Automatisation de tunnels IPSec imbriqués pour établir des communications anonymes

Hervé Aiache, Matteo Lauriano, Corinne Sieux, Cédric Tavernier

Automatisation de tunnels IPSec imbriqués pour établir des communications anonymes

Hervé AIACHE, Matteo LAURIANO, Corinne SIEUX et Cédric TAVERNIER

Thales Communications

160, bd de Valmy
92 704 Colombes Cedex
FRANCE
firstname.name}@fr.thalesgroup.com

RÉSUMÉ. *De nos jours, la sécurité et le respect de la vie privée sont devenues des challenges techniques critiques pour les courantes et futures générations de système de communication. Depuis les années 1980, beaucoup de systèmes définissent des solutions, principalement dérivées de l'approche bien connue des réseaux de MIX (développée par Chaum), visant à assurer l'anonymat des flux au niveau du réseau. Bien qu'efficaces, ces solutions souffrent d'un manque d'intégration d'approches IP standardisées et par conséquent, ne permettent pas une large adoption par le grand public. Ce papier propose une solution d'établissement de circuits anonymes, dérivée du très célèbre concept MIX, et bénéficiant du standard IPSec. Cette solution a été implémentée et expérimentée sur un vrai démonstrateur dans l'optique d'analyser les impacts de la solution sur la transmission de flux de bout en bout.*

ABSTRACT: *Nowadays, security and privacy are becoming two of the most critical issues for current and future generation of communications systems. Since the 80's, many efficient systems have been proposed to ensure flows anonymity, mainly derived from the so-called Chaum's Mix networks. However, these solutions suffer from a lack of integration with standardized IP approaches and therefore missed a wide adoption by the general public. This paper proposes an anonymous circuit establishment scheme derived from the powerful Mix networks concept and inheriting from the IPSec Framework. This solution has been implemented and experimented over a real testbed in view to analyze its impacts on multimedia flows end-to-end transmission.*

MOTS-CLÉS : : *Anonymous routing, Privacy, Traffic Flows Confidentiality, IPSec, Chaum's Mix*

CFIP 2007-2008, pages 1 à 14

1. Introduction

De nos jours, la sécurité et le secret de la vie privée sont devenus deux des plus importants problèmes des systèmes de communication présents et futurs. De nombreux fournisseurs de services demandent (et dans la plupart des cas collectent) des informations personnelles (identité, numéro de compte, mot de passe, lieu, préférences…) afin d'accéder et d'exploiter ces données pour le développement de services totalement adaptés à la personne (i.e. stockage numérique, services de location, ou accès au comptes bancaires. Des personnes malveillantes qui observent simplement le réseau, peuvent acquérir facilement des données sensibles et privées sur les usagers. Ces données peuvent être chiffrées, mais, la protection des informations privées requiert d'autres techniques qui permettent de masquer non seulement la source, la destination du trafic, le chemin utilisé et les données mais aussi le type de trafic que l'usager génère, et plus généralement leur comportement au sein du réseau.
Depuis les années 80, de nombreux systèmes efficaces (e.g. [4], [5], [3]), principalement dérivés des solutions basées sur le concept de Mix [4], permettant d'assurer l'anonymat des flux de communication ont été proposés pour permettre de résoudre ces aspects de protection de le vie privée. Cependant, elles souffrent d'un manque d'intégration basées sur des approches IP standards. Ceci peut expliquer le fait que ces solutions n'aient pas été largement acceptées du grand public.
Ainsi, ce papier présente une solution d'établissement de circuit anonymes dérivée du célèbre concept Mix, et héritant du cadre d'IPSec. Ce papier est structuré en trois parties : tout d'abord la section 2 présente pour la première fois une vue fonctionnelle, décrivant la plupart des solutions de routage anonymes existantes (e.g. Tarzan[5], Mix[4], OR[6], TOR[3], MorphMix[8]), la section 3 détaille la solution de circuit anonyme basée sur des tunnels IPSec imbriqués, la section 4 rapporte les résultats obtenus lors d'une expérimentation réelle, analysant les impacts des algorithmes de chiffrement sur des flux multimédia de bout en bout. La conclusion résume les différents aspects et présente les travaux futurs.

2. Vue fonctionnelle des approches de routage anonyme

Deux groupes de solution de routage anonyme existent : les architectures peer-to-peer et celles basées sur le concept « d'oignon ». Elles sont toutes les deux principalement basées sur une approche de nœud Mix, mais développées et distribuées dans un environnement dynamique pour la première catégorie (Tarzan, Crowds[7], MorphMix), et dans un environnement plus centralisé et fixe pour les autres (Chaum's Mix-net, mixMaster, Web-Mixes, TOR). L'analyse fonctionnelle des ces différentes approches de systèmes anonymes montre que la plupart des solutions de routage anonyme se basent sur quatre principaux composants et sur un bloc définissant des politiques de transmission.

Deux de ces composants (sections 2.1 et 2.2) sont plus liés aux aspects de routage (découverte de nœud, établissement de topologie, validation de confiance), et les trois autres (sections 2.3 à 2.5) sont plus liés au plan de retransmission (i.e. notion liées à la fonctionnalité de « forwarding »). Tous ces blocs fonctionnels sont illustrés à la figure 1 et sont décrits en détail dans la suite.

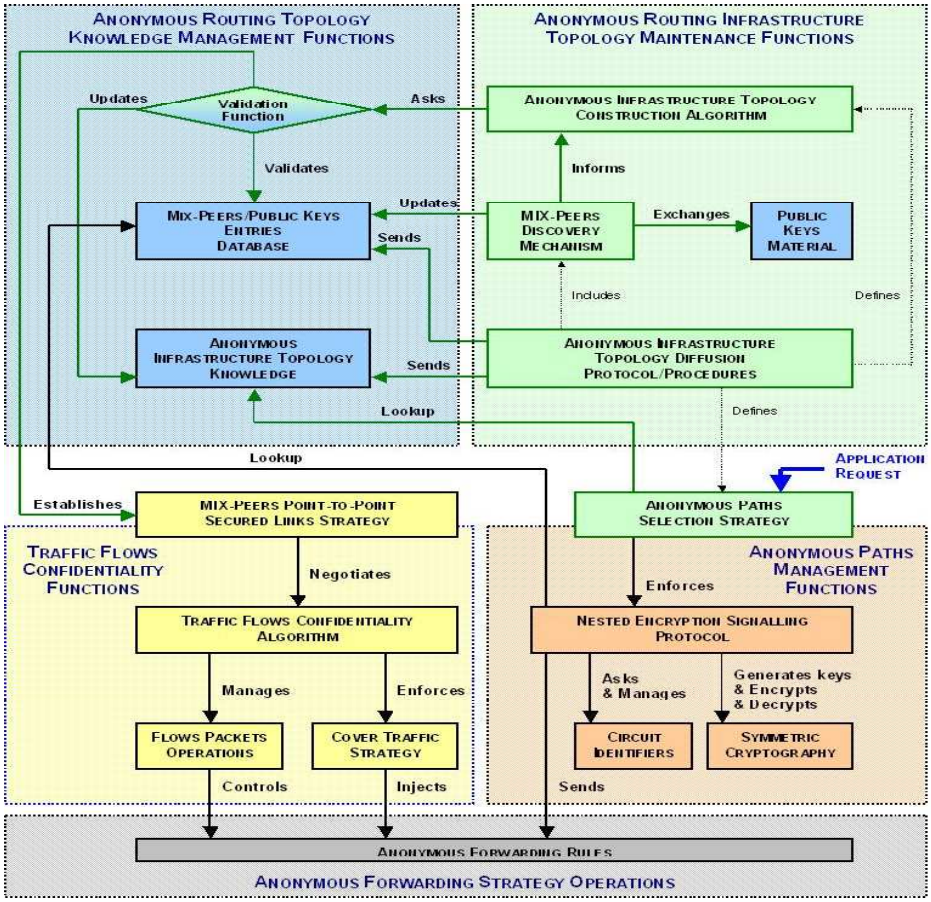

Figure 1. *Vue fonctionnelle des composants permettant une définition complète des solutions de routage anonyme*

2.1. *Maintenance de la topologie de routage anonyme (composant Anonymous Routing Infrastructure Topology Maintenance)*

Ce composant regroupe toutes les fonctions nécessaires à la gestion d'entrée et de sortie des nœuds Mix au sein de l'infrastructure de réseau anonyme. Remarquons que, en fonction du type de routage anonyme, ces opérations peuvent être opérées statiquement, par le biais de procédures (ou par un rafraîchissement basé sur un serveur central – e.g. TOR[3]) ou dynamiquement comme pour les solutions [2, 5], reposant sur des protocoles distribués spécifiques.

L'ensemble de fonctions implémentant la maintenance des topologies des infrastructures de routage anonyme définit dans la plupart des cas une construction de topologie particulière (e.g. CHORD, CAN ou Gossip pour P2P) afin de mieux contrôler l'infrastructure. Principalement associée aux fonctions de topologie, une stratégie particulière pour le choix du chemin anonyme est implémentée afin de choisir au mieux les nœuds Mix qui vont le composer.

2.2. Gestion de la connaissance de la topologie de routage anonyme (composant Anonymous Routing Topology Knowledge Management)

Ce composant intègre toutes les structures d'information (e.g. tables, bases de données) qui sont nécessaires pour savoir comment l'infrastructure est définie et quels sont les paramètres de chiffrement utilisés (principalement les clefs publiques).

Ces tables sont initiées et remplies durant la découverte des nœuds Mix et mises à jour régulièrement (et automatiquement) par le biais de protocoles de routages anonymes ou statiquement par le biais de procédures. Toutes ces informations vont être utilisées ensuite pour établir les circuits anonymes, que nous décrirons plus tard.

2.3. Composant TFC (Traffic Flows Confidentiality)

Ce composant permet d'auto-configurer les opérations permettant de masquer certaines caractéristiques de trafic des flux. Il repose principalement sur deux sous-fonctionnalités : la modification des paquets (i.e. ajout de données et opération de temps) et la gestion de « faux » paquets.

Il résout les attaques basées sur la corrélation des données analysées dans le temps. Dans la plupart des approches, ces problèmes peuvent être résolus par la définition d'algorithme particulier de TFC.

2.4. Gestion du chemin anonyme (Anonymous Paths Management)

Ce composant (une variante de l'approche basé sur le concept de Mix développé par Chaum) protége les identités de la source et/ou de la destination (i.e. les identifiants ou adresses des nœud Mix). Principalement, cette fonction basique permet d'installer un chemin anonyme basé sur une stratégie de sélection des nœuds Mix (spécifiés par le routage anonyme ou par le biais de procédures manuelles). Un message de signalisation basée sur le concept « d'oignon » (e.g. TOR[3], [6]) afin d'établir un chemin anonyme au sein de l'infrastructure anonyme en overlay. De plus, la gestion des chemins anonymes est articulée autour d'une stratégie particulière de chiffrement imbriqué (i.e. couches successives de chiffrement), qui est dans notre cas élaboré autour d'algorithmes de chiffrement symétrique.

2.5. Règles d'anonymat de la transmission (Anonymous Forwarding Rules)

Ces règles regroupent un ensemble de politiques nécessaire au niveau de l'envoi de la trame afin de garantir l'anonymat des flux de communication. Ceci est

nécessaire car les autres composants ne peuvent pas masquer toutes les informations nécessaires pour assurer proprement les règles de transmission. (i.e. les identifiants ou adresses des nœuds Mix).

3 Spécification et implémentation de la solution

Après quinze ans de recherche et de développements, les approches obtenues, bien qu'efficaces, n'ont pas été réellement intégrées au sein des infrastructures Internet. Ceci est principalement dû au fait que les solutions n'ont pas été élaborées autour des standards de sécurité basés sur IP[1]. Ainsi, pour combler ce manque, ce papier spécifie et définie une solution d'établissement de circuit anonymes chiffrés nativement inscrit au sein du standard IPSec, et une bibliothèque logicielle pour facilement l'implémenter. Cette solution répond aux attaques passives par analyse de trafic et ne permet pas de faire le lien entre la source et la destination (premier et dernier MIX). Dans notre approche fonctionnelle, cette solution correspond au composant de gestion de chemin anonyme (APSM : *Anonymous Path Signalling Manager*).

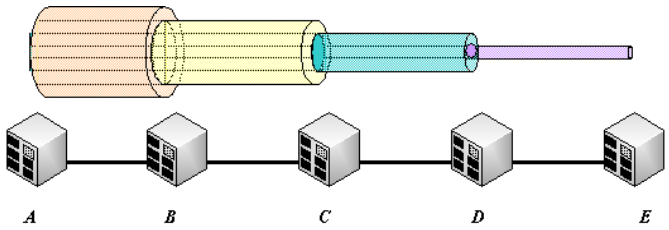

Figure 2. *Technique de tunnel imbriqués pour l'établissement d'un chemin anonyme*

Cette approche est complémentaire du standard RFC 4303 définit par l'IETF, qui propose un possible anonymat du trafic en dé-corrélant le trafic utile pour les attaques de temps (TFC). Un exemple illustrant le chemin anonyme établi, une fois les messages de signalisations échangés, est montré à la figure 2.

3.1. Spécification détaillée du logiciel

La gestion d'un chemin anonyme définie par le composant APSM peut être décomposée en deux opérations distinctes, dépendant de la position du nœud Mix considéré.

En effet, le nœud Mix peut initier l'établissement (ou le retrait d'un circuit anonyme (i.e. en étant le nœud source initiateur ou premier Mix du chemin), ou le nœud Mix peut être un nœud intermédiaire au sein du chemin anonyme sélectionné,

et dans ce cas, doit contribuer à l'établissement (ou au retrait local) de la connexion anonyme.

Ainsi, la distinction est importante afin de comprendre comment les composants logiciels (illustrés par la figure 3), vont agir différemment pour implémenter ces opérations .

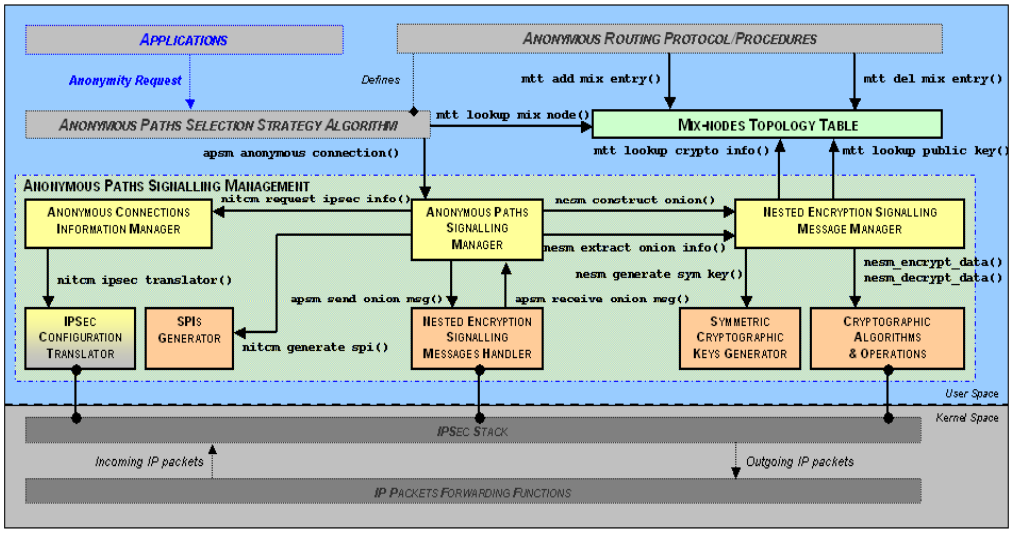

Figure 3. Détails des sous-composants de l'APSM

Dans le cas d'établissement du chemin anonyme, le nœud Mix initiateur va devoir générer les différents mécanismes de chiffrement pour établir le chemin. Dans le cas d'un nœud intermédiaire, le nœud reçoit un message de signalisation dérivé du concept « d'oignon », dont il va retirer une première couche, appliquer la configuration de tunnel IPSec correspondante et transmettre l'oignon de signalisation ainsi pelé, au nœud Mix suivant. Ces deux différentes phases sont expliquées dans les sections suivantes. L'utilisation de solution standardisée (IPSec) permet une intégration plus facile dans les infrastructures de réseau existantes.

3.2. Détails des opérations

3.2.1 *Opération d' établissement d'un chemin anonyme* (Nœud initiateur)

Après avoir reçu une demande d'établissement d'anonymat initié par une application, l'APSM (*Anonymous Paths Selection Manager*) demande l'ensemble des nœuds élus (déjà fait par le composant de topologie) pour l'établissement du chemin en appelant la fonction `apsm_anonymous_connection()`. Les adresses IP des nœuds Mix sélectionnés sont fournies comme paramètres d'entrée de cette fonction. Notons que l'ordre des adresses IP est important car le circuit anonyme va être établi en contactant un par un les nœuds correspondants dans l'ordre de départ indiqué (l'imbrication du chiffrement se fera aussi dans le même ordre).

Ensuite, le composant APSM demande la génération de plusieurs SPIs, dépendant du nombre de nœuds Mix impliqués en appelant la fonction `nitcm_generate_spi()` du sous-composant *SPIs generator*. Après cette opération, l'APSM demande la génération du message de signalisation imbriquée : ceci est fait par appel à la fonction `nesm_construct_onion()`du sous-composant *Nested Encryption Signalling Message Manager*.

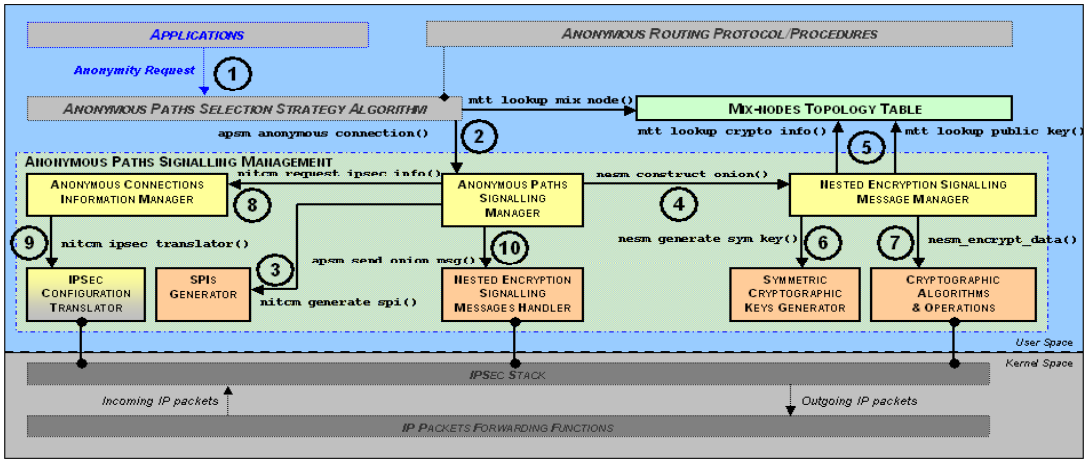

Figure 4. *Ordre d'appel des fonction de l'APSM pour l'établissement du chemin anonyme au sein du premier nœud Mix*

La liste des adresses IP des nœuds Mix et les SPIs précédemment générés sont donnés en paramètre en entrée de cette fonction.

Dans le but de générer le message imbriqué ("onion like"), le sous-composant *Nested Encryption Signalling Message Manager* récupère les clefs publiques (et les algorithmes de chiffrement associés) correspondant à la liste de nœuds Mix, en appelant la fonction mtt_lookup_public_key() (et la fonction mtt_lookup_crypto_info()). Ensuite, le sous-composant Nested Encryption Signalling Message Manager demande la génération de clefs symétriques en appelant la fonction nesm_generate_sym_key() du sous composant *Symmetric Cryptographic Key Generator*. Le message de signalisation, qui va être envoyé de noeud en nœud, avec une couche en moins à chaque nœud, est à présent construit, par appel à la fonction `nesm_encrypt_data()` du sous-composant *Cryptographic algorithms and operations*. L' APSM informe le sous-composant de gestion des circuits anonymes (*Anonymous Connections Information Manager)* par appel à la fonction `nitcm_request_ipsec_info()`. Les informations sont mémorisées localement, les politiques correspondant à la configuration de tunnels IPSec sont appliquées par appel à la fonction `nitcm_ipsec_translator()` du sous-composant IPSec configuration Translator.

Une fois les configurations appliquées, l'APSM envoie le message de signalisation au Mix suivant en appelant la fonction `apsm_send_onion_msg()`.

La figure 4 illustre l'ordre dans lequel les sous-composants opèrent entre eux.

3.2.2 Traitement d'un message de signalisation pour l'établissement du chemin anonyme (nœud intermédiaire)

Quand l'APSM reçoit un message depuis la signalisation, il arrive via le sous-composant *Nested Encryption Signalling Message Handler* au travers de la fonction `apsm_receive_onion_msg()`.

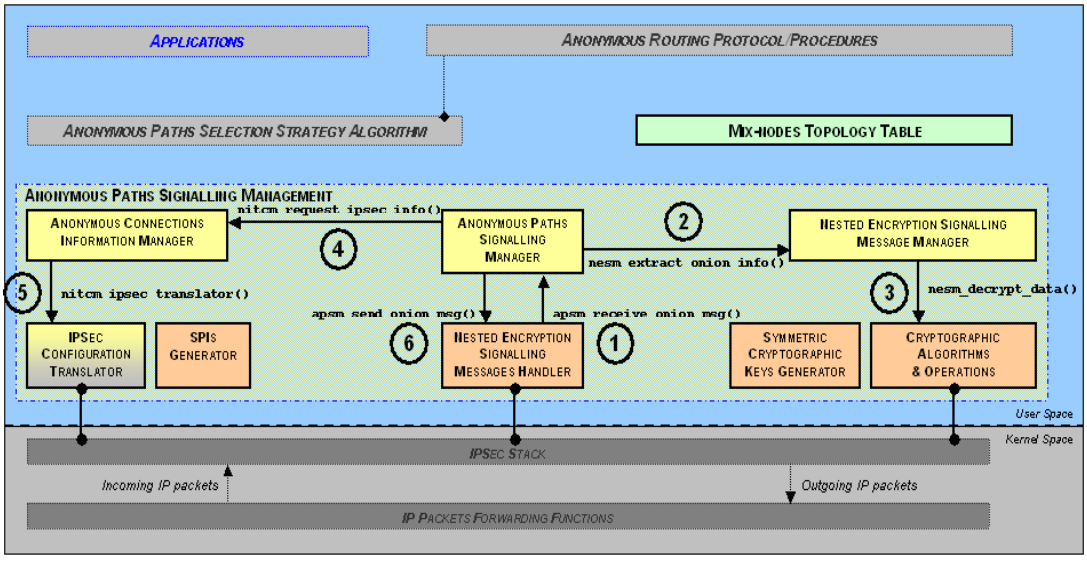

Figure 5. *Ordre d'appel des fonctions lors de l'établissement d'un chemin anonyme au sein d'un nœud Mix intermédiaire*

Une fois reçu, l'APSM retire une couche de chiffrement au message de signalisation en utilisant la fonction `nesm_extract_onion_info()` du sous-composant *Nested Encryption Signalling Message Manager*. Ensuite, ce sous-composant extrait les éléments de chiffrement au travers de la fonction `nesm_decrypt_data()` offerte par le sous-composant Cryptographic Algorithms and Operations en utilisant la clef privée du nœud Mix. Ensuite les informations concernant la configuration du tunnel IPSec (la clef symétrique, le SPIs et le prochain nœud du chemin anonyme) et celles concernant le message de signalisation imbriquée avec une couche en moins (ce message n'est pas déchiffrable car il est chiffré avec les clefs des nœuds suivants) sont séparées. Les informations concernant le tunnel sont remontées en paramètre de retour à la fonction `nesm_extract_onion_info()`. Le message pelé peut lui être directement envoyé au prochain Mix. L'APSM demande au sous-composant *Anonymous Connections Information Manager* d'appliquer les nouvelles demandes de circuit anonymes en appelant la fonction `nitcm_request_ipsec_info()`.

Comme précédemment, le sous-composant *Anonymous Connections Information Manager* crée et mémorise un nouvel état pour cette connexion anonyme. Il

demande au sous-composant *IPSec Configuration Manager* d'appliquer les différentes configurations ou commandes des tunnels IPSec en appelant la fonction `nitcm_ipsec_translator()`. Une fois informé de la mise en place du tunnel, l'APSM envoie le message de signalisation pelé d'une couche au prochain nœud Mix en appelant la fonction `apsm_send_onion_msg()`du sous-composant *Nested Encryption Signalling Message Handler.*

La figure 5 illustre l'ordre dans lequel les sous-composants opèrent entre eux.

4 Experimentations

Cette section décrit l'environnement dans lequel les mesures ont été réalisées. La plate-forme utilisée pour collecter les données est composé de 5 PCs, comme indiqué à la figure 6.

Les cinq PCs tournent sur un environnement Linux 2.6 et sont connectés en chaîne. Pour ce faire, quatre sous-réseaux sont définis. Trois d'entre eux supportent 100Mbps, le quatrième 10Mbps. Le but est de démontrer qu'il est possible de rendre anonyme et de sécuriser des communications pour du trafic temps réel.

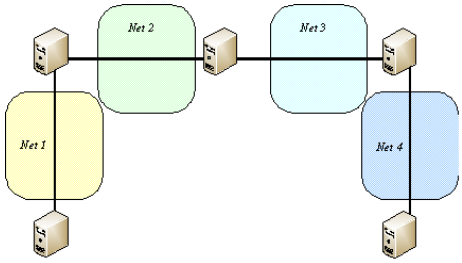

Figure 6. Plate-forme expérimentale

Dans cette section, les résultats des mesures sont présentés. Afin de simplifier la présentation, deux sous-sections sont définies. La première concerne les mesures avec ICMP, la deuxième celles réalisées avec UDP.

Cinq configurations ont été utilisées : sans tunnels, avec un, deux, trois, ou quatre tunnels. Deux différents algorithmes de chiffrement ont été utilisés : 3DES CBC et Rijndaël CBC avec ces trois longueurs de clefs différentes, i.e. 128, 192, et 256 bits. Pour chaque configuration, des mesures ont été faites pour des paquets dont la taille varie entre 64, 128, 256, 512, 1024, 2048 et 4096 octets.

4.1 Mesures avec ICMP

Le RTT (Round Trip Time) ou délai d'aller-retour dépendant de la taille des paquets pour un algorithme de chiffrement est présenté. Dans chaque figure, il y a

cinq courbes. Quatre d'entre-elles représentent les quatre configurations précédemment décrites (de 1 à 4 tunnels), la dernière représente le cas sans aucun tunnel (qui est le cas de référence).

Généralement, l'algorithme 3DES présente de mauvaise performances, comme indiqué par la figure 7. Si la taille des paquets est trop large, i.e. inférieur ou égal à 256 octets, il n'y a pas de grande différence dans le RTT, entre les configurations avec 1 ou 2 tunnels. Des différences significatives apparaissent pour des paquets de 1024 octets. Dans ce cas, la configuration avec 4 tunnels présente un RTT de 50% plus haut que le cas sans tunnel.

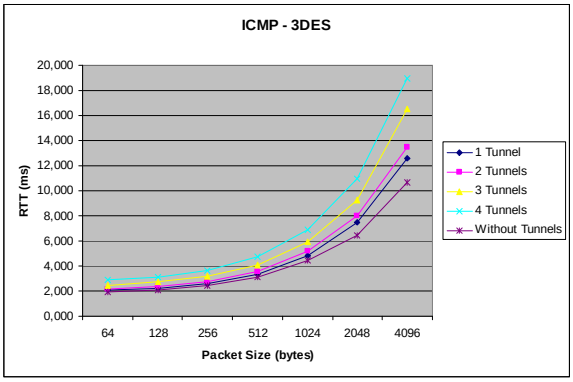

Figure 7. *RTT en fonction des paquets utilisant 3DES*

Les figures 8, 9 et 10 présente le RTT en utilisant l'algorithme de chiffrement de Rijndaël, avec respectivement des clés de 128, 192 et 256 bits. Il est important de souligner qu'entre les courbes il n'y a pas de grandes différences. Rijndaël 128 est plus rapide, mais les différences sont inférieures à 0,3 ms.

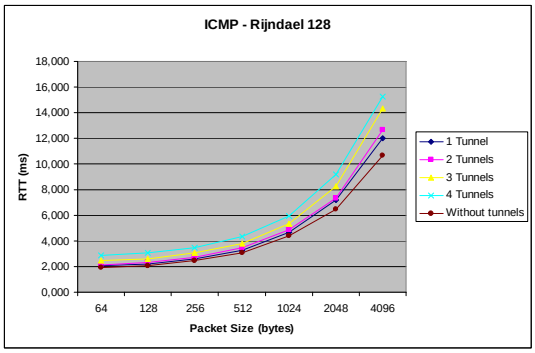

Figure 8. *RTT en fonction de la taille des paquets Rijndael 128.*

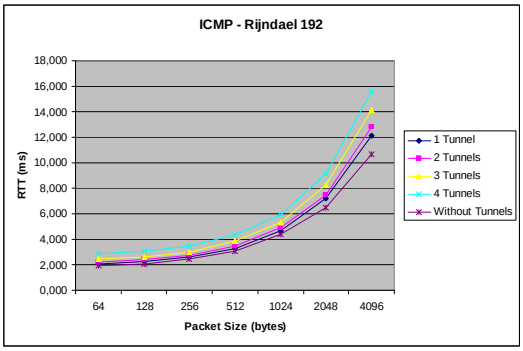

Figure 9. *RTT en fonction de la taille des paquets utilisant Rijndael 192.*

Le résultat est intéressant car cela signifie qu'il est possible d'incrémenter la sécurité de la transmission en utilisant une clé plus longue, avec un bon niveau de performance. Si la taille des paquets est trop large, i.e. égale ou supérieure à 1024 octets, de bonnes valeurs sont atteintes.

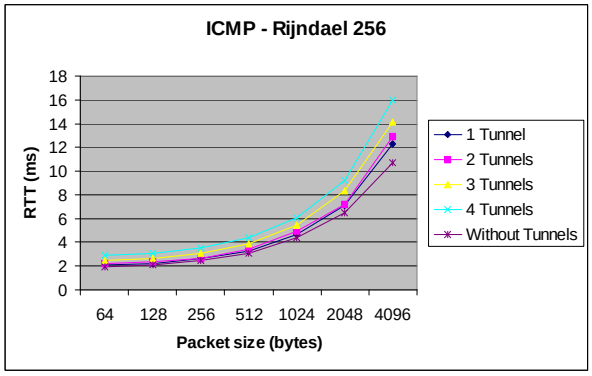

Figure 10. RTT en fonction de la taille des paquets utilisant Rijndael 256

4.2 Mesures avec UDP

Cette sous-section présente des mesures de gigue. Ces mesures sont évaluées pour un algorithme de chiffrement fixe, et de même que précédemment, on fait varier le nombre de tunnels. Pour chaque figure, différentes courbes sont disponibles, le paramètre étant la taille des paquets. La connaissance de la gigue est intéressante car la plus haute valeur ou variation de cette valeur peut déterminer des difficultés à transmettre des services temps-réel.

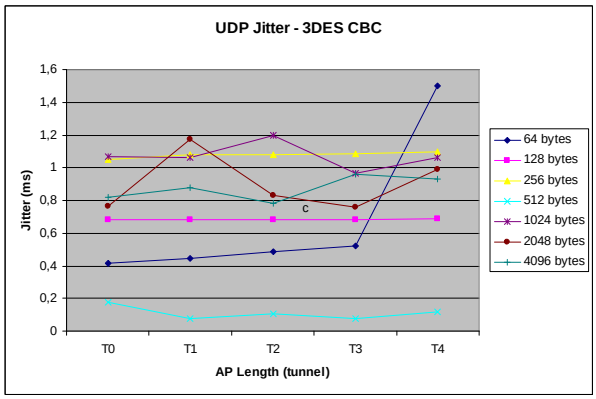

Figure 11. Gigue en fonction de la longueur du chemin anonyme en utilisant 3DES

La figure 11 montre la gigue mesurée quand l'algorithme 3DES est utilisé. 3DES n'est pas l'algorithme le plus mauvais, mais un comportement similaire est observé avec d'autres algorithmes. Les paquets de 128, 256 ou 512 octets ont le meilleur comportement car ils sont approximativement constants quand le nombre de tunnels change.

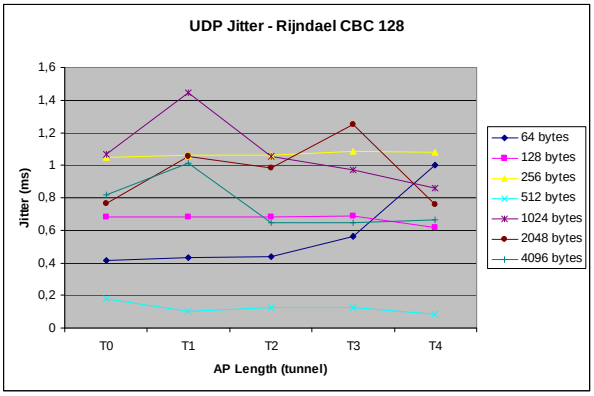

Figure 12. *Gigue en fonction de la longueur du chemin anonyme en utilisant Rijndael 128*

Le même comportement peut être observé sur les figures 12, 13 et 14. Les paquets de 512 octets ont la meilleure performance ; si 3DES ou Rijndaël 128 sont utilisés, le comportement est le plus régulier. Dans chaque cas, les paquets de 64 octets atteignent une haute valeur quand trois ou quatre tunnels sont utilisés. Les courbes pour des paquets de 1024, 2048 ou 4096 octets sont irrégulières. Généralement, tout d'abord les valeurs de gigues augmentent puis diminuent quand le nombre de tunnels augmentent.

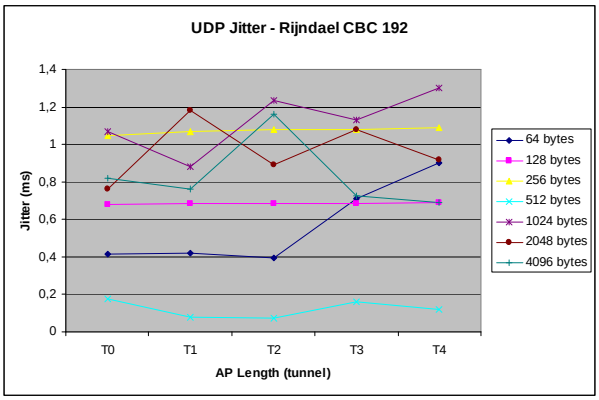

Figure 13. *Gigue en fonction de la longueur du chemin anonyme en utilisant Rijndael 192*

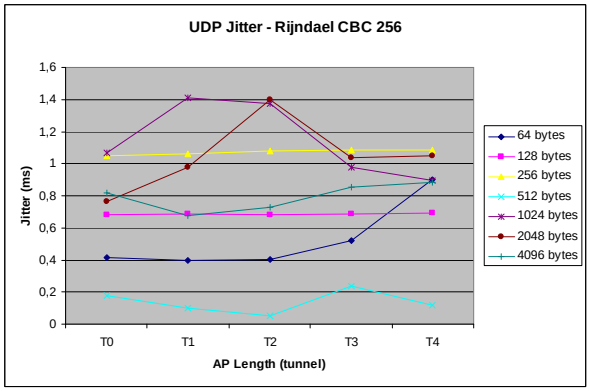

Figure 14. *Gigue en fonction de la longueur du chemin anonyme en utilisant Rijndael 256*

Les plus hautes valeurs sont atteintes pour des paquets de 1024 octets, quand Rijndaël est utilisé et quelque soit la longueur de la clé.

5. Conclusion et travaux futurs

Ce papier s'est intéressé et a présenté les principaux problèmes critiques que les réseaux IP doivent contrer aujourd'hui : la sécurité et la protection de la vie privée. Tout d'abord, ce papier a extrait un model fonctionnel qui s'applique à la majorité des solutions de routage anonymes présentées dans la littérature.

Nous avons ensuite présenté une solution basée sur des techniques de tunnels IPSec imbriqués afin d'assurer la non-traçabilité et non-observation des flux. La particularité de cette approche consiste principalement à la gestion de l'établissement de circuits anonymes dans un cadre standardisé. Un tel

environnement constitue un avantage certain pour un large déploiement ainsi que pour une adoption du grand public des solutions de routage anonyme.

De plus cette solution a été implémentée et testée sur une plate-forme expérimentale réelle afin de caractériser l'impact sur les flux multimédias. Nous avons pu observer que cette solution pouvait s'adapter à des applications nécessitant de vrais contraintes de temps réel .

Les travaux futurs consisteraient à inclure dans les performances et dans les améliorations déjà réalisées une implémentation de stratégie de découpage de message. De cette façon, les paquets IP envoyés par une source, seraient transmis à la même destination mais en utilisant des routes différentes afin de rendre plus difficiles les attaques passives. Dans ce cas, le nombre de tunnels IPSec et la longueur des circuits anonymes pourrait être diminués si de multiples routes sont utilisées. Ainsi, la performance globale serait améliorée, tout en conservant un niveau équivalent de protection et de sécurité. En d'autres mots, cette technique permettrait d'améliorer les performances de bout-en-bout en réduisant le nombre d'opérations de chiffrement et de déchiffrement. Les tests doivent aussi êtres réalisés en utilisant des applications audio et vidéo.

Nous souhaiterions remercier l'ensemble des partenaires du projet FP6 IST DISCREET, projet dans lequel nous avons défini cette solution et où nous testons en ce moment l'interaction avec le composant TFC identifié dans le model fonctionnel et le composant APSM. Pour plus d'information, merci de vous référer au site http://www.ist-discreet.org/.

7. Bibliographie et références

[1] S. Kent, "*IP Encapsulating Security Payload (ESP)",* IETF RFC 4303, December 2005.

[2] M. J. Freedman, R. Morris, "Tarzan: A Peer-to-Peer Anonymizing Network Layer", *Proceedings of the 9th ACM Conference on Computer and Communications Security (CCS 2002)*, 2002.

[3] R. Dingledine, N. Mathewson, P. Syverson, "TOR: The Second-Generation Onion Router", *Proceedings of 13th Usenix Security Simposyum* , August 2004.

[4] D. Chaum, "*Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms",* Communications of the ACM, Vol. 24 Number 2, February 1981.

[5] M. J. Freedman, E. Sit, J. Cates, R. Morris, "*Introducing Tarzan, a Peer-to-Peer Anonymizing Network Layer",* Proceedings of the 1st International Workshop on Peer-to-Peer Systems (IPTPS02), 2002.

[6] D. M. Goldschlag, M. G. Reed, P. F. Syverson, "*Hiding Routing Information"*, Workshop on Information Hiding, Cambridge, UK, May 1996.

[7] M. K. Reiter, A. D. Rubin, "*Crowds: Anonymity for web transactions"*, ACM TISSEC, June 1998.

[8] M. Rennhard, B. Plattner, "Introducing MorphMix: Peer-to-Peer based Anonymous Internet Usage with Collusion Detection", 2002.