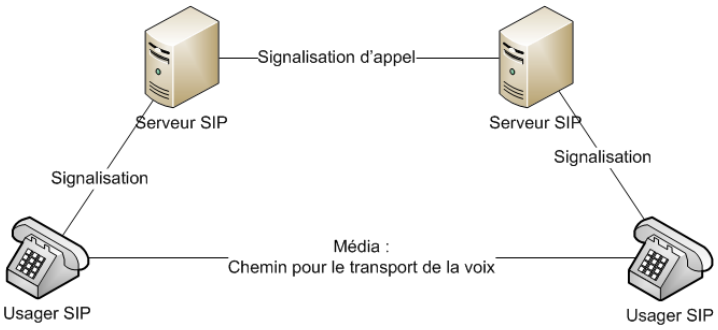

Figure 6. Architecture élémentaire SIP

L'établissement d'une communication se fait au travers d'échanges de messages entre les différents éléments du réseau. Ces échanges font partie de la signalisation. Une fois la session établie, les échanges de données (voix, images, vidéo) se font directement entre les deux extrémités. La voix est quant à elle transportée par le protocole RTP (Real-time Transport Protocol) [RFC3550].

3.2.2. *Les entités SIP*

Comme cela a été précisé dans le paragraphe précédent, il y a deux familles d'entités SIP, les usagers et les serveurs.

L'usager – ou le terminal SIP – est le User Agent (UA). Il émet et reçoit les appels. Chaque UA est à associer à un identifiant appelé URI (Uniform Resource Identifier) SIP. Les URI SIP ont une forme similaire à celle des adresses de messagerie, contenant normalement le nom d'utilisateur et le domaine d'appartenance, exemple : sip:100@enst.fr.

Concernant les serveurs, il en existe de 4 types :

- **Registrar Server** : il s'occupe exclusivement de l'enregistrement des terminaux SIP. Il reçoit les messages de type REGISTER. Il doit identifier les utilisateurs, voire les authentifier. Il doit être relié à un Proxy Server ou à un Redirect Server qui sera en charge de l'appel ;

- **Proxy Server** : il sert de relais aux messages SIP. Il joue le rôle de serveur d'un côté et de client de l'autre. Il interprète, transforme ou traduit un message avant de transférer.

- **Redirect Server** : il gère la signalisation d'appel comme le Poxy Server, mais il ne relaie pas les messages. Il redirige directement l'UA vers la destination requise en lui indiquant l'adresse IP et le port à contacter ;

- **Location Server** : il est utilisé par les deux types de serveur précédents pour obtenir des informations sur les différentes localisations possibles d'un utilisateur.

3.2.3. *Les messages SIP*

SIP a été inspiré par le modèle client/serveur particulièrement répandu dans le monde de l'Internet. Les messages sont codés en utilisant la syntaxe des messages HTTP/1.1 [RFC2616] et le codage UTF-8 [RFC2279]. Les messages échangés sont donc soit des requêtes, soit des réponses. La nature textuelle des échanges rend facilement interprétables les messages. L'association requête/réponse est appelé transaction. La figure 7 est un exemple de message SIP initiant une session avec la description des capacités pour l'échange de données voix.

```
INVITE sip:francois@192.168.1.69:7000;rinstance=1b164c0bc67dd088 SIP/2.0
Via: SIP/2.0/UDP 192.168.1.8:5060;branch=z9hG4bK33598e3a;rport
From: "thomas" <sip:100@192.168.1.8>;tag=as59c91fa4
To:<sip:francois@192.168.1.69:7000;rinstance=1b164c0bc67dd088>;tag=a51c5454
Contact: <sip:100@192.168.1.8>
Call-ID: 20c237282b8cf0c60fce5ff868413754@192.168.1.8
CSeq: 103 INVITE
User-Agent: Asterisk PBX
Max-Forwards: 70
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, SUBSCRIBE, NOTIFY
Content-Type: application/sdp
Content-Length: 286

v=0
o=root 5182 5183 IN IP4 192.168.1.96
s=session
c=IN IP4 192.168.1.96
t=0 0
m=audio 6502 RTP/AVP 3 0 8 97 101
a=rtpmap:3 GSM/8000
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-16
a=silenceSupp:off - - - -
```

Figure 7. Exemple de message SIP

Par la suite, il sera détaillé :

- les requêtes SIP ;
- les réponses SIP ;
- la structure des messages SIP.

3.2.3.1. Les requêtes

La version actuelle de SIP prévoit 6 requêtes distinctes, permettant l'établissement d'un appel, la négociation des capacités (types de média, paramètres de la session, éléments de sécurité) ou la fermeture d'une session. Ces requêtes sont détaillées dans le tableau 3.

Tableau 3. Liste des requêtes SIP

Requête	Définition
INVITE	Requête d'établissement d'une session, invitant un usager (humain ou non) à participer à une communication téléphonique ou multimédia ; l'émetteur de cette requête y indique les types de média qu'il souhaite et peut recevoir, en général au travers d'une description de session SDP (Session Description Protocol) [RFC4566].
ACK	Requête d'acquittement, émise pour confirmer que le client émetteur d'un INVITE précédent a reçu une réponse finale ; cette requête peut véhiculer une description de session qui clôt la négociation.
BYE	Requête de clôture d'un appel.
CANCEL	Requête d'annulation, signifiant au serveur de détruire le contexte d'un appel en cours d'établissement (cette requête n'a pas d'effet sur un appel en cours).
OPTIONS	Cette requête permet à un client d'obtenir de l'information sur les capacités d'un usager, sans pour autant provoquer l'établissement d'une session.
REGISTER	Requête à destination d'un serveur SIP et permettant de lui faire parvenir de l'information de localisation (machine sur laquelle se trouve l'utilisateur).

D'autres requêtes existent mais sont issues d'autres RFC comme : MESSAGE pour l'envoi de message instantané, PRACK pour la sécurisation des réponses provisoires, PUBLISH pour l'envoi d'une information relative à un état vers un serveur, INFO pour l'envoi d'information ne modifiant pas la session et UPDATE pour la mise à jour des paramètres média avant la réponse finale au premier INVITE.

3.2.3.2. Les réponses

Après réception et traitement d'une requête, un agent ou un serveur SIP génèrent un message de réponse (succès ou échec du traitement). Ces réponses sont codées par une séquence de trois chiffres, où le premier est un code de classe. Le tableau 4 donne quelques réponses possibles.

Tableau 4. Les principales familles des réponses

Code	Définition de la famille de réponse	Principales réponses
1XX	Réponse intermédiaire d'information (traitement en cours)	- 100 Trying - 180 Ringing
2XX	Succès	- 200 OK
3XX	Redirection	- 301 Moved permanently - 302 Moved temporarily
4XX	Erreur client	- 400 Bad Request - 401 Unauthorized
5XX	Erreur serveur	- 500 Server Internal Error - 501 Not Implemented
6XX	Echec global du traitement	- 600 Buzy Everywhere - 603 Decline

3.2.3.3. La structure des messages SIP

Les messages SIP se décomposent de trois parties :

- la première ligne ;
- l'en-tête ;
- le corps du message.

La première ligne sert à identifier le type de message SIP ainsi que l'adresse du destinataire. L'en-tête contient les informations permettant l'acheminement du message comme : la référence de l'émetteur, le destinataire, référence de la transaction et de la session, les éléments de sécurité. Ainsi l'en-tête permet l'établissement d'une session en termes de localisation, de nommage et d'adressage, mais c'est le corps du message qui décrit le flux multimédia mis en jeu par la session. Le corps du message contient généralement les éléments nécessaires à l'établissement du canal média. La liste des paramètres du corps du message est au format SDP (Session Description Protocol) [RFC4566]. Un exemple de message SDP est donné dans la figure 8. Les champs des en-têtes les plus usuels sont le From, le To, le Call-ID, le Cseq, le Contact. Les principaux champs avec leur signification sont donnés dans le tableau 5. Tous ne sont pas obligatoires.

Tableau 5. Les principaux champs d'en-tête des messages SIP

Champ d'en-tête	Description
Authorization :	Information d'authentification pour l'usage d'une ressource par un UA
Call-ID (*) :	Identifiant unique pour un échange d'établissement particulier
Contact :	Généralement, URL de l'usager
Content-Length :	Longueur du message en octets
Content-Type :	Type du corps du message (par exemple une description SDP)
CSeq (*) :	Identifie une requête à l'intérieur d'une session
Encryption :	Précise que le contenu est chiffré
From (*) :	Initiateur de la requête
Max-Forwards (*) :	Limite au nombre de serveurs et de proxies qui peuvent router le message
Proxy-Authenticate :	Information pour l'authentification d'un usager auprès d'un proxy
Proxy-Authorization :	Information pour l'authentification d'un usager auprès d'un proxy
Proxy-Require :	Précise un mécanisme qui doit être fourni par le proxy
Timestamp :	Date d'émission du message
To (*) :	Précise le destinataire de la requête
Unsupported :	Liste les mécanismes non supportés par le serveur
User-Agent :	Information sur l'UA qui a généré le message
Via (*) :	Dénote le chemin emprunté par la requête jusqu'à l'instant présent
WWW-Authenticate :	Inclus dans les réponses 401, dans le but d'authentifier l'émetteur de la requête

(*) : Champs d'en-tête obligatoires dans les 6 requêtes SIP du tableau 2 quelque soit le contexte.

Un des principaux corps de message SIP est SDP qui permet la négociation du canal média. Pour cela, SDP véhicule les informations suivantes :

- le nom de la session de communication ;
- le but (ou l'objet) ;
- les dates et heures d'activité de cette session ;
- les divers flux audio, vidéo ou autres qui la composent ;
- tout paramètre caractérisant ces flux (adresses, ports, formats,…) ;
- de façon optionnelle, de l'information additionnelle, précisant par exemple la bande passante requise ou une information de contact de la personne responsable.

3.2.4. *Fonctionnement d'un appel téléphonique SIP*

Cette partie s'articule en deux parties :

- une première partie pour présenter les principaux messages pour l'établissement d'un appel entre deux usagers, sans tenir compte des relais pour faciliter la compréhension du processus d'établissement d'une session (cf. figure 8) ;
- une deuxième partie pour présenter les différentes transactions dans une architecture type SIP (cf. figure 9).

D'une manière générale, l'établissement d'une session commence par une requête INVITE. Par ce message, Bob signifie à Alice son souhait d'établir une session. Le téléphone d'Alice signifie par une réponse 180 RINGING que l'INVITE est bien arrivé et qu'un signal est émis pour avertir de la demande d'établissement de session. La réponse 200 OK signifie qu'Alice accepte la session. Les messages 180 et 200 contiennent également les éléments pour l'établissement de la session voix comme par exemple les codecs supportés. Les derniers messages BYE et 200 OK permettent la libération de l'appel.

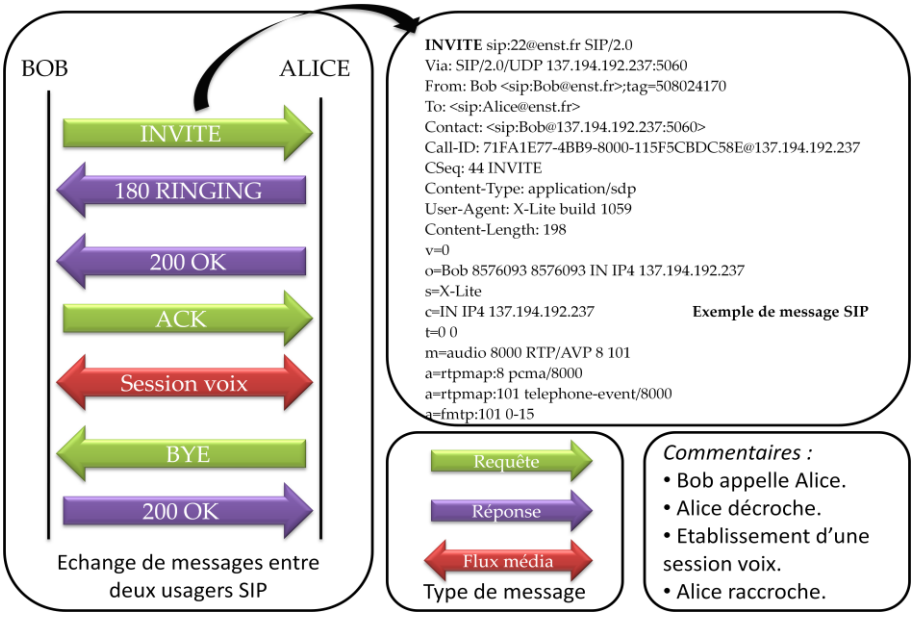

Figure 8. Echange de messages SIP pour l'établissement d'une session

Evidemment cette description ne tient pas compte des serveurs SIP qui traitent et relaient l'appel. La figure 18 issue de [RFC3261] montre le cheminement de l'appel au travers de deux domaines SIP. La réponse 100 correspond à une réponse provisoire envoyée par les proxys pour signifier que la demande est en cours de traitement.

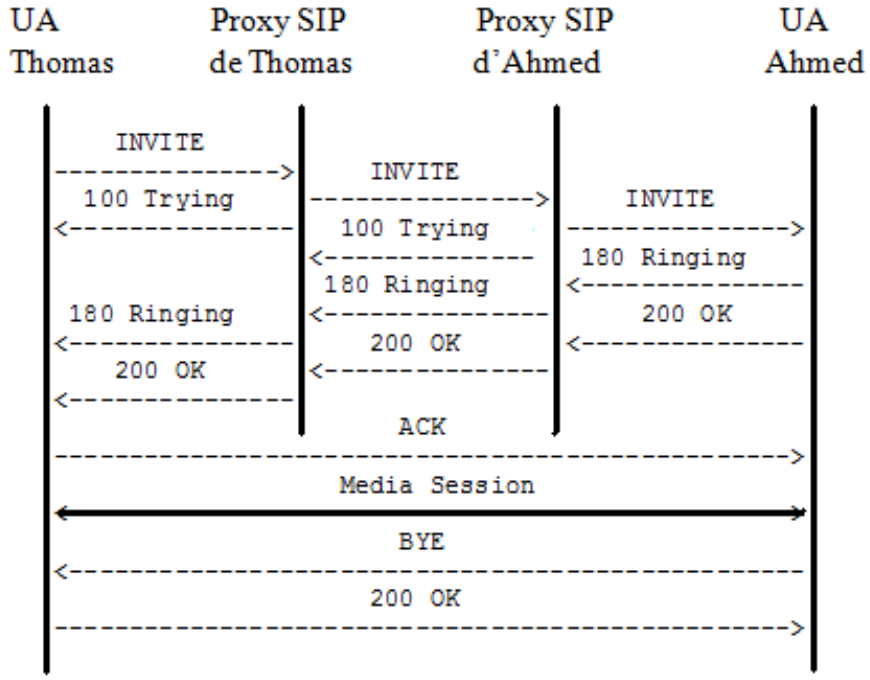

Figure 9. Etablissement d'un appel dans une infrastructure SIP

L'utilisation des services SIP nécessite au préalable un enregistrement du client auprès du serveur REGISTRAR. En dehors des mécanismes de sécurité pouvant être implémentés, cette procédure se résume par l'envoi d'un message REGISTER par l'UA qui reçoit une réponse 200 OK du REGISTRAR. Cette description, comme celle des appels ci-dessus, ne prennent pas en compte le mécanisme d'authentification HTTP Digest relativement usuel dans l'implémentation de SIP. Ce mécanisme modifie la nature des transactions. Ce point sera traité ultérieurement.

3.3. Typologie des attaques

3.3.1. *Typologie des attaques*

Comme cela a été présenté dans le paragraphe 2.4., il existe plusieurs types de risques et d'attaques en téléphonie sur IP. Une typologie orientée SIP est présentée dans le tableau 6.

Tableau 6. Typologie des attaques dans un environnement SIP

Type d'attaque	Application à l'architecture SIP
Interception et modification	- Interception des paquets SIP pour connaître les destinataires des appels ; - Interception des paquets RTP pour écouter la communication ; - Interception et injection de paquet RTP pour dégrader la qualité d'une conversation ; - Interception et manipulation du corps SDP pour faciliter une écoute ou la manipulation des messages SIP ; - Injection de message détournant une session.
Fraude et abus de service	- Usurpation d'identité en s'enregistrant avec le profil SIP d'un usager ; - Falsifier son identité d'appelant ; - Usurpation d'un serveur SIP ; - Manipulation des données de facturation.
Interruption de service ou déni de service	- Interception et modification des messages SIP entrainant une annulation annulant l'initialisation d'une session ou interrompant une session ; - Interception et injection de messages SIP entrainant le désenregistrement d'un usager (client injoignable) ; - Inondation de messages SIP vers un serveur ou client.

Nos contributions concernant l'authentification, nous illustrons par la suite des attaques illustrant l'importance de cette propriété de sécurité.

3.3.2. *Les attaques par déni de service*

3.3.2.1. L'attaque par la méthode du BYE

L'attaque par la méthode du BYE (cf. figure 10) est dirigée contre les usagers. L'attaquant génère un BYE et interrompt une conversation [CHE09]. Pour réaliser cette attaque, le pirate écoute le trafic prend les informations nécessaires (comme par exemple le Call-Id, le From ou encore le To) pour générer un BYE frauduleux correspondant à la session qui est injecté sur le réseau. Le BYE n'étant pas authentifié, celui qui reçoit l'information l'exécute.

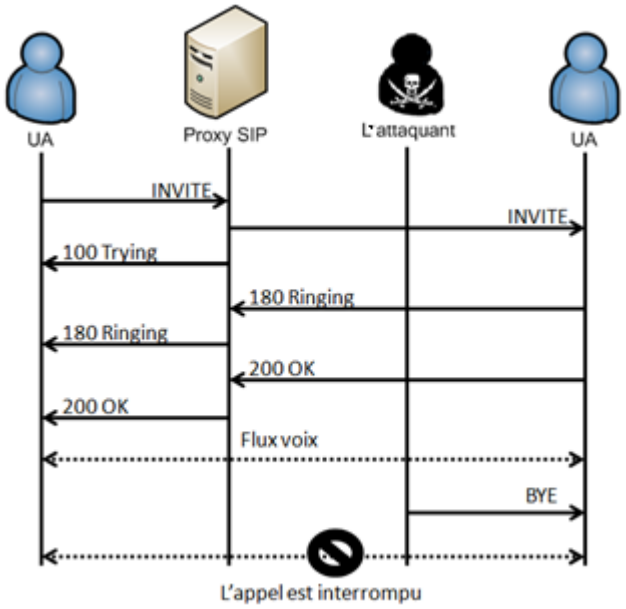

Figure 10. Attaque par le BYE

3.3.2.2. L'attaque par la méthode du CANCEL

L'attaque par la méthode du CANCEL (cf. figure 11) est dirigée contre un usager. Une partie tierce génère un CANCEL pendant l'établissement d'une session [CHE09]. Il opère de la même manière que pour l'attaque du BYE mais cette fois avant l'établissement de la session. Le serveur ou les usagers pensent que l'appelant a annulé. Cette attaque est possible car le CANCEL n'est pas authentifié.

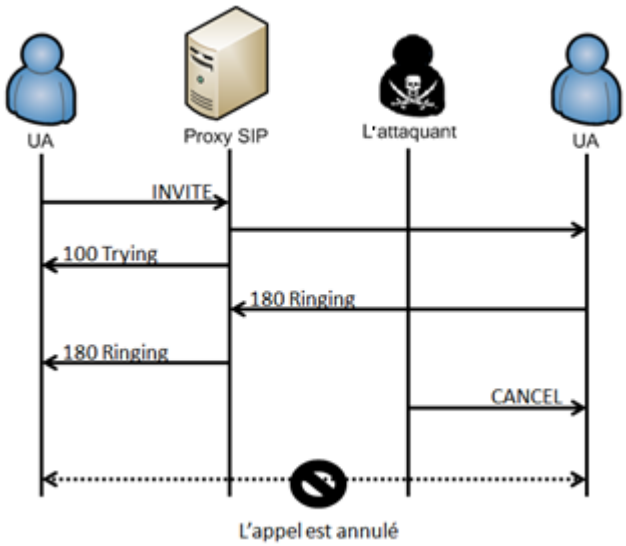

Figure 11. Attaque par le CANCEL

3.3.2.3. L'attaque par la méthode du REGISTER

L'attaque par la méthode du REGISTER (cf. figure 12) est dirigée contre l'usager. En écoutant le réseau un attaquant récupère l'identifiant d'un usager. Il contrefait un message REGISTER avec le champ « expires »[19] égal à zéro ce que le REGISTRAR traduit comme un désenregistrement [BRE06]. L'UA n'est donc plus joignable. Cette attaque est possible si l'usager ne doit pas s'authentifier auprès du REGISTRAR.

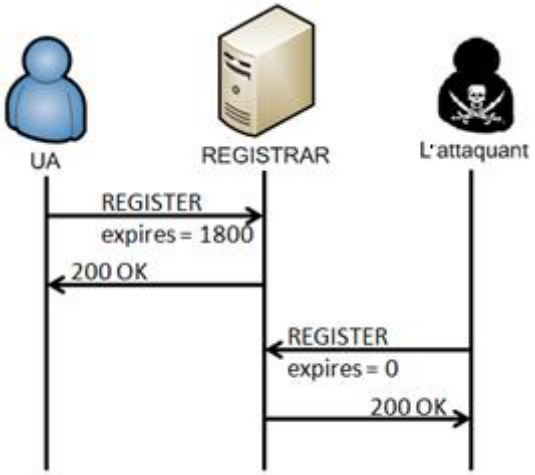

Figure 12. Attaque sur le REGISTER

3.3.3. *L'usurpation d'identité*

L'usurpation d'identité peut être la conséquence d'une authentification faible ou absente. En effet faute de pouvoir vérifier l'identité d'un usager, un proxy SIP peut fournir un service d'établissement de session à tout ce qui le demande. Des appels frauduleux sont alors imputés abusivement à des comptes SIP.

L'usurpation d'identité peut également concerner les serveurs. En usurpant l'adresse IP de ces derniers, un équipement illégitime reçoit tout le trafic SIP des usagers concernés. Sans authentification du serveur, le client continue d'émettre des requêtes SIP sans savoir qu'il dialogue avec un équipement pirate. L'attaquant peut alors avoir le détail de tous les appels et les contrôler. A partir de ce détournement de trafic, le pirate peut faire du déni de service et avoir la connaissance de tout le trafic émis par l'usager. L'authentification mutuelle est donc une nécessité dans le contexte de la téléphonie sur IP.

[19] Expires : le champ d'en-tête « expires » donne l'heure relative après laquelle le message expire. La signification dépend du message. Pour un REGISTER, un champ « expires » égale à zero est interprété comme une déconnexion. La valeur du champ est un nombre entier de secondes entre 0 et 2^{32}-1 mesuré depuis la réception de la demande.

3.4. SIP et la sécurité

Le RFC de SIP prévoit un certain nombre de mécanismes de sécurité pour assurer la confidentialité, l'intégrité, l'anonymat et l'authentification au travers de la signalisation. D'autres mécanismes de sécurité existent comme SRTP pour protéger la voix mais ne sont pas mentionnées dans le RFC. La figure 13 présente leur position dans la pile protocolaire SIP. Bien que les solutions de sécurité existent, il n'y a pas d'obligation à les utiliser. Cette situation conduit généralement à dire que la sécurité n'est pas le point fort de SIP qui reporte cette problématique au niveau des couches sous-jacentes.

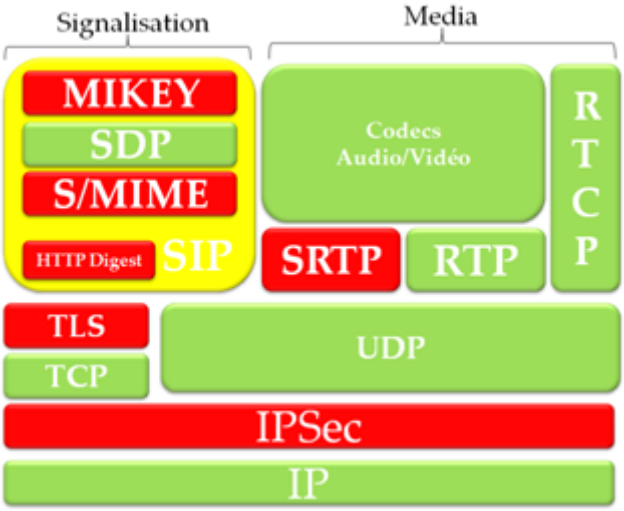

Figure 13. Pile protocolaire SIP avec les éléments de sécurité

La sécurité peut s'envisager de deux manières :

- soit de **proche-en-proche** entre un usager et un serveur ou entre serveurs. Les propriétés de sécurité ne sont alors définies qu'entre deux entités seulement ;
- soit de **bout-en-bout** entre les deux usagers.

De plus SIP fait intervenir la sécurité de deux façons : dans les messages SIP ou dans les couches sous-jacantes. Ainsi, les solutions comme S/MIME [RFC3851] ou HTTP Digest [RFC2617] sont dans les messages SIP alors que TLS [RFC2246] intervient au niveau Transport. Seuls les mécanismes intégrés aux messages SIP (i.e. dans la signalisation) peuvent permettre une solution bout-en-bout.

3.4.1. *La sécurisation de la signalisation*

3.4.1.1. L'authentification HTTP

L'authentification HTTP (méthode « Digest » et méthode « Basic ») [RFC2617] est un mécanisme basé sur un challenge/réponse. Il permet tout d'abord au client SIP de s'enregistrer auprès du REGISTRAR et ensuite d'avoir accès aux différentes ressources quand le serveur lui demande : une authentification est généralement demandée pour une requête INVITE. La figure 14 illustre l'authentification dans le cas d'un enregistrement.

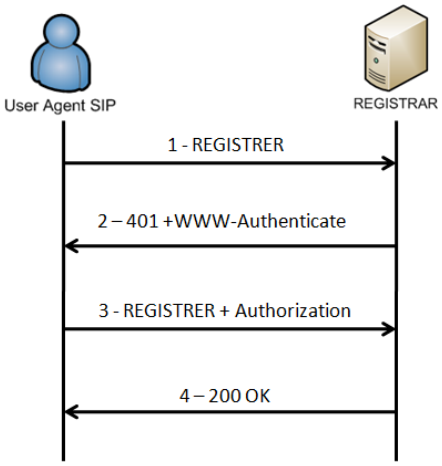

Figure 14. Enregistrement et authentification HTTP Digest dans un contexte SIP

Le principe de l'authentification HTTP Digest est classique. Le serveur envoie un challenge au client (« nonce » cf. figure 15), ce dernier répond par une valeur (« response » cf. figure 15) dérivée de ce challenge et d'un secret qu'il partage avec le serveur, généralement fourni avec le login par l'opérateur. Le serveur s'assure alors que le client possède effectivement le secret en calculant à son tour la réponse et en vérifiant la cohérence des deux. Bien que ce mécanisme ne soit pas particulièrement robuste, il est massivement implémenté dans les infrastructures de ToIP/SIP car il permet une grande mobilité. La version 2 de SIP déconseille la version « Basic » HTTP qui nécessite l'envoi du mot de passe en clair.

[RFC2617] permet également l'authentification mutuelle, néanmoins pour des raisons de compatibilité avec la version précédente du protocole SIP ce mode peut ne pas être possible. En effet le client ne peut demander une authentification mutuelle que si le serveur insère le champ « qop » dans le premier challenge. Cette condition limite considérablement l'utilisation de ce mode.

Les échanges de messages pour un enregistrement et le principe de l'authentification sont illustrés dans la figure 15. Le premier message informe le serveur du souhait du client de s'enregistrer par l'envoi d'une requête REGISTER. La réponse « 401 Unauthorized » permet au serveur d'envoyer son challenge sous la forme du champ « nonce » inclus dans le message SIP. Le client calcule la réponse « response » avec le secret pré-partagé qui renvoie dans une nouvelle requête REGISTER. Si la valeur « response » est conforme à l'attente du serveur, ce dernier envoie donc une réponse « 200 Ok ». Le client est enregistré, il peut donc téléphoner mais il n'a aucune certitude qu'il dialogue avec le serveur légitime.

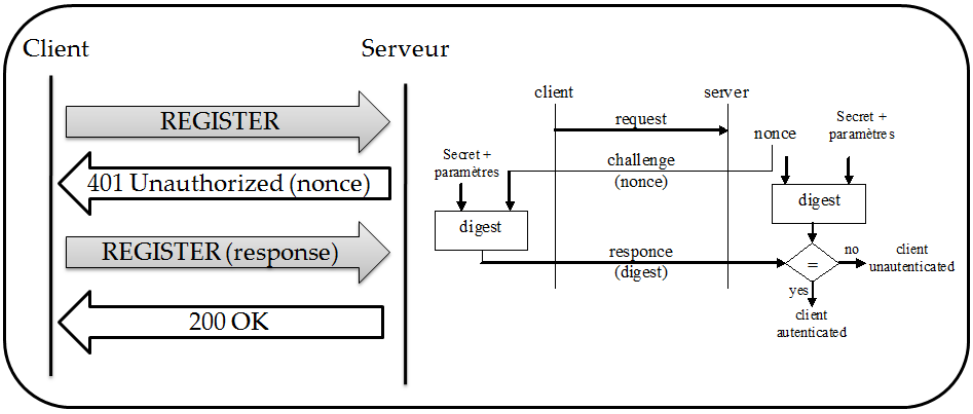

Figure 15. Authentification HTTP Digest SIP pour un message REGISTER

L'authentification est incluse dans la syntaxe des messages SIP. Un message 401 permet à un REGISTRAR de contester l'identité d'un usager à l'enregistrement. Pour les autres cas, le serveur SIP conteste l'identité du client avec un message 407.

Au message d'enregistrement REGISTER du client, le serveur répond par le message suivant en insérant le champ « WWW-Authenticate » qui contient le « nonce » (cf. figure 16) :

```
SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP 137.194.192.237:5060;received=137.194.192.237
From: <sip:ahmed@enst.fr>
To: <sip:ahmed@enst.fr>;tag=as7b4af592
Call-ID: D8A5240D579C4D6E8CE1@enst.fr
CSeq: 7168 REGISTER
User-Agent: Asterisk PBX
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, SUBSCRIBE,
NOTIFY
Max-Forwards: 70
Contact: <sip:ahmed@137.194.192.228>
WWW-Authenticate: Digest realm="asterisk", nonce="64d45b88"
Content-Length: 0
```

Figure 16. Exemple de message « 401 Unauthorized » SIP

Le client reçoit donc un challenge dans le message dit « 401 Unauthorized » sous la forme du « nonce ». Il forge alors la réponse en appliquant la formule suivante :

$$response = H(H(username||realm||password)||nonce||H(METHOD||Request-URI))$$

- *|| : correspond à un processus de concaténation ;*
- *H : H est par défaut la fonction de hachage[20] MD5 [RFC1321] (cf. [RFC3261]).*

[20] Fonction de hachage : fonction qui transforme une chaîne de caractères en une chaîne de caractères de taille inférieure et fixe appelé résultat du hachage ou condensat. Cette fonction satisfait deux propriétés. Il est difficile pour une image de la fonction de calculer l'antécédent associé. Il est difficile pour un antécédent de la fonction de calculer un antécédent différent ayant la même image.

L'application à notre exemple (cf. fig. 16) donne :

$$\text{response} = \text{H}(\text{H}(\text{ahmed}||\text{asterisk}||< password) > ||64d45b88||H(REGISTER||sip{:}enst.fr))$$

Le client renvoie un REGISTER avec un champ « Authorization » et la valeur « response » (cf. figure 17) :

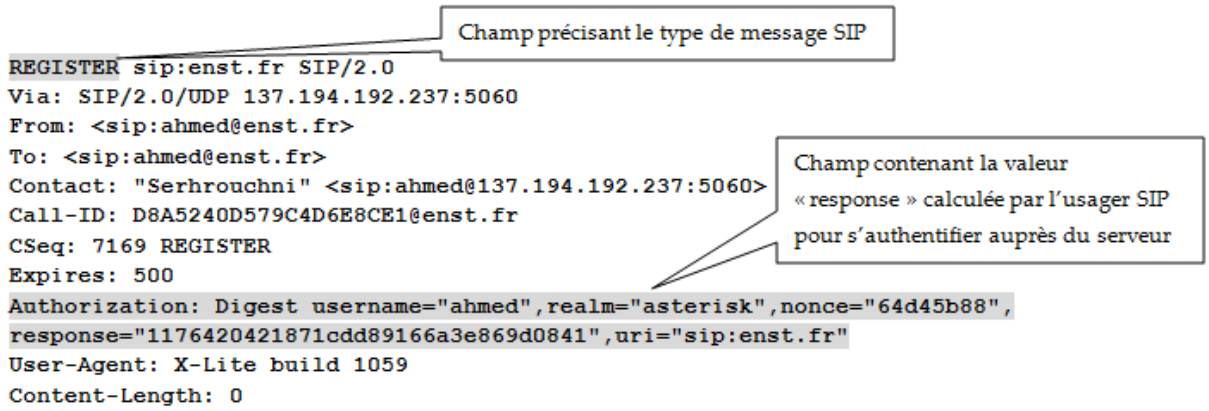

Figure 17. Exemple de message REGISTER du protocole SIP avec le champ « response »

SIP fournit donc un mécanisme d'authentification simple basé sur HTTP Digest directement intégré dans l'en-tête des messages SIP. A chaque requête d'un usager, le serveur SIP peut demander une authentification (sauf pour ACK).

Par ailleurs, cette méthode autorise les attaques par dictionnaire ou par force brute pour découvrir les mots de passe : il suffit pour cela d'avoir une association nonce/response pour refaire le calcul avec la méthode du RFC. Plus le mot de passe est court plus l'attaque est facile. Il est donc recommander d'avoir des mots de passe d'une longueur conséquente utilisant des minuscules, des majuscules, des caractères spéciaux et des chiffres comme le montre le tableau 7.

Tableau 7. Entropie d'un mot de passe [AUT07]

Caractéristiques du mot de passe	Nombres total de symboles	10 symboles (chiffres)			26 symboles (lettres)			62 symboles (chiffres, majuscules, minuscules)			90 symboles (jeu de caractères complet)		
	Nombre de symboles par mot de passe	4	7	10	8	10	16	8	10	16	8	10	16
Taille de clé équivalente (bits)		13	23	33	38	47	75	48	60	95	52	65	104
Ordre de grandeur du temps d'énumération du dictionnaire des mots de passe possibles par un ordinateur personnel		~0	~0	3 min	1h	1 mois	∞	1 mois	5 siècles	∞	2 ans	200 siècles	∞

Cette méthode présente donc quelques limites :

- d'une manière générale seul le client s'authentifie. L'usurpation de serveur SIP est donc possible [SHA09] ;
- la méthode est sensible à l'attaque par force brute sur le mot de passe. Il faut donc privilégier des mots de passe longs et non triviaux [RFC2617] ;
- cette solution n'apporte aucune confidentialité.

Cette authentification associant un nom de compte à un mot de passe, cette méthode est significative pour un domaine. Par ailleurs, un serveur peut décider de ne pas authentifier les usagers. L'identité d'un usager SIP peut alors être utilisée par tout le monde, de même que le champ anonyme [§ 3.4.7.], par définition sans mot de passe. Enfin, il faut noter que HTTP Digest est le seul mécanisme de sécurité entièrement situé dans l'en-tête SIP.

3.4.1.2. S/MIME

Les messages SIP portent des corps de type MIME[21] et peut donc utiliser sa version sécurisée S/MIME [RFC3851]. S/MIME permet de sécuriser une partie des messages SIP en utilisant le principe de chiffrement clé publique. Il permet d'assurer la confidentialité, l'authentification et l'intégrité. Les certificats permettent soit de chiffrer, soit de signer les messages SIP. La confidentialité et l'intégrité sont assurées par l'utilisation de la clé publique du destinataire. L'authentification et l'intégrité sont quant à eux assurés en utilisant la clé privée de l'émetteur. S/MIME dans un contexte SIP permet trois utilisations ; la transmission d'un certificat, la signature et le chiffrement.

Le chiffrement de tout le message SIP de bout-en-bout pour des besoins de confidentialité n'est pas approprié à cause des intermédiaires du réseau qui ont besoin de voir certains champs des en-têtes afin d'acheminer correctement les messages : si les intermédiaires sont exclus des associations de sécurité, les messages ne sont pas acheminables. Une sécurité bout-en-bout (intégrité et confidentialité) est envisageable pour le corps des messages SIP, incluant une authentification mutuelle des usagers. Le mode « tunnel » permet d'étendre la sécurité à l'en-tête.

Un des gros défauts de cette solution est l'absence d'infrastructure de certificats largement déployée pour les vérifier. Il est toujours possible de s'échanger des certificats avec SIP mais un attaquant peut toujours intercepter et modifier le message S/MIME. Ce dispositif nécessite également d'associer à chaque URI une clé publique ce qui n'est pas forcément facile. Enfin cette solution augmente d'une manière très significative la taille des messages SIP.

[21] MIME : Multipurpose Internet Mail Extensions est spécifié dans plusieurs RFC.

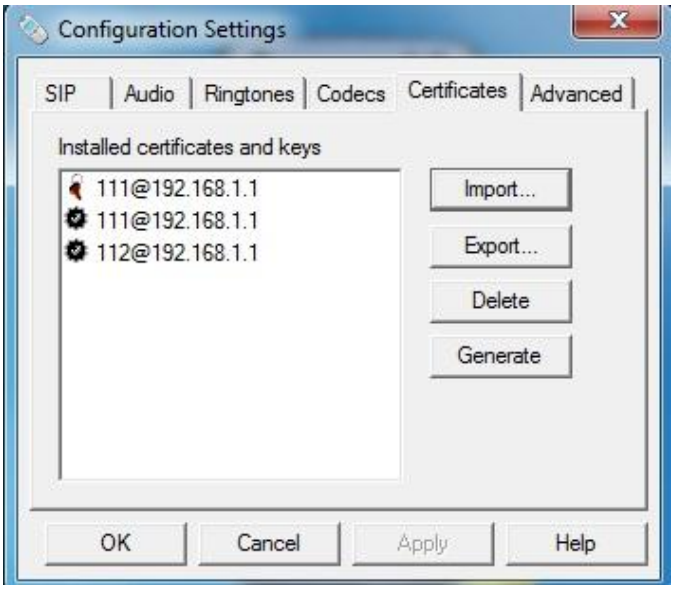

Figure 18. Interface graphique de Lynxphone pour la gestion des clés (publique et privées)

[RFC3261] décrit l'application de S/MIME au contexte SIP. Il définit en particulier un mode tunnel qui assure la confidentialité, l'intégrité et l'authentification. Cette propriété est fournie à deux niveaux dans la signature et le chiffrement. En effet S/MIME utilise la clé privée de l'usager pour signer l'en-tête du message SIP, l'appelant s'authentifie ainsi. L'authentification du destinataire est quant à elle garantie en appliquant sa clé publique au corps du message. Le choix des champs à chiffrer a principalement un impact sur la confidentialité et l'intégrité mais pas sur l'authentification, qui repose sur l'utilisation des certificats. La syntaxe utilisée est celle de Cryptographic Message Syntax CMS [RFC2630], lui-même issue de la syntaxe Public-Key Cryptography Standards PKCS#7 [RFC2315]. Le corps des messages construit selon S/MIME présente deux types de contenu issus de [RFC2315] (cf. figure 19) :

- les contenus signés qui sont identifiés par l'extension « .p7s » ;
- les contenus chiffrés qui sont identifiés par l'extension « .p7m ».

La figure 19 illustre un message SIP avec un corps de type S/MIME. Cette trame est issue des essais réalisés dans le cadre de cette thèse avec des softphones Lynxphone [LYN] (cf. figure 18) qui supportent ce mécanisme de sécurité ; l'acheminement de la signalisation étant assuré par l'IPBX Trixbox [TRI]. Les certificats sont au format Privacy Enhanced Mail (PEM), les extensions « .pem » et « .cert » sont donc usuelles dans le contexte.

```
INVITE sip:112@192.168.1.1 SIP/2.0
Via: SIP/2.0/UDP 192.168.1.2:23601;branch=z9hG4bK-d8754z-e04042155d4a285d-1---
d8754z-;rport
Max-Forwards: 70
Contact: <sip:111@192.168.1.2:23601>
To: <sip:112@192.168.1.1>
From: "thomas"<sip:111@192.168.1.1>;tag=1e55e630
Call-ID: N2U4OWE0NGRjYzFiMjc5ZTY4MTgwYTYyOGIyYjI3OWY.
CSeq: 2 INVITE
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, NOTIFY, REFER
Content-Type:
multipart/signed;boundary=7d53ad671e02307d;micalg=sha1;protocol="application/pkcs7-
signature"
User-Agent: LynxPhone 0.6.7 - Bitlynx Technologies Inc [WinVista]
Authorization: Digest
username="111",realm="asterisk",nonce="2c97195a",uri="sip:112@192.168.1.1",response
="9a208754a4f582459925a8551fc01c98",algorithm=MD5
Content-Length: 1362
--7d53ad671e02307d
Content-Type: application/pkcs7-mime;smime-type=enveloped-data;name=smime.p7m
Content-Disposition: attachment;handling=required;filename=smime.p7m
Content-Transfer-Encoding: binary
0%82%02%9d%06%09*%86H%86%f7%0d%01%07%03%a0%82%02%8e0%82%02%8a%02%01%001%81%d20%81%c
f%02%01%0008001%140%12%06%03U%04%0a%13%0b192.168.1.11%180%16%06%03U%04%03%14%0f112@
192.168.1.1%02%04{e@k0%0d%06%09*%86H%86%f7%0d%01%01%01%05%00%04%81%80'%94b%8bCc%05%
9a%00%aa%e4%a7(;%08%f1%1dvp%06US%b3%f3%da%192%0c%82E%af,b;%e4%d2%e1%b6C%0e%df%aeQ%a
b%a6%04%e1%18R:%97%fb%a6%a7.?y%02%ac%f0%c4%08<%b5)%fcf%b8%d2%02G%d3%9c~~%9e%a2%d8%1
3%e8%dc%86%03%9cTc%db@%b9%08%b7%dd7=%04%7f%86=s%bf%a9%e7W%8e%04%f3!n%d8%175%14%96N%
ee%cb%f3%1fh%9a%a9s%a0%e9%86%aa%85%130%82%01%ae%06%09*%86H%86%f7%0d%01%07%010%1d%06
%09`%86H%01e%03%04%01%02%04%10!%f8%82%8e%b8%95$%fc%dbV%a3@NA%d5^%80%82%01%80%d5%cc%
86%80%c6%c0=%a3%830%df%80}R%0d%e7v%09N%ce%de%02%93B"%d2%0a%d1c%f7I|%0d%ed%fe%b4%f0%
16rR%e6(%ab@%0a%a1`%ca%e2%d6}{%a5%cf%a6%03>`S%c1%07%eb%1b%a3%99%ee%96%17q%10%fc%f8%
b8%df%a8o%a6H%c1%cdPv%a9>%f62%15%05v%0c%b9%f4%da%89q%12%fa%15%88%c1%c9%81o%19)%a7%8
c%b2*%fc%b9b;X%06K%bb%eb%82%eeg%06Yl%e3i(%f4%b0l%d9%cdG%0f%01%13O%f9%ff\T%df'%c8%99
%ab%1a%f1%ede%f7%afQ%e9%18%8a%85H>%89%10%cb%f7%9e%02%02%c7%fc%c8%d0%9e%e8%e7<9C%a9%
fd%80%8fH%b7%1dX%e0%da%dd%0b%0dcU%fb%17I_n%fa~%b8%7f&x+d%8c%ee%e6%0b%bf%bb%z2H%aa%1
b%8dZ%b1%e8%e8H?34oR%f2%0dpZy%cfX%14%df%b0P%cc%ed!%88t<%c5i%cb%99%eeu%e3%85ut;n6@%1
8%1dw%bc%abe_%b9W%b4%c3%b5%f1Jg%03Kw%de%c8%9c]iL%e4%00%89%d0/#l%16y"%ae%e7%90%b3%7f
%f3%9f%08s%0a%96%a5m%cbj_jHq%c5%e6%ba%a2*:%e7`%99%b5%0d%c62h%88%0e%f9%1b%06%fe%80%0
c:%d8%e4%a4%c4%e7%dc*a%92%b8%01%10%86MK%88%bbb%0a%90%0dDSK%bf%f1Gu%9ar%95%93]%f0%9f
L%c0%ff%9b%82Lm%f2%e7%91b&%15
--7d53ad671e02307d
Content-Type: application/pkcs7-signature;name=smime.p7s
Content-Disposition: attachment;handling=required;filename=smime.p7s
Content-Transfer-Encoding: binary
0%82%01%0f%06%09*%86H%86%f7%0d%01%07%02%a0%82%01%000%81%fd%02%01%011%0b0%09%06%05+%
0e%03%02%1a%05%000%0b%06%09*%86H%86%f7%0d%01%07%011%81%dd0%81%da%02%01%0108001%140%
12%06%03U%04%0a%13%0b192.168.1.11%180%16%06%03U%04%03%14%0f111@192.168.1.1%02%04_%b
0;%a80%09%06%05+%0e%03%02%1a%05%000%0d%06%09*%86H%86%f7%0d%01%01%01%05%00%04%81%80a
%07%1d%d1%f7I%dfC*7%fe%b9n%9cs%04;%c7%bae{%84%a4@Jj%a5}%ef;@%94Y%f5s%1d%c88%ce%e7%0
8<%ac%a7^%cb,%17%cb%edoi@%e4%c1%de>&%1c%c5M*Ooe%a4%b7%c7%9f%faB%97%d4%b8H%f1%ee%c7%
e8%ed%d5%10I%f0}%ea:%93%1ckUTAF%b8%cb%12<%04%fa%10h%ad%c6%d9|%15%13%b6S%85"%d1%e8%a
9M%fb%f9}%ed2%e0%b0T%ac!%14
--7d53ad671e02307d—
```

Figure 19. Message SIP avec la solution de sécurité S/MIME

S/MIME apporte une solution de sécurité de bout-en-bout pour l'authentification SIP. Certes la notion de confidentialité est limitée puisque l'entête est en clair pour permettre l'acheminement de la signalisation. De même, l'intégrité est restreinte à certain champ puisque certains serveurs SIP rajoutent des champs. Ces restrictions ne concernent pas l'authentification mutuelle. En effet, un usager en déchiffrant avec la clé publique de son correspondant l'authentifie et s'authentifie en déchiffrant la partie chiffrée avec sa clé publique.

Cette méthode présente quelques limites :

- elle nécessite une certaine maturité de l'utilisateur qui doit associer à chaque correspondant un certificat, ce qui dénote déjà une certaine volonté de sécurité inexistante pour un particulier actuellement. De plus dans le cadre de la mobilité, les paires adresses/certificats doivent être associés à l'utilisateur et non à l'équipement. En effet, si les certificats sont associés à une machine, l'utilisateur doit forcément utiliser cette dernière ;
- certains intermédiaires modifient ou complètent les messages SIP : rajout de champ « via », modification du SDP. Ce qui signifie que S/MIME peut empêcher ces serveurs de travailler et donc de faire aboutir l'appel ;
- elle nécessite une infrastructure prévalente de clés publiques pour les usagers. Il est toujours possible d'utiliser les certificats auto-signés. La vulnérabilité vient alors pendant l'échange. Dans la mesure où ce certificat ne peut pas être vérifié, s'il y a une modification du certificat pendant sa transmission, il n'est pas possible de s'en rendre compte ;
- elle augmente la taille des messages (voir la différence de taille entre le message INVITE de la figure 16 et celle de la figure 19).

S/MIME permet une authentification mutuelle entre l'appelant et l'appelé. Cette solution bout-en-bout nécessite l'échange des certificats auparavant. Cette démarche reste encore marginale dans le cadre traditionnel de la téléphonie. Le dispositif pourrait alors être porté par le fournisseur qui déploierait les softphones avec les associations adresse/certificat. Cette configuration est envisageable dans un domaine de confiance. SIP prévoit d'autres mécanismes d'authentification pour la signalisation mais dans les couches sous-jacentes.

3.4.1.3. TLS et SIPS

SIP prévoit la sécurisation des échanges au niveau de la couche Transport avec Transport Layer Security (TLS) [RFC2246]. TLS, anciennement nommé Secure Sockets Layer (SSL), est un protocole de sécurisation des échanges sur Internet. TLS est un protocole modulaire dont le but est de sécuriser les échanges Internet entre le client et le serveur indépendamment de tout type d'application. TLS agit comme une couche supplémentaire au-dessus de TCP. Ainsi TLS ne s'occupe pas de fiabilité de couche Transport ni du maintien de la connexion. Les services offerts sont : l'authentification, l'intégrité et la confidentialité. Son implémentation native dans de nombreux navigateurs a fait de TLS le standard de sécurisation des applications Web : HTTPS correspondant à l'association d'HTTP avec TLS. Son utilisation est principalement associée à l'utilisation des certificats X.509[22] pour l'authentification des serveurs et le chiffrement des échanges (i.e. la signalisation). Le RFC initial de SIP ne décrivant que très sommairement l'association SIP/TLS, [RFC4474] a été édité pour précisé le fonctionnement des deux protocoles.

[22] X.509 : X.509 est une norme de cryptographie de l'Union internationale des télécommunications pour les infrastructures à clés publiques (PKI). X.509 établit entre autres les formats standards de certificats électroniques.

TLS fournit la sécurité de la couche Transport en mode connecté. L'utilisation de TLS est spécifiée dans le champ via de l'en-tête SIP ou dans l'URI SIP. Ce choix entraine l'ouverture d'une connexion TLS classique (cf. figure 20) permettant par la suite des échanges de messages SIP chiffrés. Par ailleurs, TLS est bien adapté aux architectures dans lesquelles la sécurité de proche-en-proche est demandée alors qu'il n'existe pas de relations de confiance. Les serveurs peuvent s'échanger leur certificat et les faire vérifier auprès d'une autorité de confiance. TLS est spécifique à une application qui est explicitement associé à un port (5061 pour SIP/TLS, 5060 pour SIP/TCP ou UDP).

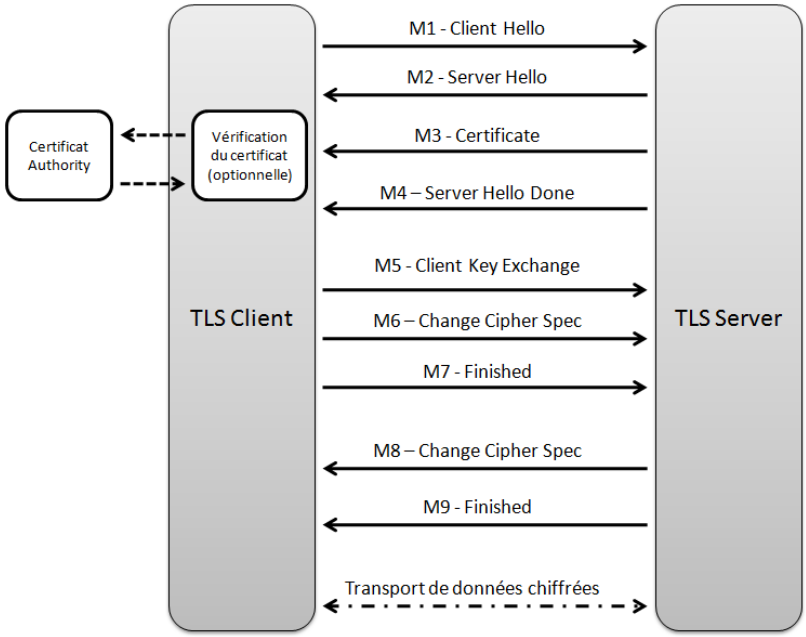

Figure 20. Echanges de messages TLS

TLS spécifie les fonctions suivantes :

- l'échange de messages d'ouverture de session, incluant la proposition des algorithmes et la méthode de négociation des clés de session ;
- l'authentification du serveur voire du client ;
- l'élaboration d'une clé « maître » ;
- la dérivation des clés de session à partir de la clé « maître » ;
- la transmission des paramètres des algorithmes de chiffrement et des aléas ;
- la vérification des paramètres.

La figure 20 illustre les échanges TLS permettant le chiffrement des échanges SIP. « Client Hello » et « Server Hello » initialisent l'échange. Le serveur fournit ensuite certains paramètres mais surtout le certificat qui permettra au client d'authentifier le serveur. Ensuite ce sont les échanges pour l'élaboration de la clé « maître ». Ces processus terminés, la signalisation de SIP est ensuite transportée chiffrée par TLS et TCP.

Concernant la mise en œuvre à la création d'un compte SIP, le fournisseur ou l'entreprise doit distribuer le certificat avec la clé publique du serveur SIP. La figure 21 présente le certificat généré pour les essais réalisés dans le cadre de cette thèse.

```
-----BEGIN CERTIFICATE-----
MIICzjCCAjegAwIBAgIBADANBgkqhkiG9w0BAQQFADCBkjELMAkGA1UEBhMCRl
IxDjAMBgNVBAgTBVBhcmlzMQ4wDAYDVQQHEwVQYXJpczERMA8GA1UEChMIM0NY
IEx0ZC4xIDAeBgNVBAsTF1RlbGVjb21tdW5pY2F0aW9ucyAgUEJYMREwDwYDVQ
QDEwgzQ1hQSE9ORTEbMBkGCSqGSIb3DQEJARYMaW5mb0AzY3guY29tMB4XDTEw
MDIyNTEzMTUyN1oXDTIwMDIyMzEzMTUyN1owgZIxCzAJBgNVBAYTAkZSMQ4wDA
YDVQQIEwVQYXJpczEOMAwGA1UEBxMFUGFyaXMxETAPBgNVBAoTCDNDWCBMdGQu
MSAwHgYDVQQLExdUZWxlY29tbXVuaWNhdGlvbnMgIFBCWDERMA8GA1UEAxMIM0
NYUEhPTkUxGzAZBgkqhkiG9w0BCQEWDGluZm9AM2N4LmNvbTCBnzANBgkqhkiG
9w0BAQEFAAOBjQAwgYkCgYEA2DEO5HClSZmB4v86YqVKKkrY2L6yutNHDk0L8m
UW/Scs7oAKT0dKpD+yioeRYZVTuZum/6L6D/VPsWH+AkImEcCu7UMteOaACbFp
9qMPglLCs9a3TWgJbK+t8SC7cRtX/N5D+4GxnLewgdsSimITNivywpldT6+o2G
NzPgMPNa8CAwEAAaMyMDAwDwYDVR0TAQH/BAUwAwEB/zAdBgNVHQ4EFgQU0KFy
CX15dAiYpCmMnPJe6P1n+cQwDQYJKoZIhvcNAQEEBQADgYEAe46oBpa+CG99jx
D013p99vHRYkcLY1AKmtASkLFxo9pxQDErg4QWz3cNjyEZe1T5akMg9p/bN3H1
tyf7rsO8irKDz+xrXprGaHkNzQE62msNqNNpYiP8h3+ywasWma6Qbn+3vNZQjg
MxmJzxFnVZ+hnMI6ntx8VC393/WOBaJVQ=
-----END CERTIFICATE-----
```

Figure 21. Exemple de certificat d'un serveur SIP

Pour réaliser la mise en œuvre de cette solution, un IPBX 3CX [3CX] qui supporte TLS a été installé sur un PC. Ce dispositif a permis d'observer au travers des trames la solution TLS appliquée à SIP. La figure 22 présente la phase d'authentification, d'envoi de certificat et l'élaboration de la clé « maître ».

137.194.192.243	137.194.192.240	TCP	49638 > sip-tls [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=2
137.194.192.240	137.194.192.243	TCP	sip-tls > 49638 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=0
137.194.192.243	137.194.192.240	TCP	49638 > sip-tls [ACK] Seq=1 Ack=1 Win=65700 [TCP CHECKSUM INCORRECT] Len=0
137.194.192.243	137.194.192.240	SSL	Client Hello
137.194.192.240	137.194.192.243	TLSv1	Server Hello, Certificate, Server Hello Done
137.194.192.243	137.194.192.240	TLSv1	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
137.194.192.240	137.194.192.243	TLSv1	Change Cipher Spec, Encrypted Handshake Message
137.194.192.243	137.194.192.240	TLSv1	Application Data, Application Data
137.194.192.240	137.194.192.243	TCP	sip-tls > 49638 [ACK] Seq=927 Ack=981 Win=64555 Len=0
137.194.192.240	137.194.192.243	TLSv1	Application Data, Application Data
137.194.192.243	137.194.192.240	TCP	49638 > sip-tls [ACK] Seq=981 Ack=1529 Win=65700 [TCP CHECKSUM INCORRECT] Len=0
137.194.192.243	137.194.192.240	TLSv1	Application Data, Application Data

Figure 22. Exemple d'échanges TLS pour SIP

TLS permet au client d'authentifier le serveur. L'utilisation d'un certificat client autoriserait une authentification mutuelle au niveau Transport, mais obligerait le serveur à posséder le certificat avec la clé publique de tous les usagers : cela compliquerait significativement le système. De plus, ce cas n'est pas décrit formellement dans [RFC3261]. Ce rajout de messages n'est d'ailleurs pas forcément utile car le serveur peut authentifier le client avec HTTP Digest. On peut considérer que HTTP Digest + TLS permet une authentification mutuelle. Concernant une connexion TLS entre deux domaines, le RFC de SIP préconise fortement une authentification mutuelle.

L'utilisation de TLS est également très liée au schéma URI SIPS pour une solution bout-en-bout. Cette syntaxe signifie que chaque saut sur lequel la demande est transmise doit être sécurisé avec TLS. SIPS permet aux ressources de spécifier qu'elles devraient être jointes de manière sécurisée. Utiliser TLS sur chaque saut signifie que les usagers s'enregistrent avec une URI SIPS. S'assurer que c'est bien le cas est assez complexe. Il est

toujours possible pour un serveur compromis ou non conforme de ne pas suivre les règles de transmission associés à SIPS. Les limites décrites dans le RFC montrent qu'il est délicat de garantir et de contrôler l'application de cette configuration. D'ailleurs peu de clients implémentent SIPS et TLS [GEN05].

Enfin SIP étant régulièrement utilisé avec UDP, un projet de RFC a été proposé pour faire fonctionner SIP sur DTLS [JEM07] l'équivalent de TLS pour TCP. Cependant comme le constate [PAR08], DTLS [RFC4347] n'est pas communément déployé.

3.4.1.4. IPSec

Pour protéger les échanges dans les réseaux, une des solutions usuelles consiste à utiliser le protocole IPsec (IP security) [RFC2401], la version sécurisée d'IP. De même que SIP prévoit la sécurisation des échanges au niveau de la couche Transport, il envisage une protection au niveau Réseau avec IPSec. Ce protocole permet en effet d'authentifier l'origine de paquets IP, de garantir l'intégrité voire la confidentialité. IPSec permet donc de protéger des communications et la signalisation entre deux entités. Deux modes sont possibles : le mode transport ou le mode tunnel. Quelque soit le mode, le serveur SIP peut modifier les en-têtes SIP et permettre l'établissement de l'appel. D'une manière générale, les clients SIP n'implémentent pas cette solution. IPSec est donc principalement utilisé pour protéger le trafic entre deux domaines [SAW06].

IPSec permet l'encapsulation des datagrammes IP. Toutes les applications dont celles basées sur SIP peuvent donc bénéficier de ses propriétés de sécurité. IPSec est un ensemble de protocoles complètement indépendant de SIP spécifiant essentiellement deux aspects :

- l'encapsulation des datagrammes IP dans d'autres datagrammes IP de manière à fournir des services de sécurité classiques : intégrité, confidentialité, authentification ;
- la négociation des clés et des associations de sécurité utilisées lors de l'encapsulation.

Deux protocoles sont définis pour l'encapsulation, AH (Authentication Header, cf. figure 23) et ESP (Encapsulating Security Payload, cf. figure 24). AH fournit le service d'authentification et l'antirejeu. ESP par rapport à AH fournit en plus la confidentialité.

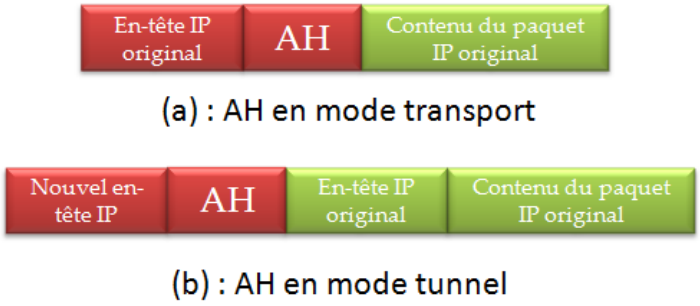

Figure 23. IPSec en mode AH

Les modes AH et ESP peuvent être utilisés en mode transport ou en mode tunnel. Le mode transport est utile pour sécuriser des communications de bout en bout (par exemple, de PC à PC). Le mode tunnel est principalement utilisé pour sécuriser les échanges transitant entre passerelles de sécurité. Ce dernier mode est utilisé pour la construction des VPN IPSec. Pour la gestion des clés, IPSec utilise IKE (Internet Key Exchange).

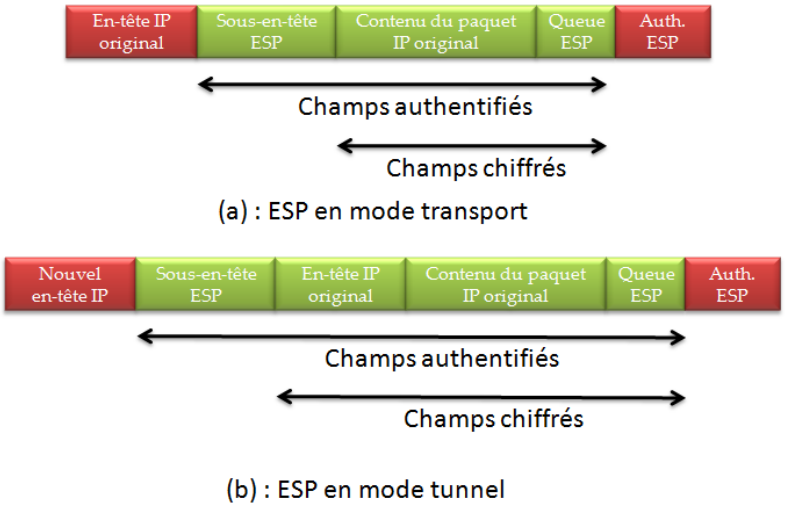

Figure 24. IPSec en mode ESP

Le fonctionnement d'IPSec étant complètement indépendant de SIP, le fonctionnement détaillé de ce protocole ne présente pas d'intérêt dans le cadre de ces travaux. Par ailleurs tous les modes offrent un service d'authentification de l'origine des paquets. IPSec est principalement utilisé pour les VPN entre deux sites distants. Dans notre cas, l'authentification concerne alors le flux voix et la signalisation. Il pourrait être envisagé d'utiliser IPSec entre le client et le serveur mais il faut être administrateur pour la configuration : dans ce cas on préférera TLS car l'usager a toute latitude pour insérer le certificat de l'IPBX dans son softphone.

3.4.2. *La sécurisation de la voix*

Comme cela a été mentionné précédemment, la voix est transportée par le protocole RTP. Pour protéger ce flux, IPSec peut être une solution pour chiffrer les échanges voix entre deux points (cf. paragraphe 3.4.1.4.). Il existe une autre solution de sécurité propre à RTP. Secure Real-time Transport Protocol (SRTP) [RFC3711] permet de chiffrer une conversion de bout-en-bout. Bien que le RFC de SIP ne fasse pas mention de SRTP puisque datant de 2004 (alors que SIP version 2 est de 2002), cette solution est incontournable dans le contexte SIP. Ce standard proposé par CISCO et Ericson permet le chiffrement de la voix en garantissant la confidentialité, l'intégrité, le non-rejeu et l'authentification des paquets voix.

La construction des paquets SRTP est assez proche de celle des paquets RTP. La différence vient du chiffrement d'une partie du paquet et du rajout des champs permettant la sécurité. Deux nouveaux champs ont été créés :

- MKI (Master Key Identifier) : ce champ est défini et utilisé par le protocole de gestion de clé. Il identifie la « Master Key » à partir duquel les clés de session sont dérivées pour les opérations de chiffrement et de déchiffrement ;
- « Authentication Tag » ; ce champ permet d'authentifier le paquet RTP et fournit un mécanisme contre le rejeu.

SRTP définit un mécanisme de gestion des clés de session. A partir de la « Master Key », le protocole génère des clés de session qui sont utilisées directement pour le chiffrement des données. Concernant la génération de la « Master Key », SRTP fait appel à un autre mécanisme pour la négociation de clés. (cf. figure 25). Deux solutions ont été développés spécifiquement pour SRTP. Ce sont MIKEY [RFC3830] et ZRTP [ZRTP10]. Pour le cas de MIKEY, ce protocole permet aux usagers de se mettre d'accord sur la clé « Master Key » à partir d'un secret, d'une clé publique (PKI) ou d'un échange Diffie Hellman, et sur l'algorithme cryptographique. On peut citer également le protocole SDP Security Descriptions for Media Streams [RFC4568]. C'est cette méthode qui est illustrée dans la figure 25 où sont présents deux messages générées pendant les mises en œuvre de la thèse. L'usager SIP à l'origine de l'appel envoie un message INVITE avec, dans le corps SDP, plusieurs méthodes pour la protection des paquets RTP. L'appelant choisit une des méthodes, valide la « Master Key » et renvoi ces éléments dans le corps SDP de la réponse 200 OK. Concernant l'algorithme de chiffrement, SRTP utilise par défaut l'algorithme AES (Advance Encryption Standard).

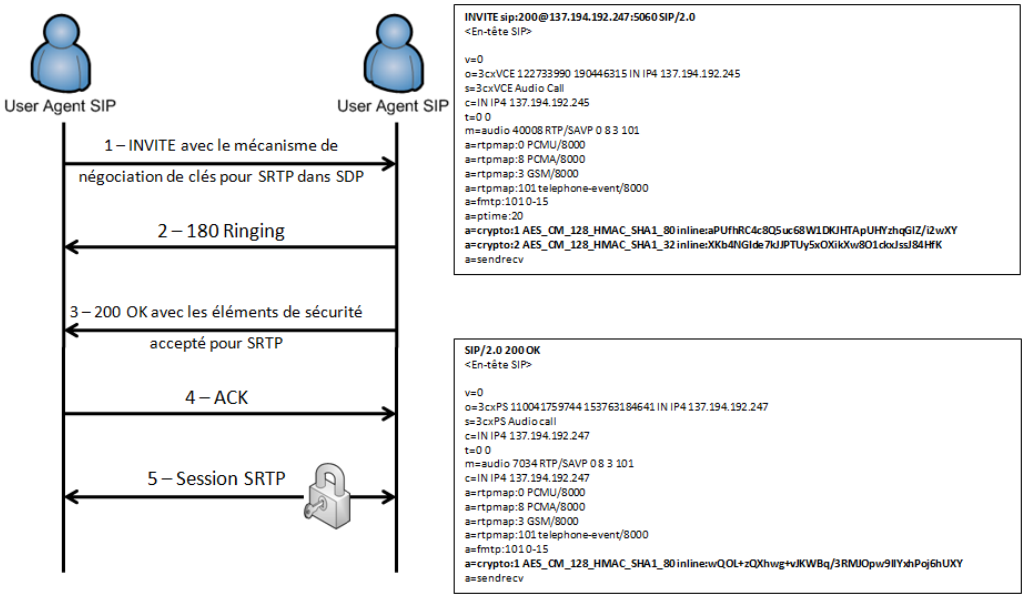

Figure 25. SRTP dans un contexte SIP

SRTP n'est pas encore massivement déployé dans les infrastructures de ToIP. Comme le précise [PAR08], les problèmes de performances (voir l'annexe VI sur l'augmentation d'occupation de la bande passante), de complexité de l'implémentation, et d'interopérabilité limitent l'adoption de ce mécanisme.

3.4.3. *SIP et la notion d'anonymat*

L'utilisation des réseaux sociaux comme Facebook et Twitter a popularisé la notion de respect de la vie privée liée à l'utilisation des réseaux Internet. La téléphonie sur IP n'échappe pas à cette réflexion puisque, pour établir une session, SIP échange un certain nombre d'informations comme votre identifiant. Dans la mesure où il n'y a pas de solutions garantissant le chiffrement de la signalisation, ces données sont la plupart du temps échangées en clair. La simple interception peut donc permettre de connaître les habitudes et les correspondants d'un usager. Cette connaissance du trafic émis par une personne peut s'expliquer dans le cas de son opérateur pour la facturation par exemple. Mais les requêtes SIP étant transmises de proxy SIP en proxy SIP pour localiser l'appelé, d'autres entités peuvent enregistrer les demandes de connexion d'un usager (cf. figure 26). Les en-têtes SIP contiennent l'origine et le destinataire de l'appel. Dans la mesure où ces données sont en clair, tous les serveurs traitant l'appel enregistrent les détails de la session.

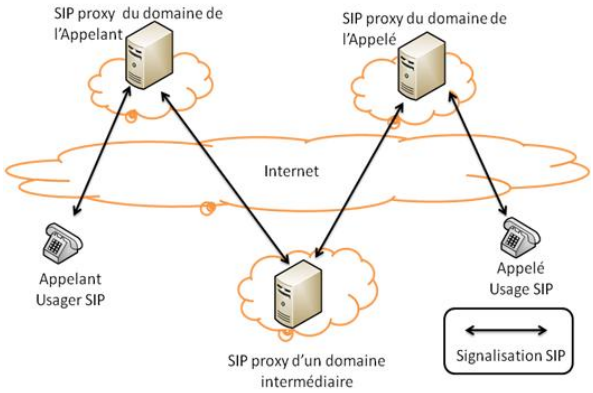

Figure 26. Cheminement de la signalisation SIP dans les réseaux IP

Le RFC de SIP prévoit une forme d'anonymat au travers de l'utilisation du From. Ce champ indique l'entité logique de l'initiateur d'une requête. Cela peut être l'adresse d'enregistrement mais pas nécessairement. From peut contenir une URI qui est une adresse d'affichage. L'origine de l'appelé est alors « déguisée » par une URI générique du type : From : Anonyous <sip:anonymous@enst.fr>. Le champ From permet à une entité SIP de déterminer les règles de traitement à appliquer aux messages. L'utilisation d'une telle URI nécessite donc une configuration particulière du serveur SIP comme pour l'authentification ; il peut alors supporter un nom d'utilisateur anonyme qui n'a pas de mot de passe. Une URI de type « Anonymous » peut également être associée à S/MIME. Le message avec le From de l'appelant est chiffré avec la clé publique de l'appelé alors que l'en-tête en clair contient un From anonyme.

Naturellement l'anonymat peut être également traité au travers des propriétés de sécurité comme la confidentialité ou l'authentification. La confidentialité permet de chiffrer certains champs, néanmoins pour l'établissement de l'appel, il est bien nécessaire que le destinataire et l'appelant puissent être connus par une entité pour permettre l'établissement de la session. L'authentification fait également partie des solutions puisqu'elle permet dans le cas d'une authentification du serveur par un usager de s'assurer que le serveur est bien légitime. [RFC3325] est consacré à ce sujet et en propose une analyse du besoin, mais surtout préconise des manières de remplir les champs des en-têtes SIP pour préserver la vie privée des usagers.

3.5. Analyse des solutions actuelles

3.5.1. *Les limites des solutions actuelles*

Le protocole SIP n'est pas facile à sécuriser. La présence de nombreux intermédiaires dans son architecture avec des relations de confiance de différents niveaux ne facilite pas le déploiement de solutions de sécurité. En effet l'acheminement des messages pour l'établissement d'une session nécessite qu'une partie des en-têtes soit en clair pour permettre l'analyse des requêtes. Par ailleurs l'absence de solutions imposées dans le standard laisse à chacun l'appréciation de l'implémentation de SIP. Cette situation ne facilite donc pas une sécurité de bout-en-bout des sessions.

TLS et IPsec n'offrent pas de solution bout-en-bout sauf si l'appel s'effectue dans un domaine ou entre plusieurs domaines qui adoptent une même politique de sécurité. Néanmoins, il est difficile de connaître la politique de sécurité du domaine auquel appartient son correspondant. S/MIME peut permettre l'authentification mutuelle entre usagers à condition évidemment que ces derniers puissent échanger leur certificat. S/MIME permet également d'assurer l'intégrité des messages. Cette propriété ne peut cependant pas s'appliquer à l'ensemble du message SIP puisque les serveurs intermédiaires peuvent rajouter de nouveaux champs. De même la confidentialité de l'ensemble du message SIP n'est pas chose facile dans la mesure où les serveurs intermédiaires doivent lire les en-têtes pour l'acheminement des messages. Les limitations sont donc nombreuses. Dans ce contexte, cette analyse s'est focalisée sur la problématique de l'authentification.

3.5.2. *Les enjeux pour l'authentification dans une architecture SIP*

Le paragraphe 3.4. a permis de comprendre les différentes interactions entre les mécanismes de sécurité de SIP et son fonctionnement. Cela a également permis d'établir les limites de la sécurité telle qu'elle est définie dans le RFC. Parmi les différentes propriétés apportées par ces mécanismes, c'est l'authentification qui a été plus particulièrement étudiée dans ce manuscrit. Cette fonction de sécurité occupe une place centrale dans les réseaux d'aujourd'hui, car elle permet de prouver son identité, voire de vérifier que son interlocuteur est bien celui qu'il prétend être. C'est généralement le premier mécanisme de sécurité auquel est confronté un usager ou un équipement dans un contexte de sécurité. Que ce soit pour un accès à des réseaux locaux ou étendus, que ces réseaux soient filaires ou sans fil, qu'ils soient en architecture client-serveur ou répartie, l'authentification des équipements ou des usagers est nécessaire pour vérifier que le service est fourni de manière légitime.

Cette protection est généralement demandée quand des données ou informations personnelles sont communiquées. La téléphonie est donc directement concernée par cette recommandation. L'intérêt de l'authentification va au delà du simple problème de divulgation d'informations en téléphonie sur IP. L'absence de vérification d'identité peut également entraîner des problèmes de déni de service ou des usurpations d'identité.

L'authentification la plus courante dans les réseaux consiste à associer un mot de passe à un identifiant, le fameux login/password. L'utilisation de SIP passe généralement par la

génération d'un profil basé sur un identifiant et un secret. Or, les procédures d'authentification classiques par identifiant et mot de passe ne suffisent plus. Sur les réseaux locaux comme sur Internet, l'écoute du trafic permet de récupérer dans certain cas facilement et pratiquement sans risque de détection les informations personnelles d'un utilisateur. Rien de plus simple ensuite pour l'attaquant que de se connecter à son tour, soit utilisant le mot de passe, soit en rejouant les mêmes valeurs et ainsi se faire passer pour un utilisateur autorisé. Il s'agit alors d'usurpation d'identité.

Pour résoudre ce problème ou éviter la multiplication des authentifications générant des mots de passe triviaux, d'autres méthodes ou procédures ont été développés. Il sera ainsi évoqué l'utilisation des mots de passe à usage unique (OTP, One Time Password) ou les architectures SSO (Single Sign On) qui permettent une seule authentification pour plusieurs services.

Les attaques montrent l'importance et les enjeux de pouvoir vérifier l'identité de l'émetteur d'un message. Le besoin primordial est que les usagers soient tous authentifiés auprès des serveurs pour éviter l'utilisation abusive des comptes SIP. D'autre part, il serait judicieux que les serveurs soient également authentifiés pour éviter la manipulation des sessions et le contrôle par un élément illégitime du réseau.

Le RFC de SIP prévoit un certain nombre de mécanismes pour permettre l'authentification des éléments. Le problème réside dans les multitudes de solutions. En effet, d'un domaine à l'autre, les choix de sécurité peuvent être très différents. Le tableau 8 résume les différents cas possibles. Notons que dans le cas de S/MIME, les serveurs pourraient authentifier l'usager dans la mesure où ils possèdent un annuaire de certificats.

Tableau 8. Synthèse des authentifications SIP référencées dans le RFC 3261

Méthode / Cas	HTTP Digest	S/MIME	TLS	IPSec
UA ⇒ Serveur	**OUI**	**~**	**~**	**~**
Serveur ⇒ UA	**~**	**NON**	**OUI**	**~**
UA ⇔ UA	**~**	**OUI**	**NON**	**NON**
Serveur ⇒ Serveur	**NON**	**NON**	**~**	**OUI**
Serveur ⇔Serveur	**NON**	**NON**	**~**	**OUI**

Légende du tableau 8 :

x ⇒ y : authentification simple, x s'authentifie auprès de y ;

⇔ : authentification mutuelle ;

OUI : possible ;

~ : possible mais pas usuel (faute de description ou soumis à condition) ;

NON : ne s'applique pas.

3.6. Conclusion

SIP spécifie les échanges d'informations pour la gestion de sessions multimédias et par extension ceux des appels en téléphonie sur IP. Ce protocole décrit particulièrement la signalisation qui permet au travers des messages SIP de pouvoir établir, modifier et terminer une communication vocale. Le fonctionnement de SIP prévoit donc l'intervention d'intermédiaires avec ou sans relations de confiance entre eux pour l'acheminement des appels ou une relation directe entre usagers. Ce modèle distribué entraîne donc une grande variété d'environnement rendant sa sécurisation délicate.

SIP propose un large panel de mécanismes pour sécuriser une infrastructure ToIP. Plutôt que de définir de nouveaux mécanismes, SIP fait appel aux mécanismes usuels de monde IP. HTTP Digest permet principalement l'authentification des usagers. TLS et S/MIME permettent de sécuriser la signalisation, SRTP chiffre la voix et IPSec protège à la fois la voix et la signalisation. Ces propositions ont un coût, en temps de calcul, en bande passante et ne sont pas toujours interopérables avec les implémentations existantes. Néanmoins la plupart de ces solutions garantissent une sécurité maîtrisée dans un domaine ou sur un lien mais difficilement sur la totalité d'une communication.

Concernant l'authentification, SIP hérite du niveau de protection des applications basées sur le modèle client/serveur. L'environnement étant de confiance, la délivrance d'un service était basée sur un simple challenge/réponse. Les risques liés à une authentification simple ont été exposés dans ce chapitre. Les solutions comme TLS ou S/MIME permettent de s'en prémunir mais nécessitent une gestion des certificats, ce qui n'est pas usuel à l'heure actuelle dans les infrastructures de ToIP comme cela déjà a été mentionné. C'est ainsi que SIP est généralement déployé avec HTTP Digest dans sa version authentification simple alors qu'une authentification mutuelle est dorénavant systématique dans les nouvelles architectures comme la 3G avec AKA [3GPP]. Les contributions présentées par la suite s'attacheront donc à renforcer l'authentification au niveau de la signalisation soit en diminuant les vulnérabilités de l'authentification simple, soit en envisageant une autre forme d'authentification mutuelle avec HTTP Digest.