Vers l'évaluation globale des classifieurs bayésiens pour la détection d'intrusions

Salem Benferhat, Tayeb Kenaza

Vers une évaluation globale des classifieurs Bayésiens pour la détection d'intrusion

Salem Benferhat, Tayeb Kenaza

Université Lille-Nord de France,
Artois, F-62307 Lens, CRIL, F-62307 Lens
CNRS UMR 8188, F-62307 Lens
{benferhat,kenaza}@cril.univ-artois.fr

RÉSUMÉ. *Cet article propose plusieurs critères d'évaluation d'un ensemble de classifieurs, illustrés par des réseaux Bayésiens naïfs, dans le contexte de la détection d'intrusion dans un réseau informatique. Chaque réseau Bayésien naïf codera les connaissances disponibles pour la détection d'un comportement anormal. Le but de ce travail est alors d'évaluer globalement les performances de ces classifieurs Bayésiens pour l'analyse simultanée d'un ensemble de connexions qui peuvent contenir plusieurs événements anormaux. Nous illustrons nos critères d'évaluation par une étude expérimentale pour la détection des attaques sévères dans le cadre du projet PLACID*[1].

ABSTRACT. *This paper proposes several criteria for evaluating a set of classifiers, illustrated by naive Bayesian classifiers, in the context of network intrusion detection. Each naive Bayes will encode available knowledge for the detection of an abnormal behavior. The aim of this work is then to assess the overall performance of Bayesian classifiers for a simultaneous analysis of a set of connections that may contain several anomalous events. We illustrate our evaluation criteria with a case study for the detection of severe attacks within the PLACID project.*

MOTS-CLÉS : *Détection d'attaques sévères, classifieurs Bayésiens naïfs, évaluation de classifieurs*

KEYWORDS: *Detection of severe attacks, naive Bayes classifiers, evaluation of classifiers*

1. Probabilistic graphical models and Logics for Alarm Correlation in Intrusion Detection, http ://placid.insa-rouen.fr/

1. Introduction

Cet article concerne le problème d'évaluation d'un ensemble de classifieurs. Chaque classifieur représente les connaissances disponibles concernant un problème de décision binaire donné. Par exemple pour un problème de diagnostic de voitures, chaque classifieur est spécialisé dans le diagnostic d'une composante importante (électricité, mécanique, carrosserie, etc.).

Dans la littérature, le critère dit *pourcentage de classification correcte* (PCC) est largement utilisé pour l'évaluation d'un classifieur individuel. Cet article propose plusieurs méthodes pour évaluer globalement un ensemble de classifieurs en combinant les PCC individuels associés à chacun des classifieurs.

Nous illustrons nos méthodes d'évaluation dans le cadre d'une modélisation du problème de la corrélation d'alertes basée sur les réseaux Bayésiens (Benferhat *et al.,* 2008) dans le cadre du projet PLACID. Dans notre modélisation, le problème de la corrélation d'alertes est considéré comme un problème de classification, qui est un problème d'inférence particulier, où parmi les n variables du problèmes, (n-1) variables sont observables. Les variables observées représenteront des alertes de sévérité faible ou moyenne. La $n^{\text{ème}}$ variable, appelée *classe*, est non-observable et que l'on cherche à estimer. Dans notre cas, le domaine de la variable *classe* contient le nom de l'alerte sévère qu'on cherche à estimer ainsi que l'instance "*non attaque*" Le but de la corrélation d'alertes est de détecter des attaques sévères en se basant seulement sur les alertes de faible sévérité. Dans certaines applications, les alertes sévères ne sont pas isolées et peuvent être préparées par des alertes de faible sévérité. Ces dernières alertes peuvent être vues comme des actions qui doivent être exécutées avant de réaliser ces attaques sévères. Notre but est de détecter les attaques sévères les plus plausibles et les actions qui contribuent à leur exécution. Les actions qui ne contribuent pas à la présence des attaques sévères seront considérées comme alertes non pertinentes.
Pour mieux évaluer notre étude de cas, nous proposons un ensemble de mesures qui permettent de donner une évaluation globale de plusieurs classifieurs (Bayésiens ou autres) utilisés simultanément dans le cadre de la détection d'intrusion. Chaque réseau Bayésien naïf représente la connaissance nécessaire pour la détection d'une alerte de sévérité élevé donnée.

Le reste de cet article est organisé comme suit. La section 2 présente des rappels sur la détection d'intrusion et la corrélation d'alertes. La section 3 présente des mesures pour évaluer globalement un ensemble de classifieurs. La section 4 présente une étude de cas pour la détection des attaques sévères. La dernière section conclut l'article.

2. Détection d'intrusion et corrélation d'alertes

Dans cette section, nous rappelons brièvement la détection d'intrusion et la corrélation d'alertes qui seront utilisées pour illustrer les critères d'évaluation d'un ensemble de classifieurs.

2.1. *Détection d'intrusion*

La sécurité informatique englobe tous les mécanismes et moyens matériels, logiciels et organisationnels mis en œuvre dans le but de garantir la confidentialité, l'intégrité et la disponibilité des informations et services (Anderson, 1972). Une intrusion est une tentative délibérée et non autorisée d'accéder ou de manipuler des informations ou rendre un système non opérationnel ou inaccessible. Les systèmes de détection d'intrusion collectent des traces d'audit (paquets réseaux, logs systèmes ou applicatifs, etc.) pour les analyser, en temps réel ou en différé, en vue de détecter toute activité suspecte ou malveillante, d'origines internes ou externes, et lever une alerte. Les principales approches de détection sont l'approche comportementale et l'approche par scénarios.
L'approche comportementale (Evangelista, 2004) se base sur les modèles et profils des comportements normaux, et toute déviation est interprétée comme une *éventuelle* intrusion. Il suffit alors d'élaborer des modèles et profils pour les activités normales et avoir un mécanisme permettant de comparer les activités courantes aux modèles/profils établis pour détecter des intrusions. L'approche comportementale peut détecter de nouvelles intrusions mais tout écart, même normal mais nouveau, sera considéré intrusif, d'où le taux de fausses alertes trop important pour déployer les systèmes comportementaux dans la pratique.
L'approche par scénarios (Evangelista, 2004) se base, quant à elle, sur les scénarios et signatures des attaques existantes. La détection revient alors à la recherche dans les traces d'audit de signatures d'attaques déjà connues. Cette approche, bien que très efficace dans la détection des intrusions déjà répertoriées, ne peut pas détecter les nouvelles attaques et certaines variantes d'attaques connues.

2.2. *Corrélation d'alertes*

La corrélation d'alertes consiste à rechercher des relations entre les alertes dans le but de réduire leur volume ou de détecter des attaques coordonnées. Elle a été étudiée ces dernières années par plusieurs chercheurs et plusieurs approches de corrélation ont été développées. Nous distinguons deux principaux objectifs des approches développées :

– **La réduction du volume d'alertes :** le but des approches de cette catégorie est la réduction du volume d'alertes. Valdes et Skinner (Valdes *et al.*, 2001) ont défini des mesures de similarité entre des attributs tels que : le type d'attaque, les adresses source et cible, l'identité d'utilisateur, le temps de détection, etc. Ensuite, ces mesures locales de similarité sont fusionnées afin de définir une mesure globale de similarité entre les alertes. S'il n'y a aucune méta-alerte[1] qui soit suffisamment similaire à une nouvelle alerte, alors une nouvelle méta-alerte est créée et ajoutée à la liste des méta-alertes. Sinon, la nouvelle alerte est fusionnée avec la méta-alerte adéquate (la plus similaire à cette nouvelle alerte). Une autre approche semblable a été proposée par

1. Une méta-alerte est une fusion d'un groupe d'alertes ayant des caractéristiques communes.

Cuppens (Cuppens, 2001) et par Dain (Dain *et al.,* 2001). Debar et Wespi (Debar *et al.,* 2001) ont proposé une solution pour l'agrégation et la corrélation d'alertes qui est mise en application dans l'outil commercial "Risk Manager". Julish (Julisch, 2001) a proposé d'utiliser un mécanisme de fouille de données connu sous le nom "AOI" (Attribute-Oriented Induction) pour regrouper les alertes en clusters.

– **La détection des attaques coordonnées :** le but des approches de cette catégorie est de rechercher les relations entre les alertes pour établir des scénarios d'attaque. Certaines approches utilisent les pré-conditions et les post-conditions des actions pour construire implicitement des scénarios d'attaque (Cuppens *et al.,* 2002, Steven *et al.,* 2000, Ning *et al.,* 2002). D'autres approches, tout simplement, introduisent la description des scénarios dans le système (Dain *et al.,* 2001). Les graphes ont été utilisés aussi pour cet objectif (Wang *et al.,* 2006, Wang *et al.,* 2008).

3. Des métriques pour l'évaluation des classifieurs multiples

Dans cette section, nous proposons plusieurs métriques permettant de donner une évaluation globale d'un système composé de plusieurs classifieurs. Supposons que nous avons un ensemble de N classifieurs binaires, où pour le $i^{\text{ème}}$ classifieur le résultat de la classification est soit c_i, soit c_0 (la classe c_0 est la négation ou l'absence de c_i). Par exemple, supposons que dans un garage automobiliste on dispose de 5 ateliers (Électricité, Pneumatique, Transmission, Système de freinage, Moteur.). Chaque atelier est vu comme un classifieur binaire qui indique s'il y a un dysfonctionnement ou non dans une partie du véhicule.

Nous supposons dans cet article que les classifiuers sont représentés par des réseaux Bayésiens naïfs. (Shachter *et al.,* 1992) qui sont la forme la plus simple des réseaux Bayésiens. Ils se composent d'un graphe avec un seul parent et plusieurs nœuds feuilles, avec une forte hypothèse d'indépendance entre les feuilles dans le contexte de leur parent (Figure 1).

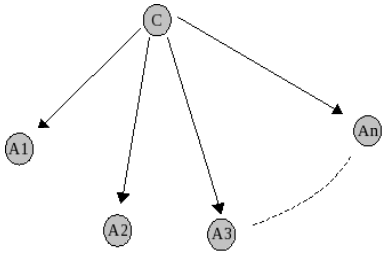

Figure 1. *Exemple de réseau Bayésien naïf*

En présence d'un individu (dans notre exemple véhicule) à diagnostiquer, chaque réseau Bayésien naïf (associé à chaque atelier) infère la probabilité (a posteriori) de présence/absence d'un dysfonctionnement. Selon le vrai état de l'individu nous dis-

tinguons 4 situations qui correspondent aux vrai positif (VP), vrai négatif (VN), faux positif (FP) et faux négatif (FN).

L'évaluation de l'efficacité de la classification est souvent (et traditionnellement) basée sur le *Pourcentage de Classification Correcte* (PCC) qui est défini comme suit :

$$PCC = \frac{VP+VN}{VP+VN+FP+FN} \quad [1]$$
$$= \frac{\text{Nombre d'individus bien classés}}{\text{Nombre total des individus}}$$

Le PCC a été largement utilisé pour évaluer un classifieur donné. Regardons maintenant comment évaluer globalement un ensemble des classifieurs. Pour cela, un individu sera représenté par un vecteur $W_r = (r_1, ..., r_n)$, où r_i = 1 (respectivement 0) signifie que W contient (respectivement ne contient pas) la classe c_i. Dans notre exemple, un individu est un véhicule qui sera représenté par un vecteur W_r. Quand $W_r = (0, 1, 0, 1, 0)$ cela signifie que le véhicule en question a un problème au niveau de la pneumatique et au niveau du système de freinage.

Soit $W_p = (p_1, ..., p_n)$ le résultat retourné par les classifieurs. Lorsque $p_i = 1$ (respectivement 0), ceci signifie que le $i^{\text{ème}}$ classifieur considère qu'il y a (respectivement, il n'y a pas) la classe c_i (*dans notre exemple, c_i représente un dysfonctionnement analysé par un atelier i*).

La question maintenant est comment proposer des mesures pour évaluer si W_r est correctement classé ou non ?

Soit T une base de test qui servira pour l'évaluation et $M = |T|$ le nombre d'élément dans cette base. La première mesure, et la plus simple, est définie par la moyenne des différents $PCC[i]$ obtenus par l'évaluation individuelle de chaque classifieur. Cette première mesure, notée PCC_1, est définie comme suit :

***Définition* 1.** *(Moyenne des PCC). Soit $PCC[i]$ le pourcentage de classification correcte du classifieur i.*

$$PCC_1 = \frac{\sum_{i=1}^{n} PCC[i]}{N}$$

Une autre façon pour évaluer ces classifieurs est de considérer uniquement la concordance exacte entre W_r et W_p. Nous définissons PCC_2 comme suit :

***Définition* 2.** *(Appariement exacte)*

$$PCC_2 = \frac{\sum_{i=1}^{M} S_i}{M}, avec$$

$$S_i = \begin{cases} 1\ si\ \forall j = 1, ..n,\ W_p^i[j] = W_r^i[j] \\ 0, sinon, \end{cases} \quad [2]$$

Où $W_p^i[j]$ désigne le résultat prédit par le classifieur j sur l'individu i de la base de test. $W_r^i[j]$ représente le résultat réel concernant la classe c_j dans le $i^{ème}$ élément de la base de test T.

Cette métrique est très stricte car elle nécessite une concordance exacte entre le vecteur réel et le vecteur prédit. Pour un individu donné, non seulement les classifieurs doivent détecter les classes concernées par cet individu, mais également aucun faux positif ne doit être généré.

Dans notre exemple, lorsque $W_r = (0, 1, 0, 1, 0)$ cela signifie que les ateliers pneumatique et système de freinage doivent absolument détecter le dysfonctionnement et les autres atelier doivent signaler un état normal du véhicule. Dans le cas contraire, l'individu est considéré mal classé.

Une troisième métrique serai de se focaliser uniquement sur les cas où il y a une classe réelle ou prédite. Nous définissons la nouvelle métrique, notée PCC_3, comme suit :

***Définition* 3.** *(Appariement restreint avec classes prédites ou réelles)*

$$PCC_3 = \frac{\sum_{i=1}^{M} S_i}{M},\ avec$$

$$S_i = \begin{cases} 1\ si\ \forall j = 1, ..n,\ W_p^i[j] = W_r^i[j] = 0 \\ \sum_{j=1}^{n}(W_r^i[j] = W_p^i[j] = 1) / \sum_{j=1}^{n}(W_r^i[j] = 1\ ou\ W_p^i[j] = 1),\ sinon. \end{cases}$$

Avec la métrique PCC_3, nous distinguons trois situations. La première situation concerne le cas où il y a une concordance exacte entre le vecteur réel et le vecteur prédit. Dans ce cas, nous affectons la valeur 1 à l'individu. La deuxième situation est lorsque le vecteur réel ne concerne aucune classe et au moins un classifieur prédit une classe. Dans ce cas, nous affectons la valeur 0 à l'individu. La dernière situation est lorsque le système détecte réellement des classes et produit en même temps du faux (positive ou négatif). Dans ce cas, le système sera pénalisé proportionnellement aux faux, positifs ou négatifs, produits.

Dans notre exemple, supposons que $W_r = (0, 0, 1, 0, 0)$. C'est-à-dire que l'individu a un problème au niveau de la transmission. Supposons que le vecteur prédit par les classifieurs est $W_p = (1, 0, 1, 0, 1)$. C'est-à-dire que le diagnostic découvre que le véhicule a des problèmes au niveau de la pneumatique, la transmission et le moteur. Notons qu'en utilisant la métrique de concordance exacte nous obtenons $PCC_2 = 0$. Maintenant, si nous utilisons PCC_3, nous obtenons $PCC_3 = \frac{1}{3}$ pour cet individu.

La dernière métrique décrite par PCC_4 considère qu'un ensemble de classifieurs est totalement satisfaisant dès qu'un classifieur détecte correctement un dysfonctionnement. Elle est définie comme suit :

***Définition* 4.** *(Interprétation disjonction des PCC)*

$$PCC_4 = \frac{\sum_{i=1}^{M} S_i}{M},\ avec$$

$$S_i = \begin{cases} 1\ si\ (\forall j, W_r^i[j] = W_p^i[j] = 0)\ ou\ (\exists j,\ W_r^i[j] = W_p^i[j] = 1) \\ 0,\ sinon. \end{cases}$$

Avec la métrique PCC_4 nous distinguons deux situations. La première situation est lorsqu'un vecteur réel ne contient aucune classe et aucun des classifieurs ne produit du faux positif. Dans ce cas, nous affectons la valeur 1 à l'individu (c'est-à-dire, cette individu est correctement classé). La deuxième situation est lorsque le système détecte des classes. Dans ce cas nous affectons la valeur 1 à cette individu si au moins un des classifieurs détecte réellement une classe, sinon nous affectons la valeur 0 à cet individu.

4. Application à la détection d'attaque sévères

Dans cette section, nous présentons un cas d'étude dans le cadre de la détection d'intrusion en sécurité informatique. Le but est de fournir un mécanisme permettant aux opérateurs de sécurité de prédire les alertes de sévérité élevée. Plus précisément, le mécanisme proposé permettra aux opérateurs de sécurité de se concentrer directement sur les alertes sévères et seulement sur les alertes (de faible sévérité) qui contribuent à réaliser ces attaques sévères.

4.1. *Présentation du problème*

Les SDI peuvent assigner une sévérité à une action, qui représente son impact sur les systèmes. Certaines actions cherchent simplement à collecter des informations sur les systèmes comme par exemple les actions de type "Probe". D'autres actions changent les systèmes avec plusieurs niveaux de sévérité (habituellement faible, moyen et élevé). Les actions de faible et moyenne sévérité peuvent changer les systèmes d'information sans vraiment compromettre leur sécurité. Cependant, en présence des actions sévères il y a une forte probabilité que la sécurité des systèmes d'information soient compromis. Dans ce qui suit, nous définissons l'échelle de niveau de sévérité comme :

sévérité = {faible, moyenne et élevée}.

Les actions de faible et moyenne sévérité peuvent avoir une influence sur les actions sévères. Les actions sévères doivent être présentées aux opérateurs de sécurité. Mais, des alertes additionnelles qui sont liées aux alertes sévères devraient être présentées également, ne serait-ce que pour un but de diagnostic.

Nous définissons les actions qui contribuent à réaliser des attaques sévères comme un ensemble $S = \{A_1, A_2, ..., A_n, H\}$, où Ai sont des instances d'actions et H est une action sévère tel que : A_i a une influence positive sur H.

4.2. *Approche Bayésienne*

Le but de notre approche est d'apprendre, à partir de l'historique des observations, un réseau Bayésien naïf quantifiant les relations entre les alertes qui contribuent à compromettre des objectifs d'intrusion. Le réseau Bayésien détermine le niveau d'influence de chaque action sur les objectifs d'intrusions en utilisant les distributions de probabilités conditionnelles calculées à partir de l'historique des observations. Une fois les distributions de probabilités des différents nœuds du réseau Bayésien calculées, ce modèle peut être utilisé pour prévoir si un objectif d'intrusion peut être atteint ou non, selon l'observation des alertes rapportées par les SDI.

Notre approche comprend deux étapes principales :

1) **Prétraitement des données :** Cette étape transforme un ensemble d'alertes en un ensemble de données formatées, qui sera utilisé pour l'apprentissage de réseau Bayésien naïf. En effet, les donnée d'entrée de notre approche est un ensemble d'alertes, tandis que dans le problème de classification l'entrée est une table (par exemple en format CSV), où la dernière colonne contient la classe et les autres colonnes représentent des variables observables. Par conséquent, nous devons transformer l'ensemble des alertes dans un tableau à partir duquel un réseau de Bayésien naïf peut être facilement appris.

2) **Apprentissage des réseaux Bayésiens naïfs et prédiction des objectifs d'intrusion :** Dans cette étape, nous calculons les distributions de probabilités conditionnelles de chaque variable nœud dans le contexte de la variable *classe* (parent). Ensuite, nous prédirons les objectifs d'intrusion par l'application des mécanismes d'inférence des réseaux Bayésiens.

4.3. *Étude expérimentale*

4.3.1. *Les données d'expérimentation*

Les données de test sont collectées en surveillant un réseau académique pendent 3 mois, en utilisant le système de détection d'intrusion "Snort". Ces données contiennent plus d'un million d'alertes. Ce volume important d'alertes confirme le problème d'inondation d'alerte qui rend les opérateurs de sécurité incapables d'analyser toutes les alertes. La plupart des alertes rapportées ne représentent pas une vraie menace

pour le système surveillé (attaque avec une faible sévérité), mais les opérateurs de sécurité ne peuvent simplement pas analyser les attaques sévères et ignorer le reste. Notre but est de montrer que certaines attaques sévères peuvent être préparées par des attaques de faible sévérité, puis nous emploierons cette propriété pour réduire le volume d'alertes et présenter aux opérateurs de sécurité un nombre réduit d'alertes qui rapportent des attaques pertinentes.

Les données contiennent des alertes de sévérité faible (62.95%), moyenne (31.07%) et élevée (5.98%). Notons que les actions de faible et moyenne sévérité représentent plus de 94% du volume des alertes rapportées. Ces données contiennent plus de 171 types d'attaque et plus de 400 machines surveillées. Notons que certaines attaques sont fréquemment rapportées (comme par exemple les 7 attaques que nous avons sélectionnées dans la section suivante), alors qu'il y a d'autres attaques qui sont très rares. Nous avons noté également que certaines machines sont concernées uniquement par quelques attaques sévères alors que d'autres machines ne sont concernées par aucune attaque sévère.

Ces données contiennent deux principales difficultés. La première est le volume important des alertes rapportées, plus d'un million, qui rend leur gestion très difficile. La deuxième est le taux élevé de fausses alertes. Pour réduire les fausses alertes, nous avons supprimé certaines alertes générées par les préprocesseurs de Snort. Ces alertes peuvent être évitées si la configuration par défaut n'est pas utilisée. Par exemple, le préprocesseur "http_inspect", qui vérifie la conformité du trafic Web aux protocoles HTTP, a généré plus de 40% des alertes. Ceci est dû au fait que les opérateurs de sécurité utilisent les outils avec la configuration par défaut. Nous avons utilisé uniquement un mois de données ce qui est suffisant pour la phase d'apprentissage.

4.3.2. *Évaluation des résultats expérimentaux*

Dans cette section, nous présentons les résultats de la détection des attaques sévères contenues dans les données de test en utilisant les différents réseaux Bayésiens naïfs appris avec les données d'apprentissage. Parmi les attaques sévères rapportées, nous avons sélectionné les 6 attaques du Tableau 1 pour l'analyse. Notre choix est basé sur plusieurs critères tels que la répartition des observations dans le temps et leur fréquence (les attaques rares sont difficiles à analyser).

1) Évaluation individuelle

Les données de test seront d'abord formatées pour construire un ensemble de fenêtres (individus). Le Tableau 2 donne les *PCC* individuels des réseaux Bayésiens associés aux attaques sévères. La classification individuelle est réalisée dans le sens suivant : soit une fenêtre de test W qui contient un certain nombre d'actions et une classe d'attaque sévères qui peut contenir la valeur 0 (ce qui signifie qu'aucune attaque sévère n'est présente dans la fenêtre de test) ou une valeur $i \in \{1, 2, 3, 4, 5, 6\}$, indiquant l'identifiant de l'attaque sévère présente dans W. Pour chaque réseau Bayésien naïf RB_j associé à une attaque sévère j, nous considérons que RB_j a correctement classé W si la prédiction de RB_j correspond à l'attaque réelle présente dans W. Le Tableau 2 montre que la majorité des réseaux Bayésiens naïfs détectent correctement les at-

Snort-ID	Attaques sévères	Nombre (1 mois)
966	WEB-FRONTPAGE .... request	17
997	WEB-IIS asp-dot attempt	23
1002	WEB-IIS cmd.exe access	42
2436	WEB-CLIENT Microsoft wmf metafile access	140
5715	WEB-MISC malformed ipv6 uri overflow attempt	14
8734	WEB-PHP Pajax arbitrary command execution attempt	14

Tableau 1. *Les attaques sévère sélectionnées. (Les réseaux Bayésiens construits et associés à ces attaques sont présentés dans la Figure 1)*

taques sévères. Ces résultats ont été confirmés par une étude récente basée également sur les réseaux bayésiens dans le cadre du projet Placid (Tabia *et al.*, 2010).

	Attaques sévères	PCC
1	WEB-FRONTPAGE .... request	95,79%
2	WEB-IIS asp-dot attempt	83,59%
3	WEB-IIS cmd.exe access	99,97%
4	WEB-MISC cross site scripting attempt	92,15%
5	WEB-CLIENT Microsoft wmf metafile access	97,28%
6	WEB-MISC malformed ipv6 uri overflow attempt	84,25%
7	WEB-PHP Pajax arbitrary command execution attempt	90,87%

Tableau 2. *Le PCC des attaques surveillées*

2) Évaluation globale

Dans la section précédente, nous avons évalué les 6 réseaux Bayésiens indépendamment des uns des autres. Dans cette section, nous nous sommes intéressés à une évaluation globale de notre système. Le Tableau 3 montre l'évaluation de notre approche en utilisant les différentes métriques définies dans la section 3. Comme il est attendu, PCC_2 est le plus sévère puisqu'il nécessite une concordance exacte. Les résultats sont très satisfaisants, par exemple avec PCC_4, plus de 90% des fenêtres testées sont correctement classées. Ces résultats confirment que la plupart des attaques sévères ont pu être prédites à partir des observations des alertes de faible sévérité. Ces résultats montre que notre approche permet de surveiller et prédire les évènements anormaux.

PCC_1	$93,44\%$
PCC_2	$85,96\%$
PCC_3	$88,04\%$
PCC_4	$90,11\%$

Tableau 3. *Évaluation globale de la détection des attaques sévères*

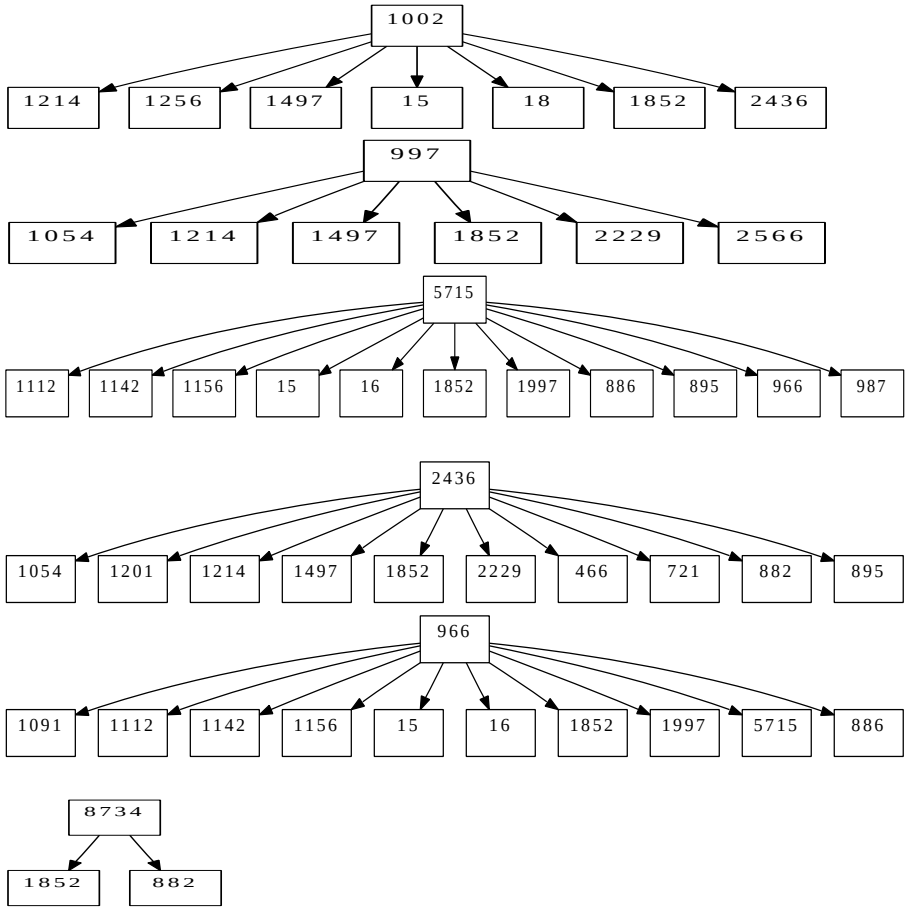

Figure 2. *Les réseaux Bayésiens des attaques surveillées. Le numéro du nœud correspond à l'identificateur de l'alerte utilisé par "Snort"*

5. Conclusion

Dans cet article, nous avons proposé plusieurs métriques qui évaluent globalement un ensemble de classifieurs. Ces métriques sont appropriées pour évaluer des systèmes de détection d'intrusion qui analysent des groupes de connexions par exemple (au lieu d'une connexion à la fois).

Nous avons en particulier illustré ces métriques dans le cadre de la détection des attaques sévères basée sur les réseaux Bayésiens naïfs.

Remerciements

Ce travail a été soutenu partiellement par le projet national PLACID. Les auteurs voudraient remercier les membres du SSIR [2] et LINA [3] pour leurs aides.

6. Bibliographie

Anderson J. P., Computer Security Technology Planning Study, Technical Report n ESD-TR-73-51, Vol. II, Electronic Systems Division, Air Force Systems Command, Bedford, MA 01730, 1972.

Benferhat S., Kenaza T., Mokhtari A., « A Naive Bayes Approach for Detecting Coordinated Attacks », *32rd IEEE International Workshop on Security, Trust, and Privacy for Software Applications (STPSA'08)*, p. 704-709, 2008.

Cuppens F., « Managing Alerts in a Multi-Intrusion Detection Environment », *ACSAC'01 : Proceedings of the 17th Annual Computer Security Applications Conference*, p. 22-31, 2001.

Cuppens F., Miège A., « Alert Correlation in a Cooperative Intrusion Detection Framework », *IEEE Symposium on Security and Privacy*, p. 202-215, 2002.

Dain O., Cunningham R. K., « Fusing a heterogeneous alert stream into scenario », *ACM Workshop on Data Mining for Security Application*, p. 1-13, 2001.

Debar H., Wespi A., « Aggregation and Correlation of Intrusion-Detection Alerts », *Recent Advances in Intrusion Detection*, p. 85-103, 2001.

Evangelista T., *Les IDS - Les systèmes de détection d'intrusions informatiques*, Dunod, 2004.

Julisch K., « Mining Alarm Clusters to Improve Alarm Handling Efficiency », *ACSAC'01 : Proceedings of the 12th Annual Computer Security Applications Conference*, p. 12-21, 2001.

Ning P., Cui Y., Reeves D. S., « Analyzing Intensive Intrusion Alerts via Correlation », *Recent Advances in Intrusion Detection*, p. 74-94, 2002.

Shachter R. D., Peot M. A., « Decision Making Using Probabilistic Inference Methods », *Uncertainty in Artificial Intelligence*, p. 276-283, 1992.

Steven J. T., Karm L., « A requires/provides model for computer attacks », *New Security Paradigms Workshop*, p. 31-38, 2000.

Tabia K., Leray P., Mé L., « From redundant/irrelevant alert elimination to handling IDSs' reliability and controlling severe attack prediction/false alarm rate tradeoffs », *The Fifth Conference on Network and Information Systems Security (SARSSI 2010, Nice, France, Mai 2010. (To appear)*, 2010.

Valdes A., Skinner K., « Probabilistic Alert Correlation », *Recent Advances in Intrusion Detection*, p. 54-68, 2001.

Wang L., Chao Y. C., Singhal A., Jajodia S., « Implementing interactive analysis of attack graphs using relational databases », *Journal of Computer Security*, vol. 16(4), p. 419-437, 2008.

Wang L., Liu A., Jajodia S., « Using graph for Correlating, Hypothesizing, and Predicting Intrusion Alerts », *Journal of Computer Communication*, 2006.

2. (Sécurité des réseaux et des systèmes d'information) équipe de Supélec, à Rennes
3. (Laboratoire d'Informatique de Nantes Atlantique