

Réseaux Bayésiens naïfs augmentés pour la
détection des attaques coordonnées

—
Délivrable n°12

Salem BENFERHAT
CRIL

Tayeb KENZA
CRIL et Ecole Militaire Polytechnique d'Alger

Philippe LERAY
LINA

Réseaux Bayésiens naïfs augmentés pour la détection des attaques coordonnées

Salem Benferhat[†] et Tayeb Kenaza[‡] et Philippe Leray[§]

La corrélation d’alertes est un mécanisme indispensable pour la réduction du volume important des alertes et pour la détection des attaques coordonnées et complexes. Les approches existantes soit se basent sur des connaissances d’experts, soit utilisent des mesures de similarité simples qui ne permettent pas de détecter des attaques complexes. Elles souffrent également d’une complexité de calcul très élevée due, par exemple, à un grand nombre de scénarios possibles pour détecter une attaque coordonnée. Dans cet article, nous proposons une nouvelle modélisation des problèmes de la corrélation d’alertes basée sur les réseaux Bayésiens naïfs augmentés. Notre modélisation implique une légère contribution des connaissances d’experts. Elle tire profit des données disponibles et fournit des algorithmes efficaces pour la détection et la prédiction des scénarios d’attaques. Nous illustrerons notre modélisation avec un cas d’étude qui montre comment prévoir des attaques coordonnées. Ce cas d’étude est réalisé sur les données de test DARPA’2000.

Mots-clés: corrélation d’alertes, prédiction des attaques, réseaux Bayésiens naïfs augmentés

1 Introduction

Dans un environnement où les technologies de l’information progressent intensivement et la complexité des attaques informatiques augmente de plus en plus, les systèmes de détection d’intrusion (SDI) jouent un rôle important pour la sécurité informatique. En effet, les SDI sont largement employés dans les systèmes d’information pour rapporter des anomalies et se protéger contre des activités malveillantes.

Nous distinguons deux principales méthodes de détection d’intrusion: méthode par signature et méthode comportementale [HDw99]. Les méthodes de détection par signature comparent les événements avec des signatures prédéfinies et produisent des alertes lorsque des signatures sont trouvées. Cette approche a l’avantage de détecter les attaques connues mais elle ne peut pas détecter des nouvelles attaques. Les méthodes de détection comportementales apprennent (ou modélisent) le comportement normal du système et produisent des alertes lorsque une déviation de la normalité est observée. Ces méthodes ont l’avantage de détecter des attaques nouvelles mais elles produisent beaucoup de faux positif.

Les SDI traditionnels se concentrent habituellement sur la détection des attaques élémentaires. Ils traitent les alertes indépendamment sans tenir compte des relations qui peuvent exister entre elles. Le résultat des SDI est généralement un ensemble d’alertes qui rapportent des attaques élémentaires.

Dans certaines situations, des intrus peuvent utiliser des attaques complexes pour atteindre leurs objectifs. Souvent, ils effectuent une série d’actions (attaques élémentaires) dans une séquence bien définie, appelée “scénario” ou “plan d’attaque”. La plupart de ces actions sont signalées par les SDI, mais les relations logiques entre ces actions ne sont pas détectées par les SDI.

En plus, les opérateurs de sécurité traitent un volume important d’alertes avec une certaine incertitude que ces alertes correspondent à de vraies attaques ou pas, et si elles correspondent à des actions isolées ou elles appartiennent à un scénario complexe. Dans une telle situation, le but de la corrélation d’alertes est de

[†] Centre de Recherche en Informatique de Lens (CNRS-UMR 8188), Université d’Artois, rue Jean Souvraz SP 18 F-62307 Lens Cedex

[‡] Ecole Militaire Polytechnique, BP 17 Bordj-ElBahri, Alger

[§] Laboratoire d’Informatique de Nantes Atlantique (CNRS-UMR 6241), Site de Polytech’Nantes, rue Christian Pauc, BP 50609 F-44306 Nantes Cedex

fournir aux opérateurs de sécurité une solution automatique pour rechercher les relations entre les alertes. Plusieurs approches de corrélation d’alertes sont proposées dans la littérature [CM02, SK00, NCR02, DC01]. La plupart de ces approches exigent beaucoup de connaissances d’experts, ou elles ne détectent pas les attaques coordonnées, comme nous le verrons plus tard.

Dans cet article, nous présentons une nouvelle modélisation de la corrélation d’alertes qui n’exige pas beaucoup de connaissances d’experts. Elle est basée sur une forme simple de réseaux Bayésiens [Jen96, Pea91] baptisée réseaux Bayésiens naïfs augmentés [FG96]. Bien entendu, les réseaux Bayésiens ont été déjà utilisés dans la détection d’intrusion ou dans la corrélation d’alertes. Cependant, comme nous le verrons dans la section des travaux existants, notre approche présente plusieurs avantages par rapport aux méthodes existantes. En particulier, concernant la contribution importante de connaissances d’experts pour modéliser la corrélation d’alertes.

Nous montrons, par une expérimentation, comment cette modélisation permet de détecter des attaques coordonnées sous forme de scénarios. Comme nous le verrons plus loin dans la section 4.1, le terme scénario est utilisé dans un sens très simplifié qui représente principalement l’ensemble d’actions impliquées dans la compromission d’un objectif d’intrusion, ou tout événement anormal qu’un opérateur de sécurité veut surveiller. Le processus de corrélation d’alertes sera considéré comme un problème de classification. Étant donné un ensemble d’actions récemment observées et un ensemble d’objectifs d’intrusion, notre but est de déterminer la plausibilité qu’un objectif d’intrusion soit compromis.

Le reste de cet article est organisé comme suit. La section 2 présente les problèmes de la corrélation d’alertes. La section 3 présente les réseaux Bayésiens naïfs augmentés. La section 4 présente notre nouvelle modélisation pour la prédiction des attaques coordonnées. La section 5 compare notre approche avec les travaux existants. La dernière section conclut le papier.

2 La corrélation d’alertes

La corrélation d’alertes consiste à rechercher des relations entre les alertes dans le but de réduire le volume d’alertes ou de détecter des attaques coordonnées. Elle a été étudiée, ces dernières années, par plusieurs chercheurs, et plusieurs approches 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. Par exemple, Valdes et Skinner [VS01] ont défini des mesures de similarité entre des attributs tels que: la classification des attaques, les adresses source et cible, l’identité des utilisateurs, 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 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 Cuppens [Cup01] et par Dain [DC01]. Debar et Wespi [DW01] ont proposé une solution pour l’agrégation et la corrélation d’alertes qui est mise en application dans l’outil commercial “Risk Manager”. Julish [Jul01] a proposé d’employer 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. Elles emploient les pré-conditions et les post-conditions des actions pour construire implicitement des scénarios d’attaque [CM02, SK00, NCR02]. D’autres approches, tout simplement, introduisent la description des scénarios dans le système [DC01].

Les méthodes existantes permettent de réduire le volume d’alertes et de détecter les plans d’attaque achevés. Cependant, pour la prédiction d’attaques ces méthodes génèrent un nombre important de scénarios.

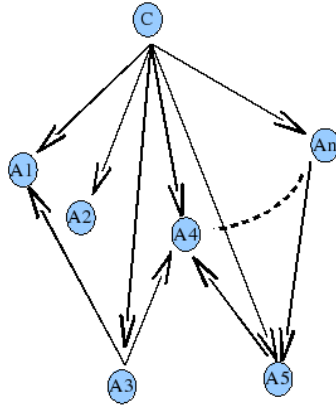


FIG. 1 – Structure d'un RBNA

3 Rappel sur les Réseaux Bayésiens

Les réseaux Bayésiens sont des modèles graphiques largement utilisés pour représenter et manipuler des informations incertaines [Jen96, Pea91, NWL⁺07]. Ils sont constitués de deux composants :

- Un composant graphique représenté par un graphe orienté sans circuit (DAG) dont les nœuds représentent les événements et les arcs représentent les relations entre ces événements.
- Un composant numérique qui consiste en une quantification des différents liens dans le graphe par une distribution des probabilités conditionnelles de chaque nœud dans le contexte de ses parents.

Les réseaux Bayésiens naïf [SP92] représentent une forme très simple des réseaux Bayésiens, qui 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. Dans le but d'améliorer les performances de la classification par les réseaux Bayésiens naïfs, Friedman et al [FG96] ont proposé d'augmenter la structure du réseau Bayésien naïf en rajoutant certains arcs entre les variables nœuds figure 1), et donc se passer de la forte hypothèse d'indépendance entre les nœuds dans le contexte du nœud parent. Dans un réseau Bayésien naïf augmenté (RBNA), un arc de A_i vers A_j implique que l'influence de A_i dans l'évaluation du nœud *classe* dépend aussi de la valeur de A_j .

Les RBNA ont donné des résultats satisfaisants pour des problèmes de classification [FG96]. La classification est assurée en considérant le nœud racine comme une variable non observée qui représente la classe d'un objet, et les nœuds feuilles comme étant des variables observées correspondant aux différents attributs spécifiant cet objet.

Une fois le réseau Bayésien quantifié, il est possible de classer tout nouvel objet, étant donnés les valeurs des attributs, en utilisant la règle de Bayes exprimée par:

$$P(c_i|A) = \frac{P(A|c_i).P(c_i)}{P(A)}, \quad (1)$$

où c_i est une valeur possible de la classe et A représente l'observation concernant les attributs.

4 Prédiction des scénarios d'attaque

Le but de cette approche est d'apprendre, à partir de l'historique des observations, les relations entre les alertes qui contribuent à compromettre des objectifs d'intrusion, sous la forme de scénarios d'attaque.

Le RBNA codera l'influence de chaque action sur les objectifs d'intrusion en calculant les distributions de probabilités conditionnelles à partir de l'historique des observations. Nous proposons d'utiliser un réseau Bayésien pour la prédiction des scénarios d'attaque, et plus précisément un RBNA qui permet de prendre en compte à la fois les dépendances entre chaque action et l'objectif d'intrusion, mais aussi les

dépendances directes entre actions. La structure augmentée et les paramètres des modèles seront automatiquement déterminés à partir de l'historique des observations. Une fois les distributions de probabilités des différents nœuds du RBNA calculées, ce modèle peut être utilisé pour prévoir si un objectif d'intrusion peut être compromis ou pas, selon une observation partielle des actions.

4.1 Les attaques coordonnées et la corrélation d'alertes

Durant la surveillance des systèmes d'informations, les SDI génèrent des alertes lorsque des actions suspectes sont observées. Les alertes rapportées chaque jour représentent des instantiations d'un ensemble fini d'actions modélisées dans le système. Par exemple, des centaines d'alertes "ICMP ping" peuvent être générées après un scan du réseau, et qui représentent des instances d'une même action "scan". Comme nous le verrons plus loin dans notre approche, les actions vont représenter les variables d'intérêt de notre RBNA.

Généralement, un intrus effectue des actions dans un ordre bien défini appelé "plan d'attaque". Dans un plan d'attaque, les premières actions modifient un système d'information ou fournissent des informations à un intrus en vue d'accomplir les dernières actions. Un plan d'attaque est modélisé comme un processus de planification d'actions qui transforment un système d'information d'un état à un autre jusqu'à ce qu'il atteigne un certain état cible, que nous appelons "objectif d'intrusion". [CM02].

Dans notre approche, nous ne nous intéressons pas à déterminer l'ordre exact dans lequel un ensemble d'actions a été exécuté de manière à atteindre un objectif d'intrusion. Nous sommes plus intéressés, d'une part à déterminer quelles sont les actions qui peuvent être impliquées dans un objectif d'intrusion, et d'autre part à développer un outil qui permet de prédire quel objectif d'intrusion peut être compromis.

Il est très important de noter que notre approche ne nécessite pas de connaissances d'experts, plus précisément elle n'exige ni les pré-conditions et post-conditions des actions comme dans [CM02, NCR02, SK00], ni une représentation explicite des scénarios d'attaque comme dans [DC01]. Elle ne nécessite même pas de préciser explicitement l'ensemble des actions impliquées dans les scénarios d'attaque. En fait, cet ensemble sera déterminé automatiquement en se basant sur les données d'apprentissage. Evidemment, elle exige un minimum de connaissances puisqu'il faut étiqueter certaines alertes pour l'apprentissage.

Dans ce qui suit, nous utilisons une définition faible d'un plan d'attaque. Un plan d'attaque est défini comme étant un ensemble $S = \{A_1, A_2, \dots, A_n, O\}$, dont les A_i 's représentent des instances d'actions et O est un objectif d'intrusion tel que:

$$A_i \text{ a une influence positive sur } O \quad (2)$$

Cette définition est plus faible que celle utilisée dans [BAC03]. Une définition possible de l'influence est:

$$A_i \text{ a une influence positive sur } O \text{ si } P(O|A_i) > P(O) \quad (3)$$

Comme nous le verrons plus loin dans cet article, l'objectif d'intrusion (O) représentera le nœud racine du RBNA, et les actions (A_i) représenteront les autres nœuds variables. L'objectif de notre approche est de détecter les plans d'attaque le plus tôt possible et de prévoir les plus plausibles. Étant donné un objectif d'intrusion, nous pouvons distinguer trois types d'actions:

- Actions avec influence négative qui diminuent la probabilité d'atteindre l'objectif d'intrusion, tel que : $P(O|A_i) < P(O)$.
- Actions avec influence positive qui augmentent la probabilité de compromettre l'objectif d'intrusion sans vraiment y parvenir, tel que: $P(O|A_i) > P(O)$ et $P(O|A_i) < Seuil$. Cela signifie que la probabilité d'atteindre l'objectif d'intrusion augmente sans dépasser un certain seuil (50% par exemple).
- Actions avec influence critique qui permettent d'atteindre directement l'objectif d'intrusion, tel que : $P(O|A_i) > P(O)$ et $P(O|A_i) > Seuil$. Cela signifie que la probabilité d'atteindre l'objectif d'intrusion dépasse un certain seuil.

La section suivante présente les principales étapes de notre approche.

4.2 Principales étapes de détection des scénarios d'attaque

Dans cette section, nous expliquons comment modéliser la corrélation d'alertes par les RBNA, en exploitant l'historique des observations. Notre approche comprend trois étapes principales :

1. Prétraitement des données: cette étape concerne le prétraitement de l'historique des observations. Le résultat de cette étape est un ensemble de données formatées.
2. Construction des réseaux Bayésiens naïfs augmentés: dans cette étape, nous calculons les distributions de probabilités conditionnelles des variables nœuds de chaque RBNA.
3. Prédiction des objectifs d'intrusion: dans cette étape nous prédirons les objectifs d'intrusion par l'application des mécanismes d'inférence des réseaux Bayésiens.

Dans cet article, un cas d'étude concernant la prévention des attaques DDoS sera présenté. Le premier scénario DARPA'2000 [DAR00] comprend un déni de service distribué (DDoS) mené par un attaquant novice. Le but de cette attaque est qu'un attaquant relativement novice, à l'aide d'une attaque en "scripte", peut compromettre plusieurs hôtes sur Internet, installer les composants nécessaires pour mener un DDoS, et ensuite lancer un DDoS. Dans ce scénario, l'attaquant exploite une faille dans l'outil Sadmin (outil d'administration à distance) pour obtenir un accès root dans trois hôtes Solaris du site Eyrie Air Force Base (AFB) [DAR00]. Les étapes du scénario d'attaque sont :

1. Scan du site AFB à partir d'un site distant.
2. Recherche des adresses IP des hôtes Solaris exécutant Sadmin.
3. Compromission des hôtes via une vulnérabilité dans Sadmin.
4. Installation du "trojan mstream" DDoS sur les trois hôtes du site AFB.
5. Lancement du DDoS.

Dans une première étape, l'attaquant effectue un IP sweep sur plusieurs sous-réseaux sur le site AFB. Il envoie des requêtes ICMP-echo dans ce balayage et écoute les réponses ICMP-echo afin de découvrir quels sont les hôtes en marche. Ensuite, les hôtes découverts sont interrogés pour déterminer ceux qui exécutent Sadmin. Par la suite, l'attaquant essaie de compromettre les hôtes exécutant Sadmin. L'attaquant tente d'exploiter Sadmin plusieurs fois dans chaque hôte, chaque fois avec des paramètres différents. À la fin de cette étape, l'attaquant obtient un accès root sur trois hôtes. Dans l'étape suivante, l'attaquant effectue une connexion Telnet sur les hôtes compromis et installe les composants nécessaires pour le DDoS (mstream serveur et mstream client). Dans la dernière étape, l'attaquant lance le DDoS contre la victime.

Nous allons maintenant décrire les trois étapes de notre approche.

4.3 Prétraitement des données

Pour construire le RBNA, nous effectuons un certain prétraitement sur les données d'observation. Les données contiennent un ensemble d'alertes qui rapportent les actions exécutées, et également des informations sur les objectifs d'intrusion (s'ils ont été atteints ou non). Nous allons d'abord regrouper les objectifs d'intrusion observés dans une seule classe appelée "Objectifs-Intrusion" et nous affectons à chaque objectif d'intrusion un numéro de 0 à N, où 0 représente aucun objectif d'intrusion et N représente le nombre maximum d'objectifs d'intrusion à protéger. Ainsi, le domaine d'Objectifs-Intrusion est $\{0, 1, 2, \dots, N\}$. Par exemple, le premier scénario de DARPA'2000 contient un seul objectif d'intrusion (une attaque DDoS). La classe va contenir alors deux valeurs $\text{Dom}(classe) = \{0, 1\}$ (0 signifie que l'objectif n'est pas atteint, et 1 signifie que l'objectif est compromis).

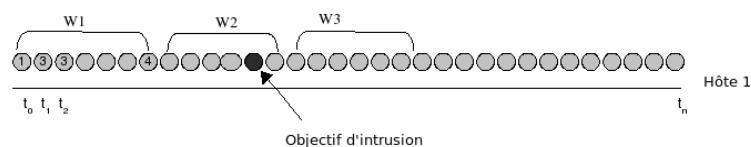


FIG. 2 – Prétraitement des données (a)

L'étape suivante consiste à trier les alertes observées en fonction de leur ordre chronologique de détection. Nous les subdivisons en sous groupes ($W1, W2, W3, etc.$) (figure 2), en fonction d'une certaine fenêtre de temps déterminée expérimentalement (de quelques minutes jusqu'à deux heures). Ces fenêtres constituent habituellement le temps nécessaire pour réaliser un plan d'attaque. Ces fenêtres sont cruciales pour déterminer l'ensemble des actions impliquées dans les scénarios.

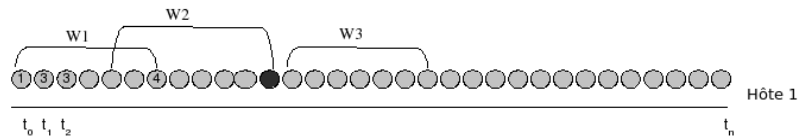


FIG. 3 – Prétraitement des données (b)

Si un objectif d'intrusion est observé dans une fenêtre, nous déplaçons cette fenêtre à gauche jusqu'à ce qu'elle se termine sur cet objectif d'intrusion (figure 3). Nous faisons cela afin de nous assurer que toutes les actions impliquées dans chaque objectif d'intrusion soient présentes dans la même fenêtre. Procéder de cette façon signifie que certaines actions peuvent être considérées sur deux fenêtres simultanément. Par exemple, dans la figure 3 l'action 4 appartient aux fenêtres $W1$ et $W2$. $W1$ contient un trafic normal et $W2$ contient un plan d'attaque (car à la fin de la fenêtre $W2$ un objectif d'intrusion est compromis). En fonction de la fréquence d'observation de l'action 4 sur des séquences normales ou anormales, nous pouvons déterminer si l'action 4 est suspecte ou non.

	$Action_1$	$Action_2$...	$Action_N$	Objectifs
w_1	vrai	faux	...	vrai	0
w_2	faux	faux	...	vrai	1
w_3	vrai	faux	...	vrai	0
w_4	faux	faux	...	vrai	0
...

TAB. 1 – Données formatées

Enfin, nous étiquetons chaque sous groupe par le numéro correspondant à l'objectif d'intrusion observé. En cas où aucun objectif d'intrusion n'a été observé, le numéro 0 est utilisé pour dire que ce trafic ne contient pas de plans d'attaque connus. Le résultat de cette première étape est un ensemble d'observations formatées sous forme de vecteurs étiquetés par un objectif d'intrusion de 0 à N (tableau 1).

En fait, les observations concernent tous les hôtes du réseau surveillé. Nous appliquons la procédure de prétraitement des données décrite ci-dessus pour chaque hôte individuellement et nous fusionnons à la fin les résultats obtenus dans un seul tableau. La procédure de prétraitement des données d'observation est résumée dans l'algorithme 1.

Algorithm 1: Prétraitement des données

Données: Historique des observations (alertes)

Result: Tableau de vecteurs

début

Grouper tous les objectifs d'intrusion dans une classe appelée "Objectifs-Intrusion";
Affecter à chaque objectif d'intrusion un numéro de 0 à N (0 représente aucun objectif d'intrusion);
pour chaque hôte **faire**
 Trier les actions observées chronologiquement;
 répéter
 Subdiviser les observations en sous groupes, selon une certaine fenêtre de temps;
 si un objectif d'intrusion est observé dans une fenêtre **alors**
 Déplacer cette fenêtre à gauche jusqu'à ce qu'elle se termine sur cet objectif;
 Déplacer le début des observations d'une certaine durée de temps;
 jusqu'à le début des observations dépasse la fin de la première fenêtre
 Étiqueter chaque sous groupe (vecteur) par le numéro correspondant à l'objectif d'intrusion observé;
Arranger tous les vecteurs dans un seul tableau;

fin

Nous allons maintenant illustrer cette première étape sur le premier scénario DARPA'2000. Ces données contiennent un trafic réseau brute capturé par un analyseur de trafic réseau pendant le plan d'attaque. Il nous faut maintenant déterminer les actions de ce plan, ceci est fait avec l'aide d'un SDI (Snort[¶]). Après l'analyse des données DARPA'2000 avec Snort, nous avons constaté que les alertes générées concernent les actions du tableau 2.

Ces actions représentent l'ensemble des variables du RBNA. Nous avons également observé une attaque DDoS réussie contre certains hôtes, donc cet objectif d'intrusion va représenter la classe du RBNA.

A ₁ : icmp_ping
A ₂ : rpc_sadmind_request
A ₃ : sadmind_ping
A ₄ : sadmind_root_query
A ₅ : sadmind_bof
A ₆ : icmp_reply
A ₇ : telnet_info
A ₈ : telnet_login_incorrect
A ₉ : telnet_bad_login
A ₁₀ : rsh_root
A ₁₁ : icmp_port_unreachable

TAB. 2 – Les actions observées dans les données DARPA'2000

Dans DARPA'2000, l'attaquant a tenté de compromettre tous les hôtes du réseau. Il a obtenu trois hôtes compromis après l'étape 4 du scénario, et il a lancé le DDoS contre la victime dans la dernière phase. L'attaque DDoS a été réalisée sur une période d'environ 3 heures sur 5 phases distinctes. Nous allons prendre 3 heures comme une fenêtre de temps pour traiter les alertes de chaque machine individuellement. Le prétraitement des données DARPA'2000 a donné 44 vecteurs marqués avec DDoS lorsque la fenêtre concerne une attaque réussie ou 0 lorsque la fenêtre concerne un trafic normal.

[¶] Snort est un système de détection d'intrusions, <http://www.snort.org>

4.4 Construction des réseaux Bayésiens naïfs augmentés

Nous construisons un RBNA pour chaque objectif d'intrusion. La raison pour laquelle nous considérons un RBNA par objectif d'intrusion, au lieu d'un seul RBNA avec une variable classe contenant tous les objectifs d'intrusion, est que les objectifs d'intrusion ne sont pas exclusifs. Il peut arriver que deux objectifs d'intrusion différents $O1$ et $O2$ soient compromis simultanément, à savoir $P(O1) = P(O2) = 1$. En définissant un RBNA par objectif d'intrusion, il est possible de représenter une telle situation. Toutefois, si un seul RBNA est utilisé, nous allons avoir $P(O1) = P(O2) = 0,5$. Et s'il y a N objectifs d'intrusion qui sont compromis, alors nous ne pouvons pas représenter une telle situation et nous allons avoir $P(O_i) = \frac{1}{N}$, ce qui signifie que la probabilité de chaque objectif d'intrusion est faible. Maintenant, sur la base de ce constat, nous allons modifier légèrement le tableau 1, en le fractionnant en plusieurs tableaux, chacun concerne un seul objectif d'intrusion. Plus précisément, pour chaque objectif d'intrusion nous remplaçons son numéro dans la colonne "Objectifs" par "vrai", et les autres objectifs d'intrusion par "faux". Ainsi, nous obtenons un tableau pour chaque objectif d'intrusion.

	$Action_1$	$Action_2$...	$Action_N$	$O1$
w_1	vrai	faux	...	vrai	faux
w_2	faux	faux	...	vrai	vrai
w_3	vrai	faux	...	vrai	faux
w_4	faux	faux	...	vrai	faux
...

TAB. 3 – Données formatées pour l'objectif $O1$

Le tableau 3 montre les données formatées pour l'objectif d'intrusion $O1$. La valeur "vrai" signifie que l'action/objectif a été observé(e) sur la fenêtre correspondante, et la valeur "faux" signifie que l'action/objectif n'a pas été observé(e) sur la fenêtre correspondante.

Même si l'hypothèse d'indépendance entre les variables nœuds dans le contexte de la variable classe dans la structure du réseau Bayésien naïfs n'est pas réaliste, ses performances en tant que classificateur ont donné des bons résultats dans l'analyse de données et la reconnaissance de formes.

Pour améliorer encore les performances des réseaux Bayésiens naïfs, des chercheurs ont proposé d'augmenter sa structure par la prise en considération des dépendances entre les variables nœuds [?]. Cependant, ajouter des arcs à cette structure est un problème très complexe, puisque il est équivalent à apprendre le meilleur réseau parmi tous les réseaux dont la classe est la racine. Friedman et al [FG96] ont proposé un algorithme qui peut apprendre un RBNA dans un temps polynômial, en imposant la restriction que la classe n'a aucun parent et que chaque nœud a comme parents la classe et un autre nœud au plus. Cet algorithme est basée sur une méthode bien connue, proposé par Clow et lieu [CL68]. Une des limites de cette approche est le fait qu'elle impose à chaque A_i d'être forcément relié aux autres par l'arbre obtenu par Chow et Liu. Une autre approche (Forest Augmented Naive Bayes) lève ce problème [SGC].

La figure 4 montre le RBNA résultant de l'apprentissage du premier scénario DARPA'2000. La structure du réseau est maintenant définie, il nous reste de calculer les distributions de probabilités.

Les données d'observations nous permettent d'estimer les distributions de probabilités conditionnelles qui peuvent être obtenues par un simple calcul de fréquences. Toutefois, lorsqu'une valeur d'un attribut ne se produit pas avec une valeur donnée de la classe, l'estimation du $P(A|C)$ produit une valeur nulle, et rend difficile l'étape de prédiction. Pour surmonter ce problème, nous utiliserons l'estimateur de Laplace. Compte tenu d'un facteur prédéfini f , s'il ya N instances de n exemple pour un problème de K valeurs, Laplace estime la probabilité par $(N + f)/(n + kf)$. Pour un problème binaire et avec $f = 1$, on obtient $(N + 1)/(n + 2)$ [KBS97].

Une fois les observations (alertes) obtenues et formatées comme dans le tableau 3, nous pouvons calculer la distribution de probabilités pour chaque variable. La procédure de construction des RBNA est résumée dans l'algorithme 2.

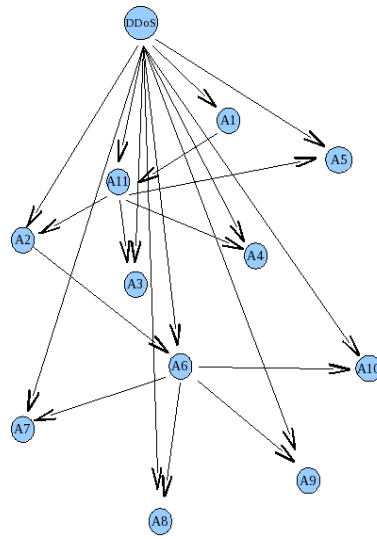


FIG. 4 – RBNA du premier scénario DARPA'2000

Algorithm 2: Construction des RBNA

Données: Tableau des vecteurs

Result: RBNA

début

pour chaque objectif d'intrusion faire

- Remplacer son numéro dans le tableau des vecteurs par "vrai" et les autres par "faux";
- Calculer les distributions de probabilités des variables du RBNA correspondant;

fin

Les distributions de probabilités de l'objectif d'intrusion du premier scénario DARPA'2000 est données dans le tableau 4. Pour ne pas encombrer l'article, nous n'avons pas présenté les distributions de probabilité des différentes actions du scénario DARPA'2000.

	Faux	Vrai
DDoS	91.3%	8.7%

TAB. 4 – Distribution de probabilités de l'objectif d'intrusion DDoS

Le tableau 4 indique qu'à priori, il existe une faible probabilité qu'un DDoS soit observé.

4.5 Prédiction des objectifs d'intrusion

Le but de l'inférence est d'estimer les valeurs des nœuds non observés, étant donné les valeurs des nœuds observés. Dans un RBNA, nous sommes intéressés à déterminer la valeur du noeud racine (la classe), étant donné les valeurs de certaines variables observées, cela peut se faire par la formule de Bayes (formule 1).

Dans notre contexte, le but de l'inférence est de calculer les nouvelles probabilités des objectifs d'intrusion étant donné que certaines actions sont observées. En présence d'une action observée, nous distinguons trois situations possibles:

1. Cette action appartient à un seul plan d'attaque. Dans ce cas, nous nous concentrons directement sur les autres actions du plan.

2. Cette action appartient à plusieurs plans d'attaque. Dans ce cas, nous nous concentrons sur les plans que cette action influence le plus.
3. Cette action n'appartient à aucun RBNA. Dans ce cas, la prédiction est seulement possible après la prochaine mise à jour du tableau 3.

Durant la détection, nous initialisons à 0 (zéro) une variable que nous appelons "timeout". Chaque nouvelle alerte générée engendrera une nouvelle probabilité de la variable classe. Selon l'influence de cette action sur les objectifs d'intrusion, la probabilité de chaque objectif d'intrusion augmentera ou diminuera. Nous nous concentrons sur les plans d'attaques (RBNA) dans lesquels la probabilité de l'objectif d'intrusion augmente.

Après chaque mise à jour, nous vérifions la nouvelle probabilité d'atteindre chaque objectif d'intrusion. Si la nouvelle probabilité dépasse un certain seuil, nous générons une alarme. Si aucune probabilité ne dépasse le seuil, nous attendons la prochaine alerte. Lorsque le timeout expire et qu'aucune probabilité ne dépasse le seuil, nous pouvons confirmer qu'aucun des plans d'attaque (les plans d'attaques modélisés par les RBNA) n'est en place. Après l'expiration du timeout, nous réinitialisons la phase de détection. La procédure de prédiction est résumée dans l'algorithme 3.

Algorithm 3: Prédiction des objectifs d'intrusion

Données: Actions observées

Result: Prédiction des objectives d'intrusion ;

début

```

Initialiser timeout;
tant que timeout n'a pas expiré faire
    si une action A est observée alors
        pour objectif  $O = O_1$  to  $O_n$  faire
            si  $influence(A, O) = positive$  alors
                Concentrer sur cet objectif;
            si  $influence(A, O) = critique$  alors
                Générer une alerte;
fin
    
```

Voyons maintenant comment chaque action du premier scénario DARPA'2000 influence l'objectif d'intrusion DDoS. A priori le RBNA du DARPA'2000 (figure 4) n'indique rien sur le plan d'attaque, mais après l'application d'un simple calcul d'influence entre les variables et la classe (l'objectif d'intrusion), nous pouvons clairement identifier les actions impliquées dans le plan d'attaque.

	$P(DDoS A_j)$
A_1	7.4%
A_2	29.1%
A_3	76.6%
A_4	76.6%
A_5	76.6%
A_6	20.1%
A_7	76.6%
A_8	76.6%
A_9	76.6%
A_{10}	76.6%
A_{11}	4.1%

TAB. 5 – Influence des actions sur le DDoS

Le tableau 5 montre l'influence de chaque action représentée par la nouvelle probabilité de l'objectif

d'intrusion. Les actions $A_3, A_4, A_5, A_7, A_8, A_9$ et A_{10} ont une influence critique sur l'objectif d'intrusion, car la probabilité d'atteindre l'objectif d'intrusion, sachant chacune de ces actions, dépasse 50% (voir tableau 6). Les actions A_2 et A_6 ont une influence positive sur l'objectif d'intrusion, car la probabilité d'atteindre l'objectif d'intrusion, sachant chacune de ces actions, augmente sans dépasser 50%. Les autres actions ont une influence négative sur l'objectif d'intrusion, car la probabilité d'atteindre l'objectif d'intrusion, sachant chacune de ces actions, diminue.

Cette analyse ne concerne que la première étape de la prédiction, à savoir si une seule action est observée. Maintenant, nous allons voir comment effectuer cette analyse sur la base des alertes rapportées.

Nous avons illustré l'influence des actions individuellement. Maintenant, nous allons illustrer la phase de prédiction avec deux scénarios réels, extraits des données DARPA'2000. Ces deux scénarios représentent respectivement un cas de succès et un cas d'échec de l'attaque DDoS contre deux hôtes distincts. Ces deux scénarios ont été retirés de l'étape d'apprentissage, à savoir le prétraitement des données et la construction du RBNA pour les utiliser dans la phase de prédiction. Ces deux scénarios seront utilisés pour tester notre approche.

Actions observées	$P(\text{DDoS} A_j)$
A_1	7.4%
A_1, A_2	25.6%
A_1, A_2, A_3	92.2%
A_1, A_2, A_3, A_4	99.8%
A_1, A_2, A_3, A_4, A_5	100%

TAB. 6 – Cas de succès du DDoS

La probabilité, avant de recevoir aucune alerte, que l'objectif d'intrusion DDoS soit atteint est 8,7% (voir tableau 4). Après avoir rejouer le premier scénario, Snort a détecté cet ensemble d'actions $\{A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8, A_9, A_{10}\}$, qui sont triées par ordre chronologique. Après avoir généré chaque alerte, nous avons mis à jour ces observations dans le RBNA et nous avons inféré la nouvelle probabilité d'atteindre le DDoS (voir tableau 6). Selon les nouvelles probabilités, il est clair qu'après la génération de l'alerte A_3 , nous pouvons confirmer que le DDoS peut être atteint directement, sans atteindre l'expiration du délai. Une alerte sera donc générée.

Actions observées	$P(\text{DDoS} A_j)$
A_1	7.4%
A_1, A_2	25.6%
A_1, A_2, A_6	47.6%
A_1, A_2, A_6, A_{11}	29.2%

TAB. 7 – Cas d'échec du DDoS

Après avoir rejoué le deuxième scénario, Snort a détecté cet ensemble d'actions $\{A_1, A_2, A_6, A_{11}\}$, qui sont triées par ordre chronologique. Après avoir généré chaque alerte, nous avons mis à jour ces observations dans le RBNA augmenté et nous avons inféré la nouvelle probabilité du DDoS (voir le tableau 7). Après avoir généré A_{11} , nous n'avons pas observé d'autres actions jusqu'à l'expiration du délai d'attente. Une fois le délai expiré, nous avons constaté que la probabilité d'atteindre l'objectif d'intrusion n'a pas dépassé le seuil. Donc, nous pouvons confirmer que le DDoS ne peut pas être atteint (le trafic est normal) et nous redémarrons la phase de détection.

5 Comparaison avec les travaux existants

Les réseaux Bayésiens ont été utilisés dans plusieurs travaux de recherche, y compris dans la détection d'intrusion [AGM⁺03, Axe04, GFV05, KMRV03, PMM03]. Cependant, peu de travaux ont appliqué les réseaux Bayésiens pour la corrélation d'alertes [GG01, QL04]. En fait, les quelques travaux qui ont appliqué les réseaux Bayésiens pour la corrélation d'alertes exigent que le scénario (sous forme d'arbre) soit

préalablement défini. Dans notre approche, une telle représentation explicite du scénario n'est pas requise, et en plus nous n'avons pas besoin de déterminer explicitement l'ensemble d'actions impliquées dans le scénario. Tout est obtenu à partir de l'historique des observations. Cette section compare notre approche aux travaux existants suivant certains critères:

1. **Approches utilisant les réseaux Bayésiens dans la détection d'intrusion:** les réseaux Bayésiens ont été introduits dans le domaine de détection d'intrusion par plusieurs chercheurs. Par exemple, ils ont été utilisés comme classificateur pour la détection d'intrusion [Axe04, ABE04, KFH05, KMRV03, PMM03]. Ils ont été également utilisés pour la détection de la cybercriminalité [AGM⁺03], la reconnaissance de plan d'attaque [GG01][QL04], la détection d'intrusions distribuée et multi-agents [BWC02][GFV05][Sco04], etc.

Abouzakhar et al [AGM⁺03] ont proposé une approche d'apprentissage des réseaux Bayésiens pour la détection de la cybercriminalité, afin de détecter les attaques distribuées le plus tôt possible.

Ben Amor et al [ABE04] ont fait une étude comparative entre l'utilisation des réseaux Bayésiens naïfs et les arbres de décision comme classificateur pour différencier entre les connexions normales et anormales.

Axelsson [Axe04] a proposé un système de détection basé sur les statistiques de Bayes combiné avec un composant de visualisation, afin de palier aux faibles taux de détection et le taux élevé de fausses alarmes. Cette approche est basée sur le principe de filtrage Bayésien, exactement comme le filtrage de Spam dans le courrier électronique. Elle permet au système de faire la différence entre les accès normaux et malicieux.

Dans [Sco04], Scott a décrit un paradigme pour la conception d'un système de détection d'intrusions réseau basé sur des modèles stochastiques. Le principe est de baser la détection d'intrusions sur les modèles stochastiques des utilisateurs combiné au comportement des intrus en utilisant le théorème de Bayes.

Plus récemment, Gowdia et al [GFV05] ont mis au point un système de détection d'intrusions probabiliste multi-agents. Ce système est une architecture coopérative multi-agents dans laquelle des agents autonomes peuvent effectuer des tâches spécifiques de détection d'intrusion et collaborer avec d'autres agents en partageant leurs croyances sur un réseau Bayésien partagé (fournie par un expert). Ces approches ont été définies dans un cadre de détection d'intrusion et non pas pour la corrélation d'alertes. Particulièrement, l'entrée de ces systèmes n'est pas un ensemble d'alertes. La sous section suivante positionne notre travail avec quelques approches qui utilisent les réseaux Bayésiens pour la corrélation d'alertes.

2. **Approches utilisant les réseaux Bayésien dans la corrélation d'alertes:** Qin et Lee [QL04] ont proposé une approche pour la reconnaissance et la prédiction des plans d'attaque en utilisant des réseaux de causalité. Dans cette approche, les auteurs utilisent des arbres de décision pour définir une bibliothèque de plans d'attaque pour corréler les alertes. Ils transforment ensuite ces arbres en réseaux Bayésiens sur lesquels ils peuvent affecter des distributions de probabilités en intégrant les domaines de connaissances nécessaires, pour enfin évaluer le risque des objectifs d'intrusion et de prédire les futures attaques.

Plus récemment dans [FW08], les auteurs ont proposé une approche basée sur les réseaux Bayésiens pour l'évaluation de la sécurité informatique. Ils interprètent dans un premier temps un graphe d'attaque donné (ce graphe est supposé obtenu par un outil automatique) comme des réseaux Bayésiens. Ensuite ils combinent les scores individuels CVSS (Common Vulnerability Scoring System) [FW08] en utilisant leurs relations causales. Enfin, ils intègrent l'effet temporel du score CVSS pour dériver une mesure de sécurité finale.

La principale différence avec notre approche est que les graphes d'attaques doivent être explicitement définis par un expert dans [QL04] ou fournis par un outil automatique externe dans [FW08]. Alors que dans notre approche, ils sont obtenues automatiquement (nous n'avons pas besoin de déterminer a priori l'ensemble des actions impliquées dans les scénarios). Ceci est un avantage important de notre approche. Notre approche est plus facile à mettre en œuvre et n'implique pas une grande contribution

des connaissances d'experts. L'opérateur de sécurité n'a qu'à déterminer les objectifs d'intrusions à protéger et mémoriser quand ces objectifs ont été compromis dans l'historique des observations.

3. **Les connaissances d'experts et les limites des méthodes basées sur le mécanisme de pré-condition et post-condition:** plusieurs chercheurs ont proposé des méthodes basées sur le mécanisme de pré-conditions et post-condition [CM02, NCR02, FW08]. Par exemple, Templeton et al [SK00] ont proposé le langage JISAW pour la description des composantes d'une attaque en terme de concepts et de capacités (dans cette méthode, les *pré-conditions* correspondent aux *requires* et les post-conditions correspondent aux *provides*).

Cuppens et al [CM02] ont utilisé les pré-conditions et les post-conditions des actions pour construire implicitement des scénarios d'attaque. Cependant, ce mécanisme de pré-condition et post-condition nécessite beaucoup de connaissances d'experts afin de définir les pré-conditions et les post-conditions des actions. De plus, dans [CM02] lorsque certaines actions ne sont pas observées, certaines alertes virtuelles sont générées. Cela augmente le nombre de scénarios possibles, et la corrélation d'alertes pondérée proposée dans [BAC03] limite seulement les conséquences de cette explosion du nombre de scénarios.

L'inconvénient majeur des méthodes de corrélation d'alertes basées sur le mécanisme de pré-condition et de post-condition est que ce dernier implique beaucoup de connaissances d'experts pour définir les pré-conditions et les post-conditions liés aux attaques élémentaires. Par exemple, dans [CM02, NCR02], il est nécessaire de prévoir chaque action qui peut être exécutée par les systèmes et les utilisateurs, ainsi que les pré-conditions et les post-conditions de ces actions. Ceci n'est pas toujours réaliste et nécessite clairement beaucoup de connaissances d'experts. Il est également difficile de demander à un expert de donner les pré-conditions et les post-conditions de toutes les actions du système, et il est simplement impossible de modéliser les actions propres aux utilisateurs.

En outre, la détection des attaques coordonnées est très sensible à la modélisation des actions. L'ajout de conditions non nécessaires aux pré-conditions ou aux post-conditions d'une action change souvent le résultat de la corrélation et produit généralement des scénarios additionnels. De même, oublier des conditions peut conduire à rater la détection de quelques scénarios plausibles. En effet, un seul scénario, dû à des conditions manquantes, peut être détecté en tant que deux scénarios indépendants.

Notre approche permet de détecter les attaques coordonnées sans demander beaucoup de connaissances d'experts.

6 Conclusions

Dans cet article, nous avons proposé une nouvelle modélisation de la corrélation d'alertes basée sur les modèles probabilistes, pour traiter le problème du faible diagnostic, fourni par les alertes élémentaires, qui ne permet pas de détecter des attaques coordonnées.

Nous avons résolu ce problème en fournissant un mécanisme qui permet de prévoir les objectifs d'intrusion en apprenant les plans d'attaque depuis l'historique de la détection d'intrusion sous forme de RBNA. Pendant l'étape de détection, chaque action observée fournira une évidence qui met à jour les RBNA (nous apprenons un RBNA pour chaque objectif d'intrusion). Selon le degré d'influence de cette attaque, la probabilité de chaque objectif d'intrusion changera positivement ou négativement.

Notre approche a pour avantage de rendre la prédiction des plans d'attaque plus facile grâce à la simplicité et l'efficacité des RBNA. Elle tire profit des données disponibles, et n'implique qu'une légère contribution des connaissances d'experts pour déterminer les objectifs d'intrusion. Contrairement aux approches existantes, les scénarios d'attaque ne sont pas explicitement fournis par des experts, mais ils sont calculés automatiquement à partir des données.

De plus, notre approche peut être facilement adaptée pour 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 attaques de faible sévérité. Ces dernières alertes peuvent être vues comme des actions qui doivent être exécutées avant de réaliser ces attaques sévères. Ce problème est clairement lié à la corrélation d'alertes, où les objectifs d'intrusion correspondent aux attaques sévères.

Cette modélisation permet implicitement de réduire le nombre d'alertes en se concentrant seulement sur les attaques sévères. Notre but est de détecter les attaques sévères les plus plausibles et les actions qui contribuent dans leur exécution. Les actions qui ne contribuent pas à la présence des attaques sévères seront considérées en tant qu'alertes non pertinentes.

Notre modélisation est fondée sur les réseaux Bayésiens naïfs augmentés en arbres, qui sont plus performants que les réseaux Bayésiens naïfs. Parmi les travaux futures, nous envisageons l'utilisation de réseaux Bayésiens naïfs augmentés en forêts [SGC], au lieu des réseaux Bayésiens naïfs augmentés en arbres, pour ne pas forcer la présence d'arcs entre tous les variables nœuds du réseau Bayésien naïf.

Références

- [ABE04] Nahla Ben Amor, Salem Benferhat, and Zied Elouedi. Naive bayes vs decision trees in intrusion detection systems. In *SAC*, pages 420–424, 2004.
- [AGM⁺03] Naser S. Abouzakhar, A. Gani, G. Manson, M. Abuitbel, and D. King. Bayesian learning networks approach to cybercrime detection. In *the 2003 PostGraduate Networking Conference*, 2003.
- [Axe04] S. Axelsson. Combining a bayesian classifier with visualisation: Understanding the ids. In *VizSEC/DMSEC-04 ACM*, pages 99–108, 2004.
- [BAC03] Salem Benferhat, Fabien Autrel, and Frédéric Cuppens. Enhanced correlation in an intrusion detection process. In *MMM-ACNS*, pages 157–170, 2003.
- [BWC02] Daniel J. Burroughs, Linda F. Wilson, and George V. Cybenko. Analysis of distributed intrusion detection systems using bayesian methods. In *21th IEEE International Conference on Performance, Computing, and Communications*, pages 329–334, 2002.
- [CL68] C.K. Chow and C.N. Liu. Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory*, 14(3):462–467, 1968.
- [CM02] Frédéric Cuppens and Alexandre Miège. Alert correlation in a cooperative intrusion detection framework. In *IEEE Symposium on Security and Privacy*, pages 202–215, 2002.
- [Cup01] Frédéric Cuppens. Managing alerts in a multi-intrusion detection environment. In *ACSAC*, pages 22–31, 2001.
- [DAR00] DARPA-2000. http://www.ll.mit.edu/IST/ideval/data/data_index.html. 2000.
- [DC01] Oliver Dain and Robert K. Cunningham. Fusing a heterogeneous alert stream into scenario. In *ACM Workshop on Data Mining for Security Application*, pages 1–13, 2001.
- [DW01] Hervé Debar and Andreas Wespi. Aggregation and correlation of intrusion-detection alerts. In *Recent Advances in Intrusion Detection*, pages 85–103, 2001.
- [FG96] Nir Friedman and Moises Goldszmidt. Building classifiers using bayesian networks. In *AAAI*, 1996.
- [FW08] M. Frigault and L. Wang. Measuring network security using bayesian network-based attack graph. In *3rd IEEE International Workshop on Security*, 2008.
- [GFV05] Vaibhav Gowadia, Csilla Farkas, and Marco Valtorta. Paid: A probabilistic agent-based intrusion detection system. *Computers & Security*, 24(7):529–545, 2005.
- [GG01] Christopher W. Geib and Robert P. Goldman. Plan recognition in intrusion detection systems. In *DISCEX*, volume 1, pages 46–55, 2001.
- [HDw99] M. Dacier H. Debar and A. wespi. Towards a taxonomy of intrusion detection systems. In *Computer Networks, Elseiver*, pages 805–822, 1999.
- [Jen96] Finn Verner Jensen. *Introduction to Bayesian networks*. UCL Press, London, 1996.
- [Jul01] Klaus Julisch. Mining alarm clusters to improve alarm handling efficiency. In *ACSAC*, pages 12–21, 2001.
- [KBS97] Ron Kohavi, Barry Becker, and Dan Sommerfield. Improving simple bayes. In *European Conference on Machine Learning*, 1997.

- [KFH05] Dae-Ki Kang, Doug Fuller, and Vasant Honavar. Learning classifiers for misuse and anomaly detection using a bag of system calls representation. In *IEEE Workshop on Information Assurance and Security*, pages 118–125, 2005.
- [KMRV03] Christopher Krügel, Darren Mutz, William K. Robertson, and Fredrik Valeur. Bayesian event classification for intrusion detection. In *ACSAC*, pages 14–23, 2003.
- [NCR02] Peng Ning, Yun Cui, and Douglas S. Reeves. Analyzing intensive intrusion alerts via correlation. In *RAID*, pages 74–94, 2002.
- [NWL⁺07] P. Naïm, P-H. Wuillemin, P. Leray, O. Pourret, and A. Becker. *Réseaux bayésiens*. Eyrolles, Paris, 3 edition, 2007.
- [Pea91] Judea Pearl. Probabilistic reasoning in intelligent systems: Networks of plausible inference. *Artif. Intell.*, 48(1):117–124, 1991.
- [PMM03] Ricardo Puttini, Zakia Marrakchi, and Ludovic Mè. A bayesian classification model for real-time intrusion detection. In *22nd International Workshop on Bayesian Inference and Maximum Entropy Methods in Science and Engineering*, volume 659, pages 150–162, 2003.
- [QL04] Xinzhou Qin and Wenke Lee. Attack plan recognition and prediction using causal networks. In *ACSAC*, pages 370–379, 2004.
- [Sco04] L. S. Scott. A bayesian paradigm for designing intrusion detection systems. In *Computational Statistics & Data Analysis*, pages 69–83. Elsevier, 2004.
- [SGC] J.P. Sacha, L. Goodenday, and K.J. Cios. Bayesian learning for cardiac spect image interpretation. *Artificial Intelligence in Medicine*.
- [SK00] J. T. Steven and L. Karm. A requires/provides model for computer attacks. In *New Security Paradigms Workshop*, pages 31–38, 2000.
- [SP92] Ross D. Shachter and Mark A. Peot. Decision making using probabilistic inference methods. In *UAI*, pages 276–283, 1992.
- [VS01] A. Valdes and K. Skinner. Probabilistic alert correlation. In *Recent Advances in Intrusion Detection*, pages 54–68, 2001.