AETA
AUDIO SYSTEMS

Codecs AoIP et sécurité des réseaux

Sommaire

1. Introduction

Comme tous les appareils qui font usage des réseaux IP, en particulier via Internet, les codecs AoIP soulèvent des questions et des problèmes vis-à-vis de la sécurité réseau. On peut rapidement distinguer deux types de préoccupations :

- L'utilisation des codecs audio peut susciter des failles dans la sécurité des systèmes informatiques, voire des attaques. Ceci est en général la préoccupation principale des gestionnaires des systèmes.
- Les procédures et moyens de protection des systèmes peuvent entraver voire bloquer l'utilisation des codecs. Cette fois c'est l'utilisateur, ou du moins celui qui met en place les moyens de transmission, qui est le premier concerné par le problème.

Bien sûr les deux types de problèmes peuvent être rencontrés simultanément… Comme pour toute utilisation d'Internet, la solution doit normalement présenter un bon compromis entre sécurité et souplesse d'exploitation.

Sans prétention d'exhaustivité, nous allons examiner ici quelques types de moyens et solutions applicables, et discuter de leur complexité et efficacité relatives.

Pré-requis : le lecteur doit posséder des connaissances de base sur les réseaux IP, les fonctions des routeurs et des pare-feu.

2. Particularités des codecs audio

2.1. Utilisation de UDP et RTP

La transmission des flux audio utilise normalement RTP/UDP. A part des implémentations très spécifiques et propriétaires, cela est le cas tant pour des liaisons directes sans signalisation que pour des sessions établies grâce au protocole SIP.

UDP est préférable à TCP pour les flux media en temps réel, mais amène certaines contraintes lorsqu'il y a de la traduction d'adresse (NAT).

Exemple concret : lors d'une phase de signalisation un codec annonce au terminal distant qu'il peut recevoir un flux RTP/UDP à son adresse IP et sur un port *n*. En fait, à cause de la présence d'un routeur avec traduction d'adresse, ces données ne sont valables que sur le réseau local du codec. Depuis le codec distant, l'adresse IP est invalide ou injoignable, et le numéro de port est incorrect. Le flux ne pourra pas parvenir au codec.

Un des moyens de contourner cet obstacle est l'utilisation de STUN (cf. + loin).

2.2. SIP

Le protocole SIP est souvent utilisé par les codecs audio, et il est préconisé dans la recommandation « N/ACIP » (UER Tech 3326).

SIP (Session Initiation Protocol) permet la signalisation d'une session media via un réseau IP entre deux « agents », tels que des téléphones VoIP, des codecs, etc. La signalisation correspond à l'établissement et la libération de la session, avec négociation (au moyen du protocole SDP) des paramètres nécessaires : codage, adresses IP et ports.

Dans les grandes lignes, le déroulement d'une session est le suivant :

- L'agent initiateur de la session envoie une demande (message INVITE) à « l'agent » distant.
- Il propose une (des) configuration(s) de session (protocole SDP), comportant codage, numéros de port pour la session RTP, etc.
- L'agent distant accepte en indiquant la configuration retenue.
- Les agents échangent leurs flux audio avec les paramètres qui viennent d'être négociés.
- Pour terminer, un des agents libère la session (message BYE).

Ceci est valable pour une liaison « pair à pair » entre les agents, et le protocole SIP n'impose pas la présence de serveurs. Le protocole SIP utilise normalement UDP.

Un système plus évolué inclura un serveur SIP qui s'interpose dans la phase de signalisation. Dans ce cas les agents s'enregistrent préalablement sur le serveur avant l'établissement de sessions. Un tel système apporte plusieurs avantages :

- Insensibilité aux modifications d'adresse IP des codecs, d'où une utilisation aussi simple que pour un système téléphonique usuel.
- Fonctions évoluées du serveur, au-delà du simple proxy/registrar.
- Meilleur contrôle des accès/autorisations des agents.
- Couplages possibles avec l'action des pare-feu.

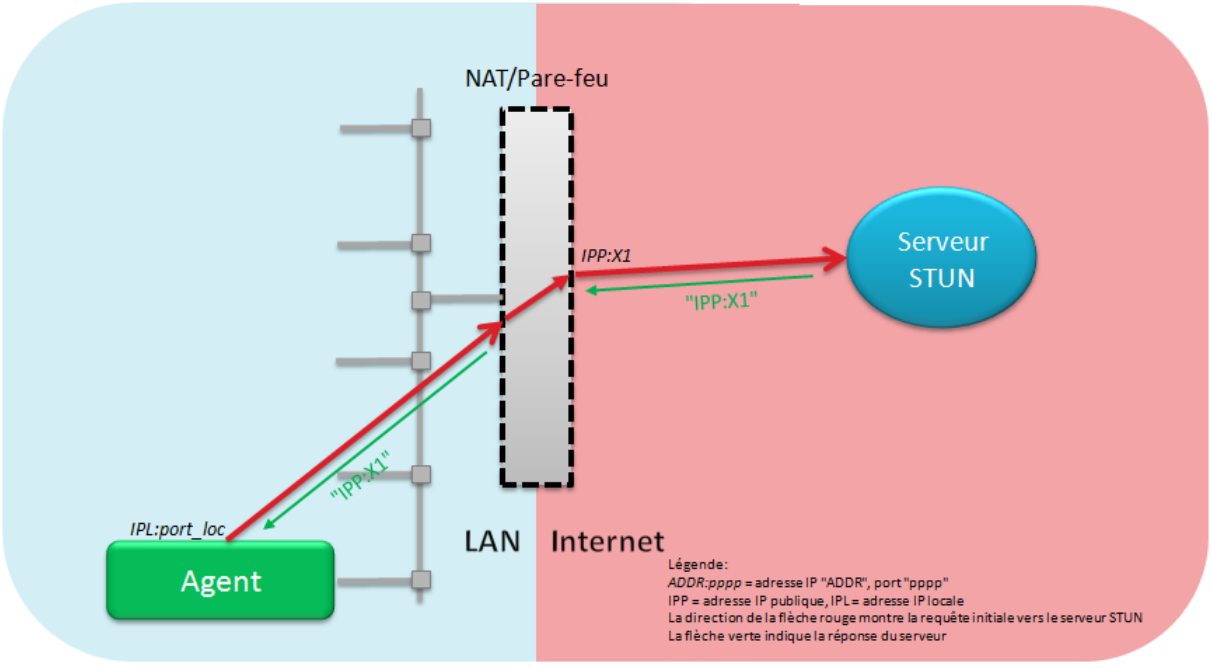
⇨ *Les codecs AETA utilisent toujours le protocole SIP, avec ou sans implication d'un serveur SIP.*

⇨ *Par ailleurs, AETA propose à ses clients les services d'un proxy/registrar SIP dédié aux applications broadcast audio sur IP (plus d'info : voir ici).*

2.3. STUN

Le protocole STUN est un moyen de traverser les routeurs avec traduction d'adresse (NAT). Il permet aux agents de découvrir leur adresse IP publique effective du côté Internet du routeur, ainsi que le numéro de port réel pour le flux RTP.

Principe de base : l'agent interroge un serveur STUN, situé en zone publique, qui lui renvoie les informations correctes sur son adresse IP publique et les numéros de ports effectivement alloués par le routeur NAT. L'agent peut ensuite utiliser ces données pour la signalisation d'une session SIP.



⇨ *Les codecs AETA ont tous la capacité d'utiliser STUN. Cette fonction peut être cependant désactivée facilement en cas de besoin.*

2.4. Autres protocoles

Les codecs peuvent aussi utiliser d'autres protocoles pour des besoins distincts de la fonction essentielle de transmission audio.

- Télécommande, d'ordinaire sous TCP/IP. Exemple : serveur html embarqué.
- Monitoring, avec TCP/IP ou UDP/IP.

Ces fonctions étant annexes, ces protocoles et les ports associés sont en général optionnels. En conséquence l'ouverture des ports nécessaires sur des pare-feu n'est pas une obligation.

3. Exemples de vulnérabilité

Nous décrivons ici des exemples d'actions malveillantes qui peuvent prendre pour cibles ou vecteurs des codecs audio.

3.1. Attaque directe sur un codec

Un hacker peut attaquer directement le codec, via son interface de gestion. L'objectif peut être par exemple :

- Prise de contrôle indésirable, ouvrant la voie à d'autres actions malveillantes.
- Tentative de planter voire endommager le codec.

3.2. Connexions indésirables

Objectif : occuper un codec pour empêcher son utilisation légitime au bon moment, ou au pire pour perturber le programme à l'antenne. Parfois cela est possible sans même une opération illicite ou du « hacking », il suffit d'appeler un codec. En fait, ce type de nuisance peut être aussi simple à effectuer qu'un appel téléphonique intempestif… D'ailleurs même une erreur de manipulation peut créer un problème similaire par un appel intempestif.

De façon plus automatique, l'attaque peut consister en une multiplicité d'appels automatiques pour surcharger le système ou le bloquer ; cela s'apparente alors à une attaque de type DoS.

3.3. Intrusion dans le serveur

L'attaque peut porter sur le serveur en essayant de s'enregistrer afin d'utiliser illicitement (et gratuitement) ses services, ou d'établir des liaisons intempestives (ce qui ramène au cas précédent). Les hackers utilisent certains outils pour détecter des mots de passe ou forcer des enregistrements.

Le hacker peut aussi tenter une intrusion dans le serveur lui-même, pour en prendre contrôle, etc.

3.4. Intrusion dans le réseau local via le codec

Pénétrer dans le codec (interface de gestion) est un moyen éventuel de faire intrusion sur le réseau local voire sur d'autres machines, au même titre que l'intrusion dans un PC du réseau (via un cheval de Troie par exemple).

Cependant, un codec audio n'est pas une cible aussi classique qu'un PC sous Windows par exemple, lorsqu'il n'est pas un PC bien sûr… Mais des attaques similaires sur des téléphones VoIP ont été déjà constatées.

3.5. Espionnage des liaisons

Les codecs, comme des téléphones IP, peuvent être la cible d'espionnage dans divers buts :

- Assez classiquement, pour écouter le contenu des conversations. Pour un codec à usage broadcast, cependant, ce n'est pas nécessairement un problème majeur.
- Pour perturber une liaison, ou usurper une identité.
- Pour obtenir des informations dans les données échangées permettant une intrusion dans le serveur, ou d'autres actions malveillantes.

3.6. Un exemple concret : SIPVicious

SIPVicious est un exemple assez fréquemment impliqué dans des interventions malveillantes sur des agents VoIP, qui peuvent relever des divers cas évoqués ci-dessus. En fait, à l'origine il s'agit d'un ensemble d'outils Open Source, conçus pour le test et l'audit de systèmes VoIP à base SIP. Ces outils ont été détournés de leur destination par des hackers qui l'utilisent pour :

- Détecter/dénombrer des serveurs ou téléphones SIP (scanner)
- Chercher à casser les mots de passe contrôlant les accès aux comptes SIP
- Tenter d'établir des liaisons pirates

Dans un scénario assez fréquent, un hacker détecte la présence d'un codec SIP sur une certaine adresse IP. Supposant probablement qu'il s'agit d'un serveur SIP, il tente ensuite de placer des appels VoIP via l'appareil vers des numéros de téléphone. En fait, ceci ne peut pas fonctionner, car c'est un serveur ou proxy SIP qui serait susceptible de traiter ces appels. Dans la pratique, le codec « sonne », reçoit un appel indésirable d'un correspondant inconnu ou non identifié, décroche la ligne. Mais il ne synchronise pas, et abandonne la ligne après quelques secondes.

Il arrive que ces appels « fantômes » se suivent à un rythme élevé, susceptible d'aller jusqu'à provoquer un plantage et/ou un redémarrage du codec.

Si vous souhaitez approfondir le sujet, voir le site http://blog.sipvicious.org .

4. Problèmes induits par les routeurs et pare-feu

4.1. Généralités

Si l'on essaie de décrire simplement sa fonction, un pare-feu doit empêcher, à la frontière entre le réseau local et l'accès Internet, le passage de données non désirées donc susceptibles de créer des problèmes. La fonction de pare-feu est souvent intégrée dans le routeur d'accès Internet.

Il ne faut pas perdre de vue que, par son principe même, un pare-feu, dont l'objet est de protéger d'actions malveillantes, peut aussi gêner l'utilisation légitime, s'il est configuré de manière erronée ou inadaptée, ou encore par excès de zèle.

Un autre point important à garder à l'esprit : au-delà du domaine réseau sous la responsabilité de l'utilisateur des codecs, le domaine réservé de l'opérateur peut aussi comporter des filtrages et blocages éventuels. Ceci est tout particulièrement le cas avec des réseaux tels que les réseaux de données mobiles.

4.2. Blocage de protocole

Un pare-feu peut éventuellement appliquer des blocages de données en fonction du protocole utilisé dans les paquets reçus ou émis, par exemple :

- Blocage de UDP en général
- Blocage de la signalisation VoIP : protocole SIP. De façon similaire, le blocage peut concerner le protocole STUN.
- Blocage des flux RTP. Par exemple, certains réseaux mobiles agissent directement à ce niveau pour empêcher l'utilisation de VoIP sur le réseau, sans nécessairement intervenir dans les phases de signalisation SIP.

4.3. Blocage de ports

Au lieu de détecter et bloquer un protocole, certaines règles visent certains numéros de ports spécifiques. Le principe est en général de bloquer tout trafic à l'exclusion des ports spécifiquement autorisés que l'on va « ouvrir » au trafic.

A l'inverse, un blocage peut viser des ports spécifiques pour bloquer un type de trafic donné. Cela se rencontre parfois dans le réseau opérateur, par exemple pour lui réserver l'accès à un type de service donné, ou encore au niveau du routeur d'accès Internet fourni et géré par l'opérateur. Les ports standard SIP et RTP sont parfois la cible de ces blocages.

Cette méthode n'est pas toujours pertinente car elle suppose le respect d'une correspondance fixe entre type de service et numéro de port. Or, par exemple, dans une session SIP les ports RTP sont susceptibles de négociation, donc leurs numéros sont dynamiques. Ouvrir de manière statique des ports ne va pas toujours fonctionner…

4.4. Routeurs NAT

Nous parlons ici de routeurs à traduction d'adresse réseau (Network Address Translation). Leur fonction première est de partager un petit nombre d'adresses IP publiques (souvent une seule) entre un nombre assez élevé d'hôtes locaux sur un réseau privé. La fonction de NAT est de ce fait quasiment toujours incluse dans un routeur d'accès Internet.

Le cas le plus fréquent est la traduction d'adresse avec traduction de port (PAT), et ce sera l'hypothèse par défaut dans le contexte de ce document[1] lorsque l'on parlera de NAT.

Une description plus détaillée du NAT peut être trouvée par exemple sur la page http://fr.wikipedia.org/wiki/Network_address_translation .

Il se trouve que les routeurs NAT apportent aussi une protection de base contre les agressions extérieures, car l'entrée de données non sollicitées n'est pas possible *a priori*. En principe un « chemin » de retour est ouvert à travers le routeur lors d'une sollicitation vers l'extérieur et les paquets de réponse emprunteront ce chemin pour parvenir à l'élément initiateur de dialogue. Le chemin est refermé après un certain temps d'inactivité de la liaison temporaire ainsi ouverte. Par conséquent il n'est pas possible pour une source externe de prendre l'initiative pour envoyer des données à un élément du réseau local.

Avec UDP et SIP, traverser un routeur NAT pose des problèmes spécifiques :

- Initier une liaison depuis l'extérieur n'est pas possible, en tout cas dans un contexte de session « pair à pair ».
- Lors de la négociation SIP, les agents indiquent l'adresse IP et le port sur lesquels ils peuvent recevoir le flux audio. Or l'agent derrière NAT (ou « NATté ») ne connaît et n'indiquera que son adresse IP et son port RTP locaux, alors que ce sont leurs pendants publics (vus du côté Internet du routeur) que le codec distant devrait connaître. Le flux ne pourra pas atteindre sa destination.

Heureusement il existe des moyens de contourner ce problème d'adressage, comme par exemple le protocole STUN, évoqué en 2.3 ci-dessus.

[1] Mais la problématique et les solutions mises en œuvre sont similaires dans les autres cas.

5. Méthodes d'accès

Nous allons ici décrire certaines méthodes utilisables pour mettre en place des codecs sur un site et permettre d'établir des liaisons audio entre un codec sur le site et un autre codec sur un site distant. Nous supposons ce dernier joignable via Internet, car une liaison par réseau privé distant est équivalente à un réseau local, et ne pose pas de problème de sécurité spécifique.

5.1. Raccordement direct à Internet

5.1.1. Principe

Il s'agit ici de la méthode la plus directe : placer le codec directement sur Internet, donc avec une adresse IP publique qui lui est allouée en propre. L'utilisation est très simple puisque proche de celle sur un réseau local : le codec peut directement créer une liaison vers l'adresse IP publique d'un autre codec, ou être joint par son adresse IP publique.

5.1.2. Avantages

- Grande simplicité de mise en œuvre. La configuration du routeur d'accès peut être un peu délicate mais c'est une opération unique.
- Pas de problèmes de connectivité, c'est-à-dire pas de blocages à craindre, puisqu'il n'y a pas de protection…
- Le codec étant isolé d'un réseau local, à première vue il ne propagera pas à d'autres équipements des problèmes de sécurité éventuels.

5.1.3. Inconvénients

- On ne peut pas imaginer plus dangereux pour la sécurité d'exploitation du codec : il ne dispose tout simplement **d'aucune protection** contre une agression ou utilisation malveillante !
  Le codec est sujet aux tentatives d'intrusion ou prise de contrôle, aux connexions intempestives, aux espionnages, etc.
- L'adresse IP publique est imposée par le réseau de l'opérateur ; sur certains accès ADSL elle est dynamique (pas forcément fixe dans le temps). Cela complique l'établissement de liaisons depuis un appareil distant.
- Le codec est hors d'un réseau local, ce qui peut limiter ses communications avec des équipements sur ce réseau.

5.1.4. Recommandations

Une seule et simple recommandation : ne pas adopter cette technique !

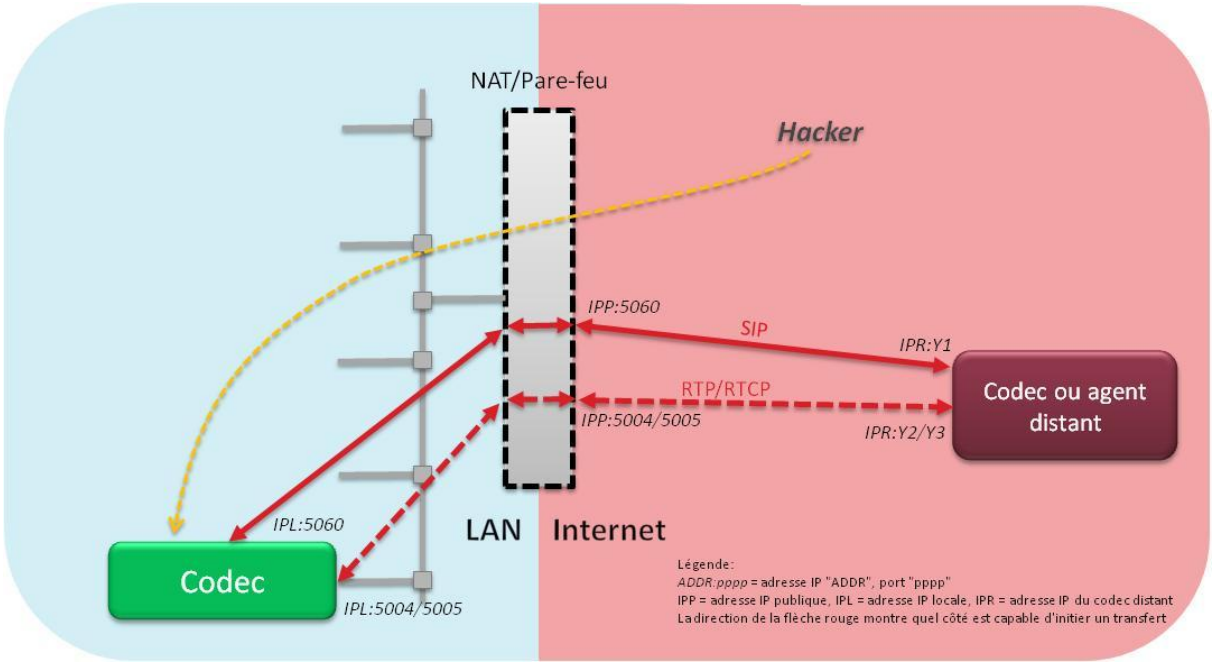
Il est bien connu qu'un appareil lâché ainsi en pâture sur Internet est très rapidement détecté et attaqué, c'est une question de minutes… Il y a peu de chances de pouvoir exploiter durablement un appareil dans de telles conditions.

5.2. NAT+DMZ

On peut considérer la mise en DMZ comme une variante de la méthode précédente.

5.2.1. Principe

Le codec est, de façon plus classique, installé derrière un routeur d'accès avec NAT, mais placé en « DMZ » : il est directement en réception de tous les paquets arrivant sur l'adresse publique[1] de l'accès Internet, avec éventuellement quelques exceptions (par exemple certains ports peuvent tout de même rester réservés au routeur ou à un autre équipement du réseau). Cependant il est sur le réseau local et son adresse IP est à portée locale, qu'elle soit attribuée statiquement ou par DHCP.



Le codec local ne connaissant pas *a priori* l'adresse publique, il va se poser le problème lié au NAT, évoqué plus haut en 4.4. Il faut utiliser un serveur STUN (cf. 2.3) pour qu'il découvre l'adresse IP publique (*serveur non représenté ci-dessus pour simplifier le schéma*).

5.2.2. Avantages

- Relative simplicité de mise en œuvre, presque aussi simple que pour le cas précédent (raccordement Internet direct).
- Pas de problèmes de connectivité, à condition d'utiliser STUN. Le codec peut directement créer une liaison vers l'adresse IP publique d'un autre codec, ou être joint par son adresse IP publique.
- En général le codec peut aussi communiquer avec d'autres codecs situés dans le même réseau local.

[1] Ou l'une des adresses publiques, si l'accès en comporte plusieurs.

5.2.3. Inconvénients

- Pour la sécurité d'exploitation du codec, ce système est aussi dangereux que le précédent ; il ne dispose **d'aucune protection** contre une agression ou utilisation malveillante !
  Le codec est sujet aux tentatives d'intrusion ou prise de contrôle, aux connexions intempestives, aux espionnages, etc.
- Le codec n'est pas forcément isolé du réseau local ; s'il est attaqué il peut au pire servir de cheval de Troie pour propager des attaques.
- Certains routeurs grand public peuvent ne pas permettre ce type de configuration. En tout cas la configuration n'est pas toujours simple.
- L'adresse IP publique est imposée par le réseau de l'opérateur ; sur certains accès ADSL elle est dynamique (pas forcément fixe dans le temps). Cela complique l'établissement de liaisons depuis un appareil distant.

5.2.4. Recommandations et variantes

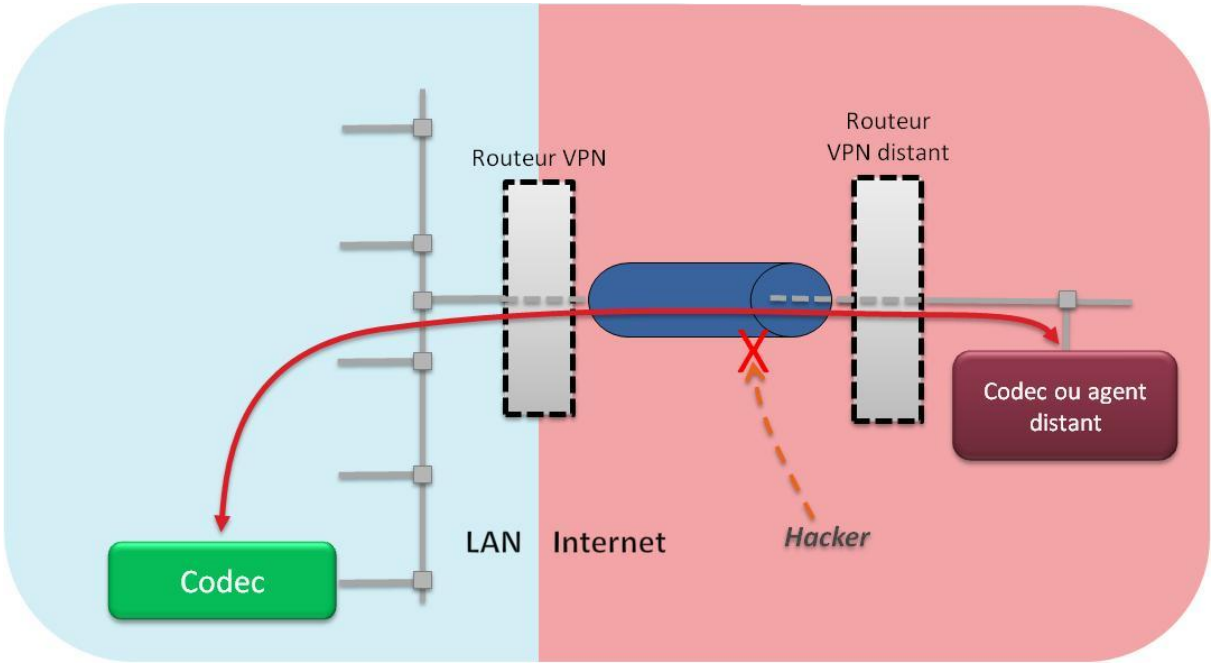
Cette configuration est à déconseiller pour les mêmes raisons que la précédente : absence de protection contre des attaques extérieures.

Une variante consiste à isoler le codec en DMZ du reste du réseau local, par exemple avec un deuxième niveau de pare-feu entre ce dernier et le codec. Plus lourde, cette architecture protège bien le réseau local mais ne change rien à la vulnérabilité du codec.

5.3. *Liaison via VPN*

5.3.1. Principe

De nombreux utilisateurs se tournent vers un système presque diamétralement opposé au précédent : le codec est sur réseau local, « caché » derrière un routeur VPN. Ce dernier va donc, comme son nom l'indique (Virtual Private Network), créer une liaison privée virtuelle avec un autre réseau local, lui aussi équipé d'un routeur VPN. L'autre réseau local est alors une extension du réseau IP privé. On utilise l'image et le terme de « tunnel » : la liaison *réelle* entre les routeurs est sécurisée et cryptée, elle abrite comme un tuyau hermétique et opaque les données échangées entre les sites.



5.3.2. Avantages

- Sécurité maximale. Une très bonne protection peut être assurée, il est extrêmement difficile de pénétrer ou espionner les données.
- L'appareil est protégé de sollicitations extérieures à l'organisation.
- L'exploitation est aussi simple que sur réseau local, puisqu'en effet elle reste inscrite dans un réseau privé.
- Une même configuration convient a priori à tous les types de codecs, quelque soient leurs protocoles, réglages et particularités.

5.3.3. Inconvénients

- Installation initiale plutôt délicate, voire complexe. En général elle est du ressort du responsable du réseau informatique, et peu envisageable sans planification préalable.
- Seuls les sites distants intégrés au réseau privé sont connectables. Il n'est pas possible d'établir une liaison à l'improviste avec un appareil sur un autre site distant[1].
- La VPN peut provoquer une augmentation du trafic extérieur, par rapport au trafic net dû aux codecs, et aussi une augmentation de la latence. Ces impacts dépendent du type de protocole utilisé.
- Si la liaison réseau est de mauvaise qualité, il arrive que la VPN se désynchronise. Il faut un certain temps pour la restaurer, qui peut se compter en secondes. Là encore, cela dépend du protocole et/ou de l'équipement VPN.
  Ce type de problème est pertinent pour une liaison via réseau mobile, qui ne sera jamais à l'abri d'interruptions occasionnelles. La VPN peut aggraver cela, dans le pire des cas, en transformant une brève coupure en une interruption de plusieurs secondes.

5.3.4. Recommandations

Cette solution est très intéressante notamment pour sécuriser des liaisons régulières entre sites bien définis et stables, et si l'on n'a pas besoin d'ouverture sur l'extérieur de l'organisation.

Si au moins un des côtés de la liaison risque des erreurs de transmission fréquentes, il faut bien évaluer la robustesse de la VPN face à ces erreurs, pour éviter qu'elles provoquent des coupures trop audibles. Attention aussi à vérifier l'impact sur la latence, selon les exigences de l'exploitation.

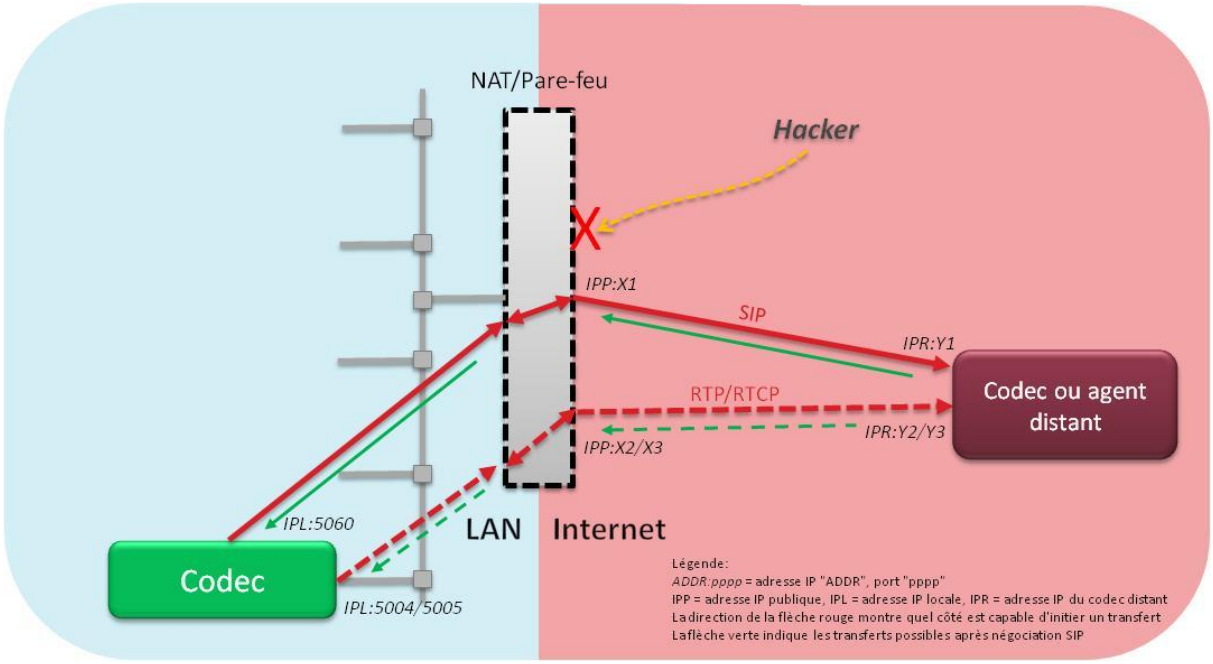
[1] Remarque : bien que cela soit paradoxal, l'utilisation d'un routeur avec accès via réseau mobile n'est pas ici considérée comme une liaison à l'improviste, puisqu'il suffit d'activer/réactiver une liaison déjà préparée (configuration des équipements et réseau mobile inchangés).

5.4. *NAT « classique »*

5.4.1. Principe

Le codec est installé, de manière classique, sur réseau local derrière un routeur d'accès avec NAT. Souvent, ce dernier remplit aussi la fonction de pare-feu pour protéger des attaques extérieures.

Le codec local ne connaissant pas *a priori* l'adresse publique, il va se poser le problème lié au NAT, évoqué plus haut en 4.4. Il faut utiliser un serveur STUN (cf. 2.3) pour qu'il découvre l'adresse IP publique (*serveur non représenté pour simplifier le schéma*).



Le schéma ci-dessus est valable pour un codec mettant en œuvre le protocole SIP. Les paquets qui doivent passer à travers le routeur d'accès sont de deux types :

- Signalisation au protocole SIP/UDP
- Flux audio codé au protocole RTP/UDP

Lorsque le codec émet un paquet vers un codec distant, ou un serveur STUN, une route (et un port correspondant) est ouverte à travers le routeur NAT. Les paquets en retour peuvent emprunter ce chemin pour atteindre le codec. C'est grâce à l'utilisation de STUN que le codec peut indiquer à son homologue les adresses et ports corrects, c'est-à-dire vus du côté accès public du routeur.

En résumé, le codec peut initier une liaison et dispose des données pour l'établir. En revanche, une session ne peut pas être déclenchée sur initiative extérieure.

5.4.2. Avantages

- La configuration est simple et ne nécessite même pas de configuration particulière du routeur NAT. Il faut cependant veiller à utiliser STUN (ou méthode similaire) dans le codec.
- Il n'est pas nécessaire d'installer ou de disposer d'un serveur d'infrastructure, tel qu'un serveur SIP. Il faut tout de même utiliser un serveur STUN, du côté public par principe même. Mais ce type de serveur est peu critique et de nombreux serveurs sont disponibles gratuitement, par exemple celui proposé par AETA (stun.aeta.com).
- Le codec peut aussi communiquer avec d'autres codecs situés dans le réseau local.
- Le système est assez bien protégé des agressions extérieures, car l'extérieur ne peut pas initier un échange simplement[1].

5.4.3. Inconvénients

- Le codec ne peut pas être contacté de l'initiative d'un codec distant. Ceci peut être un inconvénient majeur ou non, selon le mode d'exploitation souhaité.
- En particulier, il est impossible d'établir des liaisons si le codec distant est dans une disposition identique.
- Dans le cas particulier des routeurs avec NAT « symétrique », ce système ne fonctionne pas, car les ports découverts au moyen de STUN ne sont pas valables pour le transfert ultérieur avec le codec distant.

[1] Il est quand même possible de détecter les ports publics (à allocation dynamique) lors des transactions légitimes et de faire intrusion par ces voies.

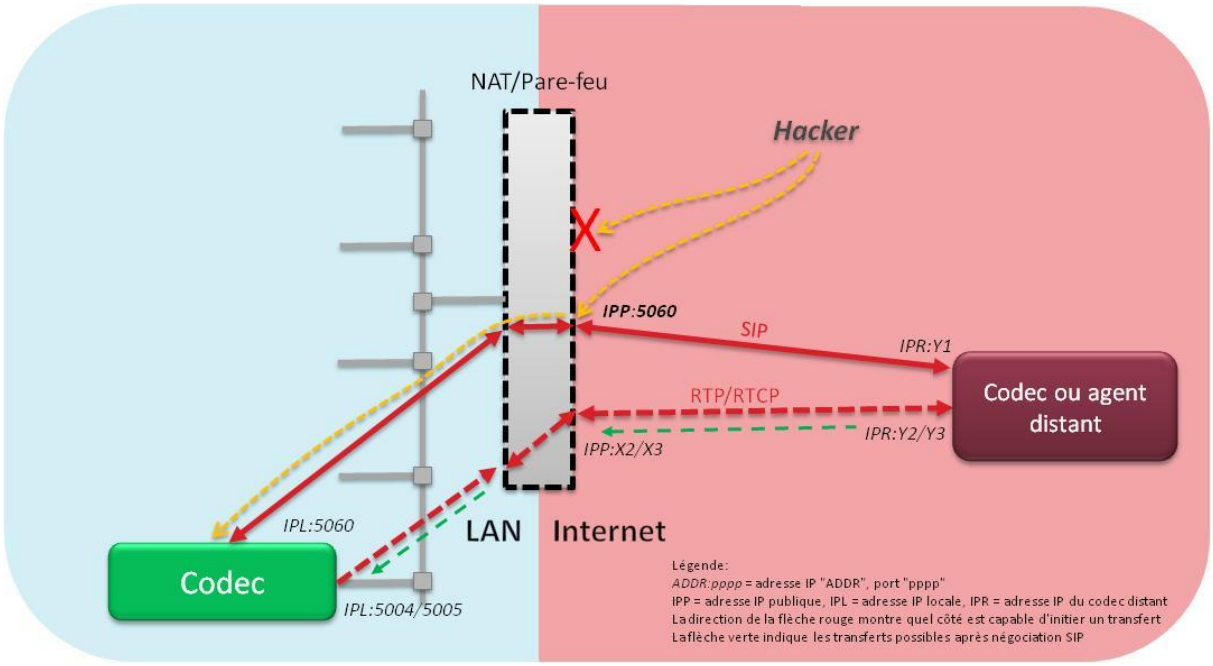
5.5. NAT et redirection de ports

5.5.1. Principe

Comme dans la situation précédente (NAT « classique »), le codec est installé sur réseau local derrière un routeur d'accès avec NAT. Souvent, ce dernier remplit aussi la fonction de pare-feu pour protéger des attaques extérieures.

Pour donner à un codec distant la possibilité d'initier une liaison, on ajoute dans le routeur/pare-feu une *redirection de port* (en Anglais « port forwarding ») : le port SIP/UDP 5060 est redirigé vers le codec.

Comme dans le cas précédent et pour les mêmes raisons, il faut utiliser un serveur STUN pour que le codec découvre l'adresse IP et les numéros de ports publics (*Serveur STUN non représenté ci-dessous pour simplifier le diagramme)*.



Le schéma ci-dessus est valable pour un codec mettant en œuvre le protocole SIP. Les paquets qui doivent passer à travers le routeur d'accès sont de deux types :

- Signalisation au protocole SIP/UDP
- Flux audio codé au protocole RTP/UDP

Lorsque le codec émet un paquet vers un codec distant, ou un serveur STUN, une route (et un port correspondant) est ouverte à travers le routeur NAT. Les paquets en retour peuvent emprunter ce chemin pour atteindre le codec. C'est grâce à l'utilisation de STUN que le codec peut indiquer à son homologue les adresses et ports corrects, c'est-à-dire vus du côté accès public du routeur.

Lorsque c'est le codec distant qui initie l'appel, il bénéficie alors de l'allocation statique du port SIP au codec ; ensuite par négociation SIP/SDP les codecs s'échangent les données pour l'établissement des flux RTP, ports à utiliser inclus.

5.5.2. Avantages

- Contrairement au cas précédent, cette fois il est possible pour un codec distant de contacter le codec et établir une liaison.
- En comparaison avec les solutions les plus basiques (voir plus haut *Raccordement direct à Internet* et *NAT+DMZ*), on n'expose pas le codec au-delà du strict nécessaire.
- Il n'est pas nécessaire d'installer ou de disposer d'un serveur d'infrastructure, tel qu'un serveur SIP. Il faut tout de même utiliser un serveur STUN, du côté public par principe même. Mais ce type de serveur est peu critique et de nombreux serveurs sont disponibles gratuitement, par exemple celui proposé par AETA (stun.aeta.com).
- Le codec peut aussi communiquer avec d'autres codecs situés dans le réseau local.

5.5.3. Inconvénients

- Sans être très complexe, la configuration du routeur est légèrement moins simple que dans le cas précédent.
- Elle n'est parfois pas possible, soit sur des routeurs très basiques, soit faute d'accès à leur configuration. Un cas flagrant et incontournable est l'accès via réseau mobile : l'opérateur seul a le contrôle sur le routeur NAT d'accès au réseau.
  La méthode est donc quasiment inutilisable en accès sur réseau mobile.
- Il n'est pas possible avec ce système de disposer plus d'un codec sur un accès, à moins de disposer de plusieurs adresses publiques (une variante exposée plus loin permet de contourner ce problème).
- L'adresse IP publique est imposée par le réseau de l'opérateur ; sur certains accès ADSL elle est dynamique (pas forcément fixe dans le temps). Cela complique l'établissement de liaisons depuis un appareil distant.
- Dans le cas particulier des routeurs avec NAT « symétrique », ce système ne fonctionne pas, car les ports découverts au moyen de STUN ne sont pas valables pour le transfert ultérieur avec le codec distant.
- La faculté d'appel depuis un codec extérieur est valable pour n'importe quel appareil ou entité extérieure, y compris les interventions indésirables (voir 3.2, Connexions indésirables) ! **Ce système crée donc une vulnérabilité en offrant à l'extérieur un accès sans contrôle.**
- Les améliorations de la sécurité impliquant le pare-feu (voir plus loin) sont assez complexes à mettre en œuvre et maintenir.

5.5.4. Recommandations et variantes

Ce système possède beaucoup d'inconvénients, en particulier au niveau de la sécurité.

Pour les limiter, tout d'abord il faut éviter de rediriger sinon ouvrir des ports non nécessaires. Par exemple, rediriger le port TCP 80 (http) est un risque inutile à moins de vouloir effectivement permettre un accès distant au serveur html embarqué du codec.

⇨ *Sur les codecs AETA, ne pas oublier de configurer un mot de passe de contrôle d'accès aux pages html si vous ouvrez cet accès à l'extérieur !*

En revanche, il ne faut pas que le pare-feu éventuel empêche l'ouverture de ces ports par le codec lorsqu'il initie une liaison :

- Envoi de paquets SIP/UDP du port 5060 du codec vers le port SIP (habituellement 5060) du codec distant.
- Envoi de paquets RTP et RTCP/UDP des ports 5004/5005 vers les ports RTP et RTCP/UDP (5004/5005 d'habitude) du codec distant.
- Envoi de requêtes vers serveur STUN (port 3478).

⇨ *Les numéros de port indiqués ci-dessus sont des valeurs standard, que l'on trouvera sur les codecs AETA. Sur ces derniers il est possible de les modifier au besoin.*

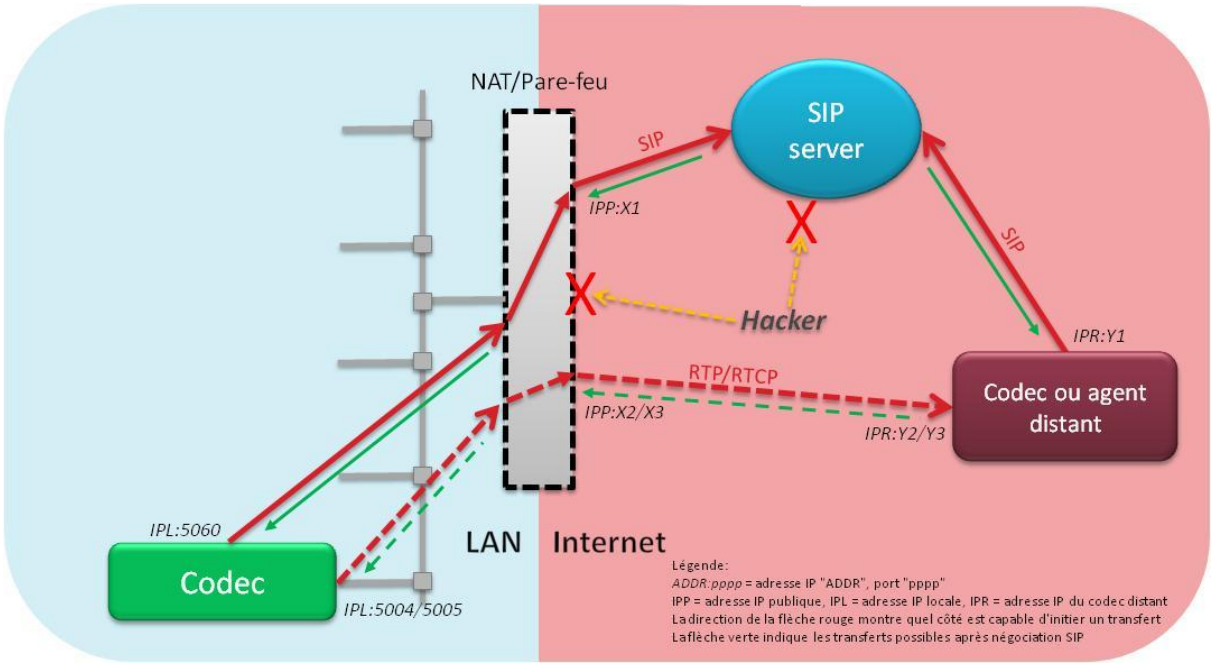
Divers variantes ou compléments peuvent améliorer le fonctionnement et/ou la sécurité :

- Pour utiliser plusieurs codecs, attribuer à chacun un port différent pour SIP, au lieu du standard 5060. *Si nécessaire, le port SIP de chaque codec AETA peut être configuré autrement que sur le numéro 5060.*
  Ainsi, une seule adresse IP publique est « consommée », et un agent distant peut choisir la destination en spécifiant le numéro de port au lieu d'utiliser implicitement 5060.
- Même avec un seul codec, il est intéressant de configurer un port autre que 5060 ou 5061. Cela va légèrement compliquer l'appel depuis un distant, mais réduit un peu la vulnérabilité aux appels intempestifs. Les hackers scannent en priorité les ports standards 5060 et 5061. Mais ceci n'est qu'un frein, pas une garantie !
- Au cas où l'on observe des attaques, il est possible dans le pare-feu de placer en liste noire les adresses IP sources de ces tentatives. Mais ceci est très provisoire.
- Plus efficace, la liste blanche : s'il est possible de définir une liste restreinte d'adresses IP distantes habilitées à établir des liaisons, configurer le pare-feu pour n'accepter que les données depuis ces adresses (vers les ports redirigés).
  Attention toutefois à ne pas bloquer les requêtes STUN !

5.6. NAT et serveur SIP

5.6.1. Principe

Pour contourner les inconvénients majeurs des méthodes décrites précédemment, il est préférable de mettre en œuvre des entités d'infrastructures telles qu'un serveur SIP. Bien entendu, ceci n'est valable que pour les codecs utilisant le protocole SIP. Dans le cas ci-dessous ce serveur est situé sur Internet :



Les paquets qui doivent passer à travers le routeur d'accès sont de deux types :

- Signalisation au protocole SIP/UDP
- Flux audio codé au protocole RTP/UDP

Avec cette architecture, c'est toujours le codec local qui est à l'initiative des ouvertures de chemins à travers le routeur :

- Requêtes SIP pour l'enregistrement du codec sur le serveur SIP ; ensuite un mécanisme maintient par relance périodique cette voie de communication codec/serveur.
- Requêtes STUN[1] pour permettre au codec de connaître les adresses et ports pour les transferts RTP.

A l'enregistrement, le serveur contrôle l'identité du codec (authentification avec compte SIP et mot de passe) et conserve une suivi de la présence et de la localisation (adresse IP et port SIP publics) de l'agent. Les transactions, à l'initiative de l'un ou l'autre des agents enregistrés, passent systématiquement par l'intermédiaire du serveur SIP.

En revanche, les flux RTP peuvent être échangés directement entre les agents sans nécessairement transiter par le serveur.

Ce système permet de se dispenser de toute route entrante fixe (ouverture et redirection de port), car le serveur gère dynamiquement les allocations de ports publics.

[1] Non représentées sur le diagramme pour simplification

5.6.2. Avantages

- Une fois installé le serveur SIP, la mise en place d'un codec sur site est assez simple.
- Le codec est identifié et joignable par son URI SIP, fixe car lié au compte SIP. Donc pas de problème avec les changements d'adresse IP, avec ou sans changement de situation géographique.
- Il n'y a pas d'obstacle à l'installation de plusieurs codecs sur un même site.
- Pas de nécessité de configurer le routeur NAT de manière spécifique (redirections, etc.).
- Les agents qui appellent ou sont appelés doivent préalablement être authentifiés pour être enregistrés. Un agent indésirable ne peut pas s'immiscer en initiant une session sans être authentifié[1].
- Le codec peut aussi communiquer directement avec d'autres codecs situés dans le réseau local.
- Il n'y a pas nécessité de créer des vulnérabilités en ouvrant statiquement des ports que des intrus pourraient exploiter. L'extérieur ne peut pas initier un échange simplement, et encore moins atteindre l'interface de contrôle du codec, à moins qu'on le permette volontairement, pour des besoins de télégestion.
- Même un accès via réseau mobile peut être utilisé pour relier un codec, sous réserve toutefois de résoudre le problème fréquent du NAT symétrique (voir plus loin).

5.6.3. Inconvénients

- Il faut disposer au départ d'un serveur SIP avec les comptes et URI ; l'installation est assez complexe. De plus le serveur est un point de panne unique pour le système global et doit donc être convenablement sécurisé.

⇨ *Cependant il est possible de s'appuyer sur un tiers ; par exemple AETA propose un serveur SIP, fiable et dédié aux applications broadcast.*

- Il existe certains cas de blocage de SIP par l'opérateur de réseau, soit au niveau du réseau, soit par le routeur, exemples rencontrés à l'occasion :
  + Blocage du port 5060 dans le routeur (mis en place et géré par l'opérateur)
  + Blocage des flux RTP dans un réseau mobile (blocage VoIP par l'opérateur)
  Dans de tels cas il faut obtenir, si possible, une levée du blocage par l'opérateur.
- De manière similaire, le fonctionnement peut être gêné ou bloqué par un pare-feu trop restrictif.
- Il existe une possibilité pour un élément extérieur de détecter les ports publics (à allocation dynamique) lors des transactions légitimes et de faire intrusion par ces voies.

[1] Pour être très précis, en fait cela *est* possible, si le serveur est configuré pour autoriser un tel comportement « permissif ». Mais cela n'est pas la pratique commune, et est évidemment déconseillé pour une exploitation sûre.

5.6.4. Recommandations et variantes

Bien qu'elle implique un effort initial d'installation, une telle architecture apporte de gros avantages, d'abord fonctionnels, mais aussi pour la sécurité, qui est le sujet principal de ce document. On peut obtenir un niveau de sécurité très appréciable pour un coût global modéré, surtout en ajoutant quelques ajustements :

- Ne pas ajouter de redirections (comme dans la méthode décrite en 5.5), qui sont inutiles dans cette configuration et peuvent seulement créer des vulnérabilités.
- En revanche, il ne faut pas que le pare-feu éventuel soit trop restrictif et empêche l'ouverture par le codec des routes vers le serveur SIP, le serveur STUN éventuel, et l'envoi du flux RTP vers le codec distant lors d'une liaison.
- Il peut être intéressant de configurer le serveur SIP avec un port non standard (autre que 5060 ou 5061). Très souvent cela peut contourner les blocages provoqués par les équipements des opérateurs.

⇨ *Avec le serveur SIP d'AETA, il est possible d'utiliser ainsi un port alternatif.*

- Pour renforcer la sécurité, il est possible de n'autoriser, au niveau du pare-feu, que l'adresse IP du serveur SIP (et celle du serveur STUN éventuel) pour tout transfert au protocole/port SIP.
  Ainsi une sollicitation de session indésirable est bloquée dans tous les cas.

5.6.5. Cas du NAT symétrique

Dans le cas particulier des routeurs avec NAT « symétrique », les ports découverts au moyen de STUN ne sont pas valables pour le transfert ultérieur avec le codec distant.

Certains serveurs SIP peuvent contourner ce problème grâce à l'association avec un proxy RTP. Dans ce cas, le serveur va tout d'abord détecter la situation de NAT symétrique au niveau du codec ; ensuite lors de la session, au lieu que les flux soient échangés de pair à pair comme d'habitude, le proxy RTP va relayer les paquets pour assurer le transfert complet de bout en bout. Cela s'effectue au prix d'un « détour » des flux par le serveur, mais la liaison est assurée malgré le(s) NAT symétrique(s) sur le chemin.

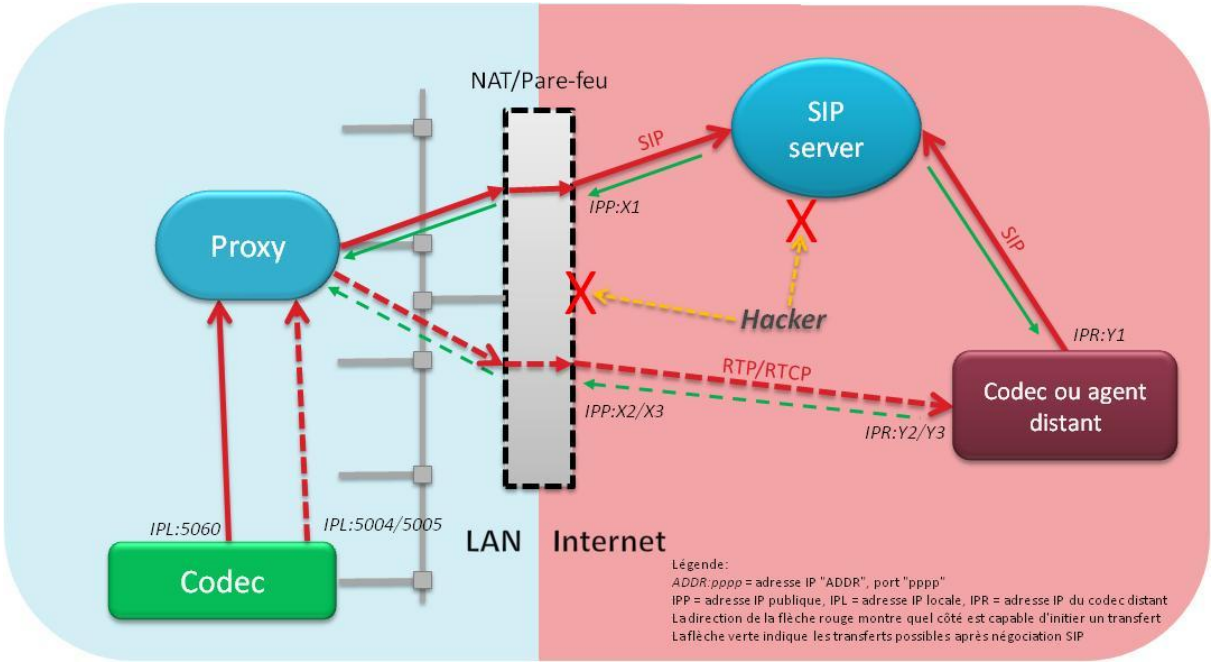
Cette solution est particulièrement utile pour les cas où le routeur NAT symétrique est inévitable, comme presque toujours via un accès réseau mobile.

⇨ *Le serveur SIP AETA propose aussi cette fonction.*
Plus généralement, le serveur SIP AETA est un bon moyen de mettre en place rapidement une organisation fondée sur l'utilisation d'un serveur SIP, mais sans avoir à se préoccuper d'une installation complexe, ni à investir dans les moyens matériels nécessaires, avec ce que cela implique en maintenance.

5.7. NAT, serveur SIP et proxy

5.7.1. Principe

Partant de l'architecture précédente, on peut renforcer la sécurité en ajoutant sur le site local un *outbound proxy* qui sert d'intermédiaire lors des sessions avec l'extérieur.



Toute la signalisation est relayée par le proxy, mais il ne gère pas les enregistrements ni les présences, qui restent le rôle du serveur SIP. Il relaye aussi les flux RTP. Seul le proxy effectue les transferts à travers le routeur vers l'extérieur.

- Le pare-feu peut bloquer totalement l'accès extérieur *en direct* à tous les codecs.
- En revanche, il peut autoriser spécifiquement le proxy à traverser le routeur, uniquement pour les protocoles utiles et seulement les ports réservés.

5.7.2. Avantages

Outre les avantages liés à l'utilisation du serveur SIP :

- Très grande sécurité : les codecs sont totalement isolés d'Internet. Même le proxy n'est exposé que pour les accès indispensables.

5.7.3. Inconvénients

- Un équipement de réseau à gérer de plus. Cependant sa configuration n'est pas nécessairement complexe (pas de base de données de comptes utilisateurs à maintenir).
- Configuration du codec légèrement plus complexe.

⇨ *Concrètement, sur les codecs AETA, ajout dans les paramètres SIP de l'adresse IP du proxy. Attention, pour un codec « nomade », cette partie du réglage est liée au lieu où se trouve le codec !*

5.7.4. Recommandations

Les recommandations pour le cas précédent s'appliquent aussi à cette architecture.

Lorsque le serveur SIP est situé à l'intérieur du réseau d'entreprise, il peut être intéressant de combiner dans le même équipement le serveur SIP et le proxy. Beaucoup de serveurs SIP offrent d'emblée cette possibilité.

5.8. NAT, serveur SIP et SBC

5.8.1. Principe

Le SBC (Session Border Controller) est un équipement qui regroupe en une entité cohérente les fonctions d'accès Internet, pare-feu et proxy SIP. Il est alors positionné en lieu et place du routeur/pare-feu des architectures décrites plus haut.

Le SBC exerce un contrôle poussé sur les sessions VoIP/AoIP à travers l'accès Internet, avec une grande variété possible dans le degré d'intervention et les fonctions supportées. Il permet d'obtenir un niveau de sécurité maximal, et ce type d'outil est notamment utilisé pour sécuriser les organisations avec une capacité importante de gestion VoIP.

Pour une description plus complète des fonctions et techniques développées dans un SBC, on peut consulter la page suivante :

http://en.wikipedia.org/wiki/Session_Border_Controller

5.8.2. Avantages

- Sécurité maximale contre toutes les formes d'attaque.
- Possibilités nombreuses d'adaptation fine aux besoins et environnement de l'application (sélection des ressources extérieures habilitées, filtrage des types de session permises, etc.).

5.8.3. Inconvénients

- L'équipement nécessaire est coûteux ; pour cette raison ce type de solution n'est pas adapté à des organisations de petite taille.
- L'installation et la gestion d'un SBC sont assez complexes, et requièrent un responsable réseau compétent. Comme pour un pare-feu sophistiqué, une connaissance insuffisante du système peut mener soit à des brèches involontaires dans la sécurité, soit au contraire à des blocages excessifs gênant l'exploitation.

AETA peut vous conseiller si vous souhaitez examiner ce type de système (consultez l'équipe commerciale).

6. Conclusions

Ce tour d'horizon est certes partiel et simplifié sur certains points pour plus de clarté. Mais l'objectif est de mettre en évidence les principes habituellement appliqués et les problèmes essentiels rencontrés dans les installations.

Parmi les grandes lignes, un principe très banal : rien n'est gratuit. Les solutions qui sont ou paraissent très simples à mettre en œuvre ne sont pas fiables ou pas sûres. Au bout du compte, après ajout de divers palliatifs on en arrive à un système qui n'est plus vraiment simple.

A l'inverse, au prix d'un effort initial pour s'appuyer sur une infrastructure fiable, l'exploitation au jour le jour peut s'avérer réellement simple, et sans mettre en péril la sécurité.