CHAPITRE 1

SÉCURITÉ DES RÉSEAUX INFORMATIQUE ET CAH

1. Généralités sur la sécurité informatique

Les exigences de la sécurité de l'information au sein des organisations ont conduit à deux changements majeurs au cours des dernières décennies. Avant l'usage généralisé d'équipements informatiques, la sécurité de l'information était assurée par des moyens physiques ou administratifs. Avec l'introduction de l'ordinateur, le besoin d'outils automatisés pour protéger fichiers et autres informations stockées est devenu évident. On donne à cette collection d'outils conçus pour protéger les données et contrecarrer les pirates le nom de sécurité informatique.

1.1. La sécurité informatique :

La sécurité informatique est l'ensemble des moyens matériels et logiciels mis en œuvre pour minimiser la vulnérabilité d'un système contre des menaces accidentelles ou intentionnelles, permettant ainsi aux systèmes informatiques de fonctionner normalement. Elle consiste, aussi, à s'assurer que celui qui modifie ou consulte les données du système en a l'autorisation et qu'il peut le faire car le service est disponible. [5]

1.2. Principaux services de la sécurité :

Pour remédier aux failles et pour contrer les attaques, la sécurité informatique se base sur un certain nombre de services qui permettent de mettre en place une réponse appropriée à chaque menace. Les principaux services sont : **[6]**

- **La confidentialité** : La confidentialité consiste à rendre l'information inintelligible à d'autres personnes que les seuls acteurs de la transaction.
- **L'intégrité des données** : Vérifier l'intégrité des données consiste à déterminer si les données n'ont pas été altérées durant la communication (de manière fortuite ou intentionnelle).
- **La disponibilité** : Permettant de maintenir le bon fonctionnement du système informatique.
- **Le non répudiation** : Permettant de garantir qu'une transaction ne peut être niée.
- **L'authentification** : L'authentification consiste à assurer l'identité d'un utilisateur, c'est-à dire de garantir à chacun des correspondants que son partenaire est bien celui qu'il croit être.

1.3. Différents types d'attaques :

Tout ordinateur connecté à un réseau informatique est potentiellement vulnérable à une attaque.

1.3.1. Les attaques réseaux :

Les attaques réseaux les plus connues aujourd'hui sont :

- **Spoofing IP :** [6]

Le spoofing IP est une technique permettant à un pirate d'envoyer à une machine des paquets semblant provenir d'une adresse IP autre que celle de la machine du pirate. Le spoofing IP n'est pas pour autant un changement d'adresse IP. Plus exactement, il s'agit d'une mascarade de l'adresse IP au niveau des paquets émis, c'est-à-dire une modification des paquets envoyés afin de faire croire au destinataire qu'ils proviennent d'une autre machine.

- **Spoofing ARP : [6]**

Le spoofing ARP est une technique qui modifie le cache ARP. Le cache ARP contient une association entre les adresses matérielles des machines et les adresses IP, l'objectif du pirate est de conserver son adresse matérielle, mais d'utiliser l'adresse IP d'un hôte approuvé. Ces informations sont simultanément envoyées vers la cible et vers le cache. A partir de cet instant, les paquets de la cible seront routés vers l'adresse matérielle du pirate.

- **Spoofing DNS : [6]**

L'objectif de cette attaque est de fournir de fausses réponses aux requêtes DNS, c'est-à-dire indiquer une fausse adresse IP pour un nom de domaine.

Il existe deux techniques pour effectuer cette attaque :

- *DNS ID spoofing* **:** L'attaquant essaie de répondre à un client en attente d'une réponse d'un serveur DNS, avec une fausse réponse et avant que le serveur DNS ne réponde.
- *DNS Cache Poisoning* **:** L'attaquant essaie d'empoisonner le cache (table de correspondance IP- nom _machine) du serveur DNS.

- **Vol de session TCP (TCP Session Hijacking) : [10]**

Le but de cette attaque est de rediriger un flux TCP (Transmission Control Protocol) afin de pouvoir outrepasser une protection par mot de passe.

Le contrôle d'authentification s'effectuant uniquement à l'ouverture de la session, un pirate réussissant cette attaque parvient à prendre possession de la connexion pendant toute la durée

de la session. Dans un premier temps, le pirate doit écouter le réseau, puis, lorsqu'il estime que l'authentification a pu se produire (délai de n secondes par exemple), il désynchronise la session entre l'utilisateur et le serveur. Pour ce faire, il construit un paquet avec, comme adresse IP source, celle de la machine de l'utilisateur et le numéro d'acquittement TCP attendu par le serveur. En plus de désynchroniser la connexion TCP, ce paquet permet au pirate d'injecter une commande via la session préalablement établie.

1.3.2. Les attaques applicatives :

Les attaques applicatives se basent sur des failles dans les programmes utilisés, ou encore des erreurs de configuration. Toutefois, comme précédemment, il est possible de classifier ces attaques selon leur provenance.

- **Les bogues (bugs) : [6]**

Tout logiciel comporte des bogues dont certains représentent des trous de sécurité ou des anomalies qui permettent de violer le système sur lequel tourne le programme. Si c'est un programme d'application réseau, ces trous peuvent être exploités à distance via Internet. Les plus connus de ces bogues et les plus intéressants en ce qui concerne leur exploitation, sont les buffers overflows.

- *buffers overflows :* [10]

Issus d'une erreur de programmation, ils permettent l'exploitation d'un shellcode[1] à distance.

Ce shellcode permettra à une personne mal intentionnée d'exécuter des commandes sur le système distant, pouvant aller jusqu'à sa destruction.

L'erreur de programmation est souvent la même : la taille d'une entrée n'est pas vérifiée et l'entrée est directement copiée dans un buffer dont la taille est inférieure. On se retrouve donc en situation de débordement, et l'exploitant peut ainsi accéder à la mémoire.

- **Injection SQL (SQL injection) : [10]**

Tout comme les attaques de scripts, les injections SQL profitent de paramètres d'entrée non vérifiés. Comme leur nom l'indique, le but des injections SQL est d'injecter du code SQL dans une requête de base de données. Ainsi, il est possible de récupérer des informations se trouvant dans la base (exemple : des mots de passe) ou encore de détruire des données.

[1] Un shellcode est une chaîne de caractères qui représente un code binaire exécutable capable de lancer un shell('/bin/sh' sous Unix ou command.com sous DOS et Microsoft Windows par exemple). Un shellcode peut être utilisé par un cracker voulant avoir accès à la ligne de commande.

- **Homme au milieu (Man in the middle) : [10]**

Moins connue, mais tout aussi efficace, cette attaque permet de détourner le trafic entre deux stations. Imaginons un client C communiquant avec un serveur S. Un pirate peut détourner le trafic du client en faisant passer les requêtes de C vers S par sa machine P, puis transmettre les requêtes de P vers S. Et inversement pour les réponses de S vers C.

Totalement transparente pour le client, la machine P joue le rôle de proxy. Elle accédera ainsi à toutes les communications et pourra en obtenir les informations sans que l'utilisateur s'en rende compte.

1.3.3. Le Déni de service (Denial of Service): [10]

Le déni de service est une attaque visant à rendre indisponible un service. Ceci peut s'effectuer de plusieurs manières : par le biais d'une surcharge réseau, rendant ainsi la machine totalement injoignable, ou bien de manière applicative en crashant l'application à distance.

Voici quelques attaques réseau connues permettant de rendre indisponible un service :

- **SYN Flooding** : exploite la connexion en trois phases de TCP (Three Way Handshake : SYN/SYN-ACK/ACK). Le principe est de laisser un grand nombre de connexions TCP en attente. Le pirate envoie de nombreuses demandes de connexion (SYN), reçoit les SYN-ACK, mais ne répond jamais avec ACK. Les connexions en cours occupent des ressources mémoire, ce qui va entraîner une saturation et l'effondrement du système.
- **UDP Flooding** : le trafic UDP (User Datagram Protocol) est prioritaire sur TCP. Le but est donc d'envoyer un grand nombre de paquets UDP, ce qui va occuper toute la bande passante et ainsi rendre indisponibles toutes les connexions TCP.
- **Fragmentation [6] :** L'attaquant sature la connexion en envoyant des fragmentations déclenchant des exceptions (faille de la pile TCP/IP de *Windows* 95 et 98).
- **Smurfling** : le pirate fait des requêtes ICMP (Internet Control Message Protocol) ECHO à des adresses de broadcast en spoofant l'adresse source (en indiquant l'adresse de la machine cible). Cette machine cible va recevoir un nombre énorme de réponses, car toutes les machines vont lui répondre, et ainsi utiliser toute sa bande passante.

1.3.4. Les attaques virales : [6]

Il existe principalement quatre types de menaces distinctes :

- **Virus** : Se reproduisent en infectant le corps de programmes hôtes.

- **Vers** : Le vers se duplique et se propage à travers le réseau, par courrier électronique par exemple.
- **Chevaux de Troie (Trojan)** : Exécutent des tâches malignes en se cachant dans un programme sain. Il peut par exemple voler des mots de passe, copier des données, ou exécuter toute autre action nuisible.
- **Trappes (portes dérobées)** : Permet à un utilisateur externe de prendre le contrôle d'une application par des moyens détournés.

1.4. Mécanismes de sécurité :

Nous tenons d'abord à définir un mécanisme de sécurité, comme étant un ensemble de stratégies, conçues et mises en place pour détecter, prévenir et lutter contre une attaque. Avec l'évolution de la technologie, il existe une panoplie de mécanismes de sécurité [7]. Nous détaillons dans la suite les plus connus de ces mécanismes.

1.4.1. Authentification et contrôles d'accès aux ressources :

Il s'agit en premier lieu de la sécurité physique des équipements et installations. L'authentification est un mécanisme permettant de prouver l'identité d'un utilisateur (mots de passe, cartes à puce, méthodes biométriques, etc.) et de lui accorder uniquement les privilèges nécessaires pour l'accomplissement de ses tâches. [5]

1.4.2. La Cryptographie : [8]

La cryptographie est une science mathématique dans laquelle on fait les études des méthodes permettant de transmettre des données de manière confidentielle. Afin de protéger un message, on lui applique une transformation qui le rend incompréhensible; c'est ce qu'on appelle le chiffrement, qui, à partir d'un texte clair, donne un texte chiffré ou cryptogramme. Inversement, le déchiffrement est l'action qui permet de reconstruire le texte en clair à partir du texte chiffré.

Dans la cryptographie moderne, les transformations en question sont des fonctions mathématiques, appelées algorithmes cryptographiques, qui dépendent d'un paramètre appelé clef. Dans les réseaux, pour contrer les vols d'informations dans la voie de transmission, on utilise les techniques de cryptographie pour chiffrer et déchiffrer les messages transmis. Il existe à l'heure actuelle deux grands principes de cryptage : le cryptage symétrique basé sur l'utilisation d'une clef privée et le cryptage asymétrique qui repose sur un codage à deux clefs, une privée et l'autre publique.

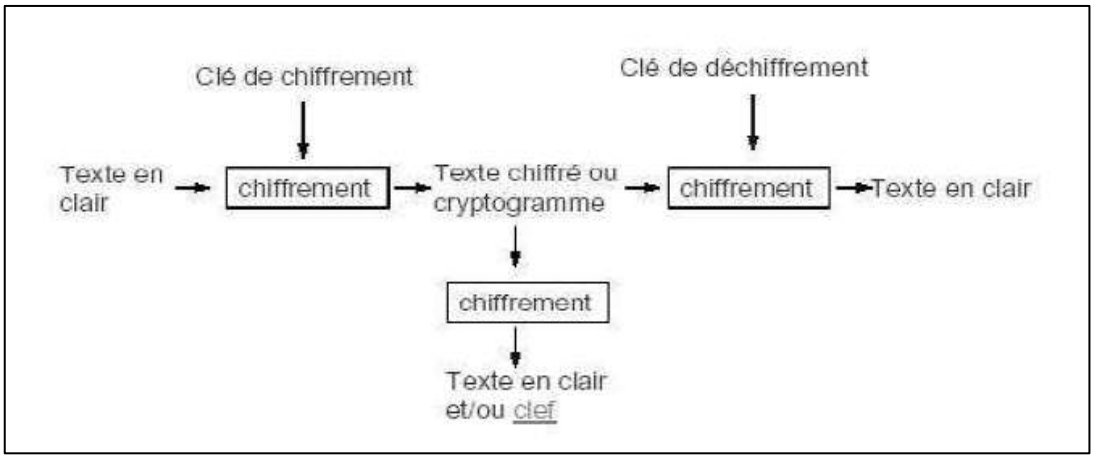

Figure 1.1: Chiffrement [8]

1.4.3. Pare-feu (fire-well) :

Un pare-feu (firewall) est une solution matérielle ou logicielle mise en place au sein de l'infrastructure du réseau afin de filtrer l'accès à des ressources réseau définies. Il ne laisse entrer que les utilisateurs autorisés, disposant d'une clef ou d'un badge, et crée une couche protectrice entre le réseau et le monde extérieur. Il est doté de filtres intégrés qui peuvent empêcher des documents non autorisés ou potentiellement dangereux d'accéder au système. Il enregistre également les tentatives d'intrusions dans un journal transmis aux administrateurs du réseau. Il permet également de contrôler l'accès aux applications et d'empêcher le détournement d'usage [8].

1.4.4. Scanners de vulnérabilités :

Les scanners de vulnérabilités automatisent la découverte des failles de sécurité. Ils sont utilisés par les attaquants pour localiser les faiblesses du réseau cible. De plus, les administrateurs peuvent en tirer profit pour corriger les vulnérabilités de leurs systèmes informatique. Cependant, malgré le grand nombre de vulnérabilités détectées, les scanners d'aujourd'hui sont inaptes à déterminer toutes les faiblesses possibles. De plus, la mise à jour de ces produits ne suit pas le rythme de la découverte des nouvelles vulnérabilités. [5]

1.4.5. Protection anti-virus :

Les logiciels anti-virus sont largement utilisés pour les stations de travail et ordinateurs personnels. Ils détectent et protègent contre les virus pouvant se propager à travers des fichiers, courrier électronique, etc. [5]

1.4.6. VPN : [9]

Le principe du VPN (Virtual Private Network) est basé sur la technique du tunnelling. Le tunneling est l'ensemble des processus d'encapsulation, de transmission et de désencapsulation. Cela consiste à construire un chemin virtuel après avoir identifié l'émetteur et le destinataire. Ensuite, la source chiffre les données et les achemine en empruntant ce chemin virtuel. Les données à transmettre peuvent appartenir à un protocole différent d'IP. Dans ce cas, le protocole de tunnelling encapsule les données en rajoutant un entête permettant le routage des trames dans le tunnel.

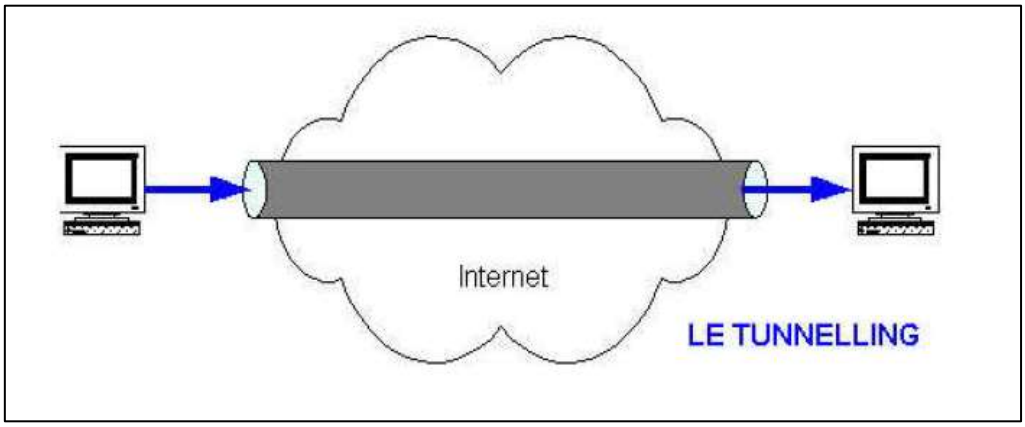

Figure 1.2: Principe de VPN [9]

1.4.7. Détection et prévention d'intrusions : [8]

Un système de détection d'intrusions (IDS) permet de surveiller constamment le réseau. Il analyse les flux de paquets de données transitant par le réseau, à la recherche d'activités non autorisées (telles que les attaques de pirates) et permet aux utilisateurs de traiter les failles de sécurité avant que les systèmes ne soient compromis.

Aujourd'hui, les systèmes IDS évoluent vers ce que l'on appelle des systèmes de prévention d'intrusions (IPS) qui, en complément de la détection, apporte une protection active. Un système IPS peut ainsi décider, suite à des remontées d'alertes, de fermer des ports et de rejeter des paquets en fonction du paramétrage qui en a été fait.

2. Système de détection d'intrusion

Afin de détecter les attaques que peut subir un système, il est nécessaire d'avoir un logiciel spécialisé dont le rôle serait de surveiller les données qui y transitent, et qui serait capable de réagir si des données semblent suspectes. Plus communément appelés IDS (Intrusion Detection Systems), les systèmes de détection d'intrusions conviennent parfaitement pour réaliser cette tâche.

2.1. Définition :

Un IDS **(***Intrusion Detection System***)** est un outil logiciel ou matériel qui permet d'écouter le trafic réseau de façon furtive dans le but de détecter des activités anormales qui pourraient être assimilées à des intrusions définies comme des tentatives pour compromettre la confidentialité, l'intégrité et la disponibilité d'une ressource ou éviter des mécanismes de sécurité de l'ordinateur ou du réseau. [8]

2.2. Terminologies :

Il convient avant d'aborder certains aspects de la détection d'intrusions de revenir sur certaines définitions qui prêtent parfois à confusion. Nous nous sommes référés pour cela aux définitions données par les RFC2828[2] [11].

- **Intrusion :** événement ou combinaison d'événements permettant d'avoir indûment accès (sans autorisation) à un système et ses ressources.
- **Détection d'intrusions :** analyse des événements ayant lieu dans un système dans le but de trouver en temps réel, en quasi temps réel ou en différé, des tentatives d'accès non autorisé et aussitôt de notifier ces tentatives à l'administrateur du système.
- **faux-positif :** détection en absence d'attaque, alarme générée par un IDS pour un évènement légal.
- **faux-négatif :** absence de détection en présence d'attaque, non génération d'alarme par un IDS pour un événement illégal.
- **Signature d'attaque :** Une signature d'attaque est un motif représentant toute l'information concernant une attaque connue. C'est par ce moyen que l'administrateur réseau configure les systèmes de détection d'intrusions. L'exemple suivant représente une signature du système de détection d'intrusion Snort :

  alert tcp $EXTERNAL_NET any − > $HOME_NET any (msg :"SCAN nmap TCP"; state-less; flags :A,12; ack :0; reference :arachnids,28; classtype :attempted-recon; sid :628; rev :3;)

  Qui se lit comme suit : si un paquet TCP provenant de l'extérieur ($EXTERNALNET) pénètre dans notre réseau ($HOME NET), peu importe les ports (any), et que ce paquet a le drapeau ACK activé, de même que les deux bits réservés

[2] Les RFC (Request For Comments) sont un ensemble de documents qui font référence auprès de la Communauté Internet et qui décrivent, spécifient, aident à l'implémentation, standardisent et débattent de la majorité des normes, standards, technologies et protocoles liés à Internet et aux réseaux en général.

(flags : A,12), et que le numéro d'acquittement est 0 (ack :0), peu importe l'état de la session (stateless), il faut alors signaler un balayage TCP fait avec nmap[3]. [13]

- **Alerte :** Une alerte représente l'information transmise par l'IDS à l'intention de l'administrateur. Elle doit être claire, nette et précise. [13]

2.3. Architecture d'un IDS : [5]

Nous décrivons dans cette section les trois composants qui constituent classiquement un système de détection d'intrusions.

- **Capteur :**

Le capteur observe l'activité du système par le biais d'une source de données et fournit à l'analyseur une séquence d'événements qui renseignent de l'évolution de l'état du système. Le capteur peut se contenter de transmettre directement ces données brutes, mais en général, un prétraitement est effectué.

On distingue classiquement trois types de capteurs en fonction des sources de données utilisées pour observer l'activité du système : les capteurs système, les capteurs réseau et les capteurs applicatifs.

- **Analyseur :**

L'objectif de l'analyseur est de déterminer si le flux d'événements fourni par le capteur contient des éléments caractéristiques d'une activité malveillante.

- **Manager :**

Le manager collecte les alertes produites par l'analyseur, les met en forme et les présente à l'opérateur. Éventuellement, le manager est chargé de la réaction à adopter qui peut être :

- ✓ Confinement de l'attaque, qui a pour but de limiter les effets de l'attaque.
- ✓ Eradication de l'attaque, qui tente d'arrêter l'attaque.
- ✓ Recouvrement, qui est l'étape de restauration du système dans un état sain.
- ✓ Diagnostic, qui est la phase d'identification du problème.

[3] Nmap est un scanner de ports libre. Il est conçu pour détecter les ports ouverts, identifier les services hébergés et obtenir des informations sur le système d'exploitation d'un ordinateur distant.

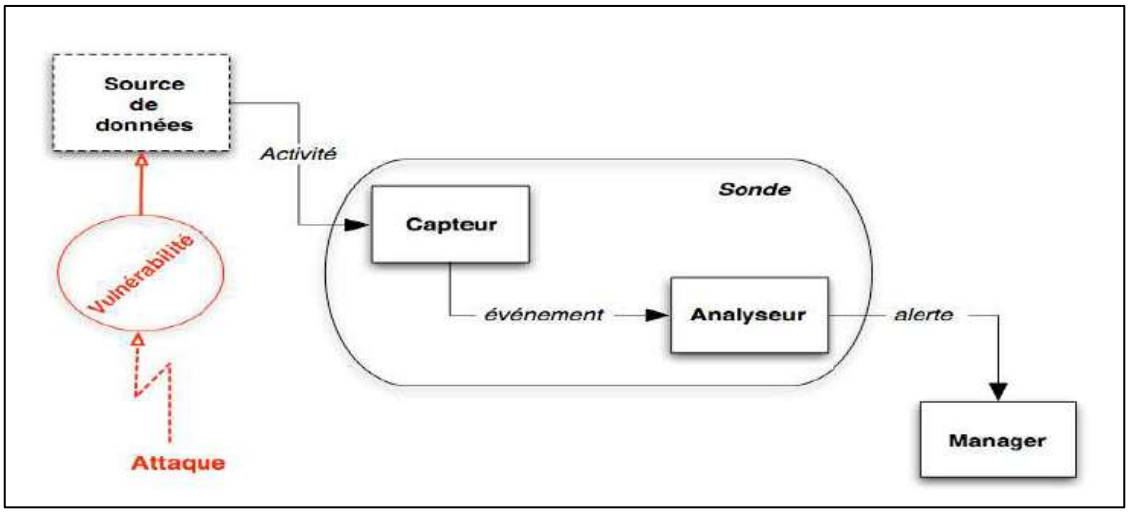

Figure 1.3: Architecture classique d'un IDS [5]

2.4. Mode de fonctionnement d'un IDS :

Il faut distinguer deux aspects dans le fonctionnement d'un IDS : le mode de détection utilisé et la réponse apportée par l'IDS lors de la détection d'une intrusion. Il existe deux modes de détection, la détection d'anomalies et la reconnaissance de signatures. De même, deux types de réponses existent, la réponse passive et la réponse active. [15]

2.4.1. Mode de détection :

Il existe deux modes de détection.

a) La détection d'anomalies :

Elle consiste à détecter des anomalies par rapport à un profil "de trafic habituel". La mise en œuvre comprend toujours une phase d'apprentissage au cours de laquelle les IDS vont "découvrir" Le fonctionnement "normal" des éléments surveillés. Ils sont ainsi en mesure de signaler les divergences par rapport au fonctionnement de référence. [15].

b) La reconnaissance de signature :

Cette approche consiste à rechercher dans l'activité de l'élément surveillé les empreintes (ou signatures) d'attaques connues. Ce type d'IDS est purement réactif ; il ne peut détecter que les attaques dont il possède la signature. De ce fait, il nécessite des mises à jour fréquentes. [15].

De plus, l'efficacité de ce système de détection dépend fortement de la précision de sa base de signature.

Une signature permet de définir les caractéristiques d'une attaque, au niveau des paquets ou au niveau protocole.

2.4.2. Réponse passive et active :

Il existe deux types de réponses, la réponse passive et la réponse active.

a) La réponse passive :

La réponse passive d'un IDS consiste à enregistrer les intrusions détectées dans un fichier de log qui sera analysé par le responsable de sécurité. Certain IDS permettent de logger l'ensemble d'une connexion identifiée comme malveillante. Ceci permet de remédier aux failles de sécurité pour empêcher les attaques enregistrées de se reproduire, mais n'empêche pas directement une attaque de se produire. [15].

b) La réponse active :

La réponse active au contraire, a pour but de stopper une attaque au moment de sa détection. [15].

2.5. Classification des systèmes de détection d'intrusions :

Nous pouvons classifier les systèmes de détections d'intrusions selon cinq critères (cités ci-dessous) qui ne sont pas forcément mutuellement exclusif :

a) Méthode de détection utilisée.

b) Sources de données à analyser.

c) Réponse aux attaques.

d) Mode de fonctionnement des mécanismes de détection

e) Fréquence d'utilisation.

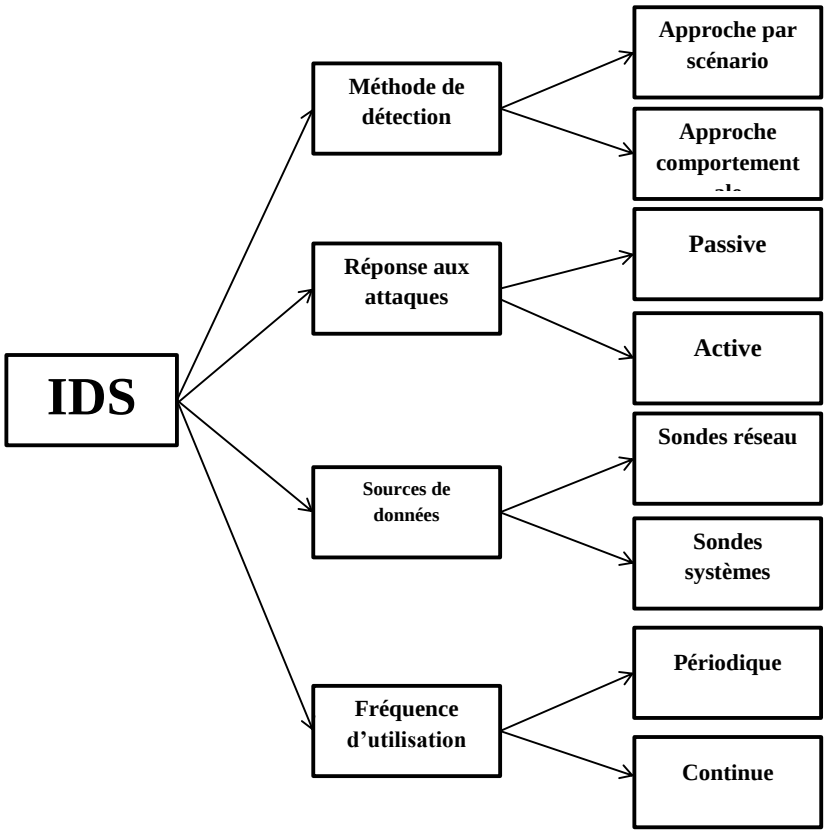

Figure 1.4: Classification des systèmes de détection d'intrusions.

2.5.1. Méthodes de détection :

On distingue deux grands types d'approches pour détecter les intrusions. La première consiste à rechercher des signatures connues d'attaques tandis que la seconde consiste à définir un comportement normal du système et à rechercher ce qui ne rentre pas dans ce comportement.

a) Approche par scénario : [5]

S'il n'est pas envisageable de décrire statistiquement le comportement d'un attaquant (on manque de données pour ce faire), il est possible de donner des règles sur sa manière de procéder. Ces règles prennent la forme de scénarios d'attaque exploitant des vulnérabilités du système précédemment identifiées. On parle parfois, pour cette approche dont la question de base est « le comportement de l'utilisateur correspond-il à une attaque connue ? », de modèle de scénario (signature d'attaque). Avec de tels systèmes, toute attaque non prévue reste indétectée.

Pour cette approche, les techniques de modélisation et d'implémentation les plus utilisées pour élaborer des IDS ayant connu un succès dans la pratique sont les systèmes experts, les algorithmes génétiques...etc

Cette approche possède les avantages suivants [6]

- censée être très puissante pour reconnaître une attaque sans générer trop de fausses alertes.
- capable de diagnostiquer rapidement l'utilisation d'une attaque ou outil d'attaque.
- ce type de détecteur reste assez facile à mettre en œuvre, ne nécessitant pas de phase d'apprentissage (ce qui élimine le risque de sur-apprentissage ou de déformation volontaire du profil).

Cependant, Ce type de détecteur est purement réactif, il ne peut détecter que les attaques dont il possède la signature, ce qui implique un taux plus élevé de faux négatifs. De ce fait, il nécessite des mises à jour quotidiennes. De plus, ce système de détection est aussi bon que la base de signatures.

b) Approche comportementale : [5]

Une approche consiste à utiliser des méthodes basées sur l'hypothèse selon laquelle l'exploitation d'une vulnérabilité du système implique un usage anormal de celui-ci. Une intrusion est donc identifiable en tant que déviation par rapport au comportement habituel d'un utilisateur. Bien sûr, une telle déviation peut avoir une autre cause qu'une attaque du système ; par exemple : un changement de fonction de l'utilisateur au sein de l'entreprise. On s'attachera donc à trouver des méthodes possédant le plus fort taux de discrimination possible (c'est-à-dire ayant le plus fort taux de détection d'intrusions et le plus faible taux de fausses alarmes). De plus, on se référera à un seuil au-delà duquel on considérera que le comportement est intrusif.

Cette approche, dont la question de base est « le comportement actuel de l'utilisateur est-il cohérent avec son comportement passé ? », est appelée approche comportementale. Pour caractériser le comportement normal d'un utilisateur (on parle de *profil*), des techniques employées pour modéliser le comportement se basent sur des méthodes statistiques. Il est également possible d'envisager l'utilisation de systèmes experts ou de réseaux de neurones.

Cette approche possède les avantages suivants : [6]

- la détection d'anomalies permet de détecter un comportement non usuel et offre ainsi la possibilité de trouver des symptômes d'une attaque sans en connaître les détails.

- peut permettre de produire de l'information qui peut être utilisée pour définir de nouvelles signatures utilisables pour les systèmes à signatures.
- pas besoin d'une base d'attaque.

Cependant, l'approche comportementale possède les inconvénients suivants : [6]

- produit une quantité énorme de fausses alertes à cause du caractère imprévisible des utilisateurs et des réseaux.
- demande une intense phase d'apprentissage pour caractériser la normalité des comportements.
- l'utilisateur peut changer lentement de comportement dans le but d'habituer le système à un comportement intrusif (faux négatifs).
- choix délicat des mesures à retenir pour un système cible donné.
- pour un utilisateur au comportement erratique, toute activité est normale.

2.5.2. Sources de données à analyser : [8]

Les sources possibles de données à analyser sont une caractéristique essentielle des systèmes de détection d'intrusions. Les données proviennent soit d'informations obtenues en écoutant le trafic réseau, soit d'un fichier généré par le système d'exploitation, soit encore d'un fichier généré par des applications.

a) Sondes réseau : [8]

Les sondes réseau (NIDS : *Network based IDS*) récoltent tout événement matérialisé sur le réseau. Ces événements sont soit à l'état brut (non encore interprétés), soit qu'ils ont déjà traversé quelques nœuds (routeurs, commutateurs, pare-feu, points d'accès, etc.) avant d'arriver à la sonde. L'avantage avec ce type de sondes est que l'analyse peut être effectuée sur un système dédié sans affecter les performances du réseau surveillé.

Cependant, avec l'augmentation des débits (on parle aujourd'hui de quelques Gigabits par seconde), l'analyse en temps réel devient de plus en plus difficile à cause du grand volume de données récoltées. L'autre complication pour les sondes réseau, est l'utilisation des protocoles cryptographiques, ce qui empêche l'analyse du contenu.

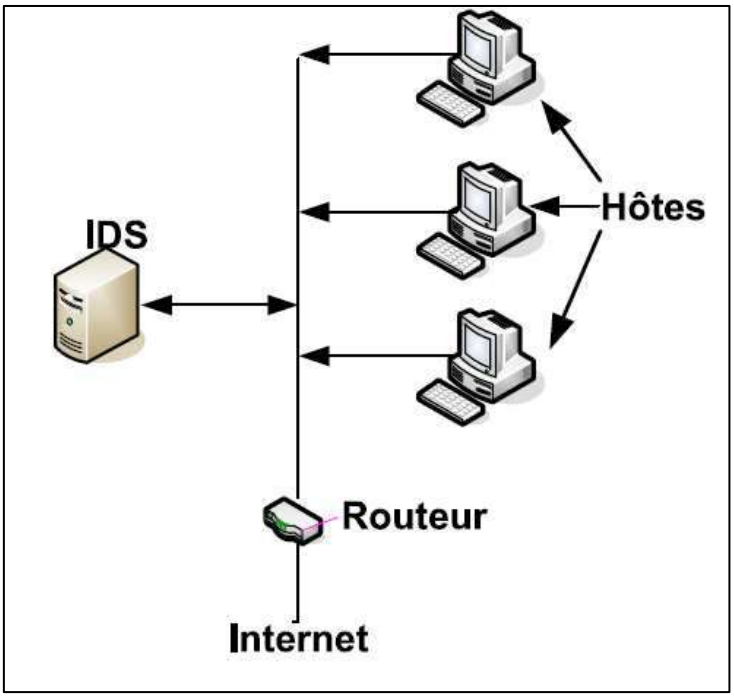

Figure 1.5: Exemple de NIDS [8].

b) Sondes systèmes : [8]

Les sondes système (HIDS : *Host based IDS*) se greffent sur les systèmes surveillés et récoltent seulement les événements matérialisés sur ces systèmes. Ces événements proviennent principalement du noyau et des modules du système d'exploitation.

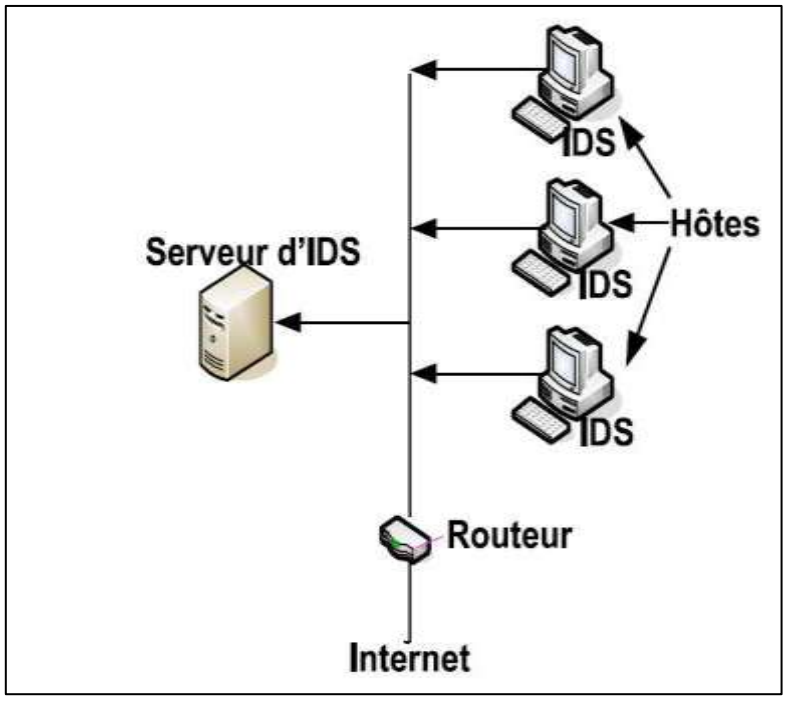

Figure 1.6: Exemple de HIDS [8].

Ces IDS utilisent deux types de sources pour fournir une information sur l'activité : les logs et les traces d'audit du système d'exploitation. Chacun a ses avantages : les traces d'audit sont plus précises, détaillées, et fournissent une meilleure information ; les logs, qui ne fournissent que l'information essentielle, sont plus petits et peuvent être mieux analysés en raison de leurs tailles.

L'inconvénient avec ce type de sondes est que les performances des systèmes surveillés sont affectées (ce qui n'est pas le cas avec des sondes réseau). Cette perte en performances est le prix à payer pour une analyse plus précise et plus fine sur les appels systèmes du noyau. En plus, on n'est plus confronté au problème des données chiffrées auquel font face les sondes réseau puisqu'au niveau du système les événements ne sont pas chiffrés.

- De ces deux familles principales, de nombreuses variantes sont issues :
  - IDS de nœud réseau NNIDS (Network Node IDS).
  - IDS basé sur une application ABIDS (Application-Based IDS).
  - IDS basé sur la pile SBID (Stack-Based IDS).
  - Pot de miel (honeypot).
  - Systèmes “capitonnés” (padded cell systems).

c) IDS hydrides (*NIDS+HIDS*) : [6]

Les systèmes de détection d'intrusions hybrides, rassemblent les caractéristiques de plusieurs systèmes de détection d'intrusions différents. En pratique, on ne retrouve que la combinaison de *NIDS* et *HIDS*. Ils permettent, en un seul outil de surveiller le réseau et l'hôte.

Les sondes sont placées dans des points stratégiques, et agissent comme *NIDS* et/ou *HIDS* suivant leurs emplacements. Toutes les sondes remontent alors les alertes à une machine qui va centraliser, agréger, et lier les informations d'origines multiples.

2.5.3. Réponses des IDS :

Les systèmes de détection d'intrusions émettent des réponses actives qui influent directement la source d'attaque (bloquer le compte d'un utilisateur, suspendre des processus malveillants, arrêter la machine, déconnecter la machine du réseau, etc.), comme ils peuvent se restreindre à des réponses passives qui inscrivent l'événement suspect (emettre un rapport, générer une alarme, créer des fichiers de sauvegarde, etc.) [5]

2.5.4. Paradigme de détection :

La détection d'intrusions s'effectue en analysant l'état courant du système ou en supervisant les transitions des états normaux aux états dangereux. Durant ces deux types d'inspection, l'IDS récupère les informations en interrogeant directement le système ou en écoutant passivement les événements. [5]

2.5.5. La fréquence d'utilisation :

Une autre caractéristique des systèmes de détection d'intrusions est leur fréquence d'utilisation : périodique ou continue. Certains systèmes de détection d'intrusions analysent périodiquement les traces d'audit à la recherche d'une éventuelle intrusion ou anomalie passée. Cela peut être suffisant dans des contextes peu sensibles.

La plupart des systèmes de détection d'intrusions récents effectuent leur analyse des traces d'audit ou des paquets réseau de manière continue afin de proposer une détection en quasi temps réel. Cela est nécessaire dans des contextes sensibles (confidentialité) ou commerciaux (confidentialité, disponibilité). C'est toutefois un processus coûteux en temps de calcul car il faut analyser à la volée tout ce qui se passe sur le système. [14]

3. La classification automatique

La classification est une méthode mathématique d'analyse de données : pour faciliter l'étude d'une population d'effectif important (animaux, plantes, malades, gènes, etc...), on les

regroupe en plusieurs classes de telle sorte que les individus d'une même classe soient le plus semblables possible et que les classes soient le plus distinctes possible. Pour cela, il y a diverses façons de procéder pouvant conduire à des résultats différents.

3.1. Le Clustering: [17]

Le Clustering, aussi connu sous le nom de Segmentation, est un regroupement en classes homogènes consistant à représenter un nuage de points d'un espace quelconque en un ensemble de groupes appelé Cluster.

C'est un traitement sur un ensemble d'objets qui n'ont pas été étiquetés par un superviseur. Généralement lié au domaine de l'analyse des données comme ACP (Analyse linéaire en Composantes Principales), ce type de méthodes vise à répondre au problème de la diminution de la dimension de l'espace d'entrée, ou pour le groupement des objets en plusieurs catégories (clusters) non définies à l'avance.

Un «Cluster» est donc une collection d'objets qui sont «similaires» entre eux et qui sont «dissemblables » par rapport aux objets appartenant à d'autres groupes.

3.2. Domaines d'Application

Les techniques de classification sont utilisées dans des applications dans de nombreux domaines. Par exemple, dans le domaine médical, les classifications de maladies, de traitements curatifs de ces maladies ou de symptômes pour ces maladies peuvent mener à des typologies très utiles. En psychiatrie, un diagnostic correct de groupes de symptômes comme la paranoïa ou la schizophrénie, est indispensable à la réussite de la thérapie. En archéologie, les chercheurs ont tenté d'établir des typologies d'outils en pierre, objets funéraires, et autres, en utilisant des techniques de classification. D'une manière générale, lorsqu'il est nécessaire de répartir une "montagne" d'informations en masses explicables, les classifications sont d'une grande utilité. [16]

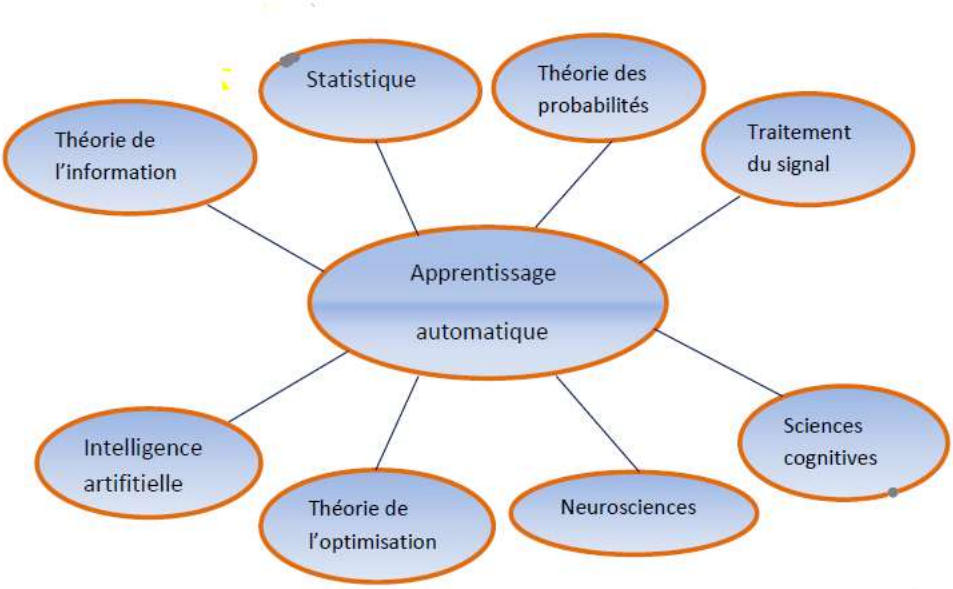

Figure 1.7: Quelques domaines d'apprentissage automatique [17].

La classification, bien que primordiale dans toutes les applications citées précédemment, n'est pas la seule phase à considérer avec attention lors du développement d'un outil d'extraction et d'analyse de l'information [18]. La figure suivante montre le parcours de l'information à classifier.

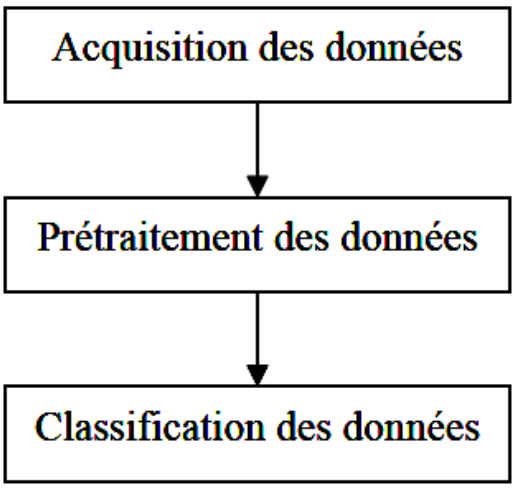

Figure 1.8: Le parcours de l'information à classifier. [14]

- **Acquisition des données :** d'une manière générale, il s'agit à ce niveau, de mettre en place l'ensemble de l'instrumentation (capteurs, matériel d'acquisition,…), de façon à reproduire le phénomène observé le plus fidèlement possible. C'est l'opération de

transformation de l'information à traiter en signaux numériques manipulables par ordinateurs appelée aussi la numérisation de l'information.

- **Prétraitement des données :** cette phase correspond au filtrage des informations en ne conservant que ce qui est pertinent dans le contexte d'étude, c'est la partie segmentation dans la classification d'images.
- **Classification des données :** Elle correspond à l'étape de décision, et pour cela, plusieurs méthodes se présentent pour la résolution.

3.3. Mesures d'éloignement : [14]

Notons $\Omega = \{i=1,\ldots, n\}$ l'ensemble des individus. Cette section se propose de définir sur $\Omega * \Omega$ différentes mesures d'éloignement entre deux individus. Les hypothèses et propriétés étant de plus en plus fortes.

3.3.1. Indice de ressemblance, ou similarité :

C'est une mesure de proximité définie de $\Omega * \Omega$ dans $R+$ et vérifiant :

$$\begin{array}{rcl} s(i,j) & = & s(j,i), \forall (i,j) \in \Omega \times \Omega : \quad \text{symétrie ;} \\ s(i,i) & = & S > 0, \forall i \in \Omega \ : \text{ ressemblance d'un individu avec lui-même ;} \\ s(i,j) & \leq & S, \forall (i,j) \in \Omega \times \Omega : \text{la ressemblance est majorée par } S. \end{array} \qquad (1)$$

Un indice de ressemblance normé s^* est facilement défini à partir de s par :

$$s^*(i,j) = \frac{1}{S} s(i,j), \forall (i,j) \in \Omega \times \Omega ; \qquad (2)$$

s^* est une application de $\Omega * \Omega$ dans [0; 1].

3.3.2. Indice de dissemblance, ou dis similarité :

Une dis similarité est une application d de $\Omega * \Omega$ dans $R+$ vérifiant :

$$\begin{array}{rcl} \forall (i,j) & \in & \Omega \times \Omega \\ d(i,j) & = & d(j,i), : \quad \text{symétrie ;} \\ d(i,j) = 0 & \Leftrightarrow & i = j. \end{array} \qquad (3)$$

Les notions de similarité et dis similarité se correspondent de façon élémentaire. Si s est un indice de ressemblance, alors

$$d(i,j) = S - s(i,j), \forall (i,j) \in \Omega \times \Omega \qquad (4)$$

Est un indice de dissemblance. De façon réciproque, si d est un indice de dissemblance avec

$D = \sup_{(i,j)\in\Omega\times\Omega} d(i,j)$ Alors $s(i,j) = D - d(i,j)$ est un indice de ressemblance. Comme *s**, un indice de dissemblance normé est défini par :

$$d^*(i,j) = \frac{1}{D} d(i,j), \forall (i,j) \in \Omega \times \Omega \qquad (5)$$

Avec *d* = 1 – s* et s* = 1 – d**. Du fait de cette correspondance immédiate, seule la notion de dissemblance, ou dis similarité, normée est considérée par la suite.

3.3.3. Distance :

Une distance sur *Ω* est, par définition, une dis similarité vérifiant en plus la propriété d'*inégalité triangulaire*. Autrement dit, une distance *d* est une application *Ω * Ω* dans *R+* vérifiant :

$$\begin{aligned} d(i,j) &= d(j,i), \ \forall (i,j) \in \Omega \times \Omega ; \\ d(i,i) &= 0 \Longleftrightarrow i = j ; \\ d(i,j) &\leq d(i,k) + d(j,k), \ \forall (i,j,k) \in \Omega^3. \end{aligned} \qquad (6)$$

Si *Ω* est fini, la distance peut être normée.

3.4. Les types des méthodes de classification :

Il a été possible de les regrouper sous la forme d'une hiérarchie de méthodes appelée taxonomie. Nous présentons ci-après une taxonomie dérivée de celle de Jain et Dubes. [18]

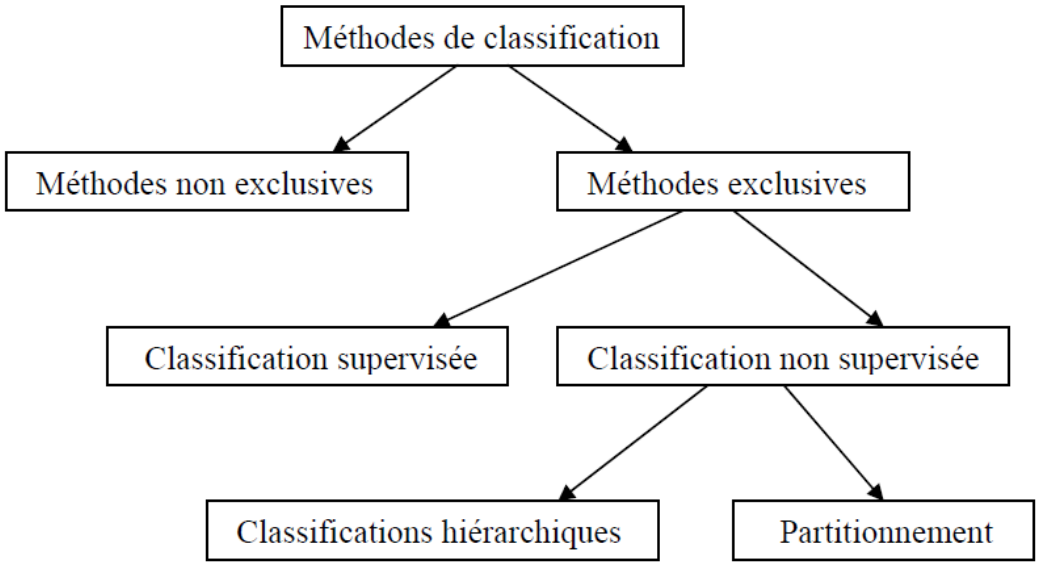

Figure 1.9: Les méthodes de classification. [18]

Classification exclusive/non exclusive : Une classification exclusive est un partitionnement des objets : un objet n'appartient qu'à une classe et une seule. Au contraire, une classification

non exclusive autorise qu'un objet appartienne à plusieurs classes simultanément. Les classes peuvent alors se recouvrir, on parle de la classification floue. [18]

Classification supervisée / non supervisée :

Ces systèmes de classification ne reçoivent aucune information extérieure (provenant d'un opérateur humain par exemple) pour créer des partitions. Cette approche porte en conséquence la dénomination de ***classification non supervisée***. De nombreuses techniques peuvent être envisagées: classification hiérarchique, classification par partition, classification statistique, classification floue (Fuzzy clustering), classification par réseaux de neurones ou classification par algorithmes génétiques.

La deuxième approche nécessite l'intervention d'un système extérieure (qui se compose le plus souvent d'un expert humain) pour étiqueter les informations avant leur classification. Dans ce cas, la classification va apprendre à trouver les groupes de données en fonction des étiquetages précédents et va donc s'entraîner à reconnaître les données. L'objectif de cette démarche est ensuite de pouvoir fournir au système entraîné des données non étiquetées pour qu'il puisse continuer à les répartir dans les groupes adéquats sans autre intervention humaine. On parle alors d'une approche ***d'apprentissage supervisé.*** On peut citer comme méthodes de classification supervisées toutes les méthodes basées sur un ensemble d'apprentissage tel que la plupart des méthodes du Data Mining (Arbre de décision, Raisonnement à base de cas, Réseaux de neurones…). [14]

Classification hiérarchique/partitionnement : Une méthode de classification hiérarchique construit une séquence de partitions imbriqués, que l'on visualise par exemple par un dendrogramme (figure 1.13), alors qu'un *partitionnement* ne construit qu'une seule partition des données. [18]

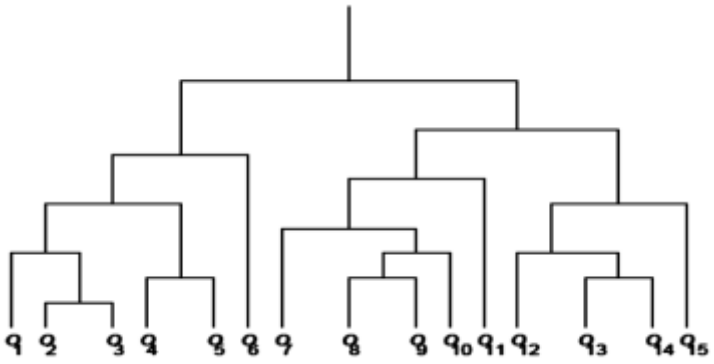

Figure 1.10 : Exemple d'un dendrogramme. [14]

3.5. Les algorithmes de clustering :

On se limitera ici aux deux techniques de classification non supervisée, et dans chaque type on va présenter seulement un algorithme :

- ✓ Le clustering par partition : les centres mobiles (*K-Means*).

- ✓ Le clustering hiérarchique : la classification ascendante hiérarchique (*CAH*).

3.5.1. K-Means [17]

L'algorithme k-means, un des plus simples algorithmes d'apprentissage non supervisé, appelée algorithme des centres mobiles, il attribue chaque point dans un cluster dont le centre (centroïde) est le plus proche. Le centre est la moyenne de tous les points dans le cluster, ses coordonnées sont la moyenne arithmétique pour chaque dimension séparément de tous les points dans le cluster c'est-à-dire chaque cluster est représentée par son centre de gravité.

- ➢ **Principe**

L'idée principale est de définir les k centroïdes arbitraires c1, c2,..., ck (k le nombre de clusters fixé a priori, chaque ci représente le centre d'une classe), Ces centroïdes doivent être placés dans des emplacements différents. Donc, le meilleur choix est de les placer le plus possible éloignés les uns des autres. La prochaine étape est de prendre chaque point appartenant à l'ensemble de données et l'associer au plus proche centroïde. C'est à dire Chaque classe Si sera représentée par un ensemble d'individus les plus proches de son ci, Les nuées dynamiques sont une généralisation de ce principe, où chaque cluster est représenté par un noyau mais plus complexe qu'une moyenne.

Lorsqu'aucun point n'est en attente, la première étape est terminée et un groupage précoce est fait. À ce point nous avons besoin de recalculer les k nouveaux centroïdes mi des groupes issus de l'étape précédente qui vont remplacer les ci (mj est le centre de gravité de la classe Sj, calculé en utilisant les nouvelles classes obtenues). Après, on réitère Le processus jusqu'à atteindre un état de stabilité où aucune amélioration n'est possible, nous pouvons constater que les k centroïdes changent leur localisation par étape jusqu'à plus de changements sont effectués.

En d'autres termes les centroïdes ne bougent plus.

- **Discussion**

Cette méthode est la plus populaire des méthodes de clustering, malgré cela, un de ses problèmes majeurs est qu'il tend à trouver des classes sphériques de même taille. En plus K-means est connu par sa complexité de « NP-difficile ». Cette approche est convergente et surtout avantageuse du point de vue calcul mais elle dépend essentiellement de la partition initiale (Des initialisations différentes peuvent mener à des clusters différents «problèmes de minima locaux ») cela risque d'obtenir une partition qui ne soit pas optimale pourtant qu'elle donne surement une partition meilleure que la partition initiale. De plus, la définition de la classe se fait à partir de son centre, qui pourrait ne pas être un individu de l'ensemble à classer, d'où le risque d'obtenir des classes vides.

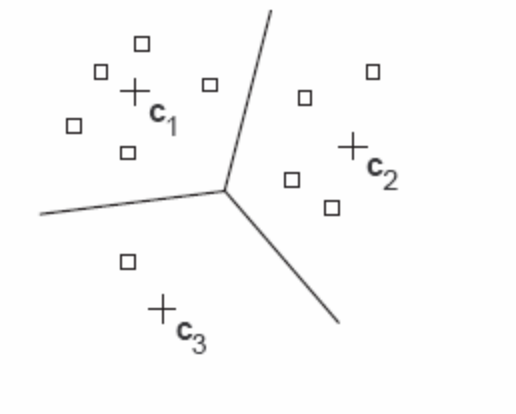

Figure 1.11 : Exemple de partition obtenue par les centres mobiles. [5]

- **Algorithme :** [18]

(1) **Tantque** l'inertie intraclasse ne s'est pas stabilisée **faire**

(2) Générer une nouvelle partition *P'* en affectant chaque objet à la classe dont le centre est le plus proche.

(3) Calculer les centres de gravité des classes de la nouvelle partition *P'*.

(4) $P \leftarrow P'$.

(5) **Fin Tantque**

(6) **retourner***P*.

3.5.2. Classification Ascendante Hiérarchique (CAH)

L'objectif de cet algorithme est de rassembler des objets dans des classes de plus en plus larges, en utilisant certaines mesures de similarité ou de distance. Les résultats de ce type de

classification sont habituellement représentés sous la forme d'un dendrogramme (arbre de la classification hiérarchique).

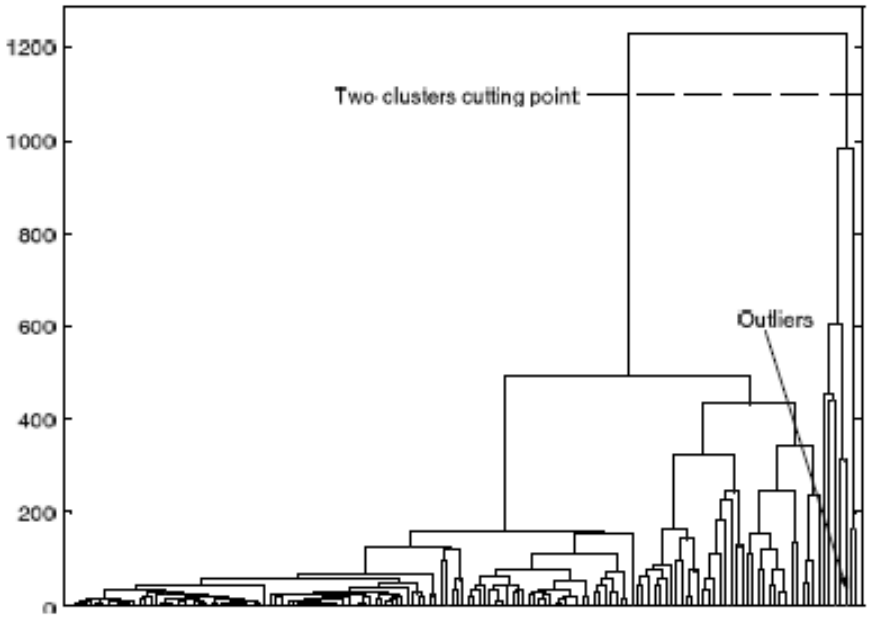

Figure 1.12 : Classification Ascendante Hiérarchique [4]

Ce type de clustering consiste à effectuer une suite de regroupements en Clusters de moins en moins fines en agrégeant à chaque étape les objets (simple élément) ou les groupes d'objets les plus proches. Ce qui nous donne une arborescence de clusters [14].

La méthode CAH suppose qu'on dispose d'une mesure de dissimilarité entre les individus, dans le cas de points situés dans un espace euclidien, on peut utiliser la *distance* comme mesure de dis similarité.

La classification ascendante hiérarchique est dite ascendante car elle part d'une situation où tous les individus sont seuls dans une classe, puis sont rassemblés en classes de plus en plus grandes. [14]

- **Discussion : [14]**

La CAH ne nécessite pas de connaître le nombre de clusters a priori. De plus, il n'y a pas de fonction d'initialisation, ainsi une seule construction d'un cluster (équivalent à une itération pour les méthodes de partitionnement). En ce qui concerne généralement les méthodes hiérarchiques le problème qu'on peut rencontrer réside dans la sélection d'une ultra-métrique (distance pour calculer la similarité entre clusters) soit la plus proche de la métrique utilisée pour les individus, car ces méthodes sont heuristiques, pour cela y a

plusieurs techniques permet de le faire : Saut minimal (single linkage), Saut maximal (complète linkage), Saut moyen, Barycentre…etc ,une autre faiblesse est : la complexité de temps d'au moins O(n2), où *n* est le nombre d'objets au total, ainsi qu'on pourrait jamais défaire ce qui a été fait précédemment. Il est difficile parfois d'apporter une justification aux méthodes hiérarchique (CAH, CDH..), Cependant, dans une interprétation probabiliste de la CAH, basée sur une estimation par maximum de vraisemblance des modèles de mélange, est proposée comme solution pour mieux interpréter les résultats.

Un autre inconvénient de ce type de méthodes est qu'une action effectuée (fusion ou décomposition), elle ne peut être annulée. Cela permet de réduire le champ d'exploration, mais une telle astuce ne peut corriger une décision erronée.

Afin améliorer la qualité d'une classification hiérarchique, on peut profiter de deux techniques :

- Analyser attentivement les liens entre objets à chaque étape.
- Améliorer la partition obtenue avec une méthode de deuxième type de clustering (partitionnement)**.**

Algorithme : [14]

(1) **Initialisation** :

Chaque individu est placé dans son propre cluster, Calcul de la matrice de ressemblance *M* entre chaque couple de clusters.

(2) **Répéter** :

- Sélection dans *M* des deux clusters les plus proches *Ci* et *Cj*.
- Fusion de *Ci* et *Cj* par un cluster *CG* plus général.
- Mise à jour de *M* en calculant la ressemblance entre *CG* et les clusters existants Jusqu'à fusionner les 2 derniers clusters.

a) Mesure des Distances : [16]

La Classification Ascendante Hiérarchique (CAH) utilise des mesures de dissemblance ou de distance entre les objets pour former des classes. Ces distances peuvent être basées sur une ou plusieurs dimensions. La méthode la plus directe pour calculer des distances entre objets dans un espace multidimensionnel consiste à calculer les distances Euclidiennes. Si nous avons un espace à deux ou trois dimensions, cette mesure est celle des distances géométriques normales entre les objets dans l'espace. L'algorithme de la CAH ne tient pas compte de la

nature des distances qui lui sont indiquées ; peu importe donc qu'il s'agisse de véritables distances, ou d'autres mesures dérivées qui peuvent être plus explicites pour un chercheur. Ce sera au chercheur de choisir la méthode adaptée à son application spécifique. Le module Classifications permet de calculer de nombreux types de mesures de distances, mais l'utilisateur peut aussi calculer une matrice de distances par ses propres moyens afin de l'utiliser directement dans la procédure.

Distance Euclidienne. C'est probablement le type de distance le plus couramment utilisé. Il s'agit simplement d'une distance géométrique dans un espace multidimensionnel. Elle se calcule ainsi :

$$\text{distance}(x, y) = \{S_i (x_i - y_i)^2 \}^{½} \qquad (7)$$

Notez que les distances Euclidiennes (et Euclidiennes au carré) sont calculées à partir des données brutes, et non des données centrées-réduites. C'est la méthode de calcul qui est habituellement utilisée, et elle présente certains avantages (en particulier, la distance entre deux objets quelconques n'est pas affectée par l'introduction de nouveaux objets dans l'analyse, qui peuvent être des points atypiques). Toutefois, les distances peuvent être largement affectées par les différences d'unités de mesure des dimensions pour lesquelles ces distances sont calculées, les distances Euclidiennes ou distances Euclidiennes au carré résultantes (calculées sur de multiples dimensions) pourront s'en trouver largement affectées, et par conséquent, les résultats de la classification pourront être très différents. Naturellement, vous avez la possibilité d'effectuer tout type de standardisation ou de changement d'échelle en utilisant les fonctionnalités de gestion des données deSTATISTICA.

Distance Euclidienne au carré. Vous pouvez élever la distance Euclidienne standard au carré afin de "sur-pondérer" les objets atypiques (éloignés). :

$$\text{Distance}(x, y) = S_i (x_i - y_i)^2 \qquad (8)$$

Distance du City-block (Manhattan). Cette distance est simplement la somme des différences entre les dimensions. Dans la plupart des cas, cette mesure de distance produit des résultats proches de ceux obtenus par la distance euclidienne simple. En revanche, notez

qu'avec cette mesure, l'effet des différences simples importantes (points atypiques) est atténué (puisque ces distances ne sont pas élevées au carré). Cette distance se calcule ainsi :

$$\text{Distance } (x, y) = S_i |x_i - y_i| \qquad (9)$$

Distance de Tchebychev. Cette mesure de distance est adaptée lorsque nous considérons deux objets comme étant "différents" à partir du moment où ils sont différents sur l'une des dimensions. La distance de Tchebychev se calcule ainsi :

$$\text{Distance } (x, y) = \text{Maximum } |x_i - y_i| \qquad (10)$$

Distance à la puissance. Nous pouvons parfois souhaiter augmenter ou diminuer la pondération progressive associée à des dimensions pour lesquelles les objets respectifs sont très différents. Cette opération est rendue possible par la distance à la puissance. La distance à la puissance se calcule ainsi :

$$\text{Distance } (x, y) = (S_i |x_i - y_i|^{p})^{1/r} \qquad (11)$$

Où r et p sont des paramètres définis par l'utilisateur. Vous verrez rapidement comment fonctionne cette mesure à l'aide d'exemples. Le paramètre p contrôle la pondération progressive affectée aux différences entre les dimensions individuelles, tandis que le paramètre r contrôle la pondération progressive affectée aux grandes différences entre les objets. Si r et p sont égaux à 2, cette distance équivaut à la distance euclidienne.

Percent disagreement. Cette mesure est particulièrement utile si les données des dimensions utilisées dans l'analyse sont de nature catégorielle. Cette distance se calcule ainsi :

$$\text{distance}(x, y) = (\text{Nombre de } x_i \neq y_i)/i \qquad (12)$$

b) Règles d'Agrégation : [16]

À la première étape, lorsque chaque objet représente sa propre classe, les distances entre ces objets sont définies par la mesure de distance choisie. En revanche, une fois l'agrégation de plusieurs objets, nous avons besoin donc d'une règle d'agrégation pour déterminer le moment où deux classes seront suffisamment similaires pour n'en former qu'une seule.

Saut minimum ou "single linkage" (distance minimum) : la distance entre deux classes est ici déterminée par la distance entre les deux objets les plus proches (les plus proches voisins) dans les différentes classes. Cette règle provoque des chaînes d'objets assemblés en classes, et les résultats obtenus ressemblent à de longues chaînes.

Diamètre ou "complete linkage" (distance maximum). Dans cette méthode, les distances entre classes sont déterminées par la plus grande distance existant entre deux objets de classes différentes (c'est-à-dire les "voisins les plus éloignés"). Cette méthode donne souvent de bons résultats lorsque les objets forment déjà naturellement des "groupes" bien distincts. Si les classes ont plutôt une forme allongée, ou sont en forme de "chaîne", cette méthode sera mal adaptée.

Méthode de Ward (méthode du moment d'ordre 2). Cette méthode se distingue de toutes les autres en ce sens qu'elle utilise une analyse de la variance approchée afin d'évaluer les distances entre classes. En résumé, cette méthode tente de minimiser la Somme des Carrés (SC) de tous les couples (hypothétiques) de classes pouvant être formés à chaque étape. Référez-vous à Ward (1963) pour les détails concernant cette méthode. En général, cette méthode est très efficace, même si elle tend à créer des classes de petite taille.

3.6. La Classification (Clustering) et la détection d'intrusion : [1] [2]

Les techniques de classification et de clustering dans Data mining sont utiles pour une grande variété d'applications en temps réel traitant de grandes quantités de données. Dans la sécurité de l'information, la détection d'intrusion est le fait de détecter les actions qui tentent de compromettre la confidentialité, l'intégrité ou la disponibilité d'une ressource. Les systèmes de détection d'intrusion sont des systèmes logiciels pour identifier les écarts par rapport au comportement normal et l'utilisation du système. Ils détectent les attaques en utilisant les techniques de classification du data mining et les algorithmes de clustering.

Les approches du data mining peuvent être appliquées à la fois à l'approche comportementale et par scénario. L'échantillon de données est un ensemble de propriétés du système, ce qui représente le comportement du système/utilisateur. Les techniques de classification sont utilisées pour apprendre un modèle à l'aide de la formation d'un ensemble d'échantillons de données. Le modèle est utilisé pour classer les échantillons de données comme instance de comportement anormal ou instance de comportement normal. Tout

échantillon de données ayant des caractéristiques différentes des clusters formés est considéré comme étant une instance d'un comportement anormal. Ces techniques de classification peuvent détecter de nouvelles attaques.

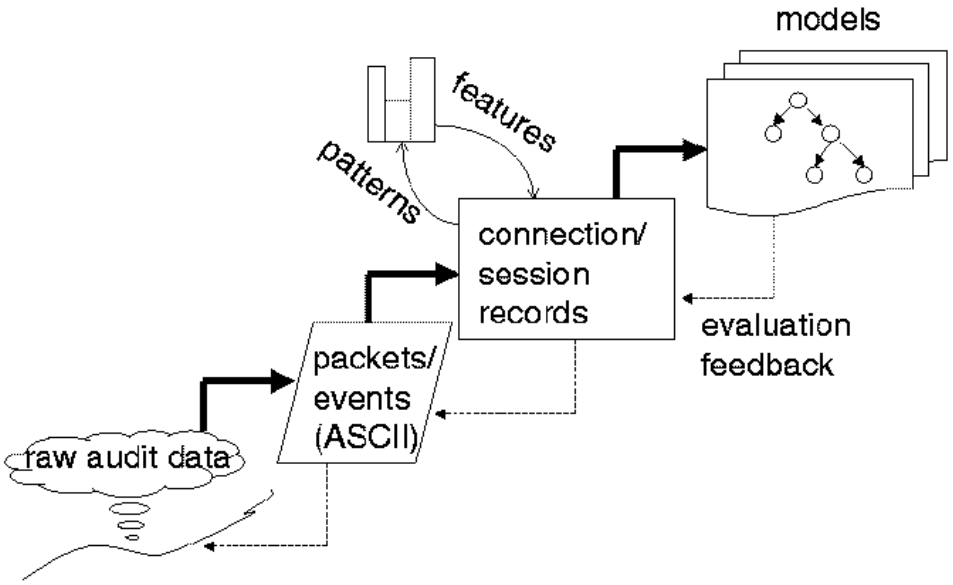

Figure 1.13 : Le processus du «data mining» de la construction de modèles de détection d'intrusion. [3]

3.7. Les limites de Clustering : [17]

Il y a un certain nombre de problèmes avec le clustering. Parmi eux :

- Les techniques de clustering actuelles ne traitent pas tous les besoins de façon adéquate (et simultanément), comme le fait que si nous n'avons pas des variables continues (la longueur), mais les catégories nominales, comme les jours de la semaine. Dans ces cas encore, la connaissance du domaine doit être faite pour formuler le clustering appropriée.
- Traitement d'un grand nombre de dimensions et de grand nombre de données, question peut être problématique en raison de la complexité du temps de calcule.
- L'efficacité de la méthode dépend de la définition de «distance» utilisée.
- Si la mesure de la distance n'existe pas, nous devons la «définir», ce qui n'est pas toujours facile, surtout dans des espaces multidimensionnels.
- Le résultat de l'algorithme de clustering peut être interprété de différentes manières.
- Beaucoup d'algorithmes de clustering exigent la spécification du nombre de clusters à produire en entrée de l'ensemble de données, avant l'exécution de l'algorithme. i.e. :

connaissance de la valeur correcte à l'avance, la valeur appropriée doit être déterminée, un problème pour lequel un certain nombre de techniques ont été développées.

Conclusion :

Ce chapitre est divisé en trois parties sont généralités sur la sécurité informatique, Système de détection d'intrusion et la classification automatique. Dans la première partie nous avons parlé sur les notions fondamentales relatives à la sécurité informatique.

Dans la deuxième partie, nous avons présenté les IDS avec leurs architectures, fonctionnement et classifications, selon cette dernière on peut distinguer deux méthodes de détection d'intrusion des IDS qu'ils sont l'approche par signature et l'approche comportementale.

Dans la dernière partie du chapitre nous avons présenté les méthodes de classification, ces méthodes (partitionnements, hiérarchiques) opèrent avec des algorithmes de classifications à savoir respectivement le k-Means et CAH.

D'après cette étude bibliographique, nous avons conclu le besoin urgent de l'utilisation de la classification pour la détection d'intrusion.

Le chapitre suivant sera consacré à la rédaction de l'état de l'art.