hakin9

# Sécurité pour le systeme Voice over IP – protocoles SIP et RTP

Tobias Glemser, Reto Lorenz

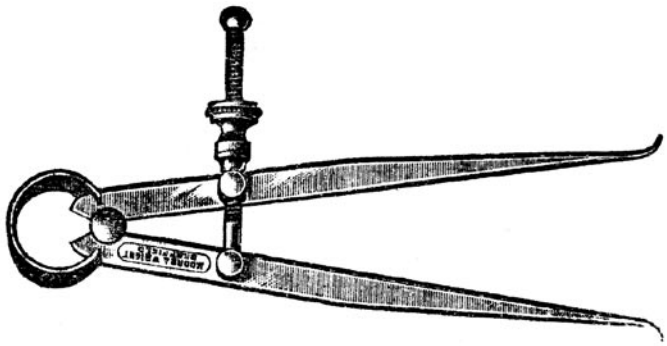
Article publié dans le numéro 5/2005 du magazine *hakin9*



Magazine *hakin9*, Software – Wydawnictwo, ul. Piaskowa 3, 01-067 Varsovie, Pologne, *fr@hakin9.org*

# Sécurité pour le système Voice over IP – protocoles SIP et RTP

Tobias Glemser, Reto Lorenz



**Depuis le salon CeBit tenu en mars 2005, le système appelé Voice over IP (VoIP, Voix sur IP) est non seulement devenu le mot à la mode dans le domaine informatique mais représente également un grand espoir pour les fournisseurs et les fabricants. Ce système VoIP sera bientôt prêt à remplacer les solutions de téléphonie stationnaires. Mais les dangers que représente ce nouveau système passent inaperçus dans la jungle des offres.**

Aujourd'hui, la technologie VoIP fait partie intégrante des offres d'accès Internet à haute débit. Le succès de cette technologie s'explique également par les appels gratuits entre les utilisateurs de VoIP d'un même fournisseur et les offres bon marché et tout compris pour l'interface vers les systèmes de téléphonie classique. Les utilisateurs de petites et moyennes entreprises ne sont pas les seuls à avoir découvert VoIP, les entreprises s'intéressent également à l'immense synergie potentielle de cette technique. Les entreprises peuvent connecter des filiales au moyen d'une seule fibre optique leur permettant de communiquer à la fois des données et des messages oraux. Quelque soit l'endroit où se trouvent les employés, ces derniers sont joignables par un seul numéro de téléphone et en raison de l'utilisation partagée de l'infrastructure réseau, les coûts de maintenance et d'investissements se résument à l'installation et la mise en marche de composants réseaux actifs et passifs. Les problèmes basiques ne commencent qu'à être évoqués *une fois* la technique répandue par les fabricants (comme d'habitude). Au tout début, ces derniers élaborent d'excellentes stratégies de migration et des services surtaxés.

Les média ont rendu tristement célèbre l'histoire d'une adolescente de treize ans, décédée parce que le numéro des urgences américaines (le 911) n'avait pas été routé dans le réseau VoIP que sa mère utilisait. Il n'existe aucune disposition légale relative au routage des appels d'urgence dans la plupart des pays, mais le sujet est actuellement débattu.

## Cet article explique...

- connaissances de base des protocoles SIP (*Session Initiation Protocol*),
- plusieurs techniques d'attaques concrètes et possibles dirigées contre les utilisateurs et fournisseurs de VoIP.

## Ce qu'il faut savoir...

- connaissances de base des protocoles réseaux,
- vous êtes censés savoir réaliser des attaques vers un LAN commuté au moyen d'un empoisonnement ARP,
- vous devriez connaître la famille des protocoles modernes de télécommunication.

Focus

## SIP – le strict nécessaire

Les paquets SIP contiennent les paramètres nécessaires au réglage initial d'appel. Tous les autres paramètres, comme les attributs pour une connexion RTP, sont inclus dans le protocole appelé SDP (en anglais *Session Description Protocol*), intégré sous forme de corps de paquet dans les paquets SIP. Les paquets SIP sont divisés en paquets requêtes et paquets réponses. Les messages sont quant à eux codés en UTF-8 standard. En raison du recours à ce codage, ces messages sont directement lisibles en l'absence d'autres mécanismes de sécurité.

Les messages SIP sont très semblables au HTTP. Les champs d'en-tête exigés pour chaque message requête sont exposés dans le Tableau 1. En dehors de ces éléments de protocole, vous pouvez constater que les définitions du protocole ont pour rôle d'assister une communication en relation avec le contexte, même si celle-ci est envoyée au moyen d'un protocole sans état comme UDP.

Les composants SIP désormais décrits, vous pouvez aborder les messages de requêtes littérales (voir le Tableau 2), qu'il faut diviser en plusieurs méthodes de requêtes. Les nouvelles méthodes de requêtes peuvent venir améliorer le protocole SIP, mais vous n'allez évoquer que les méthodes basiques (si vous souhaitez en connaître de plus particulières, veuillez consulter les excellents documents RFC). Ces méthodes et leur message correspondant constitueront une première introduction, dont le but est de vous montrer vers où sont dirigées des attaques de différentes sortes. On renonce à décrire les différentes classes de réponse et leur utilisation dans cet article.

Les messages sont intégrés dans le contexte de la communication. Les segments contextuels sont désignés sous le nom de *dialogs* (*dialogues*) et de *transactions*, où les dialogues peuvent contenir plusieurs transactions. Ainsi, par exemple, un appel effectué au moyen de VoIP est considéré comme un dialogue SIP, composé des transactions `INVITE`, `ACK` et `BYE`. Les agents utilisateurs concernés doivent être capables de stocker le statut d'un dialogue pendant une période de temps assez longue afin de pouvoir générer des messages dotés des paramètres appropriés.

En raison de la présence des *dialogues*, il existe de nombreux paramètres, autres que `Call-ID`, par exemple *balise* (*tag)* et *branche* (*branch)*. Il n'est guère pertinent d'approfondir ces paramètres tout de suite. Contentez vous pour l'instant de noter que les cohérences systématiques entre les valeurs contextuelles spécifiques et le comportement des agents utilisateurs connectés ne sont pas aussi transparentes que d'autres définitions du protocole SIP. Ce qui explique en partie les implémentations non fiables et problématiques qui présentent des failles de sécurité.

Après une commutation d'appel réussie vers le serveur mandataire (ou proxy) SIP, la communication en voix réelle est réalisée au moyen du protocole RTP. Grâce au codage ainsi échangé, les messages vocaux sont transférés entre les deux partenaires (si possibilité de communication IP directe). Le serveur mandataire SIP n'est requis que pour la libération de la communication.

En plus d'une telle déficience organisationnelle, il existe un grand nombre d'attaques dirigées contre les infrastructures techniques. Avant de les aborder, il est essentiel de bien comprendre le fonctionnement de base de la sécurité sur le protocole SIP (en anglais *Session Initiation Protocol*). Nous nous concentrerons sur le protocole SIP, dans la mesure où le développement se dirige clairement du protocole H.323 vers le SIP.

Mais il ne s'agit pas non plus d'évoquer les tâches réalisables grâce au protocole (voir l'Encadré *SIP – le strict nécessaire*). Le présent article a pour objectif de démontrer comment des attaques peuvent être menées contre la technologie VoIP et comment s'en protéger. Les attaques décrites dans le présent article proviennent d'un environnement VoIP couramment utilisé, dont le protocole de signalisation est le SIP. Par ailleurs, les attaques sont basées sur des méthodes elles aussi couramment utilisées. Ne seront donc pas évoquées, dans le présent article, les méthodes aux implémentations particulières.

## Le protocole SIP et ses avatars

Lorsque le problème de la communication est évoqué, il faut parler dans le cas de la voix sur IP de plusieurs protocoles, chargés de régler et de terminer un appel. On distingue en général différents partenaires de communication pour la signalisation, le transfert vocal ou les messages passerelles. Contrairement à votre bon vieux téléphone, où la communication est gérée à partir d'un seul câble selon le point de vue de l'utilisateur, il existe différentes voies de communications avec la technologie VoIP. Voici ci-dessous les protocoles parmi les plus importants :

- pour la signalisation : SIP et SDP pour l'échange de propriétés de la transmission en continu,
- pour le transport : UDP, TCP, SCTP,
- pour la transmission en continu : RTP, sRTP, RTCP,
- pour les passerelles : SIP, MGCP.

Les protocoles mentionnés plus haut fournissent les fonctions de base pour une utilisation de VoIP et sont de plus en plus utilisés dans d'autres implémentations. Bien sûr, il existe d'autres protocoles, mais les évoquer ici serait trop long et dépasserait le sujet du présent article.

Afin de mieux évaluer les attaques possibles, il serait utile d'observer le paramétrage d'un appel basique. Vous n'utiliserez qu'un seul serveur mandataire SIP dans les exemples qui suivront. Ce serveur mandataire fait partie de la commutation de signalisation et de composition du numéro. Dans la pratique, il y a en général deux ou plus serveurs mandataires SIP de commutation, notamment si les deux appelants ne se trouvent pas sur le même environnement réseau. Dans le cas de plusieurs serveurs mandataires, les messages SIP sont également échangés entre ces serveurs mandataires, de manière à former plus de couches de communication. Avant d'entrer dans les détails, vous pourrez observer dans la Figure 1 une présentation générale de ces phénomènes. Les fonctionnalités des protocoles utilisés n'ont rien d'extraordinaire. Au contraire, le protocole SIP

**Tableau 1.** *Champs d'en-tête de requêtes SIP*

| Champ d'en-tête | Descriptif |
|---|---|
| `Request-URI` | Contient la méthode désignée sous le nom de `Request-URI` ainsi que la version SIP utilisée. La méthode `Request-URI` est généralement composée de la même adresse que le champ appelé `To` (exception : méthode `REGISTER`). |
| `To` | Il s'agit de la cible d'un message et de la méthode de liaison. La cible est un récipient logique, car au début de l'opération, il n'est pas sûr que le message atteigne le récipient nommé. Selon le contexte de la communication, la valeur de la balise peut également être jointe. |
| `From` | Identification logique de l'émetteur de la requête. Le champ `From` doit contenir la valeur de la balise, choisie par le client. |
| `CSeq` | Permet de contrôler l'ordre du message dans la portée d'une transaction. `CSeq` est composé d'une valeur entière et d'un identifiant de la méthode de requête. |
| `Call-ID` | Une valeur `Call-ID` est censée être non récurrente, comme pour l'ensemble des messages d'un dialogue. Cette valeur est censée être choisie au moyen de méthodes cryptographiques. |
| `Max-Forwards` | Ce paramètre est utilisé pour éviter les situations de boucles. En cas d'absence de critère pour certaines valeurs, la valeur 70 doit être choisie. |
| `Via` | Ce champ montre la manière de transmission et l'emplacement vers lequel devrait être envoyée une réponse. Ce champ doit contenir la valeur `Branch-ID` non récurrente pour l'agent utilisateur pertinent. L'identifiant `Branch-ID` débute toujours par *z9hG4bK* et indique que la transaction débute par cette requête. |

**Tableau 2.** *Méthodes des en-têtes de requêtes SIP*

| Méthode | Descriptif |
|---|---|
| `REGISTER` | Avec cette méthode, un client peut s'inscrire ou annuler son inscription d'un serveur mandataire. Il est donc prêt et disponible pour la communication VoIP. Pour annuler l'inscription, la valeur pour la période est réglée sur `0`. |
| `INVITE` | Il s'agit de la méthode la plus importante. Sans elle, vous n'auriez pas besoin du protocole SIP. Toutes les méthodes de traçage lui sont subordonnées, même si elles sont toutes utilisées seules. |
| `ACK` | Si un appel (par exemple une vidéo conférence) est paramétré, c'est une requête `ACK` indépendante qui en prend connaissance au final. Juste après la requête `ACK`, la connexion en continu est paramétrée. |
| `BYE` | Ce message a pour but de terminer un appel de manière régulière. Grâce à ce message, il devient possible de mettre fin à une transaction établie au moyen de la méthode `INVITE`. Un message `BYE` est traité au moyen du paramètre du dialogue approprié (`Call-ID` ou balise). |
| `CANCEL` | Avec `CANCEL`, une connexion établie peut être interrompue avant que l'appel ne soit établi. Cette méthode est également utilisée dans des situations d'erreurs. |
| `OPTIONS` | Cette méthode de requête est utilisée afin d'échanger les méthodes de requêtes supportées ou les attributs des médias pour la transmission. |
| `NOTIFY` | Méthode de requête supplémentaire définie dans le document RFC 3265. Elle permet d'échanger les messages de statut d'une ressource vers laquelle un client est connecté. Par exemple, le client peut recevoir une indication d'arrivée de nouveaux messages vocaux. |

a recours à des techniques largement connues comme par exemple les éléments fondamentaux de HTTP. Le protocole RTP a été défini il y presque 10 ans maintenant, et sa dernière mise à jour date de 2003.

## Attaques SIP/ARP dirigées contre VoIP

Il existe plusieurs attaques dont les exigences sont différentes pour la personne malveillante. Vous allez étudier ici sept des attaques les plus répandues, les plus efficaces et les plus connues pour apprendre à les utiliser par vous-mêmes.

La vulnérabilité de VoIP, par rapport au service téléphonique ordinaire, s'explique en partie par l'utilisation d'un *média partagé*. En effet, aucune ligne destinée à la transaction d'un appel n'est disponible, mais seulement un réseau exploité par de nombreux utilisateurs et un grand nombre d'applications différentes. Ce qui permet bien sûr à une personne malveillante de s'introduire plus facilement dans la communication, au moyen des systèmes informatiques.

Aspirer des appels téléphoniques et les rediffuser devant les partenaires de communication est sans aucun doute l'une des attaques

sur VoIP les plus impressionnantes. Comme mentionné plus haut, la signalisation est réalisée au moyen d'un serveur mandataire SIP, et la communication entre plusieurs partenaires est elle-même possible grâce à la technique *peer-to-peer*. Dans cet exemple, vous souhaitez écouter l'appel entre Alice et Bob. Pour ce faire, vous devez lancer ce qu'on appelle en anglais une attaque *man-in-the-middle* au moyen d'un empoisonnement ARP (voir *Détection du sniffing dans les réseaux commutés*, *hakin9* N° 4/2005) dans le but de convaincre à la fois le serveur mandataire et les téléphones VoIP d'Alice et de Bob qu'ils veulent communiquer avec vous et non entre eux.

Vous trouverez dans la Figure 2 un plan illustrant l'aspiration d'une transmission VoIP. Tout d'abord, l'appel est paramétré. Alice envoie la requête pour appeler Bob au serveur mandataire SIP. Ce message est intercepté puis transmis à la personne malveillante. Le serveur mandataire SIP tente désormais de joindre Bob pour lui indiquer qu'Alice souhaite l'appeler. Ce message est également intercepté puis transmis à la personne malveillante. Après une initialisation de l'appel réussie, l'appel actuel (qui a recours au protocole RTP) entre Alice et Bob commence. Cette communication RTP est également interceptée puis transmise par la personne malveillante.

Si vous faites appel à un outil comme Ethereal pour aspirer une communication, vous recevrez également les données utiles RTP en continu. Si vous souhaitez retenter l'expérience, vous pouvez charger les données aspirées dans un décodeur vocal de type *Firebird DND-323 Analyzer* ou avoir directement recours à Ethereal si G.711 U-law (PCMU) ou G.711 A-law (PCMA) ont été utilisés en tant que codecs (il s'agit des transmissions téléphoniques internationales de codage et de décodage standards).

Il existe, en plus de l'empoisonnement ARP, un petit outil pratique

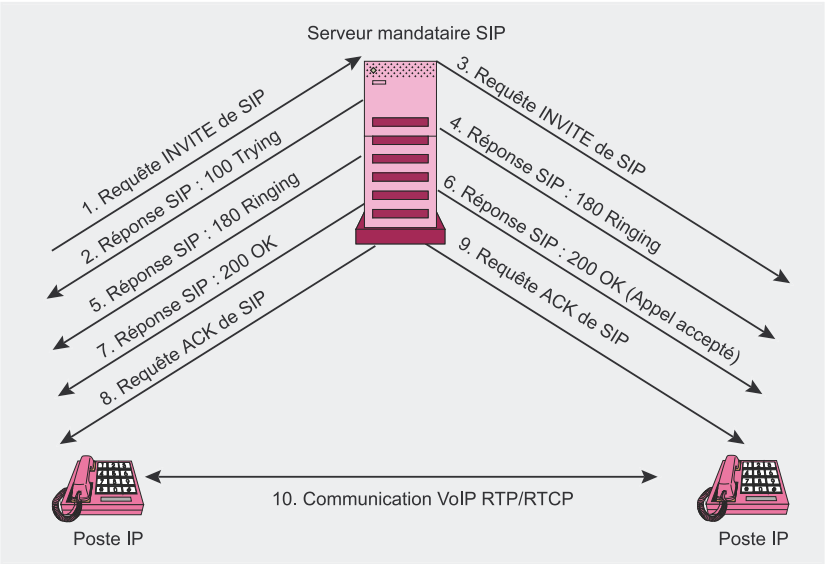



**Figure 1.** *Présentation générale schématique de réglage d'un appel au moyen du protocole SIP*

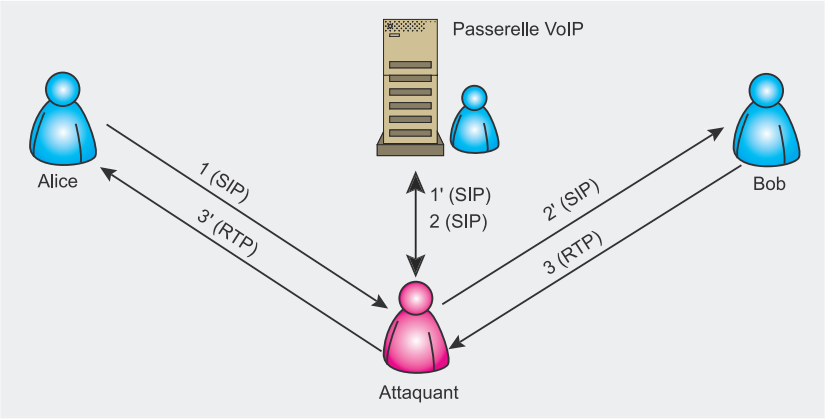



**Figure 2.** *Aspiration d'une transmission VoIP*

capable de réaliser le décodage vocal appelé Cain & Abel (voir l'Encadré *Sur Internet*). Après l'avoir installé et exécuté, vous devriez contrôler l'ensemble des hôtes existants sur votre sous-réseau (au moyen de requêtes ARP) en cliquant sur le symbole *plus*. Ces hôtes sont désormais visibles sous l'onglet intitulé *Sniffer* et peuvent être choisi en tant que victimes potentielles dans le sous-onglet intitulé *ARP*. Dans ce cas, vous sélectionnerez les adresses IP d'Alice, de Bob et du serveur mandataire SIP. Après avoir cliqué sur le bouton *Start/Stop ARP*, l'empoisonnement ARP est initialisé et il ne reste plus qu'une chose à faire pour la personne malveillante : attendre et observer. Cain & Abel se charge du reste (voir la Figure 3). Si un appel a été établi puis achevé entre Alice et Bob, cet appel sera stocké sous forme de fichier WAV de manière automatique puis apparaîtra dans l'onglet intitulé *VoIP*. Vous pourrez alors écouter leur conversation à l'aide de n'importe quel lecteur audio. Et il faut mentionner que, si certains mots de passe ont été échangés entre les deux partenaires de communication (comme par exemple POP3), la personne malveillante pourra les voir apparaître dans l'onglet intitulé *Passwords*.

Il est donc possible d'affirmer sans trop de risque qu'une personne malveillante ne rencontrera aucun problème sur un réseau local pour aspirer une communication et la rediffuser, en l'absence d'installation de mécanismes de sécurité supplémentaire.

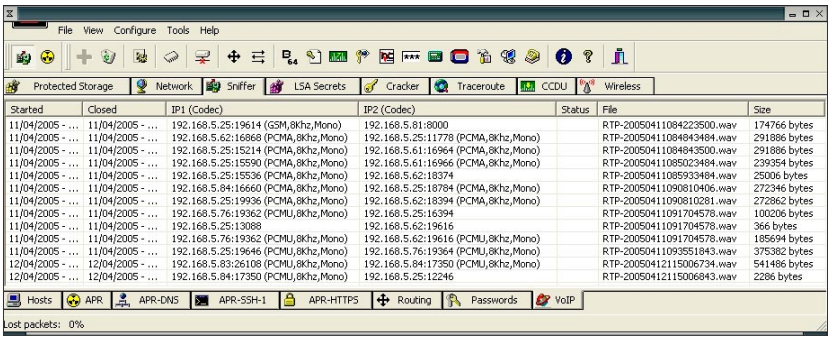



**Figure 3.** *Décodage vocal réalisé par l'outil Cain & Abel*

## Vol d'identité et détournement d'inscription

En règle générale, l'inscription sur un serveur mandataire SIP nécessite un nom d'utilisateur ainsi qu'un mot de passe. Comme il a été mentionné plus haut, l'ensemble des messages SIP ne sont pas cryptés. Si une personne malveillante aspire le processus d'authentification (comme avec l'attaque ARP), elle peut alors utiliser une combinaison de nom d'utilisateur/mot de passe pour être authentifé sur le serveur mandataire SIP.

Toutefois, une telle attaque n'est plus possible avec les dernières implémentations VoIP. Le processus d'authentification (Voir l'Encadré *Mécanismes de sécurité au sein des protocoles VoIP*), comme la manipulation d'autres fonctions pertinentes, ont recours à la fonctionnalité *authentification digest*. Tout d'abord, le client tente de s'authentifier par rapport au serveur mandataire SIP (voir le Listing 1). Le serveur mandataire rejette cette tentative d'authentification en envoyant le code de statut *401 Unauthorized* (voir le Listing 2) puis demande au client de se connecter au moyen de l'authentification digest. À la ligne débutant par la chaîne *WWW-Authenticate*, la valeur *nonce*, qui est censée être choisie de manière aléatoire, est soumise.

Lors de la troisième étape (voir le Listing 3), le client s'authentifie de nouveau. Cette fois, il envoie également le message *Authorization*. Celui-ci contient son nom d'utilisateur, le RMA donné et la valeur nonce envoyée par le serveur. La partie la plus importante est la valeur de réponse. Il s'agit en général d'informations parasites MD5 formées sur les valeurs du nom de l'utilisateur, du mot de passe, d'une valeur nonce envoyée par le serveur, de la méthode HTTP ainsi que de la requête URI. Ce message est traité par le serveur qui élabore des informations parasites MD5, composées des mêmes paramètres contenus dans le serveur. Si les deux types d'informations parasites sont identiques, l'authentification réussit et est reconnue par un message de statut émis par le serveur (voir le Listing 4).

Les informations parasites envoyées dans la troisième phase disposent de deux fonctionnalités qui permettent d'éviter une fausse authentification ou une fausse réutilisation d'une fonction aspirée. Ces informations ne sont valides que pour une valeur nonce choisie aléatoirement et contiennent un nom d'utilisateur et un mot de passe sous forme de valeur parasite. Grâce à ce mécanisme, il est quasiment impossible pour une personne malveillante d'intercepter le mot de passe sur une période de temps réaliste.

## DoS – ou Déni de Service

Comme sur chaque système offrant des services, il est toujours possible de détruire un serveur ou un service si vous disposer d'un débit suffisant. Dans le cas d'un serveur mandataire SIP, il serait possible de lancer une attaque dite *register-storm* capable de surcharger le service. Par ailleurs, il serait également possible d'attaquer le service lui-même au moyen d'attaques dites de déni de service, si ce service est vulnérable. Il est même possible d'obtenir un accès au serveur au moyen d'attaques par surcharge de la mémoire tampon contre ce service. Une telle attaque était possible en 2003 grâce au serveur libre PBX Asterisk (CAN-2003-0761). En raison de la faiblesse de traitement des paramètres dotés des messages `MESSAGE` et `INFO`, une personne aux intentions malveillantes était capable de lancer des commandes locales dans le contexte d'un service *asterisk*, normalement démarré par *root*.

### Mécanismes de sécurité au sein des protocoles VoIP

En plus des mécanismes relatifs à la communication contextuelle, il existe plusieurs autres mécanismes de sécurité sous le protocole SIP, bien qu'ils ne soient pas intégrés aux implémentations. Ces mécanismes de sécurité se rapportent principalement à l'authentification et la sécurité cryptographique de la communication.

Plusieurs possibilités permettent de réaliser l'authentification. Une des méthodes les plus implémentées est l'*authentification digest*. Il s'agit d'une simple méthode de *challenge-response*, qui peut être demandée pour chaque requête.

Il existe une autre méthode censée sécurisée les paquets SIP : l'utilisation du très connu S/MIME. Grâce à S/MIME, le corps d'un message SIP peut être sécurisé au moyen des certificats S/MIME. L'utilisation de S/MIME garantie l'établissement d'une infrastructure à clé publique ainsi que l'implémentation des mécanismes nécessaire à la vérification des certificats. Dans le cas de SIP, S/MIME sécurise généralement les messages SDP. L'utilisation de S/MIME demande beaucoup de temps et d'efforts, en l'absence de structures établies.

D'autres mécanismes de sécurité exigent des éléments de protocole supplémentaires. Il est possible à la fois pour les protocoles SIP et RTP de nommer TLS (serveur de mail sécurisé) comme méthode éventuelle. Dans le cas du protocole SIP, la protection ne s'effectue que par saut de sorte qu'il est impossible de savoir si le partenaire de communication a recours à un téléphone activé sur TLS.

Le service SIP devient impropre dans la mesure où les messages SIP invalides dépendent de l'implémentation. Si le serveur ne dispose d'aucun mécanisme lui permettant de gérer (ou tout simplement ignorer) les messages invalides, il est fort probable que celui-ci ne fonctionne plus. Afin de contrôler son comportement, il existe l'outil PROTOS Test Suite, censée être lancée par tous les propriétaires de PBX (en anglais *Private Branch Exchange*) sur leur boîte (voir à ce sujet l'Encadré *Sur Internet*).

Une autre attaque différente par déni de service est appelée *user-supported* DoS (ou déni de service supporté par l'utilisateur). La Figure 4 expose un message UDP envoyé au téléphone SIP au moyen de la connexion 14 et de l'IP 192.168.5.84 à partir du serveur mandataire SIP 192.168.5.25. En envoyant ce message, le serveur mandataire (ou la personne malveillante) signale à l'utilisateur qu'il vient de recevoir un nouveau message vocal dans sa boîte à messages. Vous pourriez reconnaître cette technique en jetant un coup d'oeil au corps du message `Messages-Waiting: yes` ainsi qu'à celui de `Voice-Message: 1/0`. À la place d'un message vocal, il pourrait également s'agir d'un message par fax par exemple. Le premier chiffre (`1` dans ce cas) indique le nombre de nouveaux messages stockés, et le second (`0` dans ce cas) indique le nombre d'anciens messages stockés.

Comme vous pouvez le constater, ce paquet a été édité. Cette manœuvre est relativement facile à réaliser grâce à l'outil Packetyzer (voir l'Encadré *Sur Internet*) pour Windows, techniquement basé sur Ethereal. Chacun des paquets peut être édité. Certaines sommes de contrôle incorrectes sont également affichées et peuvent être corrigées. Il est donc possible d'envoyer ce message vers des récipients arbitraires ; il suffit de disposer de l'IP et de l'identifiant de connexion de l'utilisateur, qui est en général le même que le numéro de téléphone. Afin de clarifier cette démonstration et de prouver qu'aucune autre information n'est requise, réglez tous les autres

**Listing 1.** *Première phase d'inscription SIP*

```
REGISTER sip:sip.example.com SIP/2.0
Via: SIP/2.0/UDP 10.10.10.1:5060;rport; ←
  branch=z9hG4bKBA66B9816CE44C848BC1DEDF0C52F1FD
From: Tobias Glemser <sip:123456@sip.example.com>;tag=1304509056
To: Tobias Glemser <sip:123456@sip.example.com>
Contact: "Tobias Glemser" <sip:123456@10.10.10.1:5060>
Call-ID: 2FB73E1760144FC0978876D9D69AE254@sip.example.com
CSeq: 20187 REGISTER
Expires: 1800
Max-Forwards: 70
User-Agent: X-Lite
Content-Length: 0
```

**Listing 2.** *Deuxième phase d'inscription SIP – rejet*

```
SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP 10.10.10.1:5060;rport=58949; ←
  branch=z9hG4bKBA66B9816CE44C848BC1DEDF0C52F1FD
From: Tobias Glemser <sip:123456@sip.example.com>;tag=1304509056
To: Tobias Glemser <sip:123456@sip.example.com>; ←
  tag=b11cb9bb270104b49a99a995b8c68544.a415
Call-ID: 2FB73E1760144FC0978876D9D69AE254@sip.example.com
CSeq: 20187 REGISTER
WWW-Authenticate: Digest realm="sip.example.com", ←
  nonce="42b17a71cf370bb10e0e2b42dec314e65fd2c2c0"
Server: sip.example.com ser
Content-Length: 0
```

**Listing 3.** *Troisième phase d'inscription SIP – nouvelle authentification*

```
REGISTER sip:sip.example.com SIP/2.0
Via: SIP/2.0/UDP 10.10.10.1:5060;rport; ←
  branch=z9hG4bK913D93CF77A5425D9822FB1E47DF7792
From: Tobias Glemser <sip:123456@sip.example.com>;tag=1304509056
To: Tobias Glemser <sip:123456@sip.example.com>
Contact: "Tobias Glemser" <sip:123456@10.10.10.1:5060>
Call-ID: 2FB73E1760144FC0978876D9D69AE254@sipgate.de
CSeq: 20188 REGISTER
Expires: 1800
Authorization: Digest username="123456",realm="sip.example.com", ←
  nonce="42b17a71cf370bb10e0e2b42dec314e65fd2c2c0", ←
  response="bef6c7346eb181ad8b46949eba5c16b8",uri="sip:sip.example.com"
Max-Forwards: 70
User-Agent: X-Lite
Content-Length: 0
```

**Listing 4.** *Quatrième phase d'inscription SIP – réussie*

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP 10.10.10.1:5060;rport=58949; ←
  branch=z9hG4bK913D93CF77A5425D9822FB1E47DF7792
From: Tobias Glemser <sip:123456@sip.example.com>;tag=1304509056
To: Tobias Glemser <sip:1888819@sipgate.de>; ←
  tag=b11cb9bb270104b49a99a995b8c68544.017a
Call-ID: 2FB73E1760144FC0978876D9D69AE254@sip.example.com
CSeq: 20188 REGISTER
Contact: <sip:123456@10.10.10.1:5060>;q=0.00;expires=1800
Server: sip.example.com ser
Content-Length: 0
```

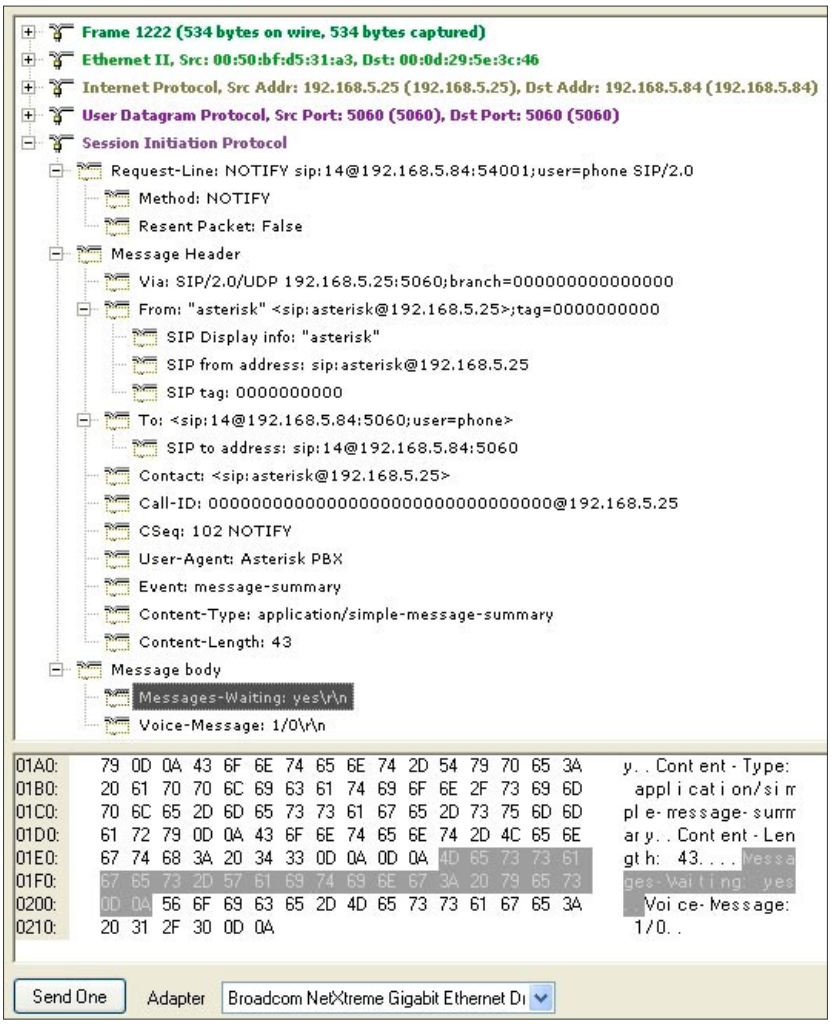



**Figure 4.** *Paquet SIP modifié*

champs sur la valeur `0`. D'autres champs comme `User-Agent` ne sont bien évidemment pas utiles ici.

Et maintenant, comment falsifier un tel message ? Il ne contient aucune donnée sensible. La plupart des téléphones (on a testé un Cisco 9750 ainsi qu'un Grandstream BT-100) traitent ces messages (mêmes ceux contenant de fausses sommes de contrôle) et les montrent aux utilisateurs. En règle générale, une icône ou l'ensemble de l'affichage commencent par clignoter. L'utilisateur appelle désormais sa boîte de message parce qu'il a reçu un nouveau faux message. Comme il n'y a aucun nouveau message stocké pour lui, il va sans doute penser qu'il s'agit d'une erreur et va l'ignorer. Peu de temps après, l'affichage clignote de nouveau. L'utilisateur appelle le support, qui va chercher l'origine de l'erreur, intéressante à voir puisqu'il n'y a en réalité aucune erreur réelle.

Si les personnes malveillantes commencent par envoyer de tels messages à chaque utilisateur du réseau, les utilisateurs et le support passeront tous les deux du temps à chercher la cause de l'erreur. De même, si vous envoyez un tel message à de nombreux utilisateurs, et si chacun appelle sa boîte de messages, un flux de demandes va être engendré voire un plantage de serveur.

## Interruption d'appel

Nombreuses sont les personnes à mettre immédiatement fin à leur appel à la réception d'un simple message `BYE` émis par un rapport de revues. Or ce n'est pas si simple. Tout d'abord, n'oubliez pas que la personne malveillante doit connaître l'identifiant de l'appel du dialogue donné. Selon le RFC 3261 : *le champ en-tête de l'identifiant Call-ID agit en tant qu'unique identifiant afin de rassembler une série de messages. Cet identifiant DOIT être le même pour l'ensemble des requêtes et des réponses envoyées par n'importe quel agent utilisateur dans un dialogue.*

Il n'existe aucune règle rigoureuse devant être observée par l'identifiant basé sur des informations parasites ou devant être comptée dans les lignes supérieures, mais dans la plupart des implémentations, c'est exactement ce comportement qui est observé : des identifiants choisis de manière aléatoire. Par conséquent, la personne malveillante doit aspirer l'initialisation de l'appel afin d'y mettre fin au moyen de l'identifiant. Si cette personne se trouve dans cette situation, le contenu de l'appel pourrait bien être plus intéressant que d'y mettre tout simplement fin.

## Piratage téléphonique

Le piratage téléphonique (en anglais *phreaking*) le plus répandu par le passé est la fraude aux services téléphoniques, qui consiste à envoyer des systèmes de tonalités spéciales dans des cabines téléphonique publiques. Cette fraude est en passe de resurgir. En raison du découplage des données utiles (flux RTP) et de la signalisation (SIP), le scénario suivant semble assez réaliste, mais

### Sur Internet

- *http://www.ee.oulu.fi/research/ouspg/protos/testing/c07/sip/* – PROTOS Test Suite,
- *http://www.ethereal.com* – sniffeur réseau Ethereal,
- *http://www.packetyzer.com* – Packetyzer – sniffeur TCP/IP Windows basé sur Ethereal,
- *http://www.asterisk.org* – serveur libre PBX Asterisk,
- *http://www.oxid.it* – outil Cain & Abel.

## PROGRAMME DE PARTENARIAT
Software-Wydawnictwo

# Inscrivez-vous et commencez à gagner

**Le programme de partenariat de la Maison d'édition Software** est une forme de la collaboration adressée aux propriétaires et aux administrateurs des services Internet.

L'inscription au programme d'affiliation est gratuite et peut vous apporter de grands avantages.
Il suffit que vous placiez le lien (bannière, bouton) à notre boutique Internet sur votre site et vous commencez à gagner de l'argent !

**Vous recevrez 10 % de valeur de chaque achat effectué dans notre boutique via votre site Web.**

www.pp.software.com.pl/fr

ne peut pas encore être réalisé aujourd'hui.

Un client préparé règle un nouvel appel vers une autre client lui aussi prêt. Tous les deux se connectent via un serveur mandataire SIP et se comportent de manière tout à fait normale. Dès que l'appel est établi, le serveur mandataire reçoit un signal lui ordonnant de mettre fin à l'appel que les deux clients peuvent lire. Mais ils n'interrompent pas le RTP en continu et l'appel lui-même continue, le serveur SIP ne remarque pas ce phénomène.

Si les deux clients se trouvent dans le même sous-réseau, l'appel ne se terminera en aucun cas en raison des voix P2P en continu. Si une coupure apparaît sur le serveur mandataire SIP (par exemple, connexion vers un autre réseau), la communication RTP est relancée par le serveur mandataire. Il doit donc mettre fin lui-même à la communication RTP en continu. Il faut donc que le serveur mandataire reconnaisse, au moyen de SIP, qu'une terminaison d'appel a été signalée puis transfère cette information directement au contrôle de la communication RTP.

Une autre attaque éventuelle de piratage téléphonique dépend de l'implémentation du serveur mandataire SIP. En effet, certaines implémentations, comme la dernière sortie d'Asterisk, exigent une réauthentification via l'authentification digest comme vous l'avez constaté dans les Listings 1 à 4 pour la grande majorité des communications client-serveur. D'autres implémentations nécessitent une réauthentification uniquement après un certain temps. Si tel est le cas, il est possible, grâce au scénario suivant, de générer des charges pour le fournisseur.

Une personne malveillante envoie un message `INVITE` valide au serveur mandataire SIP, au moyen des informations qu'il a obtenues d'un utilisateur authentifié avec succès. Le serveur mandataire SIP va désormais initialiser l'appel. Les paquets restants nécessaires à réussir l'initialisation de l'appel pourraient être envoyés par la personne malveillante (ce qui nécessite du temps sans connaître les paquets réponses issus du serveur). Certains modèles commerciaux de numéros de services spéciaux prévoient des quantités astronomiques de paquets pour un appel, indépendamment de la durée de l'appel. Compte tenu de cette situation, une personne malveillante pourrait réaliser des taux importants grâce à des appels très courts, à la charge des autres utilisateurs.

## SPIT (Spam over IP Telephone)

L'attaque appelée SPIT est l'un des dangers les plus fréquemment mentionnés en cas d'installation de la technologie VoIP. Cette attaque consiste à envoyer des messages vocaux de la même façon que les publicités e-mail non désirées ou spam. Contrairement aux appels automatisés dans le monde des services téléphoniques ordinaires, les appels VoIP ne génèrent pas de coûts dès le début. A l'instar des spams traditionnels bien connus, le *spitter* utilise l'adresse d'une victime, dans ce cas, il ne s'agit pas de son adresse e-mail, mais de son adresse SIP. Avec le succès que rencontre la téléphonie IP, ce n'est qu'une question de temps pour obtenir facilement de nombreuses adresses SIP valides, surtout si les livres d'adresses centrales vont être réellement utilisés.

Le spitter appelle un numéro SIP trouvé, le serveur mandataire SIP de la victime va traiter cet appel et la victime elle-même n'a d'autre choix que d'écouter le message spam élaborée par le spitter sur par exemple la virilité de quelqu'un. Tout comme les spammers, le spitter n'a besoin que d'un seul élément, le débit. En effet, les messages vocaux consomment bien plus de ressources que les e-mails. Un message de 15 secondes par exemple (aucune victime ne supporterait un message plus long) équivaut à 120 Ko au moyen d'un codec de 64 Kbps. En ayant recours à des chevaux de Troie, encore une fois comme les spams, un utilisateur Internet non protégé pourrait être induit à envoyer une attaque SPIT au travers de sa bande passante.


### À propos des auteurs

Les auteurs sont tous deux consultants en informatique. Tobias Glemser travaille pour le compte de la société Tele-Consulting GmbH (*http://www.tele-consulting.com*), en Allemagne depuis plus de quatre ans, alors que Reto Lorenz occupe le poste de directeur commercial de la même boîte.


## Composeurs

L'utilisation des composeurs, déclarés hors service depuis le succès des autres technologies comme le DSL ou les câbles, est en passe de devenir une nouvelle attaque. Avec le recours à un client SIP, nous disposons du même scénario que celui qui consiste à composer un numéro au moyen d'un modem ou de lignes ISDN pour appeler des numéros surtaxés. Par exemple, un composeur pourrait infecter un client SIP et installer un certain nombre de préfixes ou entrer un nouveau serveur mandataire SIP très onéreux. Les appels seraient générés via ces numéros surtaxés sans que l'utilisateur n'ait aucun moyen de le savoir jusqu'à l'arrivée de la première facture.

À l'heure actuelle, de tels composeurs n'ont pas encore été remarqués, mais ce n'est qu'une question de temps avant d'entendre parler de la première attaque de ce genre réussie (ou échouée ?).

## Conclusion

Il est évident que la technologie VoIP est l'une des techniques de ces dernières années parmi les plus prometteuse et est en passe de devenir une nouvelle application phare d'Internet, destinée aux réseaux des sociétés mais aussi aux réseaux privés. Suite à la récente attention des médias sur les problèmes de sécurité de VoIP, vous n'aurez pas tort de penser que la combinaison des protocoles SIP et RTP est d'une incroyable faiblesse. Bien sûr, il est indispensable d'envisager les problèmes de sécurité avant d'adopter une nouvelle technologie. ■