Sécurité de la téléphonie sur IP

Thomas Guillet

Thèse

présentée pour obtenir le grade de docteur
de TELECOM ParisTech

Spécialité **: Informatique et Réseaux**

Thomas GUILLET

Sécurité de la téléphonie sur IP

Soutenue le 29 octobre 2010 devant le jury composé de :

Rapporteurs :
Professeur Bernard COUSIN Université de Rennes 1
Professeur Pascal LORENZ Institut Universitaire de Technologie de Colmar

Examinateurs :
Professeur Omar ABOU KHALED University of Applied Sciences of Western Switzerland
Capitaine de vaisseau Henri d'AGRAIN Etat-major de la marine
Professeur Elena MUGELLINI University of Applied Sciences of Western Switzerland
Docteur ingénieur Michel PELLET Direction Générale pour l'Armement
Professeur Pascal URIEN TELECOM ParisTech

Directeur de Thèse :
Professeur Ahmed SERHROUCHNI TELECOM ParisTech

A Caroline
A mes parents

« J'ai un journal de route tenu au jour le jour. Je me suis aperçu en le parcourant dernièrement qu'il n'était pas intéressant du tout, et ne donnait pas l'idée de ce que j'avais vu. C'est l'inconvénient de tout journal de route. Pour donc qu'il me reste de ce voyage des notes me rendant aussi exactement que possible, non pas les événements de tous les jours, mais la synthèse des divers états d'âme qui se sont succédé, j'ai entrepris ce travail qui s'éloigne complètement de la forme « Journal » et qui ne retiendra de ce long voyage que ce qui m'a paru essentiel et présentant un intérêt général.»

Ernest Psichari

Remerciements

En tout premier lieu je tiens à remercier très sincèrement mon directeur de thèse M. Ahmed Serhrouchni. Il a su tout au long de ces quatre dernières années accompagner mes recherches et mes réflexions par ses précieux conseils. Cette collaboration a donné naissance à une amitié qui m'est chère.

Je remercie Messieurs Bernard Cousin et Pascal Lorenz d'avoir accepté d'être les rapporteurs de cette thèse. Leur intérêt pour ce travail de recherche m'a particulièrement touché.

Mes remerciements s'adressent également à Monsieur Pascal Urien qui m'a fait l'honneur de présider mon jury de thèse.

Un grand merci à Madame Elena Mugellini et Monsieur Abou Khaled d'avoir été des lecteurs attentifs de cette thèse et d'avoir fait le chemin depuis la Suisse pour participer au jury.

Je remercie également Monsieur Michel Pellet et Monsieur le capitaine de vaisseau Henri d'Agrain de leur bienveillance et d'avoir pris de leur temps pour participer à l'évaluation des mes travaux de recherche.

Enfin je dédie ce manuscrit à mon épouse Caroline et à mes parents. Leur patience et leur soutien m'ont permis de mener à bien ce défi personnel.

Résumé

Ces travaux portent sur la sécurité de la téléphonie déployée dans les réseaux Internet. Ce service est sans aucun doute, après le Web et la messagerie, l'application qui imposera l'infrastructure IP (Internet Protocol) comme le standard de transport de tout type d'information ou de média. Cette migration de la téléphonie classique vers le tout IP semble être incontournable mais elle pose des problèmes en matière de sécurité. Si des attaques existaient déjà avec la téléphonie classique, l'usage d'un réseau IP les rend plus facilement réalisables. Notre analyse souligne les limites des solutions usuelles, principalement au travers des problèmes d'interopérabilité. De plus eu égard à l'hétérogénéité des infrastructures de ToIP, la protection de bout-en-bout des appels n'est pour le moment pas considérée, sauf par les services étatiques.

Dans un premier temps, nous avons cherché les possibilités de renforcer la sécurité de SIP (Session Initiation Protocol) de l'IETF, protocole actuellement massivement adopté dans les infrastructures de téléphonie sur IP. Nous avons proposé des solutions innovantes et validées pour consolider les mécanismes existants de manière complètement transparente pour les infrastructures. Nous avons choisi de nous focaliser sur l'authentification, car c'est le premier mécanisme rencontré par les usagers ou les systèmes. Les solutions présentées ci-après proposent de nouvelles propriétés de sécurité en définissant une sémantique pour des champs dit « opaques ». Ces contributions consolident la sécurité entre l'usager et son serveur.

Dans un second temps, nous nous sommes intéressés aux solutions permettant une sécurité de bout-en-bout des appels. L'analyse des solutions applicatives comme « Future Narrow Band Digital Terminal » et « Secure Voice over IP Simple Protocol » nous a permis de formaliser les spécifications d'une architecture permettant la protection des conversations quelque soient les spécificités et l'hétérogénéité des réseaux de ToIP. Cette approche utilise le flux d'informations voix pour mette en œuvre une signalisation de sécurité, ce qui rend cette solution complètement compatible avec les infrastructures existantes. Par ailleurs notre étude atteste de l'intérêt de mettre en place des entités de confiance dédiées à la sécurité des appels.

Enfin la conclusion reprend et positionne les différentes contributions relatives à ces travaux dans le contexte de la téléphonie sur IP. Notre volonté d'être interopérable avec les infrastructures sous-jacentes voire indépendantes peut être considérée comme un service à valeur ajoutée.

Abstract

This work focuses on the security of telephony networks deployed in the Internet. Without any doubt after the Web and messaging application, this service will impose IP (Internet Protocol) infrastructures like the standard for all types of information or media. This migration from traditional telephony to all-IP appears to be inevitable but it poses security problems. If the attacks existed with the traditional telephony, the use of an IP network makes them more feasible. Our analysis highlights the limits of usual solutions, mainly through interoperability issues. Moreover, given the heterogeneity of IP telephony infrastructure, protecting end-to-end calls is currently not considered, except by state services.

Initially, we look for opportunities to strengthen the security of SIP (Session Initiation Protocol) IETF protocol currently massively adopted in the telephony infrastructure. We propose and validate innovative solutions to strengthen existing mechanisms in a completely transparent way for infrastructures. We choose to focus on authentication, because it is the first mechanism encountered by users or systems. The solutions presented below propose new security properties by defining a semantic field called "opaque". These contributions strengthen the security between the user and the server.

In a second step, we are interested in security solutions for end-to-end calls. Analysis of application solutions as “Future Narrow Band Digital Terminal” and “Simple Secure Voice over IP protocol“ allows us to formalize the specification of an architecture to protect conversations whatever the specificities and heterogeneity of networks ToIP are. This approach uses the media channel to implement a security warning, which makes this solution completely compatible with existing infrastructure. Furthermore, our study demonstrates the interest of establishing trusted entities dedicated to call security.

Finally, the conclusion takes over and takes positions on the various contributions to this work in the context of IP telephony. Our will to be interoperable with the underlying or independent infrastructure can be considered as a value added service.

Table des matières

Remerciements **4**

Résumé **5**

Abstract **6**

Table des matières **7**

Liste des figures **9**

Liste des tableaux **11**

1. Problématique **13**

1.1. Intégration de la téléphonie dans les systèmes d'information 13
1.2. Les menaces et les risques de la téléphonie sur IP 14
1.3. Etat et limites de la sécurité de la téléphonie sur IP 15
1.4. Une contribution pour la consolidation de la sécurité de la téléphonie sur IP 16
1.5. L'organisation de ce mémoire 16

2. Analyse des problèmes et des solutions de sécurité de la téléphonie sur IP **19**

2.1. Cadre de la téléphonie sur IP : concepts, architectures et protocoles 19
2.2. Les risques et typologie des attaques 22
2.3. Les différentes solutions 23
2.4. La typologie des méthodes pour sécuriser la téléphonie sur IP 31
2.5. Méthodes et outils d'évaluation de la sécurité de la téléphonie sur IP 32
2.6. Conclusion 33

3. Evaluation des solutions de sécurité du protocole SIP **35**

3.1. La place du protocole SIP dans la téléphonie sur IP 35
3.2. Architecture et protocoles dans un environnement SIP 36
3.3. Typologie des attaques 42
3.4. SIP et la sécurité 46
3.5. Analyse des solutions actuelles 60
3.6. Conclusion 62

4. Définition et conception de solutions de sécurité pour SIP **63**

4.1. Analyse des solutions d'authentification de SIP 63
4.2. Un complément au protocole d'authentification HTTP Digest SIP 74
4.3. Une authentification optimisée par les mots de passe à usage unique 76
4.4. Une solution pour le déni de service 79
4.5. Conclusion 81

5. Validation des solutions de sécurité du protocole SIP **83**

5.1. Méthodes et outils de validation 83
5.2. La validation formelle de l'authentification HTTP Digest SIP 87
5.3. Les validations sur plate-forme logicielle 90
5.4. Résultats et analyse 90
5.5. Conclusion 96

6. Définition d'une architecture de téléphonie sur IP sécurisée 97
6.1. Sécuriser les appels de bout-en-bout 97
6.2. Sécurité de bout en bout basée sur une infrastructure hétérogène 98
6.3. Spécifications d'une architecture de téléphonie sur IP sécurisée 105
6.4. Conclusion 110
7. Conclusion générale et perspectives 111
7.1. Conclusion générale 111
7.2. Perspectives d'évolutions et futurs travaux 112
Liste des acronymes 113
Bibliographie 115
Publications associées à ces travaux 122
Annexe I : Codes pour la validation de HTTP Digest SIP renforcée 123
Annexe II : Code en C d'HOTP 124
Annexe III : Descriptions en HLPSL réalisées pour cette thèse 125
Annexe IV : Brevet issu de l'analyse de l'authentification HTTP Digest SIP 127
Annexe V : Impact de la sécurité du flux de voix sur les performances réseaux .. 130
Annexe VI : Fichiers de configuration d'Asterisk pour les validations 131
Annexe VII : Interface graphique des softphones et des outils de validation 132

Liste des figures

Figure 1. Architecture générique de la téléphonie sur IP 20
Figure 2. Principe de la méthode EBIOS 24
Figure 3. Architecture ToIP au sein d'une enclave type entreprise 27
Figure 4. L'environnement de KIF [ABD08] 33
Figure 5. La pile protocolaire de SIP 36
Figure 6. Architecture élémentaire SIP 37
Figure 7. Exemple de message SIP 38
Figure 8. Echange de messages SIP pour l'établissement d'une session 41
Figure 9. Etablissement d'un appel dans une infrastructure SIP 42
Figure 10. Attaque par le BYE 44
Figure 11. Attaque par le CANCEL 44
Figure 12. Attaque sur le REGISTER 45
Figure 13. Pile protocolaire SIP avec les éléments de sécurité 46
Figure 14. Enregistrement et authentification HTTP Digest dans un contexte SIP 47
Figure 15. Authentification HTTP Digest SIP pour un message REGISTER 48
Figure 16. Exemple de message « 401 Unauthorized » SIP 48
Figure 17. Exemple de message REGISTER du protocole SIP avec le champ « response » 49
Figure 18. Interface graphique de Lynxphone pour la gestion des clés (publique et privées) 51
Figure 19. Message SIP avec la solution de sécurité S/MIME 52
Figure 20. Echanges de messages TLS 54
Figure 21. Exemple de certificat d'un serveur SIP 55
Figure 22. Exemple d'échanges TLS pour SIP 55
Figure 23. IPSec en mode AH 56
Figure 24. IPSec en mode ESP 57
Figure 25. SRTP dans un contexte SIP 58
Figure 26. Cheminement de la signalisation SIP dans les réseaux IP 59
Figure 27. Emploi du champ Integrity_Auth dans un message BYE 65
Figure 28. Formule du champ Integrity_Auth proposé par [GEN08] 65
Figure 29. Exemple de message avec le mécanisme AIB [RFC3893] 66
Figure 30. Authentification dans le domaine de confiance de [SRI05] 67
Figure 31. Authentification AKA appliquée à SIP [RFC3310] 69
Figure 32. Utilisation d'un mot de passe à usage unique dans SIP via un téléphone portable. 70
Figure 33. Architecture SIP avec un serveur RADIUS 71
Figure 34. Application de l'authentification SSO dans le contexte SIP 72
Figure 35. Exemple de message « 401 Unauthorized » SIP 74
Figure 36. Exemple de message REGISTER du protocole SIP 74
Figure 37. Principe du renforcement de l'authentification HTTP Digest SIP 75
Figure 38. Processus de génération d'un mot de passe à usage unique 76
Figure 39. Authentification OTP asynchrone 77
Figure 40. Authentification OTP asynchrone. 77
Figure 41. Fonction HMAC 78
Figure 42. Utilisation d'HOTP dans le calcul du challenge/réponse HTTP Digest 78
Figure 43. Intégration dans la signalisation d'HOTP dans l'authentification SIP 79
Figure 44. Authentification HOTP intégrée dans le Call-Id de SIP 79
Figure 45. Champ « Branch » avec sémantique 80
Figure 46. Solution au Dos par BYE frauduleux 80
Figure 47. Architecture de fonctionnement d'AVISPA 84

Figure 48. Plate-forme de validation....85
Figure 49. Schéma de principe d'ASTERISK....86
Figure 50. Authentification HTTP Digest SIP visualisée avec SPAN....87
Figure 51. Résultats d'une analyse avec AVISPA....88
Figure 52. Attaque élaborée sur HTTP Digest SIP construite avec AVISPA....88
Figure 53. Authentification HTTP renforcée visualisée avec SPAN....89
Figure 54. Résultats de l'analyse de notre contribution avec AVISPA....89
Figure 55. Twinkle modifié vérifiant la légitimité du serveur SIP....91
Figure 56. Intégration dans la signalisation d'HOTP dans l'authentification SIP....92
Figure 57. Message SIP contenant une authentification HOTP dans le Call-ID....93
Figure 58. Mesures de temps dans l'établissement d'une session....94
Figure 59. BYE intégrant un « branch » calculé selon notre processus....95
Figure 60. Application FNBDT dans l'infrastructure de téléphonie sur IP....98
Figure 61. Architecture protocolaire FNBDT....99
Figure 62. Etablissement d'un appel FNBDT....100
Figure 63. Architecture matérielle de SVSP....101
Figure 64. Scenario d'appel SVSP [BAS05]....103
Figure 65. Echanges au niveau applicatif....106
Figure 66. Etablissement d'un appel sécurisé....107
Figure 67. Architecture protocolaire générique d'une solution de ToIP....108
Figure 68. Pile protocolaire de la signalisation de sécurité....109
Figure 69. Principe générique d'une authentification simple....128
Figure 70. Principe de l'authentification mutuelle objet du brevet....129
Figure 71. Dispositif pour la mesure de trafic voix....130
Figure 72. Interfaces graphiques de 3CX et de Twinkle....132
Figure 73. Interface graphique de Wireshark....132
Figure 74. Interface graphique de CommView....133
Figure 75. Interface graphique du logiciel SPAN....133
Figure 76. Interface graphique de SIPNess....134

Liste des tableaux

Tableau 1. Les principaux risques de la ToIP 22
Tableau 2. Typologie des attaques 23
Tableau 3. Liste des requêtes SIP 39
Tableau 4. Les principales familles des réponses 39
Tableau 5. Les principaux champs d'en-tête des messages SIP 40
Tableau 6. Typologie des attaques dans un environnement SIP 43
Tableau 7. Entropie d'un mot de passe [AUT07] 49
Tableau 8. Synthèse des authentifications SIP référencées dans le RFC 3261 61
Tableau 9. Description des notations de [SRI05] 68
Tableau 10. Résultat des expérimentations de l'authentification HTTP Digest renforcée 91
Tableau 11. Temps de calcul cryptographique pour HTTP Digest renforcé 92
Tableau 12. Synthèse des résultats concernant l'authentification HOTP 93
Tableau 13. Temps de calcul cryptographique pour l'authentification HOTP 94
Tableau 14. Synthèse des résultats concernant la validation de la solution au DoS 95
Tableau 15. Tableau de synthèse des validations pratiques 96
Tableau 16. Comparaison des solutions de sécurité bout-en-bout pour la ToIP 104
Tableau 17. Qualité d'écoute en fonction du délai de transmission [DEO07] 110

CHAPITRE 1
PROBLEMATIQUE

Ce chapitre définit la problématique de sécurité liée à la migration de la téléphonie vers le réseau Internet. En effet ce dernier s'affirme de plus en plus comme le réseau de transport de tout type de média. Et comme le « Web » ou la messagerie, la téléphonie sur Internet n'échappe pas aux problèmes de sécurité : les écoutes illicites, le déni de service, l'usurpation d'identité, l'usurpation de droits, le détournement d'appel voire même le SPAM téléphonique. Certes, ces menaces existaient déjà avec la téléphonie classique, mais l'usage des réseaux Internet les rend plus facilement réalisables. Les solutions de sécurité existent mais elles sont propres à chaque infrastructure. Elles ne tiennent donc pas compte de l'hétérogénéité des protocoles et des implémentations. Ce manuscrit propose de définir des solutions interopérables pour renforcer la sécurité de la téléphonie sur IP. Deux sujets ont été particulièrement traités : l'authentification dans les architectures basées sur le protocole Session Initiation Protocol (SIP) et la sécurité de bout-en-bout des appels.

1. Problématique

1.1. Intégration de la téléphonie dans les systèmes d'information

La téléphonie sur IP[1] (ToIP) est sans aucun doute l'application – après le Web et la messagerie – qui imposera l'infrastructure IP (Internet Protocol) comme le standard de transport de tout type d'information ou de média. Elle a de nombreux avantages comme la mobilité, la réduction des coûts et l'intégration de nouveaux services.

L'adoption d'une infrastructure unique pour le transport de tout type de données représente un avantage économique très significatif pour les opérateurs, les entreprises et les particuliers. Actuellement, les opérateurs ou les entreprises gèrent deux réseaux : le réseau orienté IP dédié aux systèmes d'information et le réseau dédié à la téléphonie. Cette situation génère une multitude de coûts, tant en immobilisation de matériel qu'en ressources humaines. Concernant les usagers, certains possèdent un abonnement dit « Réseau de Téléphonie Commuté » RTC et une ou plusieurs solutions à base de technologie IP. La fusion des solutions a principalement comme objectif la réduction du coût des infrastructures, des abonnements et donc des communications. Ainsi, la migration des services de téléphonie classique vers le tout IP semble être incontournable.

[1] Téléphonie sur IP : ce terme se traduit en anglais par « Telephony over IP ». C'est pour cette raison que l'acronyme est ToIP.

A l'aspect financier, il faut ajouter les perspectives d'intégration d'autres services. Les solutions de la téléphonie sur IP sont d'autant plus tentantes qu'elles intègrent toutes les fonctions classiques de télécommunications téléphoniques : répondeur, fax, photocopieur, audioconférence voire visioconférence. Pour les entreprises, le Couplage Téléphonie Informatique (CTI) offre de nouveaux services : pour la relation clients, un numéro de téléphone peut être associé à son dossier (fiche client, état de la commande, etc.). De nombreuses sociétés ou organismes ont d'ores er déjà adopté la ToIP comme Renault, le Crédit Agricole, le CHU de Clermont-Ferrand, le rectorat d'Orléans, etc. Les architectures de ToIP sont déjà une réalité.

Ce ralliement n'est pas seulement un choix économique mais aussi technologique. La téléphonie est désormais établie comme un service informatique. Cette évolution technique s'est traduite par l'intégration de la signalisation d'appel et le développement de mécanismes de transport de la voix dans les architectures réseaux IP. La téléphonie sur IP est à la frontière de deux mondes : celui des télécommunications, qui a inventé le service de communication à distance, et celui de l'Internet qui, après avoir révolutionné le réseau de transport de l'information, souhaite s'approprier ce service.

Deux principaux problèmes se posent dans cette situation [OUA08] :

- tout d'abord la qualité de service, qui consiste à attribuer des ressources pour le bon fonctionnement du service. Ce sujet n'est pas l'objet de ce travail de recherche ;
- ensuite la sécurité. Les réseaux IP sont largement connus pour leurs vulnérabilités. A cela, il faut ajouter celles propres à la téléphonie. Nos travaux ont donc recherché les points où le renforcement de la sécurité de la ToIP est essentiel.

1.2. Les menaces et les risques de la téléphonie sur IP

Plusieurs travaux ont décrit les menaces et les risques auxquels sont exposées les infrastructures de téléphonie sur IP [KUH05] [GUP07] [ABD08] [DUB07] : les écoutes illicites, le déni de service, l'usurpation d'identité, l'usurpation de droits, le détournement d'appel voire même le SPAM téléphonique. Ces attaques sont déjà présentes dans la téléphonie classique, mais l'usage d'un réseau IP les rend plus facilement réalisables (accès distant et large diffusion des outils d'attaques) et moins coûteuses. Initialement, l'univers de la téléphonie et des télécommunications était relativement épargné par les menaces habituellement rencontrées par les systèmes d'informations. La téléphonie dite « classique » offre des garanties de sécurité qui sont principalement dues à son infrastructure dédiée. Mais la continuité des services et le besoin d'accès au web nécessite d'ouvrir les accès vers d'autres réseaux privés ou publiques (opérateur, Internet), impactant par là même la sécurité de la ToIP.

Les propriétés de sécurité sont généralement liées à une infrastructure ou à une application : le paiement en ligne, les réseaux informatiques, Internet, la téléphonie cellulaire (UMTS, GSM, etc.), les réseaux sans fil (WiFi, Bluetooth, etc.). Elles sont généralement conçues comme étant une propriété intrinsèque de chaque système. La ToIP est quant à elle un assemblage de plusieurs de ces systèmes. Ces derniers

interopèrent globalement entre eux pour assurer l'établissement d'appel. Cependant la continuité de service ne concerne que la signalisation de base et l'acheminement de la voix. La sécurité ne faisant pas l'objet de spécifications aux interconnexions, la ToIP cumule les vulnérabilités propres à la téléphonie et celles des réseaux IP.

1.3. Etat et limites de la sécurité de la téléphonie sur IP

Plusieurs efforts ont déjà été consentis pour définir des solutions de sécurité pour la ToIP. Des recommandations [KUH05] ont été établies pour la conception des architectures et l'implémentation des protocoles. Les mécanismes usuels du monde de l'Internet ont ainsi été mis à contribution :

- IPSec [RFC2401] pour la protection de la signalisation et de la voix entre des sites distants ;
- TLS [RFC2246] pour sécuriser les échanges entre un usager et son serveur, évidemment quand TCP [RFC793] est mis en œuvre pour transporter la signalisation ou la voix ;
- SRTP [RFC3711] pour garantir la confidentialité de la communication.

Comme expliqué ci-dessus, les interconnexions entre domaines n'assurent pas la continuité des propriétés de sécurité. Les solutions citées précédemment ne protègent que partiellement un appel téléphonique. Par ailleurs, la présence de certificats n'est pas généralisé dans les postes téléphoniques ce qui limite de facto l'emploi de solutions comme TLS ou S/MIME [RFC3851]. De plus ces solutions ne sont pas forcément adaptées aux contraintes de la téléphonie sur IP [DIA07]. Les limitations sont nombreuses, et c'est vraisemblablement la raison pour laquelle un standard de sécurité pour la ToIP n'a pas émergé.

Devant ce constat, déjà avéré dans les années 80, la NSA[2] avait envisagé une solution de sécurité « universelle » appelée « Future Narrowband Digital Terminal » (FNBDT) [FNBDT], permettant une sécurité de bout-en-bout des appels téléphoniques. Après établissement de l'appel, le système met en œuvre une signalisation de sécurité dans le canal audio. Depuis, le standard a évolué pour prendre en compte la technologie IP avec la version appelée « Secure Communications Interoperability Protocol » SCIP. Il reste que cette approche, usuelle pour les militaires, reste marginale pour les particuliers. Des produits non destinés à des usages gouvernementaux émergent progressivement comme TopSec de Rhodes&Schwarz [ROH].

[2] NSA : la National Security Agency est un organisme gouvernemental des États-Unis, responsable de la collecte et de l'analyse de toutes formes d'informations.

Le constat est donc le suivant :

- des solutions pour sécuriser la téléphonie sur IP existent mais ne sont pas forcément adaptées à ce service. La sécurité peut donc être garantie dans un domaine bien circonscrit (physiquement et/ou logiquement). Par ailleurs, les solutions de la ToIP ne sont pas exportables dans d'autres environnements comme la téléphonie commutée ;
- un principe de sécurisation « universel » existe en utilisant le flux d'informations voix pour la mise en œuvre d'une signalisation de sécurité. Bien que ce principe ait été éprouvé avec le protocole FNBDT, cette solution n'a pas été intégrée massivement aux équipements du monde civil.

1.4. Une contribution pour la consolidation de la sécurité de la téléphonie sur IP

Motivé par la nécessité de sécuriser la téléphonie sur IP, notre travail propose des solutions complètement interopérables avec les architectures déjà déployées. Les principales contributions de ce manuscrit sont les suivantes :

- une analyse des mécanismes de sécurité existants pour définir une typologie des solutions et ainsi justifier notre approche ;
- un renforcement de la sécurité, en particulier l'authentification SIP, sans modifier les échanges et en garantissant une totale interopérabilité avec les infrastructures existantes tout en minimisant les impacts sur les performances du service. Nous avons proposé des solutions innovantes et validées en spécifiant une sémantique pour des champs contenant normalement des valeurs aléatoires ;
- la définition et la spécification d'une architecture pour une sécurité de bout-en-bout de la téléphonie sur IP. Cette architecture se distingue par une mise en œuvre d'une signalisation sur le canal media. Cette approche permet d'être indépendante des infrastructures de ToIP sous-jacentes.

1.5. L'organisation de ce mémoire

Ces travaux sont organisés en 7 parties. La première partie (chapitre 1) est l'introduction de ce manuscrit. Elle permet au lecteur de saisir les buts et les problèmes de la téléphonie sur IP dans les systèmes d'information, et de comprendre quels sont les enjeux pour la sécurité. Cette pénétration de la téléphonie dans le monde IP pose ainsi un problème complexe en terme d'interopérabilité compte de tenu de l'hétérogénéité des architectures.

La deuxième partie (chapitre 2) analyse les principaux problèmes de sécurité de la téléphonie sur IP et présente comment les architectures actuelles permettent de s'en prémunir. Au travers de la typologie des méthodes proposée, cette étude montre que compte tenu de la nature hétérogène des architectures de ToIP l'endroit le plus opportun

pour apporter des mécanismes de sécurité est la couche applicative : soit dans la signalisation, soit dans le canal média. Ce choix permet de faire abstraction des propriétés du réseau de transport et d'offrir aux usagers des services de sécurité indépendant de l'infrastructure sous-jacente.

La troisième partie (chapitre 3) est consacrée au protocole Session Initiation Protocol SIP et aux mécanismes de sécurité associés. SIP est un des protocoles de signalisation les plus populaires de la ToIP. L'analyse montre les limites de la sécurité telle qu'elle est envisagée dans SIP et souligne en particulier les menaces liées aux solutions d'authentification.

La quatrième partie (chapitre 4) présente les contributions à la sécurité dans les architectures SIP. A partir de l'analyse de la troisième partie, il a été défini un cahier des charges pour faciliter l'émergence de nouvelles solutions de sécurité. Ces travaux ont principalement contribué à améliorer l'authentification. Après avoir analysé les autres propositions faite par la communauté scientifique, il a été recherché toutes les opportunités d'améliorer l'authentification sans modifier la signalisation mais en proposant de nouvelles sémantiques pour des champs déjà existants. Les faiblesses des algorithmes cryptographiques ne sont pas traitées

La cinquième partie (chapitre 5) présente les validations des contributions. Les expérimentations réalisées sur plates-formes logicielles s'appuient sur des produits « opensource » modifiés pour implémenter les nouveaux mécanismes de sécurité. Une des propositions fait également l'objet d'une validation formelle de sécurité avec le logiciel AVISPA. Ces validations confirment la totale compatibilité de nos contributions avec l'existant, ce qui permet le cas échéant un déploiement progressif et transparent pour les usagers.

La sixième partie (chapitre 6) propose la définition d'une architecture de ToIP sécurisée. L'analyse menée dans ce manuscrit nous a permis de formaliser les propriétés d'une solution de sécurité bout-en-bout pour les appels IP. Notre solution s'appuie sur une infrastructure dédiée à la sécurisation des appels. Ce modèle, complètement indépendant des infrastructures sous-jacentes de ToIP, illustre l'intérêt d'un tiers de confiance pour cette application et promeut une protection adapté au contexte de chaque appel. La déclinaison protocolaire n'est pas traitée, comme le mode de déploiement.

Dans la conclusion, nous reprenons et positionnons les différentes contributions relatives à ces travaux. Notre volonté d'être interopérable avec les infrastructures de ToIP voire indépendant peut être considéré comme un service à valeur ajoutée. Il est également nécessaire de considérer la qualité de service pour toute solution de sécurité, d'autant que la téléphonie est une application synchrone avec des contraintes temporelles. La Qos (Quality of Service) a été préservée voire améliorée dans une de nos contributions.

A la fin de ce manuscrit, on trouvera également une bibliographie, la liste des acronymes, la liste des publications associées à ces travaux ainsi que des annexes techniques.

CHAPITRE 2

ANALYSE DES PROBLEMES ET DES SOLUTIONS DE SECURITE DE LA TELEPHONIE SUR IP

Ce chapitre a pour objectif d'analyser les principaux problèmes de sécurité de la téléphonie sur IP et les solutions proposées. Au travers des principales menaces rencontrées, nous présentons quels sont les techniques existantes pour s'en prémunir. L'analyse met en évidence les problèmes d'interopérabilité dus à l'hétérogénéité des réseaux et des protocoles de ToIP. Chaque système ayant ses propres mécanismes de sécurité, la sécurité de bout-en-bout des appels n'est actuellement pas considérée, sauf pour des usagers gouvernementaux. Seule une solution basée sur le canal média semble répondre à cette problématique.

2. Analyse des problèmes et des solutions de sécurité de la téléphonie sur IP

2.1. Cadre de la téléphonie sur IP : concepts, architectures et protocoles

2.1.1. *Le périmètre de l'analyse*

La téléphonie a connu ces dernières années une véritable révolution avec l'émergence de la téléphonie sur IP, qui apporte néanmoins certains inconvénients comme la problématique de la sécurité. La téléphonie sur IP cumule les vulnérabilités de la téléphonie classique et celles des réseaux informatiques. En déployant ou en adoptant une solution de ToIP, les entreprises et les particuliers exposent leurs systèmes à de nouvelles menaces. Le téléphone structurant très fortement notre manière d'échanger, la ToIP est une ouverture sur les données personnelles ou professionnelles. Par exemple, l'accès frauduleux à l'information au travers du téléphone est déjà une réalité pour les mafias qui ont organisé le marché noir d'informations issues des écoutes téléphoniques en Italie et en Grèce [CHA08].

Parallèlement, la ToIP correspond à une véritable transformation des télécommunications. La ligne téléphonique est dématérialisée, le service est dorénavant lié à un compte usager (caractérisé par identifiant et un mot de passe) auquel on associe des services comme la téléphonie, la visioconférence ou encore la messagerie. Tout cela est indépendamment du réseau d'accès ou de transport, de la localisation ou du type de terminal. L'utilisation d'un compte de ToIP sur un ordinateur d'hôtel pendant ses vacances permet certes de téléphoner à moindre coût, mais pose la question du stockage des données personnelles sur un équipement non maitrisé. La sécurité doit évidemment prendre en compte cette notion de mobilité.

Ainsi, la sécurité de la téléphonie constitue un domaine d'étude vaste. Ce chapitre va donc définir le périmètre de la ToIP et balayer l'ensemble des solutions existantes. Cette description a pour but de justifier la nature des solutions proposées par ce manuscrit.

Pour définir le périmètre de l'analyse, et par là même savoir où apporter les solutions de sécurité, il convient de définir les biens, les acteurs et les problèmes de la ToIP. La modélisation de la ToIP fournie par la figure 1 permet d'identifier les biens à protéger vis-à-vis des risques qui seront précisés ultérieurement. Cette vision abstraite caractérise deux grands domaines d'étude : le réseau et les échanges propres à la téléphonie. Deux sujets vont être développés dans ce chapitre :

- la sécurisation de la téléphonie au travers de l'architecture du réseau ;
- la sécurisation de la téléphonie au travers des choix protocolaires.

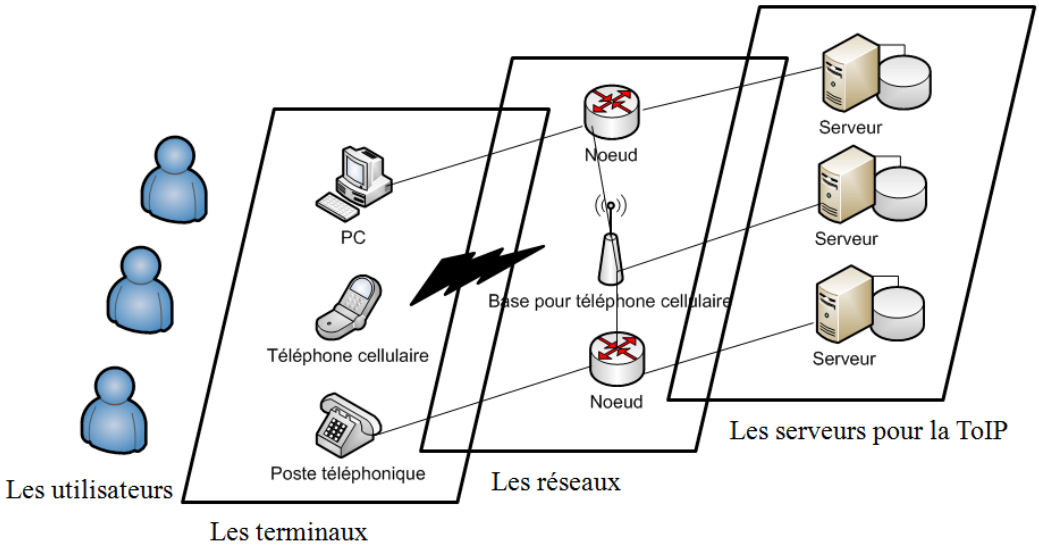


Figure 1. Architecture générique de la téléphonie sur IP

2.1.2. *Les éléments caractérisant une architecture de téléphonie sur IP*

Les infrastructures de téléphonie sur IP sont composées d'un ensemble d'équipements utilisant les technologies de l'Internet et de systèmes informatiques. Les usagers exploitent des terminaux interagissant avec différents types de serveur afin de gérer les comptes, la mobilité, la localisation et évidemment l'établissement de l'appel. Pour réaliser une telle infrastructure de téléphonie, de nombreux éléments matériels ou logiciels sont nécessaires comme :

- les autocommutateurs, qui permettent la création d'une liaison temporaire entre deux équipements de communication. Utilisés par exemple pour l'établissement d'une communication téléphonique, ces équipements sont appelés les IPBX[3] ;

[3] IPBX : Un IPBX est un PABX IP. Le terme PABX désigne un équipement qui permet de relier les postes téléphoniques d'un établissement (lignes internes) avec le réseau téléphonique public. Il permet en plus la mise en œuvre des services.

- les serveurs réseau comme : les serveurs DNS[4] ou encore les serveurs DHCP[5] ;
- les téléphones ou terminaux IP : équipement dédié à la téléphonie (hardphone) ou hébergeant un logiciel[6] de téléphonie (softphone) ;
- les routeurs : ce sont les équipements qui permettent l'aiguillage des paquets IP ;
- les cartes de commutations : elles jouent le rôle de passerelle avec les réseaux publics (RTC, GSM,…).

Le logiciel[7] est l'élément intelligent des équipements de ToIP. Il se retrouve dans les autocommutateurs (i.e. Asterisk [AST], 3CX [3CX]) et les téléphones (i.e. Twinkle [TWI], 3CX Phone [3CX]). Il permet à un équipement informatique de réaliser une fonction ou un service du monde des télécommunications. Il prend également en compte les spécificités de l'équipement d'accueil.

Les solutions issues des choix d'architecture du réseau seront évoquées dans ce mémoire mais ne feront pas l'objet d'une étude exhaustive. L'idée de notre étude n'est pas de lier la sécurité à l'infrastructure de téléphonie car, par définition, les appels doivent mettre en relation des personnes qui n'appartiennent pas forcément au même domaine ou qui ne sont pas reliés au même réseau local.

2.1.3. *Les protocoles de la téléphonie sur IP*

Un protocole est une formalisation permettant la communication entre plusieurs processus. En l'occurrence, un protocole de téléphonie doit permettre l'établissement d'une communication audio entre au moins deux personnes. Dans le monde de la ToIP, il existe une multitude de protocoles. SIP [RFC3261] est actuellement le plus populaire mais il faut également citer H323 [H323], SCCP (Skinny Client Control Protocol de Cisco) ou encore Skype [SKYPE].

Les protocoles de la ToIP se caractérisent par deux types de données :

- la signalisation qui décrit l'ensemble des échanges pour la gestion de l'appel, l'accès aux services et la négociation des paramètres pour le transport de la voix ;
- le flux vocal.

Les solutions de ToIP interopèrent globalement entre eux pour la gestion des appels. La continuité d'appel concerne principalement les fonctions de base de la signalisation et l'acheminement de la voix.

---

[4] DNS : Le Domain Name System (DNS) est un service permettant d'établir une correspondance entre une adresse IP et un nom de domaine.

[5] DHCP : Dynamic Host Configuration Protocol est un protocole dont le rôle est d'assurer la configuration automatique des paramètres IP d'un terminal, notamment en lui assignant automatiquement une adresse IP.

[6] Téléphone logiciel : cet équipement est aussi appelé softphone. Un téléphone logiciel permet à un ordinateur muni d'un microphone et d'enceintes ou d'un casque audio d'établir un appel avec un autre téléphone en utilisant les services d'un serveur dédié à la téléphonie.

[7] Logiciel : un logiciel ou application est un ensemble de programmes qui permet à un système informatique d'assurer une tâche particulière.

2.2. Les risques et typologie des attaques

L'arrivée de la ToIP constitue de nouvelles opportunités d'attaques dans le monde des systèmes d'informations. La signalisation et la voix partageant le même réseau ou au moins les mêmes technologies que les réseaux de données IP, la téléphonie partage les mêmes vulnérabilités que les réseaux de données. A cela il faut rajouter les risques propres à la signalisation de la ToIP et au transport de la voix. Le tableau 1 fournit un premier niveau de description en précisant les différents risques liés à la téléphonie sur IP.

Tableau 1. Les principaux risques de la ToIP

Risques	Méthodes	Cibles
Déni de service Dos	Attaque entraînant l'indisponibilité d'un service/système pour les utilisateurs légitimes.	Un usager Un opérateur
Ecoute clandestine	Attaque permettant d'écouter l'ensemble du trafic de signalisation et/ou de la voix. Le trafic écouté n'est pas modifié.	Un usager
Détournement de trafic	Attaque permettant de détourner le trafic au profit de l'attaquant. Le détournement peut consister à rediriger un appel vers une personne illégitime ou à inclure une personne illégitime dans la conversation.	Un usager Un opérateur
Usurpation d'identité	Attaque basées sur la manipulation d'identité.	Un usager Un opérateur
Vols de services	Attaque permettant d'utiliser un service sans avoir à rémunérer son fournisseur.	Un usager Un opérateur
Communications indésirées SPIT[8] (SPAM téléphonique)	Attaque permettant à une personne de produire massivement des appels.	Un usager

L'autre manière de définir la menace est de caractériser les attaques. Ces dernières permettent à un élément menaçant d'exploiter une vulnérabilité. En synthétisant les travaux de [ZAR05], les attaques de la ToIP peuvent se ranger en trois grandes familles explicitées dans le tableau 2.

[8] SPIT : Spam over Internet Telephony. Tout comme un spam classique par courrier électronique, le SPIT peut être généré de manière similaire à partir de serveurs visant des millions d'utilisateurs de la ToIP. Le SPIT peut ralentir notablement le fonctionnement des architectures de téléphonie sur IP (exemple en engorgeant les boîtes vocales des usagers).

Tableau 2. Typologie des attaques

Type d'attaque	Principe du mode opératoire
Interception et modification	- Collecte d'informations sur les communications ; - Collecte d'informations sur les utilisateurs et le réseau ; - Manipulation du contenu des communications ; - Détournement des communications ; - Écoute des communications (conversation, message, vidéo).
Fraude et abus de service	- Usurpation d'identité ; - Contournement, porte dérobée (back door) ; - Manipulation des données de facturation.
Interruption de service ou déni de service	- Coupure physique ; - Épuisement des ressources ; - Déni de service général ; - Perte de courant.

Un certain nombre de contributions décrivent et analysent les attaques en ToIP [CHE09] [GUP07] [SNO]. Les réponses à ces risques passent par des solutions de sécurité, dont les propriétés seront définies ci-après.

2.3. Les différentes solutions

2.3.1. *Rappels des propriétés de sécurité*

Avant de présenter les solutions de sécurité, il convient de rappeler les définitions des propriétés de sécurité :

- **l'authentification** : garantir l'identité de l'usager qui envoie le message ;
    - dans le cadre de la ToIP, cette propriété permet par exemple à un serveur de vérifier qu'il fournit le service à l'usager légitime ;
- **la confidentialité** : rendre la conversation compréhensible aux personnes concernées uniquement ;
    - dans le cadre de la ToIP, cette propriété nécessite de chiffrer le flux audio ;
- **l'intégrité** : s'assurer que les données n'ont pas été modifiées entre l'envoi d'un message et sa réception ;
    - dans le cadre de la ToIP, cette propriété permet de s'assurer que les paramètres d'un appel n'ont pas été modifiés par une tierce partie ;

- **la non répudiation de l'appel** : la non répudiation des données nécessite l'archivage des données échangées ;
    - dans le cadre de la ToIP, cette propriété permet d'associer une communication à une personne de manière certaine ;
- **le non rejeu** : éviter de mémoriser puis de re-injecter les données dans le réseau ;
    - dans le cadre de la ToIP, cette propriété permet de ne pas pouvoir rejouer des échanges protocolaires par une personne tierce souhaitant accéder au service ;
- **l'anonymat** : capacité du système à masquer l'identité de l'usager ;
    - dans le cadre de la ToIP, cette propriété peut se traduire par le masquage de l'identité de l'appelant.

Ces propriétés de sécurité permettent de décrire les objectifs qu'il faut fixer pour protéger son système, compte tenu de ses besoins et des menaces. La formalisation de cette démarche s'appelle une analyse Sécurité des Systèmes d'Information (SSI). Il existe plusieurs méthodes pour définir le besoin de sécurité des systèmes d'informations comme EBIOS [EBIOS] (Expression des Besoins et Identification des Objectifs de Sécurité, cf. figure 2) ou MEHARI [MEH] (MEthode Harmonisée d'Analyse de Risques). L'objectif de toutes ces méthodes est de répondre au mieux à l'impératif de sécurité tout en prenant en compte le contexte et le besoin des utilisateurs. Le déroulement complet de l'analyse n'est pas dans le spectre de ce travail, en particulier l'analyse du besoin qui dépend que chaque utilisateur. Cependant, il est évident qu'une banque a besoin de plus de confidentialité qu'un centre d'appel, ce dernier demande quant à lui une forte disponibilité.

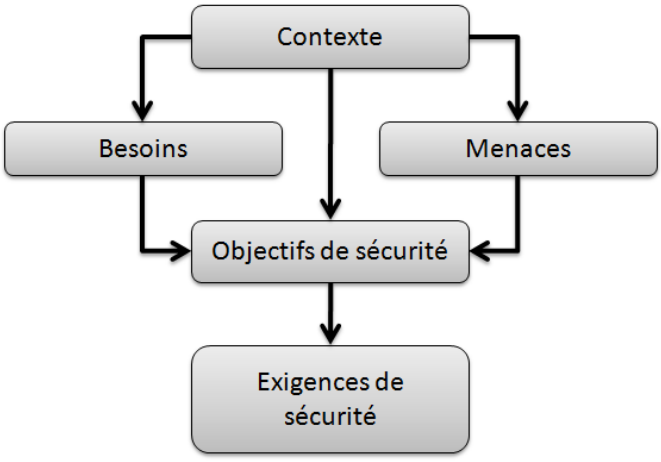


Figure 2. Principe de la méthode EBIOS

Les grandes entreprises utilisent en général ces méthodes pour la conception de leur système d'informations. Pour les particuliers ou les petites structures, il n'existe pas encore de réponse standard aux problèmes de sécurité de la téléphonie. Il semble cependant souhaitable de définir un besoin standard des particuliers pour faciliter l'adoption d'une solution unique de sécurité.

2.3.2. *Les principes de la sécurité de la téléphonique sur IP*

Les solutions pour sécuriser la ToIP existent. Elles font l'objet de recommandations formalisées par divers acteurs du domaine. Le document édité par le NIST[9] est une référence [KUH05]. Les méthodes de sécurisation usuelles s'appuient donc sur les 5 grands principes suivants :

- les bonnes pratiques ;
- la séparation des équipements voix/données ;
- l'authentification ;
- la confidentialité ;
- la sécurité périmètrique.

La sécurité physique est une partie essentielle de tout environnement sécurisé. Elle doit permettre de limiter l'accès aux locaux et donc aux équipements ainsi qu'aux données qu'ils contiennent. L'accès aux serveurs, aux équipements réseau et aux serveurs ToIP doit être restreint aux seules personnes autorisées. Les solutions dépendent du niveau de sécurité requis (pièces fermées, lecteurs de cartes, biométrie, etc.). L'objet de cette étude n'est pas d'étudier les moyens de protéger la téléphonie contre le vandalisme, les catastrophes naturelles ou encore les incendies. Néanmoins il est intéressant de retenir qu'une solution complète de sécurisation doit passer par une analyse fine et exhaustive. La suite de ce chapitre va donc se focaliser sur l'aspect réseau et protocole.

Avant de présenter les solutions de sécurité de la ToIP, rappelons les trois biens à protéger :

- l'infrastructure logiciel ou physique (serveur, téléphone,…) nécessaire pour recevoir et émettre des appels.
- la voix : la conversation téléphonique ;
- la signalisation : les informations nécessaires à l'établissement de l'appel ou aux services de téléphonie associés au compte usager.

2.3.3. *La sécurisation des architectures*

2.3.3.1. Les bonnes pratiques

La sécurité de la ToIP passe d'abord par l'application des bonnes pratiques. La ToIP faisant désormais partie intégrante des systèmes d'informations, les principes élémentaires s'appliquent donc pleinement.

[9] NIST : le National Institute of Standard and Technology dépend du ministère de l'économie américaine.

A titre d'illustration, nous citerons :

- prévoir des mesures générales :
    - politique de sécurité globale pour les systèmes de voix ;
    - formation et responsabilisation des utilisateurs et des administrateurs ;
    - redondance des équipements ;
    - étude des incidents ;
- tout système informatique étant susceptible de contenir des failles, une politique de mise à jour doit exister :
    - il est essentiel de maintenir à jour la version des logiciels grâce à un processus de management des mises à jour. Des mesures organisationnelles doivent être mises en place pour avoir des informations sur les équipements et les logiciels. Il faut pouvoir être certain d'être averti de la parution des versions et correctifs disponibles. Les évolutions doivent être testées avant d'être déployées ;
- désactiver les ports inutiles ;
- renouveler régulièrement les mots de passe des comptes utilisateurs ;
- prévoir des mots de passe longs et non triviaux ;
- définir l'utilisation des firewalls :
    - le firewall est un système permettant de protéger un ordinateur ou un réseau d'ordinateurs des intrusions. Il filtre les paquets de données échangés en analysant les entêtes. Les champs traités à minima sont les adresses IP de l'émetteur et du destinataire, les types de paquet transporté (UDP [RFC768], TCP [RFC793]) et le numéro de port associé.
- prévoir la suppression des comptes inutiles ;
- modifier les mots de passe par défaut des équipements ;
- vérifier les droits des utilisateurs ;
- prévoir le verrouillage des configurations (*) :
    - il convient que les usagers soient en mesure de modifier les paramètres de configuration de leur hardphone ou softphone ;
- définir le paramétrage par défaut des comptes (*) :
    - profil par défaut non-joignable depuis l'extérieur ;
    - profil par défaut ne pouvant pas faire suivre ses appels vers l'extérieur ;
    - auto-déconnexion la nuit ;
- contrôler les fonctionnalités de l'IPBX (*) :
    - vérifier les réglages de sécurité du serveur ;
    - définir une politique d'accès.

(*) : spécifique à la ToIP.

Cette liste n'est pas exhaustive mais elle illustre bien que la sécurité commence d'abord par des mesures simples. Les bonnes pratiques ne sont pas spécifiques à la ToIP mais contribuent à rendre son déploiement plus sain. Elles exigent des mesures organisationnelles rigoureuses pour maintenir le niveau de sécurité visé. De même, elles doivent être accompagnées d'une sensibilisation des usagers.

2.3.3.2. La séparation des équipements données et voix

Un des principes les plus recommandés pour protéger la ToIP est de séparer les équipements du réseau Data des équipements de l'infrastructure Voix. Cette séparation peut se faire de manière physique ou de manière logique. La séparation physique se traduit par deux réseaux différents avec des switches distincts. A ce choix relativement coûteux, il est préféré la séparation logique qui se décline de plusieurs manières :

- séparation des plages d'adresses IP : ce choix consiste à attribuer une plage d'adresses par réseau ; c'est-à-dire une plage pour le réseau données et une plage pour le réseau voix. Cette option nécessite que chaque réseau possède ses serveurs DHCP ou DNS ;

- séparation par VLAN[10] : cette fois, la séparation des équipements données et voix est obtenue par l'utilisation des VLAN. Il est même envisageable d'avoir des VLAN voix pour chaque catégorie d'équipements (hardphone, softphone, serveurs). Si les VLAN partagent des équipements, il est conseillé de les mettre dans une DMZ[11].

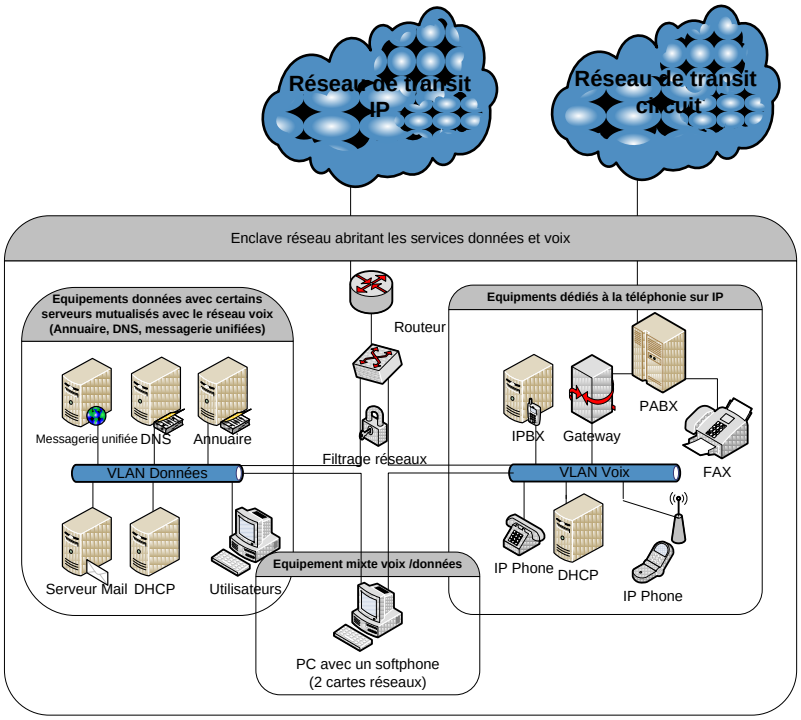


Figure 3. Architecture ToIP au sein d'une enclave type entreprise

[10] VLAN : un Virtual Local Area Network est une technique qui permet de créer un réseau local regroupant un ensemble de machines de façon logique et non physique.
[11] DMZ : en informatique, une zone démilitarisée (ou DMZ, de l'anglais demilitarized zone) est un sous-réseau isolé par un pare-feu. Ce sous-réseau contient des machines se situant entre un réseau interne et un réseau externe (typiquement, Internet). La DMZ permet à ses machines d'accéder à l'Internet et/ou de publier des services sur l'Internet sous le contrôle du pare-feu externe.

Les échanges entre les VLAN doivent être strictement contrôlés. Les équipements comme les switches ou les firewalls doivent permettre de filtrer les flux inter-VLAN. La figure 3 illustre ces recommandations pour une architecture VLAN voix et VLAN données sans serveurs mutualisés. Pour éviter de compromettre ce dispositif, il faut également penser à prendre des mesures de précautions sur les switches :

- désactiver les ports non-utilisés ;
- placer les ports inutilisés sur un VLAN inutilisé ;
- n'autoriser que les adresses MAC connues ;
- prévoir une authentification des machines.

La séparation logique des flux de voix et data est ainsi une mesure fortement recommandée. Elle doit permettre que les problèmes rencontrés sur un VLAN ne perturbent pas l'autre. L'objectif principal est bien une réduction du déni de service. D'autres dispositions sont également possibles pour renforcer la sécurité, en particulier pour limiter la possibilité de se connecter avec n'importe quelle machine sur le réseau. Les recommandations précédentes peuvent donc être complétées en mettant les ports inutiles dans un VLAN inutilisé, ou par la mise en place d'un contrôle d'accès sur l'adresse MAC et de mécanismes d'authentification.

2.3.3.3. L'authentification

L'authentification est la fonction de sécurité qui consiste à apporter et à contrôler la preuve de l'identité d'une entité (personne, message, logiciel,…). Compte tenu du périmètre de l'analyse, plusieurs mécanismes peuvent être envisagés, que ce soit au niveau réseau ou au niveau applicatif et donc dans les protocoles de ToIP. Concernant l'authentification au niveau protocolaire, ce point sera traité dans le paragraphe 2.3.4. et d'une manière plus détaillée pour SIP dans le chapitre 3.

Concernant les mécanismes d'authentification au niveau réseau, dans la mesure où la technologie IP est utilisée, les solutions usuelles de sécurité comme IPSec ou TLS peuvent répondre à l'objectif visé. Pour les équipements sans fil, les mécanismes utilisant EAP (Extensible Authentication Protocol) [RFC3748] peuvent être mis en place. Chaque couche porte ainsi sa solution de sécurité.

Généralement l'authentification est à sens unique, seul le terminal est authentifié. Ce constat est issu de modèle client/serveur dans lequel un client demande un accès à des services fournis par le serveur. Les protocoles de sécurité utilisés dans ces réseaux sont basés sur un processus type défi/réponse. Le serveur envoie un défi au client et ce dernier applique une fonction cryptographique sur le défi en utilisant un secret partagé (comme un mot de passe). Ainsi seul le client est authentifié. Cette approche ne suffit pas en ToIP. Dans la mesure où la signalisation porte des informations personnelles comme les destinataires des appels, il est plus qu'important que le client soit certain qu'il dialogue avec le serveur légitime. Cet objectif nécessite donc la mise en place d'une authentification dite « mutuelle ».

L'authentification doit également prendre en compte la mobilité. Les paramètres d'authentifications comme le mot de passe sont essentiels dans la robustesse du mécanisme. Utiliser un mot de passe sur un PC non maîtrisé est en soi une vulnérabilité. Il est essentiel de préserver ce secret. Ce sujet sera illustré dans le paragraphe 4.1.3.2.1.

2.3.3.4. La sécurité périmètrique

La sécurité périmètrique concerne les équipements placés aux frontières de l'infrastructure ToIP. Les éléments en périphérie assurent :

- la continuité des appels en permettant l'interopérabilité ou l'interconnexion ;
- éventuellement des propriétés de sécurité.

Il faut donc sécuriser les passerelles en utilisant les bonnes pratiques [§2.5.1.1.] voire en complétant le dispositif par des équipements spécifiques comme des firewalls ou des IDS (Intrusion Detection System). Rappelons que cette approche est lié à une infrastructure IP. Certes il existe des IDS dédiés à la ToIP comme le montre les travaux [NAS09]. Il convient néanmoins d'apporter des solutions de sécurité robuste dans la mise en œuvre du service téléphonique lui-même.

2.3.3.5. Confidentialité

Le chiffrement partiel ou total d'une information est nécessaire quand il existe un besoin de confidentialité. Ce besoin peut concerner la voix ou la signalisation. Concernant les couches transport ou réseau, les protocoles de sécurité usuels dans le mode de l'IP peuvent être mis à contribution pour sécuriser la ToIP. TLS et IPSec offrent un service de confidentialité. Plus largement, toutes les techniques VPN[12] peuvent permettre de protéger la voix et la signalisation entre deux sites. [DIA07] a comparé les différentes solutions et fait apparaître qu'IPSec est un bon candidat en termes de sécurité. Néanmoins les impacts sur les performances ne sont pas négligeables. Les implémentations méritent d'être optimisées. Comme nous le verrons dans le paragraphe 2.3.4, les deux principaux protocoles de la ToIP SIP et H323 préconisent l'emploi de chiffrement.

2.3.4. *La sécurisation des protocoles de téléphonie sur IP*

Les principaux protocoles de la ToIP, H323 et SIP, spécifient un cadre général pour la mise en œuvre de la sécurité. Cette dernière s'applique à la signalisation ou à la voix, voire aux deux. Les deux standards prévoient ou recommandent des mécanismes pour l'authentification, le chiffrement des données, le non-rejeu, l'anonymat et l'intégrité. Le protocole H323 se décompose en différentes recommandations qui portent chacune un ou plusieurs mécanismes de sécurité. L'authentification des usagers H323 est ainsi définie au travers de sa spécification RAS (Registration Admission Status) : l'authentification s'appuie alors sur un secret partagé ou un mécanisme à clé publique, voire après un

[12] VPN : Virtual Private Network (ou réseau privé virtuel) permet d'établir un lien sécurisé entre deux équipements.

échange de type Diffie Hellman[13]. Les modalités pour l'établissement et le contrôle d'appel peuvent être protégées au travers de la recommandation H.235 qui décrit le fonctionnement de H323 avec TLS ou IPSec. Ces protocoles peuvent alors fournir un service d'authentification, de confidentialité et d'intégrité des messages. De même SIP propose également des mécanismes de sécurité assez similaires qui seront présentés dans le chapitre 3, comme TLS ou encore IPSec. Concernant le média, les deux protocoles utilisent RTP [RFC3550] pour le transport de la voix. Le chiffrement de la communication est donc possible avec SRTP.

Les protocoles de la ToIP au travers de leurs spécifications comme SIP ou H323 reportent la sécurité sur le réseau en recommandant l'usage de TLS ou d'IPSec. Les limites de [§2.3.3.] se retrouvent alors au niveau des protocoles. Les propriétés de sécurité peuvent être très différentes d'un appel à l'autre. Il existe évidemment des mécanismes au niveau applicatif comme S/MIME pour SIP qui permet une solution de bout-en-bout en confidentialité, intégrité et une authentification mutuelle des usagers. Cette approche exige cependant que les deux terminaux implémentent les mêmes protocoles de ToIP avec les mêmes propriétés de sécurité. Or par essence, les interconnexions actuelles garantissent uniquement l'établissement de l'appel. Un usager utilisant SIP ne pourra donc pas établir un appel sécurisé de bout-en-bout avec un usager H323 à partir des spécifications des protocoles.

Ces solutions de sécurité rencontrent également de nombreuses difficultés d'implémentation et de déploiement. Le chiffrement de la signalisation ou de la voix [GUP07] nécessite un mécanisme de distribution de clés secrètes ou une infrastructure de gestion de certificats pour pouvoir être utilisé par tous les usagers ou les serveurs. Le chiffrement nécessite du temps de calcul et augmente la taille des paquets IP, ce qui n'est pas toujours conciliable avec une application temps réel comme pour le transport de la voix pour une communication. La multitude de protocoles comme SIP, H323 ou encore Skype ne facilite pas non plus l'adoption d'un standard de sécurité.

2.3.5. *La sécurité au niveau applicatif*

Fort du constat de [§2.3.4.], déjà avéré dans les années 80, la NSA[14] avait envisagé une solution de sécurité « universelle » appelée « Future Narrowband Digital Terminal » (FNBDT), permettant une sécurité de bout-en-bout. Après établissement de l'appel, le système met en œuvre une signalisation de sécurité dans le canal média. Depuis, le standard a évolué pour prendre en compte la technologie IP avec la version appelée « Secure Communications Interoperability Protocol » SCIP. Ce dernier est même devenu une norme aux Etats-Unis et à l'OTAN pour sécuriser la téléphonie. Il reste que cette approche, usuelle pour les militaires, reste marginale pour les particuliers.

---

[13] Diffie Hellmann : en cryptographie, l'échange de clés Diffie-Hellman, du nom de ses auteurs, est une méthode par laquelle deux entités peuvent se mettre d'accord sur un nombre (qu'ils peuvent utiliser comme clé pour chiffrer la conversation suivante) sans qu'une tierce personne puisse découvrir le nombre en écoutant les échanges. La sécurité de ce protocole réside dans le fondement mathématique de l'échange. [RFC2631] est une des applications.

[14] NSA : la National Security Agency est un organisme gouvernemental des États-Unis, responsable de la collecte et de l'analyse de toutes formes de communications.

La force de ce protocole est de ne faire aucune hypothèse sur le réseau sous-jacent. FNBDT fournit donc une architecture interopérable permettant de sécuriser des communications de bout-en-bout, quelque soit le réseau d'accès. Il faut néanmoins partager un équipement appelé STE (Secure Terminal Equipement). FNBDT n'est pas la seule solution applicative pour la sécurité de la ToIP mais il est le plus emblématique. Ce n'est pas non plus le premier protocole puisqu'il s'inscrit dans la continuité d'architectures plus anciennes comme SIGSALY[15].

FNBDT a inspiré les travaux [BAS05] de Carol Bassil qui a défini une architecture libre de sécurité applicative pour la voix sur IP. L'émergence d'une solution de sécurité robuste comme FNBDT ou SVSP n'a pas encore vu le jour pour les particuliers, car la perception du danger ne pousse pas encore à l'adoption de solutions de sécurité généralisées. Le monde des professionnels y vient comme le suggère [BEL09]. La sécurité de bout-en-bout des appels IP reste un sujet qui n'est pas traité systématiquement dans les architectures.

2.4. La typologie des méthodes pour sécuriser la téléphonie sur IP

De cette étude, nous proposons une typologie pour les solutions de sécurité. Les 5 grands principes pour sécuriser la ToIP sont :

- **sécuriser l'architecture** : c'est la première étape dans l'environnement des systèmes d'informations et de l'IP. L'application des bonnes pratiques est la première étape pour envisager toutes les autres solutions. Un softphone sur un PC sans anti-virus est exposé à toutes les attaques de l'Internet ;

- **sécuriser la signalisation** : dans le cas de SIP, cela revient à introduire des mécanismes comme S/MIME ou HTTP Digest. La sécurité est définie au moment de l'implémentation mais la continuité de service se limite aux domaines appliquant la politique de sécurité et utilisant le même protocole. Evidemment, il n'y a pas de continuité actuellement au changement de protocole pour permettre l'établissement de l'appel ;

- **sécuriser par extension** : il est (évidemment) toujours possible d'améliorer l'existant en rajoutant une en-tête ou d'appliquer de nouvelles méthodes de chiffrement, mais se pose alors le problème du déploiement et la gestion des différentes configurations. La compatibilité des différentes versions ne permettant pas d'imposer de nouvelles solutions. De même que précédemment cette approche ne s'applique qu'aux domaines ayant choisi cette solution ;

[15] SIGSALY : ce système est le premier système de télécommunications numériques. Il a été déployé par l'armée américaine en 1943 pour les communications entre les états-majors alliés. SIGSALY a été lancé sur la base d'un projet initié en 1942 pour pallier à l'insécurité des télécommunications analogiques brouillées et restituées en clair en temps réel par l'armée allemande. Reliant jusqu'à 12 stations lors de son retrait en 1946, son existence est restée secrète jusqu'en 1976.

- **sécuriser le canal média au moment de l'appel** : c'est le principe de SRTP. Cela nécessite que les protocoles de signalisation permettent la négociation de la clé de chiffrement ;

- **sécuriser d'une manière complètement transparente vis-à-vis du réseau** : les différentes infrastructures de ToIP interopèrent globalement entre eux pour la gestion des appels. La continuité concerne principalement les fonctions de base de la signalisation et l'acheminement de la voix. La sécurité est quant à elle propre à chaque infrastructure et ne fait l'objet d'aucune spécification au niveau des interconnexions. Faire abstraction du réseau permet donc d'envisager une solution bout-en-bout au niveau applicatif pour la sécurité : c'est l'approche de FNBDT, SCIP ou encore SVSP [§6.]. La solution de sécurité est mise en œuvre après l'établissement de l'appel dans le canal du média.

Chaque système portant ses propres mécanismes de sécurité, la sécurité de bout-en-bout des appels n'est actuellement pas considérée, sauf pour des usagers gouvernementaux ou dans le cas d'une homogénéité protocolaire entre deux usagers. Seule une solution basée sur le canal média semble répondre à cette problématique.

2.5. Méthodes et outils d'évaluation de la sécurité de la téléphonie sur IP

En complément des solutions pour sécuriser la ToIP, il existe deux processus complémentaires : l'audit et le fuzzing. Ces derniers n'apportent pas de solutions à proprement parler mais permettent de s'assurer que les mesures mises en place ne contiennent pas de failles et sont bien appliquées.

2.5.1. *L'audit*

L'audit n'est pas une technique propre à la ToIP. Il permet d'établir un diagnostic des services ou de solutions de sécurité mis en place dans l'entreprise ou l'organisme, et d'analyser les vulnérabilités en fonction de l'efficacité de ces services de sécurité. Cette démarche est essentielle pour vérifier que les mesures de sécurité sont bien appliquées, tant du point de vue organisationnel et que technique.

2.5.2. *Le fuzzing*

La deuxième méthode issue des systèmes d'informations qui peut s'appliquer à la ToIP est le fuzzing. Cette technique a émergé ces dernières années pour découvrir les vulnérabilités dans les implémentations logicielles et matérielles. Découlant du terme anglais fuzzy, signifiant flou ou brouillé, le fuzzing est une méthode s'appuyant sur des outils logiciels, des fuzzers, pour automatiser l'identification de bugs ou de failles dans des applications.

Le processus consiste à vérifier toutes les entrées possibles pour une application donnée, et à trouver les fonctionnements anormaux ou non conformes. Le fuzzer servira ainsi à bombarder l'application de codes volontairement malformés. L'opération se décompose en deux phases :

- générer des données aléatoires et/ou malveillantes ;
- injecter ces données dans l'application cible par ses divers canaux de communication et d'interaction.

Cette approche diffère de l'audit et des tests logiciels classiques en se focalisant sur les actions inattendues. La ToIP entre pleinement dans le spectre de cette approche. Des travaux ont d'ailleurs été menés sur cette problématique. Au travers d'un outil spécifique pour la voix sur IP appelé KIF (cf. figure 4), les auteurs de [ABD08] ont souligné l'intérêt de cette technique en montrant que de nombreux équipements de ToIP présentent des vulnérabilités.

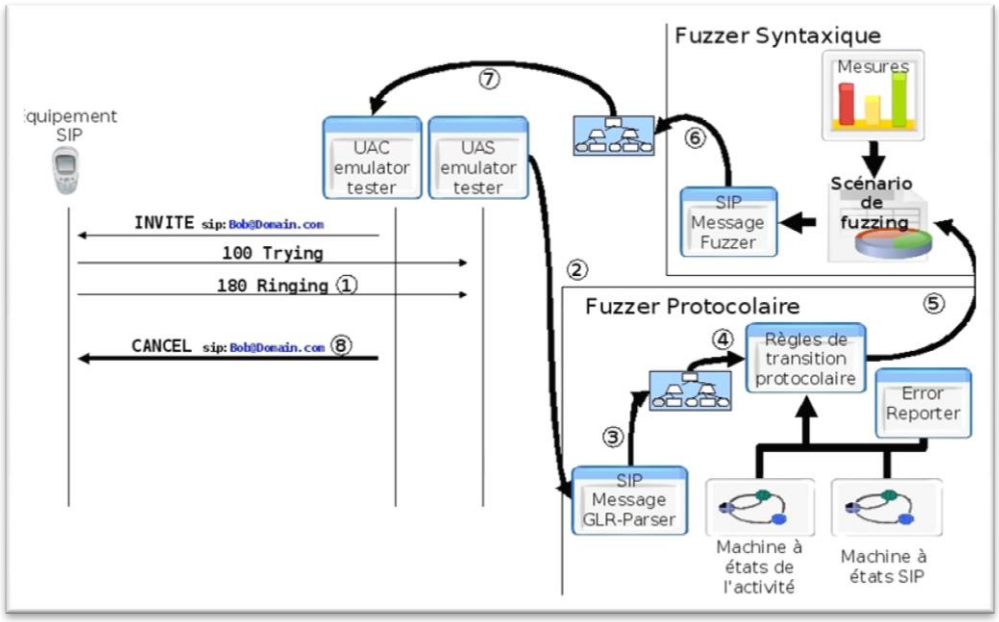


Figure 4. L'environnement de KIF [ABD08]

2.6. Conclusion

La sécurité de la téléphonie doit pouvoir répondre à trois grandes exigences : la disponibilité, l'intégrité et la confidentialité. D'autres propriétés comme l'authentification, le non-rejeu et la non-répudiation peuvent être également nécessaires. Chacune de ces exigences demande des mécanismes protocolaires ou des choix d'architecture bien particuliers. Les solutions de sécurité envisagées actuellement répondent à ces attentes mais ne tiennent cependant pas compte de la diversité des environnements de la téléphonie sur IP, sauf à considérer une solution dans le canal média comme avec FNBDT. Le contexte d'emploi des mécanismes de sécurité est souvent très restreint.

La sécurité n'est pas unifiée pour la téléphonie alors même que l'insécurité des SI augmente [LAD06]. Bien qu'il existe un consensus entre les différents opérateurs ou intégrateurs concernant la conception du réseau de téléphonie IP, il n'y a pas de standard universel pour la signalisation et donc pour la sécurité des appels de bout-en-bout. Ce constat milite pleinement pour la conception d'une solution de sécurité indépendante du réseau de transport. Cette approche ne traite cependant que partiellement le critère de

disponibilité mais permettra d'apporter confidentialité, authentification des entités et intégrité, voire plus si le contexte de l'appel le nécessite.

La fédération des solutions de sécurité viendra peut-être de l'adoption massive actuelle du standard SIP de l'IETF (Internet Engineering Task Force). Ce dernier spécifie la signalisation pour l'établissement d'un appel et les modalités pour le transport de la voix. Une large communauté a contribué à sécuriser l'environnement SIP, principalement en ajoutant de nouveaux paramètres ou en préconisant l'utilisation de protocoles de sécurité pour le transport des messages SIP ou de la voix. Ces propositions ont un coût en temps de calcul, en bande passante et ne sont pas toujours interopérables avec les implémentations existantes. Une sécurité de bout-en-bout des appels SIP reste encore à formaliser.

De facto, notre étude a recherché à renforcer la sécurité de l'existant, en particulier l'authentification, sans modifier les échanges de SIP pour faciliter l'adoption de nos solutions. L'objectif est de garantir une totale interopérabilité avec les infrastructures existantes tout en minimisant les impacts sur les performances de l'infrastructure de ToIP. Ce cahier des charges nous a permis de proposer des solutions innovantes et validées, en spécifiant une sémantique pour des valeurs aléatoires. Conscient que ces contributions sont un pis-aller, une définition d'une architecture de ToIP sécurisée est proposée dans le chapitre 6.

CHAPITRE 3

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.

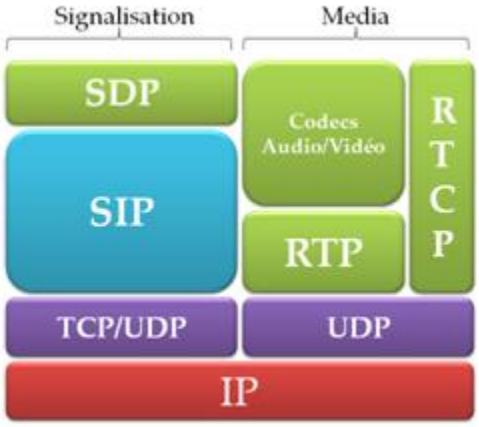


Figure 5. La pile protocolaire de SIP

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.

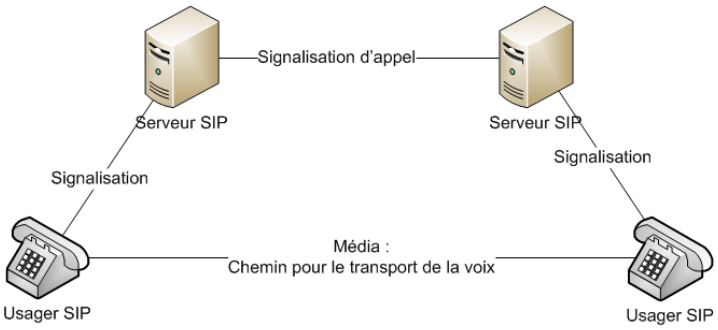


Figure 6. Architecture élémentaire SIP

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.

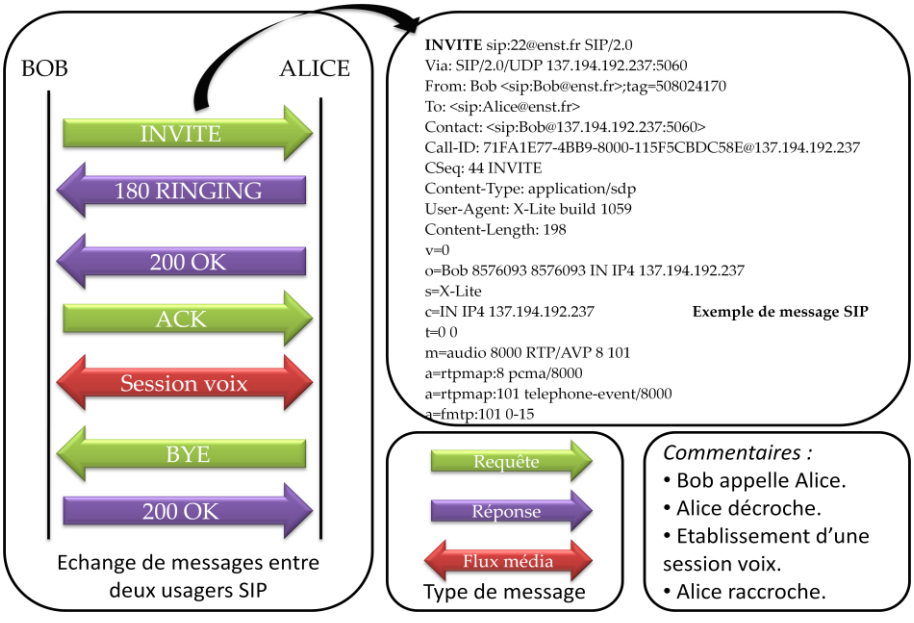


Figure 8. Echange de messages SIP pour l'établissement d'une session

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.

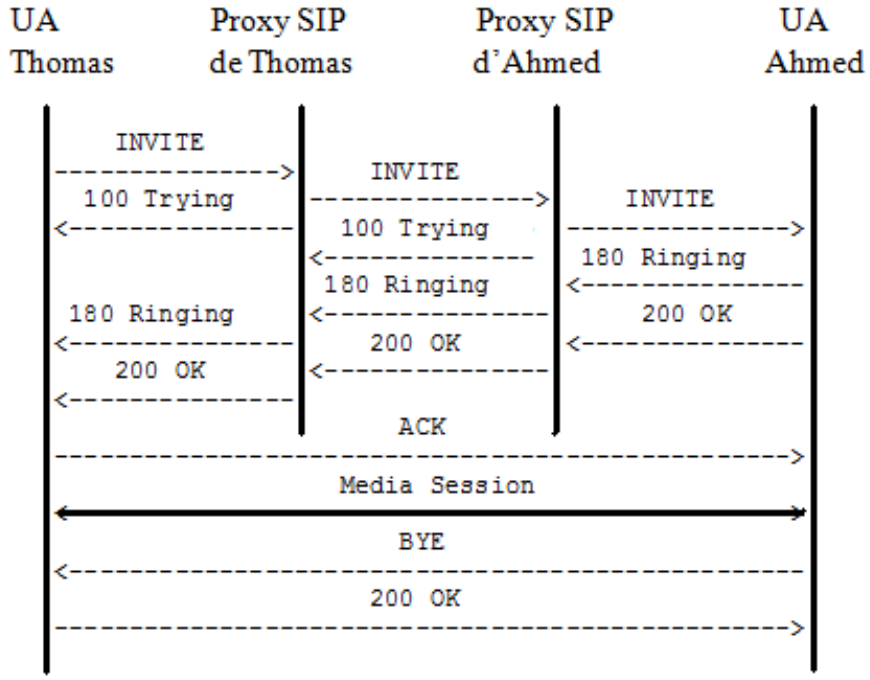


Figure 9. Etablissement d'un appel dans une infrastructure SIP

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.

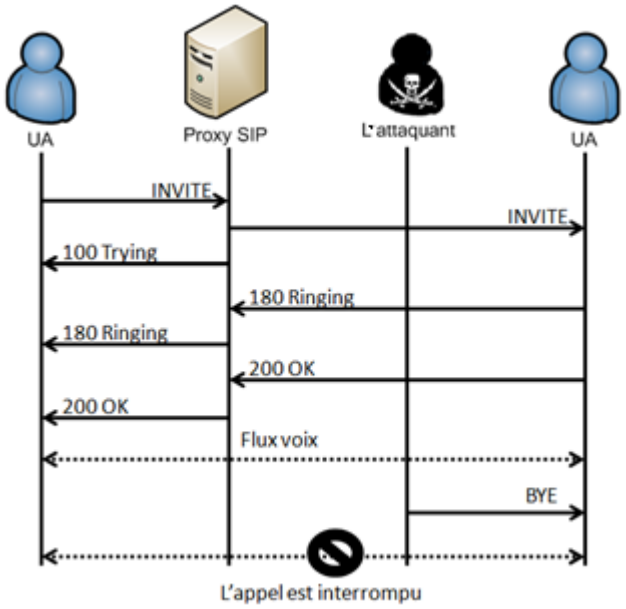


Figure 10. Attaque par le BYE

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.

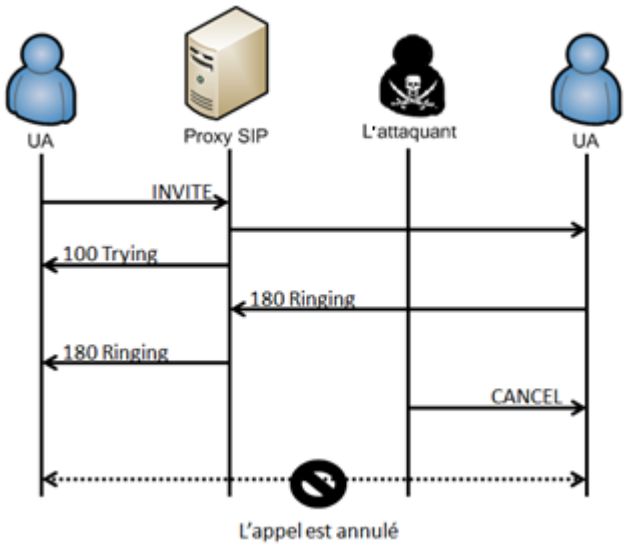


Figure 11. Attaque par le CANCEL

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.

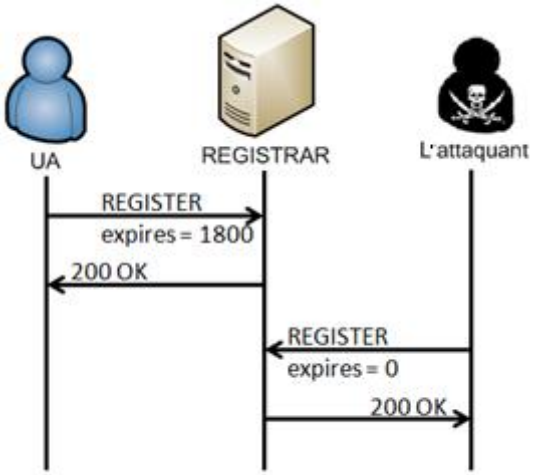


Figure 12. Attaque sur le REGISTER

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.

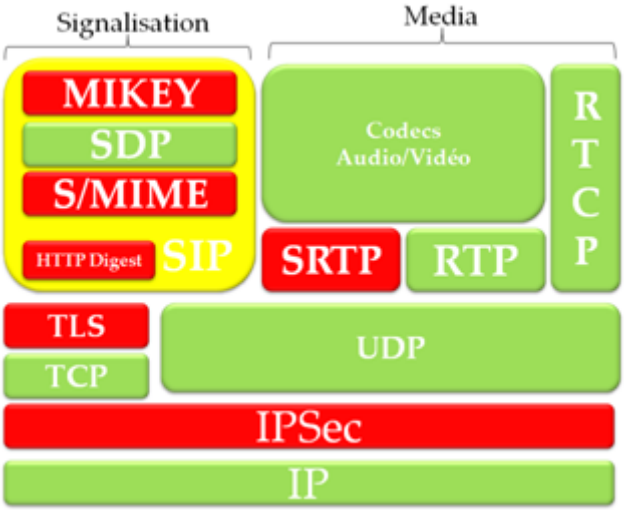


Figure 13. Pile protocolaire SIP avec les éléments de sécurité

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.

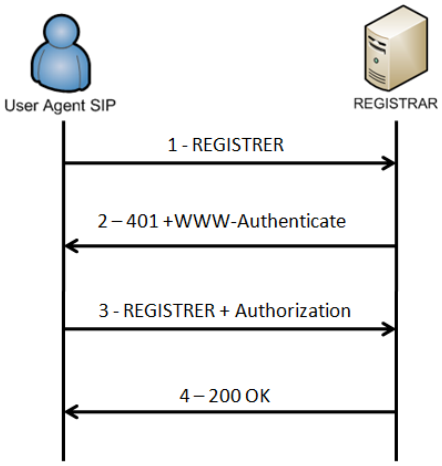


Figure 14. Enregistrement et authentification HTTP Digest dans un contexte SIP

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.

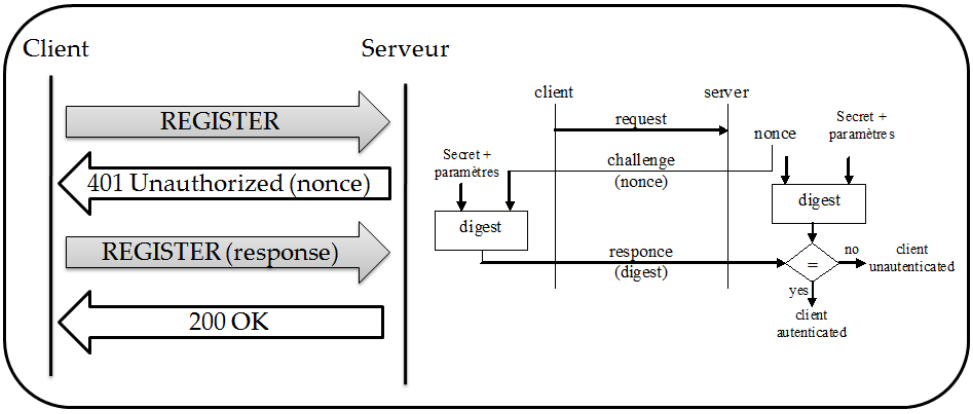


Figure 15. Authentification HTTP Digest SIP pour un message REGISTER

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) :

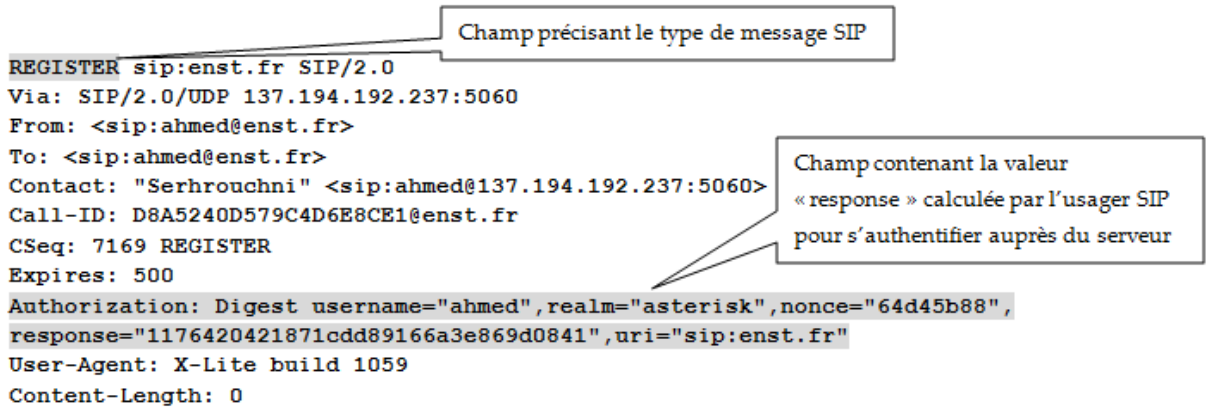


Figure 17. Exemple de message REGISTER du protocole SIP avec le champ « response »

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.

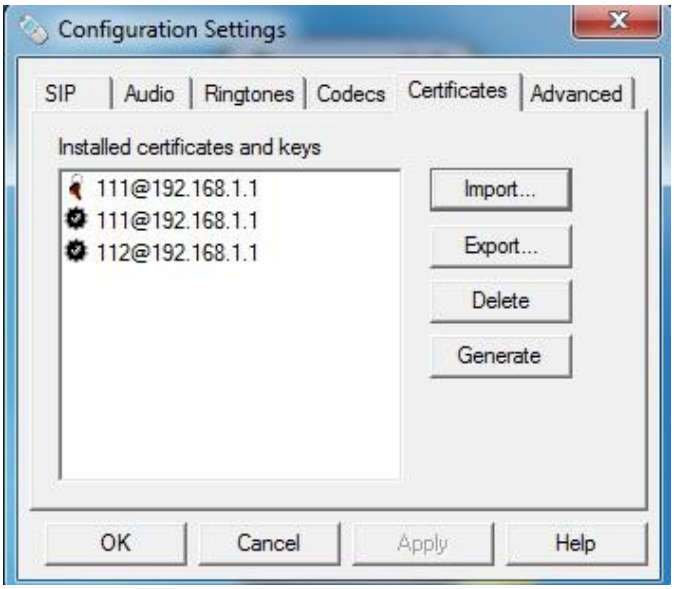


Figure 18. Interface graphique de Lynxphone pour la gestion des clés (publique et privées)

[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).

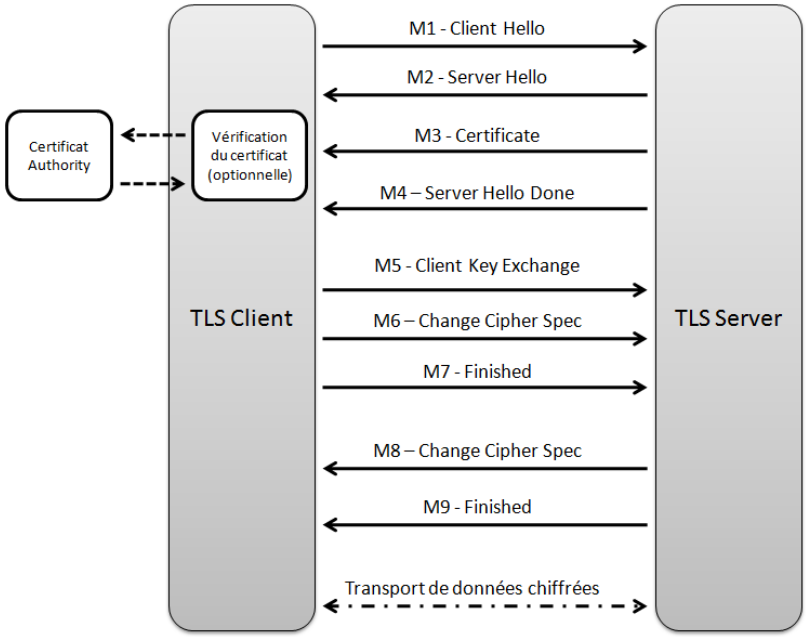


Figure 20. Echanges de messages TLS

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.

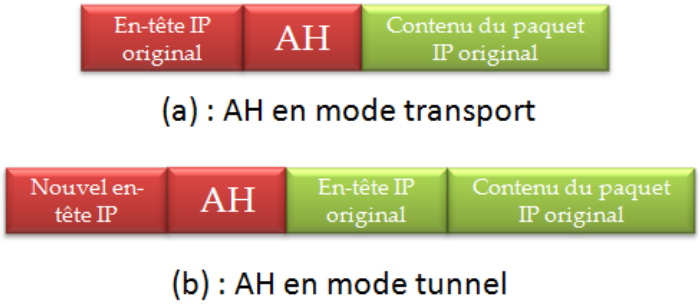


Figure 23. IPSec en mode AH

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).

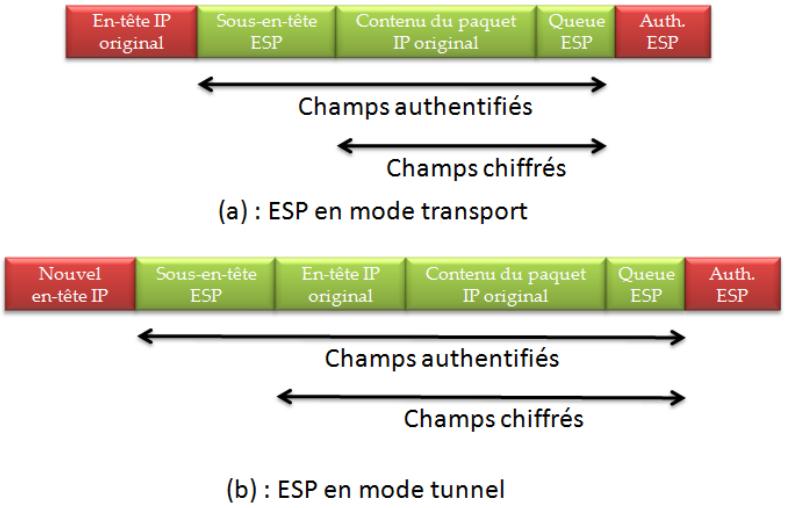


Figure 24. IPSec en mode ESP

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).

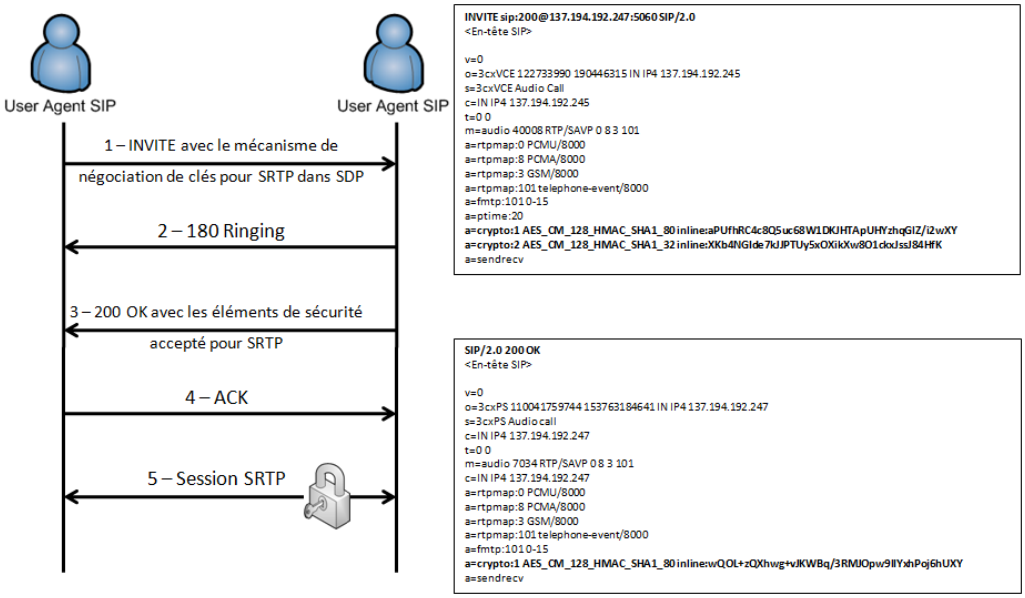


Figure 25. SRTP dans un contexte SIP

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.

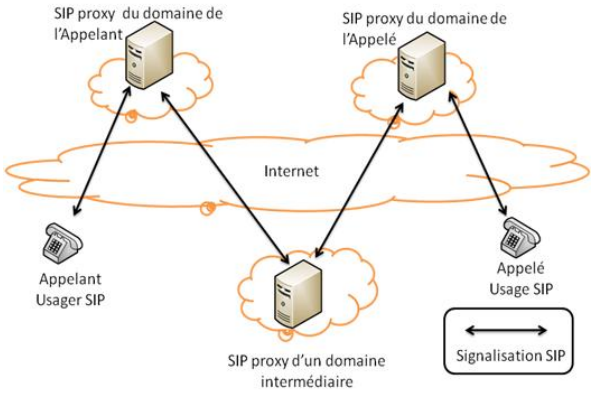


Figure 26. Cheminement de la signalisation SIP dans les réseaux IP

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.

CHAPITRE 4

SOLUTIONS POUR LA SECURITE DU PROTOCOLE SIP

Ce chapitre présente nos contributions au renforcement de l'authentification dans les infrastructures de téléphonie sur IP basé sur le protocole Session Initiation Protocol (SIP). L'émergence de nouvelles solutions étant souvent freinée par les problèmes d'interopérabilité, ce travail s'est attaché à proposer des mécanismes les plus transparents possibles pour les équipements existants. Le renforcement de la sécurité s'appuie sur la spécification d'une nouvelle sémantique pour des champs déjà existant dans les messages SIP. Au travers des mécanismes envisagés dans ce chapitre, nous proposons des solutions limitant l'usurpation d'identité et le déni de service, et optimisant les échanges protocolaires. Ces nouveaux services peuvent être considérés comme une valeur ajoutée au standard.

4. Définition et conception de solutions de sécurité pour SIP

4.1. Analyse des solutions d'authentification de SIP

4.1.1. *L'authentification*

L'authentification est la fonction de sécurité qui consiste à apporter et à contrôler la preuve de l'identité d'une personne, de l'émetteur d'un message, provenant d'un logiciel, d'un serveur logique ou d'un équipement. Par ailleurs il est habituel de faire reposer ce service sur un ou plusieurs éléments comme :

- ce que l'on sait (un mot de passe, par exemple) ;
- ce que l'on a (un certificat, par exemple) ;
- ce que l'on est (une empreinte vocale, par exemple) ;
- ce que l'on sait faire (une signature manuscrite, par exemple).

Dans une authentification intervient le demandeur qui doit prouver son identité et le receveur qui offre un service sous condition : c'est l'authentification simple. Le demandeur peut également vérifier que le receveur est une entité légitime : c'est l'authentification mutuelle. Ces échanges nécessitent un canal qui relie les différents acteurs pour réaliser ces vérifications basées sur un mot de passe ou un certificat dans le contexte de SIP [§3.4.].

4.1.2. *L'importance de l'authentification*

Le précédent chapitre a porté sur les différentes interactions entre les mécanismes de sécurité de SIP et son fonctionnement. Cela a permis d'établir les limites de la sécurité telle qu'elle est définie dans le RFC 3261. Parmi les différentes propriétés apportées par ces mécanismes, c'est l'authentification qui a été particulièrement étudiée dans ces travaux.

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.

Ce service est nécessaire 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 [§3.3.]. Ce chapitre va donc aborder les points suivants : les solutions de renforcement déjà proposées, nos motivations et les solutions proposées pour renforcer l'authentification dans un contexte SIP.

4.1.3. *Analyse des solutions déjà proposées*

4.1.3.1. Les propositions modifiant la syntaxe des messages SIP

4.1.3.1.1. Ajouter un nouveau champ pour l'authentification et l'intégrité

La sécurité de SIP se traduit par des champs destinés à l'authentification, comme HTTP Digest ou avec des en-têtes et un corps associé au formalisme S/MIME. Une des possibilités pour apporter un nouveau service de sécurité est de créer une nouvelle en-tête (cf. typologie des solutions [§2.4.]). [GEN08] a proposé un mécanisme d'authentification et d'intégrité en ajoutant le champ Integrity_Auth. La syntaxe de cette proposition est la suivante et reprend celle de SIP :

- Integrity-Auth = "lntegrity-Auth" / Integrity-auth-value
- Integrity-auth-value = credentials-value ; algorithm ; nonce
- Algorithm-"algorithm" EQUAL Alg-value
- Alg-value = "MD5|SHA1"credentials-value = quoted-string

Le principe de cette authentification est basé sur le secret pré-partagé entre le client et le fournisseur. A l'instar de l'authentification HTTP Digest, cette solution utilise une fonction de hachage et un secret pour prouver l'identité de l'émetteur. A la réception d'une requête, le serveur ou le client SIP vérifie la valeur de Integrity_Auth en calculant lui-même sa valeur à partir du message SIP, de la valeur aléatoire et du secret. La figure 27 illustre l'emploi de ce nouveau champ d'en-tête.

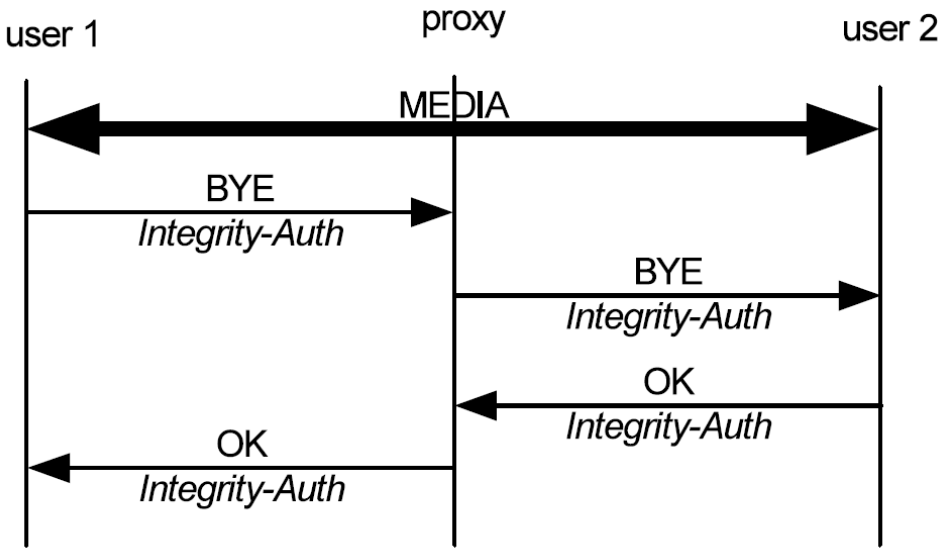


Figure 27. Emploi du champ Integrity_Auth dans un message BYE

La figure 28 présente le mode de calcul de la valeur Integrity_Auth.

$$\text{Integrity\_Auth} = \text{Hash}(\text{SIP\_MESSAGE}||\text{Random}||\text{Hash}(\text{PWDuser}\oplus\text{Random}))$$

Figure 28. Formule du champ Integrity_Auth proposé par [GEN08]

Ce mécanisme permet une authentification mutuelle entre les usagers et les *proxy* ou *registrar*. L'usurpation d'identité ou les attaques par BYE et CANCEL sont évitées.

4.1.3.1.2. Le format Authenticated Identity Body

[RFC3893] propose une alternative à la solution S/MIME de [RFC3261] en définissant un nouveau champ dans l'entête appelé Athenticated Identity Body (AIB). S/MIME offre une solution complète mais avec les limites exposées dans [§3.4.1.2.]. AIB est également basé sur l'utilisation de certificats, mais s'inscrit comme réponse intermédiaire permettant d'authentifier un usager par tous les éléments de l'architecture connaissant sa clé publique. La figure 29 donne un exemple de mise en œuvre.

Dans la mesure où le champ authentifié et intègre est non chiffrée, la solution ajoute moins d'informations que S/MIME Tunnel. La partie à authentifier est composée des éléments suivants : From, To, Call-Id, Date, Cseq et Contact. Il reste que cette contribution a besoin d'une infrastructure de gestion de certificats.

```
INVITE sip:bob@example.net SIP/2.0
Via: SIP/2.0/UDP pc33.example.com;branch=z9hG4bKnashds8
To: Bob <sip:bob@example.net>
From: Alice <sip:alice@example.com>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314159 INVITE
Max-Forwards: 70
Date: Thu, 21 Feb 2002 13:02:03 GMT
Contact: <sip:alice@pc33.example.com> Content-Type: multipart/mixed; boundary=unique-boundary-1

--unique-boundary-1

Content-Type: application/sdp
Content-Length: 147
v=0
o=UserA 2890844526 2890844526 IN IP4 example.com
s=Session SDP
c=IN IP4 pc33.example.com
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

--unique-boundary-1
Content-Type: multipart/signed; protocol="application/pkcs7-signature"; micalg=sha1;
boundary=boundary42
Content-Length: 608
--boundary42
Content-Type: message/sipfrag
Content-Disposition: aib; handling=optional
From: Alice <sip:alice@example.com>
To: Bob <sip:bob@example.net>
Contact: <sip:alice@pc33.example.com>
Date: Thu, 21 Feb 2002 13:02:03 GMT
Call-ID: a84b4c76e66710
CSeq: 314159 INVITE
--boundary42
Content-Type: application/pkcs7-signature; name=smime.p7s
Content-Transfer-Encoding: base64
Content-Disposition: attachment; filename=smime.p7s; handling=required
ghyHhHUujhJhjH77n8HHGTrfvbnj756tbB9HG4VQpfyF467GhIGfHfYT6
4VQpfyF467GhIGfHfYT6jH77n8HHGghyHhHUujhJh756tbB9HGTrfvbnj
n8HHGTrfvhJhjH776tbB9HG4VQbnj7567GhIGfHfYT6ghyHhHUujpfyF4 7GhIGfHfYT64VQbnj756 --boundary42—
--unique-boundary-1—
```

Figure 29. Exemple de message avec le mécanisme AIB [RFC3893]

4.1.3.1.3. Authentification SIP pour un domaine de confiance

[SRI05] propose une méthode d'authentification utilisant à la fois un mot de passe et les certificats. Les travaux de Srinivasan et al. ont exploré une solution où le client n'échange qu'avec le Proxy SIP, à charge pour le Proxy d'authentifier le client en échangeant avec le Registrar. Les serveurs possèdent des certificats issus d'une autorité. La figure 30 présente l'architecture de cette contribution.

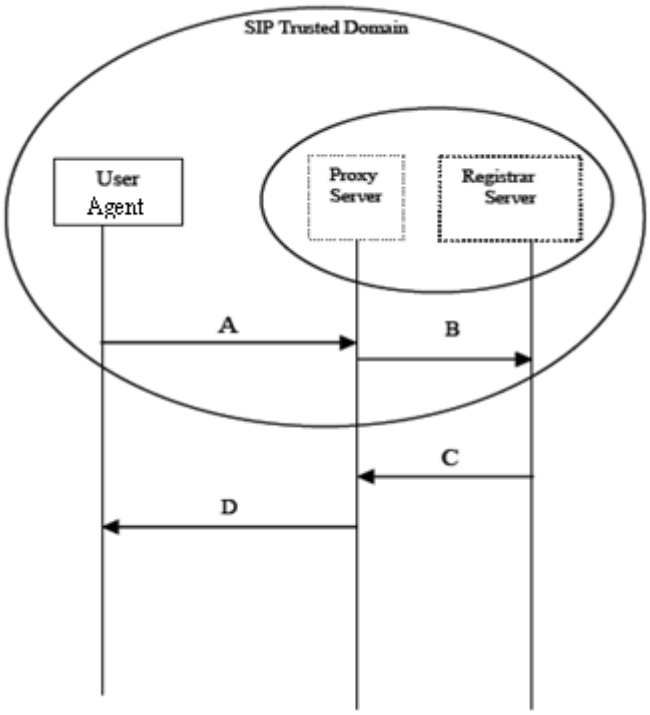


Figure 30. Authentification dans le domaine de confiance de [SRI05]

La figure 30 présente les différents messages échangés. Le principe et le détail des calculs sont explicités ci-dessous (la description des annotations est détaillée dans le tableau 9) :

Messages	Détail des calculs
Message A : UA → Proxy : n, $(R_0)_L$, I_{RS}, TS_{UC} L'UA envoie une requête vers le Proxy avec un certain nombre de valeurs – dont certaines sont dérivées du secret – et un horodatage. L'UA envoie donc A au Proxy Server	○ PW_{UC} est le secret partagé par le Registrar Server et l'UA : $PW_{UC}=H(N\|\|I_{UC})$ (N est seulement connu par le Registrar) ; ○ R_0 est une valeur aléatoire secrète pour l'UA ; ○ $r=H[N\|\|I_{RS}]\oplus H[N\|\|I_{UC}]\oplus I_{RS}\oplus I_{UC}$; ○ $n=r\oplus PW_{UC}$; ○ Valeur de l'horodatage TS_{UC} ; ○ UA génère une clé temporaire $L = H(PW_{UC}\oplus TS_{UC})$.
Message B : Proxy → Registrar : σ, n, $(R_0)L$, TSUC, S1, CPS Le proxy vérifie la valeur de l'horodatage et la compare à l'heure en usage. Si les deux valeurs sont cohérentes, le proxy envoie les données signées par sa clé privée vers le Registrar. Le Proxy envoie donc B au Registrar Server.	○ σ génère une valeur aléatoire ; ○ Le Proxy signe les éléments avec sa clé privée KR_{PS} : $S1=E_{KR_{PS}}(H(\sigma,n,(R0)L,TS_{UC},C_{PS}))$.
Message C : Registrar → Proxy : γ, $E_{KU_{PS}}(H(I_{UC}\|\|R_0))$, S2, TS_{RS}, C_{RS} Le Registrar authentifie le proxy avec sa clé publique KU_{PS} et vérifie également l'horodatage. Il extrait ensuite l'identité de l'UA et le relie à son secret en fonction du protocole pour l'authentifier. Le Registrar Server envoie alors C au proxy :	○ $I_{UC} = I_{RS} \oplus n \oplus H(N\|\|I_{RS})$; ○ Maintenant l'UA est identifié, le Registrar Server l'associe à son mot de passe PW_{UC}. Le Registrar peut alors calculer L0 et retrouve R_0 ; ○ Le Registrar génère une valeur aléatoire γ ; ○ Le Registrar signe certains éléments : $S2 =E_{KR_{RS}}(H(\sigma,\gamma,E_{KU_{PS}}(H(I_{UC}\|\|R_0))))$ ○ Le Registrar génère un horodatage TS_{RS}.

Messages	Détail des calculs
Message D : Proxy → UA : $(TC_{UC})_{SK}$. Le Proxy serveur reçoit C et vérifie d'abord l'horodatage puis authentifie le message avec le certificat. L'envoi de C vaut également pour l'authentification de l'usager. Le Proxy génère une clé de session SK qui termine le processus d'authentification :	○ $SK = H(I_{UC}) \oplus R_0$.

Tableau 9. Description des notations de [SRI05]

Notation	Description
Ii	Identité de i
Ti	Horodatage par l'entité i
Ci	Certificat de l'entité i
(M)k	Message M chiffré par une clé symétrique
Ek(M)	Message M chiffré par une clé publique
H(M)	Fonction de hachage appliqué à M
⊕	Opération XOR
||	Concaténation

Cette méthode apporte une authentification mutuelle pour chaque échange avec une solution contre les rejeux grâce à l'horodatage des messages. La synchronisation des différentes horloges est donc primordiale pour pouvoir faire fonctionner cette méthode. [SRI05] ne précise pas dans quels champs SIP sont insérer les messages A, B, C et D.

4.1.3.1.4. L'utilisation du mécanisme « Authentication and Key Agreement »

Le mécanisme d'authentification « Authentication and Key Agreement » AKA [3GPP] permet l'authentification et la distribution de clés dans les réseaux UMTS. AKA permet l'authentification mutuelle à partir d'un challenge/réponse et d'une valeur générée à partir d'une clé secrète. Dans un réseau UMTS, le réseau authentifie le client mais ce dernier en fait autant pour vérifier qu'il dialogue bien avec le réseau légitime.

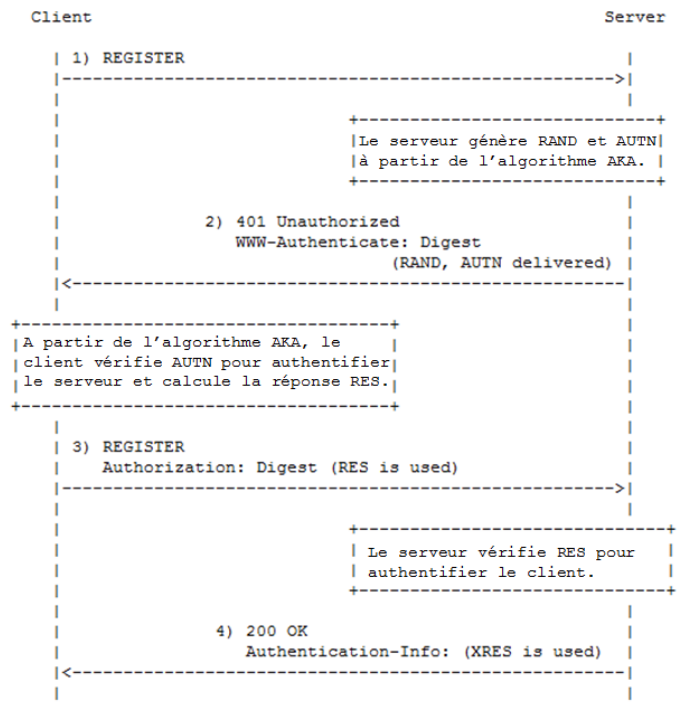


Figure 31. Authentification AKA appliquée à SIP [RFC3310]

[RFC3310] a spécifié comment modifier le message HTTP Digest en s'inspirant de la méthode AKA pour renforcer l'authentification de SIP. La figure 31 décrit le principe. Après le REGISTER, le serveur génère la valeur AUTN à partir du mot de passe selon les spécifications de [3GPP]. Le client vérifie que la valeur de AUTN a bien été construite à partir du secret et calcule la réponse à partir du mot de passe selon le même principe que la méthode HTTP Digest. Le client envoie RES, qui est le résultat du calcul. Le serveur vérifie RES en comparant avec XRES la valeur calculée par le serveur avec les éléments reçus du client. SI RES=XRES, le client est authentifié.

Les limitations sont sensiblement les mêmes que pour HTTP Digest, comme les attaques force brute ou l'usurpation d'identité (cf. le chapitre sur les considérations de sécurité de [RFC3310]). Concernant ce dernier point, il est effectivement possible pour l'attaquant de rejouer AUTN et RAND pour usurper la place du Proxy ou du Registrar. C'est pour cette raison que [RFC3310] préconise que le client vérifie si la séquence a été déjà jouée.

4.1.3.2. Les propositions modifiant les architectures SIP

4.1.3.2.1. L'utilisation des mots de passe à usage unique

Les mots de passe à usage unique ont été développés dans les années 80 [LAM81] pour éviter l'usurpation d'identité. Certains mécanismes d'authentification faisaient transiter les mots de passe en clair, une simple écoute suffisait donc pour récolter l'identité et le secret associé. [RFC1760] est le premier standard de l'IETF décrivant un mécanisme qui génère des mots de passe utilisables une seule fois. Si ces derniers sont interceptés, leur usage étant limité à une unique fois l'attaquant ne peut donc pas s'en servir : la récupération du mot de passe par l'attaquant lors de la transaction en cours n'a plus aucune utilité. C'est pourquoi on parle de « mot de passe à usage unique ».

L'utilisation des mots de passe à usage unique pour améliorer la sécurité de SIP a été explorée. [MIZ05] [HAL07] ont principalement cherché à renforcer l'authentification HTTP Digest en utilisant un mot de passe fourni par un canal tiers. Le mot de passe pour le challenge HTTP Digest [§ 3.4.1.1.] est fourni par l'intermédiaire d'un téléphone portable comme dans la figure 32. Ainsi l'emploi du mot de passe à usage unique permet de l'utiliser en toute sécurité sur un PC publique sans risque de subir une usurpation d'identité.

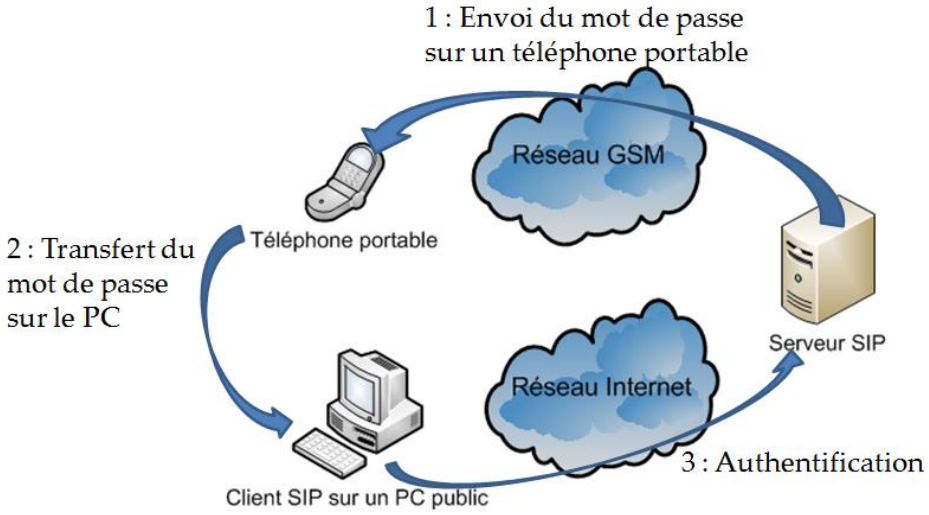


Figure 32. Utilisation d'un mot de passe à usage unique dans SIP via un téléphone portable.

Cette authentification utilisant plusieurs canaux n'intègre pas de mécanisme ou de solution OTP directement dans la signalisation SIP. Ce domaine n'a pas encore été exploré avant nos recherches.

4.1.3.2.2. Serveur RADIUS dans une architecture SIP

Le protocole RADIUS [RFC2865] est usuel pour les fonctions AAA[23] (Authorization Authentication Accounting). [RFC4590] formalise l'intégration de cette solution dans une architecture SIP. Le principe est de faire des serveurs SIP (Registrar ou Proxy) un client RADIUS afin de renvoyer les éléments de l'authentification du client SIP vers un serveur RADIUS (cf. figure 33).

[23] AAA : Les fonctions Authorization Authentication Accounting permettent de contrôler l'identité, l'accès aux services et de comptabiliser la consommation.

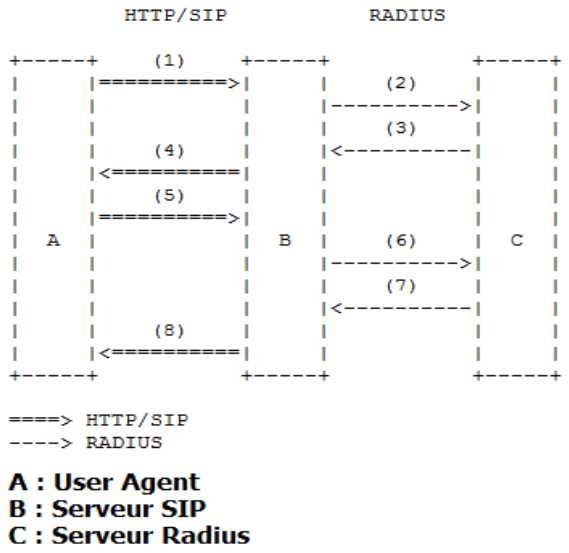


Figure 33. Architecture SIP avec un serveur RADIUS

Le détail des échanges illustrés par la figure 33 est le suivant :

(1) : A envoie une requête à son serveur SIP ;
(2) : B envoie une requête RADIUS Access Request vers C ;
(3) : C génère un nonce comme dans HTTP Digest et l'envoie vers B dans une requête Access Challenge ;
(4) : B envoie un message 407 ou 401 [§ 3.4.1.1.] avec le nonce généré par C ;
(5) : A renvoie sa requête en calculant Response à partir du nonce (similaire à authentification HTTP Digest) ;
(6) : B envoie un Access Request à C avec Response.
(7) : C vérifie Response et envoie une requête :
    a. Access Accept si la réponse est conforme ;
    b. ou Access Reject si la réponse est erronée ;
(8) : B envoie à A un 200 OK si C a envoyé un Access Accept sinon B envoie une réponse 401.

Cette architecture permet de centraliser les informations des usagers. Le serveur RADIUS possède donc tous les éléments nécessaires pour authentifier un usager comme son identité et le secret associé. Le contrôle du challenge/réponse est reporté au serveur RADIUS. Cette solution oblige à ajouter un élément dans l'architecture qu'il faudra également protéger.

4.1.3.2.3. « Single-Sign-On » dans une architecture SIP

La dernière contribution n'est pas un renforcement proprement dit de la sécurité de SIP mais prend en compte la multiplication des authentifications dans l'environnement des systèmes d'information actuels. L'idée fondatrice de l'authentification SSO (Single Sign On) provient du constat que chaque utilisateur doit s'authentifier auprès de chaque serveur pour utiliser les applications auxquelles il veut avoir accès. Cela présente certains inconvénients, dont les principaux sont le manque d'ergonomie et la manipulation malaisée de plusieurs mots de passe pour les authentifications. SSO offre donc un système d'authentification unique, commun à plusieurs serveurs, qui permet de simplifier et d'unifier les phases d'authentification des utilisateurs.

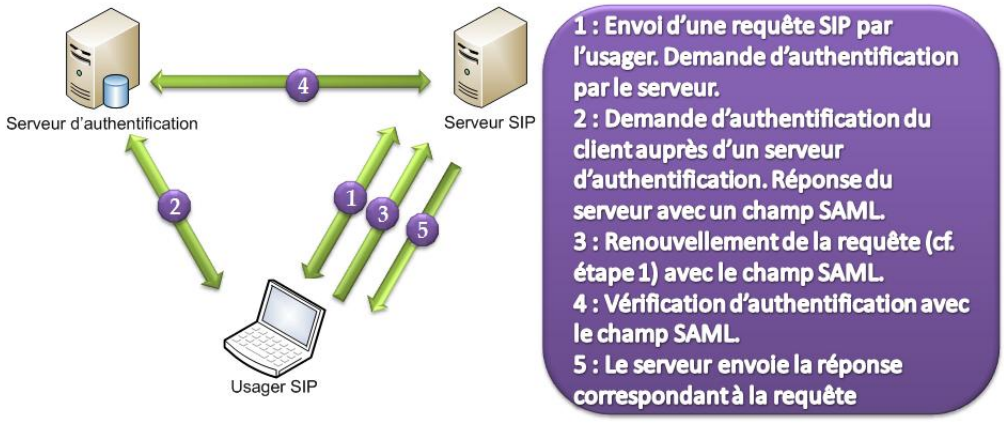


Figure 34. Application de l'authentification SSO dans le contexte SIP

Des contributions [NIE09] [TSC06] ont donc étudié les modalités d'intégrer une solution SSO dans les architecture SIP. SAML (Security assertion markup language) est à la base de ces propositions. SAML basé sur le langage XML[24] définit un protocole pour échanger des informations liées à la sécurité, principalement dans les applications SSO. [NIE09] est une des propositions pour intégrer la solution SSO de Liberty Alliance dans un contexte SIP. Le principe est décrit dans la figure 34. L'authentification est assurée par un serveur centralisant les demandes de toutes les applications. Pour [TSC06], c'est le serveur SSO qui centralise les demandes. La problématique de l'authentification SIP n'est alors plus une problématique de la téléphonie mais du système d'information.

4.1.3.3. Motivations

Les solutions pour sécuriser la signalisation SIP existent mais les différentes configurations pour avoir une sécurité robuste ne sont pas standardisées. Un opérateur ou un administrateur a toute latitude pour déployer ou non HTTP Digest, S/MIME ou encore TLS. Ce constat entraine de nombreuses lacunes dans le domaine de l'authentification. Comme cela a été précisé dans le chapitre 3, les limitations sont nombreuses : difficulté d'utiliser les certificats par absence de PKI, problème d'interopérabilité, solution liée à un choix de protocole dans les couches supports (niveau transport ou routage), difficulté d'échanger des clés symétriques, absence de politique de sécurité universelle. Ce constat nous a donc conduit à définir le cahier des charges suivant pour permettre à une solution de sécurité pour la signalisation SIP d'émerger rapidement :

- ne pas modifier le protocole SIP ;
- ne pas ajouter de nouveaux champs ;
- renforcer les mécanismes classiques présents.

Avec de telles contraintes, les évolutions restent possibles pour renforcer l'authentification. Les en-têtes SIP comportent en effet des valeurs dites « opaque » qui

[24] XML : Extensible Markup Language (XML) est un langage informatique générique. Il sert essentiellement à (re)présenter/structurer des données de type texte Unicode structurées en champs.

sont principalement générées de manière aléatoire pour le suivi des appels. Il a donc été envisagé d'utiliser ces champs en leur donnant une sémantique et ainsi une propriété de sécurité. Notre analyse a ainsi permis d'identifier plusieurs applications. Le détournement de ces champs a été inspiré par les contributions à SIP dans le domaine de stéganographie.

4.1.3.4. La stéganographie appliquée à SIP

Les solutions de sécurité présentées auparavant reposent sur des techniques basées sur la cryptographie. Mais il existe une autre manière d'envisager la sécurité, c'est la stéganographie. Il n'existe pas de solutions de sécurité usuelle en téléphonie basée sur cette technique. Néanmoins le principe et quelques applications seront présentés par la suite car ils ont inspirés les contributions de cette thèse.

La stéganographie est un mot élaboré à partir des mots grecs *steganos*, voulant dire couvert, et *graphein*, écriture. Depuis l'antiquité des formes diverses de stéganographie ont été utilisées pour préserver un secret. [SIN99] relate la manière dont le grec Histaîas envoya un message sur le principe de la dissimulation. Il rasa la tête de son messager et écrivit son message sur le crâne. Une fois les cheveux repoussés, le messager partit. Nous verrons dans ce chapitre comment notre contribution est passée de la tête d'une estafette grecque à l'en-tête des messages SIP.

La stéganographie est déjà utilisée dans le monde des images numériques : on y modifie des pixels, et les logiciels sont capables d'en extraire l'information. La stéganographie consiste à cacher une information en la noyant dans une masse de données plus grande, et non en la chiffrant. Alors que la cryptographie est l'art du secret, la stéganographie est l'art de la dissimulation. C'est ainsi que des travaux ont cherché à cacher dans les messages SIP des informations.

Dissimuler des messages secrets dans un canal voix en VoIP. C'est possible via la stéganographie. C'est ce qu'ont démontré des chercheurs polonais. Leurs travaux [MAZ06-1] [MAZ06-2] [MAZ07] [MAZ08-1] [MAZ08-2] ont proposés des applications pour la VoIP et pour SIP. Ils ont établi un certain nombre de possibilités pour créer des canaux cachés dans la signalisation SIP et dans les paquets voix RTP. Leurs travaux envisagent également des propriétés de sécurité en marquant les paquets voix.

L'application la plus pertinente vis-à-vis de nos recherches est l'utilisation de certains champs des en-têtes SIP. Ces derniers sont généralement calculés de manière aléatoire et n'ont pas de sens particuliers. Ce sont ces champs dit « opaques » (cf. [RFC3261]) que les chercheurs polonais ont utilisés pour créer un canal caché[25]. Les informations transmises sont insérées dans les champs comme Call-ID, tag, branch, ou encore Cseq. Celui qui

[25] « Un canal caché est un chemin de communication qui n'a pas été initialement prévu et/ou qui n'est pas autorisé pour transférer de l'information et qui, par conséquent, viole la politique de sécurité mise en place. Signalons qu'un canal caché nécessite donc au moins deux intervenants : celui qui donne les informations et celui qui les reçoit. » [RAY03].

reçoit collecte les informations dans les champs prévus par le protocole et reconstruit le message. Cette méthode est complètement transparente pour le réseau. C'est ce principe qui sera utilisé par la suite pour apporter de nouvelles propriétés de sécurité de manière transparente pour le protocole SIP. Les travaux cités dans le paragraphe ont montré que l'on pouvait modifier la génération de certains champs sans modifier le fonctionnement des appels. Notre approche vise ainsi à donner une sémantique aux valeurs aléatoires dans un contexte SIP pour améliorer l'authentification

4.2. Un complément au protocole d'authentification HTTP Digest SIP

Comme cela était présenté dans le paragraphe 4.1.1., l'authentification HTTP Digest permet à un *proxy* ou *registrar* SIP d'authentifier un usager avec un challenge/réponse. L'authentification est alors incluse dans la syntaxe des messages SIP. Au message d'enregistrement REGISTER du client, le serveur répond par le message suivant en insérant le champ « WWW-Authenticate » qui contient le challenge sous la forme du « nonce » (cf. figure 35).

```
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
Max-Forwards: 70
Contact: <sip:ahmed@137.194.192.228>
WWW-Authenticate: Digest realm="asterisk", nonce="64d45b88"
Content-Length: 0
```

Figure 35. Exemple de message « 401 Unauthorized » SIP

Le client forge alors la réponse en appliquant la formule [§ 3.4.1.1.] qui est la suivante :

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

Le client renvoie un REGISTER avec un champ « Authorization » et la valeur « response » (cf. figure 36) :

```
REGISTER sip:enst.fr SIP/2.0
Via: SIP/2.0/UDP 137.194.192.237:5060
From: <sip:ahmed@enst.fr>
To: <sip:ahmed@enst.fr>
Contact: "Serhrouchni" <sip:ahmed@137.194.192.237:5060>
Call-ID: D8A5240D579C4D6E8CE1@enst.fr
CSeq: 7169 REGISTER
Expires: 500
Authorization: Digest
username="ahmed",realm="asterisk",nonce="64d45b88", re-
sponse="1176420421871cdd89166a3e869d0841",uri="sip:enst.fr"
User-Agent: X-Lite build 1059
Content-Length: 0
```

Figure 36. Exemple de message REGISTER du protocole SIP

Notre cahier des charges peut donc s'appliquer à modifier la génération du « nonce » pour lui donner une sémantique. Ce dernier est généré de manière aléatoire et opaque selon les spécifications de [RFC3261]. Notre proposition pour l'élaboration du « nonce » est la suivante :

$$nonce = H(H(username||realm||password)||callid - value)\,.$$

La première partie du calcul reprend une partie de la sémantique de « response ». Par ailleurs le champ « realm » doit devenir une valeur aléatoire pour éviter les problèmes de rejeu, nous nous différencions ainsi de la proposition d'authentification SIP basée sur AKA [§4.1.3.4.]. En effet, si le nonce dépend uniquement du Call-Id, un pirate pourrait intercepter cette valeur et la rejouer, connaissant ainsi déjà la valeur « response ». Par défaut, la fonction de hachage est MD5 comme pour une authentification HTTP Digest SIP. L'authentification du serveur par le client est obtenue par la vérification du nonce de la même manière que le serveur vérifie la valeur « response ». La figure 37 résume la proposition.

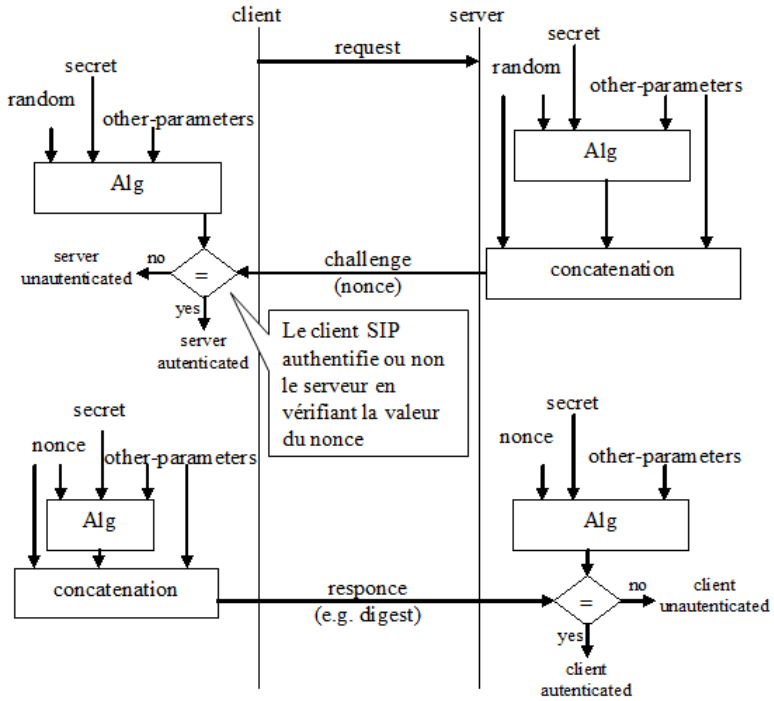


Figure 37. Principe du renforcement de l'authentification HTTP Digest SIP

Cette approche est valable pour toutes les requêtes SIP nécessitant une authentification HTTP Digest. Appliqué à cet exemple (figures 35 et 36), le nonce devient :

$$nonce = H(H(ahmed|asterisk|secret)|D8A5240D579C4D6E8CE1@enst.fr)$$

$$nonce = fa4112d21f9ba263c6fd197e6850f6c4$$

Pour faciliter le déploiement, l'authentification du serveur doit être envisagée comme un service. La réussite ou l'échec de la procédure doivent être remontés au client sous forme d'une information. Ce choix permet de rendre complètement compatible les équipements avec ou sans la solution de renforcement de l'authentification HTTP Digest.

L'intégrité des messages SIP est une problématique délicate comme le souligne [RFC3261] dans son chapitre sur l'altération des corps de message. En effet, des serveurs intermédiaires ont la possibilité de rajouter des champs. La protection en intégrité ne peut donc qu'être partielle, sachant que le chiffrement complet n'est pas possible car il

limiterait les possibilités d'acheminement. Les solutions d'intégrité prévues par le RFC reposent sur S/MIME pour les en-têtes ou sont reportées dans les couches basses. Notre contribution peut être étendue à l'intégrité de certains champs comme dans la contribution [GEN08] en ajoutant les champs à vérifier dans le calcul du nonce. Cette question n'est cependant pas traitée.

4.3. Une authentification optimisée par les mots de passe à usage unique

4.3.1. *Les différents usages des mots de passe à usage unique*

Les mots de passe sont utilisés dans les protocoles d'authentification. Les faiblesses de ces derniers sont connues [SAW06] [GUP07], comme la transmission en clair des mots de passe ou encore l'utilisation de machine non maîtrisée comme celle des cybercafés. Concernant les réseaux, une des principales attaques est l'écoute de la ligne. Tous les couples identifiant/mot de passe envoyés par un terminal vers un serveur peuvent ainsi être interceptés par un analyseur réseau, puis utilisés par un attaquant. Les mots de passe peuvent être en clair ou chiffrés, cela n'empêche pas les rejeux. C'est ce qui a motivé en particulier le développement des mots de passe à usage unique [RFC2289] [RFC1760]. D'autres facteurs favorisent également cette approche :

- Utilisation de mot de passe trivial ;
- Mot de passe inscrit sur un papier à côté d'un PC ;
- Un même mot de passe pour plusieurs applications.

Les mécanismes OTP repose sur trois éléments comme l'illustre la figure 38 : une fonction cryptographique (en général une fonction de hachage), un secret et un compteur ou challenge. Le secret n'est et ne doit jamais être transmis sur le réseau dans le cadre des échanges du protocole OTP. Il existe ainsi deux types de mécanisme OTP :

- Synchrone dans le cas de l'utilisation d'un compteur ;
- Asynchrone dans le cas de l'utilisation d'un challenge.

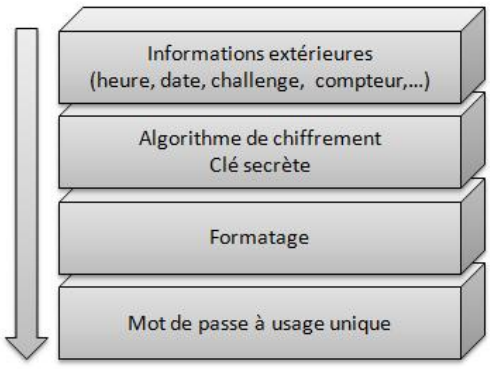


Figure 38. Processus de génération d'un mot de passe à usage unique

La première famille d'OTP est celle des mots de passe à usage unique asynchrone. « Asynchrone » caractérise le fait que le serveur va envoyer un challenge à l'usager pour lui permettre de générer l'OTP (cf. figure 39). Cela sous-entend que l'usager sollicite le

challenge. En terme de sécurité, cela signifie que pour un challenge donné, la réponse est toujours la même. La robustesse du système ne dépend donc que de la non-réutilisation du challenge. Il faut donc prévoir des challenges assez longs pour qu'un attaquant ne puisse pas réutiliser les précédents résultats.

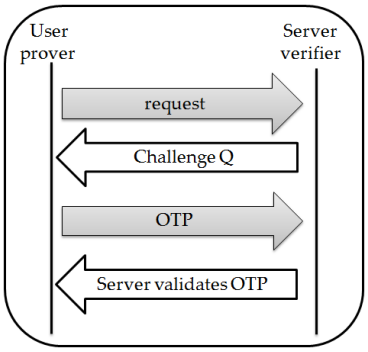


Figure 39. Authentification OTP asynchrone

La deuxième famille d'OTP est celle des mots de passe à usage unique synchrone. Contrairement aux OTP asynchrones, il n'y a pas de challenge comme le montre la figure 40. Ainsi, il suffit de générer l'OTP et de le transmettre directement au serveur. Il existe plusieurs manières de générer un OTP synchrone comme HOTP (An HMAC-Based One-Time Password Algorithm) [RFC4226]. C'est cette solution que est retenue par la suite.

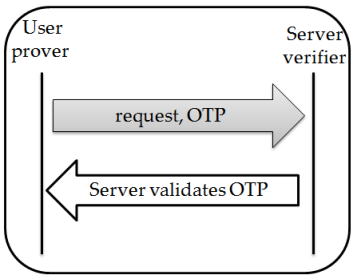


Figure 40. Authentification OTP asynchrone.

4.3.2. *HMAC-Based One-Time Password Algorithm*

HMAC-Based One-Time Password Algorithm (HOTP) est issue du RFC 4226. HOTP a pour objectif de définir un algorithme OTP basé sur une fonction d'HMAC[26], elle même basée sur SHA-1[27]. HOTP spécifie un compteur (C) s'incrémentant et une clé secrète (K) partagée entre le serveur et le client. Comme la sortie de SHA-1 est de 20 octets, elle sera tronquée pour obtenir une séquence 6 chiffres (facilement mémorisable par un usager).

[26] HMAC : Un HMAC (keyed-hash message authentication code) permet une authentification d'un message en appliquant une fonction de hachage cryptographique à la combinaison du message et d'une clé secrète.
[27] SHA1 : SHA1 (Secure Hash Algorithm) est une fonction de hachage cryptographique conçue par la National Security Agency (NSA), et publiée comme un standard fédéral pour le traitement de l'information par le NIST. Elle produit un résultat sur 20 octets.

$$HMAC_K(m) = h((K \oplus opad)||h((K \oplus ipad)||m))$$

Figure 41. Fonction HMAC

Il faut commencer par calculer le HMAC-SHA-1 sans troncature conformément à [RFC2104]. Pour rappel le HMAC-SHA-1 se calcule selon la formule de la figure 41 à partir de la clé secrète K et du compteur C sur 8 octets : h est la fonction de hachage, m le message, opad et ipad sont des blocs définis par avance. La deuxième phase est le calcul de l'OTP obtenu par troncature du $HMAC_K$ pour obtenir un résultat de 6 chiffres.

Il arrive que le client et le serveur se désynchronisent, par exemple si l'utilisateur demande des générations d'OTP sans les faire valider par le serveur. Afin de palier à ces désynchronisations, HOTP prévoit un paramètre N appelé « look-ahead window » qui défini le nombre d'OTP successifs à générer et à tester du coté du serveur. Si un des N mots de passe OTP est correct, le compteur sera alors incrémenté en conséquence coté serveur. Celui-ci sera alors de nouveau synchronisé avec le compteur de l'utilisateur.

4.3.3. *Optimisation de l'authentification avec les mots de passe à usage unique*

Nous avons également recherché les opportunités d'utiliser les mots de passe à usage unique dans le contexte de SIP. L'intégration peut s'envisager de plusieurs manières. La premières consiste à utiliser HOTP dans HTTP Digest pour générer le secret utilisé dans le challenge/réponse comme l'illustre la figure 43. Cette solution permet d'éviter les attaques par dictionnaire ou par force brute puisque le mot de passe change à chaque authentification.

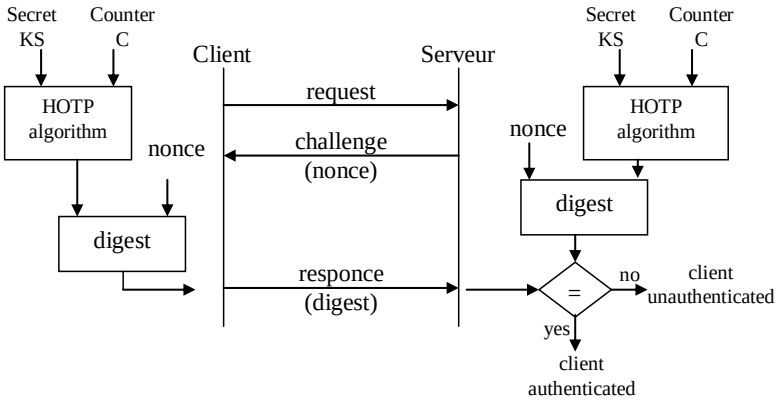


Figure 42. Utilisation d'HOTP dans le calcul du challenge/réponse HTTP Digest

Une autre piste envisagée a été d'intégrer une authentification basée sur HOTP dans le Call-ID comme le montre la figure 43. Cette option permet au serveur d'authentifier de manière implicite l'usager en interprétant le Call-ID.

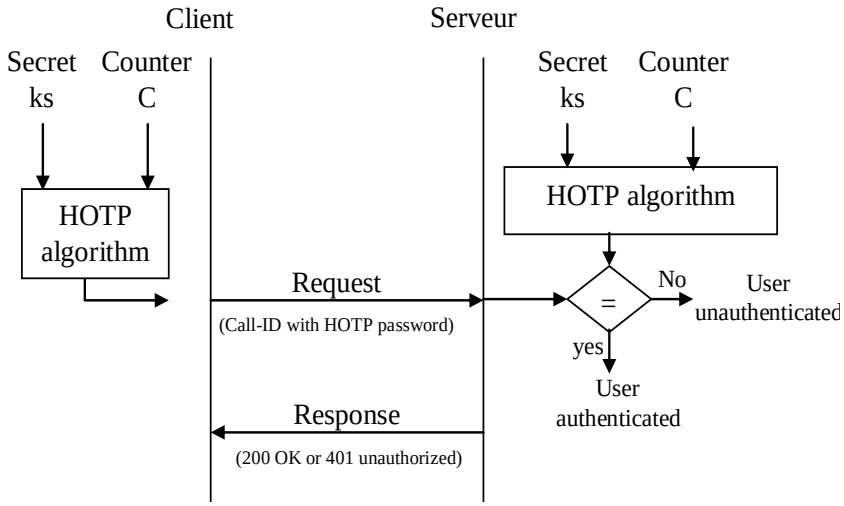


Figure 43. Intégration dans la signalisation d'HOTP dans l'authentification SIP

L'enregistrement et l'authentification du client sont réalisés avec deux messages (cf. figure 44) contre quatre messages avec HTTP Digest. Cette méthode permet d'optimiser les échanges. Ce principe vaut également pour les messages INVITE.

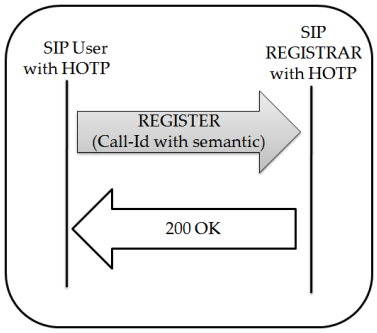


Figure 44. Authentification HOTP intégrée dans le Call-Id de SIP

Cette solution consiste à rendre l'authentification implicite par l'usage du mot de passe à usage unique synchrone. Le renforcement de l'authentification induit également une réduction des échanges pendant les enregistrements et l'initiation d'un appel.

4.4. Une solution pour le déni de service

Cette contribution concerne plus particulièrement le déni de service. Comme cela a été illustré dans [§3.3.2.1.], en absence d'authentification un attaquant peut interrompre une session en forgeant un message BYE frauduleux qui interrompt la session concernée. Si le BYE est envoyé à un serveur, ce dernier pourrait envoyer un défi avec le mécanisme HTTP Digest, mais c'est rarement le cas. De plus l'attaque est également réalisable directement sur les usagers.

Pour limiter ce déni de service, nous nous sommes intéressés au paramètre « branch » contenu dans le champ « via ». Ce dernier doit commencer par la série « z9hG4bK » suivie d'une valeur aléatoire. Une sémantique a donc été donnée à ce champ pour permettre au

proxy et au client de vérifier l'identité de l'émetteur du BYE. Le calcul est détaillé dans la figure 45.

$$Branch = z9hG4bK||F(H(Call - Id||password||From||To)$$

- *|| : correspond à une fonction de concaténation ;*
- *H : H est par défaut la fonction de hachage MD5 [RFC1321] conformément à [RFC3261] ;*
- *F : F est la fonction HOTP [RFC4226] qui permet de passer de 20 octets à une séquence de 6 chiffres. Le secret est celui fourni avec l'identifiant, et le compteur est une valeur fixe.*

Figure 45. Champ « Branch » avec sémantique

Désormais quand un proxy reçoit un BYE, il vérifie le champ « branch » pour authentifier l'émetteur ; évidemment si c'est un usager de son domaine avec lequel il partage un secret (cf. figure 46). Le client peut également vérifier l'origine du BYE en implémentant la partie comparaison. L'attaque reste possible en prenant l'identité d'un usager n'appartenant pas au domaine.

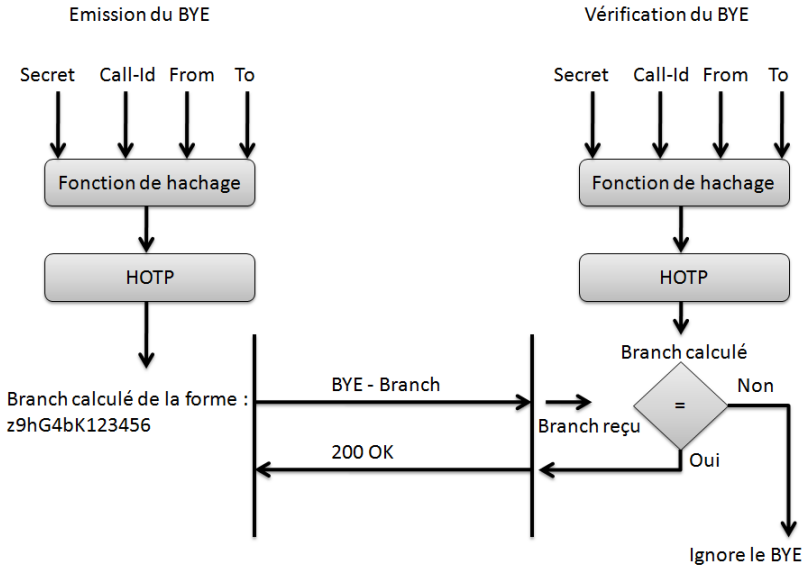


Figure 46. Solution au Dos par BYE frauduleux

Contrairement aux autres contributions qui intervenaient dans les premiers échanges d'une session, cette solution intervient à la fin. Les paramètres utilisés dans le calcul du « branch » ne font intervenir que des paramètres présents dans le message BYE ce qui permet à un proxy « stateless[28] » d'implémenter cette solution.

[28] Proxy SIP : il existe deux types de proxy : « stateless » ou « stateful ». Les proxy stateless se contente de choisir la destination d'un message SIP mais en conserve pas aucun état de la session. Les proxy « stateful » conservent les états des sessions et notamment l'état d'une transaction en cours.

4.5. Conclusion

Ce chapitre a permis de définir des solutions palliatives aux problèmes de sécurité du protocole SIP. La notion de transparence qui a prévalu dans la conception des mécanismes permettrait un déploiement progressif sans difficultés s'ils étaient retenus dans une infrastructure SIP. Cependant le fait de s'appuyer sur le secret partagé entre le serveur et le client limite le domaine d'emploi. De plus nous n'avons pas traité le mode de diffusion des extensions supportées, ce qui optimiserait l'intégration de ces nouvelles solutions.

L'analyse des solutions concurrentes montre qu'il existe deux approches pour renforcer l'authentification dans un contexte SIP. La première solution consiste à modifier la syntaxe des messages, la deuxième à modifier l'architecture et la nature des échanges. Notre logique consistant à modifier la génération de certains champs et à définir un cadre d'interprétation peut donc être considérer comme une voie innovante pour la sécurité de SIP.

L'autre notion associée à ces contributions est le principe de service. La sécurité est proposée au client sous la forme d'une information pour l'authentification HTTP Digest renforcé. La garantie d'utiliser de manière sécurisée la téléphonie est reporté au niveau de l'usager. Cela peut être considéré comme une plus-value par rapport à l'existant. Mais cette approche permet également de faire émerger dans le paysage des usagers la notion de service de sécurité pour la téléphonie. Au même titre que l'on sécurise un achat sur Internet, il faut protéger son appel téléphonique en IP.

Concernant le renforcement de l'authentification HTTP Digest dans le contexte de SIP, le principe n'est ni lié à SIP, ni au mécanisme HTTP Digest. Toutes les authentifications simples basées sur un challenge/réponse peuvent bénéficier de cette approche. Nous l'avons formalisé dans un brevet [Annexe IV]. Enfin, le principe de donner une sémantique à un champ aléatoire pour renforcer la sécurité est complètement universel.

Le dernier axe d'amélioration a été l'optimisation des échanges avec l'utilisation des mots de passe à usage unique. L'authentification HTTP Digest nécessite 4 messages pour pouvoir initier un appel ou faire un enregistrement. Alors que les appels SIP sont de plus en plus nombreux, les propositions intégrant l'authentification dans la requête pourraient avoir des impacts sur le dimensionnement des infrastructures SIP. L'évaluation de ce gain pourrait faire l'objet de travaux ultérieurs.

CHAPITRE 5

VALIDATIONS DES SOLUTIONS DE SECURITE DU PROTOCOLE SIP

Ce chapitre est consacré aux validations et aux expérimentions des propositions de renforcement de l'authentification SIP. Des logiciels de téléphonie (autocommutateurs et softphones) ont été modifiés pour démontrer la faisabilité des solutions de sécurité basées sur la sémantique des valeurs « opaques » SIP. Les tests ont permis de vérifier la totale compatibilité des équipements modifiés avec ceux qui ne supportent pas nos extensions. L'ajout des mécanismes ne provoque ainsi aucune interruption de service. Une validation formelle avec le logiciel AVISPA a été également réalisée pour valider l'intérêt de renforcer HTTP Digest.

5. Validation des solutions de sécurité du protocole SIP

5.1. Méthodes et outils de validation

5.1.1. *Les méthodes*

A l'heure où les systèmes d'informations sont de plus en plus complexes et par là même plus vulnérables, le besoin d'assurer la sécurité s'est généralisé à tous les niveaux dans les réseaux [BOI07]. La sécurité des communications repose sur l'utilisation de fonctions mathématiques et sur l'emploi de protocoles, appelés « protocole de sécurité ». Ces derniers établissent les règles d'échanges entre les différents acteurs d'un ou des réseaux. C'est bien ces mécanismes que nous avons contribué à renforcer avec nos solutions. Pour s'assurer que ces dernières répondaient bien aux attentes, les trois contributions ont été validées soit pratiquement, soit formellement. Les deux méthodes retenues sont donc :

- la validation pratique basée sur l'implémentation et l'observation ;
- la validation formelle : cette méthode repose sur le modèle d'attaque de Dolev Yao [DOL83]. Pour ce fait, deux hypothèses fortes sont posées : les algorithmes de cryptologie sont considérés comme sûrs, et l'attaquant contrôle le réseau. Ce modèle est à la base du logiciel AVISPA [AVISPA] qui a été utilisé pour cette validation.

5.1.2. *AVISPA*

AVISPA (Automated Validation of Internet Security Protocols and Applications) est un projet européen dédié au développement de techniques de validation de protocoles de sécurité [ARM05]. Cet outil permet de mettre à disposition ces techniques aux ingénieurs en charge des développements : l'architecture d'AVISPA est présentée dans la figure 47.

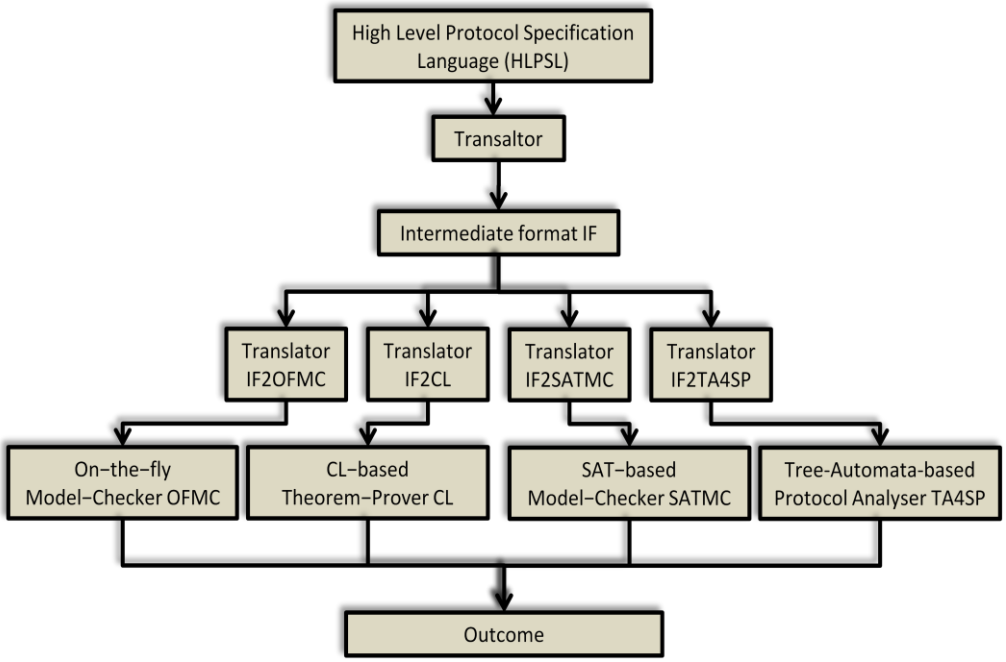


Figure 47. Architecture de fonctionnement d'AVISPA

L'outil AVISPA permet quatre méthodes d'analyse :

- **l'outil On-the-fly Model Checking (OFMC)** repose sur l'emploi de structures symbolisant plusieurs exécutions similaires d'un même protocole, réduisant ainsi le champ d'investigation ;
- **l'outil Constraint-Logic-based Attack Searcher (CLATSE)** utilisant des techniques classiques et spécifiques de résolution de contraintes, permet d'obtenir des résultats avec différents modèles de protocoles ;
- **l'outil SAT-based Model-Checker (SATMC)** consiste à savoir si une formule de logique propositionnelle (c'est-à-dire une formule construite avec les opérateurs et, ou, non) peut être vraie si l'on choisie convenablement les valeurs des variables.
- **l'outil Tree Automata based on Automatic Approximations for the Analysis of Security Protocols (TA4SP)** fournit un diagnostic moins précis que les trois autres outils mais sa spécificité est d'analyser les protocoles dans le cadre non borné du nombre de sessions.

Pour utiliser ces modules, il faut spécifier la solution de sécurité à valider dans un langage appelé HLPSL. AVISPA existe sous la forme d'un portail Web ou sous la forme d'applicatif comme [SPAN] qui a été utilisé dans les validations. L'analyse présentée plus tard s'appuie sur les modules OFMC et CLATSE qui selon [BOI07] apportent le niveau d'analyse correspondant à notre validation.

5.1.3. *Les équipements et les logiciels*

5.1.3.1. Description de la plate-forme

Pour les validations pratiques, une plate forme a été mise en place pour vérifier le bon fonctionnement des contributions envisagées dans ce mémoire. Elle a permis d'implémenter les différentes solutions et de dérouler les tests de validations. Le dispositif est décrit d'une manière générique dans la figure 48.

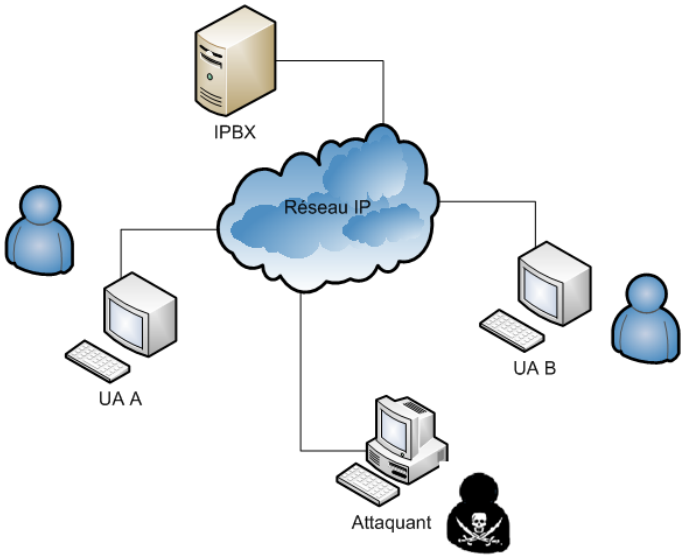


Figure 48. Plate-forme de validation

5.1.3.2. L'IPBX Asterisk

Asterisk [AST] est un logiciel de téléphonie open source conçue principalement pour fonctionner sous Linux. Ce logiciel permet de transformer la plupart des ordinateurs personnels en IPBX (IP PABX, Private Automatic Branch eXchnage). Ce logiciel créé en 1999 par Mark Spencer n'a cessé de se développer avec le concours de la communauté du logiciel libre développé en C sous Linux. Il permet à n'importe qui de créer et paramétrer son commutateur téléphonique. Asterisk comprend un nombre très élevé de fonctions permettant l'intégration complète pour répondre à la majorité des besoins en téléphonie (cf. figure 49). Il permet de remplacer totalement les équipements propriétaires.

Compatible avec les principaux protocoles de téléphonie sur IP comme SIP ou H323 et même Skype depuis peu, Asterisk permet la commutation des appels dans le monde IP. Il peut également jouer le rôle de passerelle avec les réseaux publics (RTC, GSM,…) en interfaçant sur les serveurs des cartes de commutation. La société Digium créé par le concepteur d'Asterisk commercialise principalement ces interfaces vers les réseaux de transports publics.

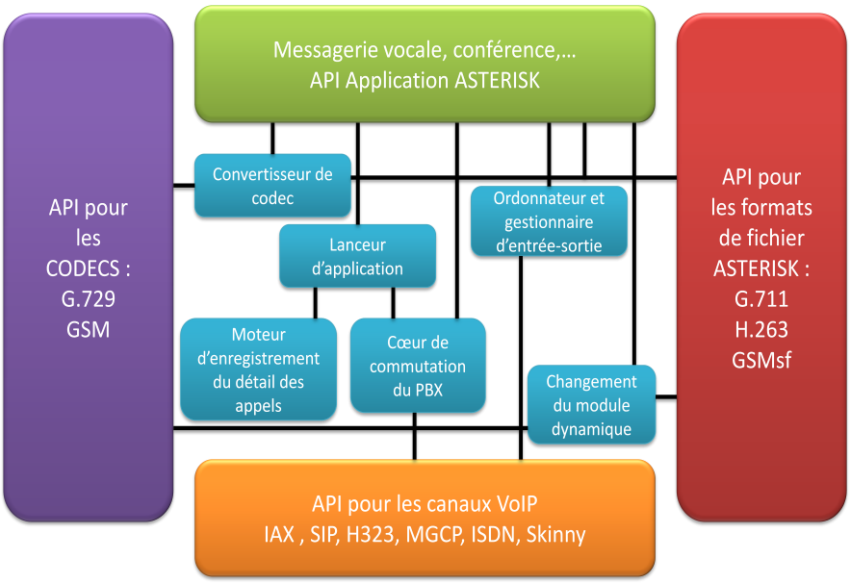


Figure 49. Schéma de principe d'ASTERISK

Asterisk peut être considéré comme la boite à outils de la téléphonie sur IP [MEG05]. C'est pour cette raison que ce logiciel a été étudié pour valider les contributions de cette thèse. Il offre une souplesse inimaginable et une multitude de services. Asterisk permet également de réaliser un domaine de téléphonie sur IP avec son propre plan d'adressage, ce qui est l'élément de base d'une validation dans le contexte de cette étude. Pour permettre les appels deux fichiers doivent être configurés « sip.conf » et « extension.conf » : le premier permet de créer des comptes SIP, le deuxième fichier contient le plan de numérotation du serveur. Des exemples de fichiers sont donnés dans l'annexe VI.

5.1.3.3. Les softphones

Deux softphones ont été utilisés sur la plate-forme de test. Tout d'abord pour implémenter les différentes solutions, nous avons utilisés TWINKLE [TWIN] qui est logiciel open source codé en C++ pour les OS Linux. L'autre application utilisée est 3CXPhone [3CX]. C'est un téléphone logiciel gratuit pour Microsoft Windows. 3CXPhone est doté d'une interface de type téléphone intelligent instantanément reconnaissable par les utilisateurs. Il a permis de tester les IPBX modifiés pour vérifier l'interopérabilité des contributions. Les interfaces graphiques sont données en annexe VII (cf. figure 72).

5.1.4. *Les outils d'audit et d'attaques*

Trois logiciels ont principalement été utilisés pour vérifier le bon fonctionnement des contributions en simulant des attaques :

- Wireshark : Wireshark [WIRE] est un analyseur de protocole (ou sniffer) qui examine les données à partir d'un réseau en direct ou à partir d'une capture stockée dans un fichier (cf. figure 73 en annexe VII).

- CommView : CommView [COMM] est un sniffer orienté téléphonie sur IP. Il permet en particulier de visualiser les échanges de messages pour chaque session (cf. figure 74 en annexe VII). Il permet également l'enregistrement des conversations téléphoniques.
- SIPNess : SIPNess est un outil basic qui permet de forger des messages SIP (cf. figure 76 en annexe VII). Il sera utilisé pour valider la solution qui limite le DoS par BYE frauduleux.

5.2. La validation formelle de l'authentification HTTP Digest SIP

5.2.1. *Rappel du principe*

Comme cela était présenté dans [§ 4.2.], la proposition d'amélioration de l'authentification HTTP Digest consiste à donner une sémantique au « nonce » pendant le challenge réponse qui permet à l'usager de s'authentifier auprès du serveur SIP. Le client peut désormais interpréter le « nonce » pour savoir si ce dernier a bien été construit par un élément légitime à partir de leur secret. Le « nonce » est construit de la manière suivante :

$$\text{nonce} = \text{H}(\text{H}(\text{username}||\text{realm}||\text{password})||\text{callid} - \text{value}).$$

5.2.2. *Validation avec AVISPA*

Comme cela a été précisé dans la présentation d'AVISPA pour vérifier formellement un protocole il faut le spécifier dans un langage. Nous nous sommes inspirés des travaux de [HAG08] pour formaliser HTTP Digest SIP et notre contribution en HLPSL. Les descriptions HLPSL sont fournies en annexe IV, elles ont été réalisées avec l'outil SPAN (Security Protocol ANimator for AVISPA) [SPAN] (cf. figure 75 en annexe VII).

Nous allons dérouler la validation avec le module OFMC comme cela était précisée précédemment. Dans un premier temps, l'authentification HTTP digest SIP est analysée. La figure 50 illustre les principes du challenge/réponse.

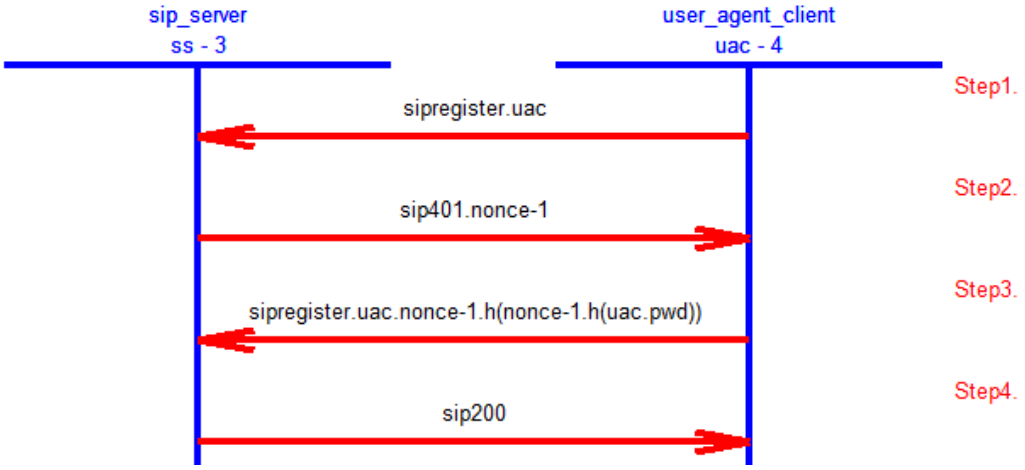


Figure 50. Authentification HTTP Digest SIP visualisée avec SPAN

L'analyse avec le module OFMC donne le résultat de la figure 51.

```
SUMMARY
UNSAFE
DETAILS
ATTACK_FOUND
PROTOCOL
HTTP DIGEST
GOAL
authentication_on_y
BACKEND
OFMC
COMMENTS
STATISTICS
parseTime: 0.01s
searchTime: 0.01s
visitedNodes: 5 nodes
depth: 3 plies
ATTACK TRACE
i -> (ss,3): sipregister.uac
(ss,3) -> i: sip401.Nonce(1)
i -> (uac,3): start
(uac,3) -> i: sipregister.uac
i -> (uac,3): sip401.Nonce(1)
(uac,3) -> i: sipregister.uac.Nonce(1).h(Nonce(1).h(uac.pwd))
i -> (ss,3): sipregister.uac.Nonce(1).h(Nonce(1).h(uac.pwd))
(ss,3) -> i: sip200
```

Figure 51. Résultats d'une analyse avec AVISPA

Un contrôle avec OFMC qualifie ainsi d'« UNSAFE » HTTP Digest, on retrouve l'attaque de l'homme au milieu. SPAN permet de visualiser l'attaque comme le montre la figure 52. Nos résultats rejoignent ceux de [HAG08]. Hagalisletto et al. rappellent également toutes les usurpations possibles liées à la faiblesse de l'authentification HTTP Digest entraînant les détournement d'appels ou d'enregistrement.

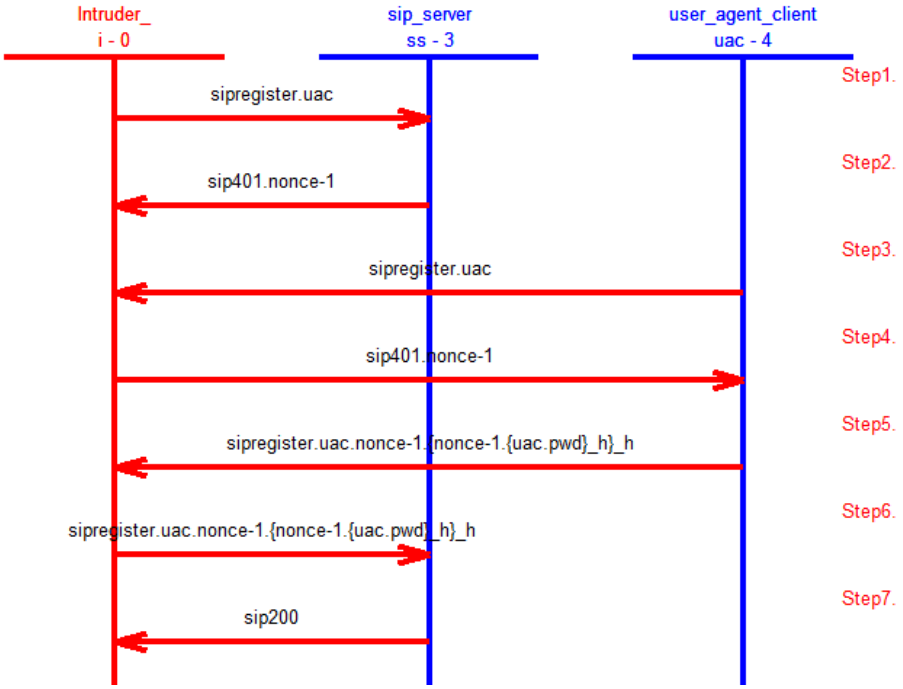


Figure 52. Attaque élaborée sur HTTP Digest SIP construite avec AVISPA

L'attaque ne correspond pas forcément à une attaque réaliste dans le contexte SIP comme le montre l'analyse de la figure 52. L'usurpation d'un serveur SIP ne nécessite pas d'interaction avec le serveur légitime, néanmoins on voit bien que l'intrus peut prendre sa place.

La même démarche a été appliquée à l'authentification renforcée. La figure 53 illustre la contribution avec le logiciel SPAN.

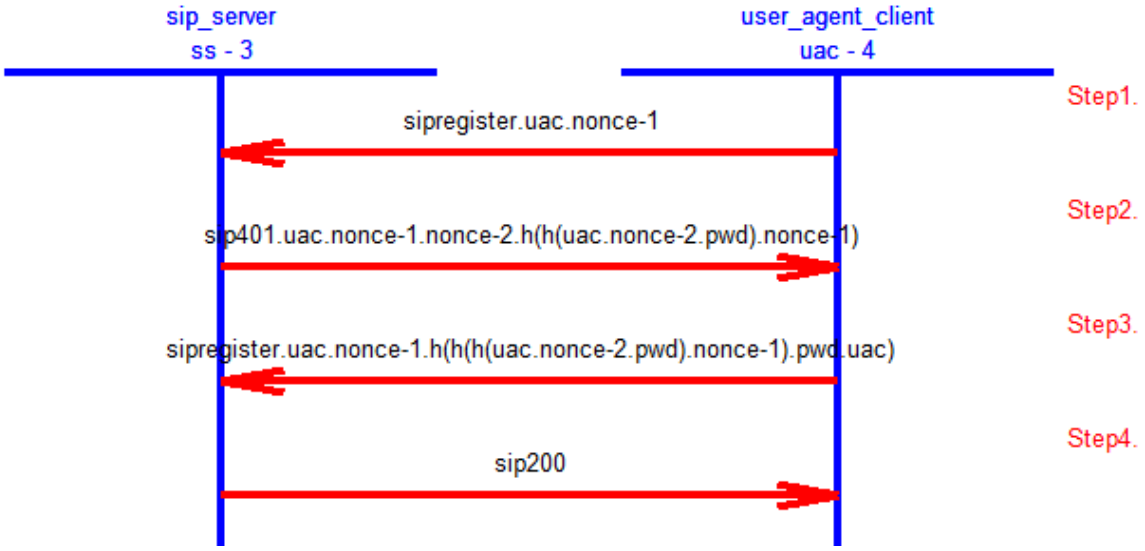


Figure 53. Authentification HTTP renforcée visualisée avec SPAN

L'analyse avec OFMC donne le résultat de la figure 54 :

```
SUMMARY
SAFE
DETAILS
BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
SIP MUTUAL AUTHENTICATION
GOAL
as_specified
BACKEND
OFMC
COMMENTS
STATISTICS
parseTime: 0.01s
searchTime: 0.01s
visitedNodes: 4 nodes
depth: 2 plies
```

Figure 54. Résultats de l'analyse de notre contribution avec AVISPA

Bien que prévisible, le résultat confirme, de manière formelle, l'état de l'art sur l'authentification HTTP Digest et la portée de notre proposition. La validation formelle a permis de bien fixer les propriétés de chaque variable.

Un contrôle avec le module OFMC (même résultats avec le module CLASTE) confirme notre analyse sur l'apport du renforcement de l'authentification HTTP Digest en la

qualifiant de SAFE. L'authentification du serveur proposée par notre solution supprime l'attaque de « l'homme au milieu » présent l'authentification HTTP Digest. L'analyse du modèle ne prend pas en compte les éventuelles faiblesses des algorithmes cryptographiques.

5.3. Les validations sur plate-forme logicielle

Les validations pratiques ont été menées sur une plate-forme logicielle. Pour chaque solution, des modifications sur les logiciels Asterisk et Twinkle ont été apportées pour intégrer les mécanismes établis dans le chapitre 4. Trois réalisations ont été développées :

- **la première concerne le renforcement d'HTTP Digest.** Nous avons modifiées la génération du « nonce » dans l'IPBX Asterisk au niveau fichier « chan_sip.c » (gère le canal SIP d'Asterisk). Twinkle a également été modifié pour intégrer notre proposition et ainsi lui permettre d'interpréter le « nonce ». Les codes sont en annexe I. Pour Twinkle, deux fichiers ont été modifiés « phone_user.cpp » qui gère les enregistrements et « dialog.cpp » qui gère les sessions ;

- **la deuxième réalisation a consistée à intégrer l'HOTP pour permettre à l'usager de s'authentifier.** Nous avons modifié la génération du Call-ID dans le logiciel Twinkle. Le code HOTP (cf. annexe III) a été inséré dans deux fichiers « phone_user.cpp » pour authentifier un REGISTER et « dialog.cpp » pour authentifier les INVITE. L'intégration est identique dans les deux cas. Du côté Asterisk, « chan_sip.c » a été modifiée pour ajouter une condition sur la valeur du Call-ID ;

- **la troisième réalisation concerne le déni de service sur le BYE.** Toujours avec le même principe que précédemment nous avons modifiées « chan_sip.c » d'Asterisk et « dialog.cpp » de Twinkle.

Pour chaque contribution, une série de tests a été déroulée. Les résultats sont fournis dans le paragraphe suivant.

5.4. Résultats et analyse

5.4.1. *HTTP Digest SIP renforcé*

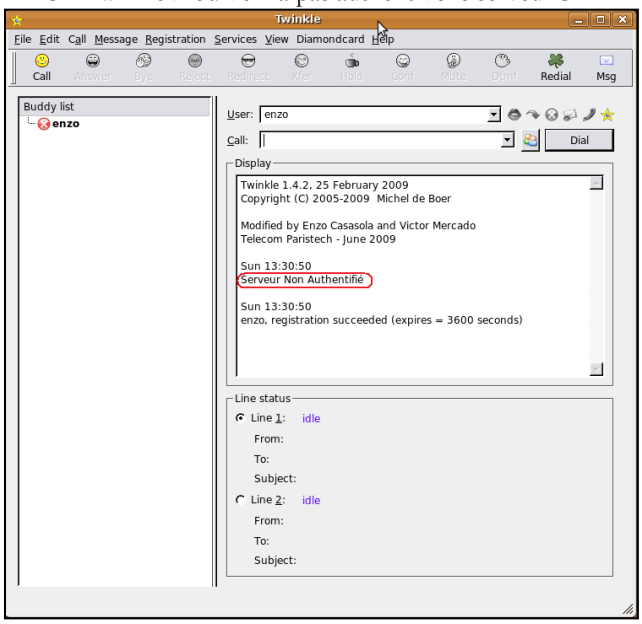
Le tableau 10 présente les résultats obtenus dans les différentes configurations. Les effets sont conformes aux prévisions, en particulier dans le cas où la solution n'est pas implémentée sur l'un des éléments. La conception de la solution fait que le service n'est pas interrompu en cas d'un serveur non authentifié.

Tableau 10. Résultat des expérimentations de l'authentification HTTP Digest renforcée

Client Normal	Client Modifié	IPBX Normal	IPBX Modifié	Comportements des équipements dans les différentes configurations possibles
✓		✓		Situation normale. Le client s'authentifie auprès du serveur.
✓			✓	L'IPBX génère un nonce avec sémantique mais le client ne l'exploite pas. Il s'authentifie néanmoins auprès du serveur en fournissant un « response » calculé à partir du nonce.
	✓	✓		L'IPBX génère un nonce de manière aléatoire. Le client constate que le nonce n'a pas de sémantique. Il est informé de cette situation par un message « server unauthenticated ». Le client fait alors le choix d'utiliser ou non l'IPBX pour téléphoner. Voir la figure 55.
	✓		✓	L'IPBX s'authentifie auprès du client en générant un nonce avec sémantique. Le client est informé par le message « server authenticated ». Ensuite il s'authentifie auprès du serveur en fournissant un « response » calculé à partir du nonce. Voir la figure 55.

✓ : équipements utilisés.

Un Twinkle modifié n'a pas authentifié le serveur SIP



Le Twinkle modifié a authentifié le serveur SIP

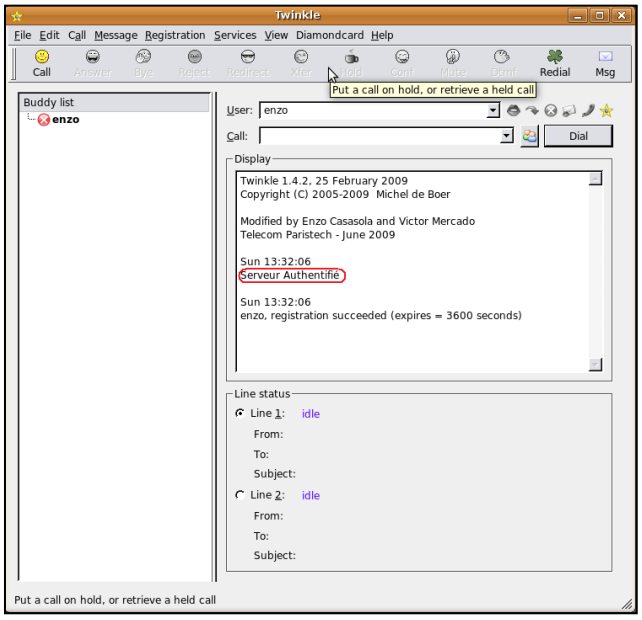


Figure 55. Twinkle modifié vérifiant la légitimité du serveur SIP

Dans notre implémentation, le choix a été fait de ne pas lier l'authentification du « nonce » avec l'authentification HTTP Digest. Dans le cas d'un serveur non authentifié, le client SIP a le choix ou non d'utiliser ce serveur pour établir des sessions. Il prend donc le risque de confier la signalisation à un serveur non légitime.

En complément des résultats du tableau 10, les validations ont été complétées par une évaluation du temps de calcul cryptographique des deux solutions. L'ordre de grandeur est de la milliseconde, ce qui est relativement petit devant le temps pour l'établissement

d'une session qui est de l'ordre de la seconde pour un appelant qui décroche immédiatement (temps observé pendant les essais). Le tableau 11 compare les deux solutions, l'augmentation du temps de calcul reste acceptable.

Tableau 11. Temps de calcul cryptographique pour HTTP Digest renforcé

Solutions	Nombre de calculs pour l'usager	Nombre de calculs pour le proxy	Nombre total de calculs	Temps de calcul sur un bloc de 64 octets pour la fonction MD5	Temps de calcul pour l'ensemble de l'authentification avec MD5
HTTP Digest	3 fonctions de hachage	3 fonctions de hachage	6 fonctions de hachage	Config. 1 : 1060 ns Config. 2 : 1460 ns	Config. 1 : 6360 ns Config. 2 : 8760 ns
HTTP Digest renforcée	6 fonctions de hachage	6 fonctions de hachage	12 fonctions de hachage	Config. 1 : 1060 ns Config. 2 : 1460 ns	Config. 1 : 12720 ns Config. 2 : 17520 ns
Remarques : Configuration 1 : Inter Core 2 Duo - 2.00 GHz - RAM 3 Go Configuration 2 : Intel Pentium 4 - 2,80 – RAM 512 Mo MD5 travaille sur des blocs de 64 octets ce qui correspond à une chaîne de 128 caractères. Cette taille est considéré comme représentative pour une application SIP. Les temps ont été calculés avec le fonction SPEED d'OpenSSL.					L'augmentation de calcul est de : Config. 1 : 6360 ns Config. 2 : 8760 ns

5.4.2. *Optimisation des échanges avec HOTP*

Comme cela était présenté dans [§4.3.], la proposition d'amélioration utilisant HOTP permet d'optimiser les échanges SIP au moment du REGISTER ou d'un INVITE. Le résultat issu du calcul HOTP est intégré dans le Call-ID comme le montre la figure 56. Cette option permet d'authentifier de manière implicite l'usager en lisant le Call-ID.

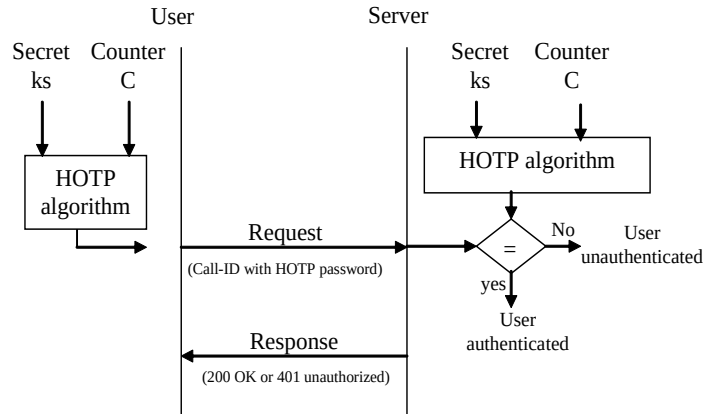


Figure 56. Intégration dans la signalisation d'HOTP dans l'authentification SIP

De la même manière que précédemment, les résultats obtenus avec l'intégration d'HOTP dans SIP ont été regroupés dans un tableau (cf. tableau 12). Dans la mesure où un élément SIP ne supporte pas la solution, les mécanismes traditionnels comme HTTP Digest reprennent le relais. La figure 57 présente un message SIP intégrant notre solution, l'INVITE est traité directement sans authentification par HTTP Digest : le Call-ID est généré selon notre processus.

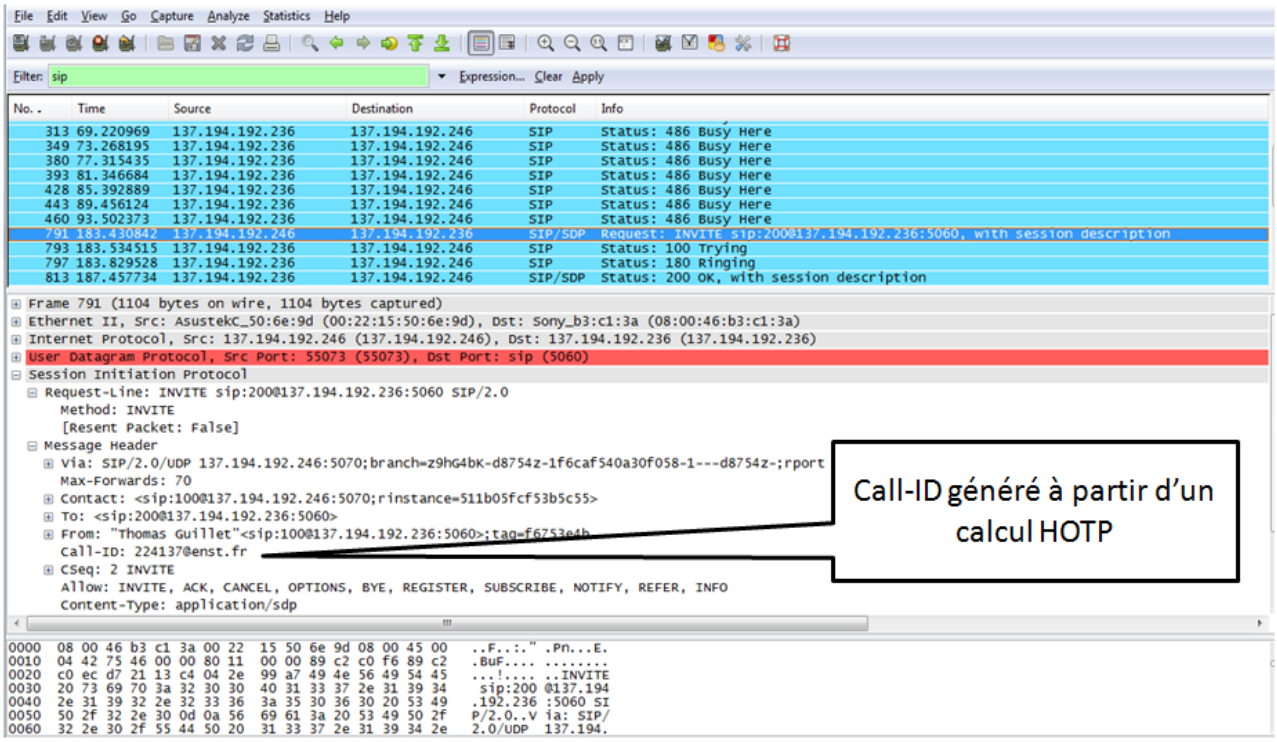


Figure 57. Message SIP contenant une authentification HOTP dans le Call-ID

Le tableau 12 reprend l'ensemble des résultats. Le déploiement de la solution n'entraîne aucun problème de fonctionnement. Dans la mesure où un élément SIP ne supporte pas la solution, les mécanismes traditionnels comme HTTP Digest reprennent le relais.

Tableau 12. Synthèse des résultats concernant l'authentification HOTP

Client normal	Client Modifié	IPBX normal	IPBX modifié	Résultats
✓		✓		L'UA est authentifié en 4 messages. Situation normale.
	✓	✓		L'UA est authentifié avec HTTP Digest car le serveur n'a pas su lire le Call-Id contenant le HOTP.
✓			✓	Le serveur vérifie la sémantique du Call-Id. Ce dernier n'ayant aucun sens, le serveur demande une authentification HTTP Digest.
	✓		✓	L'UA est authentifié par le serveur avec la sémantique du Call-Id en deux messages (cf. figure 56).

✓ : équipement utilisé

Pour évaluer le gain dans l'établissement d'une session, des mesures de temps ont été effectuées avec CommView. Le temps mesuré est celui entre l'émission du premier INVITE et la réception du Ringing ; nous ne prenons pas en compte le message 200 Ok pour faire abstraction de la réaction humaine puisque cette réponse est émise après l'acceptation de l'INVITE par l'appelé. La figure 58 illustre les mesures effectuées : la série 1 correspond à l'établissement d'une session avec une authentification normale HTTP Digest, la série 2 correspond à l'établissement d'une session avec une authentification HOTP. On observe un gain moyen de 214 ms par session.

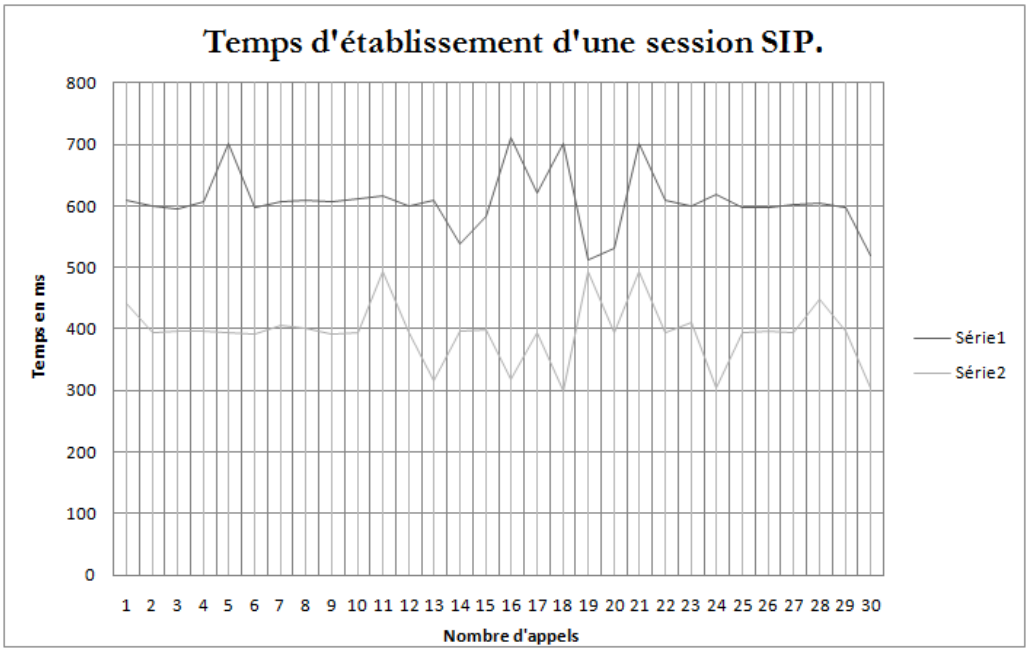


Figure 58. Mesures de temps dans l'établissement d'une session

Le temps de calcul cryptographique a également été étudié. Une comparaison entre une authentification HTTP Digest et la solution intégrant HOTP a été faite avec le commande SPEED d'OpenSSL[29]. Le tableau 13 montre que nous diminuons quelque peu le temps de calcul, le véritable gain restant au niveau de l'établissement de l'appel.

Tableau 13. Temps de calcul cryptographique pour l'authentification HOTP

Solutions	Nombre total de calculs	Temps de calcul cryptographique pour un bloc de 64 octets
HTTP Digest (MD5)	6 fonctions de hachages	Config. 1 : 6360 ns Config. 2 : 8760 ns
HOTP (SHA1)	2 calculs HOTP (2 fonctions de hachage SHA1 par calcul HOTP)	Config 1 : 4400 ns Config 2 : 7600 ns
Remarques : Configuration 1 : Inter Core 2 Duo - 2.00 GHz - RAM 3 Go Configuration 2 : Intel Pentium 4 - 2,80 – RAM 512 Mo		

5.4.3. *Une solution au déni de service*

Pour réduire les dénis de service par l'attaque sur le BYE, une authentification du message a été proposé en insérant dans la valeur « branch » du champ « Via » une donnée calculée

[29]OpenSSL : OpenSSL est une boite à outils de chiffrement comportant deux bibliothèques : une de cryptographie générale et une implémentant le protocole SSL. Les paramètres de la commande en ligne OpenSSL sont très nombreux ; ils permettent d'indiquer entre autres l'un des nombreux types de chiffrement (exemple : DES ou Triple DES, DSA, RC4, RC5, RSA...), d'encodage (base64 ou autres) et de hachage (MD5, SHA-1...). La commande SPEED permet d'évaluer le temps de calcul cryptographique pour les algorithmes de la bibliothèque.

avec une fonction HOTP à partir de plusieurs valeurs dont le secret partagé entre l'usager et le proxy. La figure 59 illustre le cas d'un client générant le BYE avec notre solution, l'IPBX pouvant interpréter ce champ l'accepte.

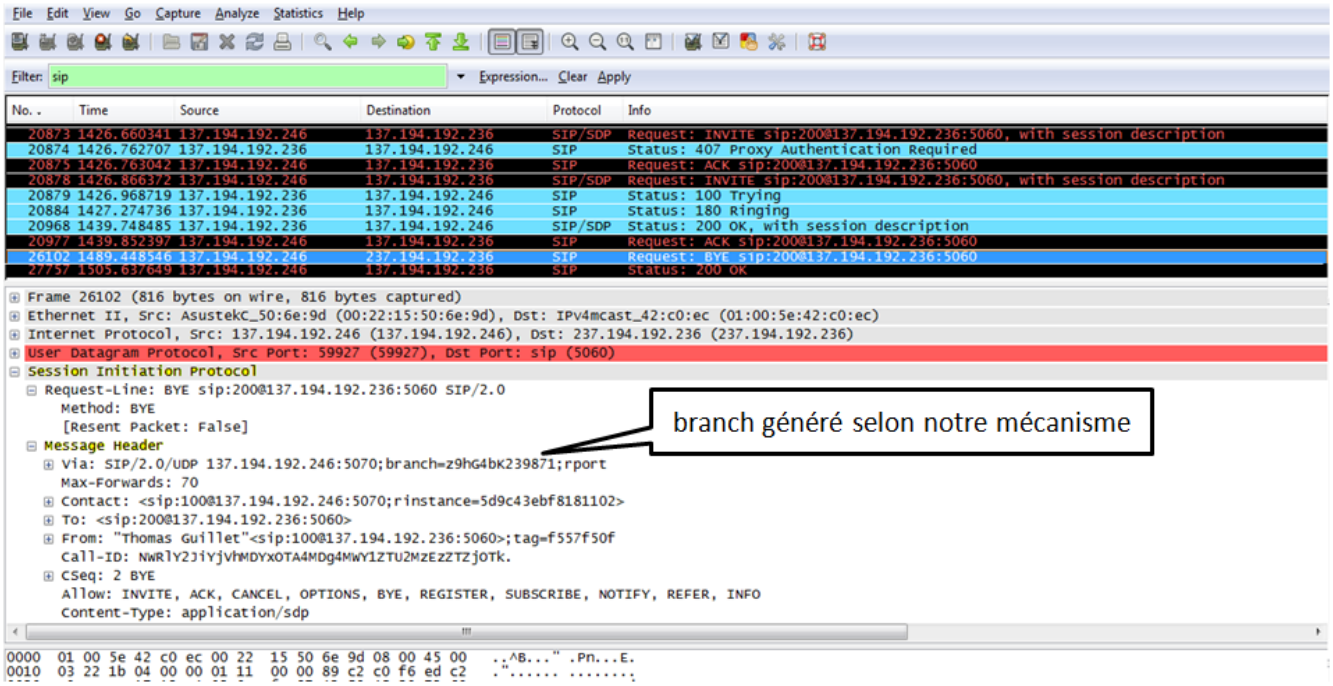


Figure 59. BYE intégrant un « branch » calculé selon notre processus

Le tableau 14 présente la synthèse des tests. Dans la mesure où nous n'avons pas traité les modalités pour connaitre les extensions supportées par les différents éléments, il y a un cas bloquant. En effet si le client supporte la solution, il attend un BYE construit selon le processus défini.

Tableau 14. Synthèse des résultats concernant la validation de la solution au DoS

Client normal	Client Modifié	IPBX Normal	IPBX Modifié	Comportements des équipements dans les différentes configurations possibles
✓		✓		Situation normale. L'attaque du BYE est possible.
✓			✓	En cas d'attaque, l'IPBX génère une authentification HTTP Digest. BYE forgé avec l'identité d'un client hors domaine reste possible.
	✓	✓		Ce cas est bloquant dans la mesure où le client refuse les BYE de l'IPBX. Il faudrait qu'à l'enregistrement le client puisse savoir si le serveur supporte cette extension. Nous n'avons pas traité complètement ce cas.
	✓		✓	L'attaque du BYE sur le serveur et le client échoue. Un BYE frauduleux a été forgé avec le logiciel SIPNess à partir de données sniffés sur le réseau.

✓ : équipement utilisé.

5.4.4. *Synthèse des analyses*

Nous proposons une synthèse des analyses dans le tableau 15.

Tableau 15. Tableau de synthèse des validations pratiques

Solutions	Compatibilité avec l'existant	Impact sur l'existant	Observations
HTTP Digest renforcé	Totale	Augmentation du temps de calcul cryptographique	L'interopérabilité est totale dans la mesure où notre solution reste au niveau de l'information de l'usager. Si notre mécanisme est un processus systèmatique, il bloquerait dans le cas d'un serveur légitime n'implémentant pas la solution.
HOTP	Totale	Diminution des échanges de message	L'interopérabilité est totale dans la mesure où notre mécanisme cohabite l'authentification HTTP Digest. Cela permet de gérer les cas où l'un des équipements n'implémente pas.
DoS	Partielle	Rajout d'un calcul cryptographique	Dans la mesure où nous n'avons pas traité le mode de déclaration des extensions supportées, il reste un cas bloquant.

5.5. Conclusion

Ce chapitre nous a permis de démontrer la faisabilité des solutions développées dans ce manuscrit. L'interopérabilité est quasiment acquise pour toutes les contributions. Le mécanisme pour limiter le déni de service nécessite cependant un complément d'étude pour être complètement interopérable. Nous avons également montré que l'impact du temps de calcul cryptographique des solutions est négligeable.

Dans la mesure où SIP n'impose pas de solutions de sécurité mais fait seulement des recommandations, les propositions faites dans cette thèse peuvent répondre à un besoin ponctuel de sécurité à un moindre coût. Le déploiement de ces solutions étant compatible avec les architectures déjà installées, leur adoption n'entraine pas d'interruption de service. Par ailleurs, l'utilisation des éléments usuels comme les champs SIP ou les fonctions de hachage déjà existants rendent les modifications simples.

CHAPITRE 6

DEFINITION D'UNE ARCHITECTURE DE TELEPHONIE SUR IP SECURISEE

Ce chapitre porte sur la définition et la spécification d'une architecture pour une sécurité de bout-en-bout de la téléphonie sur IP. Cette architecture se distingue par une mise en œuvre d'une signalisation sur le canal du media. Ceci permet une interopérabilité avec tout type d'infrastructure notamment de signalisation de ToIP. « Future Narrow Band Digital Terminal » et « Secure Voice over IP Simple Protocol » basés sur ce principe seront analysés dans ce chapitre. De cette analyse des exigences sont déduites pour la définition et la spécification d'une architecture « robuste et sécurisé » de ToIP basé sur le canal du media.

6. Définition d'une architecture de téléphonie sur IP sécurisée

6.1. Sécuriser les appels de bout-en-bout

La téléphonie sur IP est basée sur une multitude de protocoles. Ces derniers interopèrent globalement entre eux pour la gestion des appels. La continuité concerne principalement les fonctions de base de la signalisation et l'acheminement de la voix. La sécurité est quant à elle propre à chaque infrastructure et ne fait l'objet d'aucune spécification au niveau des interconnexions. La sécurité des appels de bout-en-bout repose donc sur une cohérence des protocoles et des politiques de sécurité mis en œuvre. Cependant avec la multiplication des technologies, des protocoles, des opérateurs, cette configuration n'est pas celle rencontrée.

Pour contourner cette hétérogénéité, deux protocoles Future Narrow Band Digital Terminal [FNBDT] et Secure Voice over IP Simple Protocol [BAS05] ont proposé une solution applicative complètement indépendante de l'infrastructure sous-jacente. Elles établissent une signalisation de sécurité dans le canal média après l'établissement de l'appel. Cette approche permet d'avoir une solution complètement interopérable avec l'environnement de la téléphonie sur IP actuel. Nous avons analysé ces solutions pour vérifier leur robustesse et leur adéquation avec les architectures de ToIP. Notre travail a montré qu'ils restaient encore des points achoppements pour une mise en œuvre généralisée dans un environnement IP. Pour corriger ces vulnérabilités, une architecture de ToIP sécurisée sera ainsi spécifiée.

6.2. Sécurité de bout en bout basée sur une infrastructure hétérogène

6.2.1. *Future Narrow Band Digital Terminal & Secure Communication Interoperability Protocol*

Future Narrow Band Digital Terminal (FNBDT) et son évolution Secure Communication Interoperability Protocol (SCIP) sont des solutions applicatives de sécurité pour la téléphonie. Ces protocoles proposés par la NSA pour les services gouvernementaux américains ont été depuis adoptés par l'OTAN. Ils sont décrits dans différents documents mais seul le plan de signalisation de FNBDT [FNBDT] est public comme le confirme [GAU09]. L'ensemble des spécifications (gestion des clés, condition d'interopérabilité, le vocodeur et certaines informations cryptographiques) sont réservées aux services étatiques et aux industriels.

FNBDT pose le principe d'une signalisation de sécurité échangée dans le canal média. Il ne spécifie donc pas les modalités d'établissement de l'appel qui sont à la charge de l'infrastructure téléphonique (cf. figure 60) Les échanges de messages permettent donc les services suivants :

- la signalisation de contrôle nécessaire pour initier, maintenir, et terminer les modes sécurisés ;
- la sélection du mode dit « opérationnel » : voix non sécurisée, voix sécurisée, données chiffrées ;
- la synchronisation et les choix cryptographiques.

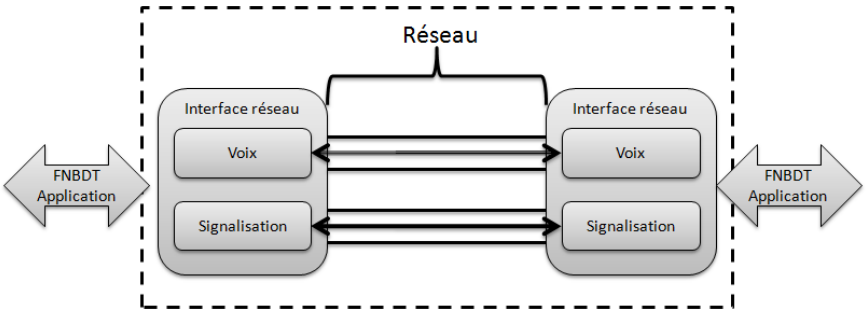


Figure 60. Application FNBDT dans l'infrastructure de téléphonie sur IP

FNBDT s'appuie sur une architecture protocolaire pour mettre en œuvre la signalisation mais également les autres services comme le transport, le chiffrement, le vocodeur. L'organisation des différents modules de l'application (cf. figure 61) est la suivante :

- les modules du haut de la pile est composé de différents applicatifs chargés de l'établissement de l'appel sécurisé, la gestion des clés, le transport des données, le vocodeur MELP[30] [SUP97] ;

[30] MELP : Mixed Excitation Linear Prediction est un vocodeur à 2400 bits/s développé par les services gouvernementaux américains.

- la couche de chiffrement qui chiffre et déchiffre les informations échangées ;
- la couche message ;
- la couche transport qui connait deux modes. Le premier « Mode Framed » permet un transport fiable avec accusé de réception. Le deuxième « Mode Fullbandwidth » qui permet une transmission tolérant des erreurs, des rejets et des pertes d'informations.

Pour garantir le succès des échanges de signalisation et des données de contrôle, FNBDT utilise des mécanismes de fiabilisation. Plusieurs mécanismes sont mis en place comme le Forward Error Control[31] (FEC), le Cyclic Redundancy Check CRC, des acquittements et des règles de rejet [DAN02-1]. Ces mécanismes sont utilisés quelles que soient les propriétés de la couche transport. Bien que conçu pour de nombreux types de liaisons (tactiques, civils, filaires, radioélectriques,...), FNBDT est une solution orientée « commutation de circuit », c'est à dire pour un lien avec des ressources réservées et des caractéristiques constantes. Le passage à l'IP (commutation de paquets) introduit de nouveaux problèmes comme la variation des délais d'acheminement et la perte de paquets.

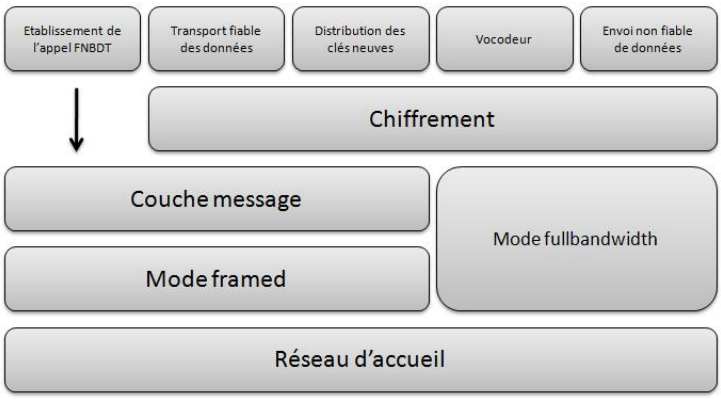


Figure 61. Architecture protocolaire FNBDT

Dès 2001 [DAN01] identifiait certaines limites. En mode trame, la transmission émise par la couche message est encapsulé dans un groupe de 127 trames maximum. En considérant les messages de début SOM (Start of message) et de fin EOM (End Of Message) ainsi que les 125 trames classiques, la super trame atteint 2556 octets. La couche transport du modèle TCP/IP va donc fractionner la transmission en fonction de la taille maximum des trames pouvant être transportée par le réseau (i.e. la taille maximale d'une trame est 1500 octets en *Ethernet*). Ce fractionnement et la gigue vont augmenter le délai de reconstruction de la transmission pouvant générer une augmentation de la surcharge protocolaire. Bien que le principe de la solution applicative semble simple, FNBDT et

[31] FEC et CRC : Forward Error Correction et Cyclic Redundancy Check sont des mécanismes de protection contre les erreurs dues à la transmission de données. L'émetteur ajoute de la redondance afin de permettre au destinataire de détecter et de corriger une partie des erreurs. Cela permet d'éviter la retransmission et d'économiser de la bande passante.

SCIP font l'objet d'études de fonctionnement sur les différents liens comme la HF [ALV07], la VHF/UHF [ALV09] et l'IP/3G [DAN02-2]. Une totale interopérabilité nécessite de nombreuses simulations et validations pour garantir le fonctionnement sur tout type de réseaux.

Nous allons maintenant étudier l'établissement de l'appel sécurisé. Quand un canal média est ouvert entre les deux terminaux FNBDT, le premier message envoyé ou reçu est le message « Capabilities ». Celui-ci contient toutes les informations qui permettent de contrôler, d'évaluer les compatibilités et de décider quels algorithmes utilisés. Au cours de ce premier échange, le mode « opérationnel » est choisi : mode clair ou chiffré, ou transfert de données. C'est à ce moment également que l'on choisit le type de clés (nationale ou OTAN).

Dans la mesure où le mode clair n'est pas retenu, les messages Capabilities sont suivis des messages suivants :

- les « Parameters/Certificate » messages contenant les paramètres associés au mode opérationnel choisi et le certificat de l'usager ;
- les « Foward and Reverse F(R) » messages pour l'établissement de la clé secrète (non défini dans [FNBDT]) ;
- les « Cryptosync » messages pour la synchronisation et le contrôle du mode chiffré.

La chronologie de l'appel est illustrée en figure 62. L'architecture de gestion des certificats et les modalités de distribution ne sont décrites dans [FNBDT].

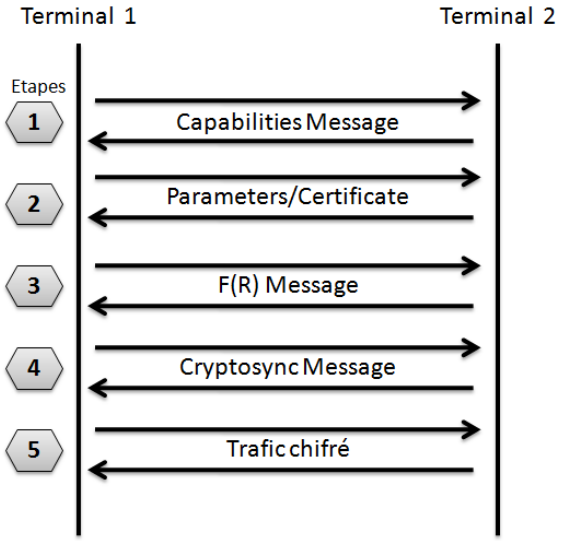


Figure 62. Etablissement d'un appel FNBDT

Cette architecture présente cependant des spécifications qui nous paraissent être des vulnérabilités :

- ce protocole ne prévoit pas d'authentification explicite des usagers avant les messages « Capabilities ».

- la signalisation est en claire ce qui permet à un attaquant en coupure de forcer le mode clair. Si la vérification des certificats n'est pas réalisée, le pirate peut également les modifier.
- il n'y a pas de mécanisme d'intégrité ;
- le protocole a été développé pour les réseaux à commutation de circuit. La couche transport doit être adaptée aux réseaux IP.

6.2.2. *Secure Voice over IP Simple Protocol*

Secure Voice over IP Simple Protocol (SVSP) défini par Carole Bassil [BAS05] est une solution qui a été conçu nativement pour la ToIP. L'appel sécurisé est également séparé en deux parties : la signalisation pour sécuriser l'appel et l'appel sécurisée proprement dit. Cette solution s'appuie cependant de manière explicite sur des éléments de confiance qui sont :

- **une carte à puce** stockant les éléments de sécurité nécessaire au fonctionnement de l'application ;
- l'entité de confiance distribuée appelée **« Trusted Authentication Autority » (TAA)** ;
- l'entité de confiance racine appelée **« Global Truster Authentication Autority » (GTAA)**.

L'architecture matérielle est illustrée par la figure 63.

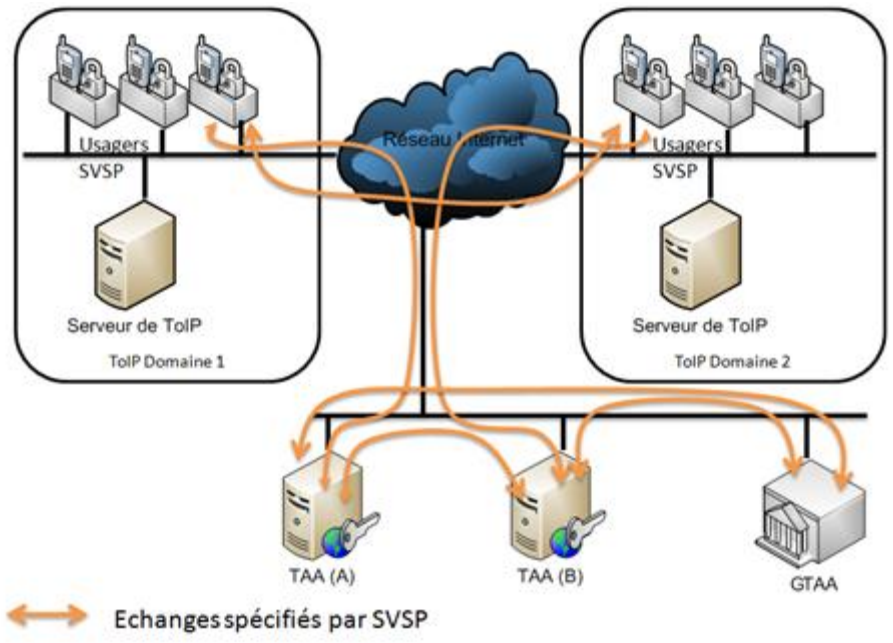


Figure 63. Architecture matérielle de SVSP

La sécurisation des messages est assurée par des certificats pour les échanges au niveau des TAA et des GTAA, et par un secret pré-partagé entre les TAA et les usagers. L'authentification mutuelle entre les entités est la brique de sécurité élémentaire de cette architecture. Cette disposition est largement justifiée par les travaux présentés dans ce manuscrit. Le GTAA est l'entité de confiance racine qui gère un ensemble de TAA. Il fournit les certificats, gère la base de révocation et assure les relations avec les autres GTAA. Les TAA délivre un identifiant unique et universel associé à un secret pour ses abonnés. Au cours d'un appel, il participe à l'élaboration d'une clé de session et au rapport de fin d'appel.

Les propriétés de sécurité offertes sont :

- pour la signalisation entre l'usager et le TAA :
    - l'authentification mutuelle ;
    - le non-rejeu ;
    - l'intégrité ;

- pour les échanges entre usagers :
    - l'authentification mutuelle des usagers ;
    - la confidentialité de la conversation et d'une partie de la signalisation ;
    - l'intégrité ;
    - le non-rejeu ;
    - la non répudiation de l'appel fournie sous la forme d'un rapport émis et signée par la TAA de l'usager.

Comme pour FNBDT, l'établissement de l'appel sécurisé se fait au travers d'échange de messages (cf. figure 64). Le déroulement du protocole SVSP débute une phase d'initialisation par l'envoi des messages en clair ou en chiffré comme : U_RUCA.req, U_RUCA.rep, etc. Ils permettent l'authentification des entités et l'échange des capacités cryptographiques. Pendant l'établissement de l'appel sécurisé, la clé de session est établie à partir du secret de l'usager qui initie l'appel et d'un nombre aléatoire généré par son TAA. Les TAA permettent l'authentification mutuelle des usagers. A l'issue, ces derniers vérifient certains paramètres et réalisent des acquittements permettant le passage en mode chiffré. La conversation est alors chiffrée à partir d'une clé de session, ainsi qu'une partie de la signalisation. Pour terminer une session, les usagers génèrent les messages dits « EndSecureMediaSession » (ESMS). A partir de ces derniers, les TAA génèrent les rapports de non-répudiation. La méthode de génération de la clé de session n'est pas imposée. Par ailleurs tous les messages font l'objet d'un contrôle d'intégrité par une fonction de hachage. Ils possèdent également tous un numéro de séquence et un horodatage. La chronologie d'un appel SVSP avec deux TAA est illustrée par la figure 64.

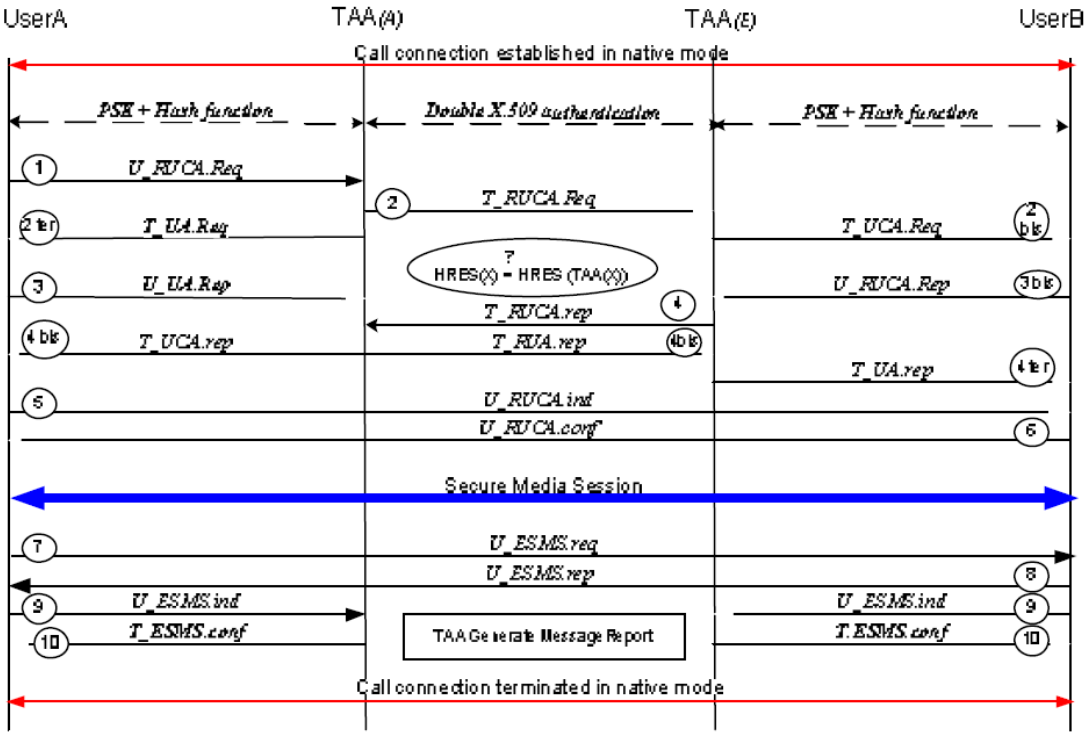


Figure 64. Scenario d'appel SVSP [BAS05]

Notre analyse de SVSP nous amène à mettre en avant les éléments suivants :

- il prévoit nativement une authentification mutuelle, ce qui limite les attaques de type « l'homme au milieu » ;
- il prévoit le chiffrement d'une partie de la signalisation ;
- il spécifie une architecture support pour la sécurité. Les entités appelées « Trusted Authentication Autority » et GTAA « Global Trusted Authentication Autority » permettent de garantir l'interopérabilité des authentifications ;
- il ne prévoit cependant pas de fiabilisation de la transmission, ce qui peut poser des problèmes avec une couche transport implémentant UDP.

SVSP a été validé dans certaines configurations mais n'a pas été déployé à grande échelle. Ce travail reste une des rares contributions à la sécurité de bout-en-bout de la téléphonie indépendante de la signalisation. La fiabilisation reste l'axe principalement d'amélioration de ce protocole.

6.2.3. *Comparaison entre les solutions de sécurité de bout-en-bout*

Dans ce manuscrit, nous avons analysé plusieurs mécanismes de sécurité. Pour bien percevoir l'intérêt des solutions de sécurité basée sur le canal média, une comparaison de plusieurs approches a été réalisée au travers du tableau 16. L'analyse confirme l'état des lieux présentée dans [§6.1.]. Seules les solutions type FNBDT ou SVSP garantissent une sécurité de bout-en-bout totalement interopérable. Certes ces protocoles ne spécifient pas la signalisation d'appel, mais ils interviennent en support pour la protection des appels. Il reste que les spécifications de la solution opérationnelle FNBDT ne sont pas publiques, ni destinées à usage civil, d'où l'intérêt de définir une solution ouverte pour les particuliers.

Tableau 16. Comparaison des solutions de sécurité bout-en-bout pour la ToIP

Protocoles	SCIP FNBDT	SIP + S/MIME	SIP + SAML	SVSP	SRTP	Skype
Concerne :						
Signalisation d'appel	Non	Oui	Oui	Non	Non	Oui
Voix	Oui	Non	Non	Oui	Oui	Oui
Propriété de sécurité						
Confidentialité	Oui	Oui	Non	Oui	Oui	Oui
Intégrité	Non	Oui	Non	Oui	Oui	-
Authentification	Oui	Oui	Oui	Oui	Oui	Oui
Non rejeu	Oui	Non	Oui	Oui	Oui	-
Non répudiation	Non	Oui	Non	Oui	Non	-
Infrastructure de sécurité	PKI	PKI	PKI	PKI & PSK	PKI ou PSK	PKI
Interopérabilité	Totale (par conception)	Limitée aux usagers SIP	Limitée aux usagers SIP	Totale (par conception)	Flux RTP de bout-en-bout Nécessite une mécanisme de négociation de clé compatible avec la signalisation	Limitée aux usagers Skype
Standard	Oui	Oui	Non	Non	Oui	Non
Spécifications	Non publiques	Publiques	Publiques	Publiques	Publiques	Non publiques

6.2.4. *Ce qu'il faut maintenir ou corriger pour définir une nouvelle architecture*

De l'analyse des solutions FNBDT et SVSP, nous retenons les éléments suivants pour la spécification d'une solution concurrente :

- spécifier une architecture support dédiée à la sécurité ;
- établir une signalisation de sécurité entre usagers dans le canal média ;
- chiffrer la signalisation dès qu'une clé de session est établie ;
- dans le cadre de la ToIP prendre en compte les spécificités des réseaux IP : latence, gigue, problème de fragmentation ;
- prévoir un mécanisme de fiabilisation pour le transport. La voix sur IP utilise généralement le protocole UDP ;
- retenir la chronologie d'un appel sécurisé spécifié par SVSP ;
- prendre en compte la notion d'interface avec l'existant : réseau IP, téléphone IP ;
- définir une application sous forme de modules ;
- prévoir un mécanisme d'intégrité ;
- définir les services de sécurité propres à chaque appel ;
- prévoir une authentification mutuelle des différentes entités avant l'établissement d'une connexion, surtout si une partie de la signalisation de sécurité est en clair.

6.3. Spécifications d'une architecture de téléphonie sur IP sécurisée

6.3.1. *Une architecture support dédiée à la sécurisation*

La définition d'une architecture de téléphonie sur IP sécurisée proposée est destinée à protéger les appels. La sécurité de bout-en-bout telle qu'elle est souhaitée peut s'envisager de deux manières soit dans la signalisation d'appel, soit dans le canal média. Le besoin d'interopérabilité sous-jacent ne permet pas à l'heure actuelle de s'appuyer sur la signalisation compte tenu de l'hétérogénéité des solutions techniques [BAS06]. Le choix du canal média est donc celui que nous avons retenu. Cette option sous-tend une architecture dédiée à la sécurité. Pour cela, l'analyse des solutions comme FNBDT, SCIP ou SVSP nous permet de définir trois entités distinctes :

- l'application de sécurité (AS) : elle s'interface avec le téléphone sur IP et permet la protection des appels et les échanges avec le tiers de confiance ;
- le tiers de confiance (TC) : il permet aux usagers au travers de l'application de sécurité de sécuriser les appels ;
- l'autorité de coordination (AC) : elle garantit l'interopérabilité entre les tiers de confiance.

Le tiers de confiance peut être considéré comme un fournisseur de service. Il fournit à l'usager les éléments secrets. Lorsqu'un client souhaite établir un appel sécurisé, il recherche le tiers de confiance de l'appelé pour créer une relation permettant la création d'une clé de session. Cela nécessite des identifiants standardisés : cette problématique n'est pas traitée dans ce manuscrit.

L'usager quant à lui doit posséder une téléphonie IP et l'application de sécurité pour permettre le déploiement de la solution de sécurité. L'application de ToIP permet l'établissement de l'appel et donc la mise en place du canal média. L'application de sécurité établit tout d'abord une connexion avec le tiers de confiance pour générer les secrets nécessaires à la protection de l'appel, puis génère une signalisation de sécurité entre usagers au travers du canal média.

L'autorité de coordination gère un groupe de tiers de confiance et les relations avec les autres autorités. Dans le cas d'emploi de certificats, elle est responsable du stockage, de la mise à jour de la liste des certificats périmés. Elle peut fournir à une autre AC le certificat d'un TC de sa responsabilité. Cette autorité a un rôle d'arbitrage et de contrôle de l'application de la politique de sécurité par les tiers de confiance. Une AC pourrait tout être un service étatique (ex. un régulateur).

La figure 65 présent cette architecture au niveau applicatif. Trois types d'échanges sont présents :

- la signalisation d'établissement d'appel pour la mise relation des usagers (cf. le fonctionnement de SIP ou H323). Elle se situe entre les différentes entités propres à la ToIP ;
- le canal média entre les usagers ;

- la signalisation de sécurité transportée directement par le réseau ou encapsulée dans le canal média.

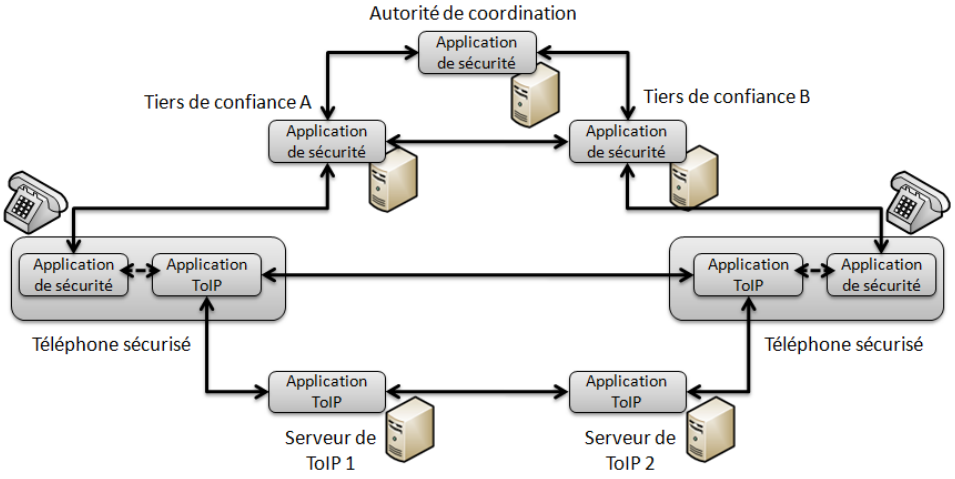


Figure 65. Echanges au niveau applicatif

Le tiers de confiance n'est pas forcément lié au fournisseur de téléphonie sur IP. Ce dernier doit être vu comme un fournisseur de services. De même le choix de l'application de ToIP est libre, la seule obligation est de pouvoir s'interfacer logiquement et/ou physiquement pour permettre les échanges nécessaires avec l'application de sécurité. Cette architecture est le socle physique et organisationnel indispensable pour faire émerger une solution de sécurité universelle pour la téléphonie sur IP. D'ailleurs deux usagers peuvent directement établir un appel sécurisé s'ils partagent déjà un secret.

6.3.2. *Les propriétés de sécurité d'un appel téléphonique*

Les propriétés de sécurité d'un appel téléphonique n'ont pas été définies. L'usager doit pouvoir spécifier ses besoins de sécurité. Comme cela a déjà été mentionné, la sécurité rajoute du temps de calcul cryptographique, une augmentation de l'occupation de la bande passante, du délai dans la transmission des paquets voix. Le mécanisme de protection doit donc être modulable pour ne pas rajouter des coûts dont l'utilité n'est pas avérée. Néanmoins deux propriétés nous paraissent indispensables quelques soient le contexte, l'authentification mutuelle des entités dans l'architecture sécurisée et le non rejeu. L'analyse de ce travail démontre sans ambigüité l'intérêt d'avoir la garanti de dialoguer avec la bonne entité. Pour le non rejeu, la garantie que les échanges ne puissent pas être réutilisés est également primordiale dans les réseaux ouverts comme Internet où les trafics peuvent être interceptés aisément.

Quant aux propriétés de sécurité comme la confidentialité, l'intégrité ou la non répudiation, elles ne nous semblent pas être nécessaires pour tous les appels téléphoniques. La confidentialité n'est pas nécessaire dans tous les cas comme pour une simple demande d'informations concernant une heure d'ouverture d'une administration. De même la non répudiation et l'intégrité ne sont pas forcément indispensable pour réserver une table au restaurant. Le niveau de sécurité doit être adapté chaque appel, limitant ainsi l'impact de la sécurité sur la QoS et l'occupation du réseau.

Cette architecture ouvre également de nouvelles perspectives dans le domaine de l'anonymat. La session voix pourrait être établie entre l'usager et le tiers de confiance. Au travers de la signalisation de sécurité le numéro de l'appelé peut être protégé en confidentialité. L'appel vers le destinataire final serait alors masqué par l'ensemble des connexions établies à partir du tiers de confiance. L'infrastructure suggérée permet une anonymisation des associations appelant/appelé.

6.3.3. *Le déroulement d'un appel téléphonique sécurisé de bout-en-bout*

La mise en relation des entités définies précédemment nécessite un séquencement dans les différentes connexions. Après l'établissement de l'appel dont le principe a déjà été établi dans le début du manuscrit, les usagers initie l'appel sécurisé en sollicitant leur tiers de confiance pour obtenir la clé de session. Chaque usager fournit son identifiant, celui du correspondant, les capacités cryptographiques et les propriétés de sécurité souhaitées. Ce principe est établi dans de nombreux protocoles comme SIP [RFC3261] qui fournit dans ses messages le « From », le « To » et ses capacités d'échange au travers du protocole SDP [RFC4568].

Les tiers de confiance se mettent en relation selon un modèle qui peut s'apparenter à celui de la téléphonie. Ils négocient la clé de session et la compatibilité des besoins de sécurité. Dès lors que les applications de sécurité possèdent la clé de session, ils peuvent échanger directement entre eux au travers du canal média. La figure 66 décrit de manière les échanges.

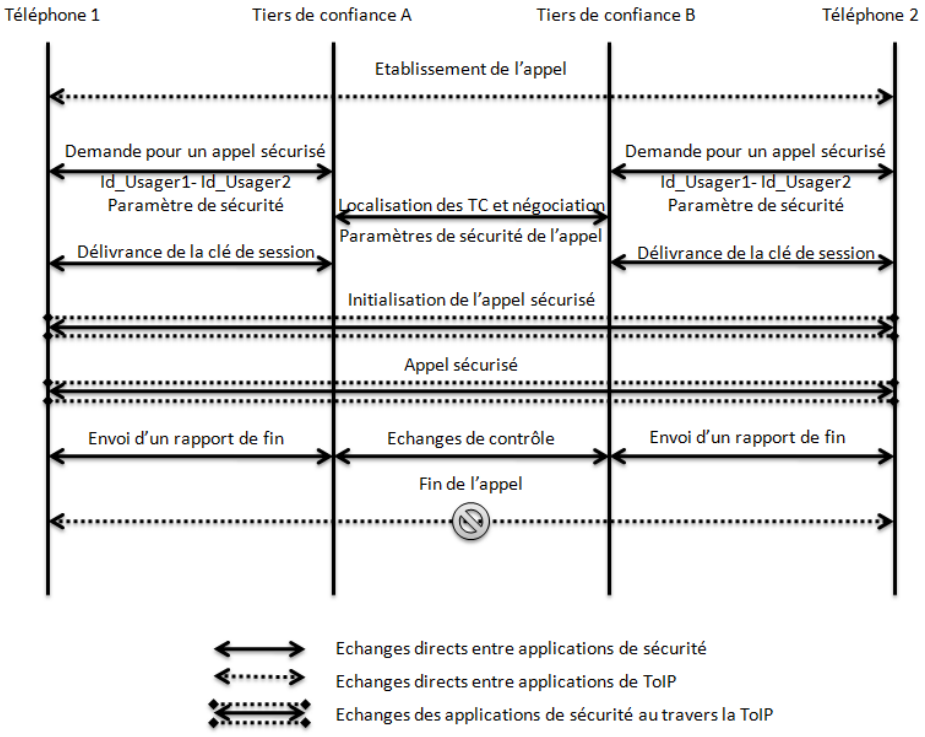


Figure 66. Etablissement d'un appel sécurisé

Les relations de confiance qui existent entre les différentes entités doivent permettre de générer une clé de session et de la diffuser de manière sécurisée. Un attaquant qui écoute

le réseau ne doit pas pouvoir trouver le secret qui sécurise l'appel. Dans la mesure où les usagers possèdent un secret pré-partagé, la sollicitation des tiers de confiance n'est pas forcément nécessaire. Les usagers peuvent passer directement de l'établissement d'appel à l'initialisation de l'appel sécurisée mais toutes les propriétés de sécurité ne pourront pas être garanties.

6.3.4. *Une architecture protocolaire*

Pour mettre en œuvre cette architecture, nous définissons une pile protocolaire cohérente avec les architectures de téléphonie sur IP. Le niveau de spécification est celui d'une analyse fonctionnelle. La pile protocolaire présentée identifie les composants logiques et les contraintes. Chaque composant est recensé, caractérisé, ordonné, hiérarchisé et valorisé. Le recensement des contraintes permet par ailleurs un dimensionnement adéquat. L'objectif est de fournir les éléments nécessaires à la réalisation de cette solution.

L'architecture type d'une solution de ToIP est illustrée par la figure 67. La sécurité applicative proposée devra s'interfacer au niveau du protocole de diffusion de la voix. Ce dernier est généralement associé à UDP [RFC768] qui est un protocole de transport fonctionnant en mode non-connecté : il n'y a pas de moyen de vérifier si tous les datagrammes envoyés sont bien arrivés à destination et ni dans quel ordre. Il n'est prévu aucun contrôle de flux ni contrôle de congestion. C'est pour cela qu'il est souvent décrit comme étant un protocole non-fiable. Cette caractéristique devra être prise en compte dans notre architecture au travers d'un mécanisme de fiabilisation.

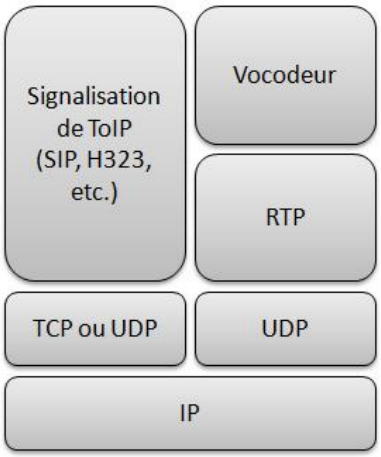


Figure 67. Architecture protocolaire générique d'une solution de ToIP

L'appel sécurisé prévoit une signalisation de sécurité et une protection du canal média. La pile protocolaire doit donc intégrer un vocodeur, un module de gestion de la signalisation, un module cryptographique. Par ailleurs comme nous l'avons précisé précédemment, le besoin de sécurité doit être adapté à chaque appel. Il faut donc également un module pour gérer les propriétés de sécurité demandées. La pile protocolaire (cf. figure 68) s'appuie sur deux types de protocoles sous-jacents :

- le protocole RTP pour la diffusion de la voix, lui-même s'appuyant sur UDP. Le canal média doit donc être considéré comme non fiable pour le transport de la signalisation ou la voix générées par l'application de sécurité entre les usagers ;
- le protocole de transport du réseau IP est soit TCP ou UDP. Dans la mesure où le canal média est considéré non fiable, nous considérons UDP pour le transport de la signalisation entre les usagers et les tiers de confiance. Cette considération sous-entend qu'un mécanisme de fiabilisation est systématiquement assuré par l'application de sécurité.

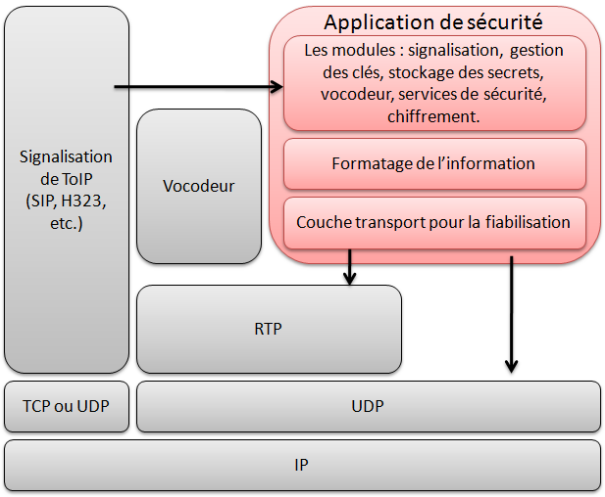


Figure 68. Pile protocolaire de la signalisation de sécurité

La pile protocolaire (cf. figure 68) se décompose en trois couches :

- la couche « Modules » :
  - le module signalisation : il permet les connexions avec le tiers de confiance et l'établissement de l'appel sécurisé avec un autre usager. IL doit accéder à l'identifiant de l'appelant et de l'appelé pour les fournir au tiers de confiance ; Ces informations sont données par le protocole de signalisation de ToIP, soit par le module « service de sécurité » ;
  - le module de chiffrement : il effectue les opérations de chiffrement et de déchiffrement en utilisant les éléments secrets du module « gestion des clés » ;
  - le module de gestion des clés : il stocke le secret fourni par le tiers de confiance, la clé de session, et éventuellement une clé pré-partagé entre usagers ;
  - le module services de sécurité : il fournit à l'établissement de l'appel sécurisé les propriétés de sécurité souhaitées par l'usager. Si une interface n'est pas possible entre l'application de sécurité et celle de ToIP, ce module reçoit les identifiants de l'appel ;
  - le vocodeur : il fournit un vocodeur à l'application de sécurité, ce qui limite l'interface avec le téléphone IP ;
- la couche « Formatage » : elle permet la mise en forme des données selon les spécifications de la signalisation ;

- la couche transport : elle fiabilise le transport. Le média est généralement transporté par UDP qui ne garantit pas la réception des paquets.

Cette pile prévoit des échanges avec d'autres éléments. Trois interfaces ont été identifiées :

- avec la couche transport UDP ;
- avec le protocole de diffusion de la voix (principalement RTP) ;
- avec la signalisation de ToIP pour obtenir les identifiants de l'appelant et de l'appelé.

Tableau 17. Qualité d'écoute en fonction du délai de transmission [DEO07]

Délai de transmission de la voix	Qualité d'écoute
< à 300 ms	Excellente
Entre 300 et 500 ms	Moyenne
Entre 500 ms et 1 s	Faible
> à 1 s	Impossible

Enfin les délais de transmission doivent être compatibles du besoin d'une application synchrone avec des contraintes temporelles comme le téléphone. Le tableau 17 donne la qualité d'écoute en fonction des temps d'acheminement de la voix. Les temps de calcul cryptographique et le volume d'informations à traiter pour sécuriser l'appel devront être compatibles avec ces délais de transmission. Cette considération devra être prise en compte dans la conception de la signalisation mettant en œuvre cette architecture.

6.4. Conclusion

L'architecture définie dans ce chapitre est dédiée à la sécurité des appels. Elle garantit une protection de bout-en-bout en utilisant le canal média pour mettre en œuvre une signalisation de sécurité. Notre approche ne nécessite pas de modifications des infrastructures déjà déployées. Compte tenu de l'hétérogénéité des solutions de ToIP, cette totale interopérabilité avec l'existant est l'élément décisif pour son adoption à grande échelle.

La faisabilité de cette approche a été démontrée par FNBDT et SVSP. Notre analyse de ces solutions nous a permis de définir une nouvelle architecture entérinant certains choix, et en corrigeant d'autres. Nous avons confirmé la nécessité d'avoir une authentification mutuelle entre chaque entité dans le contexte de la ToIP. La fiabilisation du transport a également été mis en avant dans notre conception en identifiant dans la structure protocolaire une couche dédiée à cet objectif. Nous avons enfin établi les différents flux d'informations, confirmant le besoin d'une architecture support dédiée à la sécurité.

Il reste à décliner cette architecture au niveau des échanges protocolaires.

CHAPITRE 7

CONCLUSION GENERALE ET PERSPECTIVES

Ce travail de recherche a permis d'analyser de cetains problèmes de sécurité de la téléphonie sur IP et les limites des solutions actuellement mises en œuvre. Nous avons alors montré comment d'une manière astucieuse il était possible de renforcer l'authentification dans les architectures SIP. En spécifiant le calcul d'un certain nombre d'aléas et le contexte d'interprétation, les contributions ont apporté des solutions au déni de service, à l'optimisation des échanges protocolaires et au renforcement de l'authentification HTTP Digest. Néanmoins conscient des limites de ces résultats, une architecture indépendante des infrastructures de ToIP a été proposée pour sécuriser les appels de bout-en-bout.

7. Conclusion générale et perspectives

7.1. Conclusion générale

Ce travail de recherche avait comme objectif de contribuer à la sécurité de la téléphonie sur IP. Au travers de l'utilisation de champs opaques, nous avons proposé plusieurs solutions pour améliorer l'authentification dans les architectures SIP. Inspirer par la stéganographie, nous avons détourné des valeurs aléatoires (dite « opaque ») pour leur donner des propriétés de sécurité. En structurant la génération de certains champs et en définissant un contexte d'interprétation, les contributions ont permis d'apporter une réponse au déni de service, à l'optimisation des échanges protocolaires et au renforcement de l'authentification HTTP Digest. Le besoin d'interopérabilité avec l'existant a également été pris en compte dans la conception des mécanismes. Ce point est souvent négligé, ce qui limite le déploiement de nouveaux mécanismes. Les validations ont montré que la compatibilité avec les implémentations classiques était garantie. L'approche service qui a aussi prévalu dans le cahier des charges peut sans aucun doute être considérée comme une plus-value par rapport à l'existant.

La notion d'interopérabilité que nous avons mise en avant dans le renforcement de la sécurité de SIP est également au cœur de la proposition d'architecture de ToIP sécurisée. La téléphonie est basée sur une multitude de protocoles qui interopèrent globalement entre eux pour l'établissement d'appel. La sécurité reste propre à chaque infrastructure. La sécurité de bout-en-bout des appels n'est ainsi pas traitée pour les particuliers. L'interopérabilité doit donc être la première exigence des propositions de protection des appels téléphoniques. Notre analyse a montré que l'utilisation du canal média était une voie prometteuse, compte de tenu de l'existant, pour offrir des services de sécurité à chaque usager du téléphone.

7.2. Perspectives d'évolutions et futurs travaux

La sécurité de la téléphonie sur IP est un sujet critique qui pose des problèmes difficiles à résoudre. Dans la téléphonie commutée, le système est quasi fermé limitant la réalisation d'attaques. Avec l'intégration de la téléphonie dans les systèmes d'information et dans le monde des réseaux IP, la sécurisation de cette application devient particulièrement complexe. Les besoins concernent l'authentification, la confidentialité, l'intégrité, la protection contre l'usurpation d'identité, le respect de la vie privée ou encore la non répudiation.

La question n'est donc pas de savoir si la sécurité est nécessaire, mais comment se mettra en place une solution robuste et interopérable avec les infrastructures existantes. Chaque jour, la cybercriminalité nous rappelle que la sécurité n'est plus une option mais une obligation [FIL06]. La téléphonie reste encore préservée des attaques mais la convergence des réseaux vers le tout IP va nécessairement faire augmenter les problèmes de sécurité. Ce sujet de recherche reste donc un thème d'actualité. Nos futurs travaux compléteront la définition de l'architecture de téléphonie sur IP sécurisée en définissant les échanges de sécurité sur le canal média et en précisant le rôle de l'infrastructure support de sécurité.

Liste des acronymes

AC	Autorité de Coordination
AIB	Authenticated Identity Body
ARCEP	L'Autorité de régulation des communications électroniques et des postes
ARPA	Advanced Research Projects Agency
AVISPA	Automated Validation of Internet Security Protocols and Applications
CMS	Cryptographic Message Syntax
CTI	Convergence Téléphonie Informatique
DECT	Digital Enhanced Cordless Telephone
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
DoS	Deny of Service
DTLS	Datagram Transport Layer Security
EBIOS	Expression des Besoins et Identification des Objectifs de Sécurité
FNBDT	Future Narrow Band Digital Terminal
GSM	Global System for Mobile Communications
HLPSL	High-Level Protocol Specification Language
HMAC	Hash-based Message Authentication Code
HOTP	HMAC-based One Time Password
IAX	Inter Asterisk Exchange
IETF	Internet Engineering Task Force
IGC	Infrastructures de Gestion des Clés
INPI	Institut Nationale de Propriété Intellectuel
IP	Internet Protocol
IPBX	Internet Protocol Private automatic Branch eXchnage
IPSEC	Internet Protocol Security
ITU	International Telecommunication Union
LAN	Local Area Network
LNCS	Lecture Notes in Computer Science
MEHARI	MEthode Harmonisée d'Analyse de Risques
MIME	Multipurpose Internet Mail Extensions
MS	Module de Sécurité
NAT	Network Address Translation
NSA	National Security Agency
OS	Operating system
OTP	One-Time-Password

PABX	Private Automatic Branch eXchange
PEM	Privacy Enhanced Mail
PIN	Personnal Identification Number
PKCS	Public-Key Cryptography Standards
PC	Personnal computer
PSTN	Public Switched Telephone Network
RFC	Request For Comments
RNIS	Réseau Numérique à Intégration de Services
RTC	Réseau Téléphonique Commuté
RTP	Real-time Transport Protocol
SBC	Session Border Controller
SCIP	Secure Communications Interoperability Protocol
SDP	Session Description Protocol
SIP	Session Initiation Protocol
S/MIME	Secure/Multipurpose Internet Mail Extensions
SMS	Short Message Service
SPAM	Secure Communication Interoperability Protocol
SPAN	Security Protocol ANimator for AVISPA
SPIT	Spam over IP Telephony
SRTP	Secure Real-time Transport Protocol
SSI	Sécurité des Systèmes d'Information
SSO	Single Sign One
SVSP	Secure Voice over IP Simple Protocol
TC	Tiers de Confiance
TCP	Transmission Control Protocol
TLS	Transport Layer Security
ToIP	Telephny over Internet Protocol
UA	User Agent
UDP	User Datagram Protocol
UIT	Union Internationale des Télécommunications
URI	Uniform Resource Identifier
VoIP	Voice over Internet Protocol
VLAN	Virtual Local Area Network
VPN	Virtual Private Network
WAN	Wide Area Network

Bibliographie

[3CX] Produit 3CX :http://www.3cx.fr/ (consulté le 8 mars 2010).

[3GPP] 3GPP TS 33.102, « 3GPP Security ; Security Architecture », Release 7, décembre 2006, 64 pages.

[ABD08] H. Abdelnur, O. Festor, et R. State, « Failles et VOIP », journal MISC n° 39, septembre/octobre 2008, pages : 49-57.

[ALV06] J.M. Alvermann, M. T. Kurdziel, W.N. Furman, "The Secure Communication Interoperability Protocol (SCIP) over an HF Radio Channel", In proceeding of the IEEE 2008 Military Communications Conference, Washington, USA, octobre 2006, 4 pages.

[ALV08] J.M. Alvermann, M. T. Kurdziel, "The secure communication interoperability protocol (SCIP) over a VHF/UHF radio channel", In proceeding of the IEEE 2008 Military Communications Conference, San Diego, USA, novembre 2008, 6 pages.

[ARCEP09] Rapport de l'observation de l'ARCEP, « Le marché des services de communications électroniques en France au 1er trimestre 2009 », disponible sur le site de l'ARCEP.

[ARM05] A. Armando, D. Basin, Y. Boichut, Y. Chevalier, L. Compagna, J. Cuellar, P. Hankes Drielsma, P.C. Heám, O. Kouchnarenko, J. Mantovani, S. Mödersheim, D. von Oheimb, M. Rusinowitch, J. Santiago, M. Turuani, L. Viganò, et L. Vigneron, « The AVISPA Tool for the Automated Validation of Internet Security Protocols and Applications », Computer Aided Verification Book, Spinger, volume 3576/2005, juillet 2005, pages : 281-285.

[AST] Projet ASTERISK : http://www.asterisk.org/ (consulté le 4 septembre 2009).

[AUT07] Authentification - Règles et recommandations concernant les mécanismes d'authentification de niveau de robustesse standard, n°729/SGDN/DCSSI/SDS du 12/04/2007, 29 pages.

[AVISPA] Projet Avispa : http://www.avispa-project.org/ (consulté le 5 avril 2010)

[BAS05] C. Bassil, « SVSP (Secure Voice over IP Simple Protocol) une solution pour la sécurisation de la voix sur IP », thèse présentée en décembre 2005 à l'ENST Paris.

[BEL09] S. Bellec, « MWC 2010 : Simon Bransfield-Garth (Cellcrypt) : « La sécurité des communications téléphoniques s'affaiblit » », Journal 01netPro, janvier 2009, http://pro.01net.com/editorial/512692/mwc-2010-simon-bransfield-garth-(cellcrypt)-la-securite-des-communications-telephoniques-saffaiblit/ (consulté le 2 avril 2010).

[BEN09] M. Benoist, « Le SaaS s'invite aussi sur le terrain de la téléphonie », Veille & Actualité Informatique - Electronique – Télécoms, Technique de l'ingénieur, novembre 2009, http://www.techniques-ingenieur.fr/article/article_6437/le-saas-s-invite-aussi-sur-le-terrain-de-la-telephonie.html (consulté le 10 janvier 2010).

[BER08] F. Bergé, O. Bouzereau, J. Desvouges, « Un vecteur de richesse fonctionnelle pour le Centrex IP », Journal 01netPro, mars 2008, http://pro.01net.com/editorial/374985/un-vecteur-de-richesse-fonctionnelle-pour-le-centrex-ip/ (consulté le 10 janvier 2010).

[BOI07] Y. Boichut, P.C. Héam, O. Kouchnarenko, « Vérifier automatiquement les protocoles de sécurité », Techniques de l'Ingénieur, octobre 2007, 8 pages.

[BRA09] B. Braux, « La sécurité des téléphones sans fil Dect remise en question », Journal 01netPro, janvier 2009, http://pro.01net.com/editorial/401424/la-securite-des-telephones-sans-fil-dect-remise-en-question/ (consulté le 9 janvier 2010).

[BRE06] A. Bremler-Barr, R. Halachmi-Bekel, et J. Kangasharju, « Unregister Attacks in SIP », In proceedings of the 2nd IEEE workshop on secure network protocols NPSEC, Washington, USA, novembre 2006, pages : 32–37.

[CAIN] Projet CAIN : http://www.oxid.it/cain.html (consulté le 5 avril 2010).

[CHA08] F. Chabaud, « Recherche et développement en sécurité des systèmes d'information : orientations et enjeux », Actes de la conférence SSTIC, Rennes, France, juin 2008.

[CHE09] Z. Chen, S. Guo, K. Zheng, et H. Li, « Research on Man-in-the-Middle Denial of Service Attack in SIP VoIP », In proceedings of the 2009 International Conference on Networks Security, Wireless Communication and Trusted Computing, Wuhan, China, avril 2009, pages : 263-266.

[COMM] Logiciel CommView disponible sur le site : http://www.tamos.com/products/commview/ (consulté le 11 mai).

[DAN01] E.J. Daniel, K.A. Teague, « Performance of FNBDT and Low Rate Voice (MELP) Over Packet Networks », In proceeding of the 35 th Asilomar Conference on Signals, Systems & Computers, Pacific Grove, USA, novembre 2001, pages : 1568-1572.

[DAN02-1] E.J. Daniel, K.A. Teague, R. Sleezer, J. Brewer, J. Raymond, W.J. Beck, J. Hershberger, « The Future Narrow Band Digital Terminal », In proceeding of the IEEE 2002 45th Midwest Symposium on Circuits and Systems, Stillwater, USA, août 2002, volume 2, pages : II-589 à II-592.

[DAN02-2] E.J. Daniel, K.A. Teague, « Simulation of FNBDT over Internet and 3G wireless networks », In proceeding of the IEEE 2002 45th Midwest Symposium on Circuits and Systems, Stillwater, USA, août 2002, volume 2, pages : II-330 à II-333.

[DEO07] S. Déon, « VOIP et TOIP Asterisk. La téléphonie sur IP », Eni édition, décembre 2007, 314 pages.

[DIA07] W. B. Diab, S. Tohme, C. Bassil, « Critical VPN Security Analysis and New Approach for Securing VoIP Communications over VPN Networks », In proceedings of the Third ACM International Workshop on Wireless Multimedia Networking and Performance Modeling (WMuNeP'07), La Cané, Grèce, 22-26 octobre 2007, pages : 92-96.

[DOL83] D. Dolev, and A.C.C. Yao, « On the security of public key protocols », IEEE transactions on information theory, 1983, vol. 30, no2, pages : 198-208.

[DUB07] N. Dubée, « La voix sur IP (VoIP) : une opportunité pour la sécurité ? », Proceeding of SSTIC symposium, juin 2007.

[EBIOS] Méthode EBIOS : http://www.ssi.gouv.fr/site_article45.html.

[FIL06] E. Filiol, et P. Richard, « Cybercriminalité », Dunod, novembre 2006, 212 pages.

[FNBDT] General Dynamics Communication Systems, « FNBDT Signaling Plan rev 1.1 », National Security Agency, septembre 1999.

[GAU09] F. Gauche, « An approach of the Future Narrow Band Digital Terminal protocol », In proceeding of Seventh International Conference on Computer Science and Information Technologies, Yerevan, Armenia, 28 septembre - 2 octobre 2009, pages : 457 – 462.

[GEN05] D. Geneiatakis, G. Kambourakis, T. Dagiuklas, C. Lambrinoudakis, and S. Gritzalis, « SIP Security Mechanisms: A State-of-the-art Review », In the Proceedings of the Fifth International Network Conference (INC 2005), Samos, Grèce, juillet 2005, pages : 147-155.

[GEN08] D. Geneiatakis, et C. Lambrinoudakis, « A lightweight protection mechanism against signaling attacks in a SIP-based VoIP environment », Telecommunication System revue, Volume 36 n° 4 décembre 2007, Springer, février 2008, pages : 153-159.

[GUP07] P. Gupta, V. Shmatikov, « Security Analysis of Voice-over-IP Protocols », In proceedings of 20th IEEE the Computer Security Foundations Symposium 2007, juillet 2007, pages : 49-63.

[H323] Systèmes de communication multimédia en mode paquet. Recommandations UIT-T H.323, juin 2006.

[HAG08] A. M. Hagalisletto, L. Strand, « Formal modeling of authentication in SIP registration », In proceedings of the Second International Conference of Emerging Security Information, Systems and Technologies, Cap Esterel, France, août 2008, pages : 16-21.

[HAL07] S. Hallsteinsen, I. Jøstad, and D. V. Thanh, « Using the mobile phone as a security token for unified », In proceedings of the Second International Conference on Systems and Networks Communications (ICSNC 2007), Cap Esterel, France, août 2007.

[JEM07] C. Jennings, N. Modadugu, « Session Initiation Protocol (SIP) over Datagram Transport Layer Security (DTLS) », IETF Internet Draft, draft-jennings-sip-dtls-05, 10 octobre 2007, 6 pages.

[KUH05] D. Richard Kuhn, Thomas J. Walsh, Steffen Fries, « Security Considerations for Voice Over IP Systems », Recommendations of the National Institute of Standards and Technology NIST, Us department of commerce, janvier 2005, 99 pages.

[LAD06] P. Lasbordes, « La sécurité des systèmes d'information : un enjeu majeur pour la France », La documentation française, janvier 2006, 197 pages.

[LAM81] L. Lamport, "Password Authentication with Insecure Communication", Communications of the ACM 24.11, novembre 1981, pages : 770-772.

[LYN] Produit Lynxphone : http://www.bitlynx.com/lynxphone.php (consulté le 21 mars 2010).

[MAZ06-1] W. Mazurczyk1, et Z. Kotulski, « New VoIP Traffic Security Scheme with Digital Watermarking », Computer Safety, Reliability, and Security, LCNS Volume 4166/2006, Springer Berlin / Heidelberg, 2006, pages : 170-181.

[MAZ06-2] W. Mazurczyk1, et Z. Kotulski, « New security and control protocol for VoIP based on steganography and digital watermarking », Cryptography and Security Annales UMCS, Informatica, Volume 4, 2006, 8 pages.

[MAZ07] W. Mazurczyk1, et Z. Kotulski, « Covert Channel for Improving VoIP Security », Advances in Information Processing and Protection, Springer US, septembre 2007, pages : 271-280.

[MAZ08-1] W. Mazurczyk1, et K. Szczypiorski, « Covert Channels in SIP for VoIP Signalling », In proceedings of the Global E-Security 4th International Conference, ICGeS 2008, Londres, Angleterre, 23-25 juin 2008, pages : 65-72.

[MAZ08-2] W. Mazurczyk1, et K. Szczypiorski, « Steganography of VoIP Streams », In proceedings of the 3rd International Symposium on Information Security (IS'08), Monterrey, Mexico, 10-11 novembre 2008, 18 pages.

[MEG05] J. Van Meggelen, J. Smith, et L. Madsen, « Asterisk, La Téléphonie Open Source », O'Reilly, septembre 2005, 414 pages.

[MEH] Méthode MEHARI : http://www.clusif.asso.fr/fr/production/mehari/ (consulté le 5 avril 2010).

[MIZ05] S. Mizuno, K. Yama, and K. Takahashi, « Authentication Using Multiple Communication Channels », In proceedings of the 2005 Workshop on Digital Identity Management, Fairfax, USA, November 2005, pages : 54-62.

[NAS09] M. Nassar, « Monitorage et Détection d'Intrusion dans les Réseaux Voix sur IP », thèse soutenue le 31 mars 2009, 164 pages.

[NIE09] P. Nie J.-M. Tapio, S. Tarkoma, J. Heikkinen, « Flexible Single Sign-On for SIP: Bridging the Identity Chasm », In proceedings of the IEEE International Conference on Communications ICC'09, Dresde, Allemagne, juin 2009, 6 pages.

[OUA08] L. Ouakil, G. Pujolle, « Téléphonie sur IP », Eyrolles, 2ème Edition, juin 2008, 466 pages.

[PAR08] P. Park, « Voice over IP Security », Ciscopress.com, édition septembre 2008, 361 pages.

[RFC768] J. Postel, « User Datagram Protocol », RFC 768, août 1980.

[RFC793] J. Postel, « Transmission Control Protocol STD 7 », RFC 793, septembre 1981.

[RFC1321] R. Rivest, « The MD5 Message-Digest Algorithm », RFC 1321, avril 1992.

[RFC1760] N. Haller, « The S/KEY One-Time Password System », RFC 1760, février 1995.

[RFC2104] H. Krawczyk, M. Bellare, et R. Canetti , « HMAC: Keyed-Hashing for Message Authentication », RFC 2104, février 1997.

[RFC2246] T. Dierks, and C. Allen, « The TLS Protocol », RFC 2246, janvier 1999.

[RFC2279] F. Yergeau, « UTF-8, a transformation format of ISO 10646 », RFC 2279, janvier 1998.

[RFC2289] N. Haller, C. Metz, P. Nesser, M. Straw, « A One-Time Password System », RFC 2289, février 1998.

[RFC2315] B. Kaliski, « PKCS #7: Cryptographic Message Syntax Version 1.5», RFC 2315, mars 1998.

[RFC2401] S. Kent, and R. Atkinson, « Security Architecture for the Internet Protocol », RFC 2401, novembre 1998.

[RFC2543] M. Handley, H. Schulzrinne, E. Schooler, et J. Rosenberg, « SIP: Session Initiation Protocol », RFC 2543, mars 1999.

[RFC2616] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, and T. Berners-Lee, « Hypertext Transfer Protocol -- HTTP/1.1 », RFC 2616, juin 1999.

[RFC2617] J. Franks, P. Hallam-Baker, J. Hostetler, S. Lawrence, P. Leach, A. Luotonen and L. Stewart, "HTTP Authentication: Basic and Digest Access Authentication", RFC 2617, juin 1999.

[RFC2630] R. Housley, « Cryptographic Message Syntax », RFC 2630, juin 1999.

[RFC2631] E. Rescorla, « Diffie-Hellman Key Agreement Method », RFC 2631, juin 1999.

[RFC2865] C. Rigney, S. Willens, A. Rubens, and W. Simpson, « Remote Authentication Dial In User Service (RADIUS) », RFC 2865, juin 2000.

[RFC3261] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley and E. Schooler, « SIP: Session Initiation Protocol », RFC 3261, juin 2002.

[RFC3310] A. Niemi, J. Arkko and V. Torvinen, « Hypertext Transfer Protocol (HTTP) Digest Authentication Using Authentication and Key Agreeement (AKA) », RFC 3310, septembre 2002.

[RFC3325] C. Jennings, J. Peterson and M. Watson, « Private Extensions to the Session Initiation Protocol (SIP) for Asserted Identity within Trusted Networks », RFC 3325, novembre 2002.

[RFC3550] H. Schulzrinne, S. Casner, R. Frederick and V. Jacobson, « RTP: A Transport Protocol for Real-Time Applications », RFC 3550, juillet 2003.

[RFC3711] M. Baugher, S. Casner, R. Frederick and V. Jacobson, « The Secure Real-time Transport Protocol (SRTP) », RFC 3711, mars 2004.

[RFC3748] B. Aboba, L. Blunk, J. Vollbrecht, J. Carlson, et H. Levkowetz, « Extensible Authentication Protocol (EAP) », RFC 3748, juin 2004.

[RFC3830] J. Arkko, E. Carrara, F. Lindholm, M. Naslund, et K. Norrman., « MIKEY: Multimedia Internet KEYing », IETF RFC 3830, août 2004.

[RFC3851] B. Ramsdell, « Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 3.1 Message Specification », RFC 3851, juillet 2004.

[RFC3853] J. Peterson, « S/MIME Advanced Encryption Standard (AES) Requirement for the Session Initiation Protocol (SIP) », RFC 3853, juillet 2004.

[RFC3893] J. Peterson, « Session Initiation Protocol (SIP) Authenticated Identity Body (AIB) Format », RFC 3893, septembre 2004.

[RFC4226] D. M'Raihi, M. Bellare, F. Hoornaert, D. Naccache, O. Ranen, « HOTP: An HMAC-Based One-Time Password Algorithm », RFC 4226, décembre 2005.

[RFC4347] E. Rescolar, and N. Modadugu, « Datagram Transport Layer Security », RFC 4347, avril 2006.

[RFC4474] J. Peterson, and C. Jennings, « Enhancements for Authenticated Identity Management in the Session Initiation Protocol (SIP) », RFC 4474, août 2006.

[RFC4566] M. Handley, V. Jacobson and C. Perkins, « SDP : Session Description Protocol », RFC 4566, juillet 2006.

[RFC4568] F. Andreasen, M. Baugher, et D. Wing, « Session Description Protocol (SDP) Security Descriptions for Media Streams », RFC 4568, juillet 2006.

[RFC4590] B.Sterman, D. Sadolevsky, D. Schwartz, D. Williams,et W. Beck « RADIUS Extension for Digest Authentication », RFC 4590, juillet 2006.

[ROH] Produit TOP SEC de la société Rohde et Schwarz, documentation disponible sur le lien : http://www2.rohde-schwarz.com/file_12629/TopSec-mobile_dat_en.pdf (consulté le 4 septembre 2009).

[SAW06] S. El Sawda and P. Urien, « SIP Security Attacks and Solutions : A state-of-the-art review », In proceedings of the Information and Communication Technologies 2006, ICTTA'06, April 2006, Vol.2, pages: 3187–3191.

[SCIP] SCIP-210, « SCIP Signaling Plan, Révision 3.0 », General Dynamics, avril 2006 (*document non publique*).

[SHA09] L. Shan, et N. Jiang, « Research on Security Mechanisms of SIP-based VoIP System », In proceedings of the 2009 Ninth International Conference on Hybrid Intelligent Systems, Shenyang, China, August 12-August 2009, pp. 408-410.

[SIN99] S. Singh, « L'histoire des codes secrets », Edition Le livre de poche, publié en 1999, 504 pages.

[SKYPE] Produit Skype : http://www.skype.com/intl/fr/welcomeback/ (consulté le 1er mai 2010).

[SNO] Projet SNOCER « Low Cost Tools for Secure and Highly Available VoIP Communication Services ». Rapport intitulé : « General Reliability and Security Framework for VoIP Infrastructures ». Auteurs : T. Dagiuklas et al, 74 pages.

[SPAN] Projet SPAN : http://www.irisa.fr/celtique/genet/span/ (consulté le 1er mai 2010).

[SRI05] R. Srinivasan, V. Vaidehi V, K. Harish, K. Lakshmi Narasimhan, S. LokeshwerBabu, et V. Srikanth, « Authentication of Signaling in VoIP Applications », In proceedings of the Communications Asia-Pacific Conference, Perth, Australie, Octobre 2005, pages: 530-533.

[SUP97] L. Supplee, R. Cohn, et J. Collura, « MELP : The New Federal Standard at 2400 bps », In proceedings of the IEEE International Conference on Acoustic, Speech and Signal Processing, avril 1997, pages : 1591-1594.

[TAN03] A. Tanenbaum, « Réseaux », Pearson Education France, 4ème Edition, 2003, 908 pages.

[TES05] C. Tessereau, « Protocoles SSL/TLS », Techniques de l'ingénieur, Réf H 5230, 2005, 18 pages.

[THO09] P. Thoniel, « Méthodes d'authentification », Techniques de l'ingénieur, Réf. H 5535, Avril 2009, 24 pages.

[TRI] Produit tribox, IPBX logiciel basé sur un noyau Asterisk : http://www.trixbox.org/ (consulté le 21 mars 2010).

[TSC06] H. Tschofenig, R. Falk, Siemens, J. Peterson, J. Hodges,D. Sicker, et J. Polk, « Using SAML to Protect the Session Initiation Protocol (SIP) », Revue IEEE Network, septembre/octobre 2006, pages: 14-17.

[TWI] Projet Twinkle : http://www.twinklephone.com/ (consulté le 4 septembre 2009).

[WIRE] Logiciel Wireshark disponible sur le site : http://www.wireshark.org/ (consulté le 13 mai 2010).

[ZAR05] J. Zar, "VoIP Security and Privacy Threat Taxonomy", Public Release 1.0, octobre 2005, 36 pages.

[ZRTP10] P. Zimmermann, A. Johnston , et J. Callas, « ZRTP: Media Path Key Agreement for Secure RTP », draft-zimmermann-avt-zrtp-17, janvier 2010.

Publications associées à ces travaux

En conférences internationales

[01] T. Guillet, A. Serhrouchni, M. Badra, « Mutual Authentication for SIP: A semantic meaning for the SIP opaque values », Proceeding of the second International Conference on New Technologies, Mobility and Security, Tanger, November 2008.

[02] M. Badra, T. Guillet, A. Serhrouchni, « Random Values, Nonce and Challenges: Semantic Meaning versus Opaque and Strings of Data », Proceeding of the 70th IEEE Vehicular Technology Conference, Anchorage, September 2009.

[03] T. Guillet, R. Moalla, A. Obaid, A. Serhrouchni, « SIP Authentication based on HOTP », Proceeding of Seventh International Conference on Information, Communications and Signal Processing (ICICS 2009), Macau, China, December 2009.

[04] R. Moalla, A. Serhrouchni, S. Guemara, T. Guillet, « Intégration des mots de passe à usage unique dans SIP », dans les actes de la 10ème Colloque Africain sur la Recherche en Informatique et en Mathématiques Appliquées (CARI 2010), Yamoussoukro, Côte d'Ivoire, Octobre 2010 (à paraitre).

En conférence nationale

[05] T. Guillet, A. Serhrouchni, « Authentification HTTP Digest SIP renforcée », dans les actes de la MAnifestation des JEunes Chercheurs en Sciences et Technologies de l'Information et de la Communication (Majecstic), Avignon, Novembre 2009.

Brevet

[06] Brevet n° FR2928798 « Procédé d'authentification, système d'authentification, terminal serveur, terminal client et programmes d'ordinateur correspondants », Inventeurs : T. Guillet, A. Serhrouchni, et M. Badra, demande de brevet d'invention publiée le 18 septembre 2009.

Annexe I : Codes pour la validation de HTTP Digest SIP renforcée

Cette annexe fournit les codes qui ont permis d'implémenter la proposition de renforcement de HTTP Digest SIP renforcé. La première modification a été réalisée dans le fichier appelé « chan_sip.c » d'ASTERISK :

```
char new_nonce[256];
char chaine_temp[256]; strcpy(chaine_temp,p->callid);
strcat(chaine_temp,secret);
strcat(chaine_temp,"asterisk");
ast_md5_hash(new_nonce,chaine_temp);
ast_string_field_set(p,randdata,new_nonce);
transmit_response_with_auth(p, response, req, p->randdata, reliable, respheader, 0);
sip_scheddestroy(p, DEFAULT_TRANS_TIMEOUT);
return AUTH_CHALLENGE_SENT;
```

La deuxième modification nécessaire pour valider la proposition a été réalisée dans le logiciel Twinkle :

```
string header_auth_str = r->hdr_www_authenticate.get_value();
const char* header_auth = header_auth_str.c_str();
const char* call_id;
const char* pass;
char* nonce_start=strstr(header_auth,"nonce=");
char* nonce_end=strstr(header_auth,",algo");
char val_verif[80];
char val_code[80];
t_user* current_user;
current_user=get_user_profile();
int i=nonce_end-nonce_start-6;
char nonce[i];
strncpy(nonce,nonce_start+7,i);
nonce[i-2]='\0';
call_id=r->hdr_call_id.get_value().c_str();
 pass=current_user->get_auth_pass().c_str();
strcpy(val_verif,call_id);
strcat(val_verif,pass);
strcat(val_verif,"asterisk");
ast_md5_hash(val_code, val_verif);
if(strcmp(nonce,val_code)==0) ui->cb_show_message("Serveur Authentifié");
else ui->cb_show_message("Serveur Non Authentifié");
return;
```

Annexe II : Code en C d'HOTP

```
/* Generation du HOTP*/
void
hotp(const u_char *key, size_t keylen, u_long counter, int ndigits, char *buf10, char
*buf16, size_t buflen)
{
   const int max10 = sizeof(powers10) / sizeof(*powers10);
   const int max16 = 8;
   const EVP_MD *sha1_md = EVP_sha1();
   u_char hash[EVP_MAX_MD_SIZE];
   u_int hash_len;
   u_char tosign[8];
   int offset;
   int value;
   int i;
   /* Encode counter */
   for (i = sizeof(tosign) - 1; i >= 0; i--) {
      tosign[i] = counter & 0xff;
      counter >>= 8;
   }
   /* Compute HMAC */
   HMAC(sha1_md, key, keylen, tosign, sizeof(tosign), hash, &hash_len);
   /* Extract selected bytes to get 32 bit integer value */
   offset = hash[hash_len - 1] & 0x0f;
   value = ((hash[offset] & 0x7f) << 24) | ((hash[offset + 1] & 0xff) << 16)
      | ((hash[offset + 2] & 0xff) << 8) | (hash[offset + 3] & 0xff);
   /* Sanity check max # digits */
   if (ndigits < 1)
      ndigits = 1;
   /* Generate decimal digits */
   if (buf10 != NULL) {
      snprintf(buf10, buflen, "%0*d", ndigits < max10 ? ndigits : max10,
        ndigits < max10 ? value % powers10[ndigits - 1] : value);
   }
   /* Generate hexadecimal digits */
   if (buf16 != NULL) {
      snprintf(buf16, buflen, "%0*x", ndigits < max16 ? ndigits : max16,
        ndigits < max16 ? (value & ((1 << (4 * ndigits)) - 1)) : value);
   }
}
```

Annexe III : Descriptions en HLPSL réalisées pour cette thèse

Cette annexe présente la description en HLPSL de l'authentification HTTP Digest de SIP et de notre proposition de renforcement décrite dans le chapitre 4.

Description de l'authentification HTTP Digest de SIP en HLPSL :

```
role sip_server(SS,UAC : agent, PWD : text, H : hash_func, SND, RCV : channel(dy))
played_by SS def=
local       State                   : nat,
            Nonce       : text
init State := 1
transition
0. State = 1
/\ RCV(sipregister.UAC') =|>
State' := 2
            /\ Nonce' := new()
            /\ SND(sip401.Nonce')
1. State = 2
            /\ RCV(sipregister.UAC.Nonce.H(Nonce.H(UAC.PWD))) =|>
            State' := 3
            /\ SND(sip200)
            /\ request(SS,UAC,y,sipregister.UAC.Nonce.H(Nonce.H(UAC.PWD)))
end role

role user_agent_client(UAC,SS : agent, PWD : text, H : hash_func, SND, RCV :channel(dy))
played_by UAC def=
local       State                   : nat,
            Nonce       : text
init State := 1
transition
2.          State = 1
            /\ RCV(start) =|>
State':=2
            /\ SND(sipregister.UAC)
3. State = 2
            /\ RCV(sip401.Nonce') =|>
            State':= 3
            /\ SND(sipregister.UAC.Nonce'.H(Nonce'.H(UAC.PWD)))
            /\ witness(UAC,SS,y,H(Nonce'.H(UAC.PWD)))
 4. State = 3
            /\ RCV(sip200) =|>
            State':= 4
end role

role session(UAC,SS:agent, H:hash_func, PWD:text)
def= local   SND, RCV  : channel(dy)
composition
            sip_server(SS,UAC,PWD,H,SND,RCV)
  /\        user_agent_client(UAC,SS,PWD,H,SND,RCV)
end role

role environment() def=
const       uac, ss                             : agent,
            h                                   : hash_func,
            y                                                       : protocol_id,
            sipregister             : protocol_id,
        sip401,sip200    : protocol_id,
        pwd                 : text
intruder_knowledge = {uac,ss,sipregister,sip401,sip200,h,i}
composition session(uac,ss,h,pwd)
end role

goal
authentication_on y
end goal

environment()
```

Description de l'authentification renforcée de SIP en HLPSL :

```
role sip_server(SS,UAC : agent, PWD : text, H : hash_func, SND, RCV : channel(dy))
played_by SS def=
local       State                                                   : nat,
            Callid, Realm           : text
init State := 1
transition
0.          State = 1
            /\ RCV(sipregister.UAC'.Callid') =|>
            State' := 2
            /\ Realm' := new()
            /\ SND(sip401.UAC.Callid.Realm'.H(H(UAC.Realm'.PWD).Callid))
            /\ witness(SS,UAC,yy,H(H(UAC.Realm'.PWD).Callid))
1. State = 2
/\RCV(sipregister.UAC.Callid.H(H(H(UAC.Realm.PWD).Callid').PWD.UAC)) =|>
            State' := 3
            /\ SND(sip200)
            /\ request(SS,UAC,y,H(H(H(UAC.Realm.PWD).Callid).PWD.UAC))
end role

role user_agent_client(UAC,SS : agent, PWD : text, H : hash_func, SND, RCV :channel(dy))
played_by UAC def=
local       State                                       : nat,
            Callid, Realm           : text
init State := 1
transition
2.          State = 1
            /\ RCV(start) =|>
            State':=2
            /\ Callid' := new()
            /\ SND(sipregister.UAC.Callid')
3.          State = 2
            /\ RCV(sip401.UAC.Callid.Realm'.H(H(UAC.Realm'.PWD).Callid)) =|>
            State':= 3
/\SND(sipregister.UAC.Callid.H(H(H(UAC.Realm.PWD).Callid).PWD.UAC))
/\ witness(UAC,SS,y,H(H(H(UAC.Realm.PWD).Callid).PWD.UAC))
            /\ request(UAC,SS,yy,H(H(UAC.Realm.PWD).Callid))
4.          State = 3
            /\ RCV(sip200) =|>  State':= 4
end role

role session(UAC,SS:agent, H:hash_func, PWD:text)
def= local SND, RCV  : channel(dy)
composition
            sip_server(SS,UAC,PWD,H,SND,RCV)
/\          user_agent_client(UAC,SS,PWD,H,SND,RCV)
end role

role environment() def=
const       uac, ss                                                                 : agent,
            h                                                                       : hash_func,
            y,yy                                                                    : protocol_id,
      sip401, sip200, sipregister   : protocol_id,
      pwd                                                   : text
intruder_knowledge = {uac,ss,sipregister,sip401,sip200,h,i}
composition session(uac,ss,h,pwd)
end role

goal
authentication_on y
authentication_on yy
end goal
environment()
```

Annexe IV : Brevet issu de l'analyse de l'authentification HTTP Digest SIP

1. Préambule

L'analyse menée sur l'authentification dans l'environnement SIP a permis d'établir un principe de renforcement de l'authentification générique. Cette méthode a été formalisée dans le cadre d'un brevet rédigé en collaboration avec M Mohamad Badra du laboratoire LIMOS-UMR 6158 (Laboratoire d'Informatique, de Modélisation et d'Optimisation des Systémes) de Clermont Ferrand sous la direction de M Ahmed Serhrouchni. Le principe établi pour SIP a été généralisé à l'ensemble des authentifications simples issues du modèle client/serveur basé sur un challenge réponse.

Le brevet a été enregistré auprès de l'INPI sous le numéro FR2928798. La demande de brevet d'invention a été publiée le 19 septembre 2009.

2. Analyse du contexte

Les protocoles de sécurité sont développés pour permettre, entre autres services, une authentification entre les terminaux communicants, de chiffrer et protéger les données applicatives échangées entre ces mêmes terminaux et de contrôler l'accès aux ressources et aux services du réseau.

En ce qui concerne l'authentification, deux modes sont possibles : authentification mutuelle ou authentification à sens unique. Dans le cadre de l'authentification mutuelle chaque terminal authentifie l'autre terminal et inversement. Quant au deuxième mode, l'authentification est asymétrique dans le sens où qu'un seul terminal est authentifié.

L'authentification à sens unique expose le terminal authentifié à plusieurs attaques ; notamment l'attaque de l'homme au milieu. L'attaque de l'homme au milieu consiste pour un tiers à s'interposer dans une communication sans que les terminaux concernés n'en aient conscience. C'est une attaque dans laquelle l'attaquant est capable de lire, insérer et modifier comme il le souhaite les messages chiffrés entre deux terminaux, sans que ni l'une ni l'autre ne puisse se douter que la session ou la connexion entre eux ait été compromise.

Tous les protocoles de sécurité proposent une authentification à sens unique et peu de ceux-ci implémentent une authentification mutuelle. A titre d'exemple non limitatif, le protocole SSL (Secure Sockets Layer) supporte les deux modes d'authentification cités auparavant, tandis qu'EAP-MD5 (Extensible Authentication Protocol-Message Digest5), CHAP (Challenge Handshake Authentication Protocol), les mécanismes de défi/réponse (utilisés notamment avec les réseaux GSM (Global System for Mobile), WLAN (Wireless Local Access Network) ou encore avec les applications du monde IP telles SIP (Session Initiation Protocol), WEB, courrier électronique, etc.

L'authentification simple se décrit de manière générique suivante (cf. figure 69) :

- Une requête (8) envoyée par le client (2) au serveur terminal (4) ;
- Une étape de génération d'un défi (10) par (4) ;
- Une étape de transmission du défi (10) de (4) vers (2) à travers le réseau ;
- Une étape de calcul par (2) d'une réponse (14) au défi (10). Cette étape comprend un calcul à partir d'une fonction(12), du défi (10) et du secret (6). (6) a été partagé entre (2) et (4) auparavant ;
- Une étape de transmission de (14) par (2) vers (4) à travers le réseau ;
- Une étape de calcul par (4) d'une réponse (16) au défi (10) avec la fonction (12) sur comprenant (10) et (6) ;
- Une étape de comparaison (18) par (4) entre (14) et (16) ;
- Une étape d'authentification du terminal client par le terminal serveur si la première et la deuxième réponse sont concordantes (20), sinon (22).

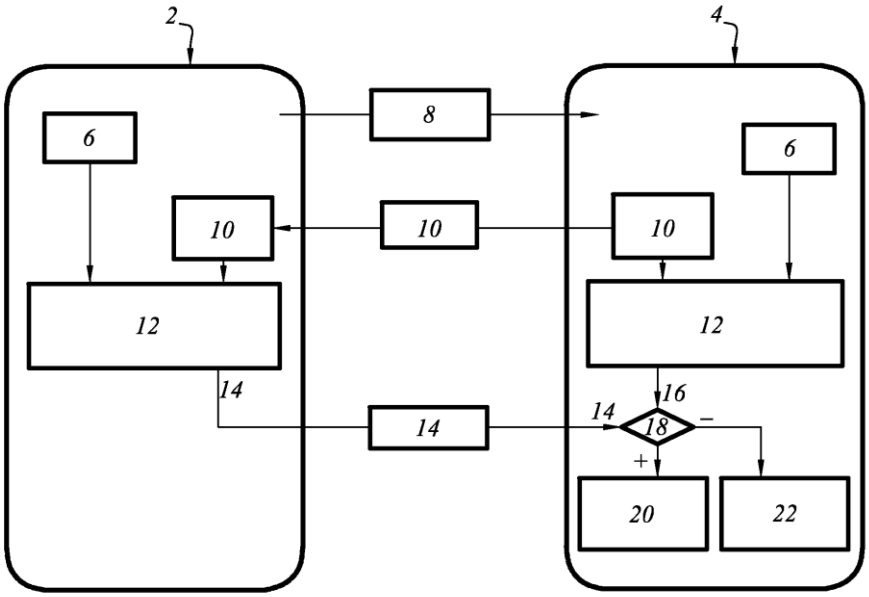


Figure 69. Principe générique d'une authentification simple

Le brevet propose donc de permettre au client de pourvoir authentifier le serveur en rétrofitant l'authentification simple par le détournement de la valeur aléatoire.

3. Résumé de l'invention

La présente invention concerne un procédé d'authentification entre un terminal client et un terminal serveur raccordés à un réseau de transmission d'informations. Ces deux entités partagent un secret (6). L'évolution par rapport à l'authentification simple objet de la figure 74 est que (10) n'est plus généré de manière aléatoire mais l'objet d'une structuration. Désormais (10) fait l'objet d'une construction à partir d'une valeur aléatoire (40), de divers paramètres (44) connus de (2) et (4). (42) et (46) sont des fonctions mathématiques ou cryptographiques qui permettent de générer (10) de manière complètement structurée. (10) est transmis par (4) vers (2). Auparavant (10) était utilisé

directement pour construire la réponse au challenge. Désormais le client vérifie (10) avant de répondre en vérifiant qu'il correspond bien à la structuration attendue. (50) correspond au processus de comparaison : l'état (52) correspond au serveur authentifié, l'état (54) correspondant à l'échec de l'authentification. (2) enchaine ensuite sur le processus d'authentification décrit précédemment. Dans la mesure où le client ne supporte pas l'extension. Il passe directement au processus (12) pour calculer la réponse (14). La figure 70 reprend les principales étapes de l'authentification renforcée proposée par le brevet.

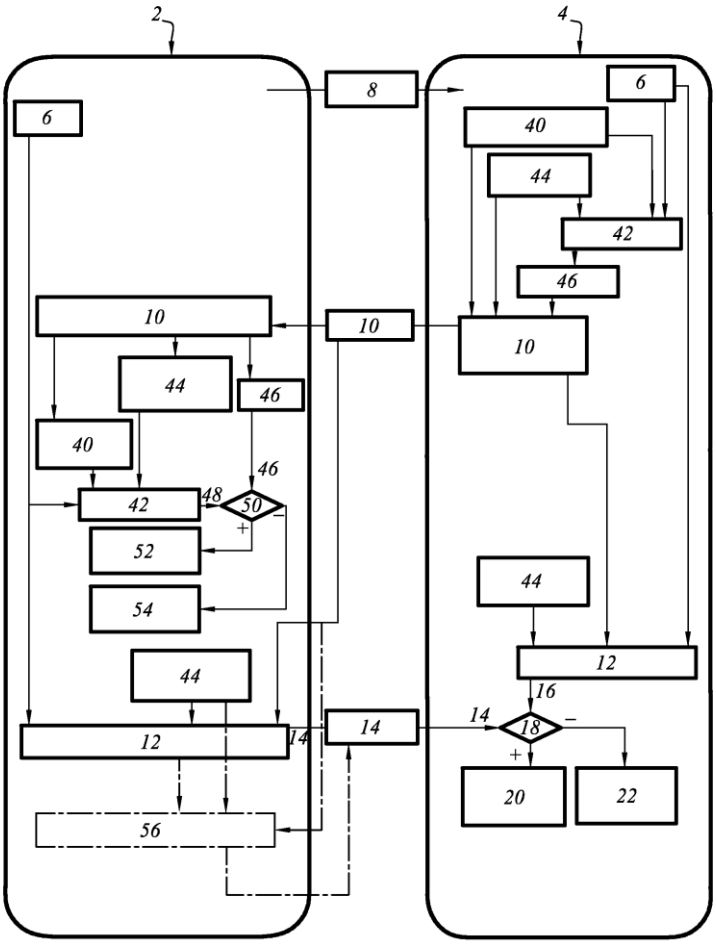


Figure 70. Principe de l'authentification mutuelle objet du brevet

Annexe V : Impact de la sécurité du flux de voix sur les performances réseaux

Dans une architecture SIP, le flux voix est transporté par le protocole RTP. Une des manières de protéger ce flux est d'utiliser SRTP. Pour illustrer l'impact de la sécurité sur les performances réseau, les tailles des paquets RTP et SRTP ont été mesurées sur une plate forme de téléphonie.

1. L'architecture de téléphonie

Pour illustrer l'impact de la sécurité sur les performances réseau, une plate-forme a été mise en place pour mesurer l'occupation la taille des paquets de voix (RTP et SRTP). L'architecture de téléphonie mis en place est à base de logiciel C3X pour l'IPBX et les softphones (cf. figure 76). Les mesures de performances ont été réalisées avec le logiciel CommView.

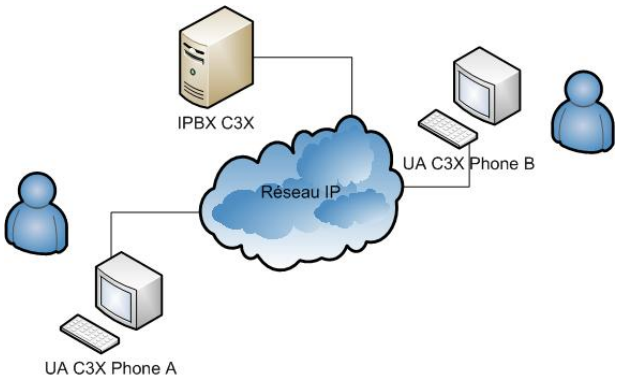
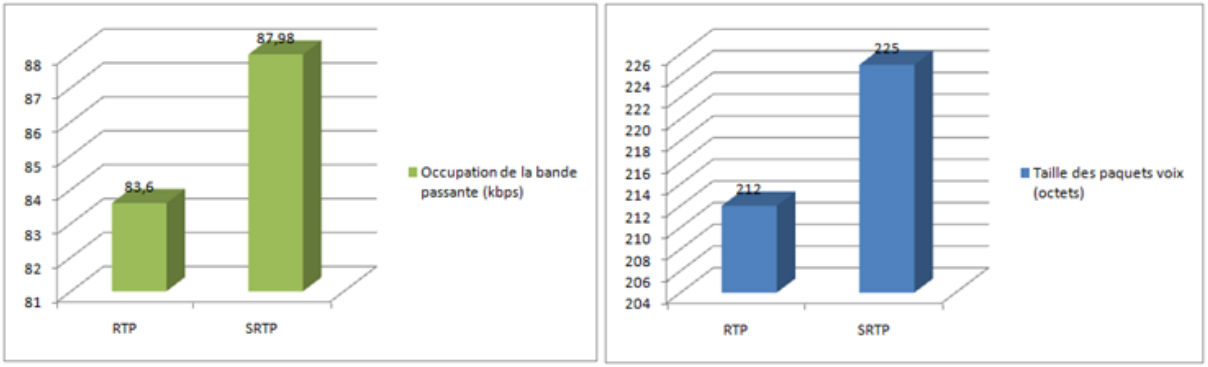


Figure 71. Dispositif pour la mesure de trafic voix

2. Les mesures

Deux mesures ont été réalisées une sur un flux RTP et une sur un flux SRTP. Les résultats sont les suivants :



Dans le cas de présent, une augmentation de 5% du trafic est constatée avec l'utilisation de SRTP. Cette mesure ne se veut pas être une référence mais une simple illustration de l'augmentation des paquets de voix avec le chiffrement.

Annexe VI : Fichiers de configuration d'Asterisk pour les validations

Plusieurs fichiers de configuration ont été utilisés pour la validation. Des exemples sont donnés ci-dessous :

- Exemple de configuration pour le plan d'adressage dans le fichier « EXTENSIONS.conf » :

```
[general]

[default]
include => voicemail
include => pabx
exten => 100,1,Dial(sip/thomas)
exten => 200,1,Dial(sip/ahmed)
```

- Exemple de configuration pour les comptes SIP dans le fichier « SIP.conf » :

```
[general]
context=default
port=5060
srvlookup=yes

[thomas]
type=friend
context=default
username=thomas
callerid="thomas" <100>
secret=tom
callgroup=1
pickupgroup=1
host=dynamic
nat=no
allow=gsm

[ahmed]
type=friend
context=default
username= ahmed
callerid=" ahmed " <200>
secret=abcd
callgroup=1
pickupgroup=1
host=dynamic
nat=no
allow=gsm
```

Annexe VII : Interface graphique des softphones et des outils de validation

Les softphones :

3CX Twinkle

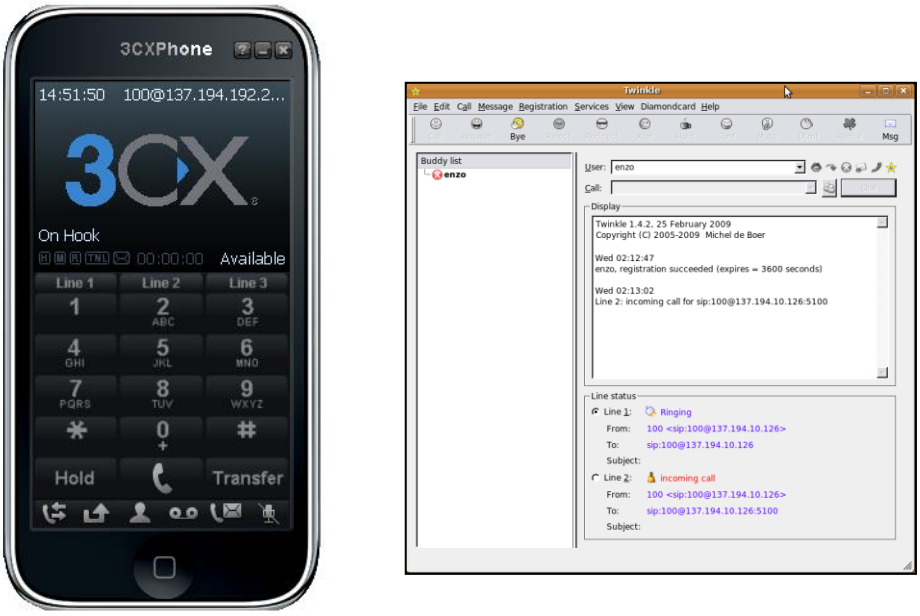


Figure 72. Interfaces graphiques de 3CX et de Twinkle

Wireshark (l'analyseur de protocole) :

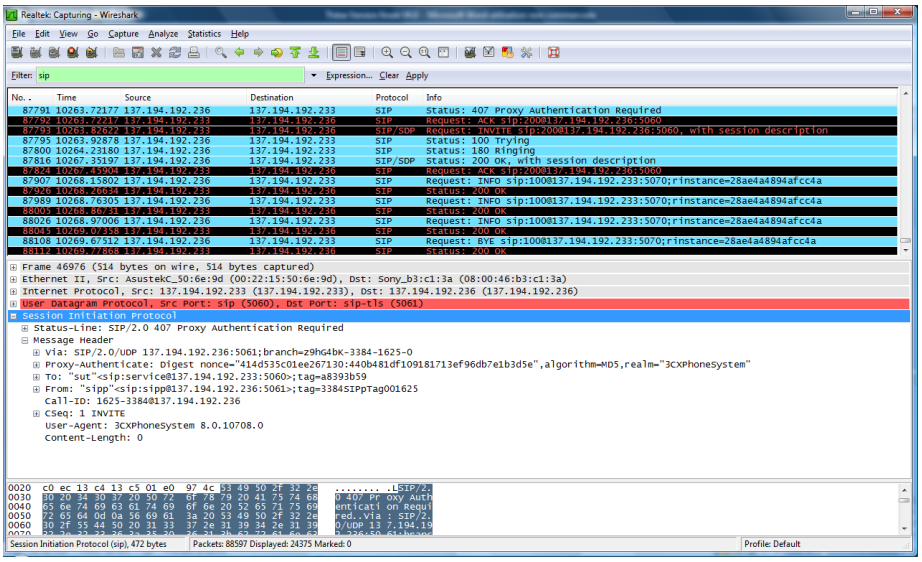


Figure 73. Interface graphique de Wireshark

CommView (sniffer orienté téléphonie sur IP) :

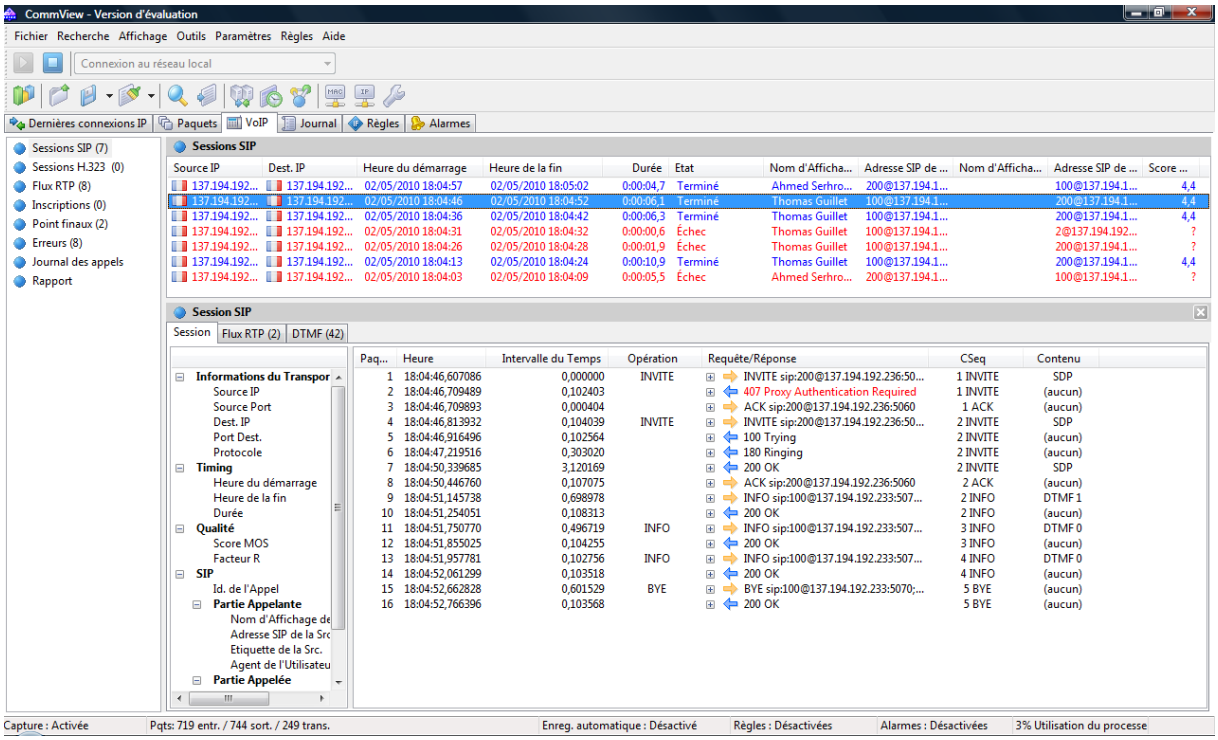


Figure 74. Interface graphique de CommView

SPAN (Security Protocol ANimator for AVISPA) :

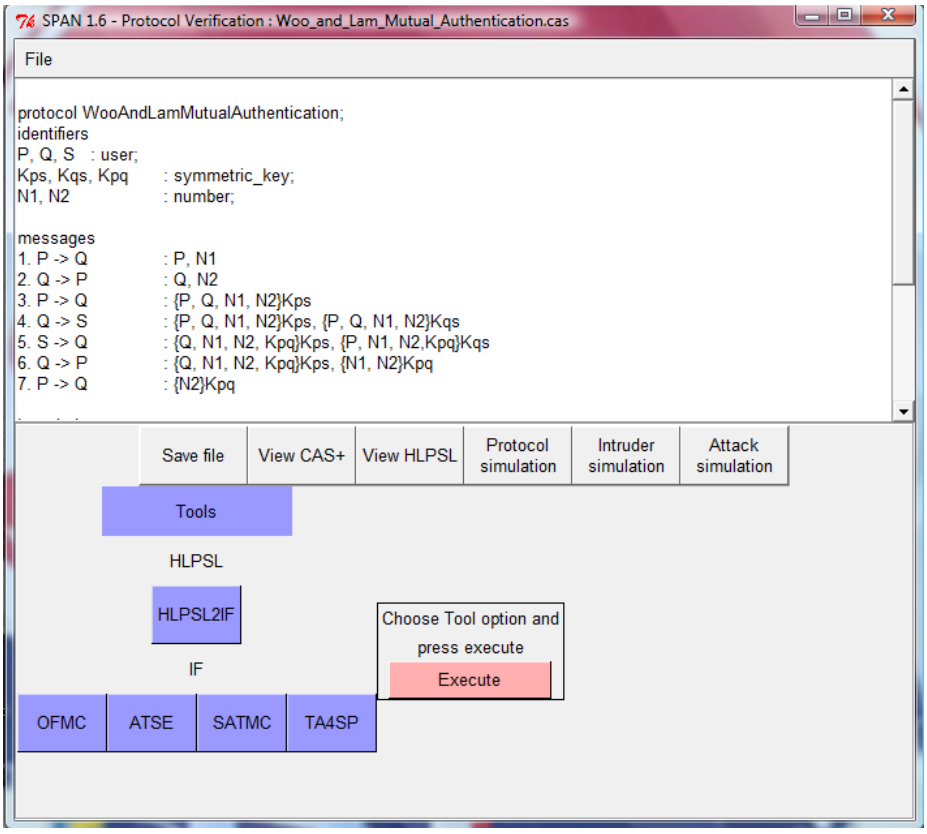


Figure 75. Interface graphique du logiciel SPAN

SIPNess :

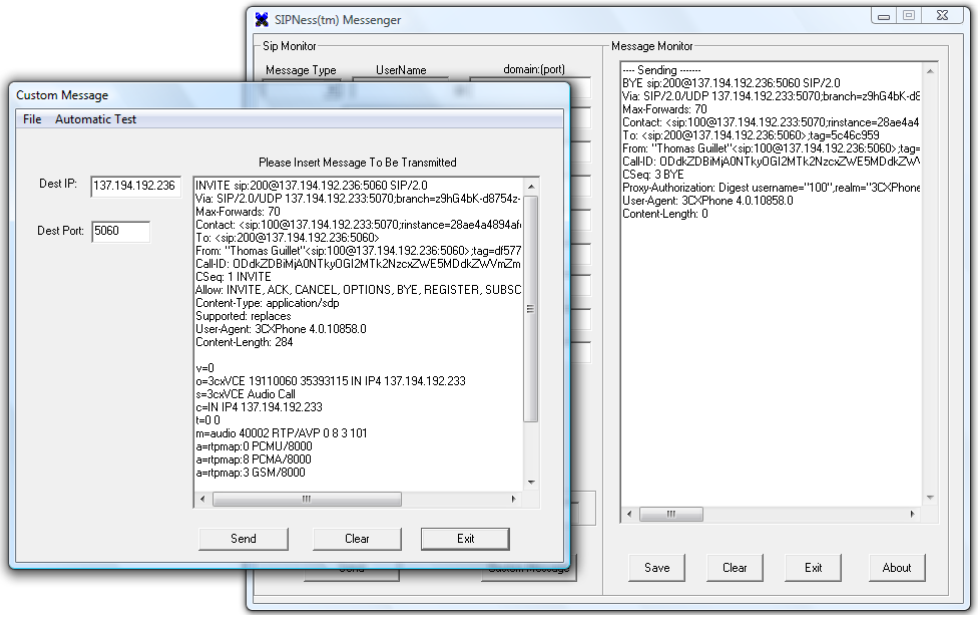


Figure 76. Interface graphique de SIPNess