Détection, classification et identification d'anomalies de trafic

Silvia Farraposo, Philippe Owezarski, Edmundo Monteiro

Détection, classification et identification d'anomalies de trafic

Silvia Farraposo[1], **Philippe Owezarski**[2], **Edmundo Monteiro**[3]

[1] School of Technology and Management -Polytechnic Institute of Leiria
Alto-Vieiro, Morro do Lena, 2411-901 Leiria, Apartado 4163, Portugal
[2] *LAAS – CNRS / Université de Toulouse*
7 Avenue du Colonel Roche, 31077 Toulouse, cedex 4, France
3 DEI/CISUC - University of Coimbra
Pólo II – Pinhal de Marrocos 3030-290 Coimbra, Portugal

e-mail: silvia@estg.ipleiria.pt, owe@laas.fr, edmundo@dei.uc.pt

RÉSUMÉ. *Cet article présente un nouvel algorithme itératif – NADA – qui détecte, classifie et identifie les anomalies d'un trafic. Cet algorithme a pour objectif de fournir, en plus de ce que font d'autres algorithmes, toutes les informations requises pour stopper la propagation des anomalies, en les localisant dans le temps, en identifiant leur classes (e.g. attaque de déni de service, scan réseau, ou n'importe quel autre type d'anomalies), et en déterminant leurs attributs comme, par exemple, les adresses et ports sources et destinations impliqués. Pour cela, NADA repose sur une approche tomographique générique, multi-échelles, multi-critères et utilisant de multiples niveaux d'agrégation. De plus, NADA utilise un ensemble exhaustif de signatures d'anomalies qui ont été définies spécifiquement pour permettre de classifier ces anomalies. Ces signatures représentées sous forme graphique permettent une classification visuelle par les opérateurs réseaux. NADA a été validé en utilisant des traces de trafic contenant des anomalies connues et documentées comme celles collectées dans le cadre du projet MétroSec.*

ABSTRACT. *This paper deals with a new iterative Network Anomaly Detection Algorithm – NADA, which accomplishes the detection, classification and identification of traffic anomalies. Our approach goes one step further than others since it fully provides all information required to limit the extent of anomalies by locating them in time, identifying their classes (e.g., if it is a Denial of Service, a Network Scan, or other type of anomalies), and giving their features as, for instance, the source and destination addresses and ports being involved. For this purpose, NADA uses a generic multi-featured approach executed at different time scales and at different levels of IP aggregation. In addition, NADA uses an exhaustive set of anomaly signatures which have been specifically defined for anomaly classification purpose. These signatures, graphically represented, make possible a visual classification of anomalies by network operators. NADA has been validated using data traces containing documented anomalies as the one gathered in the MétroSec project.*

MOTS-CLÉS : *Anomalies de trafic, Identification, Signatures d'anomalies.*

KEYWORDS: *Traffic anomaly, Identification, Anomaly Signature.*

1. Motivations

Les anomalies de trafic en général et les attaques de dénis de service (DoS) en particulier sont un réel problème qui nuit à qualité de service dans les réseaux. C'est un des problèmes majeurs que les administrateurs voudraient pouvoir résoudre à la volée. Les anomalies font partie intégrante du trafic. Il est donc important mais aussi difficile de les détecter, de les classifier et de les identifier. Le traitement des anomalies détectées sera complètement différent suivant qu'elles sont légitimes – comme des foules subites – ou illégitimes – comme des attaques DoS.

La diversité des disfonctionnements du réseau a motivé la conception et le développement de NADA (Network Anomaly Detection Algorithm) dont les objectifs sont au nombre de trois :

– La détection des anomalies, i.e. déterminer si une anomalie est en train de se produire. En particulier, l'objectif sera de détecter toutes les anomalies, y compris les plus réduites. En effet, les attaques DoS sont de plus en plus souvent distribuées, chaque source de l'attaque ne générant que très peu de trafic, afin de rester invisible le plus longtemps possible. C'est en s'agrégeant massivement près de la victime que toutes ces composantes de l'attaque provoquent une dégradation brusque et importante du niveau de service fourni par le réseau et les serveurs qui y sont connectés : d'où l'intérêt de les détecter au plus tôt près de la source, i.e. lorsqu'elles ne représentent que peu de trafic ;

– La classification des anomalies, i.e. déterminer quel type d'anomalie est en train de se produire. Cela signifie déterminer si l'anomalie est légitime ou illégitime, et quel est son type précis (foule subite HTTP, scan réseau, attaque de SYN flooding, etc.) ;

– L'identification de l'anomalie, i.e. être capable de déterminer tous les paquets et flux de l'anomalie.

De plus, NADA est complètement générique. Il peut travailler sur n'importe quelle série temporelle produite à partir du trafic entrant (ou d'une trace de trafic enregistrée). Pour illustrer NADA, dans cet article, nous ne considèrerons que les 3 séries suivantes (mais NADA peut travailler sur un nombre bien plus important et un ensemble plus varié de séries temporelles si nécessaire) :

– le nombre de paquets par unité de temps;

– le nombre d'octets par unité de temps;

– le nombre de nouveaux flots par unité de temps.

Nous insistons fortement sur la nécessité de travailler sur plusieurs séries temporelles simultanément pour permettre des détections, classifications et identifications correctes des anomalies, et cela car chaque type d'anomalie agit différemment par exemple sur la liste des paramètres du trafic cités plus haut. De plus, comme nous avons conçu notre outil NADA à la demande d'opérateurs réseaux qui veulent pouvoir l'utiliser en toute maîtrise, il doit forcément considérer des caractéristiques représentative du réseau et de son

trafic, comme des octets, des paquets ou des flots, et cela en utilisant des statistiques simples. Ainsi, les séries temporelles sur le nombre de paquets SYN ou FIN pourraient être facilement utilisées par l'algorithme si elles sont nécessaires. Dans tous les cas, en utilisant uniquement des mathématiques simples, on souhaite faire de NADA un outil facilement et efficacement exploitable par des techniciens. Dans ce but, nous utiliseront une interface de communication simple, avec des représentations graphiques aussi souvent que possible.

Les capacités de détection d'anomalies de NADA à partir de l'analyse d'un trafic entrant permettent de mettre en évidence qu'une variation significative du trafic vient de se produire. Les capacités de classification permettent de signaler parmi un ensemble d'anomalies possibles de laquelle il s'agit. Les informations nécessaires à la classification sont produites par l'analyse simultanée des différentes séries temporelles, à différentes échelles de temps et avec différents niveaux d'agrégation. Enfin, les attributs d'identification dans le cas illustré dans cet article donnent des informations sur les entités impliquées dans l'anomalie, i.e. les adresses et ports source et destination. De plus, les distributions des attributs IP caractérisant un type d'anomalie ont une représentation graphique unique facilement identifiable. C'est cette signature graphique qui sera envoyée à l'opérateur réseau (technicien ou logiciel) pour qu'il lance la procédure corrective adéquate.

D'autres approches pour détecter les anomalies du trafic existent. Cependant, à notre connaissance, aucune d'entre elles ne permet de simultanément détecter, classifier et identifier les anomalies du trafic. La plupart des travaux dans ce domaine se concentrent sur la phase de détection. C'est le cas des premiers travaux comme ceux de Bardford et al. [1] et Krishnamurthy et al. [10]. D'autres étaient déjà plutôt orientés "anomalies", comme les travaux de Hussain et al. [7] qui ont propose une taxonomie des attaques DoS, les travaux de Jung et al. [8] qui ont étudié les foules subites, ou encore le système développé par Guo et al. [6] qui pouvait détecter et protéger les réseaux des scans réseau, par exemple. Une évolution majeure dans la lutte contre les anomalies a été l'introduction de capacités de classification, i.e. la capacité de donner un nom à l'anomalie qui s'est produite, notamment en utilisant les informations des couches TCP/IP. Des contributions importantes ont été celles de Kim et al. [9], Estan et al. [5] et Lakhina et al. [11]. A noter néanmoins que ces outils de détection ne fonctionnent qu'à partir de traces de trafic enregistrées. La classification d'anomalies n'en reste pas moins, encore aujourd'hui, un sujet trop peu abordé, et la conception d'une méthode de diagnostic des anomalies reste un problème ouvert. En particulier, aucune des approches décrites dans la littérature n'ont exploité complètement la richesse des attributs IP pour fournir des informations précises permettant d'identifier les responsables d'une anomalie. On peut pour cela mettre en cause les techniques mathématiques complexes (statistiques, traitement du signal) qui sont utilisées dans la plupart des cas pour la détection d'anomalies et qui rendent improbable un retour des espaces fréquentiel ou entropique vers des

attributs réseau facilement compréhensibles par les techniciens qui opèrent les réseaux.

La suite de cet article est organisée comme suit : la seconde partie donne les principes de la détection d'anomalies avec NADA. La partie 3 aborde l'aspect classification des anomalies. Pour cela, elle définit une signature pour les anomalies et montre des extraits de la base d'anomalies qui en découle. Puis elle décrit comment se fait la classification des anomalies. La partie 4, à la suite de la détection et de la classification des anomalies, décrit comment ces dernières sont alors trivialement identifiées. La partie 5 présente la validation de NADA à partir d'une base de traces de trafics réels dans lesquels des anomalies contrôlées ont été introduites. Enfin, la partie 6 conclut cet article en décrivant les améliorations qui sont en cours sur NADA.

2. L'algorithme de détection d'anomalies NADA

NADA a été défini comme un algorithme multi-échelles (il n'existe pas une échelle de temps unique qui permette de détecter toutes les anomalies – chacune agit avec ses propres caractéristiques temporelles), multi-critères (les anomalies n'affectent pas les attributs du trafic réseau de la même façon) et avec plusieurs niveaux d'agrégation (les différents types d'anomalies se perçoivent mieux à différents niveaux d'agrégation du trafic). Pour illustrer sur un cas concret les principes de conception de NADA, l'analyse du trafic entrant se fait, dans cet article, au niveau TCP/IP, en utilisant des informations sur les paquets et les flots, les flots étant définis selon Claffy et al [3]. La détection des anomalies dans NADA se fait en deux phases. La première a pour objectif de détecter les anomalies, alors que la seconde cible les flots anormaux pour les caractériser.

La première étape de NADA est un processus récursif qui commence par des mécanismes de détection peu coûteux (en temps) pour savoir si des variations significatives sur des séries temporelles de trafic fortement agrégé se produisent. Aux étapes successives, NADA augmente la quantité d'information à traiter en réduisant le niveau d'agrégation, et réalise alors des analyses plus fines.

NADA considère qu'une anomalie est responsable d'une variation sur au moins un des critères considérés, à au moins une échelle de temps et à un niveau d'agrégation donnés. Les échelles de temps Δ choisies pour analyser le trafic vont de quelques microsecondes jusqu'à plusieurs heures. Ces valeurs dépendent des objectifs fixés pour le temps de réponse de l'algorithme, mais aussi du type d'anomalie : certaines anomalies mineures quant à leur impact sur le réseau sont plus facilement détectables en considérant des fenêtres temporelles courtes. Les niveaux d'agrégation des adresses IP peuvent aller de la taille de préfixe /0 (tous les paquets sont considérés dans un flot unique pour l'analyse) jusqu'à /32 (chaque flot composé des paquets provenant ou allant vers une adresse IP unique sont analysés individuellement), en passant par les tailles de préfixes intermédiaires (pour lesquelles les flots contenant tous les

paquets venant ou à destination des adresses ayant un préfixe donné sont analysées individuellement). C'est en découpant ainsi à chaque étape le trafic en tranches tomographiques de plus en plus fines que l'on parvient à détecter les anomalies les plus massives et celles qui sont plus discrètes. Ainsi, dans le processus récursif, à chaque étape un nouvel ensemble de séries temporelles correspondant aux critères considérés est calculé pour une taille de fenêtre et une taille de préfixe donnée. A chaque étape ces séries sont analysées. De plus, pour chaque niveau d'agrégation /n des adresses, toutes les fenêtres temporelles sont étudiées. Et tous les préfixes d'adresses qui apparaissent dans le bloc de trafic considéré sont analysés. Dans les faits, et pour réduire le temps d'exécution en évitant des redondances entre des tailles de préfixes proches, on ne considère que quelques niveaux d'agrégation (e.g. /8, /16, /24, /32, ainsi que quelques valeurs intermédiaires si cela apparaît nécessaire au cours de l'une des itérations de l'algorithme).

Les variations sur les différentes séries temporelles représentant les différents critères (octets, paquets, flots) sont détectées en utilisant l'équation suivante : X est la série temporelle directement issue du trafic entrant. P est une série obtenue en faisant la différence entre toutes les valeurs consécutives de X deux à deux. Considérer P au lieu de X est un des points forts de cet algorithme : en effet, cela permet de ne détecter que les anomalies induisant une variation brusque et significative, et de ne pas être sensible aux variations normales du trafic. Ces variations importantes sur P ont été baptisées « deltoïdes » par Cormode et al. [4] qui les a aussi utilisés pour détecter les changements brusques dans le volume de trafic transitant sur un lien.

$$X = \{x_1, x_2, ..., x_n\}, x_i = \{\#\ paquets\ |\#\ octets\ |\#\ flots\}/\Delta$$
$$P = \{p_1, p_2, ..., p_{n-1}\}, p_i = x_{i+1} - x_i$$
$$\begin{cases} pi \geq E(p) + k\sigma \rightarrow anomalie \\ pi < E(p) + k\sigma \rightarrow normal \end{cases}$$

La moyenne E(p) et l'écart-type σ sont calculés pour chaque série temporelle et utilisés pour définir un seuil. Chaque dépassement de ce seuil, est interprété comme la présence d'une anomalie à cet instant. Suivant la valeur de *k*, ce filtrage peut être plus ou moins fin, selon l'intensité des anomalies que l'on cherche. Naturellement, des grandes valeurs de *k* conduisent à un plus grand nombre de faux négatifs, alors que de petites valeurs conduisent à un plus grand nombre de faux positifs.

Le résultat de cette première phase conduit à une liste de flots qui ont présenté une variation importante à un moment donné, et induisent donc potentiellement une anomalie.

3. Classification d'anomalies à partir de signatures

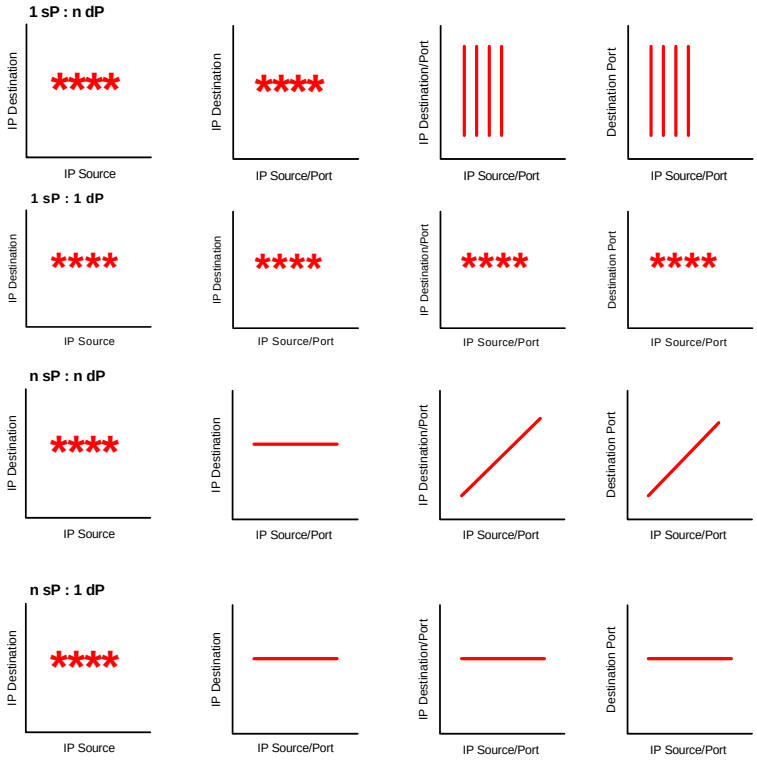

Figure 1. Exemples de s*ignatures théoriques d'attaques de DDoS de plusieurs sources IP vers 1 destination IP. Les différentes signatures représentent les cas pour lesquels un seul (resp. plusieurs) port(s) source est (resp. sont) utilisé(s) – 1 (resp. n) sP – et un (resp. plusieurs) port(s) destination est (resp. sont) ciblé(s) – 1 (resp. n) dP.*

La phase de détection précédente ayant permis d'isoler les caractéristiques des anomalies en fonction de critères relatifs aux couches TCP/IP, nous avons cherché à définir pour chaque type d'anomalies une signature unique qui leur est propre. En ne considérant que les nombres d'octets, de paquets et de flots, il est possible de définir un type d'anomalies à partir de 4 graphes bidimensionnels[1]. Sur chacun de ces graphes, chaque flot qui a été détecté comme comportant une anomalie est représenté par un point. Ces 4 graphes ont respectivement en abscisse et en ordonnée :

– l'adresse IP source et l'adresse IP destination ;

[1] Le choix de graphes bidimensionnels s'est imposé pour des raisons de lisibilité – avec des courbes en 3D les effets de perspectives étaient parfois trompeurs

– la concaténation de l'adresse et du port source, et l'adresse IP destination ;

– la concaténation de l'adresse et du port source, et la concaténation de l'adresse et du port destination ;

– la concaténation de l'adresse et du port source, et le port destination.

Ainsi, dans ces graphes, les différents types d'anomalies font apparaître des points ou des lignes : ce sont leurs signatures. A titre d'exemples, les figures 1 et 2 représentent respectivement des signatures pour des attaques DoS distribuées (DDoS) et des scans de réseaux. Prenons l'exemple d'une attaque DDoS de type *n*S, *n*P : *1*D, *n*P : plusieurs sources IP utilisant plusieurs Ports attaquent 1 destination IP sur plusieurs Ports (3ème signature sur la figure 1).

Sur le premier graphe, chaque point dénote les différentes sources (adresses IP) de l'attaque vers une seule et unique destination (au niveau IP). Lorsque l'on considère l'agrégation de l'adresse IP source avec le port source (second graphe), on voit pour une unique adresse destination une ligne droite qui montre que chaque source utilise différents ports lorsqu'il attaque la cible. Sur le troisième graphe, lorsqu'on agrège les numéros de port source et destination respectivement aux adresses IP source et destination il apparaît par la ligne droite en diagonale que plusieurs ports destination sont ciblés par l'attaque. Ceci est confirmé par le quatrième graphique par la ligne droite en diagonale montrant que plusieurs ports destination sont ciblés par l'attaque. Cette droite oblique met également en évidence une relation entre port source et port destination pour les paquets de l'attaque.

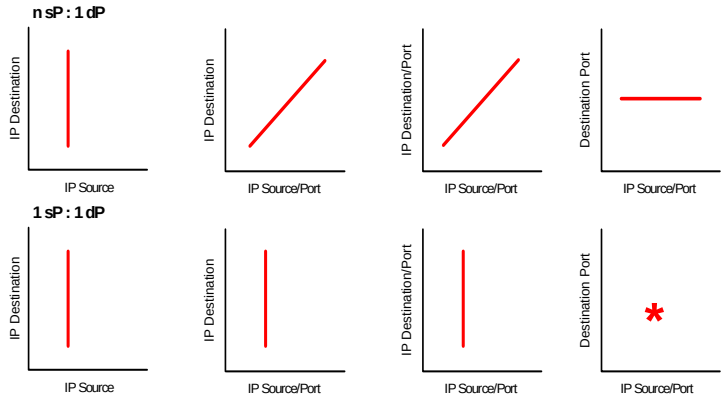

Figure 2. Exemples de s*ignatures théoriques pour des scans de réseaux (cf. figure 1 pour la légende)*

Il apparaît ainsi qu'avec ces représentations graphiques des signatures d'anomalies, il est très aisé d'analyser les caractéristiques des flots qui composent le trafic anormal et ainsi de classifier l'anomalie. Par manque de

place, les autres exemples donnés sur les figures 1 et 2 ne seront pas détaillés, de même que la base de signature complète ne peut pas être intégralement énumérée dans cet article.

Fort de cette base de signature caractérisant de manière unique un type d'anomalies, lorsqu'il détecte une anomalie, NADA la signale à l'opérateur et lui trace les 4 graphes de la signature. A partir de là, l'opérateur peut en quelques secondes déterminer quel type d'anomalie est en train de se produire, soit en analysant rapidement les 4 graphes, soit en se référant à la liste des signatures contenues dans la base et qui sont mises à sa disposition. A noter également que si une signature n'est pas encore référencée dans la base, parce qu'elle correspond à une attaque 0d (une attaque encore inconnue) par exemple, l'opérateur peut facilement comprendre son principe de fonctionnement en analysant les 4 graphes de la signature, en déduire la procédure corrective à appliquer, et enrichir la base de signatures.

4. Identification des anomalies

Etant donnés les informations continues dans la signature des anomalies détectées, l'identification des flots et paquets fautifs est immédiate. Toutefois, pour faciliter le travail de l'opérateur, celui-ci reçoit les informations requises en clair. Il n'a plus alors qu'à lancer les procédures correctives. Les données transmises en clair à l'opérateur sont la fenêtre temporelle dans laquelle l'anomalie est détectée, la liste des adresses IP et des numéros de port impliqués dans l'anomalie et la liste des adresses IP et des numéros de port ciblés

5. Validation

5.1. *Principe*

Valider NADA consiste à évaluer sa capacité à bien détecter les anomalies et à bien les classifier (l'identification étant triviale elle est de facto validée). La difficulté de la validation de détecteurs d'anomalies réside dans l'établissement des qualités de leurs performances statistiques. Les anomalies sont rarement annoncées a priori, de sorte qu'il est difficile de disposer d'un ensemble de traces, contenant des anomalies de types et caractéristiques connues, pouvant servir de base de données pour l'étalonnage des procédures de détection. Pour pallier cette difficulté, nous avons choisi de réaliser nous-mêmes un ensemble d'anomalies dont nous faisons varier les paramètres de manière contrôlée et reproductible. A partir de cette base de données que nous avons constituée dans le cadre du projet MétroSec [2], nous établissons les performances statistiques (probabilité de détections correctes versus probabilité de fausses alarmes) des procédures de détection proposées.

5.2. *La base de traces utilisées*

Cette base de traces de trafic contient plusieurs types d'anomalies, légitimes et illégitimes. Les anomalies ont été générées avec des intensités différentes pour évaluer la capacité de NADA à détecter des anomalies de faible intensité. A l'heure actuelle, notre base comporte 42 traces. Leurs caractéristiques et leur mode de production sont décrits dans [2]. Cette base contient notamment des traces d'**Attaques de DDoS** réalisées en utilisant différents types d'outils d'inondation (Iperf, Trinoo, TFN2k, etc.) qui ont permis de créer divers types d'anomalies de type « flooding ». Elle contient aussi des **Foules subites** sur un serveur Web.

5.3. *Validation des capacités de détection de NADA*

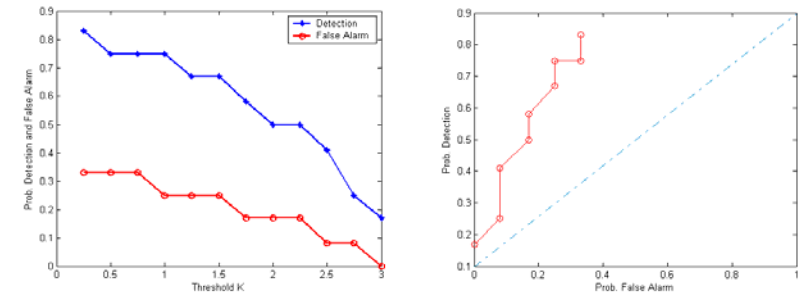

Figure 3. Performances statistiques de NADA pour la détection d'anomalies. A gauche, *Probabilité de détection PD = f(K) et probabilité de fausses alarmes PF = g(K). A droite, probabilité de détection PD en fonction de la probabilité de fausses alarmes.*

L'objectif de cette validation a consisté à confronter NADA aux traces de trafic avec anomalies contenues dans notre base. Toutefois, NADA possède un paramètre important qui influe sur les performances de l'algorithme : le paramètre de seuillage k. Pour cela, nous avons fait varier k dans l'intervalle]0, 3]. Pour chaque valeur de k, NADA a été évalué pour toutes les traces de notre base. La courbe à gauche de la figure 3 représente la probabilité de bien détecter une attaque et la probabilité de fausse alarme en fonction du paramètre k. Il montre en particulier que le taux de bonne détection dépasse toujours largement le taux de fausses alarmes. Il indique également que la valeur k = 2est une bonne valeur pour configurer NADA.

La courbe à droite de la figure 3 (obtenue par lecture sur le graphe de gauche de cette même figure) représente la probabilité de bonne détection en

fonction de la probabilité de fausse alarme. Le point idéal sur ce graphe est le coin haut à gauche pour lequel toutes les attaques sont détectées sans aucune fausse alarme. Le cas pire est la diagonale qui correspond à des résultats de détections aléatoires. Cette courbe montre que NADA est très efficace.

5.4. *Evaluation du principe de classification*

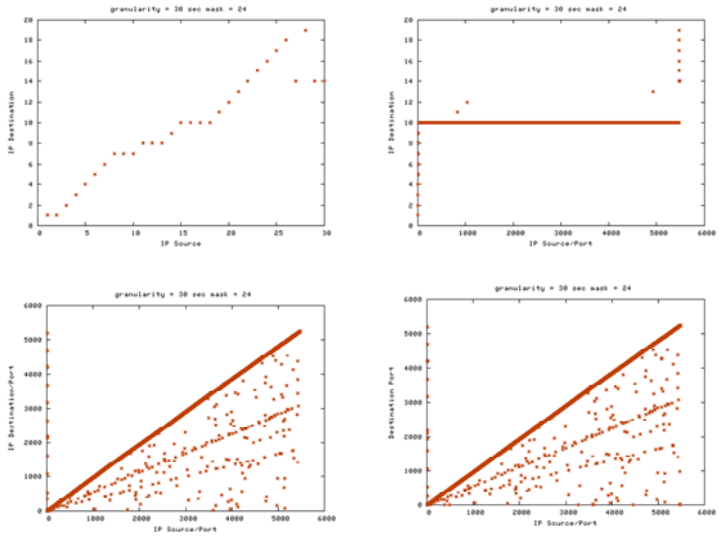

Figure 4. *Exemple d'attaque DoS par inondation*

A l'heure actuelle, NADA se limite à tracer les graphes de signatures lorsqu'il a détecté une anomalie. C'est donc l'opérateur qui est en charge de faire la classification par comparaison visuelle entre la signature de l'anomalie détectée par NADA et la base de signatures existante. De fait, cette partie ne présente pas une validation formelle, mais montre juste que les signatures connues (notamment celles représentées sur les figures 1 et 2) apparaissent clairement sur les signatures d'anomalies produites par NADA.

La figure 4 montre les 4 graphes produits par NADA lors de la détection d'une anomalie. Malgré le bruit sur les graphes (il existe des points qui ne sont pas sur les points ou les droites principaux de ces graphes), on voit clairement apparaître sur cette figure la signature d'une attaque de DDoS (la troisième de la figure 1 qui correspond à une attaque de plusieurs sources vers une cible et ce en utilisant des ports d'émission et de réception variés).

De même, malgré le bruit, il apparaît clairement sur la figure 5 la signature d'un scan réseau (cette signature est similaire à la première de la figure 2).

Ces deux exemples montrent clairement que la méthode de classification fonctionne. Les anomalies ont pu être facilement et rapidement reconnues visuellement. Cela a été le cas pour toutes les traces contenues dans notre base, mais également sur des traces inconnues qui se sont révélées contenir un certain nombre d'anomalies. Par manque de place, ce travail n'est pas rapporté dans cet article.

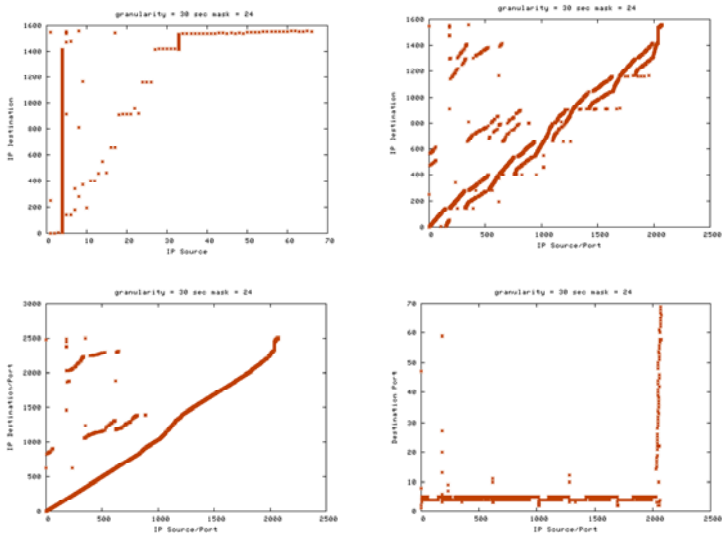

Figure 5. *Exemple de scan de réseaux*

6. Conclusion

Cet article a proposé une méthode de détection de classification et d'identification originale des anomalies du trafic. Ce travail bénéficie de nos connaissances sur les caractéristiques du trafic acquises depuis le début des années 2000 dans le cadre des projets Metropolis et MétroSec. Il en résulte un outil – NADA – très efficace qui repose sur une analyse multi-critère, multi-échelle et considérant divers niveaux d'agrégation du trafic entrant, et ce, à la volée. Une de nos contributions pour mener ce travail à bien a consisté à produire une base de traces de trafic contenant des anomalies connues et documentées, ainsi qu'une méthodologie de validation et d'évaluation statistique pour des outils de détection d'anomalies (ou d'attaques DoS).

Au niveau de la classification des anomalies, aujourd'hui, NADA laisse à l'opérateur la responsabilité de lire sur les 4 graphes de la signature le type d'anomalie qui est en train de se produire. Cette solution présente l'avantage de permettre d'analyser et de traiter ensuite de façon adéquate des anomalies inconnues. Toutefois, cela interdit de pouvoir lancer les procédures de correction automatiquement.

Aussi, pour transformer NADA en NAPA (pour assurer la **P**rotection du réseau en plus de la **D**étection des anomalies), nous allons concevoir un système de reconnaissance automatique du type d'anomalie. A priori, nous utiliserons pour cela des algorithmes de clustering permettant de mettre en évidence les points ou les lignes de fortes densités dans les graphes. Il sera alors possible de faire une comparaison automatique avec les signatures connues et de démarrer ensuite les corrections appropriées (éliminer les paquets d'une attaque par exemple, ou mieux gérer les ressources en cas de foule subite).

12. Bibliographie

[1] P. Barford, J. Kline, D. Plonka, and A. Ron. A Signal Analysis of Network Traffic Anomalies. In *Internet Measurement Workshop*, Marseille, November 2002.

[2] 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", à paraître dans *les annales des télécoms*

[3] K. Claffy, H. Braun, and G. Polyzos. A Parameterizable Methodology for Internet Traffic Flow Profiling. In *Selected Areas in Communications – IEEE Journal*, vol. 13, p 1481-1494, October 1985.

[4] G. Cormode and S. Muthukrishnan. What's New: Finding Significant Differences in Network Data Streams. In *IEEE/ACM Transactions on Networking*, vol.13, n.6, pp. 1219-1232, 2005.

[5] C. Estan, S. Savage and G. Varghese. Automatically Inferring Patterns of Resource Consumption in Network Traffic. *In ACM SIGCOMM'03*, Karlsruhe – Germany, August 2003.

[6] Guo, X., Qian, D., Liu, M., Zhang, R., and Xu, B.: Detection and Protection Against Network Scanning: IEDP. In Proceedings of ICCNMC'01, Beijing (2001)

[7] Hussain, A., Heidemann, J., and Papadopoulos, C.: A Framework for Classifying Denial of Service Attacks. In ACM SIGCOMM, Germany (2003)

[8] Jung, J., and Krishnamurthy, B., and Rabinovich, M.: Flash Crowds and Denial of Service Attacks: Characterization and Implications for CDNs and Web Sites. In WWW, Hawaii (2002)

[9] S. Kim, A. Reddy, and M. Vannucci. Detecting Traffic Anomalies through Aggregate Analysis of Packet Header Data. *In Networking 2004*, Athens – Greece, May 2004.

[10] B. Krishnamurthy, S. Sen, Y. Zhang and Y. Chen. Sketch-Based Change Detection: Methods, Evaluation, and Applications. In *Proceedings of IMC'03*, Miami – USA, October 2003.

[11] A. Lakhina, M. Crovella and C. Diot. Mining Anomalies Using Traffic Feature Distributions. *In ACM SIGCOMM*, Philadelphia – USA, August 2005.