Mesures et Caractérisation du Trafic dans le Réseau National Universitaire (RNU)

Khadija Ramah Houerbi

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR
UNIVERSITE DE MANOUBA
ECOLE NATIONALE DES SCIENCES DE L'INFORMATIQUE

THÈSE

Présentée en vue de l'obtention du diplôme de

DOCTEUR EN INFORMATIQUE

par

Khadija RAMAH HOUERBI

Mesures et Caractérisation du Trafic dans le Réseau National Universitaire (RNU)

Sous la direction du

Professeur Émérite Farouk KAMOUN

Réalisée au sein de

Soutenue le : 31 octobre 2009

Devant le jury composé de :

Président :	Professeur Émérite Mohamed Ben Ahmed
Rapporteur :	Professeur Serge Fdida
Rapporteur :	Maître de conférences Nejib Ben Hadj Alouane
Examinateur :	Professeur Moncef Tajina
Directeur de thèse :	Professeur Émérite Farouk Kamoun

Remerciements

J’aimerai, tout d’abord, remercier vivement mon directeur de thèse, Professeur Émérite Farouk Kamoun, pour m’avoir accueilli au sein du laboratoire Cristal, guidé mes travaux tout au long de ces années et m’apporté bien plus qu’un encadrement scientifique. Il a su par ses gestes d’encouragement, ses conseils et ses remarques judicieuses me donner confiance et me pousser à m’améliorer.

Mes remerciements s’adressent également à M. Kavé SALAMATIAN, Professeur à l’Université de Lancaster, pour avoir orienté mes travaux vers de nouvelles pistes. Qu’il puisse trouver ici l’expression de ma profonde reconnaissance notamment pour les innombrables discussions techniques que nous avons eu sur Skype.

Je tiens à remercier vivement Monsieur Serge Fdida Professeur à l’Université Pierre et Marie Curie (France) et Monsieur Néjib Hadj Alouane Maître de conférence à l’ENSI, pour l’honneur qu’ils me font en acceptant d’être les rapporteurs de cette thèse. Leurs commentaires, remarques et corrections m’ont permis d’améliorer la qualité du présent rapport.

Toute ma gratitude va également au Professeur Émérite Mohamed Ben Ahmed pour avoir accepter de présider le jury de ma thèse et m’avoir prodiguer conseils et corrections ainsi qu’au Professeur Moncef Tajina examinateur du présent travail.

Je suis particulièrement redevable à Mme Henda BenGhezala, Professeur à l’ENSI et Présidente de l’Université de Manouba, pour m’avoir encouragé à entreprendre ce travail de thèse, financé ma première participation à une conférence scientifique du domaine (PAM2004) et n’avoir jamais cessé de m’apporter son soutien après avoir quitté le Centre de Calcul el Khawarizmi.

Ce travail n’aurait pu être mené à bien sans le soutien de Mme Leila Saidane, Professeur et Directrice de l’ENSI qui a toujours trouvé le temps de m’écouter et de me conseiller.

Je ne saurai oublier ici les employés du Centre de Calcul el Khawarizmi que je tiens à remercier un à un pour m’avoir apporté tout le soutien matériel et logistique nécessaires à la collecte des traces de trafic. Mes remerciements s’adressent à cet effet, aux trois directeurs que j’ai côtoyé, M. Kamel BenRhouma, Mme Henda Benghazela et M. Mohamed Jemni ainsi qu’à mes anciens collègues et amis du CCK Henda, Sami, Abdelkarim, Souad, Sonia, SoniaBh ; Taeib, Mohamed, Zoubeida,….

De même, ma reconnaissance s’adresse à mes compagnons de route: mes amis et collègues que j’ai côtoyés au Laboratoire Cristal à l’ENSI ou au Laboratoire LIP6 à Paris. Je les remercie vivement pour la bonne ambiance de travail et la solidarité qu’ils m’avaient témoignée. Je remercie en particulier Hela Boucetta sans qui le déploiement de la plate-forme de mesures et l’analyse des traces aurait été beaucoup plus difficile, Hichem Ayari pour la qualité de son travail, son aide

précieuse et ses qualités humaines et Zhani Mohamed Faten pour sa maîtrise des modèles de prévision du trafic.

Enfin, ces remerciements ne peuvent s’achever, sans une pensée particulière pour mes parents, mon mari et mes enfants qui ont été soumis à rude épreuve lors de l’accompagnement de mes travaux et qui ont toujours su trouver les mots justes pour m’encourager.

Un grand merci à tous ceux qui m’ont aidé ou soutenu et dont j’ai omis de citer les noms !

Je dédie ce travail avec reconnaissance et amour

A mes chers parents à qui je dois tout
et qui par leurs prières ont toujours veillé sur moi

A mes adorables enfants Nadia et Alaa
qui ont toujours égayé ma vie

A mon cher époux Rafaa
pour sa gentillesse et sa compréhension

A ma sœur aînée Soumaya
dont j'ai toujours admiré le courage

A mes frères Helmi et Anes
qui sont toujours dans mon cœur

A mes sœurs bien aimées Samia et Inès

A tous ceux qui me sont chers

Résumé

Cette thèse porte sur la mesure et la caractérisation du trafic dans le Réseau National Universitaire (RNU) avec des applications à la détection des anomalies. Pour ce faire, une sonde de mesures passives a été déployée sur le réseau RNU. Plusieurs traces de trafic ont été collectées et analysées, exposant ainsi les caractéristiques du trafic RNU en le comparant avec les résultats de l'état de l'art.

Par la suite, deux approches de détection d'anomalies basées sur des algorithmes non paramétriques et utilisant des données faciles à collecter sont proposées. La première considère comme anomalie, tout point excentrique dans les séries temporelles de métriques de volume et utilise l'analyse en composantes principales et la distance de Mahalanobis. Sa validation, face à une trace préalablement étiquetée, a montré que l'approche proposée est efficace face aux attaques de dénis de service et plus généralement les anomalies de forte intensité. La seconde traque les variations affectant les distributions du trafic de scan dans l'espace des adresses IP et des numéros de ports visités. Son évaluation face à des traces de trafic réelles et des traces artificiellement modifiées a montré que la divergence de Kullback-Leibler de la distribution conjointe permet d'exposer la présence de tous les scans aussi bien horizontaux que verticaux.

Mots clés : Métrologie, Détection d'anomalies, Anomalie de volume, Balayage de ports, Déni de service, Réseau National Universitaire.

Abstract

This thesis focuses on measurement and characterization of traffic in Tunisian National University Network (RNU) with applications to traffic anomaly detection. For this, a passive measurement platform was deployed on RNU network and several traffic traces were collected, then these traces were analysed revealing traffic properties in RNU network with comparison to state of the art results.

Thereafter, two anomaly detection approaches based on nonparametric algorithms and easy-to-collect data are proposed. The first one marks as anomaly each eccentric points in traffic volume time series using principal component analysis and Mahalanobis distance. Its validation against a previously labelled trace showed that it is effective against Denial of Service attacks and more generally high intensity anomalies. The second proposed approach tracks changes affecting scanning traffic distributions over the space spanned by IP addresses and TCP port numbers. Its evaluation against reel traffic traces and artificially modified ones showed that the Kullback-Leibler divergence of the joint distribution exposes both the presence of horizontal scans and vertical ones.

Keywords : Metrology, Anomaly detection, Volume anomaly, Port scanning, Deny of service attack, Tunisian National University Network.

تلخيص

تهتم هذه الأطروحة بقياس و بيان خصائص حركة مرور المعلومات في الشبكة الوطنية الجامعية و تطرح تطبيقات للكشف عن التسللات والتصرفات المريبة. للقيام بذلك بدأنا بتركيز جهاز استشعار سلبي لجمع البيانات حول حركة مرور المعلومات ثم حللناها و قارننا خصائص حركة المرور في الشبكة الوطنية الجامعية بنتائج الأبحاث العلمية المماثلة.

إثر ذلك طرحنا منهاجين للكشف عن الحالات الشاذة اعتمادا علي بيانات يسهل استقاؤها من الشبكة. يعتمد الأول على المعطيات الخاصة بحجم المعلومات ويستعمل أسلوب تحليل البيانات إلى عناصرها الأساسية و مسافة ماهنوبيس . لتقيم هذا المنهاج استعملنا معطيات مبوبة فتبين أن المنهاج المقترح فعال ضد هجمات محاولات تعطيل الخدمة ، وبشكل أعم ضد أنواع الهجمات التي تولد حركة مرور كثيفة. أما المنهاج الثاني فيهتم بكشف المتغيرات التي تطرأ على محاولات النفاذ عبر الشبكة مستعملا توزيع هذه المحاولات على عناوين بروتوكول الأنترنات و أرقام المنافذ. تقييم هذا المنهاج بين أن مسافة كول-باك المحسوبة اعتمادا على التوزيع المشترك فعالة في كشف جميع أشكال مسح أرقام المنافذ الأفقية منها والرأسية.

الكلمات المفاتيح : علم القياسات ، كشف الحالات الشاذة ،اختلال في حجم حركة مرور المعلومات ، مسح أرقام المنافذ ، محاولات تعطيل الخدمة، الشبكة الوطنية الجامعية.

Table des matières

Liste des figures

Liste des tableaux

"When you can measure what you are speaking about, and express it in numbers, you know something about it; but when you cannot measure it, when you cannot express it in numbers, your knowledge is of a meagre and unsatisfactory kind" *William Thomas Thomson, Lord Kelvin, Electrical Units of Measurement (1883), Popular Lectures and Addresses (1891), Vol. I, 80-I.*

Introduction générale

La métrologie Internet, littéralement science des mesures dans les réseaux Internet, est aujourd'hui un sujet d'ingénierie et de recherche de premier plan. En effet, elle fournit aux opérateurs des réseaux des informations précieuses leur permettant d'évaluer les performances de leurs réseaux, de concevoir une ingénierie optimale de leurs trafics ou encore de planifier efficacement leurs stratégies de développement. Pour les chercheurs en réseaux, la métrologie est l'un des points de passage obligés pour concevoir et valider de nouveaux modèles du réseau ou du trafic.

Depuis l'an 2000, plusieurs projets de recherche sur la métrologie dans les réseaux IP ont été conduits par des opérateurs et des laboratoires de recherche notamment aux États Unis d'Amérique, en France et au Japon ; ils ont abouti à la collecte d'un grand nombre de traces de trafic d'une valeur inestimable pour les études en milieu industriel et académique. Malheureusement, très peu de traces sont rendues publiques sur Internet. De plus, la plupart des traces disponibles sont composées de données agrégées et anonymisées ; limitant ainsi considérablement les possibilités de leur réutilisation pour les besoins de la recherche. En effet, publier des traces de trafic est souvent délicat, car les responsables redoutent de compromettre la vie privée de leurs utilisateurs, la sécurité de leurs réseaux ou les intérêts de leurs institutions.

Par ailleurs, la prolifération des menaces liées au réseau mondial et le professionnalisme grandissant de la cybercriminalité conjuguée à la dépendance croissante des gouvernements, entreprises et particuliers vis à vis du réseau mondial [NARC07] ont abouti à une forte inquiétude au sujet de la robustesse du réseau et la protection de son infrastructure. En particulier, les réseaux de zombies[1], la propagation des vers informatiques et les attaques distribuées de déni de service ont des effets néfastes qui dépassent leurs cibles ; ils génèrent un volume de trafic indésirable important, pouvant saturer les liens et / ou congestionner les équipements actifs de tous les réseaux qu'ils traversent.

Face à ces menaces, les opérateurs de réseaux et les fournisseurs de services Internet se trouvent démunis. En effet, la plupart des systèmes de détection d'intrusion commerciaux utilisent l'approche par signatures qui est particulièrement inefficace face aux activités des réseaux zombies, des attaques distribuées ou encore la propagation des vers informatiques. En effet, ces activités ne peuvent être identifiées par une signature de paquet [Komp07].

[1] Un botnet ou réseau de zombies est composé par des centaines voire des milliers de machines sous le contrôle d'un ou plusieurs pirates distants. Ces derniers les exploitent, à l'insu de leurs propriétaires, pour lancer des attaques distribuées ou des compagnes de pourriels.

Par ailleurs, la plupart des méthodes de détection d'intrusion, issues du domaine de la recherche, adoptent des approches comportementales basées sur des algorithmes paramétriques. Ces derniers consistent à exploiter des modèles statistiques pour décrire le trafic normal, puis de marquer, comme anomalie, toute déviation par rapport à ce modèle. Par conséquent, les performances de ces approches se trouvent étroitement liées à la qualité des modèles utilisés. Or ces derniers sont, souvent, difficiles à exploiter et manquent d'évaluation.

Dans cette thèse, nous nous intéressons à la caractérisation du trafic dans le Réseau National Universitaire (RNU) et nous exploitons les résultats de cette étude pour la détection des anomalies de trafic. Les approches, que nous proposons, sont d'une part adaptées aux des Fournisseurs de Services Internet (FSI) ; d'autre part, elles se basent sur des algorithmes non paramétriques. Ainsi, nos contributions peuvent être résumées dans les quatre points suivants :

1. La collecte de traces de mesures passives détaillées sur les liens du réseau RNU et l'analyse de ces traces. La particularité de cette étude est qu'elle expose les caractéristiques de ce trafic, selon deux niveaux de granularité : paquet et flux, les interprète et les compare aux résultats obtenus par les études métrologiques rencontrées dans la littérature.
2. La proposition d'une nouvelle approche pour la détection d'anomalies de volume, son évaluation et son exploitation pour la détection des anomalies dans RNU. La particularité de cette approche est qu'elle utilise de métriques de volume multipoints.
3. La caractérisation du trafic malicieux dans le réseau RNU et l'estimation de son volume total.
4. La proposition d'une nouvelle approche pour la surveillance des activités de balayages de ports dans les réseaux. L'originalité de cette approche, par rapport aux méthodes comportementales rencontrées dans l'état de l'art, réside dans l'utilisation de distributions d'empreintes de trafic calculées sur des fenêtres disjointes de paquets SYN sans réponses. De plus, notre approche est singulière, dans le sens où, elle utilise la divergence de Kullback-Leibler ; alors que la plupart des méthodes de détection d'anomalies, dites entropiques, utilisent l'entropie de Shannon.

Le reste de ce manuscrit est organisé, comme suit. Dans le premier chapitre, nous présentons un aperçu sur la métrologie dans les réseaux Internet et nous discutons l'état de l'art des caractéristiques du trafic Internet, en mettant en évidence leur variabilité (ou au contraire leur invariabilité) selon le point de mesures ou l'origine du temps. Nous exposons, également, les principaux modèles de trafic aujourd'hui disponibles.

Dans le deuxième chapitre, nous discutons l'état de l'art des approches de détection d'anomalies et nous exposons les potentialités offertes par les télescopes réseaux et les pots de miel pour la surveillance des menaces affectant les réseaux Internet.

Le troisième chapitre introduit notre première contribution, à savoir, les aspects techniques de déploiement de l'infrastructure de mesures passives et les résultats de la caractérisation du trafic

RNU. Nous montrons, grâce à cette étude, l'importance des activités malicieuses dans le réseau RNU et la nécessité de filtrer ces trafics. Nous nous intéressons, également, à la mesure de la dépendance à long terme dans les processus d'arrivée des paquets et des flux et nous discutons les causes de ces phénomènes.

Dans le quatrième chapitre, nous décrivons une approche non-paramétrique pour la détection des anomalies de forte intensité. Cette approche, combinant l'analyse en composantes principales à la détection des points excentriques, est évaluée face à une trace artificielle composée de trafic synthétique et d'attaques perpétuées sur un réseau de test.

Le cinquième chapitre concerne nos deux dernières contributions. En effet, dans sa première partie, nous analysons le trafic lié aux adresses IP inutilisées et invalides et nous en déduisons les caractéristiques du trafic malicieux affectant le réseau RNU. Dans sa seconde partie, nous nous intéressons à la surveillance du trafic généré par les balayages de ports et nous proposons une nouvelle approche de surveillance de ce trafic. Cette dernière est fondée sur la distribution de ce trafic dans l'espace des adresses IP et des numéros de ports TCP visités.

Le dernier chapitre conclut la thèse, résume les contributions et ouvre des nouvelles perspectives de recherche.

Chapitre 1: Mesure, caractérisation et modélisation du trafic Internet

Le réseau Internet par sa croissance rapide et sa complexité grandissante suscite un intérêt croissant dans la communauté des chercheurs désireux de modéliser son trafic, maîtriser son développement ou améliorer ses performances. Pour y parvenir, les chercheurs ont besoin d'effectuer des mesures, d'où le développement de la métrologie Internet comme domaine de recherche actif. En effet, la métrologie Internet englobe des activités diverses allant de la mesure des performances et l'implémentation de mécanismes de qualité de services, à la modélisation de réseaux et des trafics en passant par l'analyse des mécanismes de gestion des files d'attente dans les routeurs ou des algorithmes de routage.

D'un autre coté, les opérateurs de réseaux ont besoin de la métrologie pour leurs activités quotidiennes de gestion, de maintenance, de diagnostic de pannes, de dimensionnement et de planification de leurs réseaux.

Dans ce chapitre, nous présenterons un aperçu sur les techniques de mesures et les principaux projets métrologiques conduits aussi bien en milieu industriel qu'en milieu académique. Par la suite, nous détaillerons les principales caractéristiques du trafic Internet relevées par les études métrologiques et nous discuterons leurs impacts sur la performance et le dimensionnement des réseaux.

1. Métrologie dans les réseaux Internet

Selon Vern Paxson, il existe deux approches pour conduire un projet de métrologie [Paxs03]. La première est l'approche « research driven », que nous pouvons traduire littéralement par « axée sur la recherche ». Elle consiste à définir, en premier lieu, les métriques recherchées avant de mettre en place l'infrastructure et les outils nécessaires, de lancer les compagnes de mesures et d'interpréter les résultats. La seconde est l'approche « measurement driven », que nous pouvons traduire par « la mesure pour la mesure ». Elle consiste à mettre en place une infrastructure de mesures afin de récupérer le maximum d'informations, puis effectuer toutes les analyses possibles sur les traces collectées et interpréter les résultats obtenus.

L'approche « axée sur la recherche » est l'approche adoptée, entre autres, par le groupe IPPM[1] de l'IETF[2] [IPPM], qui a abouti à la normalisation d'un grand nombre de métriques de trafic et au

[1] IPPM : IP Performance Metrics

[2] IETF: Internet Engineering Task Force

développement d'une multitude d'outils spécifiques[3]. De même, les travaux des groupes de travail SNMP (Simple Network Management Protocol) et RMON (Remote Monotoring Protocol) de l'IETF adoptent également cette approche. Ces protocoles, largement implémentés dans les équipements réseau, sont quotidiennement utilisés par les opérateurs pour la surveillance de leurs réseaux et la détection des pannes pouvant les affecter. Néanmoins, le fait que les métriques à mesurer, par les projets adoptant l'approche « axée sur la recherche », sont fixées à priori peut conduire à négliger certaines caractéristiques importantes du réseau ou du trafic.

Un grand nombre de projets adoptant l'approche de mesure pour la mesure ont récemment démarré dont notamment les projets IPMon [Fra01] de l'opérateur SPRINT et PacketScope [Cace00] de l'opérateur ATT ainsi que les projets PMA : Passive measurement Analysis et AMP : Active Measurement Project au NLANR[4]. Le projet MAWI [Mawi] (Measurement and Analysis on the WIDE Internet) dans le cadre du réseau WIDE au Japon, et le projet METROPOLIS en France suivent également cette approche. Tous ces projets sont motivés par les potentialités d'analyses offertes par cette approche ; en effet, elle permet d'effectuer des analyses multiples sur une même trace et des analyses multi-traces.

Au sein de ces deux approches générales, nous distinguons deux classes de techniques de mesures : les techniques de mesures passives et celles actives décrites ci dessous.

1.1 Projets et outils de mesures actives

Le principe des mesures actives consiste à générer un trafic sur le réseau et d'observer les effets des composants et des protocoles du réseau sur ce trafic. Ainsi pour les mécanismes de mesures actives, chaque paquet émis est une sonde, qui en traversant le réseau, se charge d'informations sur la qualité du lien. Les métriques, pouvant être obtenues à partir des mesures actives, ont été normalisées par le groupe IPPM de l'IETF [IPPM et Paxs98].

Les mesures actives simples sont monnaie courante dans Internet. En effet, nombreux outils de mesure active sont disponibles dans les implémentations de TCP/IP actuelles ; il s'agit notamment des outils ping et traceroute.

De même, il existe une multitude d'autres outils de mesures actives tels que clink [Down99] et pathchar [Jaco97] pour l'estimation de la capacité (d'un lien ou d'un chemin), ou Abing [Navr03] et Pipechar [Pipe] pour la mesure du débit disponible sur un chemin réseau donné. Néanmoins la

[3] Parmi ces outils nous pouvons citer la commande "ping" pour la mesure du taux de perte et des délais de transmission, la commande "traceroute" permettant de trouver l'ensemble des nœuds constituant un chemin réseau ou encore "pchar" pour la mesure de la capacité d'un lien.

[4] NLAR: National Laboratory for Applied Network Research aux États Unis.

plupart de ces outils sont peu précis ou trop longs à converger ; par conséquent ils sont inadaptés pour les réseaux ayant des trafics très variables au cours du temps [Labi05].

En outre, il existe des outils de mesures actives matériels (sous forme de matériel embarqué) tels que les sondes RIPE TTM[5] qui permettent de mesurer les délais, les taux de pertes, et la topologie du réseau reliant les différentes sondes RIPE déployées ou encore la plateforme NIMI (National Internet Measurement Infrastructure) [Paxs00]. Cette dernière a été initiée et maintenue par des universités états-unissienes, puis étendue pour couvrir d'autres pays. Un des aspects importants de cette architecture est la séparation des taches ; en effet, les mesures, l'analyse des résultats et la configuration des sondes sont effectués par des agents distincts. De plus, cette infrastructure est flexible permettant l'intégration de nouveaux outils tels que : Treno (TracerouteReno) et Poip (Poisson ping : outil de mesure des pertes et des délais sur un chemin réseau).

1.2 Projets et outils de mesures passives

Le principe des mesures passives est de regarder le trafic traversant le réseau et d'étudier ses propriétés en un ou plusieurs points du réseau. Ces mesures peuvent être effectuées soit au niveau microscopique soit au niveau macroscopique (agrégé).

Les techniques de mesures passives microscopiques tentent d'enregistrer les entêtes des paquets et leurs instants de passage par le nœud surveillé (un routeur par exemple). L'idéal pour les mesures passives microscopiques est de capter tous les paquets traversant le point de mesures, ce qui devient de plus en plus difficile lorsque les débits augmentent. Dans ce dernier cas, une solution serait d'échantillonner le trafic, c'est à dire de baser les mesures, non pas sur la totalité des paquets traversant la sonde, mais sur un sous-ensemble bien spécifié de ces paquets. Le choix d'un algorithme d'échantillonnage n'est pas évident ; d'ailleurs c'est un sujet de recherche actif au sein du groupe PSAMP[6] de l'IETF [Psamp].

Pour effectuer des mesures passives microscopiques, plusieurs outils logiciels, tous basés sur la librairie Libpcap, existent. Il s'agit, notamment des outils Tcpdump [Tcpdump], Wireshark [Wireshark] et Tstat [Mell05] qui sont largement utilisés par la communauté de recherche en réseau ainsi que par les ingénieurs pour l'opération et la gestion de leurs réseaux. La librairie Libpcap permet de lire sur une interface réseau les paquets qui y transitent, d'en récupérer une copie, pour stockage et/ou analyse. Naturellement, ces outils possèdent les avantages et les inconvénients de leur nature logicielle. Leurs principaux avantages résident dans leur facilité d'utilisation, leur disponibilité sur diverses plate-formes et leur faible coût. Concernant leurs inconvénients, ces outils souffrent de problèmes d'imprécisions temporelles et des problèmes de performances limitant leurs

[5] http://www.ripe.net/projects/ttm/index.html

[6] Psamp : Packet Sampling

capacités de capture. En effet, ils ne sont guère efficaces avec des ordinateurs de bureau classiques dès que le débit excède une dizaine de Mbps [Owez07].

La capture du trafic sur les liens à haut débit nécessite l'utilisation de cartes spécialisées telles que les cartes OCxMON [Apsi96] et les cartes DAG[7] [Dag]. Les cartes OCxMON ont été les premières cartes réseaux dédiées à la capture du trafic ; elles permettent la collecte du trafic sur les réseaux ATM (OC3MON pour les liens à 155 Mbps et OC12MON pour les liens à 622 Mbps). Pour l'analyse des traces obtenues par ces cartes, le laboratoire CAIDA[8] [Caida] a développé la suite logicielle CoralReef [Coralreef]. Aujourd'hui cette suite logicielle est étendue pour permettre l'analyse de traces de trafic collectées avec d'autres types de cartes et d'autres logiciels de capture.

Les cartes DAG, sont aujourd'hui les cartes de capture spécialisées les plus utilisées [DAG]. Elles permettent d'extraire les entêtes des paquets, de les estampiller suivant une horloge GPS (general Positionning system) et de les stocker sur un disque dur et/ou de les analyser en ligne grâce à une suite logicielle associée.

Par ailleurs, les techniques de mesures passives macroscopiques consistent à collecter des informations agrégées comme le débit total sur une interface ou le nombre de connexions traversant un point de mesure. Pour ce faire, le module NETFLOW de Cisco [Cis] est utilisé dans de nombreux projets de mesures passives macroscopiques. Ce module, une fois activé au niveau des routeurs, scrute le trafic en transit, reconstitue les flux qui le composent, génère régulièrement des informations statistiques de niveau agrégé et les envoie vers une station pour stockage et analyse. Il faut noter, que les constructeurs Juniper Networks et Huawei Technology offrent une fonctionnalité équivalente au niveau de leurs routeurs (nommée Jflow pour les routeurs Juniper et NetStream pour les routeurs Huawei). Enfin, grâce aux travaux de l'IETF, un standard nommé IPFIX (Internet Protocol Flow Information eXport) a été défini ; il spécifie un format unique pour le transfert des informations agrégées collectées au niveau des routeurs.
Concernant les projets de métrologie passive, le projet conduit par Vern Paxson [Paxs94b] et achevé depuis 1995, a été et reste une référence dans le milieu de la recherche Internet. Il a permis, via l'analyse des traces de trafic, de mettre en évidence certaines pathologies du réseau (livraison des paquets dans le désordre, duplication ou corruption des paquets) et de mesurer les pertes de paquets et les délais de transmission.

L'opérateur mondial SPRINT a démarré en 2000 le projet IPMon qui consiste à enregistrer des traces complètes des entêtes de tous les paquets, transitant par certains points de son backbone IP,

[7] Les cartes DAG, conçues au départ par l'université de Waikato en Nouvelle Zélande, sont aujourd'hui commercialisées par l'entreprise Endace Systems.

[8] CAIDA (Cooperative Association for Internet Data Analysis) est une association qui s'intéresse à la mesure et à l'analyse du trafic Internet.

grâce aux cartes de capture spécialisées DAG [IPMon]. Cette granularité macroscopique a permis d'effectuer des analyses approfondies du trafic et de la topologie du réseau.

Un peu plus proche de nous, le projet de mesures français Metropolis [Metro] s'est largement inspiré du projet IPMon de Sprint. Il a permis de mettre en place une plate-forme de mesures en différents points du réseau de recherche RENATER, du réseau ADSL de France Telecom et du réseau expérimental VTHD (Vraiment très haut débit) [Metro]. Des traces de trafic importantes ont été collectées et analysées sur plusieurs échelles du temps afin de comprendre la dynamique de ce trafic.

1.3 Comparatif mesures actives vs mesures passives

Le tableau suivant (Tableau 1.1) récapitule les avantages et les inconvénients des deux classes de techniques de mesures : les techniques passives et celles actives. Il en ressort que ces deux classes de techniques permettent la mesure de métriques différentes ; D'où l'intérêt de les utiliser d'une manière conjointe. En effet, la tendance aujourd'hui est de mettre en place des infrastructures de mesures muti-points, d'utiliser conjointement différentes techniques de mesures actives et passives afin de corréler et confronter les résultats obtenus par les différentes techniques.

Avantages		**Inconvénients**	
Mesures actives	**Mesures Passives**	**Mesures actives**	**Mesures Passives**
1. Permettent la mesure directe des paramètres de QoS[9] principalement le délai, le taux de pertes et la gigue.	1. Non intrusives 2. Permettent une mesure directe des paramètres utilisés dans l'ingénierie des réseaux.	1. Intrusives: le trafic de mesure peut, dans certains cas, fausser les mesures elles-mêmes. 2. Le fait que certains administrateurs bloquent ou limitent le trafic ICMP[10], peut fausser les résultats obtenus par les techniques actives ; car ces dernières font souvent usage de ce protocole.	1. Elles sont locales (relatives à un lien) et il est difficile de les étendre à la globalité du réseau 2. Elles ne permettent pas la mesure directe des paramètres de QoS. 3. Elles nécessitent des ressources disque et mémoire vive importantes.

Tableau 1.1: Comparatif Mesures Passives vs Mesures Actives

[9] QoS : Quality of Service.

[10] ICMP : Internet Control Message Protocol.

2. Caractérisation et analyse du trafic Internet

Le fait que le réseau mondial soit dépourvu d'un nœud central ou de centre d'administration global, constitue une grande force en termes de scalabilité et de résistance aux pannes ; mais en termes de mesure du trafic et de caractérisation, ceci constitue une grave faiblesse. En effet, les projets de métrologie, quelle que soit leur étendue, ne peuvent fournir qu'une vue restreinte d'une partie de ce réseau à un moment donné. L'extrapolation des résultats obtenus grâce à ces projets à d'autres réseaux Internet est difficile.

C'est pourquoi nous nous intéressons spécialement aux **« invariants »**, c'est à dire les caractéristiques du trafic qui ont été prouvées dans l'état de l'art, **d'une manière empirique**, d'être valables dans une large gamme d'environnements [Floy01]. Les caractéristiques du trafic présumées invariantes, d'après l'état de l'art, seront vérifiées sur les traces de trafic publiques récentes disponibles notamment sur les sites web des projets métrologiques de l'opérateur Sprint [IPMon] et celui du projet Mawi [Mawi]. Ces deux projets fournissent une source d'informations actualisée importante sur le trafic Internet dans deux environnements différents. En effet, le projet MAWI dispose essentiellement de traces de trafic collectées, depuis 1999 à ce jour (octobre 2009), sur un lien trans-pacifique d'accès à Internet (entre le Japon et les USA), alors que le projet IPMon dispose de traces de trafic collectées au niveau de différents liens tels que les liens inter-POPs[11] composant le backbone Internet de l'opérateur Sprint, les liens de peering[12] avec d'autres opérateurs ou encore des liens de type transit Internet.

2.1 Niveau de caractérisation

L'analyse des caractéristiques du trafic Internet s'effectue couramment selon trois entités de trafic, correspondant à trois échelles de temps différentes:

– Les « paquets » forment l'entité de trafic la plus fine que l'on considère dans les réseaux de données. Les paquets sont de longueur variable et leurs processus d'apparition sont très complexes, en raison notamment de la superposition de services de natures très diverses et de l'interaction des couches protocolaires. Le trafic, observé à ce niveau, possède, comme nous allons le montrer plus loin, la caractéristique largement reconnue d'auto-similarité. Les échelles

[11] POP : Point Of Presence, un point de présence Internet est un point d'accès à Internet abritant des serveurs, des routeurs et des commutateurs. Il permet de collecter le trafic provenant des abonnés directement connectés à lui. Les fournisseurs de services Internet ont généralement plusieurs POPs géographiquement dispersés et interconnectés formant ainsi un réseau de backbone.

[12] Le Peering est la pratique d'échanger du trafic Internet avec des pairs. Il est généralement gratuit contrairement au transit qui est payant.

de temps décrivant le processus des paquets sont la microseconde et la milliseconde, en fonction des ordres de grandeur du débit de transmission des liens.

– Les « flots » ou flux constituent une entité de trafic intermédiaire. Ils correspondent à des transferts plus ou moins continus de séries de paquets ayant des caractéristiques communes. En effet, d'après [Claf95], un flot est défini comme un ensemble de paquets IP répondant à une même « spécification de flots » et succédant les uns les autres à un intervalle de temps inférieur ou égal à un seuil donné appelé « Timeout » (TO). Cette notion de « Timeout » permet de garantir que les flots de paquets ne présentent pas des trous trop importants dans leurs processus d'arrivée[13]. Plus précisément, une spécification de flots a quatre dimensions [Claf95] : La directionalité des flots (mono- ou bi-directionalité) ; la prise en compte d'une ou des deux extrémités (origine ou destination des paquets), granularité des extrémités (de la plus fine, les processus générant les flux, à la plus grossière, les sous-réseaux ou les POPs) ; enfin le protocole de la couche transport. Ceci nous permet d'avoir une infinité de spécifications de flux. Par exemple, un flux peut être défini, comme étant l'ensemble de tous les paquets ayant le même quintuplé (adresse source, adresse destination, port source, port destination, protocole de transport) et délimités par un Timeout d'inactivité. Nous notons que cette spécification de flux est unidirectionnelle et qu'elle diffère sensiblement d'une connexion TCP qui correspond à un flux bidirectionnel composé de la réunion de deux flux de sens opposés (les adresses source et destination des deux flux qui composent une connexion sont inversées). Enfin, nous pouvons estimer que les flots ont une durée s'étendant de quelques secondes à quelques minutes, voire quelques heures.

– Les sessions sont des entités de plus haut niveau ; elles sont sensées rapprocher les périodes d'activités des utilisateurs, durant lesquelles ils font le transfert de plusieurs objets. En effet, la notion de session est une tentative de transposer la notion d'appel en téléphonie aux réseaux Internet ; or l'identification des sessions à partir de traces de trafic réelles n'est pas toujours possible à cause de la nature même des applications Internet qui pour la plupart ne comprennent pas de phase d'établissement de connexion. De plus, au sein d'une même session, la qualité de service perçue par les flux diffère sensiblement. Pour toutes ces raisons, nous considérons que cette entité est peu commode pour l'analyse de trafic et nous nous limiterons, dans la suite, à l'analyse et la caractérisation du trafic au niveau paquet et au niveau flux.

[13] L'introduction d'un Timeout sert pour décider qu'un flot est terminé ou non: le dépassement de TO sans nouvelle arrivée de paquets liés à une spécification de flot actif donnée permet de déclarer ce flot terminé et de le supprimer de la table d'états du routeur. Par conséquent, l'estimation d'un TO optimal revêt une importance primordiale afin d'économiser les ressources en mémoire et en temps CPU des routeurs.

2.2 Caractéristiques des paquets

Le trafic Internet, observé au niveau d'un nœud quelconque, est généré par une multitude d'applications et d'utilisateurs. De plus, les paquets de gestion (utilisant les protocoles ICMP, SNMP ou de routage) partagent le même réseau que les paquets de données utiles puisqu'il n'existe pas de réseau de signalisation séparé. En outre, les mécanismes de contrôle (tels que ceux de TCP) interagissent fortement avec le trafic offert, produisant un trafic observable fortement modifié. Toutes ces caractéristiques confèrent à la structure du trafic Internet un caractère d'une grande complexité.

Dans ce qui suit nous allons détailler quelques caractéristiques importantes relatives aux paquets, en discutant leur invariance ou au contraire leur variabilité selon l'endroit et / ou au cours du temps.

2.2.1 Débit sur un lien

Le débit sur un lien est une première information importante pour le dimensionnement des réseaux ; il permet d'avoir une idée sur la quantité de trafic transmise, sur sa variabilité en fonction des différents moments de la journée, de la semaine ou de l'année. Il est mesuré en nombre d'octets ou de paquets par seconde. Pour le calculer, des fenêtres de temps de taille fixe allant de quelques millisecondes à plusieurs minutes sont le plus souvent utilisées[14].

Les études métrologiques du débit sur un lien montrent qu'il présente des variations sur plusieurs échelles de temps et qu'il ne peut être considéré stationnaire[15] que sur des fenêtres de temps ne dépassant pas une ou deux heures. En effet, il présente des cycles journaliers et hebdomadaires bien marqués. Plus précisément, d'après [Paxs94a, Floy01 et Fra03], le débit sur un lien Internet donné suit quotidiennement le cycle des activités humaines, il commence à augmenter autour de 8 - 9 heures, atteint son pic autour de 11 heures, puis connaît une diminution autour de la pause déjeuner pour reprendre l'après midi, avant de diminuer à mesure que la journée de travail se termine. La présence du cycle hebdomadaire se manifeste par la diminution du trafic pendant les week-ends et les jours fériés.

Par ailleurs, nous signalons que le débit sur les liens de l'Internet est généralement asymétrique avec la présence d'un sens transportant beaucoup de trafic, alors que le sens opposé transporte une quantité d'informations beaucoup plus faible. Ceci a été remarqué non seulement sur les liens d'accès à Internet, mais aussi sur les liens de cœur de réseau, les liens de peering entre opérateurs de réseaux et ceux des centres d'hébergement web [Fral01],[Fral03]. D'après [Fral03], la majorité des liens bidirectionnels ont un facteur d'asymétrie variant entre 1 : 2 et 1 : 5.

[14] une fenêtre de temps de 5 minutes est le plus fréquemment choisie notamment lorsque le protocole SNMP est utilisé.

[15] un processus est dit stationnaire si ses caractéristiques ne varient pas avec la définition de l'origine du temps.

La figure 1.1 illustre la variation au cours du temps du débit sur un lien OC12 (622Mbits/s), du cœur du réseau Sprint, durant une semaine en juillet 2001 ; les deux sens de trafic sont représentés séparément, mettant en évidence l'asymétrie du trafic. De plus, la figure montre clairement la présence des cycles journalier et hebdomadaire.

Dans [Cho06], les auteurs analysent des traces de trafic collectées, entre 2004 et 2006, au niveau de sept fournisseurs de services Internet au Japon offrant des accès haut débit aux utilisateurs résidentiels via les technologies ADSL et FTTH (Fiber To The Home). Ils mettent en évidence la présence des cycles journalier et hebdomadaire, mais relèvent que la période de trafic maximal se situe en début de soirée (entre 21h et 23h) et que le trafic provenant d'utilisateurs disposant de liaisons FTTH présente une asymétrie au profit du sens de trafic ascendant[16]. Ces observations laissent deviner que le débit sur les liens de l'Internet subit des mutations et / ou qu'il dépend de facteurs géographiques et culturels.

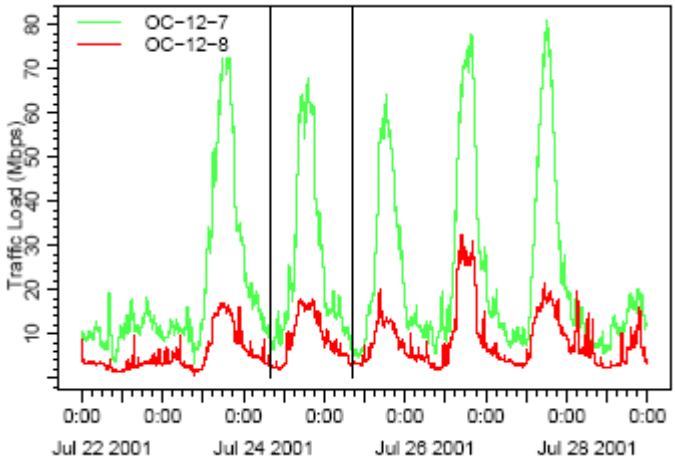

Figure 1.1: Débit sur un lien OC12 du backbone Sprint [Fral03]

2.2.2 Répartition du trafic par protocole

Au-dessus de la couche réseau IP, les protocoles de transport les plus répandus sont UDP (User Defined Protocol) et TCP (Transport Control Protocol). Le trafic TCP est largement majoritaire, comme le montre les études [cace89, Fral03, Fome04, IPMon et Mawi]. De plus, il est intéressant de noter que la répartition du trafic par protocole est restée relativement stable depuis plusieurs années. En effet, en 1989, longtemps avant l'apparition du Web et de l'accès généralisé à Internet, Caceres a montré dans [Cace89] que TCP générait 90% des octets et 80% des paquets transitant par le lien de 56kb/s reliant les laboratoires Bell au USA à l'Internet. Par la suite, l'étude conduite par CAIDA et basée sur des traces de trafic publiques datant de la période entre 1998 et 2003 [Fome04] donne,

[16] Habituellement le sens de trafic ascendant (de l'abonné vers le réseau) transporte beaucoup moins de trafic que celui descendant.

dans le Tableau 1.2, les fourchettes de proportions de trafic TCP/UDP et montre clairement la prépondérance du trafic TCP. L'étude des traces de trafic récentes[17], disponibles sur le site web du projet MAWI [Mawi], montre que le trafic TCP continue d'être majoritaire et que les quelques cas où un trafic UDP ou ICMP important ont été enregistrés, semblent être causés par des opérations de maintenance ou des anomalies.

Enfin, d'autres protocoles de transport sont présents dans les observations, mais ils ne représentent qu'une proportion négligeable du volume du trafic véhiculé. Il s'agit, par ordre d'importance, des protocoles suivants : ICMP (Internet Control Message Protocol), RSVP (Reservation Protocol), GRE (General Routing Encapsulation), SIPP-ESP (SIPP Encap Security Payload) et NHRP (NBMA Next Hop Resolution Protocol).

	TCP	**UDP**	**Autres**
% paquets	75 ± 12	22 ± 11	3 ± 2
% octets	83 ± 11	16 ± 6	1 ± 1

Tableau 1.2 : Proportions de trafic par protocole de transport [Fame04]

2.2.3 Répartition du trafic par application

Selon l'IANA[18], (Internet Assigned Numbers Authority), les numéros de ports TCP ou UDP forment trois catégories de base : les ports « bien connus » dont les numéros sont inférieurs à 1024, les ports dits « enregistrés » dont les numéros sont compris entre 1024 et 49151, et finalement les ports « dynamiques ou privés ».

Les ports « bien connus » sont assignés par l'IANA à des services Internet bien définis[19], ils ne peuvent être utilisés que par les applications serveurs comme ports d'écoute. Les ports « enregistrés » peuvent avoir une double utilisation, puisqu'ils peuvent aussi bien être utilisés comme ports sources éphémères que comme ports d'écoute (certains ports enregistrés sont assignés par l'IANA à des services particuliers). Enfin, les ports dynamiques sont rarement utilisés par des applications clientes comme ports sources et ne sont pas assignés par l'IANA à aucun service Internet.

[17] Il s'agit de traces de trafic collectées quotidiennement, et presque sans interruption, depuis 1999 à ce jour (septembre 2009), sur un lien trans-pacifique d'accès à Internet (entre le Japon et les USA).

[18] La page web http://www.iana.org/assignments/port-numbers, présente la liste complète des ports bien connus ainsi que celle des ports enregistrés.

[19] Par exemple un serveur FTP utilise le port TCP numéro 21, un serveur Web est généralement associé au port TCP numéro 80.

L'identification de l'application qui a généré une connexion TCP ou UDP, en se basant sur les numéros de ports possède plusieurs limites. D'abord, il n'est pas toujours aisé de différencier les ports clients de ceux serveurs, notamment pour les connexions UDP dépourvues de phase d'établissement de connexion. De plus, l'administrateur système n'est pas obligé de se confirmer aux spécifications de l'IANA ; il peut utiliser comme port en écoute, un numéro de port quelconque. Plus particulièrement, certaines applications serveurs écoutent sur des numéros de ports associés par l'IANA à d'autres services : Il s'agit le plus souvent du port TCP 80 qui est largement prisé par les applications serveur désirant outre passer des règles de filtrage ou de limitation de bande passante, implémentés au niveau de certains réseaux. De plus, l'identification des applications émergentes, telles que celles de partage de fichiers pair à pair, de téléphonie sur Internet ou de jeux multi-joueurs interactifs, est ardue, car elles fonctionnent indifféremment avec le protocole TCP ou UDP et négocient les numéros de ports qu'elles utilisent pour le transfert de leurs données. En effet, une étude de CAIDA [Kara04] a montré clairement que le trafic généré par les applications pair à pair est de plus en plus difficile à identifier en utilisant les numéros de ports car de plus en plus d'applications n'utilisent pas des numéros de ports fixes.

Malgré toutes les limites de l'identification des applications en se basant sur les numéros de ports, citées plus haut et identifiées notamment dans [Moor05], la répartition du trafic par numéro de port continue à être largement utilisée pour connaître les usages de l'Internet. Ceci s'explique d'une part par sa simplicité, d'autre part par les inconvénients des solutions alternatives de classification du trafic par application. En effet, les techniques de classifications basées sur l'analyse systématique du contenu des paquets nécessitent des ressources matérielles non négligeables et ne peuvent fonctionner lorsque le trafic est chiffré ou lorsque les connexions n'ont pas échangé suffisamment de données. Alors que les techniques plus récentes basées sur l'approche comportementale, consistant à associer des profils de trafic à des applications ou à des classes d'applications, sont complexes et nécessitent une phase d'apprentissage, sans quoi elles auraient un taux de faux classement important.

Contrairement à la répartition du trafic par protocole, celle par numéro de port a connu des variations importantes tout au long de l'histoire de l'Internet. En effet, durant l'ère précédant l'avènement massif du web (avant 1995), les ports TCP les plus utilisés étaient, selon les études de [Cace89 et Paxs94], ceux des protocoles SMTP (pour l'envoi des messages électroniques), FTP (pour le transfert de fichiers) et telnet (pour l'accès aux serveurs distants). Durant les années 1995 - 2000, le trafic Internet était dominé par le trafic web (port TCP 80) comme le montre l'étude très détaillée des caractéristiques et des profils de trafic [Thom97][20] ou celle de [Mccr00]. À partir de 2001, la

[20] Il s'agit d'une étude basée sur des campagnes de mesures réalisées sur le réseau dorsal de MCI : Deux liens, respectivement dédiés au trafic domestique (interne aux USA) et au trafic international (liaison USA - GB) sont observés durant une semaine. Elle expose des résultats très complets sur les poids respectifs, exprimés en flux, paquets et octets, des différentes applications entrant dans la composition du trafic.

répartition du trafic Internet par application est devenue panachée avec l'apparition d'un trafic important, utilisant des ports non associés à des applications connues.

En effet, le Tableau 1.3, dressant les répartitions de trafic TCP par application (reconnues par leurs numéros de port), permet de comparer les résultats de la campagne MCI [Thom97] avec ceux des études métrologiques effectuées sur le réseau IP de France Télécom en 2000 et 2001 [Oliv03]. Pour ces dernières mesures, les deux sens de transmission, montant (de l'abonné vers le réseau) et descendant (du réseau vers l'abonné) sont distingués. Il est remarquable que les traces de trafic françaises exhibent une répartition du trafic panachée caractérisée par un pourcentage important d'octets transportés par des ports TCP non identifiés (48% des octets transportés dans le sens montant et 22% pour l'autre sens). Il s'agit très probablement de trafic généré par des applications pair à pair utilisant des ports TCP non enregistrés. En effet, lors de téléchargement de fichiers en utilisant des applications pair à pair, la machine de l'utilisateur est simultanément client et serveur pour les autres nœuds du réseau pair à pair ce qui a pour effet d'augmenter le trafic montant et de symétriser les deux sens de trafic.

Trace	**%**	**HTTP**	**SMTP**	**POP3**	**FTP**	**NNTP**	**Autres**
MCI 1997 Lien domestique	**Paquets**	75	6	-	3	<1	15
	Octets	80	5	-	5	2	8
POP1 FT 2000 Sens montant	**Paquets**	65	2	3	5	2	23
	Octets	33	8	1	9	<1	48
POP1 FT 2000 Sens descendant	**Paquets**	65	2	3	6	2	22
	Octets	64	0	2	9	3	22
POP2 FT 2001 Sens montant	**Paquets**	79	3	3	-	-	15
	Octets	65	17	1	-	-	17
POP2 FT 2001 Sens descendant	**Paquets**	72	2	4	-	-	22
	Octets	72	<1	4	-	-	24

Tableau 1.3 : Proportions de trafic TCP par application [Oliv03]

Enfin, sur le site web du projet Mawi [Mawi], présentant les résultats d'analyse de traces de trafic collectées régulièrement depuis 1999, nous remarquons que le trafic utilisant des ports inconnus, mesuré en nombre de paquets et d'octets, a commencé à croître à partir de 2000, pour atteindre jusqu'à 80% des paquets et des octets en 2003. Mais à partir de 2004, ce trafic semble avoir diminué sensiblement (surtout lorsqu'il est mesuré en octets) au profit du trafic utilisant le port TCP 80, par conséquent le trafic TCP utilisant le port 80 est redevenu majoritaire (en effet, d'après les traces de

2007, 70% des paquets l'utilisent)[Mawi]. A notre avis, ceci ne s'explique pas par un regain d'intérêt pour les sites web classiques, mais plutôt par l'apparition de nouveaux sites dont les contenus sont crées par les internautes eux-mêmes tels que les sites de partage de vidéo (Dailymotion[21] et Youtube[22]), les blogs (blogger[23], skyblog[24]), les encyclopédies participatives (wikipedia[25]) ou encore les sites des réseaux sociaux permettant aux internautes de retrouver leurs amis et d'interagir avec eux (Facebook[26]). En outre, de nombreuses applications émergentes choisissent le port TCP 80 ; en particulier, l'étude faite dans [Silv07], montre que parmi les quatre applications de WebTV[27] populaires : PPLive, TVAnts, PPStream et SOPCast, un seul utilise UDP, alors que tous les autres utilisent majoritairement TCP et plus précisément le port TCP 80. Ceci paraît de prime abord bizarre, surtout que, théoriquement, le protocole UDP est plus adapté au transport des flux multimédias que le protocole TCP[28].

Le fait que la majorité des applications privilégient l'utilisation du protocole TCP pour le transport des flux multimédias s'explique, non seulement par l'accroissement des débits d'accès à Internet permettant d'avoir une rapidité de transmission suffisante pour les applications multimédias (même avec TCP), mais aussi par la robustesse et la fiabilité du protocole TCP, qui s'adapte aux diverses caractéristiques du réseau, aux pertes de paquets et à la congestion des liens.

Concernant le trafic UDP, la seule application généralement identifiée est l'application de résolution de noms (DNS) qui génère de nombreux flux composés pour la plupart d'un seul paquet de taille inférieure à 80 octets. Ce pendant, d'après les observations faites sur le réseau de France Télécom, le poids du trafic DNS est faible, puisqu'il représente seulement 5 à 10% des paquets (et de 2 à 8% des octets) utilisant le protocole UDP [Oliv03].

2.2.4 Distribution de la taille des paquets

Les études métrologiques montrent que la distribution de la taille des paquets est surtout sensible au sens de transmission du trafic. Ceci s'explique par le fait que la plupart des liens Internet présentent

[21] http://www.dailymotion.com

[22] http:// www.youtube.com/

[23] http://www.blogger.com

[24] http://www.skyrock.com

[25] http://www.wikipedia.org

[26] http://www.facebook.com, ce site regroupe 50 millions d'utilisateurs en juillet 2007.

[27] Les WebTVs permettent aux internautes de regarder les chaînes de télévision en ligne en utilisant la technologie du streaming (lecture en continu ou lecture progressive).

[28] Il est généralement admis que TCP n'est pas approprié pour les applications ayant des contraintes temps réel (notamment les applications de diffusion multimédia, ou les jeux multi-joueurs) puisqu'elles n'ont pas besoin, et peuvent même souffrir, des mécanismes de transport fiable de TCP.

une dissymétrie de répartition des serveurs et des usagers entre leurs deux extrémités ; de ce fait le sens serveur vers usager comporte surtout des paquets de grande taille, liés au transfert des données demandées, alors que l'autre sens (usager vers serveur) transporte essentiellement des paquets de petite taille, liés au transfert des requêtes et des informations protocolaires (accusés de réception, paquets de début et de fin de connexion TCP, ...).

La figure 1.2, illustrant les distributions des tailles des paquets sur un lien de peering appartenant au réseau dorsal de Sprint, montre que les deux sens du trafic ont des distributions sensiblement différentes. En effet, pour le sens « peer-in », les paquets de taille minimale sont majoritaires et représentent 70% des paquets. Alors que dans l'autre sens, la proportion de paquets de taille minimale est inférieure à 40%.

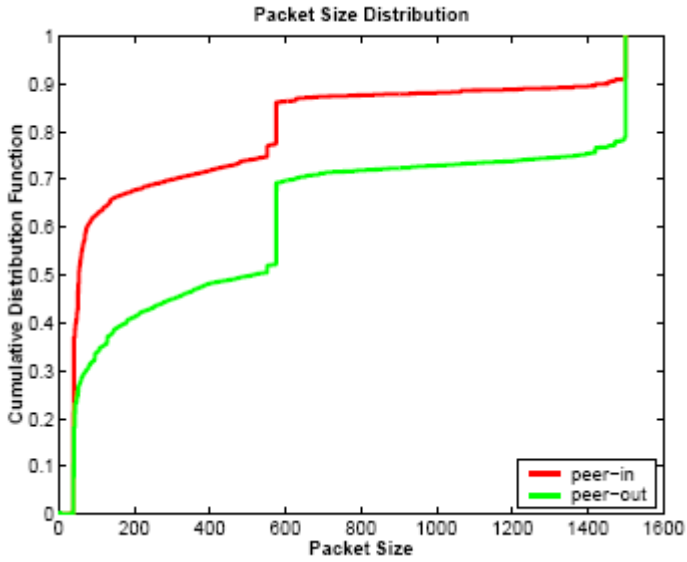

Figure 1.2 : Distribution des tailles des paquets sur un lien de peering par sens [Fral03].

En outre, la figure 1.2 montre que les deux distributions ont trois modes prépondérants correspondant aux tailles des paquets suivantes : 40, 572 et 1500 octets. Le premier correspond aux paquets de taille minimale (40 octets) ; donc aux paquets TCP de signalisation (ouverture de connexion, fermeture de connexion,..) et d'acquittement. Le mode 572 octets, correspond à la taille des paquets TCP par défaut, (lorsque aucun mécanisme de découverte de la valeur du MTU[29] n'est utilisé). Le troisième mode correspond aux paquets de taille maximale (généralement 1500 octets qui correspond à la MTU dans les réseaux de type Ethernet).

Sur les traces de trafic récentes, disponibles aussi bien sur le site du projet Mawi, que celui de IPMon, nous remarquons la disparition, ou du moins la diminution de l'importance, du mode 476 octets. La Figure 1.3, provenant du site Mawi, en est un exemple ; elle montre que la majorité des

[29] MTU : Maximum Transmission Unit, la taille maximale de données pouvant être envoyées par la couche liaison de données dans une seule trame.

paquets (60%) ont une taille minimale mais ne transportent qu'un petit pourcentage (moins de 10%) des octets ; en contre partie, les paquets de grande taille transportent la majorité des octets, malgré qu'ils ne représentent que 30% des paquets.

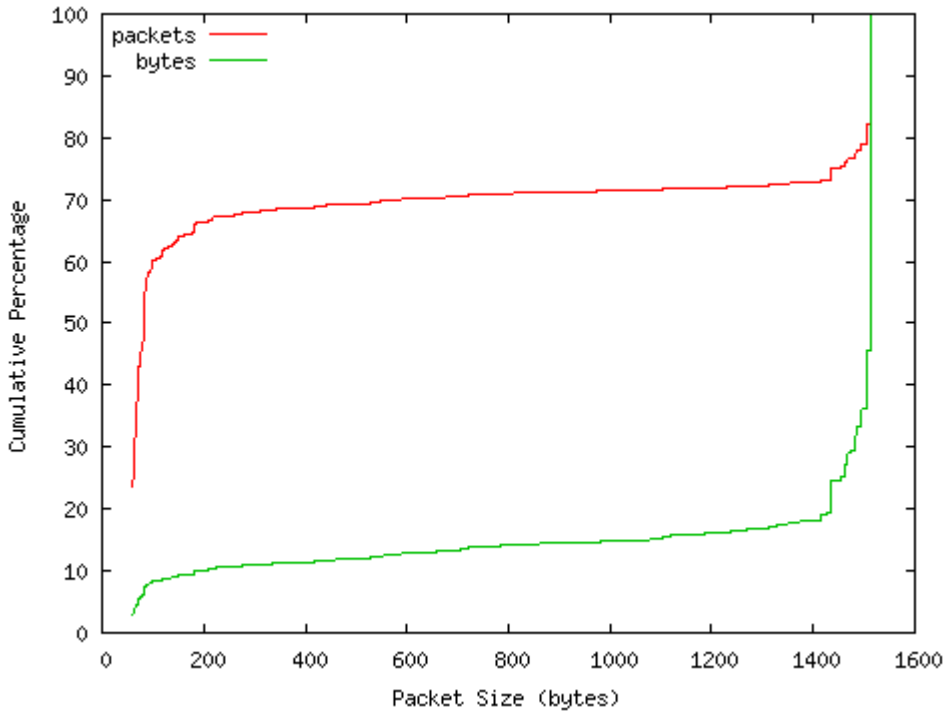

Figure 1.3 : Distribution des tailles des paquets sur un lien trans-pacifique (Janvier 2008)[30]

2.3 Caractéristiques des flux

Un flux est un ensemble de paquets ayant des caractéristiques communes ; ainsi, il apparaît, d'après cette définition, qu'il existe une multitude de spécifications de flux. Nous nous intéressons dans ce qui suit principalement aux flux unidirectionnels microscopiques définis par les 5-uplets : adresse source, port source, adresse destination, port destination et protocole de transport.

Il a été remarqué dans plusieurs études, que le nombre de flux actifs simultanément sur un lien, même à très haut débit et pour une définition de flux microscopique est relativement faible (par rapport au nombre de flux possibles). En effet, dans [Floy03], les auteurs ont remarqué que le nombre moyen de flux, définis par leurs 5-uplets, actifs simultanément sur un lien OC12 (622 Mbits/s) ne dépasse pas 50 000. Alors que sur les liens 0C48 (2,5 Gbits/s), ce nombre est inférieur à 300 000. Ces résultats démontrent, d'après [Floy03], que les techniques d'ordonnancement individuel des flux, au niveau des liens d'accès à Internet de type OC12, sont tout à fait possibles notamment au regard des performances toujours croissantes des routeurs.

[30] http://mawi.wide.ad.jp/mawi/samplepoint-F/2008/200801141400.html

D'après la Figure 1.4, nous remarquons que le nombre de flux actifs sur un lien présente le même cycle journalier observé pour le débit en paquets ou en octets. Toutefois, il est important de signaler que les pics de courtes durées visibles sur la courbe OC-12-7 de la Figure 1.4 et ne s'accompagnant pas de pics semblables au niveau de la courbe OC-12-8 (correspondant au trafic dans l'autre sens) reflètent la présence de balayage de ports ou d'attaques DoS.

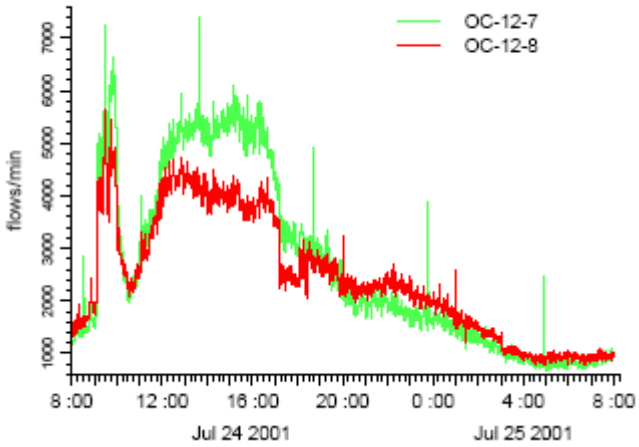

Figure 1.4 : Nombre de flux par minute sur un lien OC12 par sens du trafic [Floy03]

Par ailleurs, dans [Papa01] l'auteur définit deux classes de flux, différenciées selon leurs tailles, les flux souris et les flux éléphants. Les premiers sont les flux composés par moins de 10 paquets, alors que les seconds sont les flux composés par plus de 100 paquets. Il montre, ainsi, que la grande majorité des flux véhiculés par les liens de l'Internet sont des flux souris qui malgré leur grand nombre, ne représentent qu'un petit pourcentage du volume du trafic total. Alors que, les flux éléphants représentent un petit pourcentage du nombre de flux total, mais transportent la majorité du trafic. Ce constat est également confirmé par les études des traces collectées dans le cadre du projet français Metropolis. En effet, la Figure 1.5 montre que 3 % des flux (flux éléphants) gênèrent 80 % du volume total du trafic alors que 87 % des flux (flux souris) gênèrent à peine 4 % du trafic total.

Il est remarquable que ce constat reste valable pour une multitude de définitions de flux. En effet, à tous les niveaux d'agrégation : aussi bien pour les flux TCP, les flux définis par des préfixes réseau, les flux spécifiés par leurs systèmes autonomes ou encore les flux applicatifs, la présence de flux souris et de flux éléphants a été remarqué mais à des degrés variables, notamment dans [Broi04 et Soul03 et Bhat01].

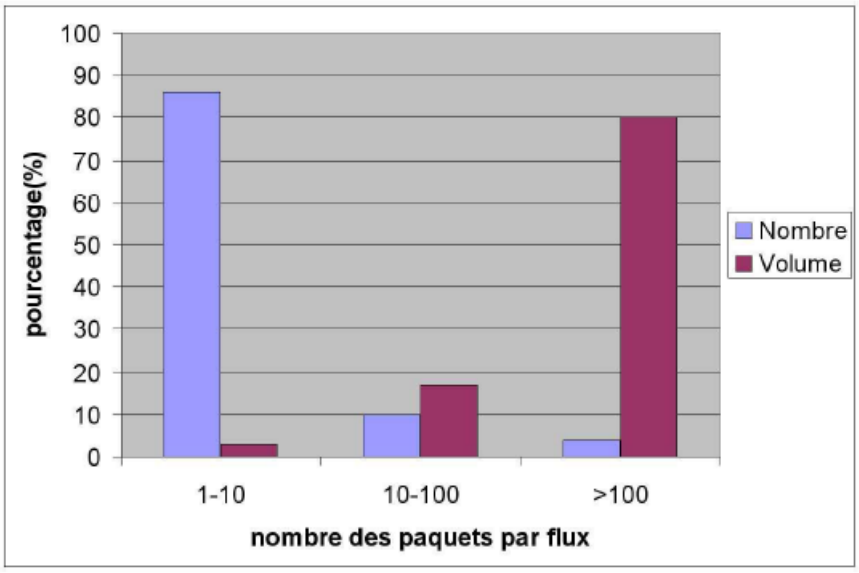

Figure 1.5 : Répartition des flux TCP selon leur taille [Larr05]

Par ailleurs, l'étude faite dans [Soule04] a montré que, pour une spécification de flux utilisant les préfixes des adresses sources et des adresses destination, la classe des flux éléphants est relativement stable sur quelques jours, c'est à dire que les flux composant cette classe restent relativement inchangés, d'où la possibilité, offerte aux administrateurs de réseaux de se focaliser sur les quelques éléphants afin de résoudre la majeure partie des problèmes à l'intérieur du réseau.

Enfin, d'autres classifications de flux plus fines existent dans la littérature. Par exemple dans [Soul04], quatre classes de flux sont définies ; il s'agit, par ordre de taille décroissant, des flux éléphants, buffles, souris et libellules. Dans [Brow02], les auteurs définirent, une double classification selon la taille des flux (éléphants / souris), et selon leur durée (tortues / libellules). En effet, les flux tortues sont les flux de longue durée (supérieure à 15 minutes) ; alors que, les flux libellules sont ceux de courte durée (inférieure à 2 secondes).

3. Modélisation du trafic Internet

Les travaux de modélisation du trafic Internet se sont principalement intéressés à la modélisation stochastique des séries temporelles matérialisant les arrivées des paquets et des flux d'informations échangés sur le réseau mondial. Les motivations de telles études résident dans l'usage qui peut être fait à posteriori de ces modélisations pour assurer le bon fonctionnement du réseau et optimiser son développement (topologie, dimensionnement, etc.).

3.1 Modélisation du trafic Internet au niveau paquet

L'auto-similarité du trafic de données (plus exactement auto-similarité asymptotique d'ordre 2 ou dépendance à long terme) a été mise en évidence dans [Lela94] et souvent confirmée depuis. Elle

signifie que la structure des variations d'amplitude du signal analysé se reproduit de manière similaire quelle que soit la finesse temporelle avec laquelle il est représenté[31]. Ce comportement est à l'opposé d'un signal Poissonnien, dont les variations d'amplitude diminuent au fur et à mesure que l'on augmente la taille de la fenêtre d'intégration. La Figure 1.6 illustre la présence de l'auto-similarité dans le processus d'arrivée des paquets. En effet, le trafic, mesuré en bits/s, et calculé sur des fenêtres de tailles croissantes (100ms, 1s, 10s et 100s), est toujours composé de rafales.

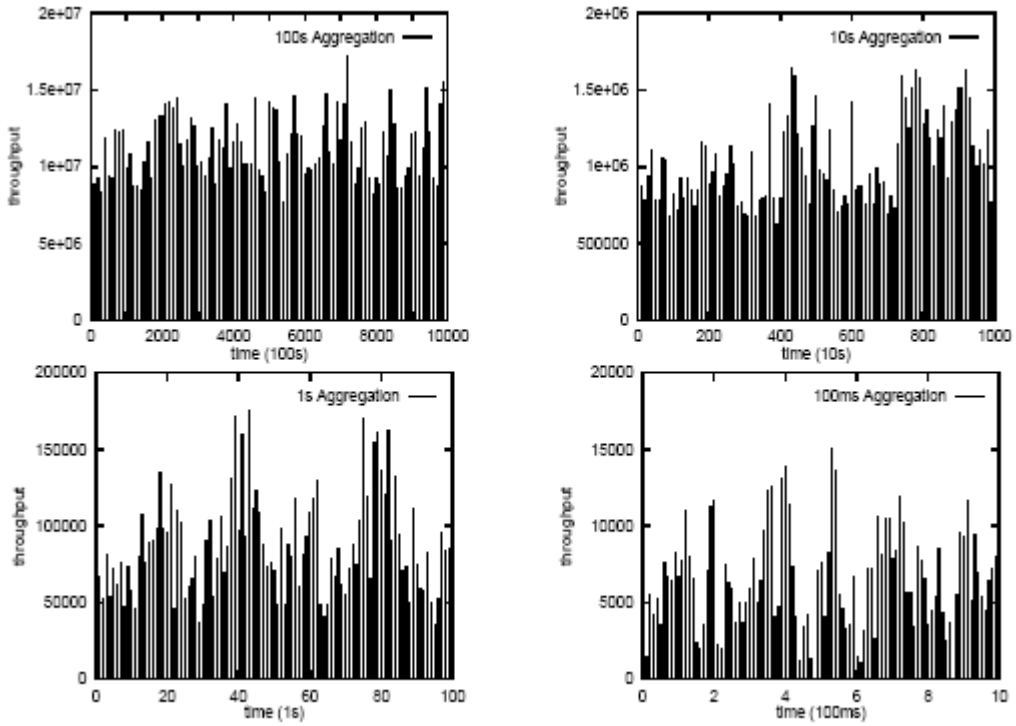

Figure 1.6 : Trafic mesuré sur des fenêtres de 100s, 10s, 1s et 100ms [Park00]

Ceci se traduit mathématiquement par le fait que la fonction d'auto-corrélation du nombre de paquets ou du nombre d'octets, transférés par unité de temps (typiquement 100 ms), se caractérise par une décroissance lente, sous forme de loi de puissance.

Il est remarquable que la dépendance à long terme (LRD) des processus d'arrivée des paquets, dans les réseaux de données et dans Internet en particulier représente ainsi **l'une des caractéristiques invariantes du trafic Internet** [Foly01]. En effet, elle a été observée dans divers contextes : dans [Lela94], elle était observée dans le trafic collecté au niveau d'un réseau Ethernet ; alors que dans [Paxs94b], elle a été mise en évidence dans le trafic Internet bien avant l'apparition du web, finalement, dans [Crov96], l'auto-similarité du trafic web a été analysée.

[31] Voir l'annexe A pour les définitions mathématiques de l'auto-similarité, de l'auto-similarité asymptotique d'ordre 2 et de la dépendance à long terme.

Dans toutes ces publications, ainsi que dans d'autres travaux, le paramètre de Hurst, relatif aux processus d'arrivée des paquets, est estimé. Les valeurs rapportées dans la littérature varient généralement entre 0,7 et 1 (voire légèrement supérieures à 1 dans certains cas). Ces valeurs indiquent l'extrême variabilité du nombre d'octets par unité de temps, et du temps d'inter arrivée des paquets.

3.1.1 Processus d'arrivée des paquets

L'auto-similarité du trafic Internet au niveau des arrivées des paquets est considérée parmi l'une des rares caractéristiques invariantes de l'Internet, c'est pourquoi plusieurs modèles stochastiques auto-similaires ont été proposés dans la littérature. Dans [Bena04], une étude comparative assez complète de ces modèles est proposée ; elle expose notamment le mouvement brownien fractionnaire, les modèles ARIMA fractionnaire, le modèle de superposition de source ON-OFF à haute variabilité et les modèles multifractals à base d'ondelettes.

À noter toutefois que les travaux de Cao dans [Cao01 et Cao02] défendent une théorie opposée à l'auto-similarité ; ils sont, de ce fait, à l'origine de nombreux débats dans la communauté de la recherche. En effet, ils ont montré, empiriquement, que la dépendance entre les arrivées des paquets diminue avec l'augmentation de la charge du réseau. Par conséquent, ils ont prédit, qu'avec la croissance de la bande passante sur Internet, l'emploi du processus de Poisson pour modéliser les arrivées des entités de trafic à tous les niveaux deviendra de plus en plus légitime. Or, les dernières études métrologiques [Borg09] montrent que les arrivées des paquets sur les liens à très haut débit sont toujours auto-similaires.

3.1.2 Discussion autour des causes et des conséquences de la LRD

La recherche des causes de la LRD observée dans les processus d'arrivée des paquets a suscité l'intérêt des chercheurs. En effet, la corrélation entre la présence de gros flux et l'auto-similarité du trafic Internet a été mise en évidence dans plusieurs études. Dans [Park96], les auteurs ont montré, grâce à des simulations, que plus les fichiers transférés sur un lien sont gros, plus le niveau de la LRD présente dans le trafic résultant est important. De même, dans [Will97], les auteurs ont montré que le trafic résultant de la superposition de plusieurs sources de trafic indépendantes de type ON/OFF, avec des périodes ON et OFF alternées et telles que les distributions des durées des périodes ON ou OFF sont à queue lourde, est un trafic auto-similaire. Enfin, dans [Larr05], l'auteur a étudié la LRD par classe de trafic ; pour ce faire, il a utilisé la double classification éléphants / souris, tortues / libellules (définie par [Brow02]). Il a trouvé ainsi, que ce sont les flux éléphants (gros flux) et les flux tortues (flux lents) qui contribuent le plus à la LRD du trafic total observé.

Par ailleurs, en effectuant une analyse de traces de trafic réelles, [Moln00] a montré que HTTP et FTP sont les principales applications contribuant, de par leur extrême variabilité, aux propriétés de

LRD détectées au niveau des paquets IP. Alors que, dans [Owez07], les auteurs affirment que c'est la taille des flux transportés sur un lien qui est la plus déterminante dans le niveau de LRD observé et ce indépendamment des applications utilisées.

En outre, les mécanismes de boucle fermée et de contrôle de congestion de TCP sont également incriminés pour leur rôle dans la LRD du trafic Internet. En effet, le fait que l'envoi des paquets suivants, d'une connexion quelconque, est conditionné par la réception d'acquittements des paquets précédents, résulte dans la création de dépendance à court terme entre les paquets d'une même fenêtre de congestion TCP. De la même façon, les deux mécanismes de contrôle de congestion de TCP (« slow-start » et « congestion avoidance ») introduisent de la dépendance à long terme entre les paquets de différentes fenêtres de congestion. Ainsi, en généralisant ces observations, il est évident que tous les paquets TCP d'une connexion sont dépendants les uns des autres.

Pour le trafic UDP (résultant du transfert de séquences vidéo à débit variable VBR selon la terminologie ATM) la présence de la LRD a été expliquée dans [Bera95] par la variabilité des paramètres de transmission liés au codage des trames et la dynamique des images.

Concernant ses conséquences, selon [Park97] la présence la LRD, se traduisant par les fortes oscillations du débit utilisé, résulte dans un gaspillage des ressources du réseau. En effet, la capacité libérée par un flux subissant une perte ne peut pas être immédiatement utilisée par un autre en raison de la phase de slow-start.

3.2 Modélisation du trafic Internet au niveau flux

La modélisation du trafic Internet au niveau flux est motivée par le fait que l'analyse des performances du réseau s'effectue plus aisément, à ce niveau. Par ailleurs, les différents schémas d'architectures de routage, de « traffic engineering » (MPLS, routage orienté QoS, etc.), voire de fourniture de services différenciés (IntServ, DiffServ), prennent en compte, également, la notion de flux selon des niveaux d'agrégation très variables.

Les modèles proposés dans la littérature concernent aussi bien la distribution de la taille et de la durée des flux, ainsi que le temps d'inter-arrivée des flux.

3.2.1 Distribution de la taille et de la durée des flux

Toutes les analyses de traces de trafic, produites dans la littérature, mettent en évidence des lois de distribution à décroissance lente, dès que l'on s'intéresse à un paramètre lié à la taille ou à la durée des flux. Ce phénomène de décroissance lente signifie que la probabilité d'obtenir de très grandes valeurs de la variable aléatoire est asymptotiquement beaucoup moins faible que pour une loi exponentielle.

La figure 1.7 montre que la distribution des tailles des flux a évolué entre 2000 et 2003 en s'éloignant, de plus en plus, de la distribution exponentielle en ayant une queue de plus en plus lourde. Ceci s'explique par le fait que la généralisation de l'accès Internet à haut débit et l'augmentation des capacités des ordinateurs ont permis aux internautes d'échanger des fichiers multimédias de plus en plus volumineux et de générer, ainsi, des flux de durée et de taille de plus en plus grandes.

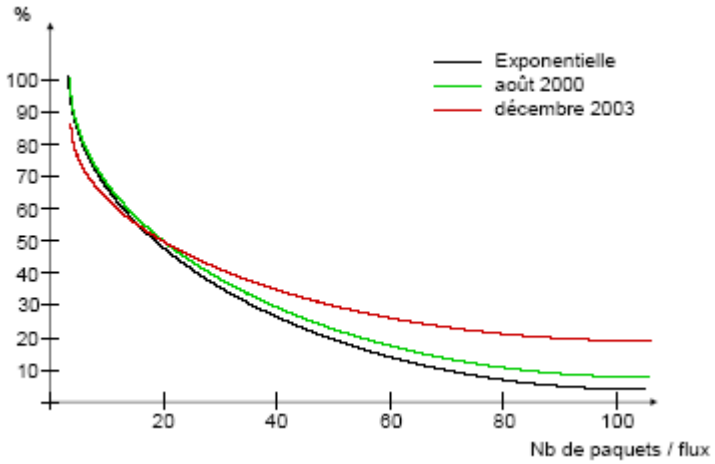

Figure 1.7 : Évolution de la distribution de la taille des flux [Owez04]

Pour modéliser les distributions des tailles ou des durées des flux, les lois de probabilité couramment utilisées sont la loi de Pareto, la loi de Weibull et la loi log-normale. Toutefois, il est important de signaler que seule la loi de Pareto est une loi à décroissance lente, puisque sa distribution (cumulative complémentaire) est proportionnelle à $x^{-\alpha}$ avec $\alpha > 0$. De plus, pour $\alpha \in]0,2[$, la moyenne de la variable aléatoire est finie, mais sa variance ne l'est pas ; alors que pour $\alpha \in]0,1]$, les deux sont infinies[32]. Les deux autres lois log-normale et Pareto ne sont pas des lois à décroissance lente. En effet, la loi log-normale est seulement **à décroissance sous-exponentielle** [Paxs94b] ; alors que la loi de Weibull n'est qu'une généralisation de la loi exponentielle qui peut avoir, dans certains cas, une décroissance sous-exponentielle.

Par ailleurs, la loi de Pareto ne fournit pas un bon modèle de représentation de l'ensemble de la distribution puisque seule la queue de la distribution est identifiée comme étant à décroissance lente. En effet, pour modéliser le corps de la distribution, [Paxs94c et Nabe98] proposent d'utiliser la loi log-normale ; alors que, dans [Jena00], les auteurs ont proposé de modéliser l'ensemble de la distribution par des mélanges de lois (log-normal / Pareto).

[32] Ce comportement est significatif d'une très grande variabilité de la variable aléatoire considérée.

3.2.2 Processus d'arrivée des flux

Les processus poissoniens ont été historiquement utilisés pour modéliser les appels téléphoniques. Par la suite, ils ont été repris par la plupart des modèles de performance décrivant les arrivées des flux dans le trafic Internet [Bara02]. Or, si on considère qu'un flux correspond, plus au moins, à une demande de transfert d'un document (page Web par exemple) ou d'une fraction d'un document (objet d'une page Web) ; alors au sein d'une même session d'un utilisateur, il est logique de s'attendre à des inter-dépendances dans le processus temporel d'apparition des flux, lequel devra alors s'écarter sensiblement du modèle Poissonnien. C'est ce que l'on vérifie expérimentalement, dans plusieurs études métrologiques [Paxs94b et Feld00 et Oliv01].

En effet, dans [Feld00 et Oliv01], les auteurs ont estimé la fonction d'auto-corrélation des temps d'inter-arrivée des flux et ont mis en évidence l'existence d'une corrélation persistante dans le temps entre les arrivées des flux (sur une ou quelques dizaines de secondes), quoique de niveau assez faible. Il est remarquable que ces deux études ont adopté chacune une spécification différente ; en effet, dans [Oliv01], les auteurs s'intéressent aux micro-flux unidirectionnels TCP ou UDP, tandis que dans [Feld00], ils s'intéressent aux connexions TCP bidirectionnelles. De plus, [Feld00] met en évidence la présence de la LRD dans le nombre d'arrivées de connexions TCP par unité de temps, sur toutes les échelles de temps au-delà d'une seconde et remarque que ce phénomène est lié au protocole http utilisé dans le web.

Par ailleurs, les auteurs de [Oliv01] montrent que les modèles de représentation des processus d'arrivée des flux sont similaires pour UDP et TCP, notamment pour leur caractère non Poisonnien.

En utilisant la double classification (souris/éléphants , tortues/libellules), l'auteur de [Larr05] a étudié les processus d'arrivée des flux de chaque classe, en utilisant des traces de trafic collectées dans le cadre du projet de métrologie français Metropolis. Il a établi que la dépendance à long terme est bien marquée au niveau des arrivées des flux souris (paramètre de Hurst estimé à 0,805 sur les traces analysées), alors que le processus d'arrivée des flux éléphants est assimilable à un processus Poissonnien (paramètre de Hurst estimé à 0,57 sur les traces analysées). Il a également établi que la différence en termes de LRD entre les flux tortues et les flux libellules n'était pas aussi importante et déduit, ainsi, que la durée des flux a beaucoup moins d'influence sur la LRD du trafic (observé au niveau flux) que le volume des flux. Il a, aussi, étudié les caractéristiques de la LRD par type d'application, il a montré qu'il n'existe pas d'application particulière à l'origine de la LRD du trafic global. En effet, toutes les applications classiques et émergentes y contribue, car chacune est capable dans certaines configurations d'utilisation de générer des flux éléphants.

4. Conclusion

La conception et la modélisation des systèmes de télécommunications a été construite sur l'identification de propriétés invariantes concernant notamment les processus d'arrivée des appels et la durée des sessions. Trouver les invariants utiles et fiables du trafic Internet a été plus difficile, en partie en raison de l'évolution et l'hétérogénéité de l'Internet lui-même [Floy01]. La synthèse des caractéristiques du trafic Internet, présentée dans ce premier chapitre, a relevé la présence d'un nombre restreint d'invariants qui sont :

1. La présence de cycles journalier et hebdomadaire, dans la variation des débits mesurés en nombre d'octets, de paquets ou de flux.
2. La présence d'une asymétrie entre les deux sens du trafic sur un lien. Toutefois le coefficient d'asymétrie varie beaucoup selon le type (lien d'accès, inter-POP ou de peering) et la capacité du lien.
3. La grande majorité du trafic utilise le protocole TCP ; en effet, même le trafic à contraintes temps réel privilégie l'utilisation de ce protocole pour sa fiabilité.
4. Les distributions de la taille et de la durée des flots sont à décroissance sous exponentielle voire lourde.
5. La distribution de la taille des paquets est dominée par la présence de deux modes correspondants aux paquets de taille minimale (40 octets) et à celui des paquets de taille maximale (1500 octets)
6. Les arrivées des paquets sur un lien présentent un caractère dépendant à long terme et peuvent être modélisées par un processus auto-similaire voire multifractal.

À part ces quelques caractéristiques largement admises comme invariantes (quoique certaines sont sujettes à des controverses limitées), toutes les autres sont très variables selon l'endroit et / ou la période de mesure, d'où la difficulté de mettre en place de modèles de trafic descriptifs fiables et efficaces.

Par ailleurs, la mise en place d'une infrastructure de mesures et d'analyses, au niveau du réseau national universitaire RNU se justifie par le besoin de confronter les résultats de cette étude au cas particulier du RNU. La description de cette infrastructure, des traces collectées et des résultats des analyses réalisées feront l'objet du troisième chapitre.

En attendant, le deuxième chapitre, s'intéresse aux intrusions via le réseau mondial et expose l'état de l'art des outils et techniques permettant de les détecter et/ ou de les surveiller.

Chapitre 2: Détection d'anomalies dans les réseaux Internet

Internet est devenu une infrastructure essentielle pour les entreprises, les administrations, les gouvernements et même les particuliers qui sont amenés à l'utiliser quotidiennement pour échanger des informations ou effectuer des transactions financières et autres. De ce fait, la sûreté du fonctionnement et la sécurité des communications sur le réseau mondial sont devenus des enjeux, de plus en plus critiques, qui concernent tous les utilisateurs de ce réseau.

En effet, dans le rapport [NARC07], les auteurs affirment que la sécurité du cyberespace est un enjeu national important pour la sécurité du USA[1]. Ils expliquent que les menaces liées au réseau mondial concernent aussi bien l'infrastructure Internet elle-même que l'exploitation de ce réseau universel pour lancer des attaques vers des cibles stratégiques. Ils détaillent comment ces menaces pourraient être concrétisées par des personnes mal intentionnées ou des organisations terroristes, afin de compromettre la sécurité des USA et porter atteinte à leurs intérêts stratégiques.

Dans ce chapitre, nous exposerons les menaces liées à l'Internet susceptibles de dégrader la qualité de service offerte par les réseaux de backbones. Ensuite, nous présenterons l'état de l'art des techniques de détection d'intrusion au niveau des cœurs de réseaux.

1. Menaces liées à Internet

Les menaces, liées à l'Internet, sont en constante augmentation ces dernières années. De plus, les techniques d'attaques sont devenues, de plus en plus, diverses et sophistiquées et les motivations des pirates sont devenues multiples (voir [Fran07 et Bail05] pour plus de détails sur la cybercriminalité). En effet, les intentions des premiers auteurs d'attaques et de logiciels malveillants, se propageant via Internet, étaient plutôt peu nuisibles ; la plupart étaient des pirates talentueux (hackers) ou des apprentis pirates informatiques (Script Kaddies) qui voyaient dans le fait d'attaquer un site, de s'introduire sur un serveur pour récupérer des informations ou de trouver une faille dans un système de sécurité, un moyen de montrer l'étendue de leurs connaissances pour les premiers et simplement de s'amuser, aux dépens des autres, pour les seconds. Aujourd'hui, les auteurs d'attaques et de logiciels malveillants sont, pour la plupart, attirés par le gain financier et utilisent des techniques

[1] Il s'agit d'un rapport de plus de 300 pages établi par une commission spéciale pour le développement de la recherche sur la sécurité du cyberespace. Il présente une étude détaillée des vulnérabilités liées à Internet et aux technologies de l'information et de la communication (TIC).

élaborées, dont notamment les réseaux de zombies ou botnets. En effet, un botnet est un réseau de machines sous le contrôle distant d'un ou plusieurs malfaiteurs, qui les utilisent pour perpétuer des activités illicites à l'insu de leurs propriétaires.

Storm botnet est l'un réseaux de zombies les plus connus notamment à cause de sa longévité et de la virulence des attaques qui l'ont exploité. En effet, il a été utilisé dans diverses activités criminelles, dont notamment des compagnes de phishing[2], de propagation d'adware[3] et de lancement d'attaques de déni de service distribuées (DDOS) vers divers sites de commerce électroniques et d'organismes actifs dans le domaine de la sécurité informatique[Storm]. De plus, il utilise un mode de propagation diversifié, faisant appel à plusieurs variantes du vers informatique du même nom : Storm worm. En outre, contrairement aux autres réseaux de zombies utilisant les canaux IRC[4], les auteurs de ce réseau de zombies, utilisent une technologie de contrôle complètement décentralisée rendant ainsi la désactivation de ce réseau complexe.

La propagation des vers informatiques et les attaques de dénis de service réseaux constituent, avec les activités des réseaux de zombies, les principales menaces liées à l'Internet qui intéressent les opérateurs de réseaux. En effet, ces trois menaces ont, contrairement aux attaques classiques (intrusion, vol d'informations confidentielles, etc.), des effets néfastes qui dépassent leurs cibles et dégradent la qualité de service des réseaux qu'elles traversent. En effet, elles ont la particularité de générer un volume de trafic indésirable important, pouvant saturer les liens et / ou congestionner les équipements actifs des réseaux traversés.

Face à ces menaces, les opérateurs de réseaux ont adopté, d'abord, une politique de sécurité consistant à se focaliser sur la protection de leurs infrastructures et serveurs stratégiques. Ils ont considéré que les attaques, perpétuées via leurs réseaux, doivent être détectées et arrêtées au niveau des machines et des réseaux cibles. Ils ont défendu le principe selon lequel, les backbones Internet devaient se focaliser sur le transport des informations, au détriment de l'analyse et du filtrage, afin d'augmenter leurs performances. Mais, après les attaques massives qui ont perturbé, d'une manière significative le fonctionnement le réseau mondial à partir de 2001, il y a eu une prise de conscience générale de l'insuffisance des moyens disponibles et de la nécessité de coopération entre opérateurs pour faire face à ces menaces [Stan02].

[2] Le phishing est une technique utilisée par les fraudeurs pour collecter des informations personnelles leur permettant d'usurper les identités de leurs victimes. Elle consiste à faire croire à la victime qu'elle s'adresse à un tiers de confiance (banque, site marchand) afin de lui soutirer des numéros de carte de crédit ou de comptes bancaires.

[3] Un adware est un logiciel propriétaire gratuit dont le financement est assuré grâce à la publicité qu'il affiche.

[4] IRC : Internet Relay Chat

Ainsi, cette prise de conscience s'est traduite par le démarrage récent de plusieurs projets de coopération entre réseaux connectés à Internet. L'objectif de ces initiatives est d'évaluer, d'une manière permanente, les menaces liées au réseau mondial et de proposer des mécanismes capables de réduire leurs effets néfastes. Parmi ces projets, nous citons le projet Shadowserver de la fondation du même nom [Shadow], le projet « network telescope » de l'association CAIDA et le projet Internet Storm Center de l'organisation SANS [ISC]. Les deux premiers utilisent des réseaux de pots de miel, installés partout dans le monde, pour collecter et analyser le trafic lié à la propagation des vers informatiques et des réseaux de zombies. Le troisième projet, Internet Storm Center (ISC), collecte et analyse continuellement des journaux d'évènements de firewalls et de systèmes de détection d'intrusion. Ces derniers sont déposés, sur le site web du projet, par les administrateurs de réseaux désirant participer au projet et contribuer ainsi à construire une image globale des menaces liées au réseau mondial.

Concernant la recherche, il a eu récemment proposition de plusieurs approches de détection d'intrusion, adaptées aux besoins des réseaux de backbone. L'état de l'art de ces approches constitue l'objet de la plus grande partie de ce chapitre. Pour ce faire, nous commencerons par une description de l'architecture globale des systèmes de détection d'intrusion et des deux approches qu'ils utilisent, à savoir la détection des malveillances et la détection d'anomalies. Puis, nous montrerons que la seconde approche (détection d'anomalies ou approche comportementale) est nettement plus intéressante pour une installation au niveau des cœurs de réseaux. Par la suite, nous exposerons une classification des techniques de détection d'anomalies avant d'aborder la problématique de validation de ces systèmes. Enfin, dans la dernière partie de ce chapitre, nous exposerons les techniques de pots de miel et de télescopes réseau en expliquant leurs intérêts pour la surveillance des réseaux et la caractérisation des menaces les affectant.

2. Systèmes de Détection d'Intrusion (IDSs)

Un système de détection d'intrusion (ou IDS : Intrusion Detection System) est un mécanisme destiné à repérer les tentatives d'intrusions sur une cible surveillée (un réseau ou un hôte) [Ande80]. Anderson définit dans [Ande80] une intrusion comme toute tentative de pénétrer un système d'informations par des personnes non autorisées ou d'accéder à de plus hauts privilèges pour des utilisateurs locaux. Alors que [Head90] adopte une définition plus large, il considère comme intrusion ou attaque[5] toute tentative de compromettre l'intégrité, la confidentialité ou la disponibilité de la cible surveillée.

Pour la détection des attaques locales, c'est à dire celles qui n'utilisent pas le réseau, mais qui sont perpétuées en utilisant un compte local sur la machine victime, les HIDS (Host based Intrusion

[5] L'annexe B, donne un aperçu sur les différentes techniques d'attaques via les réseaux.

Detection System) sont utilisés. En effet, un HIDS est un mécanisme permettant de surveiller et d'analyser l'activité d'une machine (les processus en cours, les ressources matérielles utilisées, les commandes lancées par les utilisateurs ainsi que les horaires et les durées des connexions) afin de détecter les tentatives d'attaques.

Pour détecter les attaques perpétuées via le réseau, les systèmes de détection d'intrusion réseau (Network Based Intrusion Detection System : NIDS) sont utilisés. En effet, ces derniers collectent en permanence le trafic, véhiculé par le réseau surveillé et l'analysent afin d'y déceler la présence ou pas de tentatives d'attaques.

2.1 Architecture globale d'un IDS

L'étude des techniques de détection d'intrusion, réalisée par Axelsson dans [Axel98], décrit une architecture générale des systèmes de détection d'intrusion. Cette dernière, donnée par la Figure 2.1, permet de mettre en évidence les différents modules qui composent un tel système indépendamment de la source de données (machine ou réseau) qu'il utilise et de la technique de détection qu'il adopte.

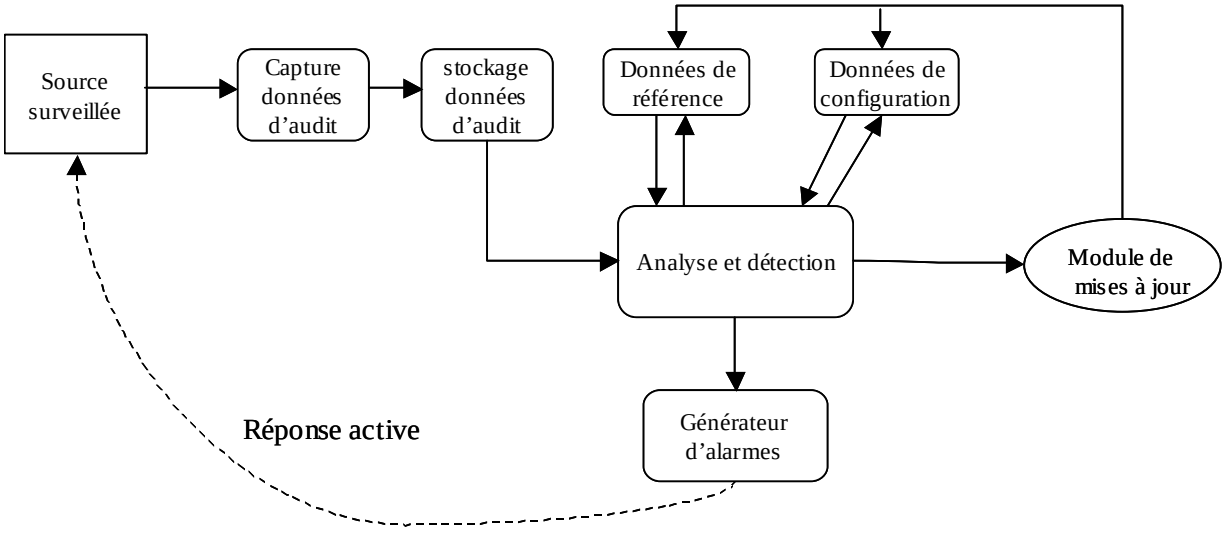

Figure 2.1 : Architecture globale d'un IDS

Ainsi, selon [Axel98], tout système de détection d'intrusion comporte les modules suivants :

- Module de capture de données d'audit : ce module sert à collecter, en permanence, les informations nécessaires à la détection des intrusions. La provenance de ces données (un hôte, un lien ou à plusieurs liens) et leur niveau de granularité varient d'un IDS à l'autre. En effet, bien que la plupart des NIDSs reposent sur la collecte de données détaillées par flux traversant la cible auditée [Snort, Paxs99, Shyu03 et Carm07], certains NIDSs utilisent des

données ayant une grosse granularité, telles que les séries temporelles du nombre d'octets, de paquets ou de flux sur un lien [Brut00 et Lakh04a].

- Module de stockage de données d'audit : les systèmes de détection d'intrusion sauvegardent les données d'audit collectées, pour qu'elles puissent être examinées par les ingénieurs de sécurité en cas de besoin. Le volume de ces données est généralement important, par conséquent, la réduction de ces données continue d'être un problème de recherche crucial pour la conception d'IDSs [Patc07].
- Données de référence : ce module contient les données de référence qui servent pour la détection des intrusions. En effet, pour détecter les tentatives d'attaques, les systèmes de détection d'intrusion comparent l'activité actuelle au niveau de la cible auditée à une référence. Pour construire les données de références, les IDSs font généralement recourt soit à des experts, soit encore à des techniques d'apprentissage automatique (machine learning) afin de permettre au système de construire lui-même les données de référence dont il a besoin.
- Module d'analyse et de détection : ce module constitue le cœur même des IDSs, c'est là où sont implémentés les algorithmes de détection d'attaques. Historiquement, les travaux de recherche sur la détection des intrusions sont axés sur ce module, d'où l'existence d'une multitude d'algorithmes de détection d'intrusion. Ces derniers peuvent être classés en deux grandes approches (qui seront détaillées plus loin) : détection de malveillances et détection d'anomalies. Les premiers utilisent comme données de référence des signatures d'attaques, alors que, les seconds utilisent des modèles décrivant le comportement normal de la cible auditée.
- Module de configuration du système : ce module contient les données relatives à la configuration de l'IDS lui-même.
- Module de génération d'alarmes : En cas de détection d'intrusion, ce module permet de générer une alarme sous forme d'un mail à l'administrateur du réseau par exemple, d'enregistrer un évènement dans un journal ou encore de lancer une action automatique afin de stopper l'intrusion. Dans ce dernier cas, on parle de système de détection d'intrusion actifs ou de système de prévention d'intrusions (Intrusion Prevention System IPS).

Dans la suite de ce chapitre, nous allons nous limiter aux systèmes de détection d'intrusion réseau (NIDS), car seules les attaques via le réseau nous intéressent.

2.2 Approches pour la détection d'intrusion via les réseaux

Il existe deux grandes familles de systèmes de détection d'intrusion réseaux : les systèmes de détection de malveillances et les systèmes de détection d'anomalies.

Les systèmes de détection des malveillances (Misuse detection en anglais) sont basés sur une approche dite par scénarios ou par signatures. Par contre, les systèmes de détection d'anomalies sont basés sur une approche dite comportementale.

2.2.1 Approche par signatures

Les systèmes de détection des malveillances réseaux s'appuient sur la connaissance préalable des techniques d'attaques utilisées par les attaquants. En effet, à chaque type d'attaque, un système de détection de malveillances associe une signature, qu'il stocke dans sa base de signatures et qui correspond à l'ensemble des caractéristiques discriminantes permettant d'identifier cette attaque. Ainsi, une signature d'attaque peut consister en la présence d'une chaîne alphanumérique particulière dans la charge utile d'un paquet ou d'une taille particulière de paquet.

La plupart des systèmes de détection des malveillances utilisent des algorithmes de recherche de motifs (Pattern matching) leur permettant de rechercher, dans le trafic qu'ils inspectent, la présence des motifs[6] prédéfinis dans leurs bases de signatures. C'est la technique utilisée par la plupart des IDSs actuels, dont par exemple, les logiciels libres Bro [Paxs99] et Snort [Snort]. Le premier est un NIDS issu du monde de la recherche (développé par Vern Paxson à l'université de Berkeley). Alors que le second est un NIDS libre disponible en deux versions : une version gratuite et une autre payante. Cette seconde version est largement déployée dans les réseaux d'entreprises.

Contrairement à Snort, qui utilise la recherche simple de motifs, Bro utilise une technique évoluée de recherche de motifs (statefull patern matching). En effet, la recherche simple de motifs consiste à inspecter les paquets, les uns indépendamment des autres, à la recherche de signatures connues, alors que la recherche évoluée consiste à analyser le trafic afin de reconstituer les flux qui le composent puis inspecter ces flux à la recherche de signatures d'attaques connues. Cette dernière technique, permet au système de détection d'intrusion de résister aux tentatives d'évasion par fractionnement du trafic (subdiviser le trafic d'attaque en plusieurs paquets).

Les IDSs à base de signatures sont aujourd'hui incontournables, à cause de leur capacité de détection face à une multitude de types d'attaques notamment applicatives. De plus, ils permettent d'identifier, pour chaque attaque détectée, sa cible (l'adresse IP), son type (la vulnérabilité utilisée et la finalité de l'attaque) et son origine (la ou les adresses IP sources qui ont généré le trafic d'attaque), offrant ainsi à l'administrateur du réseau des informations lui facilitant la prise de décision (quelles mesures adopter pour arrêter l'intrusion). Mieux encore, aujourd'hui la plupart des IDSs commerciaux intègrent un module pare-feu ou permettent une interaction avec un pare-feu existant, afin de générer automatiquement des actions défensives lorsqu'une intrusion est détectée.

[6] Un motif est une chaîne de caractères ou suite d'octets utilisant les expressions régulières.

Toutefois, les IDSs à base de signatures présentent quelques inconvénients qui limitent leur efficacité et surtout leur adaptabilité à une utilisation au niveau des backbones. Le premier de ces inconvénients, réside dans leur inefficacité face aux balayage de ports (scans) et aux attaques de déni de service distribuées (DDoS), car, ces deux types d'attaques ne peuvent être identifiées par une signature de paquet [Komp07]. Pour combler ce manque, la plupart des IDSs à base de signatures utilisent des règles heuristiques afin de détecter les scans et les attaques DDoS. Par exemple, dans Snort, une adresse IP source est marquée comme scanneur si elle essaie de se connecter, durant un intervalle de Y secondes, à au moins X ports destinations différents sur une même adresse cible (ou bien si elle essaie de se connecter au même port destination de X adresses IP différentes)[7] [Snort et Roes99].

Le deuxième inconvénient de l'approche par signature réside dans son incapacité de détecter les nouvelles attaques et les anciennes attaques modifiées. En effet, comme les signatures d'attaques stockées dans la base de signatures d'un tel système sont statiques, il suffit à un attaquant d'apporter une petite modification à une ancienne attaque pour qu'elle ne soit plus décelée. Pour remédier à ces problèmes, les systèmes de détection des malveillances, doivent mettre à jour régulièrement leurs bases de signatures pour y inclure les signatures des nouvelles attaques, ainsi que les signatures des anciennes attaques modifiées. Ceci implique une charge de travail importante pour les développeurs de ces systèmes. Les administrateurs de réseaux, adoptant ces systèmes, se trouvent dans l'obligation de souscrire des abonnements coûteux pour télécharger régulièrement les mises à jour de la base des signatures, sans quoi les IDSs qu'ils ont mis en place se trouveraient obsolètes et inefficaces. De plus, les délais entre l'apparition d'une nouvelle attaque, la publication d'une signature adéquate et la mise à jour de la base de signatures, laissent temporairement les réseaux utilisant ces IDSs sans protection.

Enfin, les systèmes de détection de malveillances voient leurs bases de signatures grossir rapidement[8] ce qui a pour effet de dégrader les performances de détection de ces systèmes, car ils doivent comparer, en temps réel, le trafic à un nombre toujours croissant de signatures d'attaques. De ce fait, ces IDSs sont plutôt adaptés à la détection des intrusions au niveau des réseaux d'extrémité où le trafic est généralement faible.

2.2.2 Approche comportementale

Les systèmes de détection d'intrusion utilisant l'approche comportementale, appelés aussi ADS (Anomaly Detection Systems), ont été proposés, pour la première fois, par Anderson dans [Ande80],

[7] Dans la configuration par défaut de Snort, X=4 et Y=3 secondes.

[8] A titre d'exemple, le site web de Snort [Snort] publie toutes les semaines, une nouvelle mise à jour de sa base de signatures. De plus, en six mois (entre octobre 2007 et mars 2008) la taille de la base des signatures de Snort a pratiquement doublé.

puis repris par Denning [Denn87], qui a exploré l'idée selon laquelle les tentatives d'attaques sur un système informatique provoquent une utilisation anormale de ce système. Par conséquent, elles peuvent être détectées comme des déviations significatives du comportement de l'utilisateur ou du réseau par rapport à un comportement normal prédéfini. Denning a proposé la modélisation du comportement normal d'un utilisateur, par un ensemble de modèles statistiques décrivant des métriques relatives à l'utilisateur, aux applications et aux ressources système utilisées.

Depuis le travail de Denning, plusieurs techniques de détection d'intrusions par approche comportementale ont été proposées. [Patc07] en dresse l'état de l'art et les classe en trois catégories : les techniques basées sur les méthodes statistiques, les techniques utilisant les méthodes de fouille de données et enfin les techniques utilisant les algorithmes d'apprentissage automatique.

Toutes ces techniques peuvent être classées en deux classes : les approches paramétriques et celles non-paramétriques. En effet, les approches paramétriques (telles que celles basées sur les méthodes statistiques) utilisent comme référence des modèles paramétriques décrivant le comportement normal du réseau. En contre partie, les approches non-paramétriques (telles que celles utilisant les méthodes de classification ou d'apprentissage automatique) se basent sur une période d'apprentissage et n'émettent aucune hypothèse concernant les modèles décrivant le comportement normal.

Indépendamment de leurs techniques, les IDS utilisant l'approche comportementale présentent deux avantages indéniables. Le premier réside dans leur capacité à détecter les nouvelles attaques. Le second réside dans le fait que ces systèmes nécessitent très peu de mises à jour une fois leurs paramètres sont bien calibrés.

Concernant leurs inconvénients, la majorité des ADSs souffrent d'un manque d'évaluation, ce qui explique, d'après [Patc07], le nombre très réduit d'implémentations commerciales utilisant l'approche comportementale. De plus, la plupart n'incluent pas de module pour l'identification de l'origine des anomalies détectées. Par conséquent, ils ne permettent pas une intégration avec un pare-feu afin d'arrêter les attaques détectées. Toutefois, ils peuvent être considérés que comme des moyens de notification précoces permettant de détecter les nouvelles attaques. Par la suite, les administrateurs pourront utiliser des techniques d'analyses poussées afin d'identifier l'origine des anomalies et décider des réactions à entreprendre.

3. Techniques de détection d'anomalies de trafic

Plusieurs études métrologiques du trafic Internet ont montré la présence assez fréquente de certaines variations inhabituelles et significatives au niveau de ses caractéristiques ; ce sont les anomalies de trafic. Pour détecter ces anomalies, puis en identifier les causes, les administrateurs de réseaux ont souvent recours à des méthodes adhoc [Barf02].

Les systèmes de détection d'anomalies de trafic (appelés en anglais Trafic Anomaly Detection System TADS) ont pour but d'automatiser tout ou partie du processus de détection et d'identification des anomalies affectant le trafic réseau. Il s'agit de systèmes de détection d'intrusion réseau (NIDS) basés sur l'approche comportementale.

Nous proposons dans ce qui suit, une classification des techniques utilisées par ces systèmes selon le type des données d'audit qu'ils utilisent. En effet, nous distinguons : Les TADSs basés sur les métriques macroscopiques de volume, les TADSs utilisant les métriques de distribution et ceux utilisant des données détaillées.

3.1 Techniques utilisant des données détaillées

Les techniques de détection d'anomalies utilisant des données détaillées, ont besoin de collecter tous les paquets traversant le lien ou le nœud surveillé à la manière des systèmes de détection d'intrusion à base de signatures.

L'approche dite «protocol anomaly detection » représente un sous-groupe des techniques de détection d'anomalies à partir de données détaillées. Cette approche nécessite la définition préalable par des experts des profils normaux des protocoles. Pour ce faire, les auteurs des articles [Seka02, Yoo04] proposent d'extraire à partir des RFCs et/ou de documents techniques publiés par des éditeurs de logiciels les différents cas d'utilisation des protocoles (IP, TCP, UDP et applicatifs) et de les modéliser par des automates à états finis. La détection d'anomalies de protocoles revient ainsi à inspecter le trafic réseau pour vérifier la conformité de l'utilisation qu'il fait des protocoles à ceux prédéfinis. Ces approches considèrent qu'un protocole décrit un ensemble fermé (limité) de comportements normaux et que les attaques résultent d'une violation de ces comportements. A titre d'exemple, ils peuvent détecter comme anomalie tout paquet dont l'un des champs dépasse une taille maximale prédéfinie ou toute requête http ne respectant pas un bon format prédéfini.

Or le fait que la plupart des protocoles présentent des zones d'ombre[9] et que des différences plus au moins importantes existent entre leurs diverses implémentations, rend le traitement de tous les cas possibles fastidieux ; D'où la difficulté d'implémenter une telle approche au niveau des réseaux de backbones. En contre partie, cette approche s'apprête bien à une implémentation sous forme de « reverse proxy » spécifiques aux applications (le plus souvent mail ou web)[10].

Bien que, les systèmes de détection d'anomalies de protocoles puissent détecter les nouvelles attaques et nécessitent peu de mises à jour en comparaison avec les techniques de détection de

[9] C'est le cas lorsque la spécification du protocole ne traite pas tous les cas possibles.

[10] C'est le cas de Portus Application system (http://www.lsli.com/pad.whitepaper.pdf, dernière consultation Juillet 2009)

malveillances ; ils échouent à détecter un grand nombre d'attaques. Il s'agit de toutes les attaques qui respectent leurs spécifications des protocoles comme par exemple des balayages de ports.

D'autres systèmes de détection d'anomalies à partir de données détaillées existent, la plupart procèdent à la reconstitution de tous les flux composant ce trafic avant d'utiliser des techniques de classification supervisée ou des techniques de clustering, afin de classifier les flux obtenus en deux ou plusieurs classes[11].

Les systèmes, utilisant les techniques de classification supervisée, ont besoin d'une phase d'apprentissage, durant laquelle, ils utilisent une trace de trafic étiquetée, c'est à dire composée par des flux normaux et des flux d'attaques préalablement identifiés par d'autres mécanismes (le plus souvent, l'étiquetage se fait par des experts humains). Par contre, les systèmes utilisant les techniques de clustering ont besoin, durant leur phase d'apprentissage, d'une trace composée exclusivement de flux normaux.

Les travaux de [Shyu03, Laza03, Bouz04 et Carm07], illustrent l'approche de détection d'anomalies à partir de données détaillées. En effet, la méthode de détection proposée par Shyu, dans [Shyu03], nécessite de décrire chaque connexion TCP ou UDP collectée par 41 attributs, parmi lesquels la durée de la connexion ou le nombre d'octets envoyés. De plus, elle consiste à effectuer une analyse en composantes principales de la matrice des connexions (formée par 41 colonnes et N lignes, avec N le nombre total de connexions) afin de conserver les composantes principales majeures expliquant 50% de la variance des données originales et les composantes principales mineures dont les valeurs propres sont inférieures à 0,2. Le principe de détection revient, alors, à marquer comme anomalie toute connexion, qui une fois projetée selon les axes principaux retenus, correspond à un point excentrique par rapport au centre des données.

L'approche utilisée dans [Bouz04] est quelque peu différente, bien qu'elle se base sur les mêmes données détaillées provenant de la trace publique [KDD99] et utilise l'analyse en composantes principales pour réduire la dimension des données nécessaires à la détection. En effet, dans [Bouz04], seules les composantes principales majeures sont retenues, de plus deux techniques de classification supervisées sont utilisées : la méthode du voisin le plus proche et la méthode des arbres de décision.

En conclusion, nous estimons que l'inconvénient principal des approches basées sur les données détaillées réside dans la taille et la nature même de leurs données. En effet, dans [Carm07], chaque connexion est décrite par plus de 200 attributs, alors que dans [Shyu03 et Bouz04] chaque connexion est décrite par 41 attributs. De plus, la plupart des attributs utilisés (tels que la durée de la connexion ou le nombre de paquets envoyés) ne peuvent être calculés qu'après la fin de la connexion. Ainsi,

[11] S'ils différencient les anomalies selon leurs types, alors ces systèmes aboutissent à une classification des flux en plusieurs classes. Sinon, ils aboutissent à deux classes de flux : flux normaux et flux anormaux.

ces systèmes ne peuvent être utilisés pour la détection d'anomalies en temps réel, et se limitent, de ce fait, à une utilisation en différé pour l'analyse post-mortem des traces de trafic.
En outre, ces méthodes (à l'exception des techniques de détection d'anomalies de protocoles) n'exploitent pas la sémantique des données utilisées. De ce fait, leur efficacité est tributaire du degré de similitude entre les données d'apprentissage et celles réelles.

3.2 Techniques utilisant des métriques de volume

Les anomalies de volume sont des variations inhabituelles et significatives qui apparaissent au niveau des métriques relatives au volume du trafic telles que le nombre de paquets, d'octets ou de flux envoyés / ou reçus sur un lien par unité de temps. Dans [Braf02], les auteurs définissent quatre types d'anomalies de volume de trafic. Il s'agit des foules subites[12], des attaques via le réseau, des pannes au niveau du réseau et des problèmes au niveau du système de mesures lui-même. Ainsi, nous constatons qu'à part les attaques réseau et, dans une moindre mesure, les foules subites, les deux autres types ne correspondent pas à des activités malicieuses ; elles concernent des évènements ayant peu d'intérêt pour l'administrateur du réseau. En effet, les pannes et les problèmes au niveau du système de mesures sont efficacement détectées par d'autres techniques, notamment en utilisant le protocole SNMP ou les outils de mesures actives. De plus, à part les attaques qui génèrent un volume de trafic considérable, notamment les dénis de service par inondation ou les balayages de ports massifs engendrés par la propagation de vers informatiques, tous les autres types d'attaques ne peuvent être détectés en utilisant les métriques de volume.

Malgré ces limites, la détection d'anomalies basée sur les métriques de volume continue de susciter l'intérêt des chercheurs dans le domaine, notamment à cause du fait qu'elle se base sur des mesures faciles à collecter et peu volumineuses. En effet, les métriques de volume sont traditionnellement collectées au niveau des réseaux d'opérateurs grâce aux systèmes de gestion de réseaux ; de plus la collecte de ces métriques est peu coûteuse en ressources matérielles puisque la taille des traces obtenues ne dépend pas du volume du trafic véhiculé par le réseau mais uniquement de la fréquence de collecte de ces métriques.

Les systèmes de détection d'anomalies de volume, proposés dans l'état de l'art, se différencient selon l'origine des données qu'ils traitent. En effet, certains détectent les anomalies affectant le volume du trafic au niveau d'un seul lien [Brut00, Balf02 et Borg07], alors que d'autres s'intéressent aux anomalies qui modifient la structure de corrélation entre les trafics de plusieurs liens [Lakh04 et Ring07].

[12] Une foule subite est le fait qu'un grand nombre de personnes sollicitent, durant le même laps du temps, le même service Internet. C'est le cas par exemple de l'engouement observé lors des événements du 11 septembre 2001 vers les sites d'informations en continu.

3.2.1 Détection d'anomalies de volume au niveau d'un lien

La plupart des approches de détection d'anomalies de volume au niveau d'un lien réseau [Brut00, barf02 et Borg07] se basent sur l'utilisation d'un modèle statistique décrivant le volume du trafic sur le lien supervisé. Le calcul des paramètres du modèle de référence nécessite une phase d'apprentissage durant laquelle le trafic réseau est supposé normal. De plus, la conception de tels systèmes se heurte à un défi de taille qui réside dans la construction de modèles de trafic capables de modéliser sa variabilité normale, tout en restant sensibles aux variations anormales engendrées par des activités malicieuses. Pour y parvenir, plusieurs modèles de trafic issus des études métrologiques ont été exploités par les TADSs proposés dans la littérature.

Dans [Brut00], l'auteur considère comme anomalie, tout intervalle du temps, durant lequel l'écart entre le trafic mesuré sur le lien supervisé et celui prédit par le modèle de prévision (construit durant la période d'apprentissage) , dépasse un seuil fixe prédéfini. Plus concrètement, il s'agit de collecter périodiquement le nombre de paquets (ou d'octets) envoyés (ou reçus) par lien. Puis, pour chaque série temporelle obtenue, un modèle de prévision du trafic, supposé décrire sa variabilité normale au cours du temps, est construit en utilisant l'algorithme de Holt-Winters[13] (qui est une généralisation de la méthode de lissage exponentiel[14]). Enfin, il s'agit de marquer comme anomalie toute observation, pour laquelle l'écart le volume du trafic mesuré et celui prédit, dépasse le seuil de détection fixé.

L'avantage principal de cette méthode réside dans son implémentation dans l'outil RRDtool[15] , permettant ainsi une exploitation facile par les administrateurs de réseau. Concernant ses limites, la méthode proposée par Brutlag est incapable de détecter les anomalies de courtes durée (moins d'une heure).

Dans [Barf02], les auteurs proposent de collecter périodiquement pour chaque lien supervisé le nombre de paquets envoyés/reçus, le nombre d'octets envoyés/reçus et le nombre de flux IP. Ensuite, ils utilisent la décomposition en ondelettes pour filtrer chacune de ces séries temporelles selon trois bandes de fréquences : la bande de hautes fréquences, celle des fréquences moyennes et

[13] L'algorithme de Holt-Winters consiste à décomposer une série temporelle en trois composantes : une référence (baseline), une composante linéaire (linear trend) et composante saisonnière. Puis d'utiliser la méthode de lissage exponentiel pour prédire chacune de ces trois composantes, ainsi la valeur prédite de la série temporelle est la somme des valeurs prédites pour ces trois composantes.

[14] La méthode de lissage exponentiel permet de prédire la valeur d'une série temporelle à l'instant t+1 en utilisant sa valeur mesurée à l'instant t, ainsi que la valeur prédite pour cet instant.

[15] RRDtool est l'acronyme de Round Robin Database, il s'agit d'un système libre pour sauvegarde et visualisation de données chronologiques (séries temporelles)

celle des basses fréquences. Puis, ils calculent la variance locale des données filtrées, si cette variance dépasse un seuil prédéfini, alors une anomalie est détectée.

Enfin, la méthode de détection d'anomalies de [Borg07] permet de détecter les anomalies via une analyse conjointe du trafic agrégé sur plusieurs intervalles de temps. Pour ce faire, elle se base sur la modélisation de la série temporelle nombre de paquets par intervalle de temps, collecté au niveau d'un lien donné, par un modèle non gaussien et à longue mémoire (utilisant les lois Gamma et Farima), et ce pour différents intervalles d'agrégation de trafic (allant du 1ms à 10s). Puis, elle calcule la distance Kullback-Leibler entre les paramètres calculés pour une fenêtre quelconque w et ceux estimés sur une fenêtre de référence composée uniquement de trafic normal ; Si cette distance dépasse un seuil fixe prédéfini alors la fenêtre w est marquée comme anormale, autrement elle est considérée normale. La particularité de la technique proposée par Borgnat [Borg07] réside dans le fait qu'elle s'intéresse aux anomalies de courtes durées et ce contrairement à [Brut00 et Barf02] qui s'intéressent uniquement aux anomalies ayant des durées supérieures à 1 heure.

En conclusion, l'inconvénient majeur des approches de détection des anomalies de volume au niveau d'un lien réside dans le fait qu'elles analysent les séries temporelles (relatives au trafic sur ce lien) les unes indépendamment des autres. Ainsi elles ne permettent pas d'exploiter les relations de corrélations qui existent entre les différentes séries supervisées (nombre de paquets, d'octets envoyés/reçus). Or, l'expérience montre que c'est grâce à la comparaison des séries temporelles les unes par rapport aux autres que les administrateurs de réseaux arrivent à détecter les anomalies et à déterminer leurs types. Par exemple, une augmentation du nombre de paquets en entrée sur lien, non accompagnée d'une augmentation équivalente de celui des paquets en sortie ou de celui des octets reflète le plus souvent des attaques de déni de service ou des balayages de ports massifs. De la même façon, le fait que ces approches analysent les séries temporelles, relatives aux différents liens d'un même réseau, les unes indépendamment des autres, ne leur permet pas d'offrir à l'administrateur une vue globale sur les anomalies affectant son réseau. De plus, ces méthodes n'exploitent pas les relations de corrélation qui existent entre les séries relatives aux différents liens du réseau.

3.2.2 Détection d'anomalies au niveau d'un réseau

Lakhina a proposé dans [Lakh04], une méthode de détection d'anomalies de volume à l'échelle d'un réseau étendu. Pour cela, il utilise conjointement toutes les séries temporelles représentant le nombre de flux OD[16] transmis (durant dix minutes) par chaque lien du réseau surveillé. Il obtient ainsi un espace à N dimensions (avec N égal au nombre du liens du réseau). En utilisant l'analyse en composantes principales, la méthode de détection proposée dans [Lakh04], décompose l'espace

[16] Un flux OD (origine/destination) est défini par une paire : adresse IP source (SIP), adresse IP destination (DIP) ; il est composé par tous les paquets ayant comme adresse IP source SIP et comme adresse IP destination DIP.

représentant ces mesures en deux sous-espaces : l'espace représentant le trafic normal et celui représentant le trafic résiduel. Le premier espace est composé par les axes principaux majeurs, alors que le second est composé par les axes principaux mineurs. Ainsi, chaque vecteur de dimension N, correspondant à une fenêtre de temps w, est décomposé en un vecteur normal et un vecteur résiduel. Par la suite, une anomalie est détectée dans la fenêtre de temps w, si la norme L2[17] du vecteur résiduel est supérieure à un seuil de détection prédéfini. Autrement la fenêtre w serait considérée normale.

L'approche de [Lakh04] présente deux avantages ; le premier réside dans son exploitation des inter-corrélations entre les volumes du trafic sur les différents liens d'un réseau. Le second réside dans le fait qu'elle ne dépend pas d'un modèle statistique pour le trafic normal.

3.3 Techniques utilisant des distributions du trafic

La détection d'anomalies dans les distributions de trafic, repose sur l'article [Kohl06] (dont la première version a été publiée en 2002). En effet, les auteurs de [kohl06] ont utilisé plusieurs traces de trafic collectées entre 1998 et 2001, et ont pu remarquer que la répartition des adresses IP destinations, présentes au niveau du trafic collecté sur un lien, est caractérisée par l'existence de trois types de zones: une zone dense, une zone d'adresses éparpillées et une zone vide. Sur la base de ces constatations, ils ont proposé un modèle multifractal à deux paramètres pour représenter cette structure d'adresses IP. Ensuite, ils ont montré que la structure des adresses IP, présentes dans le trafic d'un site donné, est stable sur les petites échelles de temps et qu'elle constitue une empreinte du trafic de ce site. De plus, ils ont remarqué que cette structure change d'une manière significative, lors de la propagation des vers informatiques.

Plus concrètement, la Figure 2.2 [Lakh05] illustre l'effet d'un balayage de ports (scan vertical) sur la distribution du trafic par numéro de port destination (Portdst) et sur celle par adresse IP destination (IPdst). Ainsi, nous pouvons remarquer que, la présence d'un scan vertical, a eu pour effet de rendre la répartition du trafic par Portdst plus dispersée et celle du trafic par IPdst plus concentrée autour d'un nombre restreint d'adresses IP.

En plus des scans verticaux, [Lakh05] décrit six autres types d'anomalies de trafic pouvant être exposés par l'analyse des répartitions du trafic. En effet, il s'agit des types d'anomalies suivants :

- Les attaques de déni de service (DoS) ;
- Les balayages de numéros de ports (ou scans verticaux) ;

[17] La norme L2 (ou norme euclidienne) d'un vecteur x à n dimensions est $\left\|(x_1,x_2,\ldots,x_n)\right\|=\sqrt{(x_1^2+x_2^2+\ldots+x_n^2)}$.

- Les balayages d'adresses IP (ou scans horizontaux), principalement engendrés par la propagation de vers informatiques via les réseaux ;
- Les foules subites ;
- Les flux alpha, engendrés par le transfert de fichiers très volumineux ;
- Les sauts de trafic, observés suite à la survenue de pannes au niveau du réseau ou à cause des maintenances programmées ;
- et les flux point à multipoints composés par plusieurs flux envoyés à partir d'une même adresse IP source vers plusieurs destinations distinctes. Ces flux sont généralement engendrés par les serveurs de distribution de contenus.

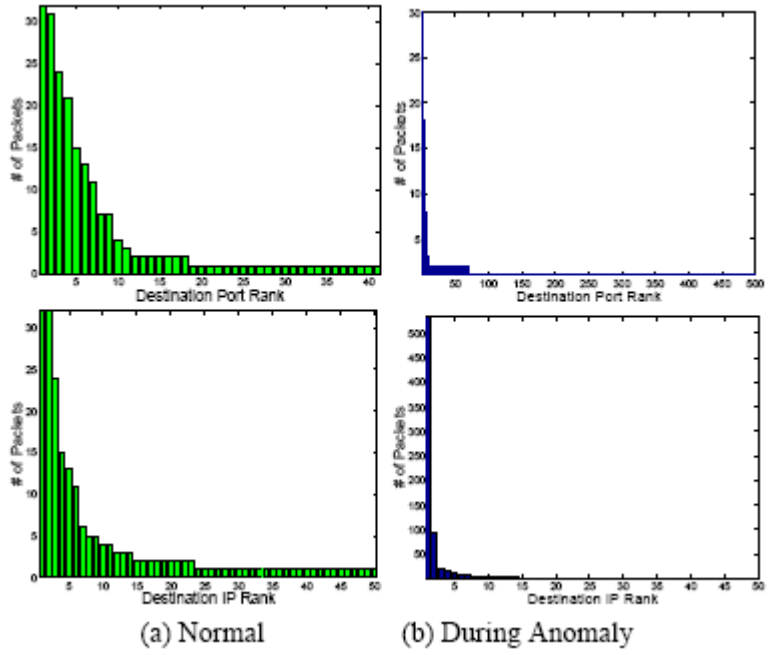

Figure 2.2: Distributions du trafic [Lakh05]

Pour détecter les types d'anomalies, décrits plus haut, certains travaux dont ceux de [Lakh05, Gu05 et Li06] utilisent des approches entropiques, alors que d'autres utilisent les sketchs [Li06 et Borg07a].

En effet, dans [Lakh05, Gu05 et Li06] l'entropie de Shannon[18] est utilisée comme mesure statistique agrégée des distributions du trafic. Plus concrètement, le problème de détection d'anomalies affectant les distributions de trafic est ramené à la détection des anomalies dans les séries

[18] Nous rappelons que, l'entropie de Shannon (notée H(P)) d'une distribution empirique P relative à une variable aléatoire X, est un nombre positif qui ne dépend que de P (i.e qu'il est indépendant des valeurs prises par X). De plus, l'entropie de Shannon mesure le degré de dispersion/concentration de la distribution P. en effet, elle est maximale lorsque P est uniforme et nulle lorsque P ne peut prendre qu'une seule valeur.

temporelles composées par les valeurs de H(P), calculées sur des fenêtres de temps de taille fixe. Pour ce faire, les techniques de détection d'anomalies de volume, décrite dans le paragraphe précédent, sont utilisées. Par exemple, dans [Lakh05], les auteurs proposent d'utiliser la méthode, détaillée dans [Lakh04] afin de détecter les anomalies, affectant les distributions de trafic collectées au niveau de N liens selon les quatre variables suivantes :IPsrc, IPdst, Portsrc et Portdst.

L'approche proposée dans [Gu05] se base sur le calcul de l'entropie relative (appelée aussi divergence Kullback-Leibler) entre une distribution calculée sur une fenêtre de temps quelconque w et celle de référence calculée en utilisant le principe de l'entropie maximale. Par la suite, le principe de détection repose sur le choix d'un seuil fixe pour la distance de Kullback. En effet, si la distance de Kullback calculée, entre une fenêtre quelconque w et la distribution de référence, dépasse un seuil de détection prédéfini, alors la fenêtre en question serait considérée anormale. Autrement, elle sera marquée comme normale.

La deuxième approche, utilisée pour la détection des anomalies dans les distributions de trafic, se base sur la représentation de ces distributions par des sketchs[19] spécialement conçus. En effet, dans [Borg07a], l'algorithme de détection d'anomalie proposé, repose sur la combinaison des sketchs et de l'approche [Borg07] précédemment utilisée pour la détection d'anomalies de volume. Dans [Li06] ; l'auteur montre que l'utilisation de sketchs multiples (représentant les adresses IP sources, les adresses IP destinations, les ports sources et les ports destinations) permet de représenter la variabilité normale du trafic, ainsi que les variations dues à la présence d'anomalies. Pour détecter les anomalies affectant les sketchs, il calcule l'entropie de chaque sketch, puis utilise l'algorithme de détection proposé dans [Lakh04].

Enfin, bien que les méthodes de détection d'anomalies dans les distributions de trafic nécessitent plus de ressources matérielles en comparaison avec celles basées sur les métriques de volume, elles permettent un gain en temps de traitement et en mémoire vive considérable, par rapport aux méthodes utilisant des informations détaillées [Krish03]. De plus, elles permettent d'exposer un plus grand nombre d'attaques, en comparaison avec les approches basées sur les métriques de volume.

3.4 Conclusion

Nous avons différencié les systèmes de détection d'anomalies du trafic (TADS), décrits dans la littérature, selon deux critères : le type des données d'audit qu'ils utilisent (données détaillées, métriques de volume et distributions du trafic) et leur provenance (un lien ou plusieurs liens). Certes

[19] Le sketch d'un flux de données est un résumé compact de ces données. Il est obtenu par projection aléatoire du flux de données en utilisant un nombre réduit de fonctions de hachage indépendantes. Il permet d'estimer la fréquence d'un élément quelconque (surtout pour les éléments ayant une haute fréquence) dans le flux de données qu'il représente.

cette classification n'expose pas la diversité des techniques de détection utilisées dans ce domaine qui sont, d'ailleurs, issues de diverses disciplines. En effet, selon [Chan09], les techniques de détection d'anomalies proviennent de plusieurs domaines : apprentissage automatique, statistiques, fouille de données, théorie spectrale, théorie de l'information.

Toutefois, la classification adoptée se justifie par le fait que le choix des métriques à surveiller influe considérablement sur les types des anomalies pouvant être détectées. En effet, les métriques de volume permettent la détection des anomalies de forte intensité (par rapport au volume du trafic total), alors que les métriques de distribution permettent la détection d'anomalies modifiant la répartition du trafic indépendamment de leurs intensités. De plus, le fait de faire la détection à partir de données provenant de plusieurs liens permet d'exposer les anomalies affectant les inter-corrélations entre les trafics véhiculés par différents liens du réseau.

Par ailleurs, nous avons distingué les techniques de détection d'anomalies paramétriques et celles dites non paramétriques. Les premières [Brut00, Braf02, Borg07 et Borg07a] nécessitent la construction d'un modèle statistique pour le trafic normal de référence et considèrent comme anomalie toute déviation par rapport à ce modèle ; par conséquent, leurs performances de détection dépendent énormément du calibrage des paramètres du modèle de référence utilisé. Alors que, les seconds [Shyu03, Lakh05] ne se basent sur aucun modèle statistique du trafic ; c'est le cas notamment sur ceux basés sur les techniques de classification et de clusturing. Le fait que les modèles du trafic, aujourd'hui disponibles, sont non seulement limités à une description grossière du trafic[20] mais aussi difficiles à calibrer [Bena04, cair05], rend les systèmes de détection d'anomalies du trafic paramétriques peu efficaces.

4. Évaluation des systèmes de détection d'anomalies

Valider un système de détection d'anomalies est une tache complexe qui nécessite de mesurer sa performance de détection face à des anomalies légitimes et illégitimes diverses. Pour cela, nous avons besoin de définir des métriques de performance globales et des métriques de performance par type d'attaque.

4.1 Métriques d'évaluation des ADSs

4.1.1 Métriques standards

Un système de détection d'anomalies peut se trouver confronté aux quatre situations différentes résumées dans le Tableau 2.1 :

[20] En effet, la plupart des modèles proposés dans la littérature décrivent uniquement les temps d'arrivée des paquets et / ou des flux et ne permettent pas de représenter leurs contenus.

- L'échantillon observé ne présente pas d'anomalie et l'algorithme de détection le marque avec le label « Normal » ; c'est un vrai négatif « True Negatif ».
- L'échantillon observé ne présente pas d'anomalie mais l'algorithme de détection le marque avec le label « anomalie » ; c'est un faux positif « Flase Positif ».
- L'échantillon observé représente réellement une anomalie et l'algorithme de détection le marque avec le label « anomalie », c'est un vrai positif « True Positif ».
- L'échantillon observé représente réellement une anomalie, mais l'algorithme de détection le marque avec le label « Normal », c'est un faux négatif « False Negatif ».

		Label prédit de l'échantillon observé	
		Normal	Anomalie
Label réel de l'échantillon	Normal	Vrai négatif (TN)	Faux positif (FP)
	Anomalie	Faux négatif (FN)	True Positif (TP)

Tableau 2.1: Métriques standards pour l'évaluation des ADSs

Les métriques principales sont le taux de détection, le taux de fausses alarmes et la précision. Elles sont définies comme suit

$$Taux\ de\ détection\ = \frac{TP}{TP+FN}$$

$$Taux\ de\ \ fausses\ alarmes\ = \frac{FP}{TN+FP}$$

$$\text{Pr}écision\ = \frac{TP}{TP+FP}$$

4.1.2 Métriques additionnelles

Toute technique de détection d'anomalies associe un score (correspondant généralement à une mesure de distance par rapport à la référence utiliséé) à chaque connexion traitée ou échantillon observé. Lorsque, le score devient supérieur à un seuil de détection prédéfini, le système de détection d'anomalies considère qu'une anomalie est survenue.

La Figure 2.3, montre l'évolution au cours du temps du score associé à chaque échantillon observé en présence d'une attaque qui s'étend sur plusieurs échantillons. En effet, les lignes verticales représentent les échantillons observés, la ligne discontinue représente la courbe réelle de l'attaque (qui est à un durant l'attaque et nulle ailleurs) et la courbe verte, dessinée en trait plein, correspond au score associé à chaque échantillon observé.

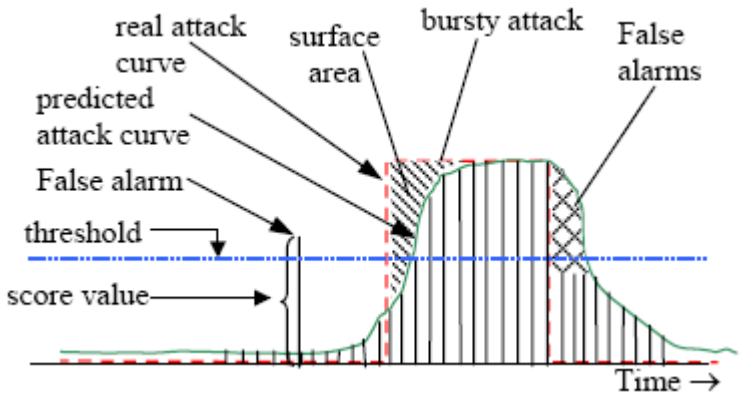

Figure 2.3: Évolution du score calculé lors d'une attaque [Laza03]

Ces deux courbes (courbe de l'évolution du score et courbe réelle de l'attaque) nous permettent de dériver des métriques d'évaluation additionnelles. En effet, plus la surface entre la courbe de l'attaque réelle et celle de l'attaque prédite (surface hachurée en \\\ dans la Figure 2.3) est petite, meilleur est l'algorithme de détection d'anomalies (de même pour la surface hachurée en x).

La Figure 2.4 définit deux nouvelles métriques:

- Burst Detection Rate (bdr) est défini pour chaque rafale, il représente le ratio entre le nombre total d'échantillons intrusifs qui ont un score supérieur au seuil de détection (n_{di}) et le nombre total d'échantillons réellement intrusifs dans la rafale d'attaque (N_{bi}). $bdr = \frac{n_{di}}{N_{bi}}$

- Le temps de réponse ($t_{reponse}$) représente le temps écoulé depuis le début de l'attaque jusqu'au moment où un échantillon est correctement classé comme intrusif.

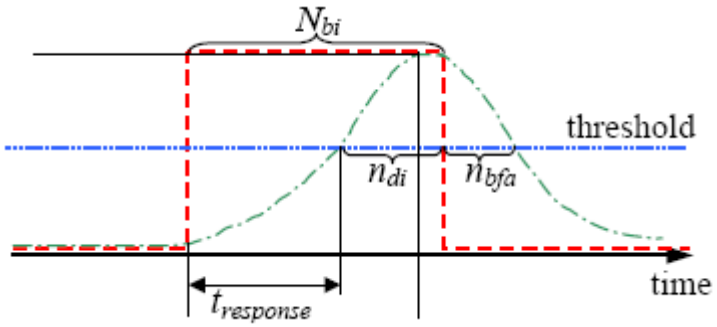

Figure 2.4: Représentation graphique des métriques additionnelles [Laza03]

4.2 Discussion autour du besoin en traces étiquetées

Le calcul des métriques d'évaluation, vues précédemment, nécessite de connaître, avec précision l'ensemble des évènements anormaux; ce qui revient à disposer de traces de trafic correctement étiquetées.

Une technique classique pour marquer les anomalies contenues dans les traces de trafic réel est de demander à un expert en sécurité de détecter tous les évènements suspects dans la trace. Pour cela, il utilise plusieurs outils adhoc et inspecte visuellement les traces selon plusieurs angles de vues (répartitions du trafic par adresse IP ou par numéro de ports, recherche des connexions TCP les plus volumineuses, …). Bien que ces spécialistes soient très expérimentés, il arrive souvent qu'ils manquent une anomalie ou bien qu'ils en ajoutent une, alors que le trafic est parfaitement normal. Mais, malgré l'imperfection de leur étiquetage, les traces manuellement étiquetées, sont très utiles pour l'évaluation des systèmes de détection d'anomalies puisqu'elles reflètent l'ensemble des anomalies qu'un algorithme de détection doit pouvoir reconnaître pour être aussi performant qu'un expert humain. Toutefois, les traces réelles manuellement étiquetées contiennent généralement un nombre réduit et peu diversifié d'anomalies, ce qui limite leur intérêt.

À ce jour, très peu de traces de trafic sont librement mis à la disposition de la communauté scientifique, pour des raisons évidentes de confidentialité. De plus, les traces publiques contenant des anomalies bien documentées sont encore plus rares. En effet, les seules traces publiques étiquetées, que nous avons pu trouver sont les suivantes :

- Les traces de KDD99 [KDD99] crées spécialement pour l'évaluation d'outils de détection d'anomalies lors du « Third International Knowledge Discovery and Data Mining Tools Competition » ;
- Et les traces DARPA98 [DARPA98], datant des années 1998-2000, elles sont issues de simulations de trafic réseau (pour la génération du trafic normal) et d'attaques expérimentales [Mchu00].

Bien que ces traces aient été utilisées pour la validation d'un grand nombre d'IDSs issus de la recherche académique, elles sont aujourd'hui obsolètes car trop anciennes pour refléter la nature du trafic Internet actuel et la diversité des attaques dont est le véhicule.

Face à la non disponibilité de traces publiques récentes étiquetées, la plupart des techniques de détection d'anomalies proposées dans la littérature ont été validées en utilisant un nombre réduit de traces de trafic réel manuellement étiquetées.

Une approche alternative proposée dans [Auss07], pour la validation des outils de détection d'anomalies, consiste à créer des anomalies artificielles sur un réseau opérationnel, puis de collecter le trafic résultant. L'intérêt principal de cette méthode est qu'elle permet de contrôler précisément les paramètres des anomalies générées (le volume, la durée ou encore le nombre de flux impliqués dans l'anomalie). Toutefois, ces traces sont généralement limitées à un nombre réduit de type d'anomalies ; par conséquent elles ne reflètent ni la diversité des anomalies réelles, ni leur structure complexe. Ainsi, dans [Auss07], seulement deux types d'anomalies ont été synthétisés : les attaques de déni de service par inondation et les foules subites.

Par ailleurs, parallèlement aux systèmes de détections d'anomalies, les administrateurs de réseaux font recours à d'autres techniques pour assurer une sécurité renforcée des réseaux Il s'agit des techniques de pots de miel et des télescopes réseaux qui seront détaillés dans le paragraphe suivant.

5. Télescopes réseaux et pots de miel

Les télescopes réseaux et les pots de miel se basent sur le fait que l'espace d'adressage IPV4 est partiellement utilisé et que les attaquants n'ont pas toujours assez d'informations leur permettant de distinguer les adresses IP valides de celles invalides ou inutilisées. Ils permettent via la collecte et l'analyse du trafic destiné aux adresses IP dites invalides ou « bogon » et celles dites « inutilisées », de relever la présence de tentatives d'attaques et de propagation de logiciels malveillants.

Les adresses IP invalides ou « bogon » sont des adresses IP réservées par l'IANA (Internet Assigned Numbers Authority) pour des usages spécifiques ; c'est le cas des adresses IP privées définies par le RFC 1918[21] ou encore des classes d'adresses D et E[22]. Dans cette catégorie, nous trouvons également les adresses IP non assignées par l'IANA à aucun Registre Internet Régional (RIR)[23]. Des listes détaillées et régulièrement actualisées de ces adresses sont publiées sur les sites web de quelques groupes de travail dont [Cidr]et [Complete]. Il apparaît, ainsi, que les adresses IP invalides représentent prés de 40% de l'espace d'adressage IP total.

Bien que les adresses invalides n'aient pas de raison d'être routées sur Internet, les études métrologiques montrent qu'il arrive fréquemment à ces adresses d'être routées sur certaines portions de l'Internet et d'être utilisées par des personnes ou des organisations malveillantes afin de conduire, d'une façon anonyme, des attaques de déni de service, des envois massifs de messages non sollicités et des activités de piratage [Feam05]. Cependant, les mêmes études montrent que les scanneurs automatisés et les vers informatiques évitent l'espace d'adressage invalide, car de toute évidence il ne comprend aucune machine vulnérable. Enfin, nous notons que l'exploitation des blocs d'adresses invalides pour des finalités malveillantes et le fait que la liste des ces adresses soit relativement stable, a poussé les administrateurs de réseaux à mettre en place des règles de filtrage rejetant le trafic lié à ces adresses afin de diminuer le risque de leur exploitation malveillante.

Par ailleurs, les adresses « inutilisées » sont des adresses IP allouées à organisme particulier qui ne les exploite pas ; ainsi la présence de trafic utilisant ces adresses est par nature suspecte. Cependant,

[21] La RFC 1918 définit un espace d'adressage privé permettant à toute organisation d'attribuer des adresses IP aux machines de son réseau interne. Il s'agit des plages d'adresses 10.0.0.0/8, 172.16.0.0/13 et 192.168.0.0/16.

[22] Les adresses de la classe D sont utilisées pour les communications multicast, alors que ceux de la classe E sont utilisées pour la recherche.

[23] Un Registre Internet Régional (RIR), alloue les adresses IP dans sa zone géographique à des opérateurs réseau et des fournisseurs Internet. Il existe aujourd'hui cinq RIRs.

une différence essentielle existe entre les adresses IP invalides et celles inutilisées : en effet, les premières sont publiées sur Internet, alors que les secondes sont inconnues sauf pour les administrateurs du réseau concerné. De ce fait, les scanneurs automatisés, les vers informatiques et les pirates cherchant à collecter des informations sur leurs prochaines victimes, ne peuvent pas éviter ces adresses, contrairement aux adresses invalides. D'où, l'intérêt de collecter le trafic destiné à ces adresses afin d'en tirer des informations sur les activités malveillantes. C'est là le rôle des télescopes réseaux.

En effet, un télescope réseau est un système installé à l'entrée d'un réseau donné afin de collecter, en permanence, le trafic destiné aux adresses IP inutilisées dans ce réseau (parfois, ces systèmes collectent également le trafic des adresses IP invalides). Il permet via l'analyse des traces collectées d'étudier les différents événements suspects affectant ce réseau. De plus, si la taille du télescope, c'est à dire le nombre d'adresses inutilisées surveillées est important, alors il permet d'étudier les évènements d'envergure qui affectent le réseau mondial dans sa globalité, notamment la propagation des vers informatiques et les attaques de déni de service[Moor04].

Par contre, un pot de miel (honeypot) est un système volontairement vulnérable destiné à attirer et à piéger les pirates informatiques en émulant le fonctionnement de véritable machine de production ; or un pot de miel n'est qu'un leurre. Plus concrètement, un pot de miel est un télescope réseau actif qui interagit avec les pirates afin d'en tirer des informations sur leurs outils, tactiques et motivations. C'est donc par ce biais que les pots de miel et les télescopes réseau aident à renforcer la sécurité des réseaux. Il est à signaler qu'il existe deux types de pots de miel, ceux à faible interaction et ceux à forte interaction. Les premiers permettent, via l'émulation de systèmes d'exploitation et de services Internet, une interaction limitée avec les attaquants potentiels ; alors que les seconds consistent à mettre en place réellement des systèmes d'exploitation et des services réseaux vulnérables afin d'avoir une interaction complète avec les éventuels attaquants.

Le trafic collecté par les télescopes réseau et les pots de miel a été observé et caractérisé par de nombreuses études, dont notamment [Moor01, Moor02, Stan02, Moor03, Zesh03 Pang04 et Yegn04]. Dans [Pang04], les auteurs ont analysé le trafic collecté par deux télescopes réseau installés à l'entrée de réseaux universitaires et l'ont dénommé « backgroud radiation » ou rayonnement de fond. En effet, ils ont décomposé ce trafic par protocole et par application et analysé ses variations au cours du temps avant de constater que les vers informatiques, les balayages de ports automatisées et les scans des « autorooter » (des logiciels malicieux semblables à des vers, mais sans auto propagation) dominent largement ce trafic.

Par ailleurs, dans [Moor01], les auteurs se sont intéressés particulièrement au trafic « backscatter » (rétro-diffusion) collecté par les télescopes réseaux. Il s'agit du trafic de réponse envoyé par les

machines victimes d'attaques de déni de service réseau de type « Syn floading »[24]. Ils ont caractérisé la distribution des adresses IP victimes de ces attaques, les types de services utilisés par les attaquants et ont pu déduire la prévalence des attaques de déni de service dans le réseau mondial.

En adoptant une méthodologie similaire, les travaux de [Moor02, Stan02, Moor03 et Zesh03] se sont intéressés aux phénomènes de propagation de vers informatiques via le réseau mondial tels que Code Red I, Code Red II et Slammer ; ils ont analysé leurs trafics et modélisé leurs différentes stratégies de scan.

Dans [Caso05], les auteurs ont exploité des traces de trafic collectées par des télescopes réseau afin d'avoir un aperçu des régions cachées d'Internet et d'estimer le nombre de réseaux utilisant le mécanisme de translation d'adresses (NAT : Network Address Translation).

En exploitant les pots de miel, les auteurs de [Krei04] ont proposé un système de génération automatique de signatures d'attaques afin de permettre aux systèmes de détection de malveillances de disposer rapidement de signatures pour les nouvelles attaques. En effet, l'algorithme proposé utilise des techniques de recherche de motifs[25] et d'analyse protocolaire afin d'extraire automatiquement des signatures d'attaques à partir de traces collectées par un ensemble de pots de miel. Dans [Yegn05], les auteurs reprennent la même idée et proposent un système de génération automatique de signatures utilisant une technique de classification non supervisée de flux.

Enfin, nous notons que les trafics collectés par les télescopes réseaux et les pots de miel ne sont pas toujours d'origine malicieuse, puisqu'ils peuvent refléter des mauvaises configurations ou des bogues logiciels. C'est le cas par exemple lorsque des postes clients sont configurés avec une adresse de serveur de noms (DNS) erronée ; résultant ainsi en l'envoi de requêtes DNS vers une adresse invalide à chaque tentative de connexion. Un autre exemple de trafic indésirable non malicieux est bien documenté dans l'article [Plon06]. En effet, l'université états-unisienne de Wisconsin a été victime en mai 2003, d'un large volume de trafic, persistant durant plusieurs jours et destiné à son serveur de temps (NTP). Ce volume de trafic a causé une saturation de la bande passante de l'université et a obligé les administrateurs de ce réseau à changer l'adresse de leur serveur NTP. Après investigation, l'origine de ce trafic a été déterminée ; il s'agit d'un bogue au niveau de centaines de milliers de routeurs ADSL destinés aux utilisateurs résidentiels d'une certaine marque à bas prix.

[24] L'attaque « syn floading » consiste à submerger la victime par un grand nombre de demandes de connexions TCP dont les adresses IP sources sont spoofées (fictives). L'annexe B fournit plus de détail sur cette attaque.

[25] L'algorithme utilisé est LCS Longest Common Substring qui permet de rechercher les similitudes entre plusieurs chaînes de caractères.

6. Conclusion

Nous avons exposé, dans ce chapitre, les menaces liées au réseau Internet et les approches de détection d'anomalies. Pour ce faire, nous avons adopté une classification des systèmes de détection d'anomalies basée sur le type des données qu'ils manipulent tout en discutant les différentes techniques de détection adoptées.

Nous avons également, exposé les potentialités offertes par les télescopes réseaux pour la caractérisation des menaces affectant les réseaux Internet (notamment la propagation des vers informatiques et les attaques de déni de services par inondation). En effet, bien que les télescopes réseaux et les pots de miel n'offrent aucune protection contre les attaques (puisqu'ils sont totalement passifs et que le trafic d'attaque peut ne pas les traverser[26]), les traces de trafic qu'ils génèrent constituent une source d'informations précieuse sur la nature des trafics d'attaques et les techniques des attaquants.

Avant d'aborder la problématique des anomalies affectant le réseau RNU et les solutions que nous proposons pour les détecter ; nous présenterons dans le chapitre suivant une étude métrologique du trafic dans le réseau RNU. Cette dernière révélera ses caractéristiques en les comparant avec l'état de l'art et montrera l'importance des anomalies qui l'affectent.

[26] C'est le cas quand l'attaquant connaît l'adresse IP de sa cible, il lui envoiera directement le trafic d'attaque ; par conséquent le télescope réseau ne verra rien.

Chapitre 3: Mesures et analyses du trafic dans le Réseau National Universitaire (RNU)

Étant donné que le trafic Internet est très variable entre sites et très évolutif dans le temps, les résultats obtenus par les groupes de travail dans le domaine, et que nous avons exposé dans le premier chapitre, ne peuvent être extrapolés au cas particulier d'un réseau national comme le RNU sans effectuer au préalable des mesures. C'est là, la première motivation de ce travail de thèse.

Le projet « Métrologie sur RNU », dans le cadre duquel s'effectue ce travail de thèse, a démarré en 2003 en collaboration avec le Centre de Calcul El Khawarizmi (CCK) et a suivi l'évolution du réseau RNU durant cinq ans. Dans ce chapitre, nous allons présenter notre contribution dans le cadre de ce projet. Pour cela, nous décrirons, dans un premier temps, le réseau national universitaire et l'infrastructure de mesures que nous avons déployé pour collecter le trafic véhiculé par le RNU. Nous présenterons, ensuite, les différentes traces de mesures passives collectées, avant de nous intéresser à la caractérisation globale de ce trafic au niveau paquet et au niveau flux. Par la suite, nous étudierons et quantifierons la dépendance à long terme dans les processus d'arrivée des paquets et des flux. Puis, nous analyserons le trafic TCP par type de connexion et étudierons les anomalies TCP. Enfin nous concluons ce chapitre par un rappel des principales caractéristiques du trafic RNU en les comparant avec les caractéristiques décrites dans la littérature.

1. Le Réseau National Universitaire (RNU)

Le Réseau National Universitaire (RNU) est crée en 1997 avec la désignation du Centre de Calcul el Khawarizmi (CCK) comme fournisseur de services Internet pour l'enseignement supérieur. Il connecte aujourd'hui l'ensemble des organismes relevant du Ministère de l'Enseignement Supérieur et de la recherche scientifique, à savoir les Établissements d'Enseignement et de Recherche (EER), les universités, la cité des sciences, le centre de publication universitaire, les offices des œuvres universitaires et les centres Internet mis à la disposition des étudiants [CCK]. Le Réseau RNU offre à ses utilisateurs (enseignants, étudiants et personnel administratif), en plus de l'accès à Internet, un ensemble de services notamment l'hébergement d'applications spécifiques telles que l'inscription en ligne, les résultats des concours nationaux, les services des œuvres universitaires et le catalogue collectif des ressources documentaires (BIRUNI). Le nombre des utilisateurs du réseau RNU n'a cessé de croître, comme le montre la Figure 3.1.

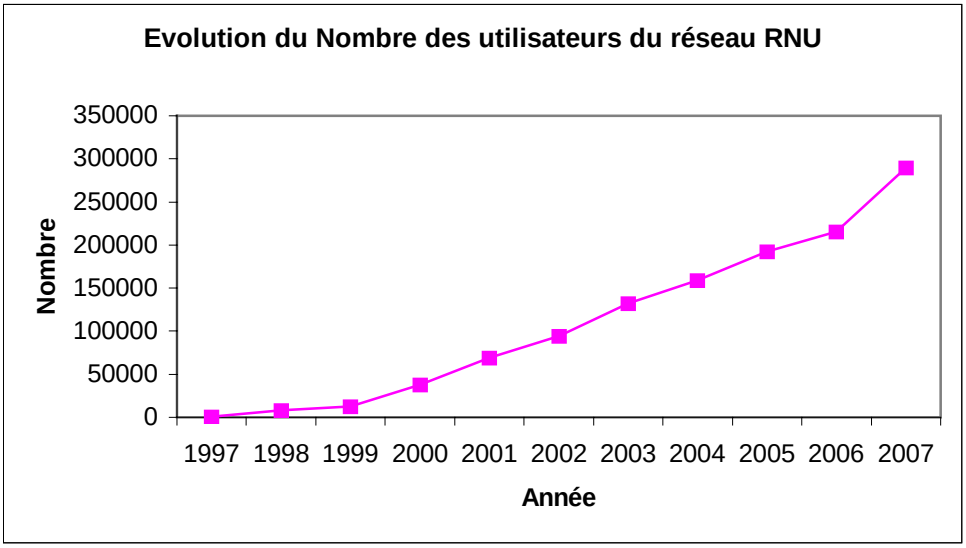

Figure 3.1 : Évolution du nombre des utilisateurs du RNU [CCK]

L'architecture du réseau RNU a connu plusieurs évolutions, pour répondre à l'augmentation du nombre d'utilisateurs et de leurs besoins. En effet, en 2003 à la naissance du projet « métrologie dans RNU » et au commencement de ce travail de thèse, le réseau RNU avait une architecture centralisée autour d'un nœud unique CCK-el Manar, comme le montre la Figure 3.2. Ce nœud, était le point de passage obligé de tout le trafic entre les institutions universitaires et Internet. En effet, toutes les institutions universitaires sont connectées à ce nœud, soit directement via des lignes spécialisées, soit indirectement via le backbone national.

Cette architecture centralisée a atteint ses limites et a été remplacée, dés Juin 2004, par une architecture plus sécurisée composée de deux nœuds (CCK-el Manar et CCK-el Kasbah). Cette dernière, illustrée par la Figure 3.3, permettait d'avoir des liaisons principales (représentées dans la figure par des flèches pleines) et des liaisons de secours (représentées par des flèches en pointillé). De plus, elle disposait de liaisons à haut débit (de type Ethernet à 100Mb/s) vers l'Internet et vers le backbone national.

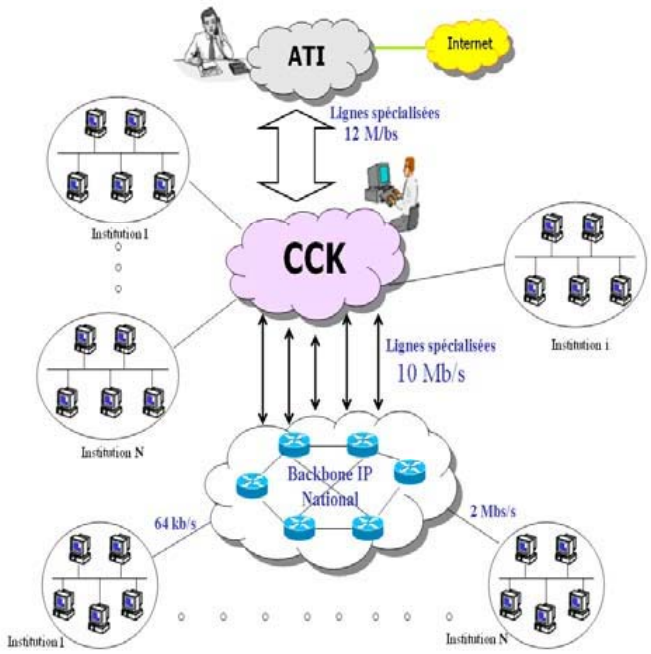

Figure 3.2 : Première architecture du réseau RNU

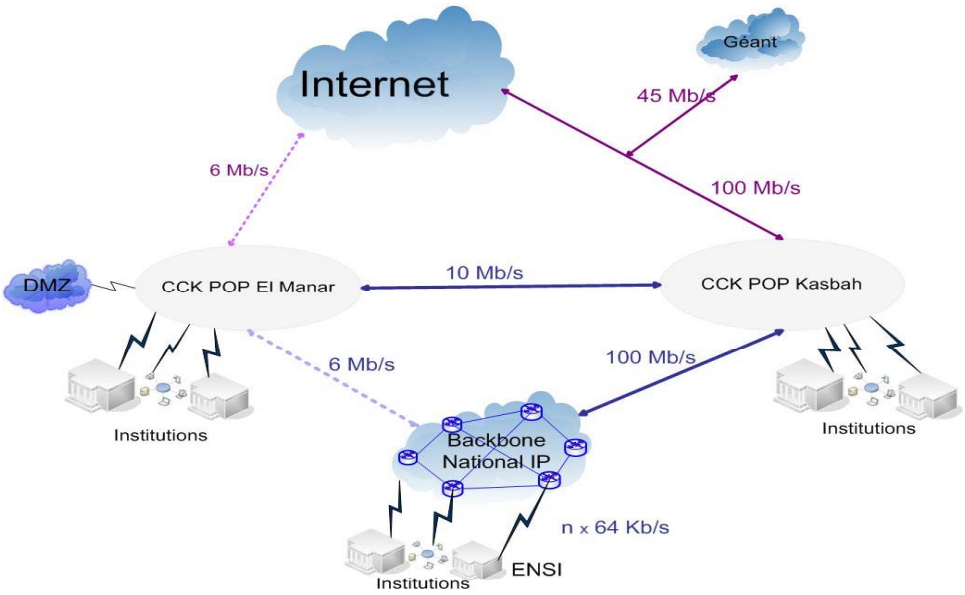

Figure 3.3 : Deuxième architecture du réseau RNU

L'évolution du réseau RNU s'est poursuivi par l'ajout de nouveaux nœuds et la migration vers des liaisons de type E3 (34 Mbits/s) et STM1 (155 Mbits/s) pour les connexions entre les nœuds du réseau. Aujourd'hui, le RNU est un réseau évolutif et performant à couverture nationale. Il dispose d'une architecture à deux niveaux, illustrée par Figure 3.4, et constituée de :

– **Les Nœuds CCK**, ce sont les points de présence du CCK dans les régions. On en compte aujourd'hui trois sur le grand Tunis (La Kasbah, El Manar et La Manouba). Tous ces nœuds sont

interconnectés entre eux pour former un réseau maillé à haut débit appelé **Epine dorsale du RNU.** Certains nœuds CCK disposent de connexions vers l'extérieur : d'une part vers le backbone de l'opérateur national Tunisie Télécom, d'autre part vers l'Internet commercial et vers le réseau de la recherche européen GEANT à travers l'Agence Nationale d'Internet (ATI).

– **Les Réseaux de Collecte**: un nœud CCK peut connecter à un ou plusieurs réseaux de collecte situés dans la région de ce nœud. Les réseaux de collecte permettent la connexion au RNU des institutions universitaires situées dans la région. Certaines institutions universitaires géographiquement éloignées de tous les nœuds CCK, ont une connexion via le backbone de Tunisie Télécom au nœud CCK le plus proche.

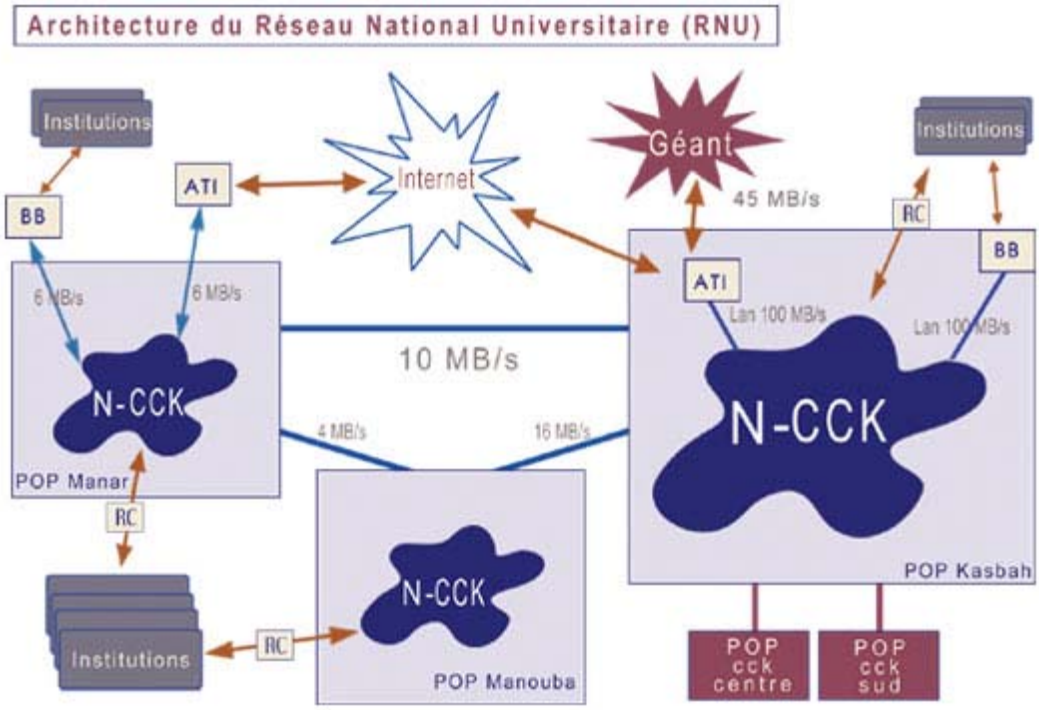

Figure 3.4 : Architecture actuelle du réseau RNU

2. Le projet « Métrologie dans RNU »

Le centre de calcul Khawarizmi, en tant qu'administrateur du réseau RNU, collecte en permanence des traces SNMP et netflow. Ces dernières servent principalement à surveiller l'évolution du trafic réseau et à détecter les pannes pouvant l'affecter. Pour ce faire, les outils MRTG, Tivoli et netflow analyzer sont utilisés pour générer des statistiques en ligne actualisées toutes les 5 minutes. Malheureusement, les traces brutes ne sont pas sauvegardées, ce qui ne permet pas d'avoir un historique détaillé ni de faire d'autres analyses plus poussées. D'où le besoin de déployer **une infrastructure de mesures et d'analyses** permettant de collecter le maximum d'informations sur l'état du réseau et la nature de son trafic. Cette infrastructure doit être à la fois flexible pour permettre l'ajout de nouveaux outils de mesure ou d'analyse et transparente pour le réseau et son

trafic ; c'est à dire qu'elle ne doit pas (ou peu) influer sur l'état du réseau. Elle doit permettre via l'analyse des traces de mesures de:

1. Connaître de manière fine la nature du trafic et sa répartition par protocole, par application, par adresse IP et par flux en général.
2. Étudier l'évolution de trafic dans le réseau RNU sur le court et le moyen termes.
3. Étudier les phénomènes de pannes affectant le réseau.
4. Étudier l'influence de l'émergence de nouvelles familles de menaces telles que les vers, les virus et les attaques sur les caractéristiques statistiques du trafic et sur les performances du réseau.
5. Mettre en place un outil de détection précoce des menaces pouvant perturber le réseau et dégrader ses performances.

Pour réaliser ces objectifs, le projet « Métrologie dans RNU » a été initié ; il s'agit d'une collaboration entre le CCK et le Laboratoire CRISTAL de l'ENSI (université de Manouba). Plus concrètement, il s'agit de mettre en place une infrastructure de mesures combinant les techniques passives et celles actives et développer des outils d'analyse en différé des traces du trafic collectées.

La plate-forme de mesures actives a été mise en place par Mr Hichem Ayari dans le cadre de son Mastère [Ayar05, Ayar05a et Ayar05b] portant sur la mesure et la caractérisation des phénomènes de pannes affectant le RNU.

La mise en place de la plate-forme de mesures passives et d'analyses en différé a fait l'objet du travail de Mastère de Melle Héla Boucetta [Bouc06, Bouc07] qui a collecté les traces T1 et T2 (voir Tableau 3.1).

Notre contribution dans le cadre de ce projet fera l'objet de la suite de ce chapitre. Elle concerne aussi bien les aspects les aspects de mise en place de la plate-forme que ceux relatifs à l'analyse des traces et l'interprétation des résultats.

3. Plate-forme de mesures passives : choix et mise en place

3.1 Description de la plateforme de mesures passives

La première contrainte pour la mise en place d'une infrastructure de mesures, vient du fait qu'il s'agit d'un réseau opérationnel, qui doit continuer à fonctionner même en cas de problèmes au niveau des sondes installées. C'est pourquoi, nous avons opté pour les plates-formes de mesures passives consistant à regarder le trafic transitant par le réseau afin d'en déduire ses caractéristiques.

Étant donné que le paquet est l'entité élémentaire mesurable dans un réseau IP, toutes les métriques relatives aux paquets, flux ou sessions, peuvent être **théoriquement mesurées, calculées ou dérivées** à partir des traces de trafic microscopiques composées par tous les paquets traversant le réseau. C'est pourquoi, nous avons opté pour **une plate-forme de mesures passives de niveau paquet,** permettant de collecter tous les paquets traversant un nœud du réseau et de leur ajouter les estampilles associées à leur date de passage.

Il existe plusieurs techniques de capture de paquets traversant un réseau donné. Parmi lesquelles :

1. Installer un « splitter » sur le lien physique de façon à dévier et amplifier une partie du signal transportant les données vers la sonde de mesure qui doit capter cette copie du trafic pour l'analyser. L'avantage principal de cette méthode est qu'elle est complètement transparente pour les équipements actifs tels que les commutateurs et les routeurs, c'est d'ailleurs pour cette raison que la plupart des compagnes de mesures passives citées dans le chapitre 1 l'ont adopté (en utilisant des splitter et des cartes DAG). Concernant les inconvénients, cette méthode est relativement coûteuse de plus elle ne peut être utilisée que par un opérateur qui construit lui-même ses liaisons et a ainsi un contrôle total sur elles, Or ceci n'est pas le cas du réseau RNU pour lequel toutes les liaisons sont louées à l'opérateur de télécommunications national ce qui limite considérablement les possibilités d'intervention sur les liens physiques.
2. Activer au niveau d'un équipement d'interconnexion (routeur ou pare-feu réseau), un module de collecte des flux transitant par ses interfaces: La plupart des constructeurs de matériel d'interconnexion proposent cette fonctionnalité et permettent d'utiliser un format standard pour l'enregistrement des traces collectées, IPFIX (IP Flow Information eXport). Toutefois l'expérience montre que l'influence de cette technique sur la performance des routeurs est non négligeable. De plus, elle ne produit que des mesures agrégées de niveau flux ce qui ne permet pas d'étudier les caractéristiques de trafic au niveau paquet.
3. Capter le trafic au niveau des commutateurs Ethernet en utilisant la technique de SPAN (Switched Port Analyzer). Cette technique permet de recopier sur un port donné tout le trafic destiné à (et/ou provenant de) un ou plusieurs autres ports. La sonde de mesure connectée au port SPAN peut ainsi surveiller le trafic provenant de n'importe quel port du commutateur sans perturber le fonctionnement du réseau.

C'est cette dernière technique, qui est adoptée pour la première plate-forme de mesures passives sur RNU. Elle a été installée pour collecter diverses traces de trafic sur RNU entre 2004 et 2006 ; elle est composée d'une machine de type PC sous Linux disposant d'un vecteur de disques SCSI de grande capacité et d'une carte réseau classique de type Ethernet 100 Mb/s connectée sur un port Ethernet d'un commutateur spécialement configuré, de façon que la sonde puisse recevoir une copie du trafic transitant par les interfaces Ethernet d'un ou de plusieurs routeurs de concentration de trafic.

Pour le logiciel, nous avons opté pour le système d'exploitation Linux, la librairie libre Libpcap (packet capture library) et les utilitaires Tcpdump [Tcpdump] et Ipsumdump [Ipsum]. Le choix de ces outils est motivé par le fait que Tcpdump est léger (puisqu'il est dépourvu d'interface graphique et qu'il n'effectue aucun traitement sur les paquets en dehors de l'ajout d'une estampille temporelle).

De plus, la solution adoptée est nettement plus économique que les deux autres décrites plus haut. En effet, le coût de mise en place d'une telle solution peut être limité à 2000 dinars, alors que celui d'une solution à base de cartes DAG s'élève à au moins à une dizaine de milliers de dollars [Dag].

Pour assurer la fiabilité et la qualité des traces de mesures, nous avons pris certaines précautions lors de la mise en place de la sonde, en s'inspirant des travaux de Vern Paxon [Paxs97] et de l'équipe du projet WAND en Nouvelle-Zélande [Clea00, Mich01a et Mich01b]. Ainsi pour limiter le volume des données enregistrées, nous avons choisi de collecter uniquement 68 octets par trame Ethernet observée ce qui correspond à un compromis entre la taille minimale des entêtes Ethernet, IP et TCP (14+20+20=54octets) et la taille maximale de ces dernières (14+60+60=134 octets) ce qui permet, en l'absence d'options IP et TCP, de capturer 14 octets de données en plus de toutes les entêtes protocolaires.

Pour assurer une bonne précision de l'estampille temporelle, il est nécessaire qu'elle soit ajoutée au moment même de la réception du paquet par l'interface réseau. Or, ceci n'est pas possible avec des cartes réseaux classiques, contrairement aux cartes de capture spécialisées (telles que les cartes DAG). En effet, lorsqu'une carte réseau classique reçoit des paquets, elle génère une interruption pour demander au noyau du système d'exploitation d'enregistrer les entêtes, grâce à la bibliothèque Libpcap, et d'ajouter l'estampille en se basant sur la variable TSC[1]. Comme le traitement de l'interruption générée par la carte réseau est dépendant de son pilote[2] et du système d'exploitation, l'ajout de l'estampille se trouve retardée ainsi plusieurs paquets peuvent être avoir la même valeur d'estampille surtout si le nombre de paquets reçus par la carte est important. D'après [Pasz02], la précision de l'estampille TSC ajoutée par Tcpdump est seulement de l'ordre de la milliseconde.
Par ailleurs, l'enregistrement des entêtes de tous les paquets transitant au niveau d'un réseau nécessite d'avoir une puissance de calcul et une capacité de stockage importantes [Mich01b]. Dans le cas du réseau RNU, nous avons défini l'espace de stockage et le débit de collecte des paquets nécessaires pour garantir que la sonde arrive effectivement à traiter tous les paquets qui la traversent. Ces besoins sont calculés dans les conditions extrêmes d'utilisation du réseau correspondants à des rafales de petits paquets de 40 octets chacun. Dans ces conditions la plate-forme de mesures doit enregistrer, pour chaque paquet IP de 40 octets, 8 octets pour l'estampille, 14 octets pour l'entête

[1] La variable TSC (timestamp counter) a une taille de 8 octets ce qui permet une résolution de 1ns.

[2] Généralement une interruption est envoyée par la carte réseau pour demander la lecture des paquets reçus, lorsque le tampon de la carte réseau est plein ou après l'écoulement d'un certain timeout prédéfini.

Ethernet ainsi que les 40 octets correspondant à la taille du paquet à enregistrer ; ce qui correspond à 62 octets de données.

Pour le cas d'une liaison symétrique à 12 Mb/s[3], le calcul montre que la plate-forme de mesures doit être capable de capter, de transférer vers la mémoire et ensuite vers le disque un débit maximal de 37,2 Mbits/s ce qui reste largement en dessous des performances d'un slot PCI standard 32 bits 66MHz (266 Moctets/s, soit 2128 Mbits/s) [Mich02b] sur lequel est connectée la carte réseau.

3.2 Description des traces collectées

Des traces de différentes durées ont été collectées entre mai 2004 et avril 2006 grâce à la sonde de mesures passives décrite plus haut (voir le Tableau 3.1 pour le détail des traces collectées).

Trace	**Période**	**Lien**	**Durée totale**	**Taille [Goctets]**	**Débit max [Mbits/s]**	**Paquets [10^6]**
T1	29 mai 2004	CCK-el Manar / ATI	24 h	8,6	12	129
T2	Du 10/08/04 au 17/08/04	CCK-el Manar / CCK-el Kasbah	7 jours	14,2	10	213
T3	Du 05/04/06 au 06/04/06	CCK-el Kasbah / ATI	24 h	25,4	50	284
T4	Du 07/04/06 au 08/04/06	CCK-el Kasbah / ATI	24 h	31,8	50	356

Tableau 3.1 : Traces collectées

Chaque trace est composée par plusieurs fichiers d'une heure chacun, et couvre ainsi une durée minimale de 24 heures. Toutes les traces sont bidirectionnelles, c'est à dire qu'elles contiennent le trafic des deux sens du lien surveillé. L'emplacement de la sonde de mesures a suivi l'évolution de l'architecture du réseau RNU et a permis de collecter le trafic au niveau de différents liens.

La trace T1, collectée alors que l'architecture du réseau RNU était centralisée (Figure 3.2), contient les entêtes de tous les paquets envoyés par tous les utilisateurs du RNU vers Internet et inversement. La trace T2, collectée alors que l'architecture du réseau RNU était celle de la Figure 3.3, contient tous les paquets traversant le lien inter-POPs CCK-el Manar et CCK-el Kasbah. La nature du trafic sur ce lien est fondamentalement différente du trafic de la première trace, puisqu'il est composé de connexions vers les serveurs hébergés au niveau de ces nœuds (essentiellement des serveurs web, mail et proxy), ainsi que du trafic d'accès à Internet des institutions universitaires de la région el Manar.

[3] 12 Mbits/s correspond à la capacité de la liaison entre CCK-el Manar et l'agence tunisienne d'Internet en 2004. Cette liaison transportait tout le trafic du réseau RNU vers Internet et inversement.

Enfin, les traces T3 et T4 collectées alors que le réseau RNU avait l'architecture actuelle (Figure 3.4), contiennent le trafic transporté par le lien reliant CCK-el Kasbah à l'Agence Tunisienne d'Internet (plus précisément les entêtes de tous les paquets envoyés par les utilisateurs du RNU vers Internet et inversement).

4. Caractéristiques générales du trafic : niveau paquet

Dans cette section, nous présentons une caractérisation générale des différentes traces collectées. Cette caractérisation concerne la décomposition du trafic par sens, son évolution temporelle, l'étude de la taille des paquets et la répartition du trafic par protocole et par application. Les résultats, donnés ci dessous, concernent pour la plupart la trace T4. Les résultats calculés à partir des autres traces ne seront mentionnées que lorsqu'ils différent sensiblement de ceux de la trace T4[4].

Sachant que le trafic Internet ne peut être considéré stationnaire que sur un intervalle de temps limité à 1 ou 2 heures maximum, nous présentons à la fois les moyennes calculées à partir de traces complètes et les résultats relatifs aux fichiers de trace de durée une heure afin de mettre en évidence l'influence de l'origine du temps sur les caractéristiques du trafic. Pour cela, nous avons utilisé deux fichiers faisant partie la trace T4 collectée au niveau du nœud el Kasbah en avril 2006. Le premier contient le trafic collecté durant une heure de forte utilisation (de 10h à 11h du matin), alors que le deuxième contient le trafic collecté sur le même lien et pendant la même journée mais durant une heure de faible trafic (de 1h à 2h du matin). Le premier fichier de trace est nommé T4_10h, alors que le deuxième T4_1h.

4.1 Décomposition du trafic par sens

Pour différencier, les deux sens de trafic sur un lien bidirectionnel, nous avons utilisé les adresses IP allouées au réseau RNU. Nous différencions, ainsi, trois sens de trafic : entrant, sortant et interne. Le trafic entrant est composé par les paquets dont uniquement les adresses IP destinations font partie de la plage d'adresses IP RNU ; alors que le trafic sortant est composé par tous les paquets dont seulement les adresses IP sources appartiennent à cette plage. Enfin, le trafic interne renferme tous les paquets dont les adresses IP sources et destinations appartiennent aux adresses IP RNU. Nous notons que, parmi les quatre traces collectées, seule la trace T2 contient les trois sens de trafic, alors que les autres ne renferment qu'un trafic entrant et un autre sortant.

[4] L'analyse des traces T1 et T2 est décrite dans [Bouc06, Bouc07].

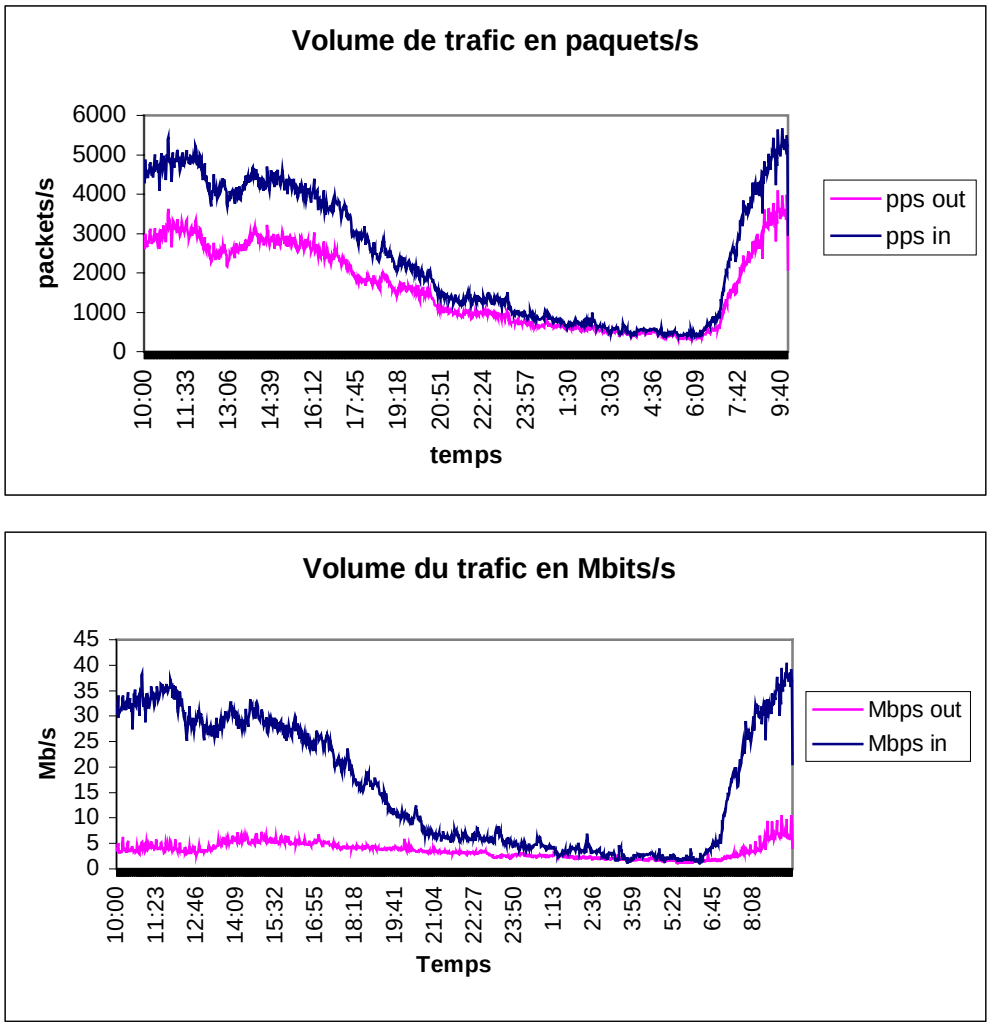

Figure 3.5 : Variation du débit utilisé en Paquets/s et Mbits/s (T4)

La Figure 3.5 représente l'évolution au cours du temps des trafics entrant et sortant, calculés à partir de la trace T4, sur des fenêtres d'une minute (mesurés en paquets/s et en Mbits/s). Nous remarquons la présence dans cette trace, ainsi que dans toutes les autres, d'un cycle journalier caractérisé par une augmentation du trafic la journée et sa diminution durant la nuit. Par ailleurs, ces figures montrent que le trafic entrant est largement supérieur à celui sortant, notamment durant la journée. En effet, le rapport d'asymétrie entre le trafic entrant et celui sortant, mesuré en Mbits/s, oscille entre 8 :1 la matinée et 1 :1 la nuit ce qui reflète, à notre avis, une différence significative entre les usages d'Internet la nuit par rapport à ceux de la journée. D'où nous pouvons conclure que le réseau RNU est un réseau de type client, puisqu'il reçoit beaucoup plus de trafic qu'il en génère.

4.2 Étude des tailles des paquets

En utilisant la trace T4, la Figure 3.6 illustre les distributions cumulatives des tailles des paquets IP des trafics entrant et sortant. Nous constatons que le trafic sortant est composé majoritairement de paquets de taille égale ou légèrement supérieure à 40 octets (65 % des paquets), alors que le trafic

entrant est majoritairement composé de paquets de grande taille (55 % des paquets ont une taille supérieure ou égale à 1420 octets). Ceci est une caractéristique des réseaux de type client qui envoient essentiellement des requêtes (donc des paquets de petite taille) et reçoivent les réponses (donc des paquets de grande taille).

Nous constatons que la taille de paquets 576 octets, correspondant à la taille maximale de paquets pouvant être envoyés sans fragmentation sur les réseaux X25 et utilisée par défaut par TCP lorsque aucun mécanisme de découverte de MTU n'est implémenté, n'apparaît pas comme un mode dominant sur ces courbes. Ceci s'explique par la généralisation au niveau des postes clients et des serveurs de l'utilisation de l'option TCP permettant de spécifier la valeur de MSS[5] et du mécanisme de découverte de MTU (Maximum Transfert unit) au niveau des routeurs de l'Internet.

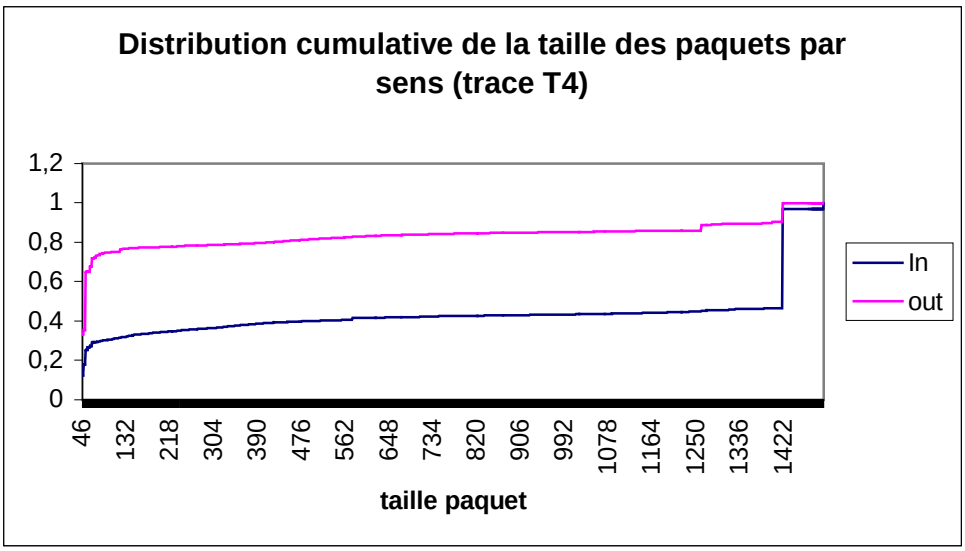

Figure 3.6 : Distributions cumulatives des tailles des paquets par sens (T4)

La deuxième remarque concerne la taille de paquet 1500 octets, correspondant à la taille maximale des paquets sur les réseaux Ethernet ; et qui n'apparaît pas comme un mode dominant sur les distributions de la Figure 3.6, puisque moins de 4% des paquets entrants et 0,03% de paquets sortants ont cette taille. Par contre, nous remarquons la présence d'un nouveau mode dominant correspondant à la taille de 1420 octets, qui concerne plus de 50% des paquets entrants et 10% de ceux sortants. En effet, plusieurs implémentations TCP annoncent une valeur de MSS égale à 1380 octets pour tenir compte des valeurs maximales des entêtes IP et TCP (1380+60+60=1500). Or, cette valeur de MSS engendre, dans le cas où aucune des options IP ou TCP n'est utilisée, des paquets de taille égale à 1420 octets et c'est, d'après nos traces, le cas de la majorité des paquets. C'est pourquoi, il nous semble plus judicieux, d'annoncer une valeur de MSS égale à 1460 octets, afin de

[5] MSS : Maximum Segment Size, c'est la taille maximale de données pouvant être transportées par TCP. Ainsi, dans le cas où, les options IP et TCP ne sont pas utilisées la valeur de MTU est égale à MSS+40 octets.

pouvoir envoyer des paquets de 1500 octets. Avec une telle valeur de MSS les paquets utilisant les options IP ou TCP auront une taille supérieure à 1500 octets et devront ainsi être fragmentés.

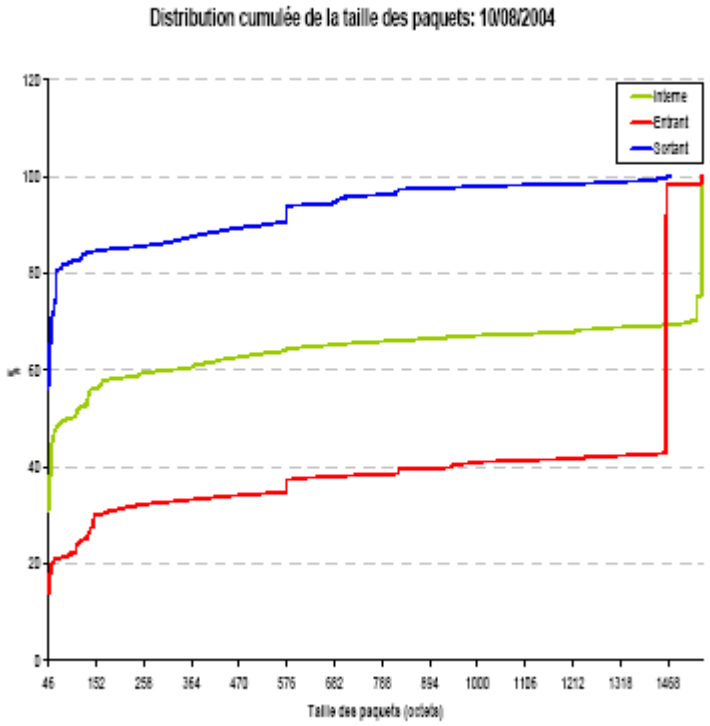

Figure 3.7 : Distributions cumulatives des tailles des paquets (T2)

La Figure 3.7, utilisant la trace inter-nœuds T2, présente les distributions cumulatives des tailles des paquets entrants, sortants et internes. Elle montre, contrairement à la Figure 3.6, la présence du mode, certes peu important, correspondant à une taille de paquet de 576 octets. Ceci montre que les systèmes ne gérant pas la découverte du MTU du chemin, continuaient à exister en 2004 bien que le RFC définissant le mécanisme de découverte de la MTU du chemin (RFC 1191) date de 1990. Enfin, nous remarquons l'importance du mode 1500 octets dans le trafic interne contrairement au deux autres sens du trafic.

La Figure 3.8 illustre la variation au cours du temps de la taille moyenne des paquets par sens calculée à partir de la trace T4. Elle montre, que la taille moyenne des paquets entrants diminue la nuit (elle passe de 900 octets durant la journée à 600 octets la nuit) ; alors que celle des paquets sortants augmente (passant de 200 octets le matin à 600 octets la nuit). Ceci peut être expliqué par des usages d'Internet qui différent selon les moments de la journée ce qui sera confirmé par l'étude des usages de l'Internet dans RNU (voir le paragraphe 4.4).

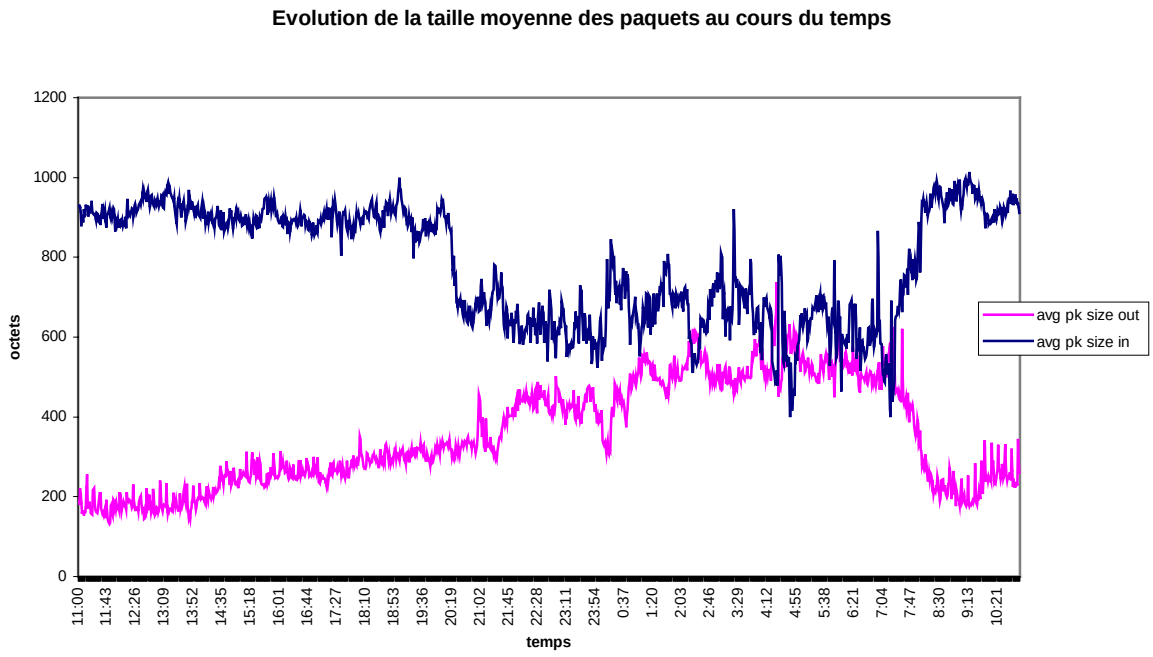

Figure 3.8 : Évolution de la taille moyenne des paquets au cours du temps (T4)

4.3 Répartition du trafic par protocole

Le Tableau 3.2, utilisant la trace T4, illustre la répartition du trafic par protocole et par sens (entrant et sortant). Au dessus de la couche IP, les protocoles les plus utilisés sont, par ordre d'importance décroissant, TCP, UDP et ICMP. Il apparaît ainsi que la majorité du trafic, aussi bien entrant que sortant, utilise TCP, ce qui est conforme aux études métrologiques publiées dans la littérature. De plus, le pourcentage du trafic TCP est d'autant plus important lorsqu'il est mesuré en octets. Ceci reste vrai pour toutes les autres traces collectées, où nous avons remarqué que le trafic TCP représente en moyenne 90% des paquets et 95% des octets.

	Répartition du trafic par protocole (%)					
	TCP		UDP		ICMP	
	paquets	octets	paquets	octets	paquets	octets
T4 entrant	96,87	98,84	1,44	0,59	1,68	0,57
T4 sortant	97,34	99,57	1,49	0,33	1,15	0,1

Tableau 3.2 : Répartition du trafic par protocole

4.4 Répartition du trafic par application

La répartition du trafic UDP par numéro de port, montre que l'application de résolution de noms (DNS) représente la majorité de ce trafic. Par exemple, dans la trace T4, 86% des paquets UDP et 82% des octets transportés par ce protocole sont générés par l'application DNS.

Concernant la répartition du trafic TCP par numéro de port, elle s'étend sur un grand nombre de numéros de ports différents. En effet, nous avons tracé toutes les connexions TCP présentes dans les différentes traces de trafic collectées, grâce aux drapeaux SYN/FIN/RST. Ensuite, nous avons identifié le numéro de port « serveur » relatif à chaque connexion. Ainsi, nous avons recensé, pour la trace T4, 8876 numéros de ports serveurs différents. Nous avons défini sept classes d'applications (voir le Tableau 3.3), en utilisant les numéros de ports enregistrés par l'IANA et les numéros de ports utilisés par défaut par certaines applications.

Classe d'application	**Numéros de ports TCP associés**
Web	http (80), https (443), http-a (8080)
P2P	Edonkey (4662), Fasttrack (1214)
FTP	ftp (21), ftp-data (20)
Mail	SMTP (25), POP3 (110)
Accès distant	SSH (22), telnet (23)
Windows	135, 137, 139, 445, 1433
Autres	Tous les autres ports

Tableau 3.3 : Définition des classes d'applications

Les classes d'applications « web », « mail », «FTP » et « accès distant » sont relativement faciles à identifier en utilisant leurs numéros de ports. Ce sont des applications Internet classiques qui utilisent généralement des ports TCP enregistrés par l'IANA.

Par contre, les applications émergentes, notamment celles permettant l'accès aux réseaux pair à pair, sont les plus difficiles à identifier, car d'une part elles n'utilisent pas des ports enregistrés, d'autre part leur fonctionnement est très peu documenté et elles peuvent utiliser, pour certaines d'entre elles, n'importe quel numéro de port non filtré. Nous avons regroupé dans la classe d'application P2P, le trafic utilisant les ports TCP configurés par défaut au niveau de deux applications pair à pair Emule et Kazaa. L'application Emule permet d'accéder au réseau Edonkey, alors que l'application Kazaa permet l'accès au réseau Fasttrack. Ces deux applications, Emule et kazaa, utilisent des ports TCP et UDP distincts pour la recherche de pairs, et le téléchargement de fichiers ; ainsi elles utilisent par défaut respectivement les ports TCP 4662 et 1214 pour le téléchargement des fichiers.

La classe d'application Windows, regroupe tout le trafic destiné aux ports TCP utilisées par les services Netbios et RPC de Windows (135, 137, 139, 445) ainsi que le trafic destiné au port MS-sql-server (1433). Tous ces ports sont connus pour être des moyens privilégiés utilisés par les vers informatiques pour se propager via Internet. En effet, ils ont été l'objet de plusieurs vulnérabilités, durant ces dernières années exploitées par plusieurs vers informatiques tels que Sasser[6], Blaster[7], .

Enfin la classe d'application « autres », renferme tout le trafic TCP n'appartenant à aucune des six autres classes.

Le Tableau 3.4 dresse la répartition du trafic TCP, mesuré en paquets et en octets, par classe d'application et par sens de trafic (entrant et sortant). Nous remarquons ainsi que la navigation sur le Web est l'application qui génère la majorité des paquets transportés par le réseau RNU (plus de 80% des paquets entrants et de 77% des paquets sortants), alors que les applications pair à pair, arrivent en deuxième position en pourcentage de paquets générés. De plus, contrairement à toutes les autres classes d'applications, le trafic pair à pair contribue beaucoup plus au volume du trafic sortant qu'à celui entrant (15,34 % des paquets et 50,88 % des octets). Ainsi, nous pouvons conclure que le trafic pair à pair a pour effet de diminuer l'asymétrie du trafic en augmentant surtout le volume du trafic sortant.

Classe d'application	**Trafic entrant**		**Trafic sortant**	
	% paquets	% octets	% paquets	% octets
Web	81,26	89,55	77,61	38,81
P2P	8,74	3,25	15,34	50,88
FTP	0,88	1,04	0,75	0,32
Mail	0,49	0,44	0,66	0,96
Accès distant	0,11	0,02	0,17	0,09
Windows	3,83	0,27	0,05	0,00
Autres	4,68	5,41	5,42	8,94

Tableau 3.4 : Répartition du trafic par classe d'application (T4)

Par ailleurs, nous remarquons que, prés de 4% des paquets entrants et 0,5% de ceux sortants sont destinés aux ports « windows ». Ces derniers sont probablement envoyés d'une manière automatique par des machines infectées cherchant à propager leurs vers via le réseau. Il remarquable que ce trafic

[6] Ce vers utilise le port 445 pour se propager (voir le site http://www.secuser.com/alertes/2004/sasser.htm pour plus de détails)

[7] Ce vers utilise le port 135 pour se propager (voir le site http://www.secuser.com/alertes/2003/blaster.htm pour plus de détails)

est beaucoup plus important lorsqu'il est mesuré en paquets que lorsqu'il est mesuré en octets (3,83% des paquets entrants mais seulement 0,27% des octets entrants). Ceci s'explique par le fait que la majorité de ce trafic ne transporte pas de charge utile (payload).

Enfin, la classe d'application « autres », renfermant le trafic TCP n'appartenant à aucune des six autres classes, représente un pourcentage non négligeable de paquets et surtout d'octets. De ce fait, il est très probable que ce trafic soit véhiculé par des applications de type pair à pair.

Pour mettre en évidence les variations des usages de l'Internet selon le moment de la journée, nous avons calculé pour chaque fichier de la trace T4, la répartition du trafic total par classe d'application. Ainsi, nous reportons, dans les figures 3.9 et 3.10, les répartitions par classe d'application calculées à partir des traces T4_10h et T4_1h. Il en ressort ainsi, que durant les heures de travail, la répartition du trafic TCP par application est dominée par le trafic de navigation sur le Web. Alors que, durant la nuit, l'utilisation de l'Internet est plus panachée, avec la présence non négligeable de trafic généré par les applications pair à pair et celles utilisant des ports TCP non enregistrés.

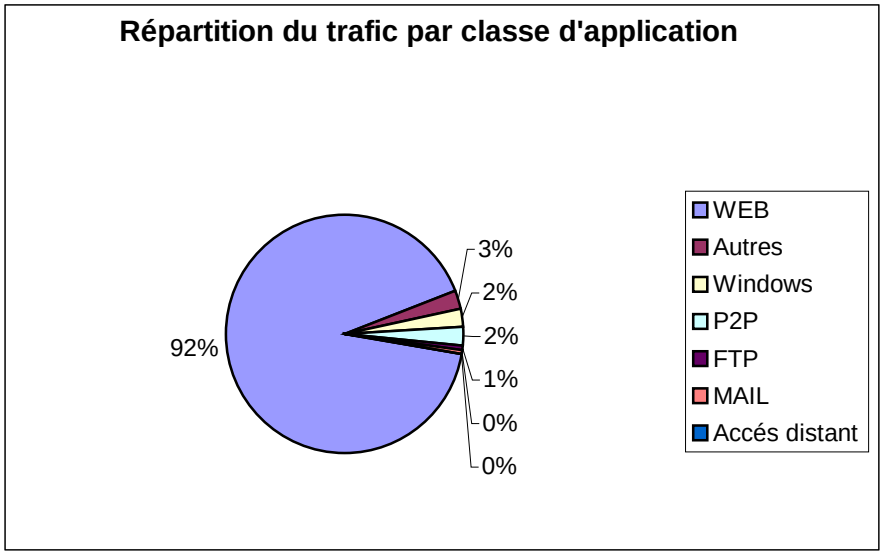

Figure 3.9 : Répartition du trafic (en paquets) par classe d'application (T4_10h)

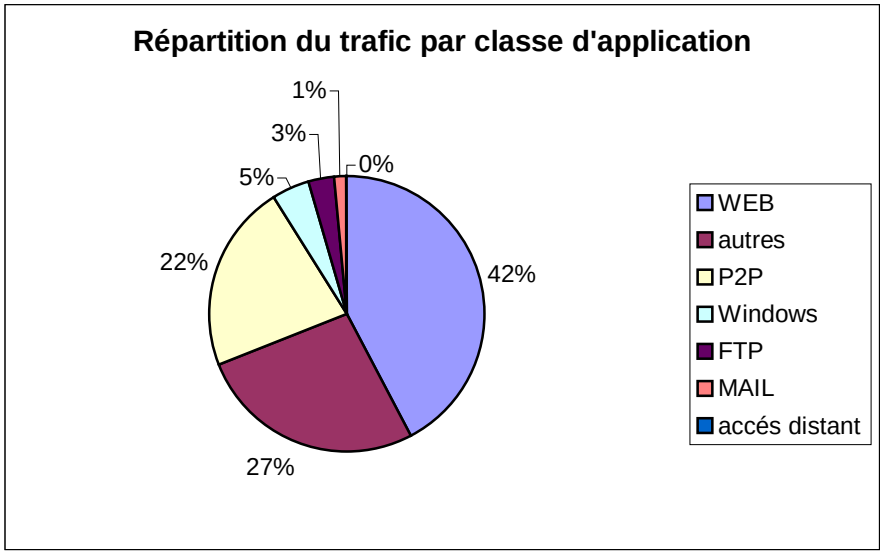

Figure 3.10 : Répartition du trafic (en paquets) par classe d'application (T4_1h)

4.5 Conclusion

Malgré quelques spécificités (présence du mode 1420 octets dans la distribution des tailles des paquets) nous pouvons considérer le réseau RNU comme un spécimen réduit du réseau mondial, puisqu'il reflète globalement les mutations qu'ont connu Internet ces dernières années. Il s'agit notamment de l'importance grandissante du trafic des applications émergeantes et celui dû à la propagation des vers sur le réseau mondial. Cette observation sera confirmée par la caractérisation du trafic au niveau des flux, présentée ci dessous.

5. Caractéristiques générales du trafic : niveau flux

Puisque que le trafic TCP est majoritaire dans toutes les traces collectées, nous nous proposons d'étudier, dans cette section, les caractéristiques des flux TCP. Pour cela nous définissons un flux TCP comme étant une connexion TCP définie par un 5-uplet et délimitée grâce aux drapeaux TCP (SYN, FIN et RESET).

5.1 Répartition des connexions TCP par application

Pour étudier la répartition des connexions TCP par application, nous avons adopté les sept classes d'applications décrites dans le Tableau 3.3. Ainsi, à partir de la trace T4, nous avons calculé la répartition des connexions TCP par classe d'application et reporté les résultats dans le Tableau 3.5. Il en ressort que cette dernière répartition diffère nettement de celle des paquets TCP, ou encore celle des octets transportés par TCP (voir le Tableau 3.4). En effet, les ports « Windows » génèrent prés de 40% des connexions, bien qu'ils ne représentent que 3,83 % des paquets entrants et 0,05 % des

paquets sortants. Alors que les applications pair à pair, représentant moins de 0,4% des connexions, génèrent prés de 9% des paquets entrants et 15% de ceux sortants.

Classe d'application	**# Connexions**	**% Connexions**
Web	4 512 009	57,30
Windows	3 132 340	39,78
FTP	38 554	0,49
P2P	29 413	0,37
Accès distant	27 750	0,35
Mail	12 436	0,16
Autres	121 308	1,54

Tableau 3.5: Répartition des connexions TCP par classe d'application

Ainsi, nous pouvons déduire que le trafic généré par les ports « windows », bien qu'il représente un faible pourcentage des paquets et un infime pourcentage des octets, risque de dégrader les performances des équipements actifs surtout les pare-feu qui doivent maintenir un état pour chaque connexion en cours. D'un autre coté, bien que le trafic pair à pair représente un nombre limité de connexions, donc vraisemblablement généré par un nombre réduit d'utilisateurs ; il a la particularité de générer un volume considérable de paquets et d'octets. Ainsi, le trafic pair à pair peut avoir un effet démesuré sur la saturation de la bande passante disponible. Par conséquent, nous suggérons que la détection et la surveillance des flux appartenant à ces deux classes d'application est indispensable, afin de prévenir les dégradations de performance qu'ils peuvent occasionner.

Par ailleurs, nous remarquons que les flux utilisant les ports « windows » correspondent à des flux souris, par contre ceux utilisant les ports des applications pair à pair sont des flux éléphants. L'étude des durées et des tailles des connexions TCP, détaillée dans suite confirmera l'existence des flux éléphants et des flux souris dans le trafic RNU.

5.2 Distribution de la durée des connexions

La durée d'une connexion TCP est le laps de temps entre l'instant correspondant à l'enregistrement, par la sonde de mesures, du premier paquet et celui correspondant à l'enregistrement du dernier paquet faisant partie de cette connexion. Comme les fichiers de trace collectés ont une durée limitée à une heure, la durée maximale que nous pouvons calculer est de 1 heure, alors que la durée minimale est égale à la résolution de l'estampille temporelle utilisée qui est de 1µs. Or la précision de cette estampille est de l'ordre du milliseconde, c'est pourquoi nous avons opté pour le filtrage de toutes les connexions ayant une durée inférieure à 10ms. Ces dernières représentent entre 0,5% et 2,5% du nombre de connexions total.

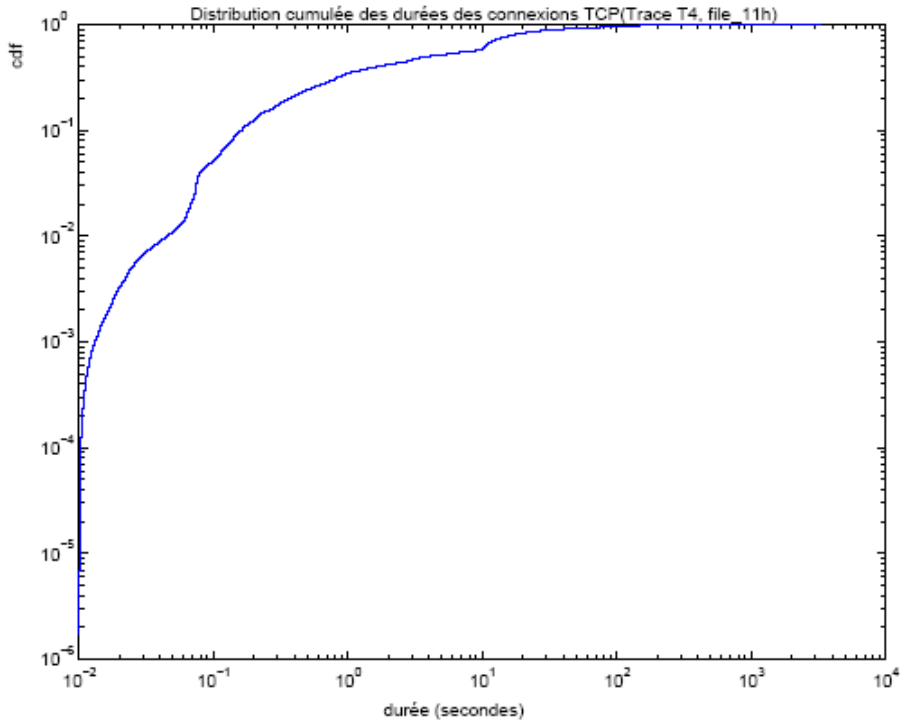

Figure 3.11 : Distribution cumulative de la durée des connexions (T4_10h)

La Figure 3.11 illustre, pour la trace T4_10h, la distribution cumulative (CDF) de la durée des connexions TCP. L'échelle logarithmique est utilisée car elle permet de souligner d'une part l'étendue des valeurs prises par cette variable aléatoire et d'autre part le fait que la majorité des connexions soient de courte durée. En effet, 50% des connexions ont une durée inférieure à 4 secondes et 90 % ont une durée inférieure à 50 secondes, alors que la durée maximale de connexion dépasse les 55 minutes.

Pour vérifier la présence de queue lourde dans la distribution de la durée des connexions TCP, nous avons approché la queue de la distribution cumulative complémentaire (CCDF) empirique par une loi exponentielle et par une loi de puissance (Pareto) en utilisant la méthode des moindres carrées avec un intervalle de confiance à 95%. Nous nous sommes limités à la queue de la distribution (10% des connexions ayant les plus longues durées de connexions), car nous n'avons pas pu approximer la totalité de la distribution par aucune loi de probabilité connue. Ceci a été déjà souligné par les études de trafic présentées dans l'état de l'art (chapitre 1, section 3.2.1).

La Figure 3.12 expose les résultats de cette approximation. Ainsi, bien que la courbe gauche montre que les deux lois exponentielle et Pareto offrent, toutes les deux, une bonne approximation (visuelle) de la loi empirique, le traçage des courbes sur une échelle log-log, montre clairement que seule la loi Pareto offre une approximation adéquate de la loi empirique. De plus, l'analyse des résidus et le test de Kolmogorov-Smirnov confirme bien ce constat. L'estimation des paramètres de la loi de Pareto, notamment le facteur α , montre qu'il est de 0,97 pour le fichier de trace T4_10h. Ceci est le signe

de la grande variabilité des durées des connexions TCP, puisqu'une loi de Pareto ayant un facteur α compris entre 0 et 1 possède une moyenne arithmétique et une variance toutes les deux infinies.

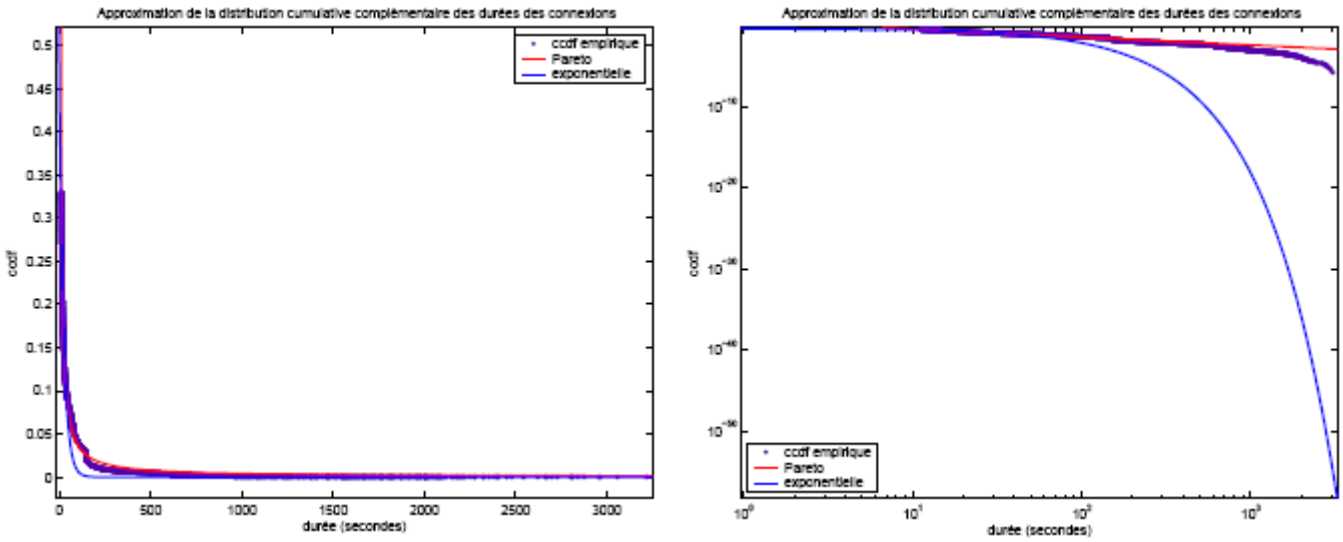

Figure 3.12 : Approximation de la distribution de durée des connexions (T4_11h)

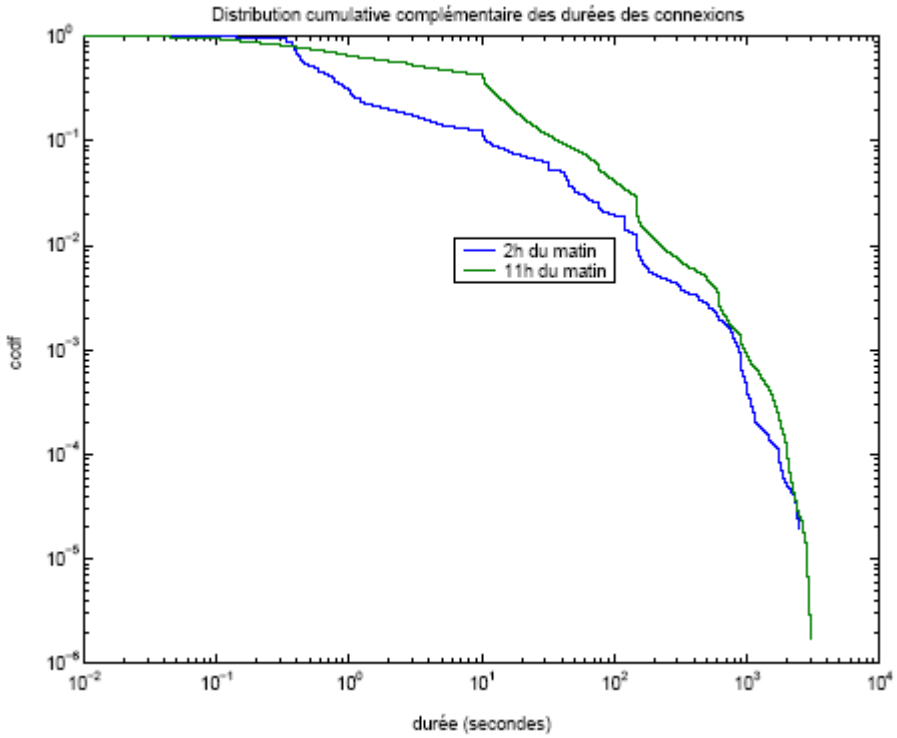

Figure 3.13 : Distributions des durées des connexions (traces T4_10h et T4_1h)

Par ailleurs, la Figure 3.13 représentant les distributions cumulatives complémentaires calculées à partir des traces T4_10h et T4_1h, montre que la CCDF empirique calculée à partir de la trace collectée durant une période de faible utilisation (T4_1h) est à décroissance plus rapide que celle calculée à partir d'une trace collectée durant une période de forte utilisation (T4_10h). Ceci est confirmé par la modélisation de la première courbe par une loi de Pareto de facteur α égal à 1,12,

alors que la deuxième est approximée par une loi de Pareto de facteur α de 0,97. Ainsi, nous pouvons conclure que la distribution des durées des connexions (du moins la queue de cette distribution) n'est pas stable mais varie selon le moment de la journée.

5.3 Distribution de la taille des connexions TCP

Pour l'étude des tailles des connexions, nous nous intéressons aux connexions TCP durant lesquelles il y a eu un transfert de données. Nous calculons ainsi, pour chaque connexion TCP, le nombre d'octets de données envoyés et celui reçus. Pour cela, nous nous basons sur la différence entre le numéro de séquence final et celui initial pour estimer le nombre d'octets de données envoyés et reçus pour une connexion TCP donnée.

L'étude des nombres d'octets envoyés et celui reçus par connexion TCP montre, pour toutes les traces étudiées, la présence de connexions éléphants et de connexions souris. En effet, dans la trace T4_10h, 15% des connexions TCP génèrent plus de 85% du trafic entrant au réseau RNU, alors que les 85% des connexions TCP restantes génèrent moins de 15% du trafic total entrant. De plus seulement 1% des connexions TCP génèrent 52% du trafic.

Pour vérifier la présence de queue lourde dans les distributions des tailles des connexions TCP, nous avons procédé à leur ajustement par des lois de Pareto en utilisant la méthode des moindres carrés. Ainsi, nous obtenons un ajustement de la distribution du nombre d'octets reçus par connexion TCP par une loi de Pareto ayant un facteur α de 0,93 et un ajustement de la distribution du nombre d'octets envoyés par connexion par une loi de Pareto avec α de 1,21. L'analyse des résidus et le test Kolmogorov-Smirnov ont permis de confirmer la validité de ces approximations.

Nous pouvons conclure que les distributions des durées et des tailles des connexions TCP sont toutes les deux des distributions à décroissance lente. De plus, leurs queues peuvent être approximées par une loi de Pareto avec α proche de 1. Ce phénomène, bien qu'observé dans toutes les traces collectées, est plus marqué dans les nouvelles traces T3 et T4.

6. Étude de la dépendance à long terme dans le trafic

Les caractéristiques statistiques que nous étudions dans cette section concernent les différentes lois d'arrivée des connexions et des paquets composant le trafic RNU. L'objectif est de vérifier l'existence et de mesurer éventuellement le degré de la dépendance à long terme (LRD) dans ces processus. L'évaluation de la LRD dans le trafic RNU, se justifie d'une part par la présence de queue lourde dans les distributions des durées et des tailles des connexions TCP et d'autre part par la présence de flux éléphants et de flux souris. Or, ces deux phénomènes sont étroitement liés au degré de LRD dans le trafic.

Le paramètre de Hurst (H) permet une quantification du phénomène de dépendance présente dans n'importe quelle série temporelle. En effet, une valeur de H inférieure à 0,5 indique l'absence de dépendance à long terme (LRD) dans la série de données analysée ; alors qu'une valeur comprise entre 0,5 et 1 indique la présence de LRD. De plus, plus la valeur de H est proche de 1, plus la portée de la dépendance dans la série temporelle est importante (couvre une large échelle de temps).

Pour le calcul du paramètre de Hurst des séries des durées inter-paquets et des séries des durées inter-connexions, nous avons utilisé le logiciel Selfis [selfis] qui implémente plusieurs méthodes d'estimation du paramètre de Hurst dont les routines Ldestimate de Abry-Veitch que nous avons utilisé pour calculer les valeurs de H à partir des traces du RNU.

La Figure 3.14 illustre la variation du paramètre de Hurst, calculé à partir des séries d'estampilles inter-paquets, durant une journée (trace T4). Nous remarquons, ainsi que toutes les valeurs de H sont comprises entre 0,5 et 1, donc nous pouvons déduire qu'il existe une dépendance à long terme entre les arrivées des paquets dans le réseau RNU. De plus le degré de dépendance augmente durant les heures de forte utilisation et diminue durant les heures creuses (notamment la nuit).

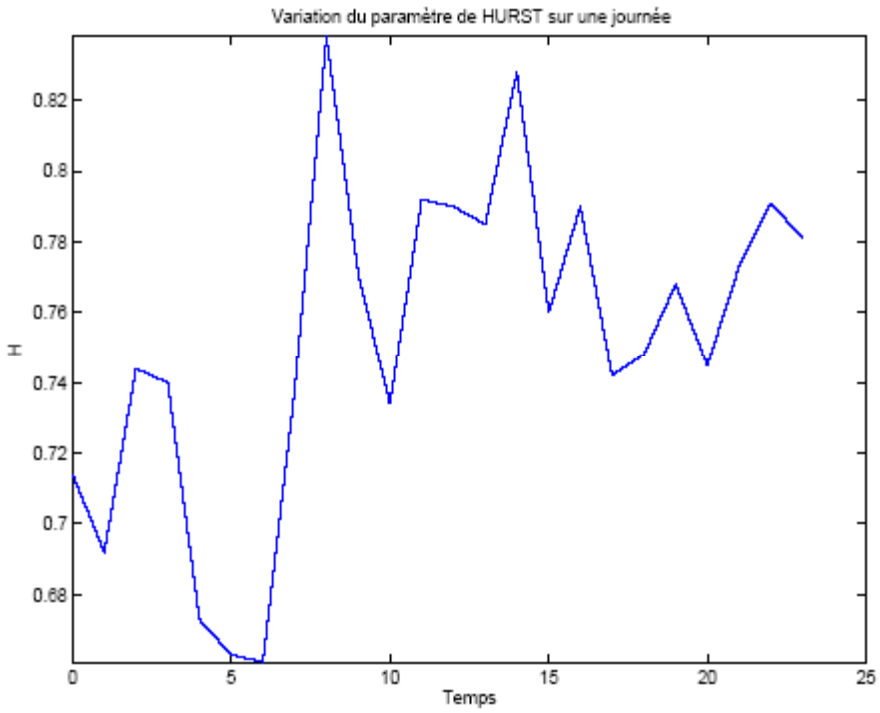

Figure 3.14 : Variation du paramètre de Hurst relatif aux arrivées des paquets (T4)

Concernant les inter-arrivées de connexions TCP, l'estimation du paramètre de Hurst par la méthode des ondelettes, donne des valeurs légèrement supérieures à 0,5 pour tous les fichiers de la trace T4. Nous pouvons ainsi conclure qu'il existe peu de dépendance à long terme entre les arrivées des connexions TCP, voire la LRD est inexistante à ce niveau (à titre d'exemple, le calcul du paramètre de Hurst à partir du fichier de trace T4_10h donne une valeur de 0,575). Par conséquent, nous

pouvons conclure que le processus d'arrivée des connexions TCP peut être modélisé par un processus Poissonnien.

La comparaison entre les valeurs du paramètre de Hurst calculées à partir des traces de 2006 (T3 et T4) et celles calculées à partir des traces plus anciennes (T1 et T2) [Bouc06], montre clairement que le niveau de LRD au niveau des arrivées des paquets a augmenté avec le changement d'architecture du réseau RNU et l'augmentation des débits au niveau de ses liens. Par contre, au niveau des arrivées des flux, le niveau de LRD a diminué.

Enfin, nous considérons ces résultats comme une nouvelle preuve empirique de la présence de la LRD dans les arrivées des paquets dans les réseaux IP. Pour les arrivées des flux, nous estimons que le niveau de LRD est assez faible ce qui permet de le négliger. Ce dernier résultat s'explique par généralisation de l'utilisation du protocole HTTP 1.1 au niveau des clients et des serveurs de l'Internet. En effet, ce dernier protocole utilise la technique des connexions persistantes qui permet d'envoyer plusieurs requêtes HTTP sur une même connexion TCP, détruisant ainsi la dépendance qui existait à cause de l'utilisation des versions antérieures du protocole http qui avaient besoin d'ouvrir une connexion TCP pour chaque objet.

7. Analyse des connexions TCP

Le déroulement d'une connexion TCP entre un client et un serveur, s'effectue selon un certain nombre de règles bien définies dans le RFC 793. En effet, l'établissement de la connexion se fait par une poignée de main en trois temps (three way hand check), durant laquelle un certain nombre paramètres (tels que les numéros de séquence initiaux) sont échangés. De plus, la rupture de connexion, utilise une poignée de main en quatre temps (avec échange de deux paires de segments FIN et ACK) permettant à chaque extrémité d'effectuer sa terminaison d'une manière indépendante.

En exploitant la sémantique d'établissement et de fermeture de connexion inhérente au protocole TCP, nous proposons une classification des connexions TCP en plusieurs types. Cette classification sera utilisée, par la suite, pour l'étude de la répartition du trafic RNU par type de connexion, avant de définir les anomalies TCP et d'étudier leur présence dans le trafic RNU.

7.1 Les types de connexions TCP

Nous définissons trois types de connexions TCP:

1. Complètes : Ce sont des connexions composées de trois phases : une phase d'ouverture de connexion comportant une poignée de main en trois temps, une phase de transfert de données et une phase de fermeture de connexion en quatre temps.
2. Reset ou annulées : Il s'agit des connexions comportant au moins un paquet avec un drapeau « RESET » mis à 1.

3. Incomplètes : Ce sont des connexions TCP durant lesquelles les étapes d'ouverture et/ou de fermeture de connexion ne sont pas respectées ou sont totalement inexistants.

Les connexions incomplètes et annulées peuvent avoir plusieurs causes, dont les balayages de ports TCP (scan de ports), le trafic « backscatter » ou de rétro-diffusion, les congestions, le comportement des utilisateurs et des applications qui à l'expiration d'un certain timeout peuvent envoyer un paquet RST pour annuler la connexion, etc.

Rappelons que les balayages de ports (ou scans) ne sont pas des attaques à proprement parler ; ils ont pour objectif de déterminer les ports ouverts sur une machine cible. De plus, ils sont généralement générés, soit par des attaquants pour collecter le maximum d'informations sur leurs cibles, soit encore par les vers informatiques qui essaient par ce biais de trouver des adresses IP vulnérables afin de les infecter. La technique de SYN scan (ou scan demi ouvert) est la plus utilisée, elle consiste à envoyer un seul paquet SYN vers le port destination d'une machine cible et déduire l'état de ce port grâce à la réception ou non d'un paquet de réponse. En effet, si un paquet SYN/ACK est envoyé par la machine cible, alors l'attaquant en déduit que le port en question est ouvert. Si par contre, la machine cible répond par un paquet RST/ACK à la demande de synchronisation envoyée par l'attaquant, alors il en déduit qu'il s'agit d'un port fermé (c'est à dire qu'aucun service n'écoute sur ce port). Enfin, si aucun paquet réponse n'est envoyé, alors l'attaquant en déduit qu'il s'agit d'un port filtré (c'est à dire qu'un pare-feu interdit l'accès au service éventuel écoutant sur ce port). Ainsi d'après ce qui précède, nous pouvons conclure que le balayage de ports de type SYN scan engendre des connexions annulées (dans le cas de ports fermés) et de connections incomplètes dans le cas de ports filtrés.

Par ailleurs, le trafic « backscatter », ou de rétro-diffusion, est le trafic de réponse engendré par les attaques de type déni de service (Deny of Service : DoS) avec usurpation d'adresse IP source. En effet, dans ce type d'attaque, l'attaquant forge plusieurs paquets en utilisant, comme adresses sources, des adresses IP usurpées ou spoofées. Tous ces paquets sont envoyés à la machine victime, qui ne pouvant pas distinguer entre les demandes de connexions envoyées par l'attaquant et celles légitimes, tente de répondre à chaque demande reçue provoquant ainsi un déni de service par une saturation de ressources. Lorsque l'attaquant usurpe une adresse IP, appartenant à une machine connectée, celle ci reçoit un paquet SYN/ACK provenant de la victime auquel elle peut soit répondre par un paquet RST pour annuler la connexion soit encore simplement ignorer ce paquet. Par conséquent, le trafic de rétro-diffusion se présente sous forme de connexions incomplètes ou annulées renfermant des paquets SYN/ACK mais dépourvues de paquet SYN.

Pour mieux analyser les origines des connexions incomplètes et annulées, la classification des connexions en trois types est affinée par la définition de sous-types pour les connexions annulées et incomplètes. Ainsi, nous distinguons deux types de connexions annulées, selon que le paquet « RST » a été envoyé par le coté client ou le coté serveur. Pour chacun de ces types, nous

distinguons deux cas selon que le paquet RST a été envoyé avant ou après la fin de la phase d'établissement de connexion. Nous signalons que les connexions TCP annulées par le serveur après un échange de poignée de main à trois temps, sont très probablement dues aux congestions, alors que celles annulées par le serveur avant la fin de la phase d'établissement de connexion sont probablement causées par des balayages de ports. Par contre, les connexions annulées par le coté client après l'échange de la poignée de main, sont vraisemblablement dues aux comportements des utilisateurs qui décident d'annuler un transfert car jugé trop long ou simplement pour lancer un autre transfert. Enfin, les connexions annulées par le coté client avant la fin de la phase d'établissement de connexion, sont probablement dues à un balayage de ports.

Pour les connexions incomplètes, nous distinguons d'abord celles ne comportant ni phase d'ouverture de connexion, ni phase de fermeture. Ce sont des connexions qui ont probablement été initialisées avant le début de la capture et qui ont duré après l'arrêt de celle ci. Il y a aussi les connexions ayant une phase d'ouverture incomplète, c'est le cas par exemple des connexions composées uniquement par un ou plusieurs paquets SYN, et de celles composées par des paquets SYN/ACK uniquement. Ces deux derniers types de connexions incomplètes sont causées des attaques de type balayage de ports pour le premières, et reflètent la présence de trafic de rétro-diffusion pour les secondes. Nous distinguons enfin les connexions TCP, dont l'établissement de connexion a été effectué dans les règles, alors qu'il n' y a pas de phase de fermeture de connexion et celles dont l'étape de fermeture a été effectuée correctement, alors que l'étape d'ouverture est inexistante. Ces deux derniers types de connexions incomplètes peuvent être causées en partie par le fait que les traces collectées ont une durée limitée à une heure. Mais peuvent également être engendrées par des attaques de types balayage de ports.

7.2 Répartition du trafic par type de connexion

Le Tableau 3.6 présente la répartition des connexions TCP par type dans les quatre traces collectées. Nous remarquons que le pourcentage des connexions annulées et celui des connexions incomplètes sont importants dans toutes ces traces.

Trace	% Complètes	% Annulées	% Incomplètes
T1	29,65	30,65	39,69
T2	31,18	53	15,8
T3	37,9	21,33	40,73
T4	27,26	18,56	54,17

Tableau 3.6: Répartition des connexions TCP par type

Le Tableau 3.7, utilisant un fichier de trace de durée 1 heure (T4_10h), présente la répartition détaillée des connexions TCP par type. Nous remarquons que seulement 24,4% des connexions TCP sont des connexions complètes, alors que plus de 5% sont annulées par le serveur probablement à cause de congestions et plus de 16% sont probablement annulées par le client suite à des temps d'attente trop longs, ou simplement pour initier d'autres connexions. De plus, 41,9% des connexions TCP sont des balayages de ports (formées par des paquets SYN isolés), alors qu'un peu plus de 1% des connexions TCP reflètent par présence d'attaques DoS (composées par paquets isolés de réponse de type Syn/Ack). L'origine du reste des connexions incomplètes, à peine 8% du nombre de connexions TCP total, ne peuvent être identifiées avec certitude : les balayages de ports et les congestions sont des explications possibles de ces connexions mais il y a aussi le fait que les fichiers de traces ont une durée limitée à une heure.

Enfin, nous signalons que le fait d'attribuer toutes les connexions incomplètes et annulées, dont les origines ne peuvent être identifiées avec certitude, à des balayages de ports permet de définir une borne supérieure du pourcentage des connexions TCP engendrées par ces activités. Par conséquent, nous pouvons conclure que les activités de balayage de ports représentent au plus 53% du trafic RNU mesuré en nombre de connexions.

<table>
<tr><th>Type de connexion</th><th colspan="4">Nombre de flux [%]</th></tr>
<tr><td>Complète</td><td colspan="4">24,43</td></tr>
<tr><td rowspan="3">Annulée</td><td colspan="2">Annulée par le coté client</td><td colspan="2">Annulée par le coté serveur</td></tr>
<tr><td>Avant la fin de la poignée de main</td><td>Après la fin de la poignée de main</td><td>Avant la fin de la poignée de main</td><td>Après la fin de la poignée de main</td></tr>
<tr><td>2,90</td><td>16,14</td><td>0,07</td><td>5,45</td></tr>
<tr><td rowspan="2">Incomplète</td><td>Paquets SYN</td><td colspan="2">Paquets SYN/ACK uniquement</td><td>Autres</td></tr>
<tr><td>41,93</td><td colspan="2">1,03</td><td>8,04</td></tr>
</table>

Tableau 3.7: Répartition détaillée des connexions TCP par type (T4_10h)

7.3 Étude des anomalies TCP

Tout flux TCP traversant un réseau peut être sujet à la duplication d'un certain nombre de ses paquets, à la perte d'autres paquets et à l'arrivée dans un ordre différent de celui de l'émission d'autres paquets. Tous ces phénomènes sont appelés, dans [Mell06], anomalies TCP. Il est à signaler que grâce aux numéros de séquence et d'acquittement, présents dans tout paquet TCP, les systèmes

terminaux peuvent remettre les données reçues dans l'ordre à l'application destinataire, détecter la duplication de paquets ou leur perte et provoquer la retransmission des paquets perdus[8].

Les études métrologiques, notamment [Paxs97 et Mell06], montrent que les anomalies TCP sont relativement fréquentes dans les réseaux IP. En effet, dans [Mell06], les auteurs ont défini une méthode de classification des anomalies TCP basée sur des heuristiques et l'ont implémenté dans l'outil Tstat [Tstat] avant de l'utiliser pour l'analyse de traces de mesures passives et d'affirmer que 5% des segments TCP entrants présentent une anomalie ; alors que dans l'autre sens, ce pourcentage atteint 8% des segments TCP.

En utilisant l'outil Tstat [Tstat], nous avons étudié des anomalies TCP présentes dans le trafic collecté sur le réseau RNU. Le Tableau 3.8 donne, pour la trace T4, le pourcentage des segments TCP enregistrés en ordre par la sonde de mesure, ainsi que celui des segments TCP arrivés en désordre. Ces valeurs sont supérieures à ceux de [Mell06]. De plus, elles montrent l'importance des congestions que subissent les connexions TCP transitant par le réseau RNU, puisque prés de 8% des segments TCP ont du être retransmis à cause des congestions.

Trace	**% Segments TCP en ordre**	**% Segments TCP en désordre**			
		Duplication	Routage	Congestion	Autres
T4_10h	87,37	0,10	0,28	7,91	4,34

Tableau 3.8 : Analyse des anomalies TCP

8. Conclusion

L'analyse du trafic RNU a mis en évidence la grande variabilité de ses caractéristiques. En effet, le volume du trafic (mesuré en nombre d'octets, de paquets ou de flux) et sa répartition (par numéro de port ou type d'application) varient selon le moment de la journée confirmant ainsi les études métrologiques précédentes (cf chapitre 1).

Plus précisément, les variations du volume du trafic dans RNU concernent une multitude d'échelles de temps. En effet, il existe une dépendance à long terme dans le processus d'arrivée des paquets, composant ce trafic; de plus le niveau de cette dépendance varie beaucoup au cours de la journée. Ce qui a pour effet de compliquer la mise en place d'un modèle de trafic pour les arrivées des paquets.

[8] En effet, le numéro de séquence permet d'identifier le nombre octets envoyés par l'émetteur, tandis que le numéro d'acquittement permet d'identifier le nombre octets de données acquittées par celui-ci. Ainsi le numéro d'acquittement au niveau de l'émetteur doit être synchronisé avec le numéro de séquence du récepteur et inversement.

Concernant les arrivées des flux, nous avons remarqué une absence de LRD, ce qui nous permet de proposer une modélisation du processus d'arrivée des flux par une loi de type Poisson.

La présence de LRD a des implications importantes sur le dimensionnement des nœuds de l'Internet : les liaisons Internet et les files d'attentes dans les routeurs doivent être surdimensionnés sans quoi le taux de pertes des paquets serait important et causerait une détérioration de la qualité de service offerte par le réseau. Plus concrètement, **le débit d'une liaison doit être de 2 à 3 fois la valeur moyenne du débit calculée sur une fenêtre de 5mn.** Cette règle est malheureusement non appliquée au niveau du RNU.

Par ailleurs, l'analyse de la répartition du trafic RNU par flux, a montré l'existence d'une majorité de flux souris générant peu de trafic, à coté d'une minorité de flux éléphants responsables de la quasi totalité du trafic. Les flux éléphants, par le volume de trafic qu'ils génèrent, peuvent saturer les liens du réseau. Alors que les flux souris, malgré le faible volume de leur trafic, peuvent dégrader considérablement les performances des équipements du réseau, notamment ceux qui maintiennent un état pour chaque flux actif (les pare-feu sont les plus touchés). Ainsi, **nous estimons qu'une gestion de réseau efficace ne peut se limiter à la surveillance des flux éléphants (comme préconisé dans [Floy03]) mais nécessite aussi la surveillance des flux souris. Cela peut se faire en utilisant par exemple l'outil Netflow**.

En outre, l'analyse de la répartition des connexions TCP par classe d'application, a montré que la plupart des flux souris sont en réalité des connexions TCP anormales puisqu'elles ne transportent aucune charge utile et utilisent les ports « Windows ». Or ces ports n'ont pas d'utilisation légitime en dehors des réseaux locaux où ils servent au partage de fichiers et d'imprimantes, à l'appel de procédures distantes et la découverte du voisinage réseau. Sur les liens de l'Internet, la présence d'un tel trafic résulte de la propagation de vers informatiques sur le réseau RNU. D'où la nécessité de filtrer ce trafic au niveau de tous les routeurs du réseau RNU. **Ce filtrage doit s'effectuer le plus prés possible des sources de ce trafic c'est à dire au niveau de tous les routeurs des institutions universitaires**. Un tel filtrage permettrait de limiter le trafic indésirable dans RNU sans toutefois pouvoir l'éliminer en totalité car les ports utilisés par un tel trafic sont très variables au cours du temps.

D'autre part, l'analyse des connexions TCP par type, a montré l'importance des balayages de ports puisqu'ils constituent au moins 42% des connexions. Ces derniers sont dus principalement aux propagations des vers informatiques sur le réseau RNU.

Bien que, les ingénieurs et les techniciens du CCK fussent déjà confrontés à la propagation massive des vers informatiques et à des attaques de dénis de service dans le réseau RNU, ils ne disposaient pas d'outils efficaces pour les détecter et les différencier des congestions et des pannes physiques de liens. En effet, suite aux réclamations des utilisateurs se plaignant de la dégradation des

performances du réseau, ils procèdent généralement par élimination et font recours à des outils adhoc pour diagnostiquer les causes de la dégradation de QoS observée.

Dans le chapitre suivant, nous proposons une méthode pour la détection d'anomalies à partir de métriques de volume collectées périodiquement au niveau de tous les liens du réseau surveillé. La méthode proposée est par la suite exploitée pour la détection et la caractérisation des anomalies de trafic affectant le réseau RNU.

Chapitre 4: Détection d'anomalies de volume

Le réseau national universitaire ou plus précisément certains de ses liens sont fréquemment confrontés à la survenue de pannes et d'anomalies qui perturbent son fonctionnement et provoquent une dégradation de la qualité de service perçue par ses utilisateurs. Pour les détecter, les techniciens du CCK se sont dotés d'un ensemble d'outils de mesures (actives et passives) et de gestion de réseaux commerciaux et du domaine public. Malgré la multiplication des outils, le processus de détection et d'identification des pannes et des anomalies demeure manuel ; il repose sur l'expertise des techniciens du CCK.

Le travail de mastère [Ayar05 et Ayar05a] a proposé une caractérisation spatiale et temporelle des phénomènes de pannes affectant le RNU. Il a montré, en utilisant les techniques de mesures actives[1], que les pannes étaient omniprésentes dans RNU. Toutefois, elles n'affectaient pas tous les liens d'une manière identique puisqu'une majorité des pannes a été engendrée par un nombre limité de liens. Ces derniers étaient sujets à des pannes répétitives mais de courtes durées. L'investigation a montré que la plupart des pannes détectées étaient soit le résultat de liaisons physiques de mauvaise qualité (présentant un taux d'erreurs important), soit encore le résultat de congestions sévères au niveau de certains liens.

La détection d'anomalies est plus délicate ; elle nécessite un choix judicieux des métriques à utiliser, la construction d'un modèle de référence pour le trafic normal et le calibrage fin des seuils de détection. Dans ce chapitre, nous présentons une nouvelle méthode, inspirée des travaux de [Shyu03] et de [Lakh04], permettant la détection des anomalies affectant le volume de trafic véhiculé par un réseau donné.

1. Description de la méthode proposée

L'approche de détection d'anomalies que nous proposons a pour objectif de détecter les anomalies de forte intensité, c'est à dire celles qui s'accompagnent de génération d'un volume de trafic important. Pour ce faire, elle consiste à :

1. Collecter périodiquement des métriques de volume au niveau des N liens du réseau supervisé. Soit p, le nombre total des métriques surveillées.

[1] La détection des pannes repose sur l'envoi périodique (toutes les minutes) d'un paquet ICMP echo request vers tous les routeurs composant le RNU. Si aucune réponse n'est reçue suite à l'envoi de deux paquets ICMP echo request, alors le chemin réseau reliant la sonde de mesures au routeur concerné est considéré en panne.

2. Calculer des estimateurs robustes de la matrice de corrélation et du vecteur moyenne des *p* métriques utilisées.
3. Utiliser l'analyse en composantes principales afin réduire la dimentionalité du problème de détection.
4. Détecter comme anomalie, tout point excentrique par rapport au centre des données, défini par le vecteur moyenne des *p* métriques utilisées.

1.1 Métriques utilisées

Le choix des métriques à utiliser influe sur les types d'anomalies pouvant être détectées. En effet, une anomalie exposée par la surveillance d'une métrique donnée (par exemple le nombre de paquets SYN par intervalle de temps) peut ne pas l'être si l'on s'intéresse à une autre métrique (par exemple le nombre d'octets par intervalle de temps).

De plus, l'expérience des techniciens du CCK, ainsi que les travaux de [Lakh04], montrent que les anomalies illégitimes (propagation des vers informatiques ou les attaques de déni de service distribuées) et légitimes (telles que les foules subites ou les changements de routes) ne peuvent être détectées en surveillant une seule métrique de volume en un seul point du réseau telle que le nombre de paquets ou d'octets. Par exemple, un saut au niveau du nombre de paquets envoyés (sur un lien donné) n'indique pas en soi la présence d'anomalie. En effet, un tel saut n'est anormal que lorsqu'il n'est pas accompagné d'un pic similaire au niveau du nombre d'octets, de celui des paquets sortants ou encore celui des paquets transitant par un autre lien du réseau.

Par conséquent, l'utilisation conjointe de métriques de volume multipoints relatives au nombre d'octets et de paquets (envoyés et reçus) est nécessaire. Ainsi, nous proposons de collecter, au niveau de chaque lien du réseau surveillé, les quatre métriques de volume suivantes :

- Le nombre de paquets entrants par intervalle de temps
- Le nombre de paquets sortants par intervalle de temps
- Le nombre d'octets entrants par intervalle de temps
- Le nombre d'octets sortants par intervalle de temps

Les métriques choisies ont un avantage indéniable puisqu'elles faciles à collecter en utilisant le protocole de gestion de réseau SNMP (Simple Network Management Protocole). En effet, la plupart des équipements réseaux actuels notamment les routeurs contiennent des objets manageables[2]

[2] Ces objets manageables peuvent être des informations matérielles, des paramètres de configuration, des statistiques de performance et autres objets qui sont directement liés au comportement en cours de l'équipement en question.

organisés sous forme d'une base de données appelée MIB (Management Information Base) pouvant être recueillis par une station de gestion de réseau en envoyant des requêtes SNMP adéquates. Dans notre cas, les métriques de volume choisies correspondent aux objets de la MIB standard « IF-MIB » suivants : IFInUcastPkts, IFOutUcastPkts, IFInOctets et IFOutOctets.

En résumé, nous choisissons de collecter périodiquement quatre métriques de volume au niveau de chacun des N liens du réseau supervisé. Dans la suite, nous désignons par p le nombre total de métriques supervisées[3].

1.2 Calcul du vecteur moyenne et de la matrice de corrélation

La première étape, dans la méthode proposée, est de collecter n échantillons des p variables aléatoires supervisées durant une phase d'apprentissage, de préférence dépourvue d'anomalies de trafic. Par la suite, il s'agit de calculer des estimateurs robustes de la matrice de corrélation et du vecteur moyenne des p variables aléatoires en utilisant la méthode de réduction (trimming). En effet, la méthode de réduction consiste à enlever de l'échantillon de données formé par les p variables aléatoires, les β% observations extrémales, c'est à dire celles qui sont les plus éloignées du centre des données selon une certaine mesure de distance. À partir, de l'échantillon réduit (ne contenant pas les observations extrémales), nous calculons les estimateurs robustes du vecteur moyenne et de la matrice de corrélation.

Plus concrètement, nous commençons par calculer les estimateurs conventionnels du vecteur moyenne $\bar{x}$ et de la matrice de covariance S^{-1} à partir de tout l'échantillon de données, puis nous calculons la distance de Mahalanobis di (donnée par l'équation 4.1) pour chaque observation i par rapport au vecteur moyenne arithmétique $\bar{x}$. Ensuite nous identifions les β% observations les plus éloignées du centre des données selon la distance de Mahalanobis, ces dernières sont éliminées avant de recalculer le vecteur moyenne et la matrice de corrélation.

Les estimateurs obtenus par cette méthode sont dits robustes car ils ne sont pas faussés par la présence de valeurs extrémales (trop grandes ou trop petites) des variables aléatoires utilisées.

$$d_i = \sqrt{(x_i - \bar{x})' S^{-1} (x_i - \bar{x})} \qquad \textbf{(4.1)}$$

La distance de Mahalanobis est utilisée dans le trimming car, à la différence de la distance euclidienne où toutes les composantes des vecteurs sont traitées de la même façon, la distance de Mahalanobis accorde un poids moins important aux composantes les plus bruitées [Shyu03].

[3] Pour un réseau composé de N liens supervisés, p vaut 4*N.

1.3 Analyse en composantes principales

La deuxième étape consiste à effectuer une analyse en composantes principales des p variables aléatoires. Nous rappelons que l'analyse en composantes principales essaye d'expliquer la structure de variance-covariance de l'ensemble des variables d'origines à travers des nouvelles variables synthétisées appelées composantes principales. Les composantes principales sont des combinaisons linéaires particulières non corrélées des p variables aléatoires d'origines X_1, X_2, ….., X_p. De plus, les composantes principales ont les propriétés suivantes :

- Les composantes principales peuvent être classées par ordre décroissant d'importance dans le sens suivant : La première composante principale a la plus grande variance, la deuxième composante principale a la seconde plus grande variance et ainsi de suite.
- La variance totale de l'échantillon par rapport aux composantes principales est égale à la variance totale sur les variables aléatoires d'origine.

Afin d'obtenir les composantes principales, une analyse des valeurs propres de la matrice de corrélation R de l'échantillon réduit (obtenu après l'étape de trimming) des p variables d'origine est effectuée. L'intérêt de l'utilisation de la matrice de corrélation au lieu de la matrice de covariance S réside dans le fait que la matrice de corrélation est invariante aux unités utilisées pour mesurer les variables aléatoires d'origines. Le calcul des composantes principales à partir de la matrice de corrélation se justifie aussi par le fait que les trafics sur les liens d'un réseau ont des ordres de grandeurs différents, avec la présence de liens de grandes capacités transmettant un trafic important à coté d'autres liens transportant très peu de trafic.

Soit M la matrice (n x p) composée par n réalisations conjointes des p variables d'origine X_1, X_2, ….., X_p et soit R la matrice de corrélation $(p \times p)$ de ces variables calculée à partir de l'échantillon réduit.

Si (λ_1, e_1), (λ_2, e_2),….., (λ_p, e_P) sont les p couples de valeur propre/ vecteur propre de la matrice R, alors la ième composante principale est donnée par l'équation (4.2).

$$y_i = e_i'(z-\bar{z}) = e_{i1}(z_1-\bar{z_1})+e_{i2}(z_2-\bar{z_2})+\cdots+e_{ip}(z_p-\bar{z_p}), i=1,2,\ldots,p \qquad \textbf{(4.2)}$$

Où

- $\lambda_1 \geq \lambda_2 \geq \ldots \lambda_p \geq 0$
- $e_i'=(e_{i1}, e_{i2}, \cdots, e_{ip})$ est le ième vecteur propre de la matrice R
- $z'=(z_1, z_2, \cdots, z_p)$ est un vecteur quelconque d'observations standardisées. Il est défini par $z_k = \frac{(x_k-\bar{x_k})}{\sqrt{s_{kk}}}$, $k=1, 2, \ldots p$ et où S_{kk} est la covariance de la variable X_k.

- et $\bar{z}'=(\bar{z}_1, \bar{z}_2,\cdots,\bar{z}_p)$ est le vecteur moyenne arithmétique de l'échantillon des *p* variables standardisées.

1.4 Détection des points excentriques

Nous formulons, à l'instar de [Shyu03 et Lakh04], le problème de détection d'anomalies comme un problème de recherche des points excentriques multi-variables. En effet, nous considérons comme anomalie toute observation éloignée du centre des données, défini par le vecteur moyenne arithmétique calculé sur l'échantillon réduit. Pour ce faire nous avons besoin de choisir une mesure de distance entre un vecteur d'observation quelconque $z'=(z_1, z_2,\cdots,z_p)$ et le vecteur moyenne.

À cet effet, nous utilisons la distance de Mahalanobis, car cette distance prend en compte les corrélations entre les variables aléatoires utilisées et ne dépend pas des unités de mesures.

Nous calculons la distance de Mahalanobis d_i, dans l'espace formé par les *p* axes principaux, qui sépare chaque observation i du centre des données. D'après [Shyu03], cette distance est équivalente à la somme des carrés des scores des *p* composantes principales standardisées puisque les composantes principales sont, par définition, non corrélées.

$$d_i^2=\sum_{k=1}^{p}\frac{y_k^2}{\lambda_k}=\frac{y_1^2}{\lambda_1}+\frac{y_2^2}{\lambda_2}+\ldots+\frac{y_p^2}{\lambda_p} \qquad \textbf{(4.3)}$$

À ce stade, le problème de détection d'anomalies revient à définir un seuil de détection pour la distance de Mahalanobis. Au-delà de ce seuil, toute observation sera considérée comme anormale.

Or, l'intérêt de l'analyse en composantes principales réside dans le fait qu'elle permet de réduire la dimensionalité du problème de détection en choisissant parmi les p composantes principales obtenues, celles à retenir. Le problème de détection reviendrait alors à calculer la distance de Mahalanobis d'une observation quelconque i par rapport au centre de données dans l'espace formé les axes principaux retenus puis comparer cette distance par rapport à un seuil prédéfini.

Contrairement à la plupart des méthodes basées sur l'analyse en composantes principales (dont par exemple [Lakh04]), Shyu [Shyu03] propose de retenir à la fois des composantes principales majeures et des composantes principales mineures. En effet, d'après [Shyu03], les observations excentriques par rapport aux axes principaux mineurs correspondent à des points excentriques multi-variables n'apparaissant pas lorsque les variables d'origine sont analysées séparément ; alors que les observations excentriques selon les axes principaux majeurs correspondent à des points excentriques par rapport à une ou à un nombre réduit de variables aléatoires d'origine. C'est pourquoi, nous nous conformons aux choix de Shyu dans [Shyu03] et nous proposons d'utiliser conjointement :

- les q composantes principales majeures qui expliqueraient, ensemble, au moins 50% de la variance totale de l'échantillon,
- et les r composantes principales mineures qui expliqueraient 20% de la variance totale de l'échantillon.

La distance de Mahalanobis d'une observation i par rapport au centre de données, calculée dans l'espace des axes principaux majeurs vaut la somme des carrés des scores des q composantes principales majeures standardisées. De même, la distance de Mahalanobis d'une observation i par rapport au centre de données, calculée dans l'espace des axes principaux mineurs est égale à la somme des carrés des scores des r composantes principales mineures standardisées.

La détection des points excentriques par rapport aux axes principaux majeurs et mineurs revient à marquer comme anomalie toute observation i qui satisfait l'équation (4.4), où c_1 et c_2 sont les seuils de détection.

$$\sum_{k=1}^{q}\frac{y_k^2}{\lambda_k}>c_1 \quad \text{ou} \quad \sum_{k=p-r+1}^{p}\frac{y_k^2}{\lambda_k}>c_2 \qquad \textbf{(4.4)}$$

Les seuils c_1 et c_2 sont déterminés selon le taux α de fausses alarmes maximal toléré. L'inégalité de Cauchy-Shwartz et celle de Bonferroni montrent que ce taux est donné par l'équation 4.5.

$$\alpha_1+\alpha_2-\sqrt{\alpha_1\alpha_2} \le \alpha \le \alpha_1+\alpha_2 \qquad \textbf{(4.5)}$$

Où $\alpha_1=P(\sum_{k=1}^{q}\frac{y_k^2}{\lambda_k}>c_1|$ x est normal) et $\alpha_2=P(\sum_{k=p-r+1}^{p}\frac{y_k^2}{\lambda_k}>c_2|$ x est normal)

Pour donner la même importance aux deux types de points excentriques (par rapport aux axes principaux majeurs et mineurs), nous choisissons $\alpha_1=\alpha_2$.

Ainsi pour un taux de fausses alarmes α=2%, nous choisissons $\alpha_1=\alpha_2=0{,}0101$ ce qui permet de vérifier l'inéquation (4.5).

Le calcul du seuil de détection c_1 s'obtient alors par la détermination des 0,9899 quantiles de la distribution empirique de $\sum_{k=1}^{q}\frac{y_k^2}{\lambda_k}$, celui du seuil c_2 s'obtient de la même façon à partir de la distribution empirique de $\sum_{k=p-r+1}^{p}\frac{y_k^2}{\lambda_k}$.

1.5 Récapitulatif de la méthode proposée

En résumé, la méthode de détection d'anomalies, décrite plus haut, nécessite deux phases : une phase d'apprentissage et une phase de détection. Durant la phase d'apprentissage, une trace de mesures composée par n observations est collectée. Chaque observation est composée par les valeurs de *4*N* variables MIB (avec N est le nombre de liens du réseau supervisé). À partir de cette trace, la méthode consiste à effectuer les étapes suivantes en différé :

1. Calculer un estimateur robuste du vecteur moyenne et de la matrice de corrélation des *4*N* variables aléatoires en utilisant la méthode de « trimming »;
2. Effectuer une analyse en composantes principales de la matrice des observations de la trace d'apprentissage ,
3. Sélectionner les composantes principales à retenir ;
4. Déterminer les valeurs des seuils de détection c_1 et c_2 à partir des distributions empiriques;

Par la suite, durant la phase de détection, il s'agit d'effectuer les étapes suivantes en temps réel :

1. Collecter, à intervalle régulier, 4 variables MIB pour chacun des N liens du réseau supervisé ;
2. Calculer, pour chaque observation *i*, la distance de Mahalanobis entre cette observation et le vecteur moyenne estimé, dans l'espace des axes principaux majeurs puis dans celui des axes principaux mineurs ;
3. Comparer les distances calculées par rapport aux seuils de détection *c1* et *c2*. Si au moins, l'une des distances calculées est supérieure au seuil de détection, alors l'observation en question sera marquée comme anormale. Autrement, elle sera considérée normale.

Concernant le coté implémentation, nous avons utilisé le logiciel de gestion de réseau WhatsUP [Whatsup] pour collecter périodiquement les valeurs des variables MIB nécessaires. Par la suite, nous avons implémenté dans le logiciel Matlab, l'algorithme de détection décrit plus haut afin de détecter, en différé, les anomalies de volume à partir des traces de trafic collectées.

2. Évaluation de la méthode proposée

Le but de l'évaluation, que nous proposons dans cette section, est de déterminer les performances de détection de la méthode face à la présence d'anomalies illégitimes. En d'autres termes, nous cherchons à évaluer son taux de détection et son taux de fausses alarmes. Pour ce faire, nous avons besoin d'une trace dans laquelle les anomalies de trafic sont bien identifiées. Pour y parvenir, nous avons choisi d'injecter des anomalies illégitimes (balayage de ports et attaques de déni de service) sur un réseau d'expérimentation afin d'obtenir une trace étiquetée avec précision.

2.1 Scénario de test

Le réseau d'expérimentation utilisé pour la génération de trace de trafic étiquetée est décrit par la Figure 4.1. Il est formé par deux sous-réseaux connectés entre eux par un routeur ; il comporte principalement une machine victime qui subit des attaques générées par deux machines « pirates » et une station pour la collecte périodique des variables MIB nécessaires à la détection d'anomalies.

Le réseau d'expérimentation était installé au CCK ; il était connecté à Internet et incluait, en plus, quelques machines qui étaient utilisées par nos collègues afin de générer un trafic Internet normal. À ce trafic, nous avons ajouté un trafic normal simulé grâce au logiciel LANTraffic [Lantraffic]. En effet, nous avons utilisé LanTraffic pour maintenir 16 connexions bidirectionnelles TCP ou UDP[4] entre la station de génération de trafic et la machine victime.

Pour injecter un trafic anormal dans le réseau d'expérimentation, nous avons utilisé les techniques d'attaques suivantes : les techniques Smurf et Syn Floading pour la génération d'attaques de déni de service sur la machine victime et le balayage de ports à destination des machines composant le réseau d'expérimentation[5]. Pour ce faire, nous avons utilisé les programmes libres PingIcmp[6], Synfload[7] et Nmap[8].

La période d'expérimentation était d'une heure, durant laquelle les quatre variables MIB (nombre de paquets en entrée/ en sortie et nombre d'octets en entrée/ en sortie du routeur d'interconnexion,) ont été recueillies par la station de gestion de réseau WhatsUP, toutes les 20 secondes et enregistrées dans un fichier trace. Trois attaques ont été lancées automatiquement durant cette période à des instants prédéfins à partir de la machine « pirate » qui a été préalablement synchronisée avec la station de gestion de réseau grâce au protocole NTP.

[4] Ces connexions sont complètement paramétrables (Temps d'inter-arrivées des paquets, loi de génération des connexions, taille des paquets,..).

[5] Voir l'annexe B pour les détails de fonctionnement de ces trois types d'attaques.

[6] http://www.frameip.com/Pingicmp

[7] http://www.frameip.com/synfload

[8] Nmap est un scanner d'adresses IP et de numéros de ports libre ; il permet de détecter les ports ouverts, d'identifier les services hébergés et d'obtenir des informations sur le système d'exploitation d'un ordinateur distant.

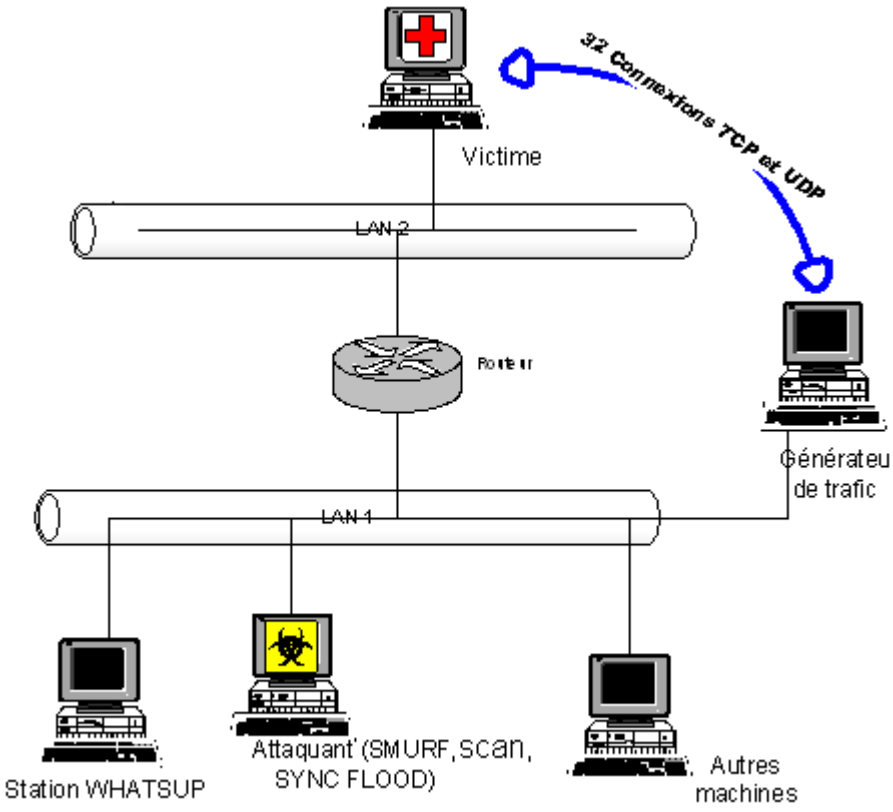

Figure 4.1 : Réseau d'expérimentation

Enfin la détection d'anomalies affectant ce réseau d'expérimentation est effectuée en différé à partir de la trace obtenue. Pour ce faire, nous avons utilisé les paramètres suivants :

- La trace utilisée pour l'apprentissage est de durée 10 minutes ;
- Le taux de « trimming » utilisé pour l'estimation robuste de la matrice de corrélation et du vecteur moyenne des 4 variables aléatoires d'origines est fixé à 0,5% ;
- Le taux de variance minimale expliquée par les composantes principales majeures est de 50% ;
- Le taux de variance minimale expliquée par les composantes principales mineures est de 20% ;

2.2 Résultats d'évaluation

Pour évaluer les performances de détection de la méthode proposée, nous avons calculé les métriques décrites par Lazarvec dans [Laza03]. Il s'agit des métriques générales suivantes : taux de faux positifs, taux de détection et précision ; ainsi que les métriques par attaque taux de détection par rafale (bdr) et temps de réponse (Trep)[9].

[9] Les définitions de toutes ces métriques sont données dans le chapitre II section 4.1. Le taux de détection, la précision et le taux de détection par attaque (bdr) sont tous mesurés en % ; alors que le temps de réponse

Les résultats de l'évaluation sont présentés dans les tableaux 4.1 et 4.2 suivants. Nous remarquons, à partir du Tableau 4.1, que le fait d'augmenter le taux fixé de fausses alarmes, permet d'améliorer le taux de détection, mais en contre partie, cela a pour effet diminuer la précision.

Taux fixé de fausses alarmes	**2%**	**4%**	**6%**
Taux de faux positifs	1,14%	1,71%	2,54%
Taux de détection	47,14%	62,86%	68,57%
Précision	91,67%	56,41%	41,74%

Tableau 4.1: Variation des métriques générales en fonction du taux de fausses alarmes

Le Tableau 4.2 montre que certaines attaques sont mieux détectées que d'autres. En effet, les attaques Smurf et Syn Fload sont correctement détectées avec un taux de détection (bdr) proche voir égal à 100%, c'est à dire que presque toutes les observations contenant l'attaque ont été correctement détectées comme anormales. De plus la détection a été effectuée très rapidement (Trep est inférieur ou égal 1), c'est à dire que la détection a été effectuée à l'instant même où l'attaque s'est produite (ou lors de la collecte de l'échantillon suivant). Par contre l'attaque Scan a été mal détectée puisque son taux de détection est faible (bdr<50%) et son temps de réponse (Trep) est important.

L'incapacité de la méthode proposée à détecter l'attaque scan s'explique par sa faible intensité, puisque nous avons utilisé un seul processus de scan. Nous estimons que le taux de détection des attaques de types scan dues à la propagation des vers informatiques via les réseaux, sera sans doute meilleur car ces derniers font intervenir une multitude (parfois des centaines) de processus légers (threads) de scan automatique à la recherche de machines vulnérables pour leur envoyer le code du vers en question.

	Taux fixé de fausses alarmes					
	2%		4%		6%	
	bdr	Trep	bdr	Trep	bdr	Trep
Smurf	93%	1	97%	0	97%	0
SYN flood	100%	0	100%	0	100%	0
SCAN	3%	19	32%	19	44%	3

Tableau 4.2 : Variation des métriques par attaque en fonction du taux de fausses alarmes

correspond au temps écoulé entre le début d'une attaque et sa détection par le système, il est mesuré en nombre d'observations.

À l'issue de cette évaluation, nous estimons que la méthode de détection est suffisamment fiable pour être utilisée pour la détection des anomalies dans un réseau de production. Pour cela, nous choisissons un taux fixé de fausses alarmes de 2% qui nous permet d'avoir une bonne précision (>90%).

3. Détection d'anomalies de volume dans le réseau RNU

Afin de détecter les anomalies affectant le réseau RNU, nous avons besoin de collecter à intervalle régulier, quatre variables MIB pour chacun des N liens composant ce réseau. Au moment de la réalisation des tests décrits dans ce paragraphe, l'architecture du réseau RNU comportait plus de 160 liens. De plus, elle était centralisée autour d'un nœud unique CCK el Manar (comme le montre la Figure 3.2). Ce dernier était structuré autour d'un pare-feu central interconnectant 6 sous-réseaux. C'est pourquoi, nous avons opté pour la collecte de 4 variables MIB pour chacun des 6 interfaces du pare-feu du nœud CCK el Manar. En effet, ces interfaces reçoivent la majorité du trafic véhiculé par le réseau RNU (le trafic inter-institutions universitaires est très faible, puisque ces dernières n'ont pas de serveurs hébergés localement), ainsi toute anomalie affectant ce réseau a théoriquement des effets propagés jusqu'au ces 6 interfaces.

Bien que la plupart des travaux utilisant les variables MIB ([Brut00, Barf02 et Lakh04]) pour la détection d'anomalies, adoptent une période de collecte de cinq minutes, nous avons préféré l'utilisation d'une période plus courte afin de pouvoir détecter aussi bien les anomalies de longue durée que celles de plus courtes durées. De plus, l'utilisation de période de collecte courte permet une précision dans l'identification des instants de début et de fin d'anomalie. Par conséquent, nous avons choisi de collecter les métriques de volume nécessaires à la détection (24 variables MIB) toutes les minutes. Pour ce faire, nous avons utilisé le logiciel WhatsUP, qui a produit un fichier trace composée de plus de 64000 observations correspondant à une période de 45 jours (du 03/04/2004 au 18/05/2004).

Afin de détecter les anomalies affectant le réseau RNU, nous avons fixé les paramètres de détection comme suit :

- Taux de variance minimale expliquée par les composantes principales majeures est de 50% ;
- Taux de variance minimale expliquée par les composantes principales mineures est de 20% ;
- Taux de réduction (trimming) de 0,5 % ;
- Taux fixé de fausses alarmes de 2% ;
- Les 20 000 premières observations sont utilisées durant la phase d'apprentissage.

Dans ces conditions, nous avons obtenu, grâce à l'analyse en composantes principales, une réduction importante (50%) du nombre de variables aléatoires nécessaires à la détection. En effet, ces dernières

sont passées de 24 à seulement 12 (7 composantes principales majeures et 5 mineures). De plus, le nombre d'observations anormales détectées s'élève à 799, pour toute la période d'expérimentation.

Afin d'étudier les caractéristiques des anomalies affectant le réseau RNU, nous avons défini deux métriques temporelles : la distribution de la durée des anomalies et la distribution du temps inter-anomalies.

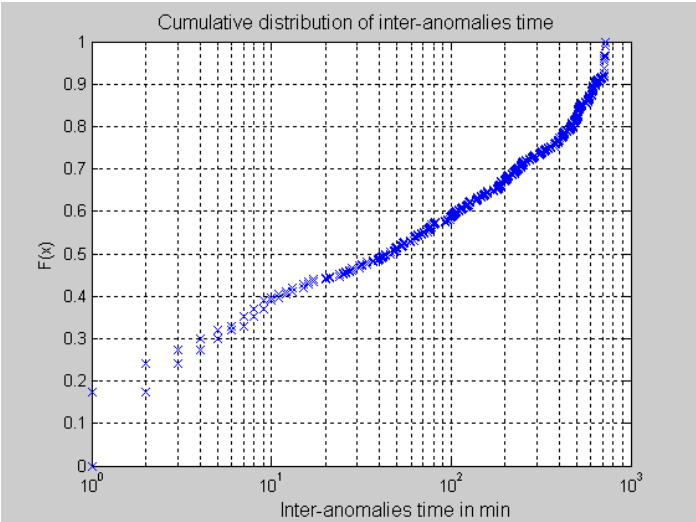

Figure 4.2: Distribution cumulative du temps inter-anomalies

La Figure 4.2 présente la distribution cumulative du temps d'inter-anomalies ; elle montre que 75% des anomalies sont suivies par la survenue d'une nouvelle anomalie au bout d'un temps inférieur à 60 minutes. Par conséquent nous pouvons déduire que les anomalies sont très fréquentes.

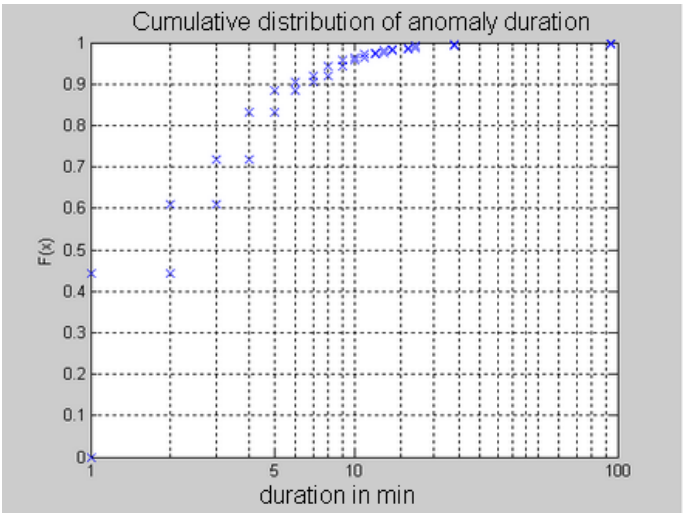

Figure 4.3 : Distribution cumulative de la durée d'anomalies

La Figure 4.3, présentant la distribution cumulative de la durée des anomalies, montre que la majorité des anomalies sont de courte durée, puisque 90% des anomalies affectant le RNU ont une durée inférieure à 5 minutes et 50 % des anomalies ont une durée d'une minute seulement.

4. Utilisation des fenêtres glissantes

Le nombre d'observations anormales détectées, dans le réseau RNU, durant la période d'expérimentation (45 jours) est de 799 pour 64 000 observations ; ce qui revient à une moyenne de plus de 17 alarmes par jour. Ceci complique énormément la tache de l'administrateur du réseau qui doit effectuer des investigations pour identifier les causes de chaque anomalie détectée.

Par ailleurs, la présence d'anomalies de trafic fréquentes, mais de courte durée, n'est pas une spécificité du réseau RNU, puisque cela a été constaté dans toutes les études métrologiques s'intéressant aux anomalies dans les réseaux Internet [Lakh04 et Pang04]. Par conséquent, il est important de pouvoir sélectionner parmi les anomalies détectées, celles qui sont les plus intéressantes, afin de faciliter la tache de l'administrateur réseau.

Pour ce faire, nous proposons d'effectuer la détection d'anomalies sur des fenêtres glissantes de taille fixe *w*, avec un glissement égal à une observation. Ainsi, nous marquons comme anormale toute fenêtre contenant au moins *s* observations anormales parmi *w*.

Pour une taille de fenêtre glissante w = 30 observations, nous avons tracé la Figure 4.4 qui montre l'évolution au cours du temps du nombre d'observations anormales par fenêtre de 30 observations. Il apparaît ainsi que les anomalies sont réparties sur toute la période d'expérimentation, mais que le nombre de fenêtres composées d'une majorité d'observations anormales est faible.

Par conséquent, fixer un seuil de détection s à 20, permet d'obtenir une réduction importante du nombre d'anomalies détectées dans le réseau RNU. En effet, ce dernier passe de 799 observations anormales à seulement 9. Ainsi l'administrateur n'aura à traiter qu'un petit nombre d'anomalies. De plus, sachant leur instant d'apparition, il pourra mener des investigations pour identifier leurs causes et implémenter les solutions adéquates.

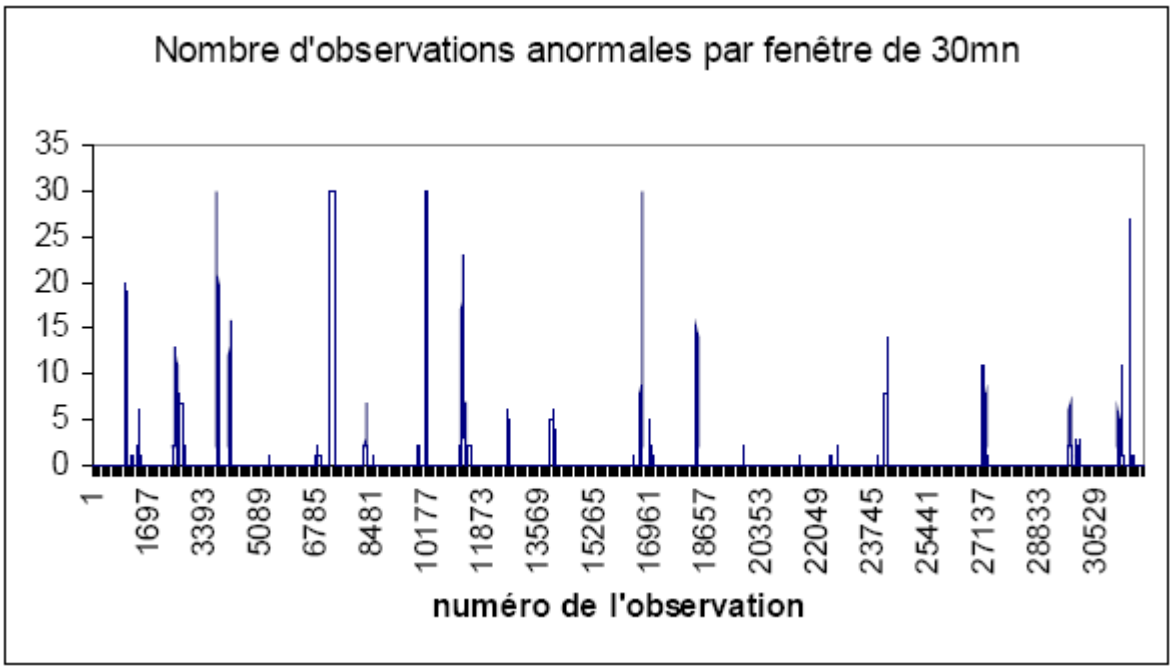

Figure 4.4: Nombre d'observations anormales par fenêtre glissante de 30 minutes

5. Conclusion

Nous avons proposé, dans ce chapitre, une méthode de détection d'anomalies de volume qui présente trois avantages indéniables. D'abord, il s'agit d'une méthode de détection d'anomalies non supervisée, c'est à dire qu'elle n'a pas besoin de trace d'apprentissage préalablement étiquetée. De plus, grâce à la technique de « trimming », elle n'a même pas besoin de trace d'apprentissage composée exclusivement de trafic normal.

Le deuxième avantage, réside dans son aspect complètement empirique, puisqu'elle n'émet aucune hypothèse sur les lois statistiques des variables aléatoires utilisées et calcule les seuils de détection à partir des distributions empiriques.

Le dernier réside dans la nature même des données utilisées pour la détection d'anomalies. En effet, il s'agit de métriques de volume multipoints faciles à collecter (via le protocole SNMP) et peu volumineuses.

Nous estimons que ces trois caractéristiques rendent notre approche adaptée à une utilisation au niveau des backbones en particulier celui du RNU.

Par ailleurs, l'exploitation de cette méthode pour la détection d'anomalies dans le réseau RNU a permis de mettre en évidence l'importance des anomalies de trafic dans ce réseau. En effet, ces dernières sont très fréquentes et majoritairement de courte durée. De plus, elles sont réparties sur toute la période d'expérimentation. Nous rappelons, que ces constatations s'accordent avec les

résultats de l'étude métrologique du trafic RNU présentée dans le troisième chapitre[10], ainsi qu'avec les résultats des études métrologiques récentes décrites dans la littérature [Yegn03, Lakh04, Pang04 et Allm07].

Enfin, le fait que les anomalies soient omniprésentes dans le trafic Internet, rend les approches de détection d'anomalies peu intéressantes d'un point de vue pratique puisqu'elles aboutissent à la génération d'un grand nombre d'alarmes. Pour remédier à ce problème nous avons proposé l'utilisation des fenêtres glissantes ce qui a permis de réduire considérablement le nombre d'alarmes générées. La contrepartie de ce gain c'est que l'approche est devenue limitée à la détection des anomalies de longue durée.

Face à l'omniprésence des anomalies, nous proposerons dans le chapitre suivant une nouvelle démarche. Il s'agit, d'abord, d'analyser le trafic des adresses IP invalides et celui des adresses inutilisées, afin d'en déduire la composition du trafic malicieux. Par la suite, nous nous intéressons à une composante essentielle du trafic malicieux, les balayage de ports, et nous proposons une nouvelle approche de surveillance de ce trafic.

[10] L'étude métrologique du trafic RNU a montré l'importance des connexions suspectes notamment les connexions semi ouvertes ou celles utilisant les ports « windows ».

Chapitre 5:Surveillance du trafic malicieux

Dans ce chapitre, nous présentons une analyse du trafic malicieux dans RNU, avant de proposer un estimateur simple du volume de ce trafic pouvant être utilisé comme indicateur simple du niveau des menaces affectant n'importe quel réseau connecté au réseau mondial. Nous proposerons par la suite, une nouvelle approche de surveillance du trafic de scan permettant de détecter les sauts dans les distributions de ce trafic et de relever ainsi la présence de nouveaux vers informatiques se propageant via le réseau.

1. Caractérisation du trafic malicieux dans RNU

Afin d'étudier les caractéristiques du trafic malicieux dans le réseau RNU, nous avons décomposé les traces de trafic T3 et T4, collectées en avril 2006, selon le type des adresses IP contenues dans les entêtes de leurs paquets. En effet, nous définissons trois types d'adresses IP qui sont : les adresses IP invalides ou « bogon », celles « inutilisées » et enfin les adresses IP valides.

Le premier type représente prés de 40% de l'espace d'adressage IPV4 mondial [Cidr], il correspond à l'ensemble des adresses IP réservées par l'IANA pour des usages spécifiques (telles que les plages d'adresses IP privées définies dans le RFC1918), ainsi que celles non assignées à aucun registre Internet régional.

Le deuxième type correspond aux adresses IP allouées au réseau RNU par l'Agence Tunisienne d'Internet mais qui ne sont pas redistribuées à des institutions universitaires, au moment de la collecte du trafic. En avril 2006 (date de collecte des traces T3 et T4), sur les 21 réseaux / 24 alloués au réseau RNU, 648 adresses appartiennent à cette catégorie, soit un pourcentage de 12,7 % du nombre total d'adresses IP du RNU. Ces adresses sont, contrairement aux adresses invalides, entièrement routées sur Internet.

Le dernier type comprend toutes les autres adresses IP.

Ainsi, nous obtenons, pour chaque trace de trafic collectée sur le réseau RNU, trois nouvelles sous-traces qui correspondent aux trafics suivants: invalide, inutilisé et valide. La sous-trace de trafic invalide contient l'ensemble des paquets ayant comme adresses sources ou destinations des adresses IP invalides. Celle du trafic «inutilisé» contient tous les paquets dont les adresses sources ou destinations appartiennent à la plage d'adresses IP allouée au RNU, mais non exploitée. Enfin, la sous-trace de trafic « valide » contient tous les paquets restants.

Nous proposons dans ce qui suit d'examiner les caractéristiques du trafic des deux premières sous-traces. En effet, ces trafics sont par nature même suspects, puisque les adresses IP invalides et inutilisées ne doivent pas figurer lors d'utilisation normale du réseau. Par conséquent, il est légitime de considérer ces trafics comme un sous-ensemble du trafic malicieux affectant le réseau.

1.1 Variation au cours du temps des trafics « inutilisé » et « invalide »

Le Tableau 5.1, montre que le trafic lié aux adresses IP inutilisées représente à peine 1,2 % des octets transitant par le lien connectant le réseau RNU à l'Internet, mais plus de 8% des connexions TCP ; ce qui laisse deviner que ce trafic est essentiellement composé par des connexions de petite taille (des souris).

Le trafic lié aux adresses invalides est beaucoup plus faible puisqu'il ne représente que 0,07% des octets, presque autant des paquets et seulement 0,04% des connexions. Ceci est dû au fait que ces dernières sont connues de tous et de ce fait, elles sont, le plus souvent, évitées par les logiciels de balayage de ports automatiques et les vers informatiques. En contre partie, elles sont principalement utilisées par des personnes ou organismes impliquées dans des activités illicites (attaques ciblées, envoi massif de mails non sollicités,..) afin de cacher leurs identités.

	Trafic inutilisé			**Trafic invalide**			**Trafic total**		
	Moctets	Pkts (10^6)	Conn (10^3)	Moctets	Pkts (10^6)	Conn (10^3)	Goctets	Pkts (10^6)	Conn (10^3)
#	262	5,5	859,2	160	0,2	4,4	218,9	347,6	10,4
%	1,20	1,60	8,23	0,07	0,06	0,04	100	100	100

Tableau 5.1 : Répartition du trafic par type d'adresses IP (T4)

Les valeurs moyennes (données dans le Tableau 5.1) sont calculées sur toute la durée de la trace T4 (24 heures) ; de ce fait elles peuvent masquer des fluctuations importantes durant la journée. Pour vérifier cette hypothèse, nous avons tracé les figures 5.1 et 5.2 qui présentent respectivement l'évolution du trafic lié aux adresses IP inutilisées et celui des adresses invalides. Ainsi, nous pouvons constater que ces trafics présentent plusieurs pics importants. En effet, le volume maximal du trafic inutilisé atteint 200 paquets/s ou 1,5 Mbits/s (ce qui correspond à 14% du volume du trafic total mesuré en paquets) ; Alors que celui généré par les adresses invalides représente jusqu'à 1,7% des paquets.

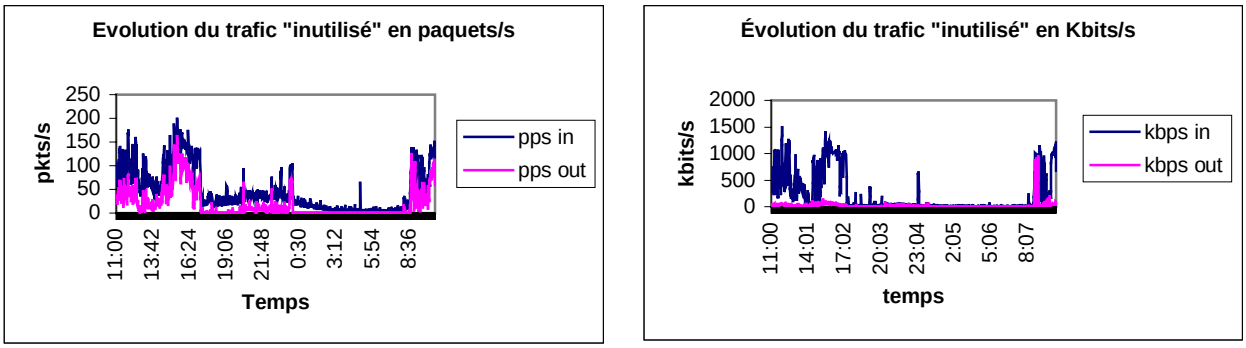

Figure 5.1: Évolution du trafic des adresses IP inutilisées (T4)

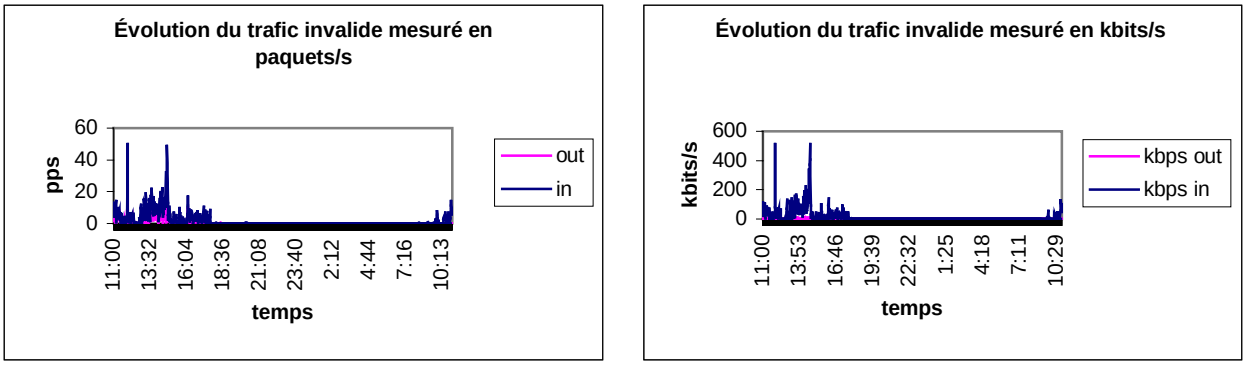

Figure 5.2 : Évolution du trafic des adresses IP invalides (T4)

Par ailleurs, nous remarquons, d'après la Figure 5.1, que le trafic des adresses IP inutilisées est présent tout au long de la période de collecte, ce qui montre l'omniprésence des activités malicieuses dont il est le reflet. Mais, ce qui nous paraît de prime abord bizarre c'est la présence d'un trafic sortant relativement important. Après investigations, il semble que ce trafic résulte essentiellement de l'existence de machines mal-configurées (utilisant ces plages d'adresses et essayant de se connecter à l'Internet) ; toutefois ces investigations ont montré l'existence de quelques adresses IP, classées comme inutilisées mais qui sont apparemment valides.

Enfin, nous signalons, d'après la Figure 5.2, que le trafic des adresses IP invalides est nul par moments surtout la nuit, ce qui reflète le caractère ponctuel des activités malicieuses liées à ce trafic.

1.2 Répartitions des trafics « inutilisé » et « invalide »

L'étude de la répartition du trafic relatif aux adresses IP inutilisées par protocole (voir Tableau 5.2) montre que la majorité de ce trafic utilise le protocole TCP. Ce qui nous paraît tout à fait logique ; En effet TCP est utilisé par la majorité des applications de l'Internet, ce qui en fait une cible privilégié des pirates qui essayent d'exploiter les vulnérabilités sous-jacentes à ce protocole ou aux applications qui l'utilisent afin de perpétuer toute sorte d'activités.

Nous remarquons aussi que la taille moyenne des paquets TCP est légèrement supérieure à la taille minimale de paquet, ce qui laisse deviner que ce trafic est essentiellement composé par des demandes de connexions (paquets SYN) et ne comporte pas ou peu de données. Alors que la taille moyenne des paquets UDP et ICMP est relativement grande, ce qui signifie que ces protocoles sont utilisés pour le transfert de données, ceci est tout à fait possible car UDP et ICMP sont des protocoles non connectés.

Protocole	% Paquets	% Octets	Taille moyenne des paquets (octets)
tcp	95,6	75,4	48
udp	3,8	23,8	498
icmp	0,6	0,8	101

Tableau 5.2 : Répartition du trafic «inutilisé» par protocole (T4)

La répartition du trafic UDP «inutilisé» par numéro de port montre que la quasi-totalité de ce trafic (94% des paquets UDP) utilise les ports UDP 1026 et 1027. Ces ports sont utilisés par le service Windows « messenger » qui permet d'envoyer des messages texte non sollicités à n'importe quelle machine Windows[1]. Ce service est très prisé par les pirates afin de lancer des compagnes de spam, voire même de phishing.

Concernant le trafic «inutilisé» ICMP, nous signalons que le fait que les traces de trafic soient limitées aux entêtes des paquets et que le protocole ICMP n'utilise pas les numéros de ports, réduit considérablement nos possibilités d'investigation et ne nous permet pas d'élucider avec précision la nature de ce trafic. Toutefois, il est fort possible qu'il soit le résultat de communications entre des machines zombies et les pirates qui les contrôlent.

Concernant la répartition du trafic TCP « inutilisé » par numéro de port (voir Figure 5.3), nous remarquons qu'elle est complètement différente de celle du trafic Internet total (illustrée par la Figure 3.9 et la Figure 3.10). En effet, prés de 94% des paquets et 74% des octets de ce trafic utilisent les ports « Windows »[2]. Ceci s'explique par le fait que les services Windows associés à ces ports sont par défaut en écoute sur toutes les versions du système d'exploitation Windows ; de plus ils constituent des sources importantes de vulnérabilités. Ces dernières sont exploitées non seulement par un grand nombre de vers informatiques afin se propager rapidement via les réseaux ; mais aussi

[1] Ces messages apparaissent dans des fenêtres surgissantes (popup) affichant un lien vers un site web commercial. Ils peuvent être envoyés par la commande « net send » lancée depuis l'invite de commande MS-DOS de Windows.

[2] Il s'agit des ports TCP 135, 137, 139 et 445.

par les pirates informatiques pour prendre le contrôle de centaines de machines vulnérables et former ainsi des réseaux de zombies (botnets).

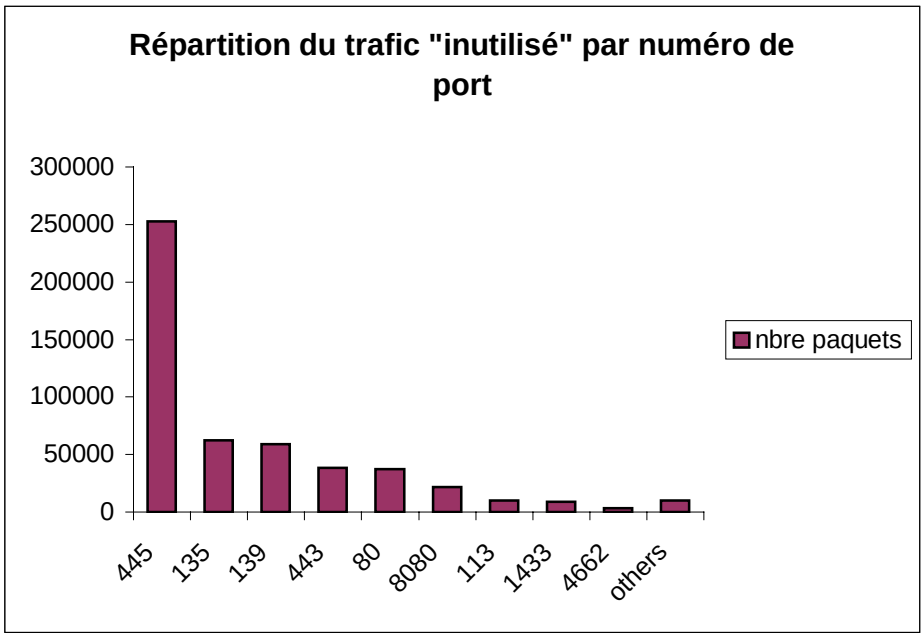

Figure 5.3 : Répartition du trafic « inutilisé » par numéro de port TCP (T4)

Par ailleurs, nous notons que le site web du projet Internet Storm Center [ISC], continue de classer les ports de partage de fichiers Windows ; ainsi que ceux de « messenger » dans le top 10 des ports des plus ciblés par des tentatives d'attaques. ; et ce depuis 2003 à nos jours (octobre 2009).

Par conséquent, le filtrage du trafic lié à ces ports est une nécessité ; il doit être effectué le plus prés possible des sources de ce trafic afin de limiter leur exploitation malveillante et réduire le volume du trafic indésirable dans le réseau. Pour ce faire, il est recommandé d'implémenter des ACLs au niveau de tous les routeurs du RNU, aussi bien au niveau de ceux installés dans les institutions universitaires que ceux installés au niveau des POPs du RNU.

Enfin, l'étude de la répartition du trafic « invalide » par protocole montre qu'il utilise essentiellement TCP (plus de 98% des paquets et des octets). De plus, la quasi-totalité de ce trafic (98% des paquets TCP) utilise les ports 80 et 8080. Ainsi, il apparaît que la répartition de ce trafic est complètement différente de celle du trafic « inutilisé » mais ressemble à celle du trafic total.

1.3 Analyse des connexions TCP des trafics « inutilisé » et « invalide »

À partir des traces du trafic des adresses IP inutilisées et celles des adresses invalides, nous avons reconstitué les connexions TCP qui les composent, puis nous les avons classées par type. Pour cela, nous avons utilisé les types, déjà adoptés pour l'étude tu trafic total (la définition de ces types est donnée dans le chapitre 3, paragraphe 7.1). Les tableaux 5.3 et 5.4 présentent les résultats de cette classification.

Il apparaît ainsi, que la quasi-totalité du trafic « inutilisé » (94,2% des connexions) est composée par des connexions semi-ouvertes entrantes (c'est à dire formées par des paquets SYN isolés envoyés depuis Internet). Ces dernières sont essentiellement le résultat des balayages de ports (scan) perpétués par des vers informatiques cherchant des nouvelles cibles à infecter et des activités de reconnaissance effectuées par des pirates qui essayent par ce biais de collecter le maximum d'informations sur leurs prochaines cibles ou de créer des réseaux de zombies.

Type de connexion	**Nombre de flux [%]**		
Complète	2		
Annulée	1,7		
Incomplète	Paquets SYN	Paquets SYN/ACK uniquement	Autres
	94,2	0,2	1,9

Tableau 5.3 : Répartition des connexions TCP « inutilisées » par type (T4)

Type de connexion	**Nombre de flux [%]**		
Complète	31,4		
Annulée	23,3		
Incomplète	Paquets SYN	Paquets SYN/ACK uniquement	Autres
	2,4	4,5	38,4

Tableau 5.4 : Répartition des connexions TCP « invalides » par type (T4)

La répartition des connexions TCP du trafic « invalide » est complètement différente. En effet 31,4 % de ces connexions sont complètes, ce qui représente un pourcentage supérieur à celui calculé à partir de la trace du trafic total (24,5%). Alors que les balayages de ports (scans) représentent 2,4% des connexions « invalides » et le trafic « backscatter »[3] seulement 4,5%. Le faible nombre de balayage de ports envoyés vers les adresses invalides est tout à fait logique, puisque ces adresses sont évitées par les vers et programmes de scans automatiques.

Par ailleurs, nous notons que les connexions complètes, liées aux adresses IP invalides, ont abouti à un échange d'informations. Malheureusement, le fait que les traces disponibles sont limitées aux

[3] Il s'agit du trafic de réponse envoyé par les machines victimes d'attaques de déni de service réseau de type Syn floading.

entêtes IP / TCP / UDP, ne nous permet pas d'élucider avec précision la nature de ces connexions ni leurs objectifs. Mais, nous soupçonnons qu'elles représentent des communications entre des contrôleurs de botnets (botnet master) usurpant des adresses IP invalides, et des machines sous leur contrôle.

1.4 Conclusion

Les analyses présentées précédemment montrent que les trafics « inutilisé » et « invalide » ont des caractéristiques complètement différentes et reflètent des activités malicieuses distinctes. En effet, le trafic « inutilisé » entrant reflète principalement les activités suivantes :

1. Balayages de ports engendrés par la propagation des vers informatiques sur le réseau et d'activités de reconnaissance par des pirates. Ces scans ciblent principalement les ports Windows.
2. Envoi massif de messages non sollicités en utilisant le protocole UDP et le service « messenger » de Windows.
3. Trafic « backscatter » résultant de la présence d'attaques de déni de service usurpant des adresses IP inutilisées et ciblant des serveurs externes.

Alors que le trafic « inutilisé » sortant expose principalement la présence de machines mal-configurées.

Par ailleurs, le trafic « invalide » reflète particulièrement les activités malicieuses des réseaux de zombies.

Par conséquent nous pouvons conclure que le trafic « inutilisé », reflète principalement des activités non ciblées alors que celui « invalide » expose la présence d'activités malicieuses ciblées. Ce constat est absolument conforme aux études métrologiques de ces trafics présentées dans l'état de l'art [Pang04 et Feam05].

2. Répartition du trafic « inutilisé » par adresse IP

L'analyse précédente a montré que trafic « inutilisé » entrant reflète essentiellement des activités malicieuses à caractère non ciblé. Par conséquent, il est légitime de s'attendre à ce que toutes les adresses du RNU, utilisées ou non, soient des cibles potentielles de ces attaques. En effet, les vers informatiques, les programmes de balayage de ports automatiques et les logiciels d'envoi de messages de spam, n'ont pas une connaissance préalable sur la répartition des adresses IP du RNU, de ce fait ils vont cibler n'importe quelle adresse IP.

Pour vérifier cette hypothèse, nous proposons d'étudier la répartition du trafic «inutilisé» par adresse IP.

L'analyse des sources du trafic « inutilisé » (4336 adresses IP différentes pour la trace T4) montre que la grande majorité de ce trafic est émis par adresses IP appartenant au réseau Internet Tunisien.

En effet, la Figure 5.4 montre que la répartition des adresses sources du trafic inutilisé est dense au alentour des adresses IP allouées à la Tunisie et éparse, voire vide partout ailleurs.

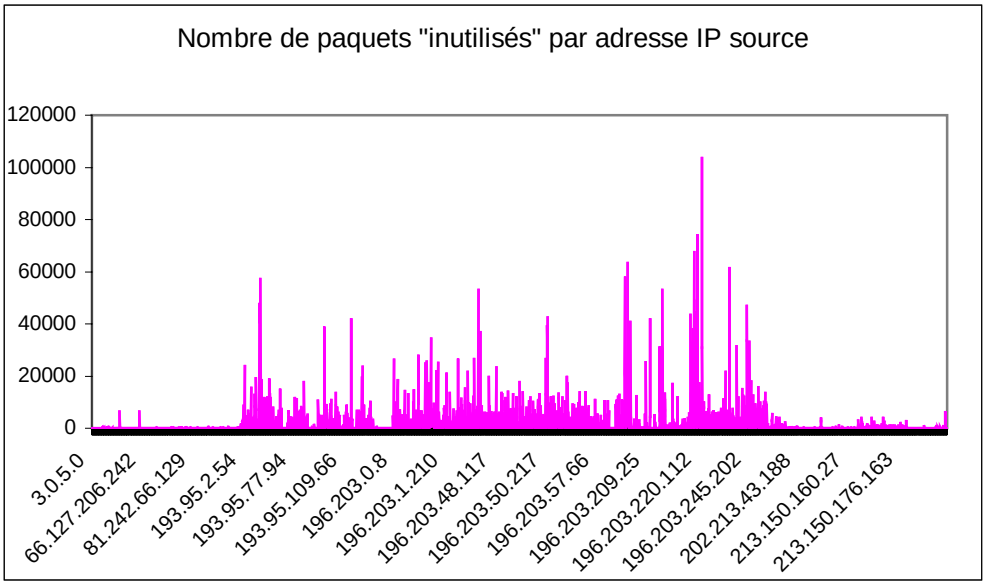

Figure 5.4 : Répartition du trafic « inutilisé » par adresse IP source (T4)

Par ailleurs, les plages d'adresses IP allouées au réseau RNU sont composées par de nombreux sous-réseaux /24 (en Avril 2006, il y avait 21 préfixe /24) qui sont pour la plupart partiellement utilisés. Ainsi, l'espace des adresses IP inutilisées (12,7% de l'espace d'adressage RNU) est composé par de nombreux petits sous-réseaux non adjacents. La Figure 5.5 illustre la répartition du trafic « inutilisé » selon les adresses IP destination. Cette figure est limitée, pour des raisons de lisibilité, à deux réseaux /24 partiellement inutilisés, elle concerne le fichier de trace (T4). Nous constatons qu'à l'intérieur d'un même réseau /24, les répartitions du nombre de paquets et de connexions par adresse IP inutilisée sont quasi-uniformes (c'est à dire que les adresses IP inutilisées appartenant au même réseau /24 ont reçu presque le même nombre de paquets et demandes de connexions TCP). Alors que ces répartitions changent complètement, lorsqu'on considère un réseau /24 différent.

Le fait que certains réseaux /24 semblent être plus ciblés par les applications malveillantes que d'autres, s'explique par le fait que certaines de ces applications (notamment les vers informatiques adoptant des stratégies de scan évoluées afin d'infecter le maximum de machines le plus rapidement possible) ne choissent pas leurs cibles d'une manière complètement aléatoire parmi l'espace d'adressage IP, mais privilégient les adresses IP appartenant au même réseau /24 ou /16 qu'eux. Ceci a été mis en évidence notamment dans le vers code Red II et Nimda [Zesh03].

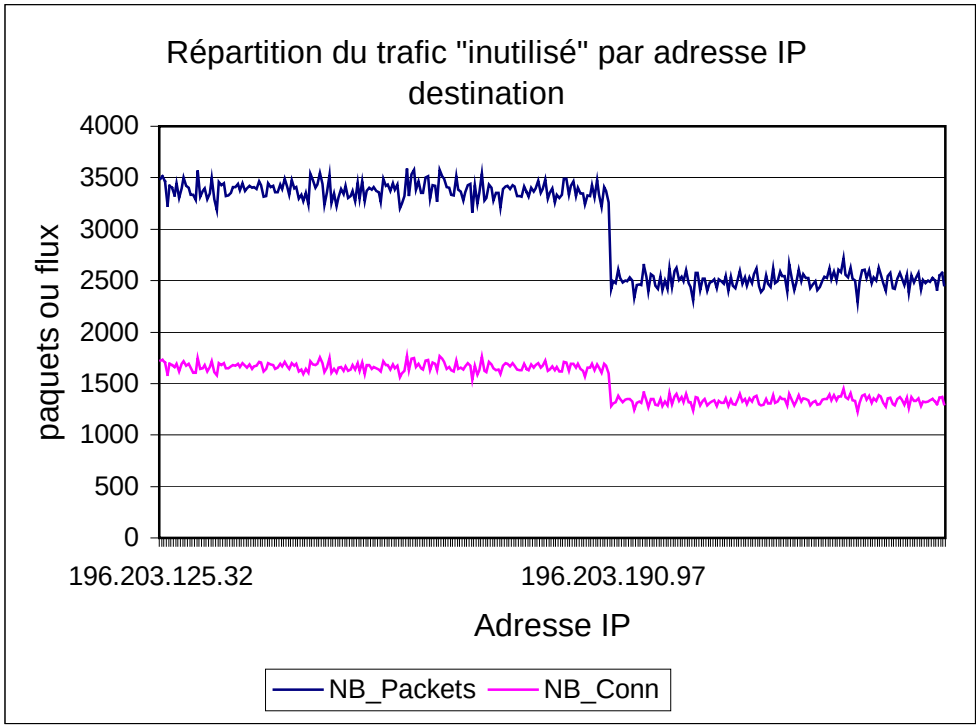

Figure 5.5 : Répartition du trafic « inutilisé » entrant par adresse IP (T4)

3. Estimation du volume du trafic malicieux entrant

L'analyse de la répartition du trafic « inutilisé » par adresse IP destination a montré que, si les adresses IP appartenant au même sous-réseau /24 reçoivent le même volume de trafic indésirable, ceci n'est pas le cas des adresses IP ne partageant pas le même préfixe /24.

Par conséquent, nous proposons d'utiliser cette dernière constatation pour calculer une estimation du volume du trafic malicieux à l'entrée de n'importe quel réseau connecté à Internet. En effet, en calculant périodiquement le volume du trafic entrant, ainsi que celui «inutilisé» pour chaque préfixe /24 alloué au réseau supervisé nous pouvons déduire que le volume du trafic malicieux entrant est donné par la formule ci-dessus.

$$MalicieuxT = \frac{\sum_{i=1}^{Q} \frac{InutiliséT_i * 256}{N_i}}{\sum_{i=1}^{q} TotalT_i} \qquad \textbf{(5.1)}$$

Où:

– *MalicieuxT* est le taux estimé du trafic malicieux entrant.

– *q* est le nombre de préfixe /24 partiellement utilisés du réseau supervisé

– *InutiliséTi* est le volume de trafic entrant envoyé aux adresses IP inutilisées faisant partie du préfixe /24, numéro i, (mesuré en paquets ou en connexions).

– *Ni* est le nombre d'adresses inutilisées appartenant au préfixe /24 numéro i.

– *TotalTi* est le volume de trafic total envoyé vers le préfixe /24 numéro i, (mesuré en paquets ou en connexions).

Ainsi, la formule précédente nous permet de déduire que le trafic malicieux envoyé vers le réseau RNU durant une seule journée (trace T4) représente prés 3,4 % des paquets entrants et 33% des connexions. Ces valeurs illustrent l'importance de ce trafic, et la nécessité de le filtrer. En fait, même si ce trafic ne peut pas saturer la bande passante des liens, il peut gravement affecter les performances des équipements réseaux actifs qui ont besoin de garder un état pour chaque flux actif (notamment les pare-feu à états).

Enfin, nous notons que l'estimateur proposé nécessite la présence d'adresses IP inutilisées dans chaque préfixe /24 du réseau supervisé. Nous estimons que ceci ne présente pas de problème dans la plupart des réseaux, puisque les préfixes /24 sont généralement découpés en sous-réseaux ce qui crée des adresses IP « inutilisées » (correspondant aux adresses réseaux et diffusions des sous-réseaux obtenus). De plus, cet estimateur est limité à l'estimation du volume des activités malicieuses à caractère non ciblé. En effet, si l'attaquant connaît l'adresse IP de sa cible, et envoie directement son trafic à elle, le volume du trafic « inutilisé » ne subira aucune modification et par conséquent l'estimateur ne relèvera pas la présence de cette attaque.

4. Surveillance du trafic de scan

Aujourd'hui, les balayages de ports forment une composante essentielle du trafic malicieux affectant n'importe quel réseau connecté à Internet ; ceci a été mis en évidence dans le réseau RNU (grâce à l'analyse du trafic des adresses IP inutilisées), mais aussi dans d'autres réseaux grâce à de nombreuses études métrologiques, dont [Yegn03, Pang04 et Allm07].

En effet, dans [Allm07], les auteurs ont analysé les activités de scan observées à l'entrée du réseau du laboratoire LBNL de l'université de Berkeley entre 1994 et 2006. Ils ont pu, ainsi mettre en évidence qu'à partir de 2001, ces activités sont devenues omniprésentes principalement en raison de la propagation rapide et à très grande échelle des vers informatiques tels que CodeRed I et II, Nimda, Slammer et Blaster. De plus, ils ont remarqué que les scanneurs sont principalement intéressés par les scans horizontaux (ciblant un port particulier au niveau de plusieurs adresses IP) au détriment des scans verticaux (ciblant plusieurs ports au niveau d'un hôte particulier).

Adoptant chacun une approche différente, les travaux de [Pang04] et de [Yegn03], confirment tous les deux l'omniprésence des balayages de ports et mettent en évidence la grande variabilité du trafic

qu'ils génèrent au cours du temps et entre sites. De plus, [Yegn03] montre la présence d'une multitude de techniques et de stratégies de scan et relate la persistance du trafic de scan provenant de certains vers informatiques (Code Red et Nimda) longtemps (plus d'un an) après la publication des correctifs de sécurité (corrigeant les failles logicielles exploitées par ces vers) par Microsoft et des signatures correspondantes par les principaux éditeurs de logiciels antivirus.

Enfin, dans [Jang04] les auteurs se sont intéressés à certaines formes bénignes de scans telles que celles générées par les moteurs de recherche ou encore par certaines applications (SSH, pair à pair et services Windows) pour rassembler des informations ou localiser les serveurs.

Pour répondre à l'omniprésence du trafic de scan, la diversité et l'ingéniosité des techniques des scanneurs[4], employant des stratégies de plus en plus furtives, nombreuses techniques de détection sont proposées dans la littérature, dont par exemple [Roes99, Paxs99, Jung04, Leck02, Wagn05 et Simo06]. Toutefois, la plupart de ces techniques souffrent d'un taux important de fausses alarmes et peuvent être facilement leurrés (via les balayages effectués dans un ordre aléatoire, avec une vitesse excessivement lente ou encore à partir de plusieurs adresses IP) [Rama09].

Par ailleurs, l'ubiquité même du trafic de scan rend les approches de détection peu efficaces. En effet, le trafic de scan peut être assimilé à un rayonnement de fond inévitable [Pang04] qui frappe en permanence tous les réseaux connectés à Internet. Cela signifie que la présence de scanneurs est une nuisance qui n'est pas problématique en soi, et qu'il n'est pas intéressant de tenter de détecter chaque scanneur individuel. Néanmoins, l'apparition de changements dans la structure du trafic de scan est un signe révélateur important qui expose la propagation d'un nouveau ver ou la survenue d'une attaque ciblée.

4.1 Collecte du trafic de scan

Surveiller le trafic généré par les balayages de ports nécessite de collecter au préalable ce trafic. Pour ce faire, nous avons choisi de n'utiliser aucune méthode détection de scans. En effet, nous proposons de collecter, au niveau du nœud surveillé, toutes les demandes de connexions TCP sans réponses. Plus concrètement, il s'agit de collecter tous les paquets TCP SYN, pour lesquels aucune réponse SYN / ACK correspondante n'a été envoyée au bout de 60 secondes.

Bien que ce trafic puisse contenir des demandes de connexions bénignes (causées notamment par des congestions ou des pannes temporaires) et qu'il ne correspond pas à tous les scans envoyés sur le réseau, il présente deux avantages principaux. D'une part, collecter ce trafic nécessite peu de ressources matérielles puisque tous les paquets n'ayant pas de drapeau SYN sont ignorés et le réassemblage complet des connexions n'est pas effectué. D'autre part, ce trafic contient, effectivement, la majorité des balayages de ports traversant le nœud surveillé. En effet, les

[4] Pour les détails des techniques de scan, se référer à l'annexe B.

techniques de scan furtives, basées sur l'utilisation de paquets invalides (avec drapeau SYN mis à zéro et une combinaison particulière des autres drapeaux), sont très peu répondues à cause de leur efficacité limitée [Nmap]. De plus, les balayages de ports, utilisant les paquets SYN, génèrent un grand nombre des paquets SYN isolés et peu de connexions TCP complètes [Paxs99, Leck02 et Simo06].

Pour toutes ces raisons, nous assimilerons, dans la suite, tout le trafic de paquets SYN sans réponses à des scans. Une fois ce trafic collecté, l'approche de surveillance des scans, que nous proposons, peut être résumée dans les trois étapes ci-dessous;

1. L'agrégation du trafic de scan collecté au niveau d'un nœud surveillé dans des fenêtres disjointes de *w* paquets.
2. Calculer, pour chaque fenêtre, les distributions empiriques du trafic dans l'espace formé par les adresses IP et les numéros de ports sources et destinations.
3. Analyser les distributions empiriques calculées et les comparer à des distributions de référence afin de détecter les changements pouvant les affecter.

4.2 Description de l'approche proposée

La méthode de détection de changements dans le trafic de scan, que nous proposons, a des fondements similaires à ceux de [Leck02 et Wagn05] puisque, nous supposons que les activités de balayages de ports peuvent être caractérisées par la distribution de leur trafic dans l'espace formé par l'adresse IP source (@ ipsrc), l'adresse IP destination (@ ipdst), le numéro de port source (# src) et le numéro de port de destination (# dst). En effet, dans [Leck02], la répartition des adresses IP destinations (@ ipdst) et celle des numéros de ports destinations (# dst) sont utilisées comme éléments discriminants pour différencier les adresses IP sources normales et de celles des scanneurs ; Alors que dans [Wagn05] l'entropie calculée à partir de la distribution des adresses IP sources (@ ipsrc) est utilisée comme l'élément clé à surveiller pour détecter la propagation des vers informatiques.

Toutefois, notre méthode est singulière, dans le sens où nous proposons de surveiller simultanément plusieurs distributions de trafic. Pour sélectionner les distributions adéquates, nous commençons par analyser l'influence des différentes stratégies de balayage de ports (horizontale, verticale et collaborative) sur les distributions du trafic dans l'espace formé par les attributs suivants : @ ipsrc, @ ipdst, # src et # dst.

4.2.1 Sélection des distributions à surveiller

Un balayage de ports horizontal affecte la distribution des numéros de ports destinations (# dst) de façon à ce qu'elle devienne concentrée autour du numéro de port cible. De plus, le fait que la

distribution des adresses IP sources (@ ipsrc) soit également concentrée (autour d'une seule adresse ou d'un nombre réduit d'adresses IP) indique qu'il s'agit probablement d'activité de scan effectué par un pirate (ou quelques pirates) à la recherche de machines vulnérables ; autrement il s'agit très probablement d'un scan engendré par la propagation massive d'un vers informatique.

D'une manière similaire un scan vertical peut être détecté comme un changement dans la distribution des adresses IP sources (@ ipsrc) et celle des adresses IP destinations (@ ipdst) qui deviennent toutes les deux denses : autour de l'adresse IP de l'attaquant pour la distribution de @ ipsrc et autour de l'adresse IP de la victime pour @ ipdst.

Les scans collaboratifs sont les plus difficiles à détecter puisque l'attaquant cible une multitude de numéros de ports sur plusieurs adresses IP destination en utilisant différentes adresses IP sources affectant ainsi toutes les distributions qui deviennent dispersées. Par conséquent, nous proposons d'utiliser la distribution conjointe formée par quadruplet suivant : adresse IP source (@ ipsrc), l'adresse IP destination (@ ipdst), le numéro de port source (# src) et le numéro de port de destination (# dst) afin d'exposer ces scans.

D'où, nous proposons de surveiller les distributions de trafic de scan relatives aux quatre attributs suivants :

- La paire (@ ipsrc, # dst) : afin de détecter l'émergence de scans horizontaux provenant de pirates individuels ou de contrôleur de botnet.
- (# dst): afin de détecter la propagation de nouveaux vers informatiques.
- La paire (@ ipsrc, @ ipdst): afin de détecter les scans verticaux.
- (@ ipsrc, @ ipdst, # src, # dst): afin de détecter les scans collaboratifs.

4.2.2 Inférence des distributions de trafic

Calculer les quatre distributions, données plus haut, pour chaque fenêtre d'observation de w paquets revient à gérer quatre vecteurs ayant une dimension totale supérieure à 2^{96} [5]. Heureusement, les nombres de flux, d'adresses IP ou de numéros de ports distincts véhiculés par le trafic Internet (et en particulier par le trafic de scan) sont nettement plus faibles. Ainsi, dans la trace décrite dans le Tableau 5.5 (correspondant à un trafic de scan collecté durant 24 heures), le nombre de paires d'adresses IP est d'environ 2 millions, alors que celui des numéros de ports TCP est à peine de 2 697. Bien que ces nombres soient beaucoup plus faibles que les 2^{64} et 2^{16} possibles, ils restent importants et il est irréalisable d'estimer avec précision de telles distributions.

Par conséquent, la solution serait de recourir à l'estimation d'un histogramme agrégé pour chaque attribut surveillé. Pour construire un tel histogramme, on pourrait appliquer un masque qui regroupe

[5]Un vecteur de dimension 2^{96} pour la distribution conjointe du quadruplet (@ ipsrc, @ ipdst, # src, # dst), un vecteur de dimension 2^{64} pour la paire (@ ipsrc, @ ipdst), un vecteur de dimension 2^{48} pour (@ ipsrc, # dst) et un vecteur 2^{16} pour (# dst).

en une seule classe toutes les valeurs partageant la même valeur du masque. Bien que cette solution ait l'avantage d'être facile à mettre en œuvre, elle présente deux limites de taille. Le premier réside dans le fait qu'elle manque de flexibilité ; puisque le niveau d'agrégation n'est pas facile à contrôler. Le second réside dans sa nature trop déterministe, ce qui permettrait à un attaquant de deviner le masque appliqué et d'en faire usage pour passer inaperçu.

Pour estimer la distribution d'un attribut, tout en évitant ces deux limites, nous proposons d'appliquer, une fonction de hachage aléatoire à chaque valeur prise par cet attribut et de calculer par la suite l'histogramme des empreintes obtenues. Cette approche est avantageuse, car elle conduit à une agrégation flexible (dans le sens où on peut fréquemment changer la fonction de hachage aléatoire utilisée) et robuste dans le sens où l'attaquant ne peut pas faire usage de cette agrégation pour passer inaperçu.

A ce stade, le problème de surveillance de trafic de scan revient à détecter les changements dans les distributions des empreintes de trafic obtenues en appliquant une fonction de hachage aléatoire aux attributs originaux.

4.2.3 Détection des changements

Détecter les changements affectant la distribution d'une v. a X, nécessite de définir une mesure de distance entre une distribution empirique P quelconque de la v. a X et une distribution de référence Q de cette même variable aléatoire. Ce problème est, en effet, central en statistiques où deux principales classes d'approches existent pour y répondre. La première catégorie regroupe les approches dites paramétriques qui consistent à mesurer la distance entre deux distributions par le biais de la mesure du changement dans les paramètres d'une distribution paramétrique modélisant la v. a X.

La deuxième regroupe les approches dites non-paramétriques c'est à dire celles qui n'utilisent aucun type particulier de distribution. En particulier, c'est le cas des approches basées sur l'entropie de Shannon ([Lack05, Wagn05 et Nych08]). Ces dernières consistent à calculer la valeur de l'entropie de Shannon d'une distribution empirique quelconque P et de la comparer à celle calculée pour la distribution de référence Q. Si la différence entre ces deux valeurs d'entropie dépasse un seuil de détection prédéfini, la distribution P serait considérée comme anormale ; autrement elle est marquée comme normale.

Rappelons que, l'entropie de Shannon [Cove91] d'une distribution P d'une v. a X définie sur un ensemble fini H est donnée par l'équation (5.2).

$$H(P) = -\sum_{x \in \mathrm{H}} p(x) \log p(x) \qquad \textbf{(5.2)}$$

L'utilisation de l'entropie de Shannon pour la détection d'anomalies est justifiée par une intuition qui stipule que tout changement dans la répartition de la v. a X, conduit à un changement proportionnel dans la valeur de l'entropie. Malheureusement, une analyse de la fonction dérivée de l'entropie montre que cette dernière n'est pas un bon indicateur des variations affectant la distribution d'une variable aléatoire.

Pour illustrer ce fait, nous allons raisonner pour le cas d'une v. a X pouvant avoir deux états possibles et ayant une distribution (p, 1-p).

Soit (p+Δ, 1-p-Δ) une deuxième distribution de cette même variable aléatoire, la fonction dérivée de H (p) est donnée par l'équation (5.3):

$$\frac{\delta H}{\delta P} = \log(\frac{p}{p-1}) \quad \textbf{(5.3)}$$

Par conséquent, une petite variation Δ dans la distribution de la v. a X résulte dans une variation approximative de $\Delta \log\left(\frac{p}{1-p}\right)$ dans la valeur de l'entropie ; ce qui signifie que la variation de l'entropie dépend fortement de p. Ainsi, pour une petite valeur de p, toute variation Δ même faible s'accompagne par une variation importante de l'entropie. Inversement, pour p proche de 0,5, une variation importante Δ de la distribution résulte dans une faible variation dans la valeur de l'entropie. Cela constitue, à notre avis, une explication théorique des faibles taux de détection obtenus par les méthodes de détection d'anomalies basées sur l'entropie de Shannon.

À la place de l'entropie, nous proposons d'utiliser l'entropie relative, également nommé divergence de Kullback-Leibler (DKL), afin de suivre l'évolution des distributions des empreintes du trafic de scan. En effet, la divergence de Kullback-Leibler est une mesure de la différence entre deux distributions de probabilités définies sur le même ensemble fini H: une distribution P et une distribution de référence Q ; Elle est donnée par l'équation (5.4).

$$D\ (P \| Q) = \sum_{x \in \mathrm{H}} P\ (x) \log \frac{P\ (x)}{Q\ (x)} \quad \textbf{(5.4)}$$

Si nous faisons une étude de sensibilité semblable à ce qui a été fait pour l'entropie de Shannon, pour la divergence de Kulback-Leibler, nous trouvons que la fonction dérivée de DKL par rapport à P est donnée par l'équation (5.5)

$$\frac{\delta D}{\delta P} = \log\left(\frac{p}{q}\right) + \log\left(\frac{1-p}{1-q}\right) \quad \textbf{(5.5)}$$

Ainsi, nous pouvons remarquer que la sensibilité de la DKL ne dépend pas de la valeur de p seule mais du ratio entre p et la probabilité q relative à la distribution de référence Q ; ce qui signifie que la DKL est très sensible aux variations de P par rapport à Q ; dans le sens où toute variation de P par

rapport à Q s'accompagne par une variation proportionnelle dans la DKL ; de ce fait la DKL constitue une métrique de mesure adéquate de la dissimilitude entre deux distributions.

Par conséquent, nous pouvons formuler le problème de détection de changements dans la distribution d'une v. a X comme suit :

Soit Q la distribution de la v. a X calculée à partir d'un trafic de scan de référence ;

Soit w une fenêtre d'observation quelconque composée de n paquets de scan

Et soit Pn la distribution empirique de la v. a X calculée sur la fenêtre w ;

Alors le problème de détection de changement dans la distribution de v. a X revient de choisir l'hypothèse valide parmi les suivantes :

- H1 : La distribution Pn suit la distribution de référence Q ; c'est à dire qu'il n' y a pas de changement dans les activités de scan.
- H2 : La distribution Pn suit une autre distribution Q' ; c'est à dire qu'il y a un changement dans le trafic de scan.

D'après les théorèmes de la théorie de l'information [Cove91], nous pouvons démontrer que le test d'hypothèses précédent peut être remplacé par un simple test sur la divergence de DKL donné par l'équation (5.6) ; où T est le seuil de détection.

$$\left| D(Pn\|Q) - D(Pn\|Q') \right| > \frac{1}{n} \log T \qquad \textbf{(5.6)}$$

De plus, le lemme de Stein stipule que pour un taux des fausses alarmes fixé, le taux minimal de faux négatifs (la probabilité qu'un changement ne soit pas détecté) qu'on peut atteindre dépend de la DKL suivante $D(Q\|Q')$.

Or, dans la pratique, la distribution Q' est inconnue ; ce qui nous conduit à remplacer l'équation (5.6) par le test suivant

$$\left| D(Pn\|Q) \right| > \frac{1}{n} \log T \qquad \textbf{(5.7)}$$

Néanmoins, le test de l'équation (5.7) n'est pas optimal et ne permet pas de dériver le taux de faux négatifs analytiquement. Par conséquent, nous proposons d'opérer d'une manière empirique en utilisant les courbes de Receiver Operation Characteristic (ROC) afin de trouver le seuil de détection optimal. En effet, les courbes ROC décrivent l'ensemble des compromis possibles entre les Faux Positifs et les Faux Négatifs d'une méthode.

4.3 Validation expérimentale

Pour la validation de l'approche proposée, nous avons utilisé deux types de trace de scan : des traces réelles collectées sur le réseau RNU et des traces modifiées ; ces dernières ont été obtenues par injection de scans expérimentaux dans une trace réelle composée de scans.

4.3.1 Traces utilisées

4.3.1.1 Trace réelle

À partir de la trace de trafic T3 (collectée sur le lien reliant le nœud CCK el Kasbah à l'ATI entre le 4 et 5 avril 2006), nous avons extrait tous les paquets SYN sans réponses. Nous avons obtenu, ainsi, la trace (T3-scan) décrite dans le Tableau 5.5. Cette dernière est composée par près de 10 millions de paquets[6] envoyés à partir de 18 388 adresses IP sources vers 56 585 cibles distinctes. La totalité de ce trafic est assimilée à des balayages de ports.

Période	**# Paquets [106]**	**# Ports**	**# @ Sources**	**# @ Destinations**	**# Paires @IP [106]**
4-5 avril 2006	10	2 697	18 388	56 585	2

Tableau 5.5 : Description de la trace de trafic de scan (T3-scan)

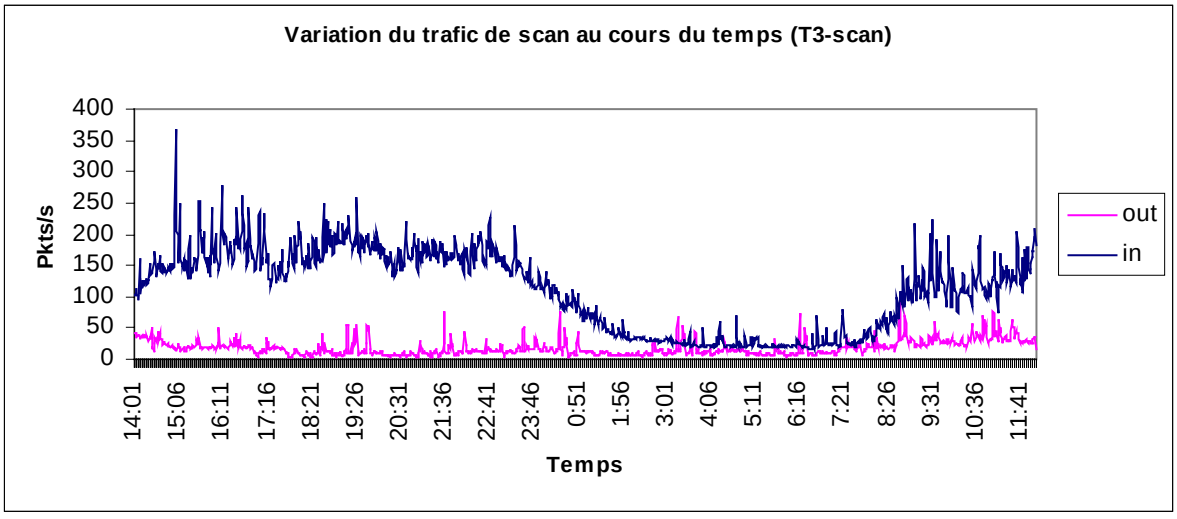

Figure 5.6 : Variation du trafic de scan au cours du temps (T3-scan)

[6] Ce qui représente 3 % des paquets de la trace T3 mais 42 % des connexions TCPde cette trace.

La Figure 5.6 illustre la variation au cours du temps du trafic de scan par sens. Nous remarquons ainsi que le volume du trafic de scan entrant est largement supérieur à celui sortant (il représente 14,5% des paquets de scan) et qu'il présente un cycle journalier, semblable à celui observé pour le trafic total, avec une diminution importante à partir de minuit.

La grande variabilité du volume de scan montre que nous ne pouvons pas simplement fixer un seuil au-dessus duquel le volume de ce trafic sera considéré normal et au-delà suspect, donc nécessitant plus d'investigations. Par conséquent, la surveillance du trafic de scan doit se baser sur d'autres paramètres que le volume.

Par ailleurs, l'analyse des sources et des destinations du trafic de scan montre que la quasi-totalité des paquets de balayage de ports entrants proviennent du réseau Internet tunisien (93,78 % des paquets de scan entrants). De même, la plupart des scans sortants (86,6 % des paquets de scan sortants) sont envoyés vers le réseau Internet tunisien.

La Figure 5.7, représentant le nombre de paquets de scan entrants par adresse cible, montre que les adresses IP partageant le même préfixe /24 reçoivent pratiquement le même nombre de balayage de ports malgré la présence de quelques creux et de quelques pics. Les creux correspondent aux adresses de réseaux et de diffusions des préfixes /24 du RNU ; elles montrent que ces adresses particulières sont généralement évitées par les scanneurs. Alors que les pics correspondent à des cibles privilégiées des scanneurs ; puisqu'elles correspondent à des adresses ayant reçu un grand nombre de balayage de ports. Il s'agit, très probablement, d'attaques ciblées telles que SYNfload ou encore des balayage de ports verticaux massifs. De plus, la Figure 5.7 expose le fait que les adresses IP, ne partageant pas le même préfixe /24, ne reçoivent pas le nombre de balayage de ports.

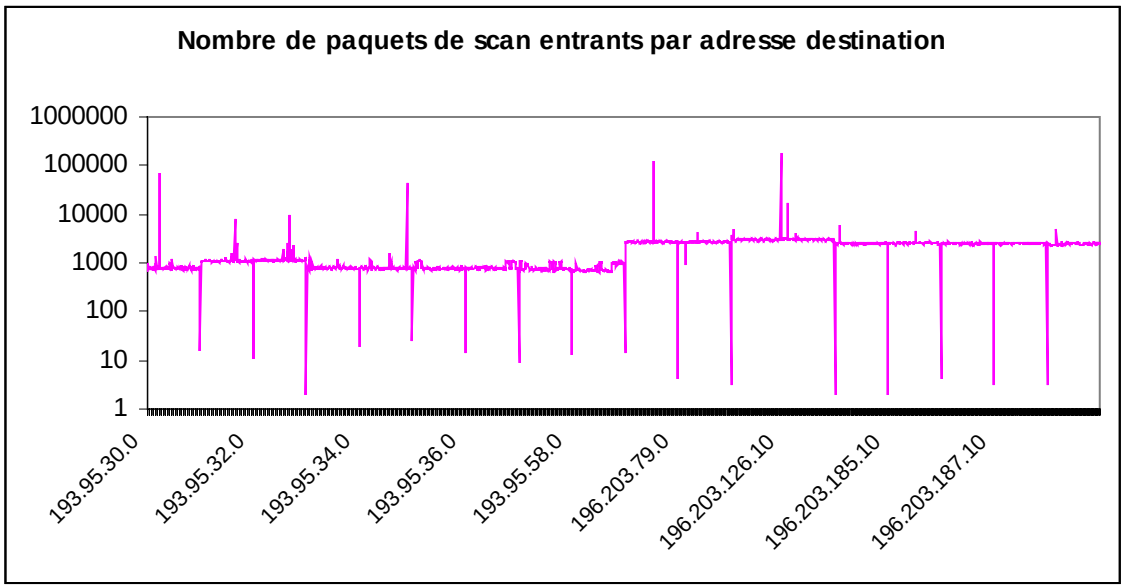

Figure 5.7 : Répartition des scans entrants par IP destination (T3-scan)

En conclusion, la Figure 5.7 constitue une autre validation empirique de l'estimateur du volume du trafic malicieux à caractère non ciblé donné par l'équation (5.1) .

Enfin, une analyse similaire faite pour les scans sortants, montre que ces scans ciblent principalement les adresses IP du réseau Internet tunisien ; mais que la répartition de ce trafic par adresse cible est pratiquement uniforme au sein d'un même préfixe /24.

4.3.1.2 Traces modifiées

Nous avons utilisé l'outil Nmap [Nmap], pour lancer cinq campagnes de balayage de ports différentes, tout en collectant le trafic de scan résultant par l'outil Tcpdump [Tcpdump]. Les traces obtenues sont décrites dans le Tableau 5.6.

Ensuite, nous avons injecté les traces expérimentales de scan dans une trace réelle de durée 1 heure (T-10h), afin d'obtenir les traces modifiées décrites dans le Tableau 5.7. En effet, les traces modifiées sont obtenues en mixant les paquets de scan provenant des traces expérimentales, selon un certain pourcentage, avec ceux du trafic de scan original.

Trace	**#Paquets**	**Type de scan**	**Stratégie de scan**
V1	15 000	Balayage de ports vertical ciblant une seule adresse IP.	Ports destination choisis d'une manière incrémentale.
V2	15 000	Balayage de ports vertical ciblant une seule adresse IP.	Ports destination choisis d'une manière aléatoire.
V3	2 500	Balayage de ports vertical ciblant une seule adresse IP.	Ports destination choisis parmi une liste de numéros de ports « reconnus ».
H1	15 000	Balayage de ports horizontal ciblant le port TCP 80.	Adresses destinations choisies d'une manière incrémentale à partir d'un préfixe donné.
H2	15 000	Balayage de ports horizontal ciblant le port TCP 80.	Adresses destinations choisies d'une manière aléatoire parmi l'espace d'adressage IPv4.

Tableau 5.6 : Traces de scans réalisés par Nmap

Trace	**Description**	**Intensité du scan ajouté**
T-10h	Trace de scan originale extraite de T3-scan (période de 10h à 11h)	0 %
T-V	Trois scans verticaux V1, V2 et V3 injectés à T-10h respectivement à partir des paquets 100 000, 215 000 et 43 000.	100 %
T-V50pc	Trois scans verticaux V1, V2 et V3 injectés à T-10h à partir des paquets 100 000, 230 000 et 460 000.	50 %
T-V20pc	Trois scans verticaux V1, V2 et V3 injectés à T-10h respectivement à partir des paquets 100 000, 260 000 et 520 000.	20 %
T-H	Deux scans horizontaux H1 et H2 injectés à T-10h respectivement à partir des paquets : 100 000 et 215 000.	100 %
T-H50pc	Deux scans horizontaux H1 et H2 injectés à T-10h respectivement à partir des paquets : 100 000 et 230 000.	50 %
T-H20pc	Deux scans horizontaux H1 et H2 injectés à T-10h respectivement à partir des paquets 100 000 et 260 000.	20 %

Tableau 5.7 : Traces de scans modifiées

4.3.2 Validation à partir des traces de scan modifiées

Afin de valider notre méthode de détection des changements dans le trafic de scan, nous l'avons, d'abord, utilisé pour détecter les scans artificiellement ajoutés dans les traces décrites dans le Tableau 5.7.

Pour ce faire, nous avons utilisé une taille de fenêtre de 10 000 paquets, ce qui correspond à une durée approximative de 60s. Pour le calcul des distributions de trafic, nous avons utilisé une fonction de hachage aléatoire à 5 bits pour les attributs suivants : (@ ipsrc, # dst), (# dst), et (@ ipsrc, @ ipdst) ; c'est à dire que les distributions de ces attributs sont estimées par des histogrammes à 32 classes chacun.

Pour l'estimation de la distribution conjointe des quatre attributs @ ipsrc, @ ipdst, # src, # dst, nous calculons les fonctions de hachage de chacun des attributs sur 3 bits, puis nous concaténons les valeurs obtenues en une seule valeur à 12 bits ; ainsi la distribution conjointe est estimée par un histogramme à 4096 classes.

Enfin, pour l'estimation des quatre distributions de référence, nécessaires à la détection de changements, nous utilisons les 50 000 premiers paquets de la trace du trafic de scan réel T-10h.

La Figure 5.8 illustre la variation des divergences de Kullback-Leibler (relatives aux quatre attributs retenus). Elle utilise la trace T-V, comportant 3 scans verticaux ajoutés à la trace réelle de scan T-10h. Nous remarquons que les trois scans ajoutés (respectivement à partir des fenêtres numéros 10, 21 et 43) apparaissent, tous, comme des sauts dans les DKLs calculées à partir des attributs (@ ipsrc, @ ipdst), (# dst), et (@ ipsrc, # src, @ ipdst, # dst). Ceci valide l'utilisation de notre approche pour la détection des changements induits par les scans verticaux.

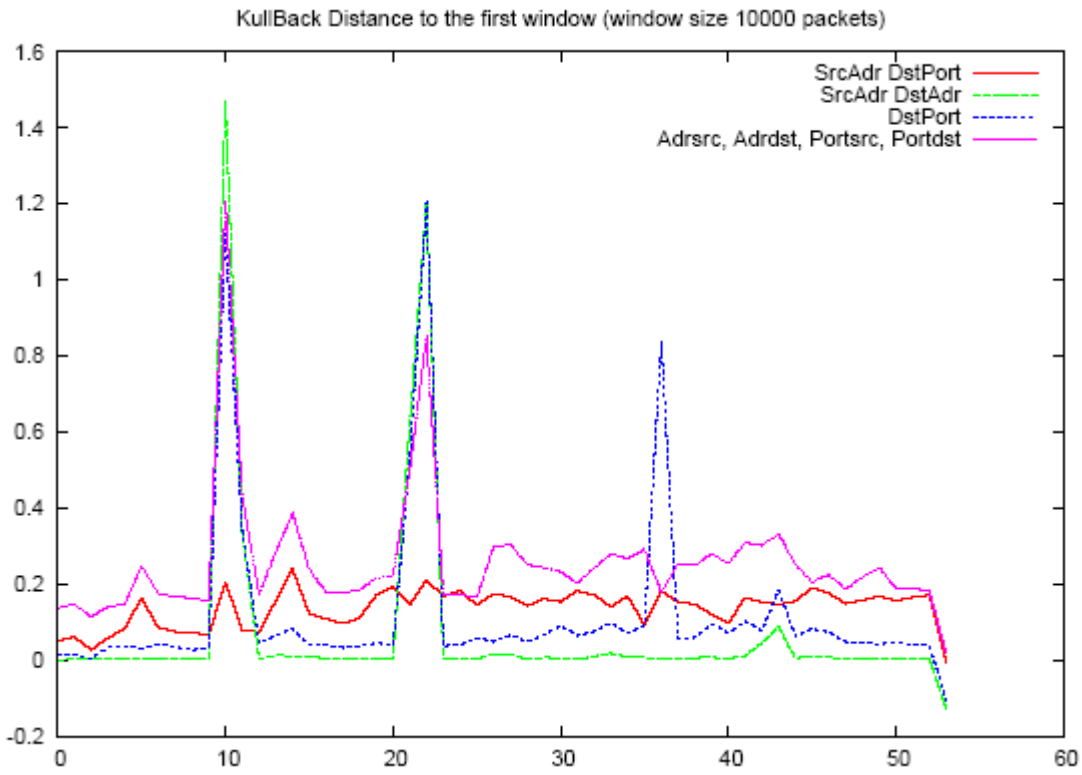

Figure 5.8 : Distances de Kullback-Leibler calculées pour 4 attributs (T-V)

Nous notons que le saut apparaissant au niveau de la fenêtre numéro 33, correspond à un scan horizontal original composé de plus de 2000 paquets et destiné au port TCP 32 000 ; alors que le fait que le saut associé au troisième scan (fenêtre 43) est moins visible dans les courbes de DKL, s'explique par sa faible intensité (il est composé de seulement 2500 paquets).

La Figure 5.9, illustre la variation des divergences de Kullback-Leibler calculées pour quatre attributs à partir de la trace T-V20pc (comportant 3 scans verticaux d'intensité 20 %). Nous remarquons que les trois scans ajoutés apparaissent tous comme des sauts dans la courbe de DKLs calculée pour l'attribut (@ ipsrc, @ ipdst). De plus, le deuxième scan vertical, utilisant une stratégie aléatoire, n'apparaît pas dans la DKL de la distribution conjointe contrairement aux deux autres.

Nous remarquons, aussi, que les sauts associés aux scans injectés, forment des paliers, contrairement à la Figure 5.8. Ceci est dû au fait que les scans ajoutés sont mixés avec un grand nombre de paquets et s'étalent sur une période de temps plus longue. De plus, l'amplitude des sauts observés a diminué reflétant, ainsi, la diminution dans l'intensité des scans ajoutés. Par conséquent, nous pouvons déduire que notre approche permet la détection des balayages verticaux intensifs que ceux furtifs s'étalant sur une longue période de temps en inspectant la courbe de la DKL calculée pour (@ ipsrc, @ ipdst).

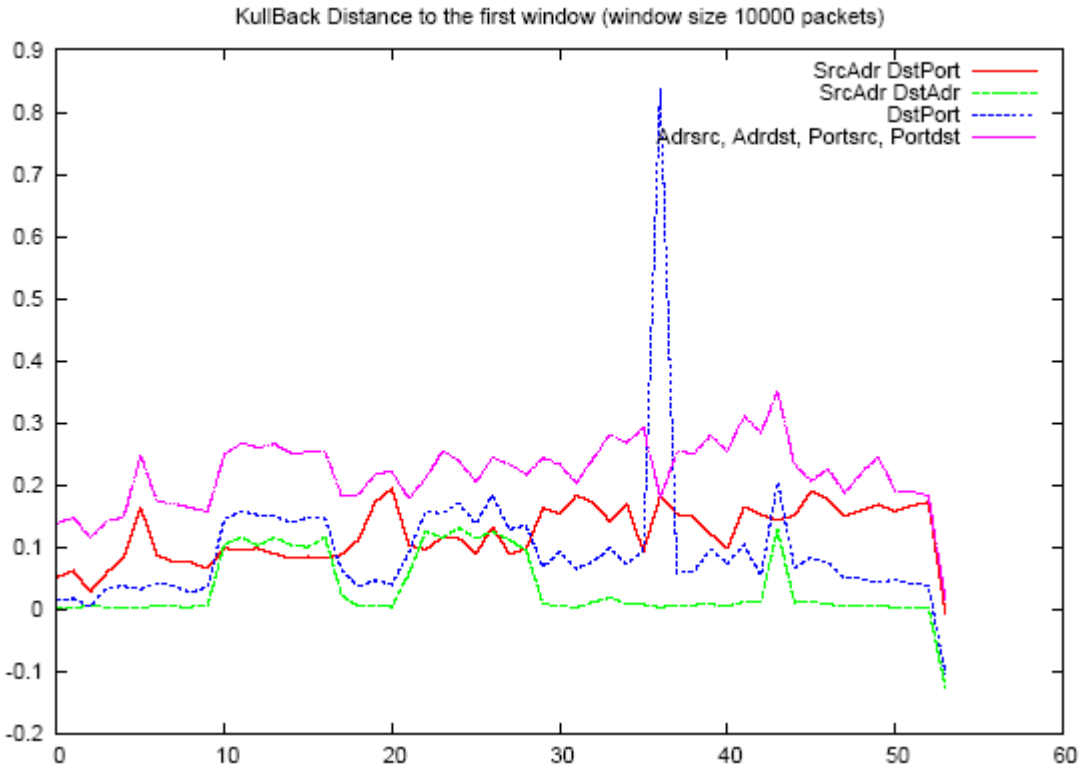

Figure 5.9 : Distances de Kullback-Leibler calculées pour 4 attributs (T-V20pc)

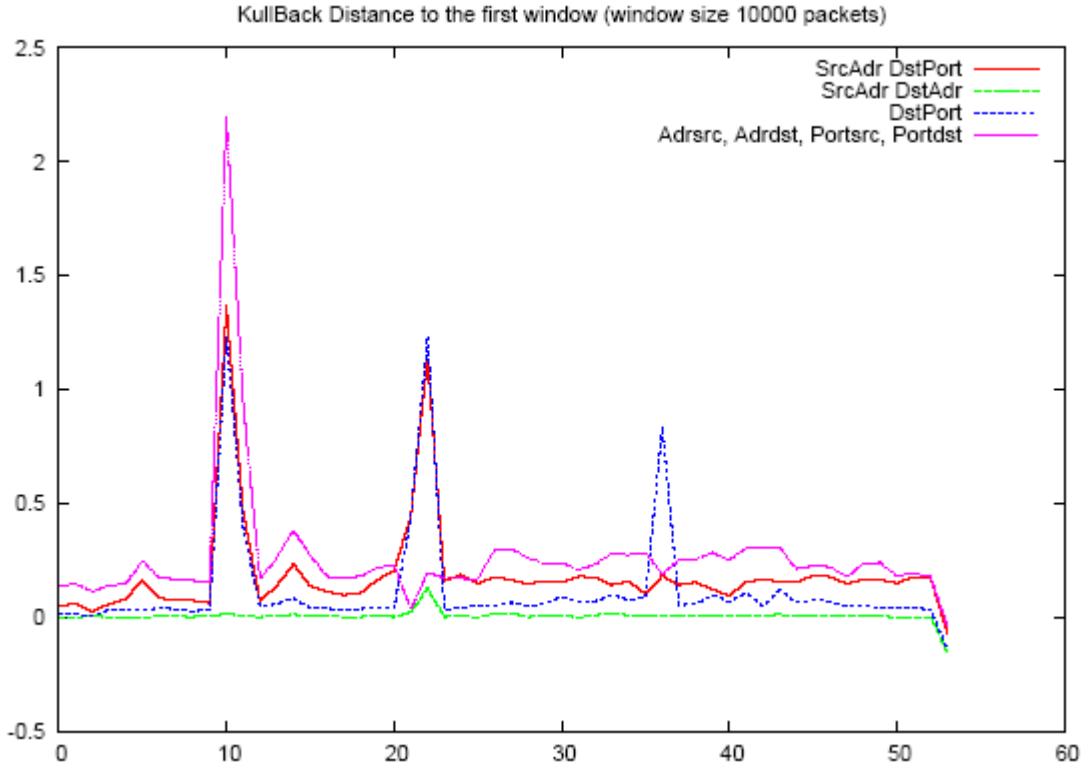

Figure 5.10 : Distances de Kullback-Leibler calculées pour 4 attributs (T-H)

La Figure 5.10 illustre la variation des divergences de Kullback-Leibler calculées pour quatre attributs à partir de la trace modifiée T-H, comportant 2 scans horizontaux ciblant le port TCP 80. Il apparaît, ainsi, que ces deux scans apparaissent, comme attendu, clairement dans la DKL calculée à partir de (@ IPsrc, # dst). De plus, au niveau de la distribution conjointe de (@ IPsrc, # src, @ IPdst, # dst), le premier scan horizontal (ciblant des adresses IP choisies séquentiellement) apparaît comme un saut, alors que le second (ciblant des adresses IP choisies d'une manière aléatoire) apparaît comme un creux.

De la même façon, la Figure 5.11 illustre la variation des divergences de Kullback-Leibler calculées à partir de la trace modifiée T-H20pc (comportant 2 scans horizontaux d'intensité égale à 20%). Il apparaît ainsi que, contrairement à la DKL calculée pour (@ ipsrc, # dst), les deux scans ajoutés apparaissent au niveau la distribution conjointe de (@ ipsrc, # src, @ ipdst, # dst). En effet, le premier scan horizontal (ciblant des adresses IP choisies séquentiellement) apparaît comme un saut ; alors que le second (ciblant des adresses IP choisies d'une manière aléatoire) apparaît comme un creux.

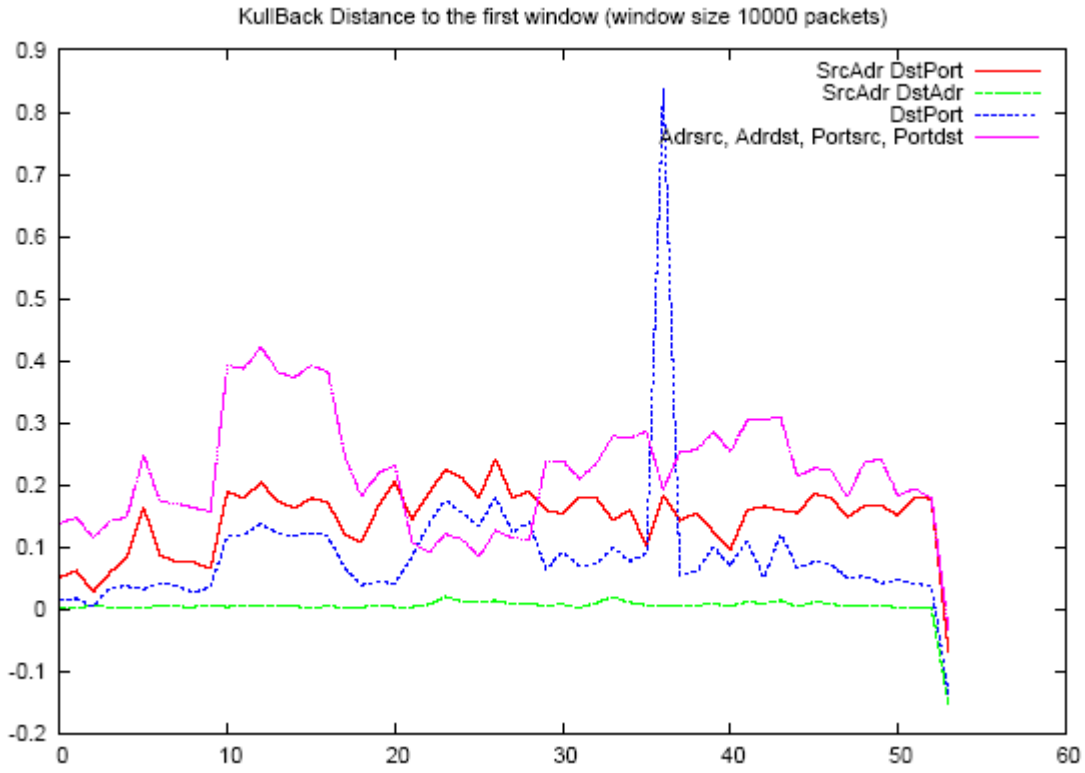

Figure 5.11 : Distances de Kullback-Leibler calculées pour 4 attributs (T-H20pc)

Par conséquent, nous pouvons conclure que la distribution conjointe permet d'exposer la présence de tous les types de scan indépendamment de la stratégie qu'ils utilisent ou de leur intensité. Toutefois, l'intérêt des trois autres distributions réside dans leur capacité de relever le type du scan.

4.3.3 Validation à partir des traces réelles

La Figure 5.12 montre l'évolution des DKLs, calculées pour quatre attributs, à partir de la trace de scan (T3-scan) ; elle présente globalement une structure plate et proche de zéro. Ceci valide notre hypothèse stipulant que les distributions de trafic de scan sont stables en dehors de la présence d'une menace de sécurité majeure (telle que la propagation d'un nouveau ver). Toutefois la DKL de (@ ipsrc, # dst) présente plusieurs pics et une structure moins plate que la DKL de (@ ipsrc, @ ipdst) ; ce qui reflète la prépondérance des balayage de ports horizontaux par rapport à ceux verticaux. Nous rappelons que la prépondérance des balayages de ports horizontaux a été déjà mise en évidence par l'analyse de cette trace (dans le paragraphe 4.3.1.1 de ce même chapitre). De plus, elle concerne tous les réseaux connectés à Internet [Allm07 et Pang04].

Par ailleurs, nous remarquons des variations importantes de la DKL de (@ ipsrc, # dst) entre les fenêtres 700 et 800 qui reflètent un changement dans le profil des activités des balayages de ports horizontaux. Ces fenêtres correspondent à la période comprise entre 3h30 et 8h du matin ; or, cette période correspond à une nette augmentation du volume des balayages de ports sortants (visible dans la Figure 5.6). L'analyse des paquets de scan relatifs à cette période confirme la présence de plusieurs scans sortants ciblant principalement les ports TCP 80 et 443 ; de plus la présence de communications via le port 445 entre les sources de ces scans et une adresse IP externe laisse deviner que ces sources forment un réseau de zombies.

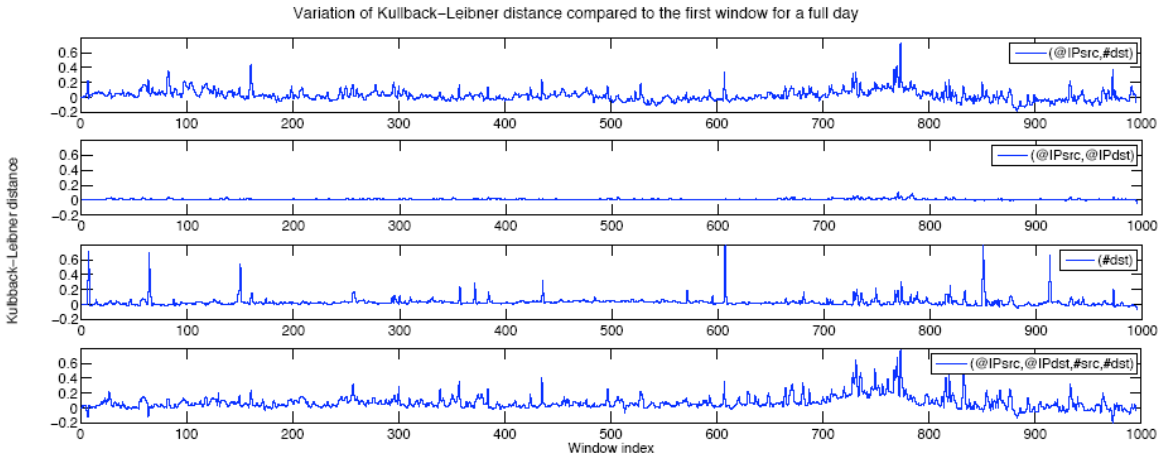

Figure 5.12 : Distances de Kullback-Leibler calculées pour 4 paramètres (T3-scan).

Enfin, le fait que cet événement soit également reflété par la courbe représentant la DKL de la distribution conjointe confirme que cette dernière permet la détection de tous les changements affectant les activités de balayages de ports.

5. Conclusion

Partant de l'hypothèse que les trafics « inutilisé » et « invalide » sont étroitement liés à ceux du trafic malicieux total, nous avons analysé deux traces de trafic « inutilisé » et « invalide » collectées sur RNU, pour en déduire la composition et le volume du trafic malicieux total affectant ce réseau.

Ainsi, nous avons mis en évidence que le trafic malicieux affectant le réseau RNU est composé par des attaques non ciblées et des attaques ciblées. Les premières, mises en évidence par l'analyse du trafic « inutilisé » ; sont engendrées principalement par des balayages de ports, l'envoi de messages non sollicités par « messenger » et le trafic « backscatter ». Alors que, la présence d'attaques ciblées est exposée par le trafic « invalide ».

Par la suite, nous avons proposé un estimateur permettant de calculer le taux du trafic malicieux entrant à caractère non ciblé. Cet estimateur peut être utilisé comme un indicateur global du niveau des menaces à caractère non ciblé affectant n'importe quel réseau connecté à Internet.
Dans la seconde partie de ce chapitre, nous avons proposé une nouvelle approche de surveillance du trafic de scan basée sur la divergence de Kullback-Leibler. En effet, la méthode proposée se base sur la collecte de tous les paquets SYN sans réponses au niveau d'un lien ou d'un nœud de concentration du trafic. De plus, elle nécessite de calculer, pour chaque fenêtre de *w* paquets, 4 distributions d'empreintes de trafic obtenues par application de fonction de hachage aléatoire sur les attributs suivants : (@ ipsrc, # dst), # dst, (@ ipsrc, @ ipdst) et (@ ipsrc, # src, @ ipdst, # dst).
Enfin, nous avons évalué l'approche proposée en utilisant une trace de trafic de scan collectée sur RNU, ainsi que des traces de scan artificiellement modifiées. Nous avons pu, ainsi, montrer que la divergence de Kullback-Leibler de la distribution conjointe constitue un compromis permettant d'exposer la présence de tous les scans aussi bien horizontaux que verticaux. De plus, nous avons établis que cette divergence est proche de zéro, en absence de propagation de nouveaux vers informatiques ou de la survenue d'attaque ciblée importante ; et ce malgré la variabilité, au cours du temps, du volume du trafic de scan.

Conclusion générale

Nous nous sommes intéressés, tout au long de cette thèse, à la mesure et la caractérisation du trafic Internet. Nous avons, également, proposé des techniques de détection d'anomalies et de surveillance du trafic malicieux.

Dans le premier chapitre, nous avons proposé un état de l'art des techniques de mesures et discuté les caractéristiques du trafic Internet telles qu'appréhendés grâce à la métrologie. Cette étude nous a permis de mettre en évidence la variabilité des caractéristiques du trafic Internet selon le temps et l'endroit où elles sont observées, mais elle nous a aussi permis de dégager la présence d'un certain nombre de caractéristiques invariantes, dont ci dessous un résumé.

- Les trafics mesurés sur les deux sens d'un même lien présentent une dissymétrie forte et ce quel que soit le type du lien ou son débit.
- Le trafic Internet est composé par des flux éléphants et des flux souris. Les premiers sont peu nombreux mais génèrent la majorité du trafic. Les seconds sont majoritaires mais ne transportent qu'un faible pourcentage du trafic total.
- Malgré la diversité des applications utilisées, le protocole TCP reste le protocole de transport privilégié.
- Les processus d'arrivée des octets, paquets ou flux sur les liens de l'Internet sont non stationnaires ; Ils présentent des cycles journalier et hebdomadaire bien marqués.
- Le processus d'arrivée des paquets sur les liens de l'Internet est dépendant à long terme.

Dans la seconde partie du premier chapitre, nous avons brièvement présenté les modèles statistiques du trafic Internet et montré leurs insuffisances. En effet, la plupart des modèles proposés dans la littérature sont difficilement exploitables et ne décrivent pas tous les aspects du trafic ; de ce fait ils ne peuvent être utilisés comme modèles pour des systèmes de détection d'anomalies efficaces.

Dans le second chapitre, nous avons exposé l'état de l'art des techniques de détection d'anomalies adaptées à une utilisation dans les réseaux de backbone. Nous avons, ainsi, montré que les approches de détection d'anomalies non-paramétriques sont les plus intéressantes. En effet, elles n'ont pas besoin d'informations à priori sur les techniques d'attaques, ni d'ailleurs de modèles statistiques pour décrire le trafic. De plus, nous avons montré que les métriques de volume, ainsi que les métriques de distributions permettent d'exposer un grand nombre d'anomalies légitimes et illégitimes.

Le troisième chapitre constitue à la fois une application au cas du réseau RNU des techniques présentées dans le premier et une approbation des conclusions de ce même chapitre. En effet, nous avons présenté dans ce chapitre une étude métrologique détaillée du trafic RNU (effectuée entre 2004 et 2006) selon deux niveaux de granularité : paquets et flux ; confirmant ainsi la validité des caractéristiques invariantes dégagées dans le premier chapitre. En particulier, nous avons constaté la présence de la dépendance à long terme (LRD) dans les processus d'arrivée des paquets et l'absence de dépendance dans les arrivées de flux. Nous avons expliqué ce phénomène par la généralisation de l'utilisation du protocole HTTP 1.1 qui grâce à son mécanisme de connexions persistantes, aboutit à la génération de gros flux indépendants contribuant, ainsi, à augmenter le niveau de dépendance entre les paquets et à diminuer celle entre les flux.

Dans le quatrième chapitre, nous avons proposé une approche non-paramétrique de détection d'anomalies de trafic basée sur la collecte périodique de métriques de volume simples. L'évaluation de l'approche proposée a requis l'utilisation d'une trace préalablement étiquetée et a prouvé son efficacité face à deux types d'attaques par déni de service, mais aussi son incapacité à détecter les balayages de ports de faible intensité ou à distinguer les anomalies légitimes de celles malicieuses.

Par la suite, nous avons exploité l'approche proposée pour la détection des anomalies dans le réseau RNU. Ainsi, nous avons pu remarquer que la présence des anomalies est monnaie courante dans RNU mais que la plupart des anomalies observées sont de courte durée.

Le chapitre cinq présente d'une part un estimateur simple du volume du trafic malicieux à l'entrée de n'importe quel réseau, d'autre part une approche originale pour la surveillance du trafic de scan.

En effet, à partir d'un spécimen du trafic malicieux dans RNU (les trafics liés aux adresses IP invalides et inutilisées) nous avons déduis le volume et la composition du trafic malicieux total. Ainsi, nous avons pu établir que le trafic entrant, lié aux adresses IP inutilisées, reflète principalement des balayages de ports et des envois de messages non sollicités. Alors que, le trafic « invalide » expose principalement des communications entre contrôleurs de botnets et machines zombies. En outre, nous avons proposé un estimateur simple du volume du trafic malicieux basé le rapport entre le volume de trafic entrant et celui non utilisé pour chaque préfixe /24.

L'omniprésence des balayages de ports automatiques dans le réseau RNU, et dans l'Internet en général ; ainsi que, l'incapacité de l'approche de détection d'anomalies de volume, proposée dans le chapitre quatre, à les détecter correctement, nous ont incité à proposer une nouvelle approche pour la surveillance du trafic de scan. Cette dernière utilise quatre distributions d'empreintes de trafic calculées dans l'espace formé par les attributs suivants : adresse IP source, adresse IP destination, port source et port destination. La validation de l'approche de surveillance du trafic de scan, sur diverses traces (réelles et artificiellement modifiées), a montré que la divergence de Kullback-Leibler de la distribution conjointe constitue un compromis permettant d'exposer la présence de tous

les scans aussi bien horizontaux que verticaux, alors que les trois autres distributions utilisées, permettent d'identifier leurs types.

Comme première perspective de ces travaux, nous proposons de valider les approches proposées dans cette thèse, face à de nouvelles traces provenant d'autres réseaux ou à défaut face à des traces de plus longues durées collectées durant des périodes de forte utilisation. De plus, il est nécessaire que ces traces soient composées de trafic normal et d'anomalies précisément identifiées par d'autres moyens (notamment des experts humains en s'aidant d'IDS commerciaux et libres). Ainsi, cela nous permettra de correctement évaluer la sensibilité des approches proposées face à divers types d'activités malicieuses, notamment la propagation de nouveaux vers informatiques, les attaques de déni de service distribuées ou les activités des réseaux zombies.

De plus, la disponibilité de telles traces nous permettrait de faire varier les paramètres des approches de détection d'anomalies proposées et d'étudier l'effet de chacun sur les performances de détection globale et par type d'anomalie. En particulier, nous pourrons étudier l'impact du nombre de composantes principales majeures retenu et celui des composantes mineures sur les performances de détection de l'approche proposée dans le quatrième chapitre.

Nous proposons également d'adapter l'approche de surveillance des scans pour la détection des anomalies de trafic. En effet, la caractérisation du trafic des adresses IP inutilisées et celui des adresses IP invalides a montré que le trafic des activités malicieuses a des répartitions (par rapport aux adresses IP et aux numéros de ports sources et destinations) radicalement différentes de celles du trafic normal. Par conséquent, il est légitime de s'attendre à ce que ces activités puissent être détectées comme des changements dans la divergence de Kullback-Leibler de la distribution conjointe par rapport à un trafic normal de référence. Néanmoins, à cause de la variabilité des usages de l'Internet selon les moments de la journée, la DKL risque d'avoir une courbe non stable au cours du temps même en absence d'anomalies. Pour y remédier, une solution possible serait de modéliser la variabilité normale de la DKL par un modèle de prévision tel que celui de Holt-Winters et de marquer comme anomalie toute déviation par rapport à ce modèle.

Par ailleurs, afin de faciliter la tâche de l'administrateur du réseau et lui permettre de trouver les meilleures réponses capables de minimiser l'impact des anomalies détectées ; tout système de détection d'anomalies nécessite d'être couplé à un outil d'identification d'anomalies. Ce dernier doit permettre d'identifier, pour chaque anomalie détectée, l'ensemble des flux qui l'ont généré. Le développement d'un tel module constitue une autre perspective de ce travail.

Enfin, nous proposons d'explorer les techniques de calcul distribuées afin de les exploiter pour la mise en place d'approches de détection d'anomalies coopératives permettant aux opérateurs de réseaux, administrativement distincts, de coopérer et de partager leurs données d'audit en toute sécurité..

Bibliographie

[Abry98] P. Abry and D. Veitch. "*Wavelet analysis of long range dependent traffic*", IEEE Trans. Information Theory, Vol 44, N 1, pp. 2–15, 1998.

[Allm07] M. Allman, V. Paxson, J. Terrell, "*A Brief History of Scanning*", In: ACM Internet Measurement Conference, pp. 77–82, 2007.

[Ande80] J. P. Anderson, "*Computer security threat monitoring and surveillance*", James P. Anderson Co, Fort Washington, PA, 1980.

[Apsi96] J. Apisdorf, k claffy, K. Thompson, and R. Wilder, "*OC3MON: Flexible, affordable, high performance statistics collection*", Proceedings of Proceedings of the 10th USENIX conference on System administration, pp. 97–112, 1996.

[Auss07] J. Aussibal, P. Borgnat, Y. Labit, G. Dewaele, N. Larrieu, L. Gallon, P. Owezarski, P. Abry, K. Boudaoud, "*Base de traces d'anomalies légitimes et illégitimes*", 6th Conference on Security in Network Architectures and Informations Systems (SAR-SSI 2007), pp. 176-185, June 2007.

[Axel98] S. Axelsson, "*Research in intrusion detection systems: a survey*", Department of computer engineering, Chalmers university of technology, Sweden, technical report, 1998.

[Ayar05] H. Ayari, K. Ramah, F. Kamoun, "*Characterization of failures in the Tunisian National University Network*", poster NetCon, Lannion France, November 2005.

[Ayar05a] H. Ayari, *"Caractérisation des pannes et des anomalies dans le réseau national universitaire"*, rapport de mastère, Université de Manouba, ENSI, 2005.

[Ayar05b] H. Ayari, K. Ramah, F. Kamoun, "*Caractérisation des pannes dans le Réseau National Universitaire*", Journées de Génie Électrique et Informatique GEI, Sousse, Tunisie, 2005.

[Bail05] M. Bailey, E. Cooke, F. Jahanian, J. Nazario, and D. Watson, "*The Internet Motion Sensor: A Distributed Blackhole Monitoring System*". In Proceedings of the Network and Distributed Security Symposium, pp. 167-179, 2005.

[Bara02] C. Barakat, P. Thiran, G. Iannaccone, C. Diot, P. Owerzaski, *"A flow-based model for Internet backbone traffic"* In proceeding of ACM SIGCOMM Internet Measurement workshop, pp. 35-47, 2002.

[Barf02] P. Barford, J. Kline, D. Plonka and A. Ron, "*A signal analysis of network traffic anomalies*", ACM SIGCOMM Internet Measurement Workshop, pp. 71-82, November 2002,.

[Bena04] N. BEN AZZOUNA, *"Étude des méthodes d'échantillonnage des flux pour la mesure dans les réseaux large bande "*, Thèse Université Pierre et Marie Curie, Paris VI, 2004

[Bera95] J. Beran, R. Sherman, M. S. Taqqu and W. Willinger, "*Long-range dependence in Variable-Bit-Rate video traffic*", IEEE Transactions on Communications, Vol. 43, N. 2/3/4, pp. 1566-1579, 1995.

[Bhat01] S. Bhattacharyya, C. Diot, J. Jetcheva, and N. Taft, "*Pop-level and access-link-level traffic dynamics in a tier-1 POP"* in Proceeding of ACM SIGCOMM Internet Measurement Workshop 2001, pp. 39–54, 2001.

[Borg07] P. Borgnat, P. Abry, G. Dewaele, A. Scherrer, N. Larrieu, P. Owezarski, Y. Labit, L. Gallon, J. Aussibal , "*Une caractérisation non gaussienne et à longue mémoire du trafic Internet et de ses anomalies : validation expérimentale et application à la détection d'attaque de DdoS*", Annales de Télécommunications, Vol 62, pp. 1401-1428, 2007.

[Borg07a] P. Borgnat, G. Dewaele, P. Abry , "*Identification d'anomalies statistiques dans le trafic Internet par projections aléatoires multirésolution*", 21eme Colloque sur le Traitement du Signal et des Images. GRETSI, 2007.

[Borg09] P Borgnat, G Dewaele, K Fukuda, P Abry, K. Cho , "*Seven Years and One Day: Sketching the Evolution of Internet Traffic*", Proceedings of INFOCOM, 2009.

[Bouc06] H. Boucetta, "*Mise en place d'une plate-forme de mesures passives pour le réseau RNU"*, rapport de mastère de l'Université de Manouba, ENSI, 2006

[Bouc07] H. Boucetta, K. Ramah, F. Kamoun, "*Caractérisation du trafic sur le réseau national universitaire*", Conférence SETIT07, Hammamet, 2007.

[Bouz04] Y. Bouzida, F. Cuppens, N. Cuppens-Boulahia, S. Gombault. "*Efficient Intrusion Detection Using Principal Component Analysis*", 3ème Conférence sur la Sécurité et Architectures Réseaux (SAR), 2004.

[Broi04] A. Broi, Y. Hyun, R.Gea, Kc.Claffy, "*Their share: Diversity in disparity in IP traffic*", In proceeding of passive and active measurement workshop (PAM), pp. 113-125, April 2004

[Brow02] N. Brownlee, "Understanding Internet Traffic Streams: Dragonflies and Tortoises", IEEE Communications magazine, Vol 40, N 10, pp. 110–117, October 2002.

[Brut00] J. Brutlag, "*Aberrant Behaviour Detection in Time Series for Network Monitoring*", in Proceeding of the USENIX System Administration Conference LISA XIV, pp. 139-146, December 2000.

[Cace00] R. Caceres, N. G. Duffield, A. Feldmann, J. Friedmann, A. Greenberg, R. Greer, T. Johson, C. Kalmanek, B. Krishnamurthy, D. Lavelle, P.P. Mirshra, K.K. Ramakrishnan, J. Rexford, F. True, J.E. Van der Merwe, "*Measurement and analysis of IP network usage and behavior"*, IEEE Communications, Vol 38, N 5, pp 144–151, May 2000.

[Cair05] C. Di Cairano-Gilfedder, R. G. Clegg, "*A decade of Internet research -- advances in models and practices"*, BT Technology Journal, Vol 23, N 4, pp. 115-128, 2005

[Cao01] J. Cao, W. S. Cleveland, D. Lin and D.X. Sun, "*Internet traffic tends to Poisson and independent as the load increases*", Technical report, Bell Labs, 2001.

[Cao02] J. Cao, W. S. Cleveland, D. Lin, and D. X. Sun, "*Internet Traffic Tends Toward Poisson and Independent as the Load Increases*", In Nonlinear Estimation and Classification, Springer eds, pp. 83-109, 2002.

[Carm07] M. F. F. Do Carmo, R. Holanda, J. E. Maia, J. N. de Sousa, "*Attack Detection based on Statistical Discriminators*", Global Information Infrastructure Symposium (GIIS), pp. 181-186, 2007.

[Casa05] M. Casado, T. Garfinkel, W. Cui, V. Paxson and S. Savage, "*Opportunistic Measurement: Extracting Insight from Spurious Traffic*", Proceedings of HOTNETS, 2005.

[Chan09] V. Chandola, A. Banerjee, and V. Kumar, "*Anomaly Detection - A Survey*", ACM Computing Surveys, Vol. 41, N 3, pp. 1-58, 2009.

[Cho06] K. Cho, K. Fukuda, H. Esaki, A. Kato, "*The impact and implications of the growth in residential user-to-user traffic*", Proceedings of Sigcomm, pp. 207-218, 2006.

[Claf95] K. Claffy, H-W Braun, G. Polyzos, *"A parametrizable methodology for Internet traffic flow profiling"*, IEEE Journal on Selected Areas in Communication, Vol. 13, N° 8, pp. 1481-1494, 1995.

[Clea00] J. Cleary, S. Donnelly, I. Graham, A. McGregor and M. Pearson, "*Design Principles for Accurate Passive. Measurement*", Proceedings of The First Passive and Active Measurement Workshop, 2000.

[Cove91] T. M. Cover and J. A. Thomas, "*Elements of information theory*", John Wiley and Sons, Inc., 1991.

[Crov96] M. Crovella and A. Bestavros. "*Selfsimilarity in world wide web traffic: Evidence and possible causes*", Proceedings of ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems, Vol 5, pp. 835–846, 1996.

[Denn87] D. E. Denning, "*An intrusion detection model*", IEEE transactions in software engineering, Vol 13, pp. 222-232, 1987.

[Down99] A. B. Downey, "*Using Pathchar to Estimate Internet Link Characteristics*", ACM SIGCOMM, pp. 222-232, 1999.

[Feam05] N. Feamster, J. Jung, H. Balakrishnan, "*An Empirical Study of Bogon Route Advertisements*", Computer Communication Review, Vol. 35, N. 1, pp. 63–70, 2005.

[Feld00] A. Feldmann, "*Characteristics of TCP connection arrivals*", In Self-similar network traffic and performance evaluation, edited by K. Park and W. Willinger, J. Wiley & Sons, 2000.

[Floy01] S. Floyd and V. Paxson , "*Difficulties in Simulating the Internet*", IEEE/ACM Transactions on Networking, Vol 9, N 4, pp. 392-403, August 2001.

[Fome04] M. Fomenkov, K. Keys, D. Moore, Kc. Claffy, "*Longitudinal study of Internet traffic from 1998-2003*", Winter International Symposium on Information and Communication Technologies, 2004.

[Fral01] C. Fraleigh, C. Diot, B. Lyles, S. Moon, P. Owezarski, D. Papagiannaki, F. Tobagi, *"Design and Deployment of a Passive Monitoring Infrastructure"*, Proceeding of Passive and Active Measurement, Lecture notes in computer science Vol. 2170, pp. 556-575, April 2001.

[Fral03] C. Fraleigh, S. Moon, B. Lyles, C. Cotton, M. Khan, D. Moll, R. Rockell, T. Seely, C. Diot, *"Packet-level traffic measurements from the sprint IP backbone"*, IEEE Network, Vol 17, N. 6, pp. 6-16, November 2003.

[Fran07] J. Franklin, V. Paxson, A. Perrig, and S. Savage, "*An Inquiry into the Nature and Causes of the Wealth of Internet Miscreants*", Proceedings of ACM conference on Computer and Communications Security, pp. 375–388, October 2007.

[Gu05] Y. Gu, A. McCallum, and D. Towsley, "*Detecting Anomalies in Network Traffic Using Maximum Entropy Estimation*", Internet Measurement Conference, pp. 345–350, 2005.

[Jaco97] V. Jacobson. "*Pathchar -- a tool to infer characteristics of internet paths*", Mathematical Sciences Research Institute (MSRI) technical report, April 1997

[Jena00] A. K. Jena, A. Popescu and P. Pruthi, "*Modeling and analysis of HTTP traffic*", In ITC Specialist Seminar on IP Traffic Modeling, Measurement and Management, September 2000.

[Jung04] J. Jung, V. Paxson, A. W. Berger, H. Balakrishnan, "*Fast Portscan Detection Using Sequential Hypothesis Testing*", IEEE Symposium on Security and Privacy, pp. 211-225, 2004.

[Kara02] T. Karagiannis, M. Faloutsos. "SELFIS: A Tool For Self-Similarity and Long-Range Dependence Analysis", Workshop on Fractals and Self-Similarity in Data Mining: Issues and Approaches, 2002.

[Kara03] T. Karagiannis, M. Faloutsos, M. Molle. "*A User-Friendly Self-Similarity Analysis Tool*", Special Section on Tools and Technologies for Networking Research and Education, ACM SIGCOMM Computer Communication Review, Vol. 33, pp. 81-93, 2003.

[Kara04] T. Karagiannis, A. Broido, N. Brownlee, Kc. Claffy and M. Faloutsos, *"Is P2P dying or just hiding"*, Globel Communication Conference, Vol.3, pp. 1532–1538, December 2004.

[Kohl06] E. Kohler, J. Li, V. Paxson, and S. Shenker, "*Observed Structure of Addresses in IP Traffic*", IEEE/ACM Transactions on Networking, Vol. 14, N. 6, pp. pp. 1207-1218, December 2006.

[Komp07] R. Kompella, S. Singh, G. Varghese: "*On Scalable Attack Detection in the Network*", IEE/ACM Transactions on Networking, Vol. 15, N. 1, pp. 14-25, 2007.

[Krei04] C. Kreibich, J. Crowcroft, "*Honeycomb-Creating Intrusion Detection Signatures Using. Honeypot*", ACM SIGCOMM Computer Communication Review, Vol. 34, N. 1, pp. 51–56, January 2004.

[Krish03] B. Krishnamurthy, S. Sen, Y. Zhang, Y. Chen, "*Sketch-based change detection: methods, evaluation, and applications*". Internet Measurement Conference, pp 234-247, 2003.

[Labi05] Y. Labit, P. Owezarski, N. Larrieu, "*Evaluation of active measurement tools for bandwidth estimation in real environment*", 3rd IEEE/IFIP Workshop on End to End Monitoring Techniques and Services (E2EMON'05), May 2005.

[Lakh04] A. Lakhina, M. Crovella, C. Diot, "*Diagnosing network traffic anomalies in traffic flows*", SIGCOMM, pp. 219-230, 2004.

[Lakh04b] A. Lakhina, M. Crovella, C. Diot, "*Characterization of network-wide anomalies in traffic flows*", Internet Measurement Conference, pp. 201-206, 2004.

[Lakh05] A. Lakhina, M. Crovella, C. Diot, "*Mining anomalies using traffic feature distributions*", SIGCOMM, pp. 217-228, 2005.

[Larr05] N. Larrieu, "*Contrôle de congestion et gestion du trafic à partir de mesures pour l'optimisation de la QoS dans l'Internet*", Thèse de doctorat de l'INSA de Toulouse, 2005.

[Laza03] A. Lazarevic, L. Ertoz., A. Ozgur, J. Srivastava, V. Kumar, "*A Comparative Study of Anomaly Detection Schemes in Network Intrusion Detection*", Proceedings of SIAM Conference on Data Mining, 2003.

[Leck02] C. Leckie, R. Kotagiri, "A probabilistic approach to detecting network scans", 8th IEEE Network Operations and Management Symposium (NOMS), pp. 359 - 372, 2002

[Lela94] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson, "*On the self-similar nature of Ethernet traffic (extended version)"*. IEEE/ACM Transactions on Networking, Vol. 2, N. 1, pp.1–15, 1994.

[Li06] X. Li, F. Bian, M. Crovella, C. Diot, R. Govindan, G. Iannaccone, A. Lakhina: "*Detection and identification of network anomalies using sketch subspaces*", Internet Measurement Conference, pp 147-152, 2006..

[Mccr00] S. McCreary and K. C. Claffy, "*Trends in Wide Area IP traffic patterns : A view from Ames Internet Exchange*", ITC Specialist Seminar on IP Traffic Modeling, Measurement and Management, 2000.

[Mchu00] J. Mchugh, "*Testing Intrusion Dectection Systems : a critique of the 1998 and 1999 DARPA Intrusion Detection System evaluations as performed by Lincoln Laboratory*", ACM Transactions on Information and System Security, Vol.3, N° 4, pp. 262–294, 2000.

[Mell05] M. Mellia, R. Lo Cigno, F. Neri, "*Measuring IP and TCP behavior on edge nodes with Tstat"*, Computer Networks, Vol. 47, N. 1, pp. 1-21, 2005.

[Mell06] M. Mellia, M. Meo, L. Muscariello, D. Rossi, "*Passive Identification and Analysis of TCP Anomalies*", IEEE International Conference of Communication (ICC'06), Vol. 2, pp. 723-728, 2006.

[Mich01a] J. Micheel, S. Donnelly, I. Graham, "*Precision Timestamping of Network Packets*", Internet measurement workshop, pp. 273 – 277, 2001.

[Mich01b] J. Micheel, H. Braun, I. Graham, "*Storage and bandwidth requirements for passive Internet header traces*", workshop in network related data management, 2001.

[Moln00] S. Molnar and T. D. Dang, "*Scaling analysis of IP traffic components"*, Proc. of ITC Spec. Seminar on IP Traffic Modeling, Measurement and Management, pp. 18–20, 2000.

[Moor01] D. Moore, G. Voelker, and S. Savage, "*Inferring Internet Denial of Service activity*", In Proceedings of the 2001 USENIX Security Symposium, pp. 9-22, August 2001.

[Moor02] D. Moore, C. Shannon, and J. Brown, "*CodeRed: a Case Study on the Spread and Victims of an Internet Worm*", In ACM SIGCOMM/USENIX Internet Measurement Workshop, pp. 273-284, 2002.

[Moor03] D. Moore, V. Paxson, S. Savage, C. Shannon, S. Staniford, and N. Weaver, "*Inside the Slammer Worm*", IEEE Security and Privacy, Vol. 1 N. 4, pp. 33–39, July 2003..

[Moor04] D. Moore, C. Shannon, G.Voelker and S. Savage, "*Network Telescopes: technical report*", Cooperative Association for Internet Data Analysis (CAIDA), July 2004.

[Moor05] A. Moore and K. Papagiannaki. "*Toward the Accurate Identification of Network Applications"*, In Passive and Active Measurement workshop, pp. 41-54, March 2005.

[Nabe98] M. Nabe, M. Murata and H. Miyahara, "*Analysis and modeling of World Wide Web traffic for capacity dimensioning of Internet access lines*", Performance Evaluation, Vol. 34, pp. 249-271, 1998.

[Navr03] J. Navratil. "*ABwE: A Practical Approach to Available Bandwidth Estimation*", In Passive and Active Measurement workshop, April 2003.

[NARC07] "*Toward a Safer and More Secure Cyberspace*", Technical report National Academy Research Council and National Academy of Engineering, October 2007.

[Nych08] G. Nychis, V. Sekar, D.G. Andersen, H. Kim, H. Zhang, "*An Empirical Evaluation of Entropy-based Anomaly Detection*", Internet Measurement Conference, pp. 151-156, 2008.

[Oliv01] P. Olivier and N. Benameur, "*Flow level IP traffic characterization*", Proceedings of International Teletraffic Congress, December 2001.

[Oliv03] P. Olivier, P. Owezarski, K. Salamatian, "*Quelques éléments caractéristiques du trafic Internet"*, Colloque International : Mesures de l'Internet, May 2003

[Owez04] P. Owezarski, N. Larrieu., "*Internet traffic characterization--An analysis of traffic oscillations*", IEEE International Conference on High Speed Networks and Multimedia Communications (HSNMC), July 2004.

[Owez07] P. Owezarski, N. Larrieu, L. Bernaille, W. Saddi, F. Guillemin, A. Soule, K. Salamatian, *"Distribution of traffic among applications as measured in the French METROPOLIS project"*, Annales des télécommunications Vol. 62, N°3-4, pp. 369-386, 2007.

[Pang04] R. Pang, V. Yegneswaran, P. Barford, V. Paxson, Vern; L. Peterson, "*Characteristics of Internet Background Radiation*", Proceedings of ACM SIGCOMM Internet Measurement Conference, pp. 27-40, 2004.

[Pang05] R. Pang, M. Allman, M. Bennett, J. Lee, V. Paxson and B. Tierney, "*A First Look at Modern Enterprise Traffic*", Proceedings of ACM SIGCOMM Internet Measurement Conference, pp. 2-2, October 2005.

[Papa01] K. Papagiannaki, N. Taft, S. Bhattacharyya, P. Thiran, K. Salamatian et C.Diot, "*On the feasibility of identifying elephants in Internet backbone traffic*", Sprint ATL technical report, 2001.

[Park96] K. Park, G. Kim, M. E. Crovella, "*On the Relationship Between File Sizes, Transport Protocols, and Self-Similar Network Traffic*". In Proceedings of the International Conference on Network Protocols, pp 171-180, October, 1996.

[Park97] K. Park, G. Kim, M. Crovella, "*On the Effect of Traffic Self-similarity on Network Performance*", SPIE International Conference on Performance and Control of Network Systems, vol. 3231, pp. 296-310, November 1997

[Park00] K. Park and W.Willinger, "*Self-similar network traffic : an overview*", In Self-similar network traffic and performance evaluation, edited by K. Park and W. Willinger, J. Wiley & Sons, 2000.

[Pasz02] A. Pasztor, D. Veitch, "*PC based precision timing without GPS*", Proceeding ACM SIGMETRICS, pp. 1-10, 2002.

[Patc07] A. Patcha, J.-M. Park, "*Network Anomaly Detection with Incomplete Audit Data*", Elsevier Computer Networks, Vol. 51, Issue 13, pp. 3935–3955, 2007.

[Paxs94a] V. Paxson, "*Growth trends in wide-area TCP connection,*", IEEE Network, vol. 8, no. 4, pp. 8-17, July 1994.

[Paxs94b] V. Paxson and S. Floyd. "*Wide area traffic: The failure of Poisson modelling*". ACM SIGCOMM, pp. 257–268, 1994.

[Paxs94c] V. Paxson, "*Empirically-derived analytic models of wide-area TCP connections*", IEEE/ACM Transactions on Networking vol. 2, no. 4, pp. 316– 336, August 1994.

[Paxs97] V. Paxson, "*Measurements and Analysis of End-to-End Internet Dynamics*", Ph.D. dissertation, university of California, Berkeley, 1997.

[Paxs98] V. Paxon, G.Almes, J.Mahdavi, M.Mathis, "*Framework for IP Performance Metrics*", RFC 2330, mai 1998.

[Paxs99] V. Paxson, "*Bro : a system for detecting network intruders in real-time*", Computer Networks, Vol. 31, N. 23-24, pp. 2435-2463, 1999.

[Paxs00] V. Paxson, A. Adams et M.Mathis, "*Experiences with NIMI*", Passive and Active Measurement Workshop, 2000.

[Paxs03] V. Paxon, "*Some not-so-pretty admissions about dealing with Internet measurements*", invited talk, Stanford Networking Seminar, 2003.

[Plon06] D. Plonka, "*Flawed Routers Flood University of Wisconsin Internet Time Server*", Technical report Wisconsin University, available on http://www.cs.wisc.edu/~plonka/netgear-sntp/, 2006.

[Rama05] K. Ramah, F. Kamoun, "*Métrologie dans les réseaux Internet : Cas du RNU*", Journées de génie électrique et informatique GEI 2005, Sousse, Tunisie, 2005.

[Rama06] K. Ramah, H. Ayari, F. Kamoun,, "*Traffic Anomaly Detection and Characterization in the Tunisian National University Network*", IFIP networking 2006, Lecture notes in computer Science 3976, pp 136-147, May 2006.

[Rama07] K. Ramah, F. Kamoun, "*Measurement of Spurious Traffic in RNU Network*", JSEABA conference, Tunis, 2007.

[Rama09] K. Ramah, K. Salamatian, F. Kamoun, "*Scan Surveillance in Internet Networks*", IFIP/TC6 NETWORKING, pp. 614-625, May 2009.

[Ring07] H. Ringberg, A. Soule, J. Rexford, C. Diot, "*Sensitivity of PCA for traffic anomaly detection*", ACM SIGMETRICS, pp 109-120, 2007.

[Roes99] M. Roesch., "*Snort: Lightweight intrusion detection for networks*", Proceedings of the 13th Conference on Systems Administration (LISA), pp. 229--238, 1999.

[Seka02] R. Sekar, Ajay Gupta, James Frullo, Tushar Shanbhag, Abhishek Tiwari, Henglin Yang and Sheng Zhou , "*Specification-based anomaly detection: a new approach for detecting network intrusions*", ACM Conference on Computer and Communications Security, October 2002.

[Shyu03] M-L. Shyu, S-C. Chen, K. Sarinnapakorn, and L. Chang, "*A Novel Anomaly Detection Scheme Based on Principal Component Classifier*", In Proceedings of the IEEE Foundations and New Directions of Data Mining Workshop, pp. 172-179, 2003.

[Silv07] T. Silverston, O. Fourmaux and K. Salamatian, "*Characterization of P2P IPTV Traffic: Scaling Analysis*", Rapport technique université Pierre et Marie Curie, Laboratoire d'informatique Paris VI, 2007.

[Simo06] G. Simon, H. Xiong, E. Eilertson, V. Kumar, "*Scan Detection: A Data Mining Approach*", SIAM International Conference on Data Mining, pp. 118-129, 2006.

[Soul04] A. Soule, K. Salamatian, N. Taft, R. Emilion, K. Papagiannaki "*Flow Classification by histograms or how to go to Safari over Internet*", In proceedings of ACM Sigmetrics, pp. 49–60, 2004.

[Stan02] S. Staniford, V. Paxson, and N. Weaver, "*How to 0wn the Internet in your Spare Time*", in Proceedings of the 11th USENIX Security Symposium, pp. 149–167, 2002.

[Thom97] K. Thompson, G. Miller, M. Wilder, "*wide-area Internet traffic patterns and characteristics (Extended version)*", IEEE network, Vol. 11, pp. 10–23, 1997.

[Wagn05] A. Wagner, B. Plattner, "*Entropy Based Worm and Anomaly Detection in Fast IP Networks*", Proceedings of the IEEE International Workshops on Enabling Technologies, pp. 172-177, 2005.

[Will97] W. Willinger, M. Taqqu, R. Sherman, D. Wilson, "*Self-Similarity Through High-Variability: Statistical Analysis of Ethernet LAN Traffic at the Source Level*", IEEE/ACM Transactions on Networking, Vol. 5, No. 1, pp. 71-86, February 1997

[Yegn03] V. Yegneswaran, P. Barford, J. Ullrichs, "*Internet Intrusions: Global Characteristics and Prevalence*", In Proceedings of ACM SIGMETRICS, pp. 138-147, June 2003.

[Yegn04] V. Yegneswaran; P. Barford, D. Plonka, "*On the Design and Utility of Internet Sinks for Network Abuse Monitoring*", In Proceedings of Symposium on Recent Advances in Intrusion Detection (RAID), pp. 146–165, 2004.

[Yegn05] V. Yegneswaran; J.T. Giffin, P. Barford, S. Jha, "*An Architecture for Generating Semantics-Aware Signatures*", Proceedings of USENIX Security Symposium, pp. 97-112, 2005.

[Yoo04] I. Yoo, "*Protocol Anomaly Detection and Verification*", Proceedings of the IEEE Information Assurance Workshop, pp. 74–81, June 2004.

[Zesh03] C. Zesheng, G. Lixin, and K. Kevin, "*Modeling the Spread of Active Worms*", IEEE INFOCOM, Vol; 3, pp. 1890-1900, 2003.

Netographie

[Caida] http://www.caida.org, dernière consultation mars 2008.

[CCK] "*Site web du Centre de Calcul el Khawarizmi*", http://www.cck.rnu.tn, dernière consultation mars 2008.

[Cidr] "*Bogon IP addresses*", http://www.cidr-report.org/bogons, dernière consultation décembre 2008.

[Cis] http://www.cisco.com/en/US/products/ps6601/products_white_paper0900aecd80406232.shtml, dernière consultation mars 2007.

[Complete] "*Bogon IP addresses*", www.completewhois.com/bogons, dernière consultation décembre 2008.

[Coralreef] "*Corelreef web site*", www.caida.org/tools/measurement/coralreef/, dernière consultation mars 2008.

[Dag] *"Cartes de capture spécialisées : DAG"*, http://www.endace.com/our-products/dag-network-monitoring-cards/, dernière consultation mars 2008.

[DARPA98] "*DARPA Intrusion Detection Data sets*", http://www.ll.mit.edu/IST/ideval/data/data_index.html, dernière consultation mars 2008.

[IANA] "*The Internet Corporation for Assigned Names and Numbers Web page"*, http://www.ianna.org, dernière consultation décembre 2008.

[IPMon] "*IPMON web site*", http://ipmon.sprint.com/ipmon.php, dernière consultation décembre 2008.

[IPPM] http://www.ietf.org/html.charters/ippm-charter.html, dernière consultation mars 2008.

[Ipsum] "*IPsumdump and IPaggcreate web site*", http://www.cs.ucla.edu/~kohler/ipsumdump/, dernière consultation avril 2009.

[ISC] "*Internet Storm Center*", http://isc.sans.org/, dernière consultation octobre 2009.

[KDD99] "*Knowledge Discovery in Databases Archive (KDD99)*", http://kdd.ics.uci.edu/, dernière consultation mars 2008.

[Lde] "*Implémentation de la méthode des ondelettes*", http://www.cubinlab.ee.unimelb.edu.au/~darryl/, dernière consultation décembre 2008.

[Mawi] "*MAWI web site*", http://www.wide.ad.jp/wg/mawi, dernière consultation octobre 2009.

[Metro] http://www.laas.fr/~owe/METROPOLIS/metropolis.html, dernière consultation décembre 2008.

[Nmap] "*Nmap free security scanner*", http://www.insecure.org/nmap/, dernière consultation mai 2009

[Pipe] J. Guojun, *"pipechar tool"*, http://dsd.lbl.gov/~jin/, dernière consultation mars 2007.

[Psamp] *"IETF Packet Sampling "*, http://www.ietf.org/html.charters/psamp-charter.html, dernière consultation mars 2008.

[Ripe] "*RIPE NCC web site*", http://www.ripe.net, dernière consultation mars 2008.

[Self] "*SELFIS web site*", http://www.cs.ucr.edu/~tkarag/Selfis/ Selfis.html, dernière consultation juillet 2008.

[Shadow] "*Shadowserver foundation*", http://www.shadowserver.org/wiki/, dernière consultation octobre 2009.

[Snort] "*SNORT web site*", http://www.snort.org, dernière consultation mai 2009.

[Storm] "*storm worm alert*", http://www.secuser.com/alertes/2007/storm-worm.htm, dernière consultation mai 2009.

[Tcpdump] "*Tcpdump web site*", http://www.tcpdump.org, dernière consultation mai 2007.

[Wireshark] "*Wireshark web site*", http://www.wireshark.org, dernière consultation mai 2007.

[Zoo] "*ZOO web site*", http://www.laas.fr/~owe/ZOO/index.htm, dernière consultation mars 2008.

Annexe A : Auto-similarité et dépendance à long terme

Un objet est dit auto-similaire s'il conserve sa forme, quelle que soit l'échelle à laquelle on l'observe. Cette définition correspond bien sûr à une vision simpliste et faisant appel à l'intuition, dans ce qui suit, nous proposons une définition mathématique formelle et rigoureuse des notions d'auto-similarité et de dépendance à long terme.

1. Définition de l'auto-similarité

Soit X=(Xt ; t=0,1,2,…) un **processus stochastique à covariance stationnaire**, c'est à dire un processus ayant les propriétés suivantes :

a) une moyenne constante $\mu = E[X_t]$

b) une variance finie $\sigma^2 = E[(X_t - \mu)^2]$

c) une fonction d'auto-corrélation donnée par (A.1) qui ne dépend que de k

$$r(k) = \frac{E[(X_t - \mu)(X_{t+k} - \mu)]}{E[(X_t - \mu)^2]} \quad (k = 0,1,2,....) \qquad \textbf{(A.1)}$$

Soit $X^{(m)} = (X_k^{(m)}; k = 1,2,3...)$ une nouvelle série temporelle obtenue en lissant la série temporelle d'origine par le calcul de la moyenne des données sur des blocs sans chevauchement de taille m. autrement dit, pour chaque m = (1,2,3,…), nous avons l'équation (A.2).

$$X_k^{(m)} = \frac{(X_{km-m+1} + ... + X_{km})}{m}, k \geq 1 \qquad \textbf{(A.2)}$$

Notons que pour chaque valeur de m, la série agrégée $X^{(m)}$ définit un processus à covariance stationnaire ; soit $r^{(m)}$ la fonction d'auto-corrélation correspondante.

Un processus X est dit **auto-similaire de paramètre de Hurst H,** si et seulement si pour tout m > 0, nous avons :

$m^{(1-H)} * X_k^{(m)}$ et X_k possèdent les mêmes distributions jointes à tous les ordres.

Notons que le paramètre de Hurst H doit être compris entre 0,5 et 1.

Cette définition signifie que si l'on modifie l'échelle sur laquelle on observe le processus par un facteur positif m et que l'on « zoome » le même processus par ce facteur élevé à la puissance (1-H), alors l'allure des deux processus obtenus est la même.

Un processus *X* est dit **exactement auto-similaire d'ordre 2 de paramètre de Hurst H** si les processus agrégés $m^{(1-H)} * X^{(m)}$ a la même moyenne et la même variance que le processus original *X* pour tout m. Autrement dit, *X* est exactement auto-similaire d'ordre 2 si les processus $X^{(m)}$ ne se distinguent pas de *X* au moins en considérant leurs propriétés statistiques d'ordre 2. Un processus exactement auto-similaire d'ordre 2 vérifie l'équation (A.3).

$$r^m(k)=r(k), \forall m=1,2,3,..., \forall k=1,2,3,... \quad \textbf{(A.3)}$$

Un processus *X* à covariance stationnaire est dit **asymptotiquement auto-similaire d'ordre 2** si $r^{(m)}(k)$ se rapproche asymptotiquement de la structure d'auto-corrélation de $r^{(k)}$ de *X* (pour m et k grands).

Un tel processus se caractérise par le fait que la variance du processus agrégé $X_k^{(m)}$ décroît lentement lorsqu'on augmente m et que la fonction d'auto-corrélation r(k) décroît aussi lentement lorsqu'on augmente le retard k.

Dans ce qui suit, nous désignons par processus auto-similaire, un processus asymptotiquement auto-similaire d'ordre 2. En effet, l'auto-similarité asymptotique d'ordre 2 est une propriété beaucoup plus faible que l'auto-similarité exacte est un concept mathématique qui ne peut être rencontré que dans les objets construits mathématiquement d'une manière itérative. De plus, cet abus de langage est adopté par les études de caractérisation et de modélisation du trafic Internet, rencontrées dans la littérature.

2. Définition de la dépendance à long terme (LRD)

Un processus **à dépendance longue** ou à mémoire longue signifie que la dépendance entre deux variables du processus ne diminue pas trop rapidement avec l'éloignement temporel. Ainsi, d'après cette définition, un processus asymptotiquement auto-similaire d'ordre 2 est un exemple de processus à mémoire longue.

Soit X un processus stochastique à covariance stationnaire, on dit que X est **à dépendance longue** s'il a une fonction d'auto-corrélation non sommable c'est à dire qu'elle satisfait l'équation (A.4) :

$$\sum_{k=-\infty}^{k=+\infty} r(k) = \infty \quad \textbf{(A.4)}$$

Pour représenter mathématiquement un processus à dépendance longue, on utilise, généralement, l'équation (A.5), avec $0 \leq \beta \leq 1$ et α_1 est une constante positive finie.

$$r(k) \sim \alpha_1 k^{-\beta} \; quand \; k \to \infty \quad \textbf{(A.5)}$$

Dans ce cas, la fonction d'auto-corrélation décroît hyperboliquement lorsque k augmente impliquant une fonction d'auto-corrélation non sommable. Dans [Park00], les auteurs ont démontré qu'un processus dépendant à long terme, est un processus asymptotiquement auto-similaire d'ordre 2, ayant comme paramètre de Hurst $H=1-\beta/2$.

D'après [Park00], les notions de processus asymptotiquement auto-similaire d'ordre 2 ayant un paramètre de Hurst *1/2 <H<1* et de processus dépendant à long terme défini par l'équation (A.5) sont équivalentes. C'est pourquoi, nous utiliserons, dans la suite, indifféremment les termes LRD et auto-similarité.

3. Calcul du paramètre de Hurst

Le paramètre de Hurst permet de mesurer le degré de LRD ou d'auto-similarité d'un processus stochastique. En effet, un facteur de Hurst compris entre 0, 5 et 1 indique la présence de LRD dans le processus analysé : plus la valeur de ce facteur est proche de 1 plus la LRD est forte. A l'inverse, si le facteur est inférieur à 0,5, il n'y a pas de LRD. Il existe plusieurs tests statistiques permettant de vérifier la présence de LRD dans une série temporelle X_t et d'estimer la valeur du paramètre de Hurst [Kara03 et Kara02]. La méthode d'Abry-veitch utilisant la décomposition en ondelettes est considérée comme étant la plus fiable [Abry98]. Elle consiste en une approximation par la méthode des moindres carrés des coefficients des ondelettes à différentes échelles de temps ; la moyenne des carrés de ces coefficients représente une estimation du paramètre de Hurst. Cette méthode est implémentée sous forme d'un ensemble de routines (LDESTIMATE) écrites en Matlab ou en C librement accessibles via Internet [Lde]. De plus, il existe des programmes avec interfaces graphiques conviviales qui intègrent l'outil LDESTIMATE et permettent ainsi de faciliter son utilisation. Il s'agit notamment des programmes Selfis [Self] et Zoo [Zoo].

Annexe B : Techniques d’attaques via les réseaux

Pour aboutir, une attaque sur un système informatique, doit exploiter au moins une vulnérabilité liée aux applications utilisées sur ce système, aux protocoles réseaux, au système d’exploitation ou encore à l’utilisateur lui même. Ainsi, il y a autant de types d’attaques que de vulnérabilités. Dans ce qui suit, nous proposons un aperçu sur les principales techniques d’attaques via les réseaux.

1 Techniques de scan

Le balayage de port, appelé portscan en anglais, est une technique pour rechercher les ports ouverts sur une machine cible. Elle est souvent utilisée par les administrateurs pour contrôler la sécurité des hôtes de leurs réseaux, et par les pirates informatiques pour tenter de la compromettre. Un balayage de port effectué sur un système tiers est généralement considéré comme tentative d’intrusion dans la mesure où il sert souvent à préparer une intrusion.

Les balayages de ports visent typiquement le protocole TCP, car c’est celui qui est utilisé par la majorité des applications. L’objectif est de savoir si un logiciel est en écoute sur un numéro de port donné ou non. Si un logiciel écoute sur un port, alors on dit qu’il est ouvert, sinon on dit qu’il est fermé. Le balayage d’un port se passe en deux étapes :

1. Envoi d’un paquet sur le port testé
2. Analyse de la réponse

Il existe une douzaine de techniques de scan, implémentées dans l’outil libre Nmap [Nmap]. Idéalement, la meilleure technique de scan est celle qui est la plus furtive afin de ne pas alerter les soupçons de la future victime. Voici une description des techniques de scan les plus répandues :

• Le scan simple : appelé aussi le scan connect(), il consiste à établir une connexion TCP complète sur une suite de ports. S’il arrive à se connecter, le port est ouvert ;sinon, il est fermé. Cette méthode de scan est très facilement détectable.

• Le scan demi-ouvert : appelé aussi scan SYN, il s’agit d’une amélioration du scan simple. Ce scan essaie également de se connecter sur des ports donnés, mais il n’établit pas complètement la connexion : il n’envoie pas de commande ACK (acquittement) après avoir reçu l’accord de se connecter. Grâce à ceci, la méthode est bien plus furtive que le scan normal.

• Les scans XMAS, NULL et FIN : se basent sur des détails de la RFC du protocole TCP pour déterminer si un port est fermé ou non en fonction de la réaction à certaines requêtes. Ces scans sont moins fiables que le scan SYN mais ils sont un peu plus furtifs. La différence entre ces trois types de scan se situe au niveau des flags TCP utilisés lors de la requête.

• Le scan à l'aveugle, appelé aussi « idle scan », s'effectue via usurpation d'adresse IP d'une machine intermédiaire (voir plus bas) ; il exploite les numéros de fragmentation IP pour inférer l'état des ports sur la machine cible. Le système attaqué, s'il dispose d'un IDS, ne peut pas remonter jusqu'à l'adresse IP du pirate et pense que le scan est réalisé par la machine intermédiaire.

2 Techniques d'usurpation d'identité

2.1. ARP Spoofing

Le but de cette attaque est de rediriger le trafic d'une machine vers une autre. Ainsi, grâce à cette redirection, une personne mal intentionnée peut se faire passer pour une autre. De plus, le pirate peut rerouter les paquets qu'il reçoit vers le véritable destinataire, ainsi l'utilisateur usurpé ne se rendra compte de rien. La finalité est la même que l'IP spoofing mais on travaille ici au niveau de la couche liaison de données.

Pour réussir cette usurpation, il faut corrompre le cache ARP de la victime. Ce qui signifie qu'il faut lui envoyer des trames ARP en lui indiquant que l'adresse IP d'une autre machine est la sienne. Les caches ARP étant régulièrement vidés, il faudra veiller à maintenir l'usurpation.

2.2. IP spoofing

Le but de cette attaque est d'usurper l'adresse IP d'une autre machine afin de se faire passer pour une machine de confiance. Cette technique peut être utile dans le cas d'authentifications basées sur une adresse IP (services tels que rlogin ou ssh par exemple).

Pour réussir cette attaque, il existe des utilitaires qui permettent de modifier les paquets IP ou de créer ses propres paquets (ex : hping2). Grâce à ces utilitaires, il est possible de spécifier une adresse IP différente de celle que l'on possède, et ainsi se faire passer pour une autre « machine ». Cependant, ceci pose un problème : en spécifiant une adresse IP différente de la notre, nous ne recevrons pas les réponses de la machine distante, puisque celle-ci répondra à l'adresse spoofée. Il existe toutefois deux méthodes permettant de récupérer les réponses :

• Source routing : technique consistant à placer le chemin de routage directement dans le paquet IP. Cette technique ne fonctionne plus de nos jours, les routeurs rejetant cette option.

• Reroutage : cette technique consiste à envoyer des paquets RIP aux routeurs afin de modifier les tables de routage. Les paquets avec l'adresse spoofée seront ainsi envoyés aux routeurs contrôlés par le pirate et les réponses pourront être également reçues par celui-ci.

2.3. DNS Spoofing

Le but de cette attaque est de fournir de fausses réponses aux requêtes DNS, c'est-à-dire indiquer une fausse adresse IP pour un nom de domaine et rediriger, ainsi à leur insu, des internautes vers des sites falsifiés où ils vont envoyer leurs identifiants en toute confiance pour être récupérés par les pirates.

3 Vol de session (TCP Session Hijacking)

Le but de cette attaque est de rediriger un flux TCP afin de pouvoir outrepasser une protection par mot de passe. En effet, le contrôle d'authentification s'effectuant uniquement à l'ouverture de la session, un pirate réussissant cette attaque parvient à prendre possession de la connexion pendant toute la durée de la session.

Le principe de cette attaque est le suivant : dans un premier temps, le pirate doit écouter le réseau, puis lorsqu'il estime que l'authentification a pu se produire (délai de n secondes par exemple), il désynchronise la session entre l'utilisateur et le serveur. Pour ce faire, il construit un paquet avec, comme adresse IP source, celle de la machine de l'utilisateur et le numéro d'acquittement TCP attendu par le serveur. En plus de désynchroniser la connexion TCP, ce paquet permet au pirate d'injecter une commande via la session préalablement établie.

4 Les dénis de service réseau

Le déni de service est une attaque visant à rendre indisponible un service, une machine ou tout un réseau. Pour y parvenir, il existe plusieurs techniques dont notamment l'inondation de la cible par des requêtes afin de saturer ses ressources et la rendre totalement injoignable.

Voici quelques techniques d'attaques réseaux permettant de rendre une cible indisponible.

• SYN Flooding : exploite le mécanisme d'ouverture de connexion en 3 phases de TCP (Three Way Handshake : SYN / SYN-ACK / ACK). Le principe est de laisser un grand nombre de connexions TCP en attente. Le pirate envoie de nombreuses demandes de connexion (SYN) aux quelles il ne répond jamais avec ACK. Les connexions en cours occupent des ressources mémoire au niveau de la machine cible, ce qui va entraîner une saturation et l'effondrement du système.

• UDP Flooding : Le but est donc d'envoyer un grand nombre de paquets UDP, ce qui va occuper toute la bande passante et ainsi rendre indisponible toutes les connexions TCP.

• Packet Fragment : exploite une faille au niveau de la gestion de la défragmentation des paquets ICMP. L'attaque "ping of death" consiste à envoyer un paquet ICMP echo request dont la taille est supérieure à MTU du réseau, donc nécessitant une fragmentation, ce qui provoque un crash système.

• Smurfling : le pirate forge des paquets ICMP Echo request ayant comme adresse source celle de la victime et les envoie vers des adresses de broadcast. Cette machine cible va recevoir un nombre énorme de réponses, car toutes les machines vont lui répondre, et ainsi toute sa bande passante sera épuisée.

5 Les attaques applicatives

Les attaques applicatives exploitent des failles au niveau des programmes utilisés. La plupart de ces failles sont dues soit à des mauvaises configurations soit à des erreurs de programmation.

5.1. Les problèmes de configuration

La plupart des logiciels sont livrés avec une configuration par défaut permettant à l'utilisateur de faire fonctionner rapidement le programme. Celles-ci sont souvent non sécurisées, elles contiennent notamment des mots de passe par défaut. De plus, des erreurs peuvent apparaître lors de la configuration d'un logiciel notamment à cause une maîtrise insuffisante du logiciel.

5.2. Les bogues ou erreurs de programmation

Les bugs sont liés à un problème dans le code source, ils peuvent amener à l'exploitation de failles par un pirate informatique. On ne peut toutefois rien faire contre ce type de problèmes, si ce n'est attendre un correctif de la part du développeur.

– Les buffer overflows

Les buffer overflows, ou dépassement de la pile, sont une catégorie de bug particulière. Issus d'une erreur de programmation, ils permettent l'exploitation d'un shellcode à distance. Ce shellcode permettra à une personne mal intentionnée d'exécuter des commandes sur le système distant, pouvant aller jusqu'à sa destruction. L'erreur de programmation est souvent la même : la taille d'une entrée n'est pas vérifiée et l'entrée est directement copiée dans un buffer dont la taille est inférieure à la taille de l'entrée. On se retrouve donc en situation de débordement, et l'exploitant peut ainsi accéder à la mémoire.

– Les scripts web

Les langages interprétés notamment ceux du web (ex : Perl, PHP, ASP), sont aussi une source importante d'erreurs de programmation. C'est le cas, lorsque des scripts web utilisent des entrées saisies par un utilisateur qui ne sont pas correctement contrôlées. L'exemple classique est l'exploitation de fichier à distance, tel que l'affichage du fichier mot de passe du système en remontant l'arborescence depuis le répertoire web.

– Les injections SQL

Tout comme les attaques de scripts, les injections SQL profitent de paramètres d'entrée non vérifiés. Comme leur nom l'indique, le but des injections SQL est d'injecter du code SQL dans une requête de base de données. Ainsi, il est possible de récupérer des informations se trouvant dans la base ou encore de détruire des données.

6 Les attaques basées sur l'ingénierie sociale

L'ingénierie sociale (social engineering en anglais) est la discipline consistant à obtenir un bien ou une information en exploitant la confiance mais parfois également l'ignorance ou la crédulité de tierces personnes.

Les pirates informatiques ont souvent recours à ces techniques pour obtenir des informations confidentielles telles que les mots de passe et les coordonnées bancaires.

L'hameçonnage, appelé en anglais phishing, est une forme d'attaque informatique reposant sur l'ingénierie sociale. La technique consiste à faire croire à la victime qu'elle s'adresse à un tiers de confiance (banque, administration, site marchant, etc.) en lui envoyant un mail d'allure authentique afin que le destinataire livre de manière consentante ses données personnelles. Généralement, le mail en question inclut un lien piégé qui reproduit le site originel et invite les internautes à mettre à jour leurs coordonnées. L'internaute, trompé à la fois par l'adresse et la page, se croit connecté à un site de confiance et n'hésite alors pas à délivrer des informations confidentielles.

7 Les attaques distribuées

Les attaques distribuées consistent à lancer une attaque à partir d'un grand nombre de machines. Pour ce faire, le pirate va tenter de se rendre maître d'un nombre important de machines, en exploitant une faille au niveau de ces machine ou en leur envoyant des messages contenant des chevaux de Troie. Il construit ainsi un réseau de machines zombies ou botnet. Une fois ceci effectué, il va pouvoir prendre le contrôle de ces machines à distance et les commander à sa guise pour lancer une attaque distribuée notamment des dénis de service distribués.

Pour réussir une telle attaque, il suffit au pirate de donner l'ordre à toutes les machines zombies d'envoyer simultanément une demande de connexion à la machine cible, de manière à ce qu'elle soit submergée par du trafic et devient ainsi totalement inaccessible. Les outils de DDOS les plus connus sont Tribal Flood Network (TFN), TFN2K et Trinoo.

Les attaques distribuées restent aujourd'hui les plus redoutables car les plus difficiles à arrêter.