Gestion des Risques dans les Infrastructures VoIP

Oussema Dabbebi

UNIVERSITÉ DE LORRAINE

École doctorale IAEM Lorraine

Gestion des risques dans les infrastructures VoIP

THÈSE

présentée et soutenue publiquement le 03 Juin 2013

pour l'obtention du

Doctorat de l'Université de Lorraine

(mention informatique)

par

Oussema DABBEBI

Composition du jury

Rapporteurs :	Frédéric CUPPENS, Professeur à TELECOM Bretagne, Brest Hervé DEBAR, Professeur à TELECOM SudParis, Evry
Examinateurs :	Claude GODART, Professeur à l'Université de Lorraine, Nancy Samuel DUBUS, Ingénieur de Recherche à Alcatel Lucent Bell Labs, Nozay Olivier FESTOR, Directeur de Recherche à l'INRIA Nancy Grand Est, Nancy Rémi BADONNEL, Maître de Conférences à TELECOM Nancy, Nancy

Laboratoire Lorrain de Recherche en Informatique et ses Applications — UMR 7503

Loria
Laboratoire lorrain de recherche en informatique et ses applications

Mis en page avec la classe thloria.

Table des matières

Table des figures

Liste des tableaux

Résumé

La téléphonie sur IP est devenue un nouveau paradigme pour établir et transmettre les communications téléphoniques directement sur les réseaux IP de manière flexible et à faible coût. Toutefois, les services VoIP sont confrontés à plusieurs problèmes de sécurité qui sont soit hérités de la couche IP soit spécifiques au service lui-même. Une grande variété de mécanismes de protection sont disponibles pour y faire face. Cependant, ces services nécessitent des performances et une disponibilité du réseau élevées, et les mécanismes de protection peuvent nuire à ces performances. La gestion des risques offre de nouvelles perspectives à l'égard de cette problématique. Nos contributions portent sur l'application et l'automatisation de la gestion de risques dans les infrastructures VoIP selon trois axes. Le première axe porte sur l'automatisation du processus de gestion des risques dans un réseau VoIP d'entreprise. Dans ce cadre, nous avons développé un modèle pour évaluer les risques, un ensemble de contremesures progressives et des algorithmes de mitigation. Nous l'avons couplé à un système de détection d'anomalies basé sur les SVM et un mécanisme d'auto-configuration qui peut fournir un retour d'expérience sur l'efficacité des contremesures. Le deuxième axe concerne l'extension de notre stratégie dans les réseaux P2PSIP. Nous avons mis en place une solution adaptée à la nature distribuée des environnements pair-à-pair. Nous nous sommes aussi intéressés à l'architecture RELOAD et avons étudié comment traiter les attaques résiduelles à travers des mécanismes de confiance. Nous avons enfin étudié les services VoIP dans le cloud où nous proposons plusieurs stratégies pour le déploiement et l'application des contremesures.

Abstract

IP telephony has become a new paradigm that permits to establish and transmit voice communications with IP networks. Its deployment has been accelerated by the standardization of dedicated signaling protocols. However, VoIP services are faced to several security issues which are inherited from the IP layer or specific to the service. A large variety of protection mechanisms are available to deal with them. However, IP telephony is a real-time service which requires high network performance. The application of countermeasures may significantly affect such a critical service. Risk management provides new perspectives for this issue. This thesis deals with the application of risk management in VoIP infrastructures. The first axis consists in the automation of the risk management process in VoIP enterprise network. In this context, we have developed a mathematical model for assessing risk, a set of progressive countermeasures to counter attackers and mitigation algorithms that evaluate the risk level and takes the decision to activate a subset of countermeasures. To improve our strategy, we have coupled it with an anomaly detection system based on SVM and a self-configuration mechanism which provides feedback about countermeasure efficiency. The second axis deals with the extension of our adaptive risk strategy to P2PSIP infrastructures. We have implemented a specific risk model and a dedicated set of countermeasures with respect to its peer-to-peer nature. For that, we have identified attack sources and established different threat scenarios. We have analysed the RELOAD framework and proposed trust mechanisms to address its residual attacks. Finally, the third axis focuses on VoIP services in the cloud where we have proposed a risk strategy and several strategies to deploy and apply countermeasures.

Introduction

Chapitre 1

Introduction générale

Sommaire

1.1 Contexte scientifique

La téléphonie sur IP (ToIP) est un nouveau paradigme qui a connu un engouement particulier ces dernières années. Elle permet la transmission de la voix sur les réseaux IP et l'Internet à travers des méthodes et techniques permettant l'établissement et la gestion des sessions d'appels. Le déploiement des services VoIP à grande échelle a été soutenu par l'accès au très haut débit qui fournit de meilleures performances réseau, par la standardisation de protocoles dédiés comme le protocole de signalisation SIP et aussi par le développement de produits logiciels conviviaux comme Skype. La téléphonie sur IP a rapidement été perçue, par les communautés de la recherche et les fournisseurs de services, comme une solution de communication potentielle permettant de réduire les coûts de déploiement et de maintenance.

L'infrastructure de la ToIP s'appuie sur l'exploitation de plateformes IP et des services associés tels que les services DNS et DHCP. Comme le réseau IP est un réseau de commutation de paquets où la bande passante peut davantage être optimisée, la ToIP représente une alternative extensible et moins coûteuse que la téléphonie classique qui est fondée sur un réseau de commutation de circuits (RTC) [35]. Une autre différence avec le réseau RTC est la distribution de services : tout service en relation avec le média est centralisé dans le cœur du réseau RTC. En revanche, dans le cas d'une infrastructure ToIP, cette intelligence peut être distribuée entre les différents équipements. Par conséquent, le service n'est pas exclusif aux opérateurs ce qui contribue à une concurrence et peut réduire les coûts. Ses inconvénients sont surtout liés à la disponibilité du service et la qualité de service. Sa disponibilité est souvent inférieure à celle de la téléphonie classique parce que son infrastructure est basée sur le réseau à commutation de paquets. Ce service dépend fortement de la bande passante et de la disponibilité des serveurs. Ainsi, la garantie de la qualité de service (QoS) est un challenge important qui inclut le temps de latence, la gigue et la perte de paquets.

La téléphonie sur IP est confrontée à différentes attaques de sécurité à savoir les attaques classiques héritées de la pile protocolaire TCP/IP et des attaques spécifiques à sa couche applicative. La classification introduite par l'alliance VoIPSA [1] considère plusieurs catégories d'attaques [112]. La première catégorie inclut les attaques d'interruption de service comme les attaques par inondation de messages SIP, les attaques de déni de service (DoS) et les attaques SPIT (*Spam over Internet Telephony*). La deuxième catégorie comprend les attaques liées à l'écoute et l'analyse du trafic de signalisation et de transfert média. La troisième catégorie d'attaques porte sur l'usurpation d'identité et l'utilisation illégitime des adresses IP et SIP-URI ainsi que la falsification de ces adresses. Pour la quatrième catégorie, nous trouvons les attaques d'accès non autorisé comme celles s'appuyant sur des techniques d'homme du milieu et également les attaques d'ingénierie sociale et les fraudes.

1.2 Problématique

Une grande variété de mécanismes de détection et de prévention ont été développés pour identifier et contrer les attaques de sécurité auxquelles est confrontée la téléphonie sur IP. Cependant, les méthodes de détection montrent rapidement leurs limites en termes de sensibilité et de spécificité. En fait, la potentialité d'une attaque est souvent difficile à estimer car elle dépend fortement de la nature de l'attaque et la durée de détection peut être importante par rapport à l'échelle de temps de cette attaque. D'autre part, les mécanismes de protection peuvent avoir un impact significatif sur les performances du service de téléphonie par rapport à sa continuité opérationnelle et sa qualité de service. Par exemple, les pare-feux peuvent bloquer l'établissement de session dès qu'un appelant ne respecte pas une règle définie.

La téléphonie sur IP pose des contraintes parfois antagonistes en termes de performances et de sécurité. Les utilisateurs attendent des performances proches du réseau téléphonique commuté (RTC) classique. C'est un service en temps réel nécessitant une qualité de service suffisamment élevée, comme un taux de rejet d'appels faible et une gigue constante afin de garantir la qualité de la communication. Il est considéré comme un service critique qui nécessite de solides performances réseau pour maintenir son état de fonctionnement à un niveau acceptable. Généralement, un appel VoIP devient incompréhensible dès que le retard de transmission de messages est supérieur à 150 ms ou la perte de paquets est supérieure à 5%.

Aussi, l'objectif de notre travail consiste à appliquer et automatiser la gestion des risques de façon dynamique pour sécuriser les architectures de téléphonie sur IP tout en maintenant la qualité du service et sa continuité opérationnelle. La gestion des risques est un processus qui consiste à identifier, estimer et traiter les risques liés à une ou plusieurs menace(s) afin de les réduire à un niveau acceptable. Son automatisation requiert un couplage fort entre les différentes opérations de ce processus.

1.3 Organisation du manuscript

Le manuscript comprend neuf chapitres organisés en trois parties. Ces trois parties correspondent respectivement à (1) l'état de l'art relatif à la téléphonie sur IP et la gestion des risques, (2) la présentation de notre approche de gestion des risques pour la téléphonie sur IP et (3) la mise en œuvre de notre approche à travers le prototypage. Les différentes contributions sont illustrées sur la figure 1.1. Ce manuscript se termine par une synthèse et l'identification d'un ensemble de travaux futurs.

1. VoIP Security Alliance

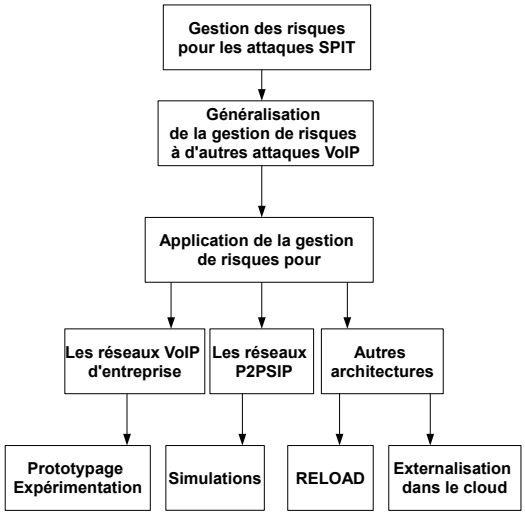


FIGURE 1.1 – Les contributions de la thèse

1.3.1 Partie I : Téléphonie sur IP et gestion des risques

La première partie présente tout à d'abord la téléphonie sur IP (ToIP), ses fondements, ses architectures et les protocoles indispensables pour sa mise en œuvre ainsi que les différents problèmes de sécurité. Nous décrivons ensuite les travaux existants, de l'évaluation au traitement des risques, afin de montrer leurs limites et positionner notre approche de gestion des risques.

Téléphonie sur IP, fondements et attaques de sécurité

Le chapitre 2 présente tout d'abord les fondements de la voix sur IP. Nous commençons par la présentation des protocoles relatifs aux services VoIP. Nous nous focalisons sur le protocole de signalisation SIP et le protocole de transfert média RTP, ainsi que les protocoles support comme SDP. Ensuite, nous nous intéressons aux architectures VoIP à savoir l'architecture VoIP centralisée typiquement basée sur un serveur IPBX. Nous décrivons les éléments de base comme le serveur proxy et le serveur d'enregistrement. Nous présentons également l'architecture P2PSIP où les services VoIP sont intégrés dans une infrastructure pair à pair s'appuyant sur une table de hachage distribuée. Nous décrivons les défis rencontrés pour adapter la VoIP à un tel support. Enfin, nous présentons les menaces de sécurité contre la signalisation, le transfert média et contre les services support.

De l'évaluation au traitement des risques

Le chapitre 3 dépeint le processus de gestion des risques. Nous commençons par définir les différents concepts de base comme l'actif, la menace, la vulnérabilité et le risque. Ensuite, nous présentons les étapes du processus en commençant par l'identification des risques, leur estimation jusqu'à la phase de traitement des risques. Lors de la première phase, le processus permet d'identifier les menaces, les vulnérabilités et les actifs critiques. La phase d'estimation peut typiquement s'appuyer sur deux grandes catégories de modèles : les modèles quantitatifs et les modèles qualitatifs. Nous distinguons plusieurs techniques pour le traitement des risques incluant les stratégies d'évitement des risques, d'optimisation ou d'acceptation des risques. Nous classons les travaux de recherche existants suivant ce processus et indiquons comment ils contribuent à ces

différents étapes. Enfin, nous montrons les limites de ces travaux pour répondre au compromis sécurité-performance indispensable dans de tels services critiques.

1.3.2 Partie II : Approche de gestion des risques pour la téléphonie sur IP

La seconde partie présente nos travaux de recherche portant sur une nouvelle approche de sécurité pour la téléphonie sur IP qui a pour objectif de minimiser l'exposition aux menaces de sécurité tout en maintenant la qualité de service. Cette approche consiste en l'application de la gestion des risques dans ces infrastructures. Cette partie inclut de nombreux résultats expérimentaux obtenus par la simulation et le prototypage. Ceux-ci permettent l'évaluation des avantages et des limites de notre solution de sécurité.

Stratégie de gestion des risques

Le chapitre 4 décrit notre stratégie de gestion de risques pour les réseaux VoIP d'entreprise. Nous appliquons et automatisons la gestion des risques d'une manière dynamique et à grains fins afin de contrôler l'exposition de cette architecture face aux menaces [79]. Nous nous focalisons dans ce premier travail sur la prévention des attaques SPIT. Cette approche vise à offrir un couplage fort entre la détection d'anomalies, l'évaluation et le traitement des risques. Elle comprend principalement quatre éléments : un système de détection d'intrusions par anomalies, un modèle quantitatif de risque, des algorithmes de mitigation et un ensemble de contremesures. Ensuite, nous montrons comment il est possible d'intégrer une technique de détection d'anomalies basée sur les machines à vecteurs supports (SVM) développée au sein de notre équipe [114] afin d'améliorer les performances de notre stratégie de sécurité [74]. Nous adaptons alors notre modèle de risque afin qu'il intègre les paramètres de cette technique de détection. Une architecture est développée pour soutenir la stratégie de gestion des risques. Enfin, des résultats expérimentaux sont présentés afin d'en quantifier les bénéfices.

Modèle de risques étendu et paramétrisation

Le chapitre 5 présente notre stratégie de gestion des risques étendue aux autres attaques VoIP. Nous proposons de généraliser notre modèle de risques et d'en évaluer son comportement pour différentes attaques VoIP [78]. Pour cela, nous classons ces attaques en trois classes suivant leur observabilité. Ensuite, nous définissons la potentialité des attaques pour chaque classe en nous basant sur sa propriété d'observabilité. Nous montrons les avantages et les limites de cette généralisation par une série d'expériences par simulation. Ces simulations portent sur différents critères comme le risque, le coût induit par l'application d'une contremesure et la disponibilité. Nous nous intéressons aussi aux différents paramètres de notre stratégie de risques comme le seuil, l'impact d'une contremesure et le nombre de contremesures disponibles. Nous proposons également une stratégie d'auto-configuration basée sur le retour d'expérience. Dans ce cas, nous présentons un mécanisme fondé sur le retour d'expérience [48] pour raffiner notre modèle quantitatif [29]. Nous focalisons ensuite notre travail sur le coût d'application d'une contremesure car ce facteur est important dans la sélection des contremesures lors du traitement des risques. Enfin, nous évaluons les performances de ce mécanisme et nous montrons l'impact du taux d'erreur d'un paramètre sur le traitement des risques.

Extension aux réseaux pair-à-pair SIP

Le chapitre 6 décrit l'extension de notre stratégie de gestion des risques dans les architectures distribuées P2PSIP [28]. Le protocole P2PSIP a pour objectif de fournir une solution pour que les utilisateurs SIP communiquent en s'appuyant sur une infrastructure pair-à-pair [1]. Il exploite une table de hachage distribuée (DHT) pour l'enregistrement et la localisation des utilisateurs. Nous étudions différents scénarios d'attaques à la frontière entre le pair à pair et la voix sur IP, notamment des scénarios relatifs au déni de service et à la manipulation de sessions. Ensuite, nous proposons une solution adaptée à cette architecture distribuée afin de contrôler dynamiquement son exposition. Elle comprend un modèle quantitatif de risques pour les menaces identifiées et un ensemble de contremesures qui respectent la nature distribuée de ce réseau comme la correction par enregistrement dans la DHT et l'authentification distribuée. Enfin, nous évaluons les performances de cette solution à travers un ensemble de simulations réalisées sur OMNET++ relatives au risque et au coût induit par l'application des contremesures.

Architectures hybrides et externalisation dans le cloud

Le chapitre 7 porte sur l'étude des architectures hybrides et l'externalisation dans le cloud. Nous nous intéressons au framework RELOAD et évaluons sa complémentarité avec nos travaux. L'infrastructure RELOAD offre une solution de sécurité centralisée pour les réseaux P2PSIP. Elle fournit un serveur de certification qui garantit trois niveaux de sécurité pour le routage, l'enregistrement dans la DHT et la communication entre les pairs SIP. Nous nous focalisons sur la remédiation des attaques résiduelles comme le refus de service [30]. La solution consiste en l'intégration de mécanismes de confiance basés sur l'algorithme *eigentrust* couplé à des contremesures pour contrôler le trafic entrant et sortant des pairs dont les scores de confiance sont en dessous d'un seuil défini. Pour les services VoIP dans le cloud, nous proposons l'externalisation des contremesures en tant que services de la couche applicative (*SaaS*) afin d'améliorer et faciliter leurs utilisations [31]. Nous proposons différentes stratégies d'application de contremesures qui offrent des alternatives de sécurité dans le cas de pannes d'une contremesure. Enfin, nous évaluons les performances de ces différentes stratégies de gestion afin de déterminer leurs impacts sur le risque et la disponibilité des services VoIP.

1.3.3 Partie III : Mise en œuvre de la solution

Le chapitre 8 présente la mise en œuvre de nos travaux par le prototypage à travers le développement de différents composants dans un serveur Asterisk [32]. Ce serveur open source offre plusieurs fonctionnalités et services VoIP à savoir la signalisation, la vidéo conférence et la messagerie vocale. Il propose des solutions pour contourner le NAT comme l'application des mécanismes STUN[2]. Il implante différents protocoles de signalisation comme H.323 et SIP et peut jouer le rôle d'un serveur proxy, d'un serveur d'enregistrement et de passerelle avec les réseaux publics. Ainsi, une architecture VoIP d'entreprise repose très souvent sur ce serveur qui englobe plusieurs services. Un serveur Asterisk offre une interface de programmation (AGI[3]) qui permet le développement et l'intégration d'applications externes. L'interface de programmation AGI permet de programmer en plusieurs langages. Nous utilisons cette interface avec Python pour implanter notre solution de gestion des risques au sein de cet IPBX. Cette implémentation respecte l'architecture de la solution de gestion des risques définie pour les réseaux

2. STUN, Simple Traversal of User Datagram Protocol, www.ietf.org/rfc/rfc3489.txt
3. Asterisk Gateway Interface

VoIP d'entreprise. Elle complète les résultats obtenus par la simulation au travers d'un ensemble d'expériences pratiques. L'implantation de notre stratégie de risques comprend le prototypage du gestionnaire de risques, du système de détection d'intrusions et du système d'auto-configuration ainsi qu'un ensemble de contremesures. Nous montrons à chaque fois leur intégration au sein du serveur Asterisk. En complément des expérimentations présentées dans les précédents chapitres, nous avons validé la mise en œuvre de notre approche par la pratique. Nous réalisons cette validation relativement aux attaques SPIT afin de vérifier l'impact de notre solution de risques sur les attaques. Les algorithmes de mitigation sont évalués au travers de différents scénarios de tests. Enfin, la conclusion fournit une synthèse des différentes contributions réalisées pendant la thèse et présente un ensemble de perspectives de recherche.

Première partie

Téléphonie sur IP et gestion des risques

Chapitre 2

Téléphonie sur IP, fondements et attaques de sécurité

2.1 Introduction

Ce chapitre présente les concepts de la téléphonie sur IP. Nous décrivons l'architecture fonctionnelle de la téléphonie sur IP. Nous traitons ensuite le protocole de signalisation SIP qui permet l'établissement, le maintien et la terminaison de sessions. Nous présentons également les protocoles média RTP/RTCP qui assurent la transmission de la voix et les protocoles supports. Nous décrivons ensuite le protocole P2PSIP qui correspond à une version décentralisée du protocole SIP. Enfin, nous décrivons et classons les différentes types d'attaques qui peuvent survenir dans ces environnements.

2.2 Concepts de la téléphonie sur IP

Nous détaillons dans cette section l'architecture fonctionnelle de la téléphonie sur IP. Nous présentons aussi les différents protocoles sous-jacents.

2.2.1 Architecture fonctionnelle

Une architecture VoIP doit garantir la gestion de sessions d'appel, de leur établissement à la terminaison et doit offrir des fonctions essentielles comme la localisation et la vérification des capacités du client [35]. Les services VoIP sont typiquement implantés dans une architecture centralisée de type client-serveur, mais peuvent aussi être déployés dans les réseaux et le cloud.

Nous trouvons classiquement dans l'architecture fonctionnelle deux types de composants à savoir l'agent client qui représente l'utilisateur et gère les requêtes, et les serveurs SIP qui reçoivent les requêtes et soutiennent l'établissement des échanges. Ces principaux composants sont représentés sur la figure 2.1 où nous montrons un scénario d'établissement de session où les deux clients SIP échangent des requêtes en exploitant les serveurs proxy et registrar.

- **L'agent client SIP :** il constitue de l'élément de base de l'architecture et se subdivise en deux parties : une partie client notée UAC (*user agent client*) qui envoie les requêtes vers les serveurs SIP et traite les réponses et une partie serveur notée UAS (*user agent server*) qui reçoit les requêtes et les traite. Le rôle d'un agent client comprend la gestion des sessions, la négociation des paramètres et l'envoi du média sous forme de messages RTP ou RTCP ainsi que son traitement par le biais de codecs.

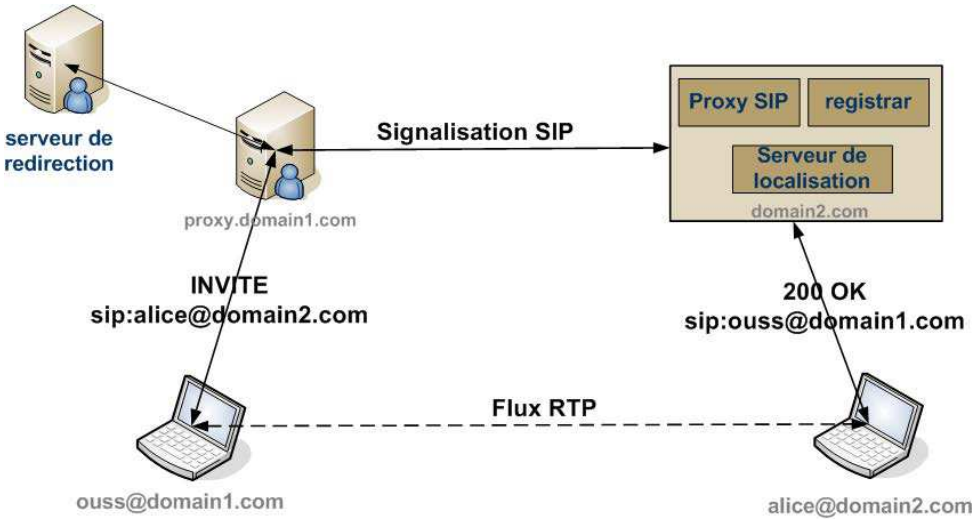


FIGURE 2.1 – Architecture fonctionnelle d'un réseau SIP

- **Les serveurs SIP :**

  - **Les serveurs d'enregistrement (*registrar*) et de localisation :** ils assurent la correspondance entre l'adresse SIP URI et l'adresse du contact (généralement sous forme d'une adresse SIP URI, une adresse IP et un port). le serveur registrar permet l'enregistrement de cette correspondance, tandis que le serveur de localisation permet de la retrouver. La difficulté dans le réseau VoIP est d'utiliser les adresses pour communiquer entre utilisateurs. Pour cette raison, SIP propose son propre adressage. Le serveur d'enregistrement répond à chaque requête REGISTER par la liste des adresses du contact correspondant au SIP URI[4] concerné. Ainsi, le protocole SIP ne limite pas le nombre d'adresses de contact par adresse SIP URI. Il permet aussi l'authentification des nouveaux clients en utilisant la technique du secret partagé [98].

  - **Le serveur proxy :** il s'agit du serveur principal de l'architecture VoIP. Il joue le rôle d'une passerelle entre les agents clients. Il reroute les requêtes entre UAC et UAS et vice-versa pour les messages réponse. Il utilise plusieurs stratégies de routage à savoir la politique de route la moins chère ou la consultation des annuaires. Il consulte aussi plusieurs bases de données pour déterminer l'adresse de contact correspondant à l'adresse SIP de l'appelant. Il utilise le serveur DNS pour trouver les différents noms de domaine. Les tâches exécutées par le serveur proxy après la réception d'une requête sont :
    - L'authentification de la requête : il s'agit de vérifier si le propriétaire de l'adresse SIP URI est la source de la requête pour éviter toute attaque d'usurpation d'identité. Cette tâche est essentielle pour le fournisseur notamment pour la facturation. Cette authentification peut être opérée par l'envoi du mot de passe déjà enregistré dans le serveur d'enregistrement.
    - La décision de routage : elle consiste à trouver à qui et où la requête est transmise. Cette décision peut être gouvernée par une politique du fournisseur ou du client. Elle a aussi pour but de déterminer le prochain nœud à qui les requêtes SIP seront envoyées.
    - La modification de la requête : le proxy a la capacité de modifier les requêtes SIP

4. Uniform Resource Identifier

conformément à la définition du protocole SIP par la mise à jour du champ REQUEST-URI dans le message SIP avec la valeur du prochain saut. Le proxy SIP met son adresse du contact dans le champ *Via* pour garantir que la réponse de la requête repasse par lui.

- **Le serveur de redirection :** il s'agit du serveur responsable du traitement des appels entrants. Il est utile dans le cas de changement d'emplacement physique de l'utilisateur afin de rediriger ses appels vers son nouvel emplacement. Ce serveur peut servir à équilibrer la charge entre les différents serveurs. Il utilise la famille de requêtes de type 3xx (redirection) pour demander au client SIP d'emprunter une autre route. Cette requête propose à l'UAC une (ou des) adresse(s) physique(s) alternative(s) de proxy SIP.

- **La passerelle média :** le rôle de la passerelle média est d'assurer l'interopérabilité d'une part entre les différents réseaux, par exemple entre le réseau IP et le RTC ou entre le réseau IP et le réseau mobile et d'autres part entre les différents protocoles de signalisation comme SIP et H.323 [5]. Elle assure surtout la conversion de signaux entre le réseau à commutation de paquets et le réseau à commutation de circuits. Dans le cas du protocole de signalisation MGCP [6], elle offre des informations de routage concernant les proxys SIP pour que les clients trouvent leurs chemins via les différents réseaux interconnectés.

- **L'agent de présence :** il informe le client inscrit à son service sur l'état de présence des autres clients. La source de cette information est l'agent client qui s'inscrit et envoie périodiquement son état de présence. Cet état peut être soit disponible, occupé ou absent ce qui rend la communication plus confortable pour l'appelé ou l'appelant.

2.2.2 Protocole de signalisation SIP

Le protocole d'initialisation de session (SIP) est un protocole qui permet aux différents équipements du réseau de communiquer entre eux pour réaliser les différents services de la voix sur IP. Il a été développé par l'IETF sous la référence RFC 2543, et suite aux modifications introduites par le même organisme, nous arrivons à la RFC 3261 [2]. Le protocole SIP est conçu principalement pour que les parties communicantes soient capables d'établir l'appel, communiquer et négocier les paramètres de session VoIP. SIP est un protocole général et non limité à la transmission de la voix. Il est également utilisé dans d'autres services tels que les appels vidéo, les jeux collectifs en ligne, la messagerie instantanée. SIP est un candidat pour toute application qui nécessite une gestion de sessions multiples. Il est conçu afin d'être indépendant du protocole de transport et est considéré comme une application de gestion de sessions VoIP.

L'adressage SIP se base sur l'utilisation d'URI (*Uniform Resource Identifier*) qui définit principalement les utilisateurs, les équipements et les services. L'URI est similaire à une adresse email. SIP prend en charge la correspondance des noms et offre un service de redirection, ce qui permet la mobilité des utilisateurs [2]. Ainsi, les utilisateurs SIP peuvent conserver un identifiant unique visible de l'extérieur quel que soit leur emplacement sur le réseau. La forme de l'URI est *sip :user@host :port ;parameters*. Une adresse SIP fait référence à une entité physique mais aussi à une entité virtuelle, se distinguent ainsi d'autres protocoles tel que HTTP [7].

5. H.323, Recommendation H.323, www.itu.int/rec/T-REC-H.323/
6. http ://tools.ietf.org/html/rfc3435
7. Hypertext Transfer Protocol

Requête	Rôle
INVITE	invitation d'un client à une session ou pour modification de celle-ci
REGISTER	enregistrement auprès d'un proxy SIP et donc dans un domaine SIP
ACK	acquittement d'une requête
BYE	terminaison d'une session ou une transaction
REFER	transfert d'appel
OPTIONS	test des capacités ou la disponibilité d'un serveur SIP
UPDATE	mise à jour d'une session établie

TABLE 2.1 – Les principales requêtes du protocole SIP

Réponse	Type	Description
1xx	Provisoire	informer l'appelant que l'appelé a reçu l'appel et qu'il est en cours de traitement
2xx	Succès	informer que la demande de session est acceptée
3xx	Redirection	indiquer que les messages empruntent un nouveau chemin
5xx, 6xx	Erreur	annoncer une panne de serveur ou une panne globale

TABLE 2.2 – Les principales familles de réponses SIP

Les principaux messages du protocole

Pour établir une session entre deux clients SIP, une séquence de messages est échangée entre les différentes entités responsables de la communication. Nous trouvons principalement deux catégories de messages dans cette séquence : les requêtes et les réponses aux requêtes. Dans SIP, la requête est envoyée d'une entité cliente vers une entité serveur et vice-versa pour les réponses. La requête doit contenir tous les éléments nécessaires pour la session à savoir le type du média et son encodage. Ensuite, l'appelé doit répondre à l'offre soit par l'acceptation afin d'indiquer qu'il supporte les exigences spécifiées, soit par le refus. Les principales requêtes, comme INVITE, REGISTER et BYE ainsi que leurs rôles sont indiquées dans le tableau 2.1. Les principales familles de réponses sont indiquées dans le tableau 2.2. Il existe aussi plusieurs champs dans le message SIP qui servent par exemple à indiquer l'adresse de l'appelé, de l'appelant, l'adresse du contact et l'identifiant du session. Ces champs sont synthétisés dans le tableau 2.3.

VIA	déterminer le protocole du transport utilisé et où la réponse doit être envoyée
FROM	indiquer l'appelant ou la source de la requête sous forme d'une adresse URI SIP
TO	déterminer la destination de la requête ou de l'appelé
REQUEST-URI	fournir à l'adresse du contact
CALL-ID	indiquer l'identifiant unique de la session, il est le même pour tous les messages d'une session

TABLE 2.3 – Les principaux champs d'un message SIP

Le trapézoïde SIP

La figure 2.2 représente le trapézoïde caractéristique de la communication entre deux clients SIP dans deux domaines différents. Avant de commencer à échanger le média (flux vidéo ou audio), l'appelant *ouss@domain1.com* doit trouver l'appelé *alice@domain2.com*, ce rôle est dévolu au proxy SIP du domaine 1. Ce dernier cherche auprès serveur de localisation l'adresse de contact correspondante. S'il ne la trouve pas, il transfère la requête au proxy SIP du domaine 2. Ce serveur proxy prend en charge l'acheminement du trafic SIP jusqu'à la destination (représenté par l'arc du trafic passant par les deux serveurs proxys dans les deux domaines). Les deux téléphones doivent se mettre d'accord pour établir une session SIP de voix et surtout vérifier si les deux équipements VoIP supportent le même encodage média. Enfin, les deux clients peuvent communiquer directement en média sans passer par les proxys SIP pour transférer le média. Cette décomposition entre le trafic SIP et le trafic RTP est une caractéristique fondamentale dans la communication de la voix sur IP qui se fait peu importe la localisation physique des serveurs SIP.

Ce protocole assure plusieurs fonctions comme :

- **L'emplacement de l'utilisateur :** l'URI SIP est indépendante de l'emplacement physique de l'utilisateur, de l'équipement et du service. Pour cela, SIP permet aux clients de se déplacer librement et garantit la correspondance entre l'adresse logique et l'adresse de contact.
- **La disponibilité de l'utilisateur :** SIP exige l'acceptation des parties avant d'établir la communication effective. Nous trouvons aussi dans les terminaux une option de mise à jour de statut de disponibilité.
- **La capacité de l'utilisateur :** SIP détermine le type de média communiqué comme la voix, la vidéo ou le texte et les paramètres de ce média.
- **La gestion de sessions :** SIP prend en charge toute opération de gestion en relation avec la session multimédia établie à savoir la négociation de ses différents paramètres, leurs modifications et la terminaison de la session.

Scénario d'enregistrement SIP

La première étape à réaliser dans un réseau VoIP pour établir une session est d'accéder à ce réseau. Le client SIP doit s'enregistrer dans son domaine VoIP auprès d'un serveur d'enregistrement (voir figure 2.3). Il envoie un message REGISTER contenant son adresse SIP URI et son adresse de contact au registrar du domaine 1. Ce dernier enregistre cette correspondance

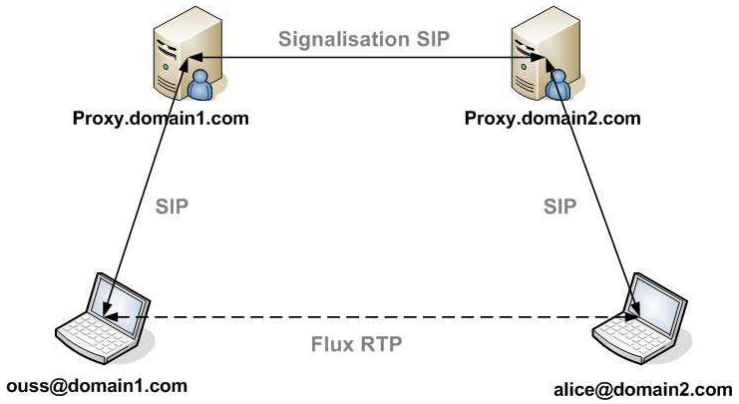


FIGURE 2.2 – Trapézoïde SIP

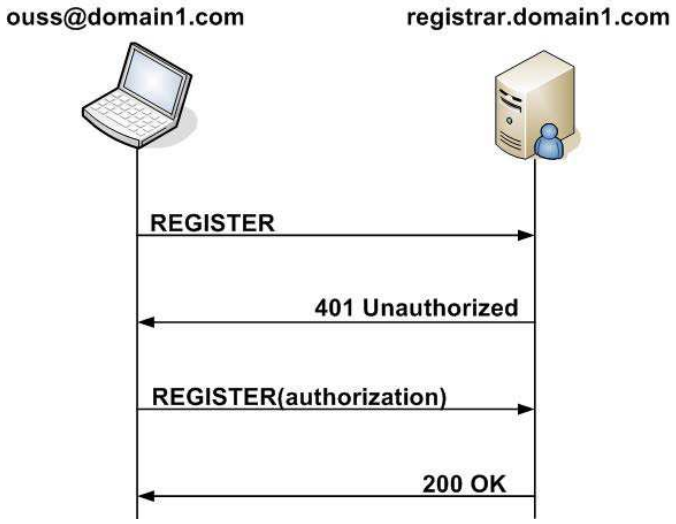


FIGURE 2.3 – Scénario d'enregistrement SIP

dans un serveur de localisation. Ainsi, le serveur proxy est capable d'accéder à la base de données et de router les appels entrants vers le client SIP. Le message REGISTER est envoyé avec les champs *To* et *From* contenant l'adresse SIP URI de l'utilisateur et le champ CONTACT contient son adresse de contact. Ce champ peut suggérer une période de temps pour l'expiration de cette correspondance. Ainsi, chaque nœud rafraichit son entrée dans le serveur de localisation par l'envoi périodique de messages REGISTER auprès du registrar. Le serveur SIP peut exiger une authentification pour chaque message d'enregistrement, il envoie un message de type *401 Unauthorized*, le client SIP doit répondre par un deuxième message REGISTER dont le champ *Authorization* contient le nom d'utilisateur, le mot de passe.

Scénario d'établissement de sessions

L'établissement d'une session VoIP se fait entre deux agents client et plus précisément entre l'UAC du premier client SIP et l'UAS du deuxième client. Dans notre exemple, *ouss* commence l'établissement de session par l'envoi d'un message SIP INVITE. Le champ *From* dans ce message contient son adresse SIP URI, le champ *To* contient l'adresse logique du destinataire et le champ *Via* contient l'adresse du contact du serveur proxy SIP du domaine VoIP (voir figure 2.4). Après la réception, *alice* répond à l'appelant par deux messages de type 1xx, en premier le message *100 Trying* afin de signaler que la requête est bien reçue. Après, il envoie le message *180 Ringing* qui signale que le téléphone sonne. Si l'appelé décroche, son UAS envoie à l'appelant le message 200 OK. L'appelant répond par le message *ACK* pour l'acquittement. Ainsi, la session VoIP est établie et les deux parties communicantes peuvent échanger des flux encapsulés dans des messages RTP.

2.2.3 Autres protocoles

Dans une architecture voix sur IP, nous trouvons principalement le protocole de signalisation SIP. Mais, nous avons aussi besoin d'autres protocoles pour le transfert média comme les protocoles RTP et RTCP, pour la négociation du session commele protocole SDP.

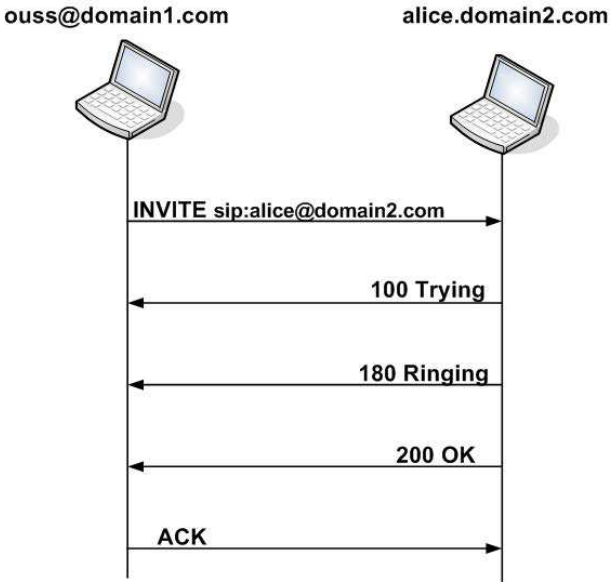


FIGURE 2.4 – Un scénario d'établissement d'une session SIP

Protocole de transfert média RTP

Le protocole RTP est défini par les RFCs 3550 et 3551 [8], il fonctionne au dessus du protocole UDP. Ce protocole de transport est le plus approprié pour gérer le flux des applications en temps réel parce qu'il est plus adéquat que les protocoles TCP. RTP se caractérise par un temps de transmission plus court qui permet de respecter la contrainte temps des applications interactives.

RTP ne garantit pas la qualité de service, il est simple et possède la même stratégie qu'UDP à savoir le best effort. Durant la conception de ce protocole, l'interactivité est considérée comme un critère plus important que la fiabilité, ce qui explique l'existence du champ *timestamp* qui définit l'instant de l'envoi des messages, du champ *codec* qui définit la qualité du flux de données et les algorithmes de correction de perte. Ainsi, le destinataire peut connaître l'algorithme de compression utilisé et est capable de minimiser la perte de données. Le protocole RTP permet de distinguer les flux média suivant leur source, d'indiquer le type de média transmis et de synchroniser le trafic grâce à l'estampillage temporel. Il permet aussi d'estimer des paramètres de qualité de service, de détecter les paquets perdus et de renvoyer à l'application des paquets ordonnés grâce au numéro de séquence. La version sécurisé d'RTP est le protocole SRTP. Il est également possible d'utiliser IPsec pour sécuriser l'échange en RTP et éviter l'écoute du trafic.

Protocole de contrôle du média RTCP

Ce protocole permet aux parties participantes dans une session d'échanger des rapports et des statistiques sur la qualité de service de la communication. Les informations échangées via RTCP n'aident pas nécessairement à trouver la source du problème mais permettent de rassembler des statistiques sur le trafic. Il offre des informations sur la gigue, le délai de latence et la perte de paquets. Ces informations sont utiles par exemple pour l'encodage adaptatif des médias. Les messages RTCP sont généralement collectés par les passerelles. Nous distinguons plusieurs types de rapports RTCP à savoir le rapport du récepteur (*DR rapport*) et le rapport de l'émetteur (*SR rapport*). Le rapport de l'émetteur est envoyé périodiquement par les appelés actifs pour

8. RTP, Real Time Protocol, www.ietf.org/rfc/rfc3550.txt

décrire la transmission et la réception des données RTP. Il contient une estampille absolue que le récepteur utilise pour synchroniser les messages RTP ce qui est surtout important dans le cas des transmissions audio et vidéo qui utilisent des repères temps différents. Le rapport du récepteur est envoyé par les appelants les moins actifs, il permet de quantifier la qualité de service de la conversation. Le volume des messages RTCP ne doit pas dépasser les 5% du volume global des messages échangés pour ne pas surcharger le trafic.

Protocole de description de session SDP

Le protocole SDP est défini par l'IETF dans le RFC 2327 et mis à jour dans le RFC 3266 pour supporter le protocole IPv6. Il sert à décrire les paramètres d'une session et les propriétés désirées entre les clients SIP. C'est un protocole similaire à HTTP. Les propriétés sont présentées dans le message sous forme de $champs = parametre_1, parametre_2, \ldots parametre_n$. Il permet aussi aux clients d'une session de partager les informations et les propriétés avec les clients qui veulent participer. Les principaux champs de SDP comme le nom de la session et le type de média du SDP sont décrits dans le tableau 2.4.

s	le nom de la session
i	la description de la session
k	la clé de cryptage envoyé en clair
m	le type de média
t	l'instant de début et de fin de la session

TABLE 2.4 – Les principaux champs du protocole SDP

Le protocole SDP est utilisé durant la phase de négociation des propriétés de la session et avant son établissement. En effet, l'appelé inclut les propriétés désirées dans la première requête de signalisation INVITE. Du côté de l'appelant, il renvoie les paramètres désirés pour la session dans le message d'acquittement ACK.

2.3 Architecture VoIP pair-à-pair

Dans cette architecture les pairs coopèrent pour fournir des services VoIP. Ils communiquent directement entre eux pour effectuer certaines activités. Le réseau pair à pair permet une plus grande fiabilité et un meilleur passage à l'échelle. L'architecture P2PSIP a été proposé par [64] pour permettre la mise en œuvre du protocole SIP dans les réseaux pair à pair.

D'après la description de P2PSIP fournies dans [1], deux modèles distincts sont proposées pour ce déploiement :

- **Le modèle *P2P-over-SIP*** : dans ce modèle, le P2PSIP utilise le protocole SIP pour la maintenance du réseau P2P. Ainsi, le protocole SIP est responsable de plusieurs tâches comme, d'une part, le maintien de l'information de localisation, l'enregistrement des clients SIP et l'établissement des sessions et d'autre part le maintien du réseau P2P. Ainsi, différents drafts [101] proposent des extensions au protocole SIP pour qu'il supporte les nouvelles fonctions de maintenance du réseau P2P. En raison de l'utilisation intensive de messages SIP REGISTER et la non flexibilité de cette architecture à d'autres services P2P, cette approche n'a pas été déployée.
- **Le modèle *SIP-using-P2P*** : plutôt que de surcharger le protocole SIP avec nouvelles fonctionnalités, le modèle *SIP-using-P2P* propose d'exploiter deux piles protocolaires pour

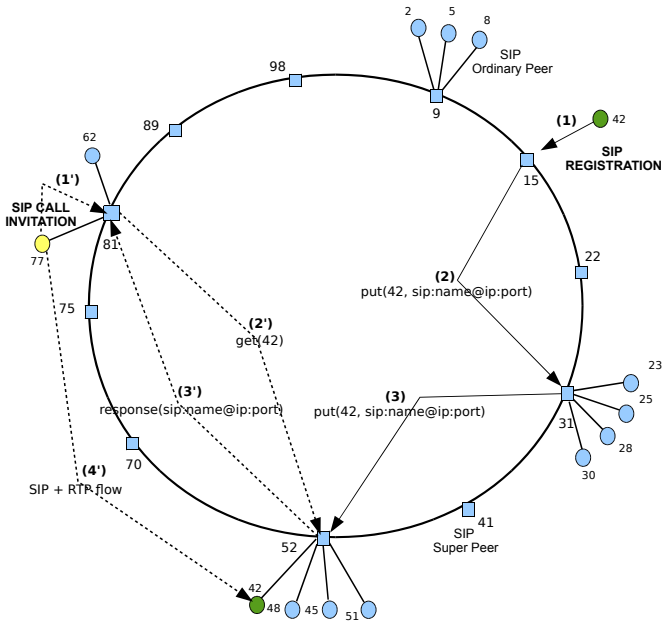


FIGURE 2.5 – Architecture liée à P2PSIP

le fonctionnement du réseau P2PSIP à savoir le protocole SIP et les protocoles P2P. Dans ce modèle, le protocole SIP est responsable de la recherche des ressources, de l'enregistrement des clients et de l'établissement de sessions tandis que l'architecture P2P est responsable de la maintenance du réseau. Ainsi, nous évitons toute surcharge et complexité d'extension du protocole SIP. Dans cette approche, SIP est considéré comme une application, d'où la possibilité de coexistence avec les autres applications sans changer les protocoles de transfert ou l'architecture du réseau P2P.

Nous nous sommes focalisés dans nos travaux sur l'architecture particulière P2PSIP définie par [100]. Les auteurs proposent une architecture basée sur l'approche du *SIP-using-P2P*, ils adoptent un réseau P2P structuré (voir figure 2.5). Pour cela, les auteurs utilisent une table de hachage distribuée DHT (*Distributed Hash Table*) afin de concevoir un réseau P2P structuré. Cette table permet de récupérer des informations sur le réseau P2PSIP. Chaque information stockée dans la table DHT est sous la forme d'une association contenant la clé et la valeur de l'information (<clé, valeur>). La responsabilité de la maintenance des informations est partagée par les nœuds et tout dysfonctionnement dans le réseau cause un minimum de problèmes. Dans le cas du réseau P2PSIP, la DHT maintient l'information de localisation des utilisateurs SIP afin qu'ils puissent solliciter l'établissement d'une session SIP.

Pour concevoir une architecture P2PSIP, trois types de table DHT ont été proposés pour les réseaux P2P :

- **DHT à base de serveurs** : les serveurs implémentent la DHT et maintiennent l'information de localisation des utilisateurs. Le client SIP se connecte auprès des serveurs afin de récupérer la correspondance entre les deux adresses logiques et physiques. Cette architecture est encore une architecture centralisée de type client serveur, elle ne respecte pas les caractéristiques des réseaux P2PSIP.
- **DHT distribuée entre les pairs** : cette forme suppose que les pairs ont la même capacité

et disponibilité. Chaque nœud héberge une fraction de la DHT.

- **Un modèle intermédiaire** : c'est un modèle intermédiaire entre les deux formes précédentes. Il distingue deux types de nœuds différents : des nœuds dont la capacité et la disponibilité sont limitées et des nœuds avec une grande capacité en termes de mémoire et bande passante. Dans ce cas, les super nœuds forment la DHT et maintiennent l'information de localisation et les autres nœuds se connectent auprès d'eux pour solliciter l'établissement d'une session. Ce modèle est dynamique parce que les super nœuds peuvent redevenir des nœuds réguliers et vice-versa.

L'architecture defénie dans [101] se base sur le modèle intermédiaire, elle définit ainsi deux types du nœuds dans le réseau. Les nœuds avec une grande capacité et disponibilité représentent les super pairs, alors que les autres représentent les pairs ordinaires. Dans cette architecture, les super pairs forment la table DHT et les autres représentent les utilisateurs SIP. Nous adoptons cette architecture pour l'application de notre stratégie de gestion des risques.

En comparant avec l'architecture fonctionnelle SIP, nous trouvons plusieurs similitudes :

- **La table DHT** : cette table joue le rôle de serveur de localisation. En effet, la table contient les correspondances entre les adresses logiques (SIP URIs) et l'adresse physique du contact ce qui permet aux différents utilisateurs de localiser les clients SIP. En plus, elle représente le serveur d'enregistrement (registrar) : un client SIP qui veut s'enregistrer auprès du réseau P2PSIP envoie un couple d'adresse logique et physique à la DHT qui prend en charge l'enregistrement.
- **Le super pair** : ce pair joue le rôle de serveur proxy SIP et de serveur de redirection. Il prend en charge l'établissement de la session entre les deux parties communicantes, il prend la décision du routage des messages et il communique avec les autres super pairs qui maintiennent la DHT pour localiser l'appelé. Ce pair communique via deux piles protocolaires à savoir les protocoles P2P et le protocole SIP.
- **Le pair ordinaire** : c'est le nœud dont la capacité et la disponibilité sont limitées, il représente l'utilisateur SIP. Il communique en SIP avec le super pair pour solliciter des services comme la recherche de localisation d'un appelé, l'établissement d'une session et la redirection d'appels.

2.3.1 Scénario d'enregistrement

Pour qu'un client s'enregistre auprès du réseau P2PSIP, il doit découvrir des super pairs du réseau pour les solliciter. Il existe plusieurs techniques pour découvrir la topologie du réseau. La plus simple consiste à envoyer un message dont la durée d'expiration est limitée (TTL [9]) pour découvrir les pairs voisins et avoir plus d'informations sur les super pairs. Une deuxième technique consiste à utiliser le protocole SLP [10] pour localiser les super pairs. Une autre alternative est de solliciter le serveur DNS pour localiser des pairs de lancement (*bootstrap*) pré-configurés.

Après cette étape de découverte, le client envoie un message SIP REGISTER contenant son adresse SIP URI dans les deux champs TO et FROM et met dans le champs de REQUEST-URI l'adresse physique du super pair sollicité. Ce dernier calcule la clé en utilisant un algorithme de hachage. Ensuite, il envoie un message P2P PUT avec la clé et l'adresse physique (<clé, adresse IP>) du client SIP. Cet enregistrement sera placé dans la table DHT en exécutant l'algorithme associé. Dans le cas où le client est un super pair, il envoie directement un message P2P PUT avec la clé et l'adresse physique. Les clients SIP envoient périodiquement des messages SIP

9. Time To Live
10. SLP, Service Location Protocol, IETF Internet Draft, www.ietf.org/rfc/rfc2608.txt

REGISTER pour rafraîchir leurs enregistrements dans la table DHT. Ils peuvent envoyer un message SIP OPTIONS pour vérifier la disponibilité des super pairs.

2.3.2 Scénario d'établissement d'une session

Pour établir une session dans le réseau P2PSIP, le client envoie un message SIP INVITE vers le super pair auquel il est attaché. Dans l'en-tête, le champs FROM contient l'adresse SIP URI de l'appelé et REQUEST-URI contient l'adresse physique du super-pair. Dans ce cas, le super pair prend en charge la localisation de l'appelé. Il calcule la clé correspondante à l'adresse SIP URI de l'appelé et envoie ensuite une requête P2P FIND à la table DHT afin de trouver l'enregistrement correspondant. Le super pair responsable de cette information, répond par une requête P2P RESPOND avec la valeur de l'enregistrement (l'adresse physique). Enfin le super pair redirige l'information de localisation vers le client SIP. Ce dernier peut contacter directement son correspondant par la redirection de requête SIP INVITE. Si l'appelé accepte de communiquer en P2PSIP, il répond par une requête de 100 TRYING, 180 RINGING et 200 OK. Ainsi, la session SIP est établie et les deux parties communicantes peuvent échanger des messages RTP. À la fin de la communication, un des deux clients SIP envoie une requête SIP BYE et le deuxième répond par un 200 OK.

2.4 Attaques de sécurité

Toute technologie, y compris la ToIP, a des problèmes de sécurité qui visent ses protocoles de communication ou ses services. La réalité est que, avec une motivation suffisante, tout problème de sécurité peut être exposé et exploité. En se basant sur l'objectif d'une personne malveillante, les attaques VoIP peuvent être classées en trois catégories : les attaques contre la signalisation, les attaques contre le transfert média et les attaques contre les services support. Dans cette partie, nous détaillons les principales attaques de sécurité contre les protocoles fondamentaux d'une infrastructure VoIP (voir figure 2.6).

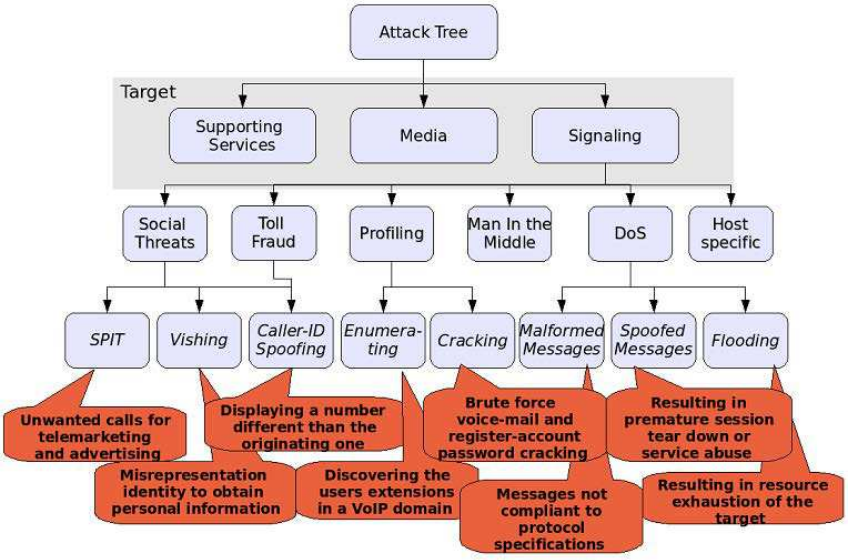


FIGURE 2.6 – Classification des attaques par protocole cible [67]

2.4.1 Attaques contre les protocoles de signalisation

La signalisation est la phase la plus importante de toute interaction de télécommunication. Elle est la cible principale d'un large éventail d'attaques de sécurité. L'objectif de l'attaque consiste à empêcher les utilisateurs légitimes d'exploiter l'équipement ou l'infrastructure attaqué.

Attaques SPIT

SPIT est une attaque centrée sur des communications indésirables. Elle consiste à appeler des utilisateurs VoIP pour des fins publicitaires et de harcèlement. Il s'agit généralement d'appels pré-enregistrés et automatiquement composés. Les infrastructures voix sur IP, comme les applications de messagerie Internet, sont susceptibles d'être abusées par des acteurs malveillants qui initient des communications non sollicitées et non désirées [85]. Par exemple, les télévendeurs et d'autres agresseurs du système téléphonique ciblent les systèmes VoIP de plus en plus, surtout que la VoIP est une application moins coûteuse à l'usage que la téléphonie classique. L'identification et le blocage des appels SPIT sont difficiles en raison de leur nature sociale. La classification d'appelants dans des listes blanches et noires ne résout pas le problème de façon définitive comme dans le cas d'un nouveau appelant avec plusieurs identités SIP. La difficulté pour contrer ce type de spam, par rapport au spam classique, repose sur le fait que le contenu de l'appel n'est pas connu à l'avance, ce qui rend le blocage d'appels illégitimes avant l'établissement de la session difficile.

Attaques de déni de service (DoS)

Cette attaque a pour objectif de réaliser des dégâts (arrêt temporaire, dysfonctionnement des appels) aux réseaux VoIP. Nous trouvons trois catégories d'attaques par inondation de messages SIP [116]. La première catégorie consiste en l'inondation par des messages SIP corrects, il s'agit d'envoyer un nombre important de messages SIP comme SIP INVITE pour réaliser un déni de services contre un terminal ou un proxy SIP. La deuxième catégorie consiste en l'inondation par des messages malformés pour faire tomber des serveurs et la troisième consiste à envoyer des messages malformés exploitant des vulnérabilités bien spécifiques. Nous trouvons aussi l'envoi massif de messages SIP INVITE avec un champ REQUEST URI valide. Dans ce cas de figure, l'attaquant appelle un téléphone SIP enregistré auprès du proxy SIP qui le surcharge par des messages SIP INVITE. Par conséquent, le serveur proxy va réserver des ressources de mémoire pour ces appels ce qui génère un débordement de mémoire auprès du serveur.

Attaques par messages malformés

L'attaque par messages malformés consiste à construire et envoyer des messages SIP malformés d'une façon aléatoire ou semi aléatoire. Cette technique de génération de trafic s'appelle le *fuzzing* [12]. Bien que le proxy SIP possède la capacité de corriger des requêtes SIP et de modifier les différents champs, ceci n'arrête pas les attaques dues aux messages malformés. Elles peuvent engendrer l'arrêt brutal des service VoIP après avoir causé des dégâts comme l'effondrement du système par une attaque du débordement de mémoire (*buffer overflow*) par exemple. L'attaquant peut aussi obtenir un accès non autorisé au système et même injecter un code pour l'exécuter à distance.

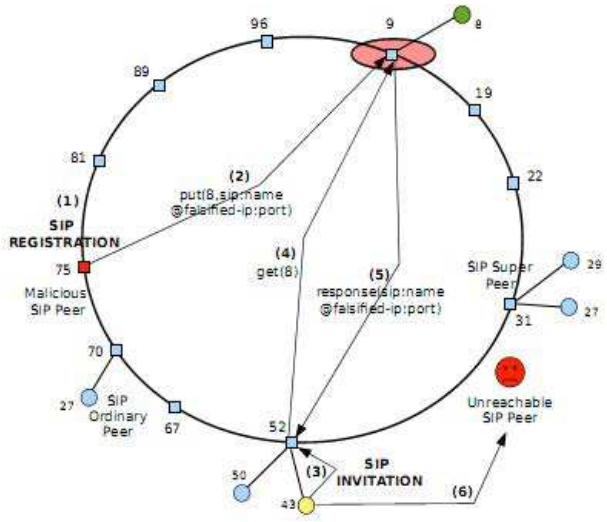


FIGURE 2.7 – Exemple d'une attaque d'usurpation d'identité en P2PSIP

Attaque de l'homme du milieu

Dans le cas où il n'y a pas des mécanismes d'authentification forte, de nombreuses situations permettent à un attaquant de se positionner entre un appelant et l'appelé. SIP met en œuvre un schéma d'authentification similaire à HTTP. Tous les serveurs SIP comme le proxy et le serveur de redirection peuvent authentifier un client SIP UAC par un défi cryptographique basé sur le secret partagé (généralement le mot de passe stocké dans le serveur) [94]. Cette authentification est généralement faite dans un seul sens, du client vers le serveur. L'attaque de l'homme du milieu peut être établie par le scénario décrit sur la figure 2.8. Le client SIP *ouss* envoie un message INVITE à son serveur proxy SIP, l'attaquant *charlie* intercepte le message et envoie une réponse de redirection forgée à *ouss* menant vers son adresse physique. Comme il n'a pas authentifié le serveur proxy, le client SIP accepte la réponse et redirige l'appel vers l'attaquant. Le vrai proxy SIP peut être neutralisé par un déni de service ou en exploitant une situation de concurrence. En même temps, l'attaquant remplace l'emplacement de l'émetteur par son adresse IP dans le champ *Contact* (ce champ indique l'adresse physique de l'appelé). L'attaquant peut aussi exploiter les champs *via* et REQUEST-URI dans les messages SIP afin de modifier l'itinéraire des requêtes SIP. Ensuite, il envoie le message falsifié vers l'appelé *alice*. Ainsi, tous les messages de signalisation communiqués entre l'appelé et l'appelant passeront dorénavant par l'attaquant. Le vol de la session de signalisation entre les deux clients est une première étape vers le vol de la session d'échange de média.

Attaque d'usurpation d'identité

Elle s'appuie sur le vol d'identité et du mot de passe d'un client SIP par l'écoute du trafic SIP ou sur l'acquisition via une autre voie du couple (nom d'utilisateur et mot de passe) et les utilise pour avoir un accès non autorisé [103]. Le vol d'identité peut se faire par une attaque de type *man-in-the-middle* ; l'utilisateur SIP entre ses coordonnées d'authentification et l'attaquant suit l'échange du trafic entre le client et le serveur et récupère les coordonnées d'un utilisateur SIP pour les utiliser ultérieurement. Sur la figure 2.7, nous présentons une attaque d'usurpation d'identité dans le cas du réseau P2PSIP. L'attaquant utilise le nom d'utilisateur de la victime pour modifier la correspondance entre l'adresse logique (de la victime) et son adresse physique.

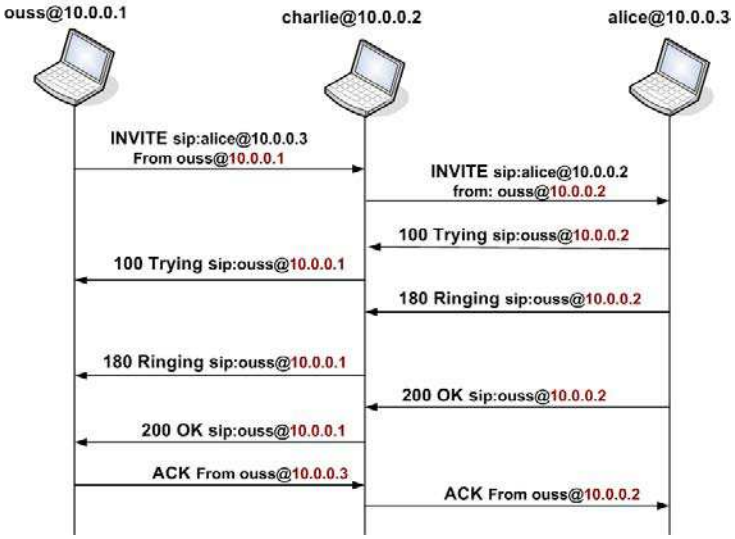


FIGURE 2.8 – Attaque de l'homme du milieu

Un pair malveillant envoie un message P2P PUT avec une nouvelle adresse physique. Ainsi, quand un utilisateur P2PSIP cherche à contacter la victime, il va chercher son entrée dans la DHT et tombe sur une adresse physique inexistante. Cette attaque simple permet de rendre les pairs victimes injoignables.

Détournement d'un enregistrement

Le processus d'enregistrement dans le réseau VoIP SIP utilise le protocole UDP pour transporter les messages ce qui simplifie la génération des messages usurpés. Le serveur d'enregistrement n'applique pas de règles fortes pour protéger les clients enregistrés. Quand l'authentification est imposée, elle peut inclure un identifiant MD5 pour le nom d'utilisateur, un mot de passe et une estampille temporelle. Cette attaque a pour but le détournement des appels vers l'attaquant ou vers une destination SIP inexistante (voir figure 2.7). L'attaquant réalise cette attaque par étapes, la première consiste à obtenir un carnet d'adresses enregistrées auprès du serveur registrar, il peut l'avoir en scannant le réseau et en cherchant des adresses SIP URI. Il peut aussi envoyer des messages SIP OPTIONS et SIP INVITE pour vérifier si une adresse SIP est valide ou pas. La deuxième étape consiste à s'authentifier auprès du serveur, cette opération est faite généralement soit par une attaque de fraude ou par une attaque de dictionnaire [81].

2.4.2 Attaques contre les protocoles médias

Dans cette section, nous nous intéressons aux attaques de protocoles médias RTP et RTCP. Les attaques présentées sont le DoS, l'écoute du média et les attaques de manipulation du trafic.

Écoute et analyse du trafic

Ceci regroupe l'ensemble des tentatives de collecte des informations sensibles concernant le média pour préparer une attaque plus développée ou pour avoir plus de connaissance sur les parties communicantes ou sur l'entreprise [3]. Cette attaque donne la possibilité à l'attaquant d'avoir le contrôle sur l'échange de messages RTP non protégés entre clients VoIP. Cette catégorie

d'attaques inclut l'analyse du trafic qui peut être active ou passive comme la collecte, l'analyse et le décodage des messages RTP. Cette attaque a pour objectif d'extraire des informations textuelles ou verbales comme les numéros de carte de crédit ou d'analyser les communications pour établir des modèles de communications que l'attaquant peut utiliser ultérieurement.

Injection de trafic malveillant

La qualité de service est essentielle dans le service VoIP. Si un attaquant réussit à dégrader la qualité de service comme la disponibilité ou la gigue, la communication SIP devient incompréhensible. Par conséquent, il suffit que l'attaquant encombre un proxy SIP, un routeur ou une passerelle média par des messages RTP malveillants pour qu'il dégrade la qualité des services VoIP [81]. L'attaquant peut aussi injecter des messages RTP (des messages vocaux) pour qu'ils perturbent les parties communicantes. Cette attaque se fait par étape : initialement, l'attaquant doit savoir qu'il y a une session établie entre deux clients SIP. Ensuite, il prépare des messages RTP usurpés par l'identité d'un des deux clients, et il les envoie massivement vers l'autre. Comme la signalisation et l'échange des messages média sont séparés par la définition du trapézoïde SIP, le protocole de signalisation n'a aucun contrôle sur la qualité de service du transfert du média et sur le chemin pris par les messages.

Attaques par manipulation de messages RTCP

L'attaquant peut perturber le fonctionnement du protocole RTCP. Ce protocole est responsable des rapports et des statistiques concernant le trafic RTP. En écoutant et analysant le trafic, l'attaquant intercepte les rapports RTCP comme le rapport de l'appelant et le rapport de l'appelé. Ensuite, il modifie le contenu et les paramètres dans ces rapports pour envoyer des fausses informations sur les échanges RTP. Par exemple, l'attaquant peut envoyer des rapports indiquant qu'il n'y a pas de perte de messages durant la conversation audio, ce qui mène à des pertes de messages plus grandes et par conséquent rend la communication incompréhensible.

2.4.3 Attaques contre les services support

Les services support jouent un rôle important dans la continuité opérationnelle et le fonctionnement des services VoIP. Nous y trouvons des services comme le DNS qui permet au serveur proxy de résoudre les noms de domaine, le service DHCP qui affecte les adresses de contact au client et d'autres services comme le transfert des fichiers pour la configuration par exemple. Parmi les services support critiques, nous trouvons le système de facturation qui représente, s'il a des dysfonctionnements, une menace financière pour le fournisseur. Nous détaillons dans cette section les plus importantes attaques contre ses services.

Attaques contre le système de paiement

La facturation est un service fondamental pour tous les services VoIP commerciaux et il a un impact direct sur chaque client VoIP. L'une des exigences les plus importantes de facturation est qu'elle doit être sécurisée et qu'elle représente un service fiable [113]. Du côté client VoIP, la facturation ne doit que lui faire payer les appels qu'il a vraiment passé et pour la durée consommée. Les systèmes de facturation existants sont basés sur la signalisation VoIP. Dans le cas du protocole SIP, elle commence à partir de la réception du message SIP 200 OK ; ce message valide que l'appelé a accepté l'appel. Ainsi, toute vulnérabilité dans le système de signalisation est une vulnérabilité potentielle de la facturation VoIP. Un attaquant peut intercepter l'établissement

de session entre un client SIP et le serveur proxy. Quand le client envoie ses coordonnées à savoir le nom d'utilisateur et le mot de passe, l'attaquant les récupère et il les utilise après la fin de session du client légitime. Dans ce cas, l'attaquant est capable de réaliser une attaque d'usurpation d'identité pour se connecter au serveur proxy SIP et il peut demander l'établissement d'une session. Cette communication est facturée au client SIP légitime.

Attaque de l'accès non autorisé

Cette attaque consiste à accéder à un service, une fonctionnalité, ou un élément de réseau sans autorisation appropriée [113]. Les attaques de cette catégorie peuvent être utilisées pour soutenir d'autres attaques, y compris des attaques de déni de service, la fraude et même l'usurpation d'identité parce que l'attaquant peut prendre le contrôle d'un équipement, d'une ressource, ou de l'accès à un réseau. La différence entre l'accès non autorisé et le masquage, c'est que l'attaquant dans le premier cas a le contrôle d'une ressource ou l'accès au réseau en exploitant une vulnérabilité comme le dépassement de mémoire, la configuration par défaut ou la signalisation non sécurisée. Par exemple, un attaquant qui contrôle un proxy SIP où il a un accès administratif peut interrompre le service de signalisation par la suppression des fichiers systèmes. Un autre exemple est celui dans lequel l'attaquant injecte un programme malveillant dans une passerelle média pour collecter les informations média qui transitent la passerelle.

Ingénierie sociale

L'ingénierie sociale est la capacité d'abuser ou de profiter des services VoIP pour un gain personnel ou financier [51]. Cette catégorie d'attaques est l'une des plus critique pour les opérateurs de télécommunications et les fournisseurs de VoIP. L'ingénierie sociale est une préoccupation importante en matière de sécurité et de confidentialité. Différentes attaques sont comprises dans cette classe comme la déclaration des informations fausses délivrées expressément dans le but de tromper ou utiliser une fausse déclaration. Nous trouvons aussi la présentation préméditée d'une fausse identité montrant des informations de quelqu'un d'autre afin de contourner les mécanismes d'authentification. Cette catégorie inclut le vol de services qui consiste à gagner illégalement des revenus provenant des services de quelqu'un d'autre, par exemple la facturation des appels VoIP.

2.5 Synthèse

Les services voix sur IP sont de plus en plus largement utilisés grâce à l'émergence des réseaux à haut débit, à la standardisation des protocoles de signalisation et à l'interopérabilité entre les terminaux. Nous trouvons dans ce contexte le protocole SIP qui est devenu le standard de signalisation pour gérer les sessions VoIP et le protocole RTP utilisé pour le transfert média. Le déploiement des services VoIP engendre l'émergence de nouvelles attaques. Elles sont soit héritées des couches inférieures comme les attaques de déni de service ou spécifiques à la couche applicative comme le SPIT ou l'attaque par inondation de messages SIP. Des mécanismes de sécurité sont disponibles mais peuvent avoir un impact non négligeable sur le fonctionnement et la qualité de service de tels services critiques. Nous défendons la thèse que le processus de gestion des risques apporte de nouvelles perspectives à cet égard. Nous proposons, par la suite, d' appliquer et d'automatiser ce processus afin de sécuriser les services VoIP.

Chapitre 3

De l'évaluation au traitement des risques

Sommaire

3.1 Introduction

La gestion des risques est un processus qui permet d'identifier, d'évaluer et de traiter les risques notamment dans les réseaux et les services. Une gestion des risques efficace est une composante importante d'une stratégie de sécurité réussie. L'objectif principal de ce processus consiste à protéger les services et leur capacité à réaliser leur mission.

Comme les infrastructures VoIP offrent des services interactifs et critiques en termes de délai et de disponibilité, elle doit maintenir une qualité de service des utilisateurs. Ainsi, la gestion des risques s'impose et joue un rôle essentiel dans la protection des équipements et services d'une telle infrastructure. Elle offre plusieurs techniques d'évaluation de risques comme l'évaluation qualitative ou quantitative qui permet de quantifier le niveau actuel de risque auquel l'infrastructure est exposée. Elle propose aussi une variété de stratégies de traitement comme l'évitement ou l'optimisation des risques qui permet d'adapter l'exposition du réseau et des services en fonction de la potentialité de la menace et des coûts induits.

Notre objectif dans ce chapitre consiste à présenter le processus de gestion des risques, puis classer les différents travaux de sécurité relatifs aux infrastructures VoIP en fonction de leur

contribution aux principales étapes de ce processus, et montrer leurs limites vis-à-vis des exigences d'un tel service.

3.2 Concepts de la gestion des risques

Dans cette section, nous présentons la terminologie et le processus de gestion des risques ainsi que les principaux modèles de risques dans le domaine.

3.2.1 Terminologie

Pour définir le risque, il faut définir ses concepts de base. Le risque résulte du fait qu'un actif (élément du service ou de l'infrastructure) possède une vulnérabilité qui peut être exploitée par une menace pour réaliser une attaque et ainsi dégrade ou rendre inopérationnel cet actif [4, 107].

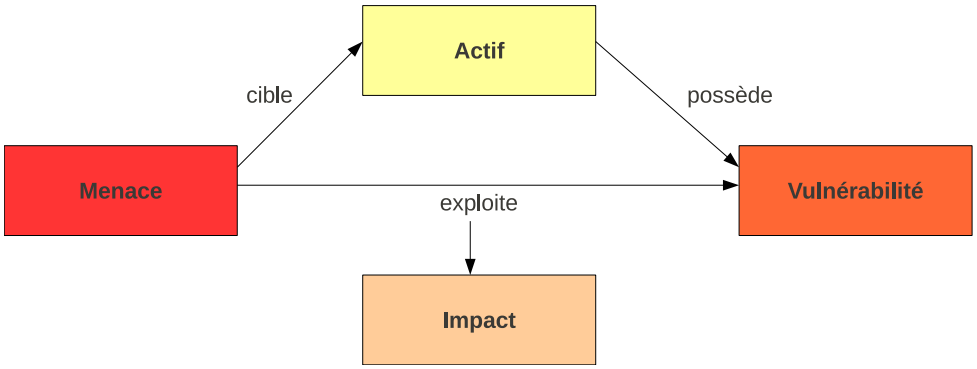


FIGURE 3.1 – Éléments composant le risque

Actif

D'après la norme ISO/IEC 27005 [4], un actif correspond à tout élément qui possède une valeur technique ou un enjeu pour l'infrastructure et ses services. Dans le contexte de la VoIP, la notion d'actif s'applique aux clients enregistrés, la base de données de localisation et l'historique des appels. Les actifs incluent aussi tout ce qui est en relation directe avec les processus de fonctionnement comme l'enregistrement du client, l'établissement de la session, la localisation de l'utilisateur et le routage de l'appel. Ainsi, cette notion rassemble les cibles potentielles d'une attaque VoIP. Il existe aussi des actifs de support qui incluent les logiciels, les serveurs et le réseau. Cette sous-catégorie inclue les systèmes d'exploitation, les piles protocolaires (TCP/IP, SIP), le protocole de transfert RTP et les serveurs (le serveur d'enregistrement, le serveur de localisation et le proxy SIP).

Menace

Une menace correspond à une source d'attaque qui peut engendrer une dégradation des actifs du service concerné. Nous distinguons plusieurs types de menaces :

- Toute action qui conduit à la réalisation d'une attaque. Dans le cas de la VoIP, nous trouvons par exemple l'écoute et l'analyse du trafic par des outils comme *wireshark* et *Nmap*.

- Toute action qui rend possible la réalisation d'une attaque comme le vol des informations personnelles d'un utilisateur ou l'usurpation d'identité.
- Tout effet caractéristique comme la congestion du réseau, un nombre élevé de requêtes SIP INVITE ou plusieurs essais d'enregistrement avec des mots de passe différents.

Une menace peut être évaluée par les conséquences potentiellement subies par l'actif en cas d'attaques réussies. Cette évaluation permet d'établir une priorité au niveau de la sécurité du service.

Vulnérabilité

Une vulnérabilité est une faiblesse qui permet à la menace d'opérer une attaque sur le service. Nous trouvons des vulnérabilités dans les services VoIP comme l'utilisation des mots de passe simples par les clients SIP, une authentification faible pour accéder au proxy SIP ou une communication SIP non cryptée. Un attaquant doit disposer d'un outil ou d'un ensemble de techniques permettant d'exploiter une vulnérabilité comme un dictionnaire de mots de passe par défaut. La gestion des vulnérabilités est la pratique cyclique de l'identification, la classification et la neutralisation des vulnérabilités du système. Cette pratique se réfère généralement à des failles logicielles relatives au service.

Attaque

Une attaque de sécurité est définie comme l'intersection de trois éléments [40] à savoir :

- un ensemble de vulnérabilités exploitables dans le service. Par exemple, une configuration par défaut dans un serveur proxy SIP, des mots de passe simples, la non-authentification dans les deux sens client SIP / proxy SIP,
- un attaquant pouvant accéder à la vulnérabilité, motivé par des raisons financières comme les appels gratuits, par la vengeance pour faire tomber les services VoIP,
- et la capacité de l'attaquant d'exploiter la faille.

Risque

D'après [4, 107], le risque est la possibilité qu'une menace donnée exploite les vulnérabilités d'un actif afin d'engendrer un dysfonctionnement ou une dégradation du service concerné. Nous trouvons plusieurs définitions du risque et en distinguons deux importantes que nous utilisons dans la suite afin de présenter notre approche de gestion des risques pour les services VoIP.

La première définition représente le risque en tant que conjonction d'un actif, d'une menace et d'un ensemble de vulnérabilités exploitées par un attaquant. Cette définition permet de classer les menaces, les actifs et les vulnérabilités par type, ainsi, avoir une classification de risque. Il s'agit d'une définition du risque statique car cette description ne dépend pas du temps et ne permet pas de déterminer un scénario qui mène vers les conséquences du risque. Par exemple, on peut définir le risque de l'écoute et de l'analyse du trafic par l'absence d'une technique de cryptage des messages échangées (la vulnérabilité), les deux clients SIP (l'actif) et la facilité d'utiliser les outils d'analyse (la menace).

La deuxième définition représente le risque en tant que conjonction d'un actif, des conséquences subies et de l'ensemble des circonstances dans lesquelles la menace peut se réaliser. Les circonstances sont généralement l'ensemble des événements qui ont mené vers la réalisation de l'attaque. Il s'agit d'une définition dynamique où le temps est un paramètre primordial. Elle permet aussi d'avoir une liaison dans le temps entre les causes et les conséquences du risque.

Cette définition favorise l'identification de risques par la signature d'attaques car elle décrit l'attaque par un enchaînement ordonné d'événements malveillants. Par exemple, le risque d'avoir une attaque du type *man-in-the-middle* ou une attaque de déroutement d'appels peuvent être décrites par une séquence d'événements ordonnée.

3.2.2 Processus de gestion

Pour déployer la gestion des risques dans les infrastructures VoIP, la première étape consiste à définir la portée du service. Dans cette tâche, les frontières de l'infrastructure sont identifiées, ainsi que les actifs et les données qui constituent le service [59]. Le processus de gestion des risques comprend trois étapes majeures : l'identification, l'évaluation et le traitement des risques. La phase d'identification consiste à identifier les actifs critiques, les menaces potentielles et les vulnérabilités dans le service concerné. Ensuite, l'estimation du risque inclut l'estimation de la potentialité d'attaques et l'évaluation de leurs conséquences. Enfin, le traitement des risques inclut plusieurs stratégies comme la stratégie d'évitement, d'optimisation et d'acceptation.

Identification des risques

L'identification des risques nécessite une bonne compréhension de l'environnement de traitement du service. Pour évaluer les risques, il faut d'abord recueillir des informations liées au service qui sont habituellement classées comme suit :

- Le matériel qui constitue la plateforme comme par exemple les terminaux et les serveurs VoIP, ainsi que son degré de sensibilité et son niveau de criticité,
- Le logiciel qui permet le fonctionnement des services comme le système d'exploitation, les services et les protocoles requis pour la VoIP, ainsi que son degré de sensibilité et son niveau de criticité,
- Les données et l'information comme les données d'enregistrement, les CDRs (informations liées à l'établissement de session) et les statistiques du trafic des appels. Elles comprennent aussi la configuration du système, la connectivité entre les services, l'historique des serveurs et le trafic provenant de l'extérieur.

Ensuite, il est nécessaire d'identifier les menaces potentielles et ses sources. Une source de menaces est définie comme toute circonstance ou événement susceptible de causer des dommages ou des dégradations au service. Les personnes peuvent être des sources de menaces par des actes intentionnels. Une attaque délibérée peut être une tentative malveillante d'accéder sans autorisation au service comme par exemple une attaque par dictionnaire contre un proxy SIP afin de compromettre l'intégrité la disponibilité, ou la confidentialité du système et de ses données.

L'identification des risques dans le service doit également inclure l'analyse des vulnérabilités associées à l'environnement du système. L'objectif de cette tâche est d'établir une liste des vulnérabilités qui peuvent être exploitées par des sources potentielles d'attaques. En analysant l'ensemble des événements qui mènent à une attaque réussie, nous pouvons identifier les vulnérabilités exploitées par l'attaquant. Parmi les vulnérabilités VoIP, nous trouvons par exemple la configuration par défaut du serveur, l'utilisation de mots de passe simples ou l'absence d'authentification lors de l'enregistrement.

Estimation de risque

Le risque s'appuie de deux paramètres principaux : le premier représente les conséquences induites si l'attaque a réussi à atteindre son objectif et le deuxième représente la probabilité

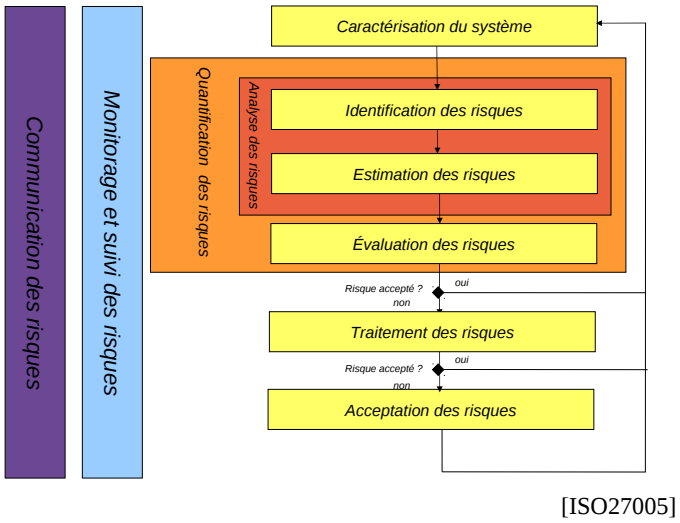


FIGURE 3.2 – Processus de gestion de risques

de l'occurrence de risque [4]. Ainsi, pour évaluer le risque induit, nous devons estimer ces deux facteurs soit d'une manière quantitative (il faut prévoir un modèle mathématique) ou soit d'une manière qualitative (il faut prévoir des niveaux de risques).

Évaluation de la potentialité : La potentialité du risque représente la probabilité qu'une ou un ensemble de vulnérabilités puisse être exploité lors de la construction de l'attaque. Les facteurs suivants doivent être considérés : la motivation et la capacité d'une menace, la nature de la vulnérabilité et l'existence et l'efficacité des contrôles de sécurité.

Il est difficile de trouver des statistiques pour estimer ces différents facteurs car les fournisseurs de services ne communiquent généralement pas ces données pour des raisons de confidentialité. Ainsi, l'évaluation de la probabilité d'occurrence d'un risque est subjective et dépend fortement du contexte d'exploitation et de déploiement de service.

Pour commencer l'évaluation, il faut définir une échelle de potentialité avec un nombre non élevé de potentialité référence. La deuxième étape consiste à évaluer la probabilité de la menace, c'est-à-dire la probabilité de l'existence de la menace indépendamment du service concerné et des mesures de sécurité mises en place. Cette valeur permet de déterminer la facilité de la mise en œuvred'une attaque. Enfin, la troisième étape consiste à évaluer le niveau de vulnérabilité dans le service.

Évaluation des conséquences : l'impact d'une attaque réussie est un élément primordial dans l'évaluation du risque. L'évaluation consiste à estimer les conséquences si la menace parvient à réaliser une attaque au sein de l'infrastructure. Elle permet d'estimer la valeur maximale des conséquences subies lors d'une attaque. Pour commencer, il faut définir une échelle qui est limitée par la conséquence la plus grave, par exemple l'arrêt du service d'établissement de sessions VoIP. Cette échelle contient des conséquences référence comme un dysfonctionnement de l'enregistrement et la dégradation de la qualité de service. En général, les conséquences tiennent en compte l'impact réel de la menace sur l'actif comme les serveurs et les terminaux, les difficultés rencontrées par les clients VoIP et les coûts de recouvrement du risque.

Traitement des risques

Le traitement des risques inclut un ensemble de méthodes qui visent à réduire le niveau de risque à un niveau acceptable. Il existe plusieurs façons pour traiter le risque : (1) soit l'accepter ce qui signifie que le service continue son fonctionnement sans un traitement du risque, (2) soit l'éviter en éliminant la cause du risque c'est-à-dire l'ensemble de vulnérabilités exploitées par la menace, (3) soit de le réduire par des contremesures qui minimisent son impact ou (4) de le transférer à une tierce partie comme l'assurance qui couvre les dégradations induites [11]. Il est souvent difficile de traiter tous les risques identifiés. Il faut dans ce cas donner la priorité au traitement des attaques dont les potentialités et/ou les conséquences sur le service sont les plus élevées.

Le traitement s'appuie typiquement sur des contremesures [73]. En fonction de la contremesure choisie, celle-ci peut soit limiter l'effet du risque, contrer la source de la menace ou éviter l'attaque. Le processus de sélection de contremesures consiste à choisir parmi une combinaison de mesures de sécurité techniques (un pare-feu applicatif, un système de prévention d'intrusions (IPS), un protocole de communication sécurisée ou toute combinaison de celles-ci) afin d'améliorer la sécurité de service. Elles peuvent être configurées pour protéger le service contre des types précis d'attaques. Ces contrôles peuvent aller de contremesures simples à des mises en œuvre complexes impliquant généralement l'architecture de service (par exemple une contremesure par test de *Turing* peut impliquer le service d'établissement de sessions).

Dans nos travaux, nous avons principalement utilisé les mesures de sécurité préventives. Elles se regroupent en plusieurs catégories :

- les mesures d'authentification : elles permet de vérifier l'identité de l'utilisateur. Elles incluent l'utilisation des mots de passe et les certificats,
- les mesures de non-répudiation : un service sûr dépend de la capacité à s'assurer que les expéditeurs ne peuvent pas nier l'envoi d'informations. Cette catégorie inclut l'utilisation de certificats et les clés publiques et privées. Ce contrôle est généralement appliqué au point d'émission ou de réception de messages,
- la communication sécurisée : dans un service distribué, la capacité d'accomplir les objectifs de sécurité est fortement dépendante des communications qui doivent être dignes de confiance. Nous trouvons par exemple l'utilisation du cryptage, des VPNs et d'IPsec,
- les mesures de confidentialité : les services nécessitent de plus en plus le respect de la vie privée des utilisateurs. Nous trouvons des outils de confidentialité des transactions comme la communication via SSL ou l'utilisation d'un *shell* sécurisé.

3.2.3 Modèles de gestion

Il existe plusieurs modèles pour soutenir la gestion des risques, comme les modèles quantitatifs qui donnent une valeur quantitative au risque, les modèles qualitatifs qui présentent le risque en terme de niveau qualitatif et les modèles mixtes sont une combinaison des deux précédents [73]. Dans cette section, nous présentons différents modèles d'évaluation du risque pour les services VoIP.

Dans [88], Dantu et al considèrent que la séquence d'actions d'un attaquant dans le réseau VoIP dépend de son comportement social, par exemple le niveau de compétences, la ténacité et la capacité financière. Ils conçoivent ainsi un mécanisme de sécurité pour évaluer le niveau de risque des ressources critiques qui pourraient être compromises. Cette évaluation est réalisée en utilisant des graphes de comportement d'attaques qui représentent tous les chemins d'attaques possibles pour les ressources critiques. Le niveau de risque est calculé sur la base de ces graphes

et de la quantification des vulnérabilités. Ce niveau de risque constitue ensuite une base efficace pour effectuer les changements appropriés à la configuration du réseau. Un profil est représenté par l'association de l'attaquant à des actifs attaqués. Ensuite, les auteurs affectent à chaque compétence une note qui distingue un attaquant d'un autre. Après la construction du profil de l'attaquant, les auteurs construisent l'arbre d'attaques qui montre comment l'attaquant peut exploiter les vulnérabilités dans le système pour atteindre son objectif. La troisième étape consiste à affecter des compétences nécessaires pour réaliser une tâche représentée par un nœud. Enfin, afin d'évaluer le risque, les auteurs utilisent les attributs, le profil d'attaquant et l'ensemble des chemins dans l'arbre d'attaque. Cette approche ne permet qu'une modélisation partielle du risque dans l'infrastructure VoIP. En particulier, le modèle de risque ne permet pas de quantifier les conséquences subies en cas d'attaque réussie, bien que cela constitue un élément important d'un tel modèle [4].

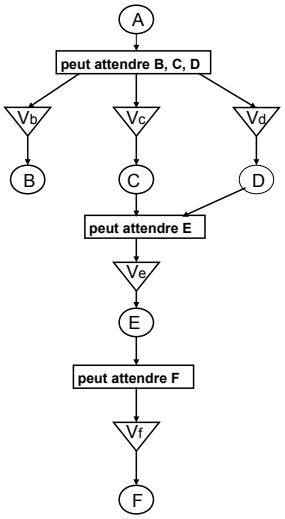


FIGURE 3.3 – Graphe d'attaques [80]

Un autre modèle de risques, proposé dans [80], consiste en une modélisation statistique du risque d'attaques. Cela permet à un fournisseur de services VoIP d'effectuer des analyses de risque en utilisant un modèle de données et un modèle d'impact au dessus d'un graphe d'attaque (voir figure 3.3). Ces deux modèles sont combinés avec un modèle statistique de compétences d'exploitation de l'attaquant. Le modèle de données décrit l'acheminement des flux de données entre les nœuds dans le réseau et comment elles sont traitées par les logiciels et les machines. Le modèle d'impact modélise comment l'exploitation des vulnérabilités affecte les flux de données à l'égard de la confidentialité, l'intégrité et la disponibilité des données. En outre, en attribuant une valeur de perte à l'ensemble des données compromises, ils estiment le coût d'une attaque réussie. Le modèle statistique permet d'incorporer les données de surveillance en temps réel dans le calcul du risque à partir d'un pot de miel. Cette approche commence par la construction d'un modèle du réseau contenant la topologie du réseau et des règles qui régissent les flux de données. Le graphe d'attaques contient trois types de nœuds : les nœuds qui représentent les conditions préalables, les nœuds pour éviter l'état de l'attaque et les nœuds qui représentent une instance de vulnérabilité. Après avoir défini le modèle de réseau et le graphe d'attaques, les auteurs définissent le modèle de flux de données et le modèle d'impact. L'impact d'une attaque réussie est modélisé selon trois aspects : la confidentialité, l'intégrité et la disponibilité. Le modèle définit l'impact

de chaque instance de vulnérabilité sur ces aspects dans une machine par un modèle qualitatif (l'impact peut être soit nul, partiel ou complet). L'inconvénient de cette approche consiste en la non existence d'une stratégie de prévention de risques. En outre, ce modèle de risque n'est pas adéquat pour un service interactif comme la VoIP car il n'analyse et n'évalue pas le risque à l'exécution. Par conséquent, il est incapable d'adapter l'exposition des services en temps réel.

Benini et al. présentent dans [66] un autre modèle de risques relatif à l'interception d'un appel VoIP. Cette situation est perçue comme une menace majeure pour les entreprises qui décident de migrer de la téléphonie traditionnelle aux services VoIP. En particulier, ces travaux prennent en considération le cas d'une société qui préfère la communication et l'échange d'informations via Internet. Ainsi, l'attaquant potentiel est présent sur Internet et son but consiste à capter une conversation en direct entre deux téléphones au sein du réseau privé. La procédure d'évaluation des risques se compose de cinq étapes :

1. La modélisation des menaces possibles par l'intermédiaire d'un arbre d'attaques [87] : le nœud racine représente l'objectif d'attaque et, de manière récursive, les nœuds représentent les sous-buts alternatifs. Chaque nœud satisfait le but parent ou des sous-buts partiels. Les feuilles de l'arbre (les nœuds finaux) représentent les vulnérabilités du système.
2. Les dépendances entre les vulnérabilités identifiées sont définies de cette manière : une vulnérabilité v dépend d'une vulnérabilité w si et seulement si v peut devenir plus facile à exploiter pour atteindre l'objectif d'attaque lorsque w est déjà exploitée.
3. A chaque vulnérabilité v dans l'arbre d'attaques est associée un indice numérique $E_0(v)$. Cet indice est appelé l'exploitabilité initiale. Il mesure les chances que v peut être exploitée avec succès.
4. L'indice de l'exploitabilité $E_i(v)$ de chaque vulnérabilité v est mis à jour par une nouvelle valeur $E_{i+1}(v)$ en prenant en compte ses dépendances jusqu'à ce que les valeurs atteignent un point fixe, c'est-à-dire lorsque les effets des dépendances ont été pleinement prises en considération.
5. Le risque associé à la menace est calculé par l'agrégation récursive des exploitabilités le long de l'arbre d'attaques. Enfin, l'exploitabilité du nœud racine, qui mesure le niveau de la faisabilité de l'attaque, est combinée avec les dommages potentiels subis pour évaluer le risque.

Ce modèle de risque permet de quantifier le risque relatif à l'interception d'appels en se basant sur un schéma de vulnérabilités. Cependant, il ne prend pas en considération la potentialité de l'attaque qui est un paramètre important. Ce paramètre permet notamment d'évaluer la probabilité de survenance d'un scénario d'attaque parmi ceux décrits dans les graphes d'attaques.

Dans nos travaux, nous nous sommes intéressés à un modèle quantitatif de risques appelé Rheostat [42]. Ce dernier est un modèle mathématique qui caractérise le risque subi par une machine. Il décrit la façon dont le risque peut être géré à l'exécution à travers l'adaptation de l'exposition d'une manière dynamique. Il s'agit d'une approche de contrôle d'accès, comme [**?**] qui s'appuie de son côté sur le modèle OrBAC. L'adaptation est obtenue en modifiant le sous-système de contrôle d'accès afin de contrôler l'attribution des permissions aux ressources de la machine et de déterminer si l'application a les droits adéquats ou non. Le système peut alors utiliser le contexte d'exécution pour faire un choix plus éclairé, ce qui resserre l'accès à une ressource quand une menace est détectée. Rheostat caractérise la menace par un enchaînement d'événements. Ainsi, il considère une quantification par signature de la menace. Rheostat la quantifie par le nombre d'événements de la signature achevés ou détectés. La définition du risque est basée sur trois paramètres : la potentialité, l'exposition et les conséquences subies. La potentialité est

définie par l'intersection entre l'ensemble des événements survenus sur la machine et la signature des attaques. Rheostat présente aussi des algorithmes de mitigation pour la gestion des verrous qui contrôlent l'accès à la ressource. Ces algorithmes sont régis par le compromis coût-bénéfice. Quand la valeur de risque atteint le seuil défini, rhesotat réduit l'exposition de l'équipement en mettant en place des mécanismes de vérification de permissions accordées à la ressource concernée. Cependant, rheostat dépend fortement d'un système de détection d'intrusions qui analyse les messages et détecte les attaques. De plus, il reste un modèle générique qui est défini pour tout système et il n'est pas spécifié pour les services VoIP. Les inconvénients de cette approche résident principalement dans la non applicabilité du modèle quantitatif dans un cas précis, ce qui prouve la difficulté de la mettre en œuvre comme elle est. De plus, il est mis en œuvre pour du contrôle d'accès d'application logicielles.

3.3 Évaluation des risques

Dans cette section, nous présentons les techniques et les approches les plus importantes proposées pour l'évaluation des risques. Ces travaux incluent des approches d'estimation des menaces ainsi que d'évaluation des vulnérabilités.

3.3.1 Quantification des menaces

Différentes approches d'identification et de quantification des menaces existent, elles incluent les *honeypots*, la détection d'intrusions par signature ou par anomalies.

Utilisation des *honeypots*

Dans [75], Nassar et al. présentent une architecture de pot de miel pour la détection et l'étude des attaques VoIP. Un pot de miel est un environnement où les vulnérabilités ont été délibérément introduites dans le système afin d'observer les menaces et les intrusions [83]. A partir des tâches attendues par le pot de miel, ils construisent un modèle pour évaluer la menace. L'agent du pot de miel est le cœur de l'architecture et la partie intelligente de l'application. Il est responsable de l'acceptation des appels entrants et d'enquêter sur d'éventuelles attaques. La base de données des profils contient plusieurs fichiers de configuration et permet ainsi à l'administrateur de choisir un profil qui convient à ses besoins, des outils de reconnaissance et le moteur d'inférence qui est capable d'interpréter automatiquement les résultats de l'enquête. Les auteurs ont aussi proposé une approche de contrôle pour distinguer entre une attaque et l'activité normale dans les réseaux VoIP SIP [74]. Cette approche montre une grande efficacité, même lorsque une quantité limitée de données est utilisée dans la phase d'apprentissage. La solution s'appuie sur le suivi d'un ensemble de 38 caractéristiques dans le flux VoIP et l'utilisation de machines à vecteurs support (SVM) pour la classification. Parmi les fonctions caractérisant le trafic SIP, les auteurs utilisent des statistiques sur le nombre de réponses, le nombre de requêtes, la durée moyenne d'appel et des statistiques sur les appelants comme le nombre d'appels effectués et la durée moyenne de leurs appels. Nous utilisons cette approche de détection d'anomalies basée sur les SVMs dans notre stratégie de gestion de risques.

Détection d'intrusions par signatures

Parmi les techniques les plus utilisées pour identifier les attaques, nous trouvons l'utilisation des systèmes de détection d'intrusions (IDS). Un (IDS) est une deuxième ligne de défense derrière

les techniques de protection tels que l'authentification, le contrôle d'accès et la cryptographie. Un IDS est défini par Anderson [14] comme un processus de surveillance de réseaux et des systèmes capables de détecter toute violation de la politique de sécurité appliquée. L'automatisation de l'analyse d'audit et le traitement des traces de grands réseaux sont les objectifs de la détection d'intrusions. En ce qui concerne les données traitées, les IDS sont classés en deux catégories : les systèmes IDS basés sur une analyse de la machine et ceux basés sur une analyse du réseau. Les IDS basés sur l'hôte (*Host* IDS) traitent généralement les informations issues du système d'exploitation et les fichiers d'enregistrement des différentes applications. L'IDS basé sur le réseau (*Network* IDS) traite le trafic du réseau et les différents proxys [34].

Dans le même contexte, Niccolini et al. [86] suggère l'utilisation de la détection d'intrusion par anomalies et la prévention au point d'entrée du réseau VoIP. Ils ont proposé deux étapes de détection afin d'améliorer la précision de l'ensemble du système : la première étape est fondée sur des techniques basées sur la connaissance, la seconde est fondée sur des techniques basées sur le comportement. En utilisant le système Snort [11], les auteurs mettent en œuvre une logique de pré-traitement SIP composée de trois étapes : la première consiste à vérifier la syntaxe des messages SIP et la deuxième analyse la sécurité du protocole SIP. La troisième analyse les messages SIP échangés après la fin de la session concernée. Les tests de performance d'un premier prototype sont évalués par le générateur du trafic BRUTE [19]. En conclusion, l'IDS caractérise la menace par sa signature. Il est capable de quantifier, à partir de cette caractérisation, la menace et évaluer la potentialité de l'attaque. Mais, malgré ses avantages, le rôle de l'IDS reste limité aux étapes de détection d'intrusions et d'évaluation de risques. En effet, il ne propose pas une stratégie de traitement contre l'attaque. Il est une solution de sécurité passive qui n'a généralement aucun impact sur le trafic provenant et sur le comportement des sources d'appels malveillants. Pour compléter ces mécanismes, nous proposons le couplage d'un modèle de risque avec des mécanismes de mitigation.

Détection d'intrusions par anomalies

Le blocage des appels SPIT est difficile en raison de leur nature sociale. La classification des appelants dans des listes blanches et noires ne résout pas catégoriquement le problème, comme dans le cas d'une nouvelle identité. Nous trouvons plusieurs travaux [86, 89, 58] sur le contrôle du risque induit par l'attaque SPIT. Dans [85], J. Quittek et al. proposent une approche de détection des appels SPIT en appliquant des tests de *turing* et comparant leurs résultats avec la communication humaine typique. Pour réussir ces tests, l'attaquant est obligé d'utiliser des ressources importantes pour générer l'attaque SPIT par des bots ce qui contredit l'objectif de l'attaquant qui consiste à appeler un nombre important d'utilisateurs VoIP. Ils ont validé sa faisabilité avec un prototype intégré dans le système de sécurité modulaire VoIP SEAL [89].

Dans le même contexte, [99] présente une approche de détection d'attaques SPIT en se basant sur un modèle de risque qualitatif. Ils proposent un algorithme de protection contre l'attaque SPIT appelé *Progressif Multi Gray-Leveling* (PMG). Cet algorithme calcule le niveau de gris d'un appelant : le niveau détermine si l'appelant est une source de spam probable ou non. Il détermine aussi si l'appel doit être acheminé en se basant sur les motifs d'appels précédents et le retour d'expérience des utilisateurs. Le niveau de gris détermine la légitimité d'un expéditeur en fonction de la situation actuelle. Selon la politique de sécurité, PMG peut également travailler avec des listes blanches et noires. Il utilise deux paramètres dans son algorithme de calcul : un paramètre de niveau de gris à court terme et un autre à long terme. La décision d'acheminer

11. http ://www.snort.org/

l'appel est prise par le calcul de deux paramètres et en les intégrant. Si le résultat est au-dessous du seuil défini, la connexion est établie, sinon, la connexion est bloquée.

Ces techniques de détection par anomalies vise uniquement l'évaluation de la potentialité d'attaques. Elles n'offrent ni un modèle complet de risques qui évalue le niveau de l'exposition du réseau face aux attaques et les conséquences subies, ni une stratégie de prévention qui protège les infrastructures VoIP et adapte son exposition face aux attaques. Nous cherchons à travers nos travaux à appliquer et automatiser la gestion des risques pour garantir le couplage entre la détection et le traitement et offrir un compromis entre la sécurité et la qualité du service.

3.3.2 Quantification des vulnérabilités

La quantification des vulnérabilités est définie comme le processus de l'identification, de la classification et l'atténuation des vulnérabilités [107]. Afin d'établir une activité sécurisée, il est nécessaire de spécifier une politique définissant l'état du système concerné et un point de départ bien connu pour identifier les vulnérabilités et la conformité aux politiques de sécurité [59]. Dans cette section, nous considérons trois techniques de quantification des vulnérabilités à savoir la description de configuration, le *fingerprinting* et le *fuzzing*.

Dans [55, 56, 57], Keromytis propose une approche qui consiste à placer les vulnérabilités connues dans un cadre structuré. Ainsi, la cartographie de l'espace des vulnérabilités selon plusieurs axes peut révéler des zones qui méritent une analyse plus approfondie. Comme point de départ, il se sert de la taxonomie fournie par VoIPSA [12] [112]. Il classifie, ainsi, les vulnérabilités par catégorie de menace et présente des statistiques sur leur répartition. Des efforts de normalisation ont été faits pour la description des vulnérabilités, en particulier avec le langage de spécification OVAL. OVAL [13] est un langage de spécification des vulnérabilités qui a pour objectif la promotion d'un contenu de sécurité ouvert et accessible au public, et de normaliser le transfert de ces informations dans l'ensemble des outils et des services de sécurité. Il est utilisé pour coder les détails du système et des vulnérabilités. C'est un langage basé sur XML qui fournit les moyens pour décrire les vulnérabilités d'une manière uniforme. Ces descriptions peuvent être ensuite utilisées pour identifier les vulnérabilités à partir des paramètres de configuration.

Parmi les techniques utilisés, nous trouvons aussi le *fingerprinting* comme une méthode importante pour l'identification et la quantification des vulnérabilités. Cela a commencé avec le travail de pionnier de Comer et Lin [26]. La précision de cette technique augmente avec la quantité des informations recueillies et analysées à partir des sources comme les informations archivées des serveurs et des sessions établies. Il est important de noter cependant que les informations provenant d'une source peuvent être intentionnellement modifiées. Plusieurs approches [45, 70, 39] présentent des méthodes pour identifier les machines ou les logiciels existants dans un réseau. Cette classification est souvent effectuée sur la base des messages de réseau générés par les machines. Les techniques de *fingerprinting* peuvent être basées sur les propriétés des paquets et sur les informations du protocole. Les techniques de *fingerprinting* basées sur les paquets considèrent que le message est la source unique pour l'analyse. La recherche des signatures typiques est basée sur les informations obtenues à partir des messages observés, par exemple, les en-têtes et l'ordre du contenu. La technique de *fingerprinting* basée sur le niveau du protocole concentre son analyse sur le comportement observé de l'entité cible. La recherche des signatures pour cette catégorie est typiquement basée sur le type de réponses à des événements spécifiques comme le retard dans la transaction requêtes / réponses. Les applications de *fingerprinting* permettent la

12. Voice over IP Security Alliance
13. Open Vulnerability and Assessment Language, www.oval.mitre.org/language/

découverte de topologie, l'identification des machines, attaques et la détection des attaques, des virus et des spams.

Ces approches de spécification de vulnérabilités permettent leur identification et leur quantification. Elles permettent aussi l'évaluation de l'exposition des services VoIP. Ainsi, ces techniques fournissent un moyen de contrôler l'exposition et de la réduire dans le cas d'une attaque qui exploite une vulnérabilité. Cependant, ces approches n'offrent pas un modèle de risques complet car elles n'évaluent ni la potentialité d'attaque ni les conséquences subies. En plus, la spécification des vulnérabilités offre souvent un ensemble de recommandations pour le recouvrement des vulnérabilités mais elle ne présente généralement pas une stratégie complète de traitement de risques.

Une autre technique utilisée pour la quantification des vulnérabilités est le *fuzzing*. KiF [12] est un outil de *fuzzing* développé pour le protocole SIP. Il contient deux composants autonomes : le *fuzzer* de syntaxe et l'évaluateur de l'état du protocole qui forment conjointement une entité de validation des données d'une façon déconnectée. Un test est généré par scénario, où un scénario représente un objectif de haut niveau. Par exemple, un scénario qui teste les requêtes SIP. Il représente une série de tests et il est conçu sur la connaissance du domaine du protocole et l'injection aléatoire de données. Les tests peuvent être semblables au comportement normal ou peut inonder l'appareil avec des messages malveillants. Le *fuzzing* et le *fingerprinting* sont des techniques de détection des vulnérabilités et des configurations malveillantes. L'inconvénient de ces techniques est qu'elles détectent principalement les attaques à base de signature comme l'empoisonnement par messages ARP ou le débordement de mémoire. Les attaques à base d'anomalies sont difficilement détectables par ces techniques parce qu'elles ne sont pas définies par une signature ou un ensemble d'événements. Pourtant, ils permettent d'atténuer l'impact des vulnérabilités (réduire l'exposition par le recouvrement des vulnérabilités). Ces approches ne proposent généralement pas un schéma de mitigation du risque couplé avec la technique de l'identification des vulnérabilités.

Nous trouvons d'autres outils d'évaluation et de découverte des vulnérabilités dans les architectures VoIP comme VoIP audit [14]. Cet outil consiste en une vérification de l'infrastructure du réseau afin de savoir si elle est prête à soutenir la technologie VoIP. Il comprend les tâches suivantes :

1. Comprendre le contexte par la collecte des informations et les entretiens avec le personnel et déterminer la période de l'audit,
2. Générer un aperçu sur la situation existante comme la situation de l'architecture physique et la qualité de service,
3. Réaliser des mesures pour évaluer la performance du réseau en terme de qualité de service,
4. Réaliser une analyse statistique qui consiste en un traitement des données saisies et la publication de graphiques de synthèse,
5. Comparer les résultats obtenus par rapport aux caractéristiques du réseau,
6. Identifier les sources possibles de menaces et les solutions d'optimisation,
7. Présenter des recommandations. Cette étape consiste à préparer un rapport de synthèse comprenant les résultats de l'analyse et les recommandations des changements pour l'optimisation de l'infrastructure.

Cet outil automatise le processus de gestion des risques pour sécuriser les architectures VoIP. Cependant, il ne propose qu'une seule stratégie de traitement du risque : la stratégie d'évitement

14. Telephony Security and Fraud Protection, www.voipshield.com/products/voipaudit-overview.php

à travers les recommandations. Ceci n'est pas toujours efficace pour traiter les menaces et les risques induits dans des architectures dynamiques comme celles de la téléphonie sur IP.

3.4 Traitement des risques

Le traitement des risques est une étape importante du processus de gestion de risques. Il consiste à mettre en œuvre une stratégie de prévention de risques couplée avec la technique de détection d'attaques et le modèle de risques. Dans cette section, nous présentons les différentes stratégies de traitement de risques, à savoir la stratégie d'évitement, d'optimisation et de rétention de risques.

3.4.1 Stratégie d'évitement

Une stratégie d'évitement est un mécanisme de prévention contre la menace qui consiste à mettre en place des solutions de défense pour supprimer le risque [11]. L'implémentation de cette stratégie se fait d'une façon anticipée de l'attaque. Dans ce contexte, plusieurs protocoles peuvent être utilisés pour assurer l'intégrité et la confidentialité des messages de signalisation SIP contre les différentes attaques. Ces recommandations portent notamment sur l'utilisation de protocoles de sécurité tels que IPsec [15], S/MIME [16], TLS [17], et DTLS. Deux critères fondamentales pour l'adoption d'un protocole de sécurité sont la facilité de mise en œuvre et le passage à l'échelle.

Afin d'éviter les attaques contre la signalisation, SIP utilise HTTP *Digest Authentication* pour fournir l'authentification et la protection contre la rediffusion de messages pour l'enregistrement, l'initiation de la session et sa terminaison [2]. En règle générale, les informations d'authentification SIP sont significatives dans un domaine spécifique. Un domaine gère les informations d'identification de ses utilisateurs, mais il ne peut pas déléguer les informations d'identification à d'autres domaines à moins qu'il y ait une relation de confiance inter-domaine définie. Les implémentations SIP peuvent appliquer un défi d'authentification à des degrés différents, ce qui ne peut pas fournir une sécurité optimale. Par exemple, une implémentation peut authentifier les demandes d'enregistrement seulement, sans le faire pour l'initiation d'une session par message INVITE et un autre peut exiger une authentification pour REGISTER et INVITE mais pas pour les requêtes BYE et CANCEL.

Parmi les protocoles adoptés par l'industrie pour soutenir la confidentialité, nous trouvons le protocole TLS (*Transport Layer Security*) [5] défini par la RFC 4346. Il offre la possibilité d'effectuer l'authentification mutuelle. Le protocole est composé de deux couches : le protocole TLS d'enregistrement et le protocole TLS de négociation. La première couche maintient une connexion sécurisée entre deux points. Alors que la deuxième couche est responsable de la négociation des propriétés cryptographiques de la connexion. Pour le protocole SIP, ce mécanisme est soutenu par l'utilisation de SIPS URI en modifiant légèrement la syntaxe des adresses utilisées, le type du protocole de transport et en utilisant un nouveau port. Bien que son utilisation ait un impact faible sur les performances du protocole SIP, il existe quelques limitations à prendre en compte. Il s'agit d'un mécanisme saut par saut et d'une authentification mutuelle qui peut ne pas être évolutive car elle nécessite le déploiement d'une infrastructure complète à clé publique (PKI). De plus, il a été conçu pour être utilisé avec un protocole de transport fiable, nécessitant que toutes les applications VoIP utilisent TCP.

15. IPsec, Security Architecture for the Internet Protocol, www.ietf.org/rfc/rfc2401.txt
16. Secure/Multipurpose Internet Mail Extensions
17. RFC 4346, SIP and TLS, tools.ietf.org/html/rfc4346

Afin d'assurer une protection équivalente à TLS pour les protocoles applicatifs qui se basent sur le protocole UDP, le protocole DTLS a été conçu et est défini dans la RFC 4347 [6]. DTLS est similaire à TLS sur plusieurs points, y compris la non nécessité d'établissement d'une nouvelle session entre les stations pour protéger les messages SIP de bout en bout. Une différence fondamentale entre TLS et DTLS est que DTLS fournit un mécanisme pour gérer la non fiabilité associée à UDP tels que la correction de perte de paquets ou la réorganisation des messages. Donc, DTLS hérite des propriétés de sécurité éprouvées de TLS, fournit des mécanismes pour compenser les limitations de TLS pour la fiabilité, détecte le renvoi des messages et utilise les cookies sans états qui protègent contre les attaques de déni de service. Mais, il a aussi des inconvénients comme par exemple la nécessité d'une infrastructure centralisée PKI et TLS pour fournir les certificats.

Un autre protocole de sécurité pour la signalisation est S/MIME [18] qui est défini dans la RFC 3851 [6]. Il fournit une confidentialité de bout en bout. MIME définit un ensemble de mécanismes pour encoder et représenter des formats de message complexes tels que les pièces jointes multimédia et les caractères linguistiques. Cette combinaison (MIME et S/MIME) offre un grand niveau de flexibilité pour l'échange des messages complexes. S/MIME est utilisé pour sécuriser les en-têtes d'un message SIP. Contrairement à TLS et DTLS, S/MIME se caractérise par la flexibilité pour une protection plus granulaire des en-têtes de l'information dans les messages SIP. Mais, il reste dépendant d'une infrastructure centrale le PKI qui centralise la distribution des clés et des certificats. En conclusion, ces protocoles de sécurité sont utilisés pour protéger la signalisation SIP et éviter des attaques de sécurité. Mais, il reste toujours des attaques dont l'évitement est techniquement difficile comme les attaques de déni de services et le SPIT.

3.4.2 Stratégie d'optimisation

La stratégie d'optimisation de risques vise à réduire l'exposition des services contre la menace. La mise en place de ce mécanisme se fait généralement d'une façon réactive, c'est-à-dire si on détecte une anomalie, la stratégie d'optimisation prend la décision d'appliquer un ensemble de contremesures [11].

Pour réduire le risque induit par les menaces, nous pouvons principalement agir sur deux paramètres : les conséquences subies et l'exposition. Concernant les conséquences de la menace, nous pouvons optimiser son impact par des solutions de recours et des alternatives qui permettent de garder la continuité opérationnelle du système comme par exemple des serveurs qui remplacent ceux qui tombent en panne. Pour optimiser l'exposition, nous devons soit optimiser l'exposition relative aux vulnérabilités par des solutions de recouvrement comme les patches [53], un audit proactif [15] qui examine les infrastructures et détermine les points faibles en termes de vulnérabilités ou des solutions de protection comme les pares feux applicatifs et les systèmes de prévention d'intrusions [44]. Des modèles de coûts élaborés [**?**] sont envisageables dans ce contexte pour soutenir la sélection des mesures de sécurité.

Dans [61], Lahmadi et al. présentent un pare feu applicatif pour le protocole SIP appelé SecSip. L'idée consiste à concevoir et mettre en œuvre un système de défense SIP qui prend en charge une analyse approfondie des messages SIP avec une fonction de suivi de l'état du protocole. Un pare feu exige un module d'analyse du protocole qui évalue la sécurité des messages et la compatibilité avec les règles définies par le protocole avant de les transférer à la destination. SecSip utilise la technique de *fuzzing* pour détecter l'existence des vulnérabilités et représente les différentes règles concernant les vulnérabilités par le langage VeTo [62]. D'abord, SecSip utilise

18. RFC 3851, S/MIME, www.ietf.org/rfc/rfc3851.txt

un moteur basé sur des règles pour exécuter les règles du modèle de vulnérabilités SIP écrit en VeTo. Ensuite, il surveille le protocole SIP pour permettre un suivi sémantique.

Dans [36, 41], Niccolini et al. analysent les exigences VoIP pour la détection d'intrusions et la prévention. Ils recommandent l'utilisation d'un système de détection d'intrusions basé sur le réseau et un système de prévention caractérisé par l'adoption d'une technique en deux étapes. D'abord, le système applique les techniques basées sur la connaissance des attaques. Si le suspect réussit à passer la première étape, le système analyse le comportement des messages envoyés pour détecter des signatures d'attaque. L'inconvénient des systèmes de prévention d'intrusions est la difficulté de garder un compromis entre la qualité de service et la sécurité par le fait que ce système est susceptible de bloquer l'appel s'il le suspecte.

Afin de réduire le risque induit par l'attaque SPIT, les auteurs dans [46] énumèrent plusieurs contremesures comme l'utilisation des listes qui distinguent entre les utilisateurs. Nous trouvons les listes blanches qui contiennent les utilisateurs des services VoIP, les listes grises qui contiennent les utilisateurs suspects de générer un trafic malicieux et la liste noire pour les attaquants. Une autre contremesure citées est la CAPTCHA audio qui permet de vérifier si l'appelant a un comportement interactif ou si les conversations sont simulées. Nous utilisons ces contremesures dans notre stratégie afin de définir notre schéma de traitement de risques.

Dans le même contexte, Niccolini et al. présentent, dans [72], un système de détection et prévention SDRS [19] contre les attaques SPIT. Ce système combine des modèles de détection connus, telles que les listes noires et les listes blanches, avec des méthodes basées sur l'analyse statistique du trafic, telles que le nombre et la durée des appels effectués d'un utilisateur. SDRS se compose de trois principaux modules : la classification des utilisateurs, la détection et la réaction contre l'attaque SPIT. En surveillant le comportement des utilisateurs, SDRS classe les utilisateurs en malveillants ou normaux et collecte également les préférences des destinataires. Le système utilise ensuite ces données comme entrée pour les composants de la détection et de la réaction. SDRS met en œuvre des réactions diverses contre les appels SPIT, tels que le rejet des appels suspects et la limitation de taux de génération d'appels. Après la détection d'un appel SPIT, le système prend des mesures en se basant sur le niveau d'appel SPIT, les politiques du fournisseur d'accès, et la catégorie de l'utilisateur. Ils proposent plusieurs contremesures comme la réduction de nombre d'appels autorisés en fixant un seuil d'appel ou la redirection des appels ou même le blocage des appels par des *blacklists*. L'inconvénient de ces mécanismes est qu'ils sont basés sur un blocage d'appels soit définitif ou temporaire ce qui n'est pas approprié pour le fournisseur d'accès.

Concernant les attaques de déni de services (DoS), nous trouvons des solutions intéressantes comme le vFDS (*VoIP Flooding Detection System*) [95], un mécanisme de détection d'anomalies en temps réel qui génère des alertes en fonction des variations anormales dans un ensemble de flux de trafic. Il fonctionne en regardant des collections de flux de paquets comme une évolution des distributions de probabilité et en mesurant les variations anormales dans leurs relations basées sur la distance de Hellinger [96]. De bonnes performances ont été montrées avec une précision de 100% à moins de 500 INVITE par seconde. Cependant, nous remarquons que cette approche ne serait pas applicable dans un environnement réel parce qu'elle ne détecte pas l'inondation SIP à haute fréquence. Toute variation de la distance de Hellinger n'est pas nécessairement une inondation. Enfin, les auteurs de [109] présentent une approche de détection et prévention contre les attaques par inondation SIP. Ils développent un système de prévention en intégrant une technique d'esquisse [104] avec la détection basée sur la distance de Hellinger. Cette approche propose un mécanisme qui est capable de résumer chacun des messages SIP entrants par un

19. Spam Detection and Reaction System

	Modèle de risques			Stratégie de prévention		
	Potentialité	Exposition	Conséquences	Évitement	Optimisation	Acceptation
Rheostat [42]	oui	oui	oui	-	oui	-
Gestion de risque de l'interception d'appels [66]	-	oui	oui	-	-	-
Modélisation statistique du risque [80]	oui	oui	oui	-	-	-
Gestion de risques par le profilage d'attaquant [88]	-	oui	-	oui	-	-
Détection par signature [14, 19]	oui	-	-	oui	-	-
Détection par anomalies [85]	oui	-	-	oui	-	-
Sécurité signalisation [2, 5, 6]	-	-	-	oui	-	-
Prévention d'intrusions [61, 62]	-	-	-	-	oui	-
Mécanismes de confiance [93, 13]	oui	-	-	oui	-	oui
Auto-génération des SIP URIs[93]	-	-	-	oui	-	oui

FIGURE 3.4 – Classification des travaux par rapport au processus de gestion de risques

ensemble de données compacts et de taille constante. Cependant, cette approche n'offre pas un ensemble de mesures de sécurité pour contrer cette attaque.

3.4.3 Stratégie de rétention et d'acceptation

La stratégie d'acceptation des risques est une stratégie de traitement de risques qui considère que dans certaines conditions on peut accepter le risque. Par exemple, dans certaine infrastructure VoIP, la menace de l'écoute du trafic ou de son analyse ne pose pas de problèmes pour les utilisateurs car le niveau de confidentialité exigé n'est pas élevé. Dans cette partie, nous nous intéressons aux problèmes de sécurité dans les architectures distribuées comme les réseaux P2PSIP où des stratégies d'acceptation de risques sont fréquemment appliquées [105].

Les mécanismes de confiance et de réputation [93, 13, 47] sont considérés comme des solutions de sécurité qui acceptent le risque. Ils sont flexibles et adaptables aux réseaux pair à pair car ils respectent la nature distribuée de ces réseaux. Ils fournissent en plus une solution du routage sécurisée. Ces solutions acceptent le risque dans certain cas et se basent principalement sur une notation mutuelle entre les différents utilisateurs pour évaluer le niveau de risque de chacun. Ces mécanismes donnent des résultats approximatifs du risque car ils se basent sur une notation subjective.

Afin de respecter la nature des réseaux pair à pair, une solution d'authentification des nœuds devrait être capable de passer à l'échelle, ainsi, elle doit être distribuée. Les systèmes de gestion de réputation offrent une forme plus faible d'authentification distribuée par l'attribution de valeurs de confiance aux nœuds [117]. Plusieurs systèmes de gestion de réputation pour les réseaux pair

à pair ont été proposés comme l'algorithme *eigentrust* [54]. Ces systèmes peuvent typiquement être dans le contexte pair à pair pour contrer les attaques d'usurpation d'identités. En effet, seuls les nœuds qui ont une bonne réputation peuvent fournir des services au réseau et utiliser les différents services proposés par les autres nœuds. La plupart des travaux de recherches sur la gestion de réputation dans les réseaux pair à pair se concentrent sur les réseaux de partage de fichiers. Cependant, nous ne trouvons pas des solution de gestion de confiance spécifiques aux réseaux P2PSIP. En outre, la plupart des systèmes existants s'appuient sur une autorité centrale comme le PKI qui facilite le routage sécurisé. Ainsi, l'intégration d'un système de gestion de réputation dans le réseau P2PSIP reste un problème ouvert.

Parmi les approches de cinfiance de réputation, l'algorithme *eigentrust* est conçu pour les systèmes de partage de fichiers dans les réseaux pair à pair [54]. Il a pour but de diminuer le nombre de téléchargements de fichiers non authentiques dans un réseau pair à pair. Il attribue à chaque pair une valeur de confiance globale et unique, basée sur l'histoire des téléchargements. Ainsi, *eigentrust* peut identifier efficacement les pairs malveillants et les isole du réseau. Le processus de calcul commence par l'estimation d'une valeur de satisfaction : chaque pair i donne une valeur de satisfaction sur les autres pairs. Cette valeur quantifie comment les différents pairs traitent le service de téléchargement. Ensuite, l'algorithme calcule une valeur de confiance locale (d'un pair i par rapport un pair j). Après, il raffine ces valeurs en consultant toutes les valeurs de confiance locales. Enfin, il donne comme résultat un vecteur de confiance qui contient des valeurs globales. Cependant, il est nécessaire de sécuriser cet algorithme contre toute manipulation. En effet, un pair malveillant peut envoyer des valeurs de satisfaction erronées, ainsi, il induit en erreur le vecteur de confiance. Pour cela, les auteurs proposent une version sécurisée de l'algorithme. Elle consiste à déterminer M nœuds qui ont le droit d'évaluer la réputation dans le réseau. Dans le même contexte, les auteurs dans [47] proposent une solution à base des relations sociales de confiance pour filtrer les appels qui sont considérés comme des attaques SPIT. Ils définissent des nœuds chercheurs de route de confiance (*trusted pathfinder*). Ils agissent comme un tiers de confiance qui fournissent une route vers le destinataire par une description du chemin de confiance anonyme.

Dans les réseaux P2PSIP où une autorité centrale est inexistante, l'auto-certification d'identités peut être utilisée pour l'authentification du contenu dans une architecture distribuée [93]. Ainsi, une identité peut être vérifiée sans consulter une partie de confiance. Par exemple, l'association d'une identité à une clé publique peut être vérifiée mathématiquement si l'identité est représentée comme le hachage de la clé publique. En P2PSIP, un utilisateur peut hacher sa clé publique pour générer un SIP URI. Il peut alors signer numériquement ses données de localisation avec sa clé privée. Toutefois, une telle approche présente certains inconvénients comme l'utilisation des identités SIP URI difficile à mémoriser par les utilisateurs. En outre, il peut y avoir des attaques de collision sur la fonction de hachage. Enfin, un utilisateur peut ne pas être sûr que l'URI appartient à l'appelé souhaité.

3.5 Synthèse

Le risque est la combinaison de la probabilité d'une attaque et de ses conséquences sur l'infrastructure VoIP en cas de réussite. La gestion des risques est le processus qui consiste à identifier les risques, les évaluer, et décider des contremesures à appliquer pour les réduire à un niveau acceptable. Dans ce chapitre, nous avons décrit le processus de gestion des risques et présenté les différents travaux dans le domaine de la sécurité des architectures VoIP. Nous les avons classés suivant leur contribution au processus de gestion. Nous avons montré l'importance

du couplage entre les trois principales étapes de la gestion des risques à savoir l'identification, l'évaluation et le traitement des risques. En effet, chaque étape a une fonction bien précise qui contribue à la réalisation de la suivante. Cependant, on ne trouve pas ce couplage dans la plupart des travaux de sécurité pour les services VoIP. Les approches proposées offrent généralement soit la détection d'intrusion, soit l'évaluation d'un paramètre du risque ou une stratégie de traitement. De plus, un modèle de risques doit inclure les paramètres suivants : la potentialité de la menace, l'exposition de l'infrastructure et les conséquences subies en cas d'attaques réussies. Toutefois, les travaux de recherche déjà cités ne présentent pas de modèles de risque complet, ces modèles sont d'ailleurs souvent implicites (voir figure 3.4). Une autre limitation de ces travaux est que souvent ils ne sont pas mis en œuvre à l'exécution et ne permettent pas de répondre dynamiquement au compromis entre sécurité et performance, si important dans ces infrastructures. Dans ce contexte, nos travaux visent à appliquer et automatiser la gestion des risques dans les réseaux VoIP pour améliorer ce couplage et apporter une réponse adéquate aux attaques tout en minimisant la dégradation du service.

Deuxième partie

Approche de gestion des risques pour la téléphonie sur IP

Chapitre 4

Stratégie de gestion des risques

Sommaire

4.1 Introduction

Nous proposons dans ce chapitre une nouvelle approche pour l'application et l'automatisation de la gestion des risques à l'exécution dans les réseaux et services VoIP [79]. Nous considérons le cas des attaques SPIT pour évaluer l'efficacité et la faisabilité de notre approche dans ce contexte. L'automatisation vise à renforcer le couplage entre l'évaluation et le traitement de risques. Une contremesure vise à empêcher la réalisation d'une attaque de sécurité, mais elle peut aussi détériorer le service en introduisant des retards supplémentaires ou en réduisant l'accès à certaines fonctionnalités spécifiques [81]. La gestion des risques est requise pour les infrastructures VoIP parce qu'elle permet de gérer le compromis entre la sécurité et la qualité du service, qui sont toutes les deux cruciales pour ces services. L'objectif est d'adapter en permanence l'exposition des équipements VoIP en activant ou désactivant les contrôles de sécurité dans le réseau d'une manière progressive et dynamique. Dans une première partie, nous décrivons la stratégie de gestion des risques ainsi que son modèle quantitatif d'évaluation des risques, les algorithmes de gestion et l'ensemble de contremesures utilisées pour contrer les attaques SPIT. Dans une deuxième partie, nous présentons le couplage entre notre approche et une technique de détection

d'anomalies et spécifions notre modèle de gestion de risques pour prendre en compte les nouveaux paramètres de détection. Ensuite, nous décrivons l'architecture qui supporte notre approche de gestion de risques ainsi que les interactions entre les différents composants. Enfin, nous présentons un ensemble d'expérimentations pour évaluer les performances de notre solution.

4.2 Modèle et algorithmes

Une grande variété de mécanismes de protection et de détection ont été développés pour identifier et bloquer les attaques de sécurité VoIP. Cependant, les méthodes de détection montrent rapidement leurs limites en termes de sensibilité et de la spécificité. La potentialité d'une attaque est souvent difficile à estimer et la durée de la détection d'une attaque peut être importante par rapport à l'échelle de temps liée à cette attaque. De plus, les mécanismes de protection (comme les protocoles de sécurité classiques TLS/DTLS et IPsec, ou les pare-feux applicatives) ont souvent un coût d'application non négligeable et peuvent avoir un impact significatif sur les performances du service de téléphonie par rapport à sa continuité opérationnelle et sa qualité de service comme par exemple le blocage des appels. En effet, la téléphonie sur IP est un service temps réel nécessitant des performances réseau suffisamment élevées comme un taux du rejet d'appel faible et une gigue constante afin de garantir la continuité opérationnelle. L'application des mécanismes de protection pourrait nuire considérablement à un tel service critique [81].

Dans ce contexte, La gestion des risques offre de nouvelles perspectives à l'égard de ce problème en offrant un compromis entre la sécurité et les performances du service [4]. Elle vise à quantifier l'intensité de la menace et à choisir les mesures de sécurité appropriées afin de minimiser l'impact sur l'infrastructure VoIP. Rheostat a défini un modèle de risques qui permet d'adapter dynamiquement l'exposition de l'infrastructure contre des risques en se basant sur un ensemble de mesures de sécurité. Ces contremesures sont activées ou désactivées en focntion de la probabilité d'attaques.

Pour appliquer le processus de gestion de risques, nous devons penser aux trois principales étapes à savoir l'identification de risques, l'estimation de risques et son traitement. Ainsi, plusieurs éléments sont indispensables pour mettre en œuvre les deux dernières étapes : un modèle de risques, des algorithmes de mitigation et un ensemble de contremesures. Pour l'identification de risques, un système de détection d'intrusions est indispensable.

Nous avons choisi de concevoir un modèle de risque quantitatif car il est plus précis et significatif pour un service critique et interactif comme la VoIP et nous permet d'avoir une idée claire sur le niveau de risque et d'automatiser le processus de risque. Pour développer notre propre modèle du risque, nous nous sommes appuyés sur le formalisme introduit par rheostat [42]. Nous avons étendu le modèle pour les réseaux VoIP centralisés en prenant en compte les propriétés de cette infrastructure et détaillons la sélection de contremesures dans le contexte d'attaques SPIT.

4.2.1 Modélisation du risque

Nous proposons d'automatiser la gestion des risques pour les réseaux VoIP d'entreprise. Notre schéma repose sur un modèle de risque qui fournit un support pour modifier dynamiquement l'exposition et contrer les attaques SPIT. Cette altération est entraînée par une analyse coût-bénéfice à l'exécution afin de fournir une réponse systématique à l'égard de la potentialité d'une attaque de sécurité. L'exposition de l'équipement VoIP est contrôlée par des contremesures auxiliaires appliquées d'une manière progressive et l'activation d'un contrôle de sécurité permet de réduire l'exposition lorsque la potentialité d'une attaque augmente, tandis que la désactivation permet

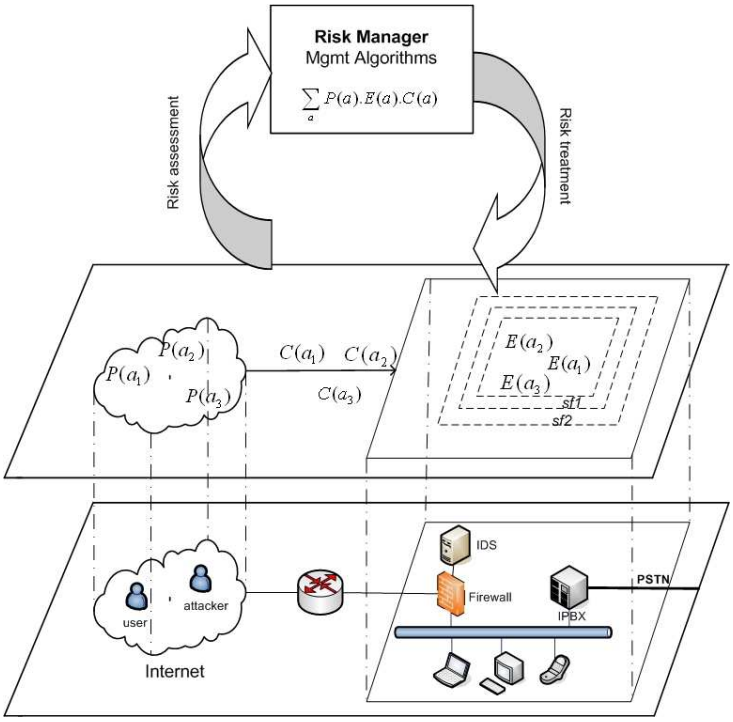


FIGURE 4.1 – Notre approche de gestion adaptative de risques

de réduire les coûts de sécurité et d'améliorer la qualité de service et la continuité opérationnelle du service VoIP lorsque la potentialité est faible.

Étant donné un système contenant un ensemble de vulnérabilités $W = \{w_1, \ldots, w_n\}$, le risque est défini par la combinaison de deux paramètres principaux : (a) la probabilité qu'une menace $a \in A$ réalise une attaque par l'exploitation d'une ou d'un ensemble de vulnérabilités de l'ensemble W et (b) les conséquences subies par le système ($\mathcal{C}(a)$) [17]. La probabilité de survenance d'une attaque est composée de deux paramètres : l'exposition du système $\mathcal{E}(a)$ et la potentialité de l'attaque $\mathcal{P}(a)$.

Ainsi, le modèle quantitatif de notre approche de gestion de risques dépend principalement de trois paramètres principaux pour évaluer le niveau de risque [33] :

- $\mathcal{P}(a)$ représente la potentialité de l'attaque. Elle quantifie soit la probabilité de son occurrence, si l'attaque est une menace à base de signature, ou son intensité, si l'attaque est une menace à base d'anomalies,
- $\mathcal{E}(a)$ représente l'exposition de l'infrastructure à l'égard de cette menace (sur la base des vulnérabilités existantes). Elle définit aussi le niveau de protection de l'infrastructure VoIP concernée,
- $\mathcal{C}(a)$ quantifie les conséquences de cette attaque sur les ressources de l'infrastructure, ce qui correspond à la dégradation subie en termes de disponibilité, qualité de service, confidentialité et intégrité.

L'ensemble des paramètres du modèle est normalisé. Sur la figure 4.1, nous représentons notre stratégie de gestion des risques superposée à une architecture du réseau VoIP d'entreprise. Dans un contexte d'attaque SPIT, les attaquants sont responsables de la génération de messages non sollicités, ainsi, c'est eux qui fixent la valeur de la potentialité comme cela est montré dans la figure. L'exposition contre l'attaque représente l'ensemble des barrières qui protègent l'équipement

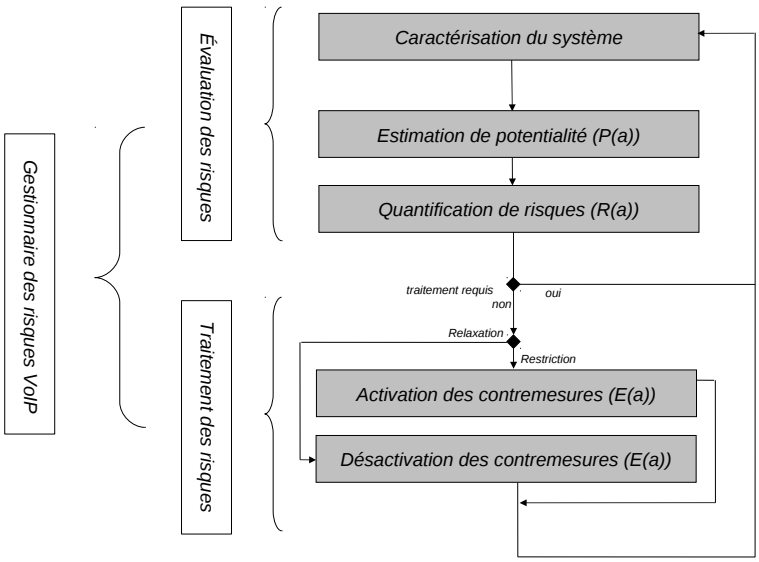


FIGURE 4.2 – Modèle du risque suivant le processus de gestion

VoIP et enfin, les conséquences d'une attaque ne sont propagées et effectives qu'après la génération du trafic malveillant.

Le niveau de risque est évalué par l'équation 4.1.

$$\mathcal{R} = \sum_{a \in A} \mathcal{P}(a) \times \mathcal{E}(a) \times \mathcal{C}(a) \tag{4.1}$$

La potentialité d'une attaque est estimée par un système de détection. L'approche rheostat, lorsqu'une signature est partiellement détectée, la séquence d'événements nécessaire pour compléter l'attaque, est utilisée comme un indicateur pour l'évaluation de degré de l'avancement de l'attaquant et par conséquent sa potentialité. Dans notre scénario, nous avons considéré l'attaque SPIT qui est une attaque détectable généralement par anomalies et non par signature. Elle est spécifiée par une liste d'anomalies comme une durée moyenne d'appels courte, un temps inter-appels, un taux de rejet élevé, etc [58]. Nous avons donc défini mathématiquement la potentialité de l'attaque SPIT $\mathcal{P}(a)$ par l'équation 4.3 avec $\{\beta, \gamma, \delta, \varepsilon, \zeta, \eta\}$ l'ensemble des facteurs de pondération des paramètres de détection.

$$\begin{aligned} \mathcal{P}(a) &= \beta \times Duration_{Rate} + \gamma \times Traffic_{Rate} + \delta \times Rejection_{Rate} \\ &+ \varepsilon \times Inter_{Rate} + \zeta \times Call_{Rate} + \eta \times Recipient_{Rate} \end{aligned} \tag{4.2}$$

Cette pondération est spécifiée en se basant sur une étude faite dans [58] avec $Duration_{Rate}$ la durée moyenne d'appel VoIP, $Traffic_{Rate}$ le taux du trafic, $Rejection_{Rate}$ le taux de rejet d'appel, $Inter_{Rate}$ la durée moyenne entre deux appels, $Call_{Rate}$ le taux d'appels, $Recipient_{Rate}$ le nombre moyen d'appelés par appelant. La gestion des risques permet de prendre en compte les limites en terme de sensibilité et de spécificité de ces critères lors de la détection des attaques.

L'exposition du réseau VoIP dépend directement des mesures de sécurité activées afin de couvrir les vulnérabilités existantes dans le réseau. Nous considérons $SF = \{sf_1, \ldots, sf_n\}$ l'ensemble des contremesures disponibles pour contrer l'attaque a, et $\phi_i(a)$ la valeur normalisée de l'impact

de la contremesure sf_i sur l'attaque a. Nous considérons aussi la fonction $active(sf_i)$ qui retourne une valeur booléenne indiquant si la contremesure sf_i est active ou non. Nous distinguons deux cas d'applications des contremesures : une activation cumulative ou non cumulative car il est possible d'appliquer un ensemble de contremesures sf_i d'une façon simultanée. Par exemple le gestionnaire de risque peut combiner l'authentification et la mise en attente pour protéger le réseau VoIP. L'exposition du réseau VoIP peut alors être définie par l'équation 4.3.

$$\mathcal{E}(a) = 1 - \sum_{sf_i \in SF} \sigma_i(a).active(sf_i) \tag{4.3}$$

La valeur de l'impact $\sigma_i(a)$ qui correspond à la contremesure sf_i dépend des autres contremesures activées. En particulier, l'impact d'une mesure de sécurité peut être réduite si une autre mesure est appliquée dans la même infrastructure VoIP. D'autre part, chaque contremesure sf_i est associée à un coût d'application $cost(sf_i)$ qui représente l'influence de la contremesure sur la disponibilité de service, sa continuité opérationnelle et surtout sa qualité de service.

Notre approche de gestion des risques a pour objectif de maintenir un compromis entre la sécurité et la performance de l'infrastructure VoIP. Ainsi, le gestionnaire de risques doit maintenir le niveau de risque au dessous du seuil défini et en même temps minimiser la somme des coûts induits par les contremesures appliquées. Nous considérons pour cela, deux algorithmes de mitigation, le premier responsable de l'activation de contremesures et le deuxième responsable de leurs désactivations si leur application n'est pas nécessaire. Ils sont régis par la formule 4.4.

$$maintenir(R_{new} \leq R_{threshold}) \;\; et \;\; minimiser(\sum_{sf_i \in SF} cost(sf_i)) \tag{4.4}$$

En développant cette formule, nous obtenons alors :

$$maintenir(\sum_{a \in A} \mathcal{P}(a) \times \mathcal{E}(a) \times \mathcal{C}(a) \leq R_{threshold}) \;\; et \;\; minimiser(\sum_{sf_i \in SF} cost(sf_i)) \tag{4.5}$$

Si la valeur de potentialité $\mathcal{P}(a)$ augmente, le risque augmente. Lorsque le seuil du risque est atteint, le gestionnaire de risque applique l'algorithme de restriction d'accès pour réduire le risque : nous pouvons agir sur les deux paramètres $\mathcal{E}(a)$ et $\mathcal{C}(a)$ la potentialité de l'attaque est une caractéristique intrinsèque qui dépend seulement du type de l'attaque et de sa source. Concernant ce paramètre, sa quantification dépend directement de son observabilité. En effet, nous distinguons par exemple les menaces qui sont caractérisées par une signature (l'ensemble d'actions permettant d'exploiter une vulnérabilité) ou par des anomalies (la variation de la durée moyenne d'appels, le taux de rejets important). Dans notre contexte (prévention de l'attaque SPIT), nous avons adopté une caractérisation de l'attaque par un ensemble de paramètres définis dans l'équation 4.3.

4.2.2 Algorithmes de mitigation

Le gestionnaire de risques exploite deux algorithmes de traitement du risque afin de contrer les attaques potentielles. Ces algorithmes adressent le compromis entre la sécurité et la qualité de service dans les infrastructures VoIP. Le niveau de risque est contrôlé d'une manière continue sur la base des résultats générés par le système de détection. Ces deux algorithmes reposent sur un compromis coût / bénéfice (voir équation 4.4) correspondant aux contremesures susceptibles d'être appliquées. Le gestionnaire de risques détermine les mesures de sécurité à activer ou à désactiver sur la base du rapport de deux critères opposés : la capacité à réduire le risque et leur impact sur la facilité d'utilisation des services VoIP (qualité de service) qui représente la performance du service VoIP.

Algorithme de restriction du risque

L'algorithme de restriction permet d'activer progressivement des mesures de sécurité pour modifier l'exposition de l'infrastructure VoIP lors de l'augmentation du niveau de risque. Chaque fois que le niveau de risque augmente jusqu'à la valeur de seuil, le gestionnaire sélectionne les contremesures appropriées permettant de réduire le risque à une valeur inférieure au seuil et présentant le meilleur rapport coût / bénéfice.

Les différentes étapes sont :

- 1ère étape : après l'évaluation du niveau de risque en se basant sur le modèle quantitatif (voir équation 4.1), le gestionnaire de risques réceptionne l'adresse SIP-URI du suspect qui et à l'origine l'augmentation de risque,
- 2ème étape : le gestionnaire de risques compare la valeur de risque actuelle au seuil de risque acceptable pour l'infrastructure VoIP concerné,
- 3ème étape : si le niveau de risque dépasse le seuil, il choisit un ensemble de contremesures qui réduisent l'exposition contre la source d'attaque et par conséquent, elles réduisent le risque tout en maintenant le niveau de service (voir formule 4.4),
- 4ème étape : il envoie l'ensemble des contremesures à appliquer sous forme d'une politique de sécurité à l'entité responsable.

Algorithme de relâchement du risque

L'algorithme de relaxation permet de désactiver les contremesures afin d'optimiser les performances du service VoIP lorsque le niveau de risque est faible. Ce niveau de risque diminue généralement quand aucun événement correspond à l'attaque a ($\in A$ l'ensemble des attaques VoIP) n'a été observé pendant une période de temps donnée.

- 1ère étape : dès la réception d'une annulation d'attaque (provenant du système de détection d'intrusions), le gestionnaire de risques lance un temporisateur T. Après son expiration, il vérifie s'il y a un événement induisant du risque ou une anomalie détectée par l'entité de détection d'intrusions et déjà survenue pendant la période T,
- 2ème étape : si on a déjà un événement induisant du risque ou une anomalie dans le trafic VoIP, le gestionnaire de risques applique l'algorithme de restriction,
- 3ème étape : sinon on relâche la barrière de sécurité déjà appliquée et on applique la barrière la moins puissante (en terme d'influence sur l'exposition de l'hôte aux menaces),
- 4ème étape : le gestionnaire de risques répète la deuxième étape jusqu'à ce qui on atteigne la contremesure sf_1 où le suspect est considéré comme un utilisateur VoIP normal.

4.2.3 Contremesures

La performance de notre schéma de gestion de risques est dépendante de l'application des contremesures en adéquation avec le niveau de risque. En effet, ces contremesures ont un impact sur l'exposition du système et sur sa performance en terme de continuité opérationnelle et de disponibilité. Dans le contexte de la VoIP, les attaques SPIT peuvent être générées par deux types différents d'entités :

- les bots qui lancent automatiquement des sessions à un ensemble d'adresses et,
- des humains qui effectuent généralement des appels à des fins de marketing.

Pour contrer l'attaque, nous proposons de traiter directement la source de l'attaque. En effet, la solution implantée dans l'infrastructure VoIP identifie la provenance des anomalies ou de la signature d'attaque et lui applique un ensemble de contremesures. Nous avons conçu, pour contrer la source de l'attaque VoIP, plusieurs contremesures qui sont résumées dans le tableau

contremesure	Description fonctionnelle
sf_1	Envoyer un message de signalisation "occupé" à l'appelant
sf_2	Demander la saisie d'un code spécifique
sf_3	Demander une réponse à une question spécifique
sf_4	Mettre l'appelant dans une file d'attente
sf_5	Bloquer systématiquement l'appelant

TABLE 4.1 – Les contremesures considérées pour notre approche

4.1. Elle se caractérise par leur impact sur l'exposition et leur impact (coût d'application) sur la performance du service VoIP. Ces mesures de sécurité sont ordonnées suivant leur impact sur l'exposition des services VoIP et sur la source de l'attaque.

- sf_1 : la première contremesure en termes d'impact sur l'attaque SPIT. Elle consiste à envoyer un message de signalisation "occupé" à l'appelant. Typiquement, le serveur IPBX répond avec un message indiquant que l'appelé est occupé. Par conséquent, si l'appelant correspond à un bot, il y a une forte probabilité qu'il abandonne son destinataire et tente d'établir des sessions d'appels avec d'autres téléphones.
- sf_2 : la deuxième contremesure consiste à demander la saisie d'un code particulier à l'appelant. Le suspect est invité à entrer une suite de chiffres. Cette séquence n'est pas un mot de passe, mais vise à détecter si l'appelant est un robot ou non. Nous considérons que cette mesure de sécurité élimine la plupart des robots utilisés qui sont incapables de taper le code choisi, tandis que la reconnaissance vocale peut cependant être exploitée pour identifier la séquence de chiffres. Cette contremesure est inefficace pour contrer les suspects humains.
- sf_3 : la troisième contremesure qui demande à l'appelant de répondre à une question spécifique. Elle bloque la plupart des robots, qui sont incapables d'interagir avec les messages vocaux, et perturbent les suspects humains parce que, dans ce cas, l'attaquant est obligé de répondre à une question chaque fois qu'il veut mettre en place une session d'appel. Cette mesure de sécurité permet de modifier de manière significative la valeur de l'exposition aux risques, mais a également un impact important sur la performance du service VoIP.
- sf_4 : la quatrième contremesure consiste à mettre l'appelant dans une file d'attente. Cette mesure de sécurité peut être implémentée en utilisant le message de signalisation d'attente et elle permet d'augmenter progressivement le temps d'attente pour l'appelant concerné. Plus l'attaquant tente d'établir des sessions d'appels, plus il doit attendre avant d'établir l'appel. Cette solution est efficace pour contrer les attaquants mais dépend fortement de l'approche choisie et du facteur d'augmentation d'attente choisi.
- sf_5 : la cinquième contremesure en termes d'impact contre l'attaque SPIT. Elle consiste à bloquer systématiquement tous les appels lancés par l'attaquant concerné. Cette mesure de sécurité est appliquée à l'infrastructure VoIP quand la potentialité de l'attaque est élevée et l'attaquant est complètement identifié par le système de détection d'intrusions. Ceci équivaut à mettre l'appelant sur une liste noire. Ainsi, le service VoIP devient indisponible pour lui. Par conséquent, l'exposition des services VoIP pour l'attaquant est réduite et est égale à 0.

4.3 Couplage avec les machines à vecteurs supports

Des nombreux travaux ont porté sur les meilleurs moyens de fournir une détection d'attaque et une protection efficace pour les services VoIP. Ils soutiennent que la détection d'intrusions

est nécessaire pour lutter contre les fraudeurs VoIP. Le principal inconvénient des systèmes de détection d'intrusions (IDS) est cependant leurs faux positifs et faux négatifs, même un faible taux de fausses alarmes rend l'utilisation d'un IDS peu pratique, voire impossible. En outre, la politique de prévention peut avoir un impact significatif sur la disponibilité et la performance du service. Par exemple, si les appels ennuyeux sont en provenance d'un pair correspondant à un fournisseur de VoIP, bloquer tous les appels provenant de ce prestataire va interdire les utilisateurs légaux de faire des appels dans le domaine de protection. Un traitement progressif et dynamique basé sur diverses contremesures est nécessaire.

Dans ce contexte, notre approche de gestion des risques nécessite un système de détection d'intrusions qui détermine s'il y a une attaque ou pas et qui fournit la valeur de potentialité d'attaque d'une manière précise. Pour cela, nous avons étudié le couplage de notre approche avec une méthode de détection à base de machines à vecteurs support (SVM) [68]. L'objectif n'est pas de définir une méthode de détection, mais de montrer comment cette méthode peut être intégrée dans un modèle de risque dynamique. Cette stratégie vise à contrôler dynamiquement l'exposition d'une infrastructure de VoIP en utilisant sur un ensemble des mesures de sécurité progressives, afin de minimiser l'impact sur la performance du service VoIP. Dans [74, 76], les auteurs ont déjà montré que les techniques de détection basées sur SVM sont efficaces et précises dans le suivi et le monitorage du trafic de signalisation VoIP. Dans cette partie, nous rappelons les principes de la détection à base de machines à vecteurs supports et sa mise en œuvre pour la gestion de risques dans une architecture de voix sur IP. Ensuite, nous décrivons comment nous pouvons détecter les attaques et déterminer la source de menace et évaluons les performances de l'approche dans notre contexte.

4.3.1 Principe des SVM

Dans le domaine de l'apprentissage, les machines à vecteurs support SVM sont des modèles d'apprentissage supervisé avec des algorithmes associés qui analysent les données et identifient des modèles de référence (dans le cas de la VoIP, des modèles de trafic), utilisés pour la classification et l'analyse de régression [43, 111, 82]. Le SVM basique prend un ensemble de données d'entrée et prédit, pour chaque entrée, laquelle des deux classes possibles constitue l'entrée, ce qui en fait un classificateur non-probabiliste binaire et linéaire. Étant donné un ensemble d'échantillons d'apprentissage, chacun est marqué comme appartenant à l'une des deux catégories. Un algorithme d'apprentissage à base de SVM construit un modèle qui assigne les nouveaux échantillons dans une catégorie ou dans l'autre. Un modèle SVM est une représentation des données comme points dans l'espace, cartographiées afin que les catégories (ou les classes) distinctes soient divisées par un écart net qui est le plus large que possible. Il est connu pour sa précision et son efficacité dans plusieurs domaines comme la bio informatique, la finance et la recherche d'informations [114] et aussi pour la détection d'intrusions [76].

Il est léger donc adapté à un système de surveillance à l'exécution comme notre approche de gestion des risques. Une classe SVM constitue un cadre géométrique où les statistiques sont mises en correspondance dans un espace caractéristique et des anomalies sont détectées dans les régions moins denses. Les SVMs sont particulièrement adaptés pour l'apprentissage non supervisée où les données propres sont difficiles à obtenir comme c'est le cas pour la VoIP.

Nous considérons pour notre approche une technique de détection SVM à une seule classe (*one class SVM*) [21] : l'idée de base du SVM mono-classe est de séparer les points de l'origine par la plus grande marge possible avec un hyperplan. Alternativement, la formulation d'hypersphères suggère plutôt de trouver la plus petite sphère renfermant les points qui représentent les données. La formulation d'un quart de sphère [63] est la plus adaptée à des fonctions caractéristiques d'un

IDS dont les valeurs sont positives. Le centre de la sphère converge vers la moyenne des valeurs des données. Le rayon joue le rôle d'un seuil qui peut être utilisé pour contrôler la spécificité et la sensibilité de la détection par anomalies. On définit le score d'anomalies d'un point est la distance de ce point par rapport au centre, c'est un paramètre important dans notre approche, car elle détermine la potentialité d'une attaque.

Mathématiquement, le principe des SVM peut être défini comme suit : soit un couple $S = (\vec{x}_l, y_l)$ où $y_l \in [-1, +1]$ qui désigne la classification correcte des données d'apprentissage, la méthode SVM tente de distinguer entre les deux classes par le moyen d'un hyperplan de séparation qui a comme équation $\vec{w}.\vec{x} + b = 0$. Si les données d'apprentissage sont linéairement séparables, la solution consiste à maximiser la marge entre les deux hyperplans donnés respectivement par les équations 4.6 et 4.7.

$$\vec{w}.\vec{x} + b = +1 \tag{4.6}$$

$$\vec{w}.\vec{x} + b = -1 \tag{4.7}$$

Le résultat du problème quadratique, où les conditions 4.6 et 4.7 sont agrégées peut être formulé par :

Trouver le vecteur $\vec{w}$ et la valeur b afin de minimiser $1/2\vec{w}.\vec{w}$ de sorte que $y_l(\vec{w}.\vec{x}_l + b > 1 \forall (\vec{x}_l, y_l)$

4.3.2 Identification des attaques

La détection d'anomalies consiste à analyser les statistiques (ou les fonctionnalités) fournies par un système de surveillance afin de révéler les situations anormales. La tâche du système de surveillance consiste à extraire les statistiques prédéfinies à partir des données brutes. L'analyse de ces statistiques se base sur un cadre mathématique et est préemptée par une période d'apprentissage où un modèle du trafic normal est construit. La stratégie de surveillance repose sur la définition d'un ensemble de sondes (ou caractéristiques) et de leur calcul périodique afin de profiler une source de données. L'extraction des données caractéristiques est une étape critique dans la détection d'intrusions car elle vise à déterminer des évidences qui peuvent être le plus utile pour l'analyse du trafic à l'exécution. Nous considérons trois importantes sources de données dans un cadre SIP/VoIP : le trafic SIP, l'historique des serveurs, les statistiques, et les enregistrements de facturation. Nous décrivons, dans cette section, un modèle de détection d'anomalies pour les réseaux et les services VoIP.

Les sources d'informations dans l'infrastructure VoIP

Plusieurs sources de données sont disponibles afin de détecter des anomalies dans une architecture VoIP. Ces sources comprennent :

- le trafic du réseau, en particulier les protocoles qui sont indispensables pour le fonctionnement normal des appels VoIP (SIP, RTP, RTCP, DNS). C'est la source des données typiques pour la détection d'anomalies basée sur le réseau. Dans [76], les auteurs ont défini 38 statistiques (ou fonctionnalités) pour le trafic SIP.
- les données historique des serveurs VoIP et des systèmes d'exploitation sous-jacents. C'est la source de données typique pour une détection d'anomalies sur une machine hôte.
- les statistiques fournies par les serveurs VoIP. En général, ces statistiques dépendent de la conception interne de ces serveurs. Par exemple, le proxy SIP OpenSIPS [20] fournit plusieurs

20. Serveur proxy OPENSIPS, www.opensips.org/

groupes de fonctions caractéristiques : le cœur, la mémoire, les statistiques sans état, les statistiques des transactions, l'emplacement et l'enregistrement de l'utilisateur.
- les enregistrements détaillés des appels (CDR) : La surveillance de cette source d'information est particulièrement importante pour la détection des fraudes et le traitement du SPIT.

Les paramètres caractérisant les appels

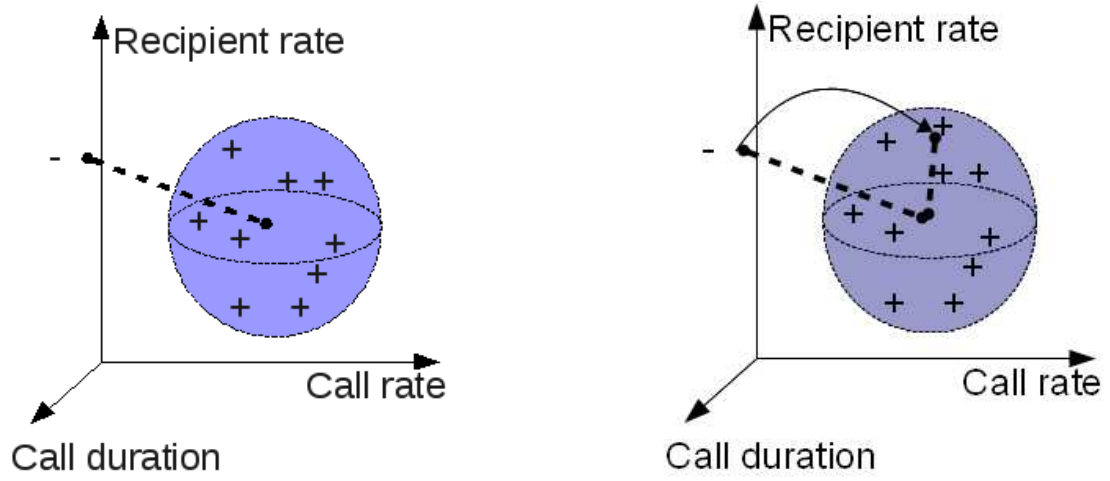
Les enregistrements détaillés d'appels (CDR) sont une source importante d'information pour le profilage d'utilisateurs, du groupe d'utilisateurs et des paramètres globaux caractérisant le trafic. Nous extrayons à partir d'un ensemble de CDRs les caractéristiques suivantes :

- le taux de rejet d'appels : c'est le pourcentage d'appels dont le statut de fin est soit "échec" ou "occupé",
- le taux d'appelant par appel : c'est le rapport de nombre de destinataires distinctes par rapport au nombre total d'appels,
- le taux distinct d'appelants : est le rapport des appelants distincts par rapport au nombre total d'appels,
- la durée moyenne de facturation : la durée de facturation est le temps entre le 200 OK et le message BYE dans un scénario de signalisation d'appel avec succès,
- la durée moyenne d'appel : La durée de l'appel est le temps entre l'invitation et le message BYE.
- le taux d'appel : c'est la fréquence des appels dans un délai de temps.
- les applications cibles : l'application cible est la dernière application réalisée lors de l'appel (par exemple le transfert d'appel, la musique d'attente, la messagerie vocale),
- le contexte d'un appel : le contexte d'un appel révèle la classe de l'extension composée (par exemple entrant/sortant, local/département/international).

Cet ensemble de données caractéristiques est général et il faut l'adapter suivant le contexte (fournisseur d'accès, entreprise, etc). Par exemple, le taux d'appel est exclu au niveau de l'utilisateur, la durée moyenne d'appel devrait être étendue au niveau global par la définition d'une répartition sur plusieurs intervalles de durée d'appel. Nous suggérons la création de profils d'appels en fonction de plusieurs paramètres contextuels.

L'identification de la source d'attaques

L'identification des sources d'attaque est basée sur le fait que la suppression de leurs effets déplace le point d'anomalie pour qu'il se trouve à nouveau dans la région normale. Pour donner un exemple, supposons que la liste des CDR est représentée uniquement par la durée moyenne des appels. Si la durée moyenne d'un appel d'une liste de CDR se révèle petite en la comparant à la moyenne calculée précédemment au cours de l'apprentissage, cette situation est considérée comme malveillante. La cause de cette anomalie est une ou plusieurs sources qui ont généré les appels de courte durée. Si nous supprimons donc tous les CDR de l'une de ces sources, la moyenne de durée des appels doit être plus proche de la valeur initiale (celle du modèle du trafic normal). Ceci est décrit par l'algorithme suivant :



(a) Fonctions représentatives en 3D (b) Élimination des sources d'attaques

FIGURE 4.3 – Processus de détection de sources d'attaques par SVM

1. ordonner toutes les sources d'appels VoIP par ordre croissant de la durée moyenne d'appel,
2. supprimer la source de l'appel dont le rang est le plus haut et la mettre dans la liste des sources suspectes,
3. recalculer la nouvelle moyenne globale de la durée d'appels,
4. tester la nouvelle configuration des point de données avec le détecteur d'anomalie,
5. si le nouveau point de données est considéré comme normal (par rapport au modèle de normalité) donc retourner la liste des sources suspectes, sinon revenir à la première étape.

Algorithme 1: Algorithme d'identification de la source d'attaque

En général, nous considérons que le score d'anomalies à un instant donné se compose des contributions individuelles des sources. Ensuite, les sources d'appels sont ordonnées suivant un ordre décroissant de leurs scores. Après, nous éliminons les sources les mieux classés (voir figure 4.3(b)). Ces sources éliminées sont considérées comme la cause de l'anomalie. Ce mécanisme d'identification, de détection et de classification des sources d'appels peut être visualisé en 2D ou en 3D (voir figure 4.3(a)).

4.3.3 Extension du modèle

Nous avons étendu le modèle de gestion de risques pour qu'il supporte les nouveaux paramètres de détection d'intrusions. Le système de détection d'intrusions délivre au gestionnaire de risques une liste de sources d'appels malveillants. Nous avons étendu le modèle pour la mitigation des attaques SPIT qui peuvent provenir d'un utilisateur dans un réseau VoIP où le blocage de tous les appels du fournisseur n'est pas une solution pratique.

Nous définissons le niveau de risque d'une source d'appel (contenant à la fois les utilisateurs

légaux, les utilisateurs malveillants ou les robots) par le nombre d'appels SPIT qui réussissent à atteindre les utilisateurs finaux et, ainsi, génèrent une gène pour eux. La potentialité $\mathcal{P}(a)$ définit, dans ce contexte, l'intensité de l'attaque SPIT, c'est-à-dire le nombre d'appels non sollicités par unité de temps, $\mathcal{E}(a)$ définit l'ensemble des contrôles de sécurité appliqués avant d'établir les appels provenant de la source malveillante. Nous présentons l'exposition des services VoIP contre la source malveillante (c) par la probabilité Pr_m que l'appelant malveillant contourne l'ensemble des contremesures appliquées dans l'infrastructure VoIP concernée. $\mathcal{C}(a)$ définit la gène côté de l'utilisateur final que nous supposons constante pour chaque appel SPIT réussi.

L'application d'un ensemble de contremesures S pour les appels entrants à partir d'une source d'appel malveillant impose un coût supplémentaire pour les appels entrants légaux à la même source. Ce coût peut être l'ajout d'un temps de retard (d) à l'établissement d'appel : l'établissement de la session ne se réalise qu'après la vérification et l'application réussie de la contremesure par la source suspecte. Nous définissons Pr_h comme la probabilité qu'un utilisateur légal contourne avec succès l'ensemble des contremesures, et L_{legal} comme le coût des appels légaux échoués. Soit $c \in C$ une source d'appel générant N_c appels légaux, le coût est défini par l'équation 4.8.

$$\mathcal{L} = \sum_{a \in A} N_a \times (Pr_h(S) \times d(S) + (1 - Pr_h(S)) \times L_{legal}) \tag{4.8}$$

$\mathcal{L}$ quantifie le coût généré suite à l'application de l'ensemble de contremesures S. Ce coût est défini par deux quantités. la première ($Pr_h(S) \times d(S)$) quantifie le délai moyen supplémentaire ajouté à la communication et la deuxième ($(1 - Pr_h(S)) \times L_{legal}$) quantifie le coût moyen induit par les appels légaux échoués.

Comme mentionné précédemment dans le modèle de gestion de risque (voir équation 4.4), notre objectif consiste à minimiser les coûts induits par l'application des contremesures tout en réduisant le risque à un niveau acceptable. La raison du choix de deux variables différentes (Pr_h) et (Pr_m), caractérisant l'exposition de l'ouverture de services VoIP pour les appels légaux et illégaux, est que les appels légaux sont généralement générés par les humains et les appels non sollicités sont généralement générés par des bots. Notre modèle peut être étendu pour tenir compte de l'identité de l'appelé ainsi que son rôle dans l'entreprise. Par exemple, atteindre le numéro de téléphone du directeur devrait être plus difficile que d'atteindre un de ses employés.

4.4 Architecture fonctionnelle

L'architecture qui prend en charge notre solution de gestion des risques vise à renforcer le couplage entre la détection des risques en s'appuyant sur la technique de SVM et l'application des contremesures. L'objectif est de prendre en compte à un stade précoce les résultats du système de détection en vue de fournir un traitement progressif et continu des risques. Cette architecture est déployée au sein d'un serveur IPBX VoIP.

L'architecture de notre solution est représentée sur la figure 4.4 et est composée des quatres composants principaux suivants :

- Moniteur de trafic : Le rôle de ce composant est d'analyser le trafic et les diverses sources de données, la collecte des sources d'information, la quantification des paramètres caractérisant le trafic tels que le taux d'appel, le taux de rejet d'appel, la durée moyenne d'un appel, le nombre d'appelé par appelant, la durée moyenne entre deux appels consécutifs. L'estimation de ces paramètres est basée sur les sources de données citées dans la sous-section 4.3.2. Le moniteur d'appel caractérise le trafic par plusieurs paramètres et envoie tous ces paramètres à un détecteur de menaces pour le traitement.

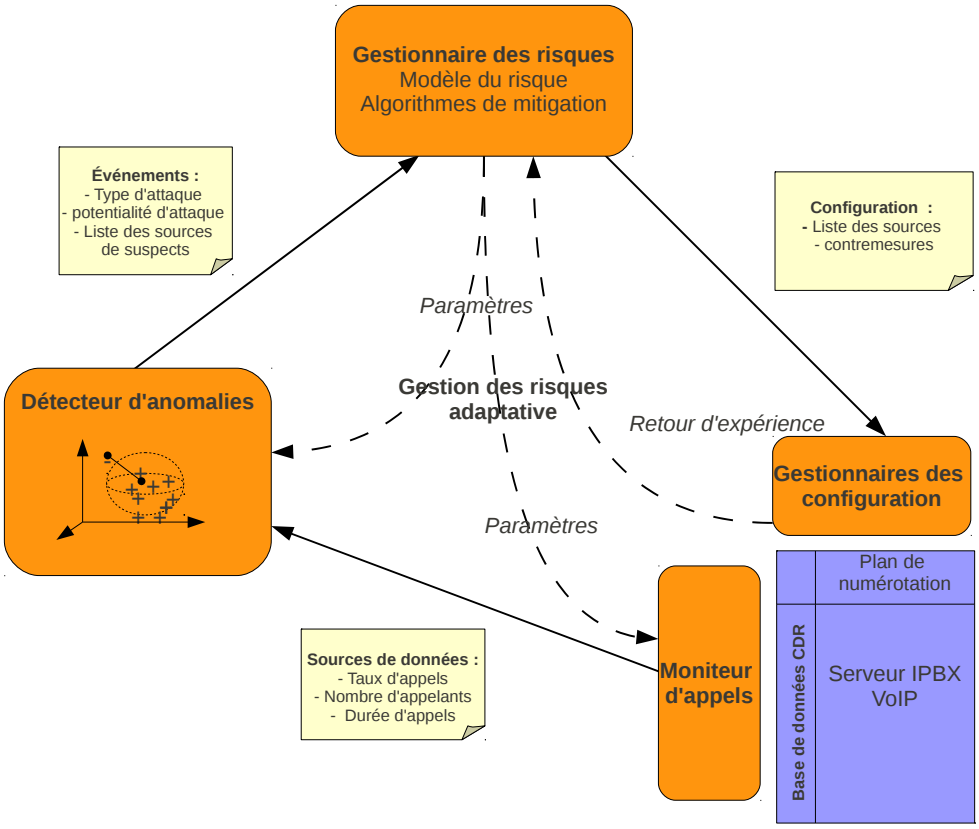
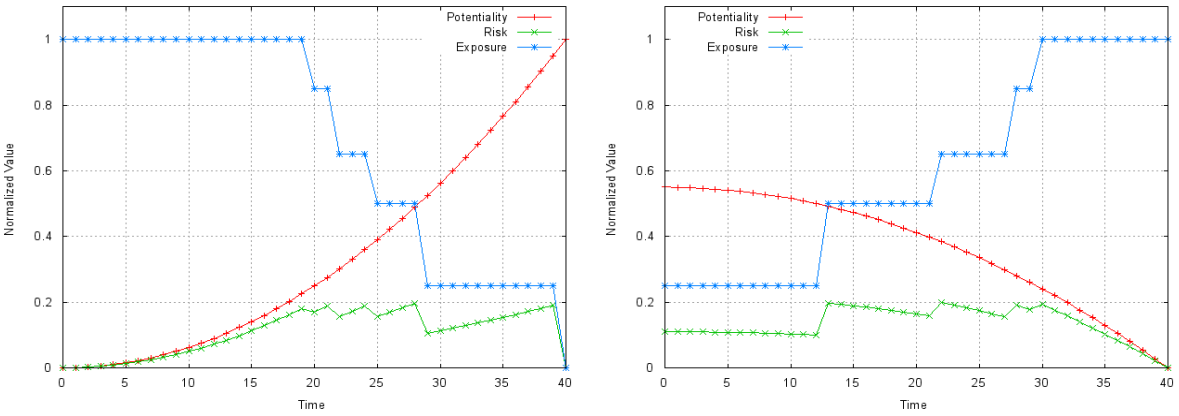


FIGURE 4.4 – Architecture de notre approche de gestion de risque

– Détecteur d'anomalies : le rôle de ce composant est d'identifier les intrusions. Il identifie le type de l'attaque, sa potentialité et la liste des sources suspectes. Il est basé sur la détection des anomalies soit par signature ou par détection. Son rôle est fondamental, car il fournit une liste des suspects possibles dont le trafic VoIP génère des anomalies par rapport au modèle du trafic normal. Typiquement, la valeur seuil permet de fixer à quel niveau de potentialité, un trafic est considéré comme malveillant. Nous avons développé une solution qui utilise la détection d'anomalies sur la base de SVM [68]. Le détecteur de menace, après, envoie tous ces paramètres au gestionnaire de risques ainsi que la liste des suspects qui génèrent le risque, le type d'attaque et sa potentialité. Le gestionnaire du risque utilise, ensuite, la potentialité pour calculer le risque et, en se basant sur cette valeur, traite les appels provenant des sources suspectes.
– Gestionnaire de risques : c'est le composant principal de l'architecture ; son rôle consiste en l'estimation et le traitement des risques. Ainsi, il est responsable des deux étapes importantes du processus de gestion des risques. Le gestionnaire des risques quantifie le risque en utilisant le modèle quantitatif. A chaque fois que le détecteur de menace envoie un changement de potentialité d'attaques, le gestionnaire de risques réévalue la valeur du risque et requantifie les deux autres paramètres du risque à savoir l'exposition contre l'attaque et les conséquences subies. L'exposition de l'infrastructure VoIP dépend fortement des contremesures appliquées. Sur la base de la valeur du risque, le gestionnaire du risque applique l'un des deux algorithmes de mitigation, soit l'algorithme de restriction, si le risque dépasse le seuil, soit l'algorithme de relâchement si le risque est inférieur au seuil. Cette gestion des



(a) Impact de la restriction sur le niveau de risque (b) Impact du relâchement sur le niveau de risque

FIGURE 4.5 – Impact des algorithmes de mitigation sur le traitement du risque

contremesures correspond à l'étape de traitement du risque. Le gestionnaire des risques est responsable de la prise des décisions soit pour activer ou pour désactiver des contremesures qui seront envoyées au gestionnaire de configuration. Il envoie avec la liste des sources d'appels malicieux la liste des contremesures associées.

– Gestionnaire de configurations de sécurité : le rôle de ce composant est de transmettre les décisions prises par le gestionnaire du risque. Ainsi, il convertit ces décisions en une politique de sécurité et communique cette politique à divers équipements VoIP. Le gestionnaire de configuration transmet les décisions prises par le gestionnaire de risques dans la plate-forme VoIP. Il peut après fournir un rapport de retour d'expérience sur l'application des contremesures. Le système de gestion de configurations est généralement composé d'un serveur de configuration et des agents déployés sur les équipements VoIP. Le traitement de risque concerne tous les équipements du réseau qui peuvent influer sur l'exposition de l'infrastructure VoIP contre le risque, ces éléments comprennent les téléphones VoIP et les serveurs VoIP, mais aussi le protocole de signalisation et de transfert de médias, les services support tels que DNS, TFTP et RADIUS et les équipements de sécurité tels que les pare-feux applicatifs.

4.5 Résultats expérimentaux

Dans cette section, nous présentons les différentes expérimentations faites pour évaluer les performances de notre solution de gestion des risques. Nous nous intéressons en premier lieu à l'évaluation des algorithmes de mitigation à savoir l'algorithme de restriction de risques et celui de relâchement, ainsi que leurs impacts sur le traitement du risque. Après, nous présentons une série d'expérimentations qui évalue le couplage avec la détection d'anomalies à base de SVM. On s'intéresse surtout aux trois paramètres suivants : le seuil de risque, la sensibilité et la spécificité.

4.5.1 Évaluation des algorithmes de mitigation

Afin d'évaluer les performances de notre approche de gestion des risques, nous avons développé un prototype du gestionnaire des risques et effectué un ensemble d'expériences. Ce prototype met en œuvre en C++ le modèle du risque proposé et les algorithmes de mitigation de risques. Il

prend en entrée les résultats du système de détection d'intrusions et fournit en sortie les mesures de sécurité pour être activées ou désactivées par le système de configuration. Nous avons élaboré différents scénarios d'attaque, afin d'évaluer le comportement et les avantages de notre solution de gestion des risques en comparaison avec d'autres stratégies traditionnelles.

Restriction du risque

Dans la première série d'expériences, nous nous sommes intéressés à l'évaluation de l'algorithme de restriction de risques et la détermination de la façon dont il influe sur le niveau de risque dans le réseau VoIP. Dans nos expériences, nous avons fait varier les paramètres de détection qui se répercutent sur la potentialité d'attaque perçue par le gestionnaire du risque. La figure 4.5(a) représente l'évolution de la potentialité d'attaque, le niveau de risque calculé par le gestionnaire du risque et l'exposition du service VoIP au cours du temps. On peut d'abord observer sur cette figure que la potentialité de l'attaque est initialement égale à zéro. Par conséquent, le niveau de risque $\mathcal{R}$ est nulle et l'exposition $\mathcal{E}(a)$ est maximale. Comme la potentialité augmente au cours du temps, le niveau de risque $\mathcal{R}$ augmente aussi. À l'instant $t = 19$, le niveau de risque atteint la valeur du seuil défini $R_{seuil} = 0.2$, le gestionnaire de risque active la première mesure de sécurité sf_1 qui utilise le message de signalisation "occupé" contre la source de la menace. L'activation permet de réduire l'exposition $\mathcal{E}(a)$ à 0.85 et d'atténuer le risque à 0.19. Alors que la première mesure de sécurité est activée, la potentialité de l'attaque (niveau de l'attaque SPIT) continue de croître jusqu'à ce que le niveau de risque atteint de nouveau la valeur de seuil R_{seuil} à l'instant $t = 21$. Le gestionnaire de risque active la deuxième contremesure sf_1 qui consiste à demander à l'appelant de taper un code donné. La valeur de l'exposition converge vers 0.65 et le niveau de risque, donc, est réduit à 0.17. Nous pouvons, ainsi, observer comment le gestionnaire de risque réussit à maintenir le niveau de risque au dessous du risque seuil : les contremesures sont successivement activées à chaque fois que la potentialité augmente.

Relâchement du risque

Dans une deuxième série d'expériences, nous avons analysé la capacité de la solution de gestion des risques à relâcher le niveau de risque quand la potentialité d'une attaque diminue. L'objectif consiste à optimiser les performances et la facilité d'utilisation du service VoIP service quand la potentialité d'une attaque est faible. La figure 4.5(b) traduit les mêmes variables (la potentialité, le niveau de risque et l'exposition) que la figure 4.5(a). Mais dans ce cas, la potentialité diminue au cours du temps. A l'instant $t = 0$, la contremesure sf_5 est mise en place, le niveau de risque R est égal à 0,12 et l'exposition $\mathcal{E}(a)$ est égale à 0,25. A l'instant $t = 13$, le gestionnaire de risque désactive la contremesure actuelle sf_5 et active la contremesure sf_4. Ainsi, Il réduit l'impact des contremesures sur les performances du réseau et maintient au même temps le niveau de risque au dessous du seuil R_{seuil}, l'exposition $\mathcal{E}(a)$ atteint la valeur de 0,5 et la potentialité $\mathcal{P}(a)$ continue à diminuer. Ainsi, le niveau de risque diminue jusqu'à atteindre la valeur de 0,15. A l'instant $t = 21$, le gestionnaire des risques désactive une nouvelle fois la contremesure sf_4 appliquée, applique une contremesure moins puissante sf_3 et relâche le niveau de risque afin de maximiser les performances des services. les contremesures sont successivement désactivées à chaque fois que la potentialité diminue. Enfin, à l'instant $t = 29$, le système relâche toutes les contremesures parce que le niveau de risque est maintenu à une valeur inférieure à R_{seuil}.

Les deux figures 4.5(a) et 4.5(b) montrent clairement comment l'exposition des services VoIP peut être modifiée ou améliorée par le gestionnaire de risques en se basant sur l'activation et la désactivation des mesures de sécurité appliquées. L'ensemble des résultats expérimentaux

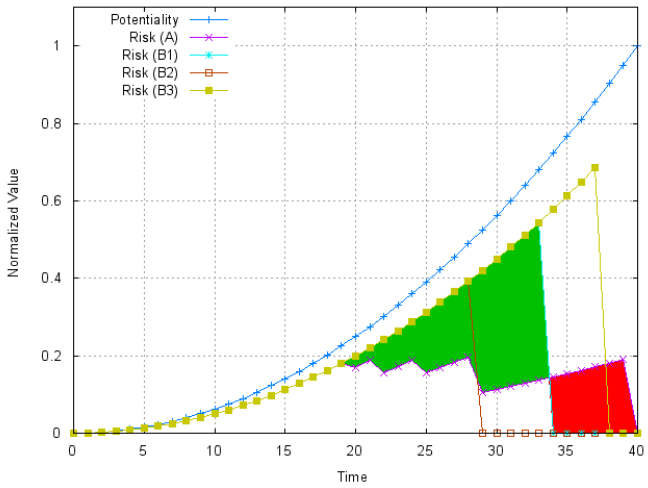


FIGURE 4.6 – Comparaison entre notre approche et d'autres stratégies traditionnelles

confirme que la solution est capable de gérer le niveau de risque d'une manière dynamique, flexible et progressive. Le gestionnaire de risque choisit les contremesures les plus appropriées afin d'optimiser le compromis entre le risque et l'impact sur les performances des services VoIP.

Comparaison avec des stratégies traditionnelles

Dans une troisième série d'expériences, nous nous sommes intéressés à comparer notre solution avec des stratégies traditionnelles et à quantifier les avantages et les limites de l'intégration du modèle de risque. La figure 4.6 décrit le niveau de risque de notre solution noté A et le niveau de risque des trois stratégies classiques notées B_1, B_2 et B_3. Ces stratégies correspondent au cas d'un système de détection, sans un modèle de risque explicite, c'est-à-dire le niveau de risque est seulement basé sur la potentialité de l'attaque. La première stratégie B_1 fournit un taux élevé de vrai positif (haute sensibilité) et consiste à bloquer les communications VoIP d'un appelant donné dès que sa potentialité d'attaque dépasse une valeur de 0,4. La deuxième stratégie B_2 consiste à bloquer la communication quand la potentialité atteint 0,5, cette solution fournit une sensibilité et spécificité moyennes et est considérée par [58] comme une méthode de détection de référence. La dernière stratégie B_3 réduit le taux de faux positif (spécificité élevée) et consiste à bloquer les communications de l'appelant lorsque la potentialité d'attaque est supérieure à 0,7.

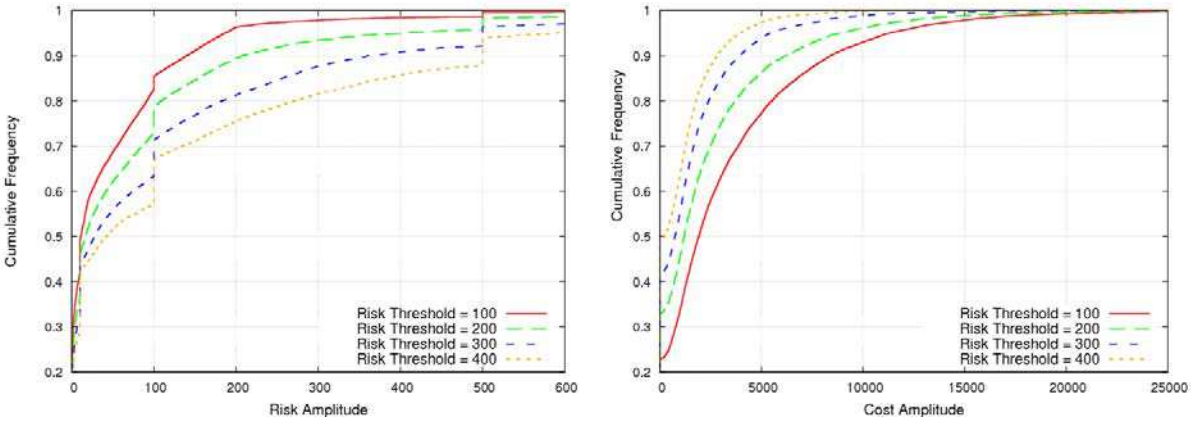
La comparaison de notre solution A avec la première stratégie B_1 montre l'avantage en termes de niveau de risque qui est faible. En effet, la stratégie B_1 rejette rapidement les communications de l'appelant (l'instant $t = 29$), et par conséquent, son niveau de risque est réduit à zéro. En revanche, notre approche offre un bénéfice en terme de disponibilité. Le bénéfice en termes de disponibilité du service est assez élevé : notre approche de gestion de risques permet de maintenir une continuité opérationnelle plus longue. D'après le graphe, le blocage de la communication se fait à l'instant $t = 40$.

Lorsque l'on compare notre solution avec la stratégie B_3, nous nous attendions à un bénéfice significatif en termes de risque, ce qui était le cas. Nous observons clairement l'avantage de notre solution de gestion des risques qui est représentée par la zone verte sur le graphe. Nous avons quantifié un bénéfice moyen de 32%. Ce bénéfice correspond à des appels SPIT évités. L'inconvénient majeur de la stratégie B_3 est que le niveau de risque peut atteindre une valeur élevée avant que l'appelant soit considéré comme un attaquant par le système de détection. En

particulier, le niveau de risque maximum généré par notre solution est relativement faible (la valeur de R_{seuil} est fixée à 0.2), alors qu'il peut être significativement plus élevé par rapport à la stratégie B_3 (un seuil de 0.7). En terme de disponibilité, les deux approches A et B_3 offrent des performances similaires, mais l'exposition contre la source d'attaque est progressivement limitée par notre approche de gestion des risques.

La comparaison avec la stratégie de B_2 offre également des résultats intéressants. L'avantage de notre approche par rapport à B_2 en termes de disponibilité du service est importante. La stratégie B_2 bloque la communication VoIP de l'appelant à l'instant $t = 29$, alors que notre solution A maintient le service VoIP jusqu'à l'instant $t = 40$. Nous avons quantifié un bénéfice en termes d'appels SPIT évités de près de 14% au cours de notre série d'expériences. Le niveau de risque moyen est similaire pour les deux approches A et B_2. Toutefois, le niveau de risque maximal de A est de 0,2 et de 0,5 pour la stratégie B_2.

4.5.2 Influence de la valeur seuil



(a) Impact du seuil sur le traitement du risque

(b) Impact du seuil sur les coûts générés

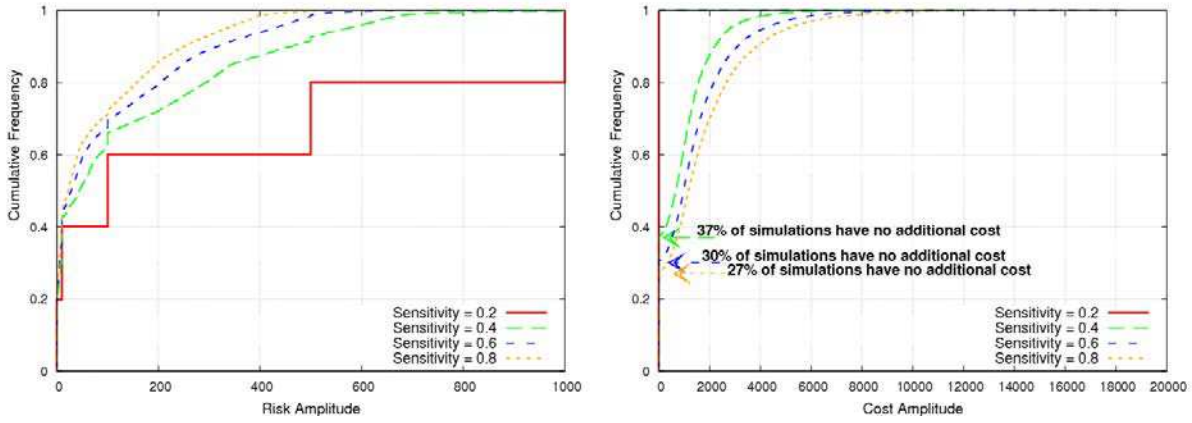
Nous avons élaboré de façon complémentaire une série de simulations afin d'analyser les performances de notre approche en termes de sensibilité, spécificité et pour évaluer l'impact de la valeur seuil sur notre solution de risque. Nous avons fixé le taux d'arrivée d'appels par une loi de Poisson avec une moyenne de 100 appels par unité de temps, la durée d'un appel par une loi exponentielle et une moyenne de 10 secondes, une intensité d'appels SPIT croissante qui varie entre 10, 100, 500 et 1000 appels SPIT par unité de temps et nous avons fixé une valeur constante de 60 secondes pour comptabiliser le coût induit par les appels normaux bloqués.

Nous définissons 5 contremesures différentes où chaque contremesure est caractérisée par trois variables : le délai d, Pr_m la probabilité qu'un appel malicieux contourne les contremesures, Pr_h la probabilité qu'un appel normal passe les contremesures ; elle est définie, pour toute contremesure, par une distribution uniforme d'intervalle entre [0,8 ; 1]. Nous définissons aussi 5 seuils de détection différents avec différentes valeurs de sensibilité / spécificité. Ces paramètres nous donnent 30 différents scénarios de simulation (intensités d'attaque normale et différente par rapport aux seuils de détection différentes). Nous réalisons 10.000 simulations de Monte Carlo par scénario ce qui permet d'obtenir un taux d'erreur de moins de 5%.

Le paramètre du seuil de risque représente le degré de l'ouverture du système dans notre modèle de gestion de risques. Un seuil de risque élevé signifie que l'infrastructure exige de minimiser les coûts induits d'application des contremesures, en dépit d'un niveau de risque élevé. Un

seuil faible de risque signifie que l'infrastructure accepte d'induire des coûts supplémentaires qui touchent à la performance des services afin de protéger le système. Les figures 4.7(a) et 4.7(b) représentent quatre scénarios avec des seuils de risque différents (100, 200, 300 et 400) de nombre d'appels SPIT acceptés par unité de temps. Évidemment, un seuil de risque fixé à 100 appels offre les meilleures performances en termes de risque et le pire en termes de coûts induits. Un seuil de risque de 200 appels ou 300 appels est un bon compromis entre le niveau de risque et le coût sur les performances. La courbe en escalier (une sensibilité égale à 0.2) à des risques égaux à 10, 100 et 500, correspond à une série d'expériences où les attaques respectives sont perçues comme acceptables. Par conséquent, aucune contremesure n'est activée afin de réduire le niveau de risque.

4.5.3 Évaluation du couplage avec SVM



(c) L'évolution de risque en fonction de la sensibilité (d) L'évolution des coûts en fonction de la sensibilité

FIGURE 4.7 – Évaluation de performances en terme de sensibilité de la détection

Il est important d'étudier les performances de notre approche en fonction des caractéristiques de détection d'anomalies. Un système de détection d'intrusions est caractérisé par deux paramètres : la sensibilité et la spécificité qui sont généralement dépendantes l'une de l'autre. Par définition, la sensibilité est le taux de vrais positifs, elle est définie aussi par les unités de temps où il y a un trafic anormal qui est correctement détecté. La spécificité est le taux de vrais négatifs ou les unités de temps où le trafic est normal qui sont correctement détectés.

Dans notre cas, le détecteur d'anomalie attribue une valeur de potentialité à une situation plutôt que binaire. Ainsi, la sensibilité est la probabilité que le détecteur d'anomalies attribue une valeur élevées de potentialité à une situation d'attaque. Respectivement, la spécificité est la probabilité d'attribuer des valeurs de potentialité faibles quand il n'y a pas d'attaques. Les figures 4.7(c) et 4.7(d) représentent quatre scénarios dont les valeurs de sensibilité sont (0,2, 0,4, 0,6 et 0,8) avec une valeur de spécificité constante égale à 0,8. Pour une spécificité égale à 0,2, toutes les attaques sont détectées mais considérées comme un risque tolérable ce qui génère un risque élevé, en revanche, aucun coût n'est induit pour ce scénario parce que les mesures de sécurité ne sont pas appliquées. À l'autre extrême, pour une spécificité égale 0,8, le niveau de risque est réduit et il y n'a pas de coûts de contremesures induits pour 37% des simulations. Le paramètre de spécificité n'a aucune influence sur le risque du système. Toutefois, une mauvaise valeur de spécificité conduit à évaluer une situation normale comme une situation de risque ce

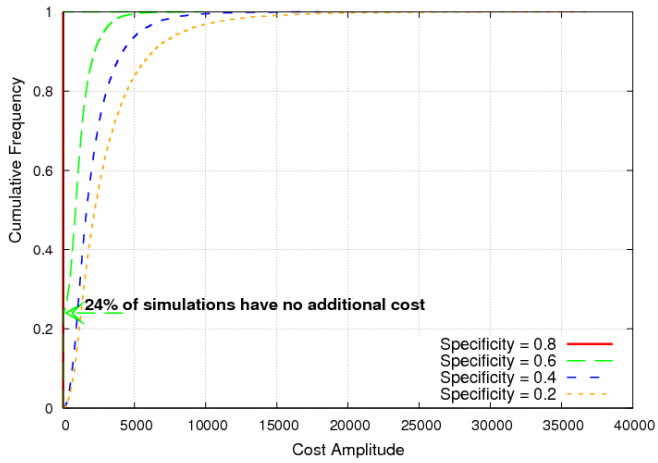


FIGURE 4.8 – Évaluation de performances en terme de spécificité de la détection

qui génère des coûts induits inutiles. Ainsi, nous évaluons les performances dans le cas où il n'y a pas d'attaques. La figure 4.7(d) illustre quatre scénarios avec des valeurs de spécificité différentes (0,2, 0,4, 0,6 et 0,8). Pour une spécificité égale à 0,8, il n'y a pas des coûts induits tant qu'aucune des situations normales n'est détectée comme une attaque. Une valeur de spécificité moyenne, qui est généralement inacceptable pour un système de détection d'intrusions réel (par exemple une spécificité égale à 0,6), réduit les coûts d'applications de contremesures à zéro pour 24% des simulations (voir la figure 4.7(d)). En plus comme indiqué, un système de détection d'anomalies ayant une valeur de spécificité très modérée altère les performances de notre approche.

4.6 Synthèse

Dans ce chapitre, nous avons proposé une solution de gestion de risques pour adapter automatiquement et d'une manière continue l'exposition des équipements VoIP contre les attaques. Cette exposition est contrôlée par l'activation et la désactivation de contremesures d'une manière progressive fondée sur un modèle de risque. Ceci permet de prévenir les risques potentiels, tout en maintenant la qualité du service de téléphonie sur IP. Nous avons conçu une architecture composée d'un système de détection d'intrusions, un gestionnaire des risques et un système de configuration. Cette architecture met en œuvre le couplage entre la détection d'attaques et l'application des contremesures en s'appuyant sur un modèle de risque utilisant le formalisme rheostat. Nous présentons l'architecture technique de notre solution ainsi que l'implantation des différents composants dans la partie 8. Nous avons aussi décrit le couplage de notre approche avec une méthode de détection basée sur SVM. Cette intégration contribue à améliorer la réponse du système contre les attaques SPIT. Nous avons montré comment les algorithmes de restriction et de relaxation de risque permettent de fournir une réponse adéquate et progressive à l'attaque SPIT en activant ou désactivant des contremesures au cours de l'exécution. Cependant, notre stratégie adaptative de risque traite uniquement l'attaque SPIT. Par ailleurs, la valeur des paramètres comme le seuil de risque se détermine en fonction de la criticité de l'infrastructure et du contexte d'utilisation des services VoIP.

Dans le chapitre suivant, nous allons étendre notre modèle de risques aux autres attaques VoIP et décrire un mécanisme d'auto-configuration pour le raffinement de certains paramètres.

Chapitre 5

Modèle de risques étendu et paramétrisation

Sommaire

5.1 Introduction

Les attaques VoIP sont très différentes dans leur objectif et leur nature, et même dans leurs spécifications techniques. Notre approche de gestion des risques, présentée dans le chapitre précédent, se focalisait uniquement sur les attaques SPIT ce qui limite la portée de la solution de sécurité pour protéger les infrastructures VoIP.

Afin de répondre à cette limite, nous proposons d'étendre notre modèle de risque aux autres attaques VoIP. Notre approche vise à adapter en permanence l'exposition du service VoIP en fonction de la potentialité des menaces en utilisant un ensemble de contremesures qui peuvent être activées d'une façon dynamique. Les avantages de cette automatisation sont fortement dépendantes des propriétés des attaques VoIP. En particulier, nous allons déterminer les performances de notre solution par rapport aux propriétés d'observabilité des attaques VoIP. Pour cela, nous allons évaluer notre stratégie de gestion des risques par un ensemble d'expériences afin d'évaluer la performance de notre solution par rapport au risque, au coût induit par l'application des contremesures et à la disponibilité de service.

Nous nous intéressons ensuite à la paramétrisation de notre solution de gestion. Les modèles mathématiques qui soutiennent la gestion de risques peuvent être qualitatifs ou quantitatifs [17]. Ces modèles permettent d'améliorer la performance des politiques de sécurité, dans les deux cas ils souffrent souvent de leur complexité [115]. Ils peuvent en effet compter un nombre élevé de paramètres à configurer. Aussi, nous proposons un mécanisme d'auto-configuration pour soutenir notre stratégie de gestion de risques dans les infrastructures VoIP. Cette approche vise à simplifier la configuration de certains paramètres du modèle de risques par leur raffinement en s'appuyant sur un mécanisme de retour d'expérience.

5.2 Modélisation et classification des attaques VoIP

Notre travail consiste d'abord en la classification et la modélisation des attaques pour adapter le modèle de risque. Les performances de notre stratégie de gestion des risques dépendent fortement de leurs propriétés d'observabilité [27]. Nous avons classé les attaques VoIP en trois classes d'observabilité : les attaques observables par les signatures (voir figure 5.1), les attaques observables par des anomalies (voir figure 5.2) et les attaques qui sont considérées comme non observables ou difficilement détectables. Cette classification sert comme support pour l'évaluation de notre modèle de gestion de risques à l'égard de la propriété d'observabilité d'attaque et elle permet d'étendre sa capacité à protéger le réseau par l'application de contremesures d'une manière progressive et efficace. L'observabilité dépend des méthodes de détection qui sont disponibles pour identifier une menace donnée. Nous présentons, pour chaque classe d'attaque, une formule mathématique modélisant l'observabilité des attaques VoIP et le calcul de la potentialité.

5.2.1 Attaques observables par signature

La première classe d'observabilité inclut les attaques qui sont identifiables en fonction de leurs signatures connues. Dans ce cas, la signature $signature(a)$ d'une attaque a est définie par une séquence d'événements $\{s_1, s_2, \ldots, s_n\}$ qui peut être soit ordonnée, partiellement ordonnée, ou non ordonnée. Les dépendances existent entre ces événements lorsque la séquence est partiellement ou totalement ordonnée. La détection d'une attaque consiste à analyser les événements générés dans l'infrastructure VoIP et à extraire ceux qui sont pertinents pour la signature. La potentialité d'une menace dépend directement du nombre d'événements de la signature qui ont déjà eu lieu dans une période de temps donnée.

Considérons le cas d'une attaque de détournement d'un enregistrement SIP où la signature est généralement décrite par une séquence d'événements ordonnés. Si l'attaquant est à l'extérieur de l'infrastructure VoIP, le premier événement consiste en l'analyse de l'infrastructure VoIP afin d'identifier les adresses SIP-URI existantes (en s'appuyant sur des messages INVITE ou OPTIONS). Ensuite, l'attaquant envoie un message REGISTER spécifique au serveur IPBX

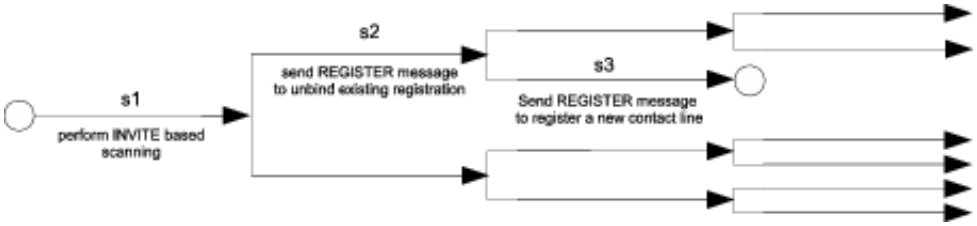


FIGURE 5.1 – Attaque observable par signatures

afin de modifier les enregistrements existants. Par exemple, cette requête malveillante précise dans l'entête de requête d'un paramètre générique pour le contact et une valeur nulle pour l'instant d'expiration, le dernier événement lié à cette attaque consiste à envoyer un deuxième message REGISTER pour enregistrer une nouvelle ligne d'enregistrement avec l'entête contenant l'adresse de l'attaquant.

Potentialité des attaques observables par détection de signatures

Quantifier la potentialité d'une attaque VoIP dépend directement de son observabilité. Concernant les attaques de sécurité appartenant à la première classe, la potentialité peut être calculée sur la base du nombre d'événements de la signature d'attaque qui sont survenus dans le réseau VoIP. Soit $\mathcal{P}_1(a)$ la potentialité des attaques observables par détection de signature, elle est alors présentée par l'équation 5.1, où E représente l'ensemble des événements survenus dans le réseau et $signature(a)$ définit la signature de l'attaque considérée.

$$\mathcal{P}_1(a) = \frac{|E \tilde{\cap} Signature(a)|}{|Signature(a)|} \tag{5.1}$$

La potentialité est alors définie par le pourcentage d'événements liés à l'attaque qui ont déjà eu lieu au cours d'une période de temps donnée (généralement spécifiée par une fenêtre glissante). L'intersection $\tilde{\cap}$ est considérée comme l'intersection entre deux ensembles dans le cas de signature non ordonnée, tandis que cette intersection est considérée comme une intersection ordonnée dans le cas de signature ordonnée. L'importance relative des événements peut également être différenciée en introduisant des facteurs de pondération.

5.2.2 Attaques observables par détection d'anomalies

La seconde classe d'observabilité inclut les attaques de sécurité qui sont détectables à l'aide des techniques de détection basées sur les anomalies. Dans ce cas, l'attaque de sécurité est identifiée par la variation de paramètres réguliers du réseau. Un modèle référentiel des performances caractérisant un comportement régulier doit être d'abord établi en analysant le trafic du réseau. Ensuite, le trafic du réseau qui s'écarte de cette ligne de base est identifié afin de détecter une attaque. Une menace est donc déterminée sur la base d'un ensemble de métriques du réseau noté $deviation(a)$:

$$deviation(a) = c_1, c_2, \ldots, c_n \tag{5.2}$$

qui caractérise l'écart par rapport à un trafic normal. Une attaque détectable par anomalies peut par exemple être caractérisée par un événement généré d'une manière répétitive et à peu près

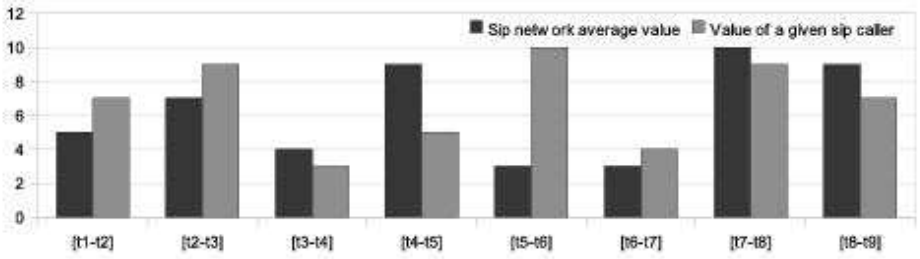


FIGURE 5.2 – Attaque observable par anomalies

constante dans le temps, par exemple l'envoi de messages INVITE pour établir plusieurs sessions à la fois et consommer les ressources d'un serveur proxy SIP. Soit a une attaque détectable par anomalies, Adr_A l'ensemble des adresses utilisées pour l'attaque et Adr_T l'ensemble des adresses des machines attaquées. Soit $e(x_j \rightarrow y_k)$, l'événement produit par l'attaquant dont l'adresse est x_j ($\in Adr_A$) qui cible l'équipement VoIP dont l'adresse est y_k ($\in Adr_T$). L'attaque a peut être définie par l'équation 5.3

$$a = \{e(x_j \rightarrow y_k)\}_{1 \leq i \leq n} \tag{5.3}$$

avec n le nombre de répétitions. Nous trouvons dans cette classe d'observabilité des attaques telles les attaques DoS/DDoS, attaque d'inondation par message, SPIT, attaque d'authentification SIP par dictionnaire.

Considérons le cas des attaques SPIT, une telle attaque peut être identifiée en fonction des paramètres réseau, y compris le taux de rejet d'appel qui est généralement plus élevé que les appels réguliers, la durée de l'appel qui est généralement de courte durée par rapport aux appels réguliers et l'intervalle de temps qui sépare deux appels consécutifs et qui est plus régulier dans le cas SPIT [84].

Soit x_0 l'adresse SIP-URI de l'attaquant et $\{y_1, y_2, \ldots, y_n\}$ l'ensemble des adresses ciblées. La modélisation de l'attaque SPIT est donnée par l'équation 5.4 avec $\{e(x_0 \rightarrow y_k)\}$ correspondant à un appel VoIP.

$$a_{SPIT} = \{e(x_0 \rightarrow y_k)\}_{1 \leq k \leq n} \tag{5.4}$$

Potentialité des attaques observables par détection d'anomalies

Dans le cas d'une attaque observable par les méthodes de détection d'anomalies, la potentialité d'une attaque VoIP est spécifiée sur la base de l'ensemble de paramètres normalisés noté $deviation(a)$ qui permet de caractériser le trafic ou le comportement normal ou anormal. La potentialité peut être alors définie par l'équation 5.5 où λ_i représente la valeur du poids associé au ième paramètre de déviation c_i. Chaque critère peut être considéré comme un symptôme, il quantifie l'anomalie liée à une attaque donnée. Soit $\mathcal{P}_2(a)$ la potentialité d'une attaque observable par détection d'anomalies. Alors, $\mathcal{P}_2(a)$ est décrite par l'équation 5.5.

$$\mathcal{P}_2(a) = \sum_{i \in deviation(a)} \lambda_i c_i \tag{5.5}$$

5.2.3 Attaques non observables

Dans le cas d'attaques non observables ou d'attaques difficilement détectables (par exemple les attaques instantanées), il est difficile ou impossible d'estimer la potentialité $\mathcal{P}(a)$ au cours du temps d'une manière appropriée. Nous proposons de considérer que $\mathcal{P}(a)$ est fixée à une valeur maximale ($\mathcal{P}_3(a) = 1$). Par conséquent, le système agit d'une manière préventive à l'égard de ces attaques, il active les mécanismes de sécurité associées d'une manière systématique, ce qui correspond à une stratégie d'évitement des risques. Ce type d'attaque peut typiquement être basé sur la survenance d'un événement unique (par exemple l'envoi d'un message incorrect ou malformé).

Soit a une attaque instantanée, elle est définie mathématiquement par un unique événement e selon l'équation 5.6.

$$a = \{e\} \tag{5.6}$$

Par exemple, une attaque par message malformé peut être simplement représentée par :

$a_malform = \{envoidumessagemalform\}$ Nous trouvons aussi d'autres attaques difficilement observables qui ne se caractérisent pas par une progression d'actions au cours du temps, comme l'analyse et l'écoute du trafic. Elles peuvent cependant être définies par des prémisses qui sont les pré conditions de l'attaque avec

$$Prem(a) = \{b_1, b_2, \ldots, b_n\} \tag{5.7}$$

Nous pouvons par exemple trouver parmi les prémisses le scan de ports de communications.

Dans cette section, nous avons classifié les attaques VoIP en trois classes suivant leurs propriétés d'observabilité. Pour chaque classe d'attaques, nous avons développé une formule spécifique de potentialité qui est adaptée à leur but et leur nature. Dans la section suivante, nous nous intéressons à une évaluation multi-critères de notre modèle étendu pour la gestion de risques afin de montrer les avantages et identifier les limites en termes de risque, de coût induit et de disponibilité du service.

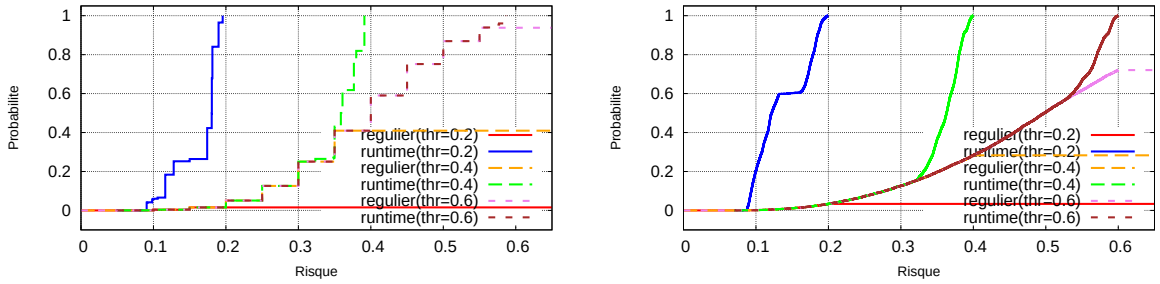
5.3 Évaluation multi-critères du modèle étendu

Nous avons évalué notre stratégie de gestion de risques à travers une série d'expériences afin d'évaluer ses performances. Ces expériences ont été menées par simulation et sont complémentaires à la mise en œuvre du prototype. Nous avons considéré différents scénarios avec un certain nombre de contremesures allant de 5 à 20, et une valeur seuil de risque variant de 0 à 1. Chaque contremesure se caractérise par un coût d'application dans l'infrastructure et un impact sur son exposition. Le coût quantifie le délai supplémentaire induit par l'application d'une contremesure et l'impact sur l'exposition quantifie la probabilité qu'une attaque donnée soit effectivement mitigée par la contremesure.

Au cours des simulations, ces deux paramètres sont définis par des distributions de probabilité uniformes autour des valeurs observées expérimentalement au cours des tests unitaires effectués avec le prototype. Nous avons considéré les attaques de sécurité appartenant à trois classes d'observabilité. Pour les attaques observables par détection de signature, nous avons considéré une signature composée d'un maximum de 20 événements, la survenance d'un événement fait suite à une probabilité de Bernoulli, mais les expériences peuvent facilement être étendues à d'autres modèles d'attaque élaborés. Pour les attaques observables par détection d'anomalies, les critères de déviation sont caractérisés par des distributions uniformes paramétrées par l'étude statistique mentionnée précédemment. Pour la troisième classe correspondant à des attaques difficilement observables, nous avons considéré une attaque de sécurité avec une taille de signature mise à 1. Nous utilisons la même variable (*seed*) pour la génération de nombres pseudo-aléatoires pour tous les scénarios.

5.3.1 Performance en termes de risque, de disponibilité et de coût

Dans une première série d'expériences, nous nous sommes intéressés à évaluer la performance globale de notre système de gestion de risques avec des classes d'attaque différentes d'observabilité. Nous avons comparé notre solution à une stratégie traditionnelle consistant à appliquer une seule contremesure de forte portée. Les figures 5.3(a) et 5.3(b) représentent la distribution de probabilité d'amplitude de risque que l'on obtient respectivement avec la première et la deuxième classe d'observabilité. La troisième classe d'attaque sera traitée dans la sous-section sur l'impact de la taille des signatures. Chaque figure est composée de 6 courbes correspondant à



(a) Traitement du risque pour la 1ère classe

(b) Traitement du risque pour la 2ème classe

FIGURE 5.3 – Impact du seuil de risque

notre stratégie (noté *runtime*) par rapport à la stratégie traditionnelle (noté régulier) pour trois valeurs différentes de seuil de risque.

Lorsque l'on compare ces deux figures, on peut d'abord observer que les allures de courbes sont différentes pour les deux classes d'observabilité. Dans la première figure, le niveau de risque génère une courbe en escalier, qui est directement corrélée à la potentialité d'attaque. Comme l'attaque de sécurité est détectée sur la base d'une signature connue, chaque étape dans la courbe correspond à la détection d'un événement donné mais l'impact de chacun n'est pas nécessairement le même sur la potentialité d'attaque. Dans la seconde figure, les attaques sont observées sur la base de la détection d'anomalies, et les valeurs de détection sont choisies par une distribution de probabilité uniforme. Par conséquent, le niveau de risque est réparti sur un grand nombre de valeurs et génère une courbe lisse. Nous pouvons remarquer que pour la première classe d'observabilité, un nombre plus élevé d'événements permet en général de lisser la courbe correspondante.

Les distributions de probabilité montrent sur ces deux figures que le niveau de risque est plus faible avec notre stratégie qu'avec la stratégie traditionnelle : cette observation peut être faite pour les deux classes d'observabilité quelle que soit la valeur de seuil examinés au cours de ces expériences. Ceci plaide en faveur de notre approche dynamique qui est capable d'adapter l'exposition de l'architecture VoIP par rapport à la potentialité de menace. L'application d'une stratégie de sécurité progressive permet de maintenir un niveau de risque à des valeurs faibles,

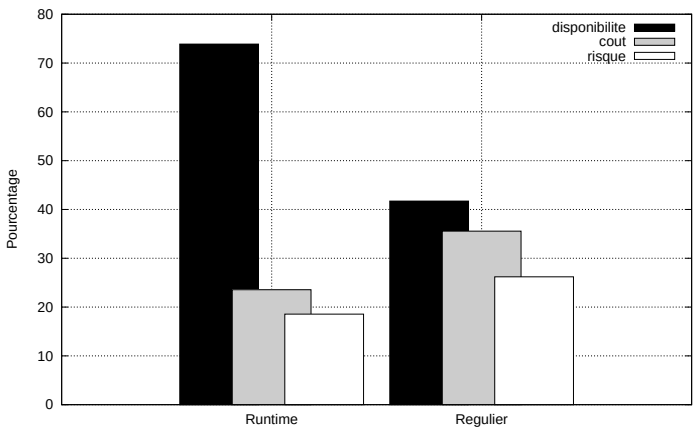


FIGURE 5.4 – Comparaison de performances en termes de risque, disponibilité et coût

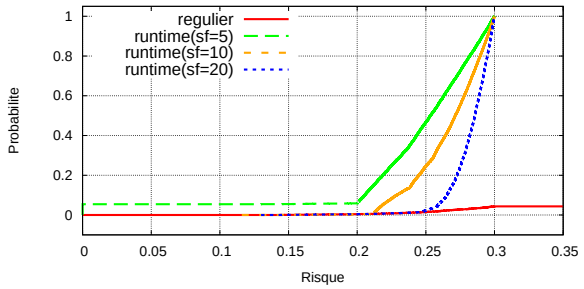
tout en optimisant les performances du service, ceci est confirmé par la figure 5.4 qui compare les performances moyennes des deux approches de sécurité en termes de risque, disponibilité et coût. Comme le montrent les figures précédentes, le niveau de risque est moins important dans notre stratégie en raison des mécanismes de mitigation soutenus par un ensemble des contremesures qui sont progressives en termes de puissance contre l'attaquant. Au cours de ces expériences, notre approche de gestion de risques réussit à maintenir la disponibilité du service à une valeur moyenne de 73,88% alors que cette valeur est réduite à 41,72% pour la stratégie classique. Ceci s'explique par l'absence des contremesures progressives dans la stratégie traditionnelle qui a un impact direct sur la disponibilité du service. De plus, le coût moyen des contremesures utilisées montre la même tendance : l'approche traditionnelle génère un coût d'application de contremesure accumulé de 35,55%, tandis que celui-ci est réduit au maximum à 23,55% avec notre approche. Ainsi, le schéma de risque progressif optimise ce coût en sélectionnant une réponse adéquate pour contrer le risque. La tendance confirme les attentes car l'impact et le coût des contremesures sont souvent corrélés.

5.3.2 Impact du nombre de contremesures

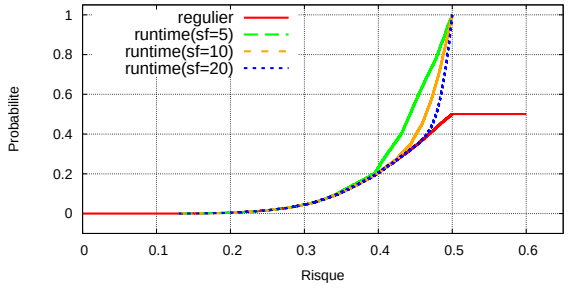
Dans une deuxième série d'expériences, nous nous sommes intéressés à évaluer dans quelle mesure le nombre de contremesures peut avoir une incidence sur notre solution de gestion des risques. Dans ces simulations, nous avons évalué la distribution de probabilité de risques en faisant varier le nombre de contremesures de 5 à 20. Nous considérons ici le nombre de contremesures de sécurité qui sont appliquées dans l'architecture VoIP, ce qui représente le nombre maximal de contremesures qui peuvent être activées simultanément pour traiter le risque. Les résultats de simulation sont représentés sur la figure 5.5 avec différentes valeurs de seuil. Celles-ci sont respectivement fixées à 0,3, 0,5 et 0,7. Sur chaque sous-figure, nous avons tracé une courbe correspondant à la stratégie traditionnelle et trois courbes correspondant à notre stratégie de gestion des risques avec respectivement 5, 10 et 20 contremesures.

Intuitivement, nous nous attendions à ce que l'augmentation du nombre de contremesures améliore les performances de notre solution ; ce qui était le cas pour les deux premières sous-figures. L'utilisation de contremesures supplémentaires et une stratégie d'application cumulative permettent de fournir une réponse plus souple et mieux adaptée au risque. Bien sûr, cela suppose que l'impact et le coût des contremesures soient suffisamment distribués et différents. Dans le cas inverse, en considérant un nombre élevé de contremesures avec des propriétés similaires, cette stratégie ne peut pas améliorer la performance de la solution. Sur les deux premières sous-figures, plus le nombre de contremesures est élevé, moins l'amplitude du risque est grande. La solution de gestion de risques avec 20 contremesures a donc généré des meilleurs résultats avec les valeurs de risque les plus basses.

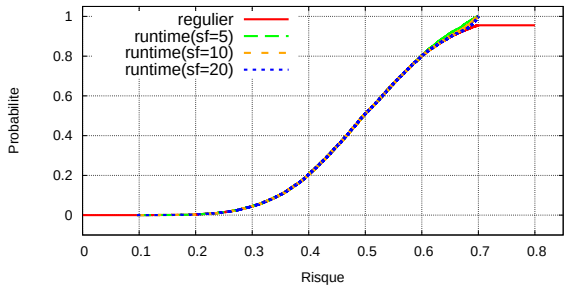
Il est également important d'analyser la dernière sous-figure qui donne des résultats intéressants : les quatre courbes convergent vers la même distribution du risque. Lorsque l'on compare les trois sous-figures avec des valeurs de seuil différentes, nous pouvons clairement observer que, lorsque nous augmentons la valeur de seuil, les avantages prévus par un nombre élevé de contremesures et leurs impacts diminuent : ce phénomène montre une corrélation entre le nombre de contremesures appliquées et la valeur seuil fixée par l'administrateur : avoir une grande variété de contremesures n'est bénéfique que si nous voulons gérer le niveau de risque d'une manière fine (suivi de risque à grains fins). C'est le cas sur la sous-figure 5.5(a) lorsque l'on considère une valeur seuil de 0,3. Cette faible valeur oblige la stratégie de risques à exécuter progressivement les différentes contremesures pour maintenir une amplitude du risque faible au fil du temps. Au même temps, un nombre restreint de contremesures peut être suffisant lorsque l'administrateur



(a) seuil = 0.2



(b) seuil = 0.4



(c) seuil = 0.6

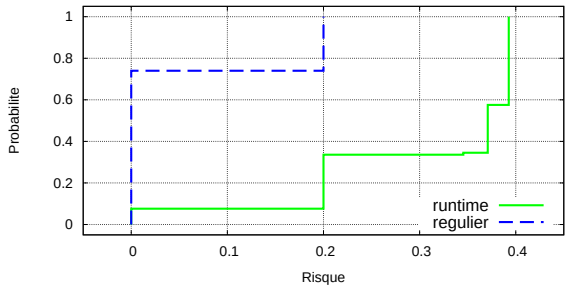
FIGURE 5.5 – Impact du nombre de contremesures sur la gestion de risques

autorise un niveau de risque élevé. C'est typiquement le cas dans la sous-figure 5.5(c) lorsque le seuil de risque est défini à 0,7.

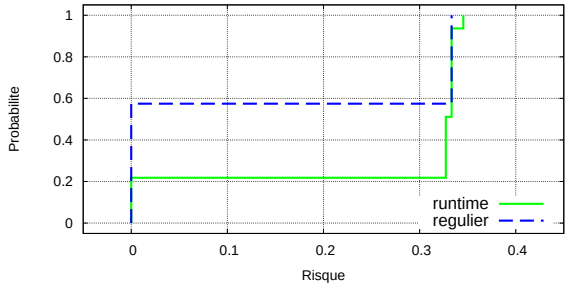
La diversité des contremesures n'a cependant pas d'impact dans ce cas sur les performances de la gestion des risques, cela signifie qu'un seuil élevé n'est pas favorable à l'utilisation d'un nombre élevé de contremesures différentes, mais limite plutôt le nombre de contremesures qui sont effectivement activées dans l'infrastructure VoIP.

5.3.3 Impact de la taille des signatures d'attaques

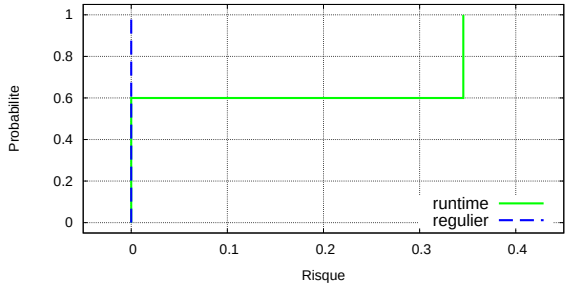
Dans une dernière série d'expériences (voir figure 5.6), nous avons évalué l'impact de la taille de la signature sur notre stratégie de gestion des risques. En particulier, nous avons évalué les performances et l'impact de notre stratégie de traitement de risques sur la dernière classe d'observabilité correspondant aux attaques VoIP non observables ou difficilement observables, en considérant une taille réduite de signature. Nous avons comparé les performances de notre stratégie de gestion des risques avec la stratégie traditionnelle tout en faisant varier la taille



(a) taille de signature = 5



(b) taille de signature = 3



(c) taille de signature = 1

FIGURE 5.6 – Impact de la taille de signature sur la gestion de risques

de la signature d'attaque. Chaque sous-figure correspond à deux courbes correspondant à notre approche et à la stratégie classique, respectivement, avec une taille de signature fixé à 1, 3 et 5 événements. Les résultats expérimentaux ont montré clairement que les avantages des approches de gestion de risques sont limités lorsque la taille de la signature est faible.

L'application d'un traitement progressif est intéressante uniquement si la potentialité d'attaque est étalée sur un ensemble de valeurs, et ne se limite pas à deux valeurs extrêmes (soit 0 et 1). Dans le cas d'une attaque non observable ou d'une attaque difficilement observable, nous plaidons en faveur d'une potentialité de menace fixée à 1 volontairement. Dans ce cas, la stratégie de gestion des risques converge vers une stratégie d'évitement du risque pour ces attaques de sécurité. Cette stratégie d'évitement consiste à éliminer les risques avant qu'ils ne surviennent. Ainsi, la solution active des contremesures d'une manière préventive, afin de contrer les attaques appartenant à cette dernière classe d'observabilité.

L'évaluation de performances en termes de risque a montré qu'un traitement adaptatif et progressif des risques comme notre stratégie le fait est indispensable pour un contrôle à grains fins des menaces. Cependant, l'évaluation de performances en termes de taille de signature montre

que notre stratégie de risques a des limites face aux attaques dites instantanées qui, vu la nature de ces attaques, exigent une approche proactive.

5.4 Modèle d'auto-configuration

La gestion des risques offre de nouvelles possibilités pour aborder le compromis entre la sécurité et la performance des services VoIP. Cette approche vise à quantifier la potentialité des menaces et à sélectionner des contremesures appropriées afin de minimiser l'impact de l'infrastructure VoIP. Cependant, notre stratégie de gestion de risques s'appuie naturellement sur un modèle de risques. La configuration de ses différents paramètres peut s'avérer difficile dans certains contextes.

Le paramétrage des modèles de risques est un défi majeur. Pour cela, nous proposons une méthode d'auto-configuration pour soutenir notre stratégie de gestion des risques [29]. Cette approche permet de simplifier la configuration de certains paramètres. Nous définissons ce mécanisme d'une façon théorique et nous décrivons ensuite son intégration au sein du modèle de risques. Nous considérons, comme cas d'étude, le paramètre du modèle qui caractérise le coût d'application d'une contremesure. Ce paramètre est crucial pour le fonctionnement de la stratégie de risques parce qu'elle vise à minimiser ce coût, tout en maintenant un niveau de risque faible. Notre approche est cependant générique et peut être appliquée d'autres paramètres du modèle de risque.

5.4.1 Complexité de la paramétrisation

La configuration des paramètres d'un modèle de risque est une activité importante et difficile, car le nombre de paramètres peut être grand (paramètres $\mathcal{P}(a)$, $\mathcal{E}(a)$ et $\mathcal{C}(a)$ sont eux-mêmes dépendants d'autres paramètres) et aussi parce que les paramètres peuvent varier en fonction du contexte. L'automatisation est nécessaire pour soutenir cette paramétrisation. Les paramètres qui sont particulièrement difficiles à configurer dans notre système de gestion des risques sont notamment les suivants :

- **Impact de la contremesure sur l'attaque** : Il quantifie la capacité d'une mesure de sécurité à protéger l'infrastructure VoIP contre une menace de sécurité. Cette quantification définit dans quelle mesure la contremesure considérée peut éviter qu'une attaque VoIP réussisse à réaliser une dégradation du service. Bien que cette quantification soit parfois évidente (dans le cas de l'application d'un patch spécifique par exemple). Dans la plupart des cas, ce paramètre est difficile à quantifier, en particulier en cas de communications non désirées (comme les attaques SPIT). Nous nous concentrons ici sur l'impact sur la source de l'attaque, et non l'impact sur la menace elle-même. Il est également intéressant de quantifier ce deuxième paramètre, même si cet impact est souvent faible en cas d'attaques générées par des robots.
- **Coût d'application d'une contremesure** : ce paramètre spécifie à quel niveau la contremesure impacte les performances du service de téléphonie. L'objectif est de déterminer si la contremesure introduit un surcoût important sur le fonctionnement normal de l'infrastructure (par exemple en termes de délai, d'indisponibilité). Il est souvent défini en termes de disponibilité du service ou de facilité d'utilisation. Il est également possible de le quantifier sur la base du nombre de clients légitimes qui ne peuvent pas surpasser l'ensemble des contremesures. Dans le cas extrême, le coût peut être considéré comme infini quand la contremesure provoque l'arrêt d'une communication VoIP. Elle ne doit pas être exécutée lorsque aucune alternative n'a été trouvée pour le traitement du risque considéré.

– **Conséquence d'une attaque réussie** : ce paramètre est généralement quantifié en termes de confidentialité, intégrité et de disponibilité. L'objectif est de déterminer si une attaque va générer des dégâts importants ou non sur l'infrastructure VoIP. Bien que la disponibilité peut être calculée d'une manière dynamique dans des scénarios spécifiques, il reste la confidentialité et l'intégrité qui sont plus difficiles et qui nécessitent souvent d'être estimées par des experts.

Les paramètres d'un modèle de risques sont nombreux dans la plupart des cas et varient par rapport à leur environnement d'application. La gestion des risques exige fortement des mécanismes d'auto-configuration pour faire face à la complexité de paramétrisation des modèles de risques.

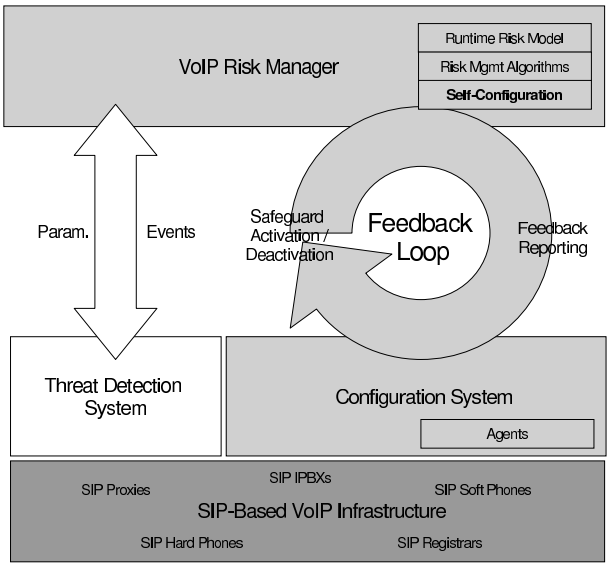


FIGURE 5.7 – Intégration de méthode d'auto-configuration dans notre approche

Nous proposons dans ce chapitre de définir une méthode d'auto-configuration pour l'amélioration de la gestion des risques dans les infrastructures VoIP (voir Figure 5.7). Cette automatisation est une condition essentielle afin de simplifier cette tâche, et afin d'adapter et d'affiner les paramètres du modèle de risques par rapport à leur contexte. Nous considérons un mécanisme de retour d'expérience pour assurer cette auto-configuration. L'objectif est de prendre en compte l'expérience du passé afin d'adapter le paramétrage et de raffiner le modèle.

5.4.2 Mécanisme de retour d'expérience

Comme le montre la figure 5.7, notre architecture VoIP contient un système de configuration qui exécute les contremesures dans l'infrastructure VoIP. La stratégie d'auto-configuration permet d'établir une boucle de retour d'expérience fondée sur les rapports effectués par les agents de configuration déployés sur les équipements VoIP. Grâce à ces rapports intégrés dans le modèle de risque, chaque application de contremesures permet d'effectuer des observations supplémentaires et de tirer partie de notre stratégie de gestion de risques. Nous avons centré notre étude sur le coût d'application de contremesures, en particulier sur les contremesures en se basant sur des tests captcha audio [108]. Il a été montré que plusieurs centaines de millions de captchas sont réalisés chaque jour, et que ces captchas pourraient représenter un coût d'un milliard de dollars en termes de perte de productivité [7, 8]. Même si cette valeur est probablement surestimée, elle illustre le besoin de bien calibrer un modèle de risque. Après l'application d'une contremesure,

l'agent estime le coût d'application et il rend sa valeur au serveur de configuration. Le serveur collecte et agrège ces statistiques qui sont après transmises au gestionnaire des risques. Le gestionnaire de risque exploite ensuite ces données afin d'affiner le coût de la contremesure considérée. Considérons le cas d'un agent UA_1 établissant une communication VoIP avec un autre agent

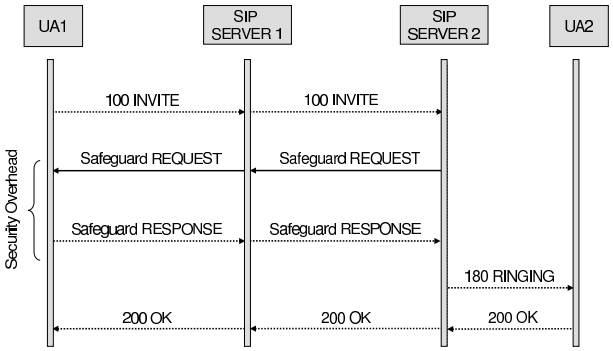


FIGURE 5.8 – Intégration de contremesures à l'initiation d'une session SIP

UA_2, comme décrit à la figure 5.8. L'agent UA_1 envoie d'abord un message INVITE pour initier une session avec le deuxième agent client. Ce message est acheminé par un proxy SIP à travers les deux domaines. Si la potentialité d'une attaque est élevée, le proxy SIP du deuxième domaine peut exiger l'application d'une mesure de sécurité afin de protéger les équipements VoIP contre les attaques qui pourraient être générées par le premier agent (cette décision a été prise par le gestionnaire de risque du domaine considéré). L'agent UA_1 est ensuite invité à interagir à cette contremesure. Par exemple, dans notre cas, le proxy SIP peut demander un test audio captcha en demandant la saisie d'un code spécifique (SAFEGUARD REQUEST). Si le deuxième agent UA_2 fournit une réponse correcte (SAFEGUARD RESPONSE), l'initiation de session peut continuer normalement avec un message SIP RINGING et un message SIP OK qui conclut l'établissement de la session. L'application de la contremesure a introduit une surcharge supplémentaire lors de l'initiation de session. Dans notre scénario, nous quantifions le coût d'application de contremesure *captcha* en terme de temps soit le délai nécessaire entre la réception de la demande d'application d'une contremesure et de fournir une réponse correcte au défi. Notre stratégie d'auto-configuration s'appuie sur un mécanisme du retour d'expérience. L'objectif est d'exploiter les résultats antérieurs d'applications d'une contremesure afin d'affiner le modèle de gestion de risques et de prédire la prochaine valeur de coût d'applications d'une manière plus efficace. Une grande variété des méthodes et techniques sont disponibles pour effectuer une telle prévision avec des performances différentes, en particulier dans le domaine de l'économétrie [90], nous avons considéré la technique ARMA couramment utilisée pour l'analyse et la prédiction de séries réelles [48]. Bien qu'elle présente certaines limitations, cette technique est entièrement en adéquation avec nos contraintes d'exécution, et nos observations peuvent facilement être mises en correspondance avec des séries chronologiques [50].

Modélisation

Un modèle ARMA est généralement défini comme la combinaison de deux modèles : le premier est un modèle auto-régressif d'ordre p noté AR(p) et le second est un modèle de moyenne mobile d'ordre q noté MA(q) [60]. Il peut donc être défini mathématiquement par l'équation 5.8.

$$y_t = \sum_{i=0}^{p} \phi_i y_{t-i} - \sum_{j=0}^{q} \theta_j \epsilon_{t-j} + \epsilon_t \tag{5.8}$$

Dans cette équation, la variable y_t correspond à la valeur prédite, alors que les variables $\{y_{t-i}\}$ représentent les valeurs précédemment prédites. La variable ϵ_t fournit l'erreur de la méthode de prédiction suivant une loi de probabilité appelée $BB(0, \sigma_t)$. Les variables $\{\Phi_i\}$ et $\theta_i\}$ sont les coefficients (positifs ou négatifs) à déterminer dans le modèle. Ces coefficients peuvent être estimés avec la méthode de maximum de vraisemblance. Nous appliquons la technique d'analyse ARMA afin d'affiner le coût d'application des contremesures. Nous notons $fcCost$ le coût prédit et $efCost$ le coût effectif de la mesure de sécurité. Dans ce cas, le coût prédit $fcCost_t$, à l'instant t, est donné par l'équation 5.8 avec ϵ_{t-j} représentant la différence entre la coût réel $fcCost_{t-j}$ et le coût prédit $efCost_{t-j}$ à l'instant t.

$$\begin{aligned} fcCost_t &= \sum_{i=0}^{p} \phi_i fcCost_{t-i} \\ &- \sum_{j=0}^{q} \theta_j \epsilon_{t-j} + \epsilon_t \end{aligned} \tag{5.9}$$

$$\epsilon_{t-j} = fcCost_{t-j} - efCost_{t-j} \tag{5.10}$$

Par conséquent, les algorithmes de mitigation (algorithme de restriction et algorithme de relaxation des risques) tentent de réduire au minimum les valeurs raffinées correspondantes aux coûts des contremesures, tout en maintenant le niveau de risque à une valeur acceptable, tel que décrit par l'équation 5.11.

$$minimize(\sum_i fcCost(sf_i)_t) \; and \; R < R_{threshold} \tag{5.11}$$

Analyse et validation

La technique d'analyse ARMA correspond à cinq étapes et inclut un test de validation comme décrit dans [49]. Nous rappelons brièvement ci-dessous ces différentes phases et leur application dans notre scénario de gestion de risques.

La première étape de cette analyse consiste à identifier et filtrer la périodicité ; cette tâche peut être effectuée par l'analyse des corrélogrammes simples ou partielles et en appliquant un test dédié tel que le test de Dickey-Fuller ou le test Philips-Perron. La deuxième étape permet de déterminer les ordres p et q du modèle ARMA ; cette tâche est généralement effectuée en se basant sur l'analyse des corrélogrammes simples et partielles. La fonction d'autocorrélation mesure la corrélation entre $efCost_t$ et $efCost_{t-k}$, et l'influence des autres variables $(efCost_{t-1}, efCost_{t-2}, \ldots, efCost_{t-k+1})$ a été retirée de l'analyse. Le coefficient d'autocorrélation d'ordre k est donné par l'équation 5.12.

$$\rho_k = \frac{cov(efCost_t, efCost_{t-k})}{\sigma_{efCost_t} \sigma_{efCost_{t-k}}} \tag{5.12}$$

La troisième étape estime les coefficients $\{\Phi_i\}$ et $\{\theta_i\}$ (positif ou négatif) du modèle ARMA : Les coefficients pondèrent respectivement les variables $fcCost_{t-i}$ et ϵ_{t-j} (voir l'équation 5.10). Leur estimation est obtenue en exploitant la méthode du maximum de vraisemblance. La quatrième étape représente la validation du procédé du modèle ARMA. Premièrement, il consiste à analyser les coefficients et les résidus, puis d'appliquer le test d'auto-corrélation de Box et Pierce en utilisant une quantité statique Q qui est donnée par l'équation 5.13.

$$Q = n \sum_{k=0}^{K} \rho_k^2 \tag{5.13}$$

Dans cette équation, n représente le nombre d'observations et ρ_k représente le coefficient d'auto-corrélation des résidus estimés d'ordre k. Cette validation permet de déterminer le terme d'erreur en fonction de la taille de l'échantillon. La dernière étape consiste à quantifier le coût de la contremesure prévue fondée sur la modélisation établie, en utilisant l'équation 5.13.

Cette analyse permet de raffiner les paramètres du modèle de risque d'une manière dynamique. Nous allons estimer l'impact de l'erreur d'estimation sur notre solution dans la sous-section correspondant aux résultats expérimentaux.

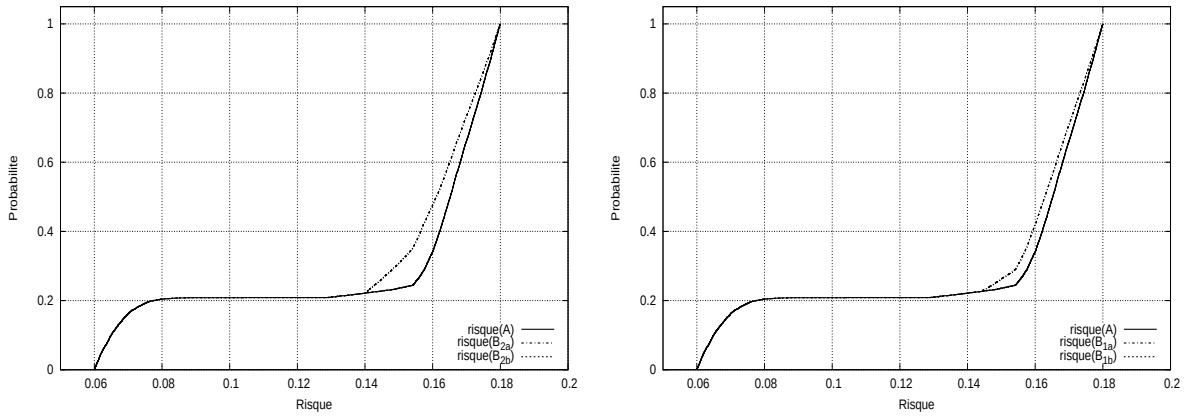
5.4.3 Évaluation du retour d'expérience

Nous avons évalué la performance de notre méthode d'auto-configuration à travers un ensemble des résultats de simulations [9]. Nous avons considéré le scénario d'attaques SPIT, comme elle est une menace très commune dans les infrastructures VoIP. Cependant, notre but n'est pas de quantifier les performances de détection (sensibilité et spécificité), mais d'évaluer l'impact de notre mécanisme du retour d'expérience sur le schéma de traitement de risques. L'arrivée d'appels suit une loi de Poisson avec une moyenne de 100 appels par unité de temps et la durée de communication est représentée par une loi exponentielle avec une moyenne de 10 secondes.

Les attaques sont représentées par 4 types différents avec une intensité croissante d'appels SPIT (de 10 à 1000 appels SPIT par unité du temps). Nous définissons 5 à 20 contremesures différentes où chaque composant de cet ensemble est caractérisé par trois variables : le coût (ce qui représente le délai supplémentaire introduit par la contremesure, avec un terme d'erreur entre 1% et 10%), la probabilité qu'un appel malveillant contourne la contremesure (suit une distribution uniforme dans l'intervalle [0,8 ;1]), et la probabilité qu'un appel légal contourne la contremesure (suit une distribution uniforme dans l'intervalle [0,8 ;1] (meilleur des cas) et [0 ;0,2] (pire cas)). Nous exposons un sous-ensemble de nos résultats expérimentaux : nous nous intéressons en particulier à évaluer les avantages et les limites du mécanisme de retour d'expérience et son impact sur notre stratégie de gestion de risques.

Impact de retour d'expérience sur l'amplitude du risque

Dans une première série d'expériences, nous avons étudié l'impact du mécanisme de retour d'expérience sur l'amplitude du risque. La figure 5.11 représente la distribution du risque pour trois cas différents : le coût avec un terme d'erreur de 0% (scénario A), un coût avec un terme d'erreur inférieure ou égale à 5% (scénario B_{2a}), et un coût avec un terme d'erreur entre 5% et 10% (scénario B_{2b}). On peut observer clairement sur la sous-figure 5.9(a), correspondant à un terme d'erreur positif, que l'amplitude de risque dans le scénario A est supérieure à celles des scénarios correspondant à B_{2a} et de B_{2b}. La distinction entre le scénario A et les autres scénarios commence avec une amplitude de risque supérieur à 0,14, les deux courbes correspondantes aux scénarios B_{2a} et B_{2b} convergent vers la même distribution.



(a) Impact de l'erreur avec un taux positif

(b) Impact de l'erreur avec un taux négatif

FIGURE 5.9 – Impact de l'auto-configuration sur le traitement des risques

Nous observons le même phénomène avec les simulations de la seconde sous-figure 5.9(b) : nous avons tracé les trois mêmes scénarios A, B_{1a} et B_{1b}, mais dans ce cas avec un sens négatif. L'amplitude de risque est une fois plus élevée avec le scénario A que avec les deux autres scénarios B_{2a} et B_{2b}, et la distribution de ces deux derniers scénarios sont également convergents. Cependant, la différence entre le scénario A et les scénarios B_{2a} et B_{2b} est moins importante en terme de risque que dans la sous-figure 5.9(a).

Ces résultats sont cohérents avec notre stratégie de gestion des risques : l'objectif des algorithmes de gestion est de minimiser le coût induit par les contremesures activées, tout en maintenant l'amplitude de risque au dessous d'une valeur seuil, ce qui permet d'expliquer les résultats expérimentaux observés dans les deux sous-figures 5.9(a) et 5.9(b). Dans la sous-figure 5.9(a), la différence entre les scénarios A et B (B_{2a} et B_{2b}) est due à l'activation d'une contremesure avec un impact plus élevé que nécessaire. Le taux d'erreur avec un sens positif contribue à la sélection d'une telle contremesure à un stade précoce, elle est choisie parce que son coût semble moins élevé que le coût effectif. Dans ce cas, le taux d'erreur conduit le modèle de risque à générer une restriction sur l'exposition de l'infrastructure plus importante que nécessaire. Cela minimise l'amplitude de risques d'une manière plus importante, mais le coût d'application de contremesures n'est pas optimisé. Ce qui signifie que le gestionnaire de risques va activer des contremesures qui ne sont pas nécessairement requises pour la protection de l'infrastructure VoIP, et introduire un retard supplémentaire dans le fonctionnement du service. La différence entre les taux d'erreur de B_{2a} et B_{2b} n'est pas assez suffisante pour modifier la sélection des contremesures dans ces deux scénarios.

Nous pouvons observer un comportement similaire dans la seconde sous-figure 5.9(b), tandis que nous nous attendions au phénomène inverse : une amplitude de risque moins importante avec le scénario A qu'avec les deux scénarios B. Dans ce cas, le coût de la sauvegarde de sécurité est diminué (taux d'erreur sens négatif) jusqu'à 10%. La contremesure concernée semble moins coûteuse que son coût réel, ce qui conduit une fois de plus les algorithmes de gestion de risques à sélectionner une contremesure plus impactante qu'effectivement nécessaire en ce qui concerne la potentialité de la menace. Une amplitude de risque moins élevée avec les scénarios B_{2a} et B_{2b} ne signifie pas que les résultats de performance sont meilleurs, mais que la solution de gestion a sous-estimé le coût de la contremesure, ce qui peut générer un impact significatif sur la performance du service.

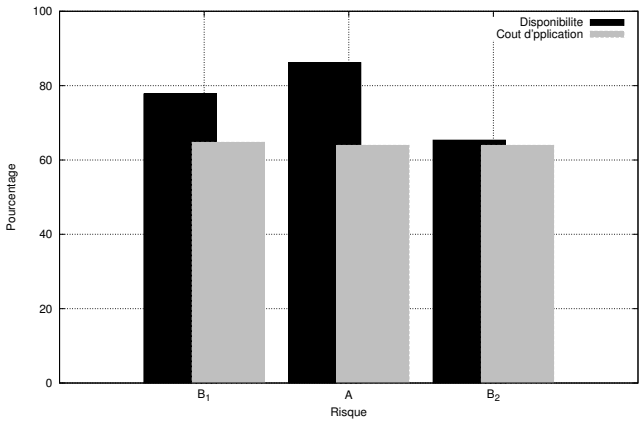


FIGURE 5.10 – Impact du retour d'expérience sur la disponibilité

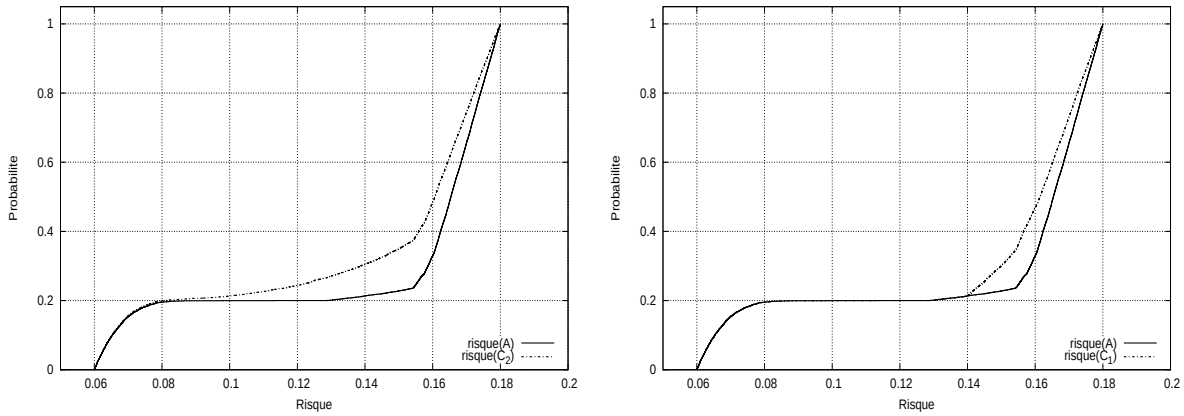
Impact du retour d'expérience sur la disponibilité du service

Une autre question intéressante est de déterminer dans quelle mesure le mécanisme du retour d'expérience impact sur la performance du service. Nous avons donc évalué dans une deuxième série d'expériences, intégrant à la fois la disponibilité du service et le coût total induit de contremesures. Nous avons évalué dans la figure 5.10 ces deux mesures d'une manière normalisée (les valeurs de disponibilité et les valeurs de coûts sont estimées entre 0% et 100%), pour les trois scénarios mentionnés précédemment A, B_{2a} et B_{2b}. Nous observons sur ce diagramme que le scénario A offre un coût effectif moins élevé par rapport aux scénarios B_{2a} et B_{2b} en raison de contremesures. Le scénario A montre également la meilleure performance de service avec une valeur pouvant aller jusqu'à 86%, tandis que les scénarios B_{2a} et B_{2b} prévoient respectivement une valeur pouvant aller jusqu'à 77% et jusqu'à 65%. En effet, l'infrastructure VoIP est surprotégée dans ces deux derniers scénarios parce que les paramètres du modèle de risques ne sont pas correctement configurés, ce qui plaide en faveur de notre mécanisme de raffinement.

Il est également important d'évaluer comment le nombre de contremesures activées peut avoir un impact sur la performance du service quand le mécanisme du retour d'expérience est activé. Nous avons quantifié la performance du service en faisant varier le nombre de contremesures de 5 à 20. Le comportement de notre approche dépend de la distribution des valeurs de coût sur l'ensemble des contremesures : plus la différence entre les coûts de deux garanties consécutives est importante, plus la solution de gestion de risques est sensible au taux d'erreur. Si l'on considère une distribution de coûts suffisamment homogènes entre les contremesures, un grand nombre de contremesures réduit la différence de coût entre les deux contremesures consécutives. Ainsi, notre modèle de risque tolère moins en moyenne le taux d'erreur. De la même manière, un nombre réduit de contremesures augmente la différence entre les coûts de deux contremesures consécutives. Ainsi, il réduit sa sensibilité à l'égard du taux d'erreur.

Analyse des cas limites

Dans une dernière série d'expériences, nous avons évalué les performances de notre stratégie d'auto-configuration dans les cas limites. On entend par cas limites, des scénarios où l'ensemble des contremesures inclut des contremesures avec un faible impact et un coût d'application élevé (le scénario C_1), ou avec un impact élevé et un coût d'application peu élevé (le scénario C_2). Les résultats expérimentaux sont donnés par la figure 5.11.



(a) Impact de l'erreur avec un taux positif (b) Impact de l'erreur avec un taux négatif

FIGURE 5.11 – Analyse des cas limites

Dans un premier scénario (voir le sous-figure 5.11(a)), nous avons considéré une contremesure dont le profil est estimé élevé (scénario C_2). Dans ce cas, la contremesure aura une priorité d'application par rapport aux autres contremesures disponibles ce qui produit un plus faible rapport coût / efficacité. Ainsi, l'erreur de configuration génère une amplitude de risque plus faible, mais le coût d'application peut détériorer sensiblement les performances du service VoIP. La divergence entre le premier scénario A et le scénario des profits élevés C_1 commence à un stade précoce à partir d'une valeur de risque égale à 0,08. La contremesure estimée à un profit élevé est activée dès que l'amplitude du risque devient élevée, et les autres contremesures ne sont pas activées parce qu'ils induisent un coût plus élevé selon le paramétrage.

Le deuxième scénario représenté sur la figure 5.11(b) correspond à une contremesure dont le coût estimé est faible (scénario noté C_1). Dans ce cas, la mesure de sécurité ne peut jamais être appliquée parce qu'elle est cachée par d'autres contremesures plus performantes, ce qui implique que les algorithmes de gestion de risques ont sélectionné des contremesures ayant un coût effectif plus élevé, en comparaison à la contremesure concerné par l'expérience.

En conclusion, le taux d'erreur impacte les deux cas limites quand le coût de la contremesure est sous-estimé ou surestimé. L'amplitude du risque est moins importante, mais cela peut considérablement influer sur le coût effectif de la contremesure. Notre mécanisme de retour d'expérience permet de réduire le terme d'erreur en affinant les paramètres de configuration d'une manière dynamique.

5.5 Synthèse

L'objectif des travaux présentés dans ce chapitre est de soutenir l'adaptation dynamique de l'exposition du réseau VoIP face à de multiples menaces de sécurité, d'étendre notre solution de gestion de risques dans ce contexte ainsi que de faciliter la configuration de certains paramètres du modèle. Cette adaptation est une exigence pour minimiser l'impact des mécanismes de protection sur un tel service critique. Par exemple, l'utilisation systématique des tests de Turing [58] introduit un retard supplémentaire et non négligeable sur la mise en place de sessions d'appels VoIP. Ces mécanismes de protection ne doivent être activés qu'en cas de besoin, d'une manière dynamique et progressive. Cette option dépend fortement des propriétés des attaques de sécurité. Dans ce contexte, nous avons classé et modélisé les attaques VoIP en considérant leurs propriétés

d'observabilité. Sur la base de cette modélisation, nous avons mathématiquement étendu notre modèle de risque afin de couvrir une plus grande variété d'attaques. Enfin, nous avons déterminé dans quelle mesure la classe d'observabilité d'attaques VoIP impacte sur les performances de notre solution en termes de risque, de disponibilité et de coût. En particulier, notre stratégie de gestion des risques est difficilement applicable avec les attaques non observables. Dans ce cas, le modèle converge vers une approche d'évitement du risque (activation systématique des contremesures). Nous avons aussi montré comment le nombre de contremesures influence sa performance et comment cette dernière est corrélée aux valeurs seuil choisies. En fait, le nombre de contremesures peut être déterminé en fonction du seuil de risque car ce dernier illustre la criticité du service.

Nous avons également amélioré la paramétrisation du modèle. Pour cela, nous avons proposé une méthode d'auto-configuration pour soutenir la gestion des risques dans les réseaux VoIP. Nous avons conclu qu'un mécanisme de retour d'expérience peut améliorer les décisions prises par le gestionnaire de risques en raffinant certains paramètres du modèle. Nos travaux se sont focalisés jusqu'à présent sur les réseaux VoIP d'entreprise s'appuyant sur un serveur VoIP IPBX central. Nous allons maintenant nous intéresser à des architectures davantage distribuées tels que les réseaux P2PSIP.

Chapitre 6

Extension aux réseaux pair-à-pair SIP

Sommaire

6.1 Introduction

De nombreux efforts sont actuellement déployés pour étendre le protocole SIP dans les réseaux distribués [100]. En particulier, le protocole P2PSIP [21] vise à fournir une solution ouverte et décentralisée où les serveurs d'enregistrement et de localisation sont remplacés par une table de hachage distribuée (DHT), qui stocke la correspondance entre l'adresse d'enregistrement SIP-URI (sip :dumont@sip.example.com) et l'adresse de contact SIP-URI (sip :dumont@1.2.3.4 :5060). L'architecture sous-jacente du réseau pair à pair est typiquement basée sur une hiérarchie à deux niveaux composée de pairs ordinaires et de super-pairs responsables du maintien de la DHT [65].

L'émergence d'un tel protocole ouvert est très prometteur pour la téléphonie IP : il ne nécessite aucun serveur centralisé, et fournit des performances intéressantes en termes de tolérance aux pannes et de passage à l'échelle. Cependant, ce protocole pose des nouveaux problèmes de sécurité : les communications VoIP sont encore plus exposés aux menaces de sécurité que dans les environnements SIP traditionnels centralisés. Un défi majeur est de fournir un service P2P VoIP ouvert qui soit à la fois (a) sécurisé, (b) performant et (c) véritablement décentralisée.

21. Concepts and Terminology for Peer to Peer SIP, IETF Internet Draft, www.p2psip.org/drafts/draft-ietf-p2psip-concepts-02.html

Afin de répondre à ce problème, nous proposons de mettre en œuvre notre stratégie de gestion des risques dans les réseaux P2PSIP afin d'adapter automatiquement les contremesures face aux attaques [28]. L'objectif est de minimiser l'exposition aux risques tout en maintenant les performances du réseau. Cette stratégie doit être adaptée à une architecture décentralisée. Elle doit contenir un modèle d'estimation distribué des risques pour que chaque pair puisse évaluer son niveau de risque d'une façon locale. Elle doit aussi permettre de déployer un ensemble de contremesures applicables en évitant l'intervention d'une tierce partie. Des mécanismes locaux de prise de décision sont nécessaires. Notre stratégie de gestion de risques vise à maintenir un compromis fort entre la continuité opérationnelle des services VoIP et leur sécurité d'une façon distribuée et locale. Nous proposons dans ce chapitre l'identification de scénarios d'attaques dans les réseaux P2PSIP, l'instanciation de notre solution de gestion de risques dans ces réseaux à partir de contremesures spécifiques et enfin un ensemble de résultats expérimentaux pour en évaluer les performances.

6.2 Identification des scénarios d'attaques

Utiliser le protocole SIP dans les réseaux P2P est intéressant pour éviter le déploiement des serveurs proxy SIP. Ce nouveau domaine de recherche a conduit à différentes extensions de ce protocole de signalisation, en particulier l'extension P2PSIP. Un réseau P2PSIP, tel que défini dans [100], est un réseau d'architecture pair à pair pour les communications SIP, qui exploite une table de hachage distribuée (une DHT Chord) pour enregistrer et localiser les utilisateurs SIP et gérer des sessions entre eux. Ce réseau est basé sur une architecture à deux niveaux hiérarchiques composée de pairs ordinaires et de super-pairs. Les pairs ordinaires, caractérisés par des ressources et / ou des capacités limitées, interagissent en tant que clients SIP, tandis que les super-pairs, caractérisés par davantage de ressources et/ou des capacités relativement importantes, interagissent en tant que serveurs SIP et ont pour rôle de maintenir la table DHT.

Dans le réseau P2PSIP, les pairs ordinaires sont associés aux super-pairs. Au cours de la phase d'enregistrement, et afin de rejoindre le réseau P2P, un nouveau pair doit d'abord repérer un ensemble de super-pairs en utilisant les techniques de multicast ou le protocole SLP [22]. Il sélectionne ensuite deux de ces super-pairs pour des raisons de redondance, et leur envoie un message SIP REGISTER. Ensuite, les super-pairs enregistrent l'identité et l'emplacement de l'utilisateur SIP dans la table de hachage distribuée, en se basant sur la valeur hachée de l'adresse d'enregistrement SIP-URI et l'algorithme de DHT pour les placer dans le pair adéquat. Cette table de hachage distribuée est maintenue par les super-pairs d'une manière dynamique.

Lors d'une initiation de session d'appel, un pair ordinaire envoie d'abord un message SIP INVITE à un de ses super-pairs. Ce dernier interroge la table de hachage distribuée qui sert de serveur d'enregistrement, afin de récupérer la localisation (l'adresse IP et le port) où il se trouve le client SIP destination. La session d'appel peut alors être établie entre les deux parties communicantes.

Ces infrastructures P2PSIP sont exposées à des attaques multiples de sécurité héritées de la VoIP et des réseaux P2P. Ces attaques contre la sécurité sont classées en trois catégories principales dans [52]. La première catégorie correspond à des attaques de sécurité visant le réseau de recouvrement P2P, telles que les attaques à base de *sybils*, les attaques de routage et de refus ou déni de service. La deuxième catégorie concerne les attaques de sécurité liées au protocole de signalisation, tels que l'attaque d'usurpation d'identité, le détournement d'appel, et les attaques SPIT [69]. La dernière catégorie comporte les attaques de sécurité ciblant les

22. SLP Service Location Protocol, IETF Internet Draft, www.ietf.org/rfc/rfc2608.txt

protocoles de transport de médias, telles que les tentatives d'écoute et l'analyse du trafic. Une grande variété de techniques a déjà été proposée pour faire face à ces attaques, comme détaillé dans la partie 6.2. Pour cela, nous sommes amené à identifier les sources d'attaque et nous nous sommes concentrés, dans ce travail, sur la première catégorie de menaces et principalement le déni de service (DoS).

6.2.1 Identification des sources d'attaques

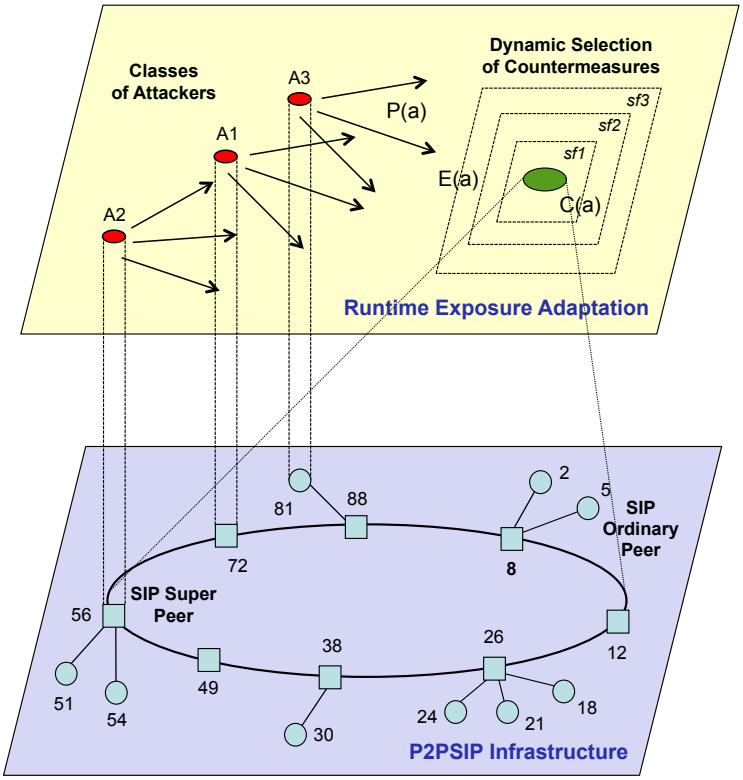


FIGURE 6.1 – Notre approche de gestion de risques appliquée aux réseaux P2PSIP

Il y a donc trois classes principales d'interactions dans une telle architecture : des interactions entre les pairs ordinaires SIP basées sur des communications en SIP, en particulier les requêtes SIP INVITE pour établir des sessions SIP, les requêtes SIP REGISTER pour s'enregistrer auprès d'un serveur d'enregistrement et les messages RTP pour le transfert média ; des interactions entre les pairs ordinaires et les super-pairs basées sur des communications en SIP qui permet aux pairs ordinaires de communiquer avec la DHT ; des interactions entre les super-pairs basées sur le protocole pair à pair implanté pour maintenir la table de DHT incluant les enregistrements SIP contenant la correspondance entre les adresses de contact et les adresses d'enregistrement.

L'attaquant peut utiliser dans le réseau P2PSIP soit le protocole SIP ou soit le protocole P2P pour réaliser une attaque P2P ou une attaque SIP. Dans nos scénarios, l'attaquant tente essentiellement de mettre en place une attaque de déni de services ou de manipulation de sessions afin de rendre un ou plusieurs nœuds injoignables en changeant la correspondance entre

l'adresse d'enregistrement et l'adresse de contact. Ces classes d'attaques sont directement liées aux capacités de l'attaquant, à savoir les piles protocolaires que celui-ci met en œuvre. Ainsi, nous considérons trois classes d'attaquants dans l'architecture P2PSIP :

- Classe d'attaquants A_1 : ces attaquants implantent la pile protocolaire P2P uniquement. Ils ne communiquent pas de messages SIP, mais peuvent modifier le contenu de la table de hachage distribuée. Par exemple, ils peuvent modifier les entrées enregistrées dans la table en envoyant des messages manipulés, ou par l'injection de pairs *sybils* à proximité du super-pair cible. Généralement, il s'agit d'un pair n'appartenant pas au réseau P2PSIP qui utilise des messages PUT pour manipuler la table DHT,
- Classe d'attaquants A_2 : cette seconde classe correspond à des attaquants qui implémentent les deux piles protocolaires P2P et SIP. Elle contient, généralement, des super-pairs. Ils peuvent générer et modifier les entrées enregistrées dans la table DHT. Ils peuvent également interagir en tant que serveurs proxy SIP afin d'effectuer des attaques *man-in-the-middle* pour manipuler les sessions SIP,
- Classe d'attaquants A_3 : cette dernière catégorie représente les attaquants qui manipulent le protocole SIP uniquement. Ces pairs SIP ordinaires envoient, généralement, des messages d'enregistrement malicieux aux super-pairs en usurpant l'identité d'un utilisateur SIP afin de polluer la table DHT de telle façon il rend la cible injoignable ou il aura la capacité d'analyser le trafic venant à elle.

6.2.2 Scénarios d'attaques

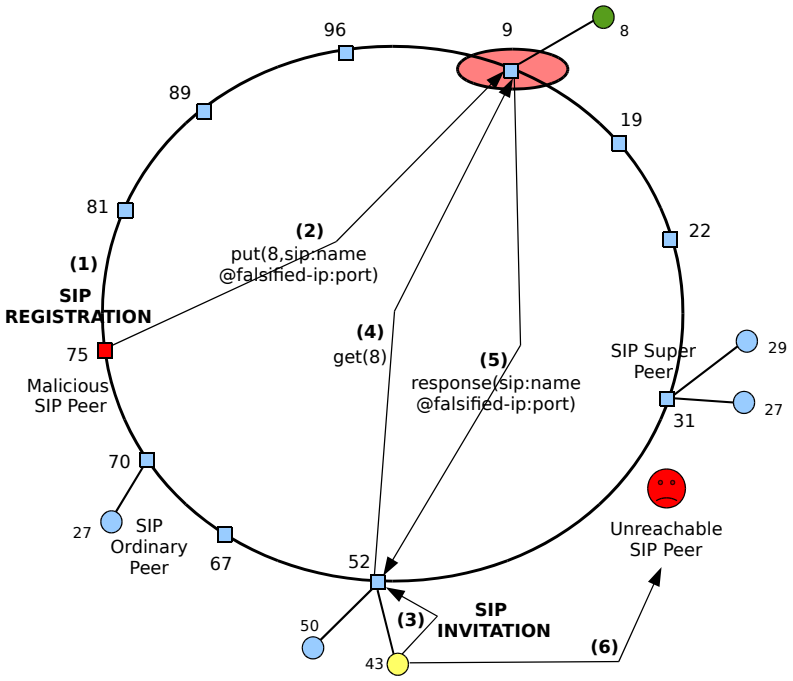


FIGURE 6.2 – Attaque par manipulation des enregistrements dans un réseau P2PSIP

Comme mentionné dans [101], les deux risques majeurs dans un réseau P2PSIP sont le déni de service et le détournement d'appels. Dans le cadre de notre travail, nous nous sommes in-

téressés notamment à deux scénarios d'attaque correspondant à la première catégorie. Le but est d'exploiter ces deux scénarios comme base pour l'expérimentation de notre stratégie de gestion des risques.

Le premier scénario, noté S_1 et représenté sur la figure 6.2, consiste à effectuer une attaque de manipulation par requêtes SIP REGISTER dans la table de hachage DHT fournissant le service d'enregistrement. L'objectif consiste à modifier une entrée d'enregistrement (étapes 1 et 2) afin de rendre le pair SIP injoignable dans le réseau P2PSIP (étapes 3 à 6). Cette attaque peut être effectuée par les attaquants des trois classes A_1, A_2 et A_3. Un attaquant qui joue le rôle d'un super-pair SIP (la classe d'attaquants A_2 est plus qualifié pour exécuter une telle attaque, par rapport aux deux autres classes. Un pair SIP ordinaire (la classe d'attaquants A_3) peut également exécuter cette attaque, mais il doit se servir d'un super-pair auquel il est associé pour transmettre les différents enregistrements. Bien qu'il n'implémente pas la pile protocolaire SIP, un attaquant de la classe A_1 est également capable de manipuler (soit par suppression ou modification) les entrées dans la table de hachage DHT.

Une autre version du scénario, notée S'_1, ciblant également le déni de service, consiste à injecter intelligemment des pairs *sybils* à côté du pair cible afin de modifier les réponses retournées par la table de hachage ; ces réponses correspondent aux requêtes envoyées par des clients SIP cherchant à établir une session SIP avec le pair attaqué. L'objectif est d'insérer des super-pairs SIP malveillants qui précèdent (l'ordre se fait par rapport à l'identité du pair) le pair cible dans l'anneau du réseau sous-jacent P2P. De cette manière, les pairs malveillants peuvent isoler le pair attaqué, et aussi ses entrées d'enregistrement SIP, en refusant de transmettre des réponses aux requêtes envoyées. Les conséquences de cette attaque sont plus importantes que celles du scénario précédent, parce que cette attaque permet de contrôler directement un segment du réseau P2PSIP. Cette attaque peut être réalisée par les classes d'attaquants A_1 et A_2, mais pas par la dernière classe d'attaquants A_3, parce qu'elle n'implémente pas la pile protocolaire P2P nécessaire pour générer et insérer des pairs *sybil*.

L'attaquant injecte périodiquement dans la table de hachage DHT, une entrée d'enregistrement correspondant à l'adresse d'enregistrement SIP-URI du pair attaqué associée à une adresse de contact falsifié. Quand un pair veut établir une session d'appel SIP avec le pair attaqué, il envoie un SIP INVITE au super-pair auquel il est attaché. Celui-ci tente de résoudre l'adresse d'enregistrement SIP-URI de la destination, en sollicitant la table DHT. Il obtient en retour l'adresse du contact falsifié. Le pair SIP ne sera pas en mesure de communiquer avec la destination parce que le service d'enregistrement mis en œuvre par la table de hachage n'est pas en mesure de fournir la bonne adresse IP.

Le deuxième scénario S_2 consiste à mettre en place la deuxième menace majeure dans le P2PSIP, la manipulation des appels. Il consiste à intercepter les appels dans le réseau P2PSIP, ainsi qu'à modifier les sessions et les manipuler. Nous avons développé un scénario de détournement d'appel basé sur l'attaque sybil [38]. Une attaque par sybils consiste à forger plusieurs identités pair à pair pour avoir plusieurs pairs dans un segment du réseau donné pour contrôler tous les messages sortants et entrants.

Notre scénario d'attaque de manipulation d'appels utilise les *sybils* pour réaliser une attaque du *man-in-the-middle* et ainsi rendre l'attaquant capable de manipuler l'appel. Les nœuds de sybil sont injectés autour du nœud porteur de l'enregistrement du pair attaqué afin de surveiller toutes requêtes de recherche de clé ou d'enregistrement de la part du pair SIP cible : les pairs *sybils* prennent le contrôle du segment responsable de la clé.

Le scénario S_2 consiste ainsi à forger plusieurs identités (node-ID) et à les affecter aux différents pairs afin qu'ils soient logiquement (en suivant l'algorithme de la DHT) proche de la clé cible. De cette façon, ils sont capables d'intercepter toute demande d'informations sur la clé con-

cernée. Ensuite, lorsqu'un pair SIP A veut contacter le pair B attaqué, les *sybils* envoient comme réponse à toute requête une adresse de contact falsifiée (par exemple une adresse IP qui mène vers un des *sybils*). Après, le nœud A utilise l'adresse du contact falsifiée pour établir une session avec la destination B. Par conséquent, ses messages de signalisation seront partis au pair sybils, qui rechange l'adresse du contact dans les requêtes SIP (champs Contact) par la bonne adresse du B et change le champs Via pour que toutes les réponses aux messages de signalisation de la part du pair SIP B passent par eux. A chaque fois que les pairs *sybils* reçoivent des requêtes SIP, ils utilisent sur les deux champs du *Via* et *Contact* afin que les deux pairs SIP A et B n'observent rien d'anormal. Dans ce contexte, l'attaquant peut par exemple changer les paramètres des codecs audio, ou les utiliser pour interpréter les messages vocaux, ou même imposer une route pour les messages RTP. Ainsi, il peut écouter l'appel et réaliser une analyse du trafic.

Il est évident que ces menaces de sécurité peuvent être évitées grâce à l'introduction de techniques d'authentification ou de certification par exemple l'utilisation d'une PKI [23]. Toutefois, le principal défi, dans le cas du P2PSIP, est de fournir un service ouvert P2P VoIP qui est sûr à la fois, performant et vraiment décentralisée. Ces techniques sont généralement en contradiction avec la dernière contrainte (quand ils s'appuient sur une autorité de certification centralisée [20, 93]) ou posent des problèmes importants à l'égard de la deuxième contrainte (en termes de performance et de capacité de déploiement).

Pour respecter les deux contraintes précédentes, nous proposons une solution dynamique, capable d'adapter les techniques de protection en cas de menace et réduire les coûts qui en résultent dans les réseaux P2PSIP. Dans la section suivante, nous présentons notre solution de risques adaptée à l'architecture distribuée liée au protocole P2PSIP.

6.3 Gestion des risques appliquée au P2PSIP

Dans ce chapitre, nous proposons une stratégie de gestion des risques pour adapter l'exposition des réseaux P2PSIP à partir d'un ensemble de contremesures. Nous avons déjà montré les avantages d'une telle approche pour les réseaux centralisés VoIP d'entreprise [78]. Ce besoin est également fort dans les réseaux P2PSIP. Comme il est souligné par Brian et al. dans [10], **"tout protocole P2PSIP doit offrir une gamme de modèles de sécurité qui peuvent être sélectionnés en fonction des besoins de l'architecture"**. Notre approche de gestion des risques fournit de nouvelles perspectives pour permettre de cette sélection d'une façon automatique, ou du moins elle suggère aux administrateurs des solutions de sécurité adaptées en fonction du contexte actuel du réseau de façon semi-automatique. La sélection des contremesures permet de contrôler dynamiquement l'exposition du réseau face aux risques.

Le risque est la combinaison de la probabilité qu'une menace donnée exerce une vulnérabilité sur un système et l'impact résultant de cet événement négatif sur ce système. Nous rappelons ici l'équation qui définit le risque où a est une attaque et A est l'ensemble des attaques de sécurité potentielles (équation 6.1).

$$\mathcal{R} = \sum_{a \in A} \mathcal{P}(a) \times \mathcal{E}(a) \times \mathcal{C}(a) \tag{6.1}$$

Le paramètre $\mathcal{P}(a)$ représente la potentialité de la menace liée à l'attaque, $\mathcal{E}(a)$ représente l'exposition de l'infrastructure face à cette menace (basée sur l'ensemble actuel de vulnérabilités),

23. Public Key Infrastructure

et $\mathcal{C}(a)$ quantifie les conséquences de cette attaque sur les ressources de l'infrastructure, ce qui correspond à la dégradation des actifs.

$$maintenir(R_{new} < R_{th})\ and\ minimiser(\sum_{i} cost(sf_i)) \tag{6.2}$$

Un autre paramètre important à prendre en compte lors de la sélection des contremesures sont les coûts induits à leurs applications. Considérons $Sf = \{sf_1, \ldots, sf_n\}$, représentant l'ensemble des contremesures disponibles, l'objectif de l'approche de gestion des risques, telle que définie par l'équation 6.2, est de maintenir le niveau de risque calculé inférieur à une valeur du seuil, notée R_{th}, tout en minimisant les coûts induits par les contremesures activées.

Afin d'appliquer notre stratégie de gestion de risques pour les réseaux P2PSIP, nous devons d'abord identifier un ensemble des contremesures dont l'application est possible dans ces réseaux. Elles doivent respecter les spécificités des architectures P2PSIP comme potentiellement l'absence d'une autorité centrale et la tolérance aux pannes. Nous serons amenés ensuite à adapter notre modèle quantitatif des risques pour l'instancier dans ce contexte spécifique.

6.3.1 Portfolio de contremesures

Nous avons identifié un ensemble de contremesures dans le contexte des scénarios d'attaques P2PSIP liés au déni de service. Le but n'est pas d'établir une liste exhaustive, mais de se concentrer sur une variété des contremesures P2PSIP pour soutenir notre gestion adaptative des risques. Nous décrivons brièvement chacune de ces contremesures et leurs propriétés ci-dessous :

- **Correction par des enregistrements SIP** : cette contremesure consiste à envoyer périodiquement des messages SIP REGISTER à une fréquence adaptée pour corriger une entrée falsifiée. L'objectif est de vérifier l'adresse de contact SIP-URI (contenant l'adresse IP) associée à une donnée d'enregistrement (AoR) SIP-URI, et de corriger cette entrée dans la DHT, dès que la falsification a été détectée. La détection est effectuée par un nœud de confiance qui interroge la valeur associée à une adresse d'enregistrement donnée. Si la valeur trouvée diffère de la valeur de contact attendue, le pair envoie un nouveau message SIP REGISTER afin de remplacer la valeur falsifiée. La fréquence des vérifications et des corrections est progressivement adaptée à la fréquence de la pollution. Cette contremesure n'est pas efficace en cas d'attaques par des pairs sybils, et peut générer un coût non négligeable quantifiable en termes de requêtes envoyées.
- **Réplication des identifiants SIP** : cette contremesure consiste à enregistrer des identifiants SIP répliqués dans la table de hachage afin d'éviter la manipulation des entrées falsifiées et avoir toujours une adresse d'enregistrement SIP-URI accessible. L'objectif est de multiplier les identifiants SIP des pairs attaqués. Cette réplication peut généralement se faire en appliquant plusieurs fois la fonction de hachage sur l'adresse SIP-URI. Ces identifiants SIP répliqués sont situés dans des segments différents de la DHT, ce qui complique l' élaboration des attaques à base de *sybils* et augmente le coût de son exécution. Cette contremesure suppose que, lors de l'initiation des sessions d'appel, le super-pair effectue plusieurs requêtes dans la DHT pour obtenir les valeurs associées aux différents réplicats de la même adresse d'enregistrement, et de les comparer afin de détecter les incohérences.
- **Adaptation du routage P2PSIP** : la stratégie de routage dans la DHT de l'architecture P2PSIP est généralement effectuée d'une manière récursive [100]. Une stratégie récursive est une technique de routage moins coûteuse par rapport à une stratégie itérative. Alors que cette dernière est plus chère en terme de trafic, la stratégie itérative du routage constitue

une approche intéressante qui permet davantage de contrôle sur les routes empruntées par les messages P2PSIP, en particulier pour la requête SIP REGISTER. L'objectif est d'éviter les pairs intermédiaires (classe d'attaquants A_2) qui peuvent modifier les messages P2PSIP pendant la phase d'enregistrement ou la phase d'initiation de session.

- **Restrictions de communication aux pairs de confiance** : cette contremesure consiste à améliorer la précédente en limitant les interactions à un sous-ensemble de pairs de confiance. Plusieurs algorithmes de confiance et de réputation ont déjà été proposés dans le contexte des réseaux pair à pair, et peuvent être facilement transférés dans les infrastructures P2PSIP. L'objectif est de minimiser la probabilité d'interagir avec un pair malveillant. Plusieurs métriques peuvent être envisagées dans ce contexte ; une métrique évidente est la probabilité qu'un pair SIP peut résoudre correctement les entrées d'enregistrement SIP qu'il prend en charge. Typiquement, les contrôles effectués par la première contremesure permettent de quantifier cette probabilité. Une autre métrique intéressante peut être évaluée à l'aide des techniques de détection proposées dans [25]. Ces techniques de détection permettent d'identifier d'éventuels pairs *sybil* en analysant la distribution des identifiants SIP et la comparant par rapport à la distribution théorique grâce à la formule de divergence de Kullback-Leibler. Ces techniques de détection sont passives et ne génèrent pas de trafic supplémentaire dans le réseau P2P. Elles permettent de minimiser la présence de pairs Sybil dans le réseau P2PSIP.
- **Élimination des pairs *sybils* de la table de routage** en utilisant la technique de détection de [36], nous pouvons éviter la manipulation des nœuds sybil. Cette contremesure dépend de la technique de détection qui attribue la probabilités d'existence des pairs sybil en fonction du nombre de bits en commun avec la clé. Son implémentation n'est pas coûteuse car elle est passive.
- **Authentification et auto-certification** : la dernière contremesure vise à authentifier les pairs SIP qui interagissent dans le réseau P2PSIP, sans violer la contrainte de décentralisation, c'est-à-dire sans utilisation d'une autorité de certification centralisée. Un mécanisme d'authentification par l'intermédiaire du service e-mail a été proposé dans [102] pour les réseaux P2PSIP. L'adresse AoR SIP-URI dois dans ce cas être identique à l'adresse e-mail du client SIP. Le service e-mail est utilisé comme un vecteur pour partager un secret entre le pair SIP qui enregistre l'entrée SIP et le pair SIP qui effectue la localisation. Cette solution est fortement dépendante du service e-mail. Nous considérons plutôt, pour cette dernière contremesure, un mécanisme d'auto-certification tel que décrit dans [94] pour permettre l'authentification des pairs SIP dans l'infrastructure P2PSIP. Dans cette approche, chaque pair SIP génère une pair de clés RSA (clé publique et clé privée) ; la clé publique est intégrée dans l'adresse d'enregistrement SIP-URI. Quand un pair SIP s'enregistre auprès de la table DHT, il signe son adresse de contact SIP-URI avec sa clé privée. Quand un utilisateur SIP veut établir une session d'appel avec un autre, il vérifie l'authenticité des valeurs d'enregistrement en utilisant la clé publique. Cette mesure de sécurité impose des contraintes de capacité et de déploiement pour les pairs, mais elle ne nécessite pas d'entité centrale.

6.3.2 Détection d'attaques dans le P2PSIP

Soit $V = \{v_1, \ldots, v_n\}$ l'ensemble des n pairs SIP dans le réseau P2PSIP. Chaque pair SIP v_i est associé à au moins une adresse d'enregistrement SIP-URI, notée $AoR(v_i)$, et une adresse de contact SIP-URI (une adresse SIP-URI associée à une adresse IP), notée $contact(v_i)$, contenant l'adresse IP. Soit h la fonction de hachage qui fournit la clé d'une entrée d'enregistrement SIP

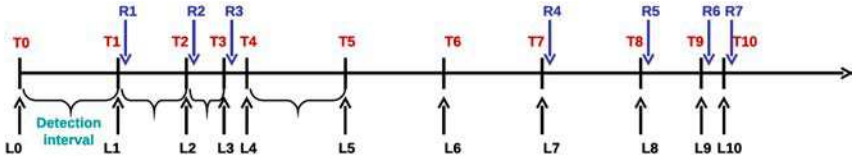


FIGURE 6.3 – Détection de l'attaque de déni de service dans le P2PSIP

dans la table de hachage distribuée. La clé d'un pair SIP est, donc, donnée par $h(AoR(v_i))$ est associée à une adresse de contact $contact(v_i)$.

La détection d'une entrée falsifiée est effectuée en procédant périodiquement à l'interrogation de la table de hachage distribuée. Soit T l'espace temps divisé en plusieurs intervalles de temps non homogènes $[t_i,\ t_{i+1}]$ où t_i indique l'instant auquel la i^{th} vérification est effectuée dans le réseau P2PSIP.

Au cours de la phase de détection, la requête auprès de la DHT est généralement effectuée par un pair SIP v_j différent du pair v_i dont l'entrée SIP enregistrée est concernée par le contrôle. L'objectif est d'empêcher l'attaquant de modifier son comportement (ou son attaque) quand il remarque que le pair SIP exécutant la requête est le propriétaire de l'entrée SIP concernée. La fréquence de ces requêtes DHT est adaptative au cours du temps et peut être augmentée quand une entrée falsifiée est détectée. Les contrôles sont effectués comme suit, avec un temps de l'instant initial t_0 mis à 0 et un intervalle de temps initial $[t_0,\ t_1]$ défini par la valeur constante ϕ :

- Initialement, le pair SIP v_j connaît à la fois l'adresse d'enregistrement SIP-URI $AoR(v_i)$ du pair v_i et son adresse de contact $contact(v_i)$. Il commence par calculer la clé $h(AoR(v_i))$ du pair v_i, en utilisant la fonction de hachage h. Il exécute ensuite une requête de recherche P2P FIND pour cette clé dans la DHT. Il obtient en retour la valeur, notée $contact'(v_i)$, associée à cette clé.
- Si la valeur reçue $contact'(v_i)$ est différente de la valeur attendue $contact(v_i)$, la fréquence de la prochaine vérification est augmentée tant que l'intervalle du temps $[t_{i-1},t_i]$ est supérieure à une valeur limite ϕ_{min} ; la prochaine vérification est faite à l'instant t_{i+1} défini par l'équation 6.3.

$$t_{i+1} = t_i + \frac{t_i - t_{i-1}}{2} \tag{6.3}$$

- si la valeur reçue $contact'(v_i)$ est égale à la valeur attendue $contact(v_I)$, l'entrée d'enregistrement n'est pas considérée comme falsifiée et la fréquence de la prochaine vérification reprend la fréquence initiale, ce qui signifie que le prochain instant du contrôle est $t_{i+1} = t_i + \phi$.

Ce système de détection peut être affiné par l'introduction d'un automate multi-états. Dans le contexte de cette méthode de détection, la potentialité est directement calculée sur la base de cette fréquence de vérification, qui montre l'intensité de l'attaque au cours du temps. Quand la potentialité augmente, notre solution de gestion des risques pour les réseaux P2PSIP détermine si le système a besoin de nouvelles contremesures à activer fans le réseau P2PSIP.

6.3.3 Modélisation du risque

En nous basant sur ce portfolio de contremesures, nous décrivons dans cette partie le modèle de risque quantitatif qui soutient notre stratégie de gestion des risques pour les réseaux P2PSIP. Ce modèle mathématique concerne les scénarios d'attaque par déni de service décrit dans la

section précédente. Il est dérivé de l'équation 6.1 qui quantifie le niveau de risque. L'activation des contremesures permet de réduire l'exposition du réseau P2PSIP quand le niveau de risque est élevé, mais il implique également une charge supplémentaire qui peut affecter les performances du réseau en fonction de sa configuration. Ce compromis préside à nouveau notre stratégie dynamique de gestion des risques.

La potentialité permet de quantifier l'intensité d'une menace de sécurité dans le réseau P2PSIP (la probabilité de son occurrence). Dans le cas des scénarios d'attaques que nous avons identifié, cette mesure est directement liée à la technique de détection des entrées falsifiées utilisée dans le réseau P2PSIP. Avec notre technique de détection, elle est calculée en fonction de la fréquence d'envoi des requêtes P2P FIND pour vérifier s'il y a une attaque ou non. La potentialité peut être donnée par l'équation 6.4 où T_{verif} représente la période entre deux requêtes P2P FIND destinées à vérifier si l'enregistrement concerné est correct ou pas dans la table DHT.

$$1/T_{verif} \tag{6.4}$$

Tel que défini dans l'équation 6.1, un autre paramètre important pour estimer le niveau de risque est la conséquence d'une attaque réussie $\mathcal{C}(a)$. Dans le cadre des scénarios d'attaque liés à un déni de service, l'objectif est de déterminer les dommages qui se produisent dans le réseau P2PSIP, lorsque l'attaquant réussit à rendre un ou plusieurs pairs SIP attaqués injoignables. Une technique intuitive pour quantifier ces conséquences consiste à considérer les sessions SIP des appels entrants qui sont perdus lorsque le client SIP appelant obtient une adresse de contact SIP-URI falsifiée.

Soit O et S deux sous-ensembles de V, représentant respectivement l'ensemble des pairs ordinaires SIP et l'ensemble des super-pairs SIP. Soit $A(v_i)$ l'ensemble des pairs SIP ordinaires logiquement rattachés à un super-pair ($v_i \in S$). Soit $InAvg$ la fonction qui donne le nombre moyen de sessions d'appels entrants pour un pair SIP donné $v_i \in V$ au cours d'une période de temps régulière. Afin de quantifier les conséquences, il est important de distinguer le cas où le nœud SIP ciblé est un pair ordinaire, du cas où il joue le rôle d'un super-pair dans le réseau P2PSIP :

- Si le pair v_i est un pair SIP ordinaire ($v_i \in O$), les conséquences d'une attaque réussie peuvent être directement calculées par la valeur $InAvg(v_i)$, qui permet d'estimer le nombre moyen de sessions d'appels perdues.
- Si le pair v_i est un super-pair SIP ($v_i \in S$), la formule des conséquences devrait également prendre en compte les nœuds SIP ordinaires qui sont logiquement attachés au super-pair SIP concerné. Nous ajoutons, ainsi, à la valeur de $InAvg(v_i)$ un autre paramètre : $\sum_{v_j \in A(v_i)} InAvg(v_j)/r$ avec r le paramètre spécifiant le niveau de réplication.

Le but de cette quantification est de déterminer l'importance d'un pair SIP donné, et donc l'impact sur le réseau si ce pair SIP est inaccessible en raison d'une attaque de déni de service. Il est possible d'affiner ce paramètre en intégrant des paramètres supplémentaires, en tenant compte, par exemple, du fait que les sessions d'appels SIP ne possèdent pas la même importance.

Le dernier paramètre de l'équation 6.1 correspond à l'exposition du réseau P2PSIP. Nous contrôlons dynamiquement cette exposition par l'activation ou la désactivation des mesures de sécurité : l'activation d'une contremesure permet de limiter l'exposition du réseau P2PSIP, tandis que sa désactivation permet d'augmenter son exposition. Soit l'ensemble $Sf = \{sf_1, \ldots, sf_i\}$ déjà défini dans la partie 6.3.1, soit $active(sf_i)$ est une fonction qui indique si la contremesure sf_i est activée ou pas. On rappelle l'exposition $\mathcal{E}(a)$ du réseau P2PSIP par rapport à une attaque donnée par l'équation 6.5.

$$\mathcal{E}(a) = 1 - \sum_i \sigma_i(a) active(sf_i) \tag{6.5}$$

Dans cette équation, la valeur du σ_i quantifie l'impact de la contremesure sf_i sur l'exposition du réseau. L'exposition est maximale quand aucune des contremesures disponibles est activée ($\forall\ sf_i \in SF, active(sf_i) = 0$). Dans les scénarios d'attaque identifiés, nous considérons trois classes d'attaquants A_1, A_2 et A_3, et nous définissons la valeur d'impact tel que spécifiée par l'équation 6.6.

$$\sigma_i = \sum_j p(A_j) \times p(sf_i|A_j) = \sum_j p(A_j) \times \alpha_{ij} \times \delta_i \tag{6.6}$$

La probabilité $p(A_j)$ représente la probabilité d'existence de la source de l'attaque A_j, tandis que $p(sf_i|A_j)$ indique la probabilité que la contremesure sf_i contre la source d'attaque A_j. Les trois probabilités $p(A_1)$, $p(A_2)$ et $p(A_3)$ ont été considérées comme équiprobables dans le contexte de notre travail, mais ces probabilités peuvent être affinées en se basant sur une analyse statistique. La probabilité $p(sf_i|A_j)$ peut être décomposée comme un produit de deux termes élémentaires. Le premier terme, noté α_{ij}, quantifie l'impact de la contremesure sf_i sur la source d'attaque A_j, tandis que le second terme, noté δ_i correspond aux caractéristiques intrinsèques de la contremesure sf_i, comme la fréquence de corrections par la requête SIP REGISTER, ou le nombre d'identités SIP répliquées dans le réseau P2PSIP.

La décomposition de la valeur de l'impact σ_i pour une contremesure donnée sf_i peut facilement être représentée comme un arbre de probabilité. L'activation des contremesures peut être exclusive ou cumulative : les impacts des différentes contremesures cumulatives peuvent se chevaucher par rapport à une attaque de sécurité donnée. Nous considérons que ce problème est un problème à part entière lors du champs d'investigation de cette thèse.

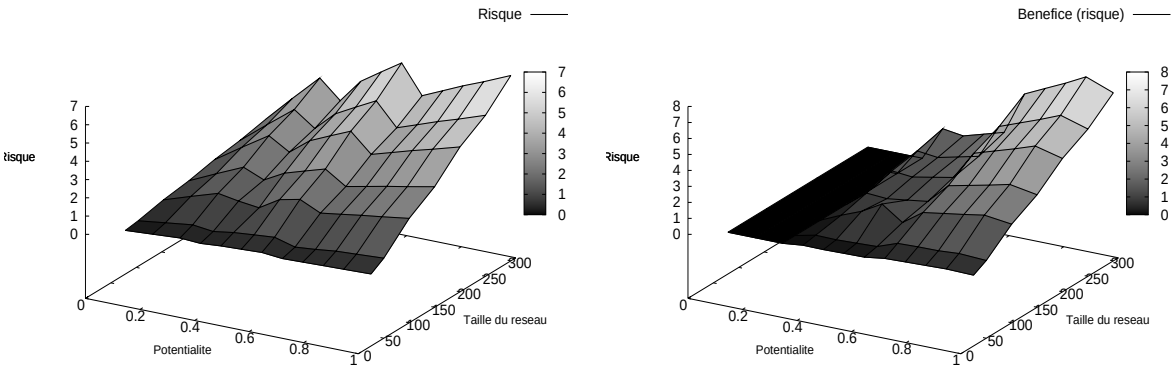
Coût d'application des contremesures

Le coût d'application d'une contremesure n'est pas nécessaire pour évaluer le niveau de risque et ne figure donc pas dans l'équation 6.1. Cependant, ce paramètre joue un rôle principal pour la sélection des contremesures, comme mentionné dans l'équation 6.2. Le mécanisme de gestion des risques tente en permanence de réduire le niveau de risque tout en minimisant le coût d'application des contremesures. Nous entendons, par coût des contremesures, le coût supplémentaire qui doit être pris en charge par l'infrastructure P2PSIP et les utilisateurs non malveillants. Dans ce cadre, nous avons décomposé le coût des contremesures en trois paramètres élémentaires : le premier paramètre correspond à la surcharge en trafic dans le réseau P2PSIP. Ainsi, pour calculer ce premier paramètre, il suffit de quantifier le nombre de messages de signalisation supplémentaires requis pour l'exécution de la contremesure. Par exemple, avec la première contremesure, ce coût en terme de trafic comprend les messages SIP REGISTER nécessaires pour corriger les entrées falsifiées. Le deuxième paramètre correspond à la surcharge en terme de temps à l'initiation d'une session d'appels SIP. Cela exprime le délai supplémentaire qu'un utilisateur régulier doit attendre avant d'effectivement établir la session d'appel. Par exemple avec la seconde contremesure, il correspond à la durée nécessaire pour obtenir et comparer les valeurs des identifiants SIP répliqués. Le dernier paramètre exprime le coût de déploiement des contremesures, ce coût est souvent non négligeable, en particulier pour la dernière contremesure.

6.4 Évaluation de la gestion de risques dans le P2PSIP

Afin d'évaluer les performances de notre solution de gestion des risques, nous avons mené un ensemble d'expériences en utilisant le simulateur OMNET++ [24], combinée avec OverSim. Au cours de ces expériences, nous avons considéré un réseau P2PSIP composé d'un maximum de 300 pairs ; ce réseau utilise un algorithme distribué pour la table de hachage DHT. Nous nous sommes intéressés au scénario d'attaques décrit dans la section 6.2.2, avec un attaquant qui tente de manipuler les entrées d'enregistrement SIP dans la table de hachage distribuée, afin de rendre un ou plusieurs pairs SIP injoignables. Nous supposons que 5% de pairs SIP sont susceptibles d'être attaqués dans le réseau P2PSIP et nous avons implémenté trois des cinq contremesures, à savoir la correction des enregistrements SIP, la réplication des identifiants SIP, et la réplication associée à la restriction de communication avec des pairs SIP de confiance. Au cours de ces expériences, nous avons mesuré le niveau de risque ainsi que le coût induit par l'application des contremesures, et nous avons comparé notre solution de gestion de risques à deux autres stratégies : une stratégie ouverte, notée ψ_1, qui consiste à minimiser systématiquement le coût induit par l'application d'une contremesure, et une stratégie fermée, notée ψ_2, qui consiste à systématiquement minimiser le niveau de risque.

6.4.1 Analyse du risque



(a) L'évolution du risque dans le réseau P2PSIP (b) Comparaison avec une stratégie régulière

FIGURE 6.4 – Traitement du risque dans le P2PSIP

Dans une première série d'expériences, nous nous sommes intéressés à analyser le comportement de notre stratégie de gestion des risques vis à vis du niveau de risque dans un réseau P2PSIP. D'après notre modèle quantitatif, le risque représente le nombre d'enregistrements dans la DHT altérés ou manipulés. Au cours de ces expériences, nous avons quantifié le niveau de risque en fonction de la potentialité de la menace pour différentes tailles de réseaux P2PSIP. Ces résultats expérimentaux sont synthétisés sur la figure 6.4. Les deux premiers axes correspondent respectivement à la potentialité de la menace et la taille du réseau, tandis que le troisième axe indique le niveau de risque mesuré. Nous varions la potentialité de la menace en faisant varier la fréquence à laquelle l'attaquant injecte des messages d'enregistrement SIP falsifiés dans la table

24. OMNeT++ Network Simulation Framework, www.omnetpp.org

de hachage distribuée. La taille du réseau varie de 0 (valeur théorique) à 300 pairs SIP et le niveau de risque mesuré correspond au niveau de risque efficace (c'est-à-dire la valeur de risque après l'application des contremesures). Nous pouvons observer clairement sur ce graphique comment la potentialité de la menace a un impact sur le niveau de risque. Initialement, la potentialité est nulle, ainsi, le niveau de risque est nulle, tandis que la valeur de l'exposition est égale à 1 car il n'y a aucune contremesure activée. Quand la potentialité augmente au cours du temps, l'algorithme de restriction du risque active progressivement les contremesures afin de restreindre l'exposition du réseau. Soit un réseau composé de 100 pairs SIP, la première contremesure activée est la correction des enregistrements SIP erronés. En effet, l'objectif de cette contremesure consiste à compenser la pollution générée par l'attaquant, en effectuant les mises à jour des enregistrements avec une fréquence de message élevée. Cette contremesure réduit l'exposition, et donc, réduit partiellement le niveau de risque. Toutefois, la potentialité ne cesse de croître dans cette expérience. Quand la potentialité atteint 0,3, l'algorithme de restriction active la deuxième contremesure, qui permet de multiplier les identifiants SIP-URI du client concerné. Le nombre de réplicats n'est pas fixe, il varie entre deux et cinq répliques pour donner plus de flexibilité à la solution de sécurité, protéger les utilisateurs SIP et en même temps contrôler le problème de charge induit par leur application. Les réplicats SIP-URI sont obtenus par application successive de la fonction de hachage sur la première SIP-URI : $h(AoR(v_i)), h(h(AoR(v_i))), h(h(h(AoR(v_i))))$.

La potentialité de la menace continue de croître et atteint une valeur de 0,5, ce qui traduit que l'attaquant poursuit son attaque. A ce niveau, l'algorithme active la troisième contremesure, qui permet de limiter les interactions avec les pairs et recommande la communication avec les pairs SIP de confiance. Ce qui conduit à une nouvelle restriction de l'exposition du réseau. Par conséquent, le niveau de risque tombe à une valeur proche de 1. Comme aucune autre contremesure n'est disponible pour restreindre l'exposition, le niveau de risque continue à croître jusqu'à ce que la potentialité atteint sa valeur maximale ; le même phénomène est observé pour les différentes tailles de réseau. Toutefois, plus la taille du réseau est grande, plus le niveau de risque est élevé. Ceci s'explique par le fait que nous considérons que 5% des pairs SIP sont susceptibles d'être attaqués dans le réseau.

Le niveau de risque le plus élevé obtenu dans les expériences, est égal à 7 ce qui signifie que l'on risque d'avoir 7 pairs SIP attaqués et inaccessibles dans le réseau. Nous pouvons également déduire le niveau de risque exprimé en terme de sessions d'appels perdus, tel que défini par l'équation 6.1. Nous avons également comparé notre solution aux deux stratégies mentionnées précédemment ψ_1 et ψ_2. Nous nous attendions à ce que notre stratégie de gestion des risques surpasse la stratégie ψ_1, ce qui a été le cas au cours de ces expériences. Nous avons tracé le bénéfice en termes de risque de notre stratégie par rapport à la stratégie ψ_1 sur la figure 6.4. Nous entendons par bénéfice la différence en terme de risque entre le niveau de risque obtenu avec ψ_1 et celui obtenu avec notre stratégie. Nous avons quantifié un bénéfice moyen d'environ 6 pairs SIP attaqués qui sont sécurisés par notre solution. Nous avons également comparé notre solution à la stratégie ψ_2 et comme attendu, il n'y a aucun avantage par rapport à ψ_2. Cependant, la stratégie ψ_2 génère plus de charge (trafic et délai) que notre approche, comme nous le verrons dans la prochaine section.

6.4.2 Coût induit en termes de trafic

Les trois contremesures considérées génèrent du trafic supplémentaire de signalisation dans le réseau P2PSIP. Ce trafic est requis pour corriger les entrées falsifiées dans le cas de la première contremesure, et pour multiplier les identités SIP dans le cas des deuxième et troisième contremesures. Nous avons également intégré dans ce coût induit le trafic utilisé pour la phase de

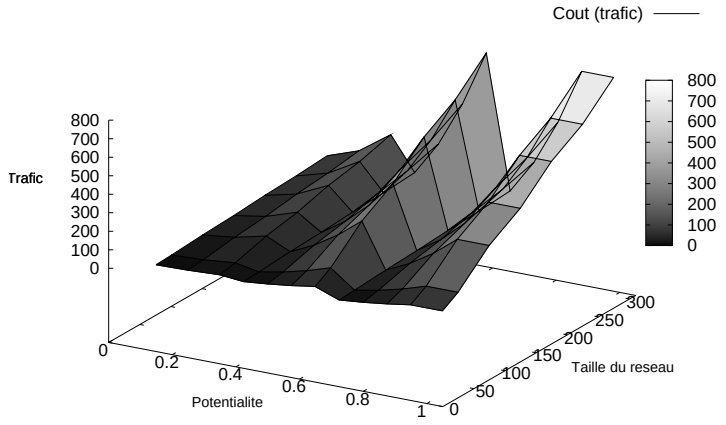


FIGURE 6.5 – Impact du traitement des risques sur le trafic

détection, comme celle-ci est fortement corrélée à la phase de traitement du risque. La surcharge en trafic dépend des contremesures activées et aussi de ses caractéristiques intrinsèques, telles que la fréquence des corrections des enregistrements SIP falsifiés et le nombre d'identités SIP répliquées, tel que discuté à la section 6.3.1.

Les résultats expérimentaux sont représentés dans la figure 6.5 indiquant le nombre de messages de signalisation, en fonction de la potentialité de la menace et de la taille du réseau. Nous pouvons observer que la première contremesure est globalement moins coûteuse que les deux autres. En effet, cette contremesure nécessite la consultation de la table de hachage distribuée du réseau P2PSIP pour vérifier l'authenticité des entrées d'enregistrement (en utilisant les messages P2P GET et leurs réponses), puis pour exécuter la correction en mettant à jour et corrigeant les entrées falsifiées (en utilisant des messages P2P PUT et leurs réponses). En comparaison, les deux autres contremesures nécessitent de générer des multiples enregistrements (en utilisant des messages P2P PUT et leurs réponses) correspondant aux différentes identités SIP répliquées. Ces résultats expérimentaux sont en accord avec les constatations sur notre stratégie de gestion des risques : quand la potentialité augmente, la stratégie réduit progressivement l'exposition du réseau, ce qui nécessite des contremesures plus coûteuses en termes de trafic et délai.

Coût induit en termes de délai d'établissement de session

Le coût peut aussi être quantifié en termes de surcharge en temps, tel que décrit à la section 6.3.1. Nous pouvons inclure dans ce paramètre le temps nécessaire pour déployer les contremesures, et le temps supplémentaire nécessaire pour établir une session d'appel SIP. Le temps de déploiement correspond à la durée de l'application des contremesures sélectionnées. Nous avons mesuré la durée entre l'émission du premier message de la contremesure, et la réception du dernier message de la même contremesure. Nous avons tracé sur la figure 6.6 ces valeurs pour différentes potentialités de la menace et différentes tailles du réseau. Comme nous l'avons constaté précédemment, la première contremesure est globalement moins coûteuse que les deux autres en terme de délai de déploiement ; elle montre un temps de déploiement maximum de 3 secondes au cours de ces expériences. Le coût de déploiement relativement élevé peut être expliqué par le temps nécessaire à la propagation des identités SIP répliquées dans la DHT.

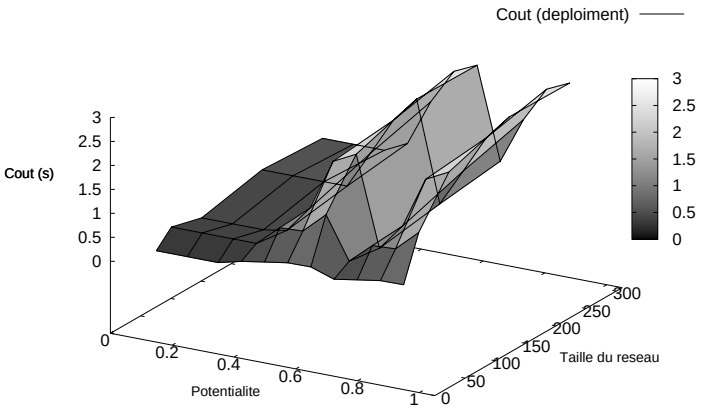
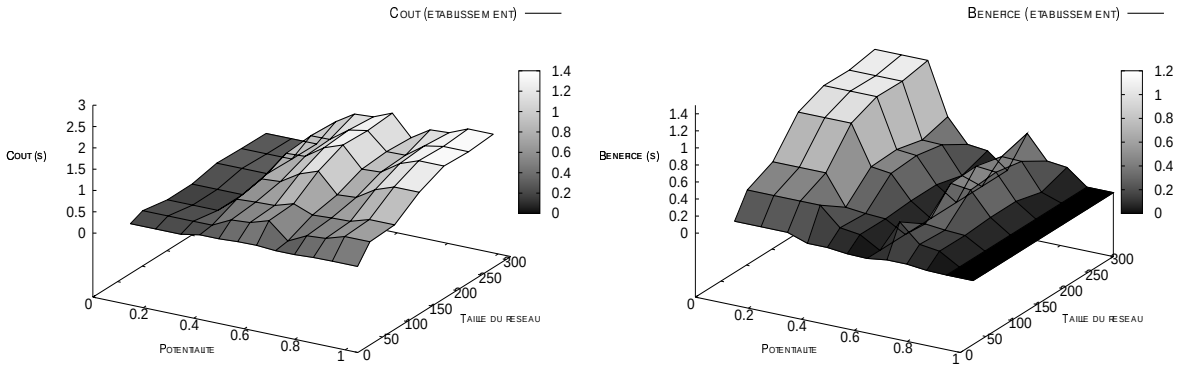


FIGURE 6.6 – Temps requis pour le déploiement des contremesures

Une autre surcharge importante est le temps supplémentaire expérimenté par les utilisateurs finaux à l'établissement d'une session SIP d'un nouvel appel. Nous avons présenté cette fois le délai supplémentaire sur la figure 6.7 en fonction de la potentialité et de la taille du réseau. Ces valeurs ont été obtenues par la comparaison des temps d'établissement d'une session (temps entre l'envoi du premier message SIP INVITE et la réception du dernier message SIP OK) avec et sans les contremesures sélectionnées. Nous pouvons clairement observer que les deux dernières contremesures sont plus coûteuses avec un délai supplémentaire d'un maximum de 1,4 secondes au cours de ces expériences. En effet, ces contremesures ont besoin d'avoir les différentes adresses SIP-URI répliquées. Ces adresses sont ensuite comparées avant de contacter le pair SIP concerné, ce qui n'est pas le cas avec la première contremesure.



(a) Surcharge en temps à l'établissement de sessions (b) Comparaison avec une stratégie régulière

FIGURE 6.7 – Impact du traitement du risque sur l'établissement de sessions

Dans le même contexte, nous avons également comparé notre stratégie de gestion des risques aux deux autres stratégies ψ_1 et ψ_2. La stratégie ψ_1 introduit un surcoût de temps relativement réduit à l'établissement de session, mais elle montre des mauvaises performances en termes de

niveau de risque, comme le montre la figure 6.4(b). La comparaison de notre stratégie avec la stratégie ψ_2 est détaillée sur la figure 6.7(b). Notre stratégie surpasse globalement la stratégie ψ_2 : la meilleure prestation étant observée avec une potentialité de menace faible.

Ces différents résultats illustrent les avantages et les limites de notre stratégi de gestion des risques. Celle-ci permet d'adapter dynamiquement l'exposition en fonction de la potentialité de menace, cette adaptation étant pilotée par un compromis entre le niveau de risque et le coût induit à l'application des contremesures.

6.5 Synthèse

Notre objectif consiste à permettre à la fois la sécurité et un niveau élevé de performances pour un service P2PSIP ouvert et décentralisé. Afin de répondre à ce compromis, nous avons défini une stratégie pour contrôler dynamiquement l'exposition d'un réseau P2PSIP grâce à un ensemble de contremesures. Nous avons décrit les mécanismes sous-jacents et détaillé l'instanciation de cette stratégie basée sur une modélisation des risques. Nous avons identifié des scénarios d'attaques et établi un ensemble de contremesures. En particulier, nous avons montré comment cette stratégie est capable de maintenir le niveau de risque tout en minimisant les coûts induits par les contremesures. Nous avons également observé comment le choix des contremesures est influencé par la potentialité de la menace et la taille du réseau. Ce travail est compatible avec les exigences exprimées précédemment dans [10] et [84] sur la pluralité des protections et leur sélection dans ces environnements.

Notre approche de gestion des risques traite les deux menaces majeures dans les réseaux P2PSIP à savoir l'interception d'appels et le déni de services. Nous avons montré que les contremesures proposées génèrent un trafic supplémentaire. De même, la technique de détection génère de trafics. Nous avons volontairement établi le cas des réseaux P2PSIP réellement décentralisés. Le framework RELOAD résout ce problème mais introduit alors un serveur de certification central dans le réseau P2PSIP.

Chapitre 7

Architectures hybrides et externalisation dans le cloud

Sommaire

7.1 Introduction

Dans ce chapitre, nous traitons des architectures VoIP hybrides notamment du framework RELOAD[25] et de l'externalisation des contremesures dans le cloud. Le framework RELOAD a été proposé dans les réseaux P2PSIP pour faciliter l'intégration de services et également traiter certains problèmes de sécurité. Il s'appuie sur un serveur central de certification qui renforce la sécurité dans le P2PSIP. En particulier, elle offre trois niveaux de sécurité, le premier pour la génération des identités de pairs, le deuxième pour l'échange de messages et le troisième pour l'enregistrement des adresses SIP-URI dans la table de hachage distribuée (DHT). Cependant, cette solution présente de nombreuses limites, en particulier contre certaines attaques telles que les attaques comme le refus et le déni de service. Pour cela, nous proposons une solution complémentaire à base d'algorithme de confiance et de réputation pour prendre en compte à ces attaques résiduelles [31].

25. REsource LOcation And Discovery (RELOAD), IETF Internet Draft, www.p2psip.org/drafts/draft-bryan-p2psip-reload-04.html

D'autre part, nous traiterons dans ce chapitre de la téléphonie IP dans le cloud. Nous proposons l'application de notre approche de gestion des risques dans ce nouveau contexte. La solution a pour objectif de gérer le risque induit par les attaques de sécurité et de fournir un ensemble de contremesure comme de service dans la couche applicative dans le cloud [31]. Ainsi, les contremesures deviennent publiques et partageables entre les différents clients. Enfin, nous proposons une modèle de gestion de tolérance qui traite le cas de panne des contremesures.

Dans une première section, nous montrons comment traiter les attaques résiduelles du framework RELOAD en utilisant un mécanisme de confiance et un ensemble de contremesures. Dans une deuxième section, nous présentons une approche pour externaliser les contremesures de notre approche de gestion dans le cloud. Nous analysons enfin les différents résultats expérimentaux pour quantifier les avantages et les limites de ces solutions en termes de risque et de coût induit induit par l'application des contremesures.

7.2 Framework RELOAD et attaques résiduelles

Nous proposons dans cette section d'étudier l'activation dynamique de contremesure pour traiter les attaques résiduelles non gérées par le framework RELOAD [30]. Ces contremesures visent à contrôler le trafic entrant aux paires vulnérables et le trafic sortant des paires dont le niveau de confiance est faible. L'objectif consiste à fournir une solution complémentaire aux mécanismes de certification afin de détecter et traiter les attaques résiduelles. Notre approche définit une stratégie de prévention basée sur la confiance qui intègre un algorithme de confiance distribué et spécifie deux mécanismes de prévention.

Ces attaques résiduelles incluent, en particulier, le déni de service qui peut être observé au cours de la phase d'enregistrement quand un attaquant refuse d'enregistrer ou de fournir une entrée de la DHT du réseau P2PSIP. Elles incluent aussi le refus de service qui peut être détecté au cours de la phase de routage quand un attaquant élimine volontairement des messages P2PSIP ou fournit des messages de routage incorrects. Le système de certification permet de signer et de contrôler les opérations qui sont effectuées par des paires dans les réseaux P2PSIP. Cependant, ce système ne permet pas d'évaluer le risque induit par leur comportement et ne protège pas le réseau contre ces attaques.

7.2.1 Sécurité dans le framework RELOAD

Le framework RELOAD ne se limite pas à la ToIP mais est également ouvert à d'autres services. RELOAD définit un réseau de stockage pair-à-pair où les enregistrements sont stockés sous des adresses numériques qui occupent le même espace que les identifiants des paires. Les pairs sont responsables d'enregistrer les données associées à un ensemble d'adresses déterminées par les identifiants de paires. RELOAD prend également en charge les utilisateurs P2P et SIP : il fournit plusieurs niveaux de sécurité. En particulier, il renforce la sécurité des enregistrements d'adresses SIP-URI dans la table DHT et le routage des messages ainsi que leurs échanges entre les paires. La sécurité du réseau P2PSIP exige à la fois l'authentification des pairs et des ressources [106].

Afin d'enregistrer des données dans la DHT, le pair doit prouver la possession de la clé privée de l'un des certificats générés par le serveur d'inscription. Ensuite, il signe les enregistrements stockés avec sa clé privée. Cet ensemble de règles définit la procédure d'enregistrement d'adresses SIP-URI dans la DHT. Quand un client veut stocker une certaine valeur, il envoie une requête P2P *Store* qui contient à la fois la valeur (les deux adresses d'enregistrement et de contact SIP-URI) et la signature de la valeur aux super paires responsables de l'enregistrement. Ces

super paires sont définis par un algorithme de construction de ressources de nom qui détermine l'emplacement d'une valeur dans la table DHT et l'identité du paire porteur. Quand le super paire reçoit la requête P2P *Store*, il doit déterminer si le client, qui a envoyé la requête, est autorisé à modifier l'enregistrement. La vérification se fait par comparaison de son identité aux exigences du modèle de contrôle d'accès par la vérification de son certificat. S'il satisfait ces exigences, le paire est autorisé à manipuler les données correspondantes à son certificat dans la table DHT soit par modification soit par suppression. Ces mécanismes offerts par le framework RELOAD fournissent un niveau de sécurité élevé pour l'enregistrement des adresses SIP-URI dans la DHT, mais certaines attaques contre les services VoIP restent réalisables. Ainsi, un attaquant qui contrôle un ou plusieurs super paires peut mettre en œuvre plusieurs attaques comme le refus de service, une attaque d'eclipse et plusieurs d'autres attaques de déni de service. En fait, il est possible que des super paires refusent d'enregistrer ou rejettent toute demande de stockage d'une valeur ou d'une ressource comme la correspondance entre les adresses d'enregistrement et de contact SIP-URI. En outre, un paire de la DHT peut nier la connaissance d'un enregistrement qu'il a préalablement accepté. Dans une certaine mesure, ces attaques peuvent être mitigées par l'utilisation d'enregistrements réplicats pour stocker les informations ou les récupérer dans la table DHT. L'inconvénient de cette solution est que le pair SIP qui cherche à récupérer des entrées de la DHT, ne sait pas quand et avec quel pair il doit déployer cette solution.

La solution d'authentification basée sur des certificats numériques empêche un attaquant d'être capable de créer ou de modifier des données détenues par les autres pairs. En outre, même si un pair subversif peut refuser de retourner les ressources de données dont il est responsable, il ne peut pas répondre aux requêtes par des données falsifiées, car il ne peut pas fournir le certificat numérique correspondant. Ainsi, la solution proposée par RELOAD consiste à faire une recherche parallèle pour trouver des enregistrements redondants correspondants à la clé concernée. Cette technique réduit l'exposition de la table DHT face à cette attaque. Cette solution de prévention réduit l'exposition de la table DHT face à ces attaques. Cependant, elle induit un délai supplémentaire dans l'établissement d'une session d'appel car la durée d'une recherche d'adresse est plus longue à cause de la recherche parallèle. Le mécanisme de sécurité du RELOAD pour la procédure d'enregistrement garantit l'intégrité des données stockées, tandis que la sécurité du routage des messages et des requêtes SIP a pour objectif d'arrêter l'attaquant dans l'exécution de déni de services qui détourne la route des messages échangés dans le réseau P2P. De plus, sécuriser la procédure de routage empêche les attaquants d'analyser le trafic et d'avoir des informations sur les sessions établies.

Le framework RELOAD intègre deux lignes de défense pour les échanges entre paires : la première consiste à utiliser les protocoles de sécurité TLS ou DTLS pour chaque lien de communication entre pairs. Ceci fournit une protection contre les attaquants qui ne font pas partie du réseau P2P. La deuxième ligne de défense consiste à signer numériquement chaque message par la clé publique du paire destinataire, ce qui empêche les pairs malveillants de modifier les messages échangés, même si ces pairs d'attaque sont sur la route des messages. Les mécanismes de sécurité de routage dans RELOAD sont conçus pour mitiger plutôt que d'éliminer les attaques sur le routage. En effet, il est encore possible pour un attaquant de mettre en place une variété d'attaques de routage. Par exemple, si un attaquant est en mesure de prendre une position sur la route d'acheminement des messages P2PSIP entre les deux pairs A et B, il peut éclipser le pair B. Il peut aussi envoyer des fausses valeurs de métriques du réseau afin de rediriger le trafic ce qui fait partie des attaques de déni de services. Le schéma de sécurité basée sur des certificats numériques sécurise l'espace d'identité des pairs, mais si un pair est compromis ou si un attaquant obtient un certificat du serveur d'inscription, un certain nombre de pairs subversifs peuvent encore apparaître dans le réseau. Bien que ces pairs ne peuvent pas falsifier les réponses

aux requêtes d'enregistrement, ils peuvent répondre avec des messages d'erreur, effectuer une attaque de déni de services sur l'enregistrement des ressources. Ils peuvent aussi compromettre le routage et rediriger le trafic échangé vers d'autres pairs compromis.

Nous proposons de traiter les attaques résiduelles par une stratégie qui s'appuie sur un algorithme de confiance et deux mécanismes de sécurité. Nous considérons que l'occurrence des attaques de refus de service, de déni de services ou des attaques de routage peut être réduite en mesurant la confiance entre les pairs qui interagissent pour réaliser ces taches [117]. Les mécanismes de sécurité permettront de protéger les pairs vulnérables et contrer si cela est nécessaire les pairs dont le niveau de confiance est faible.

7.2.2 Intégration de l'algorithme *eigentrust*

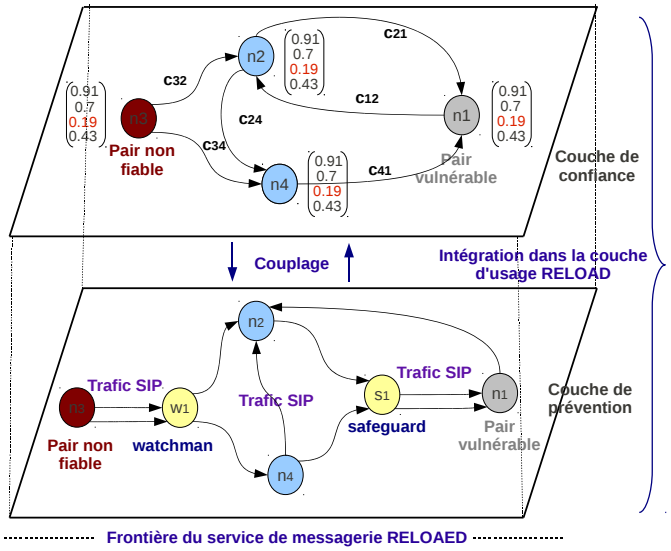


FIGURE 7.1 – Intégration d'*eigentrust* dans RELOAD

Comme le montre la figure 7.1, notre approche consiste à introduire dans le framework RELOAD un support mathématique pour quantifier le niveau de confiance des paires P2PSIP, et coupler cette modélisation avec des mécanismes de prévention. Notre modèle mathématique de confiance s'appuie sur l'algorithme distribué *eigentrust*, dont les avantages ont déjà été démontrés dans les applications pair-à-pair [54]. Il est défini dans la couche d'usage définie dans le framework RELOAD. Quand cet algorithme est exécuté, chaque paire du réseau P2PSIP attribue un score local de confiance aux différents paires P2PSIP avec lesquels il interagit dans le réseau. Les notes locales de confiance sont ensuite agrégées par l'algorithme *eigentrust* afin de converger par transitivité à un score global de confiance t_i pour chaque paire P2PSIP n_i, ce score est représenté par un vecteur de valeurs de confiance comme décrit sur la figure 7.1. Tous les paires dans le réseau P2PSIP participent dans la phase d'estimation des scores d'une manière répartie et non symétrique. Ces valeurs sont ensuite exploitées par le framework RELOAD afin de déterminer si les mécanismes de prévention doivent être exécutés quand des paires P2PSIP souhaitent communiquer entre eux.

Plusieurs stratégies de prévention peuvent être envisagées en se basant sur ce modèle qui interagit avec les composants de routage, de transmission et de stockage définis dans le framework RELOAD. L'approche la plus naturelle correspond à une stratégie d'évitement pour réduire

l'impact des paires non fiables dans le réseau P2PSIP. Dans ce cas, l'architecture permet de minimiser les interactions avec les paires P2PSIP marqués par une valeur de confiance faible. Par exemple, pendant le processus de routage, les messages P2PSIP peuvent être volontairement échangés uniquement entre les paires dont les valeurs de confiance sont élevées.

Le framework RELOAD nous permet de mettre en œuvre des stratégies plus élaborées, en particulier, nous considérons un système de prévention qui intègre des mécanismes de sécurité notés *watchman* et *safeguard*. Le composant de sécurité *safeguard*, représenté sur la couche de prévention de la figure 7.1, est sollicité par un paire P2PSIP s'il considère que son niveau de confiance locale est faible ou qu'il est vulnérable. L'objectif consiste à sécuriser un paire P2PSIP vulnérable par l'introduction d'un outil en amont qui met en œuvre des mécanismes de sécurité complémentaires. Cette technique est généralement utilisée par un paire qui est vulnérable aux messages mal formés ou à des attaques de déni de service. Dans ce cas, le mécanisme *safeguard* joue le rôle d'un paire intermédiaire qui implémente des règles de pare-feu applicatif (sur le trafic P2PSIP entrant) ou des contremesures. Le deuxième mécanisme est le *watchman*, également décrit sur la couche de prévention de la figure 7.1, qui est complémentaire au mécanisme de *safeguard* : il est fonctionnel d'une manière automatique quand un paire P2PSIP est détecté comme non fiable (c'est-à-dire, un niveau de confiance global relativement faible) par les autres paires. Dans ce cas, le *watchman* est responsable du contrôle du trafic P2PSIP sortant du paire non fiable. Ces deux composants peuvent être implémentés comme des serveurs proxy spécialisés dans le réseau P2PSIP, regroupés en *pools* de serveurs dans le framework RELOAD. Ils offrent leurs services à la demande des utilisateurs.

Modèle de confiance

Notre stratégie de prévention est soutenue par un modèle de confiance basé sur l'algorithme *eigentrust* pour évaluer la réputation des paires P2PSIP et leurs niveaux de confiance [54]. Afin d'intégrer cette base algorithmique dans le framework RELOAD, nous avons considéré trois critères principaux de confiance et de réputation. Ces critères sont directement liés aux opérations principales qu'un paire P2PSIP peut effectuer dans le réseau.

Le premier critère est trivial est porté sur la confiance envers le routage P2PSIP. Il permet de quantifier la capacité d'un paire P2PSIP à transmettre les messages correctement dans le réseau P2PSIP RELOAD sans le modifier ou le supprimer. Bien que le mécanisme de certification défini permet d'éviter qu'un paire P2PSIP compromis falsifie des message de routage, un attaquant est capable de générer des messages de routage malicieux ou de les supprimer quand ils passent par lui. L'évaluation de ce critère peut être effectuée par le composant de routage RELOAD. Cette attaque peut être réalisée par un des paires P2PSIP qui se trouvent le long de la route d'acheminement des messages du pair attaqué. Considérons un paire qui établit une session SIP avec un autre paire en utilisant le processus de routage itératif. Dans ce cas, le paire initiateur est en contact direct avec les paires intermédiaires qui sont impliqués dans le processus de routage et peut déterminer les paires P2PSIP qui éliminent les messages du routage ou envoient des messages incorrects. Ainsi, il est capable d'évaluer le niveau de confiance des autres pairs qui sont impliqués dans le processus du routage.

Conformément à l'algorithme *eigentrust*, nous définissons deux variables $sat(i,j)$ et $unsat(i,j)$. Ces variables représentent, pour un paire P2PSIP n_i, respectivement le nombre d'actions satisfaisantes et insatisfaisantes effectuées par le paire n_j par rapport l'activité de routage. Chaque fois que le paire n_i constate que le paire intermédiaire n_j supprime un message ou envoie un message inconsistant, il incrémente la variable $unsat^r(i,j)$, sinon il incrémente la variable $sat^r(i,j)$. le niveau de satisfaction pour l'activité de routage (r) est noté s^r_{ij} et est calculé en fonction de

ces deux variables. Elle est définie par l'équation suivante 7.1.

$$s_{ij}^{r} = sat^{r}(i,j) - unsat^{r}(i,j) \tag{7.1}$$

Le second critère de confiance, appelé la confiance envers la résolution d'adresses P2PSIP, a pour but de quantifier la capacité d'un paire P2PSIP à enregistrer et de fournir les entrées P2PSIP d'une façon sécurisée. En fait, ces entrées représentent la correspondance entre l'adresse SIP-URI d'enregistrement d'un paire P2PSIP et son adresse de contact SIP-URI (généralement donnée comme une adresse SIP-URI de contact contenant une adresse IP et un numéro de port). Cette résolution est nécessaire pour qu'un utilisateur établisse une session d'appel SIP avec un paire destination en utilisant son adresse SIP-URI d'enregistrement (par exemple sip : alice@example.com). Les mécanismes de certification de RELOAD permettent de vérifier que la correspondance entre les deux adresses est enregistrée par le paire approprié et n'a pas été modifiée par un paire intermédiaire. Cependant, ces mécanismes ne peuvent pas garantir que le paire contenant cette entrée qui est responsable de maintenir la table de hachage ne refusera pas de coopérer et de fournir cette entrée. Nous avons introduit dans notre modèle de confiance les deux variables $sat^{s}(i,j)$ et $unsat^{s}(i,j)$ afin de quantifier respectivement le nombre de demandes qui sont satisfaites ou insatisfaites par le paire P2PSIP qui est responsable de maintenir ces entrées. Comme défini précédemment avec le critère de confiance envers le routage, le paramètre de satisfaction d'enregistrement (s) s_{ij}^{s} est quantifié par les deux variables sat^{s} et $unsat_{ij}^{s}$.

$$s_{ij}^{s} = sat^{s}(i,j) - unsat^{s}(i,j) \tag{7.2}$$

Le dernier critère de confiance représente la confiance envers les appels P2PSIP et permet d'évaluer le comportement des paires P2PSIP en tant qu'appelants. L'objectif consiste à réduire l'impact et le nombre des communications indésirables qui pourraient être initiées par des attaquants. Après avoir résolu l'adresse de l'appelé et avoir l'adresse SIP-URI du contact, une session de communication est établie entre l'appelant et l'appelé. Alors que la certification permet d'authentifier l'appelant, il ne garantit pas que ce paire P2PSIP (ou un ensemble de paires P2PSIP) ne produit pas des sessions d'appel indésirables ou une attaque par inondation de messages SIP, comme les appels automatiques ou humains commerciaux. Nous intégrons dans notre modèle de confiance, un niveau de satisfaction, correspondant au critère de communication (c), noté s_{ij}^{c} est évalué en fonction des deux variables $unsat^{c}(i,j)$ et $sat^{c}(i,j)$ (voir équation 7.3).

$$s_{ij}^{c} = sat^{c}(i,j) - unsat^{c}(i,j) \tag{7.3}$$

Nous pouvons remarquer que les niveaux de satisfaction définis pour ces trois critères de confiance ne sont pas nécessairement corrélés, par exemple, un paire P2PSIP peut montrer un niveau élevé de satisfaction à l'égard de l'activité de routage, et un niveau de satisfaction faible à l'égard de ses appels. Ces niveaux de confiance $s_{ij}^{r}, s_{ij}^{s} et s_{ij}^{c}$ correspondent aux valeurs de confiance locales qui sont ensuite normalisées par l'algorithme *eigentrust* pour donner les variables de confiance normalisées : c_{ij}^{r}, c_{ij}^{s} et c_{ij}^{c}. Par exemple, c_{ij}^{r} est évaluée par le ratio entre $max(s_{ij}^{r}, 0)$ et $\sum_{j} max(s_{ij}^{r}, 0)$ comme décrit par l'équation 7.4.

$$c_{ij}^{r} = max(s_{ij}^{r}, 0) \quad avec \sum_{j=1}^{N} max(s_{ij}^{r}, 0) \tag{7.4}$$

De la même manière, nous quantifions c^s_{ij} et c^c_{ij} à partir des valeurs s^s_{ij} et s^c_{ij}. Dans le cas où le paire n_i ne peut pas évaluer le niveau de confiance d'un autre paire j parce que, par exemple, il n'a pas eu une relation directe ou il n'a pas sollicité le paire n_j pour un service du routage, d'enregistrement ou il n'a pas établi avec lui une session SIP. Dans ces cas, il est difficile que le paire n_i donne son avis en terme de confiance sur le paire n_j. *Eigentrust* propose une solution en se basant sur les valeurs de confiance locales établies par les connaissances du n_i. C'est une solution naturelle dans un environnement distribué. Ainsi, le pair n_i demande à ses voisins dont leurs valeurs de confiance sont élevées d'envoyer leurs valeurs locales respectives. La formule de transitivité décrite par l'équation 7.5 est appliquée par l'algorithme *eigentrust*. En particulier, cette équation montre comment la valeur de confiance locale t^r_{ij} du paire P2PSIP n_i est estimée à l'égard de l'activité de routage du paire n_j par la relation de transitivité. En fait, n_i exploite les valeurs de confiance locales calculées par les autres pairs pour quantifier d'une manière globale la réputation du pair n_j dans le réseau P2PSIP.

$$t^r_{ij} = \sum_{m=1}^{N} c^r_{im}.c^r_{mj} \; avec \; \sum_{m=1}^{N} t^r_{im} = 1 \tag{7.5}$$

Comme démontré dans [54], cette relation de transitivité représente la formule de base de l'algorithme *eigentrust* et est utilisée par le paire n_i pour quantifier la valeur locale de confiance t_{ij} (pour tout pair n_j du réseau P2PSIP). Dans notre cas, l'algorithme permet de quantifier les trois variables de confiance globales de chaque paire n_i, notées t^r_i, t^s_i et t^c_i, correspondant aux critères de confiance considérés (le routage, la résolution des adresses et l'établissement de sessions).

Afin de quantifier une valeur de confiance globale, on utilise les valeurs de confiance globale estimées par les différents pairs du réseau P2PSIP. Soit C la matrice des valeurs de confiance locales avec $C = \{c_{ij}\}_{\{i,j \in N\}}$, $\vec{t_i}$ le vecteur de confiance globale du pair n_i avec $\vec{t_i} = \{t_{ij}\}_{\{j \in N\}}$. t_{ij} la valeur de confiance globale du pair n_i envers le pair n_j. Le vecteur $\vec{t_i}$ est évalué par l'équation 7.6 avec $\vec{c_i}$ est le vecteur des valeurs de confiance locales.

$$\vec{t_i} = (C^T).\vec{c_i}, \tag{7.6}$$

Comme montré par l'équation 7.6, nous pouvons évaluer les trois vecteurs de confiance pour le routage, l'enregistrement d'adresses et l'établissement d'appels. Cependant, $\vec{t_i}$ reflète seulement l'expérience du pair n_i et de ses connaissances. Pour généraliser ces valeurs, nous pouvons utiliser les amis d'amis du pair n_i ce qui donne le vecteur de confiance $\vec{t_i}$ décrit par l'équation 7.7.

$$\vec{t} = (C^T)^2.\vec{c_i} \tag{7.7}$$

Si on continue les itérations, on obtient un vecteur de confiance généralisé et global pour tous les pairs du réseau P2PSIP. Cette estimation globale est quantifiée par l'équation 7.8 avec N est un nombre relativement grand fixé par la condition définie dans l'équation 7.9.

$$\vec{t} = (C^T)^N.\vec{c_i} \tag{7.8}$$

$$||\vec{t}^{(n+1)} - \vec{t}^{n}|| < \epsilon \tag{7.9}$$

avec ϵ un réel qui détermine la précision du calcul.

L'agrégation des valeurs de confiance locales et la quantification des valeurs de confiance globales peuvent être effectuées soit d'une manière centralisée ou d'une manière distribuée. Ces deux techniques peuvent être envisagées dans le contexte du framework RELOAD. Nous adoptons une version de calcul distribuée et sécurisée de l'algorithme *eigentrust* qui est cohérente avec la nature distribuée des réseaux P2PSIP et dont les propriétés de convergence sont rapides et ont déjà été analysées dans [54]. Nous considérons les vecteurs de confiance globales $\vec{t^r}, \vec{t^s}$ et $\vec{t^c}$ pour les trois critères d'évaluation. Ces trois vecteurs ne sont pas nécessairement corrélées (c'est-à-dire qu'il n'a y pas de dépendances entre eux). Nous les représentons par la matrice de confiance $\left(\vec{t^r}, \vec{t^s}, \vec{t^c}\right)$. Ces valeurs globales de confiance sont utilisées par notre stratégie de prévention qui est intégrée dans le framework RELOAD afin de sécuriser l'infrastructure P2PSIP. Suivant ces valeurs, la décision est prise d'utiliser un des deux mécanismes de sécurité : le *watchman* ou le *safeguard*.

7.2.3 Mécanismes de sécurité

En se basant sur le modèle de confiance, nous définissons deux mécanismes de prévention pour faire face aux attaques résiduelles de le framework RELOAD. Ces mécanismes représentent une couche de prévention couplée avec les mécanismes de confiance (voir figure 7.1). Ces mécanismes de sécurité sont intégrés dans la couche usage du framework RELOAD, et permettent de réduire la probabilité d'occurrence d'une attaque par les pairs P2PSIP qui sont identifiés comme non fiables. Ces mécanismes, décrits dans les figures 7.2 et 7.3, complètent la stratégie d'évitement (définie par RELOAD) qui consiste à éviter ou refuser les interactions avec les paires non fiables dans le réseau P2PSIP.

Mécanisme de sécurité *watchman*

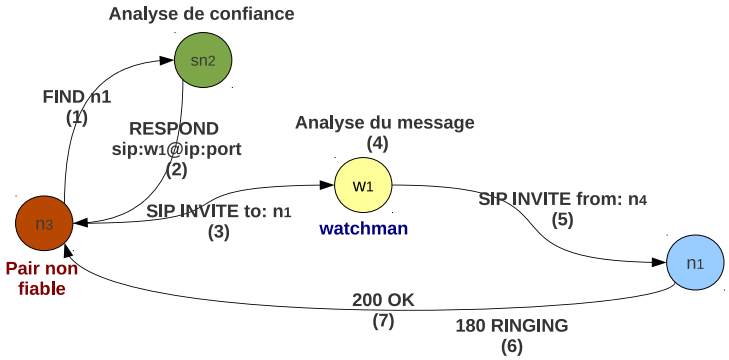


FIGURE 7.2 – Mécanisme de prévention *watchman*

Le premier mécanisme de prévention, appelé *watchman*, a pour objectif de contrôler les interactions d'un paire dans le réseau P2PSIP non sécurisé et contrôler les flux sortants : quand les mécanismes de confiance identifient un paire P2PSIP avec une valeur de confiance globale relativement faible ou critique (par rapport à un seuil de confiance ou au reste du réseau), le framework RELOAD ne bloque pas le trafic provenant de ce paire, mais active un mécanisme *watchman* chargé de surveiller le trafic sortant et de mitiger les messages P2PSIP transmis qui représentent une menace, si nécessaire. Par exemple, si la valeur de confiance t_i^c d'un paire P2PSIP n_i est relativement faible, le mécanisme *watchman* analyse la répartition des sessions

d'appels lancées par ce paire afin de détecter s'il prépare une attaque SPIT ou une attaque DoS. Par conséquent, il peut retarder ou même rejeter les messages malicieux afin de contrer l'attaque.

Le *watchman* est un composant logique de la couche de prévention, il peut être implémenté comme un serveur proxy dédié avec des fonctions de sécurité. Ces serveurs proxys peuvent être regroupés dans un *pool* de serveurs de prévention. Le réseau P2PSIP impose aux paires non fiables de passer par ce mécanisme de protection pour avoir l'autorisation d'interagir avec les autres paires. Cette règle reste active aussi longtemps que la valeur de confiance reste faible. L'algorithme *eigentrust* met à jour la valeur du confiance au cours du temps en se basant sur l'expérience des autres paires et sur le modèle de confiance défini. Le mécanisme du *watchman* participe également à cette mise à jour et peut d'une façon autonome dégrader ou augmenter cette valeur de confiance en utilisant les statistiques collectées et les résultats d'analyse.

La figure 7.2 illustre le fonctionnement du composant *watchman*. Le paire P2PSIP n_3 correspond à un paire identifié comme non fiable par l'algorithme *eigentrust* dont la valeur de confiance t_3^c est faible. Par conséquent, le réseau P2PSIP demande aux paires n_3 de communiquer et envoyer des messages via le serveur proxy qui contient le mécanisme *watchman* w_1. Quand le paire n_3 veut établir une session d'appel SIP avec le paire n_1, il contacte son super paire noté sn_2 (avec lequel il est logiquement associé) afin d'obtenir l'adresse du contact SIP-URI du paire n_3 en envoyant une requête P2P FIND (étape 1). Le super paire sn_2 connaît la valeur globale de confiance de n_3 qui provient de la couche de confiance. Comme cette valeur est faible, il redirige le paire n_3 au *watchman* w_1 (étapes 2 et 3). Ce dernier analyse les messages du paire n_3 (étape 4). Il détermine, ensuite, si le paire n_3 est autorisé ou non à poursuivre l'établissement de la session d'appel SIP. Cette autorisation dépend des résultats de l'analyse effectuée par le serveur *watchman* w_1. Il peut, dans ce cas, soit retarder ou refuser l'établissement de session si la répartition des appels lancés par le paire n_3 révèle une attaque potentielle (soit le taux d'appels est élevé, la durée moyenne d'appel est faible ou le taux de rejet d'appel est élevé). La nature des traitements exécutés par le mécanisme *watchman* peut varier en fonction de la criticité du paire (par exemple, si une nouvelle dégradation de la valeur de confiance est observée). Dans le scénario décrit dans la figure 7.2, l'autorisation est donnée au paire n_1 d'établir la session d'appel SIP avec le paire n_3 (étapes 5 à 7). Cette stratégie permet de réduire au minimum la probabilité d'occurrence d'attaques générées par le paire non fiable, tout en maintenant ce paire dans le réseau et en réduisant l'impact de ce mécanisme de sécurité sur son trafic SIP. Ceci permet de maintenir la disponibilité de service dans un réseau P2PSIP distribué et garantir en même temps un niveau de sécurité et de confiance élevé.

Mécanisme de sécurité *safeguard*

Nous proposons un second mécanisme de prévention basé sur le concept de contremesure appelé *safeguard*. L'objectif d'un *safeguard* est de protéger les paires vulnérables dans le framework RELOAD. Le mécanisme de *safeguard* est un composant logique et fait partie de la couche de prévention. Il est activé à la demande d'un paire vulnérable. Si le paire s'estime lui-même comme un paire vulnérable, il peut demander le déploiement d'un *safeguard*. Ce dernier a pour objectif d'analyser et de contrôler son trafic entrant afin de prévenir les attaques, il est implémenté comme un serveur proxy qui est capable de fournir une infrastructure de sécurité et de compléter le paire vulnérable avec des fonctionnalités de sécurité.

Un scénario typique d'utilisation du composant *safeguard* est le cas d'un paire P2PSIP sensible aux messages mal formés en raison d'implantation de plusieurs vulnérabilités. Si cette vulnérabilité ne peut pas être corrigée par la mise à niveau du paire, un *safeguard* peut être sollicité par le paire afin de renforcer sa sécurité. Le choix de solliciter ou non une mesure de

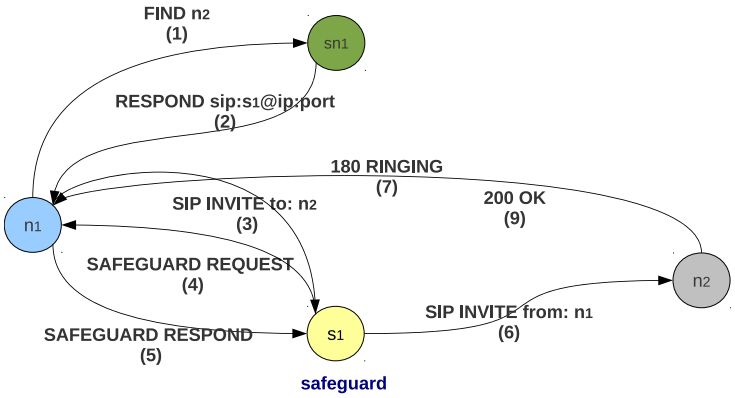


FIGURE 7.3 – Mécanisme de prévention *safeguard*

sécurité se fait d'une façon volontaire par le paire. La contremesure peut ensuite filtrer et empêcher les attaques par messages malformés en mettant en œuvre des règles pour la réception et l'autorisation des messages SIP comme les pare feux applicatifs. La nature de ce filtrage peut varier en fonction des valeurs de confiance des paires sources d'où proviennent les requêtes SIP.

Un autre scénario est le cas d'un paire P2PSIP ciblé par une attaque de déni de services distribuées. Le paire peut solliciter, dans ce contexte, un mécanisme de *safeguard* mettant en œuvre des dispositifs dédiés de sécurité tel que la technique du filtrage par test de *turing* audio [108] afin de minimiser son exposition face aux attaques. Les mécanismes du *safeguard* peuvent être également dans un ensemble de serveurs dans le framework RELOAD. D'un point de vue technique, le paire vulnérable peut modifier et mettre à jour son entrée d'enregistrement dans le réseau P2PSIP avec l'adresse du contact SIP-URI du serveur *safeguard* concerné, de sorte que son adresse publique est associée à cette contremesure. Ces opérations sont effectuées et validées par la signature par clé privée du paire concerné, en appliquant les règles de certification de RELOAD.

La figure 7.3 illustre les opérations du mécanisme de *safeguard* : le paire vulnérable n_2 est protégé par le safeguard s_1 afin de réduire son exposition face aux attaques. L'entrée dans le réseau P2PSIP qui correspond au paire n_2 associe l'adresse d'enregistrement SIP-URI à l'adresse du contact SIP-URI du serveur *safeguard* s_1 (l'adresse d'enregistrement SIP-URI sip :n2@example.com est associée à l'adresse physique sip :n2@ip_{s_1}). Considérons le paire P2PSIP n_1 qui tente d'établir une session d'appel SIP n_2. Pour cela, il sollicite son super paire sn_1 afin d'obtenir l'adresse du contact du paire n_2 (étape 1). Le super paire fournit l'adresse physique de $n2$, qui correspond en pratique à l'adresse de contact SIP-URI de serveur *safeguard* s_1 ; le paire n_1, ainsi, contacte s_1. Ce dernier analyse le trafic sortant du n_2, active les contremesures adéquates (les étapes 4 et 5), et décide si la session d'appel SIP sera finalement autorisée ou non (les étapes 6 à 8).

Les mécanisme de *safeguard* et de *watchman* sont complémentaires et ciblent les attaques résiduelles que les mécanismes de certification de RELOAD ne peuvent pas gérer seuls. En effet, en se basant sur le modèle de confiance et les mécanismes de sécurité, chaque pair peut déterminer les pairs non fiables avec lesquels le risque d'avoir une attaque de déni de service est élevé.

7.3 Externalisation des contremesures dans le cloud

Nous décrivons, dans cette section, l'externalisation de notre approche dans le contexte du cloud pour soutenir les services VoIP dans le cloud. L'objectif est d'adapter l'exposition des

services cloud VoIP en se basant sur l'activation dynamique des contremesures. Cette adaptation permet à nouveau de minimiser les coûts d'application des contremesures au cours du temps par rapport à la potentialité de la menace. Nous détaillons les concepts généraux de notre approche qui tire des avantages de l'infrastructure cloud pour permettre l'externalisation des contremesures en tant que service. Nous avons également spécifié l'architecture sous-jacente et le modèle mathématique à travers différents scénarios.

7.3.1 Contraintes et challenges de la ToIP dans le cloud

L'intégration des services VoIP dans le cloud contribue à augmenter le passage à l'échelle et améliorer la continuité opérationnelle de la téléphonie sur IP [23, 24]. Les services VoIP sont toutefois exposés à des menaces de sécurité multiples. Celles-ci peuvent être directement liées à la nature du cloud [116, 71, 110]. Par exemple, des serveurs VoIP ont été récemment attaqués par une attaque par force brute SIP effectuée via des serveurs de l'infrastructure du cloud Amazon EC2 [26]. Si les infrastructures du cloud peuvent être exploitées par des utilisateurs malveillants pour réaliser des attaques à grande échelle, nous considérons qu'elles fournissent également des nouvelles opportunités pour la protection de la téléphonie IP.

Des efforts importants ont été réalisés pour intégrer les services VoIP dans les infrastructures du cloud. Dans [23], les auteurs ont analysé l'intégration de l'architecture IMS dans cette infrastructure par l'intégration et l'implémentation de la téléphonie sur IP dans la couche SaaS (la couche des applications). La plateforme proposée améliore les performances des services multimédias en se basant sur un mécanisme d'allocation optimisée des ressources. Une autre approche dans [16] spécifie les principaux modules pour soutenir un cloud VoIP en définissant l'architecture sous-jacente et son couplage avec un réseau cellulaire.

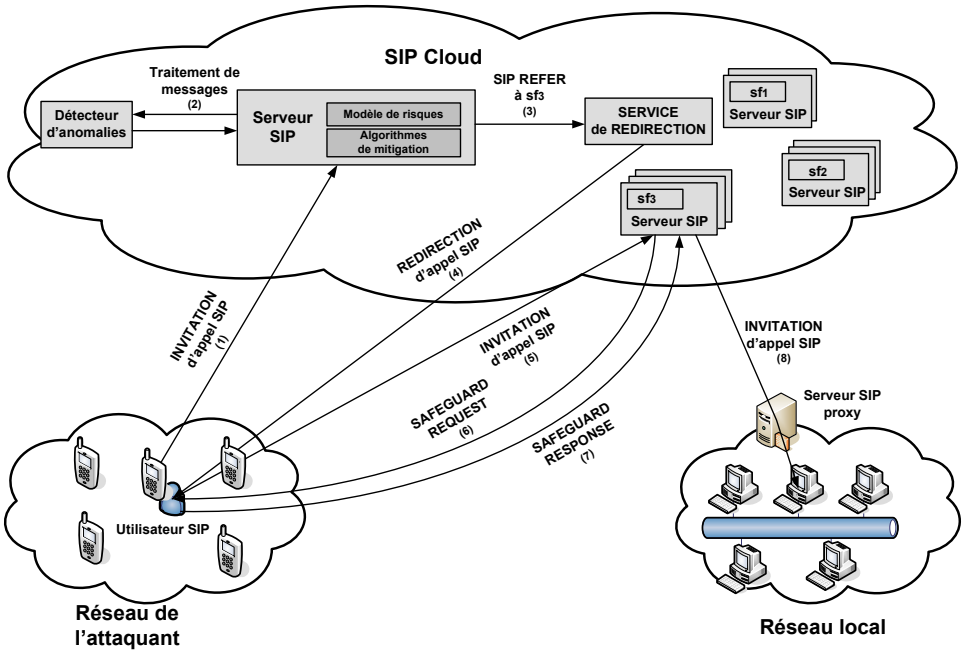


FIGURE 7.4 – Architecture de sécurité proposée pour le cloud VoIP

26. Amazon-based Attack, www.voipusersconference.org/2010/amazon-ec2-flood-attacks/

La nature de l'infrastructure du cloud pose des défis et des challenges majeurs en terme de sécurité. Les problèmes de sécurité du cloud sont généralement classés en quatre catégories principales [18]. Tout d'abord, nous considérons les questions liées au modèle multi-utilisateurs proposé par le *cloud computing* [37]. En fait, le même matériel et les mêmes ressources logicielles sont utilisés par de multiples acteurs pour exécuter des machines virtuelles, ce qui élargit la surface d'attaques. Dans ce contexte, les problèmes de sécurité sont partagés par les différents fournisseurs et utilisateurs de l'architecture du cloud, ce qui suppose d'établir des relations de confiance entre les différents acteurs. Le deuxième problème est lié à la perte de données ainsi que l'intégrité et la confidentialité. En effet, la migration des données d'une infrastructure traditionnelle vers l'infrastructure du cloud peut faciliter l'accès à des acteurs malveillants comme des utilisateurs non autorisés à cause d'un contrôle d'accès faible et non sécurisé, ce qui peut permettre aux attaquants d'altérer des données sensibles. Le troisième problème concerne l'interruption de service. Comme le cloud est une architecture ouverte et basée sur un modèle de service à la demande, elle est plus exposée à des attaques de déni de service. Les ressources de cloud peuvent également servir comme base du départ pour élaborer une attaque de déni de service [92]. Les attaquants peuvent aussi rediriger les utilisateurs du cloud vers des services illégitimes et exploiter les machines compromises comme une ressource pour de nouvelles attaques dans le cloud. Un quatrième problème est dû à la perte de contrôle et de gouvernance. Comme les ressources et les services sont confiés à des prestataires du cloud, il peut induire un manque de visibilité et de transparence sur les mécanismes de sécurité qui sont implantés par ces prestataires pour la protection des données privées, des programmes et des processus lancés par les utilisateurs. Une variété de mécanismes de protection sont disponibles pour répondre à ces questions de sécurité. Dans le contexte du cloud, l'importance des modèles de confiance et de réputation, des schémas de préservation de vie privée ainsi que la sécurité centralisée des données sont analysés dans [97]. Des directives et des recommandations ont également été élaborées, en particulier par le NIST, pour soutenir la sécurité et la confidentialité du cloud dans [110, 77]. La complexité de l'architecture cloud rend souvent ces mécanismes de sécurité plus difficile à déployer. En outre, les coûts d'application des contremesures peuvent être non négligeables pour un service critique comme la téléphonie sur IP. Dans ce contexte, nous mettons en œuvre notre solution de gestion de risques couplée avec des contremesures externalisées partageables dans le cloud. Nous proposons une stratégie de gestion pour le traitement des pannes dans ce contexte.

7.3.2 Architecture de la solution dans le cloud

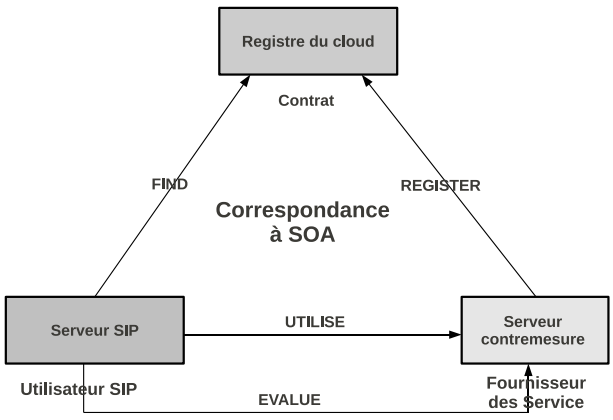
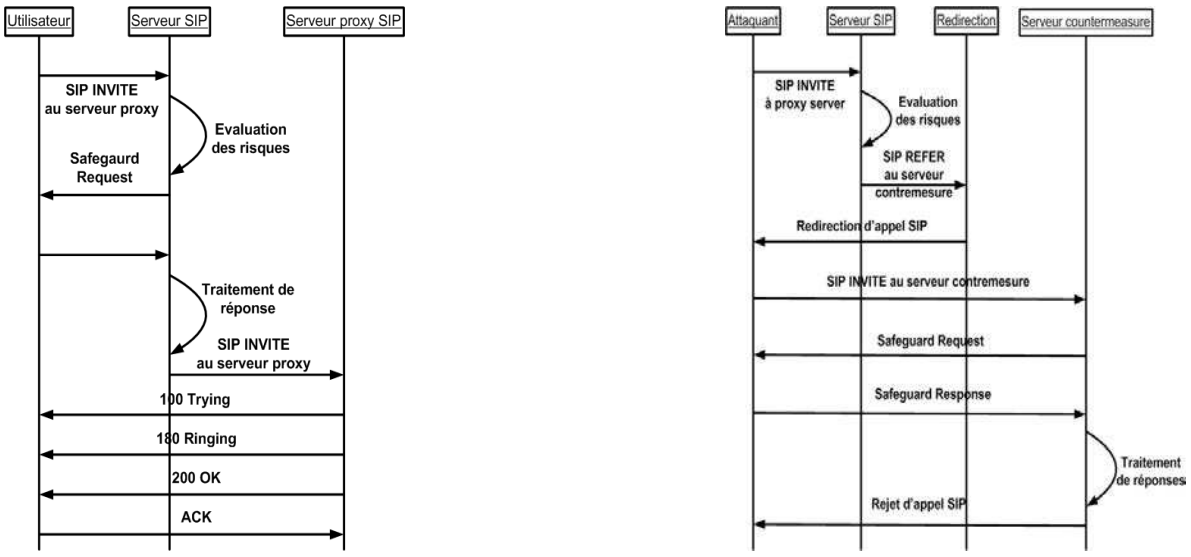


FIGURE 7.5 – Externalisation des contre-mesures dans le cloud

Notre approche de gestion de risques a pour objectif de contrôler l'exposition des services VoIP en activant des contre-mesures de sécurité en fonction de la potentialité des menaces mesurée. Ces contremesures VoIP peuvent être fournies comme des services dans le cloud. Elles sont alors déployées sur les serveurs SIP, appelés serveurs-contremesures, afin qu'elles puissent interagir avec les autres serveurs SIP en utilisant le protocole de signalisation. En particulier, des serveurs de contremesures peuvent être sollicités par le serveur SIP quand les requêtes d'établissement de sessions sont reçues. Lors de l'établissement d'une session d'appel, le serveur SIP peut ainsi réduire l'exposition face à une attaque ou contrer un appelant malveillant quand celui-ci est identifié comme un attaquant potentiel. Cette externalisation de contremesures présentent deux avantages majeurs : elle facilite l'intégration de nouvelles contremesures d'une manière flexible par le développement d'un groupe de contremesures et elle permet également d'augmenter la capacité du système de prévention en terme de passage à l'échelle. En plus, elle permet d'équilibrer la charge entre les serveurs SIP en déléguant le traitement des appels suspects à un ou plusieurs serveurs de contremesures dans le cloud.

La figure 7.5 illustre cette externalisation des contremesures VoIP dans le cloud en se basant sur un modèle SOA (*Service Oriented Architecture*). Les serveurs SIP sont considérés comme des utilisateurs de services, tandis que les serveurs de contremesures sont considérés comme des prestataires de services. Les serveurs SIP sont responsables de l'évaluation des risques. En se basant sur cette évaluation, ils déterminent les contremesures et les serveurs correspondants en sollicitant les registres ou les répertoires de service dans le cloud. Ensuite, ils utilisent ces contremesures fournies par les serveurs comme des services. Les serveurs SIP doivent interagir seulement avec les serveurs de contremesures fiables. Par ailleurs, les serveurs SIP et les serveurs de contremesures peuvent être sous la même autorité. Les mécanismes de confiance et de réputation peuvent également être envisagés pour éviter ou réduire l'impact d'un serveur de contremesures compromis. Les performances, telles que le délai de traitement et la sensibilité des contremesures peuvent être évaluées par les serveurs SIP par des mécanismes du retour d'expérience.



(a) Scénario avec un utilisateur

(b) Scénario avec un attaquant

FIGURE 7.6 – Scénarios de traitement d'attaques dans notre architecture

En se basant sur ces concepts, l'architecture de notre solution de gestion de risques est composée de plusieurs éléments décrits sur la figure 7.4. Nous retrouvons le composant principal de

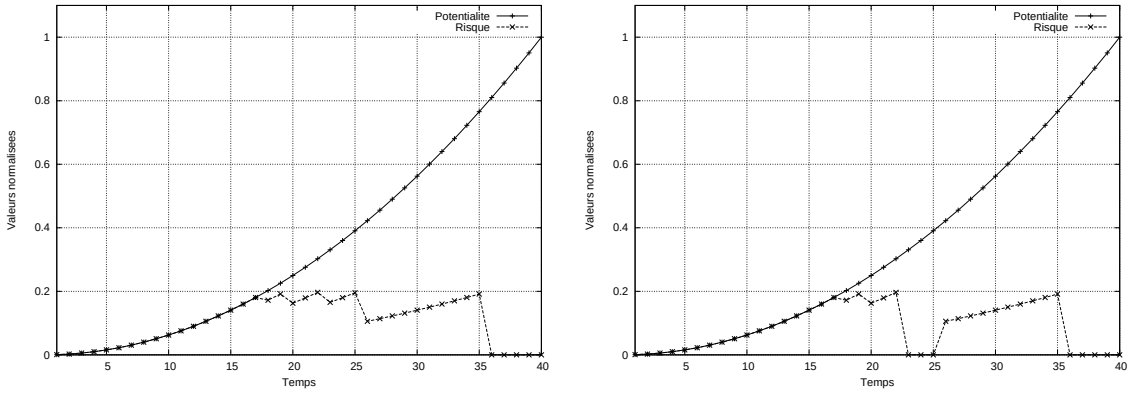
cette architecture est le serveur SIP qui gère les sessions d'appel et met en œuvre le processus de gestion des risques (dans le milieu de la figure). Il correspond au gestionnaire de risques dans notre architecture de risques. Ce serveur SIP (également appelé gestionnaire de risques) intègre le modèle quantitatif pour évaluer le niveau de risque pour les différents services, ainsi que des algorithmes de mitigation qui ont pour objectif le contrôle de risques par l'application des contremesures. Les deux autres principaux composants correspondent au système de détection de menace (le détecteur d'anomalies décrite sur la gauche de la figure) et l'ensemble des contremesures de sécurité déployées dans les serveurs dédiés (correspondant aux serveurs SIP dédiés tracés sur la droite). Nous retrouvons aussi le détecteur de menace qui applique soit des méthodes de détection d'anomalies. Nous avons déjà montré dans [68] les avantages des machines à vecteurs support (SVM) dans les réseaux SIP pour identifier des appels suspects. Un serveur de contremesures peut implémenter plusieurs contremesures. Un exemple typique d'une contremesure est la mise en œuvre des tests de *Turing* audio (qui correspondent aux CAPTCHA audio) pour contrer les attaques SPIT. Le gestionnaires des risques est responsable de la sélection du serveur de contremesures adéquat pour la réduction de risques et la mitigation d'attaque. Par ailleurs, les mêmes contremesures peuvent être implantées par différents serveurs de contremesures. En fait, cette réplication permet d'améliorer le passage à l'échelle, la robustesse des contremesures et d'équilibrer la charge induite par les appelants suspects.

Le fonctionnement de cette architecture de gestion des risques dans le cloud est illustré par les deux scénarios décrits par les diagrammes de la figure 7.6. Le premier scénario (voir figure 7.6(a)) correspond à l'établissement d'une session d'appel SIP initié par un utilisateur régulier. À la réception de l'invitation d'appel (le message SIP INVITE), le serveur SIP évalue le niveau de risque en se basant sur les résultats du détecteur de menace. Ce dernier peut s'appuyer sur les différentes sources de données déjà identifiées précédemment (voir chapitre 4).

Dans notre scénario, l'utilisateur SIP est identifié comme un candidat suspect (mais avec un risque relativement faible), il est contrôlé par une contremesure implantée directement par le gestionnaire de risques d'une manière locale. Comme l'utilisateur réussit à passer la contremesure en répondant correctement à la mesure de sécurité, le serveur SIP de gestion des risques laisse continuer l'établissement de la session d'appel entre les deux parties communicantes. Le serveur proxy SIP exploite un serveur de redirection pour transférer l'appel au serveur de contremesures approprié. Il envoie une requête SIP REFER au serveur de redirection pour indiquer l'adresse SIP-URI du contact de la mesure de sécurité. En modifiant le champs SIP Contact de la requête SIP INVITE, cette entité demande à l'appelant de contacter le serveur SIP implémentant la contremesure sélectionnée (la contremesure sf_3 sur la figure 7.4). Le serveur de contremesures interagit avec l'appelant suspect (par l'envoi d'une requête SAFEGUARD Request et la réception de SAFEGUARD Response notées dans le diagramme 7.6(b)). Si l'attaquant n'est pas capable de répondre correctement à la mesure de sécurité, son appel est rejeté par notre solution de gestion des risques. La sélection de(s) contremesures(s) dépend du niveau de risque évalué par le serveur en se basant sur le modèle de risque.

7.3.3 Stratégies d'application des contremesures

Notre approche dans le cloud est basée sur une modélisation mathématique et est définie par différentes stratégies d'application des contremesures. On rappelle que le niveau de risque est modélisé par la combinaison de la probabilité qu'une menace donnée exerce une vulnérabilité des services VoIP cloud et l'impact résultant de cet événement défavorable sur ce service [42]. Il est déjà défini par l'équation 7.10 où a représente une attaque et A est l'ensemble des attaques de sécurité potentielles.



(a) Scénario régulier (b) Scénario de panne

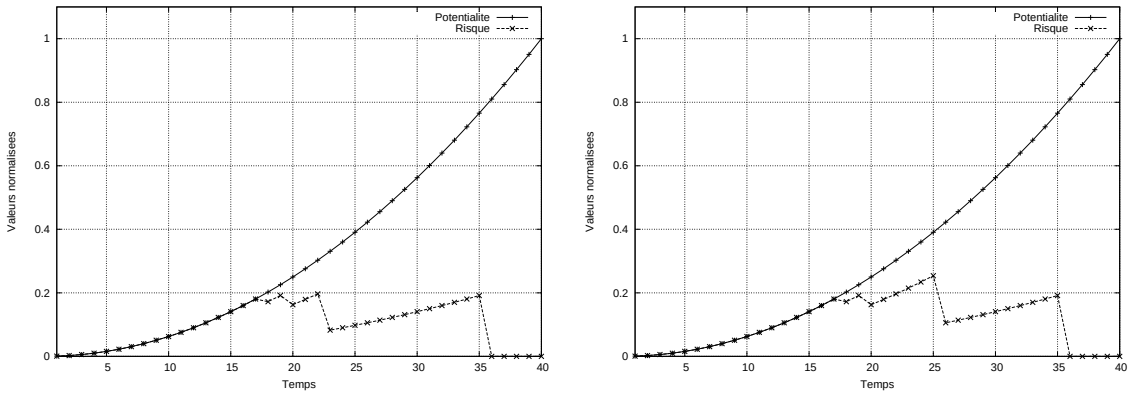
FIGURE 7.7 – Traitement du risque au cours du temps

$$\mathcal{R} = \sum_{a \in A} \mathcal{P}(a) \times \mathcal{E}(a) \times \mathcal{C}(a) \tag{7.10}$$

$$\mathcal{E}(a) = 1 - \sum_{sf_i \in SF} \sigma_i(a) active(sf_i) = 1 - \sum_{sf_i \in SFA} \sigma_i(a) \tag{7.11}$$

avec $\sigma_i(a)$ est l'impact de la contremesure sf_i sur l'exposition des services. Ce paramètre quantifie la capacité de sf_i pour contrer la source de l'attaque.

Nous rappelons aussi que notre approche repose sur deux algorithmes de mitigation de risque définis dans la section 4.2. Le gestionnaire de risques utilise ces algorithmes pour déterminer la contremesure appropriée par rapport à la potentialité de la menace mesurée. Les contremesures sont exposées en tant que services dans le cloud et peuvent subir des défaillances, une dégradation de performance ou une panne. Ce dysfonctionnement peut être du à des problèmes techniques, la surcharge d'appels mais également à des attaques de sécurité.



(a) La stratégie *upgrade* (b) La stratégie *downgrade*

FIGURE 7.8 – Les deux stratégies *upgrade* et *downgrade* dans le cas d'une panne

Comme illustré sur la figure 7.7, en comparant les résultats du traitement de risque dans les deux scénarios (le premier est un scénario régulier et le deuxième est un scénario de panne),

nous observons que la défaillance du service réduit considérablement le niveau de risque car il est équivalent à un rejet d'appel. Ainsi, le niveau de risque est nul tandis que les conséquences de panne sur les services VoIP est maximal car le service est indisponible pour l'appelantentre les instants 23 et 25 (voir la figure 7.7(b)). En effet, comme la contremesure est en panne, tout appel sera rejeté car l'appelant n'a pas répondu à son challenge.

L'externalisation utilisée par notre solution facilite la sélection de contremesures alternatives. Nous considérons trois stratégies principales d'application des contremesures : la stratégie *upgrade*, *downgrade* et la stratégie de réplication. Ces stratégies du traitement ont un impact sur l'exposition des services VoIP ainsi que sur les coûts induits.

La première stratégie de traitement, appelée la stratégie *upgrade*, consiste à appliquer comme une alternative à la contremesure tombée en panne, une autre contremesure qui est plus puissante en termes d'impact sur l'exposition. Nous ordonnons les contremesures suivant les valeurs de $\sigma_i(a)$. Comme le montre la figure 7.8(a), cette stratégie permet de maintenir le risque au dessous du seuil défini parce que la nouvelle contremesure appliquée est plus importante que celle qui devrait être appliquée. Soit l'ensemble des contremesures $SF = \{sf_1, sf_2, \ldots, sf_n\}$ définies par leurs valeurs d'impact $\{\sigma_i(a)\}$, avec $\sigma_1(a) \leq \sigma_2(a) \leq \ldots \leq \sigma_n(a)$. En cas d'échec de la contremesure sf_j, d'une défaillance ou d'un problème de disponibilité, la stratégie *upgrade* applique la contremesure suivante sf_{i+1} dont l'impact est plus important relativement ($\sigma_{j+1}(a) \geq \sigma_j(a)$). L'inconvénient de cette solution, dans ce contexte, est que cette stratégie réduit le niveau de risque plus que nécessaire car elle restreint l'accès aux services, réduit l'exposition et génère un coût supplémentaire en terme de disponibilité. Soit $failure(sf_i)$ la fonction qui indique l'état opérationnel de la contremesure sf_i. La stratégie *upgrade* réduit l'exposition des services à une nouvelle valeur $\mathcal{E}^\star(a)$ avec $SFA^\star = SFA + \{sf_{j+1}\} - \{sf_j\}$, telle que donnée par l'équation 7.12 si la contremesure sf_j tombe en panne ($failure(sf_j) = 1$).

$$\begin{aligned} \mathcal{E}^\star(a) &= 1 - \sum_{sf_i \in SFA^\star} \sigma_i(a) = 1 - \sum_{sf_i \in SFA + \{sf_{j+1}\} - \{sf_j\}} \sigma_i(a) \\ &= \mathcal{E}(a) - \sigma_{j+1}(a) + \sigma_j(a) \end{aligned} \tag{7.12}$$

La deuxième stratégie, appelée la stratégie de *downgrade*, consiste à appliquer comme une alternative à la contremesure tombée en panne ($failure(sf_j) = 1$), la contremesure qui précède celle en panne en termes d'impact sur l'exposition (suivant les valeurs de $\sigma_i(a)$). Ainsi, cette stratégie choisit d'appliquer la contremesure sf_{i-1} dont l'impact est plus faible relativement ($\sigma_{j-1}(a) \leq \sigma_j(a)$). La sous figure 7.8(b) illustre comment la stratégie de *downgrade* traite le risque. Nous pouvons clairement observer que cette approche est moins pessimiste par rapport à la stratégie précédente : elle minimise le coût total d'application des contremesures, mais génère, en même temps, un niveau de risque plus élevée (supérieure à la valeur seuil pendant la période de défaillance). Comme cela est défini par l'équation 7.13, cette stratégie augmente l'exposition des services VoIP dans le cloud à une nouvelle valeur $\mathcal{E}^\star(a)$ avec $SFA^\star = SFA + \{sf_{j-1}\} - \{sf_j\}$ si la contremesure sf_j est en panne.

$$\mathcal{E}^\star(a) = 1 - \sum_{sf_i \in SFA^\star} \sigma_i(a) = \mathcal{E}(a) - \sigma_{j-1}(a) + \sigma_j(a) \tag{7.13}$$

La dernière stratégie, appelée stratégie de réplication, consiste à implémenter dans le cloud des réplicats de contremesures. L'objectif consiste à activer une contremesure réplique de la nature même que celle tombée en panne ou désactivée dû à une défaillance. Considérons $SF_j =$

$\{sf_{j1}, \ldots, sf_{jk}\}$ les k contremesures réplicats associées à la contremesure sf_j. L'évolution du niveau de risque est similaire à celle observée avec la version régulière de gestion des risques, parce que la contremesure alternative a le même impact $\sigma_j(a)$ sur l'exposition $\mathcal{E}^\star(a) = \mathcal{E}(a)$. Cette stratégie peut générer des coûts supplémentaires dû à la mise en œuvre, la maintenance et le fonctionnement des contremesures réplicats. La défaillance ou la pannes des services sont encore possibles, mais avec une plus faible probabilité, quand toutes les réplicats de la contremesure sont en panne ou en défaillance ($\prod_{l=1}^{k} failure(sf_{jk}) = 1$). Dans ce cas, la stratégie *upgrade* ou la stratégie *downgrade* peuvent être exploitées d'une manière complémentaire au mécanisme de réplication.

D'un point de vue pratique, ces stratégies de traitement utilisent le service de redirection décrit sur la figure 7.4, afin de renvoyer les appels vers les serveurs d'application des contremesures appropriées, ce service de redirection fournit les primitives d'un serveur de redirection SIP et améliore le passage à l'échelle des services VoIP. Alors que le gestionnaire de risques détermine les opérations à effectuer en fonction du niveau de risque mesuré, ce service les redirige vers les serveurs contremesure.

Après avoir répondu au défi défini par la contremesure, l'appelant peut être rejeté, si le défi a échoué, ou son appel est transmis à la destination, si le défi est réussi. Le champ de message SIP *Contact* est utilisé pour indiquer l'adresse SIP-URI du contact du serveur de contremesures, ou celle de la destination. Nous voulons dire par destination l'appelé ou le serveur proxy utilisé pour joindre l'appelé [91]. Dans le cas d'un challenge réussi, deux versions différentes de notre solution peuvent être envisagées. La première version consiste à transmettre les requêtes de l'appelant directement par le serveur de contremesures à la destination. La deuxième version consiste à les transmettre au gestionnaire des risques. Le champ *Via* de messages SIP est utilisé par le serveur de contremesures afin de déterminer le chemin du retour. Un serveur de contremesures peut également rediriger les requêtes de l'appelant vers un autre serveur de contremesures. La première version permet de maintenir le contrôle au niveau du gestionnaire de risques, afin de transmettre les requêtes de l'appelant à d'autres serveurs contremesures, par exemple. La première permet de minimiser la charge au niveau du gestionnaire de risques en déléguant la transmission au serveur de contremesures.

7.4 Résultats expérimentaux

Nous avons évalué à la fois les performances de la stratégie appliquée pour le framework RELOAD, et l'externalisation de notre approche de gestion des risques dans le cloud par un ensemble d'expérience. Cette évaluation est faite par rapport au risque, la disponibilité et le coût d'application des contremesures.

7.4.1 Évaluation des mécanismes de confiance

Afin d'évaluer la performance de notre stratégie pour le framework RELOAD, nous avons mené un ensemble d'expériences avec le simulateur p2ptrust [27]. Pour élaborer ces expériences, nous avons considéré un réseau P2PSIP composé d'un maximum de 300 paires. Un sous-ensemble de ces paires P2PSIP interagissent comme des paires malveillants en générant un trafic malicieux. Quand une attaque est réalisée avec succès contre un paire donné du P2PSIP, le paire cible peut être neutralisé du réseau (compromission neutre), ou devenir un paire malveillant (compromission

27. P2P Trust Simulator, University of Pennsilvania, Penn Engineering School, rtg.cis.upenn.edu/qtm, 2009

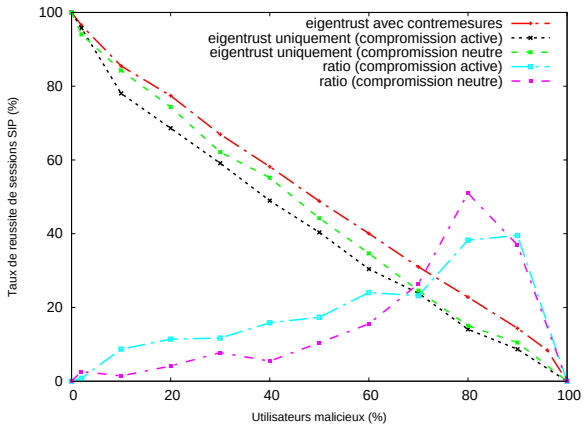


FIGURE 7.9 – Évaluation du taux de réussite dans le réseau P2PSIP

active). Un autre sous-ensemble de paires représente un ensemble de serveurs proxys qui peuvent être activés comme des watchmen dans le framework RELOAD.

Dans une première série d'expériences, nous nous sommes intéressés à évaluer les avantages et les limites de notre stratégie de prévention (application de l'algorithme *eigentrust* couplé avec des contremesures) en comparaison à une stratégie d'évitement (application de *eigentrust* uniquement). Nous avons quantifié le pourcentage des sessions SIP correctement établies dans le réseau P2PSIP en faisant varier le pourcentage de paires malveillants de 0% jusqu'à 100% dans le réseau.

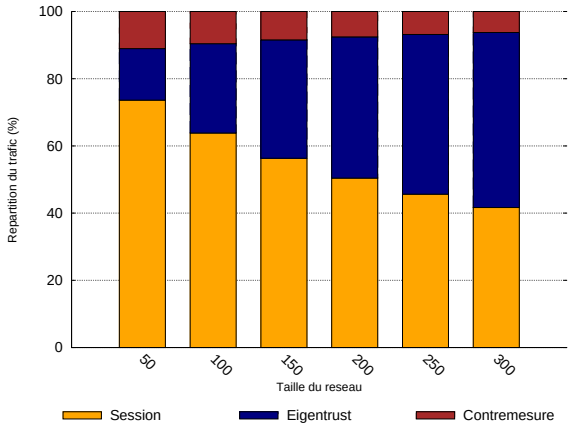


FIGURE 7.10 – Évaluation de charge induite par la solution

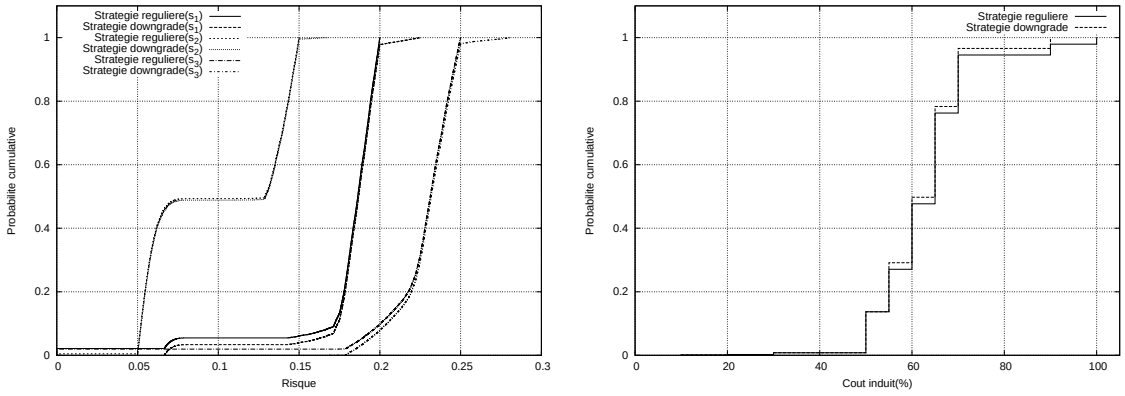
Dans cette série, le nombre de paires est fixé à 150 et les mécanismes de *watchmen* représentent 15% des paires. Les résultats expérimentaux sont décrits sur la figure 7.9, où nous avons tracé, respectivement le taux de réussite des appels avec notre stratégie de prévention (compromission active) et la stratégie d'évitement (compromissions active et neutre). Nous avons également tracé deux courbes supplémentaires correspondant au rapport entre ces deux stratégies. Au cours de ces expériences, nous avons observé un avantage de 8% par rapport à la stratégie d'évitement en cas de compromission active : comme on s'y attendait, cet avantage est moins important en cas de compromission neutre. En fait, la différence avec notre stratégie de prévention est de 5% en moyenne pour le cas de compromission active, et tombe à 3% en moyenne pour le cas de

compromission neutre. Il est évident qu'il y a aucun avantage à l'utilisation de notre stratégie si le pourcentage de paires malveillants est de 0% ou 100% (tous les paires de réseau P2PSIP sont malveillants), mais ces cas de scénarios sont extrêmes.

Dans une deuxième série d'expériences, nous nous sommes focalisés sur une autre question importante qui concerne l'évaluation de passage à l'échelle de cette solution. Nous avons quantifié la surcharge du trafic généré par les mécanismes de prévention tout en faisant varier la taille du réseau P2PSIP de 50 à 300 paires. Nous avons considéré que 20% des paires du réseau sont malveillants. La figure 7.10 représente la distribution du trafic utilisé pour la signalisation pour les différentes tailles de réseau. Il montre l'importance relative des messages réguliers dû à l'établissement de sessions P2PSIP, par rapport aux messages générés par l'algorithme *eigentrust* et les contremesures. Cette figure montre clairement que l'application de l'algorithme *eigentrust* et sa mise à jour représentent la partie la plus importante de la surcharge du trafic : par exemple, pour 50 paires, les messages du surcharge générés par l'application d'*eigentrust* et les mécanismes de prévention représentent 25% du trafic global de signalisation. Cette charge supplémentaire représente une limite de notre stratégie pour le framework RELOAD.

On peut décomposer ces 25% du trafic en 14% induit par l'application de l'algorithme *eigentrust* et 11% dû à l'application des mécanismes de prévention. Quand on varie la taille du réseau à 300 paires, ce pourcentage passe à 59% du trafic, avec 52% du trafic généré par *eigentrust* et 7% dû à l'application des contremesures. En analysant le nombre de messages (les valeurs absolues), nous obtenons les mêmes constatations : la courbe représentant les messages générés par les contremesures se caractérise par une allure linéaire, tandis que la courbe des messages générés par *eigentrust* ont une allure quadratique. Notre stratégie de prévention est, ainsi, fortement dépendante de la performance de l'algorithme *eigentrust*, et sa capacité de passage à l'échelle est directement liée à celle de l'algorithme *eigentrust*.

7.4.2 Impact de l'externalisation sur le traitement du risque



(a) Impact sur le risque

(b) Impact sur le coût

FIGURE 7.11 – Impact de la valeur de seuil sur la stratégie *downgrade*

Nous avons également évalué les performances de notre solution de gestion de risques dans le cloud. En particulier, notre évaluation a porté sur l'analyse des différentes stratégies de traitement (la stratégie *upgrade*, *downgrade* et la stratégie de réplication) décrites dans la section 7.3.3. Nous nous sommes intéressés à évaluer l'évolution du risque et du coût obtenu avec ces stratégies de traitement. Les expériences sont effectuées par simulation de Monte Carlo et les graphes

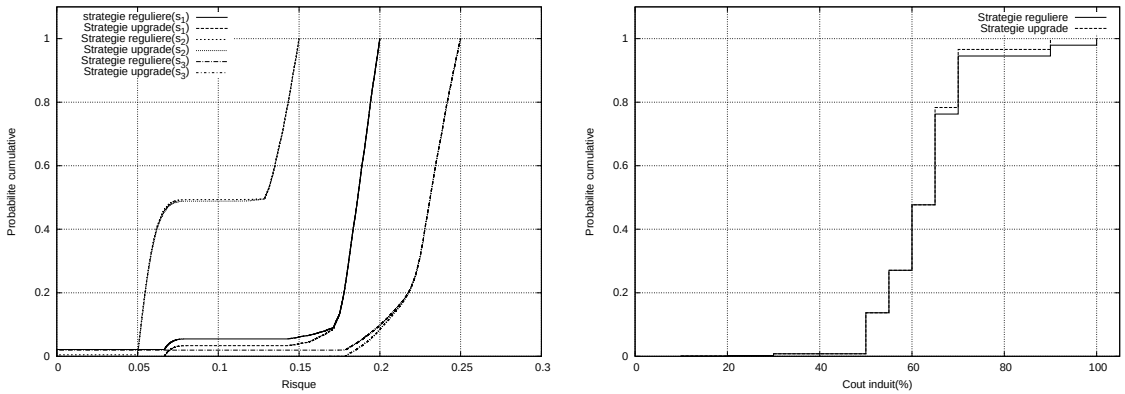
correspondent à des distribution de probabilités cumulées. Nous avons examiné différentes valeurs de seuil variant de 0,15 à 0,25, chaque contremesure est caractérisée par un coût d'application et un impact sur l'exposition. Le coût quantifie le délai supplémentaire induit par l'application de la contremesure sur le service VoIP, tandis que l'impact définit celui de la contremesure sur l'exposition de service (la valeur $\sigma_i(a)$ influence l'exposition $\mathcal{E}(a)$). Nous considérons dans ces expériences des attaques VoIP observables par détection d'anomalies comme les attaques SPIT [78]. Les contremesures sont exposées à des défaillances ou à des pannes qui peuvent être dues à des problèmes techniques ou des attaques de sécurité. Ainsi, ces contremesures ne sont plus disponibles quand elles sont en panne. La probabilité de défaillance suit une loi de Bernoulli avec un paramètre de 0,1, soit une contremesure a 10% de probabilité d'être affectée par une défaillance dans l'infrastructure cloud VoIP. Comme notre approche expérimentale est extensible, les expériences peuvent être étendues à d'autres modèles de défaillance. Nous avons réalisé des simulations de Monte Carlo permettant un taux d'erreur de moins de 5%.

Dans une première série d'expériences, nous avons évalué la performance de la stratégie de *downgrade* par rapport à l'approche de gestion de risques régulière. La figure 7.11 décrit la distribution de probabilité du risque et de coûts avec les différentes valeurs de seuil s_1, s_2 et s_3. On peut d'abord observer sur la première courbe 7.11(a) (correspondant à l'évolution du risque) que les trois courbes représentant la stratégie régulière pour les différentes valeurs de seuil sont au-dessus des courbes représentant la stratégie *downgrade*. Par conséquent, d'après cette série d'expérience, on a plus de probabilité d'avoir de risque avec la stratégie *downgrade* que avec la stratégie de risques régulières dans le cas de panne d'une contremesure. Cela est dû au fait que la stratégie *downgrade* consiste à remplacer les contremesures tombées en panne par une contremesure moins puissante en termes d'impact sur l'exposition en cas d'une défaillance tandis que, et dans la même condition, une stratégie classique bloque la communication et rejette l'appel. Ainsi, les services VoIP cloud seront plus exposées aux attaques de sécurité et le niveau de risque est globalement plus élevé que celui avec la solution de risque régulière.

La courbe 7.11(a) montre également un phénomène auquel nous ne nous attendions pas : la stratégie régulière de gestion des risques se caractérise par une probabilité d'avoir un risque égal à zéro non nulle (égal à 2,31%). En fait, quand la contremesure concernée tombe en panne, cette stratégie n'est pas capable de trouver et d'appliquer une solution alternative. L'appel VoIP est, donc, rejeté par le gestionnaire de risques, ce qui réduit le niveau de risque à zéro, mais peut aussi rendre le service de VoIP indisponible pour les utilisateurs légaux. Nous n'observons pas ce phénomène avec la stratégie de risques *downgrade* dans ces graphes : l'appel suspect est pris en charge, dans le cas d'échec, par une autre mesure de prévention alternative qui permet de maintenir la disponibilité du service VoIP et garantir sa continuité opérationnelle. En même temps, la stratégie *downgrade* génère une distribution de risque qui peut dépasser la valeur seuil. En particulier, on observe un risque qui dépasse la valeur de 0.225 supérieure au seuil de risque avec la courbe correspondante à s_2.

La figure 7.11(b) évalue l'amplitude des coûts induits avec les deux mêmes approches de risques. Le coût d'une contremesure de sécurité correspond à son impact sur les performances de service à savoir le délai d'établissement d'une session VoIP, l'utilisabilité et la disponibilité. Comme le graphe représente des distributions de probabilités cumulées, nous remarquons que les deux courbes qui correspondent aux deux approches de gestion des risques ont une forme en escalier. Cette forme correspond à l'ensemble discret des valeurs de coût $\{cost(sf_1), \ldots, cost(sf_n)\}$ associé à l'ensemble des contremesures $\{sf_1, \ldots, sf_n\}$. En plus, la courbe correspondante à la stratégie *downgrade* montre des valeurs de probabilité supérieures à celles correspondantes à la stratégie de risque régulière. Cela ne signifie pas que cette stratégie est plus coûteuse en termes de performance que l'autre stratégie régulière. En fait, l'approche régulière rejette l'appel si la

contremesure à appliquer est en panne (le service n'est pas disponible), tandis que la stratégie *downgrade* sélectionne une contremesure avec un coût inférieur à la contremesure concernée (le service est maintenu).



(a) Impact sur le risque

(b) Impact sur le coût

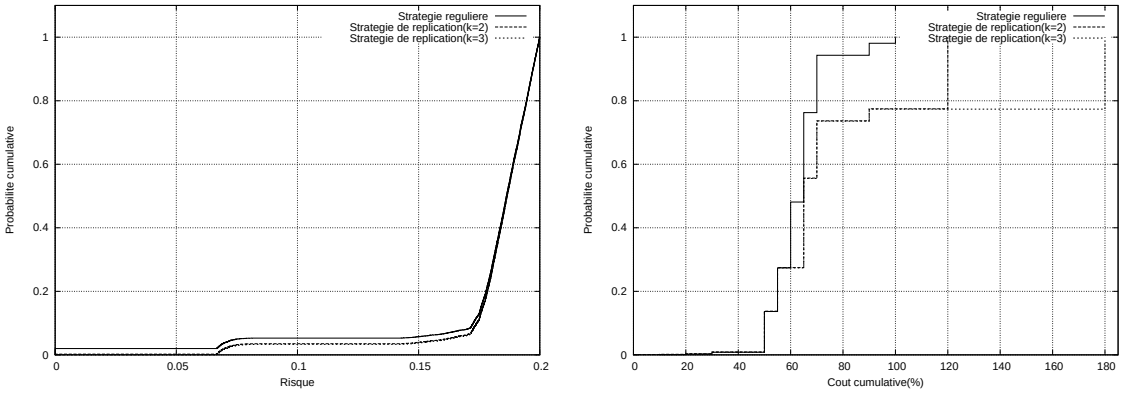
FIGURE 7.12 – Impact de la valeur de seuil sur la stratégie *upgrade*

Dans une deuxième série d'expériences, nous avons analysé la performance de la stratégie de gestion de risques *upgrade*. Les résultats expérimentaux sont présentés sur la figure 7.12 où nous avons tracé l'amplitude de risque et celle du coût d'application des contremesures. Les expériences ont été faites dans les mêmes conditions expérimentales (les mêmes valeurs seuils et le même modèle de défaillance). Sur la première figure 7.12(a), nous pouvons remarquer que la courbe correspondant à l'approche régulière est une fois de plus au dessus de celle de la stratégie de risques *upgrade*. Cela signifie que l'amplitude du risque est globalement faible avec l'approche régulière par rapport l'approche *upgrade*, comme le risque converge vers zéro lorsque la contremesure tombe en panne. En même temps, la différence en termes de risques entre les deux courbes est inférieure en comparaison à celle de la première série d'expérience.

La stratégie *upgrade* est capable de mieux protéger l'infrastructure du réseau en activant une contremesure dont l'impact est plus élevé que celui de la contremesure tombée en panne. La sous figure 7.12(b) illustre l'analyse des coûts : les coûts générés par la stratégie de risques *upgrade* sont relativement plus élevés que ceux correspondant à la stratégie *downgrade* et sont plus proches des valeurs de coût correspondant à la stratégie régulière. Ce coût supplémentaire permet de maintenir le niveau de risque à une valeur inférieure au seuil.

Dans la troisième série d'expériences, nous avons évalué la stratégie de réplication. Cette stratégie est basée sur la réplication des contremesures afin de remédier aux problèmes dus aux défaillances. Les résultats expérimentaux sont représentés par la figure 7.13 où nous avons tracé le risque et le coût d'application d'une contremesure pour cette stratégie avec un nombre de réplicats ($k = 2$) et ($k = 3$) avec k représentant le nombre d'instances de la contremesure. Si la première contremesure tombe en panne, l'appelant est pris en charge par la première duplicata de contremesure et si la première duplicata tombe en panne, la deuxième contremesure duplicata (le cas échéant) est sollicitée par le gestionnaire de risque.

Les différents résultats sont comparés aux résultats de la stratégie régulière. Nous pouvons observer sur la figure 7.13(a) que le niveau de risque induit par la stratégie régulière est au-dessus des deux versions de stratégie de réplication. Ce phénomène est similaire à celui observée dans les deux séries d'expériences précédentes : la distribution de probabilité montre que les valeurs de risque sont globalement plus faibles que celles de la stratégie de réplication, ce qui



(a) Impact sur le risque

(b) Impact sur le coût

FIGURE 7.13 – Impact de la réplication sur les performances de gestion de risques

est dû au fait que le risque est égal à zéro si des problèmes dus aux défaillances apparaissent. Toutefois, le service VoIP est indisponible dans ce cas avec la stratégie régulière. Cette figure montre également que la différence entre les deux stratégies de réplication ($k = 2$ et $k = 3$) est relativement faible au cours des expériences. Ces résultats sont expliqués par une probabilité de défaillance relativement faible du service VoIP quand cette stratégie est soutenue par une ou deux contremesure duplicatas. Nous avons également estimé l'amplitude du coût induit par la stratégie de réplication. La différence entre les deux versions de réplication est plus visible (voir figure 7.13(b)) car la transmission des appels à traiter aux contremesures répliques introduit des délais supplémentaires et leur instanciation nécessite plus de ressources à la disposition.

Dans une dernière série d'expériences, nous avons combiné la stratégie de réplication (avec une seule contremesure réplicat $k = 1$) et la stratégie *upgrade*. Ainsi, dans le cas de défaillance de la contremesure, le gestionnaire de risques sollicite une contremesure réplicats, si cette dernière tombe en panne, le gestionnaire de risque fait appel à la stratégie *upgrade* en sollicitant une contremesure dont l'impact est relativement plus important sur l'exposition. Cette stratégie hybride de gestion de risques fournit un compromis entre le coût d'application de contremesure et une bonne performance de service. En fait, l'instanciation de nouvelles contremesure duplicatas est plus coûteuse que l'utilisation d'une contremesure existante dont l'impact sur l'exposition est plus important. D'autres combinaisons peuvent être envisagées telles que la combinaison de la réplication et la stratégie *downgrade*. Ces stratégies de traitement de risques sont complémentaires des mécanismes de retour d'expériences dans le cas de sélection des fournisseurs de contremesures.

7.5 Synthèse

Nous avons traité dans ce chapitre des architectures hybrides de type RELOAD et l'externalisation de notre approche de gestion des risques dans le cloud. Nous avons notamment complété le framework RELOAD pour prévenir les attaques résiduelles. Cette solution est basée sur l'intégration d'un algorithme de confiance couplé à des mécanismes de sécurité. Ces mécanismes permettent de minimiser les attaques provenant des pairs non fiables et de protéger ceux qui sont vulnérables. Nous avons également évalué l'approche proposée par un ensemble de simulations. Nous nous sommes aussi intéressés aux services VoIP dans le cloud. Nous avons exploité de la flexibilité de l'infrastructure cloud afin d'utiliser les contremesures comme des services. Cette approche facilite l'intégration de nouvelles contremesures et permet de déléguer le traitement des

appels suspects à un ou plusieurs serveurs. Nous avons proposé dans ce cadre plusieurs stratégies pour prendre en charge la défaillance de contremesures et les avons évaluées. En conclusion, notre solution permet de soutenir le framework RELOAD pour contrer les attaques résiduelles. Cependant, son passage à l'échelle est directement dépendant du passage à l'échelle de l'algorithme eigentrust utilisé pour calculer le niveau de confiance. Concernant les services VoIP dans le cloud, nous avons montré les bénéfices et les limites de l'externalisation des contremesures dans le cadre du cloud.

Troisième partie

Mise en œuvre

Chapitre 8

Prototypage dans un serveur VoIP

Sommaire

8.1 Introduction

L'architecture qui prend en charge notre solution de gestion de risques permet d'assurer le couplage entre la détection des risques et l'application des traitements correspondants. L'objectif consiste à prendre en compte à un stade avancé les résultats du système de détection en vue de fournir un traitement progressif et continu des risques. L'architecture de notre approche est composée de quatre composants principaux à savoir le système de détection, le gestionnaire de risques, les contremesures de sécurité et le système de configuration.

Nous avons développé un prototype de notre solution au sein d'un serveur proxy Asterisk typiquement utilisé dans un réseau VoIP d'entreprise. Dans ce prototype, le gestionnaire de risques est directement intégré au serveur Asterisk en utilisant son interface de programmation [32]. Nous considérons que ce type de serveur joue un rôle central pour observer et contrôler un réseau VoIP centralisé. Le gestionnaire de risques peut être exploité à la fois pour des attaques de sécurité observables par signatures ou par anomalies. Nous nous sommes cependant focalisés sur les attaques détectables par anomalies en développant un module de détection basé sur une technique d'analyse par machines à vecteurs supports (comme décrit dans le chapitre 4).

Nous décrivons dans ce chapitre notre architecture fonctionnelle pour les réseaux VoIP d'entreprise ainsi que les composants considérés. Nous faisons une description technique de l'implantation de chaque module. Nous décrivons ensuite les interactions entre ces composants, et enfin

présentons un ensemble de tests complémentaires aux résultats expérimentaux décrits dans les précédents chapitres.

8.2 Serveur VoIP Asterisk

Nous allons tout d'abord présenter le serveur VoIP Asterisk au sein duquel notre stratégie des risques a été intégrée.

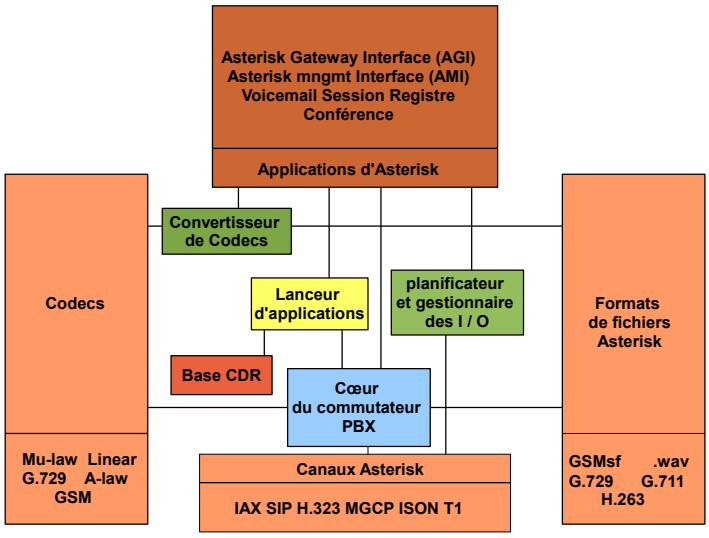


FIGURE 8.1 – Architecture du serveur VoIP Asterisk

Le serveur VoIP Asterisk est un logiciel qui met en œuvre un autocommutateur privé de téléphones, typiquement utilisé pour la téléphonie sur IP (IPBX). Comme tout logiciel IPBX, il permet aux téléphones attachés de faire des appels, de se connecter à d'autres services téléphoniques, y compris le réseau téléphonique public commuté et d'avoir accès aux services de la téléphonie sur IP. Asterisk est publié sous une licence double modèle, en utilisant la GNU comme licence de logiciel libre ainsi qu'une licence de logiciel propriétaire.

Les avantages du serveur Asterisk sont la richesse en termes de fonctionnalités par rapport aux autres serveurs, la non nécessité d'installer une passerelle pour se connecter aux réseaux RTC. De plus, il propose des solutions pour contourner le problème du NAT et est configurable pour répartir la charge des appels en se basant sur une politique de priorités. Nous avons choisi Asterisk pour ces avantages et surtout pour les interfaces qu'il offre et qui permettent de développer des applications et étendre ses modules. Les interfaces s'appellent AGI (*Asterisk Gateway Interface*). Asterisk offre de nombreuses fonctionnalités disponibles dans les systèmes PBX propriétaires (voir figure 8.1), comme par exemple la messagerie vocale, la conférence téléphonique, la réponse vocale interactive et la distribution automatique des appels. Il comprend un cœur de commutateur PBX qui assure le routage des appels, un convertisseur de codecs qui assure la compatibilité entre les terminaux ainsi qu'une base CDR qui contient des informations sur les appels.

Les utilisateurs peuvent créer de nouvelles fonctionnalités en écrivant des scripts dans le plan de numérotation via plusieurs extensions de langages ou par l'implémentation de programmes AGI. Il supporte une large variété de protocoles vidéo et de voix sur IP, y compris le protocole SIP, le protocole MGCP et le protocole H.323. Asterisk peut interopérer avec la plupart des téléphones SIP agissant à la fois comme un serveur d'enregistrement et une passerelle entre

les téléphones IP et le réseau de commutation de circuits RTC. Les applications dans Asterisk sont des modules chargeables qui effectuent des opérations spécifiques telles que composer un numéro (*appDial*), assurer des services de conférence (*appMeetme*), ou manipuler des opérations de messagerie vocale (*appVoicemail*).

```
[200]
type = friend
context = loriaSIP  ; contexte auquel appartient le numéro
secret = loria  ; définir le mot de passe
quality=yes  ; activer la surveillance de la qualité de service
callerid = "Bob" <200>  ; définir l'identité de l'utilisateur
username = 200  ; login
canreinvite = yes  ; rediriger par défaut
dtmfmode = rfc2833
disallow = all  ; autoriser un usage d'un codec spécifique
mailbox = 200@default  ; boîte vocale du contexte par défaut
```

FIGURE 8.2 – Exemple de fichier de configuration *sip.conf* du serveur Asterisk

En assurant à la fois des services de téléphonie traditionnelle et des services VoIP, Asterisk permet aux utilisateurs de construire des nouveaux systèmes téléphoniques, ou de migrer progressivement les systèmes de téléphonie existants avec les nouvelles technologies. Certains sites utilisent des serveurs Asterisk pour remplacer les PBXs qui sont sous licence propriétaire, d'autres pour fournir des fonctionnalités supplémentaires (comme la messagerie vocale ou les menus de réponse vocale) ou réduire les coûts en réalisant des appels interurbains via Internet.

Le serveur Asterisk est configuré par un ensemble de fichiers de configuration. L'un d'eux est le fichier *extensions.conf* qui contient le schéma de flux communiqués via Asterisk (voir figure 8.2). Un langage de script natif est utilisé pour définir les éléments de contrôle des processus, à savoir les variables, les macros de procédure, les contextes, les extensions et les actions. Les codes de numérotation, appelés "extensions" sont les points de départ pour les scripts qui contrôlent Asterisk et déterminent la façon de traiter les appels effectués à ces numéros dans ce contexte. Les contextes définissent les sources de l'appel et les extensions définissent leurs destinations. Notre implantation est basée sur le suivi du profil d'appels des utilisateurs. Dans ce contexte, chaque compte utilisateur est considéré comme une source d'appels. Cette approche ne peut pas être mise à l'échelle pour l'attaque SPIT provenant de l'intérieur parce que nous ne pouvons pas construire des profils pour tous les appelants possibles externes. Par ailleurs, les attaquants peuvent facilement changer leurs identités. Pour cela, nous suggérons le suivi des adresses IP (ou noms de domaine) en tant que sources d'appels. Lorsque le système de détection d'anomalies (formé par le moniteur d'appels et le détecteur d'anomalies) révèle une situation anormale, les sources d'appels suspectes sont transmises au gestionnaire de risques, ainsi que leurs potentialités respectives. Le gestionnaire de risques décide alors d'appliquer des mesures de sécurité à chaque source d'appel en fonction de sa potentialité.

8.3 Composants du prototype

Nous décrivons maintenant l'architecture de déploiement de notre solution de gestion de risques au sein de ce serveur (voir figure 8.3). Nous utilisons les attaques SPIT comme étude de cas pour évaluer les performances de notre prototype. Les attaques SPIT peuvent être divisées

en deux catégories : les attaques SPIT provenant de l'intérieur et l'attaque SPIT provenant de l'extérieur. Les attaques SPIT provenant de l'intérieur sont générées par les utilisateurs enregistrés dans le service VoIP alors que le SPIT provenant de l'extérieur consistent en l'envoi d'un ensemble d'appels non sollicités ciblant notre domaine.

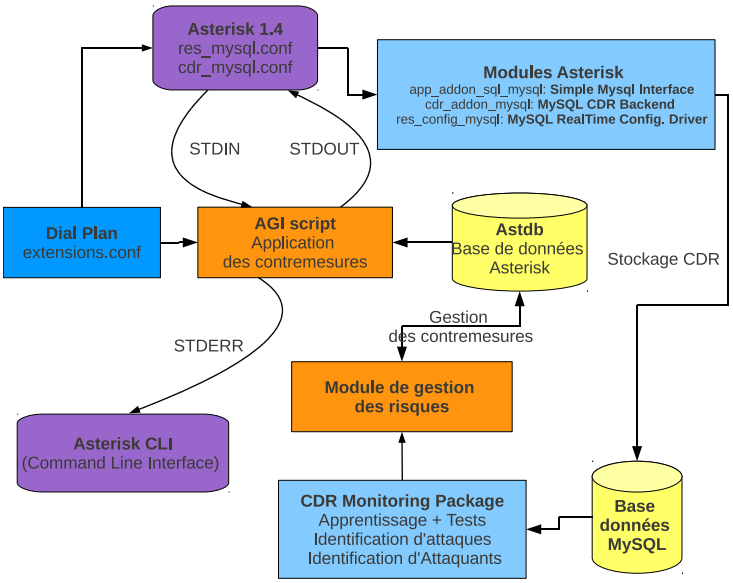


FIGURE 8.3 – Implantation de notre solution de risques

8.3.1 Système de détection

Notre système de détection d'intrusions est composé de deux éléments :

- Le moniteur d'appel : il a pour rôle d'extraire toutes les informations utiles à la détection d'intrusions à partir de plusieurs sources de données. Plusieurs sources de données sont disponibles pour effectuer le monitorage des appels dans une architecture VoIP. Ces sources comprennent le trafic réseau SIP, RTP et DNS, l'historique des serveurs SIP et notamment les enregistrements détaillés des appels (CDR) (voir figure 8.5) ainsi que les fichiers de configurations (voir figure 8.2 et 8.4).
- le détecteur d'anomalies : ce composant est chargé de mesurer la potentialité d'une attaque. Son rôle consiste à surveiller les éléments critiques du réseau et d'identifier les situations d'attaques. Nous considérons dans notre prototype un système de détection d'intrusions du réseau afin de suivre et contrôler le comportement des éléments VoIP à l'échelle du réseau. Des bonnes performances en termes de sensibilité et de spécificité du système de détection sont nécessaires. Typiquement, une valeur seuil permet de déterminer quand une attaque est considérée comme effective. Les résultats du processus de détection sont transmis au gestionnaire de risques à un stade précoce, même si ces résultats sont partiels et la potentialité d'attaque n'est pas très élevée. Ils sont progressivement intégrés dans le modèle de risques afin d'activer les mesures de sécurité d'une manière continue et adaptative.

Nous avons utilisé les pilotes Asterisk ainsi que ses modules pour nous connecter à la base de données MySQL et stocker les enregistrements détaillés des appels (CDR). La base de données est accessible par le module du moniteur d'appels afin d'interroger une liste des CDR extraite sur un intervalle de temps et éventuellement d'autres sources de données. Le module de surveillance des CDRs (qui fait partie du moniteur d'appels) possède plusieurs modes de fonctionnement :

```
[loria]
;exten => nom, priorité,application(), paramètres

exten => 200,1,Wait,1 ; attendre une seconde puis passer à l'action suivante
exten => 200,2,Dial(SIP/2001,8,tT) ; appeler le numéro 2001 via le protocole SIP
; " tT " permet à l'appelé de transférer l'appel

exten => 200,3,VoiceMail(200) ; passer l'appel à la messagerie vocale
exten => 200,4,Hangup() ; raccrocher
```

FIGURE 8.4 – Exemple de fichier de configuration *extensions.conf* du serveur Asterisk

- En ligne / hors ligne : dans le mode en ligne, une fenêtre de surveillance doit être définie (par exemple 5 minutes). A la fin de chaque fenêtre de surveillance, nous interrogeons tous les appels qui sont achevés dans cette période. En mode hors ligne, nous interrogeons tous les appels qui ont eu lieu entre le début et la fin de la période requise.
- Apprentissage / test : Dans le mode apprentissage, nous extrayons les statistiques et construisons un profil normal. En mode test, nous extrayons les statistiques et prédirons si elles révèlent un comportement anormal ou non
- Individuel / groupe / global : nous suivons une source unique (par exemple une adresse IP, un compte d'utilisateur) ou un groupe de sources.

L'algorithme de détection d'anomalies est basé sur une classe SVM implantée dans la bibliothèque libsvm [22], ce système génére comme sortie une valeur (la valeur de la fonction de décision) qui est considérée comme le score d'anomalie. L'algorithme de détection identifie la présence de SPIT ou d'autres anomalies et révèle la liste des sources d'attaque potentielles. L'identité de l'acteur est représentée par le compte utilisateur pour un utilisateur enregistré et par l'adresse IP pour les appelants externes. Le module de surveillance transmet les résultats au module de gestion de risques.

8.3.2 Gestionnaire de risques

Ce composant joue un rôle très important dans notre solution de gestion des risques car il intègre le modèle de risque (présenté dans le chapitre 3) et les algorithmes de mitigation de risques (l'algorithme de restriction et de relâchement de risques). Le gestionnaire de risques contrôle les menaces par l'estimation du niveau de risque et l'évaluation de l'état actuel de chaque élément du réseau VoIP. Le calcul de risques (phase d'estimation) est basé sur l'exposition de l'équipement VoIP, la potentialité d'une attaque et ses conséquences sur l'infrastructure VoIP. En se basant sur le niveau de risque, le gestionnaire des risques détermine (phase d'évaluation) si le niveau de risque est acceptable ou non, puis il sélectionne si nécessaire les traitements à appliquer sur l'infrastructure VoIP (phase de traitement). Cette tâche consiste typiquement à activer ou désactiver les mesures de sécurité afin d'optimiser le compromis entre la sécurité et la qualité du service VoIP.

Concernant son implantation, le module de gestion de risques gère la liste des sources d'appels suspectes dans la base Asterisk (AstDB). Cette liste est composée soit par des adresses d'enregistrement SIP-URI ou des adresses IP. La base de données Asterisk est une implémentation simple basée sur la version 1 de la base de données Berkeley. Nous choisissons d'utiliser AstDB car elle est simple et efficace pour travailler dans un environnement temps réel et d'une

```
Event : Cdr
Privilege : cdr,all
AccountCode :
Source :
Destination : 200
DestinationContext : loria
CallerID :
Channel : Console/dsp
DestinationChannel :
LastApplication : Hangup
LastData :
StartTime : 2011-10-10 11 :37 :09
AnswerTime : 2011-10-10 11 :37 :09
EndTime : 2011-10-10 12 :01 :25
Duration : 24 :16
BillableSeconds : 0
Disposition : ANSWERED
AMAFlags : DOCUMENTATION
UniqueID : 1282570041.3
UserField :
```

FIGURE 8.5 – Exemple d'un CDR du serveur Asterisk

forte interactivité. Le module de gestion des risques attribue une contremesure à chaque source d'attaque en se basant sur son niveau de risque et son ancien état.

Les extensions dans le plan de numérotation sont protégées par un programme script AGI (*Asterisk Gateway Interface*). Nous avons codé notre script AGI en Python en utilisant les commandes dans l'infrastructure AGI python. Quand un équipement VoIP est appelé, le script AGI est exécuté en premier : il prend les paramètres du canal de communications comme arguments et il sollicite la base de données Astdb pour voir si une mesure de sécurité est déjà appliquée contre cette source d'attaques.

Nous identifions l'appelant par le paramètre *agi* du canal créé par Asterisk. Comme définis précédemment, ce paramètre correspond au nom du compte de l'appelant si l'appelant est un utilisateur (enregistré) d'Asterisk ou l'adresse IP si l'appel provient de l'extérieur. Le script AGI échange les informations avec Asterisk à travers les canaux systèmes *stdin*, *stdout* et *stderr*. Notre script AGI Python implémente les différentes contremesures.

8.3.3 Mesures de sécurité

En fonction du niveau de risques, nous appliquons un ensemble de contre-mesures pour contrôler le risque dans la plateforme VoIP. Nous considérons un système de prévention implémenté dans le programme script AGI. Notre script AGI implémente les contremesures suivantes :

- La réponse avec un message occupé à l'appel de la source suspecte,
- La mise en attente de l'appel pour une période du temps,
- La demande de composer un signal DTMF spécifique afin d'établir l'appel,
- L'utilisation de la machine de réponse automatique (AMD) qui réalise des tests afin de savoir si l'appelant est un humain ou une machine,
- Le transfert de l'appel vers un autre équipement destination (par exemple permettant de

```
env = agi.env
CDR = agi.database_get("blacklist", env["agi_callerid"])
contremesure = int (agi.database_get(env["agi_callerid"],
    "verrou"))

if contremesure == 1:
  try:
    contremesure_ant = int(agi.database_get(env["
        agi_callerid"],env["agi_dnid"]))
    if contremesure_ant == 1:
      None

  except AGIError:
    agi.database_put(env["agi_callerid"],env["agi_dnid"
        ],1)
    agi.stream_file("/var/lib/asterisk/cdr_statistics/
        sounds/text_to_speech_recordings/EssayerPlustard")
    agi.hangup()
```

FIGURE 8.6 – Exemple d'implantation en AGI de la contremesure 1

réaliser le filtrage d'appels par le secrétariat),

– L'inscription de l'appelant dans une liste spécifique et le blocage de l'appel.

Nous avons testé ces mesures de sécurité contre des outils d'attaque SPIT disponibles comme Spitter / Asterisk, Warvox et Voipbot. Les tests mettent en évidence la différence et la complémentarité des contremesures en termes de coûts et avantages.

Pour la première contremesure (voir le code 8.6) qui consiste à répondre par un message occupé, le script AGI exécute en premier un fichier audio appelé "EssayerPlusTard" qui demande à l'appelant de renouveler son appel. Nous exécutons ensuite la commande agi.hangup() qui permet de mettre fin à la session SIP. La deuxième contremesure consiste à mettre l'appel en attente. Le script lance un fichier audio appelé "Appuyer" qui invite la source d'appel à entrer un numéro en exécutant la commande AGI saynumber(nombre). Si l'utilisateur tape ce nombre correctement avant une période de temps, le script établit la session SIP, sinon il exécute agi.hangup(). La troisième contremesure (voir le code 8.7) demande de composer une tonalité DTMF spécifique afin d'établir l'appel. Pour cette mesure, le script AGI lance le fichier audio que "vousDevezAttendre" et met l'appelant en attente en exécutant la commande time.sleep(period). La quatrième contremesure (voir le code 8.8) consiste à effectuer des test AMD pour détecter si l'appelant est un homme ou une machine. Nous vérifions le comportement de l'appelant en exécutant la commande AMD agi.appexec("AMD", "500", "2000").

Enfin, la cinquième contremesure redirige l'appel vers une autre destination (par exemple, le filtrage d'appel fait par le secrétariat), et le blocage de l'appel si l'appelant, paraît pour la tierce partie, suspect.

8.3.4 Système de configuration

Ce dernier composant est responsable de l'exécution des mesures de sécurité décidées par le gestionnaire de risques dans l'infrastructure VoIP à travers un ensemble d'opérations de con-

```
env = agi.env
CDR = agi.database_get("blacklist", env["agi_callerid"])
contremesure = int (agi.database_get(env["agi_callerid"],
    "verrou"))

elif verrou == 3:

  sayit("/var/lib/asterisk/cdr_statistics/sounds/
      text_to_speech_recordings/sorry")
  sayit("/var/lib/asterisk/cdr_statistics/sounds/
      text_to_speech_recordings/you_have_to_wait")
  saynumber(5)
  sayit("/var/lib/asterisk/cdr_statistics/sounds/
      text_to_speech_recordings/seconds")
  sys.stderr.write("SLEEP 5 Seconds \n")
  sys.stderr.flush()
  time.sleep(5)
```

FIGURE 8.7 – Exemple d'implantation en AGI de la contremesure 3

figuration. L'application du même traitement du risque peut varier en fonction de la classe des équipements VoIP. Le traitement du risque concerne tous les éléments de réseau qui peuvent avoir un impact sur l'exposition de l'infrastructure VoIP. Ces éléments comprennent les téléphones et les serveurs VoIP, mais aussi les protocoles de signalisation et du transfert multimédia, les services du support tels que DNS, TFTP et RADIUS, les services intégrés tels que les services de messagerie instantanée et les équipements de sécurité tels que les pare-feux applicatifs.

Ce système peut également permettre de fournir un retour d'expérience sur les opérations précédement exécutées au sein de l'environnement. Par exemple, nous avons travaillé sur le coût effectig d'une contre-mesure. ces données peuvent ensuite être exploitées par le gestionnaire de risques.

8.4 Interactions entre composants

Notre architecture met en œuvre les principaux modèles considérées par notre approche de gestion de risques à savoir ceux requis par le gestionnaire de risques et par la détection d'anomalies dans un serveur VoIP Asterisk. Elle est composée de quatre principaux composants fonctionnels : le système de détection incluant le moniteur d'appel et la détection par anomalies, le gestionnaire de risques (qui intègre le modèle mathématique et les deux algorithmes de mitigation) et le système de configuration.

Le moniteur d'appel est directement relié à la source de données (en l'occurrence, la base de données de CDR). Il estime l'ensemble des caractéristiques définies selon un calendrier périodique. Il transmet, ensuite, les caractéristiques du trafic observé sous un format approprié au détecteur d'anomalies. Ce dernier est responsable de révéler les situations d'attaques et de mesurer leur potentialité en se basant sur des machines à vecteurs supports. Le seuil de détection est un paramètre important à configurer pour aboutir à un bon compromis entre la sensibilité et la spécificité du détecteur d'anomalies. Les mécanismes de retour d'expérience peuvent contribuer

```
env = agi.env
CDR = agi.database_get("blacklist", env["agi_callerid"])
contremesure = int (agi.database_get(env["agi_callerid"],
    "verrou"))

elif contremesure == 4:

  agi.appexec("RINGING")
  agi.appexec("WAIT", "2" )
  agi.answer()
  agi.appexec("AMD", "500 2000" )
  amdstatus= agi.get_variable("AMDSTATUS")
  amdcause= agi.get_variable("AMDCAUSE")
  if amdstatus == "MACHINE":
    agi.stream_file("/var/lib/asterisk/cdr_statistics/
        sounds/text_to_speech_recordings/machine")
    agi.hangup()
  else:
    agi.stream_file("/var/lib/asterisk/cdr_statistics/
        sounds/text_to_speech_recordings/congratulations")
    agi.stream_file("/var/lib/asterisk/cdr_statistics/
        sounds/text_to_speech_recordings/human")
```

FIGURE 8.8 – Exemple d'implantation en AGI de la contremesure 4

à son raffinement.

Pour le gestionnaire des risques, il permet de faire face aux limites intrinsèques du détecteur. Les résultats de la détection d'anomalies sont transmis directement au gestionnaire de risques à un stade précoce (c'est-à-dire même avec des valeurs de potentialité faibles ou partielles). Ces résultats sont intégrés progressivement dans le modèle afin d'estimer le niveau de risque à l'exécution. Le gestionnaire de risque intègre la mise à jour progressive et les algorithmes de mitigation pour contrôler le plan de numérotation de l'IPBX en fonction du niveau de risque estimé et il est responsable de la sélection d'une liste de contremesures quand une situation est détectée comme risquée.

Le système de configurations reçoit des requêtes de configuration du gestionnaire de risques. La requête de configuration contient une liste de couples de données : chaque couple représente une source d'appels et une liste de contremesures à activer (ou désactiver) pour tous les appels provenant de cette source d'appel. Le gestionnaire de configuration agit au niveau du plan de numérotation pour protéger les extensions (les numéros) composées. Une deuxième fonctionnalité consiste à réaliser un retour d'expérience sur l'application des contremesures : le système de configuration envoie un retour au gestionnaire de risques, par exemple des indicateurs montrant si une contremesure a été appliquée avec succès ou non, ou s'il a remarqué des modifications après avoir appliqué une certaine politique. De même, le gestionnaire de risques est capable de régler certains paramètres au niveau du détecteur d'anomalies comme le seuil de détection.

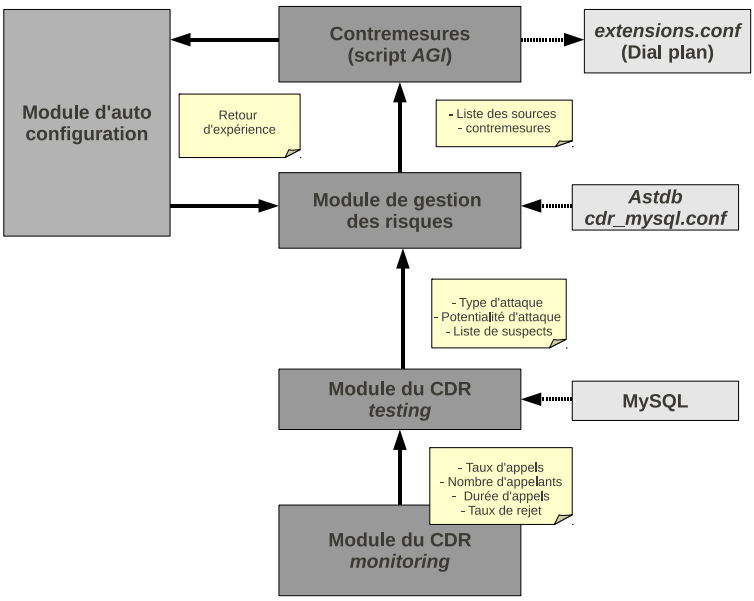


FIGURE 8.9 – Interactions entre les composants

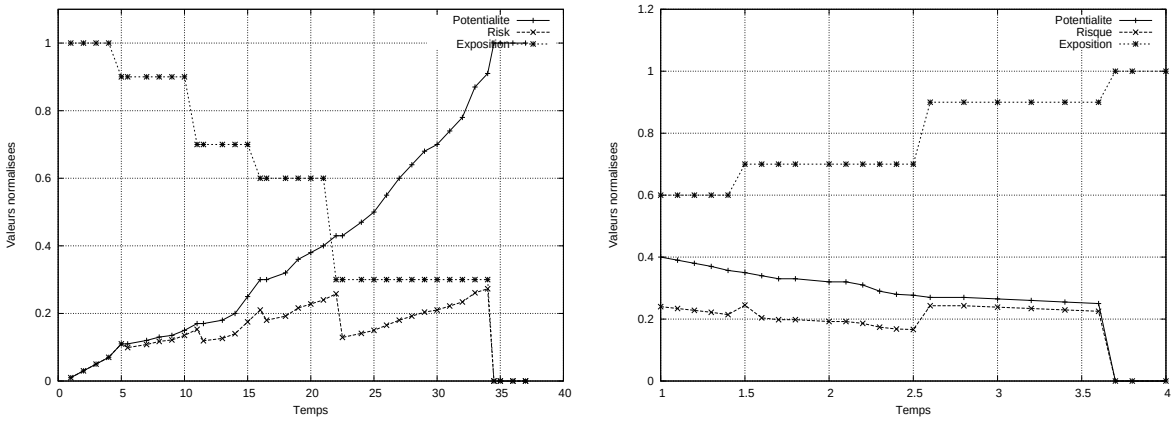
8.5 Scénarios de tests

Nous décrivons ici des scénarios de tests qui ont été mis en œuvre pour évaluer notre implantation.

8.5.1 Évaluation de l'implantation

En complément des expérimentations présentées précédemment, nous avons validé la mise en œuvre de notre approche par expérimentation. Notre approche de prévention est caractérisée par deux algorithmes : l'algorithme de restriction de risques vise à réduire l'exposition de la plateforme VoIP et limiter son accès, le deuxième algorithme est celui du relâchement qui vise à élargir l'exposition de la plateforme VoIP. Nous générons un scénario d'attaques SPIT en augmentant le nombre d'appels malveillants afin d'accroître la potentialité de l'attaque. Toutes les valeurs sont normalisées. Nous avons fixé le seuil de risque à 0,3 et nous avons défini cinq contremesures. Nous retrouvons la contremesure CAPTCHA audio et la mise en attente de l'appelant. Cette implantation teste l'approche sur plusieurs points : nous validons le fonctionnement des algorithmes de prévention ainsi que l'applicabilité des contremesures.

Dans la figure 8.10, nous avons varié la potentialité de 0 à 100% (une attaque complète) et nous avons suivi la variation du risque. Comme nous l'avons noté, le risque augmente en fonction de la potentialité jusqu'à ce qu'il atteigne 0,3 où notre stratégie de risques applique la première contremesure (voir figure 8.10(a)). Ainsi, la valeur de l'exposition varie de 1 à 0,5, ce qui réduit l'accessibilité de l'infrastructure et la valeur du risque tombe à 0,15. Comme la potentialité continue à augmenter et que l'exposition est fixée à 0,5, le risque continue à augmenter progressivement pour atteindre à nouveau 0,3 où notre stratégie de risque applique la deuxième contremesure. Par conséquent, la valeur de l'impact devient égale à 0,4 et le risque retombe à 0,24. Cette procédure de gestion se poursuit avec l'augmentation de la potentialité des menaces. La dernière contremesure (le blocage d'appel) est appliquée et réduit l'exposition à 0,1 et la



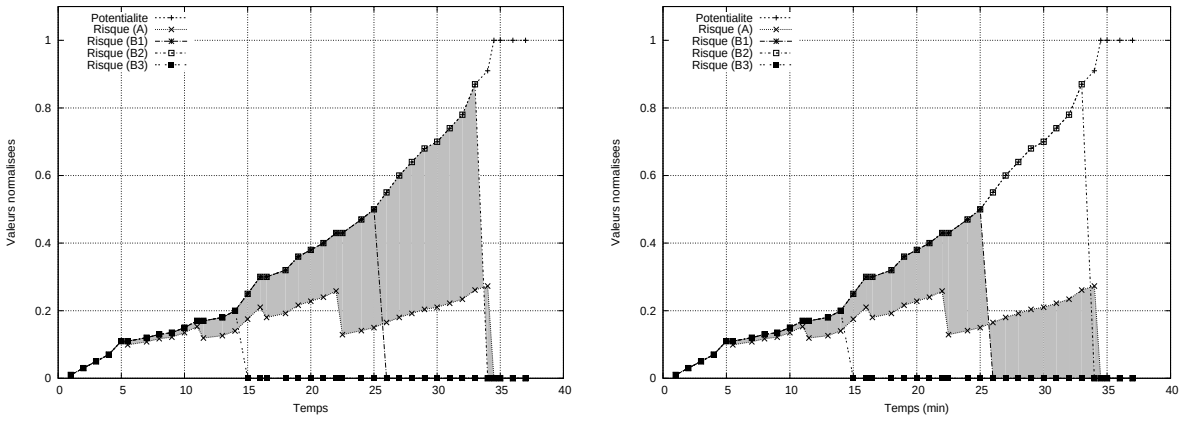
(a) Validation du mécanisme de restriction (b) Validation du mécanisme de restriction

FIGURE 8.10 – Impact des mécanismes de mitigation sur le traitement du risque

valeur de risque est réduit à 0,1.

La deuxième expérimentation valide le fonctionnement du deuxième algorithme de gestion, à savoir l'algorithme de relaxation. Nous avons généré une attaque SPIT dans laquelle la potentialité diminue de 100% à 0%. L'exposition de la plateforme contre l'attaque commence avec une valeur de 0,1 qui signifie que la stratégie de risques applique la quatrième contremesure la plus importante. Comme la potentialité diminue progressivement, l'approche applique son schéma de prévention et élargit l'exposition de 0,1 à 0,4 afin de rendre la plate-forme VoIP plus ouverte. Sur la figure 8.10(b), on constate que le risque global diminue en fonction de l'exposition, avec des pics de risque à chaque fois qu'on change l'exposition de la plateforme. Ce processus se poursuit tant que la potentialité de l'attaque continue à diminuer. Cette expérimentation valide le schéma de traitement de risques de notre architecture quand l'application de contremesures n'est pas nécessaire.

8.5.2 Comparaison des performances



(a) Comparaison par rapport la stratégie B1 (b) Comparaison par rapport la stratégie B2

FIGURE 8.11 – Comparaison avec d'autres stratégies régulières

Nous nous sommes intéressés à comparer notre solution avec des schémas traditionnels de traitements d'attaques et avons quantifié les avantages et les limites de l'intégration des modèles de risque à des systèmes de détection d'intrusions. La figure 8.11 décrit le niveau de risque de notre solution noté A, ainsi que le niveau de risque de trois stratégies alternatives notées B_1, B_2 et B_3. Ces stratégies correspondent au cas d'un système de détection sans un modèle de risque explicite : le niveau de risque est uniquement basée sur la potentialité de l'attaque. Ils diffèrent par leurs sensibilités et leurs spécificités par rapport à la détection des attaques. La première stratégie B_1 fournit un haut taux de vrais positifs (une haute sensibilité) et elle consiste à bloquer les communications VoIP d'un appelant donné dès que la potentialité dépasse une valeur de 0,3. La deuxième stratégie B_2 consiste à bloquer les communications lorsque la potentialité atteint la valeur de 0,5. Cette solution offre une sensibilité et une spécificité moyenne et est considérée par [58] comme une approche de détection de référence. La dernière stratégie de risques B_3 réduit le taux de faux positifs (une spécificité élevée) et bloque la communication de l'appelant lorsque la potentialité de l'attaque est supérieure à 0,7.

Comparaison avec des stratégies traditionnelles

La comparaison de notre solution A avec la première stratégie B_1 montre que le bénéfice de notre stratégie en termes de risque est faible. En effet, la stratégie B_1 rejette rapidement les communications de l'appelant, et le niveau de risque est réduit à zéro. Nous pouvons l'observer sur la figure 8.11(a) à l'instant $t = 15$. Toutefois, le bénéfice en termes de disponibilité du service est assez élevé : notre solution de gestion de risques permet de maintenir le service VoIP pendant une période plus longue, car la contremesure s_5, qui consiste à bloquer les communications de l'appelant, ne prend effet qu'à l'instant t = 34.

Quand nous comparons notre stratégie avec la stratégie B_3, nous nous attendions à un bénéfice significatif en termes de risque, ce qui était le cas. Nous remarquons clairement l'avantage de notre solution de gestion de risques qui est représentée par la zone grise sur la courbe représentant A sur la figure 8.11(a). Au cours des expériences, nous avons quantifié un bénéfice moyen de 32% en termes de risques. Le bénéfice correspond à un nombre d'appels illégaux évités.

L'inconvénient majeur de la stratégie B_3 est que le niveau de risque peut atteindre une valeur élevée avant que l'appelant ne soit considéré comme un attaquant par le système de détection. En particulier, le niveau de risque maximal généré par notre solution est relativement faible (R_{seuil} mis à 0,25) alors qu'il peut être significativement plus élevé avec la stratégie B_3 (une valeur de 0,7 sur la figure 8.11(a)). En termes de disponibilité du service, les deux approches A et B_3 fournissent des performances similaires, mais le service VoIP est progressivement limité avec notre approche de gestion de risques.

La comparaison de notre stratégie de gestion de risques avec la stratégie B_2 révèle également des résultats intéressants. L'avantage en termes de disponibilité du service est important : la stratégie B_2 bloque la communication VoIP de l'appelant à l'instant t = 26 alors que notre solution A maintient le service VoIP jusqu'à l'instant t = 34 sur la figure 8.11(b). Nous avons quantifié un bénéfice allant jusqu'à 14% dans notre série d'expériences. La quantité de risque moyenne est similaire pour les deux approches A et B_2. Toutefois, le niveau de risque maximal est de 0,25 avec notre solution et de 0,5 avec la stratégie B_2. Ce phénomène est similaire à celui observé avec la stratégie B_3 et est dû à l'application progressive des mesures de sécurité.

Comparaison avec d'autres comportements d'attaque temporelles

Nous avons comparé les performances de notre solution avec d'autres comportements temporels des attaques SPIT. Nous avons considéré quatre types de comportements temporels et évalué le bénéfice de notre approche par rapport à la stratégie B_2 (qui garantit un compromis entre la sensibilité et la spécificité). Nous avons expérimenté un premier type de comportement C_1 consistant à augmenter la potentialité de l'attaque (le niveau SPIT) sur une courte période de temps. Cela correspond à un scénario où le système de détection peut facilement identifier l'agresseur. Un second type de comportement C_2 augmente la potentialité jusqu'à la mi-période du temps, puis elle la diminue. Ceci illustre le cas d'une attaque qui n'est pas continue au cours du temps, c'est à dire des augmentations et des diminutions de la potentialité d'attaque. Un troisième type d'attaque C_3 vise à augmenter la puissance jusqu'à ce que le seuil de l'attaque de la stratégie B_2 est atteint puis la réduire par la suite ; ce qui réduit les avantages du schéma de gestion des risques après la valeur seuil. Pour le quatrième type comportemental C_4, nous avons réduit la potentialité juste avant le seuil de risque de B_2 est atteint. Comme prévu, le type C_1 de comportement génère les pires performances tandis que les autres comportements C_2, C_3 et C_4 ont fourni un avantage en termes de risque pouvant aller jusqu'à 41% au cours des expériences.

8.6 Synthèse

Nous avons implanté notre solution de gestion de risques dans un serveur IPBX Asterisk typiquement utilisé dans le contexte d'infrastructure VoIP d'entreprise. L'objectif était de compléter l'évaluation de performances faite par simulation et d'effectuer une analyse pratique par le prototypage. L'architecture de notre solution comprend principalement un système de détection contenant deux composants (un moniteur d'appels et un détecteur d'anomalies), un gestionnaire de risques qui implémente les algorithmes de mitigation et le modèle quantitatif de risques, un système de configuration contenant un mécanisme de retour d'expérience sur le fonctionnement du traitement. Cette implantation a permis d'évaluer l'impact du traitement sur l'évolution du risque dans les infrastructures VoIP. Pour cela, nous avons élaboré un ensemble d'attaques SPIT et suivi le comportement de notre stratégie de gestion de risques. Par ailleurs, nous avons pu comparer notre approche à d'autres stratégies traditionnelles et également la confronter à d'autres comportements d'attaques temporels. En conclusion, nous avons montré que notre stratégie de gestion de risques peut être mise en œuvre au sein d'un serveur IPBX Asterisk pour mitiger les risques dans le contexte d'un réseau d'entreprise VoIP et en avons évalué les bénéfices et les limites.

Conclusion

Chapitre 9

Conclusion générale

Sommaire

9.1 Résumé des contributions

La sécurité est considérée comme un défi majeur pour les services VoIP où les conversations sont généralement moins confinées en comparaison à la téléphonie classique. L'émergence de la téléphonie IP a conduit à l'apparition de nouvelles menaces de sécurité. Celles-ci incluent les attaques héritées de la couche IP comme le déni de service, l'écoute et l'analyse du trafic, mais aussi les attaques spécifiques à la couche applicative comme les attaques SPIT et les attaques par messages SIP malformés qui exploitent les vulnérabilités d'implantation du protocole. Dans ce contexte, un ensemble de mécanismes de détection et de protection a été proposé pour identifier et bloquer les attaques contre les services VoIP. Cependant, ces mécanismes peuvent avoir un impact non négligeable sur les performances et le fonctionnement du service. La gestion des risques se présente comme une solution pour répondre à ce compromis entre sécurité et performance.

La gestion des risques est un processus qui consiste à identifier les risques, les évaluer et prendre des mesures pour les réduire à un niveau acceptable. L'objectif de cette thèse est d'appliquer et d'automatiser ce processus dans les réseaux et services VoIP, afin de contrer dynamiquement les attaques de sécurité, en adaptant l'exposition de ces services à travers l'activation et la désactivation de contremesures. L'approche repose notamment sur un modèle quantitatif de risques qui prend en charge l'estimation et l'évaluation des risques, un ensemble de contremesures qui a pour objectif de réduire l'exposition de l'infrastructure VoIP, ainsi que des algorithmes de mitigation qui permettent le couplage entre le modèle de risques et les contremesures.

Nos travaux de recherche sur la gestion des risques dans les réseaux et services VoIP comprennent trois axes principaux. Nous avons dans un premier temps travaillé sur l'application de la

gestion des risques dans les réseaux VoIP d'entreprise, en s'intéressant aux méthodes de détection par machines à vecteurs supports et au problème de paramétrisation du modèle de risques. Nous avons ensuite étendu notre approche au cas des réseaux décentralisés P2PSIP en définissant et en évaluant des contremesures spécifiques. Enfin, nous avons quantifié la complémentarité de la solution avec le framework de sécurité RELOAD et étudié sa possible externalisation dans le contexte du cloud.

9.1.1 Gestion des risques dans les architectures VoIP d'entreprise

Une stratégie adaptative contre les attaques SPIT

Nous avons tout d'abord proposé une stratégie de gestion des risques dans les réseaux VoIP en se focalisant sur la prévention des attaques SPIT [79]. Ces dernières peuvent considérablement altérer le fonctionnement du réseau téléphonique VoIP. La gestion des risques permet de répondre au compromis entre sécurité et performance qui sont toutes deux des éléments très importants pour la téléphonie sur IP. L'objectif consiste à adapter en permanence l'exposition des équipements et des services VoIP en activant ou désactivant les contremesures de sécurité d'une façon dynamique. Le gestionnaire de risques s'appuie sur un modèle quantitatif qui a pour but l'estimation et l'évaluation des risques. Le modèle définit le risque à travers trois paramètres, qui sont la potentialité de la menace, l'exposition de l'infrastructure et les conséquences subies lorsque l'attaque se réalise. Le gestionnaire s'appuie sur un ensemble de contremesures qui a pour but de contrer les attaques et leur source d'une manière progressive, et sur deux algorithmes de mitigation (un algorithme de restriction et un algorithme de relâchement) qui permettent le traitement des risques à travers soit l'activation de contremesures lorsque le niveau de risque est élevé soit la désactivation de celles-ci lorsque le niveau de risque est faible afin de réduire le coût induit par celles-ci.

Dans ce travail, nous avons pu montrer la faisabilité de notre stratégie de risques et pu évaluer ses performances dans le cadre des attaques SPIT. Nous avons en particulier quantifié l'impact des algorithmes de mitigation sur ces attaques et comparé la solution à d'autres approches traditionnelles. Les mécanismes de restriction et de relâchement permettent de fournir une réponse progressive à travers la mise en œuvre de différentes contremesures au cours de l'exécution. Si l'application et l'automatisation de la gestion de risques permet de répondre au compromis entre sécurité et performance pour les services VoIP, les performances d'une telle stratégie sont également fortement dépendantes de celles de la méthode de détection adoptée.

Couplage de la stratégie de risques aux machines à vecteurs supports

Nous avons couplé notre stratégie à une méthode de détection d'anomalies basée sur les machines à vecteurs supports (SVM) [68]. L'objectif étant de montrer comment une telle méthode peut être intégrée à notre modèle de risques. Ce couplage permet un contrôle plus précis de l'exposition de l'infrastructure en exploitant une méthode de détection de plus grande sensibilité et spécificité. L'utilisation de machines à vecteurs supports a déjà montré son efficacité pour observer et contrôler le trafic de signalisation VoIP dans [74]. Nous avons intégré les paramètres de détection de cette méthode en étendant notre modèle de risques et conçu une architecture pour soutenir notre stratégie [79]. Nous avons décrit le déploiement de cette dernière dans un réseau d'entreprise qui repose sur un serveur VoIP. Le système de détection d'anomalies permet de caractériser le trafic et le comportement normal des utilisateurs, et peut ensuite identifier des anomalies par déviation de ce comportement. A partir de ces anomalies, notre stratégie de gestion peut ensuite traiter les attaques potentielles via l'activation de contremesures. Les performances

du couplage ont été évaluées par le prototypage et un ensemble de simulations. Nous avons quantifié l'impact de la valeur seuil de risque sur le fonctionnement de notre système, ainsi que l'influence de la sensibilité et de la spécificité de la détection sur les performances globales de l'approche.

Dans ce travail, nous avons montré comment l'intégration d'une méthode de détection d'anomalies à base de machines à vecteurs supports est possible au sein de notre stratégie de gestion des risques pour la téléphonie sur IP. A travers une évaluation des performances, nous avons constaté que cette intégration peut clairement contribuer à une réponse plus appropriée face aux attaques SPIT. La méthode de détection d'anomalies complète notre solution de gestion de risques. Si nous l'avons expérimenté pour les attaques SPIT, cette méthode de détection est également exploitable pour d'autres menaces de la téléphonie sur IP.

Modèle étendu de la stratégie de gestion

Nous avons ensuite étendu notre modèle de risques à une plus large variété d'attaques [78]. L'objectif étant toujours d'adapter l'exposition des services VoIP au regard de la potentialité des attaques, à travers l'activation ou la désactivation dynamique des contremesures. Pour ce faire, nous avons commencé par classer les attaques de sécurité relatives à la téléphonie sur IP par rapport à leurs propriétés d'observabilité. Nous avons distingué trois classes d'attaques de sécurité dans les infrastructures VoIP : les attaques VoIP observables par détection d'anomalies, les attaques VoIP observables par détection de signatures et les attaques VoIP difficilement ou non observables. Nous avons ensuite analysé comment ces propriétés influent sur les performances de notre stratégie de gestion de risques. Les bénéfices et limites de notre approche sont en effet fortement liés à ces propriétés. Nous avons notamment évalué l'impact de la valeur seuil sur le traitement des attaques correspondant aux différentes classes, ainsi que les performances de notre solution par rapport à des attaques dont la taille de signature est restreinte. Nous avons également évalué comment le nombre de contremesures disponibles peut avoir un impact sur l'évolution du risque.

Dans ce travail, et au regard de l'évaluation des performances réalisée, nous avons montré que l'application de la gestion des risques n'apporte pas nécessairement de bénéfices pour l'intégralité des attaques VoIP. Les attaques difficilement observables ne peuvent être traitées d'une façon progressive et continue, et peuvent nécessiter la mise en oeuvre systématique de contremesures. En revanche, la gestion des risques peut apporter des résultats satisfaisants pour mitiger dynamiquement aussi bien les attaques détectables par signatures que celles détectables par anomalies, à la condition de disposer également de contremesures adéquates.

Paramétrisation et auto-configuration

Les modèles de risques intègrent généralement un grand nombre de paramètres, ce qui peut considérablement complexifier la tâche de configuration et de déploiement. De plus, les valeurs de ces paramètres peuvent varier suivant l'environnement et la criticité des services VoIP. Dans ce contexte, nous avons proposé une méthode d'auto-configuration pour soutenir le paramétrage de notre solution [29]. Cette méthode permet de simplifier la tâche de configuration du modèle de risques en s'appuyant sur des mécanismes de retour d'expérience. Nous l'avons spécifié de façon théorique, puis l'avons intégré dans notre modèle de risques afin d'en configurer certains paramètres. Nous avons considéré comme étude de cas, le paramètre correspondant au coût induit par l'application d'une contremesure. Ce paramètre est crucial car le processus de gestion de risques vise à minimiser ce coût tout en maintenant un niveau de risque faible. Nous avons

évalué les bénéfices de cette méthode d'auto-configuration pour les réseaux VoIP d'entreprise et quantifié l'impact du taux d'erreur d'un paramètre sur le comportement de notre stratégie.

Dans ce travail, nous avons montré qu'une telle méthode d'auto-configuration basée sur le retour d'expérience permet de raffiner certains paramètres de notre modèle de gestion de risques et ainsi contribue à une sélection plus précise des contremesures. Nous avons également observé, par l'évaluation de performances, qu'une erreur relative à l'évaluation du coût d'application des contremesures peut conduire dans certains cas au dysfonctionnement des services VoIP, voire même mener à leur indisponibilité.

9.1.2 Extension aux infrastructures P2PSIP

Contrôle d'exposition dans les réseaux P2PSIP

Une part importante du travail a aussi consisté à étendre notre solution de gestion de risques à des infrastructures VoIP davantage décentralisées, notamment celles s'appuyant sur le protocole P2PSIP où les services d'enregistrement et de localisation sont implantés à travers une table de hachage distribuée (DHT). Nous avons étudié dans ce cadre différents scénarios d'attaques, notamment ceux reposant sur la manipulation des enregistrements au sein du réseau P2PSIP. Nous avons analysé de nouvelles techniques de détection et de prévention pour ces attaques. Notre travail a consisté à distribuer les différentes étapes du processus de gestion de risques dans le réseau P2PSIP [28]. Nous avons considéré un ensemble de contremesures spécifiques pour cet environment, telles que des méthodes de corrections des enregistrements, d'authentification des pairs et des techniques de réplication pour la table de hachage distribuée. Le modèle de risques prend en compte les différentes caractéristiques intrinsèques de ces contremesures pour une meilleure évaluation de leurs impacts. Les bénéfices et limites ainsi que le passage à l'échelle de la solution ont été évalués à travers un ensemble d'expériences réalisées avec le simulateur OMNeT++. Nous avons notamment quantifié le coût induit par les contremesures en termes de trafic et de délai supplémentaire à l'établissement des sessions d'appels.

Dans ce travail, nous avons montré que notre stratégie de gestion de risques peut également être appliquée aux infrastructures P2PSIP et qu'elle peut offrir de meilleures performances comparativement à d'autres stratégies de sécurité, en particulier lorsque l'on s'intéresse à des réseaux réellement décentralisés. La gestion de risques dépend de la technique de détection envisagée dans ces réseaux. Dans le cas d'une configuration décentralisée, la méthode de détection exige généralement un niveau élevé de confiance entre les différents pairs qui peuvent coopérer ensemble pour détecter et estimer la potentialité de l'attaque.

Intégration de mécanismes de confiance dans l'architecture RELOAD

Nous nous sommes également intéressés à l'architecture RELOAD qui offre une infrastructure de sécurité s'appuyant sur un serveur central de certification pour les réseaux P2PSIP [28]. Nous sommes alors dans une configuration où la sécurité reste centralisée. L'architecture RELOAD définit plusieurs niveaux de sécurité : un niveau de sécurité pour la communication entre les différents nœuds et un deuxième niveau qui consiste à crypter les données enregistrées dans la DHT par une clé privée afin de renforcer leur authenticité. Nous nous sommes focalisés sur le traitement des attaques résiduelles de cette architecture. En effet, cette solution de sécurité ne traite pas certaines attaques comme le refus de service. Nous avons proposé une solution complémentaire qui consiste à intégrer un modèle de confiance basé sur l'algorithme *eigentrust* pour traiter ces attaques. L'objectif est d'exploiter les informations fournies par les modèles de

confiance pour contrôler les nœuds vulnérables ou plus généralement les nœuds ayant un niveau de confiance faible et mettre en oeuvre les mécanismes de sécurité requis.

Dans ce travail, nous avons montré que l'intégration de mécanismes de confiance dans l'architecture RELOAD permet de mitiger les attaques résiduelles en limitant l'impact des pairs malveillants dont le niveaux de confiance a été évalué comme faible. Une estimation de la valeur de confiance permet aux clients SIP de les éviter et d'utiliser prioritairement les pairs dont le niveau de confiance est élevé pour router leurs requêtes et enregistrer leurs informations dans le réseau P2PSIP.

9.1.3 Externalisation des contremesures comme services dans le cloud

Nous nous sommes enfin intéressés à la téléphonie sur IP dans le cloud. Ce dernier représente un nouveau paradigme pour l'accès et la distribution de ressources informatiques d'une manière virtualisée. Il fournit ces ressources à grande échelle et à la demande. Le cloud offre un niveau d'abstraction élevé pour permettre aux utilisateurs de déployer dynamiquement des infrastructures et exploiter différents services. Si le cloud a récemment servi comme support pour réaliser des attaques telles que des attaques par déni de services, nous considérons qu'il offre aussi de nouvelles perspectives pour le déploiement des mécanismes de sécurité. L'objectif de cette intégration est double. D'un côté, elle vise à protéger d'une façon dynamique les services VoIP dans le cloud. De l'autre côté, la solution doit permettre de profiter de la robustesse et de la flexibilité du cloud pour l'instanciation de notre solution de gestion de risques. En particulier, la conception de contremesures VoIP comme des services facilite leur réplication et l'intégration de nouvelles contremesures. Elle permet également de plus facilement remédier aux défaillances et pannes auxquelles les contremesures peuvent être elles-mêmes soumises dans ce contexte.

Dans ce travail, nous avons proposé plusieurs stratégies pour assurer l'application des contremesures. Nous avons montré comment ces dernières peuvent être sélectionnées ou répliquées lorsque l'une d'entre elles n'est plus disponible ou est soumise à une panne ou à une attaque. Pour évaluer les performances de ces stratégies pour le cloud, nous avons réalisé un ensemble d'expérimentations par simulation pour montrer comment le traitement du risque est impacté par la défaillance de contremesures et comment nos stratégies de correction permettent de maintenir le niveau de sécurité.

9.2 Perspectives

9.2.1 Application à d'autres services temps réel

Parmi nos perspectives de recherche, nous sommes intéressés par étendre notre stratégie de gestion de risques à d'autres services temps réel comme la vidéo à la demande (VoD) et la télévision IP ainsi qu'à d'autres architectures multimédia telles que celles fondées sur IMS[28]. Ces infrastructures offrent des services qui sont à la fois sensibles en termes de sécurité et de performance. Le travail consiste à évaluer si notre approche de gestion de risques est facilement instanciable dans ces environnements et si de nouvelles contremesures sont également envisageables. Dans le même contexte, nous souhaitons poursuivre nos travaux dans le cloud en étudiant de nouveaux mécanismes pour la fiabilisation des contremesures et le partage potentiel de celles-ci entre différents pairs de confiance, de façon complémentaire à une solution où les contremesures sont uniquement proposées par des fournisseurs certifiés.

28. IP Multimedia Subsystem

9.2.2 Conception d'une algèbre pour les contremesures

Les contremesures ont un rôle important dans la phase de traitement des risques car elles permettent de réduire l'exposition des services VoIP face aux attaques. Cependant, elles ont des natures et des caractéristiques différentes. Il peut donc être difficile de quantifier automatiquement l'impact d'un ensemble de contremesures cumulées. En particulier, il existe des contremesures qui traitent des attaques communes. Ainsi, l'impact d'une contremesure peut être moins important (voire nul) lorsqu'une autre contremesure donnée est déjà présente. Pour cela, il faut prendre en compte ces caractéristiques lorsque l'on évalue l'exposition après une accumulation de contremesures. Le travail consistera à définir les différentes relations qui peuvent exister entre les contremesures, comme l'accumulation ou l'exclusivité dans leurs applications, et à élaborer mathématiquement un formalisme qui permette ensuite de simplifier l'évaluation de leur impact sur l'infrastructure. Nous pourrons ainsi plus facilement évaluer l'impact de toute nouvelle contremesure sur l'ensemble de celles déjà appliquées.

9.2.3 Couplage à des modèles de vulnérabilités

L'exposition d'un service dépend de ses vulnérabilités. L'analyse fine des vulnérabilités de configuration est devenue possible avec la standardisation de langages de description tels que le langage OVAL [29]. La gestion des vulnérabilités consiste à observer la configuration des équipements, identifier la présence de configurations vulnérables et effectuer les opérations de maintenance nécessaires, comme la modification de paramètres de configuration et / ou l'application de certains correctifs de sécurité. Le langage OVAL permet de décrire de façon standard une vulnérabilité sous la forme d'une combinaison logique de tests à réaliser sur un équipement ou un système. Ces descriptions sont ensuite utilisées pour identifier les vulnérabilités. Les modèles de vulnérabilités sont complémentaires aux contremesures qui peuvent ensuite être appliquées sur l'infrastructure. Ils sont souvent mis en œuvre en amont pour appliquer les correctifs nécessaires lorsqu'ils sont disponibles. L'intégration des modèles de vulnérabilités au sein de notre solution de gestion de risques apparaît ici comme prometteur et contribuera à améliorer la quantification de l'exposition du système et ainsi améliorer l'estimation du risque.

29. OVAL, Open Vulnerability and Assessment Language, MITRE, http ://oval.mitre.org/

9.3 Publications relatives

1. **Conférences internationales avec actes et comité de lecture**

   - Dabbebi O., Badonnel R., Festor O., Automated Runtime Risk Management for Voice over IP Networks and Services. In Proceedings of the IEEE/IFIP Network Operations And Management Symposium (IEEE/IFIP NOMS 2010), Osaka, Japan, April 2010, pp.57-64

   - Dabbebi O., Badonnel R., Festor O., Managing Risks at Runtime in VoIP Networks and Services. In Proceedings of the IFIP International Conference on Autonomous Infrastructure, Management and Security (IFIP AIMS 2010), Zurich, Switzerland, Juin 2010, pp.89-92

   - Nassar M., Dabbebi O., Badonnel R., Festor O., Risk Management in VoIP Infrastructures using Support Vector Machines. In Proceedings of the IEEE/IFIP International Conference on Network Network and Service Management (IEEE/IFIP CNSM 2010), taux d'acceptation : 15.3%, Canada, October 2010, pp.48-55

   - Dabbebi O., Badonnel R., Festor O., Econometric Feedback for Runtime Risk Management in VoIP Architectures. In Proceedings of the IFIP International Conference on Autonomous Infrastructure, Management and Security (IFIP AIMS 2011), Nancy, France, Juin 2011, pp.26-37

   - Dabbebi O., Badonnel R., Festor O., A Broad-Spectrum Strategy for Runtime Risk Management in VoIP Enterprise Architectures. In Proceedings of the IEEE/IFIP International Conference on Integrated Network Management (IEEE/IFIP IM 2011), Dublin, Ireland, May 2011, pp.478-484

   - Dabbebi O., Badonnel R., Festor O., Dynamic Exposure Control in P2PSIP Networks. In Proceedings of the IEEE/IFIP Network Operations And Management Symposium (IEEE/IFIP NOMS 2012), taux d'acceptation de 26.4%, USA, April, 2012, pp. 261-268

   - Dabbebi O., Badonnel R., Festor O., A Trust-based Strategy for Addressing Residual Attacks in RELOAD Architectures. In Proceedings of the IEEE International Conference on Communications (IEEE ICC 2012), Ottawa, Canada, Juin 2012

2. **Autres travaux de recherche soumis**

   - Dabbebi O., Badonnel R., Festor O., An Online Risk Management Strategy for VoIP Enterprise Infrastructures, INRIA Research Report 2012, article soumis à un journal international

   - Dabbebi O., Badonnel R., Festor O., Leveraging Countermeasure as a Service for VoIP Security in the Cloud, INRIA Research Report 2012, article soumis à un journal international

Bibliographie

[1] Concepts and Terminology for Peer to Peer SIP, IETF Internet Draft, www.p2psip.org/drafts/draft-ietf-p2psip-concepts-02.html.

[2] SIP, Session Initiation protocol, www.ietf.org/rfc/rfc3261.txt.

[3] VoIP audit, Telephony Security and Fraud Protection, www.voipshield.com/products/voipaudit-overview.php.

[4] ISO/IEC 27005, Information Security Risk Management, www.iso.org.

[5] RFC 4346, SIP and TLS, tools.ietf.org/html/rfc4346.

[6] RFC 3851, S/MIME, www.ietf.org/rfc/rfc3851.txt.

[7] Computer Literacy Tests : Are You Human ?, Lev Grossman, Times Magazine, June 2008.

[8] CAPTCHA cost, virtualblight, www.virtualblight.com/articles/featu- re/hidden-cost-of-captcha-one-billion-dollars-every-month/ .

[9] R language, R project, cran.r-project.org/.

[10] A P2P Approach to SIP Registration, IETF Internet Draft, www.p2psip.org/drafts/draft-bryan-p2psip-dsip-00.html.

[11] Guidelines for Automatic Data processing Physical Security and Risk Management. *National Bureau of Standards, Department of Commerce, USA*, 1974.

[12] H. Abdelnur, R. State, and O. Festor. KiF : a Stateful SIP Fuzzer. In *Proc. of the Principles, Systems and Applications of IP Telecommunications. First International Conference, (IPTComm'07)*, pages 47–56, New York, USA, July, 2007.

[13] A. Abdul-Rahman and S. Hailes. A distributed trust model. In *Proc. of the 1997 Workshop on New Security Paradigms (NSPW '97)*, pages 48–60, New York, NY, USA, 1997. ACM.

[14] J. P. Anderson. Computer Security Threat Monitoring and Surveillance. Technical report, Technical Report Contract 79F26400.

[15] B. Barak, A. Herzberg, D. Naor, and E. Shai. The Proactive Security Toolkit and Applications. In *Proc. of the 6th ACM conference on Computer and communications security (CCS'99)*, pages 18–27, New York, USA, 1999. ACM.

[16] E. J. Basart. Cloud VoIP System With Bypass for IP Media, April, 2011. Un brevet.

[17] T. Bedford and R. Cooke. *Probabilistic Risk Analysis : Foundations and Methods.* Cambridge ; New York : Cambridge University Press, April, 2001.

[18] A. Behl. Emerging Security Challenges in Cloud Computing : An Insight to Cloud Security Challenges and their Mitigation. In *Proc. of the Information and Communication Technologies conference (WICT'11)*, dec. 2011.

[19] N. Bonelli, S. Giordano, G. Procissi, and R. Secchi. BRUTE : a High Performance and Extensible Traffic Generator. In *Proc. of International Symposium on Performance and Extensible Traffic Generator (SPECTS'05)*, PA, USA, July 2005.

[20] M. Castro, P. Druschel, A. Ganesh, A. Rowstron, and D. Wallach. Secure Routing for Structured Peer-to-peer Overlay Networks. *SIGOPS Oper. Syst. Rev.*, 36(SI) :299–314, December 2002.

[21] C. Chang and C. Lin. *LIBSVM : a Library for Support Vector Machines*, 2001. Software available at www.csie.ntu.edu.tw/ cjlin/libsvm.

[22] C. Chang and C. Lin. LIBSVM : A Library for Support Vector Machines. *ACM Transactions on Intelligent Systems and Technology*, 2 :27 :1–27 :27, 2011. Software available at www.csie.ntu.edu.tw/ cjlin/libsvm.

[23] J. Chen, S. Wu, Y. T. Larosa, P. Yang, and Y. Li. IMS Cloud Computing Architecture for High-quality Multimedia Applications. In *Proc. of the International Wireless Communications and Mobile Computing Conference (IWCMC'11)*, 2011.

[24] Jiann-Liang Chen, Szu-Lin Wu, Yanuarius Teofilus Larosa, Pei-Jia Yang, and Yang-Fang Li. Ims cloud computing architecture for high-quality multimedia applications. *Proc. of the 7th International Wireless Communications and Mobile Computing Conference (IWCMC'11)*, pages 1463–1468, July 2011.

[25] T. Cholez, I. Chrisment, and O. Festor. Evaluation of Sybil Attacks Protection Schemes in KAD. In *Proc. of the 3rd International Conference on Autonomous Infrastructure, Management and Security : Scalability of Networks and Services*, (AIMS'09), pages 70–82, Berlin, Heidelberg, 2009. Springer-Verlag.

[26] D. Comer and J. C. Lin. Probing TCP implementations. In *Proc. of the USENIX Security Symposium (USENIX'94)*, 1994.

[27] O. Dabbebi, R. Badonnel, and O. Festor. A Broad-spectrum Strategy for Runtime Risk Management in VoIP Enterprise Architectures. In *Proc. of the 12th IFIP/IEEE Integrated Network Management (IM'11)*, pages 478–484, Dublin, Ireland, May, 2011.

[28] O. Dabbebi, R. Badonnel, and O. Festor. Dynamic exposure control in P2PSIP networks. In *Proc. of the IEEE Network Operations and Management Symposium (NOMS'12)*, pages 261–268, Maui, HI, April, 2012.

[29] O. Dabbebi, R. Badonnel, and O. Festor. Econometric Feedback for Runtime Risk Management in VoIP Architectures. In *Proc. of the 5th International Conference on Autonomous Infrastructure, Management, and Security (AIMS'11)*, pages 26–37, Nancy, France June 2011.

[30] O. Dabbebi, R. Badonnel, and O. Festor. Trust-based Strategy for Addressing Residual Attacks in the RELOAD Architecture. In *Proc. of the International Conference on Communications (ICC'12)*, pages 261–268, Ottawa, Canada, June, 2012.

[31] O. Dabbebi, R. Badonnel, and O. Festor. Leveraging Countermeasure as a Service for VoIP Security in the Cloud. In *INRIA Research Report, papier invité à être soumis au journal international de IJNM*, Septembre 2012.

[32] O. Dabbebi, R. Badonnel, and O. Festor. An Online Risk Management Strategy for VoIP Enterprise Infrastructures. In *INRIA Research Report, papier invité à être soumis au journal international de JNSM*, Septembre 2012.

[33] O. Dabbebi, R. Badonnel, and O. Festor. Managing Risks at Runtime in VoIP Networks and Services. In *Proc. of the 4th International Conference on Autonomous Infrastructure, Management and Security (AIMS'10)*, pages 89–92, Zurich, Switzerland, June, 2010.

[34] D. E. Denning. An Intrusion-Detection Model. *IEEE Transactions on Software Engineering*, 13(2) :222–232, 1987.

[35] S. Deon. *VoIP et ToIP Asterisk : La téléphonie sur IP (conception, installation, configuration, déploiement)*. Eni editions, 2001.

[36] N. d'Heureuse, J. Seedorf, S. Niccolini, and T. Ewald. Protecting SIP-based Networks and Services from Unwanted Communications. In *Proc. of IEEE/Global Telecommunications Conference (GLOBECOM'08)*, December 2008.

[37] T. S. Dillon, C. Wu, and E. Chang. Cloud Computing : Issues and Challenges. In *Proc. of The IEEE International Conference on Advanced Information Networking and Applications (AINA'10)*, pages 27–33, 2010.

[38] J. R. Douceur. The sybil attack. In *Proc. of the First International Workshop of the Peer-to-Peer Systems (IPTPS'02)*, volume 2429 of *Lecture Notes in Computer Science*, pages 251–260. Springer, 2002.

[39] D.Watson, M. Smart, G. R. Malan, and F. Jahanian. Protocol scrubbing : network security through transparent flow modification. 12 :261–273, 2004.

[40] H. Dwivedi. *Hacking VoIP : Protocols, Attacks, and Countermeasures.* No Starch Press, San Francisco, CA, USA, 1st edition, 2008.

[41] S. Ehlert, Y. Rebahi, and T. Magedanz. Intrusion Detection System for Denial of Service Flooding Attacks in SIP Communication Networks. *Int. J. Secur. Netw.*, 4(3) :189–200, July 2009.

[42] A. Gehani and G. Kedem. RheoStat : Real Time Risk Management. In *Proc. of the 7th International Symposium on Recent Advances in Intrusion Detection (RAID'04)*, pages 15–17, Sophia Antipolis, France, 2004.

[43] H. Peter Graf, E. Cosatto, L. Bottou, I. Durdanovic, and V. Vapnik. Parallel Support Vector Machines : The Cascade SVM. In *Proc. of the Advances in Neural Information Processing (NIPS'04)*, Vancouver, Canada, september 2004.

[44] M. Guimaraes and M. Murray. Overview of Intrusion Detection and Intrusion Prevention. In *Proc. of the 5th annual conference on Information security curriculum development (InfoSecCD'08)*, pages 44–46, New York, NY, USA, 2008. ACM.

[45] M. Handley, V. Paxson, , and C. Kreibich. Network intrusion detection : evasion, trafic normalization, and end-to-end protocol semantics. In *Proc. of the USENIX Security Symposium (USENIX'01)*, pages 9–9, Berkeley, USA, 2001.

[46] M. Hansen and al. Developing a Legally Compliant Reachability Management System as a Countermeasure against SPIT. In *Proc. of the 3rd Voice Over IP Security Workshop (VSW 06)*, June 2008.

[47] J. Heikkil and A. Gurtov. Filtering SPAM in P2PSIP Communities with Web of Trust. In *Proc. of the Security and Privacy in Mobile Inf. and Comm. Systems Conference (MobiSec'09)*, volume 17, pages 110–121, Turin, Italy, June 2009.

[48] A. Hossain, M. Nassar, and A. Rahman. Comparison of Finite mixture of ARMA-GARCH, Back Propagation Neural Networks and Support-Vector Machines in Forecasting Returns. *Departement of Finance & Banking, Rajshahi University.*

[49] J. G. Caldwell. *The Box-Jenkins Forecasting Technique.* PhD thesis, University of North Carolina, 1971.

[50] J. Hamilton. *Time Series Analysis.* Princeton Univ. Press., 1994.

[51] J. Hamilton. *Social Engineering : The Art of Human Hacking.* Wiley Publidhing, Inc, 2011.

[52] C. Jampathon. P2PSIP Security. Master's thesis, Helsinki University of Technology, 2008.

[53] J. T. Kajiya. Ray Tracing Parametric Patches. In *Proc. of the 9th annual conference on Computer graphics and interactive techniques (SIGGRAPH'82)*, pages 245–254, New York, NY, USA, 1982. ACM.

[54] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina. The Eigentrust Algorithm for Reputation Management in P2P Networks. In *Proc. of the 12th International Conference on World Wide Web*, (WWW'03), pages 640–651, New York, USA, 2003. ACM.

[55] A. D. Keromytis. A Survey of Voice over IP Security Research. In *Proc. of the 5th International Conference Information Systems Security, (ICISS'09)*, volume 5905, pages 1–17. Springer, Kolkata, India, December, 2009.

[56] A. D. Keromytis. Voice over IP : Risks, Threats and Vulnerabilities. In *Proc. (electronic) of the Cyber Infrastructure Protection Conference (CIP'09).* Springer, new York, USA, June, 2009.

[57] A. D. Keromytis. A look at VoIP Vulnerabilities. In *Proc. of the 18th USENIX Security Symposium (USENIX'10)*, San Diego, Febrary, 2009.

[58] H. Kim, M. J. Kim, Y. Kim, and H. C. Jeong. DEVS-Based Modeling of VoIP Spam Callers' behavior for SPIT level Calculation. *Elsevier Journal on Simulation Modelling Practice and Theory*, Sep 2008.

[59] D. R. Kuhn, T. J. Walsh, and S. Fries. Security Considerations for Voice Over IP Systems. *National Institute of Standards and Technology, csrc.nist.gov/publications/*, 2005.

[60] L. A. Gajanan. *Financinal Forecasting comparison of ARIMA, FFNN and SVR Models.* PhD thesis, Indian Institute of Technology, Bombay, 2008.

[61] A. Lahmadi and O. Festor. SecSip : A Stateful Firewall for SIP-based Networks. In *Proc. of the 11th IFIP/IEEE International Symposium on Integrated Network Management(IM'09)*, Long Island, USA, June, 2009.

[62] A. Lahmadi and O. Festor. VeTo : An Exploit Prevention Language from Known Vulnerabilities in SIP Services. In *Proc. of the IEEE/IFIP Network Operations and Management Symposium (NOMS'10)*, pages 216–223, Osaka, Japan, April 2010.

[63] P. Laskov, C. Schäfer, and I. Kotenko. Intrusion Detection in Unlabeled Data with Quarter-sphere Support Vector Machines. In *Proc. of the 1st Conference on Detection of Intrusions and Malware and Vulnerability Assessment (DIMVA'04)*, pages 71–82, Dortmund, Germany, 2004.

[64] D. Liben-Nowell, H. Balakrishnan, and D. Karger. Analysis of the Evolution of Peer-to-Peer Systems. In *21st ACM Symposium on Principles of Distributed Computing (PODC'02)*, July 2002.

[65] E. K. Lua, J. Crowcroft, M. Pias, R. Sharma, and S. Lim. A Survey and Comparison of Peer-to-Peer Overlay Network Schemes. *IEEE Communications Surveys and Tutorials*, 7 :72–93, 2005.

[66] M. Benini and S. Sicari. Assessing the Risk of Intercepting VoIP Calls. *Elsevier Journal on Computer Networks*, 52(12) :2432–2446, May 2008.

[67] M. Nassar. *VoIP Networks Monitoring and Intrusion Detection.* PhD thesis, Universite de Lorraine, 2009.

[68] M. Nassar and O. Dabbebi and R. Badonnel and O. Festor. Risk Management in VoIP Architectures using Support Vector Machines. In *Proc. of 6th IFIP/IEEE International Conference on Network and Service Management (CNSM'10)*, October 2010.

[69] R. Macintosh and D. Vinokurov. Detection and Mitigation of Spam in IP Telephony Networks using Signaling Protocol Analysis. In *Proc. of the IEEE/Sarnoff Symposium on Advances in Wired and Wireless Communication (SAWWC'05)*, April 2005.

[70] G. Malan, D. Watson, F. Jahanian, and P. Howell. Transport and application protocol scrubbing. In *Proc. of the 19th Annual Joint Conference of the IEEE Computer and Communications Societies(INFOCOM'00)*, volume 3, pages 1381–1390, 2000.

[71] T. Mather, S. Kumaraswamy, and S. Latif. *Cloud Security and Privacy : An Enterprise Perspective on Risks and Compliance.* O'Reilly Media, Inc., 2009.

[72] B. Mathieu, S. Niccolini, and D. Sisalem. SDRS : A Voice-over-IP Spam Detection and Reaction System. *IEEE Security and Privacy*, 6(6) :52–59, 2008.

[73] M. Modarres. *Risk Analysis in Engineering.* Taylor & Francis, 2006.

[74] M. Nassar, R. State, and O. Festor. Monitoring SIP Traffic Using Support Vector Machines. In *Proc. of the 11th International Symposium on Recent Advances in Intrusion Detection (RAID'08)*, pages 311–330. Springer-Verlag, 2008.

[75] M. Nassar, R. State, and O. Festor. Voip Honeypot Architecture. In *Proc. of the IFIP/IEEE International Symposium on Integrated Network Managment (IM'07)*, pages 109–118. IEEE Communications Society, Munich, Allemagne, May, 2007.

[76] M. Nassar, R. State, and O. Festor. A Framework for Monitoring SIP Enterprise Networks. In *proc. of the 4th International Conference on Network and System Security, (NSS'10)*, pages 1–8, Victoria, Australia, 2010.

[77] National Institute of Standards and Technology (NIST). Guidelines on Security and Privacy in Public Cloud Computing. *Special Publication 800-144*, February 2011.

[78] O. Dabbebi and R. Badonnel and O. Festor. A Broad-Spectrum Strategy for Runtime Risk management in VoIP Entreprise Architectures. In *Proc. of the 12 IFIP/IEEE international Symposium on Integrated network Management (IM'11)*, May 2011.

[79] O. Dabbebi and R. Badonnel and O. Festor. Automated Runtime Risk Management for Voice over IP Networks and Services. In *Proc. of the 12th IEEE/IFIP Network Operations and Management Symposium (NOMS'10)*, pages 57–64, Osaka, Japan, April 2010.

[80] T. Olsson. Assessing Security Risk to a Network using a Statistical Model of Attacker Community Competence. In *Proc. of the 11th International Conference on Information and Communications Security (ICICS'09)*, pages 308–324. Springer-Verlag, 2009.

[81] P. Thermos and A. Takanen. *Securing VoIP Networks : Threats, Vulnerabilities, and Countermeasures.* Addison-Wesley Professional, 2007.

[82] D. Pechyony and V. Vapnik. On the Theory of Learnining with Privileged Information. In *Proc. of the 24th Annual Conference on Neural Information Processing Systems (NIPS'10)*, pages 1894–1902, British Columbia, Canada, December 2010.

[83] F. Pouget and M. Dacier. Honeypot-based Forensics. In *Proc. of the Asia pacific information technology security conference (AusCERT'04)*, Brisbane, Australia, May, 2004.

[84] V. M. Quinten, R. van de Meent, and A. Pras. Analysis of Techniques for Protection Against Spam over Internet Telephony . In *Proc. of 13th Open European Summer School EUNICE 2007*, July 2007.

[85] J. Quittek, S. Niccolini, S. Tartarelli, and Roman Schlegel. Prevention of Spam over IP Telephony (SPIT). In *NEC Technical Journal Vol.1 No.2*, 2006.

[86] J. Quittek, S. Niccolini, S. Tartarelli, M. Stiemerling, M. Brunner, and T. Ewald. Detecting SPIT Calls by checking human communication patterns. In *Proc. of the International Conference on Communications (ICC'07)*, June 2007.

[87] R. Dantu and P. Kolan and J. Cangussu. Network Risk Management using Attacker Profiling. *Security and Communication Networks*, 2(1) :83–96, 2009.

[88] R. Dantu and P. Kolan and J. W. Cangussu. Network Risk Management using Attacker Profiling. *Security and Communication Networks*, 2 :83–96, 2009.

[89] R. Schlegel and S. Niccolini and S. Tartarelli and M. Brunner. Spam over Internet Telephony (SPIT) Prevention Framework. In *Proc. of the IEEE Global Communications Conference (GLOBECOM'06), San Francisco, USA*, San Francisco, USA, November 2006.

[90] R. Vitalta and C. V. Apte and J. L. Hellerstein and S. Ma and S. M. Weiss. Predictive Algorithms in the Management of Computer Systems. *IBM System journal*, 41(3), 2003.

[91] T. Russell. *Session Initiation Protocol (SIP) : Controlling Convergent Networks*. McGraw-Hill Osborne Media, 1st edition, 2008.

[92] F. Sabahi. Cloud Computing Security Threats and Responses. *Computer Engineering*, pages 245–249, 2011.

[93] J. Seedorf. Security challenges for peer-to-peer sip. *IEEE Network*, 20(5) :38–45, 2006.

[94] J. Seedorf. Using Cryptographically Generated SIP-URIs to protect the Integrity of Content in P2P-SIP. In *Proc. of the 3rd Annual VoIP Security Workshop (VSW'06)*, Berlin, June 2006.

[95] H. Sengar, H. Wang, D. Wijesekera, and S. Jajodia. Detecting VoIP Floods Using the Hellinger Distance. *IEEE Trans. Parallel Distrib. Syst.*, 19(6) :794–805, 2008.

[96] H. Sengar, H. Wang, D. Wijesekera, and S. Jajodia. Detecting VoIP Floods Using the Hellinger Distance. *IEEE Trans. Parallel Distrib. Syst.*, 19(6) :794–805, 2008.

[97] S. Sengupta, V. S. Kaulgud, and V. S. Sharma. Cloud Computing Security-Trends and Research Directions. In *Proc. of the (SERVICES'11)*, 2011.

[98] A. Shamir. How to Share a Secret. volume 22, pages 612–613, November 1979.

[99] D. Shin and C. Shim. Progressive Multi Gray-Leveling : A Voice Spam Protection Algorithm. *IEEE Network Magazine*, 20 :18–24, Sep/Oct 2006.

[100] K. Singh and H. Schulzrinne. Peer-to-peer Internet Telephony Using SIP. In *Proc. of the International Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV'05)*, pages 63–68. ACM, 2005.

[101] K. Singh and H. Schulzrinne. SIPpeer : A SIP-based Peer-to-Peer Internet Telephony Adaptor. In *Proc. of the International Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV '05)*, pages 63–68, New York, NY, USA, 2005. ACM.

[102] K. N. Singh. *Reliable, Scalable and Interoperable Internet Telephony*. PhD thesis, 2006.

[103] D. Sisalem, J. Floroiu, J. Kuthan, U. Abend, and H. Schulzrinne. *SIP Security.* Wiley, 2009.

[104] H. Son and Y. Lee. Detecting Anomaly Traffic using Flow Data in the real VoIP network. In *Proc. of the 2010 10th IEEE/IPSJ International Symposium on Applications and the Internet (SAINT'10)*, pages 253–256, Washington, DC, USA, 2010. IEEE Computer Society.

[105] H. Song and M. Matuszewski. Security Requirements in Peer-to-Peer Session Initiation Protocol, IETF Internet Draft, tools.ietf.org/id/draft-matuszewski-p2psip-security-requirements-04.html, November 2008.

[106] H. Song, M. Matuszewski, and D. York. P2PSIP Security 0verview and Risk Analysis, IETF Internet Draft, tools.ietf.org/html/draft-matuszewski-p2psip-security-requirements-06, Jan 2010.

[107] G. Stoneburner and Alice G. Risk Management Guide for Information Technology Systems Recommendations of the National Institute of Standards and Technology. Technical report, USA/NIST.

[108] J. Tam, J. Simsa, S. Hyde, and L. von Ahn. Breaking audio captchas. In *NIPS*, pages 1625–1632, 2008.

[109] J. Tang and Y. Cheng. Quick Detection of Stealthy SIP Flooding Attacks in VoIP Networks. In *Proc. of the IEEE International Conference on Communications (ICC'11)*, pages 1–5, Kyoto, Japan, June 2011.

[110] A. Tripathi. Cloud Computing Security Considerations. *Interface*, (April) :1–18, 2011.

[111] V. Vapnik. Learning Hidden Information : SVM+. In *proc. of the IEEE International Conference on Granular Computing (GrC'06)*, page 22, Georgia, USA, May 2006.

[112] Voice over IP Security Alliance. VoIP Security and Privacy Threat Taxonomy. *www.voipsa.org/Activities/taxonomy.php*, October 2005.

[113] W. Allsopp. *Unthorised Access : Physical Penetration Testing for IT Security Teams.* John Wiley & Sons Ltd, 2009.

[114] Wang. *Support Vector Machines : Theory and Applications*, volume 177 of *Studies in Fuzziness and Soft Computing.* Springer Berlin, 2005.

[115] J. Wickboldt, L. Bianchin, R. Lunardi, F. Andreis, R. Luis dos Santos, B. Dalmazo, W. Cordeiro, A. Rabelo de Sousa, L. Zambenedetti Granville, L. Paschoal Gaspary, and C. Bartolini. Computer-Generated Comprehensive Risk Assessment for IT Project Management. In *Proc. of 20th IFIP/IEEE International Distributed Systems, Operations and Management Workshop (DSOM'09)*, October 2009.

[116] G. Zhang, S. Ehlert, T. Magedanz, and D. Sisalem. Denial of Service Attack and Prevention on SIP VoIP Infrastructures Using DNS Flooding. In *Proc. of the Principles, Systems and Applications of IP Telecommunications conference (IPTComm'07)*, 2007.

[117] X. Zheng and V. A. Oleshchuk. Trust-based Framework for Security Enhancement of P2PSIP Communication Systems. In *Proc. of the 4th International Conference for Internet Technology and Secured Transactions, (ICITST'09)*, London, England, 2009.

Glossaire

AGI Asterisk Gateway Interface. 136
AR AutoRegressive model. 81
ARMA AutoRegressive Moving Average. 80

CDR Call Details Record. 58

DHCP Dynamic Host Configuration Protocol. 3
DHT Distributed Hash Table. 21
DNS Domain Name System. 3
DoS Denial of Service. 4
DTLS Distributed Transport Layer Security. 41

IDS Intrusions Detection System. 38
IETF The Internet Engineering Task Force. 14
IP Internet Protocol. 30
IPS Intrusions Prevention System. 34
ISO Organisation Internationale de Normalisation. 30

MA Moving Average model. 81

NIST National institute of Standards and Technology. 116

OVAL Open Vulnerability and Assessment Language. 39

P2PSIP Pair to Pair Signaling Initiation Session. 5, 87
PKI Public Key Infrastructure. 42

RELOAD REsource LOcation And Discovery. 106
RSA Rivest Shamir Adleman. 95
RTC Réseau téléphonique commuté. 3
RTCP Real Time Control Protocol. 5, 19
RTP Real Time Protocol. 5, 17

S/MIME Secure/Multipurpose Internet Mail Extensions. 42
SDP Session Description Protocol. 5, 19

SIP Session Initiation Protocol. 5, 14

SPIT Spam over Internet Telephony. 4

SVM Support Vector Machines. 37

TLS Transport Layer Security. 41

UDP User Datagram Protocol. 14

URI Uniform Resource Identifier. 14

VoD Video on Demand. 151