République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université de Bordj Bou Arreridj

Faculté des Mathématiques et d'Informatique Département d'Informatique

Mémoire de Fin d'Études Pour l'Obtention du Diplôme de Master en Informatique

Spécialité : Réseaux et multimédia

Thème :

Sélection d'ensemble de caractéristiques pour les systèmes de détection d'intrusion

Présenté par :

Bousmaha Fatima Zohra

Mebarkia Ala

Soutenu le:19/06/2024

Devant le jury composé de:

M. Attia Abdelouahab Président

M. Maza Sofiane Encadreur

M. Benabid Sonia Examinateur

2023/2024

Remerciements

Au nom d'Allah, le Tout Miséricordieux, le Très Miséricordieux.

Je tiens à exprimer ma gratitude infinie envers Allah, le Tout-Puissant, pour m'avoir accordé la force, la patience et la guidance tout au long de ce parcours académique.

Je souhaite également adresser mes sincères remerciements à Monsieur Maza Sofiane, mon encadreur, pour sa bienveillance, son dévouement et ses conseils précieux qui ont été d'une importance capitale. Sa disponibilité et son soutien infaillible ont été une source d'inspiration et d'encouragement.

Mes remerciements vont également à tous les membres du jury pour leur temps, leur expertise et leurs précieuses contributions lors de l'évaluation de ce travail.

Je tiens à exprimer ma profonde reconnaissance envers Monsieur Attia Abdelouahab., chef du département d'informatique, pour son appui constant, ses encouragements et son soutien institutionnel qui ont facilité la réalisation de ce projet.

Enfin, je souhaite exprimer ma gratitude envers ma famille, mes amis et tous ceux qui ont été à mes côtés tout au long de ce parcours, pour leur soutien indéfectible et leurs encouragements sans faille.

Dédicaces

الحمد لله

À mes parents, qui ont toujours cru en moi et m'ont soutenu à chaque étape de ma vie. Vos sacrifices, votre amour inconditionnel et votre encouragement ont été la force motrice derrière chaque réussite, y compris la réalisation de ce mémoire. Vous êtes ma source d'inspiration infinie.

À mes frères et sœurs, pour leur soutien constant, leurs encouragements et leur compréhension tout au long de ce voyage académique. Votre présence a rendu chaque défi surmontable et chaque succès plus significatif.

À mes amis les plus chers, pour les moments de joie partagés, les encouragements sans fin et le soutien inconditionnel. Votre amitié précieuse a illuminé les jours sombres et a rendu cette aventure mémorable.

À M.Maza Sofian, dont la patience, la sagesse et les conseils avisés ont été une boussole dans ce voyage académique. Votre mentorat m'a permis de grandir professionnellement et personnellement.

À toutes les personnes qui ont croisé mon chemin et ont contribué, de près ou de loin, à ce travail, je vous adresse mes plus sincères remerciements. Ce mémoire vous est dédié en reconnaissance de votre soutien, de votre amitié et de votre inspiration.

Résumé:

Ce mémoire se focalise sur le problème des attaques et des intrusions dans les systèmes d'information, en mettant particulièrement l'accent sur la sécurité informatique et les systèmes de détection d'intrusion (IDS). Il discute et analyse divers classificateurs intelligents utilisés pour la détection automatique et la classification des attaques réseau dans les IDS. Une partie importante du travail porte sur la sélection de caractéristiques d'ensemble, mettant en lumière ses avantages, défis potentiels et bonnes pratiques pour une utilisation efficace. De plus, il explore les modèles de base des systèmes IDS, leur classification, les méthodes de détection disponibles, ainsi que les mesures d'évaluation des performances. Enfin, le mémoire guide les lecteurs dans la création de leurs propres systèmes de détection d'intrusion en utilisant l'outil Weka et l'ensemble de données NSL KDD, afin de renforcer la protection des réseaux contre les cyberattaques.

Abstract:

This dissertation focuses on the problem of attacks and intrusions into information systems, with particular emphasis on computer security and intrusion detection systems (IDS). It discusses and analyzes various intelligent classifiers used for automatic detection and classification of network attacks in IDSs. A significant part of the work focuses on ensemble feature selection, highlighting its advantages, potential challenges and best practices for effective use. In addition, it explores the basic models of IDS systems, their classification, available detection methods, as well as performance evaluation metrics. Finally, the brief guides readers in creating their own intrusion detection systems using the Weka tool and the NSL KDD dataset, to enhance network protection against cyber attacks.

ملخص :

تركز هذه الأطروحة على مشكلة الهجمات والاقتحامات في أنظمة المعلومات، مع التركيز بشكل خاص على أمن الكمبيوتر و أنظمة كشف التسلل (IDS). يناقش ويحلل العديد من المصنفات الذكية المستخدمة للكشف التلقائي وتصنيف هجمات الشبكة في IDS. يركز جزء مهم من العمل على اختيار الميزات الشاملة، وتسليط الضوء على فوائدها والتحديات المحتملة وأفضل الممارسات للاستخدام الفعال. بالإضافة إلى ذلك، فإنه يستكشف النماذج الأساسية لأنظمة IDS وتصنيفها وطرق الكشف المتاحة بالإضافة إلى مقاييس تقييم الأداء. أخيرًا، يرشد الموجز القراء إلى كيفية إنشاء أنظمة كشف التسلل الخاصة بهم باستخدام أداة Weka ومجموعة بيانات NSL KDD، لتعزيز حماية الشبكات ضد الهجمات الإلكترونية.

Mots-Clé: NSL KDD , Weka , Attaque , Cyberattaques ,IDS

Table des matières

Liste des Figures

Liste des tableaux

Introduction générale

Introduction générale

Dans le paysage complexe de la sécurité informatique, la détection d'intrusion occupe une place prépondérante pour prévenir les cyberattaques et protéger les réseaux informatiques. Les systèmes de détection d'intrusion (IDS) jouent un rôle essentiel en surveillant en permanence le trafic réseau pour identifier et signaler les activités malveillantes.

Cependant, la tâche de détection d'intrusion est devenue de plus en plus difficile en raison de la sophistication croissante des attaques et de la diversité des menaces. Dans ce contexte, la sélection d'ensemble de caractéristiques émerge comme une étape cruciale pour améliorer l'efficacité des IDS.

La sélection d'ensemble de caractéristiques consiste à identifier les attributs les plus pertinents et discriminants dans un ensemble de données, tout en réduisant la dimensionnalité de ces données. Cette étape revêt une importance capitale car elle permet de concentrer les efforts de détection sur les caractéristiques les plus informatives, tout en réduisant les coûts de calcul et en améliorant la capacité des IDS à détecter les intrusions, même dans des environnements complexes et dynamiques.
Dans ce contexte, l'ensemble de données KDD Cup 1999, largement utilisé dans la recherche en sécurité informatique, fournit une base de données réaliste pour évaluer les performances des IDS.

L'objectif principal de la sélection d'ensemble de caractéristiques pour les systèmes de détection d'intrusion est donc de maximiser la capacité de discrimination entre les activités normales et anormales, tout en minimisant les fausses alertes. Cela nécessite l'utilisation de techniques avancées d'apprentissage automatique et de traitement des données pour identifier les caractéristiques les plus significatives, tout en prenant en compte des facteurs tels que la variabilité des attaques, la taille et la diversité des ensembles de données, ainsi que les contraintes de ressources informatiques.

Dans cette optique, cette étude se propose d'explorer et d'évaluer différentes méthodes de sélection d'ensemble de caractéristiques pour les systèmes de détection d'intrusion, en mettant l'accent sur leur efficacité, leur robustesse . En examinant les avantages et les limites de chaque approche, nous pourrons mieux comprendre comment optimiser la conception et les performances des IDS pour faire face aux défis actuels de la sécurité informatique.

Notre mémoire est organisé comme suit :
Dans le premier chapitre, nous expliquons en profondeur les principes fondamentaux, les techniques et les applications de la sélection de caractéristiques d'ensemble. Nous mettons en avant les avantages de cette approche, notamment l'amélioration de la performance des modèles, la réduction du surajustement et la robustesse aux données bruitées. De plus, nous abordons les défis potentiels, les considérations pratiques et les bonnes pratiques pour une utilisation efficace de la sélection de caractéristiques d'ensemble dans divers contextes et domaines d'application.

Dans le deuxième chapitre, nous avons exploré les systèmes de détection d'intrusions (IDS), en présentant leur modèle de base et leur classification. Nous avons également examiné en détail les différentes méthodes de détection disponibles, ainsi que les mesures d'évaluation des performances des systèmes IDS.

Dans le dernier chapitre, nous apprenons à fabriquer nos propres systèmes de détection d'intrusion. Nous utilisons un outil appelé Weka et un ensemble de données appelé NSLKDD pour nous aider. Nous découvrons ce dont nous avons besoin pour créer ces systèmes, comment les assembler, et comment les tester pour nous assurer qu'ils fonctionnent correctement. En utilisant Weka et NSLKDD, nous pouvons construire des systèmes de détection d'intrusion solides pour protéger nos réseaux contre les cyberattaques.

Chapitre 01 : Sélection ensemble des attributs

1.1 Introduction

Dans le domaine de l'apprentissage automatique, la sélection des caractéristiques joue un rôle essentiel dans la création de modèles performants. En effet, choisir les bonnes caractéristiques parmi un ensemble de données peut être comparé à chercher des trésors cachés dans un océan d'informations. Cette quête de caractéristiques pertinentes est souvent complexe et peut être laborieuse, surtout lorsque les ensembles de données sont volumineux et riches en variables.

La sélection de caractéristiques est un processus essentiel en science des données et en apprentissage automatique, visant à identifier les caractéristiques les plus pertinentes d'un ensemble de données. Ces caractéristiques, souvent appelées variables ou attributs, sont les éléments qui décrivent chaque instance ou observation dans un ensemble de données. Cependant, dans de nombreux cas, les ensembles de données contiennent un grand nombre de caractéristiques, dont certaines peuvent être redondantes, bruitées ou non informatives. La sélection de caractéristiques vise donc à réduire la dimensionnalité de l'ensemble de données en ne conservant que les caractéristiques les plus utiles pour la tâche d'apprentissage ou de modélisation.

Les techniques d'ensemble pour la sélection de caractéristiques peuvent être regroupées en deux catégories principales : les méthodes heuristiques et les méthodes basées sur l'apprentissage automatique. Les méthodes heuristiques, telles que le vote majoritaire ou les techniques de pondération, combinent les décisions individuelles des algorithmes de sélection de caractéristiques pour parvenir à une décision finale. En revanche, les méthodes basées sur l'apprentissage automatique utilisent des modèles d'apprentissage automatique, tels que les arbres de décision ou les réseaux de neurones, pour apprendre à sélectionner les caractéristiques les plus pertinentes.

Dans ce chapitre, on explore les principes fondamentaux, les techniques et les applications de la sélection de caractéristiques d'ensemble. Nous discuterons des avantages de cette approche, tels que l'amélioration de la performance des modèles, la réduction du surajustement et la robustesse aux données bruitées. De plus, nous aborderons les défis potentiels, les considérations pratiques et les bonnes pratiques pour une utilisation efficace de la sélection de caractéristiques d'ensemble dans différents contextes et domaines d'application.

1.2 Définition de la sélection de caractéristique

La sélection de caractéristique est le processus qui recherche le(s) meilleur(s) sous-ensemble(s) de fonctionnalités qui garantissent un fonctionnement optimal description des données.[1] Quatre raisons justifient l'utilisation des techniques de sélection des caractéristiques :

-Optimiser les modèles afin de faciliter leur compréhension par les chercheurs/utilisateurs,
- diminuer la durée de l'apprentissage,
-éviter le problème de la dimension,
-améliorer la généralisation en diminuant les ajustements excessifs. [2]

La sélection d'attributs, c'est choisir seulement les caractéristiques les plus importantes ou utiles d'un groupe. C'est comme décider quels détails sont les plus pertinents pour la tâche à accomplir, et ne garder que ceux-là. Cela rend les choses plus simples et plus efficaces. [1]

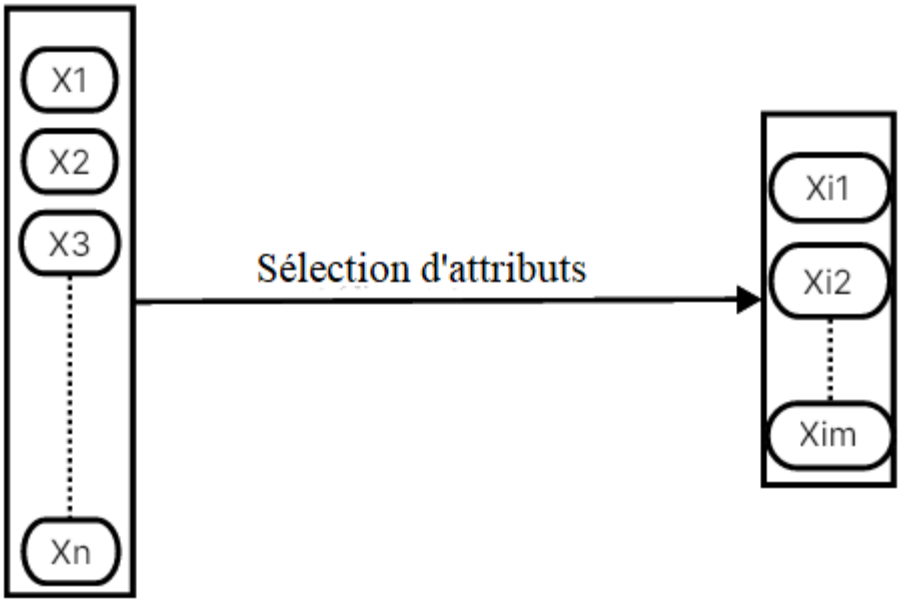

Figure 1: La sélection d'attributs [3]

1.3 Principe de sélection d'attributs

La sélection d'attributs est une étape importante dans le processus de modélisation des données, car elle vise à identifier les attributs les plus pertinents pour la construction d'un modèle prédictif. Voici quelques principes de base de la sélection d'attributs :

1.3.1 Pertinence

Les attributs sélectionnés doivent être pertinents pour la tâche de modélisation en question. Cela signifie qu'ils doivent avoir un impact significatif sur la variable cible que l'on cherche à prédire.

1.3.2 Redondance

Il est important d'éliminer les attributs redondants, c'est-à-dire ceux qui apportent des informations similaires à d'autres attributs déjà présents dans le jeu de données. La présence d'attributs redondants peut entraîner une surcharge de données et compliquer l'interprétation des résultats.

1.3.3 Cohérence

Les attributs sélectionnés doivent être cohérents avec les hypothèses du modèle et les connaissances préalables sur le domaine. Il est important de choisir des attributs qui ont un sens logique par rapport au problème étudié.

1.3.4 Facilité d'interprétation

Il est souvent préférable de privilégier des attributs faciles à interpréter, car cela permet de mieux comprendre le fonctionnement du modèle et de prendre des décisions basées sur des informations claires.

1.3.5 Performance du modèle

Enfin, la sélection d'attributs doit également viser à améliorer les performances du modèle en réduisant le bruit et en augmentant la précision des prédictions. Il est donc important de choisir des attributs qui contribuent de manière significative à la qualité des prédictions.

En résumé, la sélection d'attributs consiste à choisir judicieusement les variables qui seront utilisées pour construire un modèle prédictif, en tenant compte de leur pertinence, de leur non-redondance, de leur cohérence, de leur facilité d'interprétation et de leur impact sur les performances du modèle.

1.4 Objectif de sélection d’attributs

La performance du système de reconnaissance de forme diminue lorsque le nombre de caractéristiques augmente, ce qui entraîne une augmentation du temps de calcul. Dans certaines situations, certaines caractéristiques ne contribuent pas à la distinction entre les classes. En d'autres termes, il existe des caractéristiques redondantes ou non pertinentes qui sont inutiles dans la

classification de l'objet, ce qui justifie la nécessité de choisir les caractéristiques les plus pertinentes pour faciliter et accélérer l'apprentissage.

Effectivement, les objectifs principaux de la sélection des caractéristiques sont les suivants :

- Repérer les traits pertinents
-Minimiser les fondements d'apprentissage et de test
-Réduire la taille de l'espace d'entrée en supprimant les informations non pertinentes et redondantes
-Améliorer les performances en classification
-Réduire le temps d'apprentissage [3]

1.5 Processus de sélection d'attributs

- **Une procédure de génération :**pour produire la prochaine catégorie de caractéristiques. [4] L'objectif est de choisir un sous-ensemble d'attributs qui soit le plus informatif et le plus pertinent possible pour la tâche d'apprentissage ou de modélisation spécifique.
- **Une fonction d'évaluation :** une étape cruciale pour déterminer si le sous-ensemble choisi améliore réellement les performances du modèle par rapport à l'utilisation de l'ensemble complet d'attributs.
- **Un critère d'arrêt :** est une condition utilisée pour déterminer quand arrêter le processus de sélection d'attributs.[4]
- **Une procédure de validation :** permettant de confirmer la validité du sous-ensemble choisi. [5]

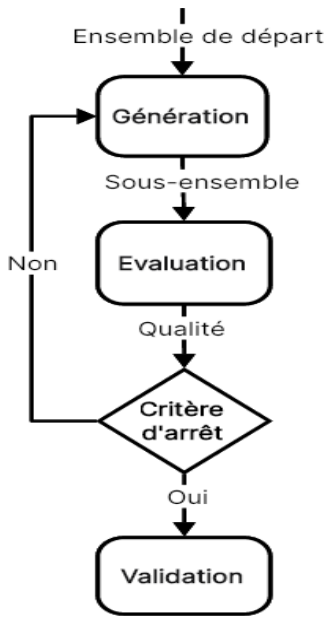

Figure 2: processus de sélection [4]

1.5.1 La Procédure de génération

La génération est une méthode de recherche qui permet d'explorer l'espace de recherche afin de créer les diverses combinaisons de caractéristiques. Il s'agit principalement d'un processus de recherche heuristique qui, à chaque étape, identifie un sous-ensemble potentiel dans l'espace de recherche pour l'évaluation. Cette étape se distingue par la mise en place d'une stratégie de recherche et d'une direction de recherche. [6]
Il ya trois méthodes de recherche sont possibles pour faire face à ce problème de taille deL'espace de recherche : la génération exhaustive, la génération heuristique et la générationaléatoire. [7]

1.5.1.1 Génération exhaustive

La génération exhaustive est une méthode qui explore systématiquement toutes les combinaisons possibles d'attributs pour la sélection d'attributs. Cette approche examine chaque sous-ensemble d'attributs, y compris les ensembles vides et complets, afin de trouver celui qui optimise la performance du modèle selon un critère donné. Le problème majeur de cette approche est que le nombre de combinaisons croît exponentiellement en fonction du nombre de variables. [8]

1.5.1.2 Génération heuristique

Dans cette catégorie, on utilise une méthode heuristique pour orienter la recherche. Cette approche est généralement utilisée par des algorithmes itératifs, où chaque itération permet de choisir ou de rejeter une ou plusieurs caractéristiques.
Ces algorithmes présentent des avantages tels que leur simplicité et leur rapidité. D'autre part, ils ne permettent pas d'explorer l'ensemble de l'espace de recherche. Les trois sous-catégories les plus répandues de cette approche sont : Dans la littérature : [8]

- **Forward Selection (Sélection ascendante)** les variables sont ajoutées une par une au modèle en commençant par le modèle le plus simple et en sélectionnant à chaque étape la variable qui améliore le plus les performances du modèle.
- **Backward Selection (Sélection descendante)** cette approche procède d'une façon inverse à ”Forward ”.[7] les variables sont retirées une par une du modèle en commençant par le modèle complet et en retirant à chaque étape la variable dont l'exclusion entraîne la plus petite perte de performance du modèle.
- **Stepwise Selection (Sélection pas à pas**) combine à la fois la sélection ascendante et descendante. Elle commence par un modèle vide et ajoute ou retire des variables à chaque étape en fonction de leur impact sur les performances du modèle.

1.5.1.3 Génération aléatoire

La génération aléatoire d'attributs est une approche où de nouveaux attributs sont créés de manière aléatoire à partir des attributs existants dans un ensemble de données. Cette méthode permet d'explorer de nouvelles caractéristiques potentiellement utiles, mais elle nécessite des techniques de validation rigoureuses pour évaluer l'efficacité des attributs générés dans la résolution d'un problème spécifique.

1.5.2 L'Évaluation

L'évaluation constitue une partie importante de la sélection d'attribut. Les algorithmes de sélection de fonctionnalités utilisent principalement trois catégories principales :

- Approche filtre (filter);
- Approche enveloppante (wrapper);
- Approche intégrée (embedded). [7]

1.5.2.1 Approche filtre (filter) :

La méthode du filtre consiste à évaluer les caractéristiques sans tenir compte des modèles d'apprentissage automatique. L'importance de chaque caractéristique est évaluée à l'aide de mesures statistiques ou heuristiques, puis un sous-ensemble est choisi en fonction de ces mesures. Par la suite, les caractéristiques choisies sont employées afin d'entraîner un modèle d'apprentissage automatique. Des mesures telles que l'ANOVA, le test du chi-deux, la corrélation de Pearson, etc. sont couramment employées. La procédure du modèle "filter" est illustrée dans la figure ci-dessous:

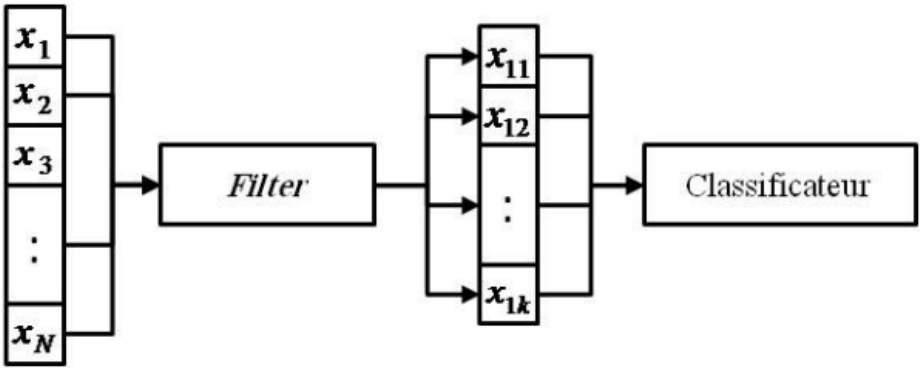

Figure 3: La procédure du modèle « filtrer » [11]

1.5.2.2 Approche enveloppante (wrapper) :

La sélection de fonctionnalités est considérée comme un problème de recherche dans cette approche enveloppante, où différents sous-ensembles de caractéristiques sont évalués en fonction des performances d'un modèle d'apprentissage automatique. On utilise différents algorithmes de recherche heuristiques, comme la recherche séquentielle avant/arrière (forward/backward), afin d'explorer l'espace des caractéristiques et de choisir le sous-ensemble optimal afin de maximiser les performances du modèle. Souvent, cette méthode est plus onéreuse en termes de calcul car elle implique de former et d'évaluer le modèle à chaque fois.
En ce qui concerne la génération de résultats, l'approche filtre est plus rapide que l'approche Wrapper. Toutefois, celle-ci présente l'avantage de donner généralement des résultats plus appropriés pour la nomenclature.[4]

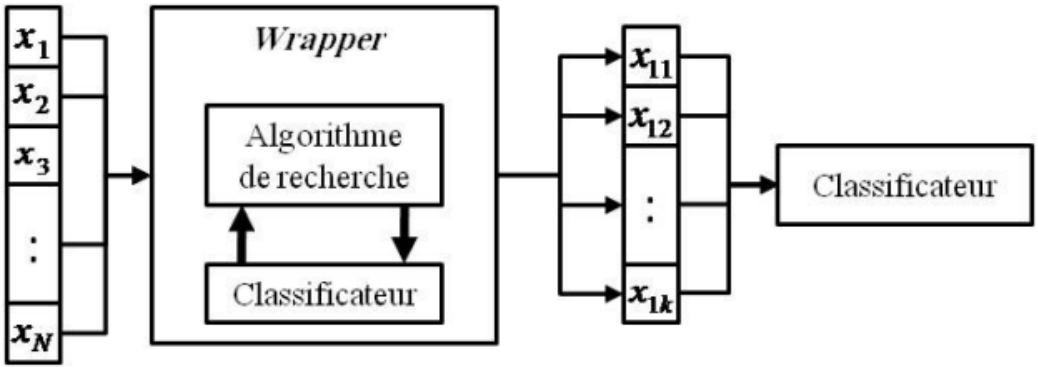

Figure 4: La procédure du modèle « wrapper » [11]

1.5.2.3 Approche intégrée (embedded) :

Dans ce type de méthode, le groupe de variables est sélectionné lors de l'apprentissage du Classificateur. L'apprentissage est utilisé pour identifier le sous-ensemble le plus optimal, comme la mise à jour de la fonction objectif. Il est possible d'utiliser des méthodes déjà mentionnées pour éliminer les variables non pertinentes. Ces méthodes sont essentielles aux méthodes intégrées. Les méthodes intégrées présentent un problème en raison de l'obligation d'adopter une stratégie adaptée au type du classifieur. L'Optimal Brain Damage ne s'applique qu'aux 12 Perceptrons multicouches. La plupart des méthodes de type Intégrées sont basées sur les méthodes connexionnistes. De manière habituelle, elles reposent sur des critères heuristiques qui permettent d'évaluer l'impact d'une ou de plusieurs variables sur la performance globale du système. [7]

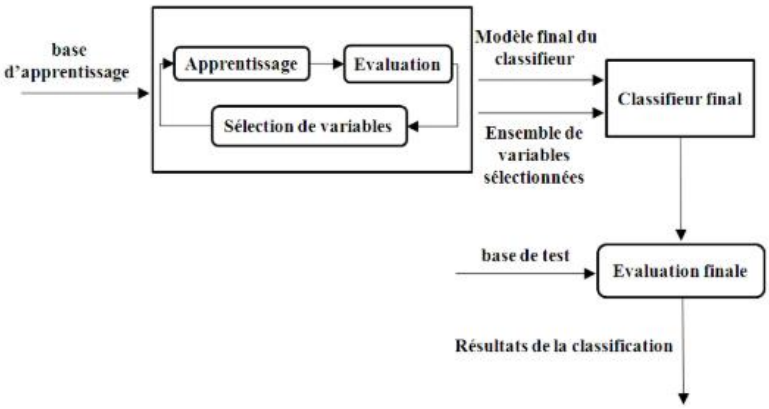

Figure 5: La procédure du modèle « Embedded » [8]

1.5.3 Critère d'arrêt

Pour mettre fin à la procédure de recherche ou identifier tous les attributs à choisir.
Après avoir établi le critère d'évaluation des variables et la méthode de recherche, tous les sous-ensembles proposés sont évalués et celui qui est choisi est celui qui est le plus pertinent dans le sens du critère d'évaluation. Pour les méthodes de sélection séquentielle, la mesure de pertinence des variables influence grandement le critère d'arrêt. La quête est suspendue. Quand aucune des autres variables n'est considérée comme pertinente. On peut déterminer la pertinence d'une variable en effectuant des tests statistiques. Quand il est impossible de calculer ces tests en raison de leur complexité, une dernière option consiste à utiliser des heuristiques. L'une des heuristiques fréquemment employées consiste à estimer l'erreur de généralisation. [10]

1.5.4 Validation

La validation n'est pas une étape du processus de sélection des caractéristiques lui-même, mais elle permet de vérifier la validité du sous-ensemble des caractéristiques sélectionnées en effectuant plusieurs tests sur des exemples de données générées artificiellement et/ou sur des données réelles. [9]
Dash et Liu suggèrent l'intégration d'une quatrième composante dans un algorithme de sélection de caractéristiques : une procédure de preuve. En fonction de la nature des données utilisées lors de cette procédure, deux options sont suggérées : artificielles ou réelles. En général, on crée une base

de données synthétique afin de tester un concept ou une application spécifique. Par conséquent, on connaît et on identifie les variables pertinentes. Il sera donc facile de valider un algorithme car il s'agit simplement de vérifier si le sous-ensemble choisi contient les variables pertinentes. En ce qui concerne les données réelles, il est courant que les variables pertinentes ne soient pas connues. Ainsi, la procédure consiste à évaluer la précision de la classification obtenue en utilisant le sous-ensemble de variables sélectionnées. [5]

1.6 Sélection de caractéristiques d'ensemble

Dans l'apprentissage d'ensemble, une collection de modèles uniques de classification ou de régression est formé et la sortie de l'ensemble est obtenue en agrégeant les sorties des modèles individuels, par exemple par vote majoritaire dans le cas de la classification, ou par calcul de la moyenne dans le cas de la régression, Dietterich [8] montre que le résultat de l'ensemble peut être plus performant que les modèles individuels lorsque des modèles faibles (instables) sont combinés. Principalement pour trois raisons :

a)plusieurs hypothèses différentes mais également optimales peuvent exister et l'ensemble réduit le risque de choisir une mauvaise hypothèse,
b) les algorithmes d'apprentissage peuvent aboutir à différents optima locaux, et l'ensemble peut donner une meilleure approximation du modèle. l'ensemble peut donner une meilleure approximation de la vraie fonction, et

c) la vraie fonction ne peut être représentée par aucune des hypothèses de l'espace d'hypothèses de l'apprenant. Et en agrégeant les sorties des modèles individuels,l'espace d'hypothèses peut être élargi. [13]

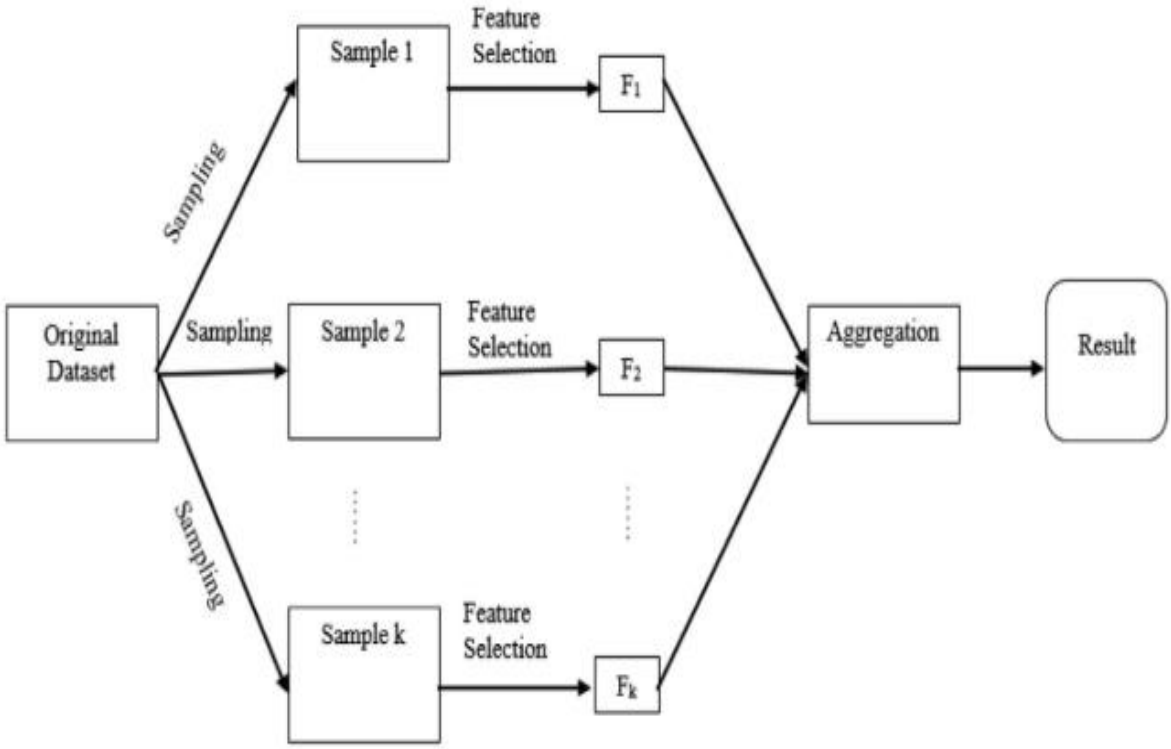

Figure 6: sélection de caractéristiques d'ensemble [14]

1.6.1 Type d'ensemble de sélection d'attributs

L'approche globale pour l'apprentissage a récemment élargi la motivation dans d'autres domaines de l'apprentissage automatique, comme la sélection de fonctionnalités. Ainsi, l'idée consiste à combiner les résultats de différents modèles de sélection de caractéristiques uniques, ce qui permettra d'obtenir de meilleurs résultats que l'utilisation d'une approche de sélection de fonctionnalité unique. Cependant, ce progrès ne se limite pas à la présence de plusieurs modèles, comme c'est également le cas pour la classification des ensembles, mais également à la variété des sous-ensembles de fonctionnalités obtenus. [12]

En général, lors de la conception d'un ensemble de sélection de caractéristiques Plusieurs décisions principales doivent être prises :

• Les méthodes FS individuelles à utiliser. Trois types de méthodes sont disponibles : wrappers, filtres et embedded. En utilisant plus qu'une seule méthode FS a inévitablement un coût de calcul, des filtres et les méthodes intégrées sont préférées aux wrappers pour être incluses dans un

ensemble. Chaque méthode individuelle a ses avantages et ses inconvénients, et les méthodes employées doivent garantir la diversité tout en augmentant la régularité du processus FS, afin d'en tirer parti pour améliorer les performances. [12]

•Le nombre de différentes méthodes FS à utiliser. Comme mentionné précédemment, il y a un Il est nécessaire d'atteindre un juste équilibre entre la complexité, la variété et la stabilité du processus. Dans le cas des ensembles de classification, les études portant sur
La nécessité d'une détermination a priori de la taille de l'ensemble est rare.
Les résultats suggèrent que l'utilisation d'autant de méthodes individuelles que d'étiquettes de classe est la meilleure option. Toutefois, dans le cas des ensembles de sélection de caractéristiques, ces études n'ont pas encore été abordées.
Et donc, à l'heure actuelle, les tests statistiques sont principalement utilisés pour déterminer le meilleur nombre de composants. [12]

•Le nombre et la taille des différents ensembles de formation à utiliser. En ce qui concerne ces deux paramètres font l'objet de quelques études dans la littérature qui visent à déterminer la taille optimale de l'ensemble de données d'entraînement pour les ensembles à des fins de classification et de prédiction. Encore une fois, dans le cas de la sélection des caractéristiques, aucune étude n'a été rapportée sur la taille des ensembles de formation optimaux pour les ensembles, bien que certains Des chercheurs ont étudié les conséquences de la distribution de l'ensemble de formation en ce qui concerne le nombre de caractéristiques et l'utilisation de méthodes de classement. [12]

•La méthode d'agrégation (également appelée combinaison) à utiliser. Différentes Méthodes sont disponibles, et la littérature scientifique ont exploré ces méthodes de combinaison, ainsi que différentes stratégies impliquant une pondération linéaire et non linéaire des classificateurs de base, l'utilisation d'algorithmes génétiques, leur relation
avec les classificateurs de base choisis, etc.

•La méthode de seuil à utiliser si les méthodes FS sont des classificateurs, c'est-à-dire, si les méthodes renvoient une liste ordonnée de toutes les caractéristiques impliquées dans le problème. Pour la plupart des études, les seuils choisis sont basés sur un pourcentage fixe d'éléments retenus, par exemple 25 %, 50 %, ou un pourcentage fixe d'éléments des caractéristiques principales, D'autres auteurs ont essayé de dériver un seuil basé sur différentes métriques. [12]

Les ensembles pour la sélection des caractéristiques peuvent être classés selon divers critères concernant un ou plusieurs des aspects ci-dessus, mais la division la plus simple concerne le type de sélecteur de base utilisé. Si les sélecteurs de base sont tous du même type, l'ensemble est dit homogène ; dans le cas contraire, l'ensemble est hétérogène. [12]

1.7 Conclusion

En conclusion, la sélection de caractéristiques est une étape cruciale dans le processus demodélisation des données et d'apprentissage automatique. En identifiant les caractéristiquesles plus pertinentes pour une tâche donnée, elle permet de réduire la dimensionnalité desdonnées, d'améliorer l'interprétabilité des modèles, et souvent d'augmenter les performancesprédictives.

Au cours de ce chapitre, nous avons exploré diverses méthodes de sélection de caractéristiques, chacune avec ses propres avantages et limitations. Des approches classiquestelles que le filtrage, l'enveloppement et les méthodes intégrées ont été examinées, ainsi quedes techniques plus avancées comme le Relief, les algorithmes basés sur l'informationmutuelle, et les méthodes de pondération basées sur l'apprentissage automatique.Il est important de noter que le choix de la méthode de sélection de caractéristiques dépendsouvent de la nature spécifique du problème, y compris la taille et la complexité des données,ainsi que des objectifs de modélisation. Il n'y a pas de méthode universelle qui convient àtoutes les situations, et il est souvent judicieux d'expérimenter plusieurs approches et decomparer leurs performances pour trouver celle qui convient le mieux à une tâche donnée.

Enfin, la sélection de caractéristiques est un domaine de recherche en évolution constante,avec de nouvelles méthodes et techniques émergentes régulièrement. En restant informé desdéveloppements récents dans ce domaine, les praticiens de l'apprentissage automatiquepeuvent améliorer leurs capacités à extraire des informations pertinentes à partir des donnéeset à développer des modèles prédictifs plus performants.

Chapitre 2 : Les systèmes de détection d’intrusion (IDS)

2.1 Introduction

En raison de l'évolution rapide de la technologie des réseaux, en particulier les réseaux sans fil, la protection de ces réseaux et de ses terminaux connectés contre différentes menaces intentionnelles ou accidentelles est devenue un enjeu majeur.
Il est essentiel de garantir la sécurité de toutes les données liées aux technologies Internet, des données stockées dans des bases de données et qui sont transmises sur le réseau. Les intrusions représentent de véritables dangers, qu'il s'agisse d'activités non autorisées ou d'utilisations malveillantes des ressources d'information qui compromettent les politiques de sécurité.La plupart du temps, les systèmes et les méthodes classiques de prévention des intrusions tels que les pare-feu, le cryptage et le contrôle d'accès ne sont pas efficaces face à l'évolution des nouvelles menaces avancées.
Il est essentiel de faire face aux défis de la cybersécurité, de repérer les intrusions et de préserver nos données, car il s'agit d'un défi crucial qui ne doit jamais être négligé.
En 1980, James E Anderson a introduit un nouveau concept de détection d'intrusion afin de repérer toute activité non autorisée dans un réseau [15].
Au cours de ce chapitre, nous avons introduit les systèmes de détection d'intrusions (IDS), en définissant le modèle de base de ces systèmes. Ensuite, nous avons examiné en détail la classification des IDSs et exposé une analyse des diverses méthodes de détection possibles, ainsi que les différentes mesures d'évaluation des systèmes IDSs.

2.2 Problématique de sécurité

Les attaques contre les systèmes informatiques sont quotidiennes, dans de nombreux lieux à la fois, au point qu'il est impossible de les énumérer. Cette situation soulève des interrogations cruciales quant à la manière de protéger efficacement les systèmes informatiques contre ces menaces persistantes. La détection précoce des activités suspectes, la mise en œuvre de mesures de prévention robustes, la gestion proactive des vulnérabilités, la sensibilisation et la formation des utilisateurs, ainsi que la mise en place de plans de réponse aux incidents efficaces, sont autant de défis auxquels les organisations sont confrontées. De plus, la nécessité de se conformer aux réglementations en matière de protection des données et de sécurité des informations ajoute une dimension supplémentaire à cette problématique déjà complexe.Effectivement, la sécurité préventive, bien que cruciale, peut souvent être contournée ou insuffisante pour protéger un système informatique contre des attaques sophistiquées et en constante évolution. C'est pourquoi la couche de détection d'intrusion (IDS) joue un rôle crucial dans la stratégie globale de sécurité informatique. Les IDS sont conçus pour détecter les activités suspectes ou malveillantes sur un réseau ou un système informatique et peuvent contribuer à compléter les mesures de sécurité préventive en identifiant les menaces en temps réel ou après coup. [16]

2.3 Système de détection d'intrusion

Un système de détection d'intrusions (« Intrusion Détection Systems » ou IDS) est unappareil ou une application qui alerte l'administrateur en cas de faille de sécurité, de violation de règles ou d'autres problèmes susceptibles de compromettre son réseau informatique.[17]
le système de détection d'intrusion est constitué d'un ensemble de éléments matériels et informatiques élaborés dans le but d'automatiser le processus de détection des intrusions.[15]
Ce genre de logiciel de sécurité informatique est associé à la sécurité du réseau de la même manière qu'un pare-feu. Il est perçu comme une deuxième protection pour repérer les diverses activités malveillantes qui ne peuvent pas être détectées par un pare-feu classique. De manière dynamique, il surveille les événements qui se déroulent dans un système et évalue si ces événements sont des signes d'attaque ou représentent une utilisation légitime du tout.
Ces systèmes sont devenus essentiels pour renforcer la sécurité et repérer les menaces avant qu'elles ne causent des dommages importants grâce à leurs performances et à leur capacité à détecter les intrusions. [15]

2.4 Les méthodes de détection

2.4.1 Détection basée sur la signature

Cette méthode implique de confronter le trafic réseau ou les événements système à une base de données de signatures préalablement établies qui représentent des modèles d'activités malveillantes déjà identifiés. Une fois que l'IDS détecte une correspondance, il déclenche une alerte afin de signaler l'activité suspecte. [18]

Avantages

- Efficacité contre les attaques connues
- Faible taux de faux positifs
- Facilité de déploiement

Inconvénients

- Incapacité à détecter de nouvelles menaces
- Besoin de mises à jour régulières
- Dépendance vis-à-vis des fournisseurs

2.4.2 Détection basée sur l'anomalie

Dans cette méthode, on établit un profil du trafic ou des comportements système normaux, puis on cherche des écarts importants par rapport à ce comportement prévu. Les activités qui dépassent les seuils définis comme anormaux déclenchent des alertes. [18]

Avantages

- Détection des nouvelles menaces
- Flexibilité et adaptabilité
- Réduction des faux négatifs

Inconvénients

- Taux élevé de faux positifs
- Complexité de configuration
- Coût élevé

2.5 Les types IDS

En raison de la variété des attaques qui peuvent être mises en place, il est nécessaire de détecter les intrusions à différents niveaux. Les IDS peuvent être classés en fonction de leur localisation et de ce qu'ils contrôlent (les « sources d'information ») ou de leurs fonctions. Ces diverses catégories ont été suggérées dans [19].

2.5.1 IDS Réseau (NIDS - Network-based IDS)

Les IDS réseau assurent la surveillance du trafic sur le réseau en analysant les paquets de données afin de repérer les activités suspectes ou malveillantes. Ils sont situés généralement à des endroits stratégiques tels que les passerelles, les commutateurs ou les routeurs, afin de surveiller le trafic entrant et sortant. Les NIDS peuvent repérer diverses menaces, telles que les scans de ports, les attaques par déni de service (DDoS), les intrusions, etc. [20]

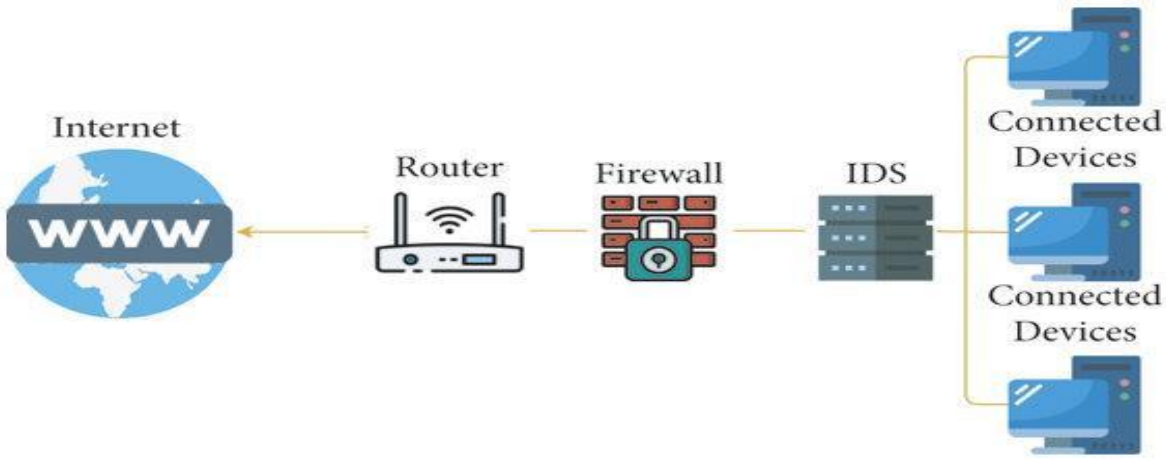

Figure 7: Exemple de NIDS [21]

2.5.2 IDS Hôte (HIDS - Host-based IDS)

Les IDS hôtes sont chargés de surveiller les activités sur un système informatique particulier. Dans le but de repérer les comportements suspects ou malveillants, ils examinent les journaux système, les fichiers système et d'autres activités sur l'ordinateur. En règle générale, les HIDS sont installés sur des serveurs ou des postes de travail essentiels afin de surveiller les activités locales, comme les tentatives de modification des fichiers système, les modifications de configuration, etc.[20]

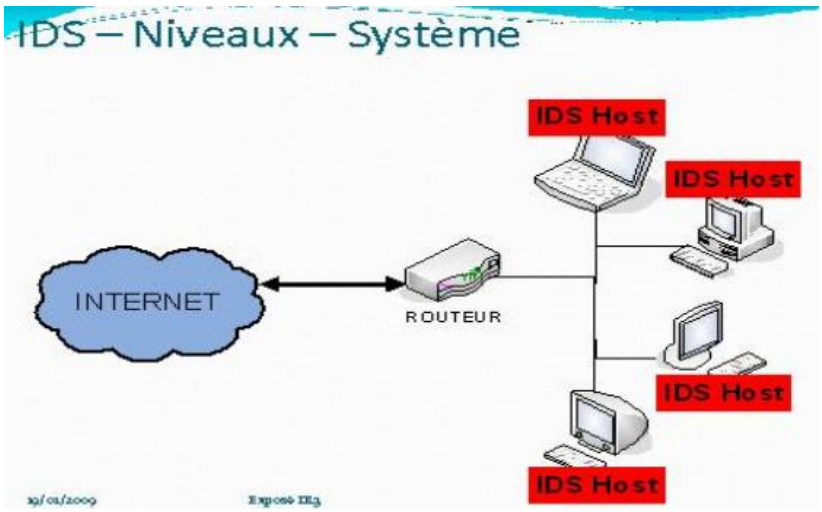

Figure 8: Exemple de HIDS [21]

2.5.3 IDS Hybride

L'IDS hybride combine les capacités de l'IDS réseau et de l'IDS hôte pour offrir une protection plus complète. Ils peuvent surveiller le trafic réseau et l'activité du système sur un hôte, ce qui leur permet de détecter un plus large éventail de menaces et de fournir des informations plus détaillées sur les attaques.[20]

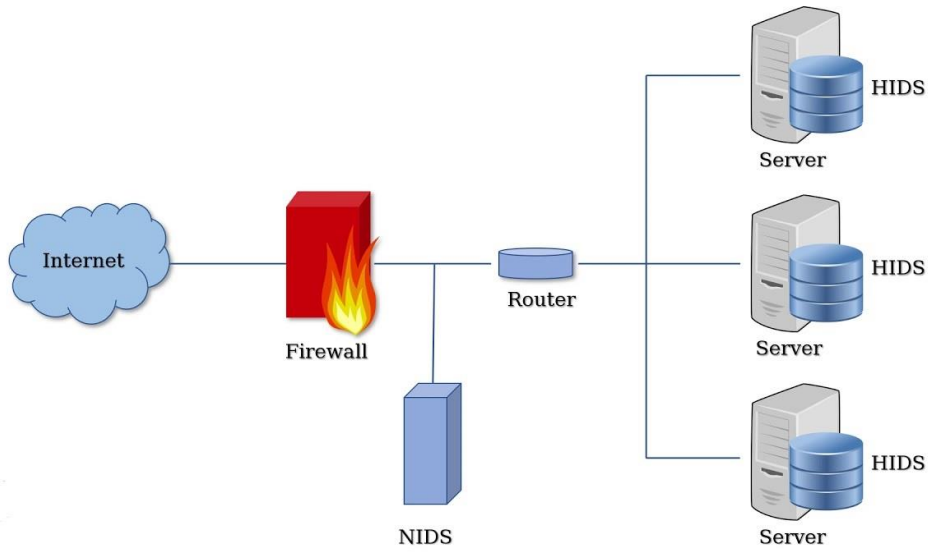

Figure 9: Exemple d'Hybride [21]

2.6 Architecture des IDS

L'architecture d'un système de détection d'intrusion (IDS) peut varier en fonction de divers facteurs, tels que le type d'IDS (réseau, hôte, hybride), la complexité du réseau surveillé et les besoins de sécurité spécifiques. Cependant, nous pouvons identifier certains composants courants couramment observés dans l'architecture IDS [20] :

2.6.1 Capteurs (Sensors)

Les capteurs sont les dispositifs chargés d'analyser le flux de données réseau ou l'activité du système. Dans un IDS de réseau, il est possible d'installer des capteurs à divers endroits du réseau afin d'analyser le trafic entrant et sortant. Les capteurs sont des éléments informatiques qui sont installés sur chaque machine à surveiller dans un hôte IDS.

2.6.2 Moteur d'analyse (Analysais Engine)

Le moteur d'analyse a pour mission d'examiner les informations recueillies par les capteurs afin de repérer les activités suspectes ou les infractions à la sécurité. Il a la possibilité d'utiliser des méthodes de détection qui reposent sur des signatures, des comportements anormaux ou une combinaison des deux.

2.6.3 Base de données de signatures (Signature Database)

La base de données de signatures est employée dans les IDS basés sur la détection basée sur la signature afin de stocker les modèles d'activité malveillante identifiés. Les données recueillies par les capteurs sont comparées à ces signatures par le moteur d'analyse afin de repérer les menaces identifiées.

2.6.4 Base de données de profil (Profile Database)

La base de données de profil est employée dans les IDS basés sur la détection basée sur l'anomalie afin de conserver les profils de comportement normal du réseau ou des systèmes. Ces profils sont utilisés par le moteur d'analyse afin de repérer les activités inhabituelles.

2.6.5 Console de gestion (Management Console)

Les administrateurs système peuvent configurer et surveiller l'IDS, visualiser les alertes générées, consulter les journaux d'événements et prendre des mesures de réponse en cas de détection d'une menace grâce à la console de gestion.

2.6.6 Système de corrélation (Correlation System)

Pour détecter les attaques coordonnées ou les modèles d'activité malveillante qui pourraient ne pas être détectés par un IDS individuel, il est possible d'utiliser un système de corrélation dans les environnements de sécurité plus complexes.

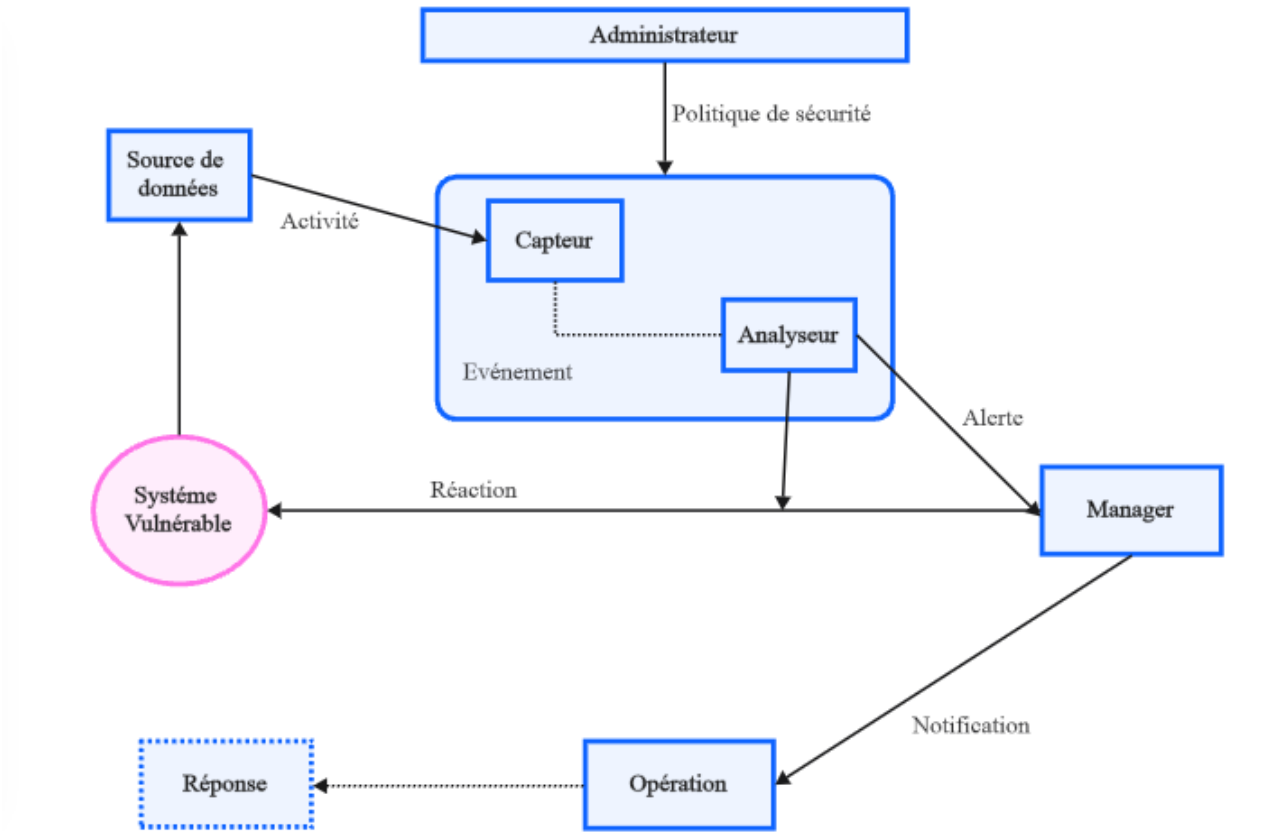

Figure 10: Architecture fonctionnelle d'un IDS[22]

2.7 Classification des systèmes de détection d'intrusion

Les systèmes de détection d'intrusion (IDS) peuvent être classés de différentes manières en fonction de plusieurs critères qui sont :[21]

2.7.1 La méthode de détection

- IDS basés sur les signatures : Ces IDS recherchent des modèles de signatures connus de menaces dans le trafic réseau ou les fichiers.
- IDS basés sur les anomalies : Ces IDS utilisent des modèles de comportement normal pour détecter les activités anormales ou suspectes.

2.7.2 Le comportement du système après la détection

- IDS passifs : Ces IDS se contentent de détecter les intrusions et de générer des alertes sans prendre d'action directe sur le système.

- IDS actifs : Ces IDS peuvent prendre des mesures pour répondre aux intrusions détectées, telles que bloquer le trafic suspect ou déconnecter un utilisateur.

2.7.3 La source des données

- IDS réseau (NIDS) : Ces IDS analysent le trafic réseau en temps réel pour détecter les activités suspectes.
- IDS hôte (HIDS) : Ces IDS surveillent les activités sur des hôtes individuels, tels que des serveurs ou des postes de travail.
- IDs application(AIDS) : Les informations à examiner proviennent directement d'une application, comme les fichiers de journalisation générés par les serveurs FTP et les serveurs Web. L'avantage de cette catégorie réside dans le fait que les données générées sont extrêmement synthétiques, riches et de volume modéré. En général, ces informations sont incluses dans les IDS basés sur l'hôte. [21]

2.6.4 La fréquence d'utilisation

- IDS en continu : Ces IDS fonctionnent en permanence, surveillant en temps réel le trafic réseau ou les activités des hôtes.
- IDS à la demande : Ces IDS sont utilisés de manière ponctuelle pour effectuer des analyses sur des journaux d'événements ou des captures de trafic en différé.

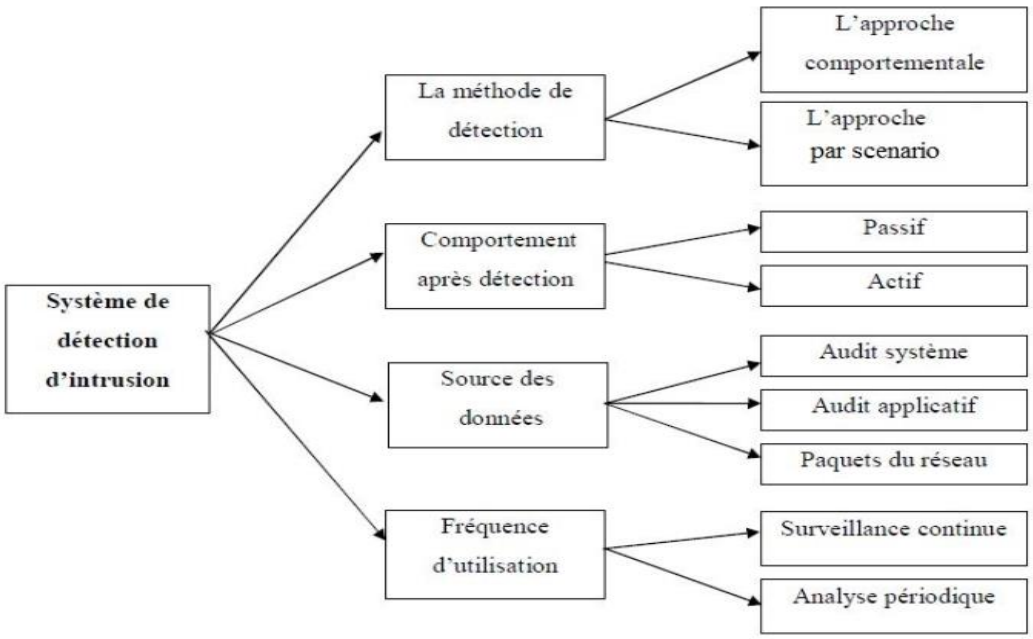

Figure 11: classification d'un système de détection d'intrusion [21]

2.7 L'efficacité des IDSs

Il existe cinq mesures qui nous permettent d'évaluer l'efficacité des systèmes de détection d'intrusion : [23]

Précision : La précision d'un IDS se réfère à sa capacité à identifier correctement les activités malveillantes tout en minimisant les faux positifs. Un IDS précis est crucial pour éviter de fausses alertes qui pourraient surcharger les équipes de sécurité avec des événements non pertinents

La performance de traitement : La qualité du traitement d'un système de détection d'intrusion est évaluée en fonction de la rapidité avec laquelle les événements d'audit sont traités. Sont pris en charge. Lorsque les performances de traitement du système de détection d'intrusion sont médiocres, il est impossible de détecter en temps réel. [23]

Complétude : La complétude mesure la capacité d'un IDS à détecter un large éventail de menaces et d'activités malveillantes. Un IDS complet est capable de détecter différents types d'attaques, des attaques basées sur des signatures aux attaques inconnues basées sur des comportements suspects.

Tolérance aux pannes : Cela se réfère à la capacité d'un IDS à continuer à fonctionner efficacement malgré les pannes matérielles ou logicielles. Un IDS résilient garantit une surveillance continue du réseau même en cas de défaillance d'une partie du système

Rapidité : La rapidité fait référence à la vitesse à laquelle un IDS peut détecter, signaler et réagir aux intrusions. Une réponse rapide est essentielle pour minimiser les dommages causés par les attaques et pour réduire leur impact sur les opérations commerciales

2.8 Conclusion

En conclusion, les systèmes de détection d'intrusion (IDS) sont des outils essentiels pour protéger les réseaux informatiques contre les menaces et les attaques. Leur capacité à surveiller en temps réel le trafic réseau et les activités des systèmes hôtes permet de détecter rapidement les comportements anormaux et les tentatives d'intrusion. En alertant les administrateurs de sécurité, les IDS permettent une réponse rapide aux incidents, contribuant ainsi à renforcer la sécurité des données et à assurer la continuité des opérations commerciales. Toutefois, pour garantir leur efficacité, il est crucial de choisir une solution adaptée, de la déployer correctement et de la maintenir régulièrement à jour. En intégrant les IDS dans une stratégie de sécurité globale comprenant d'autres outils, des politiques appropriées et une sensibilisation à la sécurité, les organisations peuvent mieux se protéger contre les menaces informatiques et les attaques potentielles.

Chapitre 03 : Architecture & Conception

3.1 Introduction

Ce chapitre présente une proposition de système de détection d'intrusion. Tout d'abord, nous exposons notre structure du système, ensuite nous exposons le langage de programmation, l'environnement de travail et les outils utilisés pour mettre en place cette IDS (Weka ,Matlab). En ce qui concerne les données d'apprentissage, nous utilisons les fichiers ARFF de NSL-KDD. Finalement, on présente le fonctionnement de notre système et sa fiabilité dans la détection d'intrusion en démontrant certaines mesures.

3.2 L'outil de développement

MATLAB est un langage de programmation multi-paradigmes propriétaire et un environnement informatique numérique développé par MathWorks. MATLAB permet les manipulations matricielles, le traçage de fonctions et de données, la mise en œuvre d'algorithmes, la création d'interfaces utilisateur et l'interface avec des programmes écrits dans d'autres langages. Wikipedia , Nous l'avons utilisé pour "FIREFLY"

weka(Waikato Environment for Knowledge Analysis) est une suite de logiciels d'apprentissage automatique écrite en Java et développée à l'université de Waikato en Nouvelle-Zélande. Weka est un logiciel libre disponible sous la Licence publique générale GNU. [26] Nous l'avons utilisé pour" KDDNSL"

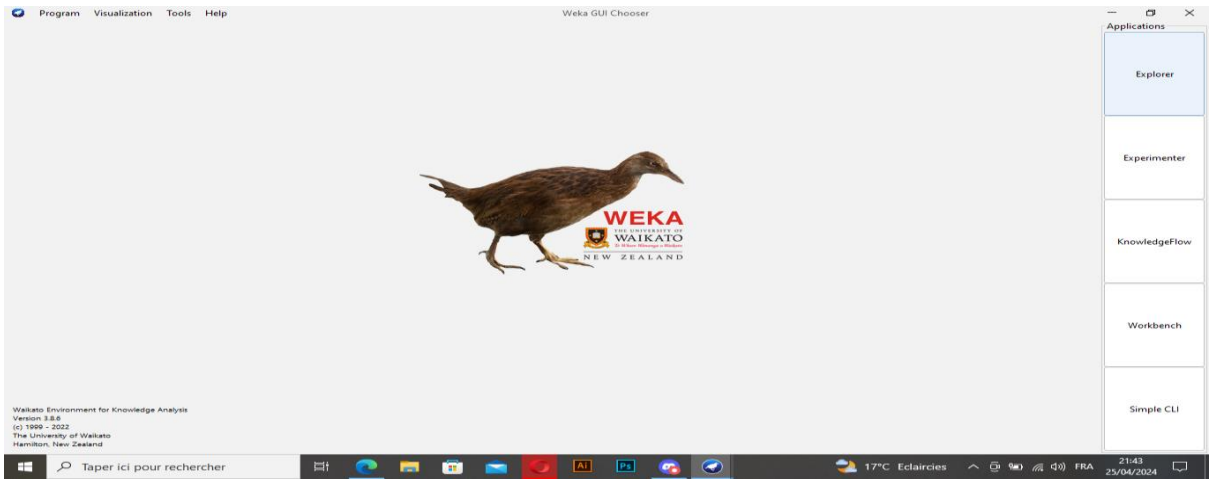

Figure 12: Interface graphique de Weka

3.3 Dataset NSL-KDD

Pour détecter l'intrusion, on utilise le dataset NSL-KDD, qui est composé de données sélectionnées provenant de l'ensemble du dataset KDD'99. En utilisant NSL-KDD (Network Security Layer-Knowledge Discovery in Database)[27] , on réduit la surcharge de données dans le dataset original KDD'99, ce qui rend les données plus réalistes pour détecter les attaques. Il y a 41 caractéristiques et 1 classe identifiée comme normal ou anomalie dans le dataset, et elle est utilisée pour évaluer un total de 24 types d'attaques de formation dans le groupe de formation et un autre 14 types d'attaques dans les données de test.

La mise en œuvre de NSL-KDD dans le domaine de la recherche en sécurité informatique a entraîné l'émergence de nouvelles méthodes et méthodes pour détecter les intrusions. Cet ensemble de données est utilisé par les chercheurs pour former des modèles d'apprentissage automatique, évaluer les performances des systèmes de sécurité, développer de nouvelles techniques de détection des attaques dans les réseaux.

Étant donné que le modèle est vérifié pour des attaques inconnues, la détection est désormais plus réaliste. Toutes les données de formation et de test dans le dataset NSL-KDD sont numérotées de 1 à 21, qui indiquent la gravité de l'attaque. Le système d'apprentissage trouve que l'attaque numéro 21 est la plus facile à repérer, l'attaque numéro 1 est la plus difficile, et d'autres attaques se trouvent entre elles. De plus, une partie du dataset NSL-KDD est dépourvue de tout dataset ayant un niveau de difficulté 21 sur 21.

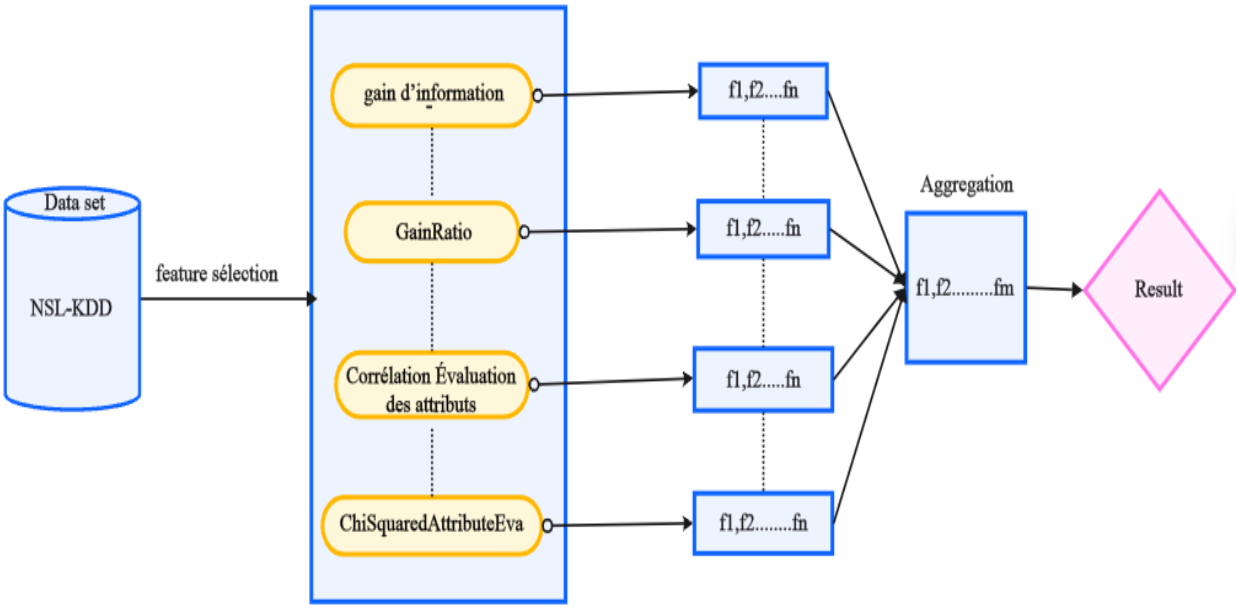

Figure 13: Architecture de Dataset KDD-NSL

Les Fichiers renseignements

Plusieurs enquêtes statistiques ont exposé les faiblesses intrinsèques de l'ensemble de données KDD-Cup-99 , qui ont eu un impact sur la précision de détection de nombreux modèles IDS.
enquêteur. Le jeu de données NSL-KDD est son précurseur. Tous les enregistrements fondamentaux de l'ensemble de données KDD y sont contenus.
Les chercheurs peuvent télécharger un certain nombre de fichiers différents. Le tableau 1 en contient une liste.

KDDTrain+.ARFF	Le train plein NSL-KDD fixé avec des étiquettes binaires au format ARFF
KDDTrain+.TXT	La rame NSL-KDD complet, y compris les étiquettes de type d'attaque et le niveau de difficulté au format CSV
KDDTrain+_20Percent.ARFF	Un sous-ensemble de 20% du dossier de l'KDDTrain +fichier .arff
KDDTrain+_20Percent.TXT	Un sous-ensemble de 20% du + fichier .txtKDDTrain
KDDTest+.ARFF	Le test complet NSL-KDD fixer avec des étiquettes binaires au format ARFF
KDDTest+.TXT	L'ensemble de test NSL-KDD complet, y compris les étiquettes de type d'attaque et le niveau de difficulté au format CSV
KDDTest-21.ARFF	Un sous-ensemble du dossier de l'KDDTest + qui ne comprennent pas les documents avec le niveau de difficulté de 21 sur 21
KDDTest-21.TXT	Un sous-ensemble du fichier KDDTest + .txt qui ne comprennent pas les documents avec le niveau de difficulté de 21 sur 21

Tableau 1: Liste des fichiers dans le système de données NSL-KDD, avec une description [38]

3.4 L'architecture de NSL-KDD

Pour permettre aux classificateurs de générer une sortie impartiale, les enregistrements redondants sont éliminés.Il existe une quantité adéquate d'enregistrements dans les ensembles de données de formation et de test, ce qui est logique et permet de mener des recherches sur l'ensemble de la collectionLa quantité d'entrées choisies dans chaque groupe de niveaux difficiles est inversement proportionnelle à la proportion d'entrées dans la collecte initiale de données KDD.Chaque enregistrement contient 41 attributs qui révèlent divers aspects du flux, chacun avec un nom le désignant soit comme un type typique, soit comme un type d'agression.Les tableaux **2**, **3**, **4** et **5** contiennent une liste des spécificités des qualités, y compris leur nom, leur description et des exemples de données. Les informations de type pour chacun des 41 attributs inclus dans la collecte de données NSL-KDD sont fournies dans le tableau **6**.Les données concernant les cinq classes différentes de vecteurs de connexion réseau (une classe normale et quatre classes d'attaque) sont

rassemblées dans la propriété 42. Les quatre classes d'attaques sont en outre divisées en groupes DoS, Probe, R2L et U2R. La façon dont les classes d'assaut sont décrites.

Attribute No.	Attribute Name	Description
1	Duration	Longueur de la connexion.
2	Protocol_type	Protocole de connexion.
3	service	Service de destination.
4	flag	Indicateur d'état de la connexion.
5	Src_bytes	Octets envoyés de la source à la destination
6	Dst_bytes	Octets envoyés de la destination à la source
7	land	1 si vient de/vers le même hôte/port ; sinon 0
8	Wrong_fragment	Nombre de mauvais fragments
9	Urgent	Nombre de paquets urgents

Tableau 2: Les fonctions essentielles de chaque élément de connexion réseau [27]

Attribute No.	Attribute Name	Description
10	Hot	Nombre d' indicateurs « hot » dans le contenu tel que: entrer dans un répertoire système, créer programmes et exécution programmes
11	Num_failed_logins	tentatives de connexion
12	Logged_in	Statut de connexion : 1 si avec succès connecté;sinon 0
13	Num_compromised	Nombre de conditions compromises
14	Root_shell	1 si la coque racine est obtenue ; sinon 0
15	Su_attempt ed	1 si la commande su root a été tentée ; sinon 0
16	Num_root	Nombre d'accès root
17	Num_file_creations	Nombre d'opérations de création de fichiers
18	Num_shells	Nombre d'invités du shell
19	Num_access_files	Nombre d'opérations sur les fichiers de contrôle d'accès
20	Num_outbound_cmds	Nombre de commandes sortantes dans une session FTP
21	Is_hot_login	1 si le login appartient à la hot list ; 0 sinon
22	Is_guest_login	1 si la connexion est une connexion invité ; sinon 0

Tableau 3: Caractéristiques de chaque vecteurs de connexion réseau liés au contenu [27]

Attribute No.	Attribute Name	Description
23	Count	Nombre de connexions à le même destination hôte en tant que actuel connexion dans les deux derniers
24	Srv_count	Nombre de connexions à le même service (port numéro) comme le actuel connexion dans les deux derniers secondes
25	Serror_rate	Le pourcentage de connexions qui ont activé le drapeau (4) s0, s1, s2 ou s3, parmi les Connexions regroupés dans compter (23)
26	Srv_serror_rate	Le pourcentage de connexions qui ont activé le drapeau (4) s0, s1, s2 ou s3, parmi les Connexions regroupés dans srv_count (24)
27	Rerror_rate	Le pourcentage de connexions qui ont activé le drapeau (4) REJ, parmi les Connexions regroupés dans compter (23)
28	Srv_rerror_rate	Le pourcentage de connexions qui ont activé le drapeau (4) REJ, parmi les Connexions regroupés dans srv_count (24)
29	Same_srv_rate	Le pourcentage de connexions c'était à le même service, parmi le Connexions regroupés dans compter (23)
30	Diff_srv_rate	Le pourcentage de connexions vers différents services, parmi les connexions regroupées en nombre (23)
31	Srv_diff_host_ rate	Le pourcentage de connexions qui étaient vers des machines de destination différentes parmi les connexions agrégées dans srv_count (24)

Attribute No.	Attribute Name	Description
32	Dst_host_coun t	Nombre de Connexions avoir le même destination adresse IP de l'hôte
33	Dst_host_srv_ count	Nombre de Connexions avoir le même port nombre
34	Dst_host_same _srv_rate	Le pourcentage de connexions qui étaient au même service, parmi les Connexions regroupés dans dst_host_count (32)
35	Dst_host_diff_ srv_rate	Le pourcentage de connexions c'était à différent prestations de service, parmi les Connexions dst_host_count(32)
36	Dst_host_same_src_port_rate	Le pourcentage de connexions qui étaient au même source port, parmi le Connexions regroupés dans dst_host_srv_c nombre (33)
37	Dst_host_srv_diff_host_rate	Le pourcentage de connexions c'était à différent destination Machines, parmi les Connexions regroupés dans dst_host_srv_c
38	Dst_host_serro r_rate	Le pourcentage de connexions qui ont activé le drapeau (4) s0, s1, s2 ou s3, parmi les Connexions regroupés dans dst_host_count (32)
39	Dst_host_srv_s error_rate	Le pourcentage de Connexions qui ont activé le drapeau (4) s0, s1, s2 ou s3, parmi les Connexions regroupés dans dst_host_srv_c nombre (33)
40	Dst_host_rerro r_rate	Le pourcentage de connexions qui ont activé le drapeau (4) REJ, parmi les Connexions regroupés dansdst_host_count (32)

Tableau 5: Caractéristiques de chaque vecteurs de connexion réseau liés au temps [27]

Type	Caractéristiques
Nominal	Protocol_type(2), Service(3), Flag(4)
Binaire	Land(7), logged_in(12), root_shell(14), su_attempted(15), is_host_login(21),, is_guest_login(22)
Numérique	Duration(1), src_bytes(5), dst_bytes(6), wrong_fragment(8), urgent(9), hot(10), num_failed_logins(11), num_compromised(13), num_root(16), num_file_creations(17), num_shells(18), num_access_files(19), num_outbound_cmds(20), count(23) srv_count(24), serror_rate(25), srv_serror_rate(26), rerror_rate(27), srv_rerror_rate(28), same_srv_rate(29) diff_srv_rate(30), srv_diff_host_rate(31), dst_host_count(32), dst_host_srv_count(33), dst_host_same_srv_rate(34), dst_host_diff_srv_rate(35), dst_host_same_src_port_rate(36), dst_host_srv_diff_host_rate(37), dst_host_serror_rate(38), dst_host_srv_serror_rate(39), dst_host_rerror_rate(40), dst_host_srv_rerror_rate(41)

Tableau 6: Caractéristiques d'un réseau de connexion vectorielle basé sur un hôte [27]

3.5 KDD Cup 1998 Data

Prévention des intrusions réseaux et protection d'un réseau contre les utilisateurs non autorisés. L'apprentissage du détecteur d'intrusions consiste à élaborer un modèle prédictif (ou classificateur) qui permet de différencier les connexions "mauvaises", connues sous le nom d'intrusions ou d'attaques, des connexions "bonnes" normales.
Le dataset contient une grande quantité de données de trafic réseau qui illustrent un environnement informatique de réseau. **MIT Lincoln Lab** l'a développé en utilisant des données brutes de TCP pour simuler un réseau local (LAN) pendant 9 semaines. La simulation du trafic de réseau englobe diverses formes d'attaques, ainsi que des activités normales.[28]
Les classes d'attaques dans l'ensemble des données NSL-KDD sont classées en quatre catégories principales en fonction du type d'attaques qu'elles représentent. Le **tableau 7** met en évidence ce point. Ces groupes comprennent [29][37] :

1. Déni de Service (DoS) : Ce type d'attaque englobe les tentatives de submerger un système ou un réseau cible avec un flux de demandes, le rendant incapable de répondre aux utilisateurs légitimes. L'attaque SYN flood, l'attaque UDP flood et l'attaque HTTP flood sont des exemples d'attaques DoS.

2. Probe : Les activités de reconnaissance englobent les tentatives d'un attaquant de collecter des données sur un réseau cible afin de repérer des vulnérabilités ou des points d'entrée potentiels. Les attaques par sonde comprennent la recherche de ports et l'enregistrement de l'OS.

3. Utilisateur à Administrateur (U2R) : Ce type d'attaque englobe les tentatives d'un utilisateur non autorisé ayant des privilèges restreints de s'élever pour obtenir un accès de niveau

administrateur sur un système. Les attaques U2R comprennent des exploits de dépassement de tampon et des attaques d'escalade de privilèges.

4. À Distance à Local (R2L) : Les attaques de cette catégorie comprennent les tentatives d'un attaquant d'obtenir un accès non autorisé à un système cible depuis un lieu éloigné. En général, ces attaques utilisent des failles dans les services réseau afin d'accéder. Les attaques R2L peuvent se manifester par le déchiffrement de mots de passe et l'utilisation de mécanismes d'authentification déficients.

Classe d'attaque	**Type d'attaque**
DoS	Back, Land, Neptune, Pod, Smurf,Teardrop,Apache2, Udpstorm, Processtable, Worm (10)
Probe	e Satan, Ipsweep, Nmap, Portsweep, Mscan, Saint (6)
R2L	Guess_Password, Ftp_write, Imap, Phf, Multihop, Warezmaster, Warezclient, Spy, Xlock, Xsnoop, Snmpguess, Snmpgetattack, Httptunnel, Sendmail, Named (16)
U2R	Buffer_overflow, Loadmodule, Rootkit, Perl, Sqlattack, Xterm, Ps (7)

Tableau 7: Cartographie de l'attaque de la class et du type [39]

Il est important de souligner que les données de test diffèrent des données d'entraînement en ce qui concerne la distribution de probabilité. Elles incluent des types spécifiques d'attaques qui ne sont pas présents dans les données d'entraînement. Cela rend les tâches plus concrètes. Selon certains spécialistes de l'intrusion, la majorité des informations sont erronées.
Une attaque est une variation d'une attaque déjà existante, et la "signature" d'une attaque déjà existante peut être utilisée pour capturer de nouveaux variants. Au total, l'ensemble de données comprend 24 types d'attaques en cours de formation, ainsi que 14 types additionnels.Seulement des données de test**.**

3.6 classification des algorithmes de sélection de fonctionnalités

Nous présentons les algorithmes bien connus qui ont été développés dans la sélection de fonctionnalités pour détection d'intrusion basée sur cette taxonomie. Nous spécifions l'approche de sélection de l'algorithme entre filtre(gain d'information, Correlation Coefficient (CC),Chi-Squared,, and Gain Ratio,...), wrapper (firefly). Dans chaque méthode, nous définissons leurs techniques utilisées avec les résultats obtenus.

I. Méthode du filtre

3.6.1.1 Méthode gain d'information

La méthode nommée InfoGainAttributeEval évalue la valeur d'un attribut en mesurant le gain d'information par rapport à la classe. [30]

Infogain (classe, attribut) = H (Classe) - H (Classe | Attribut)

Donc, le gain d'information mesure l'efficacité d'un attribut A avec la réduction de l'entropie. Pour calculer l'entropie E :

Soit S l'ensemble d'exemples. Supposons que l'attribut à prédire prenne M valeurs distinctes définissant M classes C1,…,CM L'entropie E(S) est définie par :

$$E(S) = -\sum_{i=1}^{M} p_i \log_2 p_i$$

Où pi désigne la proportion d'exemples de S appartenant à la i-ème C

A partir de l'entropie, on peut calculer InfoGain :

$$InfoGain(S, A) = E(S) - \sum_{K}^{K=m} \frac{|S_K|}{|S|} \mathrm{E}(S_K)$$

Où l'attribut A prend les valeurs a1,.., am et Sk est le sous-ensemble de S pour lequel l'attribut A prend la valeur ak. [31]

Pour l'évaluation des attributs du gain d'information, nous traitons l'ensemble de formation NSL-KDD et récupère les résultats. Cet algorithme utilise méthode des classificateurs sur les caractéristiques et évalue les caractéristiques en les classant du plus important au moins important. [30]

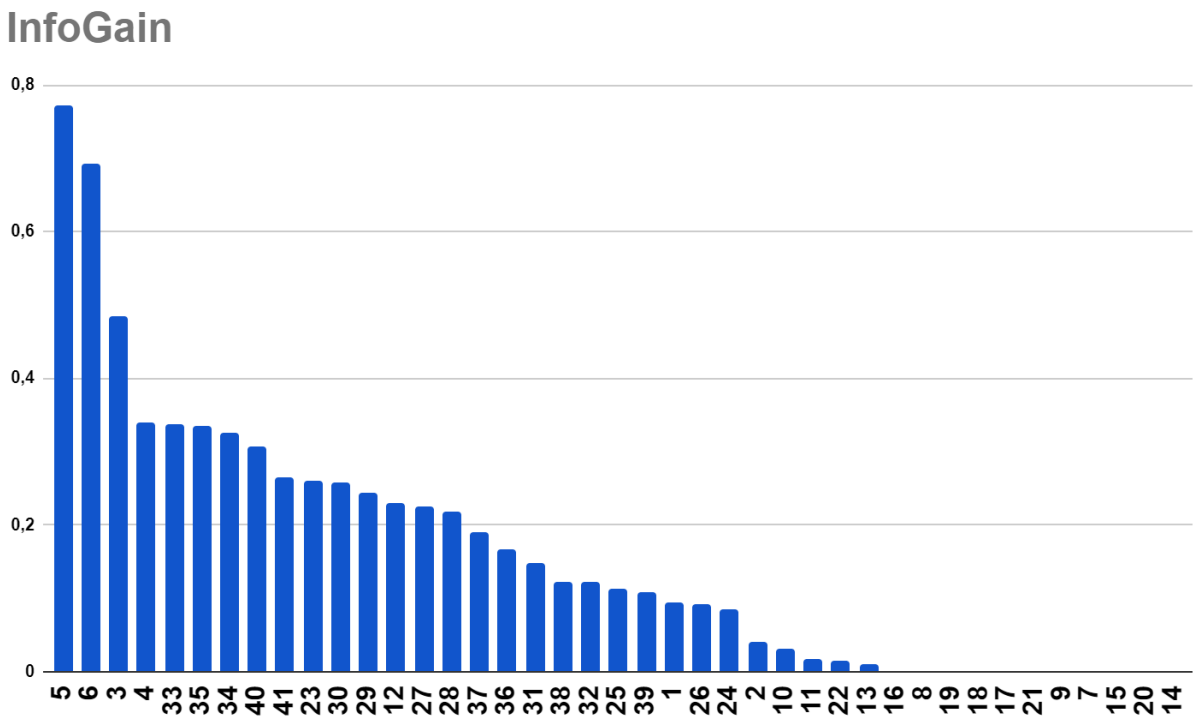

Figure 14: Gain d'information Évaluation des attributs

Le tableau 8 : montre le score de gain d'informations pour les 10 principales fonctionnalités sur la base du calcul effectué pendant la phase de formation à l'aide du filtre InfoGain calculé à l'aide des formules mentionnées précédemment.

N° d'attribut	5	6	3	4	33	35	34	40	41	23
InfoGain	0.773	0.693	0.485	0.341	0.337	0.334	0.326	0.307	0.266	0.26

Tableau 8: les meilleurs attributs selon InfoGainAttrEval

3.6.1.2 Méthode GainRatio

La méthode nommée GainRatioAttributeEval évalue la valeur d'un attribut en mesurant le rapport de gain par rapport à la classe, [31]Il utilise une extension du gain d'information en utilisant le GainRatio . [30]

GainRatio (classe, attribut) = (H (Classe) - H (Classe | Attribut)) / H (Attribut).

En introduisant une information de partitionnement, GainRation ajuste InfoGain avec l'entropie du partitionnement. On peut le calculer avec les formules suivantes :[31]

$$SplitInfo(S, A) = E(S) - \sum_{K}^{K=m} \frac{|S_k|}{|S|} log_2 \frac{|S_k|}{|S|}$$

$$GainRatio(S, A) = \frac{InfoGain(S,A)}{SplitInfo(S,A)}$$

Nous utilisons des données d'entraînement et appliquons l'algorithme d'évaluation des attributs du rapport de gain sur les données, cet algorithme utilise la méthode des classificateurs sur les caractéristiques et évalue les caractéristiques en les classant. La mesure du gain d'information est biaisée en faveur des tests comportant de nombreux résultats.
GainRatio préfère sélectionner des attributs ayant un grand nombre de valeurs. Il utilise une extension du gain d'information.[30]

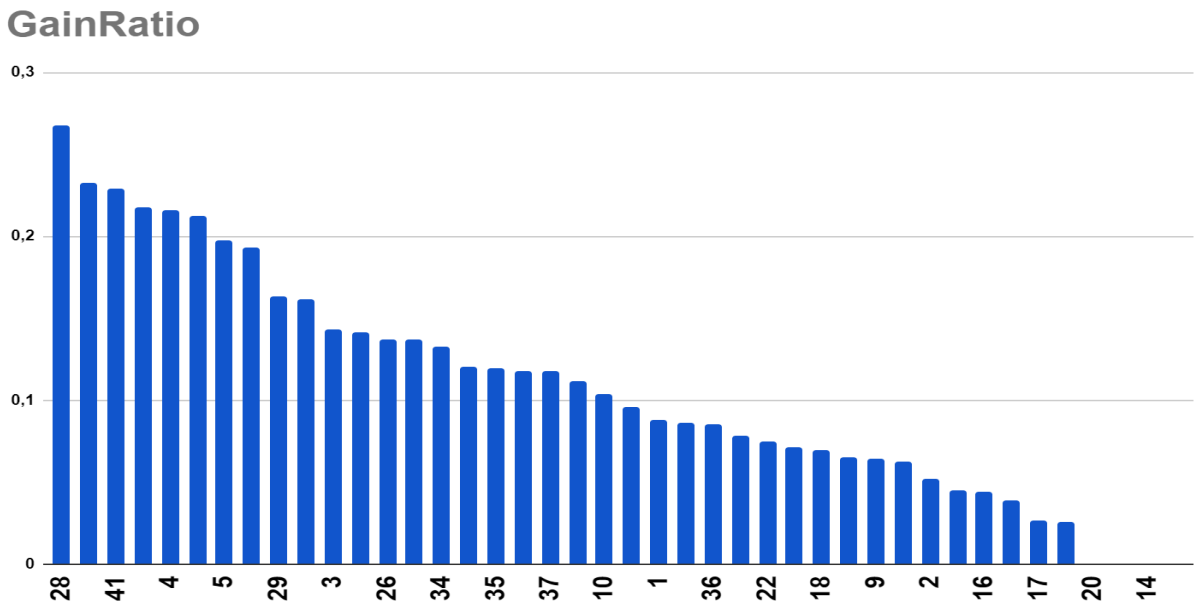

Figure 15: Évaluation des attributs du rapport de gain

Le tableau 9 : montre le score obtenus pour les 10 meilleurs attributs selon GainRatio

N° d'attribut	28	12	41	27	4	6	5	30	29	40
GainRatio	0.268	0.232	0.229	0.218	0.216	0.213	0.198	0.193	0.163	0.162

Tableau 9: les meilleurs attributs selon GainRatioAttributeEval

3.6.1.3 Corrélation Évaluation des attributs

La corrélation spécifie la dépendance des caractéristiques les unes par rapport aux autres. Elle représente la relation linéaire entre les variables ou les caractéristiques. Évaluation de l'attribut de corrélation, cet algorithme classe les caractéristiques de l'ensemble de formation NSL-KDD en fonction de leur corrélation. entre elles, la corrélation spécifie la dépendance des caractéristiques les unes par rapport aux autres.[30]

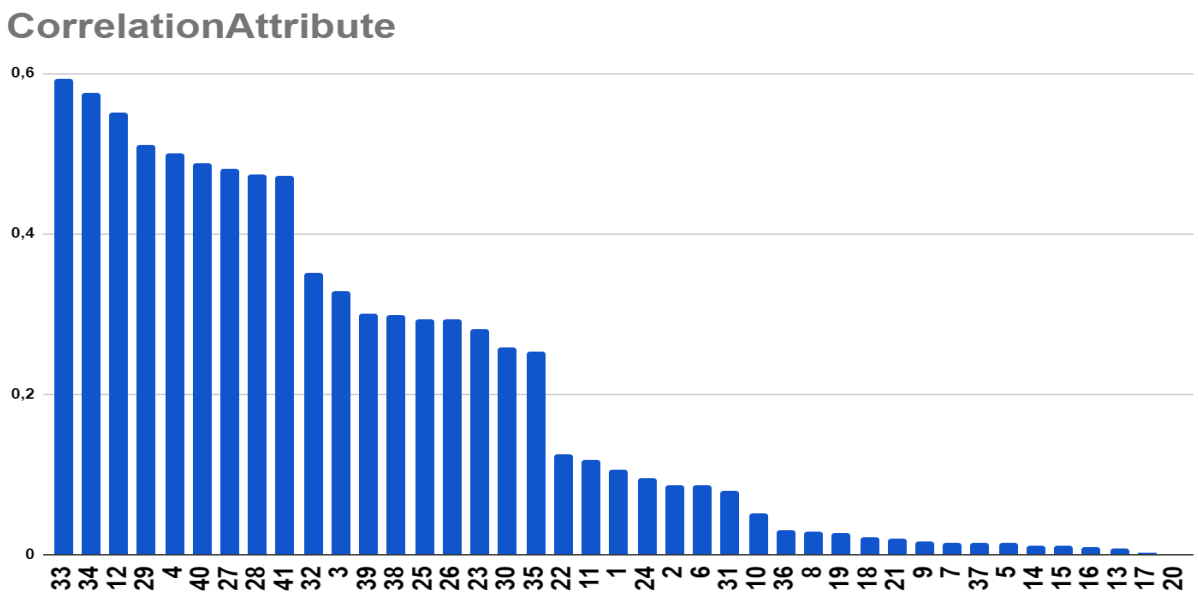

Figure 16: Correlation Attribute Evaluation

tableau 10 :montre le score obtenus pour les 10 meilleurs attributs selon CorrelationAttributesEval

N° d'attribut	33	34	12	29	4	40	27	28	41	32
CAE	0.593	0.576	0.551	0.511	0.5	0.488	0.481	0.475	0.473	0.352

Tableau 10: les meilleurs attributs selon CorrelationAttributesEval

3.6.1.4 ChiSquaredAttributeEval

Évalue la valeur d'un attribut en calculant la valeur de la statistique chis-quarée par rapport à la classe. [32] Il permet de calculer la probabilité que les lignes et les colonnes d'un tableau croisé soient autonomes. Cela veut dire que :

- L'appartenance à l'une des modalités de la première variable n'affecte pas la modalité d'appartenance de la seconde variable ;
- Les pourcentages des lignes du tableau croisé sont identiques pour toutes les lignes ;
- Les pourcentages des colonnes du tableau croisé sont identiques pour toutes les colonnes. [33]

La méthode du Khi-carré classe les caractéristiques en fonction de leur score et, comme pour les autres méthodes de sélection des caractéristiques, nous choisissons les caractéristiques les plus utiles,[34]

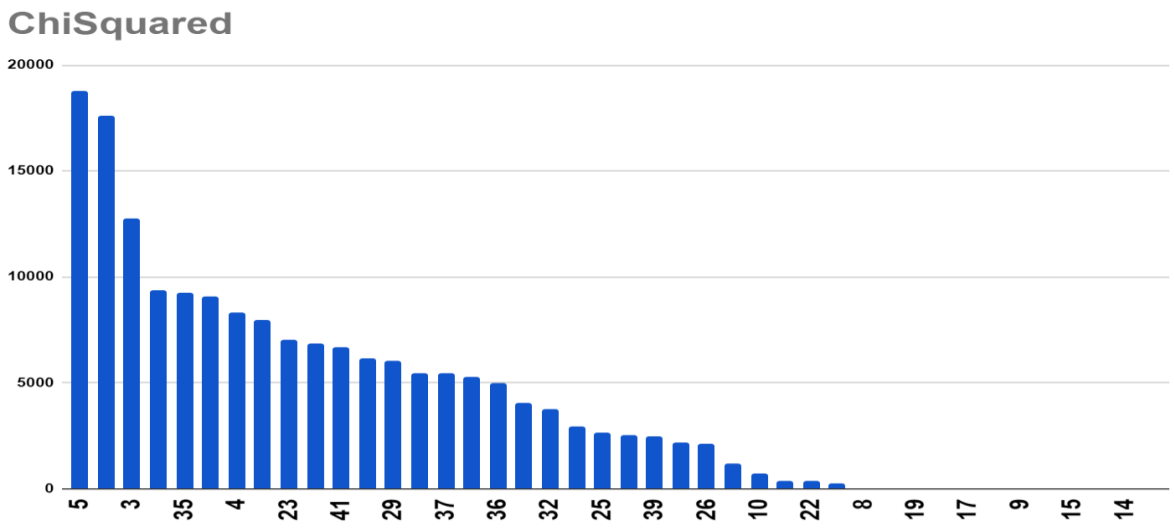

Figure 17: ChiSquaredAttributeEval

tableau 11 :montre le score obtenus pour les 10 meilleurs attributs selon ChiSquaredAttributeEval

N° d'attribut	5	6	3	33	35	34	4	40	23	12
chisquared	18756.114	17603.944	12765.71	9351.305	9238.184	9081.41	8300.685	7971.536	7020.502	6848.328

Tableau 11: les meilleurs attributs selon ChiSquaredAttributeEval

3.6.1.5 CfsSubsetEval

Cette méthode évalue la valeur d'un sous-ensemble d'attributs en considérant la capacité individuelle de chaque caractéristique ainsi que le degré de redondance entre eux. Les sous-ensembles de caractéristiques qui sont corrélation avec la classe tout en ayant une faible intercorrélation avec les autres attributs sont préférés.[32] Pour obtenir les sous-ensembles d'attributs, nous avons employé le logiciel open source Weka en utilisant le filtre CfsSubsetEval, en utilisant quatre algorithmes de recherche distincts.[33]

- **BestFirst:**Cette méthode permet de rechercher l'espace des sous-ensembles d'attributs par escalade gourmande augmentée d'une fonction de retour en arrière. La définition du nombre de nœuds consécutifs non améliorants autorisés contrôle le niveau de retour en arrière effectué. Meilleur d'abord peut commencer par un ensemble vide d'attributs et rechercher vers l'avant, ou commencer avec l'ensemble complet des attributs et rechercher vers l'arrière, ou commencer à n'importe quel point et dans les deux sens (en tenant compte de tous les tous les ajouts et suppressions possibles d'un seul attribut à un point donné). [32]
- **GeneticSearch:**Il effectue une recherche à l'aide de l'algorithme génétique simple. [32]
- **GreedyStepwise:**Il effectue une recherche avide vers l'avant ou vers l'arrière dans l'espace des sous-ensembles d'attributs. Peut commencer avec aucun/toutes les attributs ou à partir d'un point arbitraire de l'espace. Elle s'arrête lorsque l'ajout/la suppression de tout attribut restant entraîne une diminution de l'évaluation. Peut également produire une liste d'attributs classés en parcourant l'espace d'un côté à l'autre de l'espace et en enregistrant l'ordre dans lequel les attributs sont sélectionnés. [32]

3.7 Résultats et discussions

Nous avons utilisé Weka (3.8.6), un outil de machine learning, pour réaliser l'analyse comparative. Dans ce document, différentes combinaisons de méthodes de sélection des caractéristiques sont essayées et incluent: [32]

N°	Méthode de sélection des caractéristiques	Nombre de caractéristiques sélectionnées	Caractéristiques sélectionnées
1	Bestfirst+CFSSubsetEval	9 feature	f5,f6,f12,f25,f28,f30,f31,f37,f41
2	GeneticSearch+CFSSubsetEval	21 feature	f1,f4,f5,f6,f11,f12,f17, f18,f22,f25,f26,f27,f28, f29,f30,f31,f32,f35,f36,f37,f41
3	GreedyStepwise+CFSSubsetEval	9 feature	f5,f6,f12,f25,f28,f30,f31,f37,f41
4	Ranker+InfoGainAttributeEval	10 feature	f5,f6,f3,f4,f33,f35,f34,f40,f41,f23
5	Ranker+GainRatioAttributeEval	10 feature	f28,f12,f41,f27,f4,f6,f5,f30,f29,f40
6	Ranker+ChiSquaredAttributeEval	10 feature	f5,f6,f3,f33,f35,f34,f4,f40,f23,f12
7	Ranker+CorrelationAttributesEval	10 feature	f33,f34,f12,f29,f4,f40,f27,f28,f41,f32

Tableau 12: Liste des caractéristiques sélectionnées par différentes méthodes de sélection des caractéristiques

3.8 Apprentissage automatique discriminatif Algorithme

Sur l'ensemble de données réduit, nous avons appliqué plusieurs algorithmes d'apprentissage automatique discriminants. pour entraîner les algorithmes d'apprentissage automatique et l'ensemble de données de test est donné séparément. L'utilisation d'un ensemble de données d'entraînement et d'un ensemble de données de test séparés nous donne l'avantage de vérifier la précision de la détection des attaques, même pour les attaques inconnues. car l'ensemble d'entraînement contient 24 types d'attaques et l'ensemble de test contient des données supplémentaires. et l'ensemble de test contient 14 attaques supplémentaires par rapport aux 24 attaques précédentes. La détection est donc plus précise, car le modèle est également vérifié pour les attaques inconnues. [30]

3.8.1 Naive Bayes

Le classificateur Naive Bayes est un groupe de classificateurs simples en utilisant le théorème de probabilité de Bayes avec une forte hypothèse d'indépendance entre les caractéristiques de ce qui fait l'objet d'une classification binaire (avec deux états “ oui ou non”). [35]

3.8.2 J48

J48 est un algorithme basé sur un arbre de décision de l'algorithme C4.5, introduit par Ross Quinlan. Son objectif principal est de travailler sur l'apprentissage supervisé, la classification pour produire un arbre de décision. [36]

3.8.3 Random Forest

la méthode Random Forest (RF). RF est un classificateur d'ensemble qui combine les résultats de différents modèles utilisant de nombreux modèles d'arbres aléatoires. Ici, il n'est pas nécessaire d'élaguer arbres et overfitting n'est pas un problème. [35]

3.8.4 Random Tree

Random Tree (RT). L'arbre aléatoire utilise un certain nombre d'attributs choisis au hasard à chaque nœud d'un arbre de décision. Il s'agit d'un modèle prédictif qui utilise un ensemble de règles binaires et peut être utilisé pour des applications de classification ou de régression. Il est assez facile d'interpréter les règles de décision. La classification est rapide une fois les règles conçues. [35]

3.9 Résultats et performances comparaison

1. **Le tableau 13 :** présente les résultats, sans sélection de caractéristiques, sur l'ensemble de données NSL-KDD [2] avec 41 caractéristiques. Les résultats en termes de précision de détection pour différents algorithmes d'apprentissage des benchmarks précédents. [30]

Classifier	Detection Accuracy (%)	Incorrectly Classified Instances	Total Number of Instances	Root mean squared error
Naive Bayes	81.1003 %	18.8997 %	2254	87.3487 %
J48	98.5359 %	1.4641 %	2254	22.3715 %
Random Forest	98.7048 %	1.2952 %	22544	19.9075 %
Random Tree	98.1547 %	1.8453 %	22544	26.9359 %

Tableau 13: Précision de détection sans sélection de caractéristiques

2. Les tableaux suivants présentent les résultats obtenus après la sélection des caractéristiques sur l'ensemble de données NSL-KDD. [30]

2.1 CfsSubsetEval avec l'algorithme de la meilleure première recherche (9 attributs)

Classifier	Detection Accuracy (%)	Incorrectly Classified Instances	Total Number of Instances	Root mean squared error
Naive Bayes	79.1563 %	20.8437 %	22544	91.4038 %
J48	97.8087 %	2.1913 %	22544	27.3039 %
Random Forest	97.822 %	2.178 %	22544	24.9067 %
Random Tree	97.6002 %	2.3998 %	22544	28.113 %

Tableau 14: Résultats de l'évaluation Bestfirst+CFSSubset

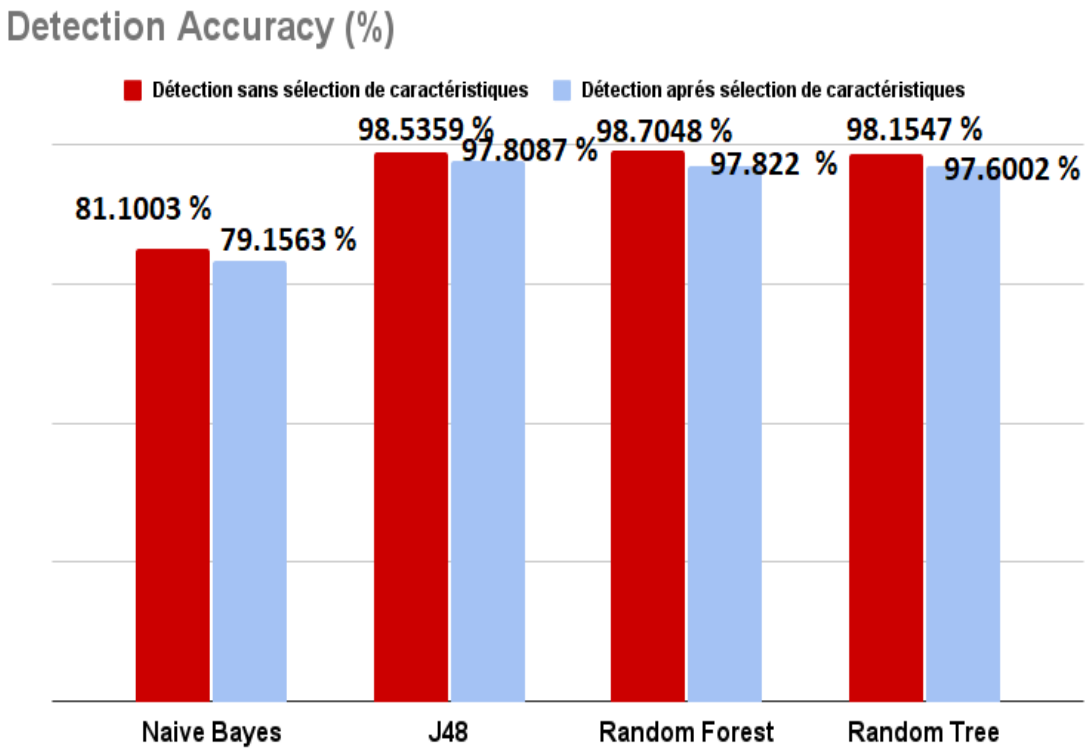

Figure 18: Résultats de l'évaluation Bestfirst+CFSSubset

2.2 CfsSubsetEval avec l'algorithme de la meilleure première recherche (11 attributs)

Classifier	Detection Accuracy (%)	Incorrectly Classified Instances	Total Number of Instances	Root mean squared error
Naive Bayes	78.7216 %	21.2784 %	22544	92.6079 %
J48	98.1592 %	1.8408 %	22544	24.9021 %
Random Forest	98.2967 %	1.7033 %	22544	22.339 %
Random Tree	97.7821 %	2.2179 %	22544	28.5324 %

Tableau 15: Résultats de l'évaluation GeneticSearch+CFSSubsetEval

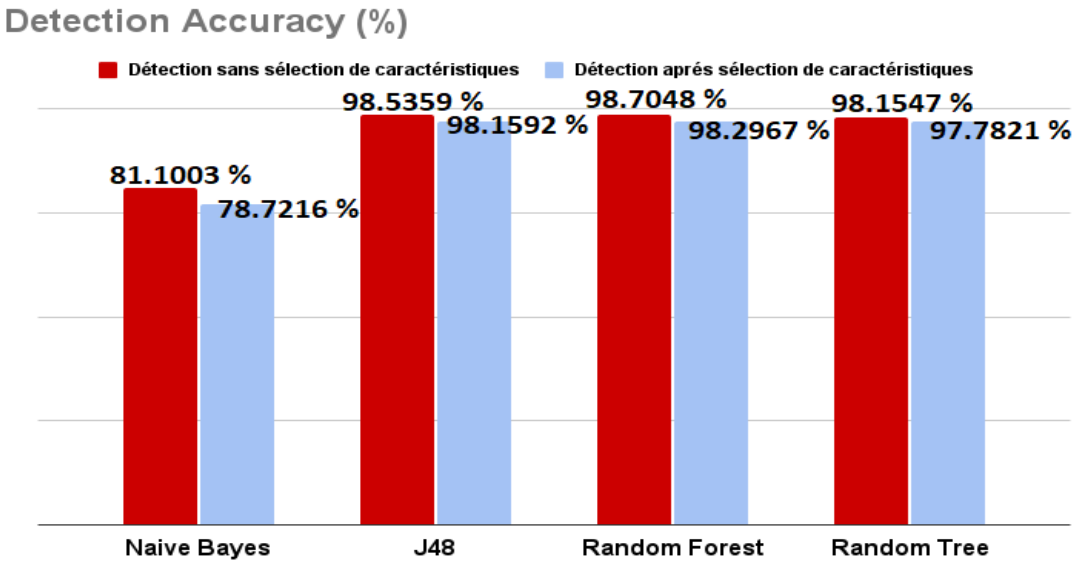

Figure 19: Résultats de l'évaluation GeneticSearch+CFSSubsetEval

2.3 InfoGainAttributeEval and Ranker (10 Attributes)

Classifier	Detection Accuracy (%)	Incorrectly Classified Instances	Total Number of Instances	Root mean squared error
Naive Bayes	80.1943 %	19.8057 %	22544	85.4644 %
J48	98.5185 %	1.4815 %	22544	21.913 %
Random Forest	98.6959 %	1.3041 %	22544	19.3308 %
Random Tree	98.2479 %	1.7521 %	22544	24.9457 %

Tableau 16: Résultats de l'évaluation Ranker+InfoGainAttributeEval

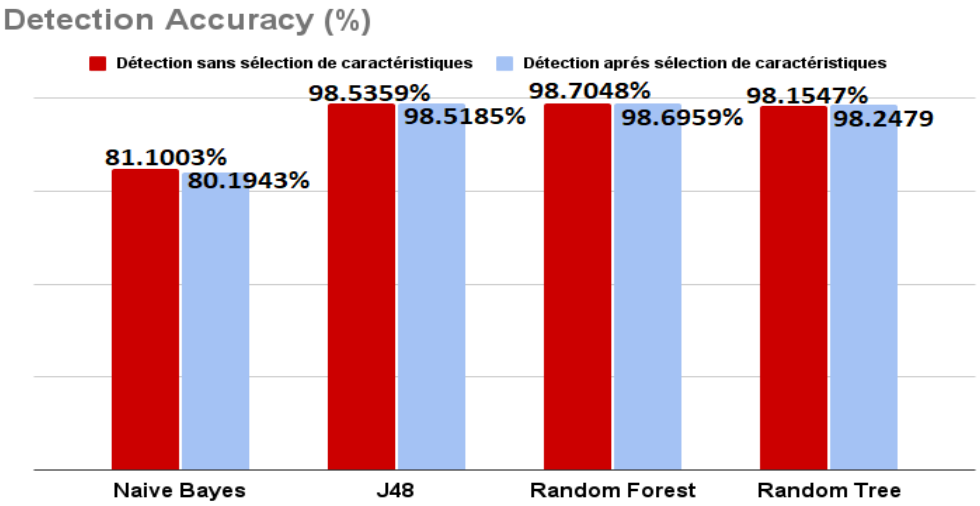

Figure 20: Résultats de l'évaluation Ranker+InfoGainAttributeEval

2.4 GainRatioAttributeEval and Ranker (10 Attributes)

Classifier	Detection Accuracy (%)	Incorrectly Classified Instances	Total Number of Instances	Root mean squared error
Naive Bayes	77.3643 %	22.6357 %	22544	95.614 %
J48	97.7644 %	2.2356 %	22544	27.1311 %
Random Forest	97.9418 %	2.0582 %	22544	24.4704 %
Random Tree	97.72 %	2.28 %	22544	28.0794 %

Tableau 17: Résultats de l'évaluation Ranker+GainRatioAttributeEval

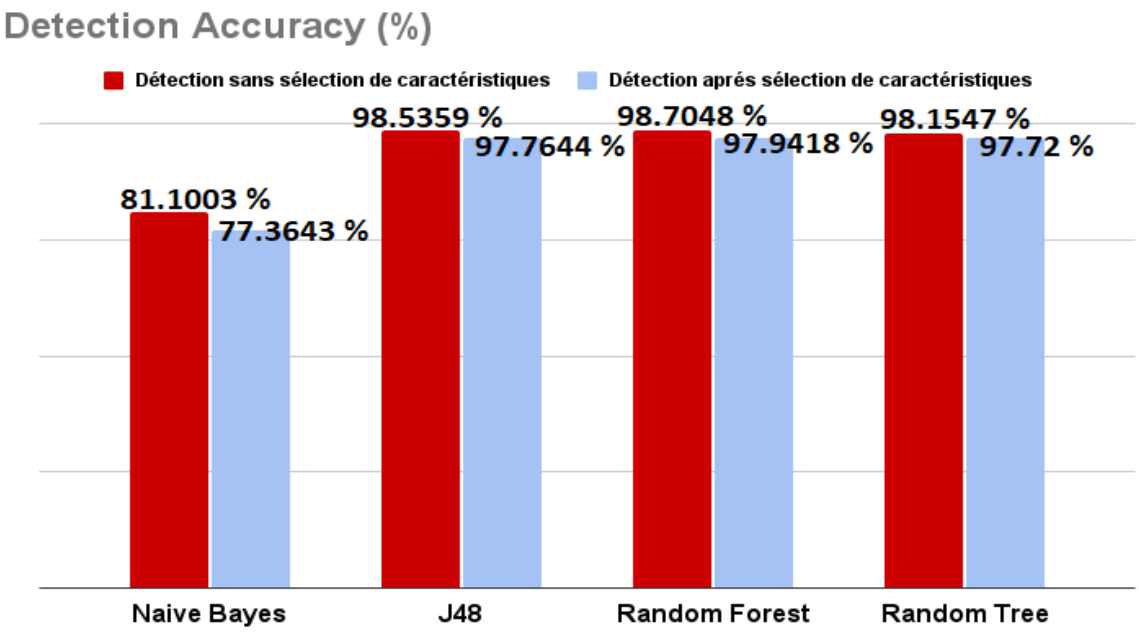

Figure 21: Résultats de l'évaluation Ranker+GainRatioAttributeEval

2.5 Chi-SquareAttributeEval and Ranker (10 Attributes)

Classifier	Detection Accuracy (%)	Incorrectly Classified Instances	Total Number of Instances	Root mean squared error
Naive Bayes	80.0302 %	19.9698 %	22544	82.9353 %
J48	98.4652 %	1.5348 %	22544	22.1106 %
Random Forest	98.6914 %	1.3086 %	22544	19.3374 %
Random Tree	98.3676 %	1.6324 %	22544	23.8862 %

Tableau 18: Résultats de l'évaluation Ranker+ChiSquaredAttributeEval

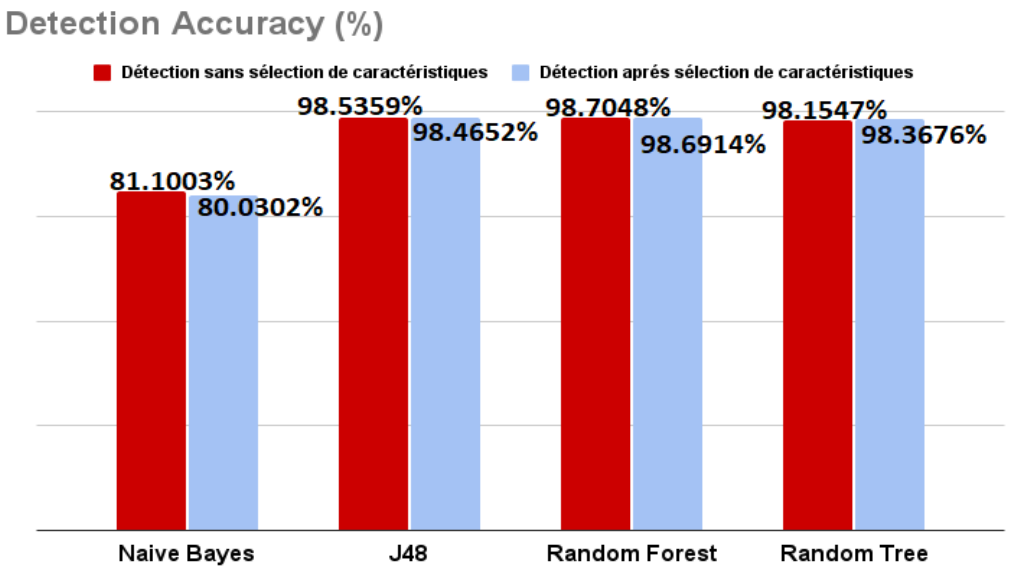

Figure 22: Résultats de l'évaluation Ranker+Chi-SquareAttributeEval

2.6 CorrelationAttributesEval and Ranker (10 Attributes)

Classifier	Detection Accuracy (%)	Incorrectly Classified Instances	Total Number of Instances	Root mean squared error
Naive Bayes	78.278 %	21.722 %	22544	93.0029 %
J48	91.5055 %	8.4945 %	22544	47.6108 %
Random Forest	91.9624 %	8.0376 %	22544	45.7292 %
Random Tree	91.2394 %	8.7606 %	22544	51.6991 %

Tableau 19: Résultats de l'évaluation Ranker+CorrelationAttributesEval

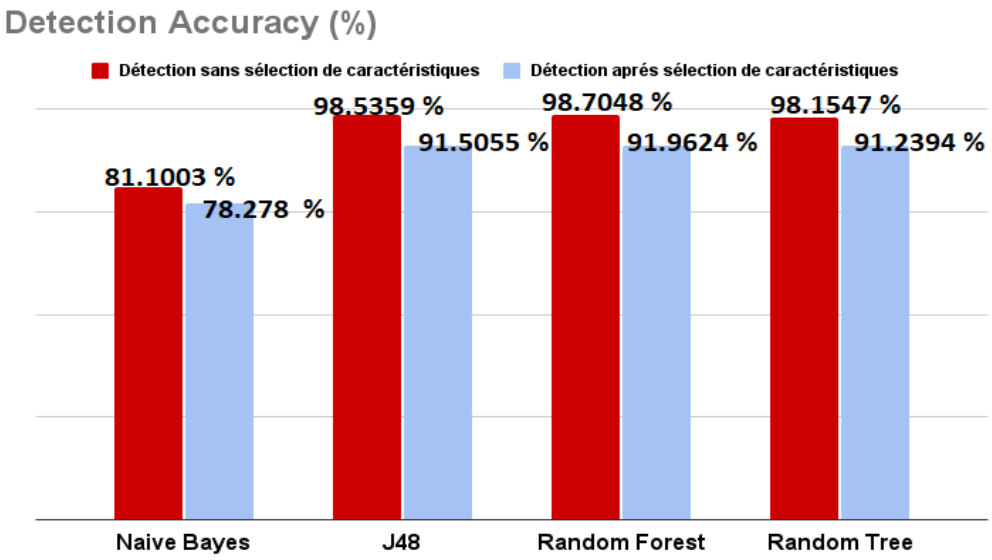

Figure 23: Résultats de l'évaluation CorrelationAttributesEval

Les Figure 16 17 18 19 20 21 représente le taux de détection de la précision. L'analyse comparative est présentée avec et sans sélection des caractéristiques.

3.10 Aggregation

L'agrégation de rangs est le processus qui combine les résultats de classement d'un ensemble fixe de candidats à partir de plusieurs fonctions de classement afin de générer un seul meilleur classement. [40]

3.10.1 Union

On applique l'opération union sur les caractéristiques sélectionnées dans le tableau 12

Ensemble de caractéristiques sélectionnées	Caractéristiques sélectionnées	union
9 21 9 10 10 10 10	9={5,f6,f12,f25,f28,f30,f31,f37,f41} 21={f1,f4,f5,f6,f11,f12,f17,f18,f22,f25,f26, f27,f28,f29,f30,f31,f32,f35,f36,f37,f41} 9={5,f6,f12,f25,f28,f30,f31,f37,f41} 10={f5,f6,f3,f4,f33,f35,f34,f40,f41,f23} 10={f28,f12,f41,f27,f4,f6,f5,f30,f29,f40} 10={f5,f6,f3,f33,f35,f34,f4,f40,f23,f12} 10={f33,f34,f12,f29,f4,f40,f27,f28,f41,f31}	f1,f3,f4,f5,f6,f11,f12,f17,f18, f22,f23,f25,f26,f27,f28,f29,f30, f31,f32,f33,f34,f35,f36,f37,f40, f41

Tableau 20: Liste des caractéristiques sélectionnées après l'opération union

3.10.2 Résultats globaux de la précision de détection

Table.20 présente les résultats obtenus après avoir sélectionné les caractéristiques suivantes: (f1,f3,f4,f5,f6,f11,f12,f17,f18,f22,f23,f25,f26,f27,f28,f29,f30,f31,f32,f33,f34,f35,f36,f37,f40,f41) sur l'ensemble de données NSL-KDD. Les caractéristiques sont maintenant réduites de 41 à 33.

Classifier	Detection Accuracy (%)	Incorrectly Classified Instances	Total Number of Instances	Root mean squared error
Naive Bayes	79.591 %	20.409 %	22544	90.6653 %
J48	98.5229 %	1.4771 %	22544	22.2621 %
Random Forest	98.6648 %	1.3352 %	22544	19.8919 %
Random Tree	98.2967 %	1.7033 %	22544	26.0317 %

Tableau 21: Résultats de la classification après l'opération union

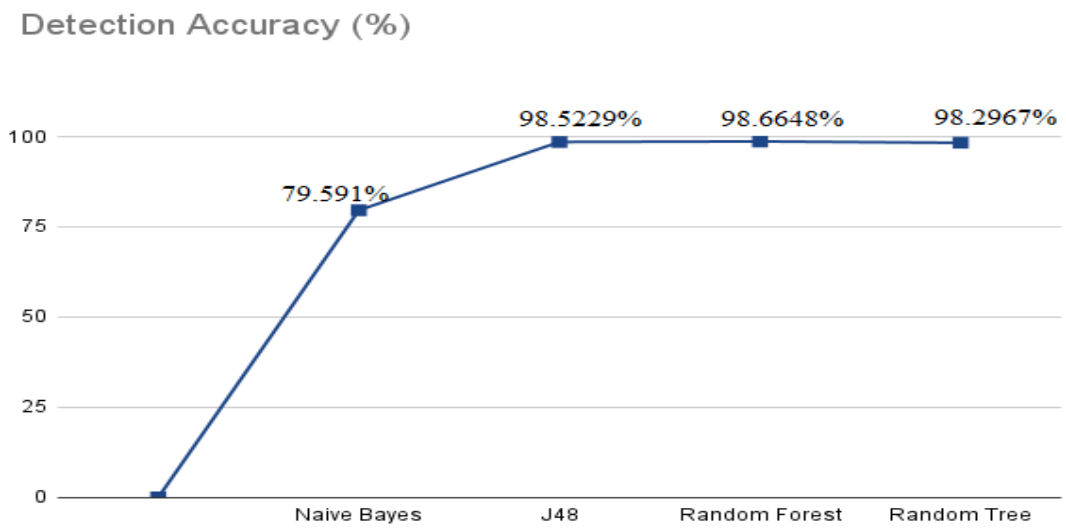

Figure 24: Graphique de comparaison de la précision de détection

Les résultats de cette analyse montrent que notre algorithme de sélection des caractéristiques basé sur l'agrégation des rangs est une technique efficace adaptée à divers types d'ensembles de données, y compris ceux dont les caractéristiques sont supérieures à 1000. Notre méthode donne une meilleure précision de classification, une meilleure mesure et une plus grande robustesse que les autres méthodes traditionnelles sur une large gamme de classificateurs. Cette méthode est particulièrement avantageuse dans les cas où il est difficile de déterminer la meilleure propriété statistique pour l'évaluation d'un ensemble de données donné. Le plus grand avantage d'une technique robuste est qu'il y aura moins de dilemmes pour décider du classificateur le plus approprié à utiliser parmi la vaste gamme de choix. [40]

II. Méthode Wrapper

1.1 Firefly

L'algorithme firefly (FA) est un algorithme évolutionnaire, inspiré du comportement des lucioles. L'algorithme FA est basé sur l'attractivité, qui est utilisée pour aller vers une plus grande intensité lumineuse entre les lucioles. L'AF prend en charge deux caractéristiques importantes pour résoudre les problèmes d'optimisation, qui sont :
la variation de l'intensité lumineuse et la formulation de l'attractivité. [27]

Iteration	Accuray	Subset Selected
1	89.9433	6
2	93.6714	4
3	93.8618	1
4	94.8826	2
5	94.8826	2
6	94.8826	2
7	94.8826	2
8	94.8826	2
9	94.8826	2
10	94.8826	2

Tableau 22: Résultats de Firefly

1.2 l'architecture de firefly

FAFS est un algorithme permettant de sélectionner le meilleur sous-ensemble de fonctionnalités qui a les meilleures performances de classification. FAFS est basé sur un wrapper approche qui utilise deux fonctions objectives : Taux de précision (AR) et taux de réduction (RR) (nombre de fonctionnalités). Le premier pas de FAFS commence par la création de la population initiale. Le la population initiale est composée sur la création des firefly, qui se compose de sous-ensembles de fonctionnalités. La prochaine étape est la position d'évaluation où nous évaluons la position de chacun firefly. La position représente le sous-ensemble de fonctionnalités. Plus loin, l'évaluation est basée sur deux objectifs, qui sont AR et RR . Dans cette évaluation, nous utilisons différents classificateurs tels que LDA, KNN et NB. Après cela, nous comparons les différents firefly pour choisir la 10 meilleure Firefly.

3.11 Conclusion

Dans ce document, nous avons souligné l'importance cruciale des systèmes de détection d'intrusion (IDS) pour sécuriser les réseaux contre les cyberattaques. En utilisant l'ensemble de données NSL-KDD, nous avons constaté que la combinaison du réseau bayésien avec les arbres aléatoires et les forêts aléatoires améliore significativement la précision de la détection des intrusions, malgré les variations de performance selon les types d'attaques. Cette approche hybride offre une solution robuste et adaptable pour renforcer la sécurité des réseaux face à diverses menaces, soulignant ainsi l'importance stratégique d'une approche proactive en sécurité informatique.

Conclusion générale

Conclusion Général

En conclusion, ce mémoire a exploré en profondeur les principes fondamentaux de la sélection d'ensemble de caractéristiques pour les systèmes de détection d'intrusion (IDS) dans le contexte de la sécurité informatique. Nous avons examiné les défis croissants posés par les cyberattaques et la nécessité d'améliorer l'efficacité des IDS pour protéger les réseaux informatiques contre ces menaces.

Nous avons souligné l'importance de la sélection d'ensemble de caractéristiques comme une étape cruciale pour maximiser la capacité de discrimination entre les activités normales et anormales, tout en minimisant les fausses alertes. En explorant différentes méthodes de sélection d'ensemble de caractéristiques et en évaluant leur efficacité, nous avons mis en lumière les avantages et les limites de chaque approche.

De plus, nous avons examiné en détail les systèmes de détection d'intrusion (IDS), leur modèle de base, leur classification, ainsi que les méthodes de détection disponibles et les mesures d'évaluation des performances. Cette compréhension approfondie nous a permis de mieux concevoir nos propres systèmes de détection d'intrusion, en utilisant des outils tels que Weka et des ensembles de données comme NSL KDD.

Enfin, en mettant en pratique nos connaissances théoriques, nous avons pu construire des systèmes de détection d'intrusion solides et robustes pour protéger nos réseaux contre les cyberattaques. Ces résultats ont des implications significatives pour la sécurité informatique, offrant des solutions pratiques pour faire face aux défis actuels et futurs de la cybersécurité.

En conclusion, ce mémoire contribue à enrichir la compréhension et les pratiques dans le domaine de la sécurité informatique en fournissant des connaissances approfondies sur la sélection d'ensemble de caractéristiques pour les systèmes de détection d'intrusion. Nous espérons que ce travail servira de base pour de futures recherches et développements visant à renforcer la sécurité des réseaux informatiques contre les menaces cybernétiques.

Bibliographie

BIBLIOGRAPHIE:

[1] : Maza1 S ; Touahria1 M (2019), « Feature selection for intrusion detection using new multi-objective estimation of distribution algorithms », Springer Nature, pdf, p.4238-4257.

[2] :https://datafranca.org/wiki/S%C3%A9lection_de_caract%C3%A9ristiques .

[3] : NEGGAZ I; "Sélection d'attributs par les techniques d'optimisation appliquée à l'analyse de visage humain", THèse de doctorat , Université des Sciences et Technologie Mohamed Boudiaf d'Oran, (2021/2022).

[4] : MENGHOUR K, "Approches Bio-inspirées pour la Sélection d'Attributs", Thèse de doctorat, UNIVERSITÉ BADJI MOKHTAR-ANNABA,(2014-2015).

[5] : BENHAMMADA S,"Etude comparative de méthodes de sélection de caractéristiques en apprentissage automatique.Proposition d'une variante", Mémoire de Master, Université Mentouri de Constantine,(2006-2007).

[6] : Touatia S ; Messaoudene S ,"Efficient Algorithm To Improve Feature Selection Accuracy "; Mémoire de master, UNIVERSITÉ ABDELHAMID IBN BADIS – MOSTAGANEM, (2021/2022).

[8] : SOUIER I et YOUBI Fatiha, «SÉLECTION DES VARIABLES A BASE DES MÉTAHEURISTIQUES», Thèse de Master Université de Tlemcen,(2015-2016).

[7] : Walid H Abdelkader A, "Evaluation des méthodes de Sélection de Variables en Apprentissage supervisé", MÉMOIRE de Master, UNIVERSITÉ ABDELHAMID IBN BADIS MOSTAGANEM , (2013-2014).

[9] : HIDAYET DJ , "Traitement parallèle pour la sélection et la classification des puces à ADN", Mémoire de master, Université Mohamed Khider – BISKRA, (2020).

[10] : Nicole CH,"Contributions à la sélection des attributs de signaux non stationnaires pour la classification", thèse de doctorat,UNIVERSITÉ DE TECHNOLOGIE DE TROYES,(2018).

[11] : Hassan CH, "Sélection de caractéristiques: méthodes et applications",thèse de doctorat,Université Paris Descartes,(2011).

[12] : V. Bolón-Canedo ; A. Alonso-Betanzos (2019), « Ensembles for feature selection: A review and future trends », information Fusion 52, p.1-12.

[13] : Yvan Saeys, Thomas Abeel, and Yves Van de Peer;Robust Feature Selection Using Ensemble Feature Selection Techniques";University, Gent, Belgium;pp. 313–325, (2008).

[14] : Utkarsh Mahadeo Khaire a ; R. Dhanalakshmi ;"Stability of feature selection algorithm: A review" ; Journal of King Saud University – Computer and Information Sciences; (2019).

[15] : Bilal Maqbool Beigh et M.A Peer, « Intrusion Detection and Prevention System: Classification and Quick Review », *ARPN Journal of Science and Technology*, 2012, p. 661-675 .

[16] : Poulmanogo illy;"Les systèmes de détection d'intrusion (IDS)";école de Technologie Supérieure ,(2018).

[17] : Zahaf M ," Système de Détection d'Intrusion Réseau Basé sur l'Apprentissage Automatique", UNIVERSITE ABDELHAMID IBN BADIS - MOSTAGANEM, (2022-2023).

[18] : BREK B ,"Système de Détection d'Intrusion basée sur Les Systèmes Multi-Agents" ,Mémoire de master ,Université Larbi Tébessi - Tébessa ,(2019).

[19] : Nathalie Dagorn , « Détection et prévention d'intrusion : présentation et limites », (Juillet 2006).

[20] : BENAICHA S, "Système de détection d'intrusion réseau basé sur les Algorithmes Génétiques",MÉMOIRE de MASTER ;UNIVERSITÉ DE M'SILA ,(2012 12013).

[21] : BELKHATMI K ; BENAMARA Oua , "Mise en place d'un système de détection et de prévention d'intrusion", Mémoire de master , Université A/Mira de Béjaïa,(2015/2016).

[22] : ahmin A , "Système de détection d'intrusion adaptatif et distribué" , thèse de doctorat , UNIVERSITÉ BADJI MOKHTAR-ANNABA, (2014) .

[23] : Hamouda Dj , "Un système de détection d'intrusion pour la cybersécurité" , Mémoire de Master, Université de 8 Mai 1945 – Guelma (2020).

[25]:BOUROUH Mouloud - KANOUN Zakaria ,Détection d'intrusions à base des réseaux de neurones et algorithmes génétiques Mémoire de Master Université Abou Bakr Belkaid– Tlemcen Faculté des Sciences Département d'Informatique 2016-2017

[26]: Prof . Hakim Lounis ,Environnement Weka,Université de Québec À Montréal Département d'Informatique (26/01/2006)

[27]: Sofiane MAZA and Mohamed TOUAHRIA,Feature Selection Algorithms in Intrusion Detection System: A Survey ,,University of Ferhat Abbas Sétif-1, Setif 19000; Algeria,Department of Computer Science(October 31, 2018)
[28]: Osmani Youcef,Un system de détection d'intrusion à base d'apprentissage supervisé,Mémoire de Master Faculté des mathématiques et de l'informatique Département d'Informatique (2014 / 2015)
[29]: Sapna S. Kaushik, Dr. Prof.P.R.Deshmukh," Detection of Attacks in an Intrusion Detection System", International Journal of Computer Science and Information Technologies, Vol. 2 (3), 2011, 982-986
[30]: Amit Arora.Karan Bajaj.Improving the Intrusion Detection using Discriminative?Machine Learning Approach and Improve the Time Complexity by Data Mining Feature Selection Methods Chitkara university Himachal Pradesh 2013
[31]:CHIKH née Kadri Djamila .BENACHENHOU Sidi Mohammed.BOUHADJER Nawel.Hiérarchisation intelligente des éléments de services contribuant à la satisfaction globale. université Abou Bakr Belkaid.2017
[32]: Megha Aggarwal, Amrita;Performance Analysis Of Different Feature Selection Methods In Intrusion Detection ; Department of CSE, SHARDA UNIVERSITY, Greater Noida, India ;INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH VOLUME 2 ;2013
[33]:BOUBLENZA Amina ;Coopération entre classifieurs hétérogènes pour la reconnaissance des données médicales;THÈSE DE DOCTORAT ;UNIVERSITÉ ABOU-BEKR BELKAID – TLEMCEN;22 février 2017
[34]: Tuba Parlar,Gokalp Cinarer;Feature selection methods for intrusion detection using machine learning methods Mustafa Kemal University;Bozok University, 2022
[35]: Mohanad Albayati.Biju Issac.Analysis of Intelligent Classifiers and Enhancing the Detection Accuracy for Intrusion Detection System.School of Computing, Teesside University, Middlesbrough, England, UK .2015
[36]: Gaurav Meena.Ravi Raj Choudhary;A Review Paper on IDS Classification using KDD 99 and NSL KDD Dataset in WEKA .University of Rajasthan, Ajmer, India.2017
[37]:https://blog.netwrix.fr/2018/07/04/les-10-types-de-cyberattaques-les-plus-courants/
[38]: https://dataportal.asia/dataset/212582112_kdd-cup-1998-data
[39]: Mohanad Albayati .Biju Issac International Journal of Computational Intelligence Systems,Analysis of Intelligent Classifiers and Enhancing the Detection Accuracy for Intrusion Detection System ,School of Computing, Teesside University, Vol. 8, No. 5 (2015) 841-853.
[40]: Chandrima Sarkar,Sarah Cooley,Jaideep Srivastava ,Robust Feature Selection Technique using Rank Aggregation,University of Minnesota at Twin Cities,2014