RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE

MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE Dr. TAHAR MOULAY SAIDA

FACULTE : TECHNOLOGIE

DEPARTEMENT : INFORMATIQUE

MÉMOIRE DE MASTER

Option : spécialité master

Utilisation de la distance de Kullback-Leibler pour la classification des menaces sécuritaires

---

Présenté par :

Mahmoudi Fatima Zohra

Slimani Chahrazed

Encadré par :

Dr. Benamara Djillali

Année Universitaire 2020-2021

Using Kullback – Leibler Distance for Security Threat Classification

Abstract — The increasing number of network attacks causes growing problems for network operators and users. Thus, detecting anomalous traffic is of primary interest in IP networks management. In this memoir, we address the problem considering a method based on PCA for detecting network anomalies. In more detail, this memoir presents a new technique that extends the state of the art in PCA-based anomaly detection. Indeed, by means of multi-scale analysis and Kullback–Leibler divergence, we are able to obtain great improvements with respect to the performance of the 'classical' approach. Moreover, we also introduce a method for identifying the flows responsible for an anomaly detected at the aggregated level.

Keywords : Anomaly identification system, Principal component analysis, Kullback–Leibler divergence.

Utilisation de la distance de Kullback–Leibler pour la classification des menaces sécuritaires

Résumé — Le nombre croissant d'attaques de réseau pose des problèmes croissants aux opérateurs de réseau et aux utilisateurs. Ainsi, la détection de trafic anormal est d'un intérêt primordial dans la gestion des réseaux IP. Dans ce mémoire, nous abordons le problème en considérant une méthode basée sur l'ACP pour détecter les anomalies du réseau. Plus en détail, ce mémoire présente une nouvelle technique qui étend l'état de l'art dans la détection d'anomalies basée sur l'ACP. En effet, grâce à l'analyse multi-échelle et à la divergence de Kullback-Leibler, nous sommes en mesure d'obtenir de grandes améliorations par rapport aux performances de l'approche « classique ». Par ailleurs, nous introduisons également une méthode d'identification des flux responsables d'une anomalie détectée au niveau agrégé.

Mots clés : Système d'identification des anomalies, Analyse en composantes principales, divergence de Kullback–Leibler.

استخدام شُللبَك - شِبلر ِستَذ لتصنيف التهديدات الأمنية

الملخص ---

يتسبب العدد المتزايد لهجمات الشبكة في مشاكل متزايدة لمشغلي الشبكات والمستخدمين. وبالتالي ، فإن اكتشاف الحركة الشاذة له أهمية أساسية في إدارة شبكات بروتوكول الإنترنت. في هذه المذكرات ، نعالج المشكلة بالنظر إلى طريقة تعتمد على صرا لاكتشاف الحالات الشاذة في الشبكة. بمزيد من التفصيل ، تقدم هذه المذكرات تقنية جديدة توسع من أحدث التقنيات في اكتشاف الشذوذ المستند إلى صرا. في الواقع ، من خلال التحليل متعدد المقاييس وتباعد شُللبَك-شِبلر ، يمكننا الحصول على تحسينات كبيرة فيما يتعلق بأداء النهج الكلاسيكي. علاوة على ذلك ، نقدم أيضًا طريقة لتحديد التدفقات المسؤولة عن حالة شاذة تم اكتشافها على المستوى المجمع.

نظام تحديد الشذوذ ، تحليل المكون الرئيسي ، اختلاف كولباك - ليبلر.

ات المفتاحية :

REMERCIEMENT

Tout d'abord nous remercions ALLAH le tout puissant d'avoir nous donner le courage, la volonté et la patience de mener à terme le présent travail.

La réalisation de ce mémoire a été possible grâce au concours de plusieurs personnes à qui nous voudrons témoigner toute nos gratitudes.

Nous voudrons adresser toutes nos reconnaissances au directeur de ce mémoire, monsieur BENAMARA Djilleli, pour sa patience, sa disponibilité et surtout ses judicieux conseils, qui ont contribué à alimenter notre réflexion.

Nous désirons aussi remercier les professeurs de l'université de MOULAY Taher de saida , qui nous ont fourni les outils nécessaires à la réussite de nos études universitaires.

Enfin, nous tenons à témoigner toute nos gratitudes aux membres de jury pour leurs conseils en évaluons notre travail

TABLE DES MATIÈRES

Liste des tableaux

Table des figures

CHAPITRE 1

INTRODUCTION GÉNÉRALE

Au commencement, c'était de simples réseaux conçus spécifiquement pour le partage de ressources particulières telle l'imprimante, parmi un certain nombre d'ordinateurs, qui à leur tour peuvent inter-échanger des informations. Les réseaux de l'époque, avaient a assuré ces services de base, au sein d'une organisation unique, telle que les universités, les compagnies et les bases militaires. . . etc. Les réseaux connectaient des utilisateurs de la même organisation, d'où la nécessité de sécuriser les transmissions n'était pas sensible.

La venue d'Internet, et son utilisation universellement émergeante, et sa politique démocratique permettant à toute machine d'être connectée au réseau, et à toute personne d'en bénéficier ainsi que de proposer ses propres services. Cette vision d'échange libre de l'information impliquant un accès aussi libre, nous laisse à réfléchir sur la nature de l'information elle-même, sur plusieurs aspects, l'un des plus importants et surement crucial est le degré de confidentialité de cette information. Une confidentialité qui devrait être garantie avec un contrôle rigoureux des accès à cette information, et les opérations engagées. Afin d'empêcher les accès non-autorisés ou les opérations nondéfinies, une politique de sécurité est recommandée, pour cette fin, plusieurs stratégies et protocoles sécuritaires ont été élaborés. Parmi les solutions proposées, la mise en place des systèmes de détection d'intrusions est devenue nécessaire. Cependant, la complexité croissante des attaques nécessite une amélioration de telles techniques de détections.

Les activités d'intrusion peuvent s'exercer au niveau du réseau comme au niveau des machines hôtes, et donc les méthodes et outils de défense devront être engagés à

ces mêmes deux niveaux précédents, d'où on parle de systèmes de détection d'intrusion basés réseaux et ceux basés hôtes. Notre projet traite les systèmes de détection basés hôte, qui représente en fait, une couche de défense.

L'IDS est de deux natures, celui qui reconnait l'attaque par un modèle d'action et, celui qui reconnait le comportement de l'activité intrusive. Un comportement est assimilé progressivement, est donc demande des techniques possédant la capacité d'apprentissage, d'où l'implication des algorithmes d'apprentissage dans le processus du design de l'IDS.

Nous étudions dans notre projet, la méthode de l'analyse des composent principales basée sur une distance de Kullback Liebler.

1.1 La structure du mémoire

Afin de répondre à cet objectif, ce mémoire est structuré de la façon suivante :

Le deuxième chapitre est consacré aux notions mathématiques préliminaires. Nous allons nous focalisé sur les notions mathématiques directement liées au calcul des méthodes utilisées pour l'objectif de ce mémoire. A savoir la divergence Kullback-Leiber et l'analyse en composante principales.

Le troisième chapitre est une présentation de la notion des systèmes de détection d'intrusions, avant de donner les définitions relatives à cette notion, nous allons illustrer des aspects théoriques comparatifs .Ensuite une classification des IDS sera abordée. Pour entamer ensuite les mesures partielles qui peuvent être utilisées pour tester l'efficacité des IDS.

Le quatrième chapitre est dédié à la présentation des concepts relatifs à la classification non supervisé avec la méthode de la divergence Kullback-Leiber et l'analyse en composante principales.

Nous conclurons ce mémoire par une conclusion générale et des perspectives.

CHAPITRE 2

PRÉLIMINAIRE

2.1 Introduction

Ce chapitre est consacré aux notions mathématiques péliminaires. Nous allons nous focalisé sur les notions mathématiques directement liées au calcul des méthodes utilisées pour l'objectif de ce mémoire. A savoir la divergence Kullback-Leiber et l'analyse en composante principales.

2.2 Préliminaire

2.2.1 Espaces de probabilités

Un espace de probabilité (Ω, P) est constitué de :

— Ω un ensemble (l'ensemble des résultats d'une expérience aléatoire)
— P une probabilité sur Ω.

Un élément $w \in \Omega$ est appelé une réalisation, c'est un résultat possible d'une expérience aléatoire. Un sous-ensemble $A \subset \Omega$ est appelé un événement. C'est un ensemble de réalisations L'ensemble des événements est donc l'ensemble P(Ω) des parties (ou sous-ensembles) de Ω.

2.2.1.1 évènements

Un évènement est un sous-ensemble de Ω, ou une réunion d'évènements élémentaires. On dit qu'un évènement est réalisé si un des évènements élémentaires qui le constitue est réalisé. Les évènements sont des ensembles, représentes souvent par des lettres capitales.

2.2.1.2 Probabilite

Une probabilité sur Ω est une application $P : P(\Omega) \rightarrow [0,1]$, définie sur les événements, telle que : P(Ω) = 1, pour toute suite $(A_n)_n$ d'événements disjoints deux à deux :

$$P(\cup_n A_n) = \sum_n P(A_n) \tag{2.1}$$

Si un événement A vérifie P(A) = 0, on dit que A est négligeable ; et si P(A) = 1, on dit que A est presque sûr, ou que A a lieu presque sûrement, abrégé « p.s. »

2.2.2 Propriétés des probabilités

1. P(∅) = 0
2. Pour tout événement A, $P(A)^c$ = 1 – P(A)
3. Si $A \subset B, alors P(A) \leq P(B)$
4. Pour tous événements A et B, P(A ∪ B) = P(A) + P(B) – P(A ∩ B)
5. Pour toute suite croissante $(A_n)_n$ d'événements (c'est-à-dire An⊂ $A_n + 1$ pour tout n)

   $$P(\cup_n A_n) = \lim_n \uparrow P(A_n) \tag{2.2}$$

6. Pour toute suite décroissante $(A_n)_n$ d'événements (c'est-à-dire $A_{n+1} \subset A_n$ pour tout n),

   $$P(\cap_n A_n) = \lim_n \downarrow P(A_n) \tag{2.3}$$

2.2.3 Probabilités conditionnelles

On considéré l'espace probabilisé (Ω,A,P) et un événement particulier B de A tel que P(B)>0 .la connaissance de la réalisation de B modifie la probabilité réalisation d'un

événement élémentaire , puisque l'ensemble des résultats possibles est devenu B et non plus Ω.

$$P(A|B) = \frac{P(A \cap B)}{P(B)} \tag{2.4}$$

2.2.4 Variables aléatoires

Soit (Ω,P) un espace de probabilité. Une variable aléatoire est une application X : $\Omega \rightarrow R$ Soit X une variable aléatoire. La loi de X est la probabilité PX sur R définie par : pour tout B ⊂ R, PX peut aussi être vue comme une probabilité sur X(Ω), ensemble des valeurs prises par X, aussi appelé support de PX. On note parfois X ≈ $PX pour indiquer que X suit la loi PX. La seconde égalité est une nouvelle notation : on note X \in$ B l'événement formé des éventualités w pour lesquelles X(w) ∈ B, et on abrège P(X ∈ B) = P(X ∈ B).

2.2.5 Variance d'une variable aléatoire

Soit X une variable aléatoire. La variance de X est l'espérance des carrés des écarts de X à sa moyenne :

$$Var(X) = E\left[(X - E(X))^2\right] \geq 0 \tag{2.5}$$

L'écart type de X est :

$$\sigma(X) = \sqrt{Var(X)} \tag{2.6}$$

À la différence de la variance, l'écart type σ(X) est homogène à X : si par exemple X est une distance, alors σ(X) est une distance aussi.

2.2.6 Densité de probabilité

Pour une variable continue, on travaille la plupart du temps avec un ensemble de définition sur les réels. La probabilité ponctuelle P(X = x) = f(x) est la fonction de densité. La fonction de répartition F(x) = P(X < x) est définie par :

$$F(x) = \int_{-\infty}^{x} F(t)dt \tag{2.7}$$

La densité de probabilité d'une variable aléatoire continue est la dérivée première par rapport à x de la fonction de répartition. Cette dérivée prend le nom de fonction de densité, notée

$$f(x) = \frac{dF(x)}{dx} \tag{2.8}$$

Elle est équivalente à P(X = x) dans le cas des variables discrètes.

Pour calculer la probabilité P(a ≤ X < b) dans le cas d'une variable continue, le calcul est le suivant :

$$\int_a^b F(x)dx \tag{2.9}$$

Développons cette expression

$$P(a \leq X < b) = P(X < b) - P(X < a) = \int_{-\infty}^{b} f(t)dt - \int_{-\infty}^{a} f(t)dt = F(b) - F(a) \tag{2.10}$$

Graphiquement, cela se traduit par la surface comprise entre a et b en dessous de la courbe de densité. Propriété : De la même façon que pi ≥ 0 et

2.2.7 Probabilité mixte

Il existe des lois de probabilité qui ne sont ni discrètes, ni absolument continues, ni singulières, elles sont parfois appelées lois mixtes. La loi de probabilité réelle suivante est un exemple de loi mixte obtenue en mélangeant une loi discrète, définie par ses atomes $\{x_k, k \in \mathbb{N}\}$ et sa fonction de masse P, avec une loi absolument continue de densité F

$$P(dx) = \alpha f(x)dx + (1-\alpha) \sum_{k \in \mathbb{N}} p(x_k)\delta_{x_k}(dx) \tag{2.11}$$

2.2.8 Entropie

L'entropie est un moyen de mesurer l'incertitude/le caractère aléatoire d'une variable aléatoire X. En d'autres termes, l'entropie mesure la quantité d'informations dans une variable aléatoire. Il est normalement mesuré en bits.

(2.12) $$H(X) = -\sum_{x} p_x \log_2(p_x)$$

Entropie conjointe : L'entropie conjointe d'une paire de variables aléatoires discrètes X, Y ≈ p (x, y) est la quantité d'informations nécessaire en moyenne pour spécifier leurs deux valeurs.

(2.13) $$H(X,Y) = -\sum_{x,y} p_{x,y} \log_2(p_{x,y})$$

2.3 la Décomposition en valeurs singulières (svd)

L'idée essentielle de la SVD est de décomposer la matrice de données en trois matrices simples : deux orthogonales et une diagonale. Du fait qu'elle produise une estimation aux moindres carrés de la matrice de données de même dimension et d'un rang inférieur, elle est équivalente à PCA, et importante autant qu'elle. L'un des avantages de la SVD est son pouvoir de réduction des données après leur blanchissement. En effet, cette technique fournit une description plus compacte des données contenues dans une matrice, exprimée par les premiers modes statistiques. Elle peut être considérée comme une méthode permettant de construire une partition de la variance d'une base de données, c'est à dire qu'elle fournit la base orthogonale qui maxime la variance au sens des moindres carrés. La décomposition en valeurs singulières utilise la décomposition en valeur propre d'une matrice semi définie positive obtenue par la multiplication d'une matrice par sa transposé, pour dériver une décomposition similaire applicable à toutes les matrices rectangulaires composées de nombres réels

2.4 série temporelle

Une série temporelle, ou série chronologique, est une suite de valeurs numériques représentant l'évolution d'une quantité spécifique au cours du temps. De telles suites de variables aléatoires peuvent être exprimées mathématiquement afin d'en analyser le comportement, généralement pour comprendre son évolution passée et pour en prévoir le comportement futur. Une telle transposition mathématique utilise le plus souvent des concepts de probabilités et de statistique.

2.5 Analyse en composantes principales

Les composants principaux d'une collection de points dans un espace de coordonnées réel sont une séquence de p vecteurs unitaires, où le ième vecteur est la direction d'une ligne qui correspond le mieux aux données tout en étant orthogonal aux premiers i-1 vecteurs. Ici, une ligne la mieux ajustée est définie comme une ligne qui minimise la distance quadratique moyenne des points à la ligne. Ces directions constituent une base orthonormée dans laquelle différentes dimensions individuelles des données sont linéairement décorrélées. L'analyse en composantes principales (ACP) est le processus consistant à calculer les composantes principales et à les utiliser pour effectuer un changement de base sur les données, en n'utilisant parfois que les premières composantes principales et en ignorant le reste.

L'ACP est utilisée dans l'analyse exploratoire des données et pour la création de modèles prédictifs. Il est couramment utilisé pour la réduction de la dimensionnalité en projetant chaque point de données uniquement sur les premiers composants principaux pour obtenir des données de dimension inférieure tout en préservant autant de variation des données que possible. La première composante principale peut être définie de manière équivalente comme une direction qui maximise la variance des données projetées. La i-ième composante principale peut être considérée comme une direction orthogonale aux premières i-1 composantes principales qui maximise la variance des données projetées.

De l'un ou l'autre objectif, il peut être montré que les composants principaux sont des vecteurs propres de la matrice de covariance des données. Ainsi, les composantes principales sont souvent calculées par décomposition propre de la matrice de covariance des données ou décomposition en valeurs singulières de la matrice de données. L'ACP est la plus simple des véritables analyses multivariées basées sur des vecteurs propres et est étroitement liée à l'analyse factorielle. L'analyse factorielle intègre généralement des hypothèses plus spécifiques au domaine concernant la structure sous-jacente et résout les vecteurs propres d'une matrice légèrement différente. L'ACP est également liée à l'analyse de corrélation canonique (ACC). CCA définit des systèmes de coordonnées qui décrivent de manière optimale la covariance croisée entre deux ensembles de données tandis que PCA définit un nouveau système de coordonnées orthogonales qui décrit de manière optimale la variance dans un seul ensemble de données≈[1, 2, 9, 16]. Des variantes robustes et basées sur la norme L1 de l'ACP standard ont également été proposées≈[2, 10, 15].

2.5.1 Notations

Soit p variables statistiques réelles $X^j (j = 1,\ldots,p)$ observées sur n individus $i (i = 1,\ldots,n)$ affectés des poids w_i :

$$\forall i = 1,\ldots,n : w_i > 0 et \sum_{i=1}^{n} w_i = 1; \tag{2.14}$$

$$\forall i = 1,\ldots,n : x_i^j = X^j(i), \tag{2.15}$$

mesure de X^j sur le $i^{ème}$ individu.

Ces mesures sont regroupées dans une matrice X d'ordre (n x p).

— À chaque individu i est associé le vecteur x_i contenant la i-ème ligne de X mise en colonne. C'est un élément d'un espace vectoriel noté E de dimension p ; nous choisissons $\mathbb{R}^p$ muni de la base canonique ϵ et d'une métrique de matrice M lui conférant une structure d'espace euclidien : E est isomorphe à $(\mathbb{R}^p, \epsilon, M)$; E est alors appelé espace des individus.

— À chaque variable X^j est associé le vecteur X^j contenant la j-ème colonne centrée (la moyenne de la colonne est retranchée à toute la colonne) de X. C'est un élément d'un espace vectoriel noté F de dimension n ; nous choisissons $\mathbb{R}^n$ muni de la base canonique F et d'une métrique de matrice D diagonale des poids lui conférant une structure d'espace euclidien : F est isomorphe à $(\mathbb{R}^n, F, D)$ avec $D = diag(w_1;:::;w_n)$; F est alors appelé espace des variables.

2.5.2 Objectifs

Les objectifs poursuivis par une ACP sont :

— la représentation graphique "optimale" des individus (lignes), minimisant les déformations du nuage des points, dans un sous-espace Eq de dimension q (q < p),

— la représentation graphique des variables dans un sous-espace Fq en explicitant au "mieux" les liaisons initiales entre ces variables,

— la réduction de la dimension (compression), ou approximation de X par un tableau de rang q (q < p).

Les derniers objectifs permettent d'utiliser l'ACP comme préalable à une autre technique préférant des variables orthogonales (régression linéaire) ou un nombre réduit d'entrées (réseaux neuronaux).

Des arguments de type géométrique dans la littérature francophone, ou bien de type statistique avec hypothèses de normalité dans la littérature anglosaxonne, justifient la définition de l'ACP. Nous adoptons ici une optique intermédiaire en se référant à un modèle "allégé" car ne nécessitant pas d'hypothèse "forte" sur la distribution des observations (normalité). Plus précisément, l'ACP admet des définitions équivalentes selon que l'on s'attache à la représentation des individus,

2.5.3 Graphiques

2.5.3.1 Individus

Les graphiques obtenus permettent de représenter "au mieux" les distances euclidiennes inter-individus mesurées par la métrique M.

Chaque individu i représenté par x_i est approché par sa projection M- orthogonale $\hat{z_i}^q$ sur le sous-espace $\hat{E_q}$ engendré par les q premiers vecteurs principaux $\{v^1,\ldots,v^q\}$. En notant e_i un vecteur de la base canonique de E, la coordonnée de l'individu i sur v^k est donnée par :

$$\langle X_i - \bar{X}, v^k \rangle_M = (X_i - \bar{X})' M v^k = e_i' \bar{X} M v^k = c_i^k \tag{2.16}$$

Les coordonnées de la projectionM-orthogonale de $X_i - \bar{X}$ sur $\hat{E}_q$ sont les q premiers élément de la i-ème ligne de la matrice C des composantes principales.

2.5.3.2 Variables

Les graphiques obtenus permettent de représenter "au mieux" les corrélations entre les variables (cosinus des angles) et, si celles-ci ne sont pas réduites, leurs variances (longueurs).

Une variable X^j est représentée par la projection D-orthogonale $\hat{Q}_q X^j$ sur le sous-espace F_q engendré par les q premiers axes factoriels. La coordonnée de x^j sur u^k est :

$$\langle x^j, u^k \rangle_D = x^{j'} D u^k = \frac{1}{\sqrt{\lambda_k}} x^{j'} D \bar{X} M v^k = \frac{1}{\sqrt{\lambda_k}} e^{j'} \bar{X}' D \bar{X} M v^k = \sqrt{\lambda_k v_j^k} \tag{2.17}$$

Les coordonnées de la projection D-orthogonale de x^j sur le sous-espace F_q sont les q premiers éléments de la j-ème ligne de la matrice $V\Lambda^{1/2}$.

2.5.4 Calcul de l'analyse en composantes principales

2.5.4.1 Standardisation

L'objectif de cette étape est de standardiser l'éventail des variables initiales continues afin que chacune d'entre elles contribue de manière égale à l'analyse.

Plus précisément, la raison pour laquelle il est essentiel d'effectuer une standardisation avant l'ACP, est que cette dernière est assez sensible aux variances des variables initiales. C'est-à-dire que s'il existe de grandes différences entre les plages de variables initiales, les variables avec des plages plus larges domineront sur celles avec de petites plages (par exemple, une variable comprise entre 0 et 100 dominera une variable comprise entre 0 et 1), ce qui conduira à des résultats biaisés. Ainsi, la transformation des données à des échelles comparables peut éviter ce problème.

Mathématiquement, cela peut être fait en soustrayant la moyenne et en divisant par l'écart type pour chaque valeur de chaque variable.

$$Z = \frac{valeur - moyenne}{écart - type} \tag{2.18}$$

Une fois la standardisation terminée, toutes les variables seront transformées à la même échelle.

2.5.4.2 Calcul de la matrice de covariance

Le but de cette étape est de comprendre comment les variables de l'ensemble de données d'entrée varient par rapport à la moyenne les unes par rapport aux autres, ou en d'autres termes, de voir s'il existe une relation entre elles. Parce que parfois, les variables sont fortement corrélées de telle sorte qu'elles contiennent des informations redondantes. Ainsi, afin d'identifier ces corrélations, nous calculons la matrice de covariance.

La matrice de covariance est une matrice symétrique p x p (où p est le nombre de dimensions) qui a comme entrées les covariances associées à toutes les paires possibles des variables initiales. Par exemple, pour un ensemble de données en 3 dimensions avec 3 variables x, y et z, la matrice de covariance est une matrice 3 x 3 de ceci à partir de :

$$Cov = \begin{bmatrix} \text{Cov(x,x)} & \text{Cov(x,y)} & \text{Cov(x,z)} \\ \text{Cov(y,x)} & \text{Cov(y,y)} & \text{Cov(y,z)} \\ \text{Cov(z,x)} & \text{Cov(z,y)} & \text{Cov(z,z)} \end{bmatrix}$$

Puisque la covariance d'une variable avec elle-même est sa variance (Cov(a,a)=Var(a)), dans la diagonale principale (De haut à gauche en bas à droite) nous avons en fait

les variances de chaque variable initiale. Et puisque la covariance est commutative (Cov(a,b)=Cov(b,a)), les entrées de la matrice de covariance sont symétriques par rapport à la diagonale principale, ce qui signifie que les parties triangulaires supérieure et inférieure sont égales.

2.5.4.3 Calculer les vecteurs propres et les valeurs propres de la matrice de covariance pour identifier les principales composantes

Les vecteurs propres et les valeurs propres sont les concepts d'algèbre linéaire que nous devons calculer à partir de la matrice de covariance afin de déterminer les principales composantes des données. Avant d'aborder l'explication de ces concepts, comprenons d'abord ce que nous entendons par composants principaux.

Les composantes principales sont de nouvelles variables construites sous forme de combinaisons linéaires ou de mélanges des variables initiales. Ces combinaisons sont faites de telle sorte que les nouvelles variables (c'est-à-dire les composantes principales) ne soient pas corrélées et que la plupart des informations contenues dans les variables initiales soient comprimées ou compressées dans les premières composantes. Ainsi, l'idée est que les données à 10 dimensions vous donnent 10 composants principaux, mais PCA essaie de mettre le maximum d'informations possibles dans le premier composant, puis le maximum d'informations restantes dans le second et ainsi de suite.

Organiser les informations en composants principaux de cette manière, vous permettra de réduire la dimensionnalité sans perdre beaucoup d'informations, et ce en éliminant les composants avec peu d'informations et en considérant les composants restants comme vos nouvelles variables.

Une chose importante à réaliser ici est que les composants principaux sont moins interprétables et n'ont pas de sens réel car ils sont construits comme des combinaisons linéaires des variables initiales.

Géométriquement parlant, les composantes principales représentent les directions des données qui expliquent une quantité maximale de variance, c'est-à-dire les lignes qui capturent la plupart des informations des données. La relation entre la variance et l'information ici est que, plus la variance portée par une ligne est grande, plus la dispersion des points de données le long de celle-ci est grande, et plus la dispersion le long d'une ligne est grande, plus elle contient d'informations. Pour faire simple, il suffit de considérer les composants principaux comme de nouveaux axes qui offrent le meilleur angle pour voir et évaluer les données, afin que les différences entre les observations soient mieux visibles.

Supposons que notre jeu de données soit bidimensionnel avec 2 variables x,y et que les vecteurs propres et les valeurs propres de la matrice de covariance soient les suivants :

$$v1 = \begin{bmatrix} 0.6778736 \\ 0.7351785 \end{bmatrix}$$

$\lambda_1 = 1.284028$

$$v2 = \begin{bmatrix} -0.7351785 \\ 0.6778736 \end{bmatrix}$$

$\lambda_2 = 0.04908323$

Si on classe les valeurs propres par ordre décroissant, on obtient $\lambda_1 > \lambda_2$, ce qui signifie que le vecteur propre qui correspond à la première composante principale (PC1) est v1 et celui qui correspond à la seconde composante (PC2) est v2.

Après avoir obtenu les composantes principales, pour calculer le pourcentage de variance (information) représenté par chaque composante, nous divisons la valeur propre de chaque composante par la somme des valeurs propres. Si nous appliquons ceci sur l'exemple ci-dessus, nous constatons que PC1 et PC2 portent respectivement 96% et 4% de la variance des données.

2.5.4.4 Vecteur de caractéristique

Comme nous l'avons vu à l'étape précédente, calculer les vecteurs propres et les ordonner par leurs valeurs propres dans l'ordre décroissant, nous permet de trouver les composantes principales par ordre d'importance. Dans cette étape, ce que nous faisons, c'est choisir de conserver tous ces composants ou de rejeter ceux de moindre importance (de faibles valeurs propres), et de former avec les autres une matrice de vecteurs que nous appelons vecteur de caractéristique.

Ainsi, le vecteur caractéristique est simplement une matrice qui a comme colonnes les vecteurs propres des composants que nous décidons de conserver. Cela en fait le premier pas vers la réduction de la dimensionnalité, car si nous choisissons de ne conserver que p vecteurs propres (composants) sur n, l'ensemble de données final n'aura que p dimensions.

Continuing with the example from the previous step, we can either form a feature vector with both of the eigenvectors v1 and v2 :

$$mat = \begin{bmatrix} 0.6778736 & -0.7351785 \\ 0.7351785 & 0.6778736 \end{bmatrix}$$

Ou supprimer le vecteur propre v2, qui est celui de moindre importance, et formez un vecteur caractéristique avec v1 uniquement :

$$mat = \begin{bmatrix} 0.6778736 \\ 0.7351785 \end{bmatrix}$$

La suppression du vecteur propre v2 réduira la dimensionnalité de 1 et entraînera par conséquent une perte d'informations dans l'ensemble de données final. Mais étant donné que la v2 ne portait que 4% des informations, la perte ne sera donc pas importante et nous aurons toujours 96% des informations qui sont portées par la v1.

Ainsi, comme nous l'avons vu dans l'exemple, c'est à vous de choisir de conserver tous les composants ou de rejeter ceux de moindre importance, en fonction de ce que vous recherchez. Parce que si vous souhaitez simplement décrire vos données en termes de nouvelles variables (composantes principales) non corrélées sans chercher à réduire la dimensionnalité, il n'est pas nécessaire de laisser de côté les composants moins significatifs.

2.5.4.5 Reconstituer les données le long des axes des principaux composants

Dans les étapes précédentes, à part la normalisation, vous n'apportez aucune modification aux données, vous sélectionnez simplement les composants principaux et formez le vecteur de caractéristiques, mais l'ensemble de données d'entrée reste toujours en termes d'axes d'origine (c'est-à-dire en termes de les variables initiales).

Dans cette étape, qui est la dernière, l'objectif est d'utiliser le vecteur de caractéristiques formé à l'aide des vecteurs propres de la matrice de covariance, pour réorienter les données des axes d'origine vers ceux représentés par les composantes principales (d'où le nom Analyse en composantes principales). Cela peut être fait en multipliant la transposition de l'ensemble de données d'origine par la transposition du vecteur de caractéristiques.

(2.19)

$$ensemblededonnéesfinal = Vecteurdecaractéristique^T * ensemblededonnéesoriginalstanda$$

2.6 Divergence de Kullback-Leibler

En théorie des probabilités et en théorie de l'information, la divergence de Kullback-Leibler (ou divergence K-L ou encore entropie relative) est une mesure de dissimilarité entre deux distributions de probabilités. Elle doit son nom à Solomon Kullback et Richard Leibler, deux cryptanalystes américains. Selon la NSA, c'est durant les années 1950, alors qu'ils travaillaient pour cette agence, que Kullback et Leibler ont inventé cette mesure. Elle aurait d'ailleurs servi à la NSA dans son effort de cryptanalyse pour le projet Venona.

Considérons deux distributions de probabilités P et Q. Typiquement, P représente les données, les observations, ou une distribution de probabilités calculée avec précision. La distribution Q représente typiquement une théorie, un modèle, une description ou une approximation de P. La divergence de Kullback-Leibler s'interprète comme la différence moyenne du nombre de bits nécessaires au codage d'échantillons de P en utilisant un code optimisé pour Q plutôt que le code optimisé pour P.

2.6.1 Définitions

Il existe plusieurs définitions selon les hypothèses sur les distributions de probabilités.

2.6.1.1 Premières définitions

Pour deux distributions de probabilités discrètes P et Q, la divergence de Kullback–Leibler de P par rapport à Q est définie par :

$$D_{KL}(P||Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)} \tag{2.20}$$

Pour des distributions P et Q continues, on utilise une intégrale

$$D_{KL}(P||Q) = \int_{-\infty}^{+\infty} p(x) \log \frac{p(x)}{q(x)} dx \tag{2.21}$$

où p et q sont les densités respectives de P et Q.

En d'autres termes, la divergence de Kullback-Leibler est l'espérance de la différence des logarithmes de P et Q, en prenant la probabilité P pour calculer l'espérance.

2.6.1.2 Définitions générales

On peut généraliser les deux cas particuliers ci-dessus en considérant P et Q deux mesures définies sur un ensemble X, absolument continues par rapport à une mesure μ : le théorème de Radon-Nikodym-Lebesgue assure l'existence des densités p et q avec $dP = pd\mu$ et $dQ = qd\mu$, on pose alors :

$$D_{KL}(P||Q) = \int_x p \log \frac{p}{q} d\mu \tag{2.22}$$

sous réserve que la quantité de droite existe. Si P est absolument continue par rapport à Q, (ce qui est nécessaire si $D_{KL}(P||Q)$ est finie) alors $\frac{p}{q} = \frac{dP}{dQ}$ est la dérivée de Radon-Nikodym de P par rapport à Q et on obtient :

$$D_{KL}(P||Q) = \int_x \log \frac{dP}{dQ} dP = \int_x \frac{dP}{dQ} \log \frac{dP}{dQ} dQ \tag{2.23}$$

où l'on reconnait l'entropie de P par rapport à Q.

De même, si Q est absolument continue par rapport à P, on a :

$$D_{KL}(P||Q) = -\int_x \log \frac{dQ}{dP} dQ \tag{2.24}$$

Dans les deux cas, on constate que la divergence de Kullback-Leibler ne dépend pas de la mesure μ.

Lorsque les logarithmes de ces formules sont pris en base 2 l'information est mesurée en bits ; lorsque la base est e, l'unité est le nat.

CHAPITRE 3

SYSTÈME DE DÉTECTION D'INTRUSION

3.1 Introduction

Tout d'abord, IDS signifie Intrusion Détection System. Il s'agit d'un équipement permettant de surveiller l'activité d'un réseau ou d'un hôte donné, afin de détecter toute tentative d'intrusion et éventuellement de réagir à cette tentative.

La détection d'intrusion est le processus de la surveillance des événements qui se produisent dans un ordinateur ou dans un réseau et de les analyser pour des signes d'intrusions, qui sont définis comme des tentatives qui compromettent la confidentialité, l'intégrité ou la disponibilité ou bien qui dépassent les conditions de la sécurité d'un ordinateur ou un réseau.

Les principales raisons pour l'utilisation des IDS sont :

— Pour fournir les informations possibles sur les intrusions et les tentatives qui ont eu place, permettant l'amélioration du diagnostic, la récupération, et la correction des facteurs causals.
— Pour agir et contrôler la qualité de la sécurité et l'administration, en particulier dans les grandes entreprises.
— Pour déceler les objectifs des attaques.
— Pour éviter les problèmes, on augmente la protection des risques qui sont découverts, et infliger une punition pour ceux qui voudraient attaquer ou autrement abuser du système.
— Pour détecter les attaques et les violations de sécurité qui ne sont pas prévus par

d'autres mesures de sécurité.

Dans le monde réel, IDS ressemble à l'alarme antivol dans une maison. Disons que le Dispositif de commande d'authentification, qui est situé à l'entrée de la porte ressemble au Pare-feu. Il contrôle l'accès à la maison par carte d'identité et mot de passe. Il y a plusieurs façons de contourner ce dispositif. Soit un voleur peut casser la fenêtre et pénètre dans la maison avec succès sans détecter, ou un voleur peut usurper la carte d'identité et le mot de passe. Dans les deux cas cidessus. Le voleur n'est pas détecté et il n'y a aucune information concernant les dégâts, comment il s'est pris, qui est le voleur et quand a-t-il tenté le cambriolage.

Voyons comment il est différent avec les alarmes antivol. Alarmes contre le vol sont activés dans la maison et configuré pour donner l'alarme si quelqu'un pénètre par infraction. Si le voleur entre dans la maison, le système d'alarme se déclenche, en faisant alerter le gardien de la sécurité ou le propriétaire de la demeure, il peut informer les personnes associées, il peut enregistrer l'heure du cambriolage et de plus amples informations en fonction de la configuration politique de la sécurité et la capacité de traitement de l'alarme anti-intrusion. Ainsi, les alarmes anti-intrusion réduisent les dommages d'un cambriolage, en fournissant les informations complètes et les indices pour l'enquête médico-légale et de prendre une action en justice contre les voleurs.

La détection d'intrusion comporte également des mécanismes fonctionnels similaires, qui réduisent le risque, et aide à l'enquête médico-légale, et contribue à rapporter les événements au management. Dans ce qui suit nous allons discuter quelques-uns des mythes associés à la technologie de détection des intrusions.

3.2 Classification des IDS

Nous allons présenter ici la technologie de détection d'intrusion d'une manière taxonomique. Il y a plusieurs types d'IDS disponibles aujourd'hui, caractérisé par différente approche de la surveillance et de l'analyse. Chaque approche a ses avantages et des inconvénients distincts. Toutes les approches peuvent être décrites en termes d'un modèle d'IDS.

— L'emplacement d'IDS
— Modes de détection
— Les types de réponse
— Fréquence d'utilisation

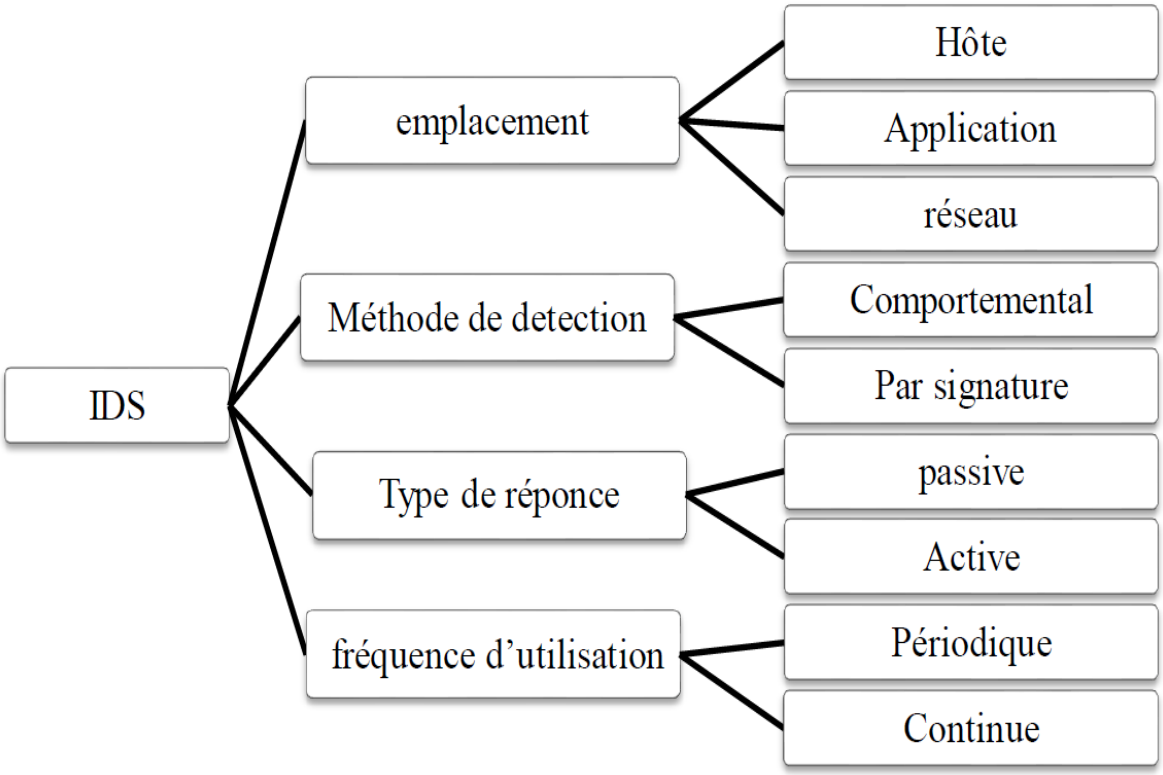

FIGURE 3.1: Les différentes sortes des IDS

3.2.1 L'emplacement d'IDS

La façon la plus commune pour classifier les IDS est de les regrouper par emplacement de l'information source où ils opèrent. Les sources des informations primaires sont : les paquets réseau capturés à partir des réseaux ou des segments de réseau local ou les systèmes d'exploitation et les fichiers critiques.

3.2.1.1 La détection d'intrusion basée sur l'hôte

Les systèmes de détection d'intrusion basés sur l'hôte ou HIDS (Host IDS) analysent exclusivement l'information concernant cet hôte. Comme ils n'ont pas à contrôler le trafic du réseau mais "seulement" les activités d'un hôte.

Avec une grande fiabilité et précision, l'IDS peut déterminer exactement quels utilisateurs ou quels processus sont impliqués dans une attaque. Avec ces types de systèmes, le résultat peut être déterminée à la différence du NIDS, le HIDS peut accéder directement et de contrôler les fichiers de données et les processus habituellement visés par les

attaques.

Ces IDS utilisent généralement les traces d'audit du système d'exploitation afin de fournir une information sur l'activité de la machine.il possède un certain nombre d'avantages : il est possible de constater immédiatement l'impact d'une attaque et donc de mieux réagir. Grâce à la quantité d'information étudiée, il est possible d'observer les activités se déroulant sur l'hôte avec précision et d'optimiser le système en fonction des activités observées.

De plus, les HIDS sont extrêmement complémentaires des NIDS. En effet, ils permettent de détecter plus facilement les attaques de type "Cheval de Troie", alors que ce type d'attaque est difficilement détectable par un NIDS. Les HIDS permettent également de détecter des attaques impossibles à détecter avec un NIDS, car elles font partie du trafic crypté.

Néanmoins, ce type d'IDS possède également ses faiblesses, qui proviennent de ses qualités : du fait de la grande quantité de données générées, ce type d'IDS est très sensible aux attaques de type DoS, qui peuvent faire exploser la taille des fichiers de logs.

Un autre inconvénient tient justement à la taille des fichiers de rapport d'alertes à examiner, qui est très contraignante pour le responsable de sécurité. La taille des fichiers peut en effet atteindre plusieurs Mégaoctets. Du fait de cette quantité de données à traiter, ils sont assez gourmands en CPU et peuvent parfois altérer les performances de la machine hôte.

Les HIDS sont en général placés sur des machines sensibles, susceptibles de subir des attaques et possédantes des données sensibles pour l'entreprise. Les serveurs web ou applicatifs, peuvent notamment être protégés par un HIDS.

3.2.1.2 Détection d'Intrusion basée sur une application

Les IDS basés sur les applications (AIDS) sont un sous-groupe des IDS hôtes. Ils contrôlent l'interaction entre un utilisateur et un programme en ajoutant des fichiers de log afin de fournir de plus amples informations sur les activités d'une application particulière. Puisque vous opérez entre un utilisateur et un programme, il est facile de filtrer tout comportement notable. Un ABIDS se situe au niveau de la communication entre un utilisateur et l'application surveillée. L'avantage de cet IDS est qu'il lui est possible de détecter et d'empêcher des commandes particulières dont l'utilisateur pourrait se servir avec le programme et de surveiller chaque transaction entre l'utilisateur et

l'application. De plus, les données sont décodées dans un contexte connu, leur analyse est donc plus fine et précise.

Par contre, du fait que cet IDS n'agit pas au niveau du noyau, la sécurité assurée est plus faible, notamment en ce qui concerne les attaques de type "Cheval de Troie". De plus, les fichiers de log générés par ce type d'IDS sont des cibles faciles pour les attaquants et ne sont pas aussi sûrs, par exemple, que les traces d'audit du système. Ce type d'IDS est utile pour surveiller l'activité d'une application très sensible, mais son utilisation s'effectue en général en association avec un HIDS. Il faudra dans ce cas contrôler le taux d'utilisation CPU des IDS afin de ne pas compromettre les performances de la machine.

3.2.1.3 La Détection d'Intrusion Réseau

La forme la plus commune de systèmes commerciaux de détection d'intrusion est basée sur le réseau. Ces systèmes détectent les attaques en capturant et en analysant les paquets réseau en écoutant sur le segment de réseau ou d'un commutateur. Ils le font en faisant correspondre entre un ou plusieurs paquets contre une base de données de signatures d'attaques connues, ou d'effectuer le décodage du protocole pour détecter les anomalies.

Le NIDS est capable à la fois de déclencher des alertes et de clôturer les connexions instantanément chaque fois qu'il remarque des activités suspectes. Le « mode Promiscuous » est la forme la plus commune du fonctionnement .ils surveillent chaque paquet qui est en transmission du segment local.

L'implantation d'un NIDS sur un réseau se fait de la façon suivante : des capteurs sont placés aux endroits stratégiques du réseau et génèrent des alertes s'ils détectent une attaque. Ces alertes sont envoyées à une console sécurisée, qui les analyse et les traites éventuellement. Cette console est généralement située sur un réseau isolé, qui relie uniquement les capteurs et la console.

3.2.2 Modes de détection

Il existe deux modes de détection, la détection d'anomalies et la reconnaissance de signatures. Il faut noter que la reconnaissance de signature est le mode de fonctionnement le plus implémenté par les IDS du marché. Cependant, les nouveaux produits tendent à combiner les deux méthodes pour affiner la détection d'intrusion.

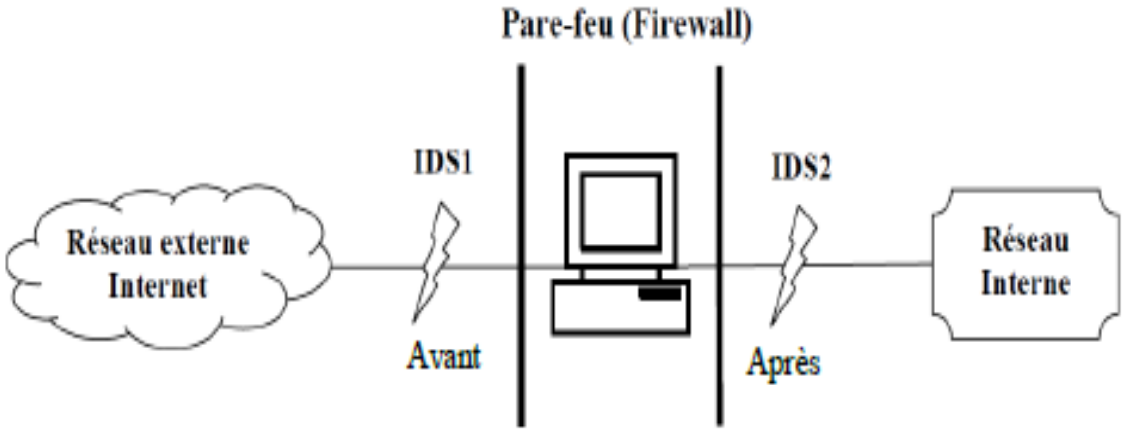

FIGURE 3.2: Installation des N-IDS

3.2.2.1 La détection d'anomalies

Elle consiste à détecter des anomalies par rapport à un profil "de trafic habituel". La mise en oeuvre comprend toujours une phase d'apprentissage au cours de laquelle les IDS vont "découvrir" le fonctionnement "normal" des éléments surveillés. Ils sont ainsi en mesure de signaler les divergences par rapport au fonctionnement de référence.

Dans le cas du HIDS, ce type de détection peut être basé sur des informations telles que le taux d'utilisation CPU, l'activité sur le disque, les horaires de connexion ou d'utilisation de certains fichiers (horaires de bureau. . .). Citons, à titre d'exemple, un projet a été sponsorisé par la DARPA (en 98 et 99), en collaboration avec le laboratoire Lincoln du MIT et qui était l'un des projets les plus ambitieux [Lippmann al, 00]. Le but était de fournir un ensemble significatif de données d'apprentissage, comprenant trafic de fond et activités intrusives (c'est-à-dire du trafic intrusif ou des événements systèmes causés par des attaques). Le trafic de fond était déduit des données statistiques collectées sur le réseau des bases de l'Air Force alors que les attaques étaient générées par des scripts créés spécialement, mais aussi par des scripts collectés à travers des sites spécialisés et des listes de diffusion. Les données collectées concernaient à la fois des HIDS (par exemple données d'audit de stations Solaris, disk dump de machines UNIX et BSM «Basic Security Monitoring ») et des NIDS.

Le jeu de test DARPA'98 utilisait environ 300 attaques (classées en 38 types d'attaques), tandis que DARPA'99 utilisait environ 50 types d'attaques.

3.2.2.2 La reconnaissance de signature

Cette approche consiste à rechercher dans l'activité de l'élément surveillé les empreintes (ou signatures) d'attaques connues. Ce type d'IDS est purement réactif ; il ne peut détecter que les attaques dont il possède la signature. De ce fait, il nécessite des mises à jour fréquentes.

De plus, l'efficacité de ce système de détection dépend fortement de la précision de sa base de signature. C'est pourquoi ces systèmes sont contournés par les pirates qui utilisent des techniques dites "d'évasion" qui consistent à maquiller les attaques utilisées. Ces techniques tendent à faire varier les signatures des attaques qui ainsi ne sont plus reconnues par l'IDS. Il est possible d'élaborer des signatures plus génériques, qui permettent de détecter les variantes d'une même attaque, mais cela demande une bonne connaissance des attaques et du réseau, de façon à stopper les variantes d'une attaque et à ne pas gêner le trafic normal du réseau Une signature permet de définir les caractéristiques d'une attaque, au niveau des paquets (jusqu'à TCP ou UDP) ou au niveau protocole (HTTP, FTP. . .).

Au niveau paquet, l'IDS va analyser les différents paramètres de tous les paquets transitant et les comparer avec les signatures d'attaques connues. Au niveau protocole, l'IDS va vérifier au niveau du protocole si les commandes envoyées sont correctes ou ne contiennent pas d'attaque. Cette fonctionnalité a surtout été développée pour HTTP actuellement.

Cet exemple nous a semblé très formateur sur le rôle des signatures et leur élaboration. Il faut savoir que les sites des vendeurs d'IDS proposent des mises à jour des signatures en fonction des nouvelles attaques identifiées. Néanmoins, plus il y a de signatures différentes à tester, plus le temps de traitement sera long, l'utilisation de signatures plus élaborées peut donc procurer un gain de temps appréciable.

Cependant, une signature mal élaborée peut ignorer des attaques réelles ou identifier du trafic normal comme étant une attaque. Il convient donc de manier l'élaboration de signatures avec précaution, et en ayant de bonnes connaissances sur le réseau surveillé et les attaques existantes.

3.2.3 Les types de réponse

Il existe deux types de réponses, suivant les IDS utilisés. La réponse passive est disponible pour tous les IDS, la réponse active est plus ou moins implémentée.

3.2.3.1 Réponse active

La réponse active au contraire a pour but de stopper une attaque au moment de sa détection. Pour cela on dispose de deux techniques : la reconfiguration du firewall et l'interruption d'une connexion TCP.

La reconfiguration du firewall permet de bloquer le trafic malveillant au niveau du firewall, en fermant le port utilisé ou en interdisant l'adresse de l'attaquant. Cette fonctionnalité dépend du modèle de firewall utilisé, tous les modèles ne permettant pas la reconfiguration par un IDS. De plus, cette reconfiguration ne peut se faire qu'en fonction des capacités du firewall.

L'IDS peut également interrompre une session établie entre un attaquant et sa machine cible, de façon à empêcher le transfert de données ou la modification du système attaqué.

Pour cela l'IDS envoie un paquet TCP reset aux deux extrémités de la connexion (cible et attaquant). Un paquet TCP reset a le flag RST positionné, ce qui indique une déconnexion de la part de l'autre extrémité de la connexion. Chaque extrémité en étant destinataire, la cible et l'attaquant pensent que l'autre extrémité s'est déconnectée et l'attaque est interrompue.

3.2.3.2 Réponse passive

La réponse passive d'un IDS consiste à enregistrer les intrusions détectées dans un fichier de log qui sera analysé par le responsable de sécurité.

Certains IDS permettent de d'enregistrer l'ensemble d'une connexion identifiée comme malveillante. Ceci permet de remédier aux failles de sécurité pour empêcher les attaques enregistrées de se reproduire, mais elle n'empêche pas directement une attaque de se produire.

3.2.4 Fréquence d'utilisation

Enfin, une dernière différenciation est la caractéristique d'utilisation qui peut se faire d'une façon : continue (online) ou périodique (offline).

3.2.4.1 Utilisation continue

La détection d'attaque se fait au moment où elle se produit, Dans la plupart des cas, avant d'attaquer un réseau, l'attaquant doit scanner l'environnement pour récolter des

informations. Par conséquent, en voyant cela, on peut détecter une attaque avant même qu'elle ne se produise et ainsi y répondre le plus tôt possible.

3.2.4.2 Utilisation périodique

Cette méthode s'exécute périodiquement et on ne voit que le résultat d'attaque, elle est préférable pour avoir une défense plus fiable du point de vue du temps de calcul que pour la première utilisation. Contrairement à l'IDS online, l'attaque ne peut pas être détectée le plus tôt possible pour l'éviter. Plus une attaque est détectée tardivement, les dommages sont importants.

3.3 Conclusion

Dans ce chapitre, nous avons présenté un état de l'art en matière de détection d'intrusions. Pour bien positionner le problème nous avons d'abord donné une définition à cette notion, ensuite nous avons présenté une classification des systèmes de détection d'intrusions (IDS) en se basant sur la source de données de ces derniers ainsi que sur leurs emplacements dans le système informatique. Cette classification comporte les IDS basés hôte (HIDS), les IDS basés sur l'application (AB-IDS) les IDS basés réseau (N-IDS).

Nous avons abordé aussi les deux grandes approches utilisées par les Modes de détection IDS à savoir : La détection d'anomalies et La reconnaissance de signature , chacun d'entre eux présent des méthodes avec leurs avantages et leurs inconvénients. Ensuite nous avons présenté les deux réponses apportées par les systèmes après détection d'une intrusion, que ce soit la réponse active qui englobe plusieurs techniques comme la reconfiguration du firewall et l'interruption d'une connexion TCP, ou bien la réponse passive dont le principe repose sur l'émission des alertes, la chose qui est assurée par la plupart des IDS actuels. Pour donner par la suite un ensemble de procédures à respecter lors de l'implémentation des IDS.

Pour conclure on peut dire que les IDS constituent une seconde ligne de défense indispensable pour assurer la sécurité opérationnelle, surtout dans un contexte d 'interconnexion et d 'ouverture croissant des systèmes informatiques. Le chapitre suivant sera consacré à la base principale du développement des IDS à savoir les attaques informatiques ainsi que des notion sur les données d'apprentissage.

CHAPITRE 4

LA RÉALISATION

4.1 Introduction

Au cours des dernières années, Internet a connu une croissance explosive. Parallèlement à la prolifération de nouveaux services, la quantité et l'impact des attaques n'ont cessé d'augmenter. Le nombre de systèmes informatiques et de leurs vulnérabilités a augmenté, tandis que le niveau de sophistication et de connaissances requis pour mener une attaque a diminué, tant le savoir-faire technique en matière d'attaque est facilement disponible sur les sites Web du monde entier.

En conséquence, de nombreux groupes de recherche ont concentré leur attention sur le développement de nouvelles techniques de détection, capables de révéler et d'identifier rapidement les attaques de réseau [3, 8, 18]. Parmi celles-ci, certaines des approches les plus prometteuses reposent sur l'utilisation de l'ACP.

L'ACP est une technique de réduction de dimensionnalité qui renvoie une représentation compacte d'un ensemble de données multidimensionnel, en projetant les données sur un sous-espace de dimension inférieure. En effet, il permet de réduire la dimensionnalité de l'ensemble de données (nombre de variables), tout en conservant l'essentiel de la variabilité originelle des données.

Un autre élément important est que la plupart des travaux proposés dans la littérature, comprenant la plupart de ceux basés sur PCA, analysent les flux de trafic uniques, ce qui les rend non évolutifs et, par conséquent, non applicables dans les réseaux backbone modernes (ce qui est le scénario considéré dans notre travail).

Pour ces raisons, dans cet article, nous nous sommes concentrés sur le développement d'un système de détection d'intrusion réseau (IDS) basé sur les anomalies, qui applique la PCA aux agrégats de trafic.

Dans ce chapitre, nous présentons l'architecture du système que nous avons mis en place pour détecter les anomalies dans le trafic réseau.

4.2 Contexte théorique

Dans cette section, nous décrivons brièvement les deux éléments clés de l'approche proposée (c'est-à-dire le Sketch et l'ACP), en fournissant une raison intuitive pour leur application à la détection d'anomalies de réseau.

4.2.1 Sketch

Les Sketches sont une famille de structures de données qui utilisent le même schéma de hachage sous-jacent pour résumer les données. Ils diffèrent dans la façon dont ils mettent à jour les compartiments de hachage et utilisent des données hachées pour dériver des estimations. Parmi les différentes sketchs, celle avec les meilleures limites temporelles et spatiales est la soi-disant sketch count-min [5], qui est essentiellement celle utilisée dans ce travail.

Plus précisément, la structure de données d'sketch est un tableau dxw à deux dimensions $T[l][j]$, où chaque ligne $l(l = 1,\ldots,d)$ est associée à une fonction de hachage h_l donnée. Dans notre travail, nous choisissons d'utiliser des fonctions appartenant à la famille de hachage 4-universelle [1] [19]. Ces fonctions donnent une sortie dans l'intervalle $(1,\ldots,w)$, et ces sorties sont associées aux colonnes du tableau. A titre d'exemple, l'élément $T[l][j]$ est associé à la valeur de sortie j de la fonction de hachage l .

Les données d'entrée sont vues comme un flux qui arrive de manière séquentielle, élément par élément. Chaque élément se compose d'une clé dièse, $i_t \in (1,\ldots,N)$, et d'un poids, c_t . Lorsque de nouvelles données arrivent, le sketch est mise à jour comme suit :

$$T[l][h_l(i_t)] \leftarrow T[l][h_l(i_t)] + c_t \tag{4.2}$$

1. Une classe de fonctions de hachage $H : (1,\ldots,N) \rightarrow (1,\ldots,w)$ est un hachage k-universel si pour tout $x_0,\ldots,x_{k-1} \in (1,\ldots,N)$ distinct, et tout $v_0,\ldots,v_{k-1} \in (1,\ldots,W)$ possible.

$$Pr_{h\in H} = Pr\{h(x_i) = v_i; \forall i \in (1,\ldots,k)\} = \frac{1}{W^k} \tag{4.1}$$

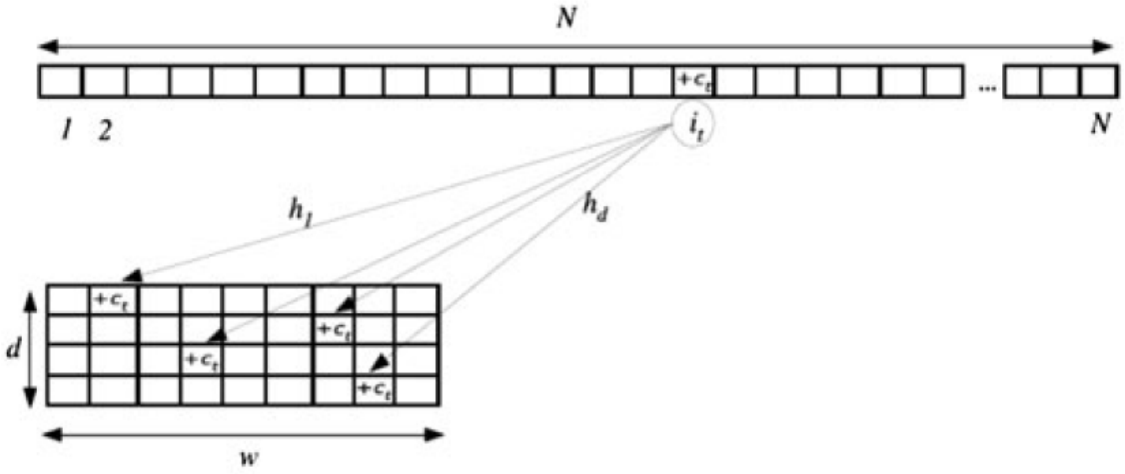

FIGURE 4.1: Sketch : fonction de mise à jour

La procédure de mise à jour est réalisée pour toutes les différentes fonctions de hachage, comme illustré à la figure 4.1.

Il est à noter qu'étant donné l'utilisation des fonctions de hachage, il est possible d'avoir des collisions dans la table de croquis. Dans ce travail, nous avons profité de ce fait, en effet avoir des collisions nous permet d'agréger aléatoirement les flux de trafic, à savoir tous les flux IP qui entrent en collision dans un même bucket seront considérés comme un agrégat. A noter qu'ainsi chaque flux IP fera partie de d agrégats aléatoires distincts, dont chacun sera analysé pour vérifier s'il présente une anomalie. Cela signifie que, dans la pratique, tout flux sera vérifié plus d'une fois ; ainsi, il sera plus facile de détecter un écoulement anormal. En effet, un flux anormal pourrait être masqué dans un agrégat de trafic donné, tout en étant détectable dans un autre.

4.2.2 Analyse en composantes principales

L'ACP est une transformation linéaire qui mappe un espace de coordonnées sur un nouveau système de coordonnées dont les axes, appelés composantes principales (PC), ont la propriété de pointer dans la direction de la variance maximale des données résiduelles (c'est-à-dire la différence entre les données d'origine et les données mappées sur les PC précédents).

Plus précisément, le premier PC capture le plus grand degré de variance des données dans une seule direction, le second capture le plus grand degré de variance des données dans les directions orthogonales restantes, et ainsi de suite.

Ainsi, les PC sont classés en fonction de la quantité de variance de données qu'ils capturent. En règle générale, les premiers PC contribuent à la majeure partie de la variance dans l'ensemble de données d'origine, de sorte que nous pouvons les décrire avec uniquement ces PC, en négligeant les autres, avec une perte de variance minimale.

En termes mathématiques, calculer les PC équivaut à calculer les vecteurs propres. Ainsi, étant donné la matrice de données $B = \{B_{i,j}\}$, avec $1 < i < t$ et $1 < j < m$ (un ensemble de données de m échantillons capturés en t intervalles de temps), chaque PC, v_i, est le $i$ème vecteur propre calculé à partir de la décomposition spectrale de $B^T B$, c'est-à-dire :

$$B^T B v_i = \lambda_i v_i, i = 1, \ldots, m \tag{4.3}$$

Où λ_i est la valeur propre « ordonnée » correspondant au vecteur propre v_i.

En pratique, le premier PC, v_1, est calculé comme suit :

$$v_1 = argmax_{\|v=1\|} \|B_v\| \tag{4.4}$$

En procédant de manière récursive, une fois les $k-1$ premiers CP déterminés, le $k$ème CP peut être évalué comme suit :

$$v_k = argmax_{\|v=1\|} \left\| \left(B - \sum_{i=1}^{k-1} B v_i {v_i}^t \right) v \right\| \tag{4.5}$$

où $\|.\|$ désigne la norme L^2.

Une fois les PC calculés, étant donné un ensemble de données et son espace de coordonnées associé, nous pouvons effectuer une transformation des données en les projetant sur les nouveaux axes.

4.3 Architecture du système

Dans cette section, nous présentons l'architecture du système que nous avons mis en place pour détecter les anomalies dans le trafic réseau.

4.3.1 Entrée système

Tout d'abord, les données d'entrée sont traitées par un module appelé, dans la figure 4.2, Data Formatting. En effet, ce module est chargé de lire les traces Netflow [4] et de les transformer en fichiers de données ASCII, au moyen des Flow-Tools [20].

Plus en détail, dans notre implémentation, nous avons utilisé des données d'entrée Netflow, mesurant le trafic passé par un routeur donné sur une tranche de temps donnée (dans ce qui suit, nous avons T tranches de temps distinctes).

4.3.2 Agrégation de flux :

Il est à noter que notre système fonctionne sur trois échelles de temps distinctes, à savoir qu'il utilise des tranches de temps de 5, 10 et 15 min. De plus, pour faciliter la détection de corrélations et de tendances périodiques dans les données, nous avons étudié différents niveaux d'agrégation. Plus précisément, le bloc appelé Flow Aggregation réalise quatre niveaux d'agrégation différents :

— IR
— OD flows
— IL
— Random aggregation

À l'aide de l'agrégation IR, les données sont organisées en fonction du routeur auquel elles sont entrées dans le réseau. Le flux OD agrège le trafic sur la base des routeurs d'entrée et de sortie utilisés par un flux IP. Au lieu de cela, l'IL agrège les flux IP avec le même IR et la même interface d'entrée. Enfin, l'agrégation aléatoire est effectuée au moyen de le sketch.

La sortie de ce bloc est donnée par 4T fichiers distincts, chaque fichier correspondant à une tranche horaire spécifique et à un agrégat spécifique.

A noter que la modularité du système permet une grande flexibilité. En effet, au lieu de considérer le nombre d'octets envoyés par une IP donnée, l'administrateur système peut facilement choisir d'utiliser un autre descripteur de trafic qui permet de mieux détecter les différentes attaques.

4.3.3 Construction de séries temporelles

Une fois les données correctement formatées, elles sont transmises en entrée au troisième module, chargé de la construction de la série temporelle.

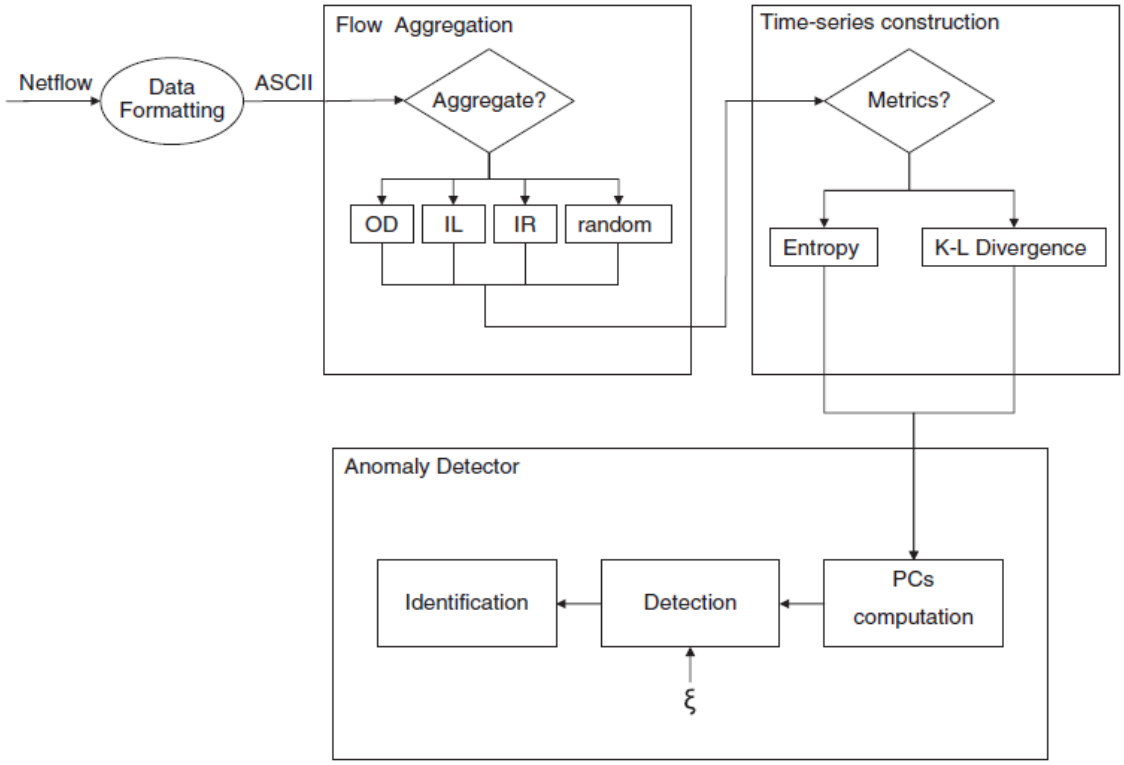

FIGURE 4.2: Architecture du système.

Dans ce travail, nous avons pris en considération la quantité b_{ij}^t , c'est-à-dire le nombre d'octets envoyés par la j ème adresse IP source du i ème agrégat dans le temps bin t . La distribution des caractéristiques a été estimée avec l'histogramme empirique. Ainsi, dans chaque tranche de temps pour chaque agrégat, nous avons évalué l'histogramme comme suit :

$$B_{i,.}^t = \left\{ \beta_{i,j}^t \right\}_{j=1}^M \tag{4.6}$$

Où :

$$\beta_{i,j}^t = \frac{b_{i,j}^t}{\sum_{j=1}^M b_{i,j}^t} \tag{4.7}$$

Malheureusement, l'histogramme est un objet de grande dimension assez difficile à manipuler avec de faibles ressources de calcul. Pour cette raison, nous avons essayé de

concentrer les informations prises par l'histogramme en une seule valeur qui est capable de contenir la plupart des informations utiles, qui, dans notre cas, est la tendance de la distribution. Des travaux antérieurs [11–13] ont souligné la possibilité d'extraire des informations très utiles de l'entropie de l'histogramme, qui fournit une métrique informatique efficace pour estimer le degré de dispersion ou de concentration d'une distribution.

Étant donné l'histogramme empirique, X^t , nous pouvons évaluer la valeur d'entropie comme suit :

$$y_{it} = -\sum_{j=1}^{M} \beta_{i,j}^{t} \log_2 \beta_{i,j}^{t} \tag{4.8}$$

Néanmoins, l'entropie n'est capable de capturer que les informations relatives à une seule tranche de temps, bien que de notre point de vue, il serait beaucoup plus important de capturer la différence entre les distributions de caractéristiques de paquets de deux tranches de temps adjacentes.

Pour cette raison, dans ce travail, nous avons également utilisé une autre métrique qui est la divergence K-L. Ainsi, étant donné deux histogrammes X^t (capturé dans le temps-intervalle t) et X^{t-1} (capturé dans le temps-intervalle t-1), la divergence K-L est définie comme suit :

$$D_{t,i} = \sum_{j=1}^{M} \beta_{i,j}^{t} \log \frac{\beta_{i,j}^{t}}{\beta_{i,j}^{t-1}} \tag{4.9}$$

Malgré la méthode utilisée, ce module produira une matrice pour chaque type d'agrégation dans laquelle, pour tous les agrégats, les valeurs de la métrique (entropie ou divergence K-L) évaluées dans chaque tranche de temps sont rapportées.

Cette matrice a la structure suivante :

$$Y = \begin{bmatrix} y_{11} & \cdots & y_{1N} \\ . & & . \\ . & & . \\ . & & . \\ y_{T1} & \cdots & y_{TN} \end{bmatrix}$$

où N est le nombre d'agrégats et T est le nombre d'intervalles de temps.

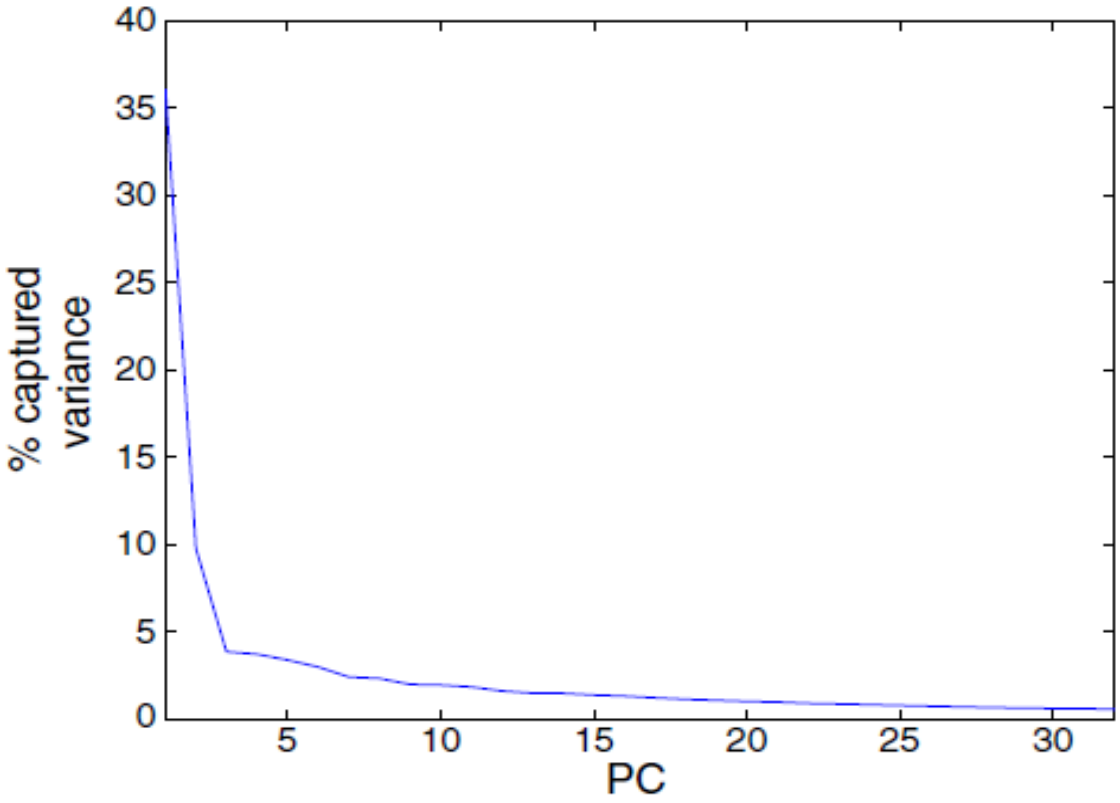

FIGURE 4.3: Scree plot

4.3.4 Calcul des PC

Une fois les séries temporelles construites, elles sont transmises au module qui applique l'ACP. Le calcul des PC peut être effectué à l'aide des équations decrites au chapitre 1. Comme décrit précédemment, il existe généralement un ensemble de PC (appelés PC dominants) qui capture la majeure partie de la variance dans l'ensemble de données d'origine. L'idée est de sélectionner les PC dominants et de décrire le comportement normal en n'utilisant que ceux-ci. Il convient de souligner que le nombre de PC dominants est un paramètre très important et doit être correctement réglé lors de l'utilisation de PCA comme détecteur d'anomalies de trafic.

Dans notre approche, l'ensemble des PC dominants est sélectionné au moyen de la méthode scree-plot. En conséquence, nous séparons les PC en deux ensembles, les PC dominants et négligeables, qui seront ensuite utilisés pour distinguer les variations normales et anormales du trafic.

Plus en détail, un scree plot est un graphique du pourcentage de variance capturé par

un PC donné. La figure 4.3 présente un scree plot d'un ensemble de données particulier. Visuellement, à partir du graphique, nous pouvons observer que les premiers r=5 PC sont capables de capturer correctement la majorité de la variance. Ces PC sont les PC dominants :

$$P = (v_1, \ldots, v_r) \tag{4.10}$$

La méthode est basée sur l'hypothèse que ces PC sont suffisants pour décrire le comportement normal du trafic.

4.3.5 Phase de détection

La phase de détection est réalisée en séparant l'espace de grande dimension des mesures de trafic en deux sous-espaces, qui capturent respectivement les variations normales et anormales.

Plus précisément, une fois la matrice P construite, nous pouvons diviser l'espace en un sous-espace normal ($\hat{S}$), couvert par les PC dominants, et un sous-espace anormal ($\tilde{S}$), couvert par les PC restants. Les composantes normales et anormales des données peuvent être obtenues en projetant le trafic agrégé sur ces deux sous-espaces. Ainsi, les données d'origine Yt , dans l'intervalle de temps t, sont décomposées en deux parties comme suit :

$$Y_t = \hat{Y}_t + \tilde{Y}_t \tag{4.11}$$

où $\hat{Y}_t$ et $\tilde{Y}_t$ sont la projection sur $\hat{S}$ et $\tilde{S}$, respectivement, et peuvent être évalués comme suit :

$$\hat{Y}_t = PP^T Y_t \tag{4.12}$$

$$\tilde{Y}_t = (I - PP^T)Y_t \tag{4.13}$$

Il est à noter que lorsqu'un trafic anormal traverse le réseau, un changement important dans la composante anormale ($\tilde{Y}_t$) se produit. Ainsi, une méthode efficace pour détecter les anomalies de trafic consiste à comparer $\|\tilde{Y}_t\|^2$ à un seuil donné ϵ.

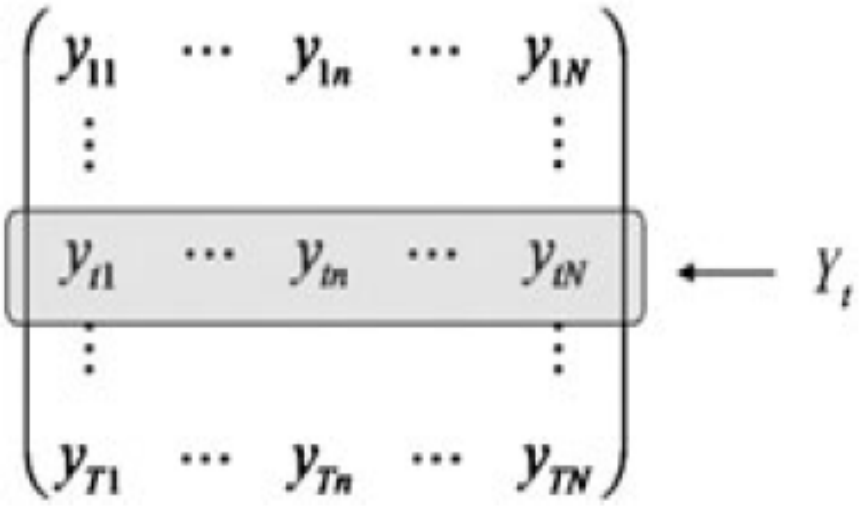

FIGURE 4.4: Matrice de données.

En particulier, si $\|\tilde{Y}_t\|^2$ dépasse ϵ, le trafic est considéré comme anormal, et nous marquons l'intervalle de temps t comme un intervalle de temps anormal (Figure 4.4). Si nous utilisons une agrégation aléatoire, la phase de détection peut être améliorée lors de l'exécution d'une étape supplémentaire.

En effet, dans ce cas, parce que nous avons utilisé d fonctions de hachage différentes h_j, nous avons d matrices de données, Y^j (une pour chaque fonction). Ainsi, l'analyse décrite précédemment (effectuée pour chaque Y^j) renvoie d réponses différentes. Ainsi, une analyse de vote est effectuée. Nous évaluons le nombre d'alarmes produites, et nous décidons si le time-bin est anormal ou non selon la règle suivante :

$$\begin{cases} \text{anormale} & \text{si le nombre d'alarmes} > \frac{d}{2} - 1 \\ \text{normale} & \text{sinon.} \end{cases}$$

4.3.6 Phase d'identification

Si une anomalie a été détectée, le système effectue une nouvelle phase appelée identification de l'anomalie. Il est à noter que PCA fonctionne sur une seule série temporelle, donc dans la phase de détection, nous sommes en mesure d'identifier la tranche de temps pendant laquelle le trafic est anormal. Cependant, à ce stade, nous ne connaissons pas l'événement réseau spécifique qui a provoqué la détection. En fait, il convient de noter qu'un intervalle de temps anormal peut contenir plusieurs événements

anormaux et qu'un seul événement anormal peut s'étendre sur plusieurs intervalles de temps.

Dans cette phase, nous voulons identifier le flux spécifique responsable de l'anomalie révélée. Plus en détail, dans un premier temps, nous recherchons l'agrégation de trafic spécifique dans laquelle l'anomalie s'est produite. Du fait qu'une anomalie est détectée lorsque $\|\tilde{Y}_t\|^2$ dépasse le seuil ϵ, la méthode d'identification consiste à rechercher l'agrégat particulier qui, s'il était retiré des statistiques précitées, le ramènerait sous le seuil.

On identifie ainsi un élément $y_t n$ du vecteur Y_t qui correspond à un ensemble A de flux IP candidats responsables de l'anomalie détectée. Il est à noter que si l'agrégation du trafic est effectuée au moyen des flux IR, OD et IL, nous ne sommes pas en mesure de déterminer le flux spécifique responsable de l'anomalie révélée. Au contraire, lorsque nous utilisons l'agrégation de trafic aléatoire, nous avons introduit une méthode qui nous permet de détecter correctement les flux spécifiques impliqués dans le trafic anormal.

Pour cette analyse, nous pouvons utiliser les informations stockées dans les matrices de données $\{Y^j\}_{j=1}^d$ qui contiennent d analyse des mêmes données de trafic en utilisant différentes fonctions de hachage. À ce stade, pour chacune de ces matrices, nous pouvons détecter l'intervalle de temps anormal et l'agrégat anormal, de sorte que nous pouvons identifier un élément $\{y_{tn}^j\}_{j=1}^d$ de chaque matrice. Chacun de ces éléments correspond à un agrégat (A_j) de flux IP anormaux candidats. Compte tenu de cela, les adresses IP responsables peuvent être trouvées en évaluant simplement l'intersection de tous ces agrégats $I = \cap_{j=1}^d A_j$.

4.4 Expériences

4.4.1 Base de données KDD CUP 99

Depuis 1999, KDD'99 [7] est l'ensemble de données le plus utilisé pour l'évaluation des méthodes de détection d'anomalies. Cet ensemble de données est préparé par Stolfo et al. [17] et est construit sur la base des données capturées dans le programme d'évaluation DARPA'98 IDS [14]. DARPA'98 est d'environ 4 gigaoctets de données tcpdump brutes (binaires) compressées de 7 semaines de trafic réseau, qui peuvent être traitées en environ 5 millions d'enregistrements de connexion, chacun avec environ 100 octets. Les deux semaines de données de test ont environ 2 millions d'enregistrements de connexion. L'ensemble de données d'entraînement KDD se compose d'environ 4 900 000 vecteurs de connexion unique, dont chacun contient 41 caractéristiques et est étiqueté comme

normal ou comme attaque, avec exactement un type d'attaque spécifique. Les attaques simulées appartiennent à l'une des quatre catégories suivantes :

1. **Attaque par déni de service (DoS) :** est une attaque dans laquelle l'attaquant rend certaines ressources informatiques ou mémoire trop occupées ou trop pleines pour traiter des demandes légitimes, ou refuse aux utilisateurs légitimes l'accès à une machine.
2. **Attaque de l'utilisateur à la racine (U2R) :** est une classe d'exploit dans laquelle l'attaquant commence par accéder à un compte d'utilisateur normal sur le système (peut-être obtenu en reniflant des mots de passe, une attaque par dictionnaire ou une ingénierie sociale) et est capable d'exploiter une vulnérabilité pour obtenir un accès root au système
3. **Attaque à distance vers locale (R2L) :** se produit lorsqu'un attaquant qui a la capacité d'envoyer des paquets à une machine sur un réseau mais qui n'a pas de compte sur cette machine exploite une vulnérabilité pour obtenir un accès local en tant qu'utilisateur de cette machine.
4. **Attaque de sondage :** est une tentative de recueillir des informations sur un réseau d'ordinateurs dans le but apparent de contourner ses contrôles de sécurité.

Il est important de noter que les données de test ne proviennent pas de la même distribution de probabilité que les données d'entraînement et qu'elles incluent des types d'attaques spécifiques qui ne figurent pas dans les données d'entraînement, ce qui rend la tâche plus réaliste. Certains experts en intrusion pensent que la plupart des nouvelles attaques sont des variantes d'attaques connues et que la signature d'attaques connues peut être suffisante pour détecter de nouvelles variantes. Les ensembles de données contiennent un nombre total de 24 types d'attaques d'entraînement, avec 14 types supplémentaires dans les données de test uniquement. Le nom et la description détaillée des types d'attaques d'entraînement sont répertoriés dans [6].

Les fonctionnalités du KDD'99 peuvent être classées en trois groupes :

1. **Caractéristiques de base :** cette catégorie encapsule tous les attributs qui peuvent être extraits d'une connexion TCP/IP. La plupart de ces caractéristiques conduisent à un retard implicite de détection.
2. **Caractéristiques du trafic :** cette catégorie comprend des caractéristiques qui sont calculées par rapport à un intervalle de fenêtre et sont divisées en deux groupes :

— **a) Fonctionnalités «même hôte» :** examinez uniquement les connexions des 2 dernières secondes qui ont le même hôte de destination que la connexion actuelle et calculez les statistiques relatives au comportement du protocole, au service, etc.
— **b) Fonctionnalités «même service» :** examinez uniquement les connexions des 2 dernières secondes qui ont le même service que la connexion actuelle.

Les deux types de caractéristiques de « trafic » susmentionnés sont appelés temporels. Cependant, il existe plusieurs attaques de sonde lente qui analysent les hôtes (ou les ports) en utilisant un intervalle de temps beaucoup plus long que 2 secondes, par exemple, une par minute. En conséquence, ces attaques ne produisent pas de schémas d'intrusion avec une fenêtre temporelle de 2 secondes. Pour résoudre ce problème, les fonctionnalités « même hôte » et « même service » sont recalculées mais en fonction de la fenêtre de connexion de 100 connexions plutôt que d'une fenêtre de temps de 2 secondes. Ces fonctionnalités sont appelées fonctionnalités de trafic basées sur la connexion.

3. **Caractéristiques du contenu :** contrairement à la plupart des attaques DoS et Probing, les attaques R2L et U2R n'ont pas de modèles séquentiels fréquents d'intrusion. C'est parce que les attaques DoS et Probing impliquent de nombreuses connexions à certains hôtes dans un laps de temps très court ; cependant, les attaques R2L et U2R sont intégrées dans les parties de données des paquets et n'impliquent normalement qu'une seule connexion. Pour détecter ces types d'attaques, nous avons besoin de certaines fonctionnalités pour pouvoir rechercher un comportement suspect dans la partie des données, par exemple, le nombre de tentatives de connexion infructueuses. Ces fonctionnalités sont appelées fonctionnalités de contenu.

4.4.2 Résultats

4.4.3 interpretation des resultats

la figure 4.5 represente les resultats de teste de notre projets et qui ont formé 94 pourcent de taux de succes. les points bleu signifie les cas ou les transactions normaux , les points vert sont des ipsweep et les points orange signifie que notre IDS a detecté une menace.

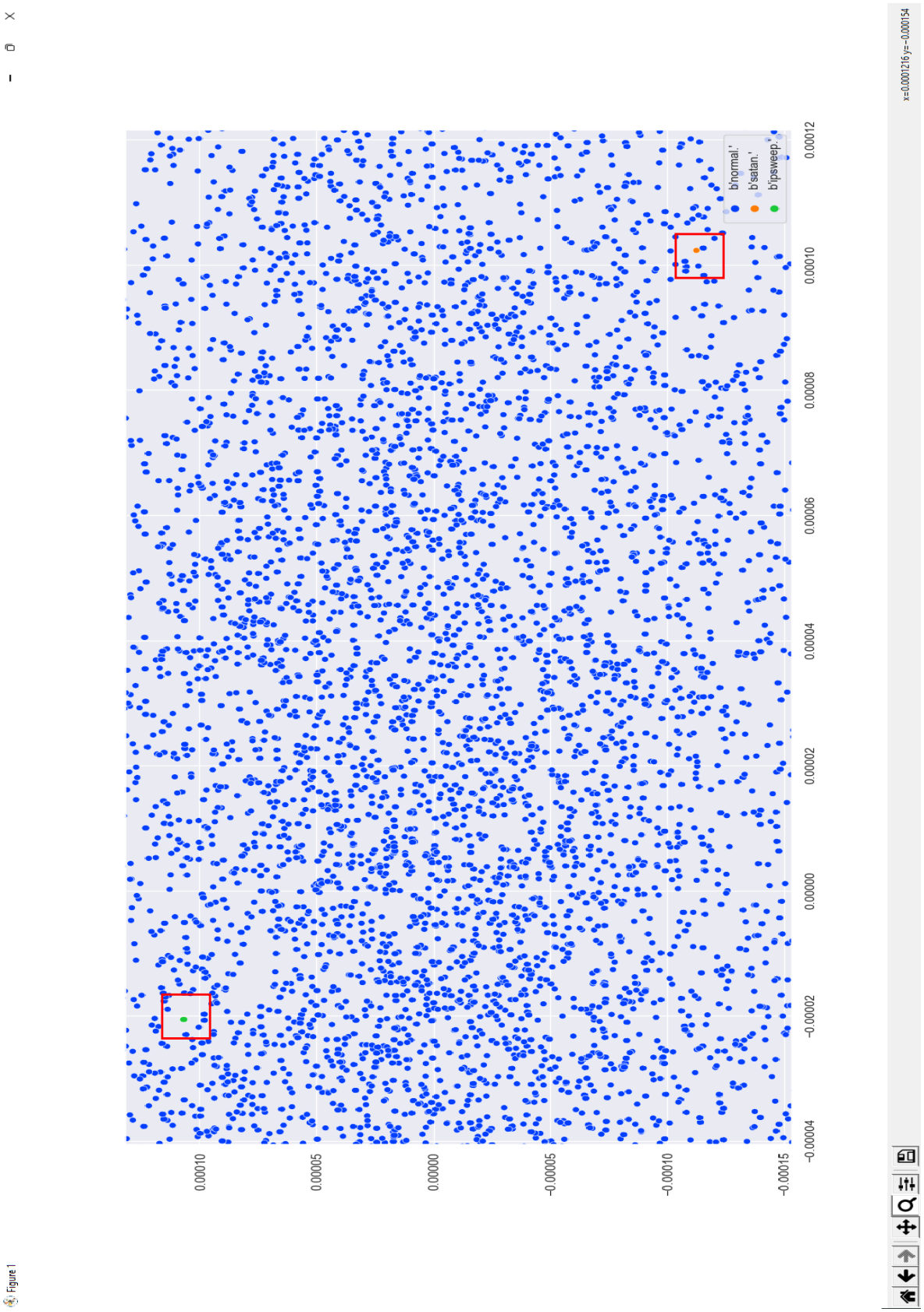

FIGURE 4.5: Visualisation du résultats de la détection.

CHAPITRE 5

CONCLUSION GÉNÉRALE

La venue d'Internet, et son utilisation universellement émergeante, et sa politique démocratique permettant à toute machine d'être connectée au réseau, et à toute personne d'en bénéficier ainsi que de proposer ses propres services. Cette vision d'échange libre de l'information impliquant un accès aussi libre, nous laisse à réfléchir sur la nature de l'information elle-même, sur plusieurs aspects, l'un des plus importants et surement crucial est le degré de confidentialité de cette information. Une confidentialité qui devrait être garantie avec un contrôle rigoureux des accès à cette information, et les opérations engagées. Afin d'empêcher les accès non-autorisés ou les opérations nondéfinies, une politique de sécurité est recommandée, pour cette fin, plusieurs stratégies et protocoles sécuritaires ont été élaborés. Parmi les solutions proposées, la mise en place des systèmes de détection d'intrusions est devenue nécessaire. Cependant, la complexité croissante des attaques nécessite une amélioration de telles techniques de détections.

Les activités d'intrusion peuvent s'exercer au niveau du réseau comme au niveau des machines hôtes, et donc les méthodes et outils de défense devront être engagés à ces mêmes deux niveaux précédents, d'où on parle de systèmes de détection d'intrusion basés réseaux et ceux basés hôtes. Notre projet traite les systèmes de détection basés hôte, qui représente en fait, une couche de défense.

L'IDS est de deux natures, celui qui reconnait l'attaque par un modèle d'action et, celui qui reconnait le comportement de l'activité intrusive. Un comportement est assimilé progressivement, est donc demande des techniques possédant la capacité d'apprentissage, d'où l'implication des algorithmes d'apprentissage dans le processus du design de l'IDS.

Nous avons étudié dans notre projet, la méthode de l'analyse des composent principales basée sur une distance de Kullback Liebler.

Afin de répondre à cet objectif, nous avons introduit les notions mathématiques préliminaires. Nous nous sommes focalisé sur les notions mathématiques directement liées au calcul des méthodes utilisées pour l'objectif de ce mémoire. A savoir la divergence Kullback-Leiber et l'analyse en composante principales.

nous avons aussi introduit la notion des systèmes de détection d'intrusions, avant de donner les définitions relatives à cette notion, nous avons illustré des aspects théoriques comparatifs .Ensuite une classification des IDS sera abordée. Pour entamer ensuite les mesures partielles qui peuvent être utilisées pour tester l'efficacité des IDS.

nous avons aussi introduit la présentation des concepts relatifs à la classification non supervisé avec la méthode de la divergence Kullback-Leiber et l'analyse en composante principales.

Nous avons conclu ce mémoire par une conclusion générale.

BIBLIOGRAPHIE

[1] T. BARNETT AND R. PREISENDORFER, *Origins and levels of monthly and seasonal forecast skill for united states surface air temperatures determined by canonical correlation analysis*, Monthly Weather Review, 115 (1987), pp. 1825–1850.

[2] D. G. CHACHLAKIS, A. PRATER-BENNETTE, AND P. P. MARKOPOULOS, *L1-norm tucker tensor decomposition*, IEEE Access, 7 (2019), pp. 178454–178465.

[3] R.-M. CHEN AND K.-T. HSIEH, *Effective allied network security system based on designed scheme with conditional legitimate probability against distributed network attacks and intrusions*, International Journal of Communication Systems, 25 (2012), pp. 672–688.

[4] B. CLAISE, G. SADASIVAN, V. VALLURI, AND M. DJERNAES, *Cisco systems netflow services export version 9*, (2004).

[5] G. CORMODE AND S. MUTHUKRISHNAN, *An improved data stream summary : the count-min sketch and its applications*, Journal of Algorithms, 55 (2005), pp. 58–75.

[6] R. K. CUNNINGHAM, R. P. LIPPMANN, D. J. FRIED, S. L. GARFINKEL, I. GRAF, K. R. KENDALL, S. E. WEBSTER, D. WYSCHOGROD, AND M. A. ZISSMAN, *Evaluating intrusion detection systems without attacking your friends : The 1998 darpa intrusion detection evaluation*, tech. rep., MASSACHUSETTS INST OF TECH LEXINGTON LINCOLN LAB, 1999.

[7] K. CUP, *Available on : http : / / kdd. ics. uci. edu / databases / kddcup99 / kddcup99. html*, 2007.

[8] Y. DJEMAIEL, S. REKHIS, AND N. BOUDRIGA, *Intrusion detection and tolerance : A global scheme*, International Journal of Communication Systems, 21 (2008), pp. 211–230.

[9] D. HSU, S. M. KAKADE, AND T. ZHANG, *A spectral algorithm for learning hidden markov models*, Journal of Computer and System Sciences, 78 (2012), pp. 1460–1480.

[10] Q. KE AND T. KANADE, *Robust l/sub 1/norm factorization in the presence of outliers and missing data by alternative convex programming*, in 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), vol. 1, IEEE, 2005, pp. 739–746.

[11] A. LAKHINA, M. CROVELLA, AND C. DIOT, *Characterization of network-wide anomalies in traffic flows*, in Proceedings of the 4th ACM SIGCOMM conference on Internet measurement, 2004, pp. 201–206.

[12] A. LAKHINA, M. CROVELLA, AND C. DIOT, *Diagnosing network-wide traffic anomalies*, ACM SIGCOMM computer communication review, 34 (2004), pp. 219–230.

[13] ——, *Mining anomalies using traffic feature distributions*, ACM SIGCOMM computer communication review, 35 (2005), pp. 217–228.

[14] R. P. LIPPMANN, D. J. FRIED, I. GRAF, J. W. HAINES, K. R. KENDALL, D. MCCLUNG, D. WEBER, S. E. WEBSTER, D. WYSCHOGROD, R. K. CUNNINGHAM, ET AL., *Evaluating intrusion detection systems : The 1998 darpa off-line intrusion detection evaluation*, in Proceedings DARPA Information Survivability Conference and Exposition. DISCEX'00, vol. 2, IEEE, 2000, pp. 12–26.

[15] P. P. MARKOPOULOS, G. N. KARYSTINOS, AND D. A. PADOS, *Optimal algorithms for l_{1}-subspace signal processing*, IEEE Transactions on Signal Processing, 62 (2014), pp. 5046–5058.

[16] P. P. MARKOPOULOS, S. KUNDU, S. CHAMADIA, AND D. A. PADOS, *Efficient l1-norm principal-component analysis via bit flipping*, IEEE Transactions on Signal Processing, 65 (2017), pp. 4252–4264.

[17] S. J. STOLFO, W. FAN, W. LEE, A. PRODROMIDIS, AND P. K. CHAN, *Cost-based modeling for fraud and intrusion detection : Results from the jam project*, in Proceedings DARPA Information Survivability Conference and Exposition. DISCEX'00, vol. 2, IEEE, 2000, pp. 130–144.

[18] B. SUN, K. WU, Y. XIAO, AND R. WANG, *Integration of mobility and intrusion detection for wireless ad hoc networks*, International Journal of Communication Systems, 20 (2007), pp. 695–721.

[19] M. THORUP AND Y. ZHANG, *Tabulation based 4-universal hashing with applications to second moment estimation.*, in SODA, vol. 4, 2004, pp. 615–624.

[20] J. VYKOPAL, *Flow-based intrusion detection in large and high-speed networks*, PhD thesis, PhD thesis, 2010.